APPENDIX A

COMPREHENSIVE COMPILATION OF

MQMS PROGRAMS

PROGRAMS — M1

D#MSQ.@MSQ00M1.PGMLIB99(URBANF)

```
//TI36URBA JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=C
//URBANF EXEC SAS6,OPTIONS='ERRORS=0 NONEWS'
//IN1 DD DSN=D#MSQ.@MSQ00.FLAT.TI36LIB1.URBANF,
//        DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.URBANF.URBANMSA,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(TRK,(10,3),RLSE)
//SYSIN DD *
*******************************************************************
* PROJ      :MSQ
* REQ BY    :TEAM MSQ
* REQ DATE  :04/21/01
* EXEC DATE :06/06/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(URBANF)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH
* INPUTS    :D#MSQ.@MSQ00.FLAT.TI36LIB1.URBANF - ON DISK
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.URBANF.URBANMSA - ON DISK
*            (URBANF) - JOB OUTPUT FILE
* PURPOSE   :This is a standalone program that creates a control
*            dataset used in a later program (YXXM1002) to develop
*            a dichotomous urban/rural variable for a beneficiary's
*            place of residence. The following assignment criteria
*            for the urban/rural was used:  (1) if the MSA code
*            exists, then that county's status is urban;
*            (2) if it doesn't exist, then the status is rural;
*            and (3) non-matches are assigned to rural status.
*            The 1999 MSA code for a given SSA state and county
*            was selected and will be used for all years.
*            This program reads in the MSABEA file off the CMS
*            web site and creates a SAS file to be used as a
*            control dataset for an urban/rural informat.
*******************************************************************;


         /**CREATE A SAS FILE THAT WE WILL USE AS A**/
         /**CONTROL DATASET FOR AN URBAN/RURAL INFORMAT**/

 ******************************************************************;
 * STEP 1. Reads in the MSABEA flat file and transforms
 *         it into a SAS control dataset.
 ******************************************************************;

DATA OUT1.URBANMSA
      (DROP=TEMP LABEL="MSABEA 99 URBAN INFORMAT DATASET");
LENGTH FMTNAME $6 TYPE $1 HLO $1 LABEL $1;
RETAIN FMTNAME 'MSABEA' TYPE 'J' DEFAULT 5;
INFILE IN1 MISSOVER END=FINI;
INPUT       @30     START       $CHAR5.
            @128    TEMP        $CHAR4.
;
LABEL=PUT(TEMP NE ' ',1.);
OUTPUT;

 ******************************************************************;
 * STEP 2. Anticipates that there will be non-matches
```

```
 *         during the merge of the control dataset to
 *         the Denominator File by allowing for
 *         non-matches to be given the value of "X".
```
D#MSQ.@MSQ00M1.PGMLIB99(URBANF) -- CONTINUED

```
 ********************************************************************;

IF FINI THEN DO;
    START='OTHER';
    HLO='O';
    LABEL='X';
    OUTPUT;
END;
RUN;

 ********************************************************************;
 * STEP 3. Proc contents of the 1999 Urban MSABEA Control Dataset
 *         for Urban Informat and a data dump of 30 observations
 ********************************************************************;

PROC CONTENTS DATA=OUT1.URBANMSA DIRECTORY;
TITLE "1999 URBAN MSABEA CONTROL DATASET FOR URBAN INFORMAT";
RUN;

PROC PRINT DATA=OUT1.URBANMSA(OBS=30);
TITLE2 "PRINT 30 OBSERVATIONS";
RUN;

 ********************************************************************;
 * STEP 4. Prints out the format as an actual control dataset
 *         for the urban/rural character informat.
 ********************************************************************;

TITLE2;
PROC FORMAT CNTLIN=OUT1.URBANMSA FMTLIB;
RUN;
```

D#MSQ.@MSQ00M1.PGMLIB92(M1PROD92)

```
//TI36M192 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD92 EXEC M1PROD,YEAR=92
```

D#MSQ.@MSQ00M1.PGMLIB93(M1PROD93)

```
//TI36M193 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD93 EXEC M1PROD,YEAR=93
```

D#MSQ.@MSQ00M1.PGMLIB94(M1PROD94)

```
//TI36M194 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD94 EXEC M1PROD,YEAR=94
```

D#MSQ.@MSQ00M1.PGMLIB95(M1PROD95)

```
//TI36M195 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD95 EXEC M1PROD,YEAR=95
```

D#MSQ.@MSQ00M1.PGMLIB96(M1PROD96)

```
//TI36M196 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD96 EXEC M1PROD,YEAR=96
```

D#MSQ.@MSQ00M1.PGMLIB97(M1PROD97)

```
//TI36M197 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD97 EXEC M1PROD,YEAR=97
```

D#MSQ.@MSQ00M1.PGMLIB98(M1PROD98)

```
//TI36M198 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD98 EXEC M1PROD,YEAR=98
```

D#MSQ.@MSQ00M1.PGMLIB99(M1PROD99)

```
//TI36M100 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD99 EXEC M1PROD,YEAR=99
```

D#MSQ.@MSQ00M1.PGMLIB00(M1PROD00)

```
//TI36M100 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD00 EXEC M1PROD,YEAR=00
```

D#MSQ.@MSQ00M1.PGMLIB01(M1PROD01)

```
//TI36M101 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=K
//* Purpose:  This job executes the cataloged procedure
//*           M1PROD for a particular year (e.g., M1PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//  JCLLIB ORDER=(D#MSQ.@MSQ00M1.PGMLIB99)
//M1PROD01 EXEC M1PROD,YEAR=01
```

D#MSQ.@MSQ00M1.PGMLIB99(M1PROD)

```
//M1PROD PROC
//*************************************************************
//********CATALOGED PROCEDURE FOR M1*****************************
//********GENERATE M1 FOR EACH YEAR IN A FEW STEPS**************
//*JUST ENTER THE SYMBOLIC PARAMETER YEAR=(2-DIGIT YEAR)
//*************************************************************
//*
//*****************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* REQ DATE  :07/10/2001
//* EXEC DATE :07/11/2001
//* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(M1PROD)
//* PGMR      :MATTHEW URATO
//* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
//* INPUTS    :SEE THE SYSIN DD FOR EACH STEP
//* OUTPUTS   :SEE THE SYSIN DD FOR EACH STEP
//* PURPOSE   :SEE ABOVE
//*************************************************************
//*CREATE AN M1 SAS FILE FROM DENOMINATOR FILE: READ ALL RECORDS
//*************************************************************
//NULLSTEP EXEC PGM=IEFBR14
// IF (&YEAR NE 00) THEN
//YXXM1001 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR')
//WORK DD SPACE=(CYL,(800,200)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M1.PGMLIB99,DISP=SHR
//IN1 DD DSN=TI03.@MSQ0000.D#MSQ.DENOM&YEAR..#010301,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1001.DENOMA&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),
//        VOL=(,,,8)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM101),DISP=SHR
//*************************************************************
// ELSE
//YXXM1001 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR')
//WORK DD SPACE=(CYL,(800,200)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M1.PGMLIB99,DISP=SHR
//IN1 DD DSN=TI03.@MSQ0000.MSQ2000.DENOM00.#100501,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1001.DENOMA&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),
//        VOL=(,,,8)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM101),DISP=SHR
// ENDIF
//*************************************************************
//*CREATE AN M1 BASE ANALYTIC DENOMINATOR FILE
//*************************************************************
//YXXM1002 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(500,100)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=*.YXXM1001.SAS.IN0,DISP=SHR
//IN1 DD DSN=*.YXXM1001.SAS.OUT1,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.URBANF.URBANMSA,DISP=SHR
//IN3 DD DSN=D#MSQ.@MSQ0000.SAS.FIPSMSA1.FIPSMSA,DISP=SHR
```

```
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENOMB&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),
```


```
//        VOL=(,,,8)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENEXB&YEAR,
//        DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=27648,LRECL=27648),
//        SPACE=(CYL,(200,100),RLSE)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENRUR&YEAR,
//        DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=27648,LRECL=27648),
//        SPACE=(CYL,(50,50),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM102),DISP=SHR
//*************************************************************
//*CREATE AN M1 COUNT FILE FOR MAJOR GROUPS
//*************************************************************
//YXXM1003 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(TRK,(500,500)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=*.YXXM1002.SAS.IN0,DISP=SHR
//IN1 DD DSN=*.YXXM1002.SAS.OUT1,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(TRK,(5,5),RLSE),DCB=(BLKSIZE=27648,LRECL=27648)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM103),DISP=SHR
//*************************************************************
//*CREATE AN M1 AGE/AGE-SEX ADJUSTMENT FILE
//*************************************************************
// IF (&YEAR = 99) THEN
//YXXM1004 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(TRK,(200,100)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M1.PGMLIB99,DISP=SHR
//IN1 DD DSN=*.YXXM1003.SAS.IN1,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1004.AGESEX&YEAR,
//        DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=27648,LRECL=27648),
//        SPACE=(TRK,(9,9),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM104),DISP=SHR
// ENDIF
//*************************************************************
//*CREATE AN M1 MAINFRAME REPORT OF COHORT COUNTS
//*BY MAJOR GROUPS
//*************************************************************
//YXXM1005 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DSN=D#MSQ.@MSQ00M1.PGMLIB99,DISP=SHR
//IN1 DD DSN=*.YXXM1003.SAS.OUT1,DISP=SHR
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM105),DISP=SHR
//*************************************************************
//*CREATE AN M1 AGE/AGE-SEX COUNT FILE
//*************************************************************
//YXXM1006 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK  DD SPACE=(CYL,(500,100)),DCB=(BLKSIZE=27648,LRECL=27648)
//WORK0 DD SPACE=(CYL,(550,50)),DCB=(BLKSIZE=27648,LRECL=27648)
//WORK1 DD SPACE=(CYL,(550,50)),DCB=(BLKSIZE=27648,LRECL=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M1.PGMLIB99,DISP=SHR
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENOMB&YEAR,DISP=SHR
```

```
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1006.M1CNTS&YEAR,
//        DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=27648,LRECL=27648),
//        SPACE=(TRK,(30,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M1.PGMLIB99(SYSM106),DISP=SHR
//*********END OF PROCEDURE
//************************************************************
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM101)

```
************************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/06/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(SYSM101)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001) - MACRO
*           :TI03.@MSQ0000.D#MSQ.DENOMXX.#010301
*                                - Y92-Y99 DENOM FILE
*         OR:TI03.@MSQ0000.MSQ2000.DENOM00.#10051
*                                - Y00 DENOM FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM1001.DENOMAXX
* PURPOSE   :FROM FLAT DENOMINATOR FILE, CREATE A SAS FILE
*            THAT INCLUDES ALL VARIABLES.  SEE MACRO YXXM1001
*            FOR MORE DETAILS
************************************************************************;
OPTIONS BUFNO=5;


        /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1001)/SOURCE2;

       /**MACRO CALL**/

%YXXM1001(YY=&SYSPARM)
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001)

```
%MACRO YXXM1001(YY=99,IN1=IN1,OUT1=OUT1);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM1001
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 APRIL 30, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      READ IN THE RAW DATA - ALL VARS - FROM 100 PERCENT
      DENOMINATOR FILE INTO SAS FOR EACH YEAR AND OUTPUT FILE TO TAPE.
      ALSO, PROVIDE BASIC FREQS OF SOME IMPORTANT VARIABLES.

8  PARAMETERS:
      THERE ARE 3 KEYWORD PARAMETERS: YY IN1 OUT1

YY
IS THE 2-DIGIT YEAR OF THE DENOMINATOR FILE BEING READ INTO SAS
E.G.:
YY=99        (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR THE FLAT INPUT DENOMINATOR FILE
CREATED FROM 100% DSAF PULL.  FILE IS ABOUT 4GB (3 TAPES).
FILE LOCATION: TI03.@MSQ0000.D#MSQ.DENOMXX.#010301 (XX=YEAR
FOR Y92-Y99) OR TI03.@MSQ0000.MSQ2000.DENOM00.#10051FOR Y00.
E.G.: IN1=IN1   (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE DENOM SAS FILE
BEING WRITTEN TO TAPE.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1001.DENOMAXX (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.  SIZE IS ABOUT 4GB (3 TAPES).
FILE IS SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

9  OTHER MACRO VARIABLES OF NOTE:
'DUPS' IS THE NAME OF A MACRO VARIABLE THAT CONTAINS THE
OBSERVATION NUMBERS OF THOSE IDS IN THE DENOMINATOR FILE THAT
HAVE AT LEAST ONE EXACT DUPLICATE IN THE FILE. WE DO NOT EXPECT
MANY OF THESE.

10  OTHER NOTES:
VARIABLE 'YXXDOBS' (XX=YEAR) IS CREATED AS UNIQUE ASCENDING NUMBER
FOR EACH RECORD. ALL NUMERIC VARS,EXCEPT 'YXXDOBS' (XX=YEAR),
SAVED WITH LENGTH=3.

FOR EACH VARIABLE THAT WE READ INTO SAS FROM THE FLAT DENOM FILE,
ITS SAS LABEL CONTAINS ITS NUMBER AS GIVEN ON CMS FILE
SPECIFICATIONS FOR THE DENOMINATOR FILE.

THE SAS WORK LIBRARY SHOULD BE ABLE TO HOLD AND MANIPULATE
```

```
AN APPROXIMATELY 500MB TEMPORARY SAS FILE  AND TO PERFORM
SOME BASIC FREQS ON THE MAIN OUTPUT FILE.
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001) -- CONTINUED

```
11 EXAMPLE MACRO CALL:
%YXXM1001(YY=92,IN1=INFILE,OUT1=OUTFILE)

*********************************************************************
*********************************************************************;

                /**CODE BEGINS**/

 *********************************************************************;
 * STEP 1. Create a temp file with the variable HICNO.
 *         The temp file holds only
 *         the variables needed for the identification of
 *         duplicates rather than sorting through the
 *         whole Denominator File.  Sort the temp file by HICNO.
 *********************************************************************;

DATA TEMP;
INFILE &IN1 MISSOVER LS=11;
INPUT @1 HICNO   $CHAR11.;
RUN;

PROC SORT DATA=TEMP;
BY HICNO;
RUN;

 *********************************************************************;
 * STEP 2. Identify duplicates and provide them in a proc print
 *********************************************************************;

     /**PICK OFF ANY DUPLICATE IDS**/

DATA TEMP2;
SET TEMP;
BY HICNO;
IF NOT(FIRST.HICNO AND LAST.HICNO) THEN DO;
   IF FIRST.HICNO THEN OUTPUT;
END;
RUN;

PROC PRINT DATA=TEMP2;
TITLE "Y&YY DENOM IDS THAT HAVE DUPLICATES";
RUN;

 *********************************************************************;
 * STEP 3. Save the observation number of the exact duplicates
 *         in order to be able to identify them on the whole
 *         Denominator File.
 *********************************************************************;

%LET FILEO = %SYSFUNC(OPEN(TEMP2));
%LET FILEN = %SYSFUNC(ATTRN(&FILEO,ANY));

%IF &FILEN EQ 1 %THEN %DO;
```

```
PROC SQL NOPRINT;
SELECT QUOTE(HICNO) FORMAT=$13.
  INTO :DUPS SEPARATED BY ','
    FROM TEMP2;
QUIT;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001) -- CONTINUED

```
%PUT THE VALUES FOR MACRO VARIABLE 'DUPS' ARE AS FOLLOWS:
&DUPS;
%END;

 ********************************************************************;
 * STEP 4. Read in the Denominator File and assigns
 *         records a unique observation number (YYYDOBS).
 ********************************************************************;

DATA   &OUT1..DENOMA&YY
       (LABEL="DENOMA&YY FILE:ALL VARS"
        SORTEDBY=Y&YY.DOBS);
LENGTH Y&YY.AGE Y&YY.AMO Y&YY.BMO Y&YY.HMOM Y&YY.BUYM Y&YY.RFRN 3;
INFILE &IN1 MISSOVER;

INPUT
@1  Y&YY.HICNO                      $CHAR11.
@12 Y&YY.BIC                        $CHAR2.
@14 Y&YY.STATE                      $CHAR2.
@16 Y&YY.CNTY                       $CHAR3.
@19 Y&YY.ZIP                        $CHAR9.
@28 Y&YY.DOB                        YYMMDD8.
@36 Y&YY.SEX                        $CHAR1.
@37 Y&YY.RACE                       $CHAR1.
@38 Y&YY.AGE                        2.
@40 Y&YY.OREC                       $CHAR1.
@41 Y&YY.CREC                       $CHAR1.
@42 Y&YY.ESRD                       $CHAR1.
@43 Y&YY.MSCD                       $CHAR2.
@45 Y&YY.ATRM                       $CHAR1.
@46 Y&YY.BTRM                       $CHAR1.
@47 Y&YY.F00                        $CHAR1.
@48 (Y&YY.BUY1-Y&YY.BUY12)          ($CHAR1.)
@60 (Y&YY.HMO1-Y&YY.HMO12)          ($CHAR1.)
@72 Y&YY.AMO                        2.
@74 Y&YY.BMO                        2.
@76 Y&YY.HMOM                       2.
@78 Y&YY.BUYM                       2.
@80 Y&YY.VDOD                       $CHAR1.
@81 Y&YY.DETH                       YYMMDD8.
@89 Y&YY.RFRN                       2.
;
Y&YY.DOBS=_N_;

 ********************************************************************;
 * STEP 5. If exact duplicates exist, exclude those
 *         observations with hicnos identified
 *         as exact duplicates.
 ********************************************************************;
```

```
%IF &FILEN EQ 1 %THEN %DO;
IF Y&YY.HICNO IN(&DUPS) THEN DELETE;
%END;

 ********************************************************************;
 * STEP 6. Create labels for the variables on the
 *         Denominator File and provides a proc contents
 *         and data dump of 20 observations for the new
```
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001) -- CONTINUED
```
 *         version of the file.
 ********************************************************************;

LABEL

 Y&YY.HICNO    = "Y&YY..1,2 BENE ACNTNUM"
 Y&YY.BIC      = "Y&YY..3 ORIGINAL BIC"
 Y&YY.STATE    = "Y&YY..4 BENE RSDNC SSASTD STATECD"
 Y&YY.CNTY     = "Y&YY..5 BENE RSDNC SSASTD CNTYCD"
 Y&YY.ZIP      = "Y&YY..6 BENE MLG CNTCT ZIPCD"
 Y&YY.DOB      = "Y&YY..7 BENE BIRTHDT"
 Y&YY.SEX      = "Y&YY..8 SEX"
 Y&YY.RACE     = "Y&YY..9 BENE RACECD"
 Y&YY.AGE      = "Y&YY..10 AGE"
 Y&YY.OREC     = "Y&YY..11 OREC"
 Y&YY.CREC     = "Y&YY..12 CREC"
 Y&YY.ESRD     = "Y&YY..13 ESRD IND"
 Y&YY.MSCD     = "Y&YY..14 BENE MDCR STUSCD"
 Y&YY.ATRM     = "Y&YY..15 ATRMCD"
 Y&YY.BTRM     = "Y&YY..16 BTRMCD"
 Y&YY.BUY1     = "Y&YY..18 BUYIN(1)"
 Y&YY.BUY2     = "Y&YY..18 BUYIN(2)"
 Y&YY.BUY3     = "Y&YY..18 BUYIN(3)"
 Y&YY.BUY4     = "Y&YY..18 BUYIN(4)"
 Y&YY.BUY5     = "Y&YY..18 BUYIN(5)"
 Y&YY.BUY6     = "Y&YY..18 BUYIN(6)"
 Y&YY.BUY7     = "Y&YY..18 BUYIN(7)"
 Y&YY.BUY8     = "Y&YY..18 BUYIN(8)"
 Y&YY.BUY9     = "Y&YY..18 BUYIN(9)"
 Y&YY.BUY10    = "Y&YY..18 BUYIN(10)"
 Y&YY.BUY11    = "Y&YY..18 BUYIN(11)"
 Y&YY.BUY12    = "Y&YY..18 BUYIN(12)"
 Y&YY.HMO1     = "Y&YY..19 HMOIND(1)"
 Y&YY.HMO2     = "Y&YY..19 HMOIND(2)"
 Y&YY.HMO3     = "Y&YY..19 HMOIND(3)"
 Y&YY.HMO4     = "Y&YY..19 HMOIND(4)"
 Y&YY.HMO5     = "Y&YY..19 HMOIND(5)"
 Y&YY.HMO6     = "Y&YY..19 HMOIND(6)"
 Y&YY.HMO7     = "Y&YY..19 HMOIND(7)"
 Y&YY.HMO8     = "Y&YY..19 HMOIND(8)"
 Y&YY.HMO9     = "Y&YY..19 HMOIND(9)"
 Y&YY.HMO10    = "Y&YY..19 HMOIND(10)"
 Y&YY.HMO11    = "Y&YY..19 HMOIND(11)"
 Y&YY.HMO12    = "Y&YY..19 HMOIND(12)"
 Y&YY.AMO      = "Y&YY..20 AMOCNT"
 Y&YY.BMO      = "Y&YY..21 BMOCNT"
 Y&YY.HMOM     = "Y&YY..22 HMO MO"
```

```
 Y&YY.BUYM     = "Y&YY..23 BUYIN MO"
 Y&YY.VDOD     = "Y&YY..24 VDODSW"
 Y&YY.DETH     = "Y&YY..25 BENE DEATHDT"
 Y&YY.RFRN     = "Y&YY..26 BENE ENRLMTRFRNC YR"
 Y&YY.DOBS     = "Y&YY UNIQUE OBSNO"
 ;
RUN;

PROC CONTENTS DATA=&OUT1..DENOMA&YY DIRECTORY POSITION;
TITLE "CONTENTS FOR DENOMINATOR A FILE Y&YY";
RUN;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1001) -- CONTINUED

```
PROC PRINT DATA=&OUT1..DENOMA&YY(OBS=20) SPLIT=' ';
TITLE2 "PRINT 20 OBS FOR Y&YY DENOM A FILE";
RUN;

 ********************************************************************;
 * STEP 7. Provide preliminary statistics (frequencies)
 *         for quality control purposes.
 ********************************************************************;

PROC FREQ DATA=&OUT1..DENOMA&YY;
TABLES Y&YY.DOB Y&YY.SEX Y&YY.RACE
       Y&YY.DETH Y&YY.MSCD;
FORMAT Y&YY.DOB YEAR4. Y&YY.DETH MMYYD7.;
TITLE "Y&YY DENOM A SELECTED FREQS ON SOME IMPORTANT VARIABLES";
RUN;


%MEND YXXM1001;                    /**MACRO ENDS**/
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM102)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/04/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM1001.DENOMAXX
                               -  BASE ANALYTIC DENOMINATOR FILE
                                  (TAPE)
*            D#MSQ.@MSQ0000.SAS.URBANF.URBANMSA
                               - URBAN CONTROL FILE (DISK)
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX
                               - M.1 FILE (TAPE)
             D#MSQ.@MSQ0000.SAS.YXXM1002.DENEXBXX
                               - M.1 EXCLUSIONS (DISK)
             D#MSQ.@MSQ0000.SAS.YXXM1002.DENRURXX
                               - M.1 FORCED RURAL NONMATCHES (DISK)
* PURPOSE   :FROM SAS  DENOMINATOR FILE, CREATE A SAS FILE
*            THAT WILL BE THE PRINCIPAL ANALYTIC FILE FOR M1 AND
*            SAVE IT TO TAPE.  ALSO, SAVE TO DISK INFO ON ALL
*            EXCLUSIONS FROM THIS FILE.
*            SEE MACRO YXXM1002 FOR MORE DETAILS
*********************************************************************;
OPTIONS BUFNO=5;


         /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1002)/ SOURCE2;

         /**MACRO CALL**/

%YXXM1002(YY=&SYSPARM)
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002)

```
%MACRO YXXM1002(YY=99,IN1=IN1,IN2=IN2,OUT1=OUT1,OUT2=OUT2,
                      OUT3=OUT3);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM1002
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 APRIL 30, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      CREATE THE BASE ANALYTIC DENOMINATOR FILE FOR M1 AND
      SAVE IT TO TAPE.
      ALSO, SAVE TO DISK INFORMATION ON THOSE RECORDS NOT SELECTED
      INTO THIS FILE.

8  PARAMETERS:
      THERE ARE 6 KEYWORD PARAMETERS:
      YY IN1 IN2 OUT1 OUT2

YY
IS THE 2-DIGIT YEAR OF THE DENOMINATOR FILE BEING READ INTO SAS
E.G.:
YY=99 (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR THE DENOM FILE CREATED IN PROGRAM
'YXXM1001'.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM1001.DENOMAXX (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.  SIZE IS ABOUT 4GB (3 TAPES).
FILE IS SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IS THE SAS LIBREF-JCL DDNAME FOR THE SAS FILE THAT WE WILL USE AS
A CONTROL DATASET FOR THE URBAN/RURAL INFORMAT.
THIS FILE WAS CREATED FROM THE MSABEA FILE OFF OF THE HCFA
WEB SITE IN PROGRAM URBANF.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.URBANF.URBANMSA
LAST QUALIFIER IS THE INTERNAL SAS NAME.
SIZE IS ABOUT 160KB.
E.G.: IN2=IN2  (THIS IS THE DEFAULT)


OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE DENOM
SAS FILE BEING WRITTEN TO TAPE.
THIS IS THE BASE ANALYTIC DENOMINATOR FILE FOR M1.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX (XX=YEAR)
LAST QUALIFIER IS THE INTERNAL SAS NAME.
SIZE WILL LIKELY BE AROUND 4GB (3 TAPES)
SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)
```

```
OUT2
IS THE SAS LIBREF-JCL DDNAME FOR THE DENOM PERMANENT SAS FILE ON DISK
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002) -- CONTINUED

THAT HOLDS INFO ON THE TYPE OF EXLUSIONS FROM THE M1
BASE ANALYTIC DENOMINATOR FILE.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1002.DENEXBXX (XX=YEAR)
LAST QUALIFIER IS INTERNAL SAS NAME.
SIZE WILL LIKELY BE AROUND 300MB.
SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
E.G.: OUT2=OUT2  (THIS IS THE DEFAULT)

OUT3
IS THE SAS LIBREF-JCL DDNAME FOR THE DENOM PERMANENT SAS FILE ON DISK
THAT HOLDS INFO ON THOSE RECORDS INCLUDED IN THE BASE ANALYTIC
DENOMINATOR FILE THAT WERE ASSUMED TO HAVE A RURAL STATUS BECAUSE
NO MSABEAMATCH.  FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1002.DENRURXX
(XX=YEAR)
LAST QUALIFIER IS INTERNAL SAS NAME.
SIZE WILL LIKELY BE AROUND 300MB.
SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
E.G.: OUT3=OUT3  (THIS IS THE DEFAULT)

9  OTHER MACRO VARIABLES OF NOTE: XX = FIRST 2 DIGITS OF 4-DIGIT YEAR

10  OTHER NOTES: NONE

11 EXAMPLE MACRO CALL:
%YXXM1002(YY=93,IN1=INFILE)

*********************************************************************
*********************************************************************;

                 /**CODE BEGINS**/

 ********************************************************************;
 * STEP 1. Create formats and informats for the key
 *         variables that allow for the later exclusion
 *         of values not appropriate for this system.
 ********************************************************************;

PROC FORMAT;
             INVALUE $STATE (DEFAULT=2)
                        /**STATE EXCLUSIONS**/
                    '54'-'64','66','67','97','99',OTHER='NO'
                        /**GUAM TRANSFORM**/
                    '65','98'='12'
                    '01'-'53'=_SAME_;

             INVALUE $RACES (DEFAULT=1)
                        /**OTHER RACE TRANSFORM**/
                    '3'-'6'='3'
                    '0'-'2'=_SAME_
                     OTHER='X';

             INVALUE $MEDST (DEFAULT=2)
                        /**MEDICARE STATUS**/
```

```
                    '10'='1' '11'='2' '20'='3' '21'='4'
                    '31'='5'
                    OTHER='0';

             VALUE FAGES (DEFAULT=3)
                        /**AGE GROUPS**/
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002) -- CONTINUED

                    0-54='1'  55-64='2'
                    65-69='3' 70-74='4' 75-79='5'
                    80-84='6' 85-89='7' 90-94='8'
                    95-HIGH='9' OTHER='X';

             INVALUE $CEND (DEFAULT=2)
                        /**CENSUS DIVISIONS**/
                     '01','18','25','44'='6'
                     '02','05','12','65','38','50','98'='9'
                     '03','06','13','27','29','32','46','53'='8'
                     '04','19','37','45'='7'
                     '07','20','22','30','41','47'='1'
                     '08','09','10','11','21','34','42','49','51'='5'
                     '14','15','23','36','52'='3'
                     '16','17','24','26','28','35','43'='4'
                     '31','33','39'='2'
                     '40','48'='0';

             INVALUE $CENR (DEFAULT=1)
                        /**CENSUS REGIONS**/
                     '1','2'='1'  '3','4'='2'  '5','6','7'='3'
                     '8','9'='4' OTHER=_SAME_;
RUN;

 ********************************************************************;
 * STEP 2. Reads the URBANF file in as a format to be
 *         used later to assign urban/rural status.
 ********************************************************************;

     /**URBAN/RURAL INFORMAT FROM SAS CONTROL DATASET**/

PROC FORMAT CNTLIN=&IN2..URBANMSA;
RUN;

                        /**DATA STEP BEGINS**/

 ********************************************************************;
 * STEP 3. Defines files to write data out to and what
 *         variables to keep in each of the files.
 ********************************************************************;

DATA &OUT1..DENOMB&YY
      (KEEP=Y&YY.SEX Y&YY.RACE Y&YY.CREG Y&YY.CDIV Y&YY.DOBS
            Y&YY.URBN Y&YY.STATE Y&YY.MDST Y&YY.DUAL Y&YY.DIE
            Y&YY.YEAR Y&YY.DOB Y&YY.CAGE Y&YY.FAGE Y&YY.DETH
            Y&YY.ZIP Y&YY.CNTY Y&YY.M13 Y&YY.M14 Y&YY.M7 Y&YY.DUALJ
            Y&YY.PARA Y&YY.PARB Y&YY.FEE Y&YY.FEEJ Y&YY.PARAJ
            Y&YY.HICNO Y&YY.VDOD
            LABEL="Y&YY DENOMB ANALYT FILE"
            SORTEDBY=Y&YY.DOBS)
```

```
       &OUT2..DENEXB&YY (KEEP=Y&YY.DOBS Y&YY.EXTP)
       &OUT3..DENRUR&YY (KEEP=Y&YY.DOBS)
       ;
LENGTH Y&YY.EXTP Y&YY.PARA Y&YY.PARB Y&YY.FEE I ENZ 2
       Y&YY.YEAR Y&YY.STATE $ 2
       Y&YY.URBN Y&YY.DIE Y&YY.RACE Y&YY.DUAL Y&YY.MDST Y&YY.FAGE
       Y&YY.CREG Y&YY.CDIV Y&YY.M13 Y&YY.M14 Y&YY.M7 Y&YY.DUALJ $ 1 ;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002) -- CONTINUED

```
SET   &IN1..DENOMA&YY;

 ********************************************************************;
 * STEP 4. Counts the months of Part A, Part B,
 *         FFS, and dual Medicaid enrollment.
 ********************************************************************;

                    /**SET BUYIN AND HMO ARRAYS**/

ARRAY BUY(12) $ Y&YY.BUY1-Y&YY.BUY12;
ARRAY HMO(12) $ Y&YY.HMO1-Y&YY.HMO12;

                     /**COUNT FFS, PART A, PART B MONTHS**/

%IF %SUBSTR(&YY,1,1) = 0 %THEN %LET XX = 20;
%ELSE %LET XX = 19;

IF YEAR(Y&YY.DETH) NE &XX&YY THEN ENZ=12;
ELSE ENZ=MONTH(Y&YY.DETH);

DO I = 1 TO ENZ ;
Y&YY.PARA=SUM((BUY(I) IN('1','3','A','C')),Y&YY.PARA);
Y&YY.PARB=SUM((BUY(I) IN('2','3','B','C')),Y&YY.PARB);
Y&YY.FEE =SUM(HMO(I)='0',Y&YY.FEE);
DUALZ=SUM(BUY(I) IN('A','B','C'),DUALZ);
END;

 ********************************************************************;
 * STEP 5. Major variable assignments
 ********************************************************************;

Y&YY.FEEJ =PUT(HMO(7)='0',1.);
Y&YY.PARAJ=PUT(BUY(7) IN('1','3','A','C'),1.);
Y&YY.DUALJ=PUT(BUY(7) IN('A','B','C'),1.);
Y&YY.M14  =PUT(Y&YY.PARA=ENZ AND Y&YY.FEE=ENZ,1.);
Y&YY.M13  =PUT(Y&YY.FEEJ='1' AND Y&YY.PARAJ='1',1.);
Y&YY.M7   =PUT(Y&YY.PARB=12 AND Y&YY.PARA=12
                  AND Y&YY.FEE=12,1.);
Y&YY.YEAR ="&YY";
Y&YY.URBN =INPUT((Y&YY.STATE!!Y&YY.CNTY),$MSABEA.);
Y&YY.DIE  =PUT((YEAR(Y&YY.DETH)=19&YY),1.);
Y&YY.STATE=INPUT(Y&YY.STATE,$STATE.);
Y&YY.RACE =INPUT(Y&YY.RACE,$RACES.);
Y&YY.DUAL =PUT(DUALZ GT 0,1.);
Y&YY.MDST =INPUT(Y&YY.MSCD,$MEDST.);
Y&YY.CAGE =FLOOR((INTCK('MONTH',Y&YY.DOB,"01JUL&XX&YY"D) -
 (1 < DAY(Y&YY.DOB))) / 12) * (YEAR(Y&YY.DOB) ~= &XX&YY);
```

```
Y&YY.FAGE =PUT(Y&YY.CAGE,FAGES.);
Y&YY.CDIV =INPUT(Y&YY.STATE,$CEND.);
Y&YY.CREG =INPUT(Y&YY.CDIV,$CENR.);

                     /**EXCLUSIONS(BITWISE FORM)**/

 ********************************************************************;
 * STEP 6. Creates the Exclusion Indicator Variable (EXTP),
 *         a three-digit variable that reflects the reason
 *         for exclusion from the MQMS Base Denominator File.
 ********************************************************************;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002) -- CONTINUED

```
Y&YY.EXTP=((Y&YY.STATE='NO')*(2**0)) +
          ((Y&YY.FAGE='X')*(2**1)) +
          ((NOT(Y&YY.SEX IN('1','2')))*(2**2)) +
          ((Y&YY.MDST='0')*(2**3)) +
          ((0 LE YEAR(Y&YY.DETH) LT &XX&YY)*(2**4)) +
          ((Y&YY.RACE='X')*(2**5)) +
          ((Y&YY.M13='0' AND Y&YY.M14='0' AND Y&YY.M7='0')*(2**6));

                /**OUTPUT**/

 ********************************************************************;
 * STEP 7. Excludes beneficiaries that meet the exclusion
 *         criteria, assigns beneficiaries with a non-match
 *         to the urban/rural file to rural status, and
 *         writes out non-excluded beneficiaries to the
 *         DENOMBYY file.
 ********************************************************************;

IF Y&YY.EXTP NE 0 THEN OUTPUT &OUT2..DENEXB&YY;
ELSE DO;
  IF Y&YY.URBN = 'X' THEN DO;
       OUTPUT &OUT3..DENRUR&YY;
       Y&YY.URBN = '0';
  END;
  OUTPUT &OUT1..DENOMB&YY;
END;

 ********************************************************************;
 * STEP 8. Creates labels for major analytic variables.
 ********************************************************************;

LABEL
Y&YY.FEE   = "Y&YY FFS MOS"
Y&YY.PARA  = "Y&YY PARA MOS"
Y&YY.PARB  = "Y&YY PARB MOS"
Y&YY.DUAL  = "Y&YY EVER DUAL"
Y&YY.DUALJ = "Y&YY DUAL JULY"
Y&YY.FEEJ  = "Y&YY FFS JULY"
Y&YY.PARAJ = "Y&YY PARTA JULY"
Y&YY.MDST  = "Y&YY MEDICARE STATUS"
Y&YY.CAGE  = "Y&YY CALCULATED AGE"
Y&YY.FAGE  = "Y&YY AGE GROUPS"
Y&YY.M14   = "Y&YY M14 COHORT:YEAR"
Y&YY.M13   = "Y&YY M13 COHORT:JULY"
```

```
Y&YY.M7    = "Y&YY M7 COHORT:DIABETES"
Y&YY.URBN  = "Y&YY URBAN STATUS"
Y&YY.DIE   = "Y&YY BENE DIED"
Y&YY.RACE  = "Y&YY RACE"
Y&YY.STATE = "Y&YY STATE"
Y&YY.CDIV  = "Y&YY CENSUS DIVISIONS"
Y&YY.CREG  = "Y&YY CENSUS REGIONS"
Y&YY.YEAR  = "Y&YY YEAR"
Y&YY.HICNO = "Y&YY ID"
Y&YY.EXTP  = "Y&YY BINARY EXCLUSION CODE"
;
                          /**DATA STEP ENDS**/
RUN;

 ******************************************************************;
```
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1002) -- CONTINUED
```
 * STEP 9. Frequency on the exclusion code (EXTP variable)
 *         and a proc contents and proc print of the
 *         DENEXBYY, DENOMBYY, and DENRURYY Files.
 ******************************************************************;

TITLE "Y&YY DENOM B: M.1 BASE ANALYTIC DENOMINATOR FILE";
PROC FREQ DATA=&OUT2..DENEXB&YY ORDER=FREQ;
TABLES Y&YY.EXTP;
FORMAT Y&YY.EXTP BINARY7.;
TITLE2 "E-X-C-L-U-S-I-O-N-S";
TITLE3 "BIT 1 TO BIT 7 VALUES (LEFT TO RIGHT) ARE AS FOLLOWS:";
TITLE4 "COHORT/RACE/DEATH/MEDST/SEX/DOB/STATE";
RUN;

TITLE2 "!!!!EXCLUSIONS FILE!!!!";
PROC CONTENTS DATA=&OUT2..DENEXB&YY DIRECTORY;
RUN;

TITLE3 "PRINT 10 OBSERVATIONS";
PROC PRINT DATA=&OUT2..DENEXB&YY(OBS=10);
RUN;


TITLE2;
PROC CONTENTS DATA=&OUT1..DENOMB&YY DIRECTORY POSITION;
RUN;

PROC PRINT DATA=&OUT1..DENOMB&YY(OBS=20) SPLIT=' ';
ID Y&YY.DOBS;
TITLE2 "PRINT 20 RECORDS";
RUN;

TITLE2 "!!!!MSABEA ASSUMED RURAL FILE!!!!";
PROC CONTENTS DATA=&OUT3..DENRUR&YY DIRECTORY;
RUN;

TITLE3 "PRINT 10 OBSERVATIONS";
PROC PRINT DATA=&OUT3..DENRUR&YY(OBS=10);
RUN;
       /**MACRO ENDS**/
```

```
%MEND YXXM1002;
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM103)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/06/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX
*                              - DENOM B FILE (TAPE)
*                              - BASE ANALYTIC DENOMINATOR FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM1003.M1CNTSXX
*                    - M.1 COUNT FILE FOR M.1.3 AND M.1.4 (DISK)
* PURPOSE   :FROM SAS M.1 PRINCIPAL ANALYTIC FILE, CREATE A SAS
*            FILE ON DISK THAT PROVIDES COUNTS FOR M.1.3 AND M.1.4
*            FOR EACH MAJOR GROUP VARIABLE.
*            SEE MACRO YXXM1003 FOR MORE DETAILS
*********************************************************************;
OPTIONS BUFNO=5 ;


         /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1003)/ SOURCE2;

         /**MACRO CALL**/

%YXXM1003(YY=&SYSPARM)
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003)

```
%MACRO YXXM1003(YY=99,IN1=IN1,OUT1=OUT1);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM1003
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 9, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      CREATE THE ANALYTIC COUNT FILE FOR M.1.3 AND M.1.4 AND
      SAVE THE THIS SMALL FILE TO DISK.

8  PARAMETERS:
      THERE ARE 3 KEYWORD PARAMETERS:
      YY IN1 OUT1

YY
IS THE 2-DIGIT YEAR OF THE BASE ANALYTIC DENOMINATOR
M.1 SAS FILE CREATED IN PROGRAM 'YXXM002'
E.G.: YY=99     (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR
M.1 DENOM SAS FILE CONSTRUCTED IN PROGRAM 'YXXM1002'.
THIS IS THE BASE ANALYTIC DENOMINATOR FILE FOR M1.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX (XX=YEAR)
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR)
SIZE WILL LIKELY BE AROUND 4GB (3 TAPES)
E.G.: IN1=IN1      (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE PRINCIPAL FILE
GENERATED BY THIS PROGRAM. THIS FILE WILL BE A SMALL SAS FILE
(ABOUT 4KB) AND WILL BE WRITTEN TO DISK.
IT CONTAINS THE DENOMINATOR COUNTS FOR M.1.3 AND M.1.4.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1003.M1CNTSXX (XX=YEAR)
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE IS SORTED BY VARIABLE 'YXXVARS' (XX=YEAR).
E.G.: OUT1=OUT1     (THIS IS THE DEFAULT)

9  OTHER MACRO VARIABLES OF NOTE:
VARIABLE 'VARS' LISTS THE VARIABLES IN THE M.1
SAS ANALYTIC FILE FOR WHICH WE WILL GENERATE FREQUENCIES.

10  OTHER NOTES:
THE WORK LIBRARY CAN BE RELATIVELY SMALL: ABOUT 100KB.

11 EXAMPLE MACRO CALL:
%YXXM1003(YY=93)

*****************************************************************
```

```
*********************************************************************;

             /**CODE BEGINS**/
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003) -- CONTINUED

```
        /**SET UP LOOP FOR BOTH COHORTS**/

 *********************************************************************;
 *STEP 1. Prepares the cohorts to create the counts.
 *        sets up do loops for the 2 cohorts with the
 *        first loop applying to M1-3 and the second to M1-4.
 *********************************************************************;


%DO J = 1 %TO 2;            /**START LOOP 1**/

%LET COHORT=M13 M14;
%LET ME=%SCAN(&COHORT,&J);

%IF &ME=M13 %THEN %LET JU=J;
%ELSE %IF &ME=M14 %THEN %LET JU= ;

%LET VARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL&JU Y&YY.FAGE Y&YY.MDST
          Y&YY.RACE Y&YY.SEX Y&YY.STATE Y&YY.URBN;

      /**SET UP LOOP FOR ALL 9 VARS**/

%DO I = 1 %TO 9;            /**START LOOP 2**/

      /**OUTPUT COUNT FILE FOR EACH VARIABLE**/

 *********************************************************************;
 * STEP 2. Outputs frequencies for each of the 9 subgroups
 *         to 9 temp files (TEMP1 - TEMP9) corresponding
 *         to the 9 subgroups.
 *********************************************************************;

PROC FREQ DATA=&IN1..DENOMB&YY (WHERE=(Y&YY&ME='1')) NOPRINT;
TABLES %SCAN(&VARS,&I)
   / OUT=TEMP&I(DROP=PERCENT);
RUN;

DATA TEMP&I;
SET TEMP&I;
IF %SCAN(&VARS,&I) = ' ' THEN DO;
PUT "*********************************************************" /
    "************************WARNING*****************************" /
    "VARIABLE %SCAN(&VARS,&I) HAS MISSING VALUES"                 /
    "THIS SHOULD NOT HAVE OCCURRED: PLEASE CHECK M.1. PROGRAMS  " /
    "*********************************************************" ;
DELETE;
END;
RUN;

     /**NAME THE VALUES OF YXXVARS AND SET THEIR**/
     /**ATTRIBUTES APPROPRIATELY**/
```

```
 ********************************************************************;
 * STEP 3. Prepares the 9 temp files to be alphabetically
 *         consolidated into one file.
 ********************************************************************;

DATA TEMP&I(DROP=%SCAN(&VARS,&I) RENAME=(COUNT=Y&YY.CT&ME));
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003) -- CONTINUED

LENGTH Y&YY.VARS $8;
SET TEMP&I;
Y&YY.VARS=COMPRESS("%SUBSTR(%SCAN(&VARS,&I),4,3)"!!%SCAN(&VARS,&I));
RUN;

%IF &I=9 %THEN %DO;

DATA TEMP10(DROP=Y&YY.CT&ME RENAME=(COUNT=Y&YY.CT&ME));
LENGTH Y&YY.VARS $8;
SET TEMP9 END=FINI;
COUNT+Y&YY.CT&ME;
IF FINI THEN DO;
    Y&YY.VARS='TOTAL';
    OUTPUT;
END;
RUN;

%END;

%END;                    /**END LOOP 2**/

            /**CONCATENATE THE COUNT FILES FOR ALL 9 VARS**/

 ********************************************************************;
 * STEP 4. Stacks the 10 temp files alphabetically to
 *         create TEMPA1 (for M1-3 cohort) and then
 *         deletes the temp1-10 files.
 ********************************************************************;

DATA TEMPA&J(LABEL="Y&YY &ME COUNT FILE"
            SORTEDBY=Y&YY.VARS);
SET TEMP1 TEMP2 TEMP3 TEMP4 TEMP5 TEMP6 TEMP7 TEMP8
    TEMP10 TEMP9;
RUN;

PROC DATASETS LIB=WORK NOLIST;
DELETE TEMP1-TEMP10;
RUN;

%END;                 /**END LOOP 1**/

    /**THIS IS THE PRINCIPAL OUTPUT FILE**/

 ********************************************************************;
 * STEP 5. Merges the 2 cohort files together.
 ********************************************************************;

DATA &OUT1..M1CNTS&YY(LABEL="Y&YY M1 COUNT FILE:M13/M14"
                      SORTEDBY=Y&YY.VARS);
MERGE TEMPA1 TEMPA2;
```

```
BY Y&YY.VARS;
IF Y&YY.CTM13=. THEN Y&YY.CTM13=0;
IF Y&YY.CTM14=. THEN Y&YY.CTM14=0;
LABEL
Y&YY.VARS = "MAJOR CATEGORY VARIABLE LEVELS"
Y&YY.CTM13  = "M.1.3 PERSON COUNTS"
Y&YY.CTM14  = "M.1.4 PERSON COUNTS"
;
RUN;
```
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1003) -- CONTINUED

```
 ********************************************************************;
 * STEP 6. Proc contents and proc print on the analytic count file.
 ********************************************************************;

PROC CONTENTS DATA=&OUT1..M1CNTS&YY DIRECTORY POSITION;
TITLE "Y&YY M.1 COUNT FILE FOR M.1.3 AND M.1.4";
RUN;

PROC PRINT DATA=&OUT1..M1CNTS&YY NOOBS;
TITLE2 "PRINT FULL M.1 COUNT FILE";
FORMAT Y&YY.CTM13 Y&YY.CTM14 COMMA10.;
RUN;

                    /**MACRO ENDS**/
%MEND YXXM1003;
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM104)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/11/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(YXXM1004)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99- MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMB99
*                              - DENOM B FILE (TAPE)
*                              - BASE ANALYTIC DENOMINATOR FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM1004.AGESEX99
*                    - M.1 AGE/AGE-SEX ADJUSTMENT FILE (DISK)
* PURPOSE   :FROM SAS M.1 BASE ANALYTIC DENOMINATOR FILE FOR Y99,
*            CREATE A SMALL SAS AGE AND AGE-SEX ADJUSTMENT
*            FILE ON DISK. THIS IS FOR Y99 ONLY.
*            SEE MACRO YXXM1004 FOR MORE DETAILS
*********************************************************************;
OPTIONS BUFNO=5 ;


         /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1004)/ SOURCE2;

         /**MACRO CALL**/

%YXXM1004
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1004)

```
%MACRO YXXM1004(YY=99,IN1=IN1,OUT1=OUT1);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM1004
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1004)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 9, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      CREATE THE ANALYTIC AGE/AGE-SEX ADJUSTMENT FILE FROM
      THE 1999 M.1.3 FILE, AND SAVE IT TO DISK.

8  PARAMETERS:
      THERE ARE 2 KEYWORD PARAMETERS:
      IN1 OUT1


IN1
IS THE SAS LIBREF-JCL DDNAME FOR THE BASE ANALYTIC DENOMINATOR FILE
CREATED IN PROGRAM 'YXXM1002'.
FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX.
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE IS ON TAPE AND IS SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR).
SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE AGE/AGE-SEX ADJUSTMENT FILE
CREATED BY THIS PROGRAM.
FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.YXXM1004.AGESEXXX.
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE IS ON DISK AND WILL ONLY ABOUT 10KB.
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

9  OTHER SYMBOLIC VARIABLES OF NOTE: NONE


10  OTHER NOTES:
THE WORK LIBRARY CAN BE SMALL: ABOUT 100KB.
THIS MACRO SHOULD BE RUN ONLY FOR YEAR 1999.

11 EXAMPLE MACRO CALL:
%YXXM1004(IN1=INFILE)

*****************************************************************
*****************************************************************;

             /**CODE BEGINS**/

 ****************************************************************;
 * STEP 1. Frequencies on age, sex, and age/sex for
 *         the M1-3 cohort and writes these counts out
```

```
 *         to a file.  This, as defined in M1PROD, is
 *         only done for 1999.
 ************************************************************************;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1004) -- CONTINUED

```
PROC MEANS DATA=&IN1..DENOMB&YY
(WHERE=(Y&YY.M13='1')) NOPRINT;
CLASS Y&YY.FAGE Y&YY.SEX;
VAR Y&YY.DOBS;
OUTPUT OUT=&OUT1..AGESEX&YY
    (DROP=_FREQ_ _TYPE_
    LABEL="SEX/AGE-SEX ADJUST FILE&YY")
    N=Y&YY.CNTS;
RUN;

 ************************************************************************;
 * STEP 2. Proc print and contents on the analytic
 *         age/age-sex adjustment file.
 ************************************************************************;

PROC PRINT DATA=&OUT1..AGESEX&YY;
FORMAT Y&YY.CNTS COMMA10.;
TITLE "FULL M.1 AGE AND AGE-SEX ADJUSTMENT FILE: 19&YY";
RUN;

PROC CONTENTS DATA=&OUT1..AGESEX&YY DIRECTORY;
RUN;
              /**MACRO ENDS**/
%MEND YXXM1004;
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM105)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/11/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM1003.M1CNTSXX
*                              - YXX M.1 COUNT FILE (DISK)
* OUTPUTS   :TEXT ONLY
* PURPOSE   :FROM SAS M.1 COUNT FILE FOR M.1.3 AND M.1.4
*            COHORTS, CREATE A REPORT OF THOSE COUNTS.
*            SEE MACRO YXXM1005 FOR MORE DETAILS
*********************************************************************;
OPTIONS BUFNO=5 ;

         /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1005)/ SOURCE2;

         /**MACRO CALL**/

%YXXM1005(YY=&SYSPARM)
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005)

```
%MACRO YXXM1005(YY=99,IN1=IN1);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM1005
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 9, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      GENERATE A REPORT OF THE M.1.3 AND M.1.4 COUNTS.

8  PARAMETERS:
      THERE ARE 2 KEYWORD PARAMETERS:
      YY IN1

YY
IS THE 2 DIGIT YEAR.
E.G.: YY=99    (THIS IS THE DEFAULT)
IN1
IS THE LIBREF-JCL DDNAME FOR THE M.1 COUNT FILE
CREATED IN PROGRAM 'YXXM1003'.
FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.YXXM1003.M1CNTSXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME.
THIS FILE IS ON DISK AND IS ABOUT 10KB.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

9  OTHER SYMBOLIC VARIABLES OF NOTE: NONE

10  OTHER NOTES:
THE WORK LIBRARY CAN BE SMALL: ABOUT 50KB.

11 EXAMPLE MACRO CALL:
%YXXM1005(IN1=INFILE)

*******************************************************************
*******************************************************************;

             /**CODE BEGINS**/

 ********************************************************************;
 * STEP 1. Creating formats to be used when writing out
 *         the reports of counts for M1-3 and M1-4 by subgroupings.
 ********************************************************************;

PROC FORMAT;

VALUE $ONE

'TOTAL'='ALL ENROLLEES '
'SEX1'-'SEX2'='SEX'
'FAG1'-'FAG9'='AGE GROUP'
'RAC0'-'RAC3'='RACE'
```

```
'MDS1'-'MDS5'='MEDICARE ELIGIBILITY STATUS'
'DUA0'-'DUA1'='OTHER DISADVANTAGED STATUS'
'URB0'-'URB1'='URBAN/RURAL AREA'
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005) -- CONTINUED

```
;
VALUE $TWO
'FAG1' ='0-54' 'FAG2'='55-64'
'FAG3' ='65-69' 'FAG4'='70-74'  'FAG5'='75-79'
'FAG6'='80-84' 'FAG7'='85-89' 'FAG8'='90-94'
'FAG9'='95-UP'

'SEX1'='MALE'
'SEX2'='FEMALE'
'RAC0'='UNKNOWN'
'RAC1'='WHITE'
'RAC2'='AFRICAN AMERICAN'
'RAC3'='OTHER'

'MDS1'='AGED WITHOUT ESRD'
'MDS2'='AGED WITH ESRD'
'MDS3'='DISABLED WITHOUT ESRD'
'MDS4'='DISABLED WITH ESRD'
'MDS5'='ESRD ONLY'

'DUA1'='ENROLLED IN MEDICAID'
'DUA0'='NOT ENROLLED IN MEDICAID'

'URB0'='RURAL'
'URB1'='URBAN'
;


VALUE $SIX
'STA01'='ALABAMA'
'STA02'='ALASKA'
'STA03'='ARIZONA'
'STA04'='ARKANSAS'
'STA05'='CALIFORNIA'
'STA06'='COLORADO'
'STA07'='CONNECTICUT'
'STA08'='DELAWARE'
'STA09'='DISTRICT OF COLUMBIA'
'STA10'='FLORIDA'
'STA11'='GEORGIA'
'STA12'='HAWAII,GUAM'
'STA13'='IDAHO'
'STA14'='ILLINOIS'
'STA15'='INDIANA'
'STA16'='IOWA'
'STA17'='KANSAS'
'STA18'='KENTUCKY'
'STA19'='LOUISIANA'
'STA20'='MAINE'
'STA21'='MARYLAND'
'STA22'='MASSACHUSETTS'
'STA23'='MICHIGAN'
```

```
'STA24'='MINNESOTA'
'STA25'='MISSISSIPPI'
'STA26'='MISSOURI'
'STA27'='MONTANA'
'STA28'='NEBRASKA'
'STA29'='NEVADA'
```
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005) -- CONTINUED

```
'STA30'='NEW HAMPSHIRE'
'STA31'='NEW JERSEY'
'STA32'='NEW MEXICO'
'STA33'='NEW YORK'
'STA34'='NORTH CAROLINA'
'STA35'='NORTH DAKOTA'
'STA36'='OHIO'
'STA37'='OKLAHOMA'
'STA38'='OREGON'
'STA39'='PENNSYLVANIA'
'STA40'='PUERTO RICO'
'STA41'='RHODE ISLAND'
'STA42'='SOUTH CAROLINA'
'STA43'='SOUTH DAKOTA'
'STA44'='TENNESSEE'
'STA45'='TEXAS'
'STA46'='UTAH'
'STA47'='VERMONT'
'STA48'='VIRGIN ISLANDS'
'STA49'='VIRGINIA'
'STA50'='WASHINGTON'
'STA51'='WEST VIRGINIA'
'STA52'='WISCONSIN'
'STA53'='WYOMING'
OTHER = ' ';


VALUE $FIVE
'CDI0'='OUTSIDE U.S.'
'CDI1'='NEW ENGLAND'
'CDI2'='MIDDLE ATLANTIC'
'CDI3'='EAST NORTH CENTRAL'
'CDI4'='WEST NORTH CENTRAL'
'CDI5'='SOUTH ATLANTIC'
'CDI6'='EAST SOUTH CENTRAL'
'CDI7'='WEST SOUTH CENTRAL'
'CDI8'='MOUNTAIN'
'CDI9'='PACIFIC'
OTHER = ' ';


VALUE $FOUR
'CRE0'='OUTSIDE U.S.'
'CRE1'='NORTHEAST'
'CRE2'='NORTH CENTRAL'
'CRE3'='SOUTH'
'CRE4'='WEST'
OTHER = ' ';

 ********************************************************************;
```

```
 * STEP 2. Orders geographic categories by census region,
 *         then the census divisions within the region,
 *         followed by the states within the region.
 ****************************************************************;

    /**FORMAT TO ORDER GEOGRAPHIC CATEGORIES**/

VALUE $THREE
'CRE0'='YY'
```
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005) -- CONTINUED
```
'CDI0'='ZZ'
'STA40','STA48'='00'
'CRE1'='01'
'CDI1'='02'
'STA07','STA20','STA22','STA30','STA41','STA47'='03'
'CDI2'='04'
'STA31','STA33','STA39'='05'
'CRE2'='06'
'CDI3'='07'
'STA14','STA15','STA23','STA36','STA52'='08'
'CDI4'='09'
'STA16','STA17','STA24','STA26','STA28','STA35','STA43'='10'
'CRE3'='11'
'CDI5'='12'
'STA08','STA09','STA10','STA11','STA21','STA34','STA42',
                     'STA49','STA51'='13'
'CDI6'='14'
'STA01','STA18','STA25','STA44'='15'
'CDI7'='16'
'STA04','STA19','STA37','STA45'='17'
'CRE4'='18'
'CDI8'='19'
'STA03','STA06','STA13','STA27','STA29','STA32','STA46','STA53'='20'
'CDI9'='21'
'STA02','STA05','STA12','STA65','STA98','STA38','STA50'='22'
OTHER ='XX';
RUN;


    /**SORT DATA WITH GEOGRAPHIC CATEGORIES APPROPRIATELY**/
    /**ORDERED AMONG THEMSELVES AT THE BOTTOM OF THE FILE**/

 ****************************************************************;
 * STEP 3. Read in counts file from YXXM1003 and
 *         sorts them to put the non-geographic variables
 *         first, followed by the ordered geographic categories.
 ****************************************************************;

DATA TEMP1;
LENGTH SORTER $ 2;
SET &IN1..M1CNTS&YY(KEEP=Y&YY.VARS Y&YY.CTM13 Y&YY.CTM14);
SORTER=PUT(Y&YY.VARS,$THREE.);
RUN;

PROC SORT DATA=TEMP1 OUT=TEMP1;
BY SORTER;
RUN;
```

```
      /**PLACE TOTAL AT THE TOP OF THE FILE AND CONVERT**/
      /**COUNTS TO THOUSANDS**/

 ********************************************************************;
 * STEP 4. Puts a total count at the top of the file
 *         and converts the counts to thousands.
 ********************************************************************;

DATA TEMP2 TEMP3;
SET TEMP1(WHERE=(Y&YY.VARS='TOTAL'))
TEMP1(WHERE=(Y&YY.VARS NE 'TOTAL'))
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1005) -- CONTINUED

;
Y&YY.CTM13=ROUND(Y&YY.CTM13/1000,1);
Y&YY.CTM14=ROUND(Y&YY.CTM14/1000,1);
IF SORTER='XX' THEN OUTPUT TEMP2;
ELSE OUTPUT TEMP3;
DROP SORTER;
RUN;

      /**BEGIN LOOP TO GENERATE 2 REPORTS: **/
      /**ONE FOR EACH COHORT: M.1.3. AND M.1.4**/

 ********************************************************************;
 * STEP 5. Sets up 2 do loops. The first loop generates
 *         two reports for M1-3.  The 1st report is generated
 *         from TEMP2 (non-geographic subgroups) and the
 *         2nd report is generated from TEMP3
 *         (geographic subgroups).  Once the reports are
 *         generated for M1-3, the 2nd do loop runs and
 *         generates the reports for M1-4.
 ********************************************************************;

%LET VARS=Y&YY.CTM13 Y&YY.CTM14;

%DO I = 1 %TO 2;

TITLE "                TABLE  %SCAN(&VARS,&I)                ";

TITLE3 "NUMBER OF MEDICARE FEE-FOR-SERVICE ENROLLEES";
TITLE4 "INCLUDED IN THE MEDICARE QUALITY MONITORING SYSTEM (MQMS)";
TITLE5 "BY AGE GROUP, SEX, RACE, MEDICARE ELIGIBILITY STATUS";
TITLE6 "             FOR Y&YY";

PROC REPORT DATA=TEMP2
(KEEP=Y&YY.VARS %SCAN(&VARS,&I))
NOWD SPACING=3 HEADLINE HEADSKIP PS=66 LS=110 NOCENTER;
COLUMN Y&YY.VARS Y&YY.VARS=NAMES %SCAN(&VARS,&I);
DEFINE Y&YY.VARS/ORDER ORDER=DATA FORMAT=$ONE.
               'DEMOGRAPHIC/CHARACTERISTICS';
DEFINE NAMES/ORDER FORMAT=$TWO. ORDER=DATA 'SUB-GROUPS';
DEFINE %SCAN(&VARS,&I)/FORMAT=COMMA9. "Y&YY/(IN,000S)";
BREAK AFTER Y&YY.VARS /OL SKIP;
RUN;

TITLE5 "FOR GEOGRAPHIC AREA FOR Y&YY";
```

```
PROC REPORT DATA=TEMP3
(KEEP=Y&YY.VARS %SCAN(&VARS,&I))
NOWD SPACING=3 HEADLINE HEADSKIP PS=66 LS=110 NOCENTER;
COLUMN Y&YY.VARS Y&YY.VARS=CDIVS Y&YY.VARS=STATES %SCAN(&VARS,&I);
DEFINE Y&YY.VARS/FORMAT=$FOUR.'CENSUS REGION';
DEFINE CDIVS/FORMAT=$FIVE. 'CENSUS DIVISION';
DEFINE STATES/FORMAT=$SIX. 'STATES';
DEFINE %SCAN(&VARS,&I)/FORMAT=COMMA9. "Y&YY/(IN,000S)";
RUN;
%END;

       /**MACRO ENDS**/

%MEND YXXM1005;
```

D#MSQ.@MSQ00M1.PGMLIB99(SYSM106)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :05/01/01
* EXEC DATE :06/06/01
* PGM       :D#MSQ.@MSQ00M1.PGMLIB99(Y99M1006)
* PGMR      :MATTHEW URATO
* MODIFIED  :HELEN MARGULIS
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M1.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMBXX
*                              - DENOM B FILE (TAPE)
*                              - BASE ANALYTIC DENOMINATOR FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM1006.M1CNTSXX
*                    - M.1 COUNT FILE FOR M.1.3 AND M.1.4 (DISK)
* PURPOSE   :THIS IS A MODIFICATION OF YXXM1003 PROGRAM TO SPLIT
*            COUNTERS BY AGE/SEX CATEGORIES
*********************************************************************;
OPTIONS BUFNO=5;

         /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM1006)/ SOURCE2;

         /**MACRO CALL**/

%YXXM1006(YY=&SYSPARM)
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1006)

```
%MACRO YXXM1006(YY=99,IN1=IN1,OUT1=OUT1);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM1006
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM1006)
3  WRITTEN BY:           MATTHEW URATO
   MODIFIED BY:          HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JULY 16, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      CREATE THE ANALYTIC COUNT FILE FOR M.1.3 AND M.1.4 AND
      SAVE THE THIS SMALL FILE TO DISK.

   MODIFICATION: COUNTERS ARE SPLIT BY AGE/SEX CATEGORIES
8  PARAMETERS:
      THERE ARE 3 KEYWORD PARAMETERS:
      YY IN1 OUT1

YY
IS THE 2-DIGIT YEAR OF THE BASE ANALYTIC DENOMINATOR
M.1 SAS FILE CREATED IN PROGRAM 'YXXM002'
E.G.: YY=99     (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR
M.1 DENOM SAS FILE CONSTRUCTED IN PROGRAM 'YXXM1002'.
THIS IS THE BASE ANALYTIC DENOMINATOR FILE FOR M1.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX (XX=YEAR)
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE SORTED BY VARIABLE 'YXXDOBS' (XX=YEAR)
SIZE WILL LIKELY BE AROUND 4GB (3 TAPES)
E.G.: IN1=IN1      (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE PRINCIPAL FILE
GENERATED BY THIS PROGRAM. THIS FILE WILL BE A SMALL SAS FILE
(ABOUT 94KB) AND WILL BE WRITTEN TO DISK.
IT CONTAINS THE DENOMINATOR COUNTS FOR M.1.3 AND M.1.4.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM1006.M1ASCNXX (XX=YEAR)
LAST QUALIFIER IS THE INTERNAL SAS NAME.
FILE IS SORTED BY VARIABLE 'YXXVARS' (XX=YEAR).
E.G.: OUT1=OUT1     (THIS IS THE DEFAULT)

9  OTHER MACRO VARIABLES OF NOTE:
VARIABLE 'VARS' LISTS THE VARIABLES IN THE M.1
SAS ANALYTIC FILE FOR WHICH WE WILL GENERATE FREQUENCIES.

10 EXAMPLE MACRO CALL:
%YXXM1006(YY=93)

*******************************************************************
*******************************************************************;
```

```
 ********************************************************************;
 * STEP 1. Sorts the Base Analytic Denominator File
D#MSQ.@MSQ00M1.PGMLIB99(YXXM1006) -- CONTINUED

 *         by beneficiary age and sex.
 ********************************************************************;

             /**CODE BEGINS**/
            /**SORT BY AGE/SEX**/

 PROC SORT DATA=&IN1..DENOMB&YY
           (KEEP= Y&YY.DUALJ Y&YY.M13  Y&YY.M14
            Y&YY.CDIV Y&YY.CREG Y&YY.DUAL    Y&YY.FAGE Y&YY.MDST
            Y&YY.RACE Y&YY.SEX Y&YY.STATE Y&YY.URBN)
           OUT=WORK0.TEMP1;
      BY  Y&YY.FAGE Y&YY.SEX;
 RUN;

 ********************************************************************;
 * STEP 2. Creates new age and sex variables to be used in
 *         the "by" statements when frequencies of the
 *         subgroups are run because age and sex are also
 *         subgroup variables.
 ********************************************************************;

        /** TO CREATE MIRRORED AGE & SEX VARS TO BE ABLE **/
    /** TO DISTINGUISH "BY" GROUPS FROM ANALYTICAL SUBGROUPS  **/

 DATA WORK1.TEMP1;
      SET WORK0.TEMP1;
      Y&YY.FAGEM = Y&YY.FAGE;
      Y&YY.SEXM=Y&YY.SEX;
 RUN;

 ********************************************************************;
 * STEP 3. Prepares the cohorts to create the counts.
 *         Sets up do loops for the 2 cohorts with the
 *         first loop applying to M1-3 and the second to M1-4.
 ********************************************************************;

%DO J = 1 %TO 2;            /**START LOOP 1**/

%LET COHORT=M13 M14;
%LET ME=%SCAN(&COHORT,&J);

%IF &ME=M13 %THEN %LET JU=J;
%ELSE %IF &ME=M14 %THEN %LET JU= ;

%LET VARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL&JU Y&YY.FAGE Y&YY.MDST
          Y&YY.RACE Y&YY.SEX Y&YY.STATE Y&YY.URBN;


      /**SET UP LOOP FOR ALL 9 VARS**/

%DO I = 1 %TO 9;            /**START LOOP 2**/

 ********************************************************************;
```

```
 * STEP 4. Outputs frequencies for each of the 9 subgroups
 *         to 9 temp files (TEMP1 - TEMP9) corresponding
 *         to the 9 subgroups.
 ********************************************************************;

      /**OUTPUT COUNT FILE FOR EACH VARIABLE**/
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1006) -- CONTINUED

```
PROC FREQ DATA=WORK1.TEMP1 (WHERE=(Y&YY&ME='1')) NOPRINT;
      BY  Y&YY.FAGEM Y&YY.SEXM;
TABLES %SCAN(&VARS,&I)
   / OUT=TEMP&I(DROP=PERCENT);
RUN;

 ********************************************************************;
 * STEP 5. Prepares the 9 temp files to be alphabetically
 *         consolidated into one file.
 ********************************************************************;

     /**NAME THE VALUES OF YXXVARS AND SET THEIR**/
     /**ATTRIBUTES APPROPRIATELY**/

DATA TEMP&I(DROP=%SCAN(&VARS,&I) RENAME=(COUNT=Y&YY.CT&ME));
LENGTH Y&YY.VARS $8;
SET TEMP&I;
Y&YY.VARS=COMPRESS("%SUBSTR(%SCAN(&VARS,&I),4,3)"!!%SCAN(&VARS,&I));
RUN;

%IF &I=9 %THEN %DO;

DATA TEMP10(DROP=Y&YY.CT&ME RENAME=(COUNT=Y&YY.CT&ME));
      SET TEMP9;
      BY  Y&YY.FAGEM Y&YY.SEXM;
      LENGTH Y&YY.VARS $8;
      RETAIN COUNT 0;
      Y&YY.VARS='TOTAL';
      COUNT =  COUNT+Y&YY.CT&ME;
      IF LAST.Y&YY.SEXM THEN DO;
         OUTPUT;
         COUNT=0;
      END;
RUN;

%END;

%END;                    /**END LOOP 2**/

 ********************************************************************;
 * STEP 6. Stacks the 10 temp files alphabetically to
 *         create TEMPA1 (for M1-3 cohort) and then
 *         deletes the temp1-10 files.
 ********************************************************************;

            /**CONCATENATE THE COUNT FILES FOR ALL 9 VARS**/

DATA TEMPA&J(LABEL="Y&YY &ME COUNT FILE"
            SORTEDBY=Y&YY.FAGEM Y&YY.SEXM Y&YY.VARS);
```

```
SET TEMP1 TEMP2 TEMP3 TEMP4 TEMP5 TEMP6 TEMP7 TEMP8
    TEMP10 TEMP9;
    BY  Y&YY.FAGEM Y&YY.SEXM;
RUN;

PROC DATASETS LIB=WORK NOLIST;
DELETE TEMP1-TEMP10;
RUN;
```

D#MSQ.@MSQ00M1.PGMLIB99(YXXM1006) -- CONTINUED

```
%END;                /**END LOOP 1**/


 ************************************************************************;
 * STEP 7. Merges the 2 cohort files together.
 ************************************************************************;

    /**THIS IS THE PRINCIPAL OUTPUT FILE**/

DATA &OUT1..M1ASCN&YY(LABEL="Y&YY M1 COUNT FILE:M13/M14"
                      SORTEDBY=Y&YY.VARS);
MERGE TEMPA1 TEMPA2;
    BY  Y&YY.FAGEM Y&YY.SEXM
        Y&YY.VARS;
IF Y&YY.CTM13=. THEN Y&YY.CTM13=0;
IF Y&YY.CTM14=. THEN Y&YY.CTM14=0;
RENAME  Y&YY.FAGEM = Y&YY.FAGE
        Y&YY.SEXM  = Y&YY.SEX;

LABEL
Y&YY.VARS = "MAJOR SUBGROUP VARIABLE LEVELS"
Y&YY.CTM13  = "M.1.3 PERSON COUNTS"
Y&YY.CTM14  = "M.1.4 PERSON COUNTS"
Y&YY.FAGEM  = "&YY. AGE GROUPS"
Y&YY.SEXM   = "&YY. SEX"
;
RUN;

 ************************************************************************;
 * STEP 8. Proc contents and proc print on the analytic count file.
 ************************************************************************;

PROC PRINT DATA=&OUT1..M1ASCN&YY NOOBS;
TITLE "Y&YY M.1 COUNT FILE FOR M.1.3 AND M.1.4";
TITLE2 "PRINT FULL M.1 COUNT FILE";
FORMAT Y&YY.CTM13 Y&YY.CTM14 COMMA10.;
RUN;
PROC CONTENTS DATA=&OUT1..M1ASCN&YY DIRECTORY POSITION;
TITLE2;
RUN;


                    /**MACRO ENDS**/
%MEND YXXM1006;
```

PROGRAMS — M2

D#MSQ.@MSQ00M2.PGMLIB92(M2PRE92)

```
//TI36PR92 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE92 EXEC M2PRE,YEAR=92,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301'
```

D#MSQ.@MSQ00M2.PGMLIB93(M2PRE93)

```
//TI36PR93 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE93 EXEC M2PRE,YEAR=93,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR93.SSLS.#010301'
```

D#MSQ.@MSQ00M2.PGMLIB94(M2PRE94)

```
//TI36PR94 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE94 EXEC M2PRE,YEAR=94,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR94.SSLS.#010401'
```

D#MSQ.@MSQ00M2.PGMLIB95(M2PRE95)

```
//TI36PR95 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE95 EXEC M2PRE,YEAR=95,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR95.SSLS.#010401'
```

D#MSQ.@MSQ00M2.PGMLIB96(M2PRE96)

```
//TI36PR96 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE96 EXEC M2PRE,YEAR=96,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR96.SSLS.#010401'
```

D#MSQ.@MSQ00M2.PGMLIB97(M2PRE97)

```
//TI36PR97 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE97 EXEC M2PRE,YEAR=97,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR97.SSLS.#010401'
```

D#MSQ.@MSQ00M2.PGMLIB98(M2PRE98)

```
//TI36PR98 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE98 EXEC M2PRE,YEAR=98,TYPE=A,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR98.SSLS.#010501'
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PRE99)

```
//TI36PR99 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE99 EXEC M2PRE,YEAR=99,TYPE=B,
//    FILEN='TI03.@MSQ0000.D#MSQ.MPAR99.SSLS.#010501'
```

D#MSQ.@MSQ00M2.PGMLIB00(M2PRE00)

```
//TI36PR00 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PRE00 EXEC M2PRE,YEAR=00,TYPE=B,
//    FILEN='TI03.@MSQ0000.MSQ2000.MPAR00.SSLS.#100501'
```

D#MSQ.@MSQ00M2.PGMLIB01(M2PRE01)

```
//TI02PR01 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose: This job executes the cataloged procedure M2PRE for a
//*          particular year (e.g., M2PRE92 for 1992).  This job
//*          specifies three parameters that are passed on to the
//*          M2PRE procedure:  (1) the 2-digit year of analysis
//*          (e.g., 92 for 1992), (2) the file name of the flat
//*          MedPAR file from the DSAF extraction (e.g.,
//*          TI03.@MSQ0000.D#MSQ.MPAR92.SSLS.#010301 for 1992),
//*          and (3) the record length of the MedPAR file (e.g., A
//*          = 500 for 1992 through 1998 and B = 514 for 1999 and
//*          2000).  To run this job for other years, simply
//*          define for that year these three parameters that will
//*          be passed on to M2PRE (the next program), which will
//*          then pass it on to all the subsequent programs.
//*          Programs SYSM2P1 through SYSM2P5 are then run based
//*          on the execution of this M2PRE procedure. Therefore,
//*          all output generated from subsequent programs will
//*          all be in one document
//*          (D#MSQ.@MSQ00M2.JOBLIBXX(M2PREXX).
//*
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB01)
//M2PRE01 EXEC M2PRE,YEAR=01,TYPE=B,
//    FILEN='TI02.@@MS0000.D#MSQ.MPAR01.SSLS.#082302'
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PRE)

```
//M2PRE PROC
//*********************************************************************
//****CATALOGED PROCEDURE FOR M.2 PRE-ANALYTICAL FILE CONSTRUCTION*****
//*********************************************************************
//*
//********ENTER THE FOLLOWING SYMBOLIC VARS WHEN CALLING THIS PROC*****
//*       YEAR  = 2-DIGIT YEAR
//*       FILEN = FILE NAME OF FLAT MEDPAR FILE FROM DSAF EXTRACTION
//*       TYPE  = LRECL TYPE OF MEDPAR FILE: A=500 B=514 BYTES
//*
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* REQ DATE  :06/21/01
//* EXEC DATE :07/11/01
//* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(M2PRE)
//* PGMR      :MATTHEW URATO
//* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
//* INPUTS    :SEE EACH STEP
//* OUTPUTS   :SEE EACH STEP
//* PURPOSE   :PREPARE MEDPAR FILE FOR THE PROPER FILE CONSTRUCTION
//*            TASKS PERFORMED IN PROC M2PROD
//*********************************************************************
//*
//*********************************************************************
//*CREATE FLAT FILE WITH ONLY SELECTED RECORDS: ALL FIELDS
//*CREATE FLAT FILE WITH ONLY SELECTED RECORDS: ID FIELD ONLY
//*********************************************************************
//M2PRE1 EXEC PGM=ICEMAN,REGION=6M,PARM=('CORE=MAX-16K')
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=&FILEN,DISP=SHR
//SORTOUT DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECT&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),VOL=(,,,8)
//SORTOF01 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE1.IDS&YEAR,
//            DISP=(NEW,CATLG,DELETE),
//            SPACE=(CYL,(200,10),RLSE),DCB=(LRECL=14,RECFM=FB)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P1&TYPE),DISP=SHR
//*
//*********************************************************************
//*CREATE A SAS FILE WITH JUST IDS AND OBSERVATION NUMBER
//*********************************************************************
//M2PRE2 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//IN1 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE1.IDS&YEAR,
//       DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMP&YEAR,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(350,30),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P2),DISP=SHR
//*
//*********************************************************************
//*CREATE FIRST MAJOR M.2 MEDPAR SAS FILE: ALL FIELDS
//*********************************************************************
//M2PRE3 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
```

```
//       'SYSPARM=&YEAR'),COND=(4,LT)
//IN1 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECT&YEAR,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR&YEAR,
```


```
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),VOL=(,,,8)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3&TYPE),DISP=SHR
//*
//*******************************************************************
//*CHECK NEW SAS FILE FOR OUT-OF-RANGE DIAGNOSIS CODES
//*******************************************************************
//M2PRE4 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//     'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SDGFORM),DISP=SHR
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR&YEAR,DISP=SHR
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P4),DISP=SHR
//*
//*******************************************************************
//*CHECK NEW SAS FILE FOR OUT-OF-RANGE PROCEDURE CODES
//*******************************************************************
//M2PRE5 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//     'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SPGFORM),DISP=SHR
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR&YEAR,DISP=SHR
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P5),DISP=SHR
//*
//*******************************************************************
//*CREATE UNIQUE ID FILE FOR DSAF XREF PROCESSING
//*******************************************************************
//M2PRE6 EXEC PGM=ICEMAN,REGION=6M,PARM=('CORE=MAX-16K'),COND=(4,LT)
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE1.IDS&YEAR,
//        DISP=SHR
//SORTOUT DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE6.XREF&YEAR,
//        DISP=(NEW,CATLG,DELETE),DCB=(LRECL=14,RECFM=FB),
//        SPACE=(CYL,(200,10),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P6),DISP=SHR
//*
//*******************************************************************
//*CREATE UNIQUE ID FILES FOR DSAF: 1 MILLION RECORDS EACH
//*******************************************************************
//M2PRE7 EXEC PGM=ICEMAN,REGION=6M,PARM=('CORE=MAX-16K'),COND=(4,LT)
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE6.XREF&YEAR,
//        DISP=SHR
//SORTOF1 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.01,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF2 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.02,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF3 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.03,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF4 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.04,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
```

```
//SORTOF5 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.05,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF6 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.06,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
```


```
//SORTOF7 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.07,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF8 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.08,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SORTOF9 DD DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREF&YEAR.09,
//        DISP=(NEW,CATLG,DELETE),
//        SPACE=(CYL,(16,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P7),DISP=SHR
//*****************************************************************
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P1A)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P1A)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :TI03.@MSQ0000.D#MSQ.MPARXX
*                              - Y92-Y98 MEDPAR FILE
* OUTPUTS   :D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECTXX (flat file
*                 with selected MedPAR records)
*            D#MSQ.@MSQ0000.FLAT.M2PRE1.IDSXX
*                               - (IDs from the MedPAR file)
* PURPOSE   :SELECT ONLY NEEDED RECORDS FROM 500 BYTE FLAT FILE
*            AND CREATE 2 FLAT FILES:
*            ONE WITH ALL FIELDS AND ONE WITH JUST IDS.
*********************************************************************

*********************************************************************;
* STEP 1. Reads in the flat MedPAR file and selects
*         short-stay acute care hospital claims (MedPAR
*         provider number special unit code = missing and
*         MedPAR provider number third position code = 0).
*********************************************************************;

  INCLUDE COND=(43,1,CH,EQ,C' ',AND,39,1,CH,EQ,C'0')

*********************************************************************;
* STEP 2. Sorts selected records (record is read in as one string)
*         and deletes duplicate records.
*********************************************************************;

  SORT FIELDS=(1,500,CH,A)
  SUM FIELDS=NONE

*********************************************************************;
* STEP 3. Writes out two flat files.
*********************************************************************;

  OUTFIL FNAMES=SORTOUT
  OUTFIL FNAMES=SORTOF01,OUTREC=(1:1,11,12:3X)
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P1B)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P1B)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :TI03.@MSQ0000.D#MSQ.MPAR99 - Y99 MEDPAR FILE
*         OR:TI03.@MSQ0000.MSQ2000.DENOM00.#10051
*                              - Y00 DENOM FILE
* OUTPUTS   :D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECTXX (flat file
*                 with selected MedPAR records)
*            D#MSQ.@MSQ0000.FLAT.M2PRE1.IDSXX
*                             - (IDs from the MedPAR file)
* PURPOSE   :SELECT ONLY NEEDED RECORDS FROM 514 BYTE MEDPAR FILE
*            AND CREATE 2 FLAT FILES:
*            ONE WITH ALL FIELDS AND ONE WITH JUST IDS.
*********************************************************************

*********************************************************************;
* STEP 1. Reads in the flat MedPAR file and selects
*         short-stay acute care hospital claims (MedPAR
*         provider number special unit code = missing and
*         MedPAR provider number third position code = 0).
*********************************************************************;

  INCLUDE COND=(43,1,CH,EQ,C' ',AND,39,1,CH,EQ,C'0')

*********************************************************************;
* STEP 2. Sorts selected records (record is read in as one string)
*         and deletes duplicate records.
*********************************************************************;

  SORT FIELDS=(1,514,CH,A)
  SUM FIELDS=NONE

*********************************************************************;
* STEP 3. Writes out two flat files.
*********************************************************************;

  OUTFIL FNAMES=SORTOUT
  OUTFIL FNAMES=SORTOF01,OUTREC=(1:1,11,12:3X)
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P2)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P2)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.FLAT.M2PRE1.IDSXX
*                              - (IDs from the MedPAR file)
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMPXX
*                              - (IDs from the MedPAR file)
* PURPOSE   :CREATE A SAS ID FILE WITH ASSOCIATED
*            OBSERVATION NUMBER.
*********************************************************************;

 ********************************************************************;
 * STEP 1. Reads in the flat MedPAR ID file, names the IDs
 *         XREF (referring to an historical ID that a beneficiary
 *         has at that point in time), assigns each record an
 *         observation number (NOBS), and writes out a SAS file.
 ********************************************************************;

DATA OUT1.XREFMP&SYSPARM
           (LABEL="MEDPAR ID FILE FOR Y&SYSPARM"
            SORTEDBY=NOBS);
INFILE IN1 MISSOVER ;
INPUT @1 XREF $CHAR11.;
NOBS=_N_;
RUN;

 ********************************************************************;
 * STEP 2. Proc contents and proc print on the SAS ID files.
 ********************************************************************;

PROC CONTENTS DATA=OUT1.XREFMP&SYSPARM DIRECTORY;
RUN;
PROC PRINT DATA=OUT1.XREFMP&SYSPARM(OBS=5);
TITLE "PRINT OUT 5 OBS FOR Y&SYSPARM MEDPAR ID FILE";
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECTXX - (92-98 flat files
*                  with selected MedPAR records)
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX (92-98 SAS files
*                  with selected MedPAR records)
* PURPOSE   :CREATE FIRST MAJOR SAS MEDPAR FILE FROM 500 BYTE LAYOUT.
*********************************************************************;

 *********************************************************************;
 * STEP 1. Reads in the flat MedPAR file that contains the
 *         short-stay acute care hospital records and writes
 *         out a SAS file.
 *********************************************************************;

DATA OUT1.MEDPAR&SYSPARM
        (LABEL="FIRST SAS Y&SYSPARM MEDPAR FILE"
         SORTEDBY = NOBS);
INFILE IN1 MISSOVER;
DROP FIL1 FIL2 FIL3 FIL4 FIL10 SSI;
INPUT
@1          XREF       $CHAR11.
@12         AGE        3.
@15         SEX        $CHAR1.
@16         RACE       $CHAR1.
@17         MSCODE     $CHAR2.
@19         BENE_ST    $CHAR2.
@21         BENE_CNT   $CHAR3.
@24         ZIP        $CHAR5.
@29         FILLER     $CHAR4.
@33         ADMDAY     1.
@34         DISSTAT    $CHAR1.
@35         GHOPAID    $CHAR1.
@36         PPSIND     $CHAR1.
@37         PROVID     $CHAR6.
@43         PROVCODE   $CHAR1.
@44         SNFIND     $CHAR1.
@45         NOOFBILL   PD2.
@47         ACDATE     PD3.
@50         FEXDATE    PD3.
@53         FDATE      PD3.
@56         THRUDATE   PD3.
@59         ADMDATE    PD3.
@62         DISDATE    PD3.
@65         CRENDATE   PD3.
@68         DOD        PD3.
@71         DODIND     $CHAR1.
@72         SSI        $CHAR4.
@76         LOS        PD3.
@79         OUTDAYS    PD2.
```

```
@81         COVDAYS    PD2.
@83         COINDAYS   PD2.
@85         LIFERES    PD2.
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A) -- CONTINUED

```
@87         FIL1       PD4.
@91         FIL2       PD4.
@95         FIL3       PD4.
@99         COINAMT    PD4.
@103        INPDED     PD4.
@107        BLOODED    PD4.
@111        PRIMAM     PD4.
@115        OUTAM      PD4.
@119        DISPROP    PD4.
@123        IMEAM      PD4.
@127        DRGPRICE   PD4.
@131        TOTPER     PD4.
@135        CAPACC     PD4.
@139        TOTDIEM    PD4.
@143        COSTIME    PD4.
@147        ACQU       PD4.
@151        CHARGE     PD4.
@155        TOT_CHRG   PD4.
@159        PMT_AMT    PD4.
@163        TOTACC     PD4.
@167        TOTDPT     PD4.
@171        (ACDAYS1-ACDAYS5)   (PD2.)
@181        (ACCHRG1-ACCHRG5)   (PD4.)
@201        (SERCHG1-SERCHG22)  (PD4.)
@289        ORGCHRG              PD4.
@293        (SERCHG24-SERCHG25) (PD4.)
@301        ACCSRIND   $CHAR23.
@324        DIAGCNT    $CHAR2.
@326        (DIAG1-DIAG10)      ($CHAR5.)
@376        SURGIND    $CHAR1.
@377        SURGCNT    2.
@379        SURGDAYS   2.
@381        (SPROC1-SPROC10)    ($CHAR4.)
@421        (SDATE1-SDATE10)    (PD3.)
@451        BLOODPIN   PD2.
@453        FIL10      $CHAR2.
@455        DRGCODE    3.
@458        DIS_DEST   2.
@460        OUTCODE    $CHAR1.
@461        PRIMPYR    $CHAR1.
@462        ESRD       2.
@464        ADMSOUR    $CHAR1.
@465        TYPEADM    $CHAR1.
@466        INTMD      $CHAR5.
@471        DIAGCODE   $CHAR5.
@476        HMONUM     PD3.
@479        HMOCODE    $CHAR1.
@480        DEATH      PD3.
@483        FIL4       $CHAR4.
@487        IMCAB      3.
@490        DATEDATA   1.
@491        SAMPLES    1.
@492        WARNIN1    IB5.
```

```
@497        WARNIN2    IB4.
;

 /******LABELS******/

 ********************************************************************;
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A) -- CONTINUED
```
 * STEP 2. Creates labels for the variables on the SAS file.
 ********************************************************************;

LABEL

XREF         =     "BENEFICIARY CLAIM#"
AGE          =     "AGE"
SEX          =     "SEX"
RACE         =     "RACE"
MSCODE       =     "STATUS CODE"
BENE_ST      =     "STATE CODE"
BENE_CNT     =     "COUNTY CODE"
ZIP          =     "ZIP CODE"
FILLER       =     "FILLER"
ADMDAY       =     "ADMISSION DAY CODE"
DISSTAT      =     "DISCHARGE STATUS CODE"
GHOPAID      =     "GHO PAID CODE"
PPSIND       =     "PPS INDICATOR CODE"
PROVID       =     "PROVIDER ID"
PROVCODE     =     "PROVIDER UNIT CODE"
SNFIND       =     "SNF/SHORT/LONG STAY"
NOOFBILL     =     "CLAIMS RECORD COUNT"
ACDATE       =     "CLAIM ACCRETION DATE"
FEXDATE      =     "LAST DATE OF COVERAGE"
FDATE        =     "SNF 3DAY FROMDATE"
THRUDATE     =     "SNF 3DAY THRUDATE"
ADMDATE      =     "INPT ADMIT DATE"
DISDATE      =     "INPT DISCHARGE DATE"
CRENDATE     =     "SNF COVERED THRUDATE"
DOD          =     "DATE OF DEATH"
DODIND       =     "DEATH VERIFICATION CODE"
SSI          =     "INTERNAL SSI"
LOS          =     "HOSP/SNF LOS"
OUTDAYS      =     "OUTLIER DAY COUNT"
COVDAYS      =     "UTILIZATION DAYCOUNT"
COINDAYS     =     "COINSURANCE DAYCOUNT"
LIFERES      =     "LIFETIME RESERVE DAYS"
FIL1         =     "FILLER"
FIL2         =     "FILLER"
FIL3         =     "FILLER"
COINAMT      =     "PART_A COINSURANCE AMT"
INPDED       =     "INPT DEDUCTIBLE AMT"
BLOODED      =     "BLOOD DEDUCTIBLE AMT"
PRIMAM       =     "PRIMARY PAYER AMT"
OUTAM        =     "DRG OUTLIER AMT"
DISPROP      =     "INPT DISSHARE AMT"
IMEAM        =     "IME AMT"
DRGPRICE     =     "DRG PRICE AMT"
TOTPER       =     "PASSTHRU AMT"
CAPACC       =     "PPS CAPITAL AMT"
```

```
TOTDIEM      =     "TOTAL PER DIEM AMOUNT"
COSTIME      =     "COSTIME"
ACQU         =     "ACQU"
CHARGE       =     "TOTAL CHARGE AMT"
TOT_CHRG     =     "TOTAL COVERED CHARGE"
PMT_AMT      =     "MEDICARE PAYMENT AMT"
TOTACC       =     "ALL ACCOMOD TOTAL CHARGE"
TOTDPT       =     "DEPARTMENTAL TOTAL CHARGE"
ACDAYS1      =     "PRIVATE ROOM DAYCOUNT"
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A) -- CONTINUED

```
ACDAYS2      =     "SEMIPRIV ROOM DAYCOUNT"
ACDAYS3      =     "WARD DAYCOUNT"
ACDAYS4      =     "INTENSIVE CARE DAYCOUNT"
ACDAYS5      =     "CORONARY CARE DAYCOUNT"
ACCHRG1      =     "PRIVATE ROOM CHARGE"
ACCHRG2      =     "SEMIPRIV ROON CHARGE"
ACCHRG3      =     "WARD CHARGE"
ACCHRG4      =     "INTENSIVE CARE CHARGE"
ACCHRG5      =     "CORONARY CARE CHARGE"
SERCHG1      =     "OTHER SERVICES CHRGAMT"
SERCHG2      =     "PHARMACY CHRGAMT"
SERCHG3      =     "MED/SURG CHRGAMT"
SERCHG4      =     "DME CHRGAMT"
SERCHG5      =     "USED DME CHRGAMT"
SERCHG6      =     "PHYSTHERAPY CHRGAMT"
SERCHG7      =     "OCCUTHERAPY CHRGAMT"
SERCHG8      =     "SPEECHPATH CHRGAMT"
SERCHG9      =     "INHALATION CHRGAMT"
SERCHG10     =     "BLOOD CHRGAMT"
SERCHG11     =     "BLOOD ADMIN CHRGAMT"
SERCHG12     =     "OPERATING_RM CHRGAMT"
SERCHG13     =     "LITHORISPY CHRGAMT"
SERCHG14     =     "CARDIOLOGY CHRGAMT"
SERCHG15     =     "ANESTHESIA CHRGAMT"
SERCHG16     =     "LABOROTORY CHRGAMT"
SERCHG17     =     "RADIOLOGY CHRGAMT"
SERCHG18     =     "MRI CHRGAMT"
SERCHG19     =     "OUTPATIENT CHRGAMT"
SERCHG20     =     "EMERGENCY CHRGAMT"
SERCHG21     =     "AMBULANCE CHRGAMT"
SERCHG22     =     "PROFESS_FEES CHGRAMT"
ORGCHRG      =     "ORGAN_AQUISIT CHRGAMT"
SERCHG24     =     "REVENUE SETTING CHRGAMT"
SERCHG25     =     "CLINIC VISIT CHRGAMT"
ACCSRIND     =     "INDICATOR CODES"
DIAGCNT      =     "DIAGNOSIS CODE COUNT"
DIAG1        =     "DIAGNOSIS CODE_1"
DIAG2        =     "DIAGNOSIS CODE_2"
DIAG3        =     "DIAGNOSIS CODE_3"
DIAG4        =     "DIAGNOSIS CODE_4"
DIAG5        =     "DIAGNOSIS CODE_5"
DIAG6        =     "DIAGNOSIS CODE_6"
DIAG7        =     "DIAGNOSIS CODE_7"
DIAG8        =     "DIAGNOSIS CODE_8"
DIAG9        =     "DIAGNOSIS CODE_9"
DIAG10       =     "DIAGNOSIS CODE_10"
SURGIND      =     "SURGICAL INDICATOR"
```

```
SURGCNT      =     "SURGICAL PROCCODES COUNT"
SURGDAYS     =     "SURGICAL PROCDATE COUNT"
SPROC1       =     "SURGICAL PROC_1"
SPROC2       =     "SURGICAL PROC_2"
SPROC3       =     "SURGICAL PROC_3"
SPROC4       =     "SURGICAL PROC_4"
SPROC5       =     "SURGICAL PROC_5"
SPROC6       =     "SURGICAL PROC_6"
SPROC7       =     "SURGICAL PROC_7"
SPROC8       =     "SURGICAL PROC_8"
SPROC9       =     "SURGICAL PROC_9"
SPROC10      =     "SURGICAL PROC_10"
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3A) -- CONTINUED

```
SDATE1       =     "SURGICAL PROCDATE_1"
SDATE2       =     "SURGICAL PROCDATE_2"
SDATE3       =     "SURGICAL PROCDATE_3"
SDATE4       =     "SURGICAL PROCDATE_4"
SDATE5       =     "SURGICAL PROCDATE_5"
SDATE6       =     "SURGICAL PROCDATE_6"
SDATE7       =     "SURGICAL PROCDATE_7"
SDATE8       =     "SURGICAL PROCDATE_8"
SDATE9       =     "SURGICAL PROCDATE_9"
SDATE10      =     "SURGICAL PROCDATE_10"
BLOODPIN     =     "BLOOD PINTS"
FIL10        =     "BIC"
DRGCODE      =     "DRG_CODE"
DIS_DEST     =     "DISCHARGE DESTINATION CODE"
OUTCODE      =     "DRG/OUTLIER STAY CODE"
PRIMPYR      =     "PRIMARY PAYER CODE"
ESRD         =     "ESRD CODE"
ADMSOUR      =     "SOURCE INPT_ADMIT CODE"
TYPEADM      =     "INPT ADMIT_TYPE CODE"
INTMD        =     "INTERMEDIARY_ID"
DIAGCODE     =     "ICD_9 ADMISSION DIAGNOSIS"
HMONUM       =     "HMONUM"
HMOCODE      =     "HMOCODE"
DEATH        =     "ADMISSION DEATH DAYCOUNT"
FIL4         =     "FILLER"
IMCAB        =     "INTERNAL_USE CODE"
DATEDATA     =     "INTERNAL_DATE CODE"
SAMPLES      =     "INTERNAL SAMPLE_SIZE CODE"
WARNIN1      =     "WARNING CODE1"
WARNIN2      =     "WARNING CODE2"
NOBS         =     "ORIGINAL MEDPAR OBS NUMBER"
;

 ********************************************************************;
 * STEP 3. Creates an observation number (NOBS) that mirrors
 *         that in the MedPAR ID file.
 ********************************************************************;

NOBS=_N_;
RUN;

 ********************************************************************;
 * STEP 4. Proc contents and proc print on the SAS MedPAR files.
 ********************************************************************;
```

```
PROC PRINT  DATA=OUT1.MEDPAR&SYSPARM(OBS=10);
TITLE "MEDPAR Y&SYSPARM FIRST SAS FILE";
RUN;
PROC CONTENTS DATA=OUT1.MEDPAR&SYSPARM DIRECTORY;
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.FLAT.M2PRE1.SELECTXX - (99-00 flat files
*                  with selected MedPAR records)
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX (99-00 SAS files
*                  with selected MedPAR records)
* PURPOSE   :CREATE FIRST MAJOR SAS MEDPAR FILE FROM 514 BYTE LAYOUT.
*********************************************************************;

 *********************************************************************;
 * STEP 1. Reads in the flat MedPAR file that contains the
 *         short-stay acute care hospital records and writes
 *         out a SAS file.
 *********************************************************************;

DATA OUT1.MEDPAR&SYSPARM
         (LABEL="FIRST SAS Y&SYSPARM MEDPAR FILE"
          SORTEDBY = NOBS);
INFILE IN1 MISSOVER;
DROP FIL1 FIL2 FIL3 FIL4 FIL10 SSI;
INPUT
@1          XREF       $CHAR11.
@12         AGE        3.
@15         SEX        $CHAR1.
@16         RACE       $CHAR1.
@17         MSCODE     $CHAR2.
@19         BENE_ST    $CHAR2.
@21         BENE_CNT   $CHAR3.
@24         ZIP        $CHAR5.
@29         FILLER     $CHAR4.
@33         ADMDAY     1.
@34         DISSTAT    $CHAR1.
@35         GHOPAID    $CHAR1.
@36         PPSIND     $CHAR1.
@37         PROVID     $CHAR6.
@43         PROVCODE   $CHAR1.
@44         SNFIND     $CHAR1.
@45         NOOFBILL   PD2.
@47         ACDATE     PD4.
@51         FEXDATE    PD4.
@55         FDATE      PD4.
@59         THRUDATE   PD4.
@63         ADMDATE    PD4.
@67         DISDATE    PD4.
@71         CRENDATE   PD4.
@75         DOD        PD4.
@79         DODIND     $CHAR1.
@80         SSI        $CHAR4.
@84         LOS        PD3.
@87         OUTDAYS    PD2.
```

```
@89         COVDAYS    PD2.
@91         COINDAYS   PD2.
@93         LIFERES    PD2.
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B) -- CONTINUED

```
@95         FIL1       PD4.
@99         FIL2       PD4.
@103        FIL3       PD4.
@107        COINAMT    PD4.
@111        INPDED     PD4.
@115        BLOODED    PD4.
@119        PRIMAM     PD4.
@123        OUTAM      PD4.
@127        DISPROP    PD4.
@131        IMEAM      PD4.
@135        DRGPRICE   PD4.
@139        TOTPER     PD4.
@143        CAPACC     PD4.
@147        TOTDIEM    PD4.
@151        COSTIME    PD4.
@155        ACQU       PD4.
@159        CHARGE     PD4.
@163        TOT_CHRG   PD4.
@167        PMT_AMT    PD4.
@171        TOTACC     PD4.
@175        TOTDPT     PD4.
@179        (ACDAYS1-ACDAYS5)   (PD2.)
@189        (ACCHRG1-ACCHRG5)   (PD4.)
@209        (SERCHG1-SERCHG25)  (PD4.)
@309        ACCSRIND   $CHAR23.
@332        DIAGCNT    $CHAR2.
@334        (DIAG1-DIAG10)      ($CHAR6.)
@394        SURGIND    $CHAR1.
@395        SURGCNT    2.
@397        SURGDAYS   2.
@399        (SPROC1-SPROC6)    ($CHAR7.)
@441        (SDATE1-SDATE6)    (PD4.)
@465        BLOODPIN   PD2.
@467        FIL10      $CHAR2.
@469        DRGCODE    3.
@472        DIS_DEST   2.
@474        OUTCODE    $CHAR1.
@475        PRIMPYR    $CHAR1.
@476        ESRD       2.
@478        ADMSOUR    $CHAR1.
@479        TYPEADM    $CHAR1.
@480        INTMD      $CHAR5.
@485        DIAGCODE   $CHAR5.
@490        HMONUM     PD3.
@493        HMOCODE    $CHAR1.
@494        DEATH      PD3.
@497        FIL4       $CHAR4.
@501        IMCAB      3.
@504        DATEDATA   1.
@505        SAMPLES    1.
@506        WARNIN1    IB5.
@511        WARNIN2    IB4.
  ;
```

```
 /******LABELS******/

*******************************************************************;
* STEP 2. Creates labels for the variables on the SAS file.
*******************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B) -- CONTINUED

```
LABEL

XREF         =     "BENEFICIARY CLAIM#"
AGE          =     "AGE"
SEX          =     "SEX"
RACE         =     "RACE"
MSCODE       =     "STATUS CODE"
BENE_ST      =     "STATE CODE"
BENE_CNT     =     "COUNTY CODE"
ZIP          =     "ZIP CODE"
FILLER       =     "FILLER"
ADMDAY       =     "ADMISSION DAY CODE"
DISSTAT      =     "DISCHARGE STATUS CODE"
GHOPAID      =     "GHO PAID CODE"
PPSIND       =     "PPS INDICATOR CODE"
PROVID       =     "PROVIDER ID"
PROVCODE     =     "PROVIDER UNIT CODE"
SNFIND       =     "SNF/SHORT/LONG STAY"
NOOFBILL     =     "CLAIMS RECORD COUNT"
ACDATE       =     "CLAIM ACCRETION DATE"
FEXDATE      =     "LAST DATE OF COVERAGE"
FDATE        =     "SNF 3DAY FROMDATE"
THRUDATE     =     "SNF 3DAY THRUDATE"
ADMDATE      =     "INPT ADMIT DATE"
DISDATE      =     "INPT DISCHARGE DATE"
CRENDATE     =     "SNF COVERED THRUDATE"
DOD          =     "DATE OF DEATH"
DODIND       =     "DEATH VERIFICATION CODE"
SSI          =     "INTERNAL SSI"
LOS          =     "HOSP/SNF LOS"
OUTDAYS      =     "OUTLIER DAY COUNT"
COVDAYS      =     "UTILIZATION DAYCOUNT"
COINDAYS     =     "COINSURANCE DAYCOUNT"
LIFERES      =     "LIFETIME RESERVE DAYS"
FIL1         =     "FILLER"
FIL2         =     "FILLER"
FIL3         =     "FILLER"
COINAMT      =     "PART_A COINSURANCE AMT"
INPDED       =     "INPT DEDUCTIBLE AMT"
BLOODED      =     "BLOOD DEDUCTIBLE AMT"
PRIMAM       =     "PRIMARY PAYER AMT"
OUTAM        =     "DRG OUTLIER AMT"
DISPROP      =     "INPT DISSHARE AMT"
IMEAM        =     "IME AMT"
DRGPRICE     =     "DRG PRICE AMT"
TOTPER       =     "PASSTHRU AMT"
CAPACC       =     "PPS CAPITAL AMT"
TOTDIEM      =     "TOTDIEM"
COSTIME      =     "COSTIME"
```

```
ACQU         =     "ACQU"
CHARGE       =     "TOTAL CHARGE AMT"
TOT_CHRG     =     "TOTAL COVERED CHARGE"
PMT_AMT      =     "MEDICARE PAYMENT AMT"
TOTACC       =     "ALL ACCOMOD TOTAL CHARGE"
TOTDPT       =     "DEPARTMENTAL TOTAL CHARGE"
ACDAYS1      =     "PRIVATE ROOM DAYCOUNT"
ACDAYS2      =     "SEMIPRIV ROOM DAYCOUNT"
ACDAYS3      =     "WARD DAYCOUNT"
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B) -- CONTINUED

```
ACDAYS4      =     "INTENSIVE CARE DAYCOUNT"
ACDAYS5      =     "CORONARY CARE DAYCOUNT"
ACCHRG1      =     "PRIVATE ROOM CHARGE"
ACCHRG2      =     "SEMIPRIV ROON CHARGE"
ACCHRG3      =     "WARD CHARGE"
ACCHRG4      =     "INTENSIVE CARE CHARGE"
ACCHRG5      =     "CORONARY CARE CHARGE"
SERCHG1      =     "OTHER SERVICES CHRGAMT"
SERCHG2      =     "PHARMACY CHRGAMT"
SERCHG3      =     "MED/SURG CHRGAMT"
SERCHG4      =     "DME CHRGAMT"
SERCHG5      =     "USED DME CHRGAMT"
SERCHG6      =     "PHYSTHERAPY CHRGAMT"
SERCHG7      =     "OCCUTHERAPY CHRGAMT"
SERCHG8      =     "SPEECHPATH CHRGAMT"
SERCHG9      =     "INHALATION CHRGAMT"
SERCHG10     =     "BLOOD CHRGAMT"
SERCHG11     =     "BLOOD ADMIN CHRGAMT"
SERCHG12     =     "OPERATING_RM CHRGAMT"
SERCHG13     =     "LITHORISPY CHRGAMT"
SERCHG14     =     "CARDIOLOGY CHRGAMT"
SERCHG15     =     "ANESTHESIA CHRGAMT"
SERCHG16     =     "LABOROTORY CHRGAMT"
SERCHG17     =     "RADIOLOGY CHRGAMT"
SERCHG18     =     "MRI CHRGAMT"
SERCHG19     =     "OUTPATIENT CHRGAMT"
SERCHG20     =     "EMERGENCY CHRGAMT"
SERCHG21     =     "AMBULANCE CHRGAMT"
SERCHG22     =     "PROFESS_FEES CHGRAMT"
SERCHG23     =     "ORGAN_AQUISIT CHRGAMT"
SERCHG24     =     "REVENUE SETTING CHRGAMT"
SERCHG25     =     "CLINIC VISIT CHRGAMT"
ACCSRIND     =     "INDICATOR CODES"
DIAGCNT      =     "DIAGNOSIS CODE COUNT"
DIAG1        =     "DIAGNOSIS CODE_1"
DIAG2        =     "DIAGNOSIS CODE_2"
DIAG3        =     "DIAGNOSIS CODE_3"
DIAG4        =     "DIAGNOSIS CODE_4"
DIAG5        =     "DIAGNOSIS CODE_5"
DIAG6        =     "DIAGNOSIS CODE_6"
DIAG7        =     "DIAGNOSIS CODE_7"
DIAG8        =     "DIAGNOSIS CODE_8"
DIAG9        =     "DIAGNOSIS CODE_9"
DIAG10       =     "DIAGNOSIS CODE_10"
SURGIND      =     "SURGICAL INDICATOR"
SURGCNT      =     "SURGICAL PROCCODES COUNT"
SURGDAYS     =     "SURGICAL PROCDATE COUNT"
```

```
SPROC1       =     "SURGICAL PROC_1"
SPROC2       =     "SURGICAL PROC_2"
SPROC3       =     "SURGICAL PROC_3"
SPROC4       =     "SURGICAL PROC_4"
SPROC5       =     "SURGICAL PROC_5"
SPROC6       =     "SURGICAL PROC_6"
SDATE1       =     "SURGICAL PROCDATE_1"
SDATE2       =     "SURGICAL PROCDATE_2"
SDATE3       =     "SURGICAL PROCDATE_3"
SDATE4       =     "SURGICAL PROCDATE_4"
SDATE5       =     "SURGICAL PROCDATE_5"
SDATE6       =     "SURGICAL PROCDATE_6"
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P3B) -- CONTINUED

```
BLOODPIN     =     "BLOOD PINTS"
FIL10        =     "BIC"
DRGCODE      =     "DRG_CODE"
DIS_DEST     =     "DISCHARGE DESTINATION CODE"
OUTCODE      =     "DRG/OUTLIER STAY CODE"
PRIMPYR      =     "PRIMARY PAYER CODE"
ESRD         =     "ESRD CODE"
ADMSOUR      =     "SOURCE INPT_ADMIT CODE"
TYPEADM      =     "INPT ADMIT_TYPE CODE"
INTMD        =     "INTERMEDIARY_ID"
DIAGCODE     =     "ICD_9 ADMISSION DIAGNOSIS"
HMONUM       =     "HMONUM"
HMOCODE      =     "HMOCODE"
DEATH        =     "ADMISSION DEATH DAYCOUNT"
FIL4         =     "FILLER"
IMCAB        =     "INTERNAL_USE CODE"
DATEDATA     =     "INTERNAL_DATE CODE"
SAMPLES      =     "INTERNAL SAMPLE_SIZE CODE"
WARNIN1      =     "WARNING CODE1"
WARNIN2      =     "WARNING CODE2"
NOBS         =     "ORIGINAL MEDPAR OBS NUMBER"
;

 ********************************************************************;
 * STEP 3. Creates an observation number (NOBS) that mirrors
 *         that in the MedPAR ID file.
 ********************************************************************;

NOBS=_N_;
RUN;

 ********************************************************************;
 * STEP 4. Proc contents and proc print on the SAS MedPAR files.
 ********************************************************************;

PROC PRINT  DATA=OUT1.MEDPAR&SYSPARM(OBS=10);
TITLE "MEDPAR Y&SYSPARM FIRST SAS FILE";
RUN;
PROC CONTENTS DATA=OUT1.MEDPAR&SYSPARM DIRECTORY;
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P4)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P4)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (SDGFORM) - SDG Format
             D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX - SAS MedPAR File
* OUTPUTS   :None
* PURPOSE   :CHECK FOR OUT-OF-RANGE DIAGNOSIS CODES.
*********************************************************************;

%INCLUDE IN0/SOURCE2;

 *********************************************************************;
 * STEP 1. Temporary data step that creates two output files -
 *         YXXD1 for primary diagnoses that do not match to an SDG and
 *         YXXD2 for any secondary diagnoses that do not match to an
 *         SDG.
 *********************************************************************;

DATA Y&SYSPARM.D1(KEEP=DIAG1)
     Y&SYSPARM.D2(KEEP=SECNDIAG
                  RENAME=(SECNDIAG=DIAG2))
;

 *********************************************************************;
 * STEP 2. Read in the SAS MedPAR file.
 *********************************************************************;

SET IN1.MEDPAR&SYSPARM
              (KEEP=PROVID PROVCODE DIAGCNT DIAG1-DIAG10);


 *********************************************************************;
 * STEP 3. Selects the number of diagnosis fields that have
 *         values (e.g., if a record only has values in diag1 - diag3,
 *         then only three fields are checked when mapped to the SDG
 *         format).
 *********************************************************************;

ARRAY DIAGS(10) DIAG1-DIAG10;

DO I = 1 TO INPUT(DIAGCNT,2.);

 *********************************************************************;
 * STEP 4. Formats the diagnoses into the SDG format.  If the
 *         diagnosis did not match, then output non-matches to YXXD1
 *         if DIAG1 did not match or YXXD2 if a secondary diagnosis
 *         (DIAG2-DIAG10) did not match.
 *********************************************************************;

SDG = PUT(LEFT(DIAGS(I)),$SDG.);
IF SDG = 'NO' AND DIAGS(I) NE ' ' THEN DO;
```

```
    IF I=1 THEN OUTPUT Y&SYSPARM.D1;
    ELSE DO;
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P4) -- CONTINUED

```
    SECNDIAG=LEFT(DIAGS(I));
    OUTPUT Y&SYSPARM.D2;
    END;
END;

END;

RUN;

TITLE "ICD9 SDG MIS-MATCHES FOR Y&SYSPARM MEDPAR FILE";

 ********************************************************************;
 * STEP 5. Frequencies on the diagnoses in D1 and D2.
 ********************************************************************;

PROC FREQ DATA=Y&SYSPARM.D1;
TABLES DIAG1;
TITLE "PRINCIPAL DIAGNOSIS ONLY";
RUN;

PROC FREQ DATA=Y&SYSPARM.D2;
TABLES DIAG2;
TITLE "SECONDARY DIAGNOSES ONLY";
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P5)

```
*******************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P5)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (SPGFORM) - SPG Format
             D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX - SAS MedPAR File
* OUTPUTS   :None
* PURPOSE   :CHECK FOR OUT-OF-RANGE PROCEDURE CODES.
*******************************************************************;

%INCLUDE IN0/SOURCE2;

 *******************************************************************;
 * STEP 1. Temporary data step that creates an output file for
 *         procedure that do not match to an SPG.
 *******************************************************************;

DATA Y&SYSPARM.P(KEEP=SURGPROC);

 *******************************************************************;
 * STEP 2. Read in the SAS MedPAR file.
 *******************************************************************;

SET IN1.MEDPAR&SYSPARM
         (KEEP=XREF PROVID PROVCODE SURGIND
               SURGCNT SPROC1-SPROC6) ;

 *******************************************************************;
 * STEP 3. Selects the number of surgical procedure fields that have
 *         values (e.g., if a record only has values in sproc1 -
 *         sproc3, then only three fields are checked when mapped to
 *         the SPG format).
 *******************************************************************;

ARRAY SURGP(6) SPROC1-SPROC6;

DO I = 1 TO 6;

 *******************************************************************;
 * STEP 4. Maps the procedure codes into the SPG format.  If the
 *         procedure code did not match, then output non-matches to
 *         YXXP.
 *******************************************************************;

SPG = PUT(SURGP(I),$SPG.);

IF SPG = 'NOT' AND SURGP(I) NE ' ' THEN DO;
    SURGPROC=SURGP(I);
    OUTPUT;
END;

END;
```

```
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P5) -- CONTINUED

```
TITLE "ICD9 SPG MIS-MATCHES FOR Y&SYSPARM MEDPAR FILE";

 ********************************************************************;
 * STEP 5. Frequencies on the procedures in YXXP.                   *;
 ********************************************************************;

PROC FREQ DATA=Y&SYSPARM.P;
TABLES SURGPROC;
TITLE2 "PROCEDURE CODES ONLY";
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P6)

```
*********************************************************************
* PROJ      :MQMS
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* REQ BY    :TEAM MQMS
* REQ DATE  :02/11/2002
* EXEC DATE :02/11/2002
* PGM       :D#MSQ.@MSQ00M2.PGMLIB92(SYSM2P6)
* PGMR      :MATTHEW URATO
* INPUTS    :SORTIN:D#MSQ.@MSQ0000.FLAT.M2PRE1.IDSXX - IDs from
*                 the MedPAR file
* OUTPUTS   :SORTOUT:D#MSQ.@MSQ0000.FLAT.M2PRE6.XREFXX -
*                 ID file to be used for cross-referencing
* PURPOSE   :GENERATE MEDPAR UNIQUE ID FLAT FILE FOR DSAF XREF
*            PROCESSING.
*********************************************************************

*********************************************************************;
* STEP 1. Sort by beneficiary ID (first 11 characters).
*********************************************************************;

   SORT FIELDS=(1,11,CH,A)

*********************************************************************;
* STEP 2. Resolve duplicates to one record.
*********************************************************************;

   SUM FIELDS=NONE
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM2P7)

```
*********************************************************************
* PROJ      :MQMS
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* REQ BY    :TEAM MQMS
* REQ DATE  :02/11/2002
* EXEC DATE :02/11/2002
* PGM       :D#MSQ.@MSQ00M2.PGMLIB92(SYSM2P7)
* PGMR      :MATTHEW URATO
* INPUTS    :SORTIN:D#MSQ.@MSQ0000.FLAT.M2PRE6.XREFXX - ID file
*            to be used for cross-referencing)
* OUTPUTS   :SORTOF:DSN=D#MSQ.@MSQ0000.FLAT.M2PRE7.XREFXX.01--09 -
*            9 1 million record length ID files
* PURPOSE   :COPY MEDPAR UNIQUE ID FLAT FILE FOR DSAF XREF
*            PROCESSING INTO 1 MILLION RECORD FILES.
*********************************************************************

*********************************************************************;
* STEP 1. Copy IDs from the unique ID file into 1 million
*         record files .
*********************************************************************;

   SORT FIELDS=COPY
   OUTFIL FNAMES=SORTOF1,ENDREC=1000000
   OUTFIL FNAMES=SORTOF2,STARTREC=1000001,ENDREC=2000000
   OUTFIL FNAMES=SORTOF3,STARTREC=2000001,ENDREC=3000000
   OUTFIL FNAMES=SORTOF4,STARTREC=3000001,ENDREC=4000000
   OUTFIL FNAMES=SORTOF5,STARTREC=4000001,ENDREC=5000000
   OUTFIL FNAMES=SORTOF6,STARTREC=5000001,ENDREC=6000000
   OUTFIL FNAMES=SORTOF7,STARTREC=6000001,ENDREC=7000000
   OUTFIL FNAMES=SORTOF8,STARTREC=7000001,ENDREC=8000000
   OUTFIL FNAMES=SORTOF9,STARTREC=8000001,ENDREC=9000000
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD92)

```
//TI36M292 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD92 EXEC M2PROD,YEAR=92
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD93)

```
//TI36M293 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD93 EXEC M2PROD,YEAR=93
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD94)

```
//TI36M294 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD94 EXEC M2PROD,YEAR=94
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD95)

```
//TI36M295 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD95 EXEC M2PROD,YEAR=95
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD96)

```
//TI36M296 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD96 EXEC M2PROD,YEAR=96
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD97)

```
//TI36M297 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD97 EXEC M2PROD,YEAR=97
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD98)

```
//TI36M298 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD98 EXEC M2PROD,YEAR=98
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD99)

```
//TI36M299 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD99 EXEC M2PROD,YEAR=99
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD00)

```
//TI36M200 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2PROD00 EXEC M2PROD,YEAR=00
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD01)

```
//TI36M201 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=H,
//  RESTART=YXXM2001.SAS
//* Purpose:  This job executes the cataloged procedure
//*           M2PROD for a particular year (e.g., M2PROD92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB01)
//M2PROD01 EXEC M2PROD,YEAR=01
```

D#MSQ.@MSQ00M2.PGMLIB99(M2PROD)

```
//M2PROD PROC
//*********************************************************************
//********CATALOGED PROCEDURE FOR M.2 PRINCIPAL ANALYTIC FILE**********
//*********************************************************************
//********ENTER SYMBOLIC PARM YEAR=(2 DIGIT YEAR)***********************
//*
//********MAKE SURE INCLUDED FILES ARE SET IN FIRST STEP: SEE SYSM200**
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* REQ DATE  :06/21/01
//* EXEC DATE :07/11/01
//* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(M2PROD)
//* PGMR      :MURATO
//* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
//* INPUTS    :SEE SYSIN DD FOR EACH STEP
//* OUTPUTS   :SEE SYSIN DD FOR EACH STEP
//* PURPOSE   :CREATE PRINCIPAL ANALYTIC FILE FOR M2 FOR EACH YEAR
//*********************************************************************
//*
//*M2XREF EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//*       'SYSPARM=&YEAR')
//*WORK DD SPACE=(CYL,(250,20)),DCB=(BLKSIZE=27648)
//*IN1 FILEREF IS ASSIGNED INTERNALLY BY SAS IN SYSM200
//*OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR&YEAR,
//*   DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(250,20),RLSE)
//*SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM200),DISP=SHR
//*********************************************************************
//*YXXM2001 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//*       'SYSPARM=&YEAR'),COND=(4,LT)
//*IN1 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR&YEAR,DISP=SHR
//*IN2 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//*OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2001.MEDPRB&YEAR,
//*        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//*        LABEL=(EXPDT=2005/365),
//*        VOL=(,,,8)
//*OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2001.XMEDB&YEAR,
//*        DISP=(NEW,CATLG,DELETE),
//*        SPACE=(TRK,(50,10),RLSE)
//*SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM201),DISP=SHR
//*********************************************************************
//*YXXM2003 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//*       'SYSPARM=&YEAR'),COND=(4,LT)
//*WORK DD SPACE=(CYL,(600,50)),DCB=(BLKSIZE=27648)
//*IN0 DD DSN=*.YXXM2001.SAS.IN2,DISP=SHR
//*IN1 DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR&YEAR,DISP=SHR
//*IN2 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMP&YEAR,DISP=SHR
//*IN3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2001.MEDPRB&YEAR,DISP=SHR
//*OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2003.MEDPRC&YEAR,
//*        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//*        LABEL=(EXPDT=2005/365),
//*        VOL=(,,,8)
//*SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM203),DISP=SHR
//*********************************************************************
//*YXXM2004 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//*       'SYSPARM=&YEAR'),COND=(4,LT)
```

```
//*WORK DD SPACE=(CYL,(900,100)),DCB=(BLKSIZE=27648)
//*IN0 DD DSN=*.YXXM2003.SAS.IN0,DISP=SHR
//*IN1 DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR&YEAR,DISP=SHR
```
D#MSQ.@MSQ00M2.PGMLIB99(M2PROD) -- CONTINUED

```
//*IN2 DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMP&YEAR,DISP=SHR
//*IN3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2003.MEDPRC&YEAR,DISP=SHR,
//*    UNIT=AFF=IN2
//*IN4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENOMB&YEAR,DISP=SHR
//*OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2004.MEDPRD&YEAR,
//*        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//*        LABEL=(EXPDT=2005/365),
//*        VOL=(,,,8)
//*OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2004.DENOMC&YEAR,
//*        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//*        LABEL=(EXPDT=2005/365),
//*        VOL=(,,,8)
//*OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2004.BADIDS&YEAR,
//*        DISP=(NEW,CATLG,DELETE),
//*        SPACE=(CYL,(25,5),RLSE)
//*SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM204),DISP=SHR
//*******************************************************************
//*YXXM2005 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//*       'SYSPARM=&YEAR'),COND=(4,LT)
//*IN0 DD DSN=*.YXXM2004.SAS.IN0,DISP=SHR
//*IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2004.MEDPRD&YEAR,DISP=SHR
//*WORK1 DD SPACE=(CYL,(400,200))
//*OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2005.MEDPRE&YEAR,
//*        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//*        LABEL=(EXPDT=2005/365),
//*        VOL=(,,,8)
//*OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2005.MARKER&YEAR,
//*        DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(5,1),RLSE)
//*SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM205),DISP=SHR
//*******************************************************************
//YXXM2006 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2005.MEDPRE&YEAR,DISP=SHR
//WORK1 DD SPACE=(CYL,(600,100))
//WORK2 DD SPACE=(CYL,(200,10))
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2006.MEDPRF&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),
//        VOL=(,,,8)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2006.TRANS&YEAR,
//        DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(500,10),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM206),DISP=SHR
//*******************************************************************
//YXXM2007 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(800,100)),DCB=(BLKSIZE=27648)
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2006.MEDPRF&YEAR,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2004.DENOMC&YEAR,DISP=SHR
//IN3 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//WORK1 DD SPACE=(CYL,(800,100))
//WORK2 DD UNIT=MCART
//WORK3 DD UNIT=MCART
```

```
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2007.MEDPRG&YEAR,
//        DISP=(NEW,CATLG,DELETE),UNIT=MCART,
//        LABEL=(EXPDT=2005/365),
//        VOL=(,,,8)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2007.CONTI&YEAR,
//        DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(10,2),RLSE)
D#MSQ.@MSQ00M2.PGMLIB99(M2PROD) -- CONTINUED

//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM207),DISP=SHR
//*******************************************************************
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM200)

```
*****************************************************************
* PROJ      :MQMS
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* REQ BY    :TEAM MQMS
* REQ DATE  :02/11/2002
* EXEC DATE :02/11/2002
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM200)
* PGMR      :MATTHEW URATO
* INPUTS    :SEE THE FILE WITH FILEREF OF IN1
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.M2XREF.FINDRXX (XX=YEAR)
*                     - UNIQUE HICNO/XREF FILE
* PURPOSE   :COMBINE ALL DSAF GENERATED CROSS REFERENCE FILES
*            FOR A GIVEN YEAR AND CONSTRUCT A FILE WITH ALL
*            UNIQUE HICNO AND XREF COMBINATIONS WHEREIN THE FORMER
*            SHALL BE THE UNIQUE ID FOR EACH PERSON IN THE MEDPAR
*            FILE FOR THAT YEAR.
*****************************************************************;

*****************************************************************;
* STEP 1. READ IN A SAS FILENAME STATEMENT THAT REFERENCES
* ALL OF THE DSAF OUTPUT FOR CROSS-REFERENCING FOR A GIVEN YEAR.
*****************************************************************;
FILENAME IN0 "D#MSQ.@MSQ00M2.PGMLIB99(M2XREF&SYSPARM)" WAIT=60;
%INCLUDE IN0 / SOURCE2;

*****************************************************************;
* STEP 2. READS ALL IDS FROM ALL THE DSAF OUTPUT FOR A GIVEN YEAR.
*****************************************************************;
DATA TEMP;
INFILE IN1 MISSOVER ;
INPUT  @1  HICNO    $CHAR11.
       @17 XREF     $CHAR11.;
RUN;

*****************************************************************;
* STEP 3. SORTS THE UNIQUE XREF/HICNO FILE BY HICNO.
*****************************************************************;
PROC SORT DATA=TEMP;
BY HICNO;
RUN;

*****************************************************************;
* STEP 4. CHECKS TO SEE IF HICNO MATCHES TO AN HISTORICAL ID.
*  IF NOT, HICNO IS ADDED AS AN HISTORICAL ID (XREF).
*****************************************************************;
DATA TEMP(KEEP=HICNO XREF);
LENGTH CHECK 3 COUNT 4 ;
SET TEMP END=FINI NOBS=OBSNUM;
BY HICNO;
IF FIRST.HICNO THEN CHECK=0;
CHECK + (HICNO=XREF);
OUTPUT;
IF LAST.HICNO THEN DO;
  IF CHECK=0 THEN DO;
     COUNT+1;
     XREF=HICNO;
```

```
     OUTPUT;
  END;
  IF FINI THEN DO;
D#MSQ.@MSQ00M2.PGMLIB99(SYSM200) -- CONTINUED

      PERCENTAGE = ROUND(((COUNT/OBSNUM)*100),.00001);
      PUT / '*' /
  COUNT 'OUT OF ' OBSNUM 'HICNOS (' PERCENTAGE'PERCENT)' /
  'HAD A HICNO WITH NO IDENTICAL XREF' / '*' /
  ;
  END;
END;
RUN;

*******************************************************************;
* STEP 5. SORT THE CONSOLIDATED FILE BY XREF AND THEN HICNO
* RESOLVING DUPLICATES TO ONE RECORD.
*******************************************************************;
PROC SORT DATA=TEMP NODUPKEY;
BY XREF HICNO;
RUN;

*******************************************************************;
* STEP 6. SELECTS ALL UNIQUE XREF/HICNO COMBINATIONS FROM THE
* CONSOLIDATED FILE.
*******************************************************************;
DATA OUT1.FINDR&SYSPARM
       (LABEL="Y&SYSPARM MEDPAR FINDER FILE");
SET TEMP;
BY XREF HICNO;
IF FIRST.XREF ;
RUN;


*******************************************************************;
* STEP 7. PROC PRINT AND PROC CONTENTS ON THE XREF/HICNO FILE.
*******************************************************************;
PROC PRINT DATA=OUT1.FINDR&SYSPARM(OBS=5);
TITLE "MEDPAR ID FINDER FILE FOR Y&SYSPARM";
TITLE2 'PRINT 5 OBS';
RUN;
TITLE2;
PROC CONTENTS DATA=OUT1.FINDR&SYSPARM(OBS=5);
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM201)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM201)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (DISK)
*                     - MACRO LOCATION, SPG FORMAT LOCATION
*            D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX  (TAPE)
*                     - MEDPAR YEAR XX SAS FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM2001.MEDPRBXX  (TAPE)
*                     - NEW MEDPAR YEAR XX SAS FILE
*            D#MSQ.@MSQ0000.SAS.YXXM2002.XMEDBXX   (DISK)
*                     - INFO FOR DELETIONS FOR YEAR XX
* PURPOSE   :CREATE A MODIFIED MEDPAR SAS FILE BY SELECTING
*            OUT OUT-OF-RANGE PROCEDURE CODES, DEMONSTRATION
*            CLAIMS, AND EMERGENCY ROOM CLAIMS.  PLEASE SEE
*            MACRO YXXM2001 FOR MORE INFORMATION.
*********************************************************************;

%INCLUDE IN2(YXXM2001)/SOURCE2;

      /**MACRO CALL**/
%YXXM2001(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2001)

```
%MACRO YXXM2001(YY=99,IN1=IN1,IN2=IN2,OUT1=OUT1,OUT2=OUT2);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2001
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2001)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 1, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      DROP RECORDS FROM THE MEDPAR FILE
      THAT MEET ONE OF THE FOLLOWING CRITERIA: 1) THEY
      ARE EMERGENCY ROOM CLAIMS, 2) THEY ARE DEMONSTRATION
      CLAIMS, 3) THEY HAVE AN OUT-OF-RANGE PROCEDURE CODE.
      SAVE THE NEW MEDPAR FILE WITHOUT THESE RECORDS TO TAPE,
      AND SAVE THE DELETED RECORDS TO DISK AS A PERMANENT FILE.

8  PARAMETERS:
   THERE ARE 5 KEYOWRD PARAMETERS: YY IN1 IN2 OUT1 OUT2

YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE CREATED IN PROGRAM 'M2PRE3',WHICH
IS STORED ON TAPE AND IS ABOUT 6GB (3 TAPES) IN SIZE.
THE FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPARXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
THIS FILE IS SORTED BY VARIABLE 'NOBS'.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IN2 DESIGNATES THE SAS FILENAME-JCL DDNAME FOR
THE SPG FORMAT PROCEDURE THAT WAS CREATED IN PROGRAM
'SPGFORM'. THE FILE LOCATION IS:
D#MSQ.@MSQ00M2.PGMLIB99.
E.G.: IN2=IN2  (THIS IS THE DEFAULT)

OUT1
OUT1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE NEW MEDPAR FILE THAT THIS PROGRAM GENERATES.
THE FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.YXXM2001.MEDPRBXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS ASSUMED THAT THIS FILE WILL BE STORED ON TAPE.
IT IS SORTED BY VARIABLE 'NOBS'. SIZE IS ABOUT 4GB.
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

OUT2
OUT2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR THE SMALL
FILE THAT CONTAINS INFO ON THE DELETED RECORDS.
THE FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.YXXM2001.XMEDBXX
```

```
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS ASSUMED THAT THIS FILE WILL BE STORED ON DISK
AS A PERMAMNENT FILE. IT IS SORTED BY VARIABLE 'NOBS'.
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2001) -- CONTINUED

```
SIZE IS ABOUT 1MB.
E.G.: OUT2=OUT2  (THIS IS THE DEFAULT)


9  OTHER SYMBOLIC VARIABLES OF NOTE:  NONE

10  OTHER NOTES: NONE

11 EXAMPLE MACRO CALL:
%YXXM2001(YY=98,IN1=INFILE)

*********************************************************************
*********************************************************************;

           /**CODE BEGINS**/

*********************************************************************;
* STEP 1. Reads in the SPG format.
*********************************************************************;

%INCLUDE &IN2(SPGFORM)/SOURCE2;

*********************************************************************;
* STEP 2. Creates two data sets.  The first data set is the new
* MedPAR file without claims that meet the exclusion criteria
* (MEDPRBXX), and the second data set contains the NOBS and the
* variable identifying the reason for exclusion for the excluded
* records (XMEDBXX).
*********************************************************************;

DATA &OUT1..MEDPRB&YY
     (DROP=Y&YY.EX BADPROC I COUNT
      LABEL="Y&YY MEDPRB FILE: NO ER/DEMO/BADPROC"
      SORTEDBY=NOBS)
     &OUT2..XMEDB&YY
     (KEEP=NOBS Y&YY.EX
      LABEL="Y&YY MEDPRB ER/DEMO/BADPROCS"
      SORTEDBY=NOBS);

*********************************************************************;
* STEP 3. Sets the length of the four variables created in this
* program (YYYEX, I, BADPROC, and COUNT).  Reads in the MedPAR file
* and sets in arrays for the variables to be used to define the
* exclusion criteria.
*********************************************************************;

LENGTH Y&YY.EX I BADPROC COUNT 2;

SET &IN1..MEDPAR&YY;

ARRAY PAYS(5) ACCHRG1-ACCHRG5;
ARRAY PROCS(6)$ SPROC1-SPROC6;
```

```
*******************************************************************;
* STEP 4. Creates the BADPROC and COUNT variables that will be used
* in the identification of claims with invalid procedure codes and
* ER claims.
*******************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2001) -- CONTINUED

```
DO I = 1 TO 6;
BADPROC=SUM(BADPROC,((PUT(LEFT(PROCS(I)),$SPG.)='NOT')
                     AND (PROCS(I) NE ' ')));
IF I NE 6 THEN COUNT=SUM(COUNT,PAYS(I) GT 0);
END;

*******************************************************************;
* STEP 5. Creates an exclusion criterion tracking variable (YYYEX).
*******************************************************************;

Y&YY.EX=(((DIS_DEST=20 OR DISSTAT='B') AND
         (SERCHG20 GT 0) AND (COUNT=0))*(1))     +
   (('880' LE (SUBSTR(PROVID,4,3)) LE '899')*(2)) +
       ((BADPROC GT 0)*(4));

*******************************************************************;
* STEP 6. If the exclusion variable (YYYEX) is greater than zero
* then the NOBS and the exclusion variable of the claim is output
* to the file XMEDBXX. Frequency on the exclusion variable (YYYEX).
*******************************************************************;

IF Y&YY.EX NE 0 THEN OUTPUT &OUT2..XMEDB&YY;
ELSE OUTPUT &OUT1..MEDPRB&YY;

LABEL Y&YY.EX="ER/DEMO/BADPROC EXCLUSION CODE";

RUN;

PROC FREQ DATA=&OUT2..XMEDB&YY;
TABLES Y&YY.EX;
FORMAT Y&YY.EX BINARY3.;
TITLE "Y&YY MEDPARB FILE EXCLUSIONS";
TITLE2 "Y&YY.EX HAS 3 BITS: FROM RIGHT TO LEFT:";
TITLE3 "BIT1=ER CLAIM BIT2=DEMO CLAIM BIT3=BAD PROC CODE";
RUN;

*******************************************************************;
* STEP 7. Proc contents and proc print on the XMEDBXX and MEDPRBXX
* files.
*******************************************************************;

PROC CONTENTS DATA=&OUT2..XMEDB&YY DIRECTORY;
TITLE2;
RUN;

PROC PRINT DATA=&OUT2..XMEDB&YY(OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;

PROC CONTENTS DATA=&OUT1..MEDPRB&YY DIRECTORY;
```

```
TITLE "Y&YY MEDPRB FILE";
TITLE2 "ER/DEMO/ AND BAD PROC CLAIMS HAVE BEEN DROPPED";
RUN;

PROC PRINT DATA=&OUT1..MEDPRB&YY(OBS=10);
TITLE3 "PRINT 10 OBS";
RUN;

        /**MACRO ENDS**/
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2001) -- CONTINUED

```
%MEND YXXM2001;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM203)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM203)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    : D#MSQ.@MSQ00M2.PGMLIB99 (DISK) - macro location
*             D#MSQ.@MSQ0000.SAS.XREF03.FINDRXX
*              - unique HICNO/XREF file
*             D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMPXX
*              - (IDs from the MedPAR file)
*             D#MSQ.@MSQ0000.SAS.YXXM2001.MEDPRBXX
*              - Y92-Y00 MedPAR File
* OUTPUTS   : D#MSQ.@MSQ0000.SAS.YXXM2003.MEDPRCXX (TAPE)
*                      - NEW MEDPAR YEAR XX SAS FILE
* PURPOSE   :CREATE A NEW MEDPAR SAS FILE BY ADDING ON
*            A UNIQUE ID FOR EACH PERSON WITH CLAIMS, AND
*            GENERATE 2 NEW VARIABLES.  PLEASE SEE
*            MACRO YXXM2003 FOR MORE INFORMATION.
*********************************************************************;

%INCLUDE IN0(YXXM2003)/SOURCE2;

      /**MACRO CALL**/
%YXXM2003(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2003)

```
%MACRO YXXM2003(YY=99,IN1=IN1,IN2=IN2,IN3=IN3,OUT1=OUT1);
%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2003
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2003)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 1, 2001
6  PROJECT:              MQMS
7  PURPOSE:
    THE PURPOSE IS TWOFOLD:
    1)MERGE ON THE 'HICNO' VARIABLE - THE UNIQUE ID FOR EACH PERSON.
    2)SELECT ONLY THE VARIABLES WE NEED FROM THE MEDPAR FILE
      GENERATED IN PROGRAM 'YXXM2001'

8  PARAMETERS:
   THERE ARE 5 KEYOWRD PARAMETERS: YY IN1 IN2 IN3 OUT1

YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE FINDER FILE THAT HAS 2 VARIABLES: 'XREF' AND 'HICNO'.
'XREF' HAS ALL OF A PERSON''S HISTORICAL IDS. 'HICNO' IS THEIR UNIQUELY
ASSIGNED ID (A UNIQUE ACTUAL HISTORICAL ID THAT WE ASSIGNED).
THE FILE LOCATION IS:  D#MSQ.@MSQ0000.SAS.M2XREF.FINDRXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS SORTED BY 'XREF'. SIZE IS ABOUT 200MB, ON DISK.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IN2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR ID FILE THAT HAS 2 VARS: 'XREF' AND 'NOBS'.
'XREF' HAS THE MEDPAR ID FOR EACH CLAIM.
'NOBS' IS THE UNIQUE NUMBER FOR EACH CLAIM.
THE FILE LOCATION IS: D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMPXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS SORTED BY 'XREF'. SIZE IS ABOUT 200MB, ON DISK.
E.G.: IN2=IN2  (THIS IS THE DEFAULT)

IN3
IN3 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE CREATED IN PROGRAM 'YXXM2001'
THE FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM2001.MEDPRBXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT WAS CREATED IN PROGRAM 'YXXM201' (XX=YEAR) AND STORED
ON TAPE. IT IS SORTED BY 'NOBS'. SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN3=IN3  (THIS IS THE DEFAULT)

OUT1
OUT1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE NEW MEDPAR FILE THAT THIS PROGRAM GENERATES.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2003.MEDPRCXX (XX=YEAR).
```

```
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON TAPE. SIZE IS ABOUT 4GB.
SORTED BY 'YXXNOBS' (XX=YEAR).
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2003) -- CONTINUED

```
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

9  OTHER SYMBOLIC VARIABLES OF NOTE:
THIS MACRO WILL GENERATE 2 OTHER SYMBOLIC VARIABLES:
'NEWNAME' - PROVIDES INFO FOR RENAME DATASET OPTION FOR IMPORTANT
VARS.
'KEEPERS' - WHICH SELECTS THE MEDPAR VARIABLES
THAT WE WILL KEEP ON THE FINAL FILE THAT THIS MACRO GENERATES.

10  OTHER NOTES:
THE WORK LIBRARY SHOULD BE ABLE TO HOLD AND SORT AN APPROX. 200MB
SAS TEMPORARY FILE ('MPAR2').
PLEASE HEED THIS IN THE JCL

11 EXAMPLE MACRO CALL:
%YXXM2003(YY=98,IN1=INFILE)

*********************************************************************
*********************************************************************;

               /**CODE BEGINS**/

   /**THESE ARE THE MEDPAR VARIABLES THAT WE WANT**/
   /**TO RENAME AND TO KEEP**/

*********************************************************************;
* STEP 1. This step is needed because the file layouts are different
* for the 1999 and 2000 MedPAR files affecting the organ acquisition
* charge variable name.
*********************************************************************;

%IF &YY=99 OR &YY=00 %THEN %LET ORG = SERCHG23=Y&YY.SCH23 ;
%ELSE %LET ORG = ORGCHRG=Y&YY.SCH23 ;

*********************************************************************;
* STEP 2. Renames the variables to include year markers and
* identifies them as NEWNAME so that these variables are easily
* selected to keep later in the program.
*********************************************************************;

%LET NEWNAME=
ADMDATE=Y&YY.ADMIT
BLOODED=Y&YY.BLOOD
COINAMT=Y&YY.COIN
DIAG1=Y&YY.DIA1
DIAG2=Y&YY.DIA2
DIAG3=Y&YY.DIA3
DIAG4=Y&YY.DIA4
DIAG5=Y&YY.DIA5
DIAG6=Y&YY.DIA6
DIAG7=Y&YY.DIA7
DIAG8=Y&YY.DIA8
DIAG9=Y&YY.DIA9
```

```
DIAG10=Y&YY.DIA10
DIAGCNT=Y&YY.DIACT
DIS_DEST=Y&YY.DDEST
DISDATE=Y&YY.DISCH
DISSTAT=Y&YY.DISST
DRGCODE=Y&YY.DRGCD
```


```
INPDED=Y&YY.INPED
NOBS=Y&YY.NOBS
PMT_AMT=Y&YY.PMT
PROVCODE=Y&YY.PROVC
PROVID=Y&YY.PRVID
SDATE1=Y&YY.SDT1
SDATE2=Y&YY.SDT2
SDATE3=Y&YY.SDT3
SDATE4=Y&YY.SDT4
SDATE5=Y&YY.SDT5
SDATE6=Y&YY.SDT6
&ORG
SPROC1=Y&YY.SPR1
SPROC2=Y&YY.SPR2
SPROC3=Y&YY.SPR3
SPROC4=Y&YY.SPR4
SPROC5=Y&YY.SPR5
SPROC6=Y&YY.SPR6
SURGCNT=Y&YY.SCNT
TOTPER=Y&YY.TOTDM
;

*********************************************************************;
* STEP 3. Specifies the variables to keep.
*********************************************************************;

%LET KEEPERS=;
%DO I = 1 %TO 75 %BY 2;
%LET KEEPERS=&KEEPERS %SCAN(&NEWNAME,&I,%STR( )=) ;
%END;


     /**MERGE ON HICNO VAR FROM THE CROSS-REFERENCE**/
    /**FINDER FILE TO THE MEDPAR ID FILE**/

*********************************************************************;
* STEP 4. Merges the XREFMPXX file (IDs from the MedPAR file) and
* the FINDRXX file (unique HICNO/XREF file).  Matches go to temp
* file MPAR2.  Non-matches to the FINDRXX file go to a temp file
* ERR.  Proc print on ERR.
*********************************************************************;

DATA MPAR2(DROP=XREF) ERR(KEEP=XREF);
MERGE &IN2..XREFMP&YY (IN=ONE)
      &IN1..FINDR&YY  (IN=TWO);
BY XREF;
IF ONE AND TWO THEN OUTPUT MPAR2;
ELSE IF ONE AND NOT(TWO) THEN OUTPUT ERR;
RUN;
```

```
PROC PRINT DATA=ERR;
TITLE "MEDPAR Y&YY IDS WITH NO CROSS-REFERENCE MATCH";
RUN;

     /**SORT THIS TEMPORARY HICNO DATASET BY THE UNIQUE CLAIM #**/

*********************************************************************;
* STEP 5. Sort MPAR2 file by NOBS to prepare for the merge to the
* MEDPRBXX file.  Rename NOBS and HICNO with year markers to match
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2003) -- CONTINUED
```
* other MEDPRBXX file names.
*********************************************************************;

PROC SORT DATA=MPAR2(
RENAME=(NOBS=Y&YY.NOBS HICNO=Y&YY.HICNO));
BY Y&YY.NOBS;
RUN;

    /**CREATE NEW MEDPAR PERMANENT DATASET WITH HICNO-**/
    /**THE NEW UNIQUE ID - AND SELECTED VARS**/

*********************************************************************;
* STEP 6. Create MEDPRCXX file after merging MEDPRBXX with MPAR2
* by NOBS. Create two payment variables - YXXTPAY and YXXBPAY.
*********************************************************************;

DATA &OUT1..MEDPRC&YY
      (LABEL="Y&YY MEDPRC: SELECTED VARS WITH HICNO"
       SORTEDBY=Y&YY.NOBS);
LENGTH I 2;
ARRAY MONEY(6) Y&YY.PMT Y&YY.TOTDM Y&YY.SCH23
               Y&YY.COIN Y&YY.INPED Y&YY.BLOOD;
MERGE &IN3..MEDPRB&YY
      (KEEP=&KEEPERS RENAME=(&NEWNAME) IN=ONE)
      MPAR2 (IN=TWO);
BY Y&YY.NOBS;

        /**CREATE 2 NEW PAYMENT VARS**/
           /**SET MISSING AND NEGATIVE AMTS TO 0**/
DO I = 1 TO 6;
    IF MONEY(I) LT 0 THEN MONEY(I)=0;
END;
DROP I;
Y&YY.TPAY=SUM(Y&YY.PMT,Y&YY.TOTDM,Y&YY.SCH23);
Y&YY.BPAY=SUM(Y&YY.COIN,Y&YY.INPED,Y&YY.BLOOD);

IF ONE AND TWO ;

*********************************************************************;
* STEP 7. Create labels for new variables.
*********************************************************************;

LABEL
Y&YY.HICNO ="Y&YY CLEAN XREFD ID"
Y&YY.NOBS="Y&YY UNIQUE CLAIM NUMBER"
Y&YY.TPAY="Y&YY CREATED TOTAL PAY"
Y&YY.BPAY="Y&YY CREATED BENE PAY"
```

```
;
RUN;

*******************************************************************;
* STEP 8. Proc contents and proc print on the MEDPRCXX file.
*******************************************************************;

PROC CONTENTS DATA=&OUT1..MEDPRC&YY DIRECTORY ;
TITLE "MEDPRC Y&YY: SELECTED VARIABLES WITH HICNO ADDED";
RUN;

PROC PRINT DATA=&OUT1..MEDPRC&YY(OBS=10);
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2003) -- CONTINUED

TITLE2 "PRINT 10 OBSERVATIONS";
RUN;

             /**MACRO ENDS**/
%MEND YXXM2003;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM204)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM204)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (DISK)
*                     - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.XREF03.FINDRXX (TAPE)
*                     - MEDPAR FINDER SAS FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.M2PRE2.XREFMPXX (TAPE)
*                     - NEW MEDPAR YEAR XX CROSS-REFERENCE SAS FILE
*            D#MSQ.@MSQ0000.SAS.YXXM2003.MEDPRCXX (TAPE)
*                     - MEDPARB SAS FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX (TAPE)
*                     -  YEAR XX M1 ANALYTIC FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YXXM2004.MEDPRDXX   (TAPE)
*                      - NEW MEDPAR FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM2004.DENOMCXX   (TAPE)
*                      -  NEW YEAR XX MODIFIED M1 DENOMINATOR FILE
*            D#MSQ.@MSQ0000.SAS.YXXM2004.BADIDSXX
*                      - not in M.1.4 duplicates on Denominator File
* PURPOSE   :CREATE A NEW MEDPAR SAS FILE BY EXCLUDING
*            CLAIMS FOR THOSE PERSONS WITH NO M1 MATCH OR WITH
*            MULTIPLE M1 FILE IDS.  ALSO CREATE A NEW MODIFIED
*            M1 FILE TO USE FOR LATER M2 PROGRAMS.  PLEASE SEE
*            MACRO YXXM2004 FOR MORE DETAILS.
*********************************************************************;

%INCLUDE IN0(YXXM2004)/SOURCE2;

      /**MACRO CALL**/
%YXXM2004(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004)

```
%MACRO YXXM2004(YY=99,IN1=IN1,IN2=IN2,IN3=IN3,IN4=IN4,
               OUT1=OUT1,OUT2=OUT2,OUT3=OUT3);
%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2004
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 2, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS TWOFOLD:
      1)CREATE A NEW M.2 ANALYTIC MEDPAR FILE THAT FLAGS
        ANY PERSON WHO EITHER DOES NOT MATCH TO FILE M.1.4 OR WHO
        DOES MATCH, BUT THAT MATCH IS FOUND TO INCLUDE TWO OR MORE
        IDS ON FILE M.1.4 THAT REPRESENT THE SAME PERSON.
      2)CREATE A MODIFIED M.1.4 FILE THAT WILL CONTAIN ONLY THOSE
        PERSONS SELECTED BY THE ABOVE CRITERIA.  THIS FILE WILL BE
        USED TO MERGE ON THE M.1 INFORMATION TO FORM THE PRINCIPAL
        M.2 ANALYTIC FILE AFTER WE HAVE FINISHED CLEANING THE MEDPAR
        CLAIMS.

8  PARAMETERS:
   THERE ARE 8 KEYOWRD PARAMETERS: YY IN1 IN2 IN3 IN4 OUT1 OUT2 OUT3

YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN1 DESIGNATES THE LIBREF OR JCL DDNAME FOR
THE FINDER FILE THAT HAS 2 VARS: 'XREF' AND 'HICNO'.
XREF HAS ALL OF A PERSON''S HISTORICAL IDS. HICNO IS THEIR UNIQUELY
ASSIGNED ID.  THE FILE IS: D#MSQ.@MSQ0000.SAS.M2XREF.FINDRXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS SORTED BY 'XREF'. SIZE IS ABOUT 200MB, ON DISK.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IN2 DESIGNATES THE LIBREF OR JCL DDNAME FOR
THE MEDPAR ID FILE THAT HAS 2 VARS: 'XREF' AND 'NOBS'.
XREF HAS THE MEDPAR ID FOR EACH CLAIM.
NOBS IS THE UNIQUE NUMBER FOR EACH CLAIM.
THE FILE IS: D#MSQ.@MSQ0000.SAS.M2PRE3.XREFMPXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS SORTED BY 'NOBS'. SIZE IS ABOUT 200MB, ON DISK.
E.G.: IN2=IN2  (THIS IS THE DEFAULT)

IN3
IN3 DESIGNATES THE LIBREF OR JCL DDNAME FOR
THE MEDPAR FILE CREATED IN 'YXXM2003'.(XX=YEAR)
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2003.MEDPRCXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT WAS CREATED IN PROGRAM 'YXXM2003' AND STORED
```

ON TAPE. IT IS SORTED BY YXXNOBS(XX=YEAR).
SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN3=IN3 (THIS IS THE DEFAULT)

IN4
IN4 DESIGNATES THE LIBREF OF JCL DDNAME FOR THE M.1.4
ANALYTICAL FILE THAT WAS CREATED IN M1 'YXXM1002'
FILE IS: D#MSQ.@MSQ0000.SAS.YXXM1002.DENOMBXX (XX=YEAR)
IT IS ON TAPE AND IS ABOUT 3GB AND IS SORTED BY 'YXXDOBS'(XX=YEAR).
E.G.: IN4=IN4 (THIS IS THE DEFAULT)

OUT1
OUT1 DESIGNATES THE LIBREF OR JCL DDNAME FOR
THE NEW MEDPAR FILE THAT THIS PROGRAM GENERATES.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2004.MEDPRDXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON TAPE. SIZE IS ABOUT 3GB.
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT2
OUT2 DESIGNATES THE LIBREF OR JCL DDNAME FOR THE NEW M.1.4 FILE
THAT WILL BE USED TO MERGE ON THE M.1 INFO TO THE FINAL M.2
ANALYTIC FILE.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2004.DENOMCXX (XX=YEAR)
THE LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS ASSUMED THAT THIS FILE WILL BE PUT ON TAPE. SIZE IS ABOUT
3GB.
E.G.: OUT2=OUT2 (THIS IS THE DEFAULT)

OUT3
OUT3 DESIGNATES THE LIBREF OR JCL DDNAME FOR THE FILE THAT CONTAINS
THE IDS AND CLAIMS RECORD NUMBERS FOR ALL MEDPAR CLAIMS AT THIS
STAGE THAT DO NOT HAVE AN M.1.4 MATCH OR A MATCH IN THE DENOMINATOR
FILE WITH MULTIPLE RECORDS.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2004.BADIDSXX (XX=YEAR)
THE LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS ASSUMED THAT THIS FILE WILL BE PUT ON DISK. SIZE IS ABOUT
10MB.
E.G.: OUT3=OUT3 (THIS IS THE DEFAULT)

9 OTHER SYMBOLIC VARIABLES OF NOTE: NONE

10 OTHER NOTES:
THE WORK LIBRARY SHOULD BE ABLE TO HOLD AND SORT AN APPROX. 200MB
SAS TEMPORARY FILE - 'MPAR2'.
PLEASE HEED THIS IN THE JCL

11 EXAMPLE MACRO CALL:
%YXXM2004(YY=98,IN1=INFILE)

```
*****************************************************************
*****************************************************************;

           /**CODE BEGINS**/

    /**RETRIEVE AND SORT ALL M.1.4 IDS**/
```

```
*******************************************************************;
* STEP 1. Read in M1 denom file and select M14 cohort and their
* HICNO and obs number.  Hicno is called XREF.  Write out temporary
* file of M14 ids sorted by XREF.
*******************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004) -- CONTINUED

```
PROC SORT DATA=&IN4..DENOMB&YY
(KEEP=Y&YY.HICNO Y&YY.DOBS Y&YY.M14 WHERE=(Y&YY.M14='1')
RENAME=(Y&YY.HICNO=XREF))
OUT=M14X(DROP=Y&YY.M14);
BY XREF;
RUN;

    /**SELECT THE CURRENT MEDPAR IDS FROM THE FINDER FILE**/

*******************************************************************;
* STEP 2. Sort the latest MedPAR file (MEDPRCXX) by HICNO and create
* a unique listing of IDs.  Create temporary dataset called FINDMP
*******************************************************************;

PROC SORT DATA=&IN3..MEDPRC&YY(KEEP=Y&YY.HICNO
      RENAME=(Y&YY.HICNO=HICNO)) NODUPKEY OUT=FINDMP;
BY HICNO;
RUN;

*******************************************************************;
* STEP 3. Sort the finder file (FINDRXX) by HICNO and write out
* a temporary file called FINDR.
*******************************************************************;

PROC SORT DATA=&IN1..FINDR&YY OUT=FINDR;
BY HICNO;
RUN;

*******************************************************************;
* STEP 4. Merge the finder file and the MedPAR ID file by HICNO
* that provides a list of everyone in the MedPAR file with their
* unique IDs and their historical ids.
*******************************************************************;

DATA FINDMP;
MERGE FINDR(IN=ONE) FINDMP(IN=TWO);
BY HICNO;
IF ONE AND TWO;
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE FINDR;RUN;
PROC SORT DATA=FINDMP;
BY XREF;
RUN;

     /**FOR THOSE PERSONS WHO HAVE SHORT-TERM STAY MEDPAR CLAIMS.**/
     /**CHECK FOR MATCHES AND MISMATCHES**/

*******************************************************************;
* STEP 5. Merge M14 ID file (M14X) and MedPAR ID list with
* historical IDs (FINDMP).  Matches go into temporary file MATCH1,
```

```
* non-matches to NOMATCH1.
*******************************************************************;

DATA MATCH1
     NOMATCH1(KEEP=HICNO)
;
MERGE M14X(IN=ONE) FINDMP(IN=TWO);
BY XREF;
IF ONE AND TWO THEN OUTPUT MATCH1;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004) -- CONTINUED
```
ELSE IF TWO AND NOT(ONE) THEN OUTPUT NOMATCH1;
RUN;

PROC DATASETS LIB=WORK NOLIST;DELETE M14X;RUN;
      /**SORT THE MATCHED IDS BY THE UNIQUE CLAIM ID**/

*******************************************************************;
* STEP 6. Sort MATCH1 by HICNO (unique ID) to identify those
* beneficiaries with duplicate IDs.  Duplicates go into temporary
* file DUPLIC1 while the rest of the beneficiaries remain in MATCH1.
*******************************************************************;

PROC SORT DATA=MATCH1;
BY HICNO;
RUN;

     /**SEPARATE THOSE MATCHED IDS IN M.1.4 INTO 2 GROUPS:**/
     /**THOSE THAT HAVE NO DUPLICATES AND THOSE THAT DO**/

DATA MATCH1 DUPLIC1(KEEP=HICNO);
SET MATCH1;
BY HICNO;
IF NOT(FIRST.HICNO AND LAST.HICNO) THEN DO;
    IF FIRST.HICNO THEN OUTPUT DUPLIC1;
    END;
ELSE OUTPUT MATCH1;
RUN;

      /**SORT THE UNMATCHED IDS BY THE UNIQUE CLAIM ID**/

*******************************************************************;
* STEP 7. Sort NOMATCH1 by HICNO in preparation for merge to MedPAR
* file to identify true non-matches.  Merge MATCH1 and NOMATCH1 by
* HICNO (unique ID).
*******************************************************************;

PROC SORT DATA=NOMATCH1 NODUPKEY;
BY HICNO;
RUN;

     /**MERGE THEM ONTO THE MATCHED IDS (NO DUPS) BY UNIQUE ID**/
     /**TO SEE WHICH ONES TRULY REPRESENT UNMATCHED PERSONS**/

DATA NOM14;
MERGE MATCH1(DROP=Y&YY.DOBS XREF IN=ONE) NOMATCH1(IN=TWO);
BY HICNO;
IF TWO AND NOT(ONE);
```

```
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE NOMATCH1;RUN;

     /**GENERATE A CONCATENATED LIST OF ALL BAD UNIQUE CLAIM IDS**/
     /**BAD = NO MATCH TO M.1.4 OR A MATCH BUT DUPLICATE IN M.1.4**/

*******************************************************************;
* STEP 8. Create BADIDS1 that includes beneficiaries with duplicate
* IDs on the Denominator File (DUPLIC1) and true non-matches to the
* M1-4 cohort (NOM14).
*******************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004) -- CONTINUED

```
DATA BADIDS1;
SET DUPLIC1 NOM14;
BY HICNO;
IF FIRST.HICNO;
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE NOM14 DUPLIC1;RUN;


      /**GET THE UNIQUE CLAIMS NUMBERS FOR THE UNIQUE IDS**/

*******************************************************************;
* STEP 9. Merge the XREFMP (IDs and NOBS from complete MedPAR
* file prior to exclusions) and FINDMP (MedPAR unique IDs and
* historical IDs) files.  Matches go into temp file FINDER.
*******************************************************************;

DATA FINDER(KEEP=HICNO NOBS);
MERGE &IN2..XREFMP&YY(IN=ONE) FINDMP(IN=TWO);
BY XREF;
IF ONE AND TWO;
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE FINDMP;RUN;

*******************************************************************;
* STEP 10. Sort FINDER by HICNO and merge with BADIDS1.  Matches
* go into the dataset BADIDSXX which provides the unique ID,
* historical ID, and NOBS information for beneficiaries to be
* excluded from the MedPAR file.
*******************************************************************;

PROC SORT DATA=FINDER;
BY HICNO;
RUN;

     /**MERGE ONTO ALL THE BAD CLAIMS UNIQUE IDS THEIR ASSOCIATED**/
     /**UNIQUE CLAIMS NUMBERS**/

DATA &OUT3..BADIDS&YY
     (LABEL="Y&YY CLAIMS #S FOR NO M14 CLAIMS OR DENOM DUPS"
      SORTEDBY=HICNO)
;
MERGE FINDER(IN=ONE) BADIDS1(IN=TWO);
BY HICNO;
IF ONE AND TWO;
```

```
LABEL HICNO="Y&YY UNIQUE XREFD ID"
      NOBS="Y&YY UNIQUE MPAR CLAIMS #"
;
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE FINDER BADIDS1;RUN;

*******************************************************************;
* STEP 11. Sort BADIDSXX by NOBS and create temporary dataset
* called BADIDS2 in preparation for merge to MEDPRCXX file.
*******************************************************************;

PROC SORT DATA=&OUT3..BADIDS&YY(KEEP=NOBS) OUT=BADIDS2;
BY NOBS;
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004) -- CONTINUED

```
     /**FROM THE MEDPAR FILE FLAG ALL OF THE CLAIMS FOR**/
     /**PERSONS THAT HAVE EITHER NO MATCH IN M.1.4 OR A MATCH**/
     /**BUT WITH DUPLICATES**/

*******************************************************************;
* STEP 12. Merge MEDPRCXX file with BADIDS2 by NOBS.  Create
* YXXNOM14 equal to '1' if they match, else equals '0'.  Create
* a new MedPAR file (MEDPRDXX).
*******************************************************************;

DATA &OUT1..MEDPRD&YY;
LENGTH Y&YY.NOM14 $1;
MERGE &IN3..MEDPRC&YY(IN=ONE) BADIDS2(IN=TWO RENAME=(NOBS=Y&YY.NOBS));
BY Y&YY.NOBS;
IF ONE;
Y&YY.NOM14=PUT(ONE AND TWO,1.);
LABEL Y&YY.NOM14="IDS WITH NO M14 MATCH OR DUPS";
RUN;
PROC DATASETS LIB=WORK NOLIST;DELETE BADIDS2;RUN;

*******************************************************************;
* STEP 13. Proc contents and proc print on MEDPRDXX.
*******************************************************************;

PROC CONTENTS DATA=&OUT1..MEDPRD&YY POSITION;
TITLE "YEAR &YY MEDPAR D FILE";
RUN;
PROC PRINT DATA=&OUT1..MEDPRD&YY(OBS=10) SPLIT=' ';
TITLE2 "PRINT 10 OBS";
RUN;

    /**CREATE A NEW MODIFIED M.1.4 FILE WITH ONLY GOOD MATCHES**/
    /**TO THE PERSONS IN OUR MEDPAR CLAIMS FILE**/
    /**THIS FILE WILL BE MERGED ONTO THE CLAIMS FILE**/
    /**AFTER WE HAVE FINISHED FURTHER CLEANING OF THE CLAIMS**/

*******************************************************************;
* STEP 14. Sort MATCH1 by YXXDOBS and then merge MATCH1 with
* DENOMBXX by YXXDOBS.  Matches go into the new Denominator
* File (DENOMCXX).
*******************************************************************;
```

```
PROC SORT DATA=MATCH1(KEEP=Y&YY.DOBS HICNO RENAME=(HICNO=Y&YY.HICNO));
BY Y&YY.DOBS;
RUN;

DATA &OUT2..DENOMC&YY;
MERGE &IN4..DENOMB&YY(WHERE=(Y&YY.M14='1') IN=ONE)
     MATCH1(IN=TWO);
BY Y&YY.DOBS;
IF ONE AND TWO;
RUN;

*****************************************************************;
* STEP 15. Proc contents and proc print on DENOMCXX.
*****************************************************************;

PROC CONTENTS DATA=&OUT2..DENOMC&YY POSITION;
TITLE "YEAR &YY DENOM C FILE";
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2004) -- CONTINUED
```

RUN;
PROC PRINT DATA=&OUT2..DENOMC&YY(OBS=10) SPLIT=' ';
TITLE2 "PRINT 10 OBS";
RUN;
          /**MACRO ENDS**/
%MEND YXXM2004;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM205)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM205)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (DISK)
*                     - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM2004.MEDPRDXX  (TAPE)
*                     - MEDPAR SAS FILE FOR YEAR XX
* OUTPUTS    D#MSQ.@MSQ0000.SAS.YXXM2005.MEDPREXX   (TAPE)
*                      - NEW MEDPAR FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM2005.MARKERXX   (DISK)
*                      - OVERLAPPING CLAIMS OBS # FOR YEAR XX
* PURPOSE   :CREATE A NEW MEDPAR SAS FILE BY EXCLUDING
*            CLAIMS THAT OVERLAP WITH OTHER CLAIMS.
*            SAVE OVERLAPPING CLAIMS INFO TO DISK. PLEASE SEE
*            MACRO YXXM2005 FOR MORE DETAILS.
*********************************************************************;

%INCLUDE IN0(YXXM2005)/SOURCE2;

      /**MACRO CALL**/
%YXXM2005(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2005)

```
%MACRO YXXM2005(YY=99,IN1=IN1,WORK1=WORK1,OUT1=OUT1,OUT2=OUT2);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2005
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2005)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 APRIL 4, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      IDENTIFY AND DROP ALL OVERLAPPING CLAIMS.
      OUTPUT INFO ON OVERLAPPING CLAIMS TO DISK FILE.
      OUTPUT THE THE NEW RESULTING MEDPAR CLAIMS FILE TO TAPE.

8  PARAMETERS:
   THERE ARE 5 KEYOWRD PARAMETERS: YY IN1 WORK1 OUT1 OUT2

YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN3 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE CREATED IN PROGRAM 'YXXM2004' (XX=YEAR).
THE FILE IS: D#MSQ.@MSQ00.SAS.YXXM2004.MEDPRDXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT WAS CREATED IN PROGRAM 'YXXM2004' AND STORED
ON TAPE. IT IS SORTED BY VARIABLE YXXNOBS (XX=YEAR).
SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

WORK1
WORK1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR THE LIMITED
AND TEMPORARY MEDPAR FILE THAT THIS PROGRAM USES TO IDENTIFY
OVERLAPPING CLAIMS.  IT IS A SUBSET OF THE FILE CREATED IN
PROGRAM 'YXXM2004' (XX=YEAR).
IT WILL BE ABOUT 500MB AND IS ASSUMED TO BE STORED
ON DISK. INTERNAL SAS NAME IS 'MARKER'.
E.G.: WORK1=WORK1  (THIS IS THE DEFAULT)


OUT1
OUT1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE THAT THIS PROGRAM GENERATES.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2005.MEDPREXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON TAPE. SIZE IS ABOUT 4GB.
FILE SORTED BY YXXNOBS.
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

OUT2
OUT2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE FILE THAT HOLDS UNIQUE CLAIMS NUMBERS FOR ALL OVERLAPPING CLAIMS.
```

```
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2005.MARKERXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON DISK. SIZE IS ABOUT 20MB.
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2005) -- CONTINUED

```
FILE SORTED BY VARIABLE 'CASENO'.
E.G.: OUT2=OUT2  (THIS IS THE DEFAULT)


9  OTHER SYMBOLIC VARIABLES OF NOTE:  NONE

10  OTHER NOTES:
THE WORK LIBRARY SHOULD BE ABLE TO HOLD AN APPROX. 20MB
SAS TEMPORARY FILE - 'TEMPCNT' (WHICH HOLDS
SOME BASIC STATS OF THESE OVERLAPPING CLAIMS).
PLEASE HEED THIS IN THE JCL

11 EXAMPLE MACRO CALL:
%YXXM2005(YY=98)

*****************************************************************
*****************************************************************;
   /**SORT THE MEDPAR FILE**/
   /**SELECT ONLY THOSE VARS NEEDED TO CHECK FOR OVERLAPS**/
   /**WRITE WORK1.MPAR1 TO DISK AS TEMPORARY FILE**/

*****************************************************************;
* STEP 1. Sort latest MedPAR File (MEDPRDXX) by HICNO, admission
* date, discharge date, and PROVID to facilitate identification of
* overlapping claims.  Output to temp file MPAR1.
*****************************************************************;

PROC SORT DATA=&IN1..MEDPRD&YY
(KEEP=Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.NOBS Y&YY.PRVID)
OUT=&WORK1..MPAR1;
BY Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.PRVID;
RUN;

  /**PULL THE UNIQUE OBSERVATION # FOR EACH CLAIM**/
  /**FOR EACH GROUP OF CLAIMS THAT FORM AN OVERLAPPING EPISODE.**/
  /**WRITE FILE TO DISK AS TEMPORARY FILE**/

*****************************************************************;
* STEP 2. Create a permanent data file (MARKERXX) that contains
* two variables - YXXNOBS (claims observation number) and CASENO
* (overlapping claims case number).
*****************************************************************;

DATA &OUT2..MARKER&YY
(KEEP=NEWOBS
      CASENO     /**UNIQUE # FOR A GROUP OF OVERLAPPING CLAIMS**/
      RENAME=(NEWOBS=Y&YY.NOBS));

LENGTH HICNO2 $ 11 MARK $ 1 PRVID2 $ 6;
RETAIN MARK ' ';

        /**DEFINE LAGGED VARIABLES**/
```

```
PRVID2=Y&YY.PRVID;
HICNO2=Y&YY.HICNO;
ADMDATE2=Y&YY.ADMIT;
DISDATE2=Y&YY.DISCH;
NOBS2=Y&YY.NOBS;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2005) -- CONTINUED

```
*******************************************************************;
* STEP 3. Read in MPAR1.
*******************************************************************;

SET &WORK1..MPAR1;

*******************************************************************;
* STEP 4. Set up algorithm to identify overlapping claims.
* Overlapping claims are those claims where: 1) HICNOs are the same;
* and 2) Admission date is less than or equal to the discharge date.
*******************************************************************;

/**DEFINE CRITERIA FOR OVERLAPPING CLAIMS**/

IF (
(HICNO2=Y&YY.HICNO)
AND (ADMDATE2 LE Y&YY.ADMIT LT DISDATE2)
    )
THEN DO;
    IF MARK=' ' THEN DO;
        CASENO+1;
        NEWOBS=NOBS2;
        OUTPUT;
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
        MARK='1';
    END;
    ELSE IF MARK='1' THEN DO;
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
    END;
END;
ELSE MARK=' ';
RUN;

PROC CONTENTS DATA=&OUT2..MARKER&YY DIRECTORY;
TITLE "YEAR &YY OVERLAPPING CLAIMS ID FILE";
RUN;
PROC PRINT DATA=&OUT2..MARKER&YY (OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;


         /**DESCRIBE BASIC INFO ON THESE OVERLAPPING CLAIMS**/

*******************************************************************;
* STEP 5. Frequency on CASENO variable and output to a temp file
* (TEMPCNT).  Run means on the counts in the temp file.
*******************************************************************;
```

```
PROC FREQ DATA=&OUT2..MARKER&YY NOPRINT;
TABLES CASENO / OUT=TEMPCNT(DROP=PERCENT);
RUN;
PROC MEANS DATA=TEMPCNT N MIN MAX MEAN STD;
VAR COUNT;
TITLE "YEAR &YY OVERLAPPING CLAIMS DELETIONS";
TITLE2 "STATS ARE FOR VARIABLE COUNT:";
TITLE3 "THE # OF CLAIMS IN EACH OVERLAPPING EPISODE";
RUN;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2005) -- CONTINUED

```
    /**SORT THESE RECORDS BY OBSERVATION # TO MERGE THEM ONTO**/
    /**THE MEDPAR FILE**/

*******************************************************************;
* STEP 6. Sort the MARKERXX file by observation number (YXXNOBS)
* to be able to merge this information onto the latest MedPAR file
* to identify overlapping claims and remove them.
*******************************************************************;

PROC SORT DATA=&OUT2..MARKER&YY;
BY Y&YY.NOBS;
RUN;

    /**GENERATE THE NEW MEDPAR FILE: OVERLAPPING CLAIMS DELETED**/

*******************************************************************;
* STEP 7. Merge MARKERXX and MEDPRDXX file by YXXNOBS.  Non-matches
* go into the new MedPAR file (MEDPREXX) - non-overlapping claims.
*******************************************************************;

DATA &OUT1..MEDPRE&YY
   (SORTEDBY=Y&YY.NOBS LABEL="MEDPAR E Y&YY: NO OVERLPS");
MERGE &IN1..MEDPRD&YY (IN=ONE)
      &OUT2..MARKER&YY (IN=TWO KEEP=Y&YY.NOBS);
BY Y&YY.NOBS;
IF ONE AND NOT(TWO);
RUN;

*******************************************************************;
* STEP 8. Proc contents and proc print on MEDPREXX file.
*******************************************************************;

PROC CONTENTS DATA=&OUT1..MEDPRE&YY DIRECTORY ;
RUN;
PROC PRINT DATA=&OUT1..MEDPRE&YY(OBS=10) SPLIT=' ';
TITLE "PRINT 1O OBS OF MEDPAR Y&YY FILE: OVERLAPS DELETED";
RUN;

      /**MACRO ENDS**/

%MEND YXXM2005;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM206)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM206)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (DISK)
*                     - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM2005.MEDPREXX  (TAPE)
*                     - MEDPAR SAS FILE FOR YEAR XX
* OUTPUTS    D#MSQ.@MSQ0000.SAS.YXXM2006.MEDPRFXX   (TAPE)
*                      - NEW MEDPAR FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM2005.TRANSXX   (DISK)
*                      - TRANSFER CLAIMS OBS #S FOR YEAR XX
* PURPOSE   :CREATE A NEW MEDPAR SAS FILE BY RESOLVING
*            TRANFER EPISODES INTO ONE CLAIMS PER EPISODE,
*            AND SAVE THIS FILE TO TAPE.
*            SAVE TRANSFER CLAIMS INFO TO DISK. PLEASE SEE
*            MACRO YXXM2006 FOR MORE DETAILS.
*********************************************************************;

%INCLUDE IN0(YXXM2006)/SOURCE2;

      /**MACRO CALL**/
%YXXM2006(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006)

```
%MACRO YXXM2006(YY=99,IN1=IN1,WORK1=WORK1,WORK2=WORK2,
                OUT1=OUT1,OUT2=OUT2);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2006
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 MAY 24, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      FROM THE MEDPAR FILE CREATED IN PROGRAM YXXM2005 (XX=YEAR)
      SELECT ALL CLAIMS THAT FORM TRANSFER EPISODES, AND FOR EACH
      EPISODE, RESOLVE THE GROUP OF CLAIMS TO ONE CLAIM, AND INSERT
      THE RESOLVED CLAIMS INTO A NEW MEDPAR FILE THAT WILL BE STORED
      ON TAPE.  ALSO, SAVE TO DISK THE OBSERVATION NUMBER AND
      CASE NUMBER FOR EACH TRANSFER CLAIM.

8  THERE ARE 6 KEYWORD PARAMETERS: YY IN1 WORK1 WORK2 OUT1 OUT2.

YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE CREATED IN PROGRAM YXXM2005 (XX=YEAR).
THE FILE IS: D#MSQ.@MSQ00.SAS.YXXM2005.MEDPREXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT WAS CREATED IN PROGRAM YXXM2005 AND STORED
ON TAPE. IT IS SORTED BY VARIABLE YXXNOBS (XX=YEAR).
SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

WORK1
WORK1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR THE LIMITED
AND TEMPORARY MEDPAR FILE THAT THIS PROGRAM USES TO IDENTIFY
TRANSFER CLAIMS.  IT IS A SUBSET OF THE FILE CREATED IN
PROGRAM YXXM2005 (XX=YEAR).
IT WILL BE ABOUT 500MB AND IS ASSUMED TO BE STORED
ON DISK. INTERNAL SAS NAME IS "MPAR1".
E.G.: WORK1=WORK1  (THIS IS THE DEFAULT)

WORK2
WORK2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR ANOTHER LIMITED
AND TEMPORARY MEDPAR FILE THAT THIS PROGRAM USES TO MANIPULATE
TRANSFER CLAIMS.  IT IS A SUBSET OF THE FILE CREATED IN
PROGRAM YXXM2005 (XX=YEAR).
IT WILL BE ABOUT 500MB AND IS ASSUMED TO BE STORED
ON DISK. INTERNAL SAS NAME IS "MPAR2".
E.G.: WORK2=WORK2  (THIS IS THE DEFAULT)

OUT1
```

```
OUT1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE THAT THIS PROGRAM GENERATES.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2006.MEDPRFXX (XX=YEAR).
```

```
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON TAPE. SIZE IS ABOUT 4GB.
FILE SORTED BY YXXNOBS.
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

OUT2
OUT2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE FILE THAT HOLDS UNIQUE CLAIMS NUMBERS FOR ALL TRANSFER CLAIMS.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2006.TRANSXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON DISK. SIZE IS ABOUT 20MB.
FILE SORTED BY VARIABLES CASENO AND COUNTS.
E.G.: OUT2=OUT2  (THIS IS THE DEFAULT)

9  OTHER SYMBOLIC VARIABLES OF NOTE:  NONE

10  OTHER NOTES:
THE WORK LIBRARY SHOULD BE ABLE TO HOLD 2 APPROX. 20MB
SAS TEMPORARY FILES - "TEMPCNT" AND "T1" (THESE HOLD
SOME BASIC STATS OF THESE TRANSFER CLAIMS).
PLEASE HEED THIS IN THE JCL

11 EXAMPLE MACRO CALL:
%YXXM2006(YY=98)

*********************************************************************
*********************************************************************;

   /***CODE BEGINS***/

*********************************************************************;
* STEP 1. Sort latest MedPAR File (MEDPREXX) by HICNO, admission
* date, discharge date, and PROVID to facilitate identification of
* transfer claims.  Output to temp file MPAR1.
*********************************************************************;

PROC SORT DATA=&IN1..MEDPRE&YY
(KEEP=Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.PRVID
      Y&YY.DDEST Y&YY.NOBS)
OUT=&WORK1..MPAR1;
BY Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.PRVID;
RUN;

  /**PULL THE UNIQUE OBSERVATION NUMBER AND CASE NUMBER"**/
  /**FOR EACH GROUP OF CLAIMS THAT FORMS A TRANSFER**/

*********************************************************************;
* STEP 2. Create a permanent data file (TRANSXX) that contains
* three variables - YXXNOBS (claims observation number), CASENO
* (transfer claims case number), and COUNTS (rank variable
* providing the order of the transfer claims in a particular
* transfer episode).
*********************************************************************;
```

```
DATA &OUT2..TRANS&YY(KEEP=NEWOBS
                   CASENO  /**UNIQUE NUMBER FOR TRANSFER EPISODE**/
                   COUNTS
              RENAME=(NEWOBS=Y&YY.NOBS)
              LABEL="YEAR &YY TRANSFER OBS AND CASE #S"
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006) -- CONTINUED

              SORTEDBY=CASENO COUNTS);
LENGTH MARK $ 1 HICNO2 $ 11 PROVID2 $ 6;
RETAIN MARK ' ';

      /**SET LAGGED VARIABLES**/

PROVID2=Y&YY.PRVID;
HICNO2=Y&YY.HICNO;
ADMDATE2=Y&YY.ADMIT;
DISDATE2=Y&YY.DISCH;
NOBS2=Y&YY.NOBS;
DDEST2=Y&YY.DDEST;

*********************************************************************;
* STEP 3. Read in MPAR1.
*********************************************************************;

SET &WORK1..MPAR1;

*********************************************************************;
* STEP 4. Set up algorithm to identify transfer claims.  Transfer
* claims are those claims where: 1) HICNOs are the same, and
*2) PROVIDs are not the same, and 3) admission date is equal to
* the previous discharge date or the previous discharge date plus 1,
* and 4) discharge destination equals 2.
*********************************************************************;

     /**SELECT TRANSFER CLAIMS**/

IF   ((HICNO2=Y&YY.HICNO)
AND  (PROVID2 NE Y&YY.PRVID)
AND  (DISDATE2 LE Y&YY.ADMIT LE DISDATE2+1)
AND  (DDEST2=2))
THEN DO;
    IF MARK=' ' THEN DO;
        CASENO+1;
        COUNTS+1;
        MARK='1';
        NEWOBS=NOBS2;
        OUTPUT;
        COUNTS+1;
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
    END;
    ELSE IF MARK='1' THEN DO;
        COUNTS+1;
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
    END;
END;
```

```
ELSE DO;
    COUNTS=0;
    MARK=' ';
END;
RUN;

*********************************************************************;
* STEP 5. Proc contents and proc print on TRANSXX file.
*********************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006) -- CONTINUED

```
PROC CONTENTS DATA=&OUT2..TRANS&YY DIRECTORY;
TITLE "YEAR &YY TRANSFER CLAIMS ID FILE";
RUN;
PROC PRINT DATA=&OUT2..TRANS&YY(OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;

PROC DATASETS LIB=&WORK1 NOLIST;
DELETE MPAR1;
RUN;

     /**CHECK BASIC INFO ON TRANSFER CLAIMS**/

*********************************************************************;
* STEP 6. Frequency on CASENO variable and output to a temp file
* (TEMPCNT).  Run means on the counts in the temp file.
*********************************************************************;

PROC FREQ DATA=&OUT2..TRANS&YY NOPRINT;
TABLES CASENO / OUT=TEMPCNT(DROP=PERCENT);
RUN;
PROC MEANS DATA=TEMPCNT N MIN MAX MEAN STD;
VAR COUNT;
TITLE "YEAR &YY TRANSFER CLAIMS STATISTICS";
TITLE2 "N=TOTAL # OF TRANSFER EPISODES";
TITLE3 "SUM=TOTAL # OF TRANSFER CLIAMS";
TITLE4 "MEAN=AVE # OF CLAIMS IN A TRANSFER EPISODE";
RUN;
PROC FREQ DATA=TEMPCNT;
TABLES COUNT;
TITLE2 "YEAR &YY FREQS FOR # OF CLMS IN A TRANSFER EPISODE";
RUN;


    /**SORT THESE RECORDS BY OBS #**/

*********************************************************************;
* STEP 7. Sort the TRANSXX file by observation number (YXXNOBS)
* and output to temp file (TOBS).
*********************************************************************;

PROC SORT DATA=&OUT2..TRANS&YY OUT=TOBS;
BY Y&YY.NOBS ;
RUN;

     /**MERGE THE TRANSFER RECORDS ONTO THE MEDPAR FILE**/
```

```
     /**TO PULL SOME OF THE MEDPAR INFO FOR THESE TRANSFERS**/

*******************************************************************;
* STEP 8. Merge TOBS and MEDPREXX file by YXXNOBS.  If match, pull
* relevant claim information for resolution of transfer cases.
*  This information is stored in temp file MPAR2.
*******************************************************************;

DATA &WORK2..MPAR2;
MERGE &IN1..MEDPRE&YY
      (IN=ONE KEEP=Y&YY.NOBS Y&YY.COIN Y&YY.BLOOD Y&YY.INPED
        Y&YY.DISCH  Y&YY.PMT Y&YY.TOTDM Y&YY.ADMIT)
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006) -- CONTINUED

      TOBS (IN=TWO)
      ;
BY Y&YY.NOBS;
IF ONE AND TWO;
RUN;

PROC DATASETS LIB=WORK NOLIST;
DELETE TOBS;
RUN;

   /**CLEAN UP THESE TRANSFER CLAIMS BY TAKING EACH UNIQUE GROUP**/
   /**AND RESOLVING ITS INFORMATION TO ONE RECORD**/

   /**SORT FILE BY CASENO AND COUNTS TO PREPARE TO CLEAN TRANSFERS**/

*******************************************************************;
* STEP 9. Sort MPAR2 by CASENO and COUNTS.  Proc print first
* 10 observations.
*******************************************************************;

PROC SORT DATA=&WORK2..MPAR2;
BY CASENO COUNTS;
RUN;

      /**TEST DUMP**/

PROC PRINT DATA=&WORK2..MPAR2(OBS=10);
BY CASENO;
TITLE "TEST PRINT OF SOME TRANSFER EPISODES IN Y&YY MEDPAR FILE";
RUN;


     /**THIS IS THE PRINCIPAL CLEANING DATA STEP FOR TRANSFERS**/

*******************************************************************;
* STEP 10. Create new temp file called MPAR1, which will contain
* the resolved transfer claims.  Set arrays for payment variables
* to be summed.  Read in MPAR2 file.
*******************************************************************;

DATA &WORK1..MPAR1
(DROP=I NEWMON: _ADMDATE LABEL="YEAR &YY CLEANED TRANSFERS"
 SORTEDBY=CASENO COUNTS);
```

```
RETAIN _ADMDATE;

       /**SET ARRAYS**/

ARRAY MONEY(5) Y&YY.COIN Y&YY.BLOOD Y&YY.INPED Y&YY.PMT Y&YY.TOTDM;
ARRAY NEWMON(5);

SET &WORK2..MPAR2;
BY CASENO;

     /**BEGIN CLEANING**/

*********************************************************************;
* STEP 11. Begin cleaning of transfers.  The admission date comes
* from the first claim in a series of transfer claims and discharge
```

```
* date is the discharge date of the last claim in a series of
* transfer claims.  Sum payment variables.
* All other variable values (including YXXNOBS) come from the
* last discharge claim in a series of transfer claims.
*********************************************************************;

IF FIRST.CASENO THEN DO;
   _ADMDATE=Y&YY.ADMIT;
   DO I = 1 TO 5;
     NEWMON(I)=MONEY(I);
   END;
END;
ELSE DO I = 1 TO 5;
    NEWMON(I)+MONEY(I);
END;

IF LAST.CASENO THEN DO;
   Y&YY.ADMIT=_ADMDATE;
   DO I = 1 TO 5;
       MONEY(I)=NEWMON(I);
   END;
   OUTPUT;
END;
RUN;

PROC PRINT DATA=&WORK1..MPAR1(OBS=10);
BY CASENO;
TITLE1 "PRINT 10 RECORDS OF RESOLVED TRANSFER RECORDS FOR MEDAPR &YY";
RUN;

*********************************************************************;
* STEP 12. Sort TRANSXX dataset by YXXNOBS and write out to temp
* file T1.  Sort MPAR1 file by YXXNOBS.
*********************************************************************;

PROC SORT DATA=&OUT2..TRANS&YY(KEEP=Y&YY.NOBS) OUT=T1;
BY Y&YY.NOBS;
RUN;
PROC SORT DATA=&WORK1..MPAR1(DROP=CASENO COUNTS);
BY Y&YY.NOBS;
RUN;
```

```
*********************************************************************;
* STEP 13. Merge on resolved claims and delete unresolved transfer
* claims.  Merge MEDPREXX, T1, and MPAR1 files.  If match to T1, but
* not to MPAR1, then delete these claims.  All other claims are kept.
* Create YXXTRF which equals '1' if a claim is a resolved transfer
* claim, else = '0' (non-transfer claim).  Create a new MedPAR file
* with the transfer claims resolved (MEDPRFXX).
*********************************************************************;

DATA &OUT1..MEDPRF&YY (LABEL="YEAR &YY MPAR F FILE"
                       SORTEDBY=Y&YY.NOBS);
LENGTH Y&YY.TRF $ 1;
MERGE &IN1..MEDPRE&YY(IN=ONE)
      T1(IN=TWO)
      &WORK1..MPAR1(IN=THREE);
BY Y&YY.NOBS;
IF ONE AND TWO AND NOT(THREE) THEN DELETE;
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2006) -- CONTINUED

Y&YY.TRF=PUT(THREE,1.);
LABEL Y&YY.TRF = "TRANSFER CLAIM INDICATOR" ;
RUN;

*********************************************************************;
* STEP 14. Proc contents and proc print on MEDPRFXX file.
*********************************************************************;

PROC CONTENTS DATA=&OUT1..MEDPRF&YY DIRECTORY ;
TITLE "NEW MEDPAR F YEAR &YY MEDPAR FILE";
RUN;
PROC PRINT DATA=&OUT1..MEDPRF&YY(OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;

         /**MACRO ENDS**/
%MEND YXXM2006;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM207)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :06/21/01
* EXEC DATE :07/11/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM207)
* PGMR      :MURATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99 (DISK)
*                     - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM2006.MEDPRFXX  (TAPE)
*                     - MEDPAR SAS FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM2004.DENOMCXX  (TAPE)
                      - MODIFIED M.1 DENOM FILE FOR YEAR XX
* OUTPUTS    D#MSQ.@MSQ0000.SAS.YXXM2007.MEDPRGXX   (TAPE)
*                      - NEW MEDPAR FILE FOR YEAR XX
*            D#MSQ.@MSQ0000.SAS.YXXM2007.CONTIXX   (DISK)
*                      - CONTIN. CARE CLAIMS OBS #S FOR YEAR XX
* PURPOSE   :CREATE A NEW MEDPAR SAS FILE BY MERGING ON
*            M.1 DENOMINATOR FILE INFORMATION: THIS FILE WILL BE
*            THE PRINCIPAL M.2/M.3 ANALYTIC FILE.
*            ALSO, IDENTIFY AND SAVE INFO TO DISK FOR CONTINUING
*            CARE CLAIMS. PLEASE SEE
*            MACRO YXXM2007 FOR MORE DETAILS.
*********************************************************************;

%INCLUDE IN3(YXXM2007)/SOURCE2;

      /**MACRO CALL**/
%YXXM2007(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007)

```
%MACRO YXXM2007(YY=99,IN1=IN1,IN2=IN2,IN3=IN3,
     WORK1=WORK1,WORK2=WORK2,OUT1=OUT1,OUT2=OUT2);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2007
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JUNE 24, 2001
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      FROM THE MEDPAR FILE CREATED IN PROGRAM YXXM2006 (XX=YEAR)
      CREATE A VARIABLE THAT MARKS CONTINUING CARE CLAIMS. SAVE
      THE CLAIM NUMBERS FOR THOSE CLAIMS TO A PERMANENT DISK FILE.
      THEN CREATE THE FINAL M.2 ANALYTIC FILE, WITH ALL APPROPRIATE
      M.1 DENOMINATOR DATA INCLUDED.

8  THERE ARE 7 KEYWORD PARAMETERS:
             YY IN1 IN2 WORK1 WORK2 OUT1 OUT2
YY
THIS SHOULD BE A 2-DIGIT YEAR.
E.G.: YY=99 (THIS IS THE DEFAULT)

IN1
IN3 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE CREATED IN PROGRAM YXXM2006 (XX=YEAR).
THE FILE IS: D#MSQ.@MSQ00.SAS.YXXM2006.MEDPRFXX
(XX=YEAR). LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT WAS CREATED IN PROGRAM YXXM2005 AND STORED
ON TAPE. IT IS SORTED BY VARIABLE YXXNOBS (XX=YEAR).
SIZE IS ABOUT 6GB (3 TAPES).
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IN2 DESIGNATES THE SAS LIBREF OR JCL DDNAME FOR THE MODIFIED
M.1.4 FILE CREATED IN PRPGRAM YXXM2004.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2004.DENOMCXX (XX=YEAR)
THE LAST QUALIFIER IS THE INTERNAL SAS NAME.
IT IS ASSUMED THAT THIS FILE WILL BE PUT ON TAPE. SIZE IS ABOUT
3GB.
E.G.: IN2=IN2  (THIS IS THE DEFAULT)

IN3
IN3 DESIGNATES THE SAS FILEREF FOR THE PARTITIONED DATASET THAT
HOLDS THE DIAGNOSTIC AND PROCEDURE GROUP PROC FORMAT STATEMENTS.
THE LOCATION IS:
D#MSQ.@MSQ00M2.PGMLIB99(SPGFORM,SDGFORM).
E.G.: IN3=IN3   (THIS IS THE DEFAULT)


WORK1
WORK1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR THE LIMITED
AND TEMPORARY MEDPAR FILE THAT THIS PROGRAM USES TO IDENTIFY
```

```
CONTINUING CARE CLAIMS.  IT IS A SUBSET OF THE FILE CREATED IN
PROGRAM YXXM2006 (XX=YEAR).
IT WILL BE ABOUT 500MB AND IS ASSUMED TO BE STORED
```

```
ON DISK. INTERNAL SAS NAME IS "MPAR1".
E.G.: WORK1=WORK1  (THIS IS THE DEFAULT)

WORK2
WORK2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR THE LIMITED
AND TEMPORARY M.1 DENOM FILE THAT THIS PROGRAM USES TO PROVIDE
INFO ON MATCHED CLAIMS.  IT WILL BE ABOUT 900MB
AND IS ASSUMED TO BE STORED ON DISK.
INTERNAL SAS NAME IS "DENOM2".
E.G.: WORK2=WORK2  (THIS IS THE DEFAULT)

OUT1
OUT1 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE MEDPAR FILE THAT THIS PROGRAM GENERATES.
THIS IS THE PRINCIPAL M.2 AND M.3 ANALYTIC FILE.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2007.MEDPRGXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON TAPE. SIZE IS ABOUT 4GB.
FILE SORTED BY YXXNOBS.
E.G.: OUT1=OUT1  (THIS IS THE DEFAULT)

OUT2
OUT2 DESIGNATES THE SAS LIBREF-JCL DDNAME FOR
THE FILE THAT HOLDS UNIQUE CLAIMS NUMBERS FOR ALL CONT. CARE CLAIMS.
THE FILE IS: D#MSQ.@MSQ0000.SAS.YXXM2007.CONTIXX (XX=YEAR).
LAST QUALIFIER IS THE INTERNAL SAS NAME. IT IS ASSUMED THAT THIS
FILE WILL BE STORED ON DISK. SIZE IS ABOUT 20MB.
E.G.: OUT2=OUT2  (THIS IS THE DEFAULT)

9  OTHER SYMBOLIC VARIABLES OF NOTE:  NONE

10  OTHER NOTES:
THE WORK LIBRARY SHOULD BE ABLE TO HOLD 3 APPROX. 20MB
SAS TEMPORARY FILES - "DENOM1". "MEDAPR1" AND "TEMP1".
PLEASE HEED THIS IN THE JCL.

11 EXAMPLE MACRO CALL:
%YXXM2007(YY=98)

*********************************************************************
*********************************************************************;

   /***CODE BEGINS***/

*********************************************************************;
* STEP 1. Sort latest MedPAR File (MEDPRFXX) by HICNO, admission
* date, discharge date, and PROVID to facilitate identification of
* continuing care claims.  Output to temp file MPAR1.
*********************************************************************;

PROC SORT DATA=&IN1..MEDPRF&YY
(KEEP=Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.PRVID
      Y&YY.NOBS Y&YY.DRGCD)
```

```
OUT=&WORK1..MPAR1;
BY Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH;
RUN;

  /**PULL THE UNIQUE OBSERVATION NUMBER**/
  /**FOR ALL CLAIMS THAT FORM CONTINUING CARE EPISODES**/
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007) -- CONTINUED

```
*********************************************************************;
* STEP 2. Create a permanent data file (CONTIXX) that contains the
* observation number (YXXNOBS).  Set the lengths of a few lagged
* variables and define all lagged variables.
*********************************************************************;

DATA &OUT2..CONTI&YY(KEEP=NEWOBS CASENO
                     RENAME=(NEWOBS=Y&YY.NOBS)
              LABEL="YEAR &YY CC OBS #")
              ;
LENGTH MARK $ 1 PROVID2 $ 6 HICNO2 $ 11;
RETAIN MARK ' ';

      /**SET LAGGED VARIABLES**/

PROVID2=Y&YY.PRVID;
HICNO2=Y&YY.HICNO;
DISDATE2=Y&YY.DISCH;
NOBS2=Y&YY.NOBS;
DRGCD2=Y&YY.DRGCD;

*********************************************************************;
* STEP 3. Read in MPAR1.
*********************************************************************;

SET &WORK1..MPAR1;

     /**SELECT CONTINUING CARE CLAIMS**/

*********************************************************************;
* STEP 4. Set up algorithm to identify continuing care claims.
* Continuing care claims are those claims where: 1) HICNOs are the
* same, and 2) PROVIDs are the same, and 3) admission date is equal
* to the previous discharge date or the previous discharge date
* plus 1, and 4) DRGCDs are the same.
*********************************************************************;

IF (
     (HICNO2=Y&YY.HICNO)
AND  (PROVID2=Y&YY.PRVID)
AND  (DISDATE2 LE Y&YY.ADMIT LE DISDATE2+1)
AND  (DRGCD2=Y&YY.DRGCD)
    )
THEN DO;
    IF MARK=' ' THEN DO;
        CASENO+1;
        MARK='1';
        NEWOBS=NOBS2;
        OUTPUT;
```

```
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
    END;
    ELSE IF MARK='1' THEN DO;
        NEWOBS=Y&YY.NOBS;
        OUTPUT;
    END;
END;
ELSE DO;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007) -- CONTINUED
```
    MARK=' ';
END;
RUN;

*******************************************************************;
* STEP 5. Proc contents and proc print on CONTIXX file.
*******************************************************************;

PROC CONTENTS DATA=&OUT2..CONTI&YY DIRECTORY;
TITLE "YEAR &YY CONTINUING CARE CLAIMS ID FILE";
RUN;
PROC PRINT DATA=&OUT2..CONTI&YY(OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;

PROC DATASETS LIB=&WORK1 NOLIST;
DELETE MPAR1;
RUN;


    /**SORT THESE RECORDS BY OBS #**/

*******************************************************************;
* STEP 6. Sort the CONTIXX file by observation number (YXXNOBS).
*******************************************************************;

PROC SORT DATA=&OUT2..CONTI&YY;
BY Y&YY.NOBS ;
RUN;

   /**PREPARE AND MERGE THE MODIFIED M.1 DENOMINATOR FILE ONTO**/
   /**THE MEDPAR FILE TO CREATE FINAL M.2 ANALYTIC FILE**/

*******************************************************************;
* STEP 7. sort DENOMCXX file by HICNO and write out to temp file
* DENOM1 keeping YXXDOBS and HICNO.  Also sort MEDPRFXX by YXXNOBS
* and HICNO and write out to temp file MEDPAR1.
*******************************************************************;

PROC SORT DATA=&IN2..DENOMC&YY
  (KEEP=Y&YY.DOBS Y&YY.HICNO)
  OUT=DENOM1;
BY Y&YY.HICNO;
RUN;

PROC SORT DATA=&IN1..MEDPRF&YY
  (KEEP=Y&YY.NOBS Y&YY.HICNO)
```

```
  OUT =MEDPAR1;
BY Y&YY.HICNO;
RUN;

*******************************************************************;
* STEP 8. Merge DENOM1 and MEDPAR1 by HICNO.  Matches are written
* out to temp file TEMP1.
*******************************************************************;

DATA TEMP1;
MERGE MEDPAR1(IN=ONE) DENOM1(IN=TWO);
BY Y&YY.HICNO;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007) -- CONTINUED
```

IF ONE AND TWO;
RUN;

PROC DATASETS LIB=WORK NOLIST;
DELETE MEDPAR1 DENOM1;
RUN;

*******************************************************************;
* STEP 9. Sort TEMP1 by YXXDOBS (Denominator file observation
* number).  Merge TEMP1 and DENOMCXX by YXXDOBS.  Matches are output
* to temp file DENOM2.
*******************************************************************;

PROC SORT DATA=TEMP1;
BY Y&YY.DOBS;
RUN;

DATA &WORK2..DENOM2;
MERGE &IN2..DENOMC&YY (IN=ONE)
      TEMP1 (IN=TWO DROP=Y&YY.HICNO)
       ;
BY Y&YY.DOBS;
IF ONE AND TWO;
RUN;

*******************************************************************;
* STEP 10. Sort DENOM2 by YXXNOBS (MedPAR file observation number).
* Read in SPG and SDG formats.  Merge DENOM2 and CONTIXX onto the
* MEDPRFXX file and write out to new MedPAR file (MEDPRGXX).
* Matches have all the Denominator File information from DENOMCXX.
* Create YXXCC that equals '1' if it is a continuing care claim,
* else equals '0'.  Create YXXSDG and YXXSPG1-YXXSPG6, the formatted
* diagnosis and procedure group variables.
*******************************************************************;

PROC SORT DATA=&WORK2..DENOM2 OUT=WORK3.DENOM2;
BY Y&YY.NOBS;
RUN;

%INCLUDE &IN3(SPGFORM) &IN3(SDGFORM) /SOURCE2;

DATA &OUT1..MEDPRG&YY
 (LABEL = "YEAR &YY MPAR G FILE" SORTEDBY = Y&YY.NOBS);
LENGTH Y&YY.CC $ 1 Y&YY.SDG $ 2 I 2;
```

```
ARRAY NEWSPG(6) $ 3 Y&YY.SPG1-Y&YY.SPG6;
ARRAY SPGS(6) $ Y&YY.SPR1-Y&YY.SPR6;
MERGE   &IN1..MEDPRF&YY(IN=ONE)
        WORK3.DENOM2
        &OUT2..CONTI&YY(IN=THREE DROP=CASENO);
BY Y&YY.NOBS;
IF ONE;
Y&YY.CC=PUT(THREE,1.);
Y&YY.SDG=PUT(Y&YY.DIA1,$SDG.);
IF Y&YY.SDG = 'NO' THEN Y&YY.SDG = ' ';
DO I = 1 TO 6;
 IF SPGS(I) NE ' ' THEN NEWSPG(I) = PUT(SPGS(I),$SPG.) ;
 ELSE NEWSPG(I) = ' ' ;
END;
LABEL
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2007) -- CONTINUED

```
Y&YY.CC   = 'CONTINUING CARE INDICATOR'
Y&YY.SDG  = 'SURVEILLANCE DIAG GROUP'
Y&YY.SPG1 = 'SURVEILLANCE PROC GROUP 1'
Y&YY.SPG2 = 'SURVEILLANCE PROC GROUP 2'
Y&YY.SPG3 = 'SURVEILLANCE PROC GROUP 3'
Y&YY.SPG4 = 'SURVEILLANCE PROC GROUP 4'
Y&YY.SPG5 = 'SURVEILLANCE PROC GROUP 5'
Y&YY.SPG6 = 'SURVEILLANCE PROC GROUP 6'
;
DROP I;
RUN;

*******************************************************************;
* STEP 11. Proc contents and proc print on MEDPRGXX file.
*******************************************************************;

PROC CONTENTS DATA=&OUT1..MEDPRG&YY DIRECTORY POSITION;
TITLE "NEW MEDPAR F YEAR &YY MEDPAR FILE";
RUN;
PROC PRINT DATA=&OUT1..MEDPRG&YY(OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;

         /**MACRO ENDS**/
%MEND YXXM2007;
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST92)

```
//TI36M292 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST92 EXEC M2POST,YEAR=92
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST93)

```
//TI36M293 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST93 EXEC M2POST,YEAR=93
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST94)

```
//TI36M294 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST94 EXEC M2POST,YEAR=94
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST95)

```
//TI36M295 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST95 EXEC M2POST,YEAR=95
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST96)

```
//TI36M296 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST96 EXEC M2POST,YEAR=96
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST97)

```
//TI36M297 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST97 EXEC M2POST,YEAR=97
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST98)

```
//TI36M298 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST98 EXEC M2POST,YEAR=98
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST99)

```
//TI36M299 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST99 EXEC M2POST,YEAR=99
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST00)

```
//TI36M200 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST00 EXEC M2POST,YEAR=00
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST01)

```
//TI36M201 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M2POST for a particular year (e.g., M2POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M2.PGMLIB99)
//M2POST01 EXEC M2POST,YEAR=01
```

D#MSQ.@MSQ00M2.PGMLIB99(M2POST)

```
//M2POST PROC
//*********************************************************************
//********CATALOGED PROCEDURE FOR M.2 TABLE GENERATION*****************
//*********************************************************************
//********ENTER SYMBOLIC PARM YEAR=(2 DIGIT YEAR)**********************
//*
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* REQ DATE  :02/26/02
//* EXEC DATE :02/26/02
//* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(M2POST)
//* PGMR      :MATTHEW URATO
//* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
//* INPUTS    :SEE SYSIN DD FOR EACH STEP
//* OUTPUTS   :SEE SYSIN DD FOR EACH STEP
//* PURPOSE   :CREATE M.2 TABLES FOR EACH YEAR
//*********************************************************************
//*
//YXXM2008 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR')
//WORK   DD SPACE=(CYL,(200,50)),DCB=(BLKSIZE=27648)
//WORK0  DD SPACE=(CYL,(200,50))
//WORK1  DD SPACE=(CYL,(200,50))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M2.PGMLIB99
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2007.MEDPRG&YEAR
//OUT DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2008.Y&YEAR.M2IN1,
//        SPACE=(CYL,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM208),DISP=SHR
//*********************************************************************
//YXXM2009 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(200,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2008.Y&YEAR.M2IN1
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1006.M1CNTS&YEAR
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2009.Y&YEAR.M2RAS,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM209),DISP=SHR
//*********************************************************************
//YXXM2010 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(200,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2008.Y&YEAR.M2IN1
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1006.M1CNTS&YEAR
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2010.Y&YEAR.M2ART,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM210),DISP=SHR
```

```
//*******************************************************************
//YXXM2011 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
```
D#MSQ.@MSQ00M2.PGMLIB99(M2POST) -- CONTINUED

```
//WORK   DD SPACE=(CYL,(200,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2008.Y&YEAR.M2IN1
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2011.Y&YEAR.M2CNT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2011.Y&YEAR.M2URT,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM211),DISP=SHR
//*******************************************************************
//YXXM2012 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(200,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M2.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2009.Y&YEAR.M2RAS
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2010.Y&YEAR.M2ART
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2011.Y&YEAR.M2CNT
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2011.Y&YEAR.M2URT
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2RAS.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2ART.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT3 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2CNT.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT4 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2URT.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM212),DISP=SHR
//*******************************************************************
//YXXM2013 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M2.PGMLIB99
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2RAS.XPT,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2ART.XPT,DISP=SHR
//IN3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2CNT.XPT,DISP=SHR
//IN4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2012.Y&YEAR.M2URT.XPT,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2013.RTFRAS&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2013.RTFART&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2013.RTFCNT&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2013.RTFURT&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
```

```
//SYSIN DD DSN=D#MSQ.@MSQ00M2.PGMLIB99(SYSM213),DISP=SHR
//*****************************************************************
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM208)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :07/19/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(YXXM2008)
* PGMR      :HELEN MARGULIS
*            (MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YXXM2007.MEDPRGXX - M2 DISCHARGE
            LEVEL ANALITICAL FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2008.YxxM2IN1
*                  - M.2 MAIN INTERMEDIATE FILE WITH COUNTERS (DISK)
* PURPOSE   :FROM SAS M.2 PRINCIPAL ANALYTIC FILE
*            CREATE A SAS FILE WITH COUNTS OF ADMISSIONS BY
*            SDG, AGE, SEX, SUBGROUPS
*            SEE MACRO YXXM2008 FOR MORE DETAILS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names and
*             a global macro variable YY.
*********************************************************************;

       /**INCLUDE MACRO**/

%INCLUDE IN0(YXXM2008)/ SOURCE2;

         /**MACRO CALL**/

%YXXM2008(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2008)

```
%MACRO YXXM2008(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2008
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2008)
3  WRITTEN BY:           HELEN MARGULIS
                         (MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JULY 16,2001
6  PROJECT:              MQMS
7  PURPOSE:
      THIS MACRO IS A MODIFIED VERSION OF MACRO YXXM1003
      WRITTEN BY MATTHEW URATO.

      THE PURPOSE IS AS FOLLOWS:
      CREATE THE ANALYTIC COUNT FILE FOR M.2. CALCULATING  # OF
      DISCHARGES FOR EACH ANALYTICAL SUBGROUP BY AGE/SEX FOR EACH
      SDG GROUP.

8  YEAR OF DATA IS SET IN THE MAIN PROGRAM YXXM2008 AS A GLOBAL MACRO
   VARIABLE YY.

9  INPUT AND OUTPUT IS DONE IN THE MAIN PROGRAM
*******************************************************************
*******************************************************************;
      /**LIST OF SUBGROUPS */

*******************************************************************;
* STEP 1. Assign major group variables to a macro variable to
* be able to do individual frequencies.
*******************************************************************;

%LET VARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL Y&YY.FAGE Y&YY.MDST
          Y&YY.RACE Y&YY.SEX Y&YY.STATE Y&YY.URBN;

              /**CODE BEGINS**/

      /**TO SORT INPUT FILE BY SDG, AGE AND SEX **/
            /** COMMENT: YxxM14=1 ALWAYS **/

*******************************************************************;
* STEP 2. Sort select variables from the Base Analytic MedPAR File
* by SDG, age group and sex for the M1-4 cohort.  Write out sorted
* variables to temp file WORK0.TEMP1 file.
*******************************************************************;

 PROC SORT DATA=IN1.MEDPRG&YY (KEEP=Y&YY.SDG &VARS Y&YY.M14
                               WHERE=(Y&YY.M14='1'))
           OUT=WORK0.TEMP1;
      BY Y&YY.SDG Y&YY.FAGE Y&YY.SEX;
 RUN;
        /**TO CREATE MIRRORED AGE & SEX VARS **/

*******************************************************************;
```

```
    * STEP 3. Read in Temp WORK0.TEMP1 file and add two variables
    * - mirrored age and sex variables. The mirrored variables are
    * created for ease of calculation for frequencies below.
    D#MSQ.@MSQ00M2.PGMLIB99(YXXM2008) -- CONTINUED

    ********************************************************************;

     DATA WORK1.TEMP1;
          SET WORK0.TEMP1(DROP=Y&YY.M14);
          Y&YY.FAGEM = Y&YY.FAGE;
          Y&YY.SEXM=Y&YY.SEX;
     RUN;

          /**SET UP LOOP FOR ALL 9 VARS**/

    ********************************************************************;
    * STEP 4. Create do loop for the nine major subgroups.
    ********************************************************************;

     %DO I = 1 %TO 9;            /**START LOOP **/

    ********************************************************************;
    * STEP 4a. Frequency on the nine major subgroups by SDG, age group,
    * and sex.  Each of these is written out respectively to nine
    * temp files (TEMP1 through TEMP9).
    ********************************************************************;

    /**OUTPUT COUNTS FOR EACH VARIABLE BY SDG/AGE/SEX **/
        PROC FREQ DATA=WORK1.TEMP1  NOPRINT;
             BY Y&YY.SDG Y&YY.FAGEM Y&YY.SEXM;
             TABLES %SCAN(&VARS,&I)
             / OUT=TEMP&I(DROP=PERCENT);
        RUN;

         /**NAME THE VALUES OF YXXVARS AND SET THEIR**/
         /**ATTRIBUTES APPROPRIATELY**/

    ********************************************************************;
    * STEP 4b. Modify the nine temp files adding a variable YXXVARS
    * that renames the values of the variables to take on the first
    * three letters of the variable name plus the variable value
    * applied as a suffix at the end (e.g., YXXSEX has values of
    * 1 and 2.  YXXVARS now has values of SEX1 and SEX2).
    ********************************************************************;

        DATA TEMP&I(DROP=%SCAN(&VARS,&I) RENAME=(COUNT=Y&YY.NSDG));
             LENGTH Y&YY.VARS $8;
             SET TEMP&I;
             Y&YY.VARS=
             COMPRESS("%SUBSTR(%SCAN(&VARS,&I),4,3)"!!%SCAN(&VARS,&I));
        RUN;

         /**COUNTS BY TOTAL FOR SDG/AGE/SEX **/

    ********************************************************************;
    * STEP 4c. Create an individual dataset (TEMP10) with a new
    *     variable     YXXVARS     that     has     a     value     of     TOTAL.
**********************************
```

```
*********************************;

    %IF &I=9 %THEN %DO;
       DATA TEMP10(DROP=Y&YY.NSDG  RENAME=(COUNT=Y&YY.NSDG ));
             SET TEMP9;
             BY  Y&YY.SDG Y&YY.FAGEM Y&YY.SEXM;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2008) -- CONTINUED
```
             LENGTH Y&YY.VARS $8;
             Y&YY.VARS='TOTAL';
             COUNT+Y&YY.NSDG;
             IF LAST.Y&YY.SEXM THEN DO;
                OUTPUT;
                COUNT=0;
             END;
       RUN;
    %END;
 %END;                    /**END LOOP **/

                     /**MACRO ENDS**/

            /**CONCATENATE THE COUNT FILES FOR ALL 9 VARS**/
           /**THIS IS THE PRINCIPAL OUTPUT FILE**/

*********************************************************************;
* STEP 5. Create a principal count file (YXXM2IN1) by stacking
* the ten temp files in alphabetical order.
*********************************************************************;

  DATA OUT.Y&YY.M2IN1(LABEL="PROJECT:MQMS, PROGRAM:Y&YY.M2008"
              SORTEDBY=Y&YY.SDG Y&YY.FAGE  y&YY.SEX  Y&YY.VARS);
       SET TEMP1 TEMP2 TEMP3 TEMP4 TEMP5 TEMP6 TEMP7 TEMP8
       TEMP10 TEMP9;
       BY  Y&YY.SDG Y&YY.FAGEM Y&YY.SEXM;
       LENGTH YEAR 2.;
       YEAR=&YY;
       LABEL
       Y&YY.VARS  = "SUBGROUP NAME"
       Y&YY.NSDG  = "M.2 discharge cnt by SUBGROUP,age,sex"
       Y&YY.FAGEM = "&YY. AGE GROUPS"
       Y&YY.SEXM  = "&YY. SEX"
   ;
       RENAME
       Y&YY.FAGEM  = Y&YY.FAGE
       Y&YY.SEXM   = Y&YY.SEX ;
  RUN;

  PROC DATASETS LIB=WORK NOLIST;
       DELETE TEMP1-TEMP10;
  RUN;

*********************************************************************;
* STEP 6. Proc print and proc contents on YXXM2IN1 file.
*********************************************************************;

PROC PRINT DATA=OUT.Y&YY.M2IN1(OBS=400);
       TITLE2 "M.2 MAIN INTERMEDIATE COUNT FILE";
  RUN;
```

```
  PROC CONTENTS DATA=OUT.Y&YY.M2IN1 DIRECTORY POSITION;
  RUN;

%MEND YXXM2008;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM209)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/07/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009)
* PGMR      :HELEN MARGULIS
*            (MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99  - MACRO LOCATION
*           -D#MSQ.@MSQ0000.SAS.YxxM1003.M1CNTSxx - file with
*            Denominator counters by subgroups
*           -D#MSQ.@MSQ0000.SAS.YxxM2008.YxxM2IN1 - intermediate file
*            with counters of admissions by age,sex,subgroups for
*            each SDG.
*           -D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99 - denominator counters
*            by age and sex for 1999 year. This file is the same for
*            all years.
*           -D#MSQ.@MSQ0000.SAS.YxxM1006.M1CNTSxx - denominator counters
*            by subgroups,age,sex
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2009.YxxM2RAS - age/sex adjusted
*            rates.
* PURPOSE   :create a SAS file with age/sex adjusted rates
*            of admissions by SDG and SUBGROUPS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*
* STEPS     :  1. create a table shell with all the rows and columns.
*              2. read in counters of discharges.
*              3. read in age/sex counters from 1999 Denominator and
*                 create weights.
*                 STEPS 4-7 ARE DONE TO CALCULATE AGE/SEX ADJUSTED
*                 RATES FOR ALL BUT AGE & SEX SUBGROUPS
*              4. read in counters from denominator.
*              5. create unadjusted rates per 1000 for each age/sex
*                 cell, using CTM14 counters.
*              6. adjust by 1999 weights.
*              7. sum on VARS/SDG level.
*                 STEPS 8-11 ARE DONE TO CALCULATE SEX ADJUSTED
*                 RATES AGE SUBGROUPS
*              8. sum denominator counters to adjust AGE groups by SEX
*              9. unadjusted rates per 1000
*             10. adjust by 1999 sex weights
*             11. sum on vars/sdg level
*                 STEPS 12-15 ARE DONE TO CALCULATE AGE ADJUSTED
*                 RATES SEX SUBGROUPS
*             12. sum denominator counters to adjust SEX groups by AGE
*             13. unadjusted rates per 1000
*             14. adjust by 1999 sex weights
*             15. sum on vars/sdg level
*             16. merge to the table shell to set missing cells to 0,
*                 write to output file.
*********************************************************************;
```

```
       /**INCLUDE MACROs**/
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM209) -- CONTINUED

```
%INCLUDE IN0(M2GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM2009)/ SOURCE2;

 ********************************************************************;
 * step 1. table shell
 ********************************************************************;
       /**MACRO CALL**/
%M2GENERL(VARNAME=SDG,VARCOUNT=95,VLENGTH=2,YY=&SYSPARM)

 ********************************************************************;
 * step 2-16. rates calculations . DATASETS: FINAL, FINALA, FINALS
 ********************************************************************;

       /**MACRO CALL**/

%YXXM2009(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(M2GENERL)

```
%MACRO M2GENERL(VARNAME=SDG,VARCOUNT=95,VLENGTH=2,YY=99);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           M2GENERL
2  LOCATION:             D#MSQ.@MSQ00.TI03LIB2(M2GENERL)
3  WRITTEN BY:           HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 6,2001
6  PROJECT:              MQMS
7  PURPOSE:
      THIS MACRO CREATES A TABLE SHELL WITH ONE ROW PER EACH
      ANALYTICAL SUBGROUP AND AS MANY COLUMNS AS THE NUMBER OF
      DIFFERENT VALUES OF THE VARIABLE VARNAME
      (FOR EXAMPLE FOR VARNAME=SDG THE TABLE WILL HAVE 95 COLUMNS).
      IT IS DESIGNED TO BE ABLE TO SET TO 0 ALL MISSING
      VARNAME COUNTER/RATE VALUES.

8  THIS MACRO COULD BE USED FOR ANY YEAR

9  PARAMETERS:
   THERE ARE 4 KEYWORD PARAMETERS:
   VARNAME, VARCOUNT, VLENGTH, YY

 VARNAME  - COLUMN VARIABLE BY WHICH STATISTICS IS DONE.
 VARCOUNT - NUMBER OF VALUES(COLUMNS) FOR VARNAME VARIABLE.
 VLENGTH  - LENGTH OF THE VARIABLE &VARNAME.
 YY       - THE TWO-DIGIT YEAR OF ANALYSIS.

10 EXAMPLE MACRO CALL:
%M2GENERL(VARNAME=SDG,VARCOUNT=95,VLENGTH=2,YY=99)

11 OTHER NOTES
IT IS ASSUMED THAT THE &VARNAME IS A CHARACTER VARIABLE.
THIS MACRO COULD BE USED FOR ANY YEAR OF DATA
*****************************************************************
*****************************************************************;

              /**CODE BEGINS**/

       /** TO GET LIST OF SUBGROUPS**/
  DATA INI1/VIEW=INI1;
       SET IN1.M1CNTS&YY;
  RUN;
  PROC SORT NODUPKEY DATA=INI1(KEEP=Y&YY.VARS) OUT=INI12;
       BY Y&YY.VARS;
  RUN;

       /** TO PRINT LIST OF SUBGROUPS  **/;
  PROC TRANSPOSE DATA=INI12 OUT=INI13 ;
       VAR Y&YY.VARS;
  RUN;
  PROC PRINT U DATA=INI13;
       TITLE "ALL THE SUBGROUPS";
  RUN;
```

```
      /** TO CREATE A TABLE SHELL  **/;
  DATA SHELL(KEEP=VARS &VARNAME);
```
D#MSQ.@MSQ00M2.PGMLIB99(M2GENERL) -- CONTINUED

```
      SET INI12(KEEP=Y&YY.VARS RENAME=(Y&YY.VARS=VARS));
      LENGTH &VARNAME $&VLENGTH.;
      DO I=1 TO &VARCOUNT;
         &VARNAME =LEFT(PUT(I,&VLENGTH..));
         OUTPUT;
      END;
  RUN;
  PROC SORT DATA=SHELL;
      BY VARS &VARNAME;
  RUN;
 %MEND M2GENERL;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009)

```
%MACRO YXXM2009(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2009
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009)
3  WRITTEN BY:           HELEN MARGULIS
                         (modified by Matthew Urato)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate age/sex adjusted rate
                         of admissions by SDG and SUBGROUPS.
                         Age subgroup rates are adjusted only by sex
                         and sex subgroup rates are adjusted only by
                         age.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

* STEPS     :  1. create a table shell with all the rows and columns.
               2. read in counters of discharges.
               3. read in age/sex counters from 1999 Denominator and
                  create weights.
                  STEPS 4-7 ARE DONE TO CALCULATE AGE/SEX ADJUSTED
                  RATES FOR ALL BUT AGE & SEX SUBGROUPS
               4. read in counters from denominator.
               5. create unadjusted rates per 1000 for each age/sex
                  cell, using CTM14 counters.
               6. adjust by 1999 weights.
               7. sum on VARS/SDG level.
                  STEPS 8-11 ARE DONE TO CALCULATE SEX ADJUSTED
                  RATES AGE SUBGROUPS
               8. sum denominator counters to adjust AGE groups by SEX
               9. unadjusted rates per 1000
              10. adjust by 1999 sex weights
              11. sum on vars/sdg level
                  STEPS 12-15 ARE DONE TO CALCULATE AGE ADJUSTED
                  RATES SEX SUBGROUPS
              12. sum denominator counters to adjust SEX groups by AGE
              13. unadjusted rates per 1000
              14. adjust by 1999 age weights
              15. sum on vars/sdg level
*             16. merge to the table shell to set missing cells to 0,
*                 write to output file.
*******************************************************************
*******************************************************************;

              /**CODE BEGINS**/

 *******************************************************************;
 * Step 1. Input counters from file YXXM2IN1 and write them out
 * to 3 files: 1 for age groups (INI2A), 1 for sex (INI2S), and one
 * for all the rest of the groups (INI2).  A fourth file (ERR1) was
 * created to write out counters where age or sex is missing as a
 * quality check.
```

```
*******************************************************************;

DATA INI2 INI2A INI2S ERR1;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009) -- CONTINUED
```
     SET IN2.Y&YY.M2IN1 ;
     IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
     ELSE
     IF SUBSTR(Y&YY.VARS,1,2)='FA' THEN OUTPUT INI2A;
     ELSE
     IF SUBSTR(Y&YY.VARS,1,2)='SE' THEN OUTPUT INI2S;
     ELSE OUTPUT INI2;
RUN;
PROC PRINT U DATA=ERR1(OBS=46);
   TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
RUN;

*******************************************************************;
* Step 1a. Sort the three files by appropriate variables to
* eventually perform merges to construct rates.
*******************************************************************;

PROC SORT DATA=INI2;
     BY  Y&YY.FAGE  Y&YY.SEX  Y&YY.VARS;
RUN;
PROC SORT DATA=INI2A;
     BY  Y&YY.SEX  Y&YY.VARS;
RUN;
PROC SORT DATA=INI2S;
     BY  Y&YY.FAGE Y&YY.VARS;
RUN;

*******************************************************************;
* Step 2. Create ASWGT99 (age/sex weight).  Write out 3 weight
* files: age/sex (WGTA_S), age (WGT_A), and sex (WGT_S) from the
* AGESEX99 file created in M1.  Proc print on the three files.
*******************************************************************;

DATA WGTA_S WGT_S WGT_A;
    IF _N_=1 THEN SET IN4.AGESEX99(WHERE=(Y99FAGE='' & Y99SEX='')
                                   RENAME=(Y99CNTS=TOTAL99));
    SET IN4.AGESEX99;
    ASWGT99=Y99CNTS/TOTAL99;
    IF Y99FAGE NE '' & Y99SEX NE '' THEN OUTPUT WGTA_S;
    ELSE
    IF Y99FAGE = '' & Y99SEX NE '' THEN OUTPUT WGT_S;
    ELSE
    IF Y99FAGE NE '' & Y99SEX = '' THEN OUTPUT WGT_A;
RENAME Y99FAGE=Y&YY.FAGE Y99SEX=Y&YY.SEX;
RUN;
PROC PRINT DATA=WGTA_S;
     TITLE "1999 AGE/SEX WEIGHTS";
RUN;
PROC PRINT DATA=WGT_S;
     TITLE "1999 SEX WEIGHTS";
RUN;
PROC PRINT DATA=WGT_A;
     TITLE "1999 AGE WEIGHTS";
```

```
RUN;

*********************************************************************;
* Step 3. Create age/sex adjusted rates for all subgroups except
* age and sex.
*********************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009) -- CONTINUED

```
*********************************************************************;
* Step 3a. Input counters from Denominator created in M.1.
*********************************************************************;

DATA INI3;
     SET IN3.M1ASCN&YY
         (KEEP=Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE  Y&YY.SEX );
RUN;

*********************************************************************;
* Step 3b. To construct the unadjusted rates per 1000 for each
* age/sex cell, merge the INI2 temp file to the Denominator temp
* count file INI3.  Create the unadjusted rate per 1000 (YXXRSDG)
* and output to TEMP1.
*********************************************************************;

DATA TEMP1 ERR2;
     MERGE INI3(IN=IN2)
           INI2(IN=IN1);
     BY  Y&YY.FAGE  Y&YY.SEX Y&YY.VARS;
     IF IN1 & NOT IN2 THEN OUTPUT ERR2;
     ELSE
     IF IN1 & IN2 THEN DO;
        Y&YY.RSDG = 1000* (Y&YY.NSDG/Y&YY.CTM14);
        OUTPUT TEMP1;
     END;
RUN;
PROC PRINT U DATA=ERR2(OBS=46);
   TITLE "!!! NB !!!!: no cells in denominator counter file";
RUN;

*********************************************************************;
* Step 3c. Add a new variable (YXXASSDG)  - the age/sex adjusted
* SDG rate - and output to TEMP2.
*********************************************************************;

DATA TEMP2 ERR3;
     MERGE TEMP1(IN=IN1)
           WGTA_S(IN=IN2);
     BY  Y&YY.FAGE  Y&YY.SEX ;
     IF IN1 & NOT IN2 THEN OUTPUT ERR3;
     ELSE
     IF IN2 & IN1 THEN DO;
        Y&YY.ASSDG = Y&YY.RSDG *ASWGT99;
        OUTPUT TEMP2;
     END;
RUN;
PROC PRINT U DATA=ERR3(OBS=46);
   TITLE "!!! NB !!!!: no weight in 1999 denominator";
```

```
RUN;

*******************************************************************;
* Step 3d: Sum the above rate (YXXASSDG) by YXXVARS, YXXSDG,
* and YEAR and output to FINAL.
*******************************************************************;

PROC SUMMARY DATA=TEMP2 NWAY;
     CLASS  Y&YY.VARS Y&YY.SDG YEAR;
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009) -- CONTINUED

     VAR Y&YY.ASSDG ;
     OUTPUT OUT=FINAL(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.ASSDG) SUM=;
RUN;

*******************************************************************;
* Step 4: Create sex adjusted rates for age subgroups.
*******************************************************************;

*******************************************************************;
* Step 4a: Sum the Denominator File counters to adjust age groups
* by sex and write out to INI3A.
*******************************************************************;

PROC SUMMARY NWAY DATA=INI3;
     CLASS  Y&YY.SEX Y&YY.VARS ;
     VAR Y&YY.CTM14 ;
     OUTPUT OUT=INI3A(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.SEX)
            SUM=;
RUN;

*******************************************************************;
* Step 4b. Create unadjusted rates per 1000 (YXXRSDG) and output
* to TEMP1A.
*******************************************************************;

DATA TEMP1A;
     MERGE INI3A(IN=IN2)
           INI2A(IN=IN1);
     BY  Y&YY.SEX Y&YY.VARS;
     IF IN1 & IN2 THEN DO;
        Y&YY.RSDG = 1000* (Y&YY.NSDG/Y&YY.CTM14);
        OUTPUT TEMP1A;
     END;
RUN;

*******************************************************************;
* Step 4c. Adjust rates by the 1999 sex weights (YXXASSDG) and
* output to TEMP2A.
*******************************************************************;

DATA TEMP2A;
     MERGE TEMP1A(IN=IN1)
           WGT_S(IN=IN2);
     BY  Y&YY.SEX ;
     IF IN2 & IN1 THEN DO;
        Y&YY.ASSDG = Y&YY.RSDG *ASWGT99;
        OUTPUT TEMP2A;
```

```
     END;
 RUN;

 ************************************************************************;
 * Step 4d. Sum the above rate (YXXASSDG) by YXXVARS, YXXSDG,
 * and YEAR and output to FINALA.
 ************************************************************************;

 PROC SUMMARY DATA=TEMP2A NWAY;
      CLASS  Y&YY.VARS Y&YY.SDG YEAR;
      VAR Y&YY.ASSDG ;
      OUTPUT OUT=FINALA(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.ASSDG) SUM=;
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009) -- CONTINUED

 RUN;

 ************************************************************************;
 * Step 5. Create age adjusted rates for sex subgroups.
 ************************************************************************;
 ************************************************************************;
 * Step 5a. Sum the Denominator File counters to adjust sex groups
 * by age and write out to INI3S.
 ************************************************************************;

 PROC SUMMARY NWAY DATA=INI3;
      CLASS  Y&YY.FAGE Y&YY.VARS ;
      VAR Y&YY.CTM14 ;
      OUTPUT OUT=INI3S(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE)
             SUM=;
 RUN;

 ************************************************************************;
 * Step 5b. Create unadjusted rates per 1000 (YXXRSDG) and output
 * to TEMP1S.
 ************************************************************************;

 DATA TEMP1S;
      MERGE INI3S(IN=IN2)
            INI2S(IN=IN1);
      BY  Y&YY.FAGE Y&YY.VARS;
      IF IN1 & IN2 THEN DO;
         Y&YY.RSDG = 1000* (Y&YY.NSDG/Y&YY.CTM14);
         OUTPUT TEMP1S;
      END;
 RUN;

 ************************************************************************;
 * Step 5c. Adjust rates by the 1999 age weights (YXXASSDG) and
 * output to TEMP2S.
 ************************************************************************;

 DATA TEMP2S;
      MERGE TEMP1S(IN=IN1)
            WGT_A(IN=IN2);
      BY  Y&YY.FAGE ;
      IF IN2 & IN1 THEN DO;
         Y&YY.ASSDG = Y&YY.RSDG *ASWGT99;
         OUTPUT TEMP2S;
```

```
     END;
 RUN;

 ******************************************************************;
 * Step 5d. Sum the above rate (YXXASSDG) by YXXVARS, YXXSDG, and
 * YEAR and output to FINALS.
 ******************************************************************;

 PROC SUMMARY DATA=TEMP2S NWAY;
      CLASS  Y&YY.VARS Y&YY.SDG YEAR;
      VAR Y&YY.ASSDG ;
      OUTPUT OUT=FINALS(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.ASSDG) SUM=;
 RUN;

 ******************************************************************;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2009) -- CONTINUED
```
 * Step 6. Create ALLS that puts all the rates (FINAL, FINALA, and
 * FINALS) together in one file.
 ******************************************************************;

 DATA ALLS;
      SET FINAL FINALA FINALS;
      BY Y&YY.VARS Y&YY.SDG ;
 RUN;

 ******************************************************************;
 * Step 7. Merge ALLS and the table shell from M2GENERL (SHELL) by
 * YXXVARS and YXXSDG and output to YXXM2RAS.  Values in SHELL that
 * do not have a match to ALLS are set to 0.
 ******************************************************************;

 DATA OUT1.Y&YY.M2RAS(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M2009"
                      SORTEDBY=Y&YY.VARS Y&YY.SDG )
      ERR4;
      MERGE ALLS(IN=IN1)
            SHELL(IN=IN2 RENAME=(SDG=Y&YY.SDG VARS=Y&YY.VARS));
      BY Y&YY.VARS Y&YY.SDG ;
      IF IN2 & NOT IN1 THEN Y&YY.ASSDG = 0;
      ELSE
      IF IN1 & NOT IN2 THEN OUTPUT ERR4;
      OUTPUT OUT1.Y&YY.M2RAS;
 run;
 PROC PRINT U DATA=ERR3(OBS=46);
    TITLE "!!! NB !!!!: no column or row in the table shell";
 RUN;
 PROC PRINT U DATA=OUT1.Y&YY.M2RAS(OBS=46);
    TITLE "M2: AGE/SEX ADJUSTED RATES";
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M2RAS;
 RUN;

                     /**MACRO ENDS**/
%MEND YXXM2009;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM210)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/07/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(YXXM2010)
* PGMR      :HELEN MARGULIS
*            (MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YxxM2008.YxxM2IN1 - intermediate file
*            with counters of admissions by age,sex,subgroups for
*            each SDG
*            D#MSQ.@MSQ0000.SAS.YxxM1004.AGESEX99 - denominator counters
*            by age and sex for 1999 year. This file is the same for
*            all years.
*            D#MSQ.@MSQ0000.SAS.YxxM1006.M1CNTSxx - denominator counters
*            by subgroups,age,sex
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2010.YxxM2ART - age adjusted
*            rates
* PURPOSE   :create a SAS file with age adjusted rate
*            of admissions  by SDG and SUBGROUPS.
*            Age subgroup rates are unadjusted.
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*
* STEPS     : 1. create table shell with all the rows and columns.
*             2. read in counters of discharges and sum them up on
*                AGE*VARS*SDG level.
*             3. read in counters from denominator and sum them up on
*                AGE*VARS level.
*             4. read in age counters from 1999 Denominator and
*                create weights
*             5. create unadjusted rates per 1000 for each age cell,
*                using CTM14 counters.
*             6. adjust by 1999 age weights all but age subroupgs.
*             7. sum on VARS/SDG level.
*             8. merge to the table shell to set missing cells to 0,
*                write to output file.
*********************************************************************;

       /**INCLUDE MACROs**/

%INCLUDE IN0(M2GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM2010)/ SOURCE2;

 *********************************************************************;
 * step 1. table shell
 *********************************************************************;
       /**MACRO CALL**/
%M2GENERL(VARNAME=SDG,VARCOUNT=95,VLENGTH=2,YY=&SYSPARM)

 *********************************************************************;
```

```
 * steps 2-7 rate calculation. dataset FINAL
 ************************************************************************;
       /**MACRO CALL**/
```
D#MSQ.@MSQ00M2.PGMLIB99(SYSM210) -- CONTINUED

```
%YXXM2010(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2010)

```
%MACRO YXXM2010(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2010
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2010)
3  WRITTEN BY:           HELEN MARGULIS
                         (modified by Matthew Urato)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate age adjusted rate
                         of admissions by SDG and SUBGROUPS.
                         Age subgroup rates are unadjusted.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

9  STEPS     : (first step is done in the main program)
*             2. read in counters of discharges and sum them up on
*                AGE*VARS*SDG level.
*             3. read in counters from denominator and sum them up on
*                AGE*VARS level.
*             4. read in age counters from 1999 Denominator and
*                create weights
*             5. create unadjusted rates per 1000 for each age cell,
*                using CTM14 counters.
*             6. adjust by 1999 age weights
*             7. sum on VARS/SDG level

*******************************************************************
*******************************************************************;

              /**CODE BEGINS**/

 *******************************************************************;
 * Step 1. Input counters from file YXXM2IN1 and write them out to
 * INI2.
 *******************************************************************;

 DATA INI2 ERR1;
      SET IN2.Y&YY.M2IN1 ;
      IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
      ELSE OUTPUT INI2;
 RUN;
 PROC PRINT U DATA=ERR1(OBS=46);
    TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
 RUN;

 *******************************************************************;
 * Step 2. Sum the count variable (YXXNSDG) by YXXFAGE, YXXVARS,
 * YXXSDG, and YEAR and output to INI21.
 *******************************************************************;

 PROC SUMMARY DATA=INI2 NWAY;
      CLASS  Y&YY.FAGE Y&YY.VARS Y&YY.SDG YEAR;
```

```
     VAR Y&YY.NSDG ;
     OUTPUT OUT=INI21(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.NSDG Y&YY.FAGE)
            SUM=;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2010) -- CONTINUED

```
 RUN;

 ********************************************************************;
 * Step 3. Sum the Denominator File counters by age group and
 * YXXVARS in preparation for the creation of age weights and write
 * out to INI31.
 ********************************************************************;

 PROC SUMMARY NWAY DATA=IN4.M1ASCN&YY
                  (KEEP=Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE  Y&YY.SEX );
      CLASS  Y&YY.FAGE Y&YY.VARS ;
      VAR Y&YY.CTM14 ;
      OUTPUT OUT=INI31(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE)
             SUM=;
 RUN;

 ********************************************************************;
 * Step 4. Create AWGT99 (age weight).  Write out an age weight
 * file WGTA). Proc print on WGTA.
 ********************************************************************;

 DATA WGTA;
     IF _N_=1 THEN SET IN3.AGESEX99(WHERE=(Y99FAGE='' & Y99SEX='')
                                    RENAME=(Y99CNTS=TOTAL99));
     SET IN3.AGESEX99(WHERE=(Y99FAGE NE '' & Y99SEX ='' ));
     AWGT99=Y99CNTS/TOTAL99;
 RENAME Y99FAGE=Y&YY.FAGE Y99SEX=Y&YY.SEX;
 RUN;
 PROC PRINT DATA=WGTA;
      TITLE "1999 AGE WEIGHTS";
 RUN;

 ********************************************************************;
 * Step 5. To construct the unadjusted rates per 1000 for each age
 * cell, merge the INI21 temp file to the Denominator temp count
 * file INI31.  Create the unadjusted rate per 1000 (YXXRSDG) and
 * output to TEMP1.
 ********************************************************************;

 DATA TEMP1 ERR2;
      MERGE INI31(IN=IN2)
            INI21(IN=IN1);
      BY  Y&YY.FAGE   Y&YY.VARS;
      IF IN1 & NOT IN2 THEN OUTPUT ERR2;
      ELSE DO;
         IF Y&YY.NSDG=. THEN Y&YY.NSDG=0;
         Y&YY.RSDG = 1000* (Y&YY.NSDG/Y&YY.CTM14);
         OUTPUT TEMP1;
      END;
 RUN;
 PROC PRINT U DATA=ERR2(OBS=46);
    TITLE "!!! NB !!!!: no cells in denominator counter file";
 RUN;
```

```
*******************************************************************;
* Step 6. Add a new variable (YXXAASDG)  - the age adjusted SDG
* rate - and output to TEMP2.  Age group rates remain unadjusted.
*******************************************************************;
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2010) -- CONTINUED

```
DATA TEMP2 ERR3;
     MERGE TEMP1(IN=IN1)
           WGTA(IN=IN2);
     BY  Y&YY.FAGE   ;
     IF IN1 & NOT IN2 THEN OUTPUT ERR3;
     ELSE
     IF IN1 & IN2 THEN DO;
        IF SUBSTR(Y&YY.VARS,1,2)='FA' THEN
        Y&YY.AASDG = Y&YY.RSDG ; *UNADJUSTED RATE FOR AGE GROUPS;
        ELSE
        Y&YY.AASDG = Y&YY.RSDG *AWGT99;
        OUTPUT TEMP2;
     END;
RUN;
PROC PRINT U DATA=ERR3(OBS=46);
   TITLE "!!! NB !!!!: no weight in 1999 denominator";
RUN;

*******************************************************************;
* Step 7. Sum the above rate (YXXAASDG) by YXXVARS, YXXSDG, and
* YEAR and output to FINAL.
*******************************************************************;

PROC SUMMARY DATA=TEMP2 NWAY;
     CLASS  Y&YY.VARS Y&YY.SDG YEAR;
     VAR Y&YY.AASDG ;
     OUTPUT OUT=FINAL(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.AASDG) SUM=;
RUN;

*******************************************************************;
* Step 8. Merge FINAL and the table shell from M2GENERL (SHELL) by
* YXXVARS and YXXSDG and output to YXXM2ART.  Values in SHELL that
* do not have a match to FINAL are set to 0.
*******************************************************************;

DATA OUT1.Y&YY.M2ART(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M2010"
                     SORTEDBY=Y&YY.VARS Y&YY.SDG )
     ERR4;
     MERGE FINAL(IN=IN1)
           SHELL(IN=IN2 RENAME=(SDG=Y&YY.SDG VARS=Y&YY.VARS));
     BY Y&YY.VARS Y&YY.SDG ;
     IF IN2 & NOT IN1 THEN Y&YY.AASDG = 0;
     ELSE
     IF IN1 & NOT IN2 THEN OUTPUT ERR4;
     OUTPUT OUT1.Y&YY.M2ART;
run;
PROC PRINT U DATA=ERR4(OBS=46);
   TITLE "!!! NB !!!!: no column or row in the table shell";
RUN;
PROC PRINT U DATA=OUT1.Y&YY.M2ART(OBS=46);
```

```
   TITLE "M2: AGE ADJUSTED RATES";
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M2ART;
 RUN;
                    /**MACRO ENDS**/
%MEND YXXM2010;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM211)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/07/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(YXXM2011)
* PGMR      :HELEN MARGULIS
*            (MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M2.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YxxM1003.M1CNTSxx - file with
*            Denominator counters by subgroups
*            D#MSQ.@MSQ0000.SAS.YxxM2008.YxxM2IN1 - intermediate file
*            with counters of admissions by age,sex,subgroups for
*            each SDG
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2011.YxxM2CNT - file with counter
*            of admissions for yeach SDG and each subgroup
*            D#MSQ.@MSQ0000.SAS.YxxM2011.YxxM2URT - file with
*            unadjusted admission rates for each SDG and each subgroup
* PURPOSE   :create SAS files with counters and unadjusted rate
*            of admissions by SDG and SUBGROUPS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*
* STEPS     : 1. create table shell with all the rows and columns.
*             2. read in counter of discharges.
*             3. sum to vars*sdg level and create final count file.
*             4. merge to the table shell to set missing cells to 0,
*                write to output file.
*             5. read in counters from Denominator and
*                create unadjusted rates per 1000, write to output
*                file.
*********************************************************************;

       /**INCLUDE MACRO**/

%INCLUDE IN0(M2GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM2011)/ SOURCE2;

 ********************************************************************;
 * step 1. table shell
 ********************************************************************;
       /**MACRO CALL**/
%M2GENERL(VARNAME=SDG,VARCOUNT=95,VLENGTH=2,YY=&SYSPARM);

 ********************************************************************;
 * step 2-5. rates calculation
 ********************************************************************;

%YXXM2011(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2011)

```
%MACRO YXXM2011(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2011
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2011)
3  WRITTEN BY:           HELEN MARGULIS
                         (modified by Matthew Urato)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate unadjusted rates
                         of admissions by SDG and SUBGROUPS.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

* STEPS     : 1. create table shell with all the rows and columns.
*             2. read in counter of discharges.
*             3. sum to vars*sdg level and create final count file.
*             4. merge to the table shell to set missing cells to 0,
*                write to output file.
*             5. read in counters from Denominator and
*                create unadjusted rates per 1000, write to output
*                file.

*******************************************************************
*******************************************************************;
              /**CODE BEGINS**/

 *******************************************************************;
 * Step 1. Input counters from file YXXM2IN1 and write them out to
 * INI2.
 *******************************************************************;

 DATA INI2 ERR1;

      SET IN2.Y&YY.M2IN1 ;
      IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
      ELSE OUTPUT INI2;
 RUN;
 PROC PRINT U DATA=ERR1(OBS=46);
    TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
 RUN;

 *******************************************************************;
 * Step 2. Sum the count variable (YXXNSDG) by YXXVARS, YXXSDG,
 * and YEAR and output to INI21.
 *******************************************************************;

 PROC SUMMARY DATA=INI2 NWAY;
      CLASS  Y&YY.VARS Y&YY.SDG YEAR;
      VAR Y&YY.NSDG ;
      OUTPUT OUT=INI21(KEEP=YEAR Y&YY.SDG Y&YY.VARS Y&YY.NSDG Y&YY.FAGE)
             SUM=;
 RUN;
```

```
 PROC PRINT U  DATA=INI21(OBS=46);
      TITLE "data check: DISCHARGE FILE SUMMED ON VARS*SDG LEVEL";
 RUN;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2011) -- CONTINUED

```
 ********************************************************************;
 * Step 3. Merge INI21 and the table shell from M2GENERL (SHELL) by
 * YXXVARS and YXXSDG and output to YXXM2CNT.  Values in SHELL that
 * do not have a match to INI21 are set to 0.
 ********************************************************************;

 DATA OUT1.Y&YY.M2CNT(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M2011"
                      SORTEDBY=Y&YY.VARS Y&YY.SDG )
      ERR2;
      MERGE INI21(IN=IN1)
            SHELL(IN=IN2 RENAME=(SDG=Y&YY.SDG VARS=Y&YY.VARS));
      BY Y&YY.VARS Y&YY.SDG ;
      IF IN2 & NOT IN1 THEN Y&YY.NSDG = 0;
      ELSE
      IF IN1 & NOT IN2 THEN OUTPUT ERR2;
      OUTPUT OUT1.Y&YY.M2CNT;
 RUN;
 PROC PRINT U DATA=ERR2(OBS=46);
    TITLE "!!! NB !!!!: no column or row in the table shell";
 RUN;
 PROC PRINT U DATA=OUT1.Y&YY.M2CNT(OBS=46);
    TITLE "M2: FINAL COUNTERS FILE";
 RUN;

 ********************************************************************;
 * Step 4. To construct the unadjusted rates per 1000 for each
 * cell, merge the YXXM2CNT file to the Denominator count file
 * M1CNTSXX.  Create the unadjusted rate per 1000 (YXXRSDG) and
 * output to YXXM2URT.
 ********************************************************************;

 DATA OUT2.Y&YY.M2URT(KEEP=YEAR Y&YY.VARS Y&YY.SDG  Y&YY.RSDG
                      LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M2011"
                      SORTEDBY=Y&YY.VARS Y&YY.SDG )
      ERR3 ERR4;
      MERGE IN1.M1CNTS&YY (KEEP=Y&YY.VARS Y&YY.CTM14 IN=IN2)
            OUT1.Y&YY.M2CNT(IN=IN1);
      BY Y&YY.VARS;
      IF IN1 & NOT IN2 THEN OUTPUT ERR3;
      ELSE
      IF IN2 & NOT IN1 THEN OUTPUT ERR4;
      ELSE DO;
         Y&YY.RSDG = 1000* (Y&YY.NSDG/Y&YY.CTM14);
         OUTPUT OUT2.Y&YY.M2URT;
      END;
 RUN;
 PROC PRINT U DATA=ERR3(OBS=46);
 TITLE "!!! NB !!!!: DENOMINATOR FILE ERROR: subgroup is missing";
 RUN;
 PROC PRINT U DATA=ERR4(OBS=46);
 TITLE "!!! NB !!!!: COUNTER DO NOT HAVE ALL ROWS/COLUMNS";
 RUN;
 PROC PRINT U DATA=OUT2.Y&YY.M2URT(OBS=46);
```

```
    TITLE "M2: FINAL RATE FILE";
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M2CNT;
      TITLE "M2: DISCHARGE COUNTERS FILE";
 RUN;
 PROC CONTENTS DATA=OUT2.Y&YY.M2URT;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2011) -- CONTINUED
```

      TITLE "M2: DISCHARGE RATE FILE";
RUN;

%MEND YXXM2011;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM212)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/13/01
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(YXXM2012)
* PGMR      :HELEN MARGULIS
*            (MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.YxxM2009.YxxM2RAS
*                          -age/sex adjusted rates
*            D#MSQ.@MSQ0000.SAS.YxxM2010.YxxM2ART
*                          -age adjusted rates
*            D#MSQ.@MSQ0000.SAS.YxxM2011.YxxM2CNT
*                          -counters of admissions
*            D#MSQ.@MSQ0000.SAS.YxxM2011.YxxM2URT
*                          -unadjusted rates
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2RAS.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2ART.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2CNT.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2URT.XPT
* PURPOSE   :create SAS transport files to download to PC
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names and
*             a global macro variable YY.
*             Global macro variable NCOL gives the number of SDG
*             groups which will be transpose to number of columns.
*********************************************************************;

        /**INCLUDE MACRO**/;

 %INCLUDE IN0(YXXM2012)/ SOURCE2;
       /**MACRO CALL**/

%YXXM2012(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2012)

```
%MACRO YXXM2012(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM2012
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM2012)
3  WRITTEN BY:           HELEN MARGULIS
                         (MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              create SAS transport file to download to PC

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

*******************************************************************
*******************************************************************;
LIBNAME OUT1 XPORT;
LIBNAME OUT2 XPORT;
LIBNAME OUT3 XPORT;
LIBNAME OUT4 XPORT;
%LET NCOL=95;

*******************************************************************;
* Step 1: Transpose the input datasets to have the subgroups in
* rows and the counts or rates as columns.
*******************************************************************;

 %MACRO TRANS(INF=,OUTF=,PREF=);
    DATA INI;
         SET &INF;
         SDGN = INPUT (Y&YY.SDG,2.);
    RUN;
    PROC SORT DATA=INI; BY Y&YY.VARS SDGN; RUN;
    PROC TRANSPOSE DATA=INI PREFIX=&PREF
         OUT=&OUTF(KEEP=Y&YY.VARS &PREF.1-&PREF.&NCOL);
         BY Y&YY.VARS;
         VAR Y&YY.&PREF;
    RUN;
 %MEND TRANS;

*******************************************************************;
* Step 2: Create SAS transport files.
*******************************************************************;
*******************************************************************;
 %TRANS(INF=IN1.Y&YY.M2RAS,OUTF=OUT1.Y&YY.M2RAS,PREF=ASSDG);
 %TRANS(INF=IN2.Y&YY.M2ART,OUTF=OUT2.Y&YY.M2ART,PREF=AASDG);
 %TRANS(INF=IN3.Y&YY.M2CNT,OUTF=OUT3.Y&YY.M2CNT,PREF=NSDG);
 %TRANS(INF=IN4.Y&YY.M2URT,OUTF=OUT4.Y&YY.M2URT,PREF=RSDG);

%MEND YXXM2012;
```

D#MSQ.@MSQ00M2.PGMLIB99(SYSM213)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :02/26/02
* EXEC DATE :02/26/02
* PGM       :D#MSQ.@MSQ00M2.PGMLIB99(SYSM213)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2RAS.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2ART.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2CNT.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM2012.YxxM2URT.XPT
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM2013.RTFRASxx
*            D#MSQ.@MSQ0000.SAS.YxxM2013.RTFARTxx
*            D#MSQ.@MSQ0000.SAS.YxxM2013.RTFCNTxx
*            D#MSQ.@MSQ0000.SAS.YxxM2013.RTFURTxx
* PURPOSE   :TO CREATE RICH TEXT FILES THAT CAN BE READ ON A
*            PC BY MICROSOFT WORD FROM THE XPORT TABLES CREATED
*            IN MACRO PROGRAM YXXM2012.
*********************************************************************;

        /**INCLUDE MACRO**/;

%INCLUDE IN0(YXXM2013)/ SOURCE2;

       /**MACRO CALL**/

%YXXM2013(YY=&SYSPARM)
```

D#MSQ.@MSQ00M2.PGMLIB99(YXXM2013)

```
%MACRO YXXM2013(YY=99,IN1=IN1,IN2=IN2,IN3=IN3,IN4=IN4,
                OUT1=OUT1,OUT2=OUT2,OUT3=OUT3,OUT4=OUT4);
%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM2013
2  LOCATION:             D#MSQ.@MSQ00M1.PGMLIB99(YXXM2013)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 FEB 01, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      PRINT OUT THE 4 MAJOR COUNT FILES FOR M.2 AND SAVE THE OUTPUT
      AS RICH TEXT FILES TO BE READ LOCALLY ON A PC WITH MICROSOFT
      WORD.

8  PARAMETERS:
      THERE ARE 9 KEYWORD PARAMETERS:
      YY IN1 IN2 IN3 IN4 OUT1 OUT2 OUT3 OUT4

YY
IS THE 2-DIGIT YEAR OF THE DENOMINATOR FILE BEING READ INTO SAS
E.G.:
YY=99 (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR A M.2 COUNT FILE CREATED IN PROGRAM
'YXXM2012' FOR AGE-SEX ADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM2012.YXXM2RAS.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IS THE SAS LIBREF-JCL DDNAME FOR A M.2 COUNT FILE CREATED IN PROGRAM
'YXXM2012' FOR AGE ADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM2012.YXXM2ART.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN3
IS THE SAS LIBREF-JCL DDNAME FOR A M.2 COUNT FILE CREATED IN PROGRAM
'YXXM2012' FOR RAW NUMERATOR COUNTS.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM2012.YXXM2CNT.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN4
IS THE SAS LIBREF-JCL DDNAME FOR A M.2 COUNT FILE CREATED IN PROGRAM
'YXXM2012' FOR UNADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM2012.YXXM2URT.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
```

FOR THE FILE IN IN1.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM2013.RTFRASXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2013) -- CONTINUED

OUT2
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN2.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM2013.RTFARTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT3
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN3.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM2013.RTFCNTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT4
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN4.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM2013.RTFURTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

9 OTHER MACRO VARIABLES OF NOTE: NONE

10 OTHER NOTES: NONE

11 EXAMPLE MACRO CALL:
%YXXM2013(YY=93)

```
**************************************************************************
**************************************************************************;

                  /**CODE BEGINS**/

*************************************************************************;
* Step 1. Replace YXXVARS values with descriptive label-like
* values to be used in tables.
*************************************************************************;

PROC FORMAT;

VALUE $GROUPS

'FAG1' ='0-54' 'FAG2'='55-64'
'FAG3' ='65-69' 'FAG4'='70-74'  'FAG5'='75-79'
'FAG6'='80-84' 'FAG7'='85-89' 'FAG8'='90-94'
'FAG9'='95-UP'
'SEX1'='MALE'
'SEX2'='FEMALE'
'RAC0'='UNKNOWN'
'RAC1'='WHITE'
'RAC2'='AFRICAN AMERICAN'
'RAC3'='OTHER'

'MDS1'='AGED WITHOUT ESRD'
'MDS2'='AGED WITH ESRD'
```

```
'MDS3'='DISABLED WITHOUT ESRD'
'MDS4'='DISABLED WITH ESRD'
'MDS5'='ESRD ONLY'

'DUA1'='ENROLLED IN MEDICAID'
'DUA0'='NOT ENROLLED IN MEDICAID'
```


```
'URB0'='RURAL'
'URB1'='URBAN'

'STA01'='ALABAMA'
'STA02'='ALASKA'
'STA03'='ARIZONA'
'STA04'='ARKANSAS'
'STA05'='CALIFORNIA'
'STA06'='COLORADO'
'STA07'='CONNECTICUT'
'STA08'='DELAWARE'
'STA09'='DISTRICT OF COLUMBIA'
'STA10'='FLORIDA'
'STA11'='GEORGIA'
'STA12'='HAWAII,GUAM'
'STA13'='IDAHO'
'STA14'='ILLINOIS'
'STA15'='INDIANA'
'STA16'='IOWA'
'STA17'='KANSAS'
'STA18'='KENTUCKY'
'STA19'='LOUISIANA'
'STA20'='MAINE'
'STA21'='MARYLAND'
'STA22'='MASSACHUSETTS'
'STA23'='MICHIGAN'
'STA24'='MINNESOTA'
'STA25'='MISSISSIPPI'
'STA26'='MISSOURI'
'STA27'='MONTANA'
'STA28'='NEBRASKA'
'STA29'='NEVADA'
'STA30'='NEW HAMPSHIRE'
'STA31'='NEW JERSEY'
'STA32'='NEW MEXICO'
'STA33'='NEW YORK'
'STA34'='NORTH CAROLINA'
'STA35'='NORTH DAKOTA'
'STA36'='OHIO'
'STA37'='OKLAHOMA'
'STA38'='OREGON'
'STA39'='PENNSYLVANIA'
'STA40'='PUERTO RICO'
'STA41'='RHODE ISLAND'
'STA42'='SOUTH CAROLINA'
'STA43'='SOUTH DAKOTA'
'STA44'='TENNESSEE'
'STA45'='TEXAS'
'STA46'='UTAH'
```

```
'STA47'='VERMONT'
'STA48'='VIRGIN ISLANDS'
'STA49'='VIRGINIA'
'STA50'='WASHINGTON'
'STA51'='WEST VIRGINIA'
'STA52'='WISCONSIN'
'STA53'='WYOMING'

'CDI0'='OUTSIDE U.S.(CD)'
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2013) -- CONTINUED
```
'CDI1'='NEW ENGLAND'
'CDI2'='MIDDLE ATLANTIC'
'CDI3'='EAST NORTH CENTRAL'
'CDI4'='WEST NORTH CENTRAL'
'CDI5'='SOUTH ATLANTIC'
'CDI6'='EAST SOUTH CENTRAL'
'CDI7'='WEST SOUTH CENTRAL'
'CDI8'='MOUNTAIN'
'CDI9'='PACIFIC'

'CRE0'='OUTSIDE U.S.(CR)'
'CRE1'='NORTHEAST'
'CRE2'='MIDWEST'
'CRE3'='SOUTH'
'CRE4'='WEST'
;

RUN;



*******************************************************************;
* Step 2. Set up do loop that reads through the 4 transport files.
* Read in SAS transport file and copy it into a standard SAS file.
* Print these files and save them as RTFs.
*******************************************************************;
OPTIONS LS=80 NODATE;

%LET ADJTYP =AGE_SEX_ADJUSTED AGE_ADJUSTED RAW_NUMERATORS UNADJUSTED;
%LET FILEN =M2RAS M2ART M2CNT M2URT;

%DO I = 1 %TO 4;


%LET ADJTYPE=%SCAN(&ADJTYP,&I);
%LET FILENM=%SCAN(&FILEN,&I);

LIBNAME IN&I XPORT;

PROC COPY IN=IN&I OUT=WORK;
RUN;

ODS RTF FILE=OUT&I TRANTAB=ASCII;

%IF &FILENM = M2CNT %THEN %DO;
TITLE "M2 DISCHARGES FOR YEAR &YY";
TITLE2 'FULL YEAR FFS MEDICARE ENROLLEES';
```

```
TITLE3 "&ADJTYPE";
%END;
%ELSE %DO;
TITLE "M2 DISCHARGE RATES PER 1000 FOR YEAR &YY";
TITLE2 'FULL YEAR FFS MEDICARE ENROLLEES';
TITLE3 "&ADJTYPE RATES";
%END;

PROC PRINT DATA=Y&YY&FILENM HEADING=V WIDTH=MIN ROUND;
FORMAT Y&YY.VARS $GROUPS.;
ID Y&YY.VARS;
RUN;
```
D#MSQ.@MSQ00M2.PGMLIB99(YXXM2013) -- CONTINUED

```
ODS RTF CLOSE;

%END;

        /**MACRO ENDS**/

%MEND YXXM2013;
```

PROGRAMS – M3

D#MSQ.@MSQ00M3.PGMLIB99(M3POST92)

```
//TI36M392 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST92 EXEC M3POST,YEAR=92
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST93)

```
//TI36M393 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST93 EXEC M3POST,YEAR=93
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST94)

```
//TI36M394 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST94 EXEC M3POST,YEAR=94
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST95)

```
//TI36M395 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST95 EXEC M3POST,YEAR=95
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST96)

```
//TI36M396 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST96 EXEC M3POST,YEAR=96
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST97)

```
//TI36M397 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST97 EXEC M3POST,YEAR=97
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST98)

```
//TI36M398 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST98 EXEC M3POST,YEAR=98
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST99)

```
//TI36M399 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST99 EXEC M3POST,YEAR=99
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST00)

```
//TI36M300 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST00 EXEC M3POST,YEAR=00
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST01)

```
//TI36M301 JOB (MSQ00,CHER),TI36MURATO,MSGCLASS=Q,CLASS=J
//*
//* Purpose:  This job executes the cataloged procedure
//*           M3POST for a particular year (e.g., M3POST92 for 1992).
//*           To run this job for other years, simply define the
//*           year of analysis in the YEAR parameter (2 digit year).
//*
//  JCLLIB ORDER=(D#MSQ.@MSQ00M3.PGMLIB99)
//M3POST01 EXEC M3POST,YEAR=01
```

D#MSQ.@MSQ00M3.PGMLIB99(M3POST)

```
//M3POST PROC
//*********************************************************************
//********CATALOGED PROCEDURE FOR M.3 TABLE GENERATION*****************
//*********************************************************************
//********ENTER SYMBOLIC PARM YEAR=(2 DIGIT YEAR)**********************
//*
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* REQ DATE  :02/26/02
//* EXEC DATE :02/26/02
//* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(M3POST)
//* PGMR      :MATTHEW URATO
//* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
//* INPUTS    :SEE SYSIN DD FOR EACH STEP
//* OUTPUTS   :SEE SYSIN DD FOR EACH STEP
//* PURPOSE   :CREATE M.3 TABLES FOR EACH YEAR
//*********************************************************************
//*
//YXXM3001 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR')
//WORK   DD SPACE=(CYL,(300,50)),DCB=(BLKSIZE=27648)
//WORK0  DD SPACE=(CYL,(300,50))
//WORK1  DD SPACE=(CYL,(300,50))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M3.PGMLIB99
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2007.MEDPRG&YEAR
//OUT DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3001.Y&YEAR.M3IN1,
//        SPACE=(CYL,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM301),DISP=SHR
//*********************************************************************
//YXXM3002 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(300,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M3.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3001.Y&YEAR.M3IN1
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1006.M1CNTS&YEAR
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3002.Y&YEAR.M3RAS,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM302),DISP=SHR
//*********************************************************************
//YXXM3003 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(300,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M3.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3001.Y&YEAR.M3IN1
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1006.M1CNTS&YEAR
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3003.Y&YEAR.M3ART,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM303),DISP=SHR
```

```
//*******************************************************************
//YXXM3004 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
```
D#MSQ.@MSQ00M3.PGMLIB99(M3POST) -- CONTINUED

```
//WORK   DD SPACE=(CYL,(300,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M3.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1003.M1CNTS&YEAR
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3001.Y&YEAR.M3IN1
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3004.Y&YEAR.M3CNT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3004.Y&YEAR.M3URT,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM304),DISP=SHR
//*******************************************************************
//YXXM3005 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(300,50)),DCB=(BLKSIZE=27648)
//IN0 DD DSN=D#MSQ.@MSQ00M3.PGMLIB99,DISP=SHR
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3002.Y&YEAR.M3RAS
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3003.Y&YEAR.M3ART
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3004.Y&YEAR.M3CNT
//IN4 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3004.Y&YEAR.M3URT
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3RAS.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3ART.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT3 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3CNT.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//OUT4 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3URT.XPT,
//        SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM305),DISP=SHR
//*******************************************************************
//YXXM3006 EXEC SAS8,REGION=6M,PARM=('ERRORS=0 NONEWS',
//       'SYSPARM=&YEAR'),COND=(4,LT)
//IN0 DD DSN=D#MSQ.@MSQ00M3.PGMLIB99,DISP=SHR
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3RAS.XPT,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3ART.XPT,DISP=SHR
//IN3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3CNT.XPT,DISP=SHR
//IN4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3005.Y&YEAR.M3URT.XPT,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3006.RTFRAS&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3006.RTFART&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3006.RTFCNT&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M3006.RTFURT&YEAR,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,2)),
//       DCB=(RECFM=FB,LRECL=80)
```

```
//SYSIN DD DSN=D#MSQ.@MSQ00M3.PGMLIB99(SYSM306),DISP=SHR
//*****************************************************************
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM301)

```
************************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :01/31/02
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM301)
* PGMR      :HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M3.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YxxM2007.MEDPRGXX - M2 DISCHARGE
             LEVEL ANALITICAL FILE
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3001.YxxM3IN1
*                  - M.3 MAIN INTERMEDIATE FILE WITH COUNTERS (DISK)
* PURPOSE   :FROM SAS M.2 PRINCIPAL ANALYTIC FILE
*            CREATE A SAS FILE WITH COUNTS OF ADMISSIONS BY
*            SPG, AGE, SEX, SUBGROUPS
*            SEE MACRO YXXM3001 FOR MORE DETAILS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
************************************************************************;
       /**INCLUDE MACRO**/
 %INCLUDE IN0(YXXM3001)/ SOURCE2;

         /**MACRO CALL**/

 %YXXM3001(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3001)

```
%MACRO YXXM3001(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM3001
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(YXXM3001)
3  WRITTEN BY:           HELEN MARGULIS
                         (MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JULY 16,2001
6  PROJECT:              MQMS
7  PURPOSE:
      THIS MACRO IS A MODIFIED VERSION OF MACRO YXXM1003
      WRITTEN BY MATTHEW URATO.

      THE PURPOSE IS AS FOLLOWS:
      CREATE THE ANALYTIC COUNT FILE FOR M.3 CALCULATING  # OF
      DISCHARGES PER EACH ANALYTICAL SUBGROUP BY ADE/SEX FOR EACH
      SPG GROUP.

8  YEAR OF DATA IS SET IN THE MAIN PROGRAM YXXM3001 AS A GLOBAL MACRO
   VARIABLE YY.

9  INPUT AND OUTPUT IS DONE IN THE MAIN PROGRAM
*******************************************************************
*******************************************************************;
      /**LIST OF SUBGROUPS */

*******************************************************************;
* STEP 1. Assign major group variables to a macro variable to
* be able to do individual frequencies.
*******************************************************************;

%LET VARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL Y&YY.FAGE Y&YY.MDST
          Y&YY.RACE Y&YY.SEX Y&YY.STATE Y&YY.URBN;


             /**CODE BEGINS**/

      /**TO CREATE A SINGLE SPG VAR FROM SPG1-SPG6 **/
            /** COMMENT: YxxM14=1 ALWAYS **/
      /**  EVERY ADMISSION IS COUNTED SCNT TIMES   **/

*******************************************************************;
* STEP 2. Read in the Base Analytic MedPAR file and create a
* single SPG variable from the six procedure variables
* (YXXSPG1-YXXSPG6).
*******************************************************************;

 DATA TEMP0(KEEP=Y&YY.SPG &VARS)/VIEW=TEMP0;
      SET IN1.MEDPRG&YY (KEEP=Y&YY.SPG1-Y&YY.SPG6 Y&YY.SCNT
                              &VARS Y&YY.M14
                         WHERE=(Y&YY.M14='1'));
      LENGTH Y&YY.SPG $3. SCOUNT 2;
      ARRAY SP(6) $ Y&YY.SPG1-Y&YY.SPG6;
```

```
      IF NOT(1 LE Y&YY.SCNT LE 6) THEN SCOUNT=6;
      ELSE SCOUNT=Y&YY.SCNT;
      DO I=1 TO SCOUNT;
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3001) -- CONTINUED

          Y&YY.SPG=SP(I);
          IF SP(I) NE '' THEN OUTPUT;
      END;
      DROP SCOUNT;
 RUN;

      /**TO SORT INPUT FILE BY SPG, AGE AND SEX **/

*******************************************************************;
* STEP 3. Sort select variables from the TEMP0 file by SPG,
* age group and sex for the M1-4 cohort.  Write out sorted
* variables to temp file WORK0.TEMP1 file.
*******************************************************************;

 PROC SORT DATA=TEMP0
           OUT=WORK0.TEMP1;
      BY Y&YY.SPG Y&YY.FAGE Y&YY.SEX;
 RUN;

        /**TO CREATE MIRRORED AGE & SEX VARS **/

*******************************************************************;
* STEP 4. Read in Temp WORK0.TEMP1 file and add two variables
* - mirrored age and sex variables. The mirrored variables are
* created for ease of calculation for frequencies below.
*******************************************************************;

 DATA WORK1.TEMP1;
      SET WORK0.TEMP1;
      Y&YY.FAGEM = Y&YY.FAGE;
      Y&YY.SEXM = Y&YY.SEX;
 RUN;

      /**SET UP LOOP FOR ALL 9 VARS**/

*******************************************************************;
* STEP 5. Create do loop for the nine major subgroups.
*******************************************************************;

 %DO I = 1 %TO 9;            /**START LOOP **/

*******************************************************************;
* STEP 5a. Frequency on the nine major subgroups by SPG, age group,
* and sex.  Each of these is written out respectively to nine
* temp files (TEMP1 through TEMP9).
*******************************************************************;

      /**OUTPUT COUNTS FOR EACH VARIABLE BY SPG/AGE/SEX **/
    PROC FREQ DATA=WORK1.TEMP1  NOPRINT;
         BY Y&YY.SPG Y&YY.FAGEM Y&YY.SEXM;
         TABLES %SCAN(&VARS,&I)
         / OUT=TEMP&I(DROP=PERCENT);
    RUN;
```

```
     /**NAME THE VALUES OF YXXVARS AND SET THEIR**/
     /**ATTRIBUTES APPROPRIATELY**/

*******************************************************************;
* STEP 5b. Modify the nine temp files adding a variable YXXVARS
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3001) -- CONTINUED

* that renames the values of the variables to take on the first
* three letters of the variable name plus the variable value
* applied as a suffix at the end (e.g., YXXSEX has values of
* 1 and 2.  YXXVARS now has values of SEX1 and SEX2).
*******************************************************************;

    DATA TEMP&I(DROP=%SCAN(&VARS,&I) RENAME=(COUNT=Y&YY.NSPG));
         LENGTH Y&YY.VARS $8;
         SET TEMP&I;
         Y&YY.VARS=
         COMPRESS("%SUBSTR(%SCAN(&VARS,&I),4,3)"!!%SCAN(&VARS,&I));
    RUN;

     /**COUNTS BY TOTAL FOR SPG/AGE/SEX **/

*******************************************************************;
* STEP 5c. Create an individual dataset (TEMP10) with a new
*    variable    YXXVARS    that    has    a    value    of    TOTAL.
*********************************
*********************************;

    %IF &I=9 %THEN %DO;
       DATA TEMP10(DROP=Y&YY.NSPG  RENAME=(COUNT=Y&YY.NSPG ));
             SET TEMP9;
             BY  Y&YY.SPG Y&YY.FAGEM Y&YY.SEXM;
             LENGTH Y&YY.VARS $8;
             Y&YY.VARS='TOTAL';
             COUNT+Y&YY.NSPG;
             IF LAST.Y&YY.SEXM THEN DO;
                OUTPUT;
                COUNT=0;
             END;
       RUN;
    %END;
 %END;                    /**END LOOP **/
          /**CONCATENATE THE COUNT FILES FOR ALL 9 VARS**/
           /**THIS IS THE PRINCIPAL OUTPUT FILE**/

*******************************************************************;
* STEP 6. Create a principal count file (YXXM3IN1) by stacking
* the ten temp files in alphabetical order.
*******************************************************************;

  DATA OUT.Y&YY.M3IN1(LABEL="PROJECT:MQMS, PROGRAM:Y&YY.M3001"
             SORTEDBY=Y&YY.SPG Y&YY.FAGE  Y&YY.SEX  Y&YY.VARS);
       SET TEMP1 TEMP2 TEMP3 TEMP4 TEMP5 TEMP6 TEMP7 TEMP8
       TEMP10 TEMP9;
       BY  Y&YY.SPG Y&YY.FAGEM Y&YY.SEXM;
       LENGTH YEAR 2.;
       YEAR=&YY;
```

```
       LABEL
       Y&YY.VARS  = "SUBGROUP NAME"
       Y&YY.NSPG  = "M.3 discharge cnt by SUBGROUP,age,sex"
       Y&YY.FAGEM = "&YY. AGE GROUPS"
       Y&YY.SEXM  = "&YY. SEX"
   ;
       RENAME
       Y&YY.FAGEM  = Y&YY.FAGE
       Y&YY.SEXM   = Y&YY.SEX ;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3001) -- CONTINUED

```
  RUN;

  PROC DATASETS LIB=WORK NOLIST;
       DELETE TEMP1-TEMP10;
  RUN;

*******************************************************************;
* STEP 7. Proc print and proc contents on YXXM3IN1 file.
*******************************************************************;

  PROC PRINT DATA=OUT.Y&YY.M3IN1(OBS=400);
       TITLE2 "M.3 MAIN INTERMEDIATE COUNT FILE";
  RUN;

  PROC CONTENTS DATA=OUT.Y&YY.M3IN1 DIRECTORY POSITION;
  RUN;

                    /**MACRO ENDS**/
%MEND YXXM3001;
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM302)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM302)
* PGMR      :HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    : D#MSQ.@MSQ00M3.PGMLIB99  - MACRO LOCATION
*           -D#MSQ.@MSQ0000.SAS.YxxM1003.M1CNTSXX - file with
*            Denominator counters by subgroups
*           -D#MSQ.@MSQ0000.SAS.YxxM3001.YxxM3IN1 - intermediate file
*            with counters of admissions by age,sex,subgroups for
*            each SPG.
*           -D#MSQ.@MSQ0000.SAS.YXXM1004.AGESEX99 - denominator counters
*            by age and sex for 1999 year. This file is the same for
*            any year.
*           -D#MSQ.@MSQ0000.SAS.YxxM1006.M1CNTSxx - denominator counters
*            by subgroups,age,sex
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3002.YxxM3RAS - age/sex adjusted
*            rates.
* PURPOSE   :create a SAS file with age/sex adjusted rate
*            of admissions  by SPG and SUBGROUPS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*
* STEPS     :  1. create a table shell with all the rows and columns.
*              STEPS 2-16 are done in MACRO YxxM2009.
*              2. read in counters of discharges.
*              3. read in age/sex counters from 1999 Denominator and
*                 create weights.
*                 STEPS 4-7 ARE DONE TO CALCULATE AGE/SEX ADJUSTED
*                 RATES FOR ALL BUT AGE & SEX SUBGROUPS
*              4. read in counters from denominator.
*              5. create unadjusted rates per 1000 for each age/sex
*                 cell, using CTM14 counters.
*              6. adjust by 1999 weights.
*              7. sum on VARS/SPG level.
*                 STEPS 8-11 ARE DONE TO CALCULATE SEX ADJUSTED
*                 RATES AGE SUBGROUPS
*              8. sum denominator counters to adjust AGE groups by SEX
*              9. unadjusted rates per 1000
*             10. adjust by 1999 sex weights
*             11. sum on vars/SPG level
*                 STEPS 12-15 ARE DONE TO CALCULATE AGE ADJUSTED
*                 RATES SEX SUBGROUPS
*             12. sum denominator counters to adjust SEX groups by AGE
*             13. unadjusted rates per 1000
*             14. adjust by 1999 sex weights
*             15. sum on vars/SPG level
*             16. merge to the table shell to set missing cells to 0,
*                 write to output file.
*********************************************************************;
       /**INCLUDE MACRO**/
```

```
%INCLUDE IN0(M3GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM3002)/ SOURCE2;
```
D#MSQ.@MSQ00M3.PGMLIB99(SYSM302) -- CONTINUED

```
 ************************************************************************;
 * step 1. table shell
 ************************************************************************;
       /**MACRO CALL**/

%M3GENERL(VARNAME=SPG,VARCOUNT=142,VLENGTH=3,YY=&SYSPARM)

 ************************************************************************;
 * step 2-15. rates calculations . DATASETS: FINAL,FINALA,FINALS
 ************************************************************************;
       /**MACRO CALL**/
%YXXM3002(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(M3GENERL)

```
%MACRO M3GENERL(VARNAME=SPG,VARCOUNT=142,VLENGTH=3,YY=99);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           M3GENERL
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(M3GENERL)
3  WRITTEN BY:           HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 6,2001
6  PROJECT:              MQMS
7  PURPOSE:
      THIS MACRO CREATES A TABLE SHELL WITH ONE ROW PER EACH
      ANALYTICAL SUBGROUP AND AS MANY COLUMNS AS THE NUMBER OF
      DIFFERENT VALUES OF THE VARIABLE VARNAME
      (FOR EXAMPLE FOR VARNAME=SPG THE TABLE WILL HAVE 142 COLUMNS).
      IT IS DESIGNED TO BE ABLE TO SET TO 0 ALL MISSING
      VARNAME COUNTER/RATE VALUES.

8  THIS MACRO COULD BE USED FOR ANY YEAR

9  PARAMETERS:
   THERE ARE 4 KEYWORD PARAMETERS:
   VARNAME, VARCOUNT, VLENGTH, YY

 VARNAME  - COLUMN VARIABLE BY WHICH STATISTICS IS DONE.
 VARCOUNT - NUMBER OF VALUES(COLUMNS) FOR VARNAME VARIABLE.
 VLENGTH  - LENGTH OF THE VARIABLE &VARNAME.
 YY       - THE TWO-DIGIT YEAR OF ANALYSIS.

10 EXAMPLE MACRO CALL:
%M3GENERL(VARNAME=SPG,VARCOUNT=142,VLENGTH=3)

11 OTHER NOTES
IT IS ASSUMED THAT THE &VARNAME IS A CHARACTER VARIABLE.
THIS MACRO COULD BE USED FOR ANY YEAR OF DATA
*****************************************************************
*****************************************************************;

             /**CODE BEGINS**/

*****************************************************************;
* STEP 1. Read in the M1CNTSXX files (Denom counts files from M1)
* Sort by YXXVARS and delete duplicates to resolve to one for
* preparation of a table shell.
*****************************************************************;

       /** TO GET LIST OF SUBGROUPS**/
  DATA INI1/VIEW=INI1;
       SET IN1.M1CNTS&YY;
  RUN;
  PROC SORT NODUPKEY DATA=INI1(KEEP=Y&YY.VARS) OUT=INI12;
       BY Y&YY.VARS;
  RUN;

*****************************************************************;
```

```
* STEP 2. Create file INI13 that takes the YXXVARS that are in
* rows and lays them out into columns.  Write out variables into
* one row.  Proc print of INI13.
```
D#MSQ.@MSQ00M3.PGMLIB99(M3GENERL) -- CONTINUED

```
*********************************************************************;

      /** TO PRINT LIST OF SUBGROUPS  **/;
  PROC TRANSPOSE DATA=INI12 OUT=INI13 ;
       VAR Y&YY.VARS;
  RUN;
  PROC PRINT U DATA=INI13;
       TITLE "ALL THE SUBGROUPS";
  RUN;

*********************************************************************;
* STEP 3. Create table shell (temp dataset SHELL).
*********************************************************************;

      /** TO CREATE A TABLE SHELL  **/;
  DATA SHELL(KEEP=VARS &VARNAME);
       SET INI12(KEEP=Y&YY.VARS RENAME=(Y&YY.VARS=VARS));
       LENGTH &VARNAME $&VLENGTH.;
       DO I=1 TO &VARCOUNT;
          &VARNAME =LEFT(PUT(I,&VLENGTH..));
          OUTPUT;
       END;
  RUN;

*********************************************************************;
* STEP 4. Proc sort SHELL by VARS (YXXVARS) and within VARS
* by VARNAME (VARNAME will be SPGs).
*********************************************************************;

  PROC SORT DATA=SHELL;
       BY VARS &VARNAME;
  RUN;
 %MEND M3GENERL;
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002)

```
%MACRO YXXM3002(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM3002
2  LOCATION:             D#MSQ.@MSQ00.TI03LIB3(YXXM3002)
3  WRITTEN BY:           HELEN MARGULIS
                         (MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate age/sex adjusted rate
                         of admissions by SPG and SUBGROUPS.
                         Age subgroup rates are adjusted only by sex
                         and sex subgroup rates are adjusted only by
                         age.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

9  STEPS     : (first step is done in the main program)
               2. read in counters of discharges.
               3. read in age/sex counters from 1999 Denominator and
                  create weights.
                  STEPS 4-7 ARE DONE TO CALCULATE AGE/SEX ADJUSTED
                  RATES FOR ALL BUT AGE & SEX SUBGROUPS
               4. read in counters from denominator.
               5. create unadjusted rates per 1000 for each age/sex
                  cell, using CTM14 counters.
               6. adjust by 1999 weights.
               7. sum on VARS/SPG level.
                  STEPS 8-11 ARE DONE TO CALCULATE SEX ADJUSTED
                  RATES FOR AGE SUBGROUPS
               8. sum denominator counters to adjust AGE groups by SEX
               9. unadjusted rates per 1000
              10. adjust by 1999 sex weights
              11. sum on vars/SPG level
                  STEPS 12-15 ARE DONE TO CALCULATE AGE ADJUSTED
                  RATES FOR SEX SUBGROUPS
              12. sum denominator counters to adjust SEX groups by AGE
              13. unadjusted rates per 1000
              14. adjust by 1999 AGE weights
              15. sum on vars/SPG level
*             16. merge to the table shell to set missing cells to 0,
*                 write to output file.

*******************************************************************

*******************************************************************;
              /**CODE BEGINS**/

 *****************************************************************;
 * Step 1. Input counters from file YXXM3IN1 and write them out
 * to 3 files: 1 for age groups (INI2A), 1 for sex (INI2S), and one
 * for all the rest of the groups (INI2).  A fourth file (ERR1) was
 * created to write out counters where age or sex is missing as a
```

```
* quality check.
*********************************************************************;
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002) -- CONTINUED

```
DATA INI2 INI2A INI2S ERR1;
     SET IN2.Y&YY.M3IN1 ;
     IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
     ELSE
     IF SUBSTR(Y&YY.VARS,1,2)='FA' THEN OUTPUT INI2A;
     ELSE
     IF SUBSTR(Y&YY.VARS,1,2)='SE' THEN OUTPUT INI2S;
     ELSE OUTPUT INI2;
    *IF _N_ > 50000 THEN STOP;
RUN;
PROC PRINT U DATA=ERR1(OBS=46);
   TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
RUN;

*********************************************************************;
* Step 1a. Sort the three files by appropriate variables to
* eventually perform merges to construct rates.
*********************************************************************;

PROC SORT DATA=INI2;
     BY  Y&YY.FAGE  Y&YY.SEX  Y&YY.VARS;
RUN;
PROC SORT DATA=INI2A;
     BY  Y&YY.SEX  Y&YY.VARS;
RUN;
PROC SORT DATA=INI2S;
     BY  Y&YY.FAGE Y&YY.VARS;
RUN;

*********************************************************************;
* Step 2. Create ASWGT99 (age/sex weight).  Write out 3 weight
* files: age/sex (WGTA_S), age (WGT_A), and sex (WGT_S) from the
* AGESEX99 file created in M1.  Proc print on the three files.
*********************************************************************;

DATA WGTA_S WGT_S WGT_A;
    IF _N_=1 THEN SET IN4.AGESEX99(WHERE=(Y99FAGE='' & Y99SEX='')
                                   RENAME=(Y99CNTS=TOTAL99));
    SET IN4.AGESEX99;
    ASWGT99=Y99CNTS/TOTAL99;
    IF Y99FAGE NE '' & Y99SEX NE '' THEN OUTPUT WGTA_S;
    ELSE
    IF Y99FAGE = '' & Y99SEX NE '' THEN OUTPUT WGT_S;
    ELSE
    IF Y99FAGE NE '' & Y99SEX = '' THEN OUTPUT WGT_A;
    RENAME Y99FAGE=Y&YY.FAGE Y99SEX=Y&YY.SEX;
RUN;
PROC PRINT DATA=WGTA_S;
     TITLE "1999 AGE/SEX WEIGHTS";
RUN;
PROC PRINT DATA=WGT_S;
     TITLE "1999 SEX WEIGHTS";
RUN;
```

```
PROC PRINT DATA=WGT_A;
     TITLE "1999 AGE WEIGHTS";
RUN;


*******************************************************************;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002) -- CONTINUED
```
* Step 3. Create age/sex adjusted rates for all subgroups except
* age and sex.
*******************************************************************;

*******************************************************************;
* Step 3a. Input counters from Denominator created in M.1.
*******************************************************************;

DATA INI3;
     SET IN3.M1ASCN&YY
         (KEEP=Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE  Y&YY.SEX );
RUN;

*******************************************************************;
* Step 3b. To construct the unadjusted rates per 1000 for each
* age/sex cell, merge the INI2 temp file to the Denominator temp
* count file INI3.  Create the unadjusted rate per 1000 (YXXRSPG)
* and output to TEMP1.
*******************************************************************;

DATA TEMP1 ERR2;
     MERGE INI3(IN=IN2)
           INI2(IN=IN1);
     BY  Y&YY.FAGE  Y&YY.SEX Y&YY.VARS;
     IF IN1 & NOT IN2 THEN OUTPUT ERR2;
     ELSE
     IF IN1 & IN2 THEN DO;
        Y&YY.RSPG = 1000* (Y&YY.NSPG/Y&YY.CTM14);
        OUTPUT TEMP1;
     END;
RUN;
PROC PRINT U DATA=ERR2(OBS=46);
   TITLE "!!! NB !!!!: no cells in denominator counter file";
RUN;

*******************************************************************;
* Step 3c. Add a new variable (YXXASSPG)  - the age/sex adjusted
* SPG rate - and output to TEMP2.
*******************************************************************;

DATA TEMP2 ERR3;
     MERGE TEMP1(IN=IN1)
           WGTA_S(IN=IN2);
     BY  Y&YY.FAGE  Y&YY.SEX ;
     IF IN1 & NOT IN2 THEN OUTPUT ERR3;
     ELSE
     IF IN2 & IN1 THEN DO;
        Y&YY.ASSPG = Y&YY.RSPG *ASWGT99;
        OUTPUT TEMP2;
     END;
```

```
RUN;
PROC PRINT U DATA=ERR3(OBS=46);
   TITLE "!!! NB !!!!: no weight in 1999 denominator";
RUN;

*********************************************************************;
* Step 3d: Sum the above rate (YXXASSPG) by YXXVARS, YXXSPG,
* and YEAR and output to FINAL.
*********************************************************************;
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002) -- CONTINUED

```
PROC SUMMARY DATA=TEMP2 NWAY;
     CLASS  Y&YY.VARS Y&YY.SPG YEAR;
     VAR Y&YY.ASSPG ;
     OUTPUT OUT=FINAL(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.ASSPG) SUM=;
RUN;

*********************************************************************;
* Step 4: Create sex adjusted rates for age subgroups.
*********************************************************************;

*********************************************************************;
* Step 4a: Sum the Denominator File counters to adjust age groups
* by sex and write out to INI3A.
*********************************************************************;

PROC SUMMARY NWAY DATA=INI3;
     CLASS  Y&YY.SEX Y&YY.VARS ;
     VAR Y&YY.CTM14 ;
     OUTPUT OUT=INI3A(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.SEX)
            SUM=;
RUN;

*********************************************************************;
* Step 4b. Create unadjusted rates per 1000 (YXXRSPG) and output
* to TEMP1A.
*********************************************************************;

DATA TEMP1A;
     MERGE INI3A(IN=IN2)
           INI2A(IN=IN1);
     BY  Y&YY.SEX Y&YY.VARS;
     IF IN1 & IN2 THEN DO;
        Y&YY.RSPG = 1000* (Y&YY.NSPG/Y&YY.CTM14);
        OUTPUT TEMP1A;
     END;
RUN;

*********************************************************************;
* Step 4c. Adjust rates by the 1999 sex weights (YXXASSPG) and
* output to TEMP2A.
*********************************************************************;

DATA TEMP2A;
     MERGE TEMP1A(IN=IN1)
           WGT_S(IN=IN2);
     BY  Y&YY.SEX ;
```

```
      IF IN2 & IN1 THEN DO;
         Y&YY.ASSPG = Y&YY.RSPG *ASWGT99;
         OUTPUT TEMP2A;
      END;
 RUN;

 ******************************************************************;
 * Step 4d. Sum the above rate (YXXASSPG) by YXXVARS, YXXSPG,
 * and YEAR and output to FINALA.
 ******************************************************************;

 PROC SUMMARY DATA=TEMP2A NWAY;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002) -- CONTINUED
```
      CLASS  Y&YY.VARS Y&YY.SPG YEAR;
      VAR Y&YY.ASSPG ;
      OUTPUT OUT=FINALA(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.ASSPG) SUM=;
 RUN;

 ******************************************************************;
 * Step 5. Create age adjusted rates for sex subgroups.
 ******************************************************************;
 ******************************************************************;
 * Step 5a. Sum the Denominator File counters to adjust sex groups
 * by age and write out to INI3S.
 ******************************************************************;

 PROC SUMMARY NWAY DATA=INI3;
      CLASS  Y&YY.FAGE Y&YY.VARS ;
      VAR Y&YY.CTM14 ;
      OUTPUT OUT=INI3S(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE)
             SUM=;
 RUN;

 ******************************************************************;
 * Step 5b. Create unadjusted rates per 1000 (YXXRSPG) and output
 * to TEMP1S.
 ******************************************************************;

 DATA TEMP1S;
      MERGE INI3S(IN=IN2)
            INI2S(IN=IN1);
      BY  Y&YY.FAGE Y&YY.VARS;
      IF IN1 & IN2 THEN DO;
         Y&YY.RSPG = 1000* (Y&YY.NSPG/Y&YY.CTM14);
         OUTPUT TEMP1S;
      END;
 RUN;

 ******************************************************************;
 * Step 5c. Adjust rates by the 1999 age weights (YXXASSPG) and
 * output to TEMP2S.
 ******************************************************************;

 DATA TEMP2S;
      MERGE TEMP1S(IN=IN1)
            WGT_A(IN=IN2);
      BY  Y&YY.FAGE ;
```

```
      IF IN2 & IN1 THEN DO;
         Y&YY.ASSPG = Y&YY.RSPG *ASWGT99;
         OUTPUT TEMP2S;
      END;
 RUN;

 ********************************************************************;
 * Step 5d. Sum the above rate (YXXASSPG) by YXXVARS, YXXSPG, and
 * YEAR and output to FINALS.
 ********************************************************************;

 PROC SUMMARY DATA=TEMP2S NWAY;
      CLASS  Y&YY.VARS Y&YY.SPG YEAR;
      VAR Y&YY.ASSPG ;
      OUTPUT OUT=FINALS(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.ASSPG) SUM=;
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3002) -- CONTINUED

 RUN;

 ********************************************************************;
 * Step 6. Create ALLS that puts all the rates (FINAL, FINALA, and
 * FINALS) together in one file.
 ********************************************************************;

 DATA ALLS;
      SET FINAL FINALA FINALS;
      BY Y&YY.VARS Y&YY.SPG ;
 RUN;

 ********************************************************************;
 * Step 7. Merge ALLS and the table shell from M3GENERL (SHELL) by
 * YXXVARS and YXXSPG and output to YXXM3RAS.  Values in SHELL that
 * do not have a match to ALLS are set to 0.
 ********************************************************************;

 DATA OUT1.Y&YY.M3RAS(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M3002"
                      SORTEDBY=Y&YY.VARS Y&YY.SPG )
      ERR4;
      MERGE ALLS(IN=IN1)
            SHELL(IN=IN2 RENAME=(SPG=Y&YY.SPG VARS=Y&YY.VARS));
      BY Y&YY.VARS Y&YY.SPG ;
      IF IN2 & NOT IN1 THEN Y&YY.ASSPG = 0;
      ELSE
      IF IN1 & NOT IN2 THEN OUTPUT ERR4;
      OUTPUT OUT1.Y&YY.M3RAS;
 run;
 PROC PRINT U DATA=ERR3(OBS=46);
    TITLE "!!! NB !!!!: no column or row in the table shell";
 RUN;
 PROC PRINT U DATA=OUT1.Y&YY.M3RAS(OBS=46);
    TITLE "M3: AGE/SEX ADJUSTED RATES";
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M3RAS;
 RUN;

                     /**MACRO ENDS**/
%MEND YXXM3002;
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM303)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/14/01
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM303)
* PGMR      :HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ00M3.PGMLIB99  - MACRO LOCATION
*            D#MSQ.@MSQ0000.SAS.YxxM3001.YxxM3IN1 - intermediate file
*            with counters of admissions by age,sex,subgroups for
*            each SPG
*            D#MSQ.@MSQ0000.SAS.YXXM1004.AGESEX99 - denominator counters
*            by age and sex for 1999 year. This file is the same for
*            all years.
*            D#MSQ.@MSQ0000.SAS.YxxM1006.M1CNTSxx - denominator counters
*            by subgroups,age,sex
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3003.YxxM3ART - age adjusted
*            rates
* PURPOSE   :create a SAS file with age adjusted rate
*            of admissions  by SPG and SUBGROUPS. Age subgroup rates
*            are unadjusted
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*
* STEPS     : calculation (steps 2-8)is done in macro YXXM3003
*             1. create table shell with all the rows and columns.
*                STEPS 2-7 ARE DONE IN MACRO YxxM3003
*             2. read in counters of discharges and sum them up on
*                AGE*VARS*SPG level.
*             3. read in counters from denominator and sum them up on
*                AGE*VARS level.
*             4. read in age counters from 1999 Denominator and
*                create weights
*             5. create unadjusted rates per 1000 for each age cell,
*                using CTM14 counters.
*             6. adjust by 1999 age weights,
*                AGE subgroups are unadjusted,
*             7. sum on VARS/SPG level
*             8. merge to the table shell to set missing cells to 0,
*                write to output file.
*********************************************************************;
       /**INCLUDE MACROS**/

%INCLUDE IN0(M3GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM3003)/ SOURCE2;

 ********************************************************************;
 * step1. table shell
 ********************************************************************;
       /**MACRO CALL**/
%M3GENERL(VARNAME=SPG,VARCOUNT=142,VLENGTH=3,YY=&SYSPARM)
```

```
 ************************************************************************;
 * steps 2-8: create dataset FINAL with rates
 ************************************************************************;
```
D#MSQ.@MSQ00M3.PGMLIB99(SYSM303) -- CONTINUED

```
       /**MACRO CALL**/
%YXXM3003(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3003)

```
%MACRO YXXM3003(YY=99);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM3003
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(YXXM3003)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate age adjusted rate of admissions
                         by SPG and SUBGROUPS. Age subgroup rates are
                         unadjusted.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

9  STEPS     : first step is done in the main program
               2. read in counters of discharges and sum them up on
                  AGE*VARS*SPG level.
               3. read in counters from denominator and sum them up on
                  AGE*VARS level.
               4. read in age counters from 1999 Denominator and
                  create weights
               5. create unadjusted rates per 1000 for each age cell,
                  using CTM14 counters.
               6. adjust by 1999 age weights
                  AGE subgroup rate are unadjusted
               7. sum on VARS/SPG level
*              8. merge to the table shell to set missing cells to 0,
*                 write to output file.
*****************************************************************
*****************************************************************;

 ****************************************************************;
 * Step 1. Input counters from file YXXM3IN1 and write them out to
 * INI2.
 ****************************************************************;

 DATA INI2 ERR1;
      SET IN2.Y&YY.M3IN1 ;
      IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
      ELSE OUTPUT INI2;
 RUN;
 PROC PRINT U DATA=ERR1(OBS=46);
    TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
 RUN;


 ****************************************************************;
 * Step 2. Sum the count variable (YXXNSPG) by YXXFAGE, YXXVARS,
 * YXXSPG, and YEAR and output to INI21.
 ****************************************************************;

 PROC SUMMARY DATA=INI2 NWAY;
      CLASS  Y&YY.FAGE Y&YY.VARS Y&YY.SPG YEAR;
```

```
     VAR Y&YY.NSPG ;
     OUTPUT OUT=INI21(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.NSPG Y&YY.FAGE)
            SUM=;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3003) -- CONTINUED

```
 RUN;

 ********************************************************************;
 * Step 3. Sum the Denominator File counters by age group and
 * YXXVARS in preparation for the creation of age weights and write
 * out to INI31.
 ********************************************************************;

 PROC SUMMARY NWAY DATA=IN4.M1ASCN&YY
                  (KEEP=Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE  Y&YY.SEX );
      CLASS  Y&YY.FAGE Y&YY.VARS ;
      VAR Y&YY.CTM14 ;
      OUTPUT OUT=INI31(KEEP= Y&YY.VARS Y&YY.CTM14 Y&YY.FAGE)
             SUM=;
 RUN;

 ********************************************************************;
 * Step 4. Create AWGT99 (age weight).  Write out an age weight
 * file WGTA). Proc print on WGTA.
 ********************************************************************;

 DATA WGTA;
     IF _N_=1 THEN SET IN3.AGESEX99(WHERE=(Y99FAGE='' & Y99SEX='')
                                    RENAME=(Y99CNTS=TOTAL99));
     SET IN3.AGESEX99(WHERE=(Y99FAGE NE '' & Y99SEX ='' ));
     AWGT99=Y99CNTS/TOTAL99;
     RENAME Y99FAGE=Y&YY.FAGE Y99SEX=Y&YY.SEX;
 RUN;
 PROC PRINT DATA=WGTA;
      TITLE "1999 AGE WEIGHTS";
 RUN;

 ********************************************************************;
 * Step 5. To construct the unadjusted rates per 1000 for each age
 * cell, merge the INI21 temp file to the Denominator temp count
 * file INI31.  Create the unadjusted rate per 1000 (YXXRSPG) and
 * output to TEMP1.
 ********************************************************************;

 DATA TEMP1 ERR2;
      MERGE INI31(IN=IN2)
            INI21(IN=IN1);
      BY  Y&YY.FAGE   Y&YY.VARS;
      IF IN1 & NOT IN2 THEN OUTPUT ERR2;
      ELSE DO;
         IF Y&YY.NSPG=. THEN Y&YY.NSPG=0;
         Y&YY.RSPG = 1000* (Y&YY.NSPG/Y&YY.CTM14);
         OUTPUT TEMP1;
      END;
 RUN;
 PROC PRINT U DATA=ERR2(OBS=46);
    TITLE "!!! NB !!!!: no cells in denominator counter file";
 RUN;
```

```
*******************************************************************;
* Step 6. Add a new variable (YXXAASPG)  - the age adjusted SPG
* rate - and output to TEMP2.  Age group rates remain unadjusted.
*******************************************************************;
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3003) -- CONTINUED

```
DATA TEMP2 ERR3;
     MERGE TEMP1(IN=IN1)
           WGTA(IN=IN2);
     BY  Y&YY.FAGE   ;
     IF IN1 & NOT IN2 THEN OUTPUT ERR3;
     ELSE
     IF IN1 & IN2 THEN DO;
        IF SUBSTR(Y&YY.VARS,1,2)='FA' THEN
        Y&YY.AASPG = Y&YY.RSPG ; *UNADJUSTED RATE FOR AGE GROUPS;
        ELSE
        Y&YY.AASPG = Y&YY.RSPG *AWGT99;
        OUTPUT TEMP2;
     END;
RUN;
PROC PRINT U DATA=ERR3(OBS=46);
   TITLE "!!! NB !!!!: no weight in 1999 denominator";
RUN;

*******************************************************************;
* Step 7. Sum the above rate (YXXAASPG) by YXXVARS, YXXSPG, and
* YEAR and output to FINAL.
*******************************************************************;

PROC SUMMARY DATA=TEMP2 NWAY;
     CLASS  Y&YY.VARS Y&YY.SPG YEAR;
     VAR Y&YY.AASPG ;
     OUTPUT OUT=FINAL(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.AASPG) SUM=;
RUN;

*******************************************************************;
* Step 8. Merge FINAL and the table shell from M3GENERL (SHELL) by
* YXXVARS and YXXSPG and output to YXXM3ART.  Values in SHELL that
* do not have a match to FINAL are set to 0.
*******************************************************************;

DATA OUT1.Y&YY.M3ART(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M3003"
                     SORTEDBY=Y&YY.VARS Y&YY.SPG )
     ERR4;
     MERGE FINAL(IN=IN1)
           SHELL(IN=IN2 RENAME=(SPG=Y&YY.SPG VARS=Y&YY.VARS));
     BY Y&YY.VARS Y&YY.SPG ;
     IF IN2 & NOT IN1 THEN Y&YY.AASPG = 0;
     ELSE
     IF IN1 & NOT IN2 THEN OUTPUT ERR4;
     OUTPUT OUT1.Y&YY.M3ART;
run;
PROC PRINT U DATA=ERR4(OBS=46);
   TITLE "!!! NB !!!!: no column or row in the table shell";
RUN;
PROC PRINT U DATA=OUT1.Y&YY.M3ART(OBS=46);
```

```
   TITLE "M3: AGE ADJUSTED RATES";
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M3ART;
 RUN;

                     /**MACRO ENDS**/
%MEND YXXM3003;
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM304)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/07/01
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM304)
* PGMR      :HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    : D#MSQ.@MSQ00M3.PGMLIB99  - MACRO LOCATION
*             D#MSQ.@MSQ0000.SAS.YxxM1003.M1CNTSxx - file with
*             Denominator counters by subgroups
*             D#MSQ.@MSQ0000.SAS.YxxM3001.YxxM3IN1 - intermediate file
*             with counters of admissions by age,sex,subgroups for
*             each SPG
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3004.YxxM3CNT - file with counter
*             of admissions for yeach SPG and each subgroup
*             D#MSQ.@MSQ0000.SAS.YxxM3004.YxxM3URT - file with
*             unadjusted admission rates for each SPG and each subgroup
* PURPOSE   :create SAS files with counters and unadjusted rate
*             of admissions  by SPG and SUBGROUPS
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*              To run this program for a particular year it is
*              necessary to change input/output data set names  and
*              a global macro variable YY.
*
* STEPS     : 1. create table shell with all the rows and columns.
*             2. read in counter of discharges.
*             3. sum to vars*spg level and create final count file.
*             4. merge to the table shell to set missing cells to 0,
*                write to output file.
*             5. read in counters from Denominator and
*                create unadjusted rates per 1000, write to output
*                file.
*********************************************************************;
       /**INCLUDE MACRO**/

%INCLUDE IN0(M3GENERL)/ SOURCE2;
%INCLUDE IN0(YXXM3004)/ SOURCE2;

 *********************************************************************;
 * step 1. table shell
 *********************************************************************;
       /**MACRO CALL**/
%M3GENERL(VARNAME=SPG,VARCOUNT=142,VLENGTH=3,YY=&SYSPARM)

 *********************************************************************;
 * steps 2-5: create datasets with counters and rates
 *********************************************************************;
       /**MACRO CALL**/ ;
%YXXM3004(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3004)

```
%MACRO YXXM3004(YY=99);

%*******************************************************************
*******************************************************************

1  MACRO NAME:           YXXM3004
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(YXXM3004)
3  WRITTEN BY:           HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate unadjusted rates of admissions
                         by SPG and SUBGROUPS.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

9  STEPS     : first step is done in the main program
*             2. read in counter of discharges.
*             3. sum to vars*spg level and create final count file.
*             4. merge to the table shell to set missing cells to 0,
*                write to output file.
*             5. read in counters from Denominator and
*                create unadjusted rates per 1000, write to output
*                file.
*******************************************************************
*******************************************************************;
              /**CODE BEGINS**/

 *******************************************************************;
 * Step 1. Input counters from file YXXM3IN1 and write them out to
 * INI2.
 *******************************************************************;

 DATA INI2 ERR1;
      SET IN2.Y&YY.M3IN1 ;
      IF Y&YY.FAGE ='' OR Y&YY.SEX='' THEN OUTPUT ERR1;
      ELSE OUTPUT INI2;
 RUN;
 PROC PRINT U DATA=ERR1(OBS=46);
    TITLE "!!! NB !!!!: DISCHARGE FILE ERROR: AGE OR SEX IS MISSING";
 RUN;

 *******************************************************************;
 * Step 2. Sum the count variable (YXXNSPG) by YXXVARS, YXXSPG,
 * and YEAR and output to INI21.
 *******************************************************************;

 PROC SUMMARY DATA=INI2 NWAY;
      CLASS  Y&YY.VARS Y&YY.SPG YEAR;
      VAR Y&YY.NSPG ;
      OUTPUT OUT=INI21(KEEP=YEAR Y&YY.SPG Y&YY.VARS Y&YY.NSPG Y&YY.FAGE)
             SUM=;
 RUN;
 PROC PRINT U  DATA=INI21(OBS=46);
      TITLE "data check: DISCHARGE FILE SUMMED ON VARS*SPG LEVEL";
 RUN;
```

```
*******************************************************************;
* Step 3. Merge INI21 and the table shell from M3GENERL (SHELL) by
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3004) -- CONTINUED

* YXXVARS and YXXSPG and output to YXXM3CNT.  Values in SHELL that
* do not have a match to INI21 are set to 0.
*******************************************************************;

DATA OUT1.Y&YY.M3CNT(LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M3004"
                     SORTEDBY=Y&YY.VARS Y&YY.SPG )
     ERR2;
     MERGE INI21(IN=IN1)
           SHELL(IN=IN2 RENAME=(SPG=Y&YY.SPG VARS=Y&YY.VARS));
     BY Y&YY.VARS Y&YY.SPG ;
     IF IN2 & NOT IN1 THEN Y&YY.NSPG = 0;
     ELSE
     IF IN1 & NOT IN2 THEN OUTPUT ERR2;
     OUTPUT OUT1.Y&YY.M3CNT;
RUN;
PROC PRINT U DATA=ERR2(OBS=46);
   TITLE "!!! NB !!!!: no column or row in the table shell";
RUN;
PROC PRINT U DATA=OUT1.Y&YY.M3CNT(OBS=46);
   TITLE "M3: FINAL COUNTERS FILE";
RUN;

*******************************************************************;
* Step 4. To construct the unadjusted rates per 1000 for each
* cell, merge the YXXM3CNT file to the Denominator count file
* M1CNTSXX.  Create the unadjusted rate per 1000 (YXXRSPG) and
* output to YXXM3URT.
*******************************************************************;

DATA OUT2.Y&YY.M3URT(KEEP=YEAR Y&YY.VARS Y&YY.SPG  Y&YY.RSPG
                     LABEL="PROJECT: MQMS, PROGRAM: Y&YY.M3004"
                     SORTEDBY=Y&YY.VARS Y&YY.SPG )
     ERR3 ERR4;
     MERGE IN1.M1CNTS&YY (KEEP=Y&YY.VARS Y&YY.CTM14 IN=IN2)
           OUT1.Y&YY.M3CNT(IN=IN1);
     BY Y&YY.VARS;
     IF IN1 & NOT IN2 THEN OUTPUT ERR3;
     ELSE
     IF IN2 & NOT IN1 THEN OUTPUT ERR4;
     ELSE DO;
        Y&YY.RSPG = 1000* (Y&YY.NSPG/Y&YY.CTM14);
        OUTPUT OUT2.Y&YY.M3URT;
     END;
RUN;
PROC PRINT U DATA=ERR3(OBS=46);
TITLE "!!! NB !!!!: DENOMINATOR FILE ERROR: subgroupis missing";
RUN;
PROC PRINT U DATA=ERR4(OBS=46);
TITLE "!!! NB !!!!: COUNTER DO NOT HAVE ALL ROWS/COLUMNS";
RUN;
PROC PRINT U DATA=OUT2.Y&YY.M3URT(OBS=46);
   TITLE "M3: FINAL RATE FILE";
RUN;
```

```
 PROC CONTENTS DATA=OUT1.Y&YY.M3CNT;
      TITLE "M3: DISCHARGE COUNTERS FILE";
 RUN;
 PROC CONTENTS DATA=OUT2.Y&YY.M3URT;
      TITLE "M3: DISCHARGE RATE FILE";
 RUN;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3004) -- CONTINUED

```
                     /**MACRO ENDS**/
%MEND YXXM3004;
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM305)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :07/05/01
* EXEC DATE :08/13/01
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM305)
* PGMR      :HELEN MARGULIS(MODIFIED BY MATTHEW URATO)
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.YxxM3002.Y99M3RAS
*                          -age/sex adjusted rates
*            D#MSQ.@MSQ0000.SAS.YxxM3003.Y99M3ART
*                          -age adjusted rates
*            D#MSQ.@MSQ0000.SAS.YxxM3004.Y99M3CNT
*                          -counters of admissions
*            D#MSQ.@MSQ0000.SAS.YxxM3004.Y99M3URT
*                          -unadjusted rates
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3005.Y99M3RAS.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.Y99M3ART.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.Y99M3CNT.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.Y99M3URT.XPT
* PURPOSE   :create SAS transport files to download to PC
*
* COMMENTS  : xx - is 2 digit for a year (for example 99).
*             To run this program for a particular year it is
*             necessary to change input/output data set names  and
*             a global macro variable YY.
*             Global macro variable NCOL gives the number of SPG
*             groups which will be transpose to number of columns.
*********************************************************************;
        /**INCLUDE MACRO**/;

 %INCLUDE IN0(YXXM3005)/ SOURCE2;

       /**MACRO CALL**/

 %YXXM3005(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3005)

```
%MACRO YXXM3005(YY=99);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM3005
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(YXXM3005)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 AUGUST 21, 2001
6  PROJECT:              MQMS
7  PURPOSE:              to calculate unadjusted rates of admissions
                         by SPG and SUBGROUPS.

8  COMMENTS  : xx - is 2 digit for a year (for example 99).

9  PURPOSE   :create SAS transport files to download to PC

*****************************************************************
*****************************************************************;
LIBNAME OUT1 XPORT;
LIBNAME OUT2 XPORT;
LIBNAME OUT3 XPORT;
LIBNAME OUT4 XPORT;
%LET NCOL=142;

*****************************************************************;
* Step 1: Transpose the input datasets to have the subgroups in
* rows and the counts or rates as columns.
*****************************************************************;

 %MACRO TRANS(INF=,OUTF=,PREF=);
    DATA INI;
         SET &INF;
         SPGN = INPUT (Y&YY.SPG,3.);
    RUN;
    PROC SORT DATA=INI; BY Y&YY.VARS SPGN; RUN;
    PROC TRANSPOSE DATA=INI PREFIX=&PREF
         OUT=&OUTF(KEEP=Y&YY.VARS &PREF.1-&PREF.&NCOL);
         BY Y&YY.VARS;
         VAR Y&YY.&PREF;
    RUN;
 %MEND TRANS;

*****************************************************************;
* Step 2: Create SAS transport files.
*****************************************************************;
*****************************************************************;
 %TRANS(INF=IN1.Y&YY.M3RAS,OUTF=OUT1.Y&YY.M3RAS,PREF=ASSPG);
 %TRANS(INF=IN2.Y&YY.M3ART,OUTF=OUT2.Y&YY.M3ART,PREF=AASPG);
 %TRANS(INF=IN3.Y&YY.M3CNT,OUTF=OUT3.Y&YY.M3CNT,PREF=NSPG);
 %TRANS(INF=IN4.Y&YY.M3URT,OUTF=OUT4.Y&YY.M3URT,PREF=RSPG);

                     /**MACRO ENDS**/
%MEND YXXM3005;
```

D#MSQ.@MSQ00M3.PGMLIB99(SYSM306)

```
************************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* REQ DATE  :02/26/02
* EXEC DATE :02/26/02
* PGM       :D#MSQ.@MSQ00M3.PGMLIB99(SYSM306)
* PGMR      :MATTHEW URATO
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.YxxM3005.YxxM3RAS.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.YxxM3ART.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.YxxM3CNT.XPT
*            D#MSQ.@MSQ0000.SAS.YxxM3005.YxxM3URT.XPT
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.YxxM3006.RTFRASxx
*            D#MSQ.@MSQ0000.SAS.YxxM3006.RTFARTxx
*            D#MSQ.@MSQ0000.SAS.YxxM3006.RTFCNTxx
*            D#MSQ.@MSQ0000.SAS.YxxM3006.RTFURTxx
* PURPOSE   :TO CREATE RICH TEXT FILES THAT CAN BE READ ON A
*            PC BY MICROSOFT WORD FROM THE XPORT TABLES CREATED
*            IN MACRO PROGRAM YXXM306.
************************************************************************;

        /**INCLUDE MACRO**/;

%INCLUDE IN0(YXXM3006)/ SOURCE2;

       /**MACRO CALL**/

%YXXM3006(YY=&SYSPARM)
```

D#MSQ.@MSQ00M3.PGMLIB99(YXXM3006)

```
%MACRO YXXM3006(YY=99,IN1=IN1,IN2=IN2,IN3=IN3,IN4=IN4,
                OUT1=OUT1,OUT2=OUT2,OUT3=OUT3,OUT4=OUT4);
%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM3006
2  LOCATION:             D#MSQ.@MSQ00M3.PGMLIB99(YXXM3006)
3  WRITTEN BY:           MATTHEW URATO
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 FEB 01, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      PRINT OUT THE 4 MAJOR COUNT FILES FOR M.3 AND SAVE THE OUTPUT
      AS RICH TEXT FILES TO BE READ LOCALLY ON A PC WITH MICROSOFT
      WORD.

8  PARAMETERS:
      THERE ARE 9 KEYWORD PARAMETERS:
      YY IN1 IN2 IN3 IN4 OUT1 OUT2 OUT3 OUT4

YY
IS THE 2-DIGIT YEAR OF THE DENOMINATOR FILE BEING READ INTO SAS
E.G.:
YY=99 (THIS IS THE DEFAULT)

IN1
IS THE SAS LIBREF-JCL DDNAME FOR A M.3 COUNT FILE CREATED IN PROGRAM
'YXXM3005' FOR AGE-SEX ADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM3005.YXXM3RAS.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN2
IS THE SAS LIBREF-JCL DDNAME FOR A M.3 COUNT FILE CREATED IN PROGRAM
'YXXM3005' FOR AGE ADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM3005.YXXM3ART.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN3
IS THE SAS LIBREF-JCL DDNAME FOR A M.3 COUNT FILE CREATED IN PROGRAM
'YXXM3005' FOR RAW NUMERATOR COUNTS.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM3005.YXXM3CNT.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

IN4
IS THE SAS LIBREF-JCL DDNAME FOR A M.3 COUNT FILE CREATED IN PROGRAM
'YXXM3005' FOR UNADJUSTED RATES.
FILE LOCATION: D#MSQ.#MSQ0000.SAS.YXXM3005.YXXM3URT.XPT (XX=YEAR).
LAST QUALIFIER IS INTERNAL SAS NAME.
E.G.: IN1=IN1  (THIS IS THE DEFAULT)

OUT1
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
```

FOR THE FILE IN IN1.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM3006.RTFRASXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT2
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN2.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM3006.RTFARTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT3
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN3.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM3006.RTFCNTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

OUT4
IS THE SAS LIBREF-JCL DDNAME FOR THE RICH TEXT FORMATTED FILE
FOR THE FILE IN IN4.
FILE LOCATION: D#MSQ.@MSQ0000.SAS.YXXM3006.RTFURTXX (XX=YEAR)
E.G.: OUT1=OUT1 (THIS IS THE DEFAULT)

9 OTHER MACRO VARIABLES OF NOTE: NONE

10 OTHER NOTES: NONE

11 EXAMPLE MACRO CALL:
%YXXM3006(YY=93)

```
**************************************************************************
**************************************************************************;

                  /**CODE BEGINS**/

*************************************************************************;
* Step 1. Replace YXXVARS values with descriptive label-like
* values to be used in tables.
*************************************************************************;

PROC FORMAT;

VALUE $GROUPS

'FAG1' ='0-54' 'FAG2'='55-64'
'FAG3' ='65-69' 'FAG4'='70-74'  'FAG5'='75-79'
'FAG6'='80-84' 'FAG7'='85-89' 'FAG8'='90-94'
'FAG9'='95-UP'
'SEX1'='MALE'
'SEX2'='FEMALE'
'RAC0'='UNKNOWN'
'RAC1'='WHITE'
'RAC2'='AFRICAN AMERICAN'
'RAC3'='OTHER'

'MDS1'='AGED WITHOUT ESRD'
'MDS2'='AGED WITH ESRD'
```

```
'MDS3'='DISABLED WITHOUT ESRD'
'MDS4'='DISABLED WITH ESRD'
'MDS5'='ESRD ONLY'

'DUA1'='ENROLLED IN MEDICAID'
'DUA0'='NOT ENROLLED IN MEDICAID'
```


```
'URB0'='RURAL'
'URB1'='URBAN'

'STA01'='ALABAMA'
'STA02'='ALASKA'
'STA03'='ARIZONA'
'STA04'='ARKANSAS'
'STA05'='CALIFORNIA'
'STA06'='COLORADO'
'STA07'='CONNECTICUT'
'STA08'='DELAWARE'
'STA09'='DISTRICT OF COLUMBIA'
'STA10'='FLORIDA'
'STA11'='GEORGIA'
'STA12'='HAWAII,GUAM'
'STA13'='IDAHO'
'STA14'='ILLINOIS'
'STA15'='INDIANA'
'STA16'='IOWA'
'STA17'='KANSAS'
'STA18'='KENTUCKY'
'STA19'='LOUISIANA'
'STA20'='MAINE'
'STA21'='MARYLAND'
'STA22'='MASSACHUSETTS'
'STA23'='MICHIGAN'
'STA24'='MINNESOTA'
'STA25'='MISSISSIPPI'
'STA26'='MISSOURI'
'STA27'='MONTANA'
'STA28'='NEBRASKA'
'STA29'='NEVADA'
'STA30'='NEW HAMPSHIRE'
'STA31'='NEW JERSEY'
'STA32'='NEW MEXICO'
'STA33'='NEW YORK'
'STA34'='NORTH CAROLINA'
'STA35'='NORTH DAKOTA'
'STA36'='OHIO'
'STA37'='OKLAHOMA'
'STA38'='OREGON'
'STA39'='PENNSYLVANIA'
'STA40'='PUERTO RICO'
'STA41'='RHODE ISLAND'
'STA42'='SOUTH CAROLINA'
'STA43'='SOUTH DAKOTA'
'STA44'='TENNESSEE'
'STA45'='TEXAS'
'STA46'='UTAH'
```

```
'STA47'='VERMONT'
'STA48'='VIRGIN ISLANDS'
'STA49'='VIRGINIA'
'STA50'='WASHINGTON'
'STA51'='WEST VIRGINIA'
'STA52'='WISCONSIN'
'STA53'='WYOMING'

'CDI0'='OUTSIDE U.S.(CD)'
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3006) -- CONTINUED
```
'CDI1'='NEW ENGLAND'
'CDI2'='MIDDLE ATLANTIC'
'CDI3'='EAST NORTH CENTRAL'
'CDI4'='WEST NORTH CENTRAL'
'CDI5'='SOUTH ATLANTIC'
'CDI6'='EAST SOUTH CENTRAL'
'CDI7'='WEST SOUTH CENTRAL'
'CDI8'='MOUNTAIN'
'CDI9'='PACIFIC'

'CRE0'='OUTSIDE U.S.(CR)'
'CRE1'='NORTHEAST'
'CRE2'='MIDWEST'
'CRE3'='SOUTH'
'CRE4'='WEST'
;

RUN;

OPTIONS LS=80 NODATE;

%LET ADJTYP=AGE_SEX_ADJUSTED AGE_ADJUSTED RAW_NUMERATORS UNADJUSTED;
%LET FILEN=M3RAS M3ART M3CNT M3URT;

*****************************************************************;
* Step 2. Set up do loop that reads through the 4 transport files.
* Read in SAS transport file and copy it into a standard SAS file.
* Print these files and save them as RTFs.
*****************************************************************;

%DO I = 1 %TO 4;

%LET ADJTYPE=%SCAN(&ADJTYP,&I);
%LET FILENM=%SCAN(&FILEN,&I);

LIBNAME IN&I XPORT;

PROC COPY IN=IN&I OUT=WORK;
RUN;

ODS RTF FILE=OUT&I TRANTAB=ASCII;

%IF &FILENM = M3CNT %THEN %DO;
TITLE "M3 DISCHARGES FOR YEAR &YY";
TITLE2 'FULL YEAR FFS MEDICARE ENROLLEES';
TITLE3 "&ADJTYPE";
%END;
```

```
%ELSE %DO;
TITLE "M3 DISCHARGE RATES PER 1000 FOR YEAR &YY";
TITLE2 'FULL YEAR FFS MEDICARE ENROLLEES';
TITLE3 "&ADJTYPE RATES";
%END;

PROC PRINT DATA=Y&YY&FILENM HEADING=V WIDTH=MIN ROUND;
FORMAT Y&YY.VARS $GROUPS.;
ID Y&YY.VARS;
RUN;

ODS RTF CLOSE;
```
D#MSQ.@MSQ00M3.PGMLIB99(YXXM3006) -- CONTINUED

```
%END;

        /**MACRO ENDS**/

%MEND YXXM3006;
```

PROGRAMS –

ACUTE MYOCARDIAL INFARCTION,

HEART FAILURE, AND STROKE

D#MSQ.@MSQ00M4.PGMLIB92(M4PROD92)

```
//TI10M492 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD92 EXEC M4PROD,YEAR=92
```

D#MSQ.@MSQ00M4.PGMLIB93(M4PROD93)

```
//TI10M493 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD93 EXEC M4PROD,YEAR=93
```

D#MSQ.@MSQ00M4.PGMLIB94(M4PROD94)

```
//TI10M494 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD94 EXEC M4PROD,YEAR=94
```

D#MSQ.@MSQ00M4.PGMLIB95(M4PROD95)

```
//TI10M495 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD95 EXEC M4PROD,YEAR=95
```

D#MSQ.@MSQ00M4.PGMLIB96(M4PROD96)

```
//TI10M496 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD96 EXEC M4PROD,YEAR=96
```

D#MSQ.@MSQ00M4.PGMLIB97(M4PROD97)

```
//TI10M497 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD97 EXEC M4PROD,YEAR=97
```

D#MSQ.@MSQ00M4.PGMLIB98(M4PROD98)

```
//TI10M498 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD98 EXEC M4PROD,YEAR=98
```

D#MSQ.@MSQ00M4.PGMLIB99(M4PROD99)

```
//TI10M499 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD99 EXEC M4PROD,YEAR=99
```

D#MSQ.@MSQ00M4.PGMLIB00(M4PROD00)

```
//TI10M400 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD00 EXEC M4PROD,YEAR=00
```

D#MSQ.@MSQ00M4.PGMLIB01(M4PROD01)

```
//TI10M401 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M4PROD FOR A PARTICULAR YEAR (E.G., M4PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M4.PGMLIB99'
//M4PROD01 EXEC M4PROD,YEAR=01
```

D#MSQ.@MSQ00M4.PGMLIB99(M4PROD)

```
//M4PROD PROC
//*********************************************************************
//* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR M4 TABLE CREATIONS
//*********************************************************************
//*
//YXXM4001 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(1300,300))
//     DD SPACE=(CYL,(1300,300))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//INL DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB&YEAR
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M2007.MEDPRG&YEAR,DISP=SHR
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMPX,
//       LABEL=(EXPDT=2002/120),UNIT=MCART
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.M4FLAGS,
//       LABEL=(EXPDT=2007/365),UNIT=(MCART,,DEFER)
//OUT3 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.M4OTHER,
//       LABEL=(EXPDT=2007/365),UNIT=(MCART,,DEFER)
//OUT4 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP4,
//       LABEL=(EXPDT=2002/120),UNIT=(MCART,,DEFER)
//OUT5 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP5,
//       LABEL=(EXPDT=2002/120),UNIT=(MCART,,DEFER)
//OUT6 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP6,
//       LABEL=(EXPDT=2002/120),UNIT=(MCART,,DEFER)
//OUT7 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.DISCH&YEAR,
//       LABEL=(EXPDT=2007/365),UNIT=(MCART,,DEFER)
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM401),DISP=SHR
//DELSTEP EXEC PGM=IEFBR14,COND=(4,LT)
//DX DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMPX,DISP=(OLD,DELETE)
//D4 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP4,DISP=(OLD,DELETE)
//D5 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP5,DISP=(OLD,DELETE)
//D6 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.TEMP6,DISP=(OLD,DELETE)
//*********************************************************************
//*PURPOSE   :CREATE SUBGROUP TO TOTAL BENE RATIOS TO BE USED
//*           :IN THE RATE PER 1000 BENES TABLES
//*********************************************************************
//YXXM402A EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(1300,300))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M1002.DENOMB&YEAR,DISP=SHR
//OUT DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M402A.GRPWT&YEAR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(1000,200),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM42A),DISP=SHR
//*********************************************************************
//* PURPOSE   :CREATE SUBGROUP TO TOTAL BENE RATIOS AGE AND
//*           :AGE/SEX ADJUSTED FOR THE 1999 NATIONAL MEDICARE FFS
//*           :POPULATION. THIS FILE WILL BE USED FOR ALL YEARS.
```

```
//*******************************************************************
//NULLSTEP EXEC PGM=IEFBR14,COND=(4,LT)
// IF (&YEAR EQ 99) THEN
```
D#MSQ.@MSQ00M4.PGMLIB99(M4PROD) -- CONTINUED

```
//YXXM402B EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(1300,300))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//OUT DD DSN=D#MSQ.@MSQ0000.SAS.Y99M402B.ADJWT99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(1000,200),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM42B),DISP=SHR
// ENDIF
//*******************************************************************
//* PURPOSE:  MERGE SUBGP/BENE RATES PER 1000 FROM DENOMINATOR FILE AND
//*           AGE SEX ADJUSTMENTS FORM THE 1999 DENOMINATOR FILE ONTO
//*           THE MEDPAR FILE
//*******************************************************************
//YXXM4003 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(1300,300))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y99M402B.ADJWT99,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M402A.GRPWT&YEAR,
//      DISP=SHR
//IN3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4001.DISCH&YEAR,DISP=SHR
//OUT DD DISP=(NEW,CATLG,KEEP),VOL=(,,,10),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4003.M4TAB&YEAR,
//       LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM403),DISP=SHR
//*******************************************************************
//* PURPOSE :TO CREATE TABLES OF DISCHARGES BY COHORT FOR ALL SUBGROUPS
//*******************************************************************
//YXXM4004 EXEC SAS8,REGION=8000K,PARM=('ERRORS=0 NEWS',
//     'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4003.M4TAB&YEAR,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE1,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE2,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE3,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE4,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT5 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE5,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT6 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE6,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT7 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE7,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT8 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE8,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT9 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE9,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
```

```
//OUT10 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE10,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT11 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE11,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT12 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE12,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
```
D#MSQ.@MSQ00M4.PGMLIB99(M4PROD) -- CONTINUED

```
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE13,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4004.TABLE14,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM404),DISP=SHR
//*********************************************************************
//* PURPOSE   :CREATE TABLES OF FREQUENCY, RATE, LENGTH OF STAY,
//*            AND COSTS OF HOSPITAL DISCHARGES FOR HCQIP CLINICAL
//*            PRIORITY AREAS OF STROKE, AMI, HEART FAILURE,
//*            AND PNEUMONIA BY GEOGRAPHIC AREA, AGE GROUP, SEX
//*            RACE, MEDICARE STATUS, AND OTHER DISADVANTAGED
//*            STATUS FOR THE YEARS 1992 - 2000
//*********************************************************************
//YXXM4005 EXEC SAS8,REGION=8000K,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YEAR'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M4.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4003.M4TAB&YEAR,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE15,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE16,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE17,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE18,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT5 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE19,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT6 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE20,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT7 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE21,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT8 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE22,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT9 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE23,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT10 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE24,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT11 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE25,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT12 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YEAR.M4005.TABLE26,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M4.PGMLIB99(SYSM405),DISP=SHR
//M4PROD PEND
//*************************************************************
```

D#MSQ.@MSQ00M4.PGMLIB99(SYSM401)

```
*********************************************************************
* PROJ      :MQMS
* REQ BY    :MQMS TEAM
* EXEC DATE :SEPTEMBER 2002
* PGM       :D#MSQ.@MSQ00M4.PGMLIB99(SYSM401)
* PGMR      :LARSEN
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUT     :D#MSQ.@MSQ0000.SAS.YXXM2007.MEDPRGXX
* OUTPUT    :D#MSQ.@MSQ0000.SAS.YXXM4001.DISCH  (FILE FOR M4 TABLES)
*           :D#MSQ.@MSQ0000.SAS.YXXM4001.M4FLAGS (COHORT DISCHARGES)
*           :D#MSQ.@MSQ0000.SAS.YXXM4001.M4OTHER (OTHER DISCHARGES)
* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR M4 TABLE CREATIONS
*********************************************************************;

 %INCLUDE IN0(YXXM4001)/SOURCE2; * MACRO TO CREATE COHORTS FOR M4;

 %YXXM4001(&SYSPARM);
```

D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001)

```
%MACRO YXXM4001(YY);

*********************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* EXEC DATE :OCTOBER 2002
* PGM       :D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001)
* PGMR      :LARSEN
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUT     :D#MSQ.@MSQ0000.SAS.Y&YYM2007.MEDPRGYY
* OUTPUTS   :D#MSQ.@MSQ0000.SAS.Y&YYM4001.DISCH
*           :D#MSQ.@MSQ0000.SAS.Y&YYM4001.M4FLAGS
*           :D#MSQ.@MSQ0000.SAS.Y&YYM4001.M4OTHER
* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR TABLE CREATIONS
* STEP1:  SORT M2 FILE BY THE X-REF HICNO
* STEP2:  CREATE THE FOLLOWING VARIABLES
*            Y&YY.SADM - SAS DATE OF Y&YYADMIT
*            Y&YY.SDIS - SAS DATE OF Y&YYDISCH
*            Y&YY.LOS - LENGTH OF STAY
* STEP3:  CREATE COHORT FLAGS - HF P1-P7 S1-S4 M1-M2
*                 FLAG=1 IF IT IS THAT TYPE OF DISCHARGE
*                 FLAG=0 IF IT IS NOT THAT TYPE OF DISCHARGE
* STEP4:  OUTPUT DISCHARGE LEVEL FILE OF COHORT DISCHARGES
*         D#MSQ.@MSQ0000.SAS.Y&YYM4001.M4FLAGS
*         OUTPUT DISCHARGE LEVEL FILE OF OTHER NON-COHORT DISCHARGES
*         D#MSQ.@MSQ0000.SAS.Y&YYM4001.M4OTHER
*         THE NON-COHORT DISCHARGE FILE WILL BE USED IN THE M5 ANALYSIS
*         TO IDENTIFY READMISSIONS
* STEP5:  CREATE TOTDIS - TOTAL NUMBER OF DISCHARGES FOR THIS HICNO
*         MAXIMUM NUMBER OF DISCHARGES IS LIMITED TO 8
* STEP6:  MERGE TOTDIS ONTO COHORT DISCHARGE FILE BY HICNO
* STEP7:  CREATE THE FOLLOWING VARIABLE
*            Y&YYCOHRT - TAKES THE VALUE OF HF, P1-P7, S1-S4, OR M1-M2
*                        DEPENDING ON THE VALUE OF THE COHORT FLAG
*                        IF MORE THAN 1 COHORT FLAG IS 1 THEN THE
*                        DISCHARGE IS WRITTEN OUT ONCE FOR EACH
*                        COHORT IT REPRESENTS
* STEP8: SORT FILE BY COHORT HICNO
* STEP9: CREATE THE FOLLOWING VARIABLES
*             BENE&YY=1 IF IT IS THE FIRST DISCHARGE FOR THAT BENE
*                    FOR THAT COHORT. OTHERWISE BENE&YY=0
*             CCDIS&YY=NUMBER OF DISCHARGES FOR THIS COHORT FOR THAT BENE
*                    MAX NUMBER OF DISCHARGES IS LIMITED TO 8
*             DISCH&YY=1 FOR EVERYONE. IT WILL BE USED TO SUM DISCHARGES
* STEP10: MERGE CCDIS&YY ONTO COHORT DISCHARGE FILE BY COHORT HICNO
* STEP11: MULTIPLY CCDIS&YY AND TOTDIS&YY BY BENE&YY SO THAT IT IS ONLY
*         ON THE FIRST DISCHARGE FOR THAT BENE FOR THAT COHORT
*         OTHERWISE IT IS 0
* STEP12:  CREATE Y&YYCCGRP TO BE 1-13 DEPENDING ON THE
*         COHORT.  THIS KEEPS THE COHORTS IN ORDER WHEN
*         PRODUCING THE TABLES
*********************************************************************;

 %INCLUDE INL(Y&YY.DIAGS)/SOURCE2;
 %INCLUDE IN0(YXXFLAGS)/SOURCE2;
```

```
 %LET VARLIST=%STR(Y&YY.STATE Y&YY.CDIV Y&YY.CREG Y&YY.URBN Y&YY.SEX
      Y&YY.FAGE Y&YY.RACE Y&YY.DUAL Y&YY.MDST);
```

D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001) -- CONTINUED

```
 %Y&YY.DIAGS;

 /*******************************************************
  * STEP1:  SORT M2 ANALYTIC FILE BY HICNO               *
  *******************************************************/

PROC SORT DATA=IN.MEDPRG&YY
   OUT=OUT1.TEMPX;  BY Y&YY.HICNO;

DATA OUT2.M4FLAGS OUT3.M4OTHER X(KEEP=Y&YY.HICNO TOTDIS&YY);
  SET OUT1.TEMPX; BY Y&YY.HICNO;

 /*******************************************************
  * STEP2:  CREATE THE FOLLOWING VARIABLES               *
  *          Y&YY.SADM - SAS DATE OF Y&YY.ADMIT          *
  *          Y&YY.SDIS - SAS DATE OF Y&YY.DISCH          *
  *          Y&YY.LOS - LENGTH OF STAY (MIN=1, MAX=90)   *
  *******************************************************/;

  Y&YY.SADM=DATEJUL(Y&YY.ADMIT);
  Y&YY.SDIS=DATEJUL(Y&YY.DISCH);
  Y&YY.LOS = MAX((Y&YY.SDIS-Y&YY.SADM),1);
  IF Y&YY.LOS>90 THEN Y&YY.LOS=90;

 /*******************************************************
  * STEP3:  CREATE COHORT FLAGS - HF P1-P7 S1-S4 M1-M2   *
  *           FLAG=1 IF IT IS THAT TYPE OF DISCHARGE     *
  *           FLAG=0 IF IT IS NOT THAT TYPE OF DISCHARGE *
  *******************************************************/;

  ARRAY D(10) $ Y&YY.DIA1-Y&YY.DIA10;
  ARRAY P(6) $ Y&YY.SPR1-Y&YY.SPR6;


  %YXXFLAGS(&YY, Y&YY.HF, Y&YY.S, Y&YY.P, Y&YY.M);

   DO K = 1 TO 6 UNTIL(Y&YY.M2=1);
      Y&YY.M2 = (Y&YY.M1 = 1 AND P(K)=:"36" AND P(K) LE "3639");
   END;

  LENGTH Y&YY.HF Y&YY.M1 Y&YY.M2 Y&YY.P1-Y&YY.P7 Y&YY.S1-Y&YY.S4 2;

 /*****************************************************************
  * STEP4: OUTPUT DISCHARGE LEVEL FILE OF COHORT DISCHARGES (M4FLAGS)
  *        OUTPUT DISCHARGE LEVEL FILE OF NON-COHORT DISCHARGES (M4OTHER)
  *****************************************************************/;

  SFLAGS=(SUM(Y&YY.HF,OF Y&YY.P1-Y&YY.P7,OF Y&YY.S1-Y&YY.S4,
         Y&YY.M1)>0);
  IF (SFLAGS>0) THEN OUTPUT OUT2.M4FLAGS;
  ELSE OUTPUT OUT3.M4OTHER;

 /*****************************************************************
```

```
  * STEP5:  CREATE TOTDIS - TOTAL NUMBER OF DISCHARGES FOR THIS HICNO *
  ********************************************************************/;

  RETAIN TOTDIS&YY;
  IF FIRST.Y&YY.HICNO THEN TOTDIS&YY=0;
  TOTDIS&YY=SUM(TOTDIS&YY,1);
```
D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001) -- CONTINUED
```
  IF (LAST.Y&YY.HICNO) THEN OUTPUT X;

RUN;

 /*******************************************************************
  * STEP6:  MERGE TOTDIS ONTO COHORT DISCHARGE FILE BY HICNO         *
  *         LIMIT THE NUMBER OF TOTAL DISCHARGES TO 8                *
  ********************************************************************/;

DATA OUT4.TEMP4(DROP=Y&YY.HF Y&YY.M1 Y&YY.P1-Y&YY.P7 Y&YY.S1-Y&YY.S4);
  MERGE
  OUT2.M4FLAGS(KEEP=&VARLIST Y&YY.SADM Y&YY.SDIS Y&YY.LOS Y&YY.BPAY
  Y&YY.TPAY Y&YY.HICNO Y&YY.DETH Y&YY.VDOD
  Y&YY.HF Y&YY.M1 Y&YY.P1-Y&YY.P7 Y&YY.S1-Y&YY.S4
  IN=INF) X; BY Y&YY.HICNO;
  IF (INF);
  IF TOTDIS&YY>8 THEN TOTDIS&YY=8;

 /*******************************************************************
  * STEP7:  CREATE THE FOLLOWING VARIABLE                            *
  *          Y&YY.COHRT - TAKES THE VALUE OF HF, P1-P7, S1-S4, OR M1 *
  *                       DEPENDING ON THE VALUE OF THE COHORT FLAG  *
  *                       IF MORE THAN 1 COHORT FLAG IS 1 THEN THE   *
  *                       DISCHARGE IS WRITTEN OUT ONCE FOR EACH     *
  *                       COHORT IT REPRESENTS                       *
  ********************************************************************/;


  FORMAT Y&YY.COHRT $2.;

  %MACRO PUTOUT(CC);
     IF (Y&YY.&CC=1) THEN DO;
        Y&YY.COHRT="&CC";
        OUTPUT OUT4.TEMP4;
     END;
  %MEND PUTOUT;

  %PUTOUT(HF);
  %PUTOUT(S1);
  %PUTOUT(S2);
  %PUTOUT(S3);
  %PUTOUT(S4);
  %PUTOUT(P1);
  %PUTOUT(P2);
  %PUTOUT(P3);
  %PUTOUT(P4);
  %PUTOUT(P5);
  %PUTOUT(P6);
  %PUTOUT(P7);
  %PUTOUT(M1);
```

```
 /*************************************************
  * STEP8:  SORT FILE BY COHORT HICNO              *
  *************************************************/;

PROC SORT DATA=OUT4.TEMP4 OUT=OUT5.TEMP5; BY Y&YY.COHRT Y&YY.HICNO
  Y&YY.SADM;

DATA OUT6.TEMP6(DROP=CCDIS&YY)
D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001) -- CONTINUED

     Z(KEEP=Y&YY.HICNO CCDIS&YY Y&YY.COHRT);
  SET OUT5.TEMP5; BY Y&YY.COHRT Y&YY.HICNO;

 /*****************************************************************
  * STEP9:  CREATE THE FOLLOWING VARIABLES                        *
  *         BENE&YY=1 IF IT IS THE FIRST DISCHARGE FOR THAT BENE  *
  *                  FOR THAT COHORT. OTHERWISE BENE&YY=0         *
  *         CCDIS&YY=NUMBER OF DISCHARGES FOR THIS COHORT FOR THAT BENE*
  *         DISCH&YY=1 FOR EVERYONE. IT WILL BE USED TO SUM DISCHARGES*
  *****************************************************************/;

  RETAIN CCDIS&YY;
  IF (FIRST.Y&YY.HICNO) THEN DO;
      BENE&YY=1;
      CCDIS&YY=0;
  END;
  CCDIS&YY=SUM(CCDIS&YY,1);
  DISCH&YY=1;
  OUTPUT OUT6.TEMP6;
  IF (LAST.Y&YY.HICNO) THEN OUTPUT Z;
RUN;

 /*****************************************************************
  * STEP10: MERGE CCDIS&YY ONTO COHORT DISCHARGE FILE BY COHORT HICNO *
  *         LIMIT THE MAXIMUN NUMBER OF COHORT DISCHARGES TO 8     *
  *****************************************************************/;

DATA OUT7.DISCH&YY; MERGE OUT6.TEMP6 Z; BY Y&YY.COHRT Y&YY.HICNO;
  IF CCDIS&YY>8 THEN CCDIS&YY=8;

 /*****************************************************************
  * STEP11: MULTIPLY CCDIS&YY AND TOTDIS&YY BY BENE SO THAT IT IS ONLY *
  *         ON THE FIRST DISCHARGE FOR THAT BENE FOR THAT COHORT   *
  *         OTHERWISE IT IS 0                                      *
  *****************************************************************/;

  CCDIS&YY=BENE&YY*CCDIS&YY;
  TOTDIS&YY=BENE&YY*TOTDIS&YY;

  /*************************************************************
   * STEP12:  CREATE Y&YY.CCGRP TO BE 1-13 DEPENDING ON THE    *
   *         COHORT.  THIS KEEPS THE COHORTS IN ORDER WHEN     *
   *         PRODUCING THE TABLES                              *
   *************************************************************/;

  FORMAT Y&YY.CCGRP 1.;
  SELECT(Y&YY.COHRT);
```

```
     WHEN('HF') Y&YY.CCGRP=1;
     WHEN('S1') Y&YY.CCGRP=2;
     WHEN('S2') Y&YY.CCGRP=3;
     WHEN('S3') Y&YY.CCGRP=4;
     WHEN('S4') Y&YY.CCGRP=5;
     WHEN('P1') Y&YY.CCGRP=6;
     WHEN('P2') Y&YY.CCGRP=7;
     WHEN('P3') Y&YY.CCGRP=8;
     WHEN('P4') Y&YY.CCGRP=9;
     WHEN('P5') Y&YY.CCGRP=10;
     WHEN('P6') Y&YY.CCGRP=11;
     WHEN('P7') Y&YY.CCGRP=12;
```
D#MSQ.@MSQ00M4.PGMLIB99(YXXM4001) -- CONTINUED

```
     WHEN('M1') Y&YY.CCGRP=13;
     OTHERWISE;
  END;
RUN;

PROC CONTENTS DATA=OUT7.DISCH&YY;
 TITLE1 'MQMS - PROGRAM=Y&YY.M4001';
RUN;

PROC PRINT DATA=OUT7.DISCH&YY(OBS=100);
 TITLE3 'DATA = D#MSQ.@MSQ0000.SAS.Y&YY.M4001.DISCH&YY';
RUN;

%MEND YXXM4001;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y92DIAGS)

```
%MACRO Y92DIAGS;

*********************************************************************

1  MACRO NAME:           Y92DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y92DIAGS)
3  WRITTEN BY:           Ann Larsen
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 September 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

  %LET HF=%STR (('40201','40211','40291','40401','40411','40491',
       '4280','4281','4289'));

  %LET IS1=%STR(('36234','4330','4331','4332','4333','4338','4339',
       '4340','4341','4349','4350','4351','4358','4359','436'));

  %LET IS2=%STR(('4330','4331','4332','4333','4338','4339','4340',
       '4341','4349','4350','4351','4358','4359','436'));

  %LET IS3=%STR(('42731'));

  %LET IS4=%STR(('3812'));

  %LET PN1=%STR(('4800','4801','4802','4808','4809'));

  %LET PN2=%STR(('481'));

  %LET PN3=%STR(('4820','4821','4822','4823','48230','48231',
       '48232','48239','4824','4828','48281',
       '48282','48283','48289','4829','483','4830','4838'));
```

```
  %LET PN4=%STR(('485','486'));
```

D#MSQ.@MSQ00M4.PGMLIB99(Y92DIAGS) -- CONTINUED

```
  %LET PN5=%STR(('4870'));

  %LET PN6_1=%STR(('0380','0381','0382','0383','03840',
       '03841','03842','03843','03844','03849','0388','0389'));

  %LET PN6_2=%STR(('1363','4661','4800','4801','4802','4808','4809',
      '481','4820','4821','4822','4823','48230','48231','48232',
      '48239','4824','48281','48282','48283','48284','48289','483',
      '4829','4830','4838','485','486','4870','5070',
      '5071','5078','514','5168','5183'));

  %LET PN7_1=%STR(('51881'));

  %LET PN7_2=%STR(('1363','4661','4800','4801','4802','4808','4809',
      '481','4820','4821','4822','4823','48230','48231','48232',
      '48239','4824','48281','48282','48283','48284','48289','483',
      '4829','4830','4838','485','486','4870','5070',
      '5071','5078','514','5168','5183'));

  %LET MI=%STR(('41000','41001','41010','41011','41020','41021',
       '41030','41031','41040','41041','41050','41051','41060',
       '41061','41070','41071','41080','41081','41090','41091'));

%MEND Y92DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y93DIAGS)

```
%MACRO Y93DIAGS;

*********************************************************************

1  MACRO NAME:           Y93DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y93DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction


*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

  %LET HF=%STR (('40201','40211','40291','40401','40411',
       '40491','4280','4281','4289'));

  %LET IS1=%STR(('36234','4330','43300','43301','4331','43310',
       '43311','4332','43320','43321','4333','43330','43331',
       '4338','43380','43381','4339','43390','43391','4340','43400',
       '4341','43401','43410','43411','4349','43490','43491','4350',
       '4351','4358','4359','436'));

  %LET IS2=%STR(('4330','43301','4331',
       '43311','4332','43321','4333','43331',
       '4338','43381','4339','43391','4340','43400',
       '43401','4341','43410','43411','4349','43490','43491','4350',
       '4351','4358','4359','436'));

  %LET IS3=%STR(('42731'));

  %LET IS4=%STR(('3812'));
```

```
  %LET PN1=%STR(('4800','4801','4802','4808','4809'));

  %LET PN2=%STR(('481'));
D#MSQ.@MSQ00M4.PGMLIB99(Y93DIAGS) -- CONTINUED


  %LET PN3=%STR(('4820','4821','4822','48230','48231',
       '48232','48239','4824','48281','48282','48283',
       '48289','4829','4830','4838'));

  %LET PN4=%STR(('485','486'));

  %LET PN5=%STR(('4870'));

  %LET PN6_1=%STR(('0380','03810','03811','03819','0382','0383',
       '03840','03841',03842','03843','03844','03849','0388','0389'));

  %LET PN6_2=%STR(('1363','4661','4800','4801','4802','4808',
       '4809','481','4820','4821','4822','48230','48231','48232',
       '48239','4824','48281','48282','48283','48284','48289',
       '4829','4830','4838','485','486','4870','5070',
       '5071','5078','514','5168','5183'));

  %LET PN7_1=%STR(('51881'));

  %LET PN7_2=%STR(('07888','1363','4661','4800','4801','4802','4808',
       '4809','481','4820','4821','4822','48230','48231','48232',
       '48239','4824','48281','48282','48283','48284','48289',
       '4829','4830','4838','485','486','4870','5070',
       '5071','5078','514','5168','5183'));

  %LET MI=%STR(('41000','41001','41010','41011','41020','41021',
       '41030','41031','41040','41041','41050','41051','41060',
       '41061','41070','41071','41080','41081','41090','41091'));

%MEND Y93DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y94DIAGS)

```
%MACRO Y94DIAGS;

*********************************************************************

1  MACRO NAME:           Y94DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y94DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

8 EXAMPLE MACRO CALL:
%YXXDIAGS

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
             );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));
```

```
      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

D#MSQ.@MSQ00M4.PGMLIB99(Y94DIAGS) -- CONTINUED

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR((
'4820','4821','4822','48230','48231',
'48232','48239','4824','48281','48282','48283',
'48289','4829','4830','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR((
'0380','0381','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR((
'1363','4661','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289',
'4829','4830','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR(('07888',
'1363','4661','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289'  ,
'4829','4830','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y94DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y95DIAGS)

```
%MACRO Y95DIAGS;

*********************************************************************

1  MACRO NAME:           Y95DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y95DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR(
D#MSQ.@MSQ00M4.PGMLIB99(Y95DIAGS) -- CONTINUED

'4820','4821','4822','48230','48231',
'48232','48239','4824','48281','48282','48283',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR((
'0380','0381','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR(('4661',
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR(('4661','07888',
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y95DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y96DIAGS)

```
%MACRO Y96DIAGS;

*********************************************************************

1  MACRO NAME:           Y96DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y96DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR(
D#MSQ.@MSQ00M4.PGMLIB99(Y96DIAGS) -- CONTINUED

'4820','4821','4822','48230','48231',
'48232','48239','4824','48281','48282','48283',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR(
'0380','0381','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR(
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR(('07888',
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR(
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y96DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y97DIAGS)

```
%MACRO Y97DIAGS;

*********************************************************************

1  MACRO NAME:           Y97DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y97DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR((
```
D#MSQ.@MSQ00M4.PGMLIB99(Y97DIAGS) -- CONTINUED
```
'4820','4821','4822','48230','48231',
'48232','48239','4824','48281','48282','48283','48284',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR(('0381',
'0380','03810','03811','03819','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'4824','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y97DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y98DIAGS)

```
%MACRO Y98DIAGS;

*********************************************************************

1  MACRO NAME:           Y98DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y98DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR(('4824',
D#MSQ.@MSQ00M4.PGMLIB99(Y98DIAGS) -- CONTINUED

'4820','4821','4822','48230','48231',
'48232','48239','48240','48241','48249','48281','48282','48283','48284',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR((
'0380','03810','03811','03819','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR(('4824',
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR(('4824',
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y98DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y99DIAGS)

```
%MACRO Y99DIAGS;

*********************************************************************

1  MACRO NAME:           Y99DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y99DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR((
D#MSQ.@MSQ00M4.PGMLIB99(Y99DIAGS) -- CONTINUED

'4820','4821','4822','48230','48231',
'48232','48239','48240','48241','48249','48281','48282','48283','48284',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR((
'0380','03810','03811','03819','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y99DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(Y00DIAGS)

```
%MACRO Y00DIAGS;

*********************************************************************

1  MACRO NAME:           Y00DIAGS
2  LOCATION:             D#MSQ.@MSQ00M4.PGMLIB99(Y00DIAGS)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 SEPTEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO DEFINE 13 DISEASE COHORTS BASED ON DIAGNOSIS AND SURGICAL
      PROCEDURES.
      HF  - heart failure
      IS1 - Stroke or TIA
      IS2 - Stroke only
      IS3 - Atrial Fibrillation
      IS4 - Carotid Endarterectomy
      PN1 - viral pneumonia
      PN2 - Pneumococcal Pneumonia
      PN3 - bacterial Pneumonia
      PN4 - Unspecified Pneumonia
      PN5 - Influenza with Pneumonia
      PN6 - principal diagnosis of septicemia and
            a secondary diagnosis of pneumonia
      PN7 - principal diagnosis of respiratory failure and
            a secondary diagnosis of pneumonia.
      MI  - acute myocardial infarction

*********************************************************************
*********************************************************************;

  %GLOBAL HF IS1 IS2 IS3 IS4 PN1 PN2 PN3 PN4 PN5
          PN6_1 PN6_2 PN7_1 PN7_2 MI;

      %LET HF=%STR ((
'40201','40211','40291','40401','40411','40491','4280','4281','4289')
              );

      %LET IS1=%STR((
'36234','43300','43301','43310','43311','43320','43321','43330','43331',
'43380','43381','43390','43391','43400','43401','43410','43411','43490',
'43491','4350','4351','4353','4358','4359','436'));

      %LET IS2=%STR((
'43301','43311','43321','43331','43381','43391','43400','43401','43410',
'43411','43490','43491','4350','4351','4353','4358','4359','436'));

      %LET IS3=%STR(('42731'));

      %LET IS4=%STR(('3812'));

      %LET PN1=%STR((
'4800','4801','4802','4808','4809'));
```

```
      %LET PN2=%STR(('481'));

      %LET PN3=%STR((
D#MSQ.@MSQ00M4.PGMLIB99(Y00DIAGS) -- CONTINUED

'4820','4821','4822','48230','48231',
'48232','48239','48240','48241','48249','48281','48282','48283','48284',
'48289','4829','4830','4831','4838'));

      %LET PN4=%STR(('485','486'));

      %LET PN5=%STR(('4870'));

      %LET PN6_1=%STR((
'0380','03810','03811','03819','0382','0383','03840',
'03841','03842','03843','03844','03849','0388','0389'));

      %LET PN6_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289',
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET PN7_1=%STR(('51881'));

      %LET PN7_2=%STR((
'1363','46611','46619','4800','4801','4802','4808' ,
'4809','481','4820','4821','4822','48230','48231','48232','48239',
'48240','48241','48249','48281','48282','48283','48284','48289'  ,
'4829','4830','4831','4838',
'485','486','4870','5070','5071','5078','514','5168','5183'));

      %LET MI=%STR((
'41000','41001','41010','41011','41020','41021','41030',
'41031','41040','41041','41050','41051','41060','41061',
'41070','41071','41080','41081','41090','41091'));

%MEND Y00DIAGS;
```

D#MSQ.@MSQ00M4.PGMLIB99(YXXFLAGS)

```
%MACRO YXXFLAGS(YR, HFFLAG, ISFLAG, PNFLAG, MIFLAG);
*****************************************************************

1  MACRO NAME:           YXXFLAGS
2  LOCATION:             D#MSQ.@MSQ00M5.TI03LIB5(YXXFLAGS)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JANUARY 30, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      BASED ON DEFINITION OF COHORTS DONE IN YXXCOHRT MACRO
      TO CREATE 13 DISEASE FLAGS IN EACH DISCHARGE RECORD

      HF - heart failure
      S1 - Stroke or TIA
      S2 - Stroke only
      S3 - Atrial Fibrillation
      S4 - Carotid Endarterectomy
      P1 - viral pneumonia
      P2 - Pneumococcal Pneumonia
      P3 - bacterial Pneumonia
      P4 - Unspecified Pneumonia
      P5 - Influenza with Pneumonia
      P6 - principal diagnosis of septicemia and
           a secondary diagnosis of pneumonia
      P7 - principal diagnosis of respiratory failure and
           a secondary diagnosis of pneumonia.
      MI - acute myocardial infarction

9 PARAMETERS:
      Parameters difine only initial letters of the output flags
      YR     - 2 digit year of data
      HFFLAG - heart failure flag
      ISFLAG - stroke flags
      PNFLAG - Pneumonia flags
      MIFLAG - acute myocardial infarction flag

8 EXAMPLE MACRO CALL:
      %YXXFLAGS(&YY, Y&YY.HF, Y&YY.S, Y&YY.P, Y&YY.MI);

*********************************************************************

      *heart failure;
     &HFFLAG   = (D(1) IN &HF);

      *Stroke or TIA;
     &ISFLAG.1 = (D(1) IN &IS1);

      *Stroke only;
     &ISFLAG.2 = (D(1) IN &IS2);

      *Atrial Fibrillation;
     &ISFLAG.3=0;
     DO I=1 TO 10 WHILE(&ISFLAG.3 = 0);
        &ISFLAG.3 = (D(I) IN &IS3);
```

```
     END;

      *Carotid Endarterectomy;
```
D#MSQ.@MSQ00M4.PGMLIB99(YXXFLAGS) -- CONTINUED
```
     &ISFLAG.4 = (Y&YR.SPR1 = &IS4 OR Y&YR.SPR2 = &IS4 OR
                  Y&YR.SPR3 = &IS4 OR Y&YR.SPR4 = &IS4 OR
                  Y&YR.SPR5 = &IS4 OR Y&YR.SPR6 = &IS4 );

      *viral pneumonia;
     &PNFLAG.1 = (D(1) IN &PN1);

      *Pneumococcal Pneumonia;
     &PNFLAG.2 = (D(1) IN &PN2);

      *bacterial Pneumonia;
     &PNFLAG.3 = (D(1) IN &PN3);

      *Unspecified Pneumonia;
     &PNFLAG.4 = (D(1) IN &PN4);

      *Influenza with Pneumonia;
     &PNFLAG.5 = (D(1) IN &PN5);

      *principal diagnosis of septicemia and
       a secondary diagnosis of pneumonia;
     &PNFLAG.6 = ((D(1) IN &PN6_1) & (
       (D(2) IN &PN6_2) OR (D(3) IN &PN6_2) OR (D(4)  IN &PN6_2) OR
       (D(5) IN &PN6_2) OR (D(6) IN &PN6_2) OR (D(7)  IN &PN6_2) OR
       (D(8) IN &PN6_2) OR (D(9) IN &PN6_2) OR (D(10) IN &PN6_2)));

       *principal diagnosis of respiratory failure and
        a secondary diagnosis of pneumonia;
     &PNFLAG.7 = ( D(1) IN &PN7_1 & (
       (D(2) IN &PN7_2) OR (D(3) IN &PN7_2) OR (D(4)  IN &PN7_2) OR
       (D(5) IN &PN7_2) OR (D(6) IN &PN7_2) OR (D(7)  IN &PN7_2) OR
       (D(8) IN &PN7_2) OR (D(9) IN &PN7_2) OR (D(10) IN &PN7_2)));

       *acute myocardial infarction;
     &MIFLAG.1 = (D(1) IN &MI);




%MEND YXXFLAGS;
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y92M1AG1)

```
//TI0292A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y92M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE
//*
//Y92M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1002.DENOMB92,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y92M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y92M1002.DENOMB92
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y92M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=92;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA      *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.        *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
D#MSQ.@MSQ00M1.PGMLIBAG(Y92M1AG1) -- CONTINUED

          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y93M1AG1)

```
//TI0293A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y93M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE
//*
//Y93M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1002.DENOMB93,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y93M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y93M1002.DENOMB93
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y93M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=93;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y94M1AG1)

```
//TI0294A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y94M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE
//*
//Y94M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1002.DENOMB94,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y94M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y94M1002.DENOMB94
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y94M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=94;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
D#MSQ.@MSQ00M1.PGMLIBAG(Y94M1AG1) -- CONTINUED

          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y95M1AG1)

```
//TI0295A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y95M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE
//*
//Y95M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1002.DENOMB95,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y95M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y95M1002.DENOMB95
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y95M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=95;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA      *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.        *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y96M1AG1)

```
//TI0296A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y96M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE
//*
//Y96M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1002.DENOMB96,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y96M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y96M1002.DENOMB96
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y96M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=96;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y97M1AG1)

```
//TI0297A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y97M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE
//*
//Y97M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1002.DENOMB97,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y97M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y97M1002.DENOMB97
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y97M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=97;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y98M1AG1)

```
//TI0298A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y98M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE
//*
//Y98M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y98M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y98M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=98;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA      *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.        *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
D#MSQ.@MSQ00M1.PGMLIBAG(Y98M1AG1) -- CONTINUED

          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y99M1AG1)

```
//TI0491A1 JOB (MSQ00,CHER),TI04BSCOTT,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y99M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE
//*
//Y99M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :030702
* PGM       :D#MSQ.@MSQ00.TI04LIB(Y99M1AG1)
* PGMR      :BSCOTT
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y99M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=99;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y00M1AG1)

```
//TI0200A1 JOB (MSQ00,CHER),TI02BBAKER,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y00M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE
//*
//Y00M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1002.DENOMB00,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :CURATO/NMCCALL
* EXEC DATE :031202
* PGM       :D#MSQ.@MSQ00.TI02LIB(Y00M1AG1)
* PGMR      :BBAKER
* COMPANY   :HEALTH ECONOMICS RESEARCH, INC.
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y00M1002.DENOMB00
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y00M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=00;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA   *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.     *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
```

```
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00M1.PGMLIBAG(Y01M1AG1)

```
//P1F701A1 JOB (MSQ00,CHER),P1F7WPAN,MSGCLASS=Q,CLASS=K
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ0000.SAS.Y01M1AG1.M13BENE
  DELETE D#MSQ.@MSQ0000.SAS.Y01M1AG1.M14BENE
//*
//Y01M1AG1 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD SPACE=(CYL,(2500,300))
//WORK1 DD SPACE=(CYL,(2500,300))
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1002.DENOMB01,DISP=SHR
//OUT13 DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1AG1.M13BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT14 DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1AG1.M14BENE,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ      :MQMS
* REQ BY    :BILL SCOTT ON BEHALF OF CURATO/NMCCALL
* EXEC DATE :1-2-03
* PGM       :D#MSQ.@MSQ00.P1F7LIB1(Y01M1AG1)
* PGMR      :WESLEY PAN
* COMPANY   :RTI
* INPUTS    :D#MSQ.@MSQ0000.SAS.Y01M1002.DENOMB01
* OUTPUTS   :(OUT1)D#MSQ.@MSQ0000.SAS.Y00M1AG1.M13BENE
*            (OUT2)D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE
* PURPOSE   :CREATING AGGREGATE FILES FOR M1-3 AND M1-4 -
*            OUT1=TOTAL BENEFICIARIES FOR M1-3 BY SUBGROUP VARIABLES
*            OUT2=TOTAL BENEFICIARIES FOR M1-4 BY SUBGROUP VARIABLES
* NOTE      :CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*            OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=01;

 ********************************************************
 * MACRO TO AGGREGATE BENE'S ACROSS ALL SUBGROUPS VIA      *
 * PROC SUMMARY. OUTPUT DATA FILE, CONTENTS, AND PRINT 50  *
 * OBS. WHERE STATEMENT DIFFERENTIATES M1-3 & M1-4.        *
 ********************************************************;

%MACRO AG1_1(N);

 %IF &N=13 %THEN %LET JU=J;
 %ELSE %IF &N=14 %THEN %LET JU= ;

  PROC SUMMARY DATA=IN1.DENOMB&YY
```

```
  (WHERE=(Y&YY.M&N='1'))
    NWAY;
    CLASS Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL&JU
D#MSQ.@MSQ00M1.PGMLIBAG(Y01M1AG1) -- CONTINUED

          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG Y&YY.CDIV;
    OUTPUT OUT=WORK1.TEMP1(DROP=_TYPE_ RENAME=(_FREQ_=Y&YY.COUNT));
  RUN;

  PROC APPEND BASE=OUT&N..M&N.BENE DATA=WORK1.TEMP1;
  RUN;

  PROC DATASETS NOLIST LIBRARY=WORK1;
   DELETE TEMP1;
  RUN;

  PROC CONTENTS DATA=OUT&N..M&N.BENE;
   TITLE1 "MQMS - PROGRAM=Y&YY.M1AG1";
  RUN;

  PROC PRINT DATA=OUT&N..M&N.BENE(OBS=50);
   TITLE2 "OUTPUT DATA=D#MSQ.@MSQ0000.SAS.Y&YY.M1AG1.M&N.BENE";
  RUN;
%MEND AG1_1;

 ************************************************
 * INVOKE MACRO FOR M1-3 MEASURE, THEN M1-4 MEASURE *
 ************************************************;

%AG1_1(13);
%AG1_1(14);
```

D#MSQ.@MSQ00AG.PGMLIB(DISCHPXX)

```
//P1H4MS00 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=G,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(1300,300))
//     DD SPACE=(CYL,(1300,300))
//IN1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4FLAGS
//IN2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.DISCH00
//OUT3 DD DISP=(NEW,CATLG,KEEP),
//       DSN=P1H4.@MSQ0000.SAS.Y00M4001.TEMP3,
//       LABEL=(EXPDT=2003/120),UNIT=(MCART,,DEFER)
//OUT4 DD DISP=(NEW,CATLG,KEEP),
//       DSN=P1H4.@MSQ0000.SAS.Y00M4001.TEMP4,
//       LABEL=(EXPDT=2003/120),UNIT=(MCART,,DEFER)
//OUT5 DD DISP=(NEW,CATLG,KEEP),
//       DSN=P1H4.@MSQ0000.SAS.Y00M4001.DISCHP00,
//       LABEL=(EXPDT=2006/120),UNIT=(MCART,,DEFER)
//SYSIN DD *
*******************************************************************
* PROJ      :MQMS
* REQ BY    :TEAM MQMS
* EXEC DATE :MAR 2003
* PGM       :P1H4.@MSQ00M4.PGMLIBAG(DISCHPxx)
* PGMR      :C PARK
* COMPANY   :ANASYS
* INPUT     :D#MSQ.@MSQ0000.SAS.Y&YYM4001.M4FLAGS
*           :D#MSQ.@MSQ0000.SAS.Y&YYM4001.DISCHYY
* OUTPUTS   :P1H4.@MSQ00M4.SAS.Y&YYM4003.DISCHPYY(NEW)
* PURPOSE   :MERGE PRVID TO DISCHYY FILES AND CREATE PROVIDER
*            LOCATION VARIABLES
*******************************************************************;

%LET YY=00;

DATA PRV; SET
IN1.M4FLAGS(KEEP=Y&YY.HICNO Y&YY.SADM Y&YY.SDIS Y&YY.PRVID);

ATTRIB Y&YY.PST LENGTH=$2.;
ATTRIB Y&YY.PCDIV LENGTH=$1.;
ATTRIB Y&YY.PREG LENGTH=$1.;

 Y&YY.PST = SUBSTR(y&yy.PRVID,1,2); *ACCORDING TO CMS 3/2001 DIRECTIVE;
 IF Y&YY.PST = '55' THEN Y&YY.PST='05';      *CA has 2 codes;
 ELSE IF Y&YY.PST = '65' THEN Y&YY.PST='12'; *SETING GUAM TO HAWAII;
 ELSE IF Y&YY.PST = '67' THEN Y&YY.PST='45'; *TX has 2 codes;
 ELSE IF Y&YY.PST = '68' THEN Y&YY.PST='10'; *FL has 2 codes;
 ELSE IF Y&YY.PST = '80' THEN Y&YY.PST='21'; *MD has 2 codes;

IF Y&YY.PST IN (40,48) OR Y&YY.PST GT 53
     then Y&YY.PREG=0;                   * PR, VI, & OTHER ARE SET TO 0;
else IF Y&YY.PST IN (07,20,22,30,31,33,39,41,47)
     then y&yy.preg=1;
else IF Y&YY.PST IN (14,15,23,36,52,16,17,24,26,28,35,43)
     then y&yy.preg=2;
else IF Y&YY.PST IN (08,09,10,11,21,34,42,49,51,01,18,25,44,04,19,37,45)
     then y&yy.preg=3;
else IF Y&YY.PST IN (03,06,13,27,29,32,46,53,02,05,12,38,50)
     then y&yy.preg=4;
```

```
RUN;

PROC SORT DATA=PRV ; BY Y&YY.HICNO
```
D#MSQ.@MSQ00AG.PGMLIB(DISCHPXX) -- CONTINUED
```
  Y&YY.SADM Y&YY.SDIS;

PROC SORT DATA=IN2.DISCH&YY OUT=OUT3.TEMP3; BY Y&YY.HICNO
  Y&YY.SADM Y&YY.SDIS;

DATA OUT4.TEMP4; MERGE OUT3.TEMP3 (IN=ONE) PRV;
 IF ONE;
 BY Y&YY.HICNO Y&YY.SADM Y&YY.SDIS;
RUN;

PROC SORT DATA=OUT4.TEMP4 OUT=OUT5.DISCHP&YY; BY Y&YY.COHRT Y&YY.HICNO
  Y&YY.SADM;

PROC CONTENTS DATA=OUT5.DISCHP&YY;
RUN;

PROC PRINT DATA=OUT5.DISCHP&YY(OBS=50);
 TITLE3 'DATA = P1H4.@MSQ0000.SAS.Y&YY.M4001.DISCHP&YY';
RUN;
```

D#MSQ.@MSQ00AG.PGMLIB(Y99M4AG1)

```
//P1H4MSQ1 JOB (MSQ00,PARK),P1H4CP,MSGCLASS=Q,CLASS=J
//*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4HFAG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS1AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS2AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS3AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS4AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN1AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN2AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN3AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN4AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN5AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN6AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN7AG
  DELETE P1H4.@MSQ0000.SAS.Y99M4AG1.M4MIAG
//*
//Y99M4AG1 EXEC SAS8,REGION=100M,PARM=('ERRORS=0 NEWS',
//        'SYMBOLGEN MPRINT MLOGIC')
//WORK DD  SPACE=(CYL,(2500,150))
//IN  DD DSN=P1H4.@MSQ0000.SAS.Y99M4001.DISCHP99,DISP=SHR
//OUT1 DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4HFAG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT2  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS1AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT3  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS2AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT4  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS3AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT5  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4IS4AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT6  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN1AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT7  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN2AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT8  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN3AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT9  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN4AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
//OUT10  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN5AG,
//          DISP=(NEW,CATLG,DELETE),
//          SPACE=(CYL,(25,25),RLSE)
//OUT11  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN6AG,
//          DISP=(NEW,CATLG,DELETE),
//          SPACE=(CYL,(25,25),RLSE)
```

```
//OUT12  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN7AG,
//         DISP=(NEW,CATLG,DELETE),
//         SPACE=(CYL,(25,25),RLSE)
```
D#MSQ.@MSQ00AG.PGMLIB(Y99M4AG1) -- CONTINUED

```
//OUT13  DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4MIAG,
//          DISP=(NEW,CATLG,DELETE),
//          SPACE=(CYL,(25,25),RLSE)
//SYSIN DD *,DLM=##

*********************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ00M4.PGMLIBAG(Y99M4AG1)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : P1H4.@MSQ0000.SAS.Y99M4001.DISCHP99
* OUTPUTS: P1H4.@MSQ0000.SAS.Y99M4AG1.M4ccAG
*            WHERE cc = HF, IS1-IS4, PN1-PN7, MI
* PURPOSE: PRODUCE aggregate FILES THAT CONTAIN COUNTS OF beneficiaries,
*          DISCHARGES, TOTAL MEDICARE PAYMENTS (TPAY), total deductibles
*          AND COPAY (BPAY), NUMBER OF DAYS of stay,and other M4
*          M4 analytic variables for each MQMS cohort summarized to the
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary and provider location variables.  The letters
*          "cc" reference the clinical cohort, e.g., when "cc"is HF,
*          this references the heart failure cohort, when "cc" isMI,
*          this references the acute myocardial infarction cohort, etc..
* NOTE   : CHANGE THE YY MACRO VAR BELOW FOR SPECIFIC YEAR
*          OF ANALYSIS
*********************************************************************;

 ***************
 * ASSIGN YEAR *
 ***************;

%LET YY=99;

 *************** *
 * SUBGROUP VARS *
 *****************;

%LET VARS=Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG
          Y&YY.PST Y&YY.PREG;

 ****************************************
 * MACRO TO AGGREGATE ACROSS ALL SUBGROUPS *
 ****************************************;

%MACRO AG1_4(WHERE,X,OUTDATA);
  PROC SUMMARY DATA=IN.DISCHP&YY
   (WHERE=(&WHERE))
    NWAY ;
    VAR DISCH&YY Y&YY.LOS Y&YY.TPAY Y&YY.BPAY
        BENE&YY CCDIS&YY TOTDIS&YY;
    CLASS &VARS;
    OUTPUT OUT=OUT&X..&OUTDATA(DROP=_TYPE_ _FREQ_) SUM=;
  RUN;
```

```
 PROC CONTENTS DATA=OUT&X..&OUTDATA;
  TITLE1 "MQMS - PROGRAM=Y&YY.M4AG1";
  TITLE2 "DATA = &YY &OUTDATA";
 RUN;
```

D#MSQ.@MSQ00AG.PGMLIB(Y99M4AG1) -- CONTINUED

```
 PROC PRINT DATA=OUT&X..&OUTDATA(OBS=50);
 RUN;
%MEND AG1_4;

 ************************************
 * INVOKE MACRO=AG1_4 FOR EACH COHORT *
 ************************************;

%AG1_4(Y&YY.CCGRP EQ 1,1,M4HFAG);

%AG1_4(Y&YY.CCGRP EQ 2,2,M4IS1AG);

%AG1_4(Y&YY.CCGRP EQ 3,3,M4IS2AG);

%AG1_4(Y&YY.CCGRP EQ 4,4,M4IS3AG);

%AG1_4(Y&YY.CCGRP EQ 5,5,M4IS4AG);

%AG1_4(Y&YY.CCGRP EQ 6,6,M4PN1AG);

%AG1_4(Y&YY.CCGRP EQ 7,7,M4PN2AG);

%AG1_4(Y&YY.CCGRP EQ 8,8,M4PN3AG);

%AG1_4(Y&YY.CCGRP EQ 9,9,M4PN4AG);

%AG1_4(Y&YY.CCGRP EQ 10,10,M4PN5AG);

%AG1_4(Y&YY.CCGRP EQ 11,11,M4PN6AG);

%AG1_4(Y&YY.CCGRP EQ 12,12,M4PN7AG);

%AG1_4(Y&YY.CCGRP EQ 13,13,M4MIAG);
```

P1H4.@MSQ00M4.PGMLIB(ADJWTS99)

```
//P1H4MSQ JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(2500,500))
//IN  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1006.M1CNTS99,DISP=SHR
//OUT DD DSN=P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99,
//       DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *
************************************************************
* PURPOSE: Computes age, sex, & age/sex adjustment weights from
*          1999 Medicare FFS beneficiary population files (standard
*          population for M1-3 and M1-4 denominator cohorts.
*          Two sets of weights are computed: one based on 9 age
*          and another one based on 18 age groups.
* INPUT  : D#MSQ.@MSQ0000.SAS.Y99M1006.m1cnts99
* OUTPUT : P1H4.@MSQ0000.SAS.Y99M1.adjwts99
************************************************************;

proc sql;
title 'Adjustment factors for M1-3 and M1-4';
  Create table adjwt1  as
    select * ,
    sum(y99ctm13) as m13total,
    sum(y99ctm14) as m14total
  from in.m1ascn99 where y99vars='TOTAL';

proc sql;
  Create table adjwt2 as
  select y99fage, y99sex, y99ctm13, y99ctm14, m13total, m14total,
    sum(y99ctm13) as m13agetot,
    sum(y99ctm14) as m14agetot,
    (calculated m13agetot/m13total) as m13aadj
     LABEL='AGE (9) ADJUSTMENT WT FOR M1-3',
    (calculated m14agetot/m14total) as m14aadj
     LABEL='AGE (9) ADJUSTMENT WT FOR M1-4'
  from adjwt1
  group by y99fage;

proc sql;
  Create table adjwt3 as select *,
    sum(y99ctm13) as m13sextot,
    sum(y99ctm14) as m14sextot,
    (calculated m13sextot/m13total) as m13sadj
     label='Sex Adjustment Wt for M1-3',
    (calculated m14sextot/m14total) as m14sadj
     label='Sex Adjustment Wt for M1-4'
  from adjwt2
  group by y99sex;

proc sql;
  CREATE TABLE ADJWT99 AS SELECT *,
    sum(y99ctm13) as m13subtot,
    sum(y99ctm14) as m14subtot,
    (calculated m13subtot/m13total) as m13asadj
     LABEL='AGE/SEX (18) ADJUSTMENT WT FOR M1-3',
    (calculated m14subtot/m14total) as m14asadj
     LABEL='AGE/SEX (18) ADJUSTMENT WT FOR M1-4'
```

```
  from adjwt3
  group by y99sex, y99fage;
```

P1H4.@MSQ00M4.PGMLIB(ADJWTS99) -- CONTINUED

```
DATA ADJWT99; SET ADJWT99;
* REGROUP AGE TO 4 CATEGORIES: <64, 65-74, 75-84, 85+;
  IF Y99FAGE IN ('1','2') THEN Y99FAGER='1';
  ELSE IF Y99FAGE IN ('3','4') THEN Y99FAGER='2';
  ELSE IF Y99FAGE IN ('5','6') THEN Y99FAGER='3';
  ELSE IF Y99FAGE IN ('7','8','9') THEN Y99FAGER='4';

proc sql;
  CREATE TABLE ADJWTS99 AS SELECT *,
    SUM(M13AADJ) AS M13AADJR
     LABEL='AGE(4) ADJUSTMENT WT FOR M1-3',
    SUM(M14AADJ) AS M14AADJR
     LABEL='AGE(4) ADJUSTMENT WT FOR M1-4'
  FROM ADJWT99
  GROUP BY Y99FAGER;

proc sql;
  CREATE TABLE OUT.ADJWTS99 AS SELECT *,
    SUM(M13ASADJ) AS M13ASADJR
     LABEL='AGE/SEX (8) ADJUSTMENT WT FOR M1-3',
    SUM(M14ASADJ) AS M14ASADJR
     LABEL='AGE/SEX (8) ADJUSTMENT WT FOR M1-4'
  FROM ADJWTS99
  GROUP BY y99sex, Y99FAGER;
QUIT;

proc sort data=out.adjwts99;
by y99fage y99sex;
run;

OPTIONS NODATE PAGENO=1 LINESIZE=130 PAGESIZE=60;

PROC PRINT DATA=OUT.ADJWTS99;
TITLE 'ADJUSTMENT WEIGHTS FOR M1-3 AND M1-4';
RUN;
```

P1H4.@MSQ00AG.PGMLIB(ADJM4CC)

```
//P1H4PN1 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//*****************************************************************
//* THIS JOB EXECUTES THE CATALOGED PROCEDURE ADJM4 TO COMPUTE AGE/SEX
//* ADJUSTED M4 MEASURES FOR A PARTICULAR CLINICAL CONDITION (CC).
//* NOTE: CC NEEDS TO BE CHANGED.
//*****************************************************************
//  JCLLIB ORDER='P1H4.@MSQ00AG.PGMLIB' RORS=0,NEWS'
//ADJM4CC EXEC ADJM4,CC=PN1
```

P1H4.@MSQ00AG.PGMLIB(ADJM4)

```
//ADJM4 PROC
//*************************************************************
//* COMPUTE AGE/SEX ADJUSTED DISCHARGE RATES
//*************************************************************
//ADRATE EXEC SAS8,REGION=8000K,PARM=('ERRORS=0,NEWS',
//      'SYSPARM=&CC'),COND=(4,LT)
//WORK  DD SPACE=(CYL,(2500,500))
//IN0   DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//NUM92 DD DSN=P1H4.@MSQ0000.SAS.Y92M4AG1.M4&CC.AG,DISP=SHR
//NUM93 DD DSN=P1H4.@MSQ0000.SAS.Y93M4AG1.M4&CC.AG,DISP=SHR
//NUM94 DD DSN=P1H4.@MSQ0000.SAS.Y94M4AG1.M4&CC.AG,DISP=SHR
//NUM95 DD DSN=P1H4.@MSQ0000.SAS.Y95M4AG1.M4&CC.AG,DISP=SHR
//NUM96 DD DSN=P1H4.@MSQ0000.SAS.Y96M4AG1.M4&CC.AG,DISP=SHR
//NUM97 DD DSN=P1H4.@MSQ0000.SAS.Y97M4AG1.M4&CC.AG,DISP=SHR
//NUM98 DD DSN=P1H4.@MSQ0000.SAS.Y98M4AG1.M4&CC.AG,DISP=SHR
//NUM99 DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4&CC.AG,DISP=SHR
//NUM00 DD DSN=P1H4.@MSQ0000.SAS.Y00M4AG1.M4&CC.AG,DISP=SHR
//NUM01 DD DSN=P1H4.@MSQ0000.SAS.Y01M4AG1.M4&CC.AG,DISP=SHR
//DEN92 DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE,DISP=SHR
//DEN93 DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE,DISP=SHR
//DEN94 DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE,DISP=SHR
//DEN95 DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE,DISP=SHR
//DEN96 DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE,DISP=SHR
//DEN97 DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE,DISP=SHR
//DEN98 DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE,DISP=SHR
//DEN99 DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE,DISP=SHR
//DEN00 DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE,DISP=SHR
//DEN01 DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1AG1.M14BENE,DISP=SHR
//ADJ   DD DSN=P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99,DISP=SHR
//OUT1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.&CC.TAB03,
//      DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=P1H4.@MSQ00AG.PGMLIB(SYSMADJ1),DISP=SHR
//*************************************************************
//* COMPUTE AGE/SEX ADJUSTED AVERAGE LENGTH OF STAY
//*************************************************************
//ADLOS EXEC SAS8,REGION=8000K,PARM=('ERRORS=0,NEWS',
//      'SYSPARM=&CC'),COND=(4,LT)
//WORK  DD SPACE=(CYL,(2500,500))
//IN0   DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//NUM92 DD DSN=P1H4.@MSQ0000.SAS.Y92M4AG1.M4&CC.AG,DISP=SHR
//NUM93 DD DSN=P1H4.@MSQ0000.SAS.Y93M4AG1.M4&CC.AG,DISP=SHR
//NUM94 DD DSN=P1H4.@MSQ0000.SAS.Y94M4AG1.M4&CC.AG,DISP=SHR
//NUM95 DD DSN=P1H4.@MSQ0000.SAS.Y95M4AG1.M4&CC.AG,DISP=SHR
//NUM96 DD DSN=P1H4.@MSQ0000.SAS.Y96M4AG1.M4&CC.AG,DISP=SHR
//NUM97 DD DSN=P1H4.@MSQ0000.SAS.Y97M4AG1.M4&CC.AG,DISP=SHR
//NUM98 DD DSN=P1H4.@MSQ0000.SAS.Y98M4AG1.M4&CC.AG,DISP=SHR
//NUM99 DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4&CC.AG,DISP=SHR
//NUM00 DD DSN=P1H4.@MSQ0000.SAS.Y00M4AG1.M4&CC.AG,DISP=SHR
//NUM01 DD DSN=P1H4.@MSQ0000.SAS.Y01M4AG1.M4&CC.AG,DISP=SHR
//ADJ   DD DSN=P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99,DISP=SHR
//OUT2  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.&CC.TAB04,
//      DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=P1H4.@MSQ00AG.PGMLIB(SYSMADJ2),DISP=SHR
```

P1H4.@MSQ00AG.PGMLIB(UNADJR)

```
//P1H4PN5 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//*****************************************************************
//* PRODUCE UNADJUSTED M4 MEASURES: TOTAL DISCHARGE & BENE COUNTS,
//* TPAY, BPAY, AND AVERAGE TPAY, # CC DISCHARGES, AND ALL DISCHARGES
//*****************************************************************
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(2500,500))
//NUM92 DD DSN=P1H4.@MSQ0000.SAS.Y92M4AG1.M4PN5AG,DISP=SHR
//NUM93 DD DSN=P1H4.@MSQ0000.SAS.Y93M4AG1.M4PN5AG,DISP=SHR
//NUM94 DD DSN=P1H4.@MSQ0000.SAS.Y94M4AG1.M4PN5AG,DISP=SHR
//NUM95 DD DSN=P1H4.@MSQ0000.SAS.Y95M4AG1.M4PN5AG,DISP=SHR
//NUM96 DD DSN=P1H4.@MSQ0000.SAS.Y96M4AG1.M4PN5AG,DISP=SHR
//NUM97 DD DSN=P1H4.@MSQ0000.SAS.Y97M4AG1.M4PN5AG,DISP=SHR
//NUM98 DD DSN=P1H4.@MSQ0000.SAS.Y98M4AG1.M4PN5AG,DISP=SHR
//NUM99 DD DSN=P1H4.@MSQ0000.SAS.Y99M4AG1.M4PN5AG,DISP=SHR
//NUM00 DD DSN=P1H4.@MSQ0000.SAS.Y00M4AG1.M4PN5AG,DISP=SHR
//NUM01 DD DSN=P1H4.@MSQ0000.SAS.Y01M4AG1.M4PN5AG,DISP=SHR
//OUT1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB01,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//OUT2  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB02,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *
************************************************************
* INPUT  : P1H4.@MSQ0000.SAS.YXXM4AG1.M4PN5AG
* OUTPUT : P1H4.@MSQ0000.SAS.YxxyyM4.PN5TABnn
************************************************************;

%MACRO unadj(YY);

*****************************************************************
* STEP 1: Produce sums and averages of M4 measures in total and by
*         stratifiers, i.e. age, sex, race, dual, Medicare status,
*         urban/rural, beneficiary census region, and state.
*****************************************************************;

DATA M4PN5AG; SET NUM&yy..M4PN5AG;
if y&yy.creg ='0' then y&yy.state='0';  *setting PR & VI to '0';
if y&yy.preg ='0' then y&yy.pst ='0';   *setting PR,VI,& Canada to '0';
if y&yy.mdst ='1' then y&yy.mdstr='1';   *recode MDST into 3 categories;
else if y&yy.mdst ='3' then y&yy.mdstr='2';
else if y&yy.mdst in ('2','4','5') then y&yy.mdstr='3';

PROC SUMMARY DATA=M4PN5AG ;
CLASS Y&yy.FAGE Y&yy.SEX
      y&yy.race y&yy.dual y&yy.mdstr y&yy.urbn y&yy.creg y&yy.state;
VAR DISCH&yy bene&yy y&yy.bpay;
OUTPUT OUT=cc1 (WHERE=(_TYPE_ IN (0,1,2,4,8,16,32,64,128)))
               SUM=DISCH&yy bene&yy bpay&yy;

DATA ccbst&yy; SET cc1;
length total $1;
if _type_ in (0) then total='1';
if disch&yy le 25 then do;
   disch&yy = .;
   bpay&yy  = .;
end;
```

```
if bene&yy le 25 then bene&yy = .;
rename y&yy.sex=sex
       y&yy.fage=fage
```
P1H4.@MSQ00AG.PGMLIB(UNADJR) -- CONTINUED

```
       y&yy.race=race
       y&yy.dual=dual
       y&yy.mdstr=mdst
       y&yy.urbn=urbn
       y&yy.creg=creg
       y&yy.state=state;
drop _type_ _freq_ ;

proc sort data=ccbst&yy;
by total fage sex race dual mdst urbn creg state;
run;

*****************************************************************
* STEP 2: Produce sums and averages of M4 measures in total and by
*         stratifiers, i.e. age, sex, race, dual, Medicare status,
*         urban/rural, provider census region, and state.
*****************************************************************;

PROC SUMMARY DATA=M4PN5AG;
class y&yy.fage y&yy.sex
      y&yy.race y&yy.dual y&yy.mdstr y&yy.urbn y&yy.preg y&yy.pst;
VAR DISCH&yy bene&yy y&yy.tpay ccdis&yy totdis&yy;
OUTPUT OUT=cc2 (WHERE=(_TYPE_ IN (0,1,2,4,8,16,32,64,128)))
               SUM=DISCH&yy bene&yy tpay&yy ccdis&yy totdis&yy;

DATA ccpst&yy; SET cc2;
length total $1;
if _type_ in (0) then total='1';
avgtpay&yy = tpay&yy/disch&yy;
avgcc&yy = ccdis&yy /bene&yy;
avgdis&yy = totdis&yy/bene&yy;
if disch&yy le 25 then do;
   tpay&yy   = .;
   avgtpay&yy = .;
end;
if bene&yy le 25 then do;
   avgcc&yy = .;
   avgdis&yy = .;
end;
rename y&yy.sex=sex
       y&yy.fage=fage
       y&yy.race=race
       y&yy.dual=dual
       y&yy.mdstr=mdst
       y&yy.urbn=urbn
       y&yy.preg=preg
       y&yy.pst =pst;
drop _type_ _freq_ ccdis&yy totdis&yy;

proc sort data=ccpst&yy;
by total fage sex race dual mdst urbn preg pst;
run;
```

```
%MEND unadj;

%unadj(92);
%unadj(93);
%unadj(94);
%unadj(95);
```
P1H4.@MSQ00AG.PGMLIB(UNADJR) -- CONTINUED

```
%unadj(96);
%unadj(97);
%unadj(98);
%unadj(99);
%unadj(00);
%unadj(01);

*********************************************************************;
* After the above programs are run for each year, the 10 yearly files
* are combined to produce a file containing the 10 years' rates
*********************************************************************;


options ls=160 ps=200;

data out1.PN5tab01;
merge ccbst92  ccbst93  ccbst94  ccbst95  ccbst96
      ccbst97  ccbst98  ccbst99  ccbst00  ccbst01;
by total fage sex race dual mdst urbn creg state;

data out2.PN5tab02;
merge ccpst92  ccpst93  ccpst94  ccpst95  ccpst96
      ccpst97  ccpst98  ccpst99  ccpst00  ccpst01;
by total fage sex race dual mdst urbn preg pst;

PROC PRINT data=out1.PN5tab01;
title 'Unadjusted M4 measures by demographic vars & bene state';

PROC PRINT data=out2.PN5tab02;
title 'Unadjusted M4 measures by demographic vars & provider state';
run;
```

P1H4.@MSQ00AG.PGMLIB(ADRATE)

```
%MACRO ADRATE(CC);

*****************************************************************
* PURPOSE : Computes age/sex adjusted rates using the aggregate data
*           files. The 1999 Medicare FFS beneficiary population (M1-3
*           cohort) serves as the standard population. Discharge
*           rates by demographic subgroups and census region are
*           adjusted using direct method, and rates by state are
*           adjusted using indirect method.
* INPUT   : Numerator file - P1H4.@MSQ0000.SAS.YxxM4AG1.M4CCAG
*         : Denomina. file - P1H4.@MSQ0000.SAS.YxxM1AG1.M14BENE
*         : Std. Pop. file - P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99
* OUTPUT : P1H4.@MSQ0000.SAS.Y9201M4.CCTAB03
*****************************************************************;

  /** Compute 1999 standard rates for national total and by age*sex
      groupings to be used in indirect adjustment **/

* Merge numerator and denominator files;
proc sql;
  Create table M4&CC.AG as
  SELECT Y99FAGE, Y99SEX,
    sum(bene99) as benestd,
    sum(disch99) as dischstd
  from num99.M4&CC.AG
  GROUP BY Y99FAGE, Y99SEX;
QUIT;

proc sql;
  Create table M4&CC.den as
  SELECT Y99FAGE, Y99SEX,
    sum(y99count) as denomstd
  from den99.m14bene
  GROUP BY Y99FAGE, Y99SEX;
QUIT;

* Compute 1999 standard rates by age*sex grouping and for total;
data M4&CC.com;
merge M4&CC.AG M4&CC.den;
by y99fage y99sex ;
stddisrate = dischstd/denomstd;
stdbenrate = benestd/denomstd;
run;

proc sql;
  Create table M4&CC.std as
  SELECT *,
   (sum(dischstd)/sum(denomstd))*1000 as nstddisrate
    label='Standard national discharge rate',
   (sum(benestd)/sum(denomstd))*1000 as nstdbenrate
    label='Standard national beneficiary discharge rate'
  from M4&CC.com;
QUIT;

%MACRO ADJ(YY);
```

```
    **BEGIN DIRECT ADJUSTMENT PROCEDURE;

    ********************************************************************
    P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED

    * STEP 1: Merge the numerator and denominator files by subgroup
    *         variables, and also merge with the 1999 standard population
    *         file containing age/sex adjustment weights.
    ********************************************************************;

    * Get numerator variables from the aggregate numerator file by
      stratifiers including beneficiary region;

    proc sql;
      Create table M4&CC.AG as
      SELECT Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
             Y&YY.URBN, Y&YY.CREG,
        sum(disch&YY) as disch,
        sum(bene&YY) as bene
      from num&yy..M4&CC.AG
      GROUP   BY   Y&YY.FAGE,   Y&YY.SEX,   Y&YY.RACE,   Y&YY.DUAL,   Y&YY.MDST,
Y&YY.URBN,
               Y&YY.CREG;
    QUIT;

    * Merge the above file with the denominator counts by stratifiers and
      then with the file containing adjustment weights;

    proc sql;
      Create table m14bene as
      SELECT Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
             Y&YY.URBN, Y&YY.CREG,
        sum(Y&YY.count) as denomcnt
      from den&yy..m14bene
      GROUP   BY   Y&YY.FAGE,   Y&YY.SEX,   Y&YY.RACE,   Y&YY.DUAL,   Y&YY.MDST,
Y&YY.URBN,
               Y&YY.CREG;
    QUIT;

    DATA &CC;
    MERGE M4&CC.AG m14BENE (IN=ONE);
    BY Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL Y&YY.MDST Y&YY.URBN Y&YY.CREG;
    if one;
    if Y&YY.mdst ='1' then Y&YY.mdstr='1';   *recode MDST into 3 categories;
    else if Y&YY.mdst ='3' then Y&YY.mdstr='2';
    else if Y&YY.mdst in ('2','4','5') then Y&YY.mdstr='3';

    DATA &CC.1;
    merge &CC (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.race=race
                       y&yy.mdstr=mdst y&yy.dual=dual y&yy.urbn=urbn
                       y&yy.creg=creg))
          ADJ.adjwts99 (keep= Y99fage Y99sex m13asadj m13aadj m13sadj
                        rename=(y99fage=fage y99sex=sex));
    by fage sex;

    ********************************************************************
    * STEP 2:  Sum up numerator and denominator counts by age*sex*subgroup
    *          categories.
```

```
*********************************************************************;

proc summary data=&CC.1;
class fage sex
      RACE DUAL MDST URBN CREG;
var disch bene denomcnt;
```
P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED

```
id m13aadj m13sadj m13asadj ;
OUTPUT OUT=ADJRATE (WHERE=(_TYPE_ IN (96,97,98,100,104,112)))
SUM=disch bene denomcnt;

*****************************************************************
* STEP 3:  Divide the numerator total by denominator total by
*          age*sex*subgroup categories and multiply by appropriate
*          adjustment weights, i.e.,
*            For adjustment rate by age, multiply by sex-adjusted wts
*            For adjustment rate by sex, multiply by age-adjusted wts
*            For adjustment rate by other subgroup variables,
*                multiply by age/sex-adjusted wts
*****************************************************************;

DATA ADJRATE2; set adjrate;
if _type_ in (96) then do; total='1';
  wtsexdrate = disch/denomcnt * m13aadj;
  wtagedrate = disch/denomcnt * m13sadj;
  wteddrate = disch/denomcnt * m13asadj;
  wtsexbrate = bene/denomcnt * m13aadj;
  wtagebrate = bene/denomcnt * m13sadj;
  wtedbrate = bene/denomcnt * m13asadj;
end;
else if _type_ in (97,98,104,112) then do;
  fage=.;  sex=.;
  wteddrate = disch/denomcnt * m13asadj;
  wtedbrate = bene /denomcnt * m13asadj;
end;
* Fixing the rates by MDST with age-based correction factors;
else if _type_ in (100) then do;
  fage=.;  sex=.;
  if mdst='1' then do;
    wteddrate = disch/denomcnt * m13asadj/0.84593;
    wtedbrate = bene /denomcnt * m13asadj/0.84593;
  end;
  else if mdst='2' then do;
    wteddrate = disch/denomcnt * m13asadj/0.15407;
    wtedbrate = bene /denomcnt * m13asadj/0.15407;
  end;
  else if mdst='3' then do;
    wteddrate = disch/denomcnt * m13asadj;
    wtedbrate = bene /denomcnt * m13asadj;
  end;
end;

*****************************************************************
* STEP 4:  Sum the multiplied products up to each categorical level
*          of a subgroup variable - This becomes the adjusted rate
*          for individual subgroup categories
*****************************************************************;
```

```
 OPTIONS NODATE PAGENO=1 LINESIZE=80 PAGESIZE=80;

PROC TABULATE DATA=Adjrate2  OUT=adjrate3
  missing noseps FORMCHAR='           ';
class total fage sex
      RACE DUAL MDST URBN CREG;
VAR disch bene denomcnt wtagedrate wtsexdrate wteddrate
                        wtagebrate wtsexbrate wtedbrate;
```
P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED

```
TABLE total,
 ((disch='' bene='' denomcnt='')*SUM=''*F=12.
  (wteddrate='' wtedbrate='')*SUM=''*F=6.2);
TABLE fage,
 ((disch='' bene='' denomcnt='')*SUM=''*F=12.
  (wtagedrate='' wtagebrate='')*SUM=''*F=6.2);
TABLE sex,
 ((disch='' bene='' denomcnt='')*SUM=''*F=12.
  (wtsexdrate='' wtsexbrate='')*SUM=''*F=6.2);
TABLE (RACE dual mdst URBN creg),
 ((disch='' bene='' denomcnt='')*SUM=''*F=12.
  (wteddrate='' wtedbrate='')*SUM=''*F=6.2)
/RTS=28 ROW=FLOAT;
KEYLABEL ALL='All enrollees'
         sum='';
         LABEL   TOTAL='All enrollees'
                  FAGE='Age Group'
                   SEX='Sex'
                   URBN='Urban/Rural'
                   RACE='Race'
                   MDST='Medicare Eligibility Status'
                   DUAL='Other Disadvantaged Status'
                   creg='Beneficiary region';
RUN;
*****************************************************************
* STEP 5:  Sum of weighted values from step 2.3 multiplied by
*          1,000 is the adjusted rate per 1,000 denominator counts
*          for individual subgroup categories
*****************************************************************;

data adjrate&yy;  set adjrate3 (rename=(disch_sum=dischcnt&yy
                bene_sum=benecnt&yy  denomcnt_sum=denomcnt&yy));

* Compute unadjusted rates;
disrate&yy = dischcnt&yy / denomcnt&yy *1000;
benrate&yy = benecnt&yy / denomcnt&yy *1000;

* Compute adjusted rates;
if total='1' then do;
     adjdrate&YY=wteddrate_sum*1000;
     adjbrate&YY=wtedbrate_sum*1000; end;
else if sex ge 0 then do;
     adjdrate&YY=wtsexdrate_sum*1000;
     adjbrate&YY=wtsexbrate_sum*1000; end;
else if fage ge 0 then do;
     adjdrate&YY=wtagedrate_sum*1000;
     adjbrate&YY=wtagebrate_sum*1000; end;
```

```
else if race ge 0 or
        mdst ge 0 or
        dual ge 0 or
        urbn ge 0 or
        creg ge 0 then do;
     adjdrate&YY=wteddrate_sum*1000;
     adjbrate&YY=wtedbrate_sum*1000; end;
else delete;

* Set the adjusted rates to . if the numerator count is 25 or less ;

if dischcnt&yy le 25 then adjdrate&YY=.;
```
P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED
```
if benecnt&yy le 25 then adjbrate&YY=.;
drop _type_ _page_ _table_ wtsexdrate_sum wtagedrate_sum wteddrate_sum
                           wtsexbrate_sum wtagebrate_sum wtedbrate_sum;

run;

   /**END OF DIRECT ADJUSTMENT PROCEDURE**/

   /**BEGIN INDIRECT ADJUSTMENT PROCEDURE**/
*********************************************************************
* For each year, obtain observed(O) and expected(E) number of numerator
* events (i.e., CC discharges and benes with CC discharges) by state
* and multiply O/E to the 1999 national unadjusted rate to compute
* the indirectly adjusted rates for individual states
*********************************************************************;

*********************************************************************
* Step 6: Merge numerator and denominator files for each year to have
*         numertor and denominator counts by age/sex/state cells.
*********************************************************************;

proc sql;
  Create table M4&CC.AG as
  SELECT Y&YY.FAGE, Y&YY.SEX, Y&YY.state,
    sum(bene&YY) as benecnt,
    sum(disch&YY) as dischcnt
  from num&YY..M4&CC.AG
  GROUP BY Y&YY.FAGE, Y&YY.SEX, Y&YY.state;
QUIT;

proc sql;
  Create table M4&CC.den as
  SELECT Y&YY.FAGE, Y&YY.SEX, Y&YY.state,
    sum(Y&YY.count) as denomcnt
  from den&YY..m14bene
  GROUP BY Y&YY.FAGE, Y&YY.SEX, Y&YY.state;
QUIT;

data combine;
merge M4&CC.AG M4&CC.den;
by Y&YY.fage Y&YY.sex Y&YY.state;
IF Y&YY.state in ('40','48') THEN Y&YY.state='0'; *SETTING PR & VI to 0;

*********************************************************************
```

```
* Step 7: Compute expected outcome events by multiplying the standard
*         state-specific rates with the denominator counts for each
*         age*sex*state cell.
*********************************************************************;

data combine2;
merge combine (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.state=state))
      M4&CC.std (rename=(y99fage=fage y99sex=sex));
by fage sex;
expdis = stddisrate * denomcnt;
expben = stdbenrate * denomcnt;
run;

*********************************************************************
* Step 8: Compute observed and expected sum of numerator events and
P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED

*         denominator counts for each state.
*********************************************************************;

proc summary data=combine2;
class state;
var expdis expben denomcnt dischcnt benecnt;
ID NSTDDISRATE NSTDBENRATE;
OUTput OUT=iadjrate
       sum=expdis&yy expben&yy denomcnt&yy dischcnt&yy benecnt&yy;

*********************************************************************
* Step 9: Compute unadjusted and indirectly adjusted rates for each
*         state.
*********************************************************************;

data irate&yy; set iadjrate;
srrdis&yy = dischcnt&yy / expdis&yy;
srrben&yy = benecnt&yy / expben&yy;
disrate&yy = dischcnt&yy / denomcnt&yy *1000;   *Unadjusted rate;
benrate&yy = benecnt&yy / denomcnt&yy *1000;
ADJDrate&yy = srrdis&yy * nstddisrate;         *Adjusted rate;
ADJBrate&yy = srrben&yy * nstdbenrate;

if _type_ in (0) then total='2';               *Setting U.S. total;
if dischcnt&yy le 25 then ADJDrate&yy=.;       *Setting rates based on;
if benecnt&yy le 25 then ADJBrate&yy=.;        *numerator events of 25;
                                                *or less to missing;
drop _type_ _freq_ expdis&yy expben&yy;

run;

   /**END OF DIRECT ADJUSTMENT PROCEDURE**/

*********************************************************************
* Step 10: Combine the two (directly adjusted & indirectly adjusted
*          rates) files
*********************************************************************;

DATA ADRATE&YY;  SET ADJRATE&YY IRATE&YY;
proc sort ;
by total fage sex race dual mdst urbn creg state;
```

```
run;

%MEND ADJ;

*********************************************************************
* Step 11:  Run the adjustment macro for each year, and then combine
*           the yearly files into one file containing 10 year data.
*********************************************************************;

%ADJ(92);
%ADJ(93);
%ADJ(94);
%ADJ(95);
%ADJ(96);
%ADJ(97);
%ADJ(98);
%ADJ(99);
%ADJ(00);
```
P1H4.@MSQ00AG.PGMLIB(ADRATE) -- CONTINUED

```
%ADJ(01);


DATA OUT1.&CC.TAB03;
MERGE ADRATE92 ADRATE93 ADRATE94 ADRATE95 ADRATE96
      ADRATE97 ADRATE98 ADRATE99 ADRATE00 ADRATE01;
by total fage sex race dual mdst urbn creg state;

PROC PRINT DATA=OUT1.&CC.TAB03;
TITLE 'ADJUSTED DISCHARGE RATES for &CC,1992-2001';
RUN;

%MEND ADRATE;
```

P1H4.@MSQ00AG.PGMLIB(ADLOS)

```
%MACRO ADLOS(CC);

*****************************************************************
* PURPOSE : Computes age/sex adjusted length of stays (LOSs) using
*           the aggregate data files. LOSs by demographic subgroups
*           and provider region are adjusted using direct method and
*           LOSs by provider state are adjusted using indirect method.
*           For direct adjustment, the 1999 Medicare FFS
*           beneficiary population (M1-3 cohort) serves as the
*           standard population.
* INPUT   : Numerator file - P1H4.@MSQ0000.SAS.YXXM4AG1.M4CCAG
*         : Std. Pop. file - P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99
* OUTPUT  : P1H4.@MSQ0000.SAS.Y9201M4.CCTAB04
*****************************************************************;

  /** Compute 1999 standard LOSs for national total and by age*sex
      groupings to be used in indirect adjustment **/

proc sql;
  Create table M4&CC.AG as
  SELECT y99fage, y99SEX,
    sum(y99los) as losas,
    sum(disch99) as dischas,
    (calculated losas/calculated dischas) as stdavglos
      label='Standard age/sex average LOS'
  from num99.M4&CC.AG
  GROUP BY y99fage, y99SEX;
QUIT;

proc sql;
  Create table M4&CC.STD as
  SELECT *,
    (sum(losas)/sum(dischas)) as nstdavglos
      label='Standard national average LOS'
  from M4&CC.AG;
QUIT;


%MACRO ADJ(YY);

  /**BEGIN DIRECT ADJUSTMENT PROCEDURE**/

*****************************************************************
* STEP 1: Merge the AGGREGATE file with the 1999 standard population
*         file containing age/sex adjustment weights.
*****************************************************************;

* Get LOS and DISCHARGE variables from the aggregate file by
  stratifiers including PROVIDER region;

proc sql;
  Create table &CC as
  SELECT Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
         Y&YY.URBN, Y&YY.PREG,
    sum(disch&YY) as disch,
    sum(y&yy.los) as LOS
```

```
  from num&yy..M4&CC.AG
  GROUP BY Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
           Y&YY.URBN, Y&YY.PREG;
```
P1H4.@MSQ00AG.PGMLIB(ADLOS) -- CONTINUED

```
QUIT;

DATA &CC.1;
merge &CC (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.race=race
                  y&yy.dual=dual y&yy.urbn=urbn y&yy.PREG=PREG))
      ADJ.adjwts99 (keep= Y99fage Y99sex m13asadj m13aadj m13sadj
                    rename=(y99fage=fage y99sex=sex));
by fage sex;

*Recode MDST into 3 categories;
if Y&YY.mdst ='1' then mdst='1';
else if Y&YY.mdst ='3' then mdst='2';
else if Y&YY.mdst in ('2','4','5') then mdst='3';
DROP Y&YY.mdst;

*********************************************************************
* STEP 2:  Sum up numerator and denominator counts by age*sex*subgroup
*          categories;
*********************************************************************;

proc summary data=&CC.1;
class fage sex
      RACE DUAL MDST URBN PREG;
var disch LOS;
id m13aadj m13sadj m13asadj ;
OUTPUT OUT=ADJLOS (WHERE=(_TYPE_ IN (96,97,98,100,104,112)))
SUM=disch LOS;

*****************************************************************
* STEP 3:  Divide the numerator total by denominator total by
*          age*sex*subgroup categories and multiply by appropriate
*          adjustment weights, i.e.,
*            For adjustment LOS by age, multiply by sex-adjusted wts
*            For adjustment LOS by sex, multiply by age-adjusted wts
*            For adjustment LOS by other subgroup variables,
*                multiply by age/sex-adjusted wts
*****************************************************************;

DATA ADJLOS2; set adjLOS;
if _type_ in (96) then do; total='1';
  wtsexLOS = LOS / DISCH * m13aadj;
  wtageLOS = LOS / DISCH * m13sadj;
  wtedLOS =  LOS / DISCH * m13asadj;
end;
else if _type_ in (97,98,104,112) then do;
  fage=.;  sex=.;
  wtedLOS =  LOS / DISCH * m13asadj;
end;

* Fix the LOSs by MDST with age-based correction factors;
else if _type_ in (100) then do;
  fage=.;  sex=.;
  if mdst='1' then
```

```
    wtedLOS = LOS / DISCH * m13asadj/0.84593;
  else if mdst='2' then
    wtedLOS = LOS / DISCH * m13asadj/0.15407;
  else if mdst='3' then
    wtedLOS = LOS / DISCH * m13asadj;
end;
```
P1H4.@MSQ00AG.PGMLIB(ADLOS) -- CONTINUED

```
*****************************************************************
* STEP 4:  Sum the multiplied products up to each categorical level
*          of a subgroup variable - This becomes the adjusted LOS
*          for individual subgroup categories
*****************************************************************;

 OPTIONS NODATE PAGENO=1 LINESIZE=80 PAGESIZE=80;

PROC TABULATE DATA=AdjLOS2   out=adjLOS3
  missing noseps FORMCHAR='           ';
class total fage sex
      RACE DUAL MDST URBN PREG;
VAR disch LOS  wtageLOS wtsexLOS wtedLOS;
TABLE total,
 ((disch='' LOS='')*SUM=''*F=12.
  (WTEDLOS='')*SUM=''*F=6.2);
TABLE fage,
 ((disch='' LOS='')*SUM=''*F=12.
  (wtageLOS='')*SUM=''*F=6.2);
TABLE sex,
 ((disch='' LOS='')*SUM=''*F=12.
  (wtsexLOS='')*SUM=''*F=6.2);
TABLE (RACE dual mdst URBN PREG),
 ((disch='' LOS='')*SUM=''*F=12.
  (WTEDLOS='' )*SUM=''*F=6.2)
/RTS=28 ROW=FLOAT;
KEYLABEL ALL='All enrollees'
         sum='';
         LABEL   TOTAL='All enrollees'
                  FAGE='Age Group'
                   SEX='Sex'
                   URBN='Urban/Rural'
                   RACE='Race'
                   MDST='Medicare Eligibility Status'
                   DUAL='Other Disadvantaged Status'
                   PREG='Provider region';
RUN;
*****************************************************************
* STEP 5:  Output both unadjusted and adjusted LOS for individual
*          subgroup categories, and set the results to missing (.)
*          if the numerator count is 25 or less.
*****************************************************************;

data ADJLOS&yy;   set ADJLOS3 (rename=(disch_sum=dischsum&yy
                LOS_sum=LOSSUM&yy ));

* Compute unadjusted LOSs;
AVGLOS&yy = LOSSUM&yy / DISCHSUM&yy;
```

```
* Compute adjusted LOSs;
if total='1' then
     ADJLOS&YY=wtedLOS_sum;
else if sex ge 0 then
     ADJLOS&YY=wtsexLOS_sum;
else if fage ge 0 then
     ADJLOS&YY=wtAGELOS_sum;
else if race ge 0 or
        mdst ge 0 or
```
P1H4.@MSQ00AG.PGMLIB(ADLOS) -- CONTINUED
```
        dual ge 0 or
        urbn ge 0 or
        PREG ge 0 then
     ADJLOS&YY=wtedLOS_sum;
else delete;

* Set the adjusted LOSs to . if the numerator count is 25 or less;
if dischSUM&yy le 25 then ADJLOS&YY=.;

drop _type_ _page_ _table_ wtsexLOS_sum wtageLOS_sum WTEDLOS_sum;
run;

  /**END OF DIRECT ADJUSTMENT PROCEDURE**/

  /**BEGIN INDIRECT ADJUSTMENT PROCEDURE**/
*********************************************************************
* For each year, obtain observed(O) and expected(E) number of numerator
* events (i.e., CC discharges and benes with CC discharges) by state
* and multiply O/E to the 1999 national unadjusted LOS to compute
* the indirectly adjusted LOSs for individual states
*********************************************************************;

*********************************************************************
* Step 6: Compute expected outcome events by multiplying the standard
*         LOSs with the denominator counts for each age*sex*state.
*********************************************************************;

data M4&CC.AGP; set num&yy..M4&CC.AG (keep=y&yy.fage y&yy.sex y&yy.pst
                                       y&yy.preg disch&yy y&yy.los);
IF y&yy.preg='0' THEN y&yy.pst='0'; *SETTING PR,VI, & Canada to 0;

proc sql;
  Create table M4&CC.AGP2 as
  SELECT y&yy.fage, y&yy.SEX, y&yy.pst,
    sum(y&yy.los) as losasp,
    sum(disch&yy) as dischasp
  from M4&CC.AGP
  GROUP BY y&yy.fage, y&yy.SEX, y&yy.pst;
QUIT;

data combine;
merge M4&CC.AGP2 (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.pst=pst))
      M4&CC.STD (rename=(y99fage=fage y99sex=sex));
by fage sex;
explos = stdavglos * dischasp;
run;
```

```
************************************************************************
* Step 7: Sum up the observed LOS and expected LOS for each state.
************************************************************************;

proc SUMMARY DATA=COMBINE;
CLASS PST;
VAR losasp dischasp explos;
id nstdavglos;
OUTPUT OUT=IADJLOS
       SUM=lossum&YY dischsum&YY PSTEXPLOS;

************************************************************************
```

P1H4.@MSQ00AG.PGMLIB(ADLOS) -- CONTINUED

```
* Step 8: Compute unadjusted and indirectly adjusted average LOS by
*         provider state
************************************************************************;

DATA ILOS&YY; SET IADJLOS;
AVGLOS&YY = lossum&YY/dischsum&YY; * UNADJUSTED AVERAGE LOS;
SRR&YY = lossum&YY/ PSTEXPLOS;
adjlos&YY = nstdavglos * SRR&YY; * INDIRECTLY ADJUSTED AVERAGE LOS;

if _type_ in (0) then total='2';              *Setting U.S. total;
IF dischsum&YY LE 25 THEN adjlos&YY=.;        *Setting small cell to .;
DROP _TYPE_ _FREQ_ pstexplos;
run;

  /**END OF INDIRECT ADJUSTMENT PROCEDURE**/

************************************************************************
* Step 9: Combine the two (directly adjusted & indirectly adjusted
*         LOSs) files
************************************************************************;

DATA ADLOS&YY;   SET ADJLOS&YY ILOS&YY;
proc sort;
by total fage sex race dual mdst urbn preg pst;
run;

%MEND ADJ;

************************************************************************
* Step 10:  Run the adjustment macro for each year, and then combine
*           the yearly files into one file containing 10 year data.
************************************************************************;

%ADJ(92);
%ADJ(93);
%ADJ(94);
%ADJ(95);
%ADJ(96);
%ADJ(97);
%ADJ(98);
%ADJ(99);
%ADJ(00);
%ADJ(01);
```

```
DATA OUT2.&CC.TAB04;
MERGE ADLOS92 ADLOS93 ADLOS94 ADLOS95 ADLOS96
      ADLOS97 ADLOS98 ADLOS99 ADLOS00 ADLOS01;
by total fage sex race dual mdst urbn preg pst;

PROC PRINT DATA=OUT2.&CC.TAB04;
TITLE 'ADJUSTED AVERAGE LOS for &CC, 1992-2001';
RUN;

%MEND ADLOS;
```

D#MSQ.@MSQ00M5.PGMLIB92(M5PRE92)

```
//TI10M592 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE92 EXEC M5PRE,YY=92
```

D#MSQ.@MSQ00M5.PGMLIB93(M5PRE93)

```
//TI10M593 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE93 EXEC M5PRE,YY=93
```

D#MSQ.@MSQ00M5.PGMLIB94(M5PRE94)

D#MSQ.@MSQ00M5.PGMLIB95(M5PRE95)

```
//TI10M595 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE95 EXEC M5PRE,YY=95
```

D#MSQ.@MSQ00M5.PGMLIB96(M5PRE96)

```
//TI10M596 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE96 EXEC M5PRE,YY=96
```

D#MSQ.@MSQ00M5.PGMLIB97(M5PRE97)

```
//TI10M597 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE97 EXEC M5PRE,YY=97
```

D#MSQ.@MSQ00M5.PGMLIB98(M5PRE98)

```
//TI10M598 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE98 EXEC M5PRE,YY=98
```

D#MSQ.@MSQ00M5.PGMLIB99(M5PRE99)

```
//TI10M599 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE99 EXEC M5PRE,YY=99
```

D#MSQ.@MSQ00M5.PGMLIB00(M5PRE00)

```
//TI10M500 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE00 EXEC M5PRE,YY=00
```

D#MSQ.@MSQ00M5.PGMLIB01(M5PRE01)

```
//TI10M501 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=K
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PRE01 EXEC M5PRE,YY=01
```

D#MSQ.@MSQ00M5.PGMLIB99(M5PRE)

```
//M5PRE PROC
//*********************************************************************
//* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR M5 TABLE CREATIONS
//*********************************************************************
//*
//YXXM5001 EXEC SAS8,REGION=8000K,
//      PARM=('ERRORS=0,NEWS',
//      'SYSPARM=&YY'),COND=(4,LT)
//WORK   DD SPACE=(CYL,(2500,50))
//WORK1  DD SPACE=(CYL,(2500,50))
//WORK2  DD SPACE=(CYL,(2500,50))
//IN0 DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN1  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M4001.M4FLAGS
//IN2  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M4001.M4OTHER
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M5001.Y&YY.M5IN1,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//OUTF DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.FLAT.Y&YY.M5001.Y&YY.FNDR,
//        DCB=(RECFM=FB,LRECL=14,BLKSIZE=23464),
//        SPACE=(CYL,(80,10),RLSE)
//OUTE  DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M5001.SDERRS,
//        SPACE=(TRK,(460,10),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM501),DISP=SHR
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001)

```
%MACRO YXXM5001(yy);

%*************************************************************************
***
*************************************************************************
***

1  MACRO NAME:           YXXM5001
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 MARCH 2002
6  PROJECT:              MQMS
7  PURPOSE: FROM THE "TARGET CONDITION" M4 DISCHARGE FILE SELECT ONLY
            DISCHARGES RELEVANT TO M5 MEASURE AND CREATE SURGICAL DATE
             VARIABLE (DISCHARGES WITH ERRORS IN SURGICAL DATE ARE WRITTEN
IN
            A  SEPARATE  FILE  FOR  FUTHER  ANALYSIS),  SELECT  ALL  OTHER
DISCHARGES
            FROM M4 "OTHER" DISCHARGES FILE FOR PEOPLE WITH M5 INDEX
            DISCHARGES, CREATE A FINDER FILE TO BE AN INPUT FILE FOR DSAF
            TO GET CROSS-REFERENCED HICNOs
            (see macro YxxM5001 for more details) .


8  STEPS:   1. set macro parametrs.
              - REQVARS, CATVARS, DEFLAGS - list of varibles to take from
M4
                discharge files
              - LOWERB- is a Julin value of Januare 1, 1990.
                          it is the same for all the years.
              - UPPERB- is created as the last day of the year of medpar
                          file. it varies from year to year.
                the last 2 paramets are lower and upper boundary for valid
                surgical codes in julian date format (all medpar date
                variables come in this format).
                There are some errors in surgical dates and
                in order to avoid sas error messages we have to restrict
                usage of DATEJUL function only for valid dates.

               2. using "M4FLAGS" discharge level file with disease flags
create
                tempopary dataset with surgical date variable and output 2
files
                 "errors in surgical procedares codes" , "list of benes
with M5
                index records".
                   - convert admission and discharge dates to SAS
                     format
                   - call macro to create surgical date variable if
                     S4 flag > 0. Output "errors in surgical dates" file
                     if there is an error in the surgical date
                   - create SUM of M5 flags variable
                   - if SUM of M5 flags > 0 then output HICNO in the file
                     to keep all discharges for this person and use as
input
```

```
                        to get cross-references from DSAF.
                     - print data check dumps.
                3. from "M4OTHER" file select discharges only for people in
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001) -- CONTINUED

                   the list of people to keep.
                4. concatenate step2 and step3 datasets
                5. delete discharges for people without M5 cohort discharges,
                   output "all discharges file" .
                6. data check dumps and proc contents.

    9  YEAR OF DATA IS SET IN THE MAIN PROGRAM Yxx99M5001 AS 2 GLOBAL MACRO
       VARIABLES YY & YEAR.
    ************************************************************************
****;

                 /**CODE BEGINS**/

    ********************************************************************;
    *step1:  set macro parametrs
    ********************************************************************;

    %let year=19&yy;
      %if &yy<10 %then %let year=20&yy;

    *category variables;
    %LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
                 Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

    *disease flags;
    %LET DEFLAGS =Y&YY.HF Y&YY.M1 Y&YY.P1-Y&YY.P7 Y&YY.S1-Y&YY.S4;

    *other required variables;
    %LET REQVARS=Y&YY.HICNO Y&YY.ADMIT Y&YY.DISCH Y&YY.NOBS Y&YY.PRVID
                 Y&YY.DETH Y&YY.DOB
                 Y&YY.DIA1-Y&YY.DIA10 Y&YY.SPR1-Y&YY.SPR6
                 Y&YY.SDT1-Y&YY.SDT6 Y&YY.SCNT Y&YY.M14 Y&YY.VDOD;

    ********************************************************************;
    *step1:  set lower and upper boundary for valid surgical codes
    ********************************************************************;

     %GLOBAL LOWERB UPPERB;
     %LET LOWERB = 1990001; *January first,1990;
     DATA _NULL_;
             LEAP = (MOD(&YEAR,400) = 0  or
                    (MOD(&YEAR,4) = 0 & MOD(&YEAR,100) NE 0)); *leap year;
             LD=(&YEAR*1000+365+LEAP);
             CALL SYMPUT ("UPPERB",PUT(LD,7.));
     RUN;
     %PUT &UPPERB;
    ********************************************************************;
    * step 2: create temporary dataset with surgical date var
    *         and output 2 files
    ********************************************************************;
     DATA WORK1.FLAGS OUTE.SDERRS
          HICS(KEEP=Y&YY.HICNO);
```

```
      SET IN1.M4FLAGS(KEEP=&CATVARS &REQVARS &DEFLAGS
                      WHERE=(Y&YY.M14='1'));
      KEEP &CATVARS &DEFLAGS
      Y&YY.HICNO  Y&YY.SADM   Y&YY.SDIS
      Y&YY.NOBS   Y&YY.PRVID
      Y&YY.VDOD   Y&YY.DETH Y&YY.DOB
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001) -- CONTINUED

      Y&YY.DIA1-Y&YY.DIA10 Y&YY.SPR1-Y&YY.SPR6
      Y&YY.SDT1-Y&YY.SDT6
      Y&YY.SRDT ERRFL ;

      BY Y&YY.HICNO;

      *SAS dates;
      Y&YY.SDIS=DATEJUL(Y&YY.DISCH);
      Y&YY.SADM=DATEJUL(Y&YY.ADMIT);

      *to create surgical date variable;
      IF Y&YY.S4 > 0 THEN DO;
         %YXXSURDT(Y&YY.SPR, Y&YY.SDT , Y&YY.SCNT, Y&YY.SADM ,
                   Y&YY.SDIS , Y&YY.SRDT, ERRFL);
      END;

      OUTPUT WORK1.FLAGS;
      IF ERRFL THEN OUTPUT OUTE.SDERRS;

      *to find people with no flags from M5 cohorts and create finders;
      RETAIN ANYFLAG5 0;

      IF FIRST.Y&YY.HICNO THEN
         ANYFLAG5=0;

      SFLAGSM5= SUM(Y&YY.HF,Y&YY.P1,Y&YY.P6,Y&YY.P7,
                  Y&YY.S1,Y&YY.S3,Y&YY.S4,Y&YY.M1);

      ANYFLAG5 = SUM(ANYFLAG5, SFLAGSM5);

      IF LAST.Y&YY.HICNO THEN DO;
          *not M5 person;
          IF  ANYFLAG5 THEN DO;
             OUTPUT HICS;

            *flat file to input in DSAF;
             FILE OUTF;
             PUT @1  Y&YY.HICNO  $CHAR11.
                 @12 '   ';
          END;
      END;
 RUN;

 *print if surgical dates are outside admission interval;
 PROC PRINT U DATA=OUTE.SDERRS(OBS=200);
      VAR Y&YY.HICNO Y&YY.NOBS Y&YY.HF Y&YY.P1-Y&YY.P7
          Y&YY.S1-Y&YY.S4 Y&YY.M1
          Y&YY.SPR1- Y&YY.SPR6  Y&YY.SDT1- Y&YY.SDT6 Y&YY.SRDT
          Y&YY.SADM Y&YY.SDIS;
          FORMAT Y&YY.SRDT Y&YY.SADM Y&YY.SDIS MMDDYY.;
```

```
      TITLE1 "surgical dates are outside admission interval";
      TITLE2
      "or there are more then one stroke4 code in a discharge";

 RUN;
*******************************************************************;
* step 3: from "M4OTHER" file select discharges only for people with
*         M5 discharges
*******************************************************************;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001) -- CONTINUED

```
 DATA WORK2.OTHER;
      MERGE IN2.M4OTHER(IN=INO KEEP=&CATVARS &REQVARS)
            HICS(IN=INF);
      BY Y&YY.HICNO;
      IF INO & INF;

      KEEP &CATVARS &DEFLAGS
      Y&YY.HICNO Y&YY.SADM  Y&YY.SDIS Y&YY.PRVID
      Y&YY.NOBS
      Y&YY.SRDT Y&YY.DETH Y&YY.VDOD Y&YY.DOB;

      *SAS dates;
      Y&YY.SDIS=DATEJUL(Y&YY.DISCH);
      Y&YY.SADM=DATEJUL(Y&YY.ADMIT);

      LENGTH &DEFLAGS  2.;
      *set flags to 0;
      ARRAY FL  &DEFLAGS;
      DO OVER FL; FL=0; END;
      Y&YY.SRDT=.;

 RUN;

*******************************************************************;
* step 4: concatenate step2 and step3 datasets
*******************************************************************;
 DATA ALLD(KEEP=&CATVARS &DEFLAGS
      Y&YY.HICNO Y&YY.SADM  Y&YY.SDIS Y&YY.PRVID
      Y&YY.NOBS
      Y&YY.SRDT Y&YY.DETH Y&YY.VDOD Y&YY.DOB)
       /VIEW=ALLD;
      SET WORK1.FLAGS
          WORK2.OTHER;
      BY Y&YY.HICNO;
 RUN;
*******************************************************************;
* step 5: delete people without M5 cohort discharges;
*******************************************************************;

 DATA OUT1.Y&YY.M5IN1(
                     LABEL="MQMS, PROGRAM: Y&YY.M5001"
                     SORTEDBY=Y&YY.HICNO );
      MERGE ALLD(IN=IN1)
            HICS(IN=IN2);
      BY Y&YY.HICNO;
      IF IN1 & IN2;
      LABEL        Y&YY.HF ="heart failure discharge flag"
```

```
                  Y&YY.M1 ="AMI discharge flag"
                  Y&YY.S1="Stroke / TIA discharge flag"
                  Y&YY.S2="Stroke only discharge flag"
                  Y&YY.S3="Atrial Fibrillation discharge flag"
                  Y&YY.S4="Carotid endarterectomy discharge flag"
                  Y&YY.P1="Viral Pneumonia discharge flag"
                  Y&YY.P2="Pneumococcal Pneumonia discharge flag"
                  Y&YY.P3="Other bacterial Pneumonia discharge flag"
                  Y&YY.P4="Unspecified Pneumonia discharge flag"
                  Y&YY.P5="Influenza with Pneumonia discharge flag"
                  Y&YY.P6="Septicemia and Pneumonia discharge flag"
                  Y&YY.P7="Acute respiratory failure discharge flag"
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5001) -- CONTINUED

                  Y&YY.HICNO   ="patient ID"
                  Y&YY.SRDT    ="stroke4 surgical date "
                  Y&YY.SEX     ="sex                          "
                  Y&YY.STATE   ="state of residence           "
                  Y&YY.FAGE    ="Age group (1-9)              "
                  Y&YY.NOBS    ="unique claim number          "
                  Y&YY.PRVID   ="provider id                  "
                  Y&YY.CDIV    ="census division              "
                  Y&YY.CREG    ="census region                "
                  Y&YY.RACE    ="race                         "
                  Y&YY.URBN    ="urban flag                   "
                  Y&YY.SDIS    ="SAS discharge date"
                  Y&YY.SADM    ="SAS admission date"
                  Y&YY.DETH    ="date of death"
                  Y&YY.VDOD    ="=V if date of death is validated"
                   ;
 RUN;

*********************************************************************;
* step 6: datacheck dumps and proc contents;
*********************************************************************;

 PROC PRINT U DATA=OUT1.Y&YY.M5IN1(OBS=46);
      TITLE "THE FIRST INTERMEDIATE M5 FILE";
      FORMAT Y&YY.SDIS
             Y&YY.SADM  YYMMDD8.;
 RUN;
 PROC CONTENTS DATA=OUT1.Y&YY.M5IN1;
 RUN;

                    /**MACRO ENDS**/
%MEND YXXM5001;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXSURDT)

```
%MACRO YXXSURDT(SC,SD,SCNT,AD,DD,S4DATE,ERRFL);

*****************************************************************

1  MACRO NAME:           YXXSURDT
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXSURDT)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 JANUARY 30, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      CREATE SURGICAL DATE VARIABLE FOR DISCHARGES WITH S4 FLAG > 0
      DO SOME CLEANING OF SELECTED DATE VARIABLE


8  PARAMETERS:
       -input :
       -    SC - surgical code variable name,
       -    SD - surgical date variable name,
       -    SCNT - surgical procedures count,
       -    AD - admission date,
       -    DD - discharge date,
       - output :
       -    S4DATE - date of stroke4 code,
       -    ERRFL - error flag

9 COMMENTS:
      THIS MACRO USES GLOBAL VARIABLES &LOWERB AND &UPPERB WHICH ARE
      CREATED IN YXXM5001 AS WELL AS YY GLOBAL VARIABLE. THESE ARE LOWER
      AND UPPER BOUNDARIES OF VALID PROCEDURE DATES. THEY ARE NEEDED TO
      CHECK VALUE OF PROCEDURE DATE BEFORE APPLYING DATEJUL FUNCTION.
      IT IS DISCOVERED THAT THERE ARE NUMBER OF NONVALID VALUES OF DATES
      WHICH COUSE SAS ERROR MESSAGES.

      &LOWERB- IS A JULIN VALUE OF JANUARE 1, 1990.
               IT IS THE SAME FOR ALL THE YEARS.
      &UPPERB- IS CREADE AS THE LAST DAY OF THE YEAR OF MEDPAR FILE,
               IT VARIES FROM YEAR TO YEAR.
      &YY    - IS THE 2-DIGIT YEAR


11 EXAMPLE MACRO CALL:
%YXXSURDT(Y&YY.SPR, Y&YY.SDT , Y&YY.SCNT, Y&YY.SADM ,
          Y&YY.SDIS , Y&YY.SRDT, ERRFL)

*****************************************************************
*****************************************************************;

             /**CODE BEGINS**/

        *3812 is the procedure code for stroke4 cohort;

        ARRAY C(6) &SC.1-&SC.6; *surgical procedure codes;
        ARRAY DT(6) &SD.1-&SD.6; *surgical procedure dates;
        LENGTH &ERRFL 2.;
```

```
        CNTS=0;      *to count number of 3812 code in the record;
        &S4DATE=&DD; *to initially set date to the discharge date ;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXSURDT) -- CONTINUED

```
        IF &SCNT>0 THEN &SCNT=6;
        IF (&SCNT>0) THEN DO;
          DO I=1 TO &SCNT;
             IF C(I)='3812' THEN DO;  *S4 proc code;

                *check if date is within the boundaries;
                IF DT(I) NE . AND &LOWERB <= DT(I) <= &UPPERB
                   THEN DATETMP=DATEJUL(DT(I));

                *if date is within the boundaries take minimum if S4
                 code has been already met;
                IF DATETMP NE .
                    THEN &S4DATE = MIN (&S4DATE, DATETMP);
                ELSE &ERRFL=1; *error flag set to 1;
                CNTS=CNTS+1;
             END;
          END;
        END; *loop through surgical procedure codes;

        IF CNTS > 1 THEN &ERRFL=1;

        *check and correct created date variable if it falles out of
         admission interval;
        IF &S4DATE < &AD THEN &S4DATE=&AD;

 %MEND YXXSURDT;
```

D#MSQ.@MSQ00M5.PGMLIB92(M5PROD92)

```
//TI10M592 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YB PARAMETER (2 DIGIT YEAR).
//*           THE 2ND YEAR IN THE Y1 PARAMETER AND THE 3RD YEAR IN
//*           Y2 PARAMETER. SET YR = ALL 3 YEARS (4 DIGIT YEARS)
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD92 EXEC M5PROD,YR=199219931994,YB=92,Y1=93,Y2=94
```

D#MSQ.@MSQ00M5.PGMLIB93(M5PROD93)

```
//TI10M593 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD93 EXEC M5PROD,YR=199319941995,YB=93,Y1=94,Y2=95
```

D#MSQ.@MSQ00M5.PGMLIB94(M5PROD94)

```
//TI10M594 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD94 EXEC M5PROD,YR=199419951996,YB=94,Y1=95,Y2=96
```

D#MSQ.@MSQ00M5.PGMLIB95(M5PROD95)

```
//TI10M595 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD95 EXEC M5PROD,YR=199519961997,YB=95,Y1=96,Y2=97
```

D#MSQ.@MSQ00M5.PGMLIB96(M5PROD96)

```
//TI10M596 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD96 EXEC M5PROD,YR=199619971998,YB=96,Y1=97,Y2=98
```

D#MSQ.@MSQ00M5.PGMLIB97(M5PROD97)

```
//TI10M597 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB99'
//M5PROD97 EXEC M5PROD,YR=199719981999,YB=97,Y1=98,Y2=99
```

M5PROD98 & M5PROD99

D#MSQ.@MSQ00M5.PGMLI00(M5PROD00)

```
//TI10M500 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB00'
//M5PROD00 EXEC M5PROD,YR=20002001,YB=00,Y1=01
```

D#MSQ.@MSQ00M5.PGMLIB01(M5PROD01)

```
//TI10M501 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M5.PGMLIB01'
//M5PROD01 EXEC M5PROD1,YR=2001,YB=01
```

D#MSQ.@MSQ00M5.PGMLIB99(M5PROD)

```
//M5PROD PROC
//*********************************************************************
//* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR M5 TABLE CREATIONS
//*********************************************************************
//*
//YXXM5002 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK   DD SPACE=(CYL,(2500,50))
//WORK1  DD SPACE=(CYL,(2500,50))
//WORK2  DD SPACE=(CYL,(2500,50))
//WORK3  DD SPACE=(CYL,(2500,50))
//WORK4  DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN2  DD DISP=SHR,DSN=TI10.@0000000.Y&YB.M5XRF.OUTPUT.NODUPE
//IN1  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5001.Y&YB.M5IN1
//IN3  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&Y1.M1001.DENOMA&Y1
//IN4  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&Y2.M1001.DENOMA&Y2
//OUT DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&YB.M5IN2,
//       LABEL=(EXPDT=2007/365),UNIT=MCART
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&YB.M5XR,
//       SPACE=(CYL,(420,5),RLSE)
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//       DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&Y1.M5DUP,
//       SPACE=(CYL,(30,10),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM52A),DISP=SHR
//*
//YXXM5003 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK   DD SPACE=(CYL,(2500,50))
//WORK1  DD SPACE=(CYL,(2500,50))
//WORK2  DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN1  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5001.Y&YB.M5IN1
//IN2  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&YB.M5IN2
//OUT DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5003.Y&YB.M5IN3,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM503),DISP=SHR
//*
//YXXM504A EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK   DD SPACE=(CYL,(2000,50))
//WORK1  DD SPACE=(CYL,(2000,50))
//WORK2  DD SPACE=(CYL,(1500,50))
//WORK3  DD SPACE=(CYL,(1500,50))
//WORKX  DD SPACE=(CYL,(1500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//INX  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB&YB
//IN2  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&YB.M5IN2
//IN3  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5003.Y&YB.M5IN3
//INXR DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&YB.M5XR
//INDP DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5002.Y&Y1.M5DUP
//INY1 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&Y1.M2007.MEDPRG&Y1
//INY2 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&Y2.M2007.MEDPRG&Y2
```

```
//OUT DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5004.Y&YB.M5IN4,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
```
D#MSQ.@MSQ00M5.PGMLIB99(M5PROD) -- CONTINUED

```
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM54A),DISP=SHR
//*
//YXXM5005 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK   DD SPACE=(CYL,(1500,50))
//WORK1  DD SPACE=(CYL,(1500,50))
//WORK2  DD SPACE=(CYL,(1500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN1  DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5004.Y&YB.M5IN4
//OUT DD DISP=(NEW,CATLG,KEEP),
//         DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5005.Y&YB.M5ANA,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//WORKOUT DD DISP=(NEW,CATLG,KEEP),
//         DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5005.WORKOUT,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM505),DISP=SHR
//*
//DELSTEP EXEC PGM=IEFBR14,COND=(4,LT)
//DX DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5005.WORKOUT,DISP=(OLD,DELETE)
//*
//YXXM5006 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5005.Y&YB.M5ANA
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//OUT DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5006.Y&YB.MEASR,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM506),DISP=SHR
//*
//YXXM5007 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5006.Y&YB.MEASR,DISP=SHR
//OUT1 DD DISP=(NEW,CATLG,CATLG),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5007.SORTM5,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//OUT2 DD DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(50,5),RLSE),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5007.GRPWT&YB
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM507),DISP=SHR
//*
//YXXM5008 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK DD SPACE=(CYL,(2500,50))
//IN0   DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5007.GRPWT&YB,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5007.SORTM5,DISP=SHR
//OUT DD DISP=(NEW,CATLG,CATLG),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5008.M5TAB&YB,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM508),DISP=SHR
```

```
//*
//YXXM5009 EXEC SAS8,REGION=8000K,COND=(4,LT),
//      PARM=('ERRORS=0,NEWS,NOCENTER','SYSPARM=&YR')
//WORK DD SPACE=(CYL,(2500,50))
//IN0   DD DISP=SHR,DSN=D#MSQ.@MSQ00M5.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5008.M5TAB&YB,DISP=SHR
```
D#MSQ.@MSQ00M5.PGMLIB99(M5PROD) -- CONTINUED

```
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5009.TABLEA,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5009.TABLEB,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YB.M5009.TABLEC,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M5.PGMLIB99(SYSM509),DISP=SHR


%INCLUDE IN0(YXXM5002)/ SOURCE2;     *PROGRAM STEPS;

       /**MACRO CALL**/

%YXXM5002(&SYSPARM);
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5002)

```
%MACRO YXXM5002(YVAR);

%*****************************************************************
*****************************************************************

1  MACRO NAME:           YXXM5002
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM5002)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 MARCH 2002
6  PROJECT:              MQMS
7  PURPOSE   :CREATE PERSON LEVEL FILE WITH "END OF ENROLLMENT"  AND
              "DEATH" VARIABLES, KEEP DEMOGRAPHIC VARIABES FROM THE FIRST
              YEAR DENOMINATOR FILE
8  STEPS     : 1. set macro parametr
                 - CATVARS list of categorical varibles
               2. check cross-reference file and correct if nesessary
               3. select one record for each HICNO with the first year
                  denominator variables
               4. from the second year denominator create date with hmo
                  or no part a eligibility
               5. merge step 4 file with xref file
               6. sort by hicno, find duplicates and write them to a
separate
                  file
               7. create date of death, disenrollment date, create
                  output file
               8. data check dumps and proc contents

*****************************************************************;
OPTIONS BUFNO=5 ;

*****************************************************************;
* step1: macro variables
*****************************************************************;

*year variables;
%let YY=%SUBSTR(&yvar,3,2);
%let YN=%SUBSTR(&yvar,7,2);
%let YEAR=%SUBSTR(&yvar,1,4);
%let YEARN=%SUBSTR(&yvar,5,4);

*category variables;
%LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
             Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

*****************************************************************;
* step2: check cross-reference file and correct if nesessary
*****************************************************************;
 DATA XREFD;
      INFILE IN2;
      INPUT  @1 Y&YY.HICNO     $CHAR11.
            @17 XREF           $CHAR11.;
      OUTPUT;
      IF XREF NE Y&YY.HICNO THEN DO;
          XREF=Y&YY.HICNO;
```

```
          OUTPUT;
      END;
 RUN;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5002) -- CONTINUED
```
 PROC SORT DATA=XREFD;
      BY XREF Y&YY.HICNO;
 RUN;
 DATA OUT1.Y&YY.M5XRF ;
      SET XREFD;
      BY XREF;
      IF FIRST.XREF;
 RUN;
 PROC PRINT U DATA=OUT1.Y&YY.M5XRF(OBS=46);
      TITLE "M5 CROSS-REFERENCES";
 RUN;
 PROC CONTENTS;RUN;
*******************************************************************;
* step3: select one record for each HICNO with first year
*        denominator variables
*******************************************************************;
 DATA WORK2.DEN&YY;
      SET IN1.Y&YY.M5IN1
          (KEEP=&CATVARS Y&YY.HICNO Y&YY.DETH Y&YY.DOB Y&YY.VDOD);
       BY Y&YY.HICNO;
       IF FIRST.Y&YY.HICNO;
 RUN;

*******************************************************************;
* step4: second year denominator
*        create date of hmo or no part a eligibility
*******************************************************************;
 DATA WORK1.TEMP(KEEP=XREF Y&YN.DETH Y&YN.VDOD Y&YN.HMOPA)
      CHECK;
      SET IN3.DENOMA&YN(KEEP=Y&YN.HICNO Y&YN.HMO1-Y&YN.HMO12
                        Y&YN.BUY1-Y&YN.BUY12 Y&YN.DETH Y&YN.VDOD
                        RENAME=(Y&YN.HICNO=XREF));

       ARRAY BUY(12) $ Y&YN.BUY1-Y&YN.BUY12 ;
       ARRAY HMO(12) $ Y&YN.HMO1-Y&YN.HMO12 ;

   /*****************************************/
   /** create month of PartA disenrollment or **/
   /** HMO enrollment in TEMPM               **/
   /*****************************************/;
       TEMPM = 0;
       IF Y&YN.DETH NE . THEN IE=MONTH(Y&YN.DETH);
       ELSE IE = 12;

       DO I=1 TO IE WHILE (TEMPM =0);
       TEMPM = I*(
                  (UPCASE(BUY(I)) NOT IN ('1','3','A','C'))
                   OR (HMO(I) NE '0')
                  );
       END;

   /*****************************************************/
   /** set date as the first day of month TEMPM,year YEARN**/
```

```
   /*********************************************************/;
       IF TEMPM > 0 THEN
           Y&YN.HMOPA=MDY(TEMPM, 1, &YEARN);

       OUTPUT WORK1.TEMP;
       IF TEMPM > 0 & _N_ < 55000 THEN OUTPUT CHECK;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5002) -- CONTINUED

```
 RUN;

   /*****************/
   /** data check  **/
   /*****************/;
 PROC PRINT U DATA=CHECK(OBS=46);
      TITLE "DENOMINATOR &YEARN. DATA CHECK";
      FORMAT Y&YN.DETH Y&YN.HMOPA  YYMMDD8.;
 RUN;

*******************************************************************;
* step5: merge step 4 file with xref file
*******************************************************************;

 PROC SORT DATA=WORK1.TEMP OUT=WORK3.TEMP;
      BY XREF;
 RUN;

 DATA WORK2.TD(DROP=XREF) NM(KEEP=XREF Y&YY.HICNO);
      MERGE WORK3.TEMP(IN=IN1)
            OUT1.Y&YY.M5XRF(IN=IN2);
      BY XREF;
      IF IN2 & NOT IN1 THEN OUTPUT NM;
      ELSE
      IF IN2 & IN1 THEN OUTPUT WORK2.TD;
 RUN;

*******************************************************************;
* step6. sort by hicno, find duplicates
*******************************************************************;

 PROC SORT DATA=WORK2.TD;
      BY Y&YY.HICNO;
 RUN;
 DATA WORK2.DEN&YN OUT2.Y&YN.M5DUP;
      SET WORK2.TD;
      BY Y&YY.HICNO;
      IF NOT FIRST.Y&YY.HICNO OR NOT LAST.Y&YY.HICNO
         THEN OUTPUT OUT2.Y&YN.M5DUP;
      ELSE OUTPUT WORK2.DEN&YN;
 RUN;

   /***********************************/
   /**          check nonmatches       **/
   /***********************************/;
 PROC SQL;
      CREATE TABLE NONM AS
        SELECT DISTINCT Y&YY.HICNO FROM NM
        WHERE Y&YY.HICNO NOT IN
              SELECT DISTINCT Y&YY.HICNO FROM WORK2.DEN&YN;
```

```
 RUN;

   /***********************************/
   /**      print nonmatches           **/
   /***********************************/;
  PROC PRINT U DATA=NONM(OBS=46);
       TITLE "NO RECORDS IN DENOMINATOR &YEARN";
  RUN;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5002) -- CONTINUED

```
*********************************************************************;
* step7: create date of death, disenrollment date, create output file
*********************************************************************;
  DATA OUT.Y&YY.M5IN2(LABEL="MQMS: PROGRAM Y&YY.M5001"
                     SORTEDBY=Y&YY.HICNO
                     KEEP= Y&YY.HICNO  Y&YY.M5DEATH Y&YY.ENDENR
                           Y&YY.DOB
                           Y&YY.DETH   Y&YY.VDOD    Y&YY.DENFL
                           Y&YN.DETH   Y&YN.VDOD    Y&YN.DENFL
                           &CATVARS);

       MERGE WORK2.DEN&YY(IN=IN1)
             WORK2.DEN&YN(IN=IN2);
       BY Y&YY.HICNO;
       LENGTH Y&YY.DENFL Y&YN.DENFL 2.;
       Y&YY.DENFL =IN1;
       Y&YN.DENFL =IN2;

      /*********************/
      /** date of death   **/
      /*********************/;
       IF Y&YN.VDOD='V' THEN Y&YY.M5DEATH = Y&YN.DETH;
       ELSE
       IF Y&YY.VDOD='V' THEN Y&YY.M5DEATH = Y&YY.DETH;
       ELSE
       IF Y&YN.DETH NE . THEN Y&YY.M5DEATH = Y&YN.DETH;
       ELSE
       IF Y&YY.DETH NE . THEN Y&YY.M5DEATH = Y&YY.DETH;
       *else M5DEATH is missing;

      /*************************************/
      /** disenrollment date variable update **/
      /*************************************/ ;
       *first set it to the third year January 1 for all the benes;
       Y&YY.ENDENR="1JAN%EVAL(&YEARN+1)"D;

       * if there are no records in denominator &yn;
       IF IN1 & NOT IN2 THEN TEMPDIS="1JAN&YEARN"D;

       ARRAY DD(3) Y&YY.M5DEATH Y&YN.HMOPA TEMPDIS;
       DO I=1 TO 3;
          IF DD(I) NE . & DD(I)< Y&YY.ENDENR THEN
              Y&YY.ENDENR=DD(I);
       END;

      /***************/
```

```
      /** labels    **/
      /***************/;
       LABEL
           Y&YY.M5DEATH  = "latest date of death"
           Y&YY.ENDENR   = "disqualification date"
           Y&YY.DENFL     = "=0 if no data in &YY denomin,=1 otherwise"
           Y&YN.DENFL     = "=0 if no data in &YN denomin,=1 otherwise"
           Y&YY.DETH     = "date of death from &YY denominator"
           Y&YN.DETH     = "date of death from &YN denominator"
           Y&YY.VDOD     = "valid date of death from &YN denominator"
           Y&YN.VDOD     = "valid date of death from &YN denominator"
       ;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5002) -- CONTINUED

```
       OUTPUT OUT.Y&YY.M5IN2;
  RUN;

*****************************************************************;
* step 8: data check dumps and proc contents;
*****************************************************************;

  PROC PRINT U DATA=OUT.Y&YY.M5IN2(OBS=46);
       TITLE1 "M5 SECOND INTERMEDIATE FILE";
       TITLE2 "DENOMINATORS &YY AND &YN ARE MERGED";
       FORMAT Y&YY.M5DEATH Y&YY.DETH Y&YN.DETH YYMMDD8.;
  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5IN2;RUN;

                    /**MACRO ENDS**/

%MEND YXXM5002;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A)

```
%MACRO YXXM502A(YVAR);

*********************************************************************

1  MACRO NAME:           YXXM502A
2  LOCATION:             D#MSQ.@MSQ00M2.PGMLIB99(YXXM502A)
3  WRITTEN BY:           HELEN MARGULIS
3  EDITED BY:            ANN LARSEN
4  COMPANY:              RTI
5  DATE:                 NOVEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE   :CREATE PERSON LEVEL FILE WITH "END OF ENROLLMENT"  AND
              "DEATH" VARIABLES, KEEP DEMOGRAPHIC VARIABES FROM THE FIRST
              YEAR DENOMINATOR FILE
8  STEPS     : 1. set macro parametr
                 - CATVARS list of categorical varibles
               2. check cross-reference file and correct if nesessary
               3. select one record for each HICNO with the first year
                  denominator variables
               4. from the second year denominator create date with hmo
                  or no part a eligibility (Y1HMOPA) keep date of death
              4a. from the third year denominator create date with hmo
                  or no part a eligibility (Y2HMOPA) keep date of death
               5. merge files from steps 4 & 4a with the xref file
               6.  sort  by  hicno,  find  duplicates  and  write  them  to  a
separate
                  file
               7. create date of death, disenrollment date, create
                  output file
               8. data check dumps and proc contents

*********************************************************************/;
OPTIONS BUFNO=5 ;

*********************************************************************;
* step1: macro variables
*********************************************************************;

*year variables;
%let YB=%SUBSTR(&YVAR,3,2);
%let Y1=%SUBSTR(&YVAR,7,2);
%let Y2=%SUBSTR(&YVAR,11,2);
%let YEARB=%SUBSTR(&YVAR,1,4);
%let YEAR1=%SUBSTR(&YVAR,5,4);
%let YEAR2=%SUBSTR(&YVAR,9,4);

*category variables;
%LET CATVARS=Y&YB.CDIV Y&YB.CREG Y&YB.DUAL  Y&YB.FAGE Y&YB.MDST
             Y&YB.RACE Y&YB.SEX  Y&YB.STATE Y&YB.URBN;

*********************************************************************;
* step2: check cross-reference file and correct if nesessary
*********************************************************************;
 DATA XREFD;
      INFILE IN2;
      INPUT  @1 Y&YB.HICNO     $CHAR11.
```

```
           @17 XREF           $CHAR11.;
      OUTPUT;
      IF XREF NE Y&YB.HICNO THEN DO;
D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A) -- CONTINUED

          XREF=Y&YB.HICNO;
          OUTPUT;
      END;
 RUN;
 PROC SORT DATA=XREFD;
      BY XREF Y&YB.HICNO;
 RUN;
 DATA OUT1.Y&YB.M5XRF ;
      SET XREFD;
      BY XREF;
      IF FIRST.XREF;
 RUN;
 PROC PRINT U DATA=OUT1.Y&YB.M5XRF(OBS=46);
      TITLE "M5 CROSS-REFERENCES";
 RUN;
 PROC CONTENTS;RUN;
*******************************************************************;
* step3: select one record for each HICNO with first year
*        denominator variables
*******************************************************************;
 DATA WORK2.DEN&YB;
      SET IN1.Y&YB.M5IN1
          (KEEP=&CATVARS Y&YB.HICNO Y&YB.DETH Y&YB.DOB Y&YB.VDOD);
       BY Y&YB.HICNO;
       IF FIRST.Y&YB.HICNO;
 RUN;
 proc print data=work2.den&yb(obs=50); run;
*******************************************************************;
* step4: second year denominator
*        create date of hmo or no part a eligibility
*******************************************************************;
 DATA WORK1.TEMP(KEEP=XREF Y&Y1.DETH Y&Y1.VDOD Y&Y1.HMOPA)
      CHECK;
      SET IN3.DENOMA&Y1(KEEP=Y&Y1.HICNO Y&Y1.HMO1-Y&Y1.HMO12
                        Y&Y1.BUY1-Y&Y1.BUY12 Y&Y1.DETH Y&Y1.VDOD
                        RENAME=(Y&Y1.HICNO=XREF));

       ARRAY BUY(12) $ Y&Y1.BUY1-Y&Y1.BUY12 ;
       ARRAY HMO(12) $ Y&Y1.HMO1-Y&Y1.HMO12 ;

   /*****************************************/
   /** create month of PartA disenrollment or **/
   /** HMO enrollment in TEMPM                **/
   /*****************************************/;
       TEMPM = 0;
       IF Y&Y1.DETH NE . THEN IE=MONTH(Y&Y1.DETH);
       ELSE IE = 12;

       DO I=1 TO IE WHILE (TEMPM =0);
       TEMPM = I*(
                  (UPCASE(BUY(I)) NOT IN ('1','3','A','C'))
                   OR (HMO(I) NE '0')
                  );
```

```
       END;

   /*******************************************************/
   /** set date as the first day of month TEMPM,year YEAR1 */
   /*******************************************************/;
       IF TEMPM > 0 THEN
D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A) -- CONTINUED

           Y&Y1.HMOPA=MDY(TEMPM, 1, &YEAR1);

       OUTPUT WORK1.TEMP;
       IF TEMPM > 0 & _N_ < 55000 THEN OUTPUT CHECK;
 RUN;

   /*****************/
   /** data check  **/
   /*****************/;
 PROC PRINT U DATA=CHECK(OBS=46);
      TITLE "DENOMINATOR &YEAR1. DATA CHECK";
      FORMAT Y&Y1.DETH Y&Y1.HMOPA  YYMMDD8.;
 RUN;
 PROC PRINT U DATA=work1.temp(OBS=46);
      TITLE "DENOMINATOR &YEAR1. DATA temp1";
      FORMAT Y&Y1.DETH Y&Y1.HMOPA  YYMMDD8.;
 RUN;


*********************************************************************;
* step4a: third year denominator
*        create date of hmo or no part a eligibility
*********************************************************************;
 DATA WORK4.TEMP(KEEP=XREF Y&Y2.DETH Y&Y2.VDOD Y&Y2.HMOPA)
      CHECK;
      SET IN4.DENOMA&Y2(KEEP=Y&Y2.HICNO Y&Y2.HMO1-Y&Y2.HMO12
                        Y&Y2.BUY1-Y&Y2.BUY12 Y&Y2.DETH Y&Y2.VDOD
                        RENAME=(Y&Y2.HICNO=XREF));

       ARRAY BUY(12) $ Y&Y2.BUY1-Y&Y2.BUY12 ;
       ARRAY HMO(12) $ Y&Y2.HMO1-Y&Y2.HMO12 ;

   /*****************************************/
   /** create month of PartA disenrollment or **/
   /** HMO enrollment in TEMPM               **/
   /*****************************************/;
       TEMPM = 0;
       IF Y&Y2.DETH NE . THEN IE=MONTH(Y&Y2.DETH);
       ELSE IE = 12;

       DO I=1 TO IE WHILE (TEMPM =0);
       TEMPM = I*(
                  (UPCASE(BUY(I)) NOT IN ('1','3','A','C'))
                   OR (HMO(I) NE '0')
                  );
       END;

   /*******************************************************/
   /** set date as the first day of month TEMPM,year YEAR2 */
   /*******************************************************/;
```

```
       IF TEMPM > 0 THEN
           Y&Y2.HMOPA=MDY(TEMPM, 1, &YEAR2);

       OUTPUT WORK4.TEMP;
       IF TEMPM > 0 & _N_ < 55000 THEN OUTPUT CHECK;
 RUN;

   /*****************/
   /** data check  **/
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A) -- CONTINUED
```
   /*****************/;
 PROC PRINT U DATA=CHECK(OBS=46);
      TITLE "DENOMINATOR &YEAR2. DATA CHECK";
      FORMAT Y&Y2.DETH Y&Y2.HMOPA  YYMMDD8.;
 RUN;
 PROC PRINT U DATA=work4.temp(OBS=46);
      TITLE "DENOMINATOR &YEAR2. DATA temp4";
      FORMAT Y&Y2.DETH Y&Y2.HMOPA  YYMMDD8.;
 RUN;

 PROC SORT DATA=WORK1.TEMP OUT=WORK3.TEMP;
      BY XREF;
 RUN;

*******************************************************************;
* step5: merge step 4 and 4a file with xref file
*******************************************************************;

 DATA WORK2.TD(DROP=XREF) NM(KEEP=XREF Y&YB.HICNO);
      MERGE WORK3.TEMP(IN=IN1)
            WORK4.TEMP(IN=IN3)
            OUT1.Y&YB.M5XRF(IN=IN2);
      BY XREF;
      Y&Y2.DENFL=(IN3);
      Y&Y1.DENFL=(IN1);
      IF IN2 & NOT IN1 THEN OUTPUT NM;
      ELSE
      IF IN2 & IN1 THEN OUTPUT WORK2.TD;
 RUN;

*******************************************************************;
* step6. sort by hicno, find duplicates
*******************************************************************;

 PROC SORT DATA=WORK2.TD;
      BY Y&YB.HICNO;
 RUN;
 proc print data=work2.td(obs=48); run;

 DATA WORK2.DEN&Y1 OUT2.Y&Y1.M5DUP;
      SET WORK2.TD;
      BY Y&YB.HICNO;
      IF NOT FIRST.Y&YB.HICNO OR NOT LAST.Y&YB.HICNO
         THEN OUTPUT OUT2.Y&Y1.M5DUP;
      ELSE OUTPUT WORK2.DEN&Y1;
 RUN;
 proc print data=work2.den&Y1(obs=48); run;
```

```
 proc print data=OUT2.y&Y1.m5dup(obs=48); run;

   /***********************************/
   /**          check nonmatches       **/
   /***********************************/;
 PROC SQL;
      CREATE TABLE NONM AS
        SELECT DISTINCT Y&YB.HICNO FROM NM
        WHERE Y&YB.HICNO NOT IN
              SELECT DISTINCT Y&YB.HICNO FROM WORK2.DEN&Y1;

 RUN;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A) -- CONTINUED

```
   /***********************************/
   /**      print nonmatches           **/
   /***********************************/;
  PROC PRINT U DATA=NONM(OBS=46);
       TITLE "NO RECORDS IN DENOMINATOR &yearb";
  RUN;

*********************************************************************;
* step7: create date of death, disenrollment date, create output file
*********************************************************************;
  DATA OUT.Y&YB.M5IN2(LABEL="MQMS: PROGRAM Y&YB.M5001"
                     SORTEDBY=Y&YB.HICNO
                     KEEP= Y&YB.HICNO  Y&YB.M5DEATH Y&YB.ENDENR
                           Y&YB.DOB
                           Y&YB.DETH Y&YB.VDOD Y&YB.DENFL
                           Y&Y1.DETH Y&Y1.VDOD Y&Y1.DENFL Y&Y1.HMOPA
                           Y&Y2.DETH Y&Y2.VDOD Y&Y2.DENFL Y&Y2.HMOPA
                           &CATVARS);

       MERGE WORK2.DEN&YB(IN=IN1)
             WORK2.DEN&Y1(IN=IN2);
       BY Y&YB.HICNO;
       LENGTH Y&YB.DENFL Y&Y1.DENFL 2. Y&Y2.DENFL 2.;
       Y&YB.DENFL =IN1;

      /*********************/
      /** date of death   **/
      /*********************/;
       IF Y&Y2.VDOD='V' THEN Y&YB.M5DEATH = Y&Y2.DETH;
       ELSE
       IF Y&Y1.VDOD='V' THEN Y&YB.M5DEATH = Y&Y1.DETH;
       ELSE
       IF Y&YB.VDOD='V' THEN Y&YB.M5DEATH = Y&YB.DETH;
       ELSE
       IF Y&Y2.DETH NE . THEN Y&YB.M5DEATH = Y&Y2.DETH;
       ELSE
       IF Y&Y1.DETH NE . THEN Y&YB.M5DEATH = Y&Y1.DETH;
       ELSE
       IF Y&YB.DETH NE . THEN Y&YB.M5DEATH = Y&YB.DETH;
       *else M5DEATH is missing;

      /*************************************/
      /** disenrollment date variable update **/
```

```
      /*************************************/ ;
       *first set it to the third year January 1 for all the benes;
       Y&YB.ENDENR="1JAN%EVAL(&yearb+3)"D;

       if (y&Y1.denfl=.) then y&Y1.denfl=0;
       if (y&Y2.denfl=.) then y&Y2.denfl=0;

       * if there are no records in denominator YEAR+1;
       IF IN1 & Y&Y1.DENFL=0 THEN TEMPDIS="1JAN&YEAR1"D;
       * if there are no records in denominator YEAR+2;
       else IF IN1 & Y&Y2.DENFL=0 THEN TEMPDIS="1JAN&YEAR2"D;

       ARRAY DD(4) Y&YB.M5DEATH Y&Y2.HMOPA TEMPDIS Y&Y1.HMOPA;
       DO I=1 TO 4;
          IF DD(I) NE . & DD(I)< Y&YB.ENDENR THEN
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM502A) -- CONTINUED
```
              Y&YB.ENDENR=DD(I);
       END;

      /***************/
      /** labels    **/
      /***************/;
       LABEL
           Y&YB.M5DEATH  = "latest date of death"
           Y&YB.ENDENR   = "disqualification date"
           Y&YB.DENFL     = "=0 if no data in &YB denomin,=1 otherwise"
           Y&Y1.DENFL     = "=0 if no data in &Y1 denomin,=1 otherwise"
           Y&Y2.DENFL     = "=0 if no data in &Y2 denomin,=1 otherwise"
           Y&YB.DETH     = "date of death from &YB denominator"
           Y&Y1.DETH     = "date of death from &Y1 denominator"
           Y&Y2.DETH     = "date of death from &Y2 denominator"
           Y&YB.VDOD     = "valid date of death from &YB denominator"
           Y&Y1.VDOD     = "valid date of death from &Y1 denominator"
           Y&Y2.VDOD     = "valid date of death from &Y2 denominator"
       ;
       OUTPUT OUT.Y&YB.M5IN2;
  RUN;

*****************************************************************;
* step 8: data check dumps and proc contents;
*****************************************************************;

  PROC PRINT U DATA=OUT.Y&YB.M5IN2(OBS=46);
       TITLE1 "M5 SECOND INTERMEDIATE FILE";
       TITLE2 "DENOMINATORS &YB AND &Y1 AND &Y2 ARE MERGED";
       FORMAT Y&YB.M5DEATH Y&YB.DETH Y&Y1.DETH Y&Y2.DETH
              Y&Y1.HMOPA Y&Y2.HMOPA
              Y&YB.ENDENR Y&YB.DOB YYMMDD8.;
  RUN;
  PROC CONTENTS DATA=OUT.Y&YB.M5IN2;RUN;
%MEND YXXM502A;
```

D#MSQ.@MSQ00M5.PGMLIB99(SYSM503)

```
%INCLUDE IN0(YXXM5003)/ SOURCE2;     *PROGRAM STEPS;
%INCLUDE IN0(YXXMEASR)/ SOURCE2;     *DEFINE MEASURES;

       /**MACRO CALL**/

%YXXM5003(&SYSPARM);
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003)

```
%MACRO YXXM5003(YVAR);

%*************************************************************************
*****
*************************************************************************
*****

1  MACRO NAME:           YXXM5003
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JUNE 2002
6  PROJECT:              MQMS
7  PURPOSE:  TO CREATE INDEX FLAGS IN ALL DISCHARGES FIRST YEAR FILE
             FOR EACH MEASURE AND EACH WINDOW. DISENROLLMENT DATE IS
             NOT TAKEN INTO CONSIDERATION

8  STEPS     : 1. definitions

               - CATVARS: tables category variables
               - DEFLAGS: disease flags
               - NUM_WIN: number of day windows
               - NUM_MEA: number of measures
               - NUM_COH: number of disease cohorts
               - MSIZE: size of matrix to keep index variables
               - $DISEASE: format to go from disease flag to disease
                           index
               - DWINDOW: string containing number of days in each
                          window
             2. define macro to set index discharges flags for each
                cohort if the corresponding disease flag > 0. Cohorts
                loop from 1 to NUM_COH. Measures loop from the first
                measure to the last measure NUM_MEA
             3. call macro YXXMEASR to define each measure by creating
                macro variables DENOM, DFLAG, COND
             4. sort &YY discharge file by hicno,admdate,disdate
             5. delete records before the first discharge with one of
                the disease flags. merge denominator variables
             6. create index flags and create an output file
               - set arrays for each disease(cohort) for input flags
               - set output variables
               - set temporary flags to keep in memory for a particular
                 HICNO
               - for the first record for a person set to zero
                 memory variables
               - put zeros in output index flags
               - if this record is flagged with any flag then call
                 macro to set index in each relevant measure.
             7. datacheck printout and proc contents
*************************************************************************
****;
*************************************************************************
****;

OPTIONS BUFNO=5 ;
%INCLUDE IN0(YXXMEASR);
```

```
*********************************************************************;
* step1: macro variables
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003) -- CONTINUED

*********************************************************************;

*year variables;
%let YY=%SUBSTR(&yvar,3,2);

      /**category vars**/;

%LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
             Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

      /**disease flags**/;
%LET DEFLAGS =Y&YY.HF  Y&YY.M1 Y&YY.P1-Y&YY.P7
               Y&YY.S1-Y&YY.S4;

   /**number of diseases, windows, measures **/;

%LET NUM_WIN = 6;  *number of day windows;
%LET NUM_MEA = 34; *number of measures;
%LET NUM_COH = 8;  *number of disease cohorts;
%LET MSIZE   = %EVAL(&NUM_WIN*&NUM_MEA) ; *matrix size;

   /**number of days in windows **/;

%LET DWINDOW = %STR(2 7 30 180 365 730);  *days in each window;

   /** format to translate disease into **/
   /** variable index, list of cohorts   **/;

PROC FORMAT;
     VALUE $DISEASE
       'HF'= 1
       'P1'= 2
       'P6'= 3
       'P7'= 4
       'S1'= 5
       'S3'= 6
       'S4'= 7
       'M1'= 8  ;
RUN;

*********************************************************************;
* step2 : to set index discharges flags for all flagged cohorts
*********************************************************************;

%MACRO COH;
     /*********************************/
     /**  indexes:                   **/
     /**  J - count measures,        **/
     /**  I - count diseases(cohorts)**/
     /**  K - count windows          **/
     /*********************************/;

     %DO J=1 %TO &NUM_MEA;
```

```
        *find index I for the flag which define this measure;
        I = INPUT(PUT("&&DFLAG&J", $DISEASE.),2.);
        IF FL(I) THEN DO;

          *if this is a BENE measure then temporary variable TINDEX
          *is equal 1 or 0 depending on FOUND flag for measure J,
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003) -- CONTINUED
```
          *for a discharge level measure TINDEX=1 ;
          IF "&&DENOM&J" = "BENE" THEN TINDEX=(NOT FOUND(&J));
          ELSE TINDEX=1;

          *turn "on" index flags for this measure for all windows;
          IF TINDEX THEN
             DO K=1 TO &NUM_WIN;
                IWM(K,&J)=TINDEX;
             END;

          *to turn "ON" FOUND flag for this measure: first index
           discharge for the bene is found for measure j;
          FOUND(&J)=1;
        END;    *end if this measure is FL(I) measure;
     %END;      *end of loop by measure j;

%MEND COH;
*********************************************************************;
* step3 : call macro to define each measure;
*********************************************************************;

 %YXXMEASR(&NUM_MEA,&NUM_COH);

*********************************************************************;
* step4: sort &YY discharge file by hicno,admdate,disdate
*********************************************************************;

  PROC SORT DATA=IN1.Y&YY.M5IN1 OUT=WORK1.TEMP1;
       BY Y&YY.HICNO Y&YY.SADM Y&YY.SDIS;
  RUN;

*********************************************************************;
* step5: delete records before the first discharge with one of
*        the disease flags, merge denominator variables
*********************************************************************;
      /** delete discharges that happend before the **/
      /** the first index discharge,                **/
      /** this is a VIEW step: it will run          **/
      /** simultaneously with the next step.        **/;

  DATA TEMP2/VIEW=TEMP2;
       SET WORK1.TEMP1;
       BY Y&YY.HICNO;
       LENGTH Y&YY.SFLAGS 2.;
       Y&YY.SFLAGS= SUM(Y&YY.HF,Y&YY.P1,Y&YY.P6,Y&YY.P7,
                        Y&YY.S1,Y&YY.S3,Y&YY.S4,Y&YY.M1);
       RETAIN ANYFLAG 0;
       IF FIRST.Y&YY.HICNO THEN ANYFLAG=0;

       ANYFLAG = MAX(ANYFLAG, Y&YY.SFLAGS);
```

```
       IF NOT ANYFLAG THEN DELETE;
  RUN;

      /** merge denominator vars  **/;

  DATA WORK2.TEMP2(DROP=ANYFLAG)
       ERR;
       MERGE TEMP2(IN=IN1)
             IN2.Y&YY.M5IN2(IN=IN2 KEEP=Y&YY.HICNO Y&YY.M5DEATH
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003) -- CONTINUED

                            Y&YY.ENDENR);
       BY Y&YY.HICNO;

       IF IN1 & NOT IN2 THEN OUTPUT ERR;
       ELSE
       IF IN1 & IN2 THEN OUTPUT WORK2.TEMP2;
 RUN;

 PROC PRINT U DATA=ERR(OBS=200);
      TITLE1 "NO RECORDS IN DENOMINATOR &YY";
      TITLE2 "ERROR!! SHOULD NOT HAPPEN";
 RUN;

*********************************************************************;
* step6 :  create index flags
*********************************************************************;

DATA OUT.Y&YY.M5IN3(KEEP=&CATVARS  &DEFLAGS
                   Y&YY.HICNO Y&YY.SADM Y&YY.SDIS Y&YY.ENDENR
                   Y&YY.M5DEATH Y&YY.SRDT
                   Y&YY.NOBS  Y&YY.PRVID
                   Y&YY.SFLAGS
                   Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                   Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                   Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                   Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                   Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                   Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA

                   LABEL="PROJECT: MQMS,PROGRAM: Y&YY.M5003"
                   SORTEDBY=Y&YY.HICNO Y&YY.SADM Y&YY.SDIS);

    SET WORK2.TEMP2;
    BY Y&YY.HICNO ;

     /** set arrays for each disease(cohort) to store  **/;
     /** input flags                                   **/;

 *cohort input 0/1 flags  ;
 ARRAY FL(&NUM_COH) Y&YY.HF Y&YY.P1 Y&YY.P6 Y&YY.P7
                      Y&YY.S1 Y&YY.S3 Y&YY.S4 Y&YY.M1;

     /** output variables **/;

 *length for index discharge flags for each window;
 LENGTH  Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
         Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
```

```
         Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
         Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
         Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
         Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA
      2.;

 *array for index discharge flags for each window;
 *w-for window (first index in array);
 *m-for measure (second index in array);
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5003) -- CONTINUED

                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

      /** temporary flags to keep in memory    **/
      /** for a particular HICNO               **/;

 *flag for each cohort to indicate if the first index disch is found;   ;
 RETAIN  FOUND1-FOUND&NUM_MEA 0;
 LENGTH  FOUND1-FOUND&NUM_MEA 2.;
 ARRAY   FOUND(&NUM_MEA) FOUND1-FOUND&NUM_MEA;

      /** if the first record for a person then zero out  **/
      /** all memory variables                            **/;

 IF FIRST.Y&YY.HICNO THEN DO;
    *set FOUND to 0 because none of the index records is found yet;
    DO J=1 TO &NUM_MEA;
       FOUND(J)=0;
    END;
 END; *first admission for HICNO;

      /** zero out output index flags **/;

 *set all index flags to 0 for this discharge;
 DO K=1 TO &NUM_WIN;
    DO J=1 TO &NUM_MEA;
      IWM(K,J) = 0;
    END;
 END;

      /** if this record is flagged at all then            **/
      /** call macro to set index in each relevant measure **/;

 IF Y&YY.SFLAGS THEN DO;
      %COH;
 END;
 *ELSE not index for any measure;

RUN;

*****************************************************************;
* step9: datacheck prinout and proc contents
*****************************************************************;
```

```
  PROC PRINT U DATA=OUT.Y&YY.M5IN3(OBS=46);
       TITLE1 "M5. THIRD INTERMEDIATE FILE";
       TITLE2 "INDEX FLAGS ARE CREATED";

  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5IN3;RUN;

                    /**MACRO ENDS**/

%MEND YXXM5003;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A)

```
    %MACRO YXXM503A(YVAR);

    %*************************************************************************
*****
    *************************************************************************
*****

    1  MACRO NAME:           YXXM503A
    2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A)
    3  WRITTEN BY:           HELEN MARGULIS
       REVISED BY:           ANN LARSEN
    4  COMPANY:              HEALTH ECONOMICS RESEARCH
    5  DATE:                 DECEMBER 2002
    6  PROJECT:              MQMS
    7  PURPOSE:  TO CREATE INDEX FLAGS IN ALL DISCHARGES FIRST YEAR FILE
                 FOR EACH MEASURE AND EACH WINDOW. DISENROLLMENT DATE IS
                 NOT TAKEN INTO CONSIDERATION

    8  STEPS     : 1. definitions

                   - CATVARS: tables category variables
                   - DEFLAGS: disease flags
                   - NUM_WIN: number of day windows
                   - NUM_MEA: number of measures
                   - NUM_COH: number of disease cohorts
                   - MSIZE: size of matrix to keep index variables
                   - $DISEASE: format to go from disease flag to disease
                               index
                   - DWINDOW: string containing number of days in each
                              window
                 2. define macro to set index discharges flags for each
                    cohort if the corresponding disease flag > 0. Cohorts
                    loop from 1 to NUM_COH. Measures loop from the first
                    measure to the last measure NUM_MEA
                 3. call macro YXXMEASR to define each measure by creating
                    macro variables DENOM, DFLAG, COND
                 4. sort &YY discharge file by hicno,admdate,disdate
                 5. delete records before the first discharge with one of
                    the disease flags. merge denominator variables
                 6. create index flags and create an output file
                   - set arrays for each disease(cohort) for input flags
                   - set output variables
                   - set temporary flags to keep in memory for a particular
                     HICNO
                   - for the first record for a person set to zero
                     memory variables
                   - put zeros in output index flags
                   - if this record is flagged with any flag then call
                     macro to set index in each relevant measure.
                 7. datacheck printout and proc contents
    *************************************************************************
****;
    *************************************************************************
****;

    OPTIONS BUFNO=5 ;
```

```
%INCLUDE IN0(YXXMEASR);

*********************************************************************;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A) -- CONTINUED
```
* step1: macro variables
*********************************************************************;

*year variables;
%let YY=%SUBSTR(&yvar,3,2);
%let YEAR=%SUBSTR(&yvar,1,4);

      /**category vars**/;

%LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
             Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

      /**disease flags**/;
%LET DEFLAGS =Y&YY.HF  Y&YY.M1 Y&YY.P1-Y&YY.P7
               Y&YY.S1-Y&YY.S4;

   /**number of diseases, windows, measures **/;

%LET NUM_WIN = 6;  *number of day windows;
%LET NUM_MEA = 34; *number of measures;
%LET NUM_COH = 8;  *number of disease cohorts;
%LET MSIZE   = %EVAL(&NUM_WIN*&NUM_MEA) ; *matrix size;

   /**number of days in windows **/;

%LET DWINDOW = %STR(2 7 30 180 365 730);  *days in each window;

   /** format to translate disease into **/
   /** variable index, list of cohorts   **/;

PROC FORMAT;
     VALUE $DISEASE
       'HF'= 1
       'P1'= 2
       'P6'= 3
       'P7'= 4
       'S1'= 5
       'S3'= 6
       'S4'= 7
       'M1'= 8  ;
RUN;

*********************************************************************;
* step2 : to set index discharges flags for all flagged cohorts
*********************************************************************;

%MACRO COH;
     /*********************************/
     /**  indexes:                   **/
     /**  J - count measures,        **/
     /**  I - count diseases(cohorts)**/
     /**  K - count windows          **/
     /*********************************/;
```

```
     %DO J=1 %TO &NUM_MEA;
        *find index I for the flag which define this measure;
        I = INPUT(PUT("&&DFLAG&J", $DISEASE.),2.);
        IF FL(I) THEN DO;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A) -- CONTINUED

```
          *if this is a BENE measure then temporary variable TINDEX
          *is equal 1 or 0 depending on FOUND flag for measure J,
          *for a discharge level measure TINDEX=1 ;
          IF "&&DENOM&J" = "BENE" THEN TINDEX=(NOT FOUND(&J));
          ELSE TINDEX=1;

          *turn "on" index flags for this measure for all windows;
          IF TINDEX THEN
             DO K=1 TO &NUM_WIN;
                IWM(K,&J)=TINDEX;
             END;

          *to turn "ON" FOUND flag for this measure: first index
           discharge for the bene is found for measure j;
          FOUND(&J)=1;
        END;    *end if this measure is FL(I) measure;
     %END;      *end of loop by measure j;

%MEND COH;
*********************************************************************;
* step3 : call macro to define each measure;
*********************************************************************;

 %YXXMEASR(&NUM_MEA,&NUM_COH);

*********************************************************************;
* step4: sort &YY discharge file by hicno,admdate,disdate
*********************************************************************;

  PROC SORT DATA=IN1.Y&YY.M5IN1 OUT=WORK1.TEMP1;
       BY Y&YY.HICNO Y&YY.SADM Y&YY.SDIS;
  RUN;

*********************************************************************;
* step5: delete records before the first discharge with one of
*        the disease flags, merge denominator variables
*********************************************************************;
      /** delete discharges that happend before the **/
      /** the first index discharge,                **/
      /** this is a VIEW step: it will run          **/
      /** simultaneously with the next step.        **/;

  DATA TEMP2/VIEW=TEMP2;
       SET WORK1.TEMP1;
       BY Y&YY.HICNO;
       LENGTH Y&YY.SFLAGS 2.;
       Y&YY.SFLAGS= SUM(Y&YY.HF,Y&YY.P1,Y&YY.P6,Y&YY.P7,
                        Y&YY.S1,Y&YY.S3,Y&YY.S4,Y&YY.M1);

        Y&YY.ENDENR="1JAN%EVAL(&YEAR+1)"D;
```

```
        Y&YY.M5DEATH=Y&YY.DETH;
        IF Y&YY.M5DEATH NE . & Y&YY.M5DEATH< Y&YY.ENDENR THEN
           Y&YY.ENDENR=Y&YY.M5DEATH;

       RETAIN ANYFLAG 0;
       IF FIRST.Y&YY.HICNO THEN ANYFLAG=0;

       ANYFLAG = MAX(ANYFLAG, Y&YY.SFLAGS);
       IF NOT ANYFLAG THEN DELETE;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A) -- CONTINUED

```
  RUN;

*******************************************************************;
* step6 :  create index flags
*******************************************************************;

DATA OUT.Y&YY.M5IN4(KEEP=&CATVARS  &DEFLAGS
                   Y&YY.HICNO Y&YY.SADM Y&YY.SDIS Y&YY.ENDENR
                   Y&YY.M5DEATH Y&YY.SRDT
                   Y&YY.NOBS  Y&YY.PRVID
                   Y&YY.SFLAGS
                   Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                   Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                   Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                   Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                   Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                   Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA
                   RENAME=(Y&YY.SADM=SADMDT
                           Y&YY.SDIS=SDISDT
                           Y&YY.PRVID=PROVID)
                   LABEL="PROJECT: MQMS,PROGRAM: Y&YY.M503A"
                   SORTEDBY=Y&YY.HICNO SADMDT SDISDT);

    SET TEMP2;
    BY Y&YY.HICNO ;

     /** set arrays for each disease(cohort) to store  **/;
     /** input flags                                   **/;

 *cohort input 0/1 flags  ;
 ARRAY FL(&NUM_COH) Y&YY.HF Y&YY.P1 Y&YY.P6 Y&YY.P7
                    Y&YY.S1 Y&YY.S3 Y&YY.S4 Y&YY.M1;

     /** output variables **/;

 *length for index discharge flags for each window;
 LENGTH  Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
         Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
         Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
         Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
         Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
         Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA
     2.;

 *array for index discharge flags for each window;
 *w-for window (first index in array);
 *m-for measure (second index in array);
```

```
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

      /** temporary flags to keep in memory    **/
      /** for a particular HICNO               **/;

 *flag for each cohort to indicate if the first index disch is found;   ;
 RETAIN  FOUND1-FOUND&NUM_MEA 0;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM503A) -- CONTINUED
```
 LENGTH  FOUND1-FOUND&NUM_MEA 2.;
 ARRAY   FOUND(&NUM_MEA) FOUND1-FOUND&NUM_MEA;

      /** if the first record for a person then zero out  **/
      /** all memory variables                            **/;

 IF FIRST.Y&YY.HICNO THEN DO;
    *set FOUND to 0 because none of the index records is found yet;
    DO J=1 TO &NUM_MEA;
       FOUND(J)=0;
    END;
 END; *first admission for HICNO;

      /** zero out output index flags **/;

 *set all index flags to 0 for this discharge;
 DO K=1 TO &NUM_WIN;
    DO J=1 TO &NUM_MEA;
      IWM(K,J) = 0;
    END;
 END;

      /** if this record is flagged at all then            **/
      /** call macro to set index in each relevant measure **/;

 IF Y&YY.SFLAGS THEN DO;
      %COH;
 END;
 *ELSE not index for any measure;

RUN;

*******************************************************************;
* step9: datacheck prinout and proc contents
*******************************************************************;
  PROC PRINT U DATA=OUT.Y&YY.M5IN4(OBS=46);
       TITLE1 "M5. THIRD INTERMEDIATE FILE";
       TITLE2 "INDEX FLAGS ARE CREATED";

  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5IN4;RUN;

                     /**MACRO ENDS**/
```

```
%MEND YXXM503A;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR)

```
%MACRO YXXMEASR( NUM_MEA, NUM_COH);

*********************************************************************

1  MACRO NAME:           YXXMEASR
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 JANUARY 30, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO WORK WITH THE TABLE OF MEASURES CREATING THE FOLLOWING
      CHARACTERISTICS FOR EACH OF 34 MEASURES (INDEX "J" IS USED
      IN ALL PROGRAMS TO LOOP THROUGH MESUARES) AND EACH OF 8
      COHORT DISEASES (INDEX "I" IS USED TO LOOP THROUGH DISEASES) :
       - DENOMj: to define whether this discharge will me counted in
                 denominator
         VALUES: ____BENE for beneficiary level measure
                 ____DISC for discharge level measure

       - DFLAGj: to define cohort for the measure
         VALUES: ____HF
                 ____P1
                 ____P6
                 ____P7
                 ____S1
                 ____S3
                 ____S4
                 ____M1

      - CONDj : to define condition for counting this discharge as a
                readmission
        VALUES:  ____ALL
                 ____SUM(Y&YY.HF)
                 ____SUM(OF Y&YY.P1-Y&YY.P7)
                 ____NOT SUM(OF Y&YY.P1-Y&YY.P7)
                 ____SUM(Y&YY.S2)
                 ____SUM(Y&YY.M1)



8 LOCAL MACROs:
      THIS MACRO HAS A LOCAL MACRO:
      macro1:  MVARS1 - to create measure specific macro variables
                        DENOM, DFLAG, COND.


9  PARAMETERS:

      NUM_MEA -number of measures
      NUM_COH -number of disease cohorts

10 COMMENT:

      YY: GLOBAL MACRO VARIABLE
```

11 EXAMPLE MACRO CALL:

D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR) -- CONTINUED

```
%YXXMEASR(NUM_MEA, NUM_COH)

**********************************************************************

**********************************************************************;
* GLOBAL PARAMETRS
**********************************************************************;

   %GLOBAL
     DENOM1 DENOM2 DENOM3 DENOM4 DENOM5 DENOM6 DENOM7 DENOM8
     DENOM9 DENOM10 DENOM11 DENOM12 DENOM13 DENOM14 DENOM15
     DENOM16 DENOM17 DENOM18 DENOM19 DENOM20 DENOM21 DENOM22
     DENOM23 DENOM24 DENOM25 DENOM26 DENOM27 DENOM28 DENOM29
     DENOM30 DENOM31 DENOM32 DENOM33 DENOM34

     DFLAG1 DFLAG2 DFLAG3 DFLAG4 DFLAG5 DFLAG6 DFLAG7 DFLAG8
     DFLAG9 DFLAG10 DFLAG11 DFLAG12 DFLAG13 DFLAG14 DFLAG15
     DFLAG16 DFLAG17 DFLAG18 DFLAG19 DFLAG20 DFLAG21 DFLAG22
     DFLAG23 DFLAG24 DFLAG25 DFLAG26 DFLAG27 DFLAG28 DFLAG29
     DFLAG30 DFLAG31 DFLAG32 DFLAG33 DFLAG34

     COND1 COND2 COND3 COND4 COND5 COND6 COND7 COND8
     COND9 COND10 COND11 COND12 COND13 COND14 COND15
     COND16 COND17 COND18 COND19 COND20 COND21 COND22
     COND23 COND24 COND25 COND26 COND27 COND28 COND29
     COND30 COND31 COND32 COND33 COND34

     ;


/*******************************************************************
*   measure description from documentation
*******************************************************************
Measure Denominator  Cohn Cohf  Discharge   Readmission
  "J"                "I"
   1   beneficiaries   1   HF    1st HF     all causes
   2   discharges      1   HF    each HF    all causes
   3   beneficiaries   1   HF    1st HF     HF
   4   discharges      1   HF    each HF    HF
   5   beneficiaries   2   P1    1st P1     all causes
   6   discharges      2   P1    each P1    all causes
   7   beneficiaries   2   P1    1st P1     pneumonia (P1 - P7)
   8   discharges      2   P1    each P1    pneumonia (P1 - P7)
   9   beneficiaries   2   P1    1st P1     not pneumonia (not P1-P7)
   10  discharges      2   P1    each P1    not pneumonia (not P1-P7)
   11  beneficiaries   3   P6    1st P6     all causes
   12  discharges      3   P6    each P6    all causes
   13  beneficiaries   3   P6    1st P6     pneumonia (P1 - P7)1
   14  discharges      3   P6    each P6    pneumonia (P1 - P7)1
   15  beneficiaries   3   P6    1st P6     not pneumonia (not P1-P7)
   16  discharges      3   P6    each P6    not pneumonia (not P1-P7)
   17  beneficiaries   4   P7    1st P7     all causes
   18  discharges      4   P7    each P7    all causes
```

```
   19  beneficiaries   4   P7    1st P7     pneumonia (P1 - P7)1
   20  discharges      4   P7    each P7    pneumonia (P1 - P7)1
   21  beneficiaries   4   P7    1st P7     not pneumonia (not P1-P7)
   22  discharges      4   P7    each P7    not pneumonia (not P1-P7)
   23  beneficiaries   5   S1    1st S1     all causes
   24  discharges      5   S1    each S1    all causes
D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR) -- CONTINUED

   25  beneficiaries   5   S1    1st S1     S2
   26  discharges      5   S1    each S1    S2
   27  beneficiaries   6   S3    1st S3     S2
   28  discharges      6   S3    each S3    S2
   29  beneficiaries   7   S4    1st S43    S2
   30  discharges      7   S4    each S43   S2
   31  beneficiaries   8   M1    1st M1     all causes
   32  discharges      8   M1    each M1    all causes
   33  beneficiaries   8   M1    1st M1     M1
   34  discharges      8   M1    each M1    M1
*****************************************************************/;

 *to create macro variables the table is transformed into the following
  macro string (blanks are substituted by dots in order substring strings
  needed to describe the measures);

%LET MEAS1  = %STR(BENE....HF....ALL );
%LET MEAS2  = %STR(DISC....HF....ALL );
%LET MEAS3  = %STR(BENE....HF....SUM(Y&YY.HF));
%LET MEAS4  = %STR(DISC....HF....SUM(Y&YY.HF));
%LET MEAS5  = %STR(BENE....P1....ALL);
%LET MEAS6  = %STR(DISC....P1....ALL);
%LET MEAS7  = %STR(BENE....P1....SUM(OF Y&YY.P1-Y&YY.P7));
%LET MEAS8  = %STR(DISC....P1....SUM(OF Y&YY.P1-Y&YY.P7));
%LET MEAS9  = %STR(BENE....P1....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS10 = %STR(DISC....P1....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS11 = %STR(BENE....P6....ALL );
%LET MEAS12 = %STR(DISC....P6....ALL );
%LET MEAS13 = %STR(BENE....P6....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS14 = %STR(DISC....P6....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS15 = %STR(BENE....P6....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS16 = %STR(DISC....P6....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS17 = %STR(BENE....P7....ALL );
%LET MEAS18 = %STR(DISC....P7....ALL );
%LET MEAS19 = %STR(BENE....P7....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS20 = %STR(DISC....P7....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS21 = %STR(BENE....P7....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS22 = %STR(DISC....P7....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS23 = %STR(BENE....S1....ALL );
%LET MEAS24 = %STR(DISC....S1....ALL );
%LET MEAS25 = %STR(BENE....S1....SUM(Y&YY.S2) );
%LET MEAS26 = %STR(DISC....S1....SUM(Y&YY.S2) );
%LET MEAS27 = %STR(BENE....S3....SUM(Y&YY.S2) );
%LET MEAS28 = %STR(DISC....S3....SUM(Y&YY.S2) );
%LET MEAS29 = %STR(BENE....S4....SUM(Y&YY.S2) );
%LET MEAS30 = %STR(DISC....S4....SUM(Y&YY.S2) );
%LET MEAS31 = %STR(BENE....M1....ALL );
%LET MEAS32 = %STR(DISC....M1....ALL );
%LET MEAS33 = %STR(BENE....M1....SUM(Y&YY.M1) );
%LET MEAS34 = %STR(DISC....M1....SUM(Y&YY.M1) );
```

```
*********************************************************************;
* macro1: to create DENOM, DFLAG, COND macro variables for each
*         measure
*********************************************************************;

%MACRO MVARS1;
      * first string for measure description:  ("&&MEAS&J",1)
      * second string for measure description: ("&&MEAS&J",2)
D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR) -- CONTINUED

      * third string for measure description:  ("&&MEAS&J",3);


     %DO J=1 %TO &NUM_MEA;
        CALL SYMPUT ("DENOM"||"&J",SCAN("&&MEAS&J",1,"...."));
        CALL SYMPUT ("DFLAG"||"&J",SCAN("&&MEAS&J",2,"...."));
        CALL SYMPUT ("COND"||"&J",SCAN("&&MEAS&J",3,"...."));
     %END;
%MEND MVARS1;


*********************************************************************;
* macros call
*********************************************************************;
DATA _NULL_;
  %MVARS1;
RUN;

*********************************************************************;
* data check print;
*********************************************************************;

%put denominator characteristics:;
%PUT &DENOM1 &DENOM2 &DENOM3 &DENOM4 &DENOM5 &DENOM6 &DENOM7 &DENOM8;
%PUT &DENOM9 &DENOM10 &DENOM11 &DENOM12 &DENOM13 &DENOM14 &DENOM15;
%PUT &DENOM16 &DENOM17 &DENOM18 &DENOM19 &DENOM20 &DENOM21 &DENOM22;
%PUT &DENOM23 &DENOM24 &DENOM25 &DENOM26 &DENOM27 &DENOM28 &DENOM29;
%PUT &DENOM30 &DENOM31 &DENOM32 &DENOM33 &DENOM34;

%put cohort flags:;
%PUT &DFLAG1 &DFLAG2 &DFLAG3 &DFLAG4 &DFLAG5 &DFLAG6 &DFLAG7 &DFLAG8;
%PUT &DFLAG9 &DFLAG10 &DFLAG11 &DFLAG12 &DFLAG13 &DFLAG14 &DFLAG15  ;
%PUT &DFLAG16 &DFLAG17 &DFLAG18 &DFLAG19 &DFLAG20 &DFLAG21 &DFLAG22 ;
%PUT &DFLAG23 &DFLAG24 &DFLAG25 &DFLAG26 &DFLAG27 &DFLAG28 &DFLAG29 ;
%PUT &DFLAG30 &DFLAG31 &DFLAG32 &DFLAG33 &DFLAG34;

%put readmission conditions:;
%PUT &COND1 &COND2 &COND3 &COND4 &COND5 &COND6 &COND7 &COND8;
%PUT &COND9 &COND10 &COND11 &COND12 &COND13 &COND14 &COND15 ;
%PUT &COND16 &COND17 &COND18 &COND19 &COND20 &COND21 &COND22;
%PUT &COND23 &COND24 &COND25 &COND26 &COND27 &COND28 &COND29;
%PUT &COND30 &COND31 &COND32 &COND33 &COND34;


%MEND YXXMEASR;
```

D#MSQ.@MSQ00M5.PGMLIB99(SYSM04)

```
       /**INCLUDE MACRO**/
%INCLUDE IN0(YXXM5004)/ SOURCE2;
%INCLUDE IN0(YXXFLAGS)/ SOURCE2;
       /**MACRO CALL**/
%YXXM5004(&SYSPARM);
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004)

```
%MACRO YXXM5004(YVAR);

%*************************************************************************
*****
*************************************************************************
*****

1  MACRO NAME:           YXXM5004
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 MARCH 2002
6  PROJECT:              MQMS
7  PURPOSE   :CREATE 2-YEAR ALL DISCHARGES FILE FOR PEOPLE WITH M5 "TARGET
              CONDITIONS" IN THE FIRST YEAR.
              SORT FILE BY HICNO, ADMISSION DATE, DISCHARGE DATE

8 STEPS     : 1. definitions

                 include the following external macros:
               - Y&YYDIAGS:  definition of 13 m4/m5 cohorts
               - YXXFLAGS:  create flag for each cohort

                 set the following macro parameters:
               - DEFLAGS: disease flags
               - CATVARS: tables category variables
               -  REQVAR  :  list  of  variales  to  take  from  M2  second  year
file
               - NUM_WIN: number of day windows
               - NUM_MEA: number of measures

              2. call macro to create macro variables in order to
                 define cohorts by diagnosis and surgical procedures

              3. delete records from cross-reference file if a person
                 has duplicate HICNO in YN denominator, sort by xref

              4. select discharges from the proximate year Medpar
                 using cross-reference file created in step2,
                 create "target conditions" flags for each discharge
                  - sort year 2 medpar by xref
                  - select discharges for M5 people
                  - create disease flags
                  - rename vars to be able to concatenate to the
                    base year discharge file
                  - sort by hicno, admission, discharge dates
                  - clear work1 library

              5. merge year 1 denominator variables

              6. concatenate 2 years discharge datasets by hicno
                 admission & discharge dates .
                 set all index flags to 0 for year 2 discharges.
                 Admission date, Dishcarge date and Provider variables
                 are renamed deleting a year in the beginning of the names
                 to indicate that these variables are from different years
```

of data.

7. datacheck prinout and proc contents

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004) -- CONTINUED

```
    ************************************************************************
****;
    ************************************************************************
****;

    OPTIONS BUFNO=5 ;

    ************************************************************************;
    * step1: macro variables
    ************************************************************************;

    *year variables;
    %let YY=%SUBSTR(&yvar,3,2);
    %let YN=%SUBSTR(&yvar,7,2);
    %let YEAR=%SUBSTR(&yvar,1,4);
    %let YEARN=%SUBSTR(&yvar,5,4);

          /**category vars**/;

    %LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
                 Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

          /**disease flags**/;
    %LET DEFLAGS =Y&YY.HF  Y&YY.M1 Y&YY.P1-Y&YY.P7
                   Y&YY.S1-Y&YY.S4;

    *other required variables;
    %LET REQVAR=Y&YN.HICNO  Y&YN.ADMIT   Y&YN.DISCH
                Y&YN.NOBS   Y&YN.PRVID
                Y&YN.DIA1-Y&YN.DIA10 Y&YN.SPR1-Y&YN.SPR6
                ;
       /**number of disease,windows measures **/;

    %LET NUM_WIN = 6;  *number of day windows;
    %LET NUM_MEA = 34; *number of measures;
    %LET NUM_COH = 8;  *number of disease cohorts;

    ************************************************************************;
    * step2: call macro to create macro variables in order to
    *        define cohorts by diagnosis and surgical procedures
    ************************************************************************;
          /**MACRO  CALL**/

    %INCLUDE INX(Y&YY.DIAGS)/ SOURCE2;
    %Y&YY.DIAGS;

    ************************************************************************;
    * step3: delete records from cross-reference file if a person
    *        has duplicate HICNO in YN denominator, sort by xref
    ************************************************************************;
      PROC SQL;
           CREATE TABLE WORKX.XR AS
```

```
        SELECT * FROM INXR.Y&YY.M5XRF
        WHERE Y&YY.HICNO NOT IN
              (SELECT DISTINCT Y&YY.HICNO FROM
               INDP.Y&YN.M5DUP)
        ORDER BY XREF;
  RUN;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004) -- CONTINUED

```
*******************************************************************;
* step4: select discharges from the proximate year Medpar
*        using cross-reference file created in step2,
*        create "target conditions" flags for each discharge
*******************************************************************;

      /** sort year 2 medpar by xref **/

  PROC SORT DATA=INY1.MEDPRG&YN(KEEP=&REQVAR RENAME=(Y&YN.HICNO=XREF))
           OUT=WORK1.TEMP1;
       BY XREF;
  RUN;

      /** select discharges for M5 people **/

  DATA WORK2.SELECT(KEEP=&DEFLAGS Y&YY.SFLAGS
                   Y&YY.HICNO Y&YN.SADM  Y&YN.SDIS
                   Y&YN.NOBS  Y&YN.PRVID);
       MERGE WORK1.TEMP1(IN=IN1)
             WORKX.XR(IN=IN2);

       BY XREF;
       IF IN1 & IN2;
       Y&YN.SDIS=DATEJUL(Y&YN.DISCH);
       Y&YN.SADM=DATEJUL(Y&YN.ADMIT);

      /** create disease flags **/;

       ARRAY D(10) Y&YN.DIA1-Y&YN.DIA10;
       %YXXFLAGS(&YN, Y&YY.HF, Y&YY.S, Y&YY.P, Y&YY.M);

      LENGTH Y&YY.SFLAGS 2.;
      Y&YY.SFLAGS= SUM(Y&YY.HF,Y&YY.P1,Y&YY.P6,Y&YY.P7,
                       Y&YY.S1,Y&YY.S3,Y&YY.S4,Y&YY.M1);
  RUN;

      /** sort by hicno, admission, discharge dates **/
  PROC SORT DATA=WORK2.SELECT;
       BY Y&YY.HICNO Y&YN.SADM Y&YN.SDIS;
  RUN;

      /** to clear work1 library **/
  PROC DATASETS NOLIST LIBRARY=WORK1;
       DELETE TEMP1;
  RUN;

*******************************************************************;
* step4: merge denominator variables
*******************************************************************;
```

```
  DATA WORK1.TEMP2
       ERR;
       MERGE WORK2.SELECT(IN=IN1)
             IN2.Y&YY.M5IN2(IN=IN2 KEEP=&CATVARS
                                Y&YY.HICNO Y&YY.M5DEATH
                                Y&YY.ENDENR);
       BY Y&YY.HICNO;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004) -- CONTINUED

```
       IF IN1 & NOT IN2 THEN OUTPUT ERR;
       ELSE
       IF IN1 & IN2 THEN OUTPUT WORK1.TEMP2;
 RUN;
 PROC PRINT U DATA=ERR(OBS=200);
      TITLE1 "NO RECORDS IN THE COMBINED DENOMINATOR FILE";
      TITLE2 "ERROR!! SHOULD NOT HAPPEN";
 RUN;

*****************************************************************;
* step6: concatenate 2 years discharge datasets by hicno,
*        admission & discharge dates
*****************************************************************;
  DATA OUT.Y&YY.M5IN4(DROP=K J
                      LABEL="MQMS PROGRAM: Y&YY.M5004"
                      SORTEDBY=Y&YY.HICNO SADMDT SDISDT);
       SET IN3.Y&YY.M5IN3(IN=IN&YY RENAME=(Y&YY.SADM=SADMDT
                                           Y&YY.SDIS=SDISDT
                                           Y&YY.PRVID=PROVID))
           WORK1.TEMP2(IN=INYN RENAME=(Y&YN.SADM=SADMDT
                                       Y&YN.SDIS=SDISDT
                                       Y&YN.PRVID=PROVID));
       BY Y&YY.HICNO SADMDT SDISDT;

 *array for index discharge flags for each window;
 *w-for window (first index in array);
 *m-for measure (second index in array);
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

       *set all index flags to 0 for year 2 discharges;
       *and set NOBS to 0 if record from another year;
       IF INYN THEN DO;
          DO K=1 TO &NUM_WIN;
             DO J=1 TO &NUM_MEA;
               IWM(K,J) = 0;
             END;
          END;
          Y&YY.NOBS=0;
       END;
       IF IN&YY THEN DO;
          Y&YN.NOBS=0;
       END;
```

```
  RUN;

*******************************************************************;
* step6: datacheck prinout and proc contents
*******************************************************************;
  PROC PRINT U DATA=OUT.Y&YY.M5IN4(OBS=46);
       TITLE1 "M5. FORTH INTERMEDIATE FILE";
       TITLE2 "TWO YEARS OF MEDPAR ARE CONCATENATED";

  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5IN4;RUN;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5004) -- CONTINUED

```
                    /**MACRO ENDS**/

%MEND YXXM5004;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A)

```
%MACRO YXXM504A(YVAR);

%*************************************************************************
*****
 *************************************************************************
*****

1  MACRO NAME:           YXXM504A
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A)
3  WRITTEN BY:           HELEN MARGULIS
   EDITED BY:            ANN LARSEN
4  COMPANY:              RTI
5  DATE:                 NOVEMBER 2002
6  PROJECT:              MQMS
7  PURPOSE   :CREATE 3-YEAR ALL DISCHARGES FILE FOR PEOPLE WITH M5 "TARGET
              CONDITIONS" IN THE FIRST YEAR.
              SORT FILE BY HICNO, ADMISSION DATE, DISCHARGE DATE

9 STEPS     : 1. definitions

                 include the following external macros:
               - Y&YBDIAGS:  definition of 13 m4/m5 cohorts
               - YXXFLAGS:  create flag for each cohort

                 set the following macro parameters:
               - DEFLAGS: disease flags
               - CATVARS: tables category variables
               -  REQVAR1:  list  of  variales  to  take  from  M2  second  year
file
               - REQVAR2: list of variales to take from M2 third year file
               - NUM_WIN: number of day windows
               - NUM_MEA: number of measures

              2. call macro to create macro variables in order to
                 define cohorts by diagnosis and surgical procedures

              3. delete records from cross-reference file if a person
                 has duplicate HICNO in Y1 denominator, sort by xref

              4. select discharges from the year 1 Medpar
                 using cross-reference file created in step2,
                 create "target conditions" flags for each discharge
                  - sort year 2 medpar by xref
                  - select discharges for M5 people
                  - create disease flags
                  - rename vars to be able to concatenate to the
                    base year discharge file
                  - sort by hicno, admission, discharge dates
                  - clear work1 library
              4a. merge year 1 denominator variables onto year 1 medpar

              5. repeat step 4 using year 2 medpar data
              5a. merge year 1 denominator variables onto year 2 medpar

              6. concatenate 3 years discharge datasets by hicno
                 admission & discharge dates .
```

```
                  set all index flags to 0 for year 2 discharges.
                  Admission date, Dishcarge date and Provider variables
                  are renamed deleting a year in the beginning of the names
```

```
                  to indicate that these variables are from different years
                  of data.

               7. datacheck prinout and proc contents

   ************************************************************************
****;
   ************************************************************************
****;

   OPTIONS BUFNO=5 ;

   ********************************************************************;
   * step1: macro variables
   ********************************************************************;

   *year variables;
   %let YB=%SUBSTR(&yvar,3,2);
   %let Y1=%SUBSTR(&yvar,7,2);
   %let Y2=%SUBSTR(&yvar,11,2);

         /**category vars**/;

   %LET CATVARS=Y&YB.CDIV Y&YB.CREG Y&YB.DUAL  Y&YB.FAGE Y&YB.MDST
                Y&YB.RACE Y&YB.SEX  Y&YB.STATE Y&YB.URBN;

         /**disease flags**/;
   %LET DEFLAGS =Y&YB.HF  Y&YB.M1 Y&YB.P1-Y&YB.P7
                  Y&YB.S1-Y&YB.S4;

   *other required variables;
   %LET REQVAR1=Y&Y1.HICNO  Y&Y1.ADMIT   Y&Y1.DISCH
               Y&Y1.NOBS   Y&Y1.PRVID
               Y&Y1.DIA1-Y&Y1.DIA10 Y&Y1.SPR1-Y&Y1.SPR6;
   %LET REQVAR2=Y&Y2.HICNO  Y&Y2.ADMIT   Y&Y2.DISCH
               Y&Y2.NOBS   Y&Y2.PRVID
               Y&Y2.DIA1-Y&Y2.DIA10 Y&Y2.SPR1-Y&Y2.SPR6;

      /**number of disease,windows measures **/;

   %LET NUM_WIN = 6;  *number of day windows;
   %LET NUM_MEA = 34; *number of measures;
   %LET NUM_COH = 8;  *number of disease cohorts;

   ********************************************************************;
   * step2: call macro to create macro variables in order to
   *        define cohorts by diagnosis and surgical procedures
   ********************************************************************;
         /**MACRO  CALL**/

   %INCLUDE IN0(YXXFLAGS)/ SOURCE2;
   %INCLUDE INX(Y&YB.DIAGS)/ SOURCE2;
   %Y&YB.DIAGS;
```

```
*******************************************************************;
* step3: delete records from cross-reference file if a person
*        has duplicate HICNO in Y1 denominator, sort by xref
*******************************************************************;
  PROC SQL;
D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A) -- CONTINUED

       CREATE TABLE WORKX.XR AS
         SELECT * FROM INXR.Y&YB.M5XRF
         WHERE Y&YB.HICNO NOT IN
               (SELECT DISTINCT Y&YB.HICNO FROM
                INDP.Y&Y1.M5DUP)
         ORDER BY XREF;
  RUN;

*******************************************************************;
* step4: select discharges from the year 2 Medpar
*        using cross-reference file created in step2,
*        create "target conditions" flags for each discharge
*******************************************************************;

      /** sort year 2 medpar by xref **/

  PROC SORT DATA=INY1.MEDPRG&Y1(KEEP=&REQVAR1 RENAME=(Y&Y1.HICNO=XREF))
            OUT=WORK1.TEMP1;
       BY XREF;
  RUN;

      /** select discharges for M5 people **/

  DATA WORK2.SELECT1(KEEP=&DEFLAGS Y&YB.SFLAGS
                  Y&YB.HICNO Y&Y1.SADM  Y&Y1.SDIS
                  Y&Y1.NOBS  Y&Y1.PRVID);
       MERGE WORK1.TEMP1(IN=IN1)
             WORKX.XR(IN=IN2);

       BY XREF;
       IF IN1 & IN2;
       Y&Y1.SDIS=DATEJUL(Y&Y1.DISCH);
       Y&Y1.SADM=DATEJUL(Y&Y1.ADMIT);

      /** create disease flags **/;

       ARRAY D(10) Y&Y1.DIA1-Y&Y1.DIA10;
       %YXXFLAGS(&Y1, Y&YB.HF, Y&YB.S, Y&YB.P, Y&YB.M);

      LENGTH Y&YB.SFLAGS 2.;
      Y&YB.SFLAGS= SUM(Y&YB.HF,Y&YB.P1,Y&YB.P6,Y&YB.P7,
                       Y&YB.S1,Y&YB.S3,Y&YB.S4,Y&YB.M1);
  RUN;

      /** sort by hicno, admission, discharge dates **/
  PROC SORT DATA=WORK2.SELECT1;
       BY Y&YB.HICNO Y&Y1.SADM Y&Y1.SDIS;
  RUN;

      /** to clear work1 library **/
```

```
  PROC DATASETS NOLIST LIBRARY=WORK1;
       DELETE TEMP1;
  RUN;

*********************************************************************;
* step4a: merge denominator variables
*********************************************************************;

  DATA WORK1.TEMP1
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A) -- CONTINUED
```
       ERR;
       MERGE WORK2.SELECT1(IN=IN1)
             IN2.Y&YB.M5IN2(IN=IN2 KEEP=&CATVARS
                                   Y&YB.HICNO Y&YB.M5DEATH
                                   Y&YB.ENDENR);
       BY Y&YB.HICNO;

       IF IN1 & NOT IN2 THEN OUTPUT ERR;
       ELSE
       IF IN1 & IN2 THEN OUTPUT WORK1.TEMP1;
 RUN;
 PROC PRINT U DATA=ERR(OBS=200);
      TITLE1 "NO RECORDS IN THE COMBINED DENOMINATOR FILE";
      TITLE2 "ERROR!! SHOULD NOT HAPPEN";
 RUN;

      /** to clear work2 library **/
  PROC DATASETS NOLIST LIBRARY=WORK2;
       DELETE SELECT1;
  RUN;
*********************************************************************;
* step5: select discharges from the year 3 Medpar
*        using cross-reference file created in step2,
*        create "target conditions" flags for each discharge
*********************************************************************;

      /** sort year 3 medpar by xref **/

  PROC SORT DATA=INY2.MEDPRG&Y2(KEEP=&REQVAR2 RENAME=(Y&Y2.HICNO=XREF))
            OUT=WORK2.TEMP2;
       BY XREF;
  RUN;

      /** select discharges for M5 people **/

  DATA WORK3.SELECT2(KEEP=&DEFLAGS Y&YB.SFLAGS
                    Y&YB.HICNO Y&Y2.SADM  Y&Y2.SDIS
                    Y&Y2.NOBS  Y&Y2.PRVID);
       MERGE WORK2.TEMP2(IN=IN1)
             WORKX.XR(IN=IN2);

       BY XREF;
       IF IN1 & IN2;
       Y&Y2.SDIS=DATEJUL(Y&Y2.DISCH);
       Y&Y2.SADM=DATEJUL(Y&Y2.ADMIT);

      /** create disease flags **/;
```

```
       ARRAY D(10) Y&Y2.DIA1-Y&Y2.DIA10;
       %YXXFLAGS(&Y2, Y&YB.HF, Y&YB.S, Y&YB.P, Y&YB.M);

      LENGTH Y&YB.SFLAGS 2.;
      Y&YB.SFLAGS= SUM(Y&YB.HF,Y&YB.P1,Y&YB.P6,Y&YB.P7,
                       Y&YB.S1,Y&YB.S3,Y&YB.S4,Y&YB.M1);
  RUN;

      /** sort by hicno, admission, discharge dates **/
  PROC SORT DATA=WORK3.SELECT2;
       BY Y&YB.HICNO Y&Y2.SADM Y&Y2.SDIS;
D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A) -- CONTINUED

  RUN;

      /** to clear work2 library **/
  PROC DATASETS NOLIST LIBRARY=WORK2;
       DELETE TEMP2;
  RUN;

*******************************************************************;
* step5a: merge denominator variables
*******************************************************************;

  DATA WORK2.TEMP2
       ERR;
       MERGE WORK3.SELECT2(IN=IN1)
             IN2.Y&YB.M5IN2(IN=IN2 KEEP=&CATVARS
                                Y&YB.HICNO Y&YB.M5DEATH
                                Y&YB.ENDENR);
       BY Y&YB.HICNO;

       IF IN1 & NOT IN2 THEN OUTPUT ERR;
       ELSE
       IF IN1 & IN2 THEN OUTPUT WORK2.TEMP2;
 RUN;
 PROC PRINT U DATA=ERR(OBS=200);
      TITLE1 "NO RECORDS IN THE COMBINED DENOMINATOR FILE";
      TITLE2 "ERROR!! SHOULD NOT HAPPEN";
 RUN;

      /** to clear work3 library **/
  PROC DATASETS NOLIST LIBRARY=WORK3;
       DELETE SELECT2;
  RUN;

*******************************************************************;
* step6: concatenate 2 years discharge datasets by hicno,
*        admission & discharge dates
*******************************************************************;
  DATA OUT.Y&YB.M5IN4(DROP=K J
                      LABEL="MQMS PROGRAM: Y&YB.M5004"
                      SORTEDBY=Y&YB.HICNO SADMDT SDISDT);
       SET IN3.Y&YB.M5IN3(IN=IN&YB RENAME=(Y&YB.SADM=SADMDT
                                           Y&YB.SDIS=SDISDT
                                           Y&YB.PRVID=PROVID))
           WORK1.TEMP1(IN=INY1 RENAME=(Y&Y1.SADM=SADMDT
```

```
                                        Y&Y1.SDIS=SDISDT
                                        Y&Y1.PRVID=PROVID))
          WORK2.TEMP2(IN=INY2 RENAME=(Y&Y2.SADM=SADMDT
                                        Y&Y2.SDIS=SDISDT
                                        Y&Y2.PRVID=PROVID));
      BY Y&YB.HICNO SADMDT SDISDT;

 *array for index discharge flags for each window;
 *w-for window (first index in array);
 *m-for measure (second index in array);
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YB.IW1M1-Y&YB.IW1M&NUM_MEA
                              Y&YB.IW2M1-Y&YB.IW2M&NUM_MEA
                              Y&YB.IW3M1-Y&YB.IW3M&NUM_MEA
                              Y&YB.IW4M1-Y&YB.IW4M&NUM_MEA
                              Y&YB.IW5M1-Y&YB.IW5M&NUM_MEA
D#MSQ.@MSQ00M5.PGMLIB99(YXXM504A) -- CONTINUED

                              Y&YB.IW6M1-Y&YB.IW6M&NUM_MEA;

      *set all index flags to 0 for year 2 & 3 Discharges;
      *and set NOBS to 0 if record from another year;
      IF (INY1 ³ INY2) THEN DO;
         DO K=1 TO &NUM_WIN;
            DO J=1 TO &NUM_MEA;
              IWM(K,J) = 0;
            END;
         END;
         Y&YB.NOBS=0;
      END;
      IF IN&YB THEN DO;
         Y&Y1.NOBS=0;
         Y&Y2.NOBS=0;
      END;
  RUN;

*******************************************************************;
* step6: datacheck prinout and proc contents
*******************************************************************;
  PROC PRINT U DATA=OUT.Y&YB.M5IN4(OBS=46);
       TITLE1 "M5. FORTH INTERMEDIATE FILE";
       TITLE2 "TWO YEARS OF MEDPAR ARE CONCATENATED";

  RUN;
  PROC CONTENTS DATA=OUT.Y&YB.M5IN4;RUN;

                    /**MACRO ENDS**/

%MEND YXXM504A;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM504B)

```
    %MACRO YXXM504B(YVAR);

    %*************************************************************************
*****
    *************************************************************************
*****

    1  MACRO NAME:           YXXM504B
    2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM504B)
    3  WRITTEN BY:           HELEN MARGULIS
    4  COMPANY:              RTI
    5  DATE:                 MARCH 2002
    6  PROJECT:              MQMS
    7  PURPOSE   :CREATE 2-YEAR ALL DISCHARGES FILE FOR PEOPLE WITH M5 "TARGET
                  CONDITIONS" IN THE FIRST YEAR.
                  SORT FILE BY HICNO, ADMISSION DATE, DISCHARGE DATE

    8 STEPS     : 1. definitions

                     include the following external macros:
                    - Y&YYDIAGS: definition of 13 m4/m5 cohorts
                    - YXXFLAGS:  create flag for each cohort

                     set the following macro parameters:
                    - DEFLAGS: disease flags
                    - CATVARS: tables category variables
                    -  REQVAR  :  list  of  variales  to  take  from  M2  second  year
file
                    - NUM_WIN: number of day windows
                    - NUM_MEA: number of measures

                  2. call macro to create macro variables in order to
                     define cohorts by diagnosis and surgical procedures

                  3. set all index flags to 0 for year 2 discharges.
                     Admission date, Dishcarge date and Provider variables
                     are renamed deleting a year in the beginning of the names
                     to indicate that these variables are from different years
                     of data.

                  7. datacheck prinout and proc contents

    *************************************************************************
****;
    *************************************************************************
****;

    OPTIONS BUFNO=5 ;

    ********************************************************************;
    * step1: macro variables
    ********************************************************************;

    *year variables;
    %let YY=%SUBSTR(&yvar,3,2);
```

```
%let YEAR=%SUBSTR(&yvar,1,4);

      /**category vars**/;
D#MSQ.@MSQ00M5.PGMLIB99(YXXM504B) -- CONTINUED


%LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
             Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

      /**disease flags**/;
%LET DEFLAGS =Y&YY.HF  Y&YY.M1 Y&YY.P1-Y&YY.P7
               Y&YY.S1-Y&YY.S4;

   /**number of disease,windows measures **/;

%LET NUM_WIN = 6;  *number of day windows;
%LET NUM_MEA = 34; *number of measures;
%LET NUM_COH = 8;  *number of disease cohorts;

*****************************************************************;
* step2: call macro to create macro variables in order to
*        define cohorts by diagnosis and surgical procedures
*****************************************************************;
      /**MACRO  CALL**/

%INCLUDE INX(Y&YY.DIAGS)/ SOURCE2;
%Y&YY.DIAGS;

  DATA OUT.Y&YY.M5IN4(DROP=K J
                      LABEL="MQMS PROGRAM: Y&YY.M504B"
                      SORTEDBY=Y&YY.HICNO SADMDT SDISDT);
       SET IN3.Y&YY.M5IN3(IN=IN&YY RENAME=(Y&YY.SADM=SADMDT
                                           Y&YY.SDIS=SDISDT
                                           Y&YY.PRVID=PROVID));
       BY Y&YY.HICNO SADMDT SDISDT;

       Y&YY.ENDENR="1JAN%EVAL(&YEAR+1)"D;
       Y&YY.M5DEATH=Y&YY.DETH;
       IF Y&YY.M5DEATH NE . & Y&YY.M5DEATH< Y&YY.ENDENR THEN
          Y&YY.ENDENR=Y&YY.M5DEATH;

 *array for index discharge flags for each window;
 *w-for window (first index in array);
 *m-for measure (second index in array);
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

  RUN;

*****************************************************************;
* step6: datacheck prinout and proc contents
*****************************************************************;
  PROC PRINT U DATA=OUT.Y&YY.M5IN4(OBS=46);
       TITLE1 "M5. FORTH INTERMEDIATE FILE";
```

```
      TITLE2 "TWO YEARS OF MEDPAR ARE CONCATENATED";

  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5IN4;RUN;

                    /**MACRO ENDS**/
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM504B) -- CONTINUED

```
%MEND YXXM504B;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXFLAGS)

```
%MACRO YXXFLAGS(YR, HFFLAG, ISFLAG, PNFLAG, MIFLAG);
*****************************************************************

1  MACRO NAME:           YXXFLAGS
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXFLAGS)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              HEALTH ECONOMICS RESEARCH
5  DATE:                 JANUARY 30, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      BASED ON DEFINITION OF COHORTS DONE IN YXXCOHRT MACRO
      TO CREATE 13 DISEASE FLAGS IN EACH DISCHARGE RECORD

      HF - heart failure
      S1 - Stroke or TIA
      S2 - Stroke only
      S3 - Atrial Fibrillation
      S4 - Carotid Endarterectomy
      P1 - viral pneumonia
      P2 - Pneumococcal Pneumonia
      P3 - bacterial Pneumonia
      P4 - Unspecified Pneumonia
      P5 - Influenza with Pneumonia
      P6 - principal diagnosis of septicemia and
           a secondary diagnosis of pneumonia
      P7 - principal diagnosis of respiratory failure and
           a secondary diagnosis of pneumonia.
      MI - acute myocardial infarction

9 PARAMETERS:
      Parameters difine only initial letters of the output flags
      YR     - 2 digit year of data
      HFFLAG - heart failure flag
      ISFLAG - stroke flags
      PNFLAG - Pneumonia flags
      MIFLAG - acute myocardial infarction flag

8 EXAMPLE MACRO CALL:
      %YXXFLAGS(&YY, Y&YY.HF, Y&YY.S, Y&YY.P, Y&YY.MI);

*********************************************************************

      *heart failure;
     &HFFLAG   = (D(1) IN &HF);

      *Stroke or TIA;
     &ISFLAG.1 = (D(1) IN &IS1);

      *Stroke only;
     &ISFLAG.2 = (D(1) IN &IS2);

      *Atrial Fibrillation;
     &ISFLAG.3=0;
     DO I=1 TO 10 WHILE(&ISFLAG.3 = 0);
        &ISFLAG.3 = (D(I) IN &IS3);
```

```
     END;

      *Carotid Endarterectomy;
D#MSQ.@MSQ00M5.PGMLIB99(YXXFLAGS) -- CONTINUED

     &ISFLAG.4 = (Y&YR.SPR1 = &IS4 OR Y&YR.SPR2 = &IS4 OR
                  Y&YR.SPR3 = &IS4 OR Y&YR.SPR4 = &IS4 OR
                  Y&YR.SPR5 = &IS4 OR Y&YR.SPR6 = &IS4 );

      *viral pneumonia;
     &PNFLAG.1 = (D(1) IN &PN1);

      *Pneumococcal Pneumonia;
     &PNFLAG.2 = (D(1) IN &PN2);

      *bacterial Pneumonia;
     &PNFLAG.3 = (D(1) IN &PN3);

      *Unspecified Pneumonia;
     &PNFLAG.4 = (D(1) IN &PN4);

      *Influenza with Pneumonia;
     &PNFLAG.5 = (D(1) IN &PN5);

      *principal diagnosis of septicemia and
       a secondary diagnosis of pneumonia;
     &PNFLAG.6 = ((D(1) IN &PN6_1) & (
       (D(2) IN &PN6_2) OR (D(3) IN &PN6_2) OR (D(4)  IN &PN6_2) OR
       (D(5) IN &PN6_2) OR (D(6) IN &PN6_2) OR (D(7)  IN &PN6_2) OR
       (D(8) IN &PN6_2) OR (D(9) IN &PN6_2) OR (D(10) IN &PN6_2)));

       *principal diagnosis of respiratory failure and
        a secondary diagnosis of pneumonia;
     &PNFLAG.7 = ( D(1) IN &PN7_1 & (
       (D(2) IN &PN7_2) OR (D(3) IN &PN7_2) OR (D(4)  IN &PN7_2) OR
       (D(5) IN &PN7_2) OR (D(6) IN &PN7_2) OR (D(7)  IN &PN7_2) OR
       (D(8) IN &PN7_2) OR (D(9) IN &PN7_2) OR (D(10) IN &PN7_2)));

       *acute myocardial infarction;
     &MIFLAG.1 = (D(1) IN &MI);

%MEND YXXFLAGS;
```

D#MSQ.@MSQ00M5.PGMLIB99(SYSM505)

```
       /**INCLUDE MACRO**/

%INCLUDE IN0(YXXMEASR)/ SOURCE2;      *define measures;
%INCLUDE IN0(YXXM5005)/ SOURCE2;     *program steps;

       /**MACRO CALL**/

%YXXM5005(&sysparm);
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005)

```
%MACRO YXXM5005(YVAR);

%*************************************************************************
*****
*************************************************************************
*****

1  MACRO NAME:           YXXM5005
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 JUNE 2002
6  PROJECT:              MQMS
7  PURPOSE:  CREATE M5 READMISSION FLAGS AND OUTPUT FINAL ANALYTICAL
             M5 FILE

8 STEPS     : 1. definitions

               - NUM_WIN: number of day windows
               - NUM_MEA: number of measures
               - NUM_COH: number of disease cohorts
               - MSIZE: size of matrix to keep index variables
               - DWINDOW: string containing number of days in each
                          window
               - $DISEASE: format to go from disease flag to disease
                           index

              2. define macro to put in memory date of this discharge
                 if this is an index discharge
              2. define macro to put in memory date of this discharge
                 if this is an index discharge
              3. define macro to fill out readmission array
              4. define macro to find the first window that includes
                 disenrollment date.
                 It should be done for an index discharge. Days are
                 counted from surgical date for S4 measures and from
                 discharge date for other measures.
              5. define macro to turn "off" indexes for windows that
                 include "end of enrollment" date and all subsequent
                 windows. It should be done only if a readmission
                 for a measure is not found for this index admission.
              5. call macro to define each measure.
              6. create readmission  flags:
                 - input variables:
                      array for input cohort flags
                      array for input index flags,
                 - output vaiables:
                      array for readmission flags
                      array for days interval,
                 - temporary variables to keep in memory for a hicno:
                      dates for previous index discharges to count
                      days to a readmission,
                      admission and discharge dates of recent index
                      discharge for each measure,
                 - set to 0 readmission flags and to -1 number of days
                   to a readmission,
```

```
                - if first record for a person then set to zero memory
                  vars and call INDFLAGS macro,
                - create readmission flags,
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

                - change days in memory if this is index discharge.
             7. work with disenrollment date:
                - sum up measure info on a discharge level,
                - turn "off" indexes for index discharges with no
                  readmission for windows on and after disenrollment
                  date.
             8. datacheck printout and proc contents
*******************************************************************;
OPTIONS BUFNO=5 ;

*******************************************************************;
* step1: definitions
*******************************************************************;

*year variables;
%let YY=%SUBSTR(&yvar,3,2);

   /** number of disease, windows, measures **/;

%LET NUM_WIN = 6;  *number of day windows;
%LET NUM_MEA = 34; *number of measures;
%LET NUM_COH = 8;  *number of disease cohorts;
%LET MSIZE   = %EVAL(&NUM_WIN*&NUM_MEA) ; *matrix size;


      /**category vars**/;

%LET CATVARS=Y&YY.CDIV Y&YY.CREG Y&YY.DUAL  Y&YY.FAGE Y&YY.MDST
             Y&YY.RACE Y&YY.SEX  Y&YY.STATE Y&YY.URBN;

      /**disease flags**/;
%LET DEFLAGS =Y&YY.HF  Y&YY.M1 Y&YY.P1-Y&YY.P7
               Y&YY.S1-Y&YY.S4;

     /** number of days in windows **/

%LET DWINDOW = %STR(2 7 30 180 365 730);  *days in each window;

     /** format to translate disease into **/
     /** variable index, list of cohorts **/;

PROC FORMAT;
     VALUE $DISEASE
       'HF'= 1
       'P1'= 2
       'P6'= 3
       'P7'= 4
       'S1'= 5
       'S3'= 6
       'S4'= 7
       'M1'= 8  ;

*******************************************************************;
```

```
* step2: macro to put in memory dates of this discharge if this is
         an index discharge
*********************************************************************;

%MACRO INDFLAGS;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

```
      /*********************************/;
      /** indexes:                     **/
      /**          J - count measures  **/
      /**          I - count diseases  **/
      /**          K - count windows   **/
      /*********************************/;

 ****for this discharge write dates into memory storage;
    %DO J=1 %TO &NUM_MEA;

       **** if this discharge is an index discharge for measure J
       **** then index flag=1 for the first window;

       IF Y&YY.IW1M&J THEN DO;

         ****for readmission remember day to count window from;
         IF "&&DFLAG&J" = "S4" THEN
             IF Y&YY.SRDT NE . THEN INDM&J=Y&YY.SRDT;
             ELSE INDM&J=SDISDT;
         ELSE INDM&J=SDISDT;

         ****for each measure only last index discharge is remembered;
         ADTS(&J) = SADMDT; *admission date;
         DDTS(&J) = SDISDT; *discharge date;

       END;  *this is an index discharge;
    %END;    *end of measure loop;
%MEND INDFLAGS;

*********************************************************************;
* step3: macro to fill out readmission array
*********************************************************************;

%MACRO READMFL;
  * if this discharge is not a readmission for a  particular measure
  * then interval is equal to -1;

  %DO J=1 %TO &NUM_MEA;

      **** if there was and index discharge for this measure
      **** then memory date for this measure > 0;
      IF INDM&J>0 THEN DO;
         IF SADMDT <= Y&YY.ENDENR THEN DO; *discharge before or on the;
                                           * date of disenrollment ;

           ****check readmission conditions;
           IF "&&COND&J"="ALL" THEN  WM(&J)=SADMDT - INDM(&J);
           ELSE
           IF &&COND&J THEN WM(&J)= SADMDT - INDM(&J);
           ELSE WM(&J)= -1;  *this discharge is not a readmission;
```

```
         END;
         ELSE WM(&J)= -1; *it means this discharge in not counted
                          *because it happens after disenrollment date;
      END;
      ELSE WM(&J)= -1; *there is no index discharge for measure J;

      ****it is a readmission for measure J;
      IF WM(&J) > 0 THEN DO;
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED
```

         ****readmission flags;
         %DO K=1 %TO &NUM_WIN;
             RWM(&K,&J)=( WM(&J) <= %EVAL(%SCAN(&DWINDOW,&K)));
         %END;     *window loop;

         *for work with end of enrollment;
         *if it is a readmission then the last window flag is =1;
         IF RWM(&NUM_WIN, &J) > 0 THEN DO;
            INDM&J=0; *readmission is found. Do not look for any more
                      readmissions for this index discharge;
            MEASURE=&J;  *measure number;
            ADT= ADTS(&J); *index discharge admission date;
            DDT= DDTS(&J); *index discharge discharge date;
            OUTPUT INDINFO;
         END;  *readmission found;
      END;

  %END;        *measure loop;
%MEND READMFL;

*********************************************************************;
* step4: to find window that includes disenrollment date.
*        It is done for index discharge. Days are counted from
*        discharge date for not S4 measures and from surgery date
*        for S4 measure.
*********************************************************************;
%MACRO MWINDOW;
    DFL = 0;
    *if there no surgical date SFL=.;
    IF Y&YY.SRDT NE . THEN SFL =0;

    %DO K=1 %TO &NUM_WIN ;

       WLENGTH=%EVAL(%SCAN(&DWINDOW,&K));

       IF DFL = 0 & WLENGTH > (Y&YY.ENDENR - SDISDT)
       THEN DFL = &K;

       IF Y&YY.SRDT NE . THEN
          IF SFL = 0 & WLENGTH > (Y&YY.ENDENR - Y&YY.SRDT)
          THEN SFL = &K;

    %END;

%MEND MWINDOW;
```

```
*******************************************************************;
* step5: macro to turn "off" indexes for windows that include and
*        after ENDENR date. It should be done only if readmission
*        for a  measure is not found  for this index discharge .
*******************************************************************;

%MACRO NULLWIN;

    %DO J=1 %TO &NUM_MEA;

       *if this is an index discharge for measure J then
       *index for the first window =1. MS(&J)=0 means that
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

       *a readmission is not found;
       IF Y&YY.IW1M&J & MS(&J)=0 THEN DO;

         IF "&&DFLAG&J" = "S4" THEN DO;
            *surgical date sould be present, if it is missing
             we take a discharge date;
            IF SFL = . THEN SFL=DFL;
            IF SFL > 0 THEN
              DO K=SFL TO &NUM_WIN;
                 IWM(K,&J)=0;          *S4 measure;
              END;
            *else window with disenrollment date is not found;
         END;
         ELSE DO;
            IF DFL > 0 THEN
              DO K=DFL TO &NUM_WIN;
                 IWM(K,&J)=0;          *not S4 measure;
              END;
            *else window with disenrollment date is not found;
         END;
       END;  *this is an index discharge for measure J;
    %END;    *end of measure loop;

%MEND NULLWIN;

*******************************************************************;
* step6 : call macro to define each measure;
*******************************************************************;

 %YXXMEASR(&NUM_MEA,&NUM_COH);

*******************************************************************;
* step7 :  create readmission  flags
*******************************************************************;


DATA /* output temp dataset with all the flags */
     WORKOUT.Y&YY.M5ANA(KEEP=&CATVARS &DEFLAGS
                    Y&YY.HICNO SADMDT SDISDT PROVID
                    Y&YY.M5DEATH Y&YY.SRDT
                    Y&YY.ENDENR Y&YY.NOBS
                    Y&YY.SFLAGS
                    Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                    Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
```

```
                    Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                    Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                    Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                    Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA

                    Y&YY.RW1M1-Y&YY.RW1M&NUM_MEA
                    Y&YY.RW2M1-Y&YY.RW2M&NUM_MEA
                    Y&YY.RW3M1-Y&YY.RW3M&NUM_MEA
                    Y&YY.RW4M1-Y&YY.RW4M&NUM_MEA
                    Y&YY.RW5M1-Y&YY.RW5M&NUM_MEA
                    Y&YY.RW6M1-Y&YY.RW6M&NUM_MEA

                    Y&YY.WM1-Y&YY.WM&NUM_MEA

                    LABEL="MQMS PROGRAM: Y&YY.M5005"
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED
```
                    SORTEDBY=Y&YY.HICNO SADMDT SDISDT)

     /* out temporary dataset to keep info about index discharges that
        have readmissions */
     INDINFO(KEEP=Y&YY.HICNO ADT DDT MEASURE)

     /* data check dataset: if not an index discharge for a bene
        is found before any index discharge then that record is
        is outputted in this dataset */
     NOT_IND;

     SET IN1.Y&YY.M5IN4;
     BY Y&YY.HICNO ;

      /***************************/
      /** input cohort flags    **/
      /***************************/;

 **** cohort input 0/1 flags  ;
 ARRAY FL(&NUM_COH) Y&YY.HF Y&YY.P1 Y&YY.P6 Y&YY.P7
                    Y&YY.S1 Y&YY.S3 Y&YY.S4 Y&YY.M1;

      /***************************/
      /** input index flags     **/
      /***************************/;
 **** array for input index discharge flags for each window;
 **** w-for window (first index in array);
 **** m-for measure (second index in array);
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

      /***************************/
      /**  output variables      **/
      /** -readmission flags     **/
      /** -interval in days      **/
      /***************************/;
```

```
****length;
LENGTH  Y&YY.RW1M1-Y&YY.RW1M&NUM_MEA
        Y&YY.RW2M1-Y&YY.RW2M&NUM_MEA
        Y&YY.RW3M1-Y&YY.RW3M&NUM_MEA
        Y&YY.RW4M1-Y&YY.RW4M&NUM_MEA
        Y&YY.RW5M1-Y&YY.RW5M&NUM_MEA
        Y&YY.RW6M1-Y&YY.RW6M&NUM_MEA
     2.
        Y&YY.WM1-Y&YY.WM&NUM_MEA
     3.;

**** readmission output flags;
ARRAY RWM(&NUM_WIN,&NUM_MEA) Y&YY.RW1M1-Y&YY.RW1M&NUM_MEA
                             Y&YY.RW2M1-Y&YY.RW2M&NUM_MEA
                             Y&YY.RW3M1-Y&YY.RW3M&NUM_MEA
                             Y&YY.RW4M1-Y&YY.RW4M&NUM_MEA
                             Y&YY.RW5M1-Y&YY.RW5M&NUM_MEA
                             Y&YY.RW6M1-Y&YY.RW6M&NUM_MEA;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

```
****interval days;
ARRAY WM(&NUM_MEA) Y&YY.WM1-Y&YY.WM&NUM_MEA;

**** output variables for INDINFO dataset;
LENGTH ADT DDT 8. MEASURE 2. ;

     /*****************************************************/;
     /** temporary variables to keep in memory           **/
     /** for a particular hicno.                         **/
     /*****************************************************/;

**** temporary variables to create readmission flags: a day for each
**** measure to count windows from.
**** Always remember last index discharge/surgery date for measure j,
**** it changes if a new index discharge found;

RETAIN INDM1-INDM&NUM_MEA 0;
ARRAY INDM(&NUM_MEA) INDM1-INDM&NUM_MEA;

**** temporary variables with index discharges info to
**** keep in memory to output into a temporary dataset
**** if this index discharge has a readmission. It is needed
**** to work with date of disenrollment;

RETAIN ADTS1-ADTS&NUM_MEA    /*admission dates */
       DDTS1-DDTS&NUM_MEA 0; /*discharge dates */
ARRAY ADTS(&NUM_MEA) ADTS1-ADTS&NUM_MEA;
ARRAY DDTS(&NUM_MEA) DDTS1-DDTS&NUM_MEA;

     /*******************************/
     /**  set to 0 readmission flags**/
     /**  for this discharge       **/
     /**  and set -1 for interval   **/
     /*******************************/;
DO JJ=1 TO &NUM_MEA;
   WM(JJ) =-1;
   DO KK=1 TO &NUM_WIN;
     RWM(KK,JJ) = 0;
```

```
   END;
END;

     /****************************************/
     /** if this record is the first record for**/
     /** a person then set to zero memory vars.**/
     /** Call INDFLAGS macro                  **/
     /****************************************/;

IF FIRST.Y&YY.HICNO THEN DO;

   **** set dates temporary to 0;
   DO JJ=1 TO &NUM_MEA;
      INDM(JJ) = 0;
      ADTS(JJ) = 0;
      DDTS(JJ) = 0;
   END;

   **** if this discharge is an index discharge;
   IF Y&YY.SFLAGS THEN DO; %INDFLAGS; END;
   ELSE OUTPUT NOT_IND; *first discharge has to be index;
```

```

END; *first discharge for HICNO;

ELSE DO;
     /*******************************/
     /** readmission flags         **/
     /*******************************/;

   **** is this discharge a readmission for any measure?;
   **** readmissions flags should be set before checking if;
   **** this discharge is an index discharge;
    %READMFL;

     /*********************************/
     /**  change days in memory if    **/
     /**  this is an index discharge  **/
     /*********************************/;
   IF Y&YY.SFLAGS THEN DO; %INDFLAGS; END;
END;

OUTPUT WORKOUT.Y&YY.M5ANA;

RUN;

*********************************************************************;
* step7: work with disenrollment date
*********************************************************************;

     /*************************************************************/;
     /** to sum up measure info on the discharge level:          **/
     /** one discharge could be an index discharge for more then **/
     /** one measure. In this case it was outputted more then    **/
     /** one time in the previous step.                          **/
     /*************************************************************/;
PROC REPORT DATA=INDINFO(OBS=300) HEADLINE HEADSKIP PANELS=2;
```

```
      COLUMNS Y&YY.HICNO ADT DDT MEASURE;
      DEFINE Y&YY.HICNO / DISPLAY "HICNO";
      DEFINE ADT / DISPLAY "ADM DATE" FORMAT=YYMMDD10.;
      DEFINE DDT / DISPLAY "DIS DATE" FORMAT=YYMMDD10.;
      DEFINE MEASURE / DISPLAY "MEASURE #";
      TITLE "DATA CHECK: INDEX RECORDS WITH A READMISSION";
 RUN;
 PROC SORT DATA=INDINFO;
      BY Y&YY.HICNO ADT DDT;
 RUN;
 DATA INDINFO1(KEEP=Y&YY.HICNO ADT DDT MS1-MS&NUM_MEA
               RENAME=(ADT=SADMDT DDT=SDISDT));
      SET INDINFO;
      BY Y&YY.HICNO ADT DDT;

      **** storage;
      LENGTH MS1-MS&NUM_MEA 2.;
      RETAIN MS1-MS&NUM_MEA 0;
      ARRAY MS(&NUM_MEA)  MS1-MS&NUM_MEA;

      **** to clear measure array;
      IF FIRST.DDT THEN
         DO J=1 TO &NUM_MEA;
            MS(J) = 0;
D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

         END;

      **** for this measure this discharge is an index discharge;
      MS(MEASURE)=1;

      IF LAST.DDT THEN OUTPUT;
 RUN;


      /*************************************************************/;
      /** for index discharges with no readmission turn "off"     **/
      /** index flags for windows which include end of enrollment **/;
DATA OUT.Y&YY.M5ANA(KEEP=&CATVARS &DEFLAGS
                   Y&YY.HICNO SADMDT SDISDT PROVID
                   Y&YY.M5DEATH Y&YY.SRDT
                   Y&YY.ENDENR Y&YY.NOBS
                   Y&YY.SFLAGS
                   Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                   Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                   Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                   Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                   Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                   Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA

                   Y&YY.RW1M1-Y&YY.RW1M&NUM_MEA
                   Y&YY.RW2M1-Y&YY.RW2M&NUM_MEA
                   Y&YY.RW3M1-Y&YY.RW3M&NUM_MEA
                   Y&YY.RW4M1-Y&YY.RW4M&NUM_MEA
                   Y&YY.RW5M1-Y&YY.RW5M&NUM_MEA
                   Y&YY.RW6M1-Y&YY.RW6M&NUM_MEA

                   Y&YY.WM1-Y&YY.WM&NUM_MEA
```

```
                  LABEL="MQMS PROGRAM: Y&YY.M5005"
                  SORTEDBY=Y&YY.HICNO SADMDT SDISDT);

      MERGE WORKOUT.Y&YY.M5ANA(IN=IN1)
            INDINFO1(IN=IN2);
      BY Y&YY.HICNO SADMDT SDISDT;

      **** measure indicators;
      ARRAY MS(&NUM_MEA)  MS1-MS&NUM_MEA;
      **** INDEX FLAGS;
 ARRAY IWM(&NUM_WIN,&NUM_MEA) Y&YY.IW1M1-Y&YY.IW1M&NUM_MEA
                              Y&YY.IW2M1-Y&YY.IW2M&NUM_MEA
                              Y&YY.IW3M1-Y&YY.IW3M&NUM_MEA
                              Y&YY.IW4M1-Y&YY.IW4M&NUM_MEA
                              Y&YY.IW5M1-Y&YY.IW5M&NUM_MEA
                              Y&YY.IW6M1-Y&YY.IW6M&NUM_MEA;

      **** this is an index discharge;
      IF Y&YY.SFLAGS THEN DO;

          **** no readmissions found;
          IF IN1 & NOT IN2 THEN
             DO J=1 TO &NUM_MEA;
                MS(J) = 0;
             END;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXM5005) -- CONTINUED

```
          **** to find the first window with disenrollment day;
          %MWINDOW;

          **** to turn "off" windows for measures without
          **** a readmission;
          %NULLWIN;
      END;
 RUN;

*******************************************************************;
* step8: datacheck printout and proc contents
*******************************************************************;
  PROC PRINT U DATA=NOT_IND(OBS=46);
       TITLE
       "!!!! WARNING !!!!!";
       TITLE2 "FIRST DISCHARGE FOR A HICNO IS NOT INDEX DISCHARGE.";
  RUN;

  PROC PRINT U DATA=OUT.Y&YY.M5ANA(OBS=46);
       TITLE1 "M5. FINAL ANALYTICAL FILE";
       TITLE2 "READMISSION FLAGS ARE CREATED";
       FORMAT Y&YY.ENDENR Y&YY.M5DEATH SADMDT SDISDT  YYMMDD10.;
  RUN;
  PROC CONTENTS DATA=OUT.Y&YY.M5ANA;RUN;

                    /**MACRO ENDS**/

%MEND YXXM5005;
```

D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASUR)

```
%MACRO YXXMEASR( NUM_MEA, NUM_COH);

*********************************************************************

1  MACRO NAME:           YXXMEASR
2  LOCATION:             D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASR)
3  WRITTEN BY:           HELEN MARGULIS
4  COMPANY:              RTI
5  DATE:                 JANUARY 30, 2002
6  PROJECT:              MQMS
7  PURPOSE:
      THE PURPOSE IS AS FOLLOWS:
      TO WORK WITH THE TABLE OF MEASURES CREATING THE FOLLOWING
      CHARACTERISTICS FOR EACH OF 34 MEASURES (INDEX "J" IS USED
      IN ALL PROGRAMS TO LOOP THROUGH MESUARES) AND EACH OF 8
      COHORT DISEASES (INDEX "I" IS USED TO LOOP THROUGH DISEASES) :
       - DENOMj: to define whether this discharge will me counted in
                 denominator
         VALUES: ____BENE for beneficiary level measure
                 ____DISC for discharge level measure

       - DFLAGj: to define cohort for the measure
         VALUES: ____HF
                 ____P1
                 ____P6
                 ____P7
                 ____S1
                 ____S3
                 ____S4
                 ____M1

      - CONDj : to define condition for counting this discharge as a
                readmission
        VALUES:  ____ALL
                 ____SUM(Y&YY.HF)
                 ____SUM(OF Y&YY.P1-Y&YY.P7)
                 ____NOT SUM(OF Y&YY.P1-Y&YY.P7)
                 ____SUM(Y&YY.S2)
                 ____SUM(Y&YY.M1)



8 LOCAL MACROs:
      THIS MACRO HAS A LOCAL MACRO:
      macro1:  MVARS1 - to create measure specific macro variables
                        DENOM, DFLAG, COND.


9  PARAMETERS:

      NUM_MEA -number of measures
      NUM_COH -number of disease cohorts

10 COMMENT:
```

YY: GLOBAL MACRO VARIABLE

11 EXAMPLE MACRO CALL:
D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASUR) -- CONTINUED

```
%YXXMEASR(NUM_MEA, NUM_COH)

**********************************************************************

**********************************************************************;
* GLOBAL PARAMETRS
**********************************************************************;

   %GLOBAL
     DENOM1 DENOM2 DENOM3 DENOM4 DENOM5 DENOM6 DENOM7 DENOM8
     DENOM9 DENOM10 DENOM11 DENOM12 DENOM13 DENOM14 DENOM15
     DENOM16 DENOM17 DENOM18 DENOM19 DENOM20 DENOM21 DENOM22
     DENOM23 DENOM24 DENOM25 DENOM26 DENOM27 DENOM28 DENOM29
     DENOM30 DENOM31 DENOM32 DENOM33 DENOM34

     DFLAG1 DFLAG2 DFLAG3 DFLAG4 DFLAG5 DFLAG6 DFLAG7 DFLAG8
     DFLAG9 DFLAG10 DFLAG11 DFLAG12 DFLAG13 DFLAG14 DFLAG15
     DFLAG16 DFLAG17 DFLAG18 DFLAG19 DFLAG20 DFLAG21 DFLAG22
     DFLAG23 DFLAG24 DFLAG25 DFLAG26 DFLAG27 DFLAG28 DFLAG29
     DFLAG30 DFLAG31 DFLAG32 DFLAG33 DFLAG34

     COND1 COND2 COND3 COND4 COND5 COND6 COND7 COND8
     COND9 COND10 COND11 COND12 COND13 COND14 COND15
     COND16 COND17 COND18 COND19 COND20 COND21 COND22
     COND23 COND24 COND25 COND26 COND27 COND28 COND29
     COND30 COND31 COND32 COND33 COND34

     ;


/*****************************************************************
*   measure description from documentation
*****************************************************************
Measure Denominator  Cohn Cohf  Discharge   Readmission
  "J"                "I"
   1   beneficiaries   1   HF    1st HF     all causes
   2   discharges      1   HF    each HF    all causes
   3   beneficiaries   1   HF    1st HF     HF
   4   discharges      1   HF    each HF    HF
   5   beneficiaries   2   P1    1st P1     all causes
   6   discharges      2   P1    each P1    all causes
   7   beneficiaries   2   P1    1st P1     pneumonia (P1 - P7)
   8   discharges      2   P1    each P1    pneumonia (P1 - P7)
   9   beneficiaries   2   P1    1st P1     not pneumonia (not P1-P7)
   10  discharges      2   P1    each P1    not pneumonia (not P1-P7)
   11  beneficiaries   3   P6    1st P6     all causes
   12  discharges      3   P6    each P6    all causes
   13  beneficiaries   3   P6    1st P6     pneumonia (P1 - P7)1
   14  discharges      3   P6    each P6    pneumonia (P1 - P7)1
   15  beneficiaries   3   P6    1st P6     not pneumonia (not P1-P7)
   16  discharges      3   P6    each P6    not pneumonia (not P1-P7)
   17  beneficiaries   4   P7    1st P7     all causes
```

```
   18  discharges      4   P7    each P7    all causes
   19  beneficiaries   4   P7    1st P7     pneumonia (P1 - P7)1
   20  discharges      4   P7    each P7    pneumonia (P1 - P7)1
   21  beneficiaries   4   P7    1st P7     not pneumonia (not P1-P7)
   22  discharges      4   P7    each P7    not pneumonia (not P1-P7)
   23  beneficiaries   5   S1    1st S1     all causes
```
D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASUR) -- CONTINUED

```
   24  discharges      5   S1    each S1    all causes
   25  beneficiaries   5   S1    1st S1     S2
   26  discharges      5   S1    each S1    S2
   27  beneficiaries   6   S3    1st S3     S2
   28  discharges      6   S3    each S3    S2
   29  beneficiaries   7   S4    1st S43    S2
   30  discharges      7   S4    each S43   S2
   31  beneficiaries   8   M1    1st M1     all causes
   32  discharges      8   M1    each M1    all causes
   33  beneficiaries   8   M1    1st M1     M1
   34  discharges      8   M1    each M1    M1
*****************************************************************/;

 *to create macro variables the table is transformed into the following
  macro string (blanks are substituted by dots in order substring strings
  needed to describe the measures);

%LET MEAS1  = %STR(BENE....HF....ALL );
%LET MEAS2  = %STR(DISC....HF....ALL );
%LET MEAS3  = %STR(BENE....HF....SUM(Y&YY.HF));
%LET MEAS4  = %STR(DISC....HF....SUM(Y&YY.HF));
%LET MEAS5  = %STR(BENE....P1....ALL);
%LET MEAS6  = %STR(DISC....P1....ALL);
%LET MEAS7  = %STR(BENE....P1....SUM(OF Y&YY.P1-Y&YY.P7));
%LET MEAS8  = %STR(DISC....P1....SUM(OF Y&YY.P1-Y&YY.P7));
%LET MEAS9  = %STR(BENE....P1....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS10 = %STR(DISC....P1....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS11 = %STR(BENE....P6....ALL );
%LET MEAS12 = %STR(DISC....P6....ALL );
%LET MEAS13 = %STR(BENE....P6....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS14 = %STR(DISC....P6....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS15 = %STR(BENE....P6....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS16 = %STR(DISC....P6....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS17 = %STR(BENE....P7....ALL );
%LET MEAS18 = %STR(DISC....P7....ALL );
%LET MEAS19 = %STR(BENE....P7....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS20 = %STR(DISC....P7....SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS21 = %STR(BENE....P7....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS22 = %STR(DISC....P7....NOT SUM(OF Y&YY.P1-Y&YY.P7) );
%LET MEAS23 = %STR(BENE....S1....ALL );
%LET MEAS24 = %STR(DISC....S1....ALL );
%LET MEAS25 = %STR(BENE....S1....SUM(Y&YY.S2) );
%LET MEAS26 = %STR(DISC....S1....SUM(Y&YY.S2) );
%LET MEAS27 = %STR(BENE....S3....SUM(Y&YY.S2) );
%LET MEAS28 = %STR(DISC....S3....SUM(Y&YY.S2) );
%LET MEAS29 = %STR(BENE....S4....SUM(Y&YY.S2) );
%LET MEAS30 = %STR(DISC....S4....SUM(Y&YY.S2) );
%LET MEAS31 = %STR(BENE....M1....ALL );
%LET MEAS32 = %STR(DISC....M1....ALL );
%LET MEAS33 = %STR(BENE....M1....SUM(Y&YY.M1) );
```

```
%LET MEAS34 = %STR(DISC....M1....SUM(Y&YY.M1) );

*********************************************************************;
* macro1: to create DENOM, DFLAG, COND macro variables for each
*         measure
*********************************************************************;

%MACRO MVARS1;
      * first string for measure description:  ("&&MEAS&J",1)
D#MSQ.@MSQ00M5.PGMLIB99(YXXMEASUR) -- CONTINUED

      * second string for measure description: ("&&MEAS&J",2)
      * third string for measure description:  ("&&MEAS&J",3);


     %DO J=1 %TO &NUM_MEA;
        CALL SYMPUT ("DENOM"||"&J",SCAN("&&MEAS&J",1,"...."));
        CALL SYMPUT ("DFLAG"||"&J",SCAN("&&MEAS&J",2,"...."));
        CALL SYMPUT ("COND"||"&J",SCAN("&&MEAS&J",3,"...."));
     %END;
%MEND MVARS1;


*********************************************************************;
* macros call
*********************************************************************;
DATA _NULL_;
  %MVARS1;
RUN;

*********************************************************************;
* data check print;
*********************************************************************;

%put denominator characteristics:;
%PUT &DENOM1 &DENOM2 &DENOM3 &DENOM4 &DENOM5 &DENOM6 &DENOM7 &DENOM8;
%PUT &DENOM9 &DENOM10 &DENOM11 &DENOM12 &DENOM13 &DENOM14 &DENOM15;
%PUT &DENOM16 &DENOM17 &DENOM18 &DENOM19 &DENOM20 &DENOM21 &DENOM22;
%PUT &DENOM23 &DENOM24 &DENOM25 &DENOM26 &DENOM27 &DENOM28 &DENOM29;
%PUT &DENOM30 &DENOM31 &DENOM32 &DENOM33 &DENOM34;

%put cohort flags:;
%PUT &DFLAG1 &DFLAG2 &DFLAG3 &DFLAG4 &DFLAG5 &DFLAG6 &DFLAG7 &DFLAG8;
%PUT &DFLAG9 &DFLAG10 &DFLAG11 &DFLAG12 &DFLAG13 &DFLAG14 &DFLAG15  ;
%PUT &DFLAG16 &DFLAG17 &DFLAG18 &DFLAG19 &DFLAG20 &DFLAG21 &DFLAG22 ;
%PUT &DFLAG23 &DFLAG24 &DFLAG25 &DFLAG26 &DFLAG27 &DFLAG28 &DFLAG29 ;
%PUT &DFLAG30 &DFLAG31 &DFLAG32 &DFLAG33 &DFLAG34;

%put readmission conditions:;
%PUT &COND1 &COND2 &COND3 &COND4 &COND5 &COND6 &COND7 &COND8;
%PUT &COND9 &COND10 &COND11 &COND12 &COND13 &COND14 &COND15 ;
%PUT &COND16 &COND17 &COND18 &COND19 &COND20 &COND21 &COND22;
%PUT &COND23 &COND24 &COND25 &COND26 &COND27 &COND28 &COND29;
%PUT &COND30 &COND31 &COND32 &COND33 &COND34;


%MEND YXXMEASR;
```

D#MSQ.@MSQ00M6.PGMLIB92(M6PROD92)

```
//TI10M692 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD92 EXEC M6PROD,YY=92
```

D#MSQ.@MSQ00M6.PGMLIB93(M6PROD93)

```
//TI10M693 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M5PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD93 EXEC M6PROD,YY=93
```

D#MSQ.@MSQ00M6.PGMLIB94(M6PROD94)

```
//TI10M694 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD94 EXEC M6PROD,YY=94
```

D#MSQ.@MSQ00M6.PGMLIB95(M6PROD95)

```
//TI10M695 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD95 EXEC M6PROD,YY=95
```

D#MSQ.@MSQ00M6.PGMLIB96(M6PROD96)

```
//TI10M696 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD96 EXEC M6PROD,YY=96
```

D#MSQ.@MSQ00M6.PGMLIB97(M6PROD97)

```
//TI10M697 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD97 EXEC M6PROD,YY=97
```

D#MSQ.@MSQ00M6.PGMLIB98(M6PROD98)

```
//TI10M698 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB99'
//M6PROD98 EXEC M6PROD,YY=98
```

D#MSQ.@MSQ00M6.PGMLIB99(M6PROD99)

D#MSQ.@MSQ00M6.PGMLIB00(M6PROD00)

```
//TI10M600 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB00'
//M6PROD00 EXEC M6PROD,YY=00
```

D#MSQ.@MSQ00M6.PGMLIB01(M6PROD01)

```
//TI10M601 JOB (MSQ00,CHER),TI10LARSEN,MSGCLASS=Q,CLASS=H
//*
//* PURPOSE:  THIS JOB EXECUTES THE CATALOGED PROCEDURE
//*           M6PROD FOR A PARTICULAR YEAR (E.G., M5PROD99 FOR 1999).
//*           TO RUN THIS JOB FOR OTHER YEARS, SIMPLY DEFINE THE
//*           YEAR OF ANALYSIS IN THE YEAR PARAMETER (2 DIGIT YEAR).
//*
//  JCLLIB ORDER='D#MSQ.@MSQ00M6.PGMLIB01'
//M6PROD01 EXEC M6PROD,YY=01
```

D#MSQ.@MSQ00M6.PGMLIB99(M6PROD)

```
//M6PROD PROC
//*********************************************************************
//* PURPOSE   :CREATE COHORTS AND OUTPUT A FILE FOR M6 TABLE CREATIONS
//*********************************************************************
//*
//YXXM6001 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YY'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M6.PGMLIB99
//IN DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M5005.Y&YY.M5ANA
//IN3 DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99
//OUT1 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.TEMP1,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//OUT2 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.TEMP2,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//OUT3 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.Y&YY.M6GRP,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//OUT4 DD DISP=(NEW,CATLG,KEEP),
//        DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.Y&YY.M6DEN,
//        LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M6.PGMLIB99(SYSM601),DISP=SHR
//DELSTEP EXEC PGM=IEFBR14,COND=(4,LT)
//D1 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.TEMP1,DISP=(OLD,DELETE)
//D2 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.TEMP2,DISP=(OLD,DELETE)
//*********************************************************************
//* PURPOSE   : SUM DISCHARGES TO CREATE DENOMINATOR VALUES
//*********************************************************************
//*
//YXXM6002 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YY'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M6.PGMLIB99
//IN  DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.Y&YY.M6DEN,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6002.TEMP1,
//       DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2002/365),UNIT=MCART
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6002.GRPWT&YY,
//       DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M6.PGMLIB99(SYSM602),DISP=SHR
//DELSTEP EXEC PGM=IEFBR14,COND=(4,LT)
//D1 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6002.TEMP1,DISP=(OLD,DELETE)
//*********************************************************************
//* PURPOSE: TO MERGE DENOMINATORS ONTO MEDPAR BY TYPE
//*********************************************************************
//*
//YXXM6003 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YY'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M6.PGMLIB99
//IN1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6002.GRPWT&YY,DISP=SHR
//IN2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6001.Y&YY.M6GRP,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6003.TEMP1,
//       DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2002/365),UNIT=MCART
//OUT DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6003.M6TAB&YY,
```

```
//       DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2007/365),UNIT=MCART
//SYSIN DD DSN=D#MSQ.@MSQ00M6.PGMLIB99(SYSM603),DISP=SHR
//DELSTEP EXEC PGM=IEFBR14,COND=(4,LT)
```
D#MSQ.@MSQ00M6.PGMLIB99(M6PROD) -- CONTINUED

```
//D1 DD  DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6003.TEMP1,DISP=(OLD,DELETE)
//*****************************************************************
//* PURPOSE: TO CREATE TABLES BY M6 MEASURE FOR ALL SUBGROUPS
//*****************************************************************
//*
//YXXM6004 EXEC SAS8,REGION=30M,PARM=('ERRORS=0 NEWS',
//      'SYSPARM=&YY'),COND=(4,LT)
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=D#MSQ.@MSQ00M6.PGMLIB99
//IN DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6003.M6TAB&YY,DISP=SHR
//OUT1 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTA1,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT2 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTB1,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT3 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTC1,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT4 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTA2,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT5 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTB2,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT6 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTC2,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT7 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTA3,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT8 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTB3,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//OUT9 DD DSN=D#MSQ.@MSQ0000.SAS.Y&YY.M6004.OUTPUTC3,
//        DISP=(NEW,CATLG,CATLG),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD DSN=D#MSQ.@MSQ00M6.PGMLIB99(SYSM604),DISP=SHR
//M6PROD PEND
//*************************************************************
```

D#MSQ.@MSQ00M6.PGMLIB99(SYSM601)

```
 %INCLUDE IN0(YXXM6001)/SOURCE2; * MACRO TO CREATE FILE FOR TABLES;
 %YXXM6001(&SYSPARM) *YEAR;
```

D#MSQ.@MSQ00M6.PGMLIB99(YXXM6001)

```
%MACRO YXXM6001(YY);

*******************************************************************
*
*  MACRO NAME:           YXXM6001
*  LOCATION:             D#MSQ.@MSQ00M6.PGMLIB99(YXXM6001)
*  WRITTEN BY:           ANN LARSEN
*  COMPANY:              RTI
*  INPUT:                D#MSQ.@MSQ0000.SAS.Y&YYM5005.Y&YYM5ANA
*                        D#MSQ.@MSQ0000.SAS.Y&YYM1004.AGESEX&YY
*  OUTPUT:               D#MSQ.@MSQ0000.SAS.Y&YYM6001.M6GRP&YY
                         D#MSQ.@MSQ0000.SAS.Y&YYM6001.M6DEN&YY
*  DATE:                 MAY 2002
*  PROJECT:              MQMS
*  PURPOSE:              TO CREATE MEASURE/WINDOW LEVEL FILE WITH
*                        NUMBER OF DAYS TO DEATH
*  COMMENTS  : XX - IS 2 DIGIT FOR A YEAR (FOR EXAMPLE 99).
*
*  STEPS     : 1. CREATE AGE & AGE/SEX WEIGHTS FROM DENOMINATOR
*              2. CREATE THE FOLLOWING VARIABLES:
*                 MEASURE - TYPE OF ADMISSION, HF P1 P6 P7 S1 S4 OR M1
*                           1=HF
                            2=P1
                            3=P6
                            4=P7
                            5=S1
                            6=S4
                            7=M1
                 dtod - days to death
              3. sort M5 analytical file by hicno, MEASURE, admission
                 date to calculate WINDOWs
              4. Create variables:
                 Y&YY.BENE=1 If it is an index admission, Else 0
                 WINDOW = 1 IF DIED DURING HOSPITALIZATION
                            2 IF DIED WITHIN 2 DAYS OF ADMISSION
                            3 IF DIED WITHIN 30 DAYS OF ADMISSION
                            4 IF DIED WITHIN 180 DAYS OF ADMISSION
                            5 IF DIED WITHIN 365 DAYS OF ADMISSION
                            6 IF DIED WITHIN 730 DAYS OF ADMISSION
              5. OUTPUT EACH ADMISSION/WINDOW TO A FILE
                 WITH AGE ADJUSTMENT
                 (AW) and Age/Sex adjustment (ASW) merged on.

*******************************************************************;
            /**CODE BEGINS**/
 *******************************************************************;
 * step1. create age & age/Sex & Sex weights from denominator
 *******************************************************************;

 DATA WGTA(KEEP=AWT1-AWT9 ASWT1-ASWT18 SWT1-SWT2);
     SET IN3.AGESEX99 END=XEND;
     RETAIN AWT1-AWT9 ASWT1-ASWT18 TOTAL99 SWT1 SWT2;
     ARRAY AW(9) AWT1-AWT9;
     ARRAY SW(2) SWT1-SWT2;
     ARRAY ASW(2,9) ASWT1-ASWT18;
     IF (Y99SEX=' ' & Y99FAGE=' ') THEN TOTAL99=Y99CNTS;
```

```
        ELSE IF (Y99SEX=' ' & Y99FAGE>' ') THEN
           AW(Y99FAGE)=Y99CNTS/TOTAL99;
        ELSE IF (Y99SEX>' ' & Y99FAGE>' ') THEN
D#MSQ.@MSQ00M6.PGMLIB99(YXXM6001) -- CONTINUED

           ASW(Y99SEX,Y99FAGE)=Y99CNTS/TOTAL99;
        ELSE IF (Y99SEX>' ' & Y99FAGE=' ') THEN
           SW(Y99SEX)=Y99CNTS/TOTAL99;
        IF (XEND) THEN OUTPUT;
    RUN;
    PROC PRINT DATA=WGTA;
         TITLE "1999 AGE WEIGHTS";
    RUN;

     %IF &YY>90 %THEN %LET YEAR=19&YY;
       %ELSE %LET YEAR=20&YY;

     %LET VARLIST=%STR(Y&YY.STATE Y&YY.CDIV Y&YY.CREG Y&YY.URBN Y&YY.SEX
       Y&YY.FAGE Y&YY.RACE Y&YY.DUAL Y&YY.MDST);
     %LET DXFLAGS=%STR(Y&YY.HF Y&YY.P1 Y&YY.P6 Y&YY.P7 Y&YY.S1 Y&YY.S4
       Y&YY.M1);

    ******************************************************************;
    * step 2. create the following vars:
    *         MEASURE - Type of admission, HF P1 P6 P7 S1 S4 or MI
    *         dtod - days to death
    *         died - 1 if Y&YY.BENE died during admission
    ******************************************************************;

    DATA OUT1.TEMP1(DROP=&DXFLAGS);
      SET IN.Y&YY.M5ANA(KEEP=Y&YY.HICNO SADMDT SDISDT Y&YY.ENDENR
       Y&YY.SRDT &VARLIST &DXFLAGS Y&YY.M5DEATH
       WHERE=(SDISDT<=MDY(12,31,&YEAR) & Y&YY.ENDENR>=SDISDT));
      FORMAT MEASURE $1.;
      MEASURE=' ';
      If (Y&YY.HF=1) THEN MEASURE='1';
      If (Y&YY.P1=1) THEN MEASURE='2';
      If (Y&YY.P6=1) THEN MEASURE='3';
      If (Y&YY.P7=1) THEN MEASURE='4';
      If (Y&YY.S1=1) THEN MEASURE='5';
      If (Y&YY.S4=1) THEN MEASURE='6';
      If (Y&YY.M1=1) THEN MEASURE='7';

      IF (Y&YY.M5DEATH>.) THEN DO;
          IF (MEASURE='6') THEN DTOD=Y&YY.M5DEATH-Y&YY.SRDT;
          ELSE DTOD=Y&YY.M5DEATH-SADMDT;
      END;

      DIED=(SADMDT<=Y&YY.M5DEATH<=SDISDT);
      If (MEASURE>' ') THEN OUTPUT OUT1.TEMP1;

    PROC PRINT DATA=OUT1.TEMP1(OBS=100);

    ******************************************************************;
    * step 3. sort file by hicno, MEASURE, admission
    *           to define index admission per Y&YY.BENE for each MEASURE of
admision
    ******************************************************************;
```

```
 PROC SORT DATA=OUT1.TEMP1 OUT=OUT2.TEMP2;
    BY Y&YY.HICNO MEASURE SADMDT;
 RUN;

 ******************************************************************;
```
D#MSQ.@MSQ00M6.PGMLIB99(YXXM6001) -- CONTINUED

```
 * step 4. Create variables:
 *      Y&YY.BENE=1 If it is an index admission, Else 0
 *      WINDOW=1 IF DIED DURING ADMISSION (per discharge)
 *      WINDOW=2 IF DIED WITHIN 2/7 DAYS OF ADMISSION (per bene)
 *      WINDOW=3 IF DIED WITHIN 30/15 DAYS OF ADMISSION (per bene)
 *      WINDOW=4 IF DIED WITHIN 180/30 DAYS OF ADMISSION (per bene)
 *      WINDOW=5 IF DIED WITHIN 365 DAYS OF ADMISSION (per bene)
 *      WINDOW=6 IF DIED WITHIN 730 DAYS OF ADMISSION (per bene)
 ******************************************************************;

 DATA OUT3.M6GRP&YY(KEEP=Y&YY.HICNO &VARLIST MEASURE WINDOW
    AGEADJ DENOM AGESEXADJ DTOD SEXADJ)
    OUT4.M6DEN&YY(KEEP=DENOM MEASURE &VARLIST);

      IF (_N_=1) THEN SET WGTA;

      SET OUT2.TEMP2; BY Y&YY.HICNO MEASURE;

      ARRAY AW(9) AWT1-AWT9;
      ARRAY SW(2) SWT1-SWT2;
      ARRAY ASW(2,9) ASWT1-ASWT18;
      AGEADJ=AW(Y&YY.FAGE);
      SEXADJ=SW(Y&YY.SEX);
      AGESEXADJ=ASW(Y&YY.SEX,Y&YY.FAGE);

      FORMAT DENOM $5.;
      DENOM='DISCH';
      OUTPUT OUT4.M6DEN&YY;
      IF (DIED=1) THEN DO;
         WINDOW=1; DENOM='DISCH'; OUTPUT OUT3.M6GRP&YY;
      END;
      IF (FIRST.MEASURE) THEN DO;
          DENOM='BENE';
          OUTPUT OUT4.M6DEN&YY;
          IF (MEASURE='1' ³ MEASURE='5' ³ MEASURE='6' ³ MEASURE='7')
             THEN DO;
             IF (.<DTOD<=2) THEN DO;
                WINDOW=2; OUTPUT OUT3.M6GRP&YY;
             END;
             IF (.<DTOD<=30) THEN DO;
               WINDOW=3; OUTPUT OUT3.M6GRP&YY;
             END;
             IF (.<DTOD<=180) THEN DO;
                WINDOW=4; OUTPUT OUT3.M6GRP&YY;
             END;
             IF (.<DTOD<=365) THEN DO;
                WINDOW=5; OUTPUT OUT3.M6GRP&YY;
             END;
             IF (.<DTOD<=730) THEN DO;
                WINDOW=6; OUTPUT OUT3.M6GRP&YY;
```

```
            END;
         END;
         ELSE IF (MEASURE='2' ³ MEASURE='3' ³ MEASURE='4') THEN DO;
            IF (.<DTOD<=7) THEN DO;
               WINDOW=2; OUTPUT OUT3.M6GRP&YY;
            END;
            IF (.<DTOD<=15) THEN DO;
               WINDOW=3; OUTPUT OUT3.M6GRP&YY;
            END;
```

```
            IF (.<DTOD<=30) THEN DO;
               WINDOW=4; OUTPUT OUT3.M6GRP&YY;
            END;
            IF (.<DTOD<=365) THEN DO;
               WINDOW=5; OUTPUT OUT3.M6GRP&YY;
            END;
            IF (.<DTOD<=730) THEN DO;
               WINDOW=6; OUTPUT OUT3.M6GRP&YY;
            END;
         END;
      END;

 ********************************************************************;
 * Step5. Output each admission to a file with age adjustment
 *     (AGEADJ) and Age/Sex adjustment (AGESEXADJ) merged on.
 ********************************************************************;
 RUN;

 PROC CONTENTS DATA=OUT3.M6GRP&YY; RUN;
 PROC PRINT DATA=OUT3.M6GRP&YY(OBS=100); RUN;

 PROC CONTENTS DATA=OUT4.M6DEN&YY(OBS=100); RUN;
 PROC PRINT DATA=OUT4.M6DEN&YY(OBS=100); RUN;

%MEND YXXM6001;
```

P1H4.@MSQ00AG.PGMLIB(Y99AGP)

```
//P1H4AG98 JOB (@MSQ00,PARK),P1H4PARK,MSGCLASS=Q,CLASS=K
//Y99AGP EXEC SAS8,REGION=8000K,OPTIONS='ERRORS=0,NEWS'
//WORK DD SPACE=(CYL,(2500,50))
//IN0  DD DISP=SHR,DSN=P1H4.@MSQ00AG.PGMLIB99
//IN DD DISP=SHR,DSN=D#MSQ.@MSQ0000.SAS.Y98M5005.Y98M5ANA
//OUT1 DD DISP=(NEW,CATLG,KEEP),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP01.TEMP1,UNIT=MCART
//OUT2 DD DISP=(NEW,CATLG,KEEP),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP01.TEMP2,UNIT=AFF=IN
//OUT3 DD DISP=(NEW,CATLG,KEEP),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP01,UNIT=AFF=OUT1
//OUT4 DD DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP02.TEMP1,UNIT=AFF=IN
//OUT5 DD DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP02.TEMP2,UNIT=AFF=OUT1
//OUT6 DD DISP=(NEW,CATLG,CATLG),LABEL=(EXPDT=2007/365),
//        DSN=P1H4.@MSQ0000.SAS.Y98AGP02,UNIT=AFF=IN
//SYSIN DD *,DLM=##
OPTIONS BUFNO=5;

 %INCLUDE IN0(YXXAGP01)/SOURCE2; * MACRO TO CREATE AGPREGATE FILE;
 %YXXAGP01(98);     *YEAR=98;
 %INCLUDE IN0(YXXAGP02)/SOURCE2; * MACRO TO CREATE AGPREGATE FILE;
 %YXXAGP02(98);     *YEAR=98;
```

P1H4.@MSQ00AG.PGMLIB(YXXAGP01)

```
                                                                        00010028
%MACRO YXXAGP01(YY);                                                    00020033
                                                                        00030028
*********************************************************************   00040028
                                                                        00050027
1  MACRO NAME:           YXXAGP01                                       00060033
2  LOCATION:             P1H4.@MSQ00AG.PGMLIB(YXXAGP01)                 00070033
3  WRITTEN BY:           ANN LARSEN (MODIFIED BY C. PARK)               00080032
4  COMPANY:              RTI                                            00090027
5  DATE:                 APRIL 2003                                     00100032
6  PROJECT:              MQMS                                           00110027
7  INPUT:                D#MSQ.@MSQ0000.SAS.YXXM5005.YXXM5ANA           00120027
8  OUTPUT:               P1H4.@MSQ0000.SAS.YXXAGP01                     00130033
                         XX - is 2 digit for a year (for example 99)    00140027
9  PURPOSE:              To create an AGGregate summary file for M5,    00150027
                         M6, and M8 BY PROVIDER STATE                   00160032
                         NUMBER OF DAYS TO READMISSION OR DEATH         00170027
                                                                        00180027
    STEP 1. Output bene level file with up to 99 admission trailers     00190027
    STEP 2. Output claim for every cohort it falls in.                  00200027
            Create the following vars:                                  00210027
            DAYS1A - days to cohort readmission                         00220027
            DAYS1B - days to non-cohort readmission                     00230027
            DAYS2 - days to death                                       00240027
            DAYS3A - days to death or cohort readmission whichever      00250027
                     is first                                           00260027
            DAYS3B - days to death or non-cohort readmission whichever  00270027
                     is first                                           00280027
            DAYS4A - days between cohort readmission and the            00290027
                     index discharge or death whichever comes first     00300027
            DAYS4B - days between non-cohort readmission and the        00310027
                     index discharge or death whichever comes first     00320027
            DAYS5 - days between index admission and disqualification   00330027
                     date                                               00340027
            DAYS6 = days between index admission and January 1 of base  00350027
                     year+3                                             00360027
            Days7 = days between index discharge and disqualification   00370027
                     date                                               00380027
            Days8 = days between index discharge and January 1 of base  00390027
                     year+3                                             00400027
            NEXTADM - admission date of next non-cohort readmission     00410027
            NEXTDIS - discharge date of next non-cohort readmission     00420027
            COHADM - admission date of next cohort readmission          00430027
            COHDIS - discharge date of next cohort readmission          00440027
            NEXTCOH - cohort of next cohort readmission                 00450027
                                                                        00460027
    STEP 3. Sort file by COHORT Y&YY.HICNO SADMDT SDISDT                00470027
            to identify the first admission in the cohort for           00480027
            bene level sums.                                            00490027
    STEP 4. Create the variable BENE=1 if first admission in cohort     00500027
*********************************************************************; 00510027
                                                                        00530007
  %IF &YY>90 %THEN %LET YEAR=19&YY;                                     00540007
    %ELSE %LET YEAR=20&YY;                                              00550007
                                                                        00560007
  %LET VARLIST=%STR(Y&YY.PST Y&YY.PCDIV Y&YY.PREG Y&YY.URBN Y&YY.SEX    00570033
```

```
    Y&YY.FAGE Y&YY.RACE Y&YY.DUAL Y&YY.MDST);                              00580007
                                                                          00590007
  %LET FLAGLIST=%STR(Y&YY.HF Y&YY.P1-Y&YY.P7 Y&YY.S1-Y&YY.S4 Y&YY.M1);  00600007
```
P1H4.@MSQ00AG.PGMLIB(YXXAGP01) -- CONTINUED

```
                                                                          00610007
 ************************************************************************; 00620007
 *  STEP 1. Output bene level file with up to 99 admission trailers     00630007
 ************************************************************************; 00640007
                                                                          00650007
  DATA OUT1.TEMP1(KEEP=&VARLIST COHORT NEXTADM NEXTDIS NEXTCOH            00660007
   Y&YY.HICNO SADMDT SDISDT SRGDT Y&YY.M5DEATH Y&YY.ENDENR                00670007
   DAYS5-DAYS8 COHADM COHDIS DAYS1A DAYS1B DAYS2 DAYS3A DAYS3B            00680007
   DAYS4A DAYS4B);                                                        00690007
  SET IN.Y&YY.M5ANA(KEEP=Y&YY.HICNO SADMDT SDISDT PROVID                  00700032
  Y&YY.SRDT &FLAGLIST Y&YY.M5DEATH Y&YY.ENDENR                            00710034
  Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL Y&YY.MDST Y&YY.URBN              00711034
  WHERE=(Y&YY.ENDENR>=SDISDT)); BY Y&YY.HICNO;                            00720007
                                                                          00731632
ATTRIB Y&YY.PST LENGTH=$2.;                                               00731732
ATTRIB Y&YY.PCDIV LENGTH=$1.;                                             00731832
ATTRIB Y&YY.PREG LENGTH=$1.;                                              00731932
                                                                          00732032
 Y&YY.PST = SUBSTR(PROVID,1,2);     *ACCORDING TO CMS 3/2001 DIRECTIVE; 00732132
 IF Y&YY.PST = '55' THEN Y&YY.PST='05';      *CA has 2 codes;            00732232
 ELSE IF Y&YY.PST = '65' THEN Y&YY.PST='12'; *SETING GUAM TO HAWAII;     00732332
 ELSE IF Y&YY.PST = '67' THEN Y&YY.PST='45'; *TX has 2 codes;            00732432
 ELSE IF Y&YY.PST = '68' THEN Y&YY.PST='10'; *FL has 2 codes;            00732532
 ELSE IF Y&YY.PST = '80' THEN Y&YY.PST='21'; *MD has 2 codes;            00732632
                                                                          00732732
IF Y&YY.PST IN (40,48) OR Y&YY.PST GT 53 THEN DO;                         00732832
     Y&YY.PCDIV=0;  Y&YY.PREG=0;  END;   * PR, VI, & OTHER ARE SET TO 0;00732932
  else IF Y&YY.PST IN (07,20,22,30,41,47) THEN do;                        00733032
     Y&YY.PCDIV=1;  y&yy.preg=1;  end;                                    00733132
  else IF Y&YY.PST IN (31,33,39) THEN do;                                 00733232
     Y&YY.PCDIV=2;  y&yy.preg=1;  end;                                    00733332
  else IF Y&YY.PST IN (14,15,23,36,52) THEN do;                           00733432
     Y&YY.PCDIV=3;  y&yy.preg=2;  end;                                    00733532
  else IF Y&YY.PST IN (16,17,24,26,28,35,43) THEN do;                     00733632
     Y&YY.PCDIV=4;  y&yy.preg=2;  end;                                    00733732
  else IF Y&YY.PST IN (08,09,10,11,21,34,42,49,51) THEN do;               00733832
     Y&YY.PCDIV=5;  y&yy.preg=3;  end;                                    00733932
  else IF Y&YY.PST IN (01,18,25,44) THEN do;                              00734032
     Y&YY.PCDIV=6;  y&yy.preg=3;  end;                                    00734132
  else IF Y&YY.PST IN (04,19,37,45) THEN do;                              00734232
     Y&YY.PCDIV=7;  y&yy.preg=3;  end;                                    00734332
  else IF Y&YY.PST IN (03,06,13,27,29,32,46,53) THEN do;                  00734432
     Y&YY.PCDIV=8;  y&yy.preg=4;  end;                                    00734532
  else IF Y&YY.PST IN (02,05,12,38,50) THEN do;                           00734632
     Y&YY.PCDIV=9;  y&yy.preg=4;  end;                                    00734732
                                                                          00736032
* Define admission variables allow for only 99 admissions per bene;      00740032
  ARRAY ADM(99) ADM01-ADM99;                                              00750007
  ARRAY DIS(99) DIS01-DIS99;                                              00760007
  ARRAY SRG(99) SRG01-SRG99;                                              00770007
  ARRAY COH(99) $ 2 COH01-COH99;                                          00780007
                                                                          00790007
  RETAIN CNT ADM01-ADM99 DIS01-DIS99 COH01-COH99 SRG01-SRG99;             00800009
```

```
  FORMAT COHORT $2.;                                                    00810007
                                                                        00820007
  *Initialize arrays to 0 for new bene;                                 00830007
  IF (FIRST.Y&YY.HICNO) THEN DO;                                        00840007
     CNT=0;                                                             00850007
     DO I=1 TO 99;                                                      00860007
P1H4.@MSQ00AG.PGMLIB(YXXAGP01) -- CONTINUED

        ADM(I)=.;  DIS(I)=.;  COH(I)='  '; SRG(I)=.;                    00870007
     END;                                                               00880007
  END;                                                                  00890007
                                                                        00900007
  CNT=SUM(CNT,1);                                                       00910007
  IF (CNT>90) THEN CNT=90;                                              00920007
                                                                        00930007
  ARRAY FLG(13) &FLAGLIST;                                              00940007
  ARRAY LBL(13) $ 2 ('HF','P1','P2','P3','P4','P5','P6','P7',          00950007
                     'S1','S2','S3','S4','M1');                        00960007
  CHECK=0;                                                              00970007
  * Check each cohort flag and create admission trailer for each cohort 00980007
    this admission falls in;                                            00990007
  DO I=1 TO 13;                                                         01000007
     IF (FLG(I)=1) THEN DO;                                             01010007
        CNT=SUM(CNT,CHECK);                                             01020007
        COH(CNT)=LBL(I);                                                01030007
        ADM(CNT)=SADMDT;                                                01040007
        DIS(CNT)=SDISDT;                                                01050007
        SRG(CNT)=Y&YY.SRDT;                                             01060007
        CHECK=1;                                                        01070007
      END;                                                              01080007
  END;                                                                  01090007
  * If not cohort admission then output admission trailer and set       01100007
    COH=XX;                                                             01110008
  IF (SUM(OF &FLAGLIST)=0) THEN DO;                                     01120007
      COH(CNT)="XX";                                                    01130007
      ADM(CNT)=SADMDT;                                                  01140007
      DIS(CNT)=SDISDT;                                                  01150007
      SRG(CNT)=Y&YY.SRDT;                                               01160007
  END;                                                                  01170007
                                                                        01180007
  *If last admission for this bene, then start checking for index       01190007
   admissions;                                                          01200007
  IF (LAST.Y&YY.HICNO) THEN DO;                                         01210007
      ********************************************************;        01220007
      *  STEP 2. Output claim for every cohort it falls in.  *;        01230007
      ********************************************************;        01240007
       DO I=1 TO CNT;                                                   01250007
          * If this is a cohort admission and the discharge date       01260007
            is in the base year then this is an index admission;       01270007
          IF (COH(I) in ('HF','S1','S3','S4','P1','P6','P7','M1') &    01280007
             DIS(I)<=MDY(12,31,&YEAR)) THEN DO;                         01290007
             COHORT=COH(I);                                             01300007
             SADMDT=ADM(I);                                             01310007
             SDISDT=DIS(I);                                             01320007
             SRGDT=SRG(I);                                              01330007
         * Create DAYS2 = date of death - index admission date;         01340007
           IF (Y&YY.M5DEATH>. & Y&YY.M5DEATH>=SRG(I) & COH(I)='S4' &    01350007
               SRG(I)>0)                                                01360007
```

```
            THEN DAYS2=Y&YY.M5DEATH-SRG(I);                              01370007
            ELSE IF (Y&YY.M5DEATH>. & Y&YY.M5DEATH>=ADM(I))              01380008
            THEN DAYS2=Y&YY.M5DEATH-ADM(I);                              01390007
            Else DAYS2=.;                                                01400007
                                                                         01410007
        * Days5 = days between index admission and disqualification      01420007
          date;                                                          01430007
          DAYS5=Y&YY.ENDENR-ADM(I);                                      01440007
                                                                         01450007
```

P1H4.@MSQ00AG.PGMLIB(YXXAGP01) -- CONTINUED

```
        * Days6 = days between index admission and January 1 of base     01460007
          year+3;                                                        01470007
          TP=&YEAR+3;                                                    01480007
          DAYS6=MDY(01,01,TP)-ADM(I);                                    01490007
                                                                         01500007
        * Days7 = days between index discharge and disqualification      01510007
          date;                                                          01520007
          IF (COH(I)='S4' & SRG(I)>0) THEN DAYS7=Y&YY.ENDENR-SRG(I);     01530007
             ELSE DAYS7=Y&YY.ENDENR-DIS(I);                              01540007
                                                                         01550007
        * Days8 = days between index discharge and January 1 of base     01560007
          year+3;                                                        01570007
          DAYS8=MDY(01,01,TP)-SDISDT;                                    01580007
                                                                         01590007
         * Initialize readmission variables to missing;                 01600007
          DAYS1A=.; NEXTADM=.; NEXTDIS=.; NEXTCOH='  ';                  01610007
          DAYS3A=.; DAYS4A=.; COHADM=.; COHDIS=.; DAYS1B=.; DAYS3B=.;    01620007
          DAYS4B=.;                                                      01630007
             *Check for readmission;                                     01640007
             IF (CNT>1 & I<CNT) THEN DO;                                 01650007
                FOUND1=0; FOUND2=0;                                      01660007
                *Check for cohort readmission;                           01670007
                DO J=(I+1) TO CNT UNTIL(FOUND1>0 & FOUND2>0);            01680007
                   SKIP=0; CTOC=0;                                       01690007
                   *If admission is written multiple times for several   01700007
                    cohorts, do not count as a readmission;              01710007
                   IF ((ADM(I)=ADM(J) & DIS(I)=DIS(J)) ³                 01720007
                      (ADM(J)<DIS(I))) THEN SKIP=1;                      01730011
                   * Check if readmission is a cohort readmission;       01740008
                   IF ((COH(I)='HF' & COH(J)='HF') ³                     01750007
                   ((COH(I) IN ('P1','P6','P7')) & 'P1'<=COH(J)<='P7') ³01760007
                   ((COH(I) IN ('S1','S3','S4')) & COH(J)='S2') ³        01770007
                   (COH(I)='M1' & COH(J)='M1')) THEN CTOC=1;             01780007
                   IF (CTOC=1 & SKIP=0 & FOUND1=0)                       01790007
                   THEN DO;                                              01800007
                      FOUND1=1;                                          01810007
                      * Create days1a = next admission-index discharge   01820007
                        for cohort readmission;                          01830007
                      IF (COH(I)='S4' & SRG(I)>0 &                       01840007
                         (ADM(J)>=DIS(I) & ADM(J)<=Y&YY.ENDENR))         01850007
                         THEN DAYS1A=ADM(J)-SRG(I);                      01860007
                        ELSE IF (ADM(J)>=DIS(I) & ADM(J)<=Y&YY.ENDENR)   01870007
                         THEN DAYS1A=ADM(J)-DIS(I);                      01880007
                      NEXTCOH=COH(J);                                    01890007
                      COHADM=ADM(J);                                     01900007
                      COHDIS=DIS(J);                                     01910007
                   END;                                                  01920007
```

```
                  *Not a cohort readmission;                                01930007
                  ELSE IF (CTOC=0 & SKIP=0 & FOUND2=0) THEN DO;             01940007
                     * Create days1b = next admission-index discharge       01950007
                       for non-cohort readmission;                          01960007
                     IF (COH(I)='S4' & SRG(I)>0 &                           01970007
                        (ADM(J)>=DIS(I) & ADM(J)<=Y&YY.ENDENR))             01980007
                        THEN DAYS1B=ADM(J)-SRG(I);                          01990007
                        ELSE IF (ADM(J)>=DIS(I) & ADM(J)<=Y&YY.ENDENR)      02000007
                           THEN DAYS1B=ADM(J)-DIS(I);                       02010007
                        NEXTADM=ADM(J);                                     02020007
                        NEXTDIS=DIS(J);                                     02030007
                        FOUND2=1;                                           02040007
P1H4.@MSQ00AG.PGMLIB(YXXAGP01) -- CONTINUED

                  END;                                                      02050007
               END; * j=i+1 to cnt;                                         02060007
            END; *CNT>1 & I<CNT;                                            02070007
            * Days3a= days between cohort readmission and the              02080007
               index admission or death whichever comes first;              02090007
            If (DAYS1A>.) THEN DAYS3A=COHADM-SADMDT;                        02100011
              Else If (DAYS2>.) THEN DAYS3A=DAYS2;                          02110007
              Else DAYS3A=.;                                                02120007
                                                                            02130007
            * Days3b= days between non-cohort readmission and the index     02140007
               admission or death whichever comes first;                    02150007
            If (DAYS1B>.) THEN DAYS3B=NEXTADM-SADMDT;                       02160011
              Else If (DAYS2>.) THEN DAYS3B=DAYS2;                          02170007
              Else DAYS3B=.;                                                02180007
                                                                            02190007
           * Days4a= days between cohort readmission and the                02200007
               index discharge or death whichever comes first;              02210007
            If (DAYS1A>.) THEN DAYS4A=DAYS1A;                               02220011
              Else If (DAYS2>.) THEN DAYS4A=Y&YY.M5DEATH-DIS(I);            02230007
              Else DAYS4A=.; IF DAYS4A<0 THEN DAYS4A=.;                     02240007
                                                                            02250007
           * Days4b= days between non-cohort readmission and the            02260007
               index discharge or death whichever comes first;              02270007
            If (DAYS1B>.) THEN DAYS4B=DAYS1B;                               02280011
              Else If (DAYS2>.) THEN DAYS4B=Y&YY.M5DEATH-DIS(I);            02290007
              Else DAYS4B=.; IF DAYS4B<0 THEN DAYS4B=.;                     02300007
            OUTPUT OUT1.TEMP1;                                              02310007
         END; *IF COHORT;                                                   02320007
      END; *I=1 TO CNT;                                                     02330007
   END; *LAST.HICNO;                                                        02340007
RUN;                                                                        02350007
                                                                            02360007
*****************************************************************;        02370007
* step 3. Sort file by COHORT Y&YY.HICNO SADMDT SDISDT                      02380007
*          to identify the first admission in the cohort for                02390007
*          bene level sums.                                                 02400007
*****************************************************************;        02410007
PROC SORT DATA=OUT1.TEMP1 OUT=OUT2.TEMP2;                                   02420007
   BY COHORT Y&YY.HICNO SADMDT SDISDT;                                      02430007
                                                                            02440007
*****************************************************************;        02450007
* step 4. Create the variable BENE=1 if first admission in cohort           02460007
*****************************************************************;        02470007
                                                                            02480007
```

```
DATA OUT3.Y&YY.AGP01;                                                    02490033
     SET OUT2.TEMP2; BY COHORT Y&YY.HICNO;                               02500007
     IF FIRST.Y&YY.HICNO THEN BENE=1; ELSE BENE=0;                       02510007
RUN;                                                                     02520007
                                                                         02530007
PROC CONTENTS DATA=OUT3.Y&YY.AGP01;                                      02540033
TITLE "CONTENTS OF M5/M6/M8 AGGREGATE FILE FOR &YEAR"; RUN;              02550025
                                                                         02560007
PROC PRINT DATA=OUT3.Y&YY.AGP01(OBS=50);                                 02570033
  FORMAT SADMDT SDISDT NEXTADM NEXTDIS COHADM COHDIS                     02580025
     Y&YY.M5DEATH Y&YY.ENDENR MMDDYY8.;  RUN;                            02590025
                                                                         02600007
%MEND YXXAGP01;                                                          02610033
```

P1H4.@MSQ00AG.PGMLIB(YXXAGP02)

```
%MACRO YXXAGP02(YY);

*********************************************************************

1  MACRO NAME:           YXXAGP02
2  LOCATION:             P1H4.@MSQ00AG.PGMLIB99(YXXAGP02)
3  WRITTEN BY:           ANN LARSEN
4  COMPANY:              RTI
5  DATE:                 April 2003
6  PROJECT:              MQMS
7  INPUT:                P1H4.@MSQ0000.SAS.YXXAGP01
8  OUTPUT:               P1H4.@MSQ0000.SAS.YXXAGP02
                         XX - is 2 digit for a year (for example 99)
9  PURPOSE:              To create 2 aggregate summary files for M5,
                         M6, and M8.  One summarizes benes and the
                         other summarizes discharges.

    STEP 1. Create measure/window level file with numerator &
            denominator variables
    STEP 2. Sort file by measure, window and 9 subgroup for the
            summations
    STEP 3. Sum new variables by measure, window and
            subgroups for beneficiary level observations.
    STEP 4. Sum new variables by measure, window and subgroups
            for discharge level observations.
    STEP 5. Combine the bene and discharges observations into one
            file and create labels for summations variables

*********************************************************************;

  %INCLUDE IN0(YXXAGFMT);
  %YXXAGFMT;

  %LET VARLIST=%STR(Y&YY.PST Y&YY.PCDIV Y&YY.PREG Y&YY.URBN Y&YY.SEX
    Y&YY.FAGE Y&YY.RACE Y&YY.DUAL Y&YY.MDST);

  %IF &YY>90 %THEN %LET YEAR=19&YY;
    %ELSE %LET YEAR=20&YY;

  %LET NEWVARS=%STR(READM DEATH M5DENOM M6DENOM TIME_TO_READM
    TIME_TO_DEATH TIME_TO_READM_OR_DEATH READM_OR_DEATH);

  %MACRO CREATE;

    DO I=1 TO 6;
       * Initialize new variables to 0;
       DO K=1 TO 8;
          NV(K)=0;
       END;

       * All Beneficiaries with index event;
       M6DENOM=1;

       * If death is within window;
       IF (.<DAYS2<=W(I)) THEN DO;
          DEATH=1;
```

```
         TIME_TO_DEATH=DAYS2;
      END;
```

P1H4.@MSQ00AG.PGMLIB(YXXAGP02) -- CONTINUED

```
      * If alive and in FFS in window or had a readmission anytime;
      IF (DAYS7>=W(I) ³ .<DAYS1<=W(6)) THEN DO;
         M5DENOM=1;

         * If readmission is within window;
         IF (.<DAYS1<=W(I)) THEN DO;
            READM=1;
            TIME_TO_READM=DAYS1;
         END;
      END;
      * If enrolled thru entire window & DEATH or readmitted;
      IF (READM=1 ³ (DEATH=1 & Y&YY.M5DEATH<=Y&YY.ENDENR)) THEN DO;
         READM_OR_DEATH=1;
         TIME_TO_READM_OR_DEATH=DAYS3;
      END;
      WINDOW=I;
      OUTPUT;
   END;
   DAYS1=.; DAYS3=.;
%MEND CREATE;
*******************************************************************;
* step 1. Create measure/window level file with numerator &
*         denominator variables
*******************************************************************;
 DATA OUT4.TEMP1(KEEP=MEASURE WINDOW BENE &VARLIST &NEWVARS
   Y&YY.HICNO SADMDT SDISDT Y&YY.ENDENR Y&YY.DEATH);
   SET OUT3.Y&YY.AGP01;

   ARRAY W(6) (2,7,30,180,365,730);
   ARRAY NV(8) &NEWVARS;

   IF (COHORT='HF') THEN DO;
    * HF -> ANY;
      IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
         DAYS1=DAYS1B; DAYS3=DAYS3B;
      END;
      MEASURE=1;
      %CREATE;
      * HF -> HF;
      IF (NEXTCOH='HF') THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      MEASURE=2;
      %CREATE;
   END;
   IF (COHORT='P1') THEN DO;
      IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
```

```
          DAYS1=DAYS1B; DAYS3=DAYS3B;
        END;
      * P1 -> ANY or death;
        MEASURE=3;
        %CREATE;
        * P1 -> P1-P7 or death;
```
P1H4.@MSQ00AG.PGMLIB(YXXAGP02) -- CONTINUED
```
        IF ('P1'<=NEXTCOH<='P7') THEN DO;
           DAYS1=DAYS1A; DAYS3=DAYS3A;
        END;
        MEASURE=5;
        %CREATE;
        * P1 -> not P1-P7 or death;
        IF (DAYS1B>.) THEN            DO;
           DAYS1=DAYS1B; DAYS3=DAYS3B;
        END;
        MEASURE=4;
        %CREATE;
     END;
     IF (COHORT='P6') THEN DO;
        IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
           DAYS1=DAYS1A; DAYS3=DAYS3A;
        END;
        ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
           DAYS1=DAYS1B; DAYS3=DAYS3B;
        END;
      * P6 -> ANY or death;
        MEASURE=6;
        %CREATE;
        * P6 -> P1-P7;
        IF ('P1'<=NEXTCOH<='P7') THEN DO;
           DAYS1=DAYS1A; DAYS3=DAYS3A;
        END;
        MEASURE=8;
        %CREATE;
        * P6 -> not P1-P7 or death;
        IF (DAYS1B>.) THEN DO;
           DAYS1=DAYS1B; DAYS3=DAYS3B;
        END;
        MEASURE=7;
        %CREATE;
     END;
     IF (COHORT='P7') THEN DO;
        IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
           DAYS1=DAYS1A; DAYS3=DAYS3A;
        END;
        ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
           DAYS1=DAYS1B; DAYS3=DAYS3B;
        END;
      * P7 -> ANY or death;
        MEASURE=9;
        %CREATE;
        * P7 -> P1-P7;
        IF ('P1'<=NEXTCOH<='P7') THEN DO;
           DAYS1=DAYS1A; DAYS3=DAYS3A;
        END;
        MEASURE=11;
```

```
      %CREATE;
      * P7 -> not P1-P7 or death;
      IF (DAYS1B>.) THEN DO;
         DAYS1=DAYS1B; DAYS3=DAYS3B;
      END;
      MEASURE=10;
      %CREATE;
   END;
   IF (COHORT='S1') THEN DO;
```
P1H4.@MSQ00AG.PGMLIB(YXXAGP02) -- CONTINUED
```
      IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
         DAYS1=DAYS1B; DAYS3=DAYS3B;
      END;
    * S1 -> ANY;
      MEASURE=12;
      %CREATE;
      * S1 -> S2 or death;
      IF (NEXTCOH='S2') THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      MEASURE=13;
      %CREATE;
   END;
   * S3 -> S2;
   IF (COHORT='S3') THEN DO;
      IF (NEXTCOH='S2') THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      MEASURE=14;
      %CREATE;
   END;
   * S4 -> S2 or death;
   IF (COHORT='S4') THEN DO;
      IF (NEXTCOH='S2') THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      MEASURE=15;
      %CREATE;
   END;
   IF (COHORT='M1') THEN DO;
      IF (.<DAYS1A<=DAYS1B ³ DAYS1B=.) THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      ELSE IF (.<DAYS1B<=DAYS1A ³ DAYS1A=.) THEN DO;
         DAYS1=DAYS1B; DAYS3=DAYS3B;
      END;
    * M1 -> ANY or death;
      MEASURE=16;
      %CREATE;
      * M1 -> M1;
      IF (NEXTCOH='M1') THEN DO;
         DAYS1=DAYS1A; DAYS3=DAYS3A;
      END;
      MEASURE=17;
```

```
      %CREATE;
   END;
RUN;
*******************************************************************;
* step 2. Sort file by measure, window and 9 subgroup for the sums
*******************************************************************;

PROC SORT DATA=OUT4.TEMP1 OUT=OUT5.TEMP2;
  BY MEASURE WINDOW &VARLIST; RUN;

PROC PRINT DATA=OUT5.TEMP2 (OBS=50);  RUN;
```

P1H4.@MSQ00AG.PGMLIB(YXXAGP02) -- CONTINUED

```
*******************************************************************;
* step 3. Sum denom and numerator vars by measure, window and
*         subgroups for bene level observations.
*******************************************************************;

PROC SUMMARY DATA=OUT5.TEMP2(WHERE=(BENE=1));
  BY MEASURE WINDOW &VARLIST;
  VAR &NEWVARS;
  OUTPUT OUT=BEN(DROP=_TYPE_ _FREQ_) SUM=;
RUN;

*******************************************************************;
* step 4. Sum denom and numerator vars by measure, window and
*         subgroups for discharge observations.
*******************************************************************;

PROC SUMMARY DATA=OUT5.TEMP2;
  BY MEASURE WINDOW &VARLIST;
  VAR &NEWVARS;
  OUTPUT OUT=DIS(DROP=_TYPE_ _FREQ_) SUM=;
RUN;

*******************************************************************;
* step 5. Combine the bene and discharges observations into one
*         file and create labels for summations variables
*******************************************************************;

DATA OUT6.Y&YY.AGP02; SET BEN(IN=INB) DIS; BY MEASURE WINDOW;
  BENE=(INB);
  LABEL
    MEASURE='Admission and Readmission combination'
    WINDOW='Window of time'
    READM='Sum of Readmissions'
    DEATH='Sum of Deaths'
    M5DENOM='Cohort Admission enrolled thru entire window'
    M6DENOM='Cohort Admission'
    TIME_TO_READM='Sum of days to readmission'
    TIME_TO_DEATH='Sum of days to death'
    TIME_TO_READM_OR_DEATH='Sum of days to readm or death'
    READM_OR_DEATH='Sum of admissions that had readm or death'
    BENE='First admission for this bene';
  RUN;

PROC CONTENTS DATA=OUT6.Y&YY.AGP02;
```

```
  TITLE "CONTENTS OF M5/M6/M8 AGGREGATE FILE FOR &YY";
RUN;

PROC PRINT DATA=OUT6.Y&YY.AGP02(OBS=40);
* FORMAT MEASURE MEASFMT. WINDOW WINFMT. Y&YY.SEX $SEXFMT.
         Y&YY.PST $STFMT. Y&YY.PCDIV $CDIVFMT. Y&YY.PREG $CREGFMT.
         Y&YY.URBN $URBFMT. Y&YY.MDST $MCDFMT.
         Y&YY.FAGE $AGEFMT. Y&YY.RACE $RACEFMT. Y&YY.DUAL $DUALFMT.;
RUN;

PROC MEANS DATA=OUT6.Y&YY.AGP02 SUM; BY MEASURE WINDOW; CLASS BENE;
  VAR &NEWVARS;
RUN;

%MEND YXXAGP02;
```

P1H4.@MSQ00AG.PGMLIB(ADJM5)

```
//P1H4M599 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//*****************************************************************
//* PRODUCE ADJUSTED READMISSION RATES
//*****************************************************************
//ADJM5 EXEC SAS8,REGION=8000K,OPTIONS='ERRORS=0,NEWS'
//WORK DD SPACE=(CYL,(2500,500))
//IN0   DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//NUM99 DD DSN=P1H4.@MSQ0000.SAS.Y99AGP02,DISP=SHR
//ADJ1  DD DSN=P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99,DISP=SHR
//ADJ2  DD DSN=P1H4.@MSQ0000.SAS.Y99M5M6.IDSTD,DISP=SHR
//OUT   DD DSN=P1H4.@MSQ0000.SAS.Y99M5.ADRATE99,
//      DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *,DLM=##
OPTIONS BUFNO=5;

 %INCLUDE IN0(ADRATEM5)/SOURCE2; * MACRO TO COMPUTE ADJUSTED RATES;
 %ADRATEM5(99);     *YEAR=99;
```

P1H4.@MSQ00AG.PGMLIB(ADRATEM5)

```
%MACRO ADRATEM5(YY);

*********************************************************************
* MACRO TO PRODUCE AGE/SEX ADJUSTED READMISSION RATES USING DIRECT
* ADJUSTMENT ON DEMOGRAPHIC STRATIFIERS AND INDIRECT ADJUSTMENT ON
* PROVIDER STATES.
*********************************************************************

  /**BEGIN DIRECT ADJUSTMENT PROCEDURE**/

*********************************************************************
* STEP 1: Merge the AGGREGATE file with the 1999 standard population
*         file containing age/sex adjustment weights.
*********************************************************************;

* Get readmission and denominator variables from the aggregate file by
  stratifiers including PROVIDER region;

proc sql;
  Create table dadj as
  SELECT BENE, MEASURE, WINDOW,
         Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
         Y&YY.URBN, Y&YY.PREG,
    sum(READM) as READMCNT,
    sum(M5DENOM) as DENOMCNT
  from num&yy..Y&YY.AGP02
  group by Y&YY.FAGE, Y&YY.SEX, BENE, MEASURE, WINDOW,
           Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
           Y&YY.URBN, Y&YY.PREG;
QUIT;

DATA dadj1;
merge dadj (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.race=race
                    y&yy.dual=dual y&yy.urbn=urbn y&yy.PREG=PREG))
      ADJ1.adjwts99 (keep= Y99fage Y99sex m13asadj m13aadj m13sadj
                     rename=(y99fage=fage y99sex=sex));
by fage sex;

*Recode MDST into 3 categories;
if Y&YY.mdst ='1' then mdst='1';
else if Y&YY.mdst ='3' then mdst='2';
else if Y&YY.mdst in ('2','4','5') then mdst='3';
DROP Y&YY.mdst;

PROC SORT DATA=DADJ1;
BY BENE MEASURE WINDOW;
RUN;

*********************************************************************
* STEP 2:  Sum up numerator and denominator counts by age*sex*subgroup
*          categories;
*********************************************************************;

proc summary data=dadj1;
BY BENE MEASURE WINDOW ;
class fage sex
```

```
      RACE DUAL MDST URBN PREG;
var readmcnt denomcnt;
id m13aadj m13sadj m13asadj ;
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM5) -- CONTINUED

```
OUTPUT OUT=DRRATE (WHERE=(_TYPE_ IN (96,97,98,100,104,112)))
SUM=readmcnt denomcnt;

*************************************************************
* STEP 3:  Divide the numerator total by denominator total by
*          age*sex*subgroup categories and multiply by appropriate
*          adjustment weights, i.e.,
*            For adjustment readm by age, multiply by sex-adjusted wts
*            For adjustment readm by sex, multiply by age-adjusted wts
*            For adjustment readm by other subgroup variables,
*                multiply by age/sex-adjusted wts
*************************************************************;

DATA DRRATE2; set DRRATE;
if _type_ in (96) then do; total='1';
  wtsexreadm = readmcnt / denomcnt * m13aadj;
  wtagereadm = readmcnt / denomcnt * m13sadj;
  wtedreadm =  readmcnt / denomcnt * m13asadj;
end;
else if _type_ in (97,98,104,112) then do;
  wtedreadm =  readmcnt / denomcnt * m13asadj;
  fage=. ;  sex=.;
end;

* Fix the readms by MDST with age-based correction factors;
else if _type_ in (100) then do;
  fage=. ;  sex=.;
  if mdst='1' then
    wtedreadm = readmcnt / denomcnt * m13asadj/0.84593;
  else if mdst='2' then
    wtedreadm = readmcnt / denomcnt * m13asadj/0.15407;
  else if mdst='3' then
    wtedreadm = readmcnt / denomcnt * m13asadj;
END;

*************************************************************
* STEP 4:  Sum the multiplied products up to each categorical level
*          of a subgroup variable - This becomes the adjusted readm
*          for individual subgroup categories
*************************************************************;

PROC TABULATE DATA=DRRATE2     out=DRRATE3
  missing noseps FORMCHAR='           ';
BY BENE MEASURE WINDOW;
class total fage sex
      RACE DUAL MDST URBN PREG;
VAR readmcnt denomcnt wtagereadm wtsexreadm wtedreadm;
TABLE total,
 ((denomcnt='' readmcnt='')*SUM=''*F=12.
  (WTEDreadm='')*SUM=''*F=6.2);
TABLE fage,
 ((denomcnt='' readmcnt='')*SUM=''*F=12.
   (wtagereadm='')*SUM=''*F=6.2);
```

```
 TABLE sex,
  ((denomcnt='' readmcnt='')*SUM=''*F=12.
   (wtsexreadm='')*SUM=''*F=6.2);
 TABLE (RACE dual mdst URBN PREG),
  ((denomcnt='' readmcnt='')*SUM=''*F=12.
   (WTEDreadm='' )*SUM=''*F=6.2)
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM5) -- CONTINUED

```
 /RTS=28 ROW=FLOAT;
 KEYLABEL ALL='All enrollees'
          sum='';
          LABEL   TOTAL='All enrollees'
                   FAGE='Age Group'
                    SEX='Sex'
                    URBN='Urban/Rural'
                    RACE='Race'
                    MDST='Medicare Eligibility Status'
                    DUAL='Other Disadvantaged Status'
                    PREG='Provider region';
 RUN;

*************************************************************
* STEP 5:  Output both unadjusted and adjusted readm for individual
*          subgroup categories, and set the results to missing (.)
*          if the numerator count is 25 or less.
*************************************************************;

data DRRATE&yy; set DRRATE3 (rename=(denomcnt_sum=denomcnt&yy
                readmcnt_sum=readmcnt&yy ));

* Compute unadjusted readms;
readmrate&yy = readmcnt&yy / denomcnt&yy * 1000;

* Compute adjusted readms;
if total='1' then
    ADJRRATE&YY=wtedreadm_sum * 1000;
else if sex ge 0 then
    ADJRRATE&YY=wtsexreadm_sum * 1000;
else if fage ge 0 then
    ADJRRATE&YY=wtAGEreadm_sum * 1000;
else if race ge 0 or
        mdst ge 0 or
        dual ge 0 or
        urbn ge 0 or
        PREG ge 0 then
    ADJRRATE&YY=wtedreadm_sum * 1000;
else delete;

* Set the adjusted readms to . if the numerator count is 25 or less;
if readmcnt&yy le 25 then ADJRRATE&YY=.;
drop _type_ _page_ _table_ wtsexreadm_sum wtagereadm_sum WTEDreadm_sum;
run;

  /**END OF DIRECT ADJUSTMENT PROCEDURE**/

  /**BEGIN INDIRECT ADJUSTMENT PROCEDURE**/

*************************************************************************
```

```
    * For each year, obtain observed(O) and expected(E) number of numerator
    * events (i.e., CC READMCNTarges and DENOMCNTs with CC READMCNTarges) by
state
    * and multiply O/E to the 1999 national unadjusted rate to compute
    * the indirectly adjusted rates for individual states
    ********************************************************************;

    * Step 1: Get sum of readmission and denominator counts by age/sex and
              provider state for each measure and window;
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM5) -- CONTINUED

```
    proc sql;
      Create table AGP as
      SELECT BENE, MEASURE, WINDOW, Y&YY.FAGE, Y&YY.SEX, Y&YY.PST,
        sum(READM) as READMCNT,
        sum(M5DENOM) as DENOMCNT
      from num&yy..Y&YY.AGP02
      GROUP BY BENE, MEASURE, WINDOW, Y&YY.FAGE, Y&YY.SEX, Y&YY.PST;
    QUIT;

    * Step 2: Merge the above file with the standard rate file and
              compute expected outcome events by multiplying the standard
              state-specific rates with the denominator counts for each
              age*sex*state cell;

    data combine ;
    merge AGP     (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.PST=PST))
          ADJ2.IDSTD (rename=(y99fage=fage y99sex=sex));
    by BENE measure window fage sex;

    *SETTING outside US to 0;
     IF PST in ('40','48') or PST GT 53 THEN PST='0';

    expreadm = STDRrate*denomcnt;
    run;

    * Step 3: Compute observed and expected sum of numerator events and
              denominator counts for each state;

    proc summary data=combine;
    BY BENE MEASURE WINDOW;
    class PST;
    var expreadm  denomcnt READMCNT;
    ID NSTDRrate;
    output out=iADJRATE
           sum=expreadm&yy  denomcnt&yy READMCNT&yy;

    * Step 4: Compute unadjusted and indirectly adjusted rates for each
              state;

    data IDRATE&YY; set iADJRATE;
    srr&yy = READMCNT&yy / expreadm&yy;
    readmrate&yy = READMCNT&yy / denomcnt&yy *1000; *Unadjusted rate;
    ADJRrate&yy = srr&yy * NSTDRrate;              *Adjusted rate;

    if _type_ in (0) then total='2';               *Setting U.S. total;
    if READMCNT&yy le 25 then ADJRrate&yy=.;       *Setting rates based on;
```

```
                                                  *numerator events of 25;
                                                   *or less to missing;
drop _type_ _freq_ expreadm&yy;
run;

    /**END OF INDIRECT ADJUSTMENT PROCEDURE**/

DATA out.ADRATE&YY;  SET DRRATE&YY IDRATE&YY;
proc sort ;
by BENE MEASURE WINDOW total fage sex race dual mdst urbn preg pst;
RUN;
```

P1H4.@MSQ00AG.PGMLIB(ADRATEM5) -- CONTINUED

```
PROC PRINT DATA=OUT.ADRATE&YY (OBS=100);
TITLE 'ADJUSTED READMISSION RATES';
RUN;

%MEND ADRATEM5;
```

P1H4.@MSQ00AG.PGMLIB(ADJM6)

```
//P1H4M699 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//*****************************************************************
//* PRODUCE ADJUSTED MORTALITY RATES
//*****************************************************************
//ADJM6 EXEC SAS8,REGION=8000K,OPTIONS='ERRORS=0,NEWS'
//WORK DD SPACE=(CYL,(2500,500))
//IN0   DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//NUM99 DD DSN=P1H4.@MSQ0000.SAS.Y99AGP02,DISP=SHR
//ADJ1  DD DSN=P1H4.@MSQ0000.SAS.Y99M1.ADJWTS99,DISP=SHR
//ADJ2  DD DSN=P1H4.@MSQ0000.SAS.Y99M5M6.IDSTD,DISP=SHR
//OUT   DD DSN=P1H4.@MSQ0000.SAS.Y99M6.AMRATE99,
//      DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *,DLM=##
OPTIONS BUFNO=5;

 %INCLUDE IN0(ADRATEM6)/SOURCE2; * MACRO TO COMPUTE ADJUSTED RATES;
 %ADRATEM6(99);     *YEAR=99;
```

P1H4.@MSQ00AG.PGMLIB(ADRATEM6)

```
%MACRO ADRATEM6(YY);

*********************************************************************
* MACRO TO PRODUCE AGE/SEX ADJUSTED MORTALITY RATES USING DIRECT
* ADJUSTMENT ON DEMOGRAPHIC STRATIFIERS AND INDIRECT ADJUSTMENT ON
* PROVIDER STATES.
*********************************************************************

  /**BEGIN DIRECT ADJUSTMENT PROCEDURE**/

*********************************************************************
* STEP 1: Merge the AGGREGATE file with the 1999 standard population
*         file containing age/sex adjustment weights.
*********************************************************************;

* Get MORTALITY and denominator variables from the aggregate file by
  stratifiers including PROVIDER region;

proc sql;
  Create table dadj as
  SELECT BENE, MEASURE, WINDOW,
         Y&YY.FAGE, Y&YY.SEX, Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
         Y&YY.URBN, Y&YY.PREG,
    sum(DEATH) as DEATHCNT,
    sum(M6DENOM) as DENOMCNT
  from num&yy..Y&YY.AGP02
  where measure in (1,3,6,9,12,14,15,16)  /* 8 cc groups */
  group by Y&YY.FAGE, Y&YY.SEX, BENE, MEASURE, WINDOW,
           Y&YY.RACE, Y&YY.DUAL, Y&YY.MDST,
           Y&YY.URBN, Y&YY.PREG;
QUIT;

DATA dadj1;
merge dadj (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.race=race
                    y&yy.dual=dual y&yy.urbn=urbn y&yy.PREG=PREG))
      ADJ1.adjwts99 (keep= Y99fage Y99sex m13asadj m13aadj m13sadj
                     rename=(y99fage=fage y99sex=sex));
by fage sex;

*Recode MDST into 3 categories;
if Y&YY.mdst ='1' then mdst='1';
else if Y&YY.mdst ='3' then mdst='2';
else if Y&YY.mdst in ('2','4','5') then mdst='3';
DROP Y&YY.mdst;

PROC SORT DATA=DADJ1;
BY BENE MEASURE WINDOW;
RUN;

*********************************************************************
* STEP 2:  Sum up numerator and denominator counts by age*sex*subgroup
*          categories;
*********************************************************************;

proc summary data=dadj1;
BY BENE MEASURE WINDOW ;
```

```
class fage sex
      RACE DUAL MDST URBN PREG;
var DEATHcnt denomcnt;
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM6) -- CONTINUED

```
id m13aadj m13sadj m13asadj ;
OUTPUT OUT=DRRATE (WHERE=(_TYPE_ IN (96,97,98,100,104,112)))
SUM=DEATHcnt denomcnt;

*************************************************************
* STEP 3:  Divide the numerator total by denominator total by
*          age*sex*subgroup categories and multiply by appropriate
*          adjustment weights, i.e.,
*            For adjustment DEATH by age, multiply by sex-adjusted wts
*            For adjustment DEATH by sex, multiply by age-adjusted wts
*            For adjustment DEATH by other subgroup variables,
*                multiply by age/sex-adjusted wts
*************************************************************;

DATA DRRATE2; set DRRATE;
if _type_ in (96) then do; total='1';
  wtsexDEATH = DEATHcnt / denomcnt * m13aadj;
  wtageDEATH = DEATHcnt / denomcnt * m13sadj;
  wtedDEATH =  DEATHcnt / denomcnt * m13asadj;
end;
else if _type_ in (97,98,104,112) then do;
  wtedDEATH =  DEATHcnt / denomcnt * m13asadj;
  fage=. ;  sex=.;
end;
* Fix the DEATHs by MDST with age-based correction factors;
else if _type_ in (100) then do;
  fage=. ;  sex=.;
  if mdst='1' then
    wtedDEATH = DEATHcnt / denomcnt * m13asadj/0.84593;
  else if mdst='2' then
    wtedDEATH = DEATHcnt / denomcnt * m13asadj/0.15407;
  else if mdst='3' then
    wtedDEATH = DEATHcnt / denomcnt * m13asadj;
END;

*************************************************************
* STEP 4:  Sum the multiplied products up to each categorical level
*          of a subgroup variable - This becomes the adjusted DEATH
*          for individual subgroup categories
*************************************************************;

PROC TABULATE DATA=DRRATE2     out=DRRATE3
  missing noseps FORMCHAR='           ';
BY BENE MEASURE WINDOW;
class total fage sex
      RACE DUAL MDST URBN PREG;
VAR DEATHcnt denomcnt wtageDEATH wtsexDEATH wtedDEATH;
TABLE total,
 ((denomcnt='' DEATHcnt='')*SUM=''*F=12.
  (WTEDDEATH='')*SUM=''*F=6.2);
TABLE fage,
 ((denomcnt='' DEATHcnt='')*SUM=''*F=12.
   (wtageDEATH='')*SUM=''*F=6.2);
```

```
 TABLE sex,
  ((denomcnt='' DEATHcnt='')*SUM=''*F=12.
   (wtsexDEATH='')*SUM=''*F=6.2);
 TABLE (RACE dual mdst URBN PREG),
  ((denomcnt='' DEATHcnt='')*SUM=''*F=12.
   (WTEDDEATH='' )*SUM=''*F=6.2)
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM6) -- CONTINUED

```
 /RTS=28 ROW=FLOAT;
 KEYLABEL ALL='All enrollees'
          sum='';
          LABEL   TOTAL='All enrollees'
                   FAGE='Age Group'
                    SEX='Sex'
                    URBN='Urban/Rural'
                    RACE='Race'
                    MDST='Medicare Eligibility Status'
                    DUAL='Other Disadvantaged Status'
                    PREG='Provider region';
 RUN;

*************************************************************
* STEP 5:  Output both unadjusted and adjusted DEATH for individual
*          subgroup categories, and set the results to missing (.)
*          if the numerator count is 25 or less.
*************************************************************;

data DRRATE&yy; set DRRATE3 (rename=(denomcnt_sum=denomcnt&yy
                DEATHcnt_sum=DEATHcnt&yy ));

* Compute unadjusted DEATHs;
MRATE&YY = DEATHcnt&yy / denomcnt&yy * 1000;

* Compute adjusted DEATHs;
if total='1' then
    ADJMRATE&YY=wtedDEATH_sum * 1000;
else if sex ge 0 then
    ADJMRATE&YY=wtsexDEATH_sum * 1000;
else if fage ge 0 then
    ADJMRATE&YY=wtAGEDEATH_sum * 1000;
else if race ge 0 or
        mdst ge 0 or
        dual ge 0 or
        urbn ge 0 or
        PREG ge 0 then
    ADJMRATE&YY=wtedDEATH_sum * 1000;
else delete;

* Set the adjusted DEATHs to . if the numerator count is 25 or less;
if DEATHcnt&yy le 25 then ADJMRATE&YY=.;
drop _type_ _page_ _table_ wtsexDEATH_sum wtageDEATH_sum WTEDDEATH_sum;
run;

  /**END OF DIRECT ADJUSTMENT PROCEDURE**/

  /**BEGIN INDIRECT ADJUSTMENT PROCEDURE**/

*********************************************************************
```

```
* For each year, obtain observed(O) and expected(E) number of numerator
* events (i.e., deaths) by stateand multiplythe ratio, O/E,
* to the 1999 national unadjusted mortality rate to compute
* the indirectly adjusted mortality rates for individual states
*********************************************************************;

* Step 1: Get sum of MORTALITY and denominator counts by age/sex and
          provider state for each measure and window;
```

P1H4.@MSQ00AG.PGMLIB(ADRATEM6) -- CONTINUED

```
proc sql;
  Create table AGP as
  SELECT BENE, MEASURE, WINDOW, Y&YY.FAGE, Y&YY.SEX, Y&YY.PST,
    sum(DEATH) as DEATHCNT,
    sum(M6DENOM) as DENOMCNT
  from num&yy..Y&YY.AGP02
  where measure in (1,3,6,9,12,14,15,16)  /* 8 cc groups */
  GROUP BY BENE, MEASURE, WINDOW, Y&YY.FAGE, Y&YY.SEX, Y&YY.PST;
QUIT;

* Step 2: Merge the above file with the standard rate file and
          compute expected outcome events by multiplying the standard
          state-specific rates with the denominator counts for each
          age*sex*state cell;

data combine ;
merge AGP     (rename=(y&yy.fage=fage y&yy.sex=sex y&yy.PST=PST))
      ADJ2.IDSTD (where=(measure in (1,3,6,9,12,14,15,16))
                  rename=(y99fage=fage y99sex=sex));
by BENE measure window fage sex;

*SETTING OUTSIDE US to 0;
 IF PST in ('40','48') OR PST GT '53' THEN PST='0';

EXPDEATH = STDMRATE*denomcnt;
run;

* Step 3: Compute observed and expected sum of numerator events and
          denominator counts for each state;

proc summary data=combine;
BY BENE MEASURE WINDOW;
class PST;
var expDEATH  denomcnt DEATHCNT;
ID NSTDMRATE;
output out=iADJRATE
       sum=expDEATH&yy  denomcnt&yy DEATHCNT&yy;

* Step 4: Compute unadjusted and indirectly adjusted rates for each
          state;

data IDRATE&YY; set iADJRATE;
srr&yy = DEATHCNT&yy / expDEATH&yy;
MRATE&YY = DEATHCNT&yy / denomcnt&yy *1000;    *Unadjusted rate;
ADJMRATE&YY = srr&yy * NSTDMRATE;              *Adjusted rate;

if _type_ in (0) then total='2';               *Setting U.S. total;
```

```
if DEATHCNT&yy le 25 then ADJMRATE&YY=.;       *Setting rates based on;
                                                *numerator events of 25;
                                                 *or less to missing;
drop _type_ _freq_ expDEATH&yy;
run;

    /**END OF INDIRECT ADJUSTMENT PROCEDURE**/

DATA out.AMRATE&YY;  SET DRRATE&YY IDRATE&YY;
proc sort ;
by BENE MEASURE WINDOW total fage sex race dual mdst urbn preg pst;
RUN;
```
P1H4.@MSQ00AG.PGMLIB(ADRATEM6) -- CONTINUED

```
PROC PRINT DATA=OUT.AMRATE&YY (where=(bene=1 and measure=16));
TITLE 'ADJUSTED MORTALITY RATES';
RUN;

%MEND ADRATEM6;
```

P1H4.@MSQ00AG.PGMLIB(MERGEM5)

```
//P1H4MEM5 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN92 DD DSN=P1H4.@MSQ0000.SAS.Y92M5.ADRATE92,DISP=SHR
//IN93 DD DSN=P1H4.@MSQ0000.SAS.Y93M5.ADRATE93,DISP=SHR
//IN94 DD DSN=P1H4.@MSQ0000.SAS.Y94M5.ADRATE94,DISP=SHR
//IN95 DD DSN=P1H4.@MSQ0000.SAS.Y95M5.ADRATE95,DISP=SHR
//IN96 DD DSN=P1H4.@MSQ0000.SAS.Y96M5.ADRATE96,DISP=SHR
//IN97 DD DSN=P1H4.@MSQ0000.SAS.Y97M5.ADRATE97,DISP=SHR
//IN98 DD DSN=P1H4.@MSQ0000.SAS.Y98M5.ADRATE98,DISP=SHR
//IN99 DD DSN=P1H4.@MSQ0000.SAS.Y99M5.ADRATE99,DISP=SHR
//IN00 DD DSN=P1H4.@MSQ0000.SAS.Y00M5.ADRATE00,DISP=SHR
//IN01 DD DSN=P1H4.@MSQ0000.SAS.Y01M5.ADRATE01,DISP=SHR
//OUT  DD DSN=P1H4.@MSQ0000.SAS.Y9201M5.READMP,
//     DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *
*******************************************************************
* PURPOSE: Outputs the 1992-2001 data on M5 READMISSION RATES
*          in the report format, to be downloaded into excel
*          spreadsheet.
*******************************************************************;

DATA OUT.READMP;
MERGE IN92.ADRATE92 IN93.ADRATE93 IN94.ADRATE94 IN95.ADRATE95
      IN96.ADRATE96 IN97.ADRATE97 IN98.ADRATE98 IN99.ADRATE99
      IN00.ADRATE00 IN01.ADRATE01;
by bene measure window total fage sex race dual mdst urbn preg pst;

PROC PRINT DATA=OUT.READMP (OBS=100);
TITLE 'ADJUSTED READMISSION RATES, 1992-2001';
RUN;
```

P1H4.@MSQ00AG.PGMLIB(MERGEM6)

```
//P1H4MEM6 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN92 DD DSN=P1H4.@MSQ0000.SAS.Y92M6.AMRATE92,DISP=SHR
//IN93 DD DSN=P1H4.@MSQ0000.SAS.Y93M6.AMRATE93,DISP=SHR
//IN94 DD DSN=P1H4.@MSQ0000.SAS.Y94M6.AMRATE94,DISP=SHR
//IN95 DD DSN=P1H4.@MSQ0000.SAS.Y95M6.AMRATE95,DISP=SHR
//IN96 DD DSN=P1H4.@MSQ0000.SAS.Y96M6.AMRATE96,DISP=SHR
//IN97 DD DSN=P1H4.@MSQ0000.SAS.Y97M6.AMRATE97,DISP=SHR
//IN98 DD DSN=P1H4.@MSQ0000.SAS.Y98M6.AMRATE98,DISP=SHR
//IN99 DD DSN=P1H4.@MSQ0000.SAS.Y99M6.AMRATE99,DISP=SHR
//IN00 DD DSN=P1H4.@MSQ0000.SAS.Y00M6.AMRATE00,DISP=SHR
//IN01 DD DSN=P1H4.@MSQ0000.SAS.Y01M6.AMRATE01,DISP=SHR
//OUT  DD DSN=P1H4.@MSQ0000.SAS.Y9201M6.MORTALP,
//     DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *
*******************************************************************
* PURPOSE: Combines the 1992-2001 data on M6 mortality rates.
*******************************************************************;

DATA OUT.MORTALP;
MERGE IN92.AMRATE92 IN93.AMRATE93 IN94.AMRATE94 IN95.AMRATE95
      IN96.AMRATE96 IN97.AMRATE97 IN98.AMRATE98 IN99.AMRATE99
      IN00.AMRATE00 IN01.AMRATE01;
by bene measure window total fage sex race dual mdst urbn preg pst;

PROC PRINT DATA=OUT.MORTALP (OBS=100);
TITLE 'ADJUSTED MORTALITY RATES, 1992-2001';
RUN;
```

P1H4.@MSQ00AG.PGMLIB(TABLEUNA)

```
//P1H4TABU JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN0  DD DSN=P1H4.@MSQ00M4.PGMLIB,DISP=SHR
//IN1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB01,DISP=SHR
//IN2  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB02,DISP=SHR
//SYSIN DD *
*********************************************************************
* PURPOSE: Outputs the 1992-2001 data on inadjusted M4 measures for
*          each clinical condition by stratifiers in the report format,
*          to be downloaded into excel spreadsheet.
* INPUT:   P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB01
*          P1H4.@MSQ0000.SAS.Y9201M4.PN5TAB02
* OUTPUT:  P1H4.@MSQ00M4.PN5TABLE(TABLEBn) where n is the table number.
*
* NOTE:    This program is submitted separately for each clinical
*          condition. The condition name (e.g., HF, MI, IS1..IS4, and
*          PN1..PN5) needs to be changed for each submission.
*********************************************************************;

%include IN0(TABLEFMT);    * Read in formats for the stratifiers;
%TABLEFMT;

*********************************************************************
* Read in data that contain unadjusted M4 measures by stratifiers
* including BENEFICIARY region and state
*********************************************************************;

data PN5tab01;  set in1.PN5tab01;

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if creg='0' then creg='5';
  if state='0' then state='90';

* Divide total BPAY amount by 1,000 to display the amount in $1,000s;

  array bpay(10) bpay92 bpay93 bpay94 bpay95 bpay96 bpay97 bpay98
                 bpay99 bpay00 bpay01;
  do i=1 to 10;
     bpay(i) = bpay(i)/1000;
  end;

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if creg  ne ' ' then do;  cat=8;  catgr=creg;  end;
else if state ne ' ' then do;  cat=9;  catgr=state; end;

* use the cat variables and the catgr variables
```

```
to determine the group for a statistic;

LENGTH ROWLAB $25 ;
```
P1H4.@MSQ00AG.PGMLIB(TABLEUNA) -- CONTINUED
```

if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
else if cat=7 then rowlab=put(catgr,$urbfmt.);
else if cat=8 then rowlab=put(catgr,$cregfmt.);
else if cat=9 then rowlab=put(catgr,$stfmt.);

proc sort; by cat race creg state;
RUN;
*****************************************************************
* Read in data that contain unadjusted M4 measures by stratifiers
* including PROVIDER region and state
*****************************************************************;

data PN5tab02;  set in2.PN5tab02;

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if preg='0' then preg='5';
  if pst ='0' then pst ='90';

* Divide total TPAY amount by 1,000 to display the amount in $1,000s;

  array tpay(10) tpay92 tpay93 tpay94 tpay95 tpay96 tpay97 tpay98
                 tpay99 tpay00 tpay01;
  do i=1 to 10;
     tpay(i) = tpay(i)/1000;
  end;

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if preg  ne ' ' then do;  cat=8;  catgr=preg;  end;
else if pst   ne ' ' then do;  cat=9;  catgr=pst ;  end;

* use the cat variables and the catgr variables
to determine the group for a statistic;

LENGTH ROWLAB $25 ;
if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
```

```
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
else if cat=7 then rowlab=put(catgr,$urbfmt.);
else if cat=8 then rowlab=put(catgr,$cregfmt.);
else if cat=9 then rowlab=put(catgr,$stfmt.);

proc sort; by cat race preg pst;
```
P1H4.@MSQ00AG.PGMLIB(TABLEUNA) -- CONTINUED
```
RUN;

*****************************************************************
* Output individual tables to the directory named below using the
* following macro program
*****************************************************************;

filename fileout 'P1H4.@msq00M4.PN5table';

%macro table(i,n,var);

data _null_;
set PN5tab0&i; by cat;
CC='PN5'; tablenum = &n; var="&var";
file fileout(tableb&n) header=toppage ls=132 ps=250  ;

if first.cat then do;
if      cat=1 then  put // @1 ' ' ;
else if cat=2 then  put  @1 'Gender' ;
else if cat=3 then  put  @1 'Age Group' ;
else if cat=4 then  put  @1 'Race' ;
else if cat=5 then  put  @1 'Medicaid Buy-In Status' ;
else if cat=6 then  put  @1 'Medicare Eligibility Status' ;
else if cat=7 then  put  @1 'Urban/Rural' ;
else if cat=8 then  put  @1 'Census Region' ;
else if cat=9 then  put  @1 'State' ;
end;

if tablenum in (1,2,7,9) then do;                    *sum measures;
put @1 rowlab $25. @27 &var.92 comma10. @37 &var.93 comma10.
    @47 &var.94 comma10. @57 &var.95 comma10. @67 &var.96 comma10.
    @77 &var.97 comma10. @87 &var.98 comma10. @97 &var.99 comma10.
    @107 &var.00 comma10. @117 &var.01 comma10.;
end;

else if tablenum in (8,10,11) then do;              *average measures;
put @1 rowlab $25. @27 &var.92 comma10.2 @37 &var.93 comma10.2
    @47 &var.94 comma10.2 @57 &var.95 comma10.2 @67 &var.96 comma10.2
    @77 &var.97 comma10.2 @87 &var.98 comma10.2 @97 &var.99 comma10.2
    @107 &var.00 comma10.2 @117 &var.01 comma10.2;
end;
return;
toppage:

if tablenum in (7,9) then do;                *Put $ amounts in 1,000s;
PUT #1 @1 CC $CCFMT. #2 @1 'Table B.' @9 tablenum @11 var $varfmt. /
#4 @75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001'/
@67 '(in thousands of dollars)';
```

```
end;

else do;
PUT #1 @1 CC $CCFMT. #2 @1 'Table B.' @9 tablenum @11 var $varfmt. /
#4 @75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001';
end;
run;
```
P1H4.@MSQ00AG.PGMLIB(TABLEUNA) -- CONTINUED

```
%mend;

%table(1,1,disch);
%table(1,2,bene);
%table(1,9,bpay);
%table(2,7,tpay);
%table(2,8,avgtpay);
%table(2,10,avgcc);
%table(2,11,avgdis);
```

P1H4.@MSQ00AG.PGMLIB(TABLEADJ)

```
//P1H4TABP JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN0  DD DSN=P1H4.@MSQ00M4.PGMLIB,DISP=SHR
//IN1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.IS4TAB03,DISP=SHR
//IN2  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.IS4TAB04,DISP=SHR
//SYSIN DD *
*****************************************************************
* PURPOSE: Outputs the 1992-2001 data on M4 measures by stratifiers
*          in the report format, to be downloaded into excel
*          spreadsheet.
* INPUT:   P1H4.@MSQ0000.SAS.Y9201M4.IS4TAB03
*          P1H4.@MSQ0000.SAS.Y9201M4.IS4TAB04
* OUTPUT:  P1H4.@MSQ00M4.IS4TABLE(TABLEBn) where n is the table
*                                          number.
*****************************************************************;

%include IN0(TABLEFMT);
%TABLEFMT;

*****************************************************************
* Read in data that contain adjusted M4 measures by stratifiers
* including BENEFICIARY region and state
*****************************************************************;

data IS4tab03;  set in1.IS4TAB03;

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if creg='0' then creg='5';
  if state='0' then state='90';

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if creg  ne ' ' then do;  cat=8;  catgr=creg;  end;
else if total='2'    then do;  cat=9;  catgr=total; end;
else if state ne ' ' then do;  cat=10; catgr=state; end;

* use the cat variables and the catgr variables
to deterIS4ne the group for a statistic;

LENGTH ROWLAB $25 ;
if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
else if cat=7 then rowlab=put(catgr,$urbfmt.);
```

```
else if cat=8 then rowlab=put(catgr,$cregfmt.);
else if cat=9 then rowlab=put(catgr,$totfmt.);
else if cat=10 then rowlab=put(catgr,$stfmt.);
```


```
proc sort; by cat race creg state;
RUN;

*****************************************************************
* Read in data that contain adjusted M4 measures by stratifiers
* including PROVIDER region and state
*****************************************************************;

data IS4tab04;  set in2.IS4TAB04;

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if preg='0' then preg='5';
  if pst ='0' then pst ='90';

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if Preg  ne ' ' then do;  cat=8;  catgr=Preg;  end;
else if total='2'    then do;  cat=9;  catgr=total; end;
else if pst   ne ' ' then do;  cat=10; catgr=pst;   end;

* use the cat variables and the catgr variables
to deterIS4ne the group for a statistic;

LENGTH ROWLAB $25 ;
if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
else if cat=7 then rowlab=put(catgr,$urbfmt.);
else if cat=8 then rowlab=put(catgr,$cregfmt.);
else if cat=9 then rowlab=put(catgr,$totfmt.);
else if cat=10 then rowlab=put(catgr,$stfmt.);

proc sort; by cat race preg pst;
RUN;

*****************************************************************
* Output individual tables to the directory named below using the
* following macro program
*****************************************************************;

filename fileout 'P1H4.@msq00m4.IS4table';
```

```
%macro table(i,n,var);

data _null_;
set IS4tab0&i; by cat;
cc='IS4'; tablenum = &n;  var="&var";
```
P1H4.@MSQ00AG.PGMLIB(TABLEADJ) -- CONTINUED
```
file fileout(tableb&n) header=toppage ls=132 ps=250  ;

if first.cat then do;
if      cat=1 then  put // @1 ' ' ;
else if cat=2 then  put  @1 'Gender' ;
else if cat=3 then  put  @1 'Age Group' ;
else if cat=4 then  put  @1 'Race' ;
else if cat=5 then  put  @1 'Medicaid Buy-In Status' ;
else if cat=6 then  put  @1 'Medicare Eligibility Status' ;
else if cat=7 then  put  @1 'Urban/Rural' ;
else if cat=8 then  put  @1 'Census Region' ;
else if cat=9 then  put //@1 ' ' ;
else if cat=10 then  put  @1 'State' ;
end;

put @1 rowlab $25. @27 &var.92 comma10.2 @37 &var.93 comma10.2
    @47 &var.94 comma10.2 @57 &var.95 comma10.2 @67 &var.96 comma10.2
    @77 &var.97 comma10.2 @87 &var.98 comma10.2 @97 &var.99 comma10.2
    @107 &var.00 comma10.2 @117 &var.01 comma10.2;
return;
toppage:

if tablenum in (4,5) then do;               *discharge rates;
PUT #1 @1 cc $ccfmt. #2 @1 'Table B.' @9 tablenum @11 var $varfmt. //
@75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001'/
@67 '(rate per 1,000 beneficiaries)';
end;

ELSE if tablenum in (6) then do;            *average LOS;
PUT #1 @1 cc $ccfmt. #2 @1 'Table B.' @9 tablenum @11 var $varfmt. //
@75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001'/
@67 'Average Length of Stay (days)';
end;
run;

%mend;

%table(3,4,adjdrate);
%table(3,5,adjbrate);
%table(4,6,adjlos);
```

P1H4.@MSQ00AG.PGMLIB(TABLEM5)

```
//P1H4T5HF JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN0  DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//IN1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M5.READMP,DISP=SHR
//*UT  DD DSN=P1H4.@MSQ0000.SAS.Y9201M4.ADRATEP,
//*    DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSIN DD *
*******************************************************************
* PURPOSE: Outputs the 1992-2001 data on M5 READMISSION RATES
*          in the report format, to be downloaded into excel
*          spreadsheet.
*******************************************************************;

%include IN0(TABM5FMT);
%TABM5FMT;

*******************************************************************
* Read in data that contain adjusted M5 measures by stratifiers
* including PROVIDER region and state
*******************************************************************;

%macro TABLE(b,m,n);

data readmp;  set in1.readmp (where=(bene=&b and measure=&m));

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if preg='0' then preg='5';
  if pst ='0' then pst ='90';

if window in (4,5) then adjrrate01=.;
else if window=6 then do; adjrrate00=.; adjrrate01=.;  END;

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if Preg  ne ' ' then do;  cat=8;  catgr=Preg;  end;
else if total='2'    then do;  cat=9;  catgr=total; end;
else if pst   ne ' ' then do;  cat=10; catgr=pst;   end;

* use the cat variables and the catgr variables
to determine the group for a statistic;

LENGTH ROWLAB $25 ;
if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
```

```
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
else if cat=7 then rowlab=put(catgr,$urbfmt.);
else if cat=8 then rowlab=put(catgr,$cregfmt.);
```


```
else if cat=9 then rowlab=put(catgr,$totfmt.);
else if cat=10 then rowlab=put(catgr,$stfmt.);

proc sort; by WINDOW cat race preg pst;
RUN;

*****************************************************************
* Output individual tables to the directory named below using the
* following macro program
*****************************************************************;

filename fileout 'P1H4.@msq00m4.hftable';

data _null_;
set READMP; by WINDOW CAT;
tablenum = &n;
file fileout(tableb&n) header=toppage ls=132 ps=180  ;

if first.cat then do;
if first.window then put // @1 window winfmt.;
if      cat=1 then  put // @1 ' ' ;
else if cat=2 then  put  @1 'Gender' ;
else if cat=3 then  put  @1 'Age Group' ;
else if cat=4 then  put  @1 'Race' ;
else if cat=5 then  put  @1 'Medicaid Buy-In Status' ;
else if cat=6 then  put  @1 'Medicare Eligibility Status' ;
else if cat=7 then  put  @1 'Urban/Rural' ;
else if cat=8 then  put  @1 'Census Region' ;
else if cat=9 then  put //@1 ' ' ;
else if cat=10 then  put  @1 'State' ;
end;

put @1 rowlab $25. @27 adjrrate92 comma10.2 @37 adjrrate93 comma10.2
    @47 adjrrate94 comma10.2 @57 adjrrate95 comma10.2
    @67 adjrrate96 comma10.2 @77 adjrrate97 comma10.2
    @87 adjrrate98 comma10.2 @97 adjrrate99 comma10.2
    @107 adjrrate00 comma10.2 @117 adjrrate01 comma10.2;
return;
toppage:

PUT #1 @1 measure measfmt. #2 @1 tablenum TITLEfmt. //
@75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001'/;

%mend;

*table(1,1,12); *bene-level,  CC-->Any, table 12;
*table(1,2,13); *bene-level,  CC-->CC,  table 13;
%table(0,1,14); *disch-level, CC-->Any, table 14;
%table(0,2,15); *disch-level, CC-->CC,  table 15;
```

P1H4.@MSQ00AG.PGMLIB(TABLEM6)

```
//P1H4TM6 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=C,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(50,10))
//IN0  DD DSN=P1H4.@MSQ00AG.PGMLIB,DISP=SHR
//IN1  DD DSN=P1H4.@MSQ0000.SAS.Y9201M6.MORTALP,DISP=SHR
//SYSIN DD *
*******************************************************************
* PURPOSE: Outputs the 1992-2001 data on M6 READMISSION RATES
*          in the report format, to be downloaded into excel
*          spreadsheet.
*******************************************************************;

%include IN0(TABM6FMT);
%TABM6FMT;

*****************************************************************
* Read in data that contain adjusted M6 measures by stratifiers
* including PROVIDER region and state
*****************************************************************;

%macro TABLE(b,m,n);

data MORTALP; set in1.MORTALP (where=(bene=&b and measure=&m));

* Change the values of some variables to be the last category for table
  display;
  if race='0' then race='4';
  if preg='0' then preg='5';
  if pst ='0' then pst ='90';

* For 2000 and 2001, rates for longer window period are not available;
if window in (4,5) then adjmrate01=.;
else if window=6 then do; adjmrate00=.; adjmrate01=.; end;

length catgr $2 ;
if total='1'         then do;  cat=1;  catgr=total; end;
else if sex   ne ' ' then do;  cat=2;  catgr=sex;   end;
else if fage  ne ' ' then do;  cat=3;  catgr=fage;  end;
else if race  ne ' ' then do;  cat=4;  catgr=race;  end;
else if dual  ne ' ' then do;  cat=5;  catgr=dual;  end;
else if mdst  ne ' ' then do;  cat=6;  catgr=mdst;  end;
else if urbn  ne ' ' then do;  cat=7;  catgr=urbn;  end;
else if Preg  ne ' ' then do;  cat=8;  catgr=Preg;  end;
else if total='2'    then do;  cat=9;  catgr=total; end;
else if pst   ne ' ' then do;  cat=10; catgr=pst;   end;

* use the cat variables and the catgr variables
to determine the group for a statistic;

LENGTH ROWLAB $25 ;
if cat=1 then rowlab=put(catgr,$totfmt.);
else if cat=2 then rowlab=put(catgr,$sexfmt.);
else if cat=3 then rowlab=put(catgr,$agefmt.);
else if cat=4 then rowlab=put(catgr,$racefmt.);
else if cat=5 then rowlab=put(catgr,$dualfmt.);
else if cat=6 then rowlab=put(catgr,$mcdfmt.);
```

```
else if cat=7 then rowlab=put(catgr,$urbfmt.);
else if cat=8 then rowlab=put(catgr,$cregfmt.);
else if cat=9 then rowlab=put(catgr,$totfmt.);
```
P1H4.@MSQ00AG.PGMLIB(TABLEM6) -- CONTINUED

```
else if cat=10 then rowlab=put(catgr,$stfmt.);

proc sort; by WINDOW cat race preg pst;
RUN;

*****************************************************************
* Output individual tables to the directory named below using the
* following macro program
*****************************************************************;

filename fileout 'P1H4.@msq00M6.M6tables';

data _null_;
set MORTALP; by WINDOW CAT;
tablenum ="&n";
file fileout(table&n) header=toppage ls=132 ps=700  ;

if first.cat then do;
if first.window then put / @1 '' / @1 window winfmt.;
if      cat=1 then  put // @1 ' ' ;
else if cat=2 then  put  @1 'Gender' ;
else if cat=3 then  put  @1 'Age Group' ;
else if cat=4 then  put  @1 'Race' ;
else if cat=5 then  put  @1 'Medicaid Buy-In Status' ;
else if cat=6 then  put  @1 'Medicare Eligibility Status' ;
else if cat=7 then  put  @1 'Urban/Rural' ;
else if cat=8 then  put  @1 'Census Region' ;
else if cat=9 then  put //@1 ' ' ;
else if cat=10 then  put  @1 'State' ;
put /;
end;

put @1 rowlab $25. @27 adjmrate92 comma10.2 @37 adjmrate93 comma10.2
    @47 adjmrate94 comma10.2 @57 adjmrate95 comma10.2
    @67 adjmrate96 comma10.2 @77 adjmrate97 comma10.2
    @87 adjmrate98 comma10.2 @97 adjmrate99 comma10.2
    @107 adjmrate00 comma10.2 @117 adjmrate01 comma10.2;
return;
toppage:

PUT #1 @1 tablenum $tabfmt. //
@75 'Year'/ @1 'Beneficiary Characteristics'
@30 '1992' @40 '1993' @50 '1994' @60 '1995' @70 '1996' @80 '1997'
@90 '1998' @100 '1999' @110 '2000' @120 '2001'/ ;

%mend;

%table(1,1,1b);   * HF, bene-level;
%table(0,1,1d);   * HF, disch-level;

%table(1,12,12b);   * S1, bene-level;
%table(0,12,12d);   * S1, disch-level;
```

```
%table(1,15,15b);   * S4, bene-level;
%table(0,15,15d);   * S4, disch-level;

%table(1,16,16b);   * MI, bene-level;
%table(0,16,16d);   * MI, disch-level;
```

PROGRAMS –

PATIENT SAFETY INDICATORS

PSIB1

```
//TRA0PS JOB (MSQ00000000),'NHK',CLASS=K,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR92,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR93,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR94,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR95,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="96"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR96,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR97,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
```

```
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR98,DISP=SHR
PSIB1 -- CONTINUED

//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR99,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR00,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='3000,1500',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2PRE3.MEDPAR01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSIDAT),
//          DISP=(SHR,PASS,KEEP)
//*
```

PSIDAT

```
OPTIONS COMPRESS=YES;
proc format;
  value $drgmdc
'001'='01'
'002'='01'
'003'='01'
'004'='01'
'005'='01'
'006'='01'
'007'='01'
'008'='01'
'009'='01'
'010'='01'
'011'='01'
'012'='01'
'013'='01'
'014'='01'
'015'='01'
'016'='01'
'017'='01'
'018'='01'
'019'='01'
'020'='01'
'021'='01'
'022'='01'
'023'='01'
'024'='01'
'025'='01'
'026'='01'
'027'='01'
'028'='01'
'029'='01'
'030'='01'
'031'='01'
'032'='01'
'033'='01'
'034'='01'
'035'='01'
'036'='02'
'037'='02'
'038'='02'
'039'='02'
'040'='02'
'041'='02'
'042'='02'
'043'='02'
'044'='02'
'045'='02'
'046'='02'
'047'='02'
'048'='02'
'049'='03'
'050'='03'
'051'='03'
'052'='03'
'053'='03'
```

'054'='03'
'055'='03'
'056'='03'
PSIDAT -- CONTINUED

'057'='03'
'058'='03'
'059'='03'
'060'='03'
'061'='03'
'062'='03'
'063'='03'
'064'='03'
'065'='03'
'066'='03'
'067'='03'
'068'='03'
'069'='03'
'070'='03'
'071'='03'
'072'='03'
'073'='03'
'074'='03'
'075'='04'
'076'='04'
'077'='04'
'078'='04'
'079'='04'
'080'='04'
'081'='04'
'082'='04'
'083'='04'
'084'='04'
'085'='04'
'086'='04'
'087'='04'
'088'='04'
'089'='04'
'090'='04'
'091'='04'
'092'='04'
'093'='04'
'094'='04'
'095'='04'
'096'='04'
'097'='04'
'098'='04'
'099'='04'
'100'='04'
'101'='04'
'102'='04'
'103'='05'
'104'='05'
'105'='05'
'106'='05'
'107'='05'
'108'='05'
'109'='05'

'110'='05'
'111'='05'
'112'='05'
'113'='05'
'114'='05'
'115'='05'
PSIDAT -- CONTINUED

'116'='05'
'117'='05'
'118'='05'
'119'='05'
'120'='05'
'121'='05'
'122'='05'
'123'='05'
'124'='05'
'125'='05'
'126'='05'
'127'='05'
'128'='05'
'129'='05'
'130'='05'
'131'='05'
'132'='05'
'133'='05'
'134'='05'
'135'='05'
'136'='05'
'137'='05'
'138'='05'
'139'='05'
'140'='05'
'141'='05'
'142'='05'
'143'='05'
'144'='05'
'145'='05'
'146'='06'
'147'='06'
'148'='06'
'149'='06'
'150'='06'
'151'='06'
'152'='06'
'153'='06'
'154'='06'
'155'='06'
'156'='06'
'157'='06'
'158'='06'
'159'='06'
'160'='06'
'161'='06'
'162'='06'
'163'='06'
'164'='06'
'165'='06'

'166'='06'
'167'='06'
'168'='03'
'169'='03'
'170'='06'
'171'='06'
'172'='06'
'173'='06'
'174'='06'
PSIDAT -- CONTINUED

'175'='06'
'176'='06'
'177'='06'
'178'='06'
'179'='06'
'180'='06'
'181'='06'
'182'='06'
'183'='06'
'184'='06'
'185'='03'
'186'='03'
'187'='03'
'188'='06'
'189'='06'
'190'='06'
'191'='07'
'192'='07'
'193'='07'
'194'='07'
'195'='07'
'196'='07'
'197'='07'
'198'='07'
'199'='07'
'200'='07'
'201'='07'
'202'='07'
'203'='07'
'204'='07'
'205'='07'
'206'='07'
'207'='07'
'208'='07'
'209'='08'
'210'='08'
'211'='08'
'212'='08'
'213'='08'
'214'='08'
'215'='08'
'216'='08'
'217'='08'
'218'='08'
'219'='08'
'220'='08'
'221'='08'

'222'='08'
'223'='08'
'224'='08'
'225'='08'
'226'='08'
'227'='08'
'228'='08'
'229'='08'
'230'='08'
'231'='08'
'232'='08'
'233'='08'
PSIDAT -- CONTINUED

'234'='08'
'235'='08'
'236'='08'
'237'='08'
'238'='08'
'239'='08'
'240'='08'
'241'='08'
'242'='08'
'243'='08'
'244'='08'
'245'='08'
'246'='08'
'247'='08'
'248'='08'
'249'='08'
'250'='08'
'251'='08'
'252'='08'
'253'='08'
'254'='08'
'255'='08'
'256'='08'
'257'='09'
'258'='09'
'259'='09'
'260'='09'
'261'='09'
'262'='09'
'263'='09'
'264'='09'
'265'='09'
'266'='09'
'267'='09'
'268'='09'
'269'='09'
'270'='09'
'271'='09'
'272'='09'
'273'='09'
'274'='09'
'275'='09'
'276'='09'
'277'='09'

'278'='09'
'279'='09'
'280'='09'
'281'='09'
'282'='09'
'283'='09'
'284'='09'
'285'='10'
'286'='10'
'287'='10'
'288'='10'
'289'='10'
'290'='10'
'291'='10'
'292'='10'

PSIDAT -- CONTINUED

'293'='10'
'294'='10'
'295'='10'
'296'='10'
'297'='10'
'298'='10'
'299'='10'
'300'='10'
'301'='10'
'302'='11'
'303'='11'
'304'='11'
'305'='11'
'306'='11'
'307'='11'
'308'='11'
'309'='11'
'310'='11'
'311'='11'
'312'='11'
'313'='11'
'314'='11'
'315'='11'
'316'='11'
'317'='11'
'318'='11'
'319'='11'
'320'='11'
'321'='11'
'322'='11'
'323'='11'
'324'='11'
'325'='11'
'326'='11'
'327'='11'
'328'='11'
'329'='11'
'330'='11'
'331'='11'
'332'='11'
'333'='11'

'334'='12'
'335'='12'
'336'='12'
'337'='12'
'338'='12'
'339'='12'
'340'='12'
'341'='12'
'342'='12'
'343'='12'
'344'='12'
'345'='12'
'346'='12'
'347'='12'
'348'='12'
'349'='12'
'350'='12'
'351'='12'
PSIDAT -- CONTINUED

'352'='12'
'353'='13'
'354'='13'
'355'='13'
'356'='13'
'357'='13'
'358'='13'
'359'='13'
'360'='13'
'361'='13'
'362'='13'
'363'='13'
'364'='13'
'365'='13'
'366'='13'
'367'='13'
'368'='13'
'369'='13'
'370'='14'
'371'='14'
'372'='14'
'373'='14'
'374'='14'
'375'='14'
'376'='14'
'377'='14'
'378'='14'
'379'='14'
'380'='14'
'381'='14'
'382'='14'
'383'='14'
'384'='14'
'385'='15'
'386'='15'
'387'='15'
'388'='15'
'389'='15'

'390'='15'
'391'='15'
'392'='16'
'393'='16'
'394'='16'
'395'='16'
'396'='16'
'397'='16'
'398'='16'
'399'='16'
'400'='17'
'401'='17'
'402'='17'
'403'='17'
'404'='17'
'405'='17'
'406'='17'
'407'='17'
'408'='17'
'409'='17'
'410'='17'
PSIDAT -- CONTINUED

'411'='17'
'412'='17'
'413'='17'
'414'='17'
'415'='18'
'416'='18'
'417'='18'
'418'='18'
'419'='18'
'420'='18'
'421'='18'
'422'='18'
'423'='18'
'424'='19'
'425'='19'
'426'='19'
'427'='19'
'428'='19'
'429'='19'
'430'='19'
'431'='19'
'432'='19'
'433'='20'
'434'='20'
'435'='20'
'436'='20'
'437'='20'
'438'=' '
'439'='21'
'440'='21'
'441'='21'
'442'='21'
'443'='21'
'444'='21'
'445'='21'

'446'='21'
'447'='21'
'448'='21'
'449'='21'
'450'='21'
'451'='21'
'452'='21'
'453'='21'
'454'='21'
'455'='21'
'456'=' '
'457'=' '
'458'=' '
'459'=' '
'460'=' '
'461'='23'
'462'='23'
'463'='23'
'464'='23'
'465'='23'
'466'='23'
'467'='23'
'468'=' '
'469'=' '

PSIDAT -- CONTINUED

'470'=' '
'471'='08'
'472'=' '
'473'='17'
'474'=' '
'475'='04'
'476'=' '
'477'=' '
'478'='05'
'479'='05'
'480'=' '
'481'=' '
'482'=' '
'483'=' '
'484'='24'
'485'='24'
'486'='24'
'487'='24'
'488'='25'
'489'='25'
'490'='25'
'491'='08'
'492'='17'
'493'='07'
'494'='07'
'495'=' '
'496'='08'
'497'='08'
'498'='08'
'499'='08'
'500'='08'
'501'='08'

```
'502'='08'
'503'='08'
'504'='22'
'505'='22'
'506'='22'
'507'='22'
'508'='22'
'509'='22'
'510'='22'
'511'='22'
;
run;

data out1.PSIdat;
 length pr1-pr6 $4 age sex race asource atype disp mdc pay1
        prday1-prday6 2 sdt1-sdt6 4 los drg 3;
 set in1.medpar&sysparm.(rename=(race=crace sex=csex));
 key=xref;
 race=input(crace,1.);

 if race=0 then race=6;
 else if race=3 then race=6;

 sex=input(csex,1.);

 pay1=6;
```

PSIDAT -- CONTINUED

```
 if primpyr='A' then pay1=3;
 else if primpyr='B' then pay1=3;
 else if primpyr='C' then pay1=6;
 else if primpyr='D' then pay1=3;
 else if primpyr='E' then pay1=6;
 else if primpyr='F' then pay1=6;
 else if primpyr='G' then pay1=3;
 else if primpyr='H' then pay1=6;
 else if primpyr='I' then pay1=6;
 else if primpyr='J' then pay1=3;
 else if primpyr='Z' then pay1=1;
 else if primpyr=' ' then pay1=1;

 hospstco=compress(substr(provid,1,2)!!'000',' ');

 hospid=provid;

 disp=dis_dest;

 if disp=8 then disp=6;
 else if disp=30 then disp=1;

 if admsour='7' then asource=1;
 else if admsour in ('4','A') then asource=2;
 else if admsour in ('5','6') then asource=3;
 else if admsour='8' then asource=4;
 else asource=5;

 atype=input(typeadm,1.);
```

```
 if atype=0 then atype=6;
 else if atype=9 then atype=6;

 drg=drgcode;
 mdc=input(put(put(drgcode,z3.),$drgmdc.),2.);

 dx1=diag1;
 dx2=diag2;
 dx3=diag3;
 dx4=diag4;
 dx5=diag5;
 dx6=diag6;
 dx7=diag7;
 dx8=diag8;
 dx9=diag9;
 dx10=diag10;

 pr1=left(sproc1);
 pr2=left(sproc2);
 pr3=left(sproc3);
 pr4=left(sproc4);
 pr5=left(sproc5);
 pr6=left(sproc6);

 admdt=input(put(admdate,7.),julian7.);

 if sdate1>0 then sdt1=input(put(sdate1,7.),julian7.);
 if sdate2>0 then sdt2=input(put(sdate2,7.),julian7.);
 if sdate3>0 then sdt3=input(put(sdate3,7.),julian7.);
 if sdate4>0 then sdt4=input(put(sdate4,7.),julian7.);
```

PSIDAT -- CONTINUED

```
 if sdate5>0 then sdt5=input(put(sdate5,7.),julian7.);
 if sdate6>0 then sdt6=input(put(sdate6,7.),julian7.);

 if sdt1>.z then prday1=(sdt1-admdt)+1;
 if sdt2>.z then prday2=(sdt2-admdt)+1;
 if sdt3>.z then prday3=(sdt3-admdt)+1;
 if sdt4>.z then prday4=(sdt4-admdt)+1;
 if sdt5>.z then prday5=(sdt5-admdt)+1;
 if sdt6>.z then prday6=(sdt6-admdt)+1;

 keep key age los race sex pay1 hospstco hospid
      disp asource atype drg mdc dx1-dx10 pr1-pr6
      sdt1-sdt6 prday1-prday6;
run;

proc contents position;
run;

proc print data=out1.PSIdat(obs=50);
run;
```

PS1B1

```
//TRA0P1 JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1992"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR92,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1993"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR93,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1994"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR94,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1995"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR95,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1996"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR96,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
```

```
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1997"'
PS1B1 -- CONTINUED

//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR97,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1998"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR98,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1999"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR99,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="2000"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR00,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="2001"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR01,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PSI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PSI(PSIFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS1),
//          DISP=(SHR,PASS,KEEP)
//*
```

PSSAS1

```
  *================================================================;
  *  Title:  PROGRAM 1  ASSIGN AHRQ PATIENT SAFETY INDICATORS
  *          TO INPATIENT RECORDS
  *
  *  Description:
  *         ASSIGNS THE OUTCOME OF INTEREST AND POPULATION AT RISK
  *         INDICATORS TO INPATIENT RECORDS.  ALSO ASSIGNS
  *         COMORBIDITY CATEGORIES.
  *
  *                >>>  VERSION 2.1  <<<
  *
  * USER NOTE:  Make sure you have created the format library
  *              using PSFMTS.SAS BEFORE running this program.
  *
  *================================================================;
  *FILENAME CONTROL 'TRA0.@MSQ0000.PSI(CONTROL)' DISP=SHR;
  *<===USER MUST modify;

%INCLUDE FMTS;
%INCLUDE CONT;

OPTIONS COMPRESS=YES;

 TITLE2 'PROGRAM 1';
 TITLE3 'AHRQ PATIENT SAFETY INDICATORS: ASSIGN PSIS TO INPATIENT DATA';

  * ---------------------------------------------------------------- ;
  * --- PATIENT SAFETY INDICATOR (PSI) NAMING CONVENTION:      --- ;
  * --- THE FIRST LETTER IDENTIFIES THE PATIENT SAFETY         --- ;
  * --- INDICATOR AS ONE OF THE FOLLOWING:
               (O) OBSERVED RATES
               (R) RISK-ADJUSTED RATES
               (S) SMOOTHED RATES
               (T) NUMERATOR ("TOP")
               (P) POPULATION ("POP")
  * --- THE SECOND LETTER IDENTIFIES THE PSI AS A PROVIDER (P) --- ;
  * --- OR AN AREA (A) LEVEL INDICATOR.  THE                   --- ;
  * --- NEXT TWO CHARACTERS ARE ALWAYS 'PS'. THE LAST TWO      --- ;
  * --- DIGITS ARE THE INDICATOR NUMBER (WITHIN THAT SUBTYPE). --- ;
  * ---------------------------------------------------------------- ;

 DATA OUT1.&OUTFILE1.(DROP=DISP PAY1 RACE SEX
                      DX1-DX&NDX.
                      PR1-PR&NPR. SDT1-SDT&NPR.
                      PRDAY1-PRDAY&NPR.
                      I J DXVALUE A1-A35
                      HTN HTNCX HTNCHF HCHFRF HTNRF DMNB
                      CARDFLG PERIFLG CEREFLG PULMFLG DIABFLG
                      HYPOFLG RENALFLG RENFFLG LIVERFLG
                      ULCEFLG HIVFLG LEUKFLG CANCFLG ARTHFLG
                      NUTRFLG ANEMFLG ALCFLG);

   SET IN1.&INFILE1.;

  * ---------------------------------------------------------------- ;
  * --- DIAGNOSIS AND PROCEDURE MACROS ------------------------- ;
```

```
  * ------------------------------------------------------------- ;

 %MACRO MDX(FMT);
```
PSSAS1 -- CONTINUED

```

 (%DO I = 1 %TO &NDX.-1;
  (PUT(DX&I.,&FMT.) = '1') OR
  %END;
  (PUT(DX&NDX.,&FMT.) = '1'))

 %MEND;

 %MACRO MDX1(FMT);

 ((PUT(DX1,&FMT.) = '1'))

 %MEND;

 %MACRO MDX2(FMT);

 (%DO I = 2 %TO &NDX.-1;
  (PUT(DX&I.,&FMT.) = '1') OR
  %END;
  (PUT(DX&NDX.,&FMT.) = '1'))

 %MEND;

 %MACRO MPR(FMT);

 (%DO I = 1 %TO &NPR.-1;
  (PUT(PR&I.,&FMT.) = '1') OR
  %END;
  (PUT(PR&NPR.,&FMT.) = '1'))

 %MEND;

 %MACRO MPR2(FMT,EQ);

 (%DO I = 2 %TO &NPR.-1;
  (PUT(PR&I.,&FMT.) = '1' AND PRDAY&I. NE . AND PRDAY&I. &EQ.) OR
  %END;
  (PUT(PR&NPR.,&FMT.) = '1' AND PRDAY&I. &EQ.))

 %MEND;

 %MACRO MPR2B(FMT);

 (%DO I = 2 %TO &NPR.-1;
  (PUT(PR&I.,&FMT.) = '1') OR
  %END;
  (PUT(PR&NPR.,&FMT.) = '1'))

 %MEND;

 %MACRO MDR(FMT);

 (PUT(PUT(DRG,Z3.),&FMT.) = '1')
```

```
%MEND;

  * ------------------------------------------------------------- ;
  * --- DEFINE COMORBIDITY CATEGORIES --------------------------- ;
  * ------------------------------------------------------------- ;
```

PSSAS1 -- CONTINUED

```
ATTRIB
CHF      LENGTH=3
  LABEL='CONGESTIVE HEART FAILURE'
ARYTHM   LENGTH=3
  LABEL='ARRHYTHMIAS'
VALVE    LENGTH=3
  LABEL='VALVULAR DISEASE'
PULMCIRC LENGTH=3
  LABEL='PULMONARY CIRCULATION DISEASE'
PERIVASC LENGTH=3
  LABEL='PERIPHERAL VASCULAR DISEASE'
HTN_C    LENGTH=3
  LABEL='HYPERTENSION'
PARA     LENGTH=3
  LABEL='PARALYSIS'
NEURO    LENGTH=3
  LABEL='OTHER NEUROLOGICAL DISORDERS'
CHRNLUNG LENGTH=3
  LABEL='CHRONIC PULMONARY DISEASE'
DM       LENGTH=3
  LABEL='DIABETES'
DMCX     LENGTH=3
  LABEL='DIABETES W CHRONIC COMPLICATIONS'
HYPOTHY  LENGTH=3
  LABEL='HYPOTHYROIDISM'
RENLFAIL LENGTH=3
  LABEL='RENAL FAILURE'
LIVER    LENGTH=3
  LABEL='LIVER DISEASE'
ULCER    LENGTH=3
  LABEL='PEPTIC ULCER DISEASE X BLEEDING'
AIDS     LENGTH=3
  LABEL='ACQUIRED IMMUNE DEFICIENCY SYNDROME'
LYMPH    LENGTH=3
  LABEL='LYMPHOMA'
METS     LENGTH=3
  LABEL='METASTATIC CANCER'
TUMOR    LENGTH=3
  LABEL='SOLID TUMOR WOUT METASTASIS'
ARTH     LENGTH=3
  LABEL='RHEUMATOID ARTHRITIS COLLAGEN VAS'
COAG     LENGTH=3
  LABEL='COAGULOPTHY'
OBESE    LENGTH=3
  LABEL='OBESITY'
WGHTLOSS LENGTH=3
  LABEL='WEIGHT LOSS'
LYTES    LENGTH=3
  LABEL='FLUID AND ELECTROLYTE DISORDERS'
```

```
 BLDLOSS  LENGTH=3
   LABEL='CHRONIC BLOOD LOSS ANEMIA'
 ANEMDEF  LENGTH=3
   LABEL='DEFICIENCY ANEMIAS'
 ALCOHOL  LENGTH=3
   LABEL='ALCOHOL ABUSE'
 DRUG     LENGTH=3
   LABEL='DRUG ABUSE'
 PSYCH    LENGTH=3
```
PSSAS1 -- CONTINUED
```
   LABEL='PSYCHOSES'
 DEPRESS  LENGTH=3
   LABEL='DEPRESSION'
;

 LENGTH A1  A2  A3  A4  A5  A6  A7  A8  A9  A10
        A11 A12 A13 A14 A15 A16 A17 A18 A19 A20
        A21 A22 A23 A24 A25 A26 A27 A28 A29 A30
        A31 A32 A33 A34 A35 DXVALUE $8
;

  * ------------------------------------------------------------- ;
  * --- CREATION OF COMORBIDITY VARIABLES                       ;
  * ---                                                         ;
  * --- Creates comorbidity variables based on the              ;
  * --- presence of secondary diagnoses and redefines           ;
  * --- comorbidity group by eliminating DRGs directly          ;
  * --- related to them.                                        ;
  * -------------------------------------------------------------;

  * ------------------------------------------------------------- ;
  * ---  Declare variables as array elements                    ;
  * ------------------------------------------------------------- ;

   ARRAY ARRY1{&NDX.}  DX1 - DX&NDX.;

   ARRAY ARRY2{35}  CHF      HTNCHF   HCHFRF   ARYTHM   VALVE
                    PULMCIRC PERIVASC HTN      HTNCX    PARA
                    NEURO    CHRNLUNG DM       DMCX     DMNB
                    HYPOTHY  RENLFAIL HTNRF    LIVER    ULCER
                    AIDS     LYMPH    METS     TUMOR    ARTH
                    COAG     OBESE    WGHTLOSS LYTES    BLDLOSS
                    ANEMDEF  ALCOHOL  DRUG     PSYCH    DEPRESS ;

   ARRAY ARRY3{35}  A1  A2  A3  A4  A5  A6  A7  A8  A9  A10
                    A11 A12 A13 A14 A15 A16 A17 A18 A19 A20
                    A21 A22 A23 A24 A25 A26 A27 A28 A29 A30
                    A31 A32 A33 A34 A35 ;


  * ------------------------------------------------------------- ;
  * ---  Initialize ARRY2 to 0 and assigns the variable         ;
  * ---  name from ARRY2 as the VALUE of ARRY3                  ;
  * ------------------------------------------------------------- ;

   DO I = 1 TO 35;
      ARRY2(I) = 0;
```

```
      CALL VNAME(ARRY2(I),ARRY3(I));
   END;

  * ------------------------------------------------------------ ;
  * ---  Looking at the secondary DXs and using formats,         ;
  * ---  create DXVALUE to define each comorbidity group         ;
  * ---                                                          ;
  * ---  If DXVALUE is equal to the comorbidity name in          ;
  * ---  array ARRY3 then a value of 1 is assigned to the        ;
  * ---  corresponding comorbidity group in array ARRY2          ;
  * ------------------------------------------------------------ ;
```

PSSAS1 -- CONTINUED

```
   DO I = 2 TO &NDX.;
      DXVALUE = PUT(ARRY1(I),$RCOMFMT.) ;
      DO J = 1 TO 35;
         IF DXVALUE = ARRY3(J) THEN ARRY2(J) = 1;
      END;
   END;

  * ------------------------------------------------------------ ;
  * ---  SET UP CODE TO ONLY COUNT THE MORE SEVERE COMORBIDITY   ;
  * ------------------------------------------------------------ ;

   IF HTNCHF = 1 THEN DO; CHF = 1; HTNCX = 1; END;
   IF HCHFRF = 1 THEN DO; CHF = 1; RENLFAIL = 1; HTNCX = 1; END;
   IF HTNRF = 1 THEN DO; RENLFAIL = 1; HTNCX = 1; END;
   IF DMNB = 1 AND AGE < 1 THEN DMCX = 1;
   IF HTNCX = 1 THEN HTN = 0 ;
   IF METS = 1 THEN TUMOR = 0 ;
   IF DMCX = 1 THEN DM = 0;

  * ------------------------------------------------------------ ;
  * ---  EXAMINE DRG AND SET FLAGS TO IDENTIFY A PARTICULAR      ;
  * ---  DRG GROUP                                               ;
  * ------------------------------------------------------------ ;

   IF PUT(DRG,CARDDRG.) = 'YES' THEN CARDFLG = 1;
   IF PUT(DRG,PERIDRG.) = 'YES' THEN PERIFLG = 1;
   IF PUT(DRG,CEREDRG.) = 'YES' THEN CEREFLG = 1;
   IF PUT(DRG,PULMDRG.) = 'YES' THEN PULMFLG = 1;
   IF PUT(DRG,DIABDRG.) = 'YES' THEN DIABFLG = 1;
   IF PUT(DRG,HYPODRG.) = 'YES' THEN HYPOFLG = 1;
   IF PUT(DRG,RENALDRG.) = 'YES' THEN RENALFLG = 1;
   IF PUT(DRG,RENFDRG.) = 'YES' THEN RENFFLG = 1;
   IF PUT(DRG,LIVERDRG.) = 'YES' THEN LIVERFLG = 1;
   IF PUT(DRG,ULCEDRG.) = 'YES' THEN ULCEFLG = 1;
   IF PUT(DRG,HIVDRG.) = 'YES' THEN HIVFLG = 1;
   IF PUT(DRG,LEUKDRG.) = 'YES' THEN LEUKFLG = 1;
   IF PUT(DRG,CANCDRG.) = 'YES' THEN CANCFLG = 1;
   IF PUT(DRG,ARTHDRG.) = 'YES' THEN ARTHFLG = 1;
   IF PUT(DRG,NUTRDRG.) = 'YES' THEN NUTRFLG = 1;
   IF PUT(DRG,ANEMDRG.) = 'YES' THEN ANEMFLG = 1;
   IF PUT(DRG,ALCDRG.) = 'YES' THEN ALCFLG = 1;


   * ------------------------------------------------------------ ;
```

```
   * ---  REDEFINING COMORBIDITIES BY ELIMINATING THE DRG DIRECTLY ;
   * ---  RELATED TO COMORBIDITY, THUS LIMITING THE SCREENS TO     ;
   * ---  PRINCIPAL DIAGNOSES NOT DIRECTLY RELATED TO COMORBIDITY  ;
   * ---  IN QUESTION                                              ;
   * ------------------------------------------------------------ ;

   IF CHF AND CARDFLG  THEN    CHF = 0 ;
   IF ARYTHM AND CARDFLG  THEN   ARYTHM = 0;
   IF VALVE AND CARDFLG  THEN  VALVE = 0;
   IF PULMCIRC AND ( CARDFLG OR DRG = 88 ) THEN PULMCIRC = 0;
   IF PERIVASC AND PERIFLG THEN PERIVASC = 0;
   IF HTN AND DRG =134  THEN HTN = 0;
   IF HTNCX AND ( CARDFLG OR RENALFLG OR DRG = 134 ) THEN HTNCX = 0  ;
   IF NEURO AND MDC = 1  THEN   NEURO = 0;
   IF CHRNLUNG AND PULMFLG THEN  CHRNLUNG = 0;
```

PSSAS1 -- CONTINUED

```
   IF DM AND DIABFLG THEN   DM = 0;
   IF DMCX AND DIABFLG THEN  DMCX = 0 ;
   IF HYPOTHY AND HYPOFLG THEN HYPOTHY = 0;
   IF RENLFAIL AND RENFFLG THEN   RENLFAIL = 0;
   IF LIVER AND LIVERFLG  THEN LIVER = 0;
   IF ULCER AND ULCEFLG THEN  ULCER = 0;
   IF AIDS AND HIVFLG THEN  AIDS = 0;
   IF LYMPH AND LEUKFLG  THEN  LYMPH = 0;
   IF METS AND CANCFLG  THEN  METS = 0;
   IF TUMOR AND CANCFLG  THEN  TUMOR = 0;
   IF ARTH AND ARTHFLG THEN ARTH = 0;
   IF COAG AND DRG = 397 THEN  COAG = 0;
   IF OBESE AND (NUTRFLG OR DRG = 288 )  THEN  OBESE = 0;
   IF WGHTLOSS AND NUTRFLG THEN WGHTLOSS = 0;
   IF LYTES AND NUTRFLG THEN LYTES = 0;
   IF BLDLOSS AND ANEMFLG THEN BLDLOSS = 0;
   IF ANEMDEF AND ANEMFLG THEN  ANEMDEF = 0;
   IF ALCOHOL AND ALCFLG THEN  ALCOHOL = 0;
   IF DRUG AND ALCFLG THEN DRUG = 0;
   IF PSYCH AND DRG = 430 THEN PSYCH = 0;
   IF DEPRESS AND DRG = 426 THEN DEPRESS = 0;

   * ------------------------------------------------------------ ;
   * --- Combine HTN and HTNCX into HTN_C                         ;
   * ------------------------------------------------------------ ;
   IF HTN=1 OR HTNCX=1 THEN HTN_C=1;
     ELSE HTN_C=0;


  * ------------------------------------------------------------ ;
  * --- DELETE RECORDS WITH MISSING VALUES FOR AGE OR SEX--------- ;
  * ------------------------------------------------------------ ;
 IF SEX LE 0 THEN DELETE;
 IF AGE LT 0 THEN DELETE;

  * ------------------------------------------------------------ ;
  * --- DEFINE GEOGRAPHIC AREAS ---------------------------------- ;
  * ------------------------------------------------------------ ;
 ATTRIB AREA LENGTH=$5
   LABEL='AREA ASSIGNED FROM HOSPSTCO';
```

```
 *AREA = INPUT( PUT(HOSPSTCO,AREA.), $5.);
AREA = HOSPSTCO;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: PAYER CATEGORY ----------------------- ;
 * ------------------------------------------------------------- ;
ATTRIB PAYCAT LENGTH=$1
  LABEL='(M)edicar Me(D)icaid (P)riv (S)elf (O)th';

SELECT (PAY1);
  WHEN (1)  PAYCAT = 'M';
  WHEN (2)  PAYCAT = 'D';
  WHEN (3)  PAYCAT = 'P';
  WHEN (4)  PAYCAT = 'S';
  OTHERWISE PAYCAT = 'O';
END; * SELECT PAY1 ;

 * ------------------------------------------------------------- ;
```

PSSAS1 -- CONTINUED

```
 * --- DEFINE STRATIFIER: RACE CATEGORY ------------------------ ;
 * ------------------------------------------------------------- ;
ATTRIB RACECAT LENGTH=$1
  LABEL='(W)hite (B)lack (H)ispanic (O)ther';

SELECT (RACE);
  WHEN (1)  RACECAT = 'W';
  WHEN (2)  RACECAT = 'B';
  WHEN (3)  RACECAT = 'H';
  OTHERWISE RACECAT = 'O';
END; * SELECT RACE ;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: AGE CATEGORY  ------------------------ ;
 * ------------------------------------------------------------- ;
ATTRIB AGECAT LENGTH=$1
  LABEL='AGE CATEGORIES';

SELECT;
  WHEN (0  <= AGE < 55)  AGECAT = '1';
  WHEN (55 <= AGE < 65)  AGECAT = '2';
  WHEN (65 <= AGE < 70)  AGECAT = '3';
  WHEN (70 <= AGE < 75)  AGECAT = '4';
  WHEN (75 <= AGE < 80)  AGECAT = '5';
  WHEN (80 <= AGE < 85)  AGECAT = '6';
  WHEN (85 <= AGE < 90)  AGECAT = '7';
  WHEN (90 <= AGE < 95)  AGECAT = '8';
  WHEN (95 <= AGE     )  AGECAT = '9';
END; * SELECT AGE ;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: SEX CATEGORY  ------------------------ ;
 * ------------------------------------------------------------- ;
ATTRIB SEXCAT LENGTH=$1
  LABEL='(M)ale (F)emale';

SELECT (SEX);
  WHEN (1)  SEXCAT = 'M';
```

```
  WHEN (2)  SEXCAT = 'F';
END; * SELECT SEX ;

 * ------------------------------------------------------------- ;
 * --- DEFINE PROVIDER LEVEL INDICATORS ----------------------- ;
 * ------------------------------------------------------------- ;

%MACRO LBL;

ATTRIB
TPPS01 LENGTH=8
  LABEL='COMPLICATIONS OF ANESTHESIA  (Numerator)'
TPPS02 LENGTH=8
  LABEL='DEATH IN LOW MORTALITY DRGS  (Numerator)'
TPPS02A LENGTH=8
  LABEL='DEATH IN LMD - ADULT MEDICAL (Numerator)'
TPPS02B LENGTH=8
  LABEL='DEATH IN LMD - PED MEDICAL   (Numerator)'
TPPS02C LENGTH=8
  LABEL='DEATH IN LMD - ADLT SURGICAL (Numerator)'
TPPS02D LENGTH=8
```



```
  LABEL='DEATH IN LMD - PED SURGICAL  (Numerator)'
TPPS02E LENGTH=8
  LABEL='DEATH IN LMD - OBSTETRIC     (Numerator)'
TPPS02F LENGTH=8
  LABEL='DEATH IN LMD - NEONATAL      (Numerator)'
TPPS02G LENGTH=8
  LABEL='DEATH IN LMD - PSYCHIATRIC   (Numerator)'
TPPS03 LENGTH=8
  LABEL='DECUBITUS ULCER              (Numerator)'
TPPS04 LENGTH=8
  LABEL='FAILURE TO RESCUE            (Numerator)'
TPPS05 LENGTH=8
  LABEL='FOREIGN BODY LEFT IN DURING PROC (Numer)'
TPPS06 LENGTH=8
  LABEL='IATROGENIC PNEUMOTHORAX      (Numerator)'
TPPS07 LENGTH=8
  LABEL='INFECTION DUE TO MEDICAL CARE (Numeratr)'
TPPS08 LENGTH=8
  LABEL='POSTOPERATIVE HIP FRACTURE   (Numerator)'
%IF &PRDAY. = 1 %THEN %DO;
TPPS09 LENGTH=8
  LABEL='POSTOP HEMORRHAGE OR HEMATOMA (Numeratr)'
%END;
%ELSE %DO;
TPPS09 LENGTH=8
  LABEL='POSTOP HEMOR OR HEMAT-NO PRDAY (Numertr)'
%END;
TPPS10 LENGTH=8
  LABEL='POSTOP PHYSIO METABOL DERANGMNT (Numrtr)'
TPPS11 LENGTH=8
  LABEL='POSTOP RESPIRATORY FAILURE   (Numerator)'
%IF &PRDAY. = 1 %THEN %DO;
TPPS12 LENGTH=8
  LABEL='POSTOPERATIVE PE OR DVT      (Numerator)'
%END;
```

```
 %ELSE %DO;
 TPPS12 LENGTH=8
   LABEL='POSTOP PE OR DVT-NO PRDAY    (Numerator)'
 %END;
 TPPS13 LENGTH=8
   LABEL='POSTOPERATIVE SEPSIS         (Numerator)'
 TPPS14 LENGTH=8
   LABEL='POSTOPERATIVE WOUND DEHISCENCE (Numertr)'
 TPPS15 LENGTH=8
   LABEL='TECH DIFFICULTY W PROCEDURE  (Numerator)'
 TPPS16 LENGTH=8
   LABEL='TRANSFUSION REACTION         (Numerator)'
 TPPS17 LENGTH=8
   LABEL='BIRTH TRAUMA                 (Numerator)'
 TPPS18 LENGTH=8
   LABEL='OB TRAUMA - VAGINAL W INSTRUMENT (Numer)'
 TPPS19 LENGTH=8
   LABEL='OB TRAUMA - VAGINAL WO INSTRUMENT (Numr)'
 TPPS20 LENGTH=8
   LABEL='OB TRAUMA - C-SECTION        (Numerator)'
;

  * ------------------------------------------------------------- ;
  * --- DEFINE AREA LEVEL INDICATORS ---------------------------- ;
```

PSSAS1 -- CONTINUED

```
  * ------------------------------------------------------------- ;
 ATTRIB
 TAPS21 LENGTH=3
   LABEL='FOREIGN BODY LEFT IN DURING PROC (Numer)'
 TAPS22 LENGTH=3
   LABEL='IATROGENIC PNEUMOTHORAX      (Numerator)'
 TAPS23 LENGTH=3
   LABEL='INFECTION DUE TO MEDICAL CARE (Numeratr)'
 TAPS24 LENGTH=3
   LABEL='POSTOPERATIVE WOUND DEHISCENCE (Numertr)'
 TAPS25 LENGTH=3
   LABEL='TECH DIFFICULTY W PROCEDURE  (Numerator)'
 TAPS26 LENGTH=3
   LABEL='TRANSFUSION REACTION         (Numerator)'
;

 %MEND;

 %LBL;


  * ------------------------------------------------------------- ;
  * --- CONSTRUCT PROVIDER LEVEL INDICATORS --------------------- ;
  * ------------------------------------------------------------- ;

   * --- COMPLICATIONS OF ANESTHESIA                          --- ;

   IF %MDR($SURGIDR.) THEN DO;

      TPPS01 = 0;

      IF %MDX2($ANES1ID.) OR %MDX2($ANES2ID.) THEN TPPS01 = 1;
```

```
      *** Exclude Active Drug Dependence, Active nondependent abuse
          of drugs, and self-inflicted injury;

      IF %MDX2($ANES1ID.) AND
         (%MDX($ADRUGID.) OR %MDX($ANONDID.) OR %MDX($SELFIID.))
      THEN TPPS01 = .;

   END;

   * --- DEATH IN LOW MORTALITY DRGS                          --- ;

   IF %MDR($LOWMODR.) THEN DO;

      TPPS02 = 0;

      IF DISP IN (20) THEN TPPS02 = 1;

      *** Exclude Trauma, Immunocompomised state, and cancer;

      IF %MDX($TRAUMID.) OR %MDR($TRAUMDR.) OR
         %MDX($IMMUNID.) OR %MPR($IMMUNIP.) OR %MDR($IMMUNDR.) OR
         %MDX($CANCEID.) OR %MDR($CANCEDR.)
      THEN TPPS02 = .;

      SELECT (PUT(PUT(DRG,Z3.),$LOWM2DR.));
        WHEN ('M')  TPPS02A = TPPS02;
```

PSSAS1 -- CONTINUED

```
        WHEN ('PM') TPPS02B = TPPS02;
        WHEN ('S')  TPPS02C = TPPS02;
        WHEN ('PS') TPPS02D = TPPS02;
        WHEN ('O')  TPPS02E = TPPS02;
        WHEN ('N')  TPPS02F = TPPS02;
        WHEN ('Y')  TPPS02G = TPPS02;
      END; * SELECT DRG ;

   END;

   * --- DECUBITUS ULCER                                      --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TPPS03=0;

      IF %MDX2($DECUBID.) THEN TPPS03 = 1;

      *** Exclude Hemiplegia, Paraplagia, Quadriplagia;

      IF %MDX($HEMIPID.) THEN TPPS03 = .;

      *** Exclude MDC 9;

      IF MDC IN (9) THEN TPPS03 = .;

      *** Exclude LOS < 5, Admitted from LTC;

      IF LOS < 5 OR ASOURCE IN (3) THEN TPPS03 = .;
```

```
   END;

   * --- FAILURE TO RESCUE                                        --- ;

   IF (%MDX2($FTR1DX.) AND
       NOT (%MDX1($AMIID.) OR %MDX1($CARDRID.) OR %MDX1($CARDIID.) OR
            %MDX1($HEMORID.) OR %MDX1($SHOCKID.)))
      OR
      (%MDX2($FTR2DX.) AND
       NOT (%MDX1($DEEPVID.)))
      OR
      (%MDX2($FTR3DX.) AND
       NOT (%MDX1($FTR3EXA.) OR %MDX2($FTR3EXB.) OR %MDX($IMMUNID.) OR
            %MDR($IMMUNDR.) OR %MPR($IMMUNIP.) OR MDC IN (4)))
      OR
      (%MDX2($FTR4DX.) AND
       NOT (%MDX($IMMUNID.) OR %MDR($IMMUNDR.) OR %MPR($IMMUNIP.) OR
            %MDX1($INFECID.) OR %MDR($INFECDR.) OR LOS > 3))
      OR
      ((%MDX2($FTR5DX.) OR %MPR($FTR5PR.)) AND
       NOT (%MDX1($HEMORID.) OR %MDX1($TRAUMID.) OR %MDR($TRAUMDR.) OR
            MDC IN (4,5)))
      OR
      (%MDX2($FTR6DX.) AND
       NOT (%MDX1($FTR6EX.) OR %MDX1($TRAUMID.) OR %MDR($TRAUMDR.) OR
            %MDX1($DRUGAID.) OR MDC IN (6,7)))
   THEN DO;

      TPPS04 = 0;
```

PSSAS1 -- CONTINUED

```
      IF DISP IN (20) THEN TPPS04 = 1;

      *** Exclude Transfers from an acute care or LTC facility
          or Transferred to an acute care facility;

      IF ASOURCE IN (2,3) OR DISP IN (2) THEN TPPS04 = .;

   END;

   * --- FOREIGN BODY LEFT IN DURING PROCEDURE                    --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TPPS05 = 0;

      IF %MDX2($FOREIID.) THEN TPPS05=1;

   END;

   * --- IATROGENIC PNEUMOTHORAX                                  --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TPPS06=0;
```

```
      IF %MDX2($IATROID.) THEN TPPS06=1;

      *** Exclude Trauma, Thoracic surgery, Lung or pleural biopsy,
          or Cardiac surgery;

      IF %MDX($TRAUMID.) OR %MDR($TRAUMDR.) OR %MPR($THORAIP.) OR
         %MPR($LUNGBIP.) OR %MDR($CARDSDR.)
      THEN TPPS06 = .;

   END;

   * --- INFECTION DUE TO MEDICAL CARE                          --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TPPS07=0;

      IF %MDX2($IDTMCID.) THEN TPPS07 = 1;

      *** Exclude Immunocompromised state and Cancer;

      IF %MDX($IMMUNID.) OR %MDR($IMMUNDR.) OR %MPR($IMMUNIP.) OR
         %MDX($CANCEID.) OR %MDR($CANCEDR.)
      THEN TPPS07 = .;

   END;

   * --- POSTOPERATIVE HIP FRACTURE                             --- ;

   IF %MDR($SURGIDR.) AND AGE >= 18 THEN DO;

      TPPS08 = 0;
```

PSSAS1 -- CONTINUED

```
      IF %MDX2($HIPFXID.) THEN TPPS08 = 1;

      *** Excludes Seizure, Syncope, Stroke, Coma, Cardiac arrest,
          Poisoning, Trauma, Delirium and other psychoses, Anoxic
          brain injury;

      IF %MDX1($SEIZUID.) OR %MDX1($SYNCOID.) OR %MDX1($STROKID.) OR
         %MDX1($COMAID.)  OR %MDX1($CARDIID.) OR %MDX1($POISOID.) OR
         %MDX1($TRAUMID.) OR %MDR($TRAUMDR.)  OR %MDX1($DELIRID.) OR
         %MDX1($ANOXIID.)
      THEN TPPS08 = .;

      *** Excludes Metastatic cancer, Lymphoid malignancy,
          Bone malignancy, Self-inflicted injury;

      IF %MDX($METACID.) OR %MDX($LYMPHID.) OR %MDX($BONEMID.) OR
         %MDX($SELFIID.)
      THEN TPPS08 = .;

      *** Exclude MDC 8;

      IF MDC IN (8) THEN TPPS08 = .;
```

```
  END;

  * --- POSTOPERATIVE HEMORRHAGE OR HEMATOMA                --- ;

%MACRO PS9;

%IF (&PRDAY. = 1 AND PRDAY1 NE .) %THEN %DO;

  IF %MDR($SURGIDR.) THEN DO;

     TPPS09 = 0;

     IF (%MDX2($POHMAID.) AND %MPR2($HEMATIP.,>= PRDAY1)) OR
        (%MDX2($POHMRID.) AND %MPR2($HEMORIP.,>= PRDAY1))
     THEN TPPS09 = 1;

     *** Exclude MDC 14, 15;

     IF MDC IN (14,15) THEN TPPS09 = .;

  END;

%END;
%ELSE %DO;

  IF %MDR($SURGIDR.) THEN DO;

     TPPS09 = 0;

     IF (%MDX2($POHMAID.) AND %MPR2B($HEMATIP.)) OR
        (%MDX2($POHMRID.) AND %MPR2B($HEMORIP.))
     THEN TPPS09 = 1;

     *** Exclude MDC 14, 15;

     IF MDC IN (14,15) THEN TPPS09 = .;
```

PSSAS1 -- CONTINUED

```
  END;

%END;

%MEND;

%PS9;

  * --- POSTOPERATIVE PHYSIOLOGIC AND METABOLIC DERANGEMENTS --- ;

  IF %MDR($SURGIDR.) AND ATYPE IN (3) THEN DO;

     TPPS10 = 0;

     IF %MDX2($PHYSIDA.) OR (%MDX2($PHYSIDB.) AND %MPR($DIALYIP.))
     THEN TPPS10 = 1;

     *** Exclude Ketoacidosis, Hyperosmolarity or other Coma and
         Diabetes;
```

```
      IF %MDX2($PHYSIDA.) AND %MDX1($DIABEID.) THEN TPPS10 = .;

      *** Exclude Acute renal failure and AMI, Cardiac arrhythmia,
          Cardiac arrest, Shock, Hemorrhage, or Gastrointestinal
          hemorrhage;

      IF %MDX2($PHYSIDB.) AND (%MDX1($AMIID.) OR %MDX1($CARDRID.) OR
         %MDR($CARDRDR.) OR %MDX1($CARDIID.) OR %MDX1($SHOCKID.) OR
         %MDX1($HEMORID.) OR %MDX1($GASTRID.))
      THEN TPPS10 = .;

      *** Exclude MDC 14, 15;

      IF MDC IN (14,15) THEN TPPS10 = .;

   END;

   * --- POSTOPERATIVE RESPIRATORY FAILURE                    --- ;

   IF %MDR($SURGIDR.) AND ATYPE IN (3) THEN DO;

      TPPS11 = 0;

      IF %MDX2($ACURFID.) THEN TPPS11 = 1;

      *** Exclude MDC 4, 5;

      IF MDC IN (4,5) THEN TPPS11 = .;

      *** Exclude MDC 14,15;

      IF MDC IN (14,15) THEN TPPS11 = .;

   END;

   * --- POSTOPERATIVE PE OR DVT                              --- ;

   IF %MDR($SURGIDR.) THEN DO;
```

PSSAS1 -- CONTINUED

```
      TPPS12 = 0;

      IF %MDX2($DEEPVID.) OR %MDX2($PULMOID.) THEN TPPS12 = 1;

      *** Exclude Deep vein thrombosis;

      IF %MDX1($DEEPVID.) THEN TPPS12 = .;

      *** Exclude Interruption of vena cava;

 %MACRO PS12;

 %IF (&PRDAY. = 1  AND PRDAY1 NE .) %THEN %DO;

      IF %MPR2($VENACIP.,<= PRDAY1) THEN TPPS12 = .;
```

```
%END;
%ELSE %DO;

     IF %MPR2B($VENACIP.) THEN TPPS12 = .;
%END;

%MEND;

%PS12;

     *** Exclude MDC 14,15;

     IF MDC IN (14,15) THEN TPPS12 = .;

  END;

  * --- POSTOPERATIVE SEPSIS                                    --- ;

  IF %MDR($SURGIDR.) AND ATYPE IN (3) THEN DO;

     TPPS13=0;

     IF %MDX2($SEPTIID.) THEN TPPS13 = 1;

     *** Exclude Infection, Immunocompromised or Cancer;

     IF %MDX1($INFECID.) OR %MDR($INFECDR.) OR %MDX($IMMUNID.)  OR
        %MDR($IMMUNDR.)  OR %MPR($IMMUNIP.) OR %MDX($CANCEID.) OR
        %MDR($CANCEDR.)
     THEN TPPS13 = .;

     *** Exclude Length of stay less then 4 days;

     IF LOS < 4 THEN TPPS13 = .;

     *** Exclude MDC 14,15;

     IF MDC IN (14,15) THEN TPPS13 = .;

  END;

  * --- POSTOPERATIVE WOUND DEHISCENCE                          --- ;
```

PSSAS1 -- CONTINUED

```
  IF %MPR($ABDOMIP.) THEN DO;

     TPPS14 = 0;

     IF %MPR2B($RECLOIP.) THEN TPPS14 = 1;

     *** Exclude MDC 14,15;

     IF MDC IN (14,15) THEN TPPS14 = .;

  END;

  * --- TECH DIFFICULTY WITH PROCEDURE                          --- ;
```

```
      IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;
         TPPS15 = 0;
         IF %MDX2($TECHNID.) THEN TPPS15 = 1;
         *** Exclude MDC 14,15;
         IF MDC IN (14,15) THEN TPPS15 = .;
      END;
      * --- TRANSFUSION REACTION                                  --- ;
      IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;
         TPPS16 = 0;
         IF %MDX2($TRANFID.) THEN TPPS16 = 1;
      END;
      * --- BIRTH TRAUMA                                          --- ;
      IF (%MDX($LIVEBID.) OR %MDR($LIVEBDR.)) AND ATYPE IN (4) THEN DO;
         TPPS17 = 0;
         IF %MDX($BIRTIDA.) OR %MDX($BIRTIDB.) OR %MDX($BIRTIDC.) THEN TPPS17
= 1;
         *** Exclude Subdural or cerebral hemmorrhage and preterm;
         IF %MDX($BIRTIDA.) AND %MDX($PRETEID.) THEN TPPS17 = .;
         *** Exclude Injury to skeleton and osteogenesis imperfecta;
         IF %MDX($BIRTIDB.) AND %MDX($OSTEOID.) THEN TPPS17 = .;
      END;
      * --- OB TRAUMA - VAGINAL WITH INSTRUMENTATION             --- ;
      IF %MDR($VAGINDR.) AND %MPR($INSTRIP.) THEN DO;
```

PSSAS1 -- CONTINUED

```
         TPPS18 = 0;
         IF %MDX($OBTRAID.) OR %MPR($OBTRAIP.) THEN TPPS18 = 1;
      END;
      * --- OB TRAUMA - VAGINAL WITHOUT INSTRUMENTATION          --- ;
      IF %MDR($VAGINDR.) AND NOT %MPR($INSTRIP.) THEN DO;
```

```
      TPPS19 = 0;

      IF %MDX($OBTRAID.) OR %MPR($OBTRAIP.) THEN TPPS19 = 1;

   END;

   * --- OB TRAUMA - C-SECTION                                  --- ;

   IF %MDR($CESARDR.) THEN DO;

      TPPS20 = 0;

      IF %MDX($OBTRAID.) OR %MPR($OBTRAIP.) THEN TPPS20 = 1;

   END;



  * ------------------------------------------------------------- ;
  * --- CONSTRUCT AREA LEVEL INDICATORS ------------------------- ;
  * ------------------------------------------------------------- ;

   * --- FOREIGN BODY LEFT IN DURING PROCEDURE                --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TAPS21 = 0;

      IF %MDX($FOREIID.) THEN TAPS21=1;

   END;

   * --- IATROGENIC PNEUMOTHORAX                              --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TAPS22=0;

      IF %MDX($IATROID.) THEN TAPS22=1;

      *** Exclude Trauma, Thoracic surgery, Lung or pleural biopsy,
          or Cardiac surgery;

      IF %MDX($TRAUMID.) OR %MDR($TRAUMDR.) OR %MPR($THORAIP.) OR
         %MPR($LUNGBIP.) OR %MDR($CARDSDR.)
      THEN TAPS22 = .;

   END;
```

PSSAS1 -- CONTINUED

```
   * --- INFECTION DUE TO MEDICAL CARE                        --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TAPS23=0;

      IF %MDX($IDTMCID.) THEN TAPS23 = 1;
```

```
      *** Exclude Immunocompromised state and Cancer;

      IF %MDX($IMMUNID.) OR %MDR($IMMUNDR.) OR %MPR($IMMUNIP.) OR
         %MDX($CANCEID.) OR %MDR($CANCEDR.)
      THEN TAPS23 = .;

   END;

   * --- POSTOPERATIVE WOUND DEHISCENCE                        --- ;

   IF %MPR($ABDOMIP.) THEN DO;

      TAPS24 = 0;

      IF %MPR($RECLOIP.) THEN TAPS24 = 1;

      *** Exclude MDC 14,15;

      IF MDC IN (14,15) THEN TAPS24 = .;

   END;

   * --- TECH DIFFICULTY WITH PROCEDURE                        --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TAPS25 = 0;

      IF %MDX($TECHNID.) THEN TAPS25 = 1;

      *** Exclude MDC 14,15;

      IF MDC IN (14,15) THEN TAPS25 = .;

   END;

   * --- TRANSFUSION REACTION                                  --- ;

   IF %MDR($MEDICDR.) OR %MDR($SURGIDR.) THEN DO;

      TAPS26 = 0;

      IF %MDX($TRANFID.) THEN TAPS26 = 1;

   END;
RUN;

  * ------------------------------------------------------------ ;

PROC MEANS DATA=OUT1.&OUTFILE1. N NMISS MIN MAX MEAN SUM;
PSSAS1 -- CONTINUED

RUN;

PROC CONTENTS DATA=OUT1.&OUTFILE1. POSITION;
RUN;
```

```
PROC PRINT DATA=OUT1.&OUTFILE1. (OBS=24);
TITLE4 "FIRST 24 RECORDS IN OUTPUT DATA SET &OUTFILE1.";
RUN;
```

CONTROL

```
* ------------------------------------------------------------- ;
* ---                      VERSION 2.1                       --- ;
* ------------------------------------------------------------- ;
* The Patient Safety Indicator (PSI) module of the AHRQ Quality
  Indicators software includes the following four programs, in
  addition to the programs CONTROL.SAS and PSFMTS.SAS:
      Program 1.   PSSAS1:  Assigns HCUP Patient Safety
                     Indicators to inpatient records.
      Program P2.  PSSASP2:  Calculates observed provider rates
                     for Patient Safety Indicators.
      Program P3.  PSSASP3:  Calculates risk-adjusted and smoothed
                     provider rates and merges with observed
                     provider rates.
      Program A2.  PSSASA2:  Calculates observed area rates for
                     Patient Safety Indicators.

*************************************************************;
**************************PLEASE READ************************;
*************************************************************;
*The USER MUST modify portions of the following code in order to
 run this software.  With one exception (see NOTE immediately
 following this paragraph), the only changes necessary to run
 this software using your data are changes you make in this
 program (CONTROL.SAS).  The modifications you make include such
 items as specifying the name and location of your input data
 set, the year of population data to be used, and the name and
 location of output data sets.

*NOTE:  CONTROL.SAS is written so that you have the option to
        read in data from and write out data to different
        locations.  For example, "LIBNAME1 IN1" points to the
        location or pathname of your input data set for program
        PSSAS1 and "LIBNAME OUT1" points to the location of the
        output data set created by the PSSAS1 program.  These
        locations can be different.  This pattern of allowing
        you to specify the location of each input and output
        data set is true for each of the different programs
        referenced in CONTROL.SAS.  However, if you wish to read
        in and write out all data to the same location you can.
        The easiest way to do this is to make a global
        change in CONTROL.SAS, changing "D:\Files\8908\PSIs\TEST" to the
        location you wish to use.

*NOTE:  In each of the four programs included with this
        package, as well as in the program which generates
        formats, there is a line of code that begins with
        "FILENAME CONTROL".  The USER MUST include after
        "FILENAME CONTROL" the location of the CONTROL.SAS file.

*Generally speaking, a first-time user of this software would
 proceed as follows:
      1.  Modify CONTROL.SAS.  (MUST be done.)
      2.  In the program generating needed formats, i.e., in
          PSFMTS.SAS, specify location (path name) of
          CONTROL.SAS. (MUST be done.)
```

3. Run PSFMTS.SAS. (MUST be done.)
4. In Program 1, i.e., in PSSAS1, specify location of CONTROL.SAS program. (MUST be done.)

CONTROL -- CONTINUED

5. Run PSSAS1. (MUST be done.)
6. If you want to generate provider-level Patient Safety Indicators:
    a. In Program P2, i.e., in PSSASP2, specify location of CONTROL.SAS program. (MUST be done if going to run Program P2.)
    b. Run PSSASP2. (MUST have already run Program 1.)
    c. In Program P3, i.e., in PSSASP3, specify location of CONTROL.SAS program. (MUST be done if going to run Program P3.)
    d. Run PSSASP3. (MUST have already run Programs 1 and P2.)
7. If you want to generate area-level Patient Safety Indicators:
    a. In Program A2, i.e., in PSSASA2, specify location of CONTROL.SAS program. (MUST be done if going to run Program A2.)
    b. Run PSSASA2. (MUST have already run Program 1.)

```
 * ------------------------------------------------------------- ;
 * ---                      All Programs                       --- ;
 * ------------------------------------------------------------- ;
TITLE "MEDPAR DATA - &SYSPARM";                 *<===USER may modify;
 *LIBNAME LIBRARY  'D:\Files\8908\PSIs\TEST';  *<==USER MUST modify;
%LET PRINT   = 1;                               *<===USER may modify;

 * ------------------------------------------------------------- ;
 * ---                        Program 1                        --- ;
 * ------------------------------------------------------------- ;
 *LIBNAME IN1  'D:\Files\8908\PSIs\TEST';       *<==USER MUST modify;
 *LIBNAME OUT1  'D:\Files\8908\PSIs\TEST';      *<==USER MUST modify;
 * ------------------------------------------------------------- ;
 * --- MODIFY INPUT AND OUTPUT FILE                              --- ;
 * ------------------------------------------------------------- ;
 * --- PROGRAM DEFAULT ASSUMES THERE ARE                         --- ;
 * ---     10 DIAGNOSES (DX1-DX10)                               --- ;
 * ---      6 PROCEDURES (PR1-PR6)                               --- ;
 * ------------------------------------------------------------- ;
 * --- MODIFY NUMBER OF DIAGNOSES, PROCEDURES AND RELATED        --- ;
 * --- VARIABLES TO MATCH INPUT DATA                             --- ;
 * ------------------------------------------------------------- ;
%LET INFILE1  = PSIdat;                         *<==USER MUST modify;
%LET OUTFILE1 = PS1;                            *<===USER may modify;
%LET NDX = 10;                                  *<===USER may modify;
%LET NPR = 6;                                   *<===USER may modify;
 * - SET TO 1 IF DATA AVAILABLE REGARDING NUMBER OF DAYS FROM   --- ;
 * - ADMISSION TO SECONDARY PROCEDURES                         --- ;
%LET PRDAY  = 1;                                *<===USER may modify;
 * ------------------------------------------------------------- ;

 * =============================================================;
 * ============PROVIDER-LEVEL Patient Safety INDICATORS=========;
 * =============================================================;
```

```
 * ---                   Program P2-Observed Rates              --- ;
 * ------------------------------------------------------------- ;
 * ------------------------------------------------------------- ;
 *LIBNAME INP2 'D:\Files\8908\PSIs\TEST';        *<===USER MUST modify;
 *LIBNAME OUTP2 'D:\Files\8908\PSIs\TEST';       *<===USER MUST modify;
```

CONTROL -- CONTINUED

```
 * ------------------------------------------------------------- ;
 * --- TYPELVLP indicates the levels (or _TYPE_) of              --- ;
 * --- summarization to be kept in the output.                   --- ;
 * ---                                                           --- ;
 * ---  TYPELVLP      stratification                             --- ;
 * ---  --------  ------------------------------------          --- ;
 * ---     0      overall                                       --- ;
 * ---     1                                      race          --- ;
 * ---     2                               payer                --- ;
 * ---     3                               payer * race         --- ;
 * ---     4                         sex                        --- ;
 * ---     5                         sex          * race        --- ;
 * ---     6                         sex * payer                --- ;
 * ---     7                         sex * payer * race         --- ;
 * ---     8                  age                               --- ;
 * ---     9                  age                * race         --- ;
 * ---    10                  age        * payer                 --- ;
 * ---    11                  age        * payer * race          --- ;
 * ---    12                  age * sex                          --- ;
 * ---    13                  age * sex          * race          --- ;
 * ---    14                  age * sex * payer                  --- ;
 * ---    15                  age * sex * payer * race           --- ;
 * ---    16       provider                                     --- ;
 * ---    17       provider                   * race            --- ;
 * ---    18       provider              * payer                --- ;
 * ---    19       provider              * payer * race         --- ;
 * ---    20       provider       * sex                         --- ;
 * ---    21       provider       * sex       * race            --- ;
 * ---    22       provider       * sex * payer                 --- ;
 * ---    23       provider       * sex * payer * race          --- ;
 * ---    24       provider * age                               --- ;
 * ---    25       provider * age             * race            --- ;
 * ---    26       provider * age       * payer                 --- ;
 * ---    27       provider * age       * payer * race          --- ;
 * ---    28       provider * age * sex                         --- ;
 * ---    29       provider * age * sex       * race            --- ;
 * ---    30       provider * age * sex * payer                 --- ;
 * ---    31       provider * age * sex * payer * race          --- ;
 * ---                                                           --- ;
 * --- The default TYPELVLP (0,16) will provide an overall       --- ;
 * --- and provider-level totals.                                --- ;
 * ---                                                           --- ;
 * ------------------------------------------------------------- ;
 %LET TYPELVLP = IN (0,16);                    *<===USER may modify;

 * ------------------------------------------------------------- ;
 * --- SET NAME OF MICRODATA INPUT (OUTPUT FROM PROGRAM 1) AND   --- ;
 * --- SUMMARY OUTPUT FILE OF OBSERVED PROVIDER RATES.           --- ;
 * ------------------------------------------------------------- ;
 %LET INFILEP2 = PS1;                          *<===USER may modify;
```

```
%LET OUTFILP2 = PSP2;                                *<===USER may modify;

 * ------------------------------------------------------------------ ;
 * --- INDICATE IF WANT TO CREATE A COMMA-DELIMITED FILE FOR    --- ;
 * --- EXPORT INTO A SPREADSHEET.                               --- ;
 * ------------------------------------------------------------------ ;
%LET TEXTP2=0;                                       *<===USER may modify;
 * ------------------------------------------------------------------ ;
 * --- IF YOU CREATE A COMMA-DELIMITED FILE, SPECIFY THE LOCATION - ;
```

CONTROL -- CONTINUED

```
 * --- OF THE FILE.                                             --- ;
 * ------------------------------------------------------------------ ;
 *FILENAME PSTEXTP2 'D:\Files\8908\PSIs\TEST\PSTEXTP2.TXT';
 *<===USER MUST modify;

 * ------------------------------------------------------------------ ;
 * ---       Program P3-Risk-Adjusted and Smooothed Rates       --- ;
 * ------------------------------------------------------------------ ;
 * ------------------------------------------------------------------ ;
 *LIBNAME INP3 'D:\Files\8908\PSIs\TEST';      *<===USER MUST modify;
 *LIBNAME OUTP3 'D:\Files\8908\PSIs\TEST';     *<===USER MUST modify;
 * ------------------------------------------------------------------ ;
 * ------------------------------------------------------------------ ;
 * --- SET NAME OF MICRODATA INPUT (OUTPUT FROM PROGRAM 1)      --- ;
 * ------------------------------------------------------------------ ;
%LET INFILP3X  = PS1;                                *<===USER may modify;
 * ------------------------------------------------------------------ ;
 * --- SET NAME OF COVARIATE FILE.                                - ;
 * --- THIS FILE CONTAINS THE REGRESSION COEFFICIENTS FOR EACH     - ;
 * --- COVARIATE.  THERE IS ONE OBSERVATION PER QI.               - ;
 * ------------------------------------------------------------------ ;
 *FILENAME COVARP 'D:\Files\8908\PSIs\TEST\COVARP00.TXT';
 *<===USER MUST modify;
 * ------------------------------------------------------------------ ;
 * --- SET NAME OF MSX STATISTICS FILE                            - ;
 * ---   FILE CONTAINS A SET OF THREE ARRAYS.  THE FIRST ARRAY     - ;
 * ---   CONTAINS NOISE ESTIMATES, THE SECOND SIGNAL ESTIMATES,    - ;
 * ---   AND THE THIRD MEAN PROVIDER RATES FOR EACH INPATIENT      - ;
 * ---   QUALITY INDICATOR.                                       - ;
 * ------------------------------------------------------------------ ;
 *FILENAME MSXP 'D:\Files\8908\PSIs\TEST\MSXP00.TXT';
 *<===USER MUSTmodify;
 * ------------------------------------------------------------------ ;
 * ------------------------------------------------------------------ ;
 * --- SET NAME OF INPUT RATES FILE (OUTPUT FROM PROGRAM P2) AND--- ;
 * --- PROVIDER LEVEL OUTPUT FILE.                              --- ;
 * ------------------------------------------------------------------ ;
%LET INFILP3Y  = PSP2;                               *<===USER may modify;
%LET OUTFILP3  = PSP3;                               *<===USER may modify;

 * ------------------------------------------------------------------ ;
 * --- INDICATE IF WANT TO CREATE A COMMA-DELIMITED FILE FOR    --- ;
 * --- EXPORT INTO A SPREADSHEET.                               --- ;
 * ------------------------------------------------------------------ ;
%LET TEXTP3=0;                                       *<===USER may modify;
 * ------------------------------------------------------------------ ;
 * --- IF YOU CREATE A COMMA-DELIMITED FILE, SPECIFY THE LOCATION - ;
```

```
  * --- OF THE FILE.                                                  --- ;
  * ------------------------------------------------------------------ ;
  *FILENAME PSTEXTP3 'D:\Files\8908\PSIs\TEST\PSTEXTP3.TXT';
  *<===USER may modify;

  * ==================================================================;
  * ============AREA-LEVEL Patient Safety INDICATORS=================;
  * ==================================================================;
  * ---                   Program A2-Observed Rates                --- ;
  * ------------------------------------------------------------------ ;
  * ------------------------------------------------------------------ ;
  * NOTE:  In program A2 the user should select the file with        ;
CONTROL -- CONTINUED

  *        population data for the year that best matches the        ;
  *        discharge data.                                           ;
  * In program A2,                                                   ;
  *       select one from:  QIPOP97.TXT, QIPOP98.TXT, QIPOP99.TXT    ;
  *                         QIPOP00.TXT, QIPOP01.TXT                 ;
  * ------------------------------------------------------------------ ;


  *LIBNAME INA2 'D:\Files\8908\PSIs\TEST';        *<===USER MUST modify;
  *LIBNAME OUTA2 'D:\Files\8908\PSIs\TEST';       *<===USER MUST modify;
  * ------------------------------------------------------------------ ;
  * --- TYPELVLA indicates the levels (or _TYPES_) of              --- ;
  * --- summarization to be kept in the output.                    --- ;
  * ---                                                            --- ;
  * ---  TYPELVLA      stratification                              --- ;
  * ---  --------  ------------------------                       --- ;
  * ---     0      overall                                        --- ;
  * ---     1                          race                       --- ;
  * ---     2                    sex                              --- ;
  * ---     3                    sex * race                       --- ;
  * ---     4              age                                    --- ;
  * ---     5              age *       race                       --- ;
  * ---     6              age * sex                              --- ;
  * ---     7              age * sex * race                       --- ;
  * ---     8       area                                          --- ;
  * ---     9       area  *             race                      --- ;
  * ---    10       area  *       sex                             --- ;
  * ---    11       area  *       sex * race                      --- ;
  * ---    12       area  * age                                   --- ;
  * ---    13       area  * age *       race                      --- ;
  * ---    14       area  * age * sex                             --- ;
  * ---    15       area  * age * sex * race                      --- ;
  * ---                                                            --- ;
  * --- The default TYPELVLA (0,8) will provide an overall         --- :
  * --- and area-level totals                                      --- ;
  * ---                                                            --- ;
  * ------------------------------------------------------------------ ;
 %LET TYPELVLA = IN (0,8);                      *<===USER may modify;

  * ------------------------------------------------------------------ ;
  * --- SET NAME OF MICRODATA INPUT (OUTPUT FROM PROGRAM 1) AND    --- ;
  * --- SUMMARY OUTPUT FILE OF OBSERVED AREA RATES.                --- ;
  * ------------------------------------------------------------------ ;
 %LET INFILEA2 = PS1;                           *<===USER may modify;
```

```
 %LET OUTFILA2 = PSA2;                          *<===USER may modify;
  * ------------------------------------------------------------- ;
  * --- SET NAME OF POPULATION FILE ------------------------------ ;
  * ------------------------------------------------------------- ;
  *FILENAME POPFILA2 'D:\Files\8908\PSIs\TEST\QIPOP97.TXT';
  *<===USER MUST modify

  * ------------------------------------------------------------- ;
  * --- INDICATE IF WANT TO CREATE A COMMA-DELIMITED FILE FOR    --- ;
  * --- EXPORT INTO A SPREADSHEET.                               --- ;
  * ------------------------------------------------------------- ;
 %LET TEXTA2=0;                                 *<===USER may modify;
  * ------------------------------------------------------------- ;
  * --- IF YOU CREATE A COMMA-DELIMITED FILE, SPECIFY THE LOCATION - ;
  * --- OF THE FILE.                                             --- ;
CONTROL -- CONTINUED

  * ------------------------------------------------------------- ;
  *FILENAME PSTEXTA2 'D:\Files\8908\PSIs\TEST\PSTEXTA2.TXT';
  *<===USER MUST modify;
```

PSFMTS

```
    *=================================================================================;
    * Title:
    *     Format Library for Patient Safety Indicators Program;
    *
    * Description:
    *     FORMAT LIB FOR AHRQ PATIENT SAFETY INDICATORS;
    *
    *         >>> VERSION 2.1 <<<
    *
    *     Note:  Codes valid from 10/1/1994 to 9/30/2001
    *
    *=================================================================================;
    *FILENAME  CONTROL  'D:\Files\8908\PSIs\TEST\CONTROL.SAS';   *<===USER  must
modify;

    %* INCLUDE CONTROL;

    PROC FORMAT; /* DDNAME=library; */

     /* ---------------------------------------------------------------- */
     /* --- FORMATS FOR STRATIFIERS ------------------------------------ */
     /* ---------------------------------------------------------------- */

     /* --- FORMAT FOR SEX CATEGORY ------------------------------------ */
    VALUE $SEXCAT
     'M' = 'Male'
     'F' = 'Female'
     ;

     /* --- FORMAT FOR AGE CATEGORY ------------------------------------ */
    VALUE $AGECAT
     '1' = '0 to 17'
     '2' = '18 to 39'
     '3' = '40 to 64'
     '4' = '65+'
     ;

     /* --- FORMAT FOR RACE CATEGORY ----------------------------------- */
    VALUE $RACECAT
     'W' = 'White'
     'B' = 'Black'
     'H' = 'Hispanic'
     'O' = 'Other'
     ;

     /* --- FORMAT FOR PAYER CATEGORY ---------------------------------- */
    VALUE $PAYCAT
     'M' = 'Medicare'
     'D' = 'Medicaid'
     'P' = 'Private'
     'S' = 'Self Pay'
     'O' = 'Other'
     ;
```

```
     * ------------------------------------------------------------- ;
     * --- PSI DEFINING FORMATS ------------------------------------ ;
     * ------------------------------------------------------------- ;
```

PSFMTS -- CONTINUED

```
     * ------------------------------------------------------------- ;
     * --- FORMATS - PROVIDER AND AREA LEVEL INDICATORS ------------ ;
     * ------------------------------------------------------------- ;

    /* Abdominopelvic */;
    /* ICD-9-CM procedure codes: */;

    VALUE $ABDOMIP
    '3804' = '1' /* INCISION OF AORTA */
    '3806' = '1' /* INCISION OF ABDOMINAL ARTERIES */
    '3807' = '1' /* INCISION OF ABDOMINAL VEINS */
    '3814' = '1' /* ENDARTERECTOMY OF AORTA */
    '3816' = '1' /* ENDARTERECTOMY OF ABDOMINAL ARTERIES */
    '3834' = '1' /* RESECTION OF AORTA WITH ANASTOMOSIS */
    '3836' = '1' /* RESECTION OF ABDOMINAL ARTERIES WITH ANASTOMOSIS */
    '3837' = '1' /* RESECTION OF ABDOMINAL VEINS WITH ANASTOMOSIS */
    '3844' = '1' /* RESECTION OF AORTA, ABDOMINAL WITH REPLACEMENT */
    '3846' = '1' /* RESECTION OF ABDOMINAL ARTERIES WITH REPLACEMENT */
    '3847' = '1' /* RESECTION OF ABDOMINAL VEINS WITH REPLACEMENT */
    '3857' = '1' /* LIGATION AND STRIPPING OF VARICOSE VEINS, ABDOMINCAL VEINS
*/
    '3864' = '1' /* OTHER EXCISION OF AORTA, ABDOMINAL */
    '3866' = '1' /* OTHER EXCISION OF ABDOMINAL ARTERIES */
    '3867' = '1' /* OTHER EXCISION OF ABDOMINAL VEINS */
    '3884' = '1' /* OTHER SURGICAL OCCLUSION OF AORTA, ABDOMINAL */
    '3886' = '1' /* OTHER SURGICAL OCCLUSION OF ABDOMINAL ARTERIES */
    '3887' = '1' /* OTHER SURGICAL OCCLUSION OF ABDOMINAL VEINS */
    '391'  = '1' /* INTRA-ABDOMINAL VENOUS SHUNT */
    '3924' = '1' /* AORTA-RENAL BYPASS */
    '3925' = '1' /* AORTA-ILIAC-FEMORAL BYPASS */
    '3926' = '1' /* OTHER INTRA-ABDOMINAL VASCULAR SHUNT OR BYPASS */
    '4052' = '1' /* RADICAL EXCISION OF PERIAORTIC LYMPH NODES */
    '4053' = '1' /* RADICAL EXCISION OF ILIAC LYMPH NODES */
    '412'  = '1' /* SPLENOTOMY */
    '4133' = '1' /* OPEN BIOPSY OF SPLEEN */
    '4141' = '1' /* MARSUPIALIZATION OF SPLENIC CYST */
    '4142' = '1' /* EXCISION OF LESION OR TISSUE OF SPLEEN */
    '4143' = '1' /* PARTIAL SPLENECTOMY */
    '415'  = '1' /* TOTAL SPLENECTOMY */
    '4193' = '1' /* EXCISION OF ACCESSORY SPLEEN */
    '4194' = '1' /* TRANSPLANTATION OF SPLEEN */
    '4195' = '1' /* REPAIR AND PLASTIC OPERATIONS ON SPLEEN */
    '4199' = '1' /* OTHER OPERATIONS ON SPLEEN */
    '4240' = '1' /* ESOPHAGECTOMY, NOS */
    '4241' = '1' /* PARTIAL EXOPHAGECTOMY */
    '4242' = '1' /* TOTAL ESOPHAGECTOMY */
    '4253' = '1' /* INTRATHORACIC ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    SMALL BOWEL */
    '4254' = '1' /* OTHER INTRATHORACIC ESOPHAGOENTEROSTOMY */
    '4255' = '1' /* INTRATHORACIC ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    COLON */
    '4256' = '1' /* OTHER INTRATHORACIC ESOPHAGOCOLOSTOMY */
```

```
    '4263' = '1' /* ANTESTERNAL ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    SMALL BOWEL */
    '4264' = '1' /* OTHER ANTESTERNAL ESOPHAGOENTEROSTOMY */
    '4265' = '1' /* ANTESTERNAL ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    COLON */
    '4266' = '1' /* OTHER ANTESTERNAL ESOPHAGOCOLOSTOMY */
```

PSFMTS -- CONTINUED

```
    '4291' = '1' /* LIGATION OF ESOPHAGEAL VARICES */
    '430'  = '1' /* GASTROSTOMY */
    '4319' = '1' /* OTHER GASTROSTOMY */
    '433'  = '1' /* PYLOROMYOTOMY */
    '4342' = '1' /* LOCAL EXCISION OF OTHER LESION OR TISSUE OF STOMACH */
    '4349' = '1' /* OTHER DESTRUCTION OF LESION OR TISSUE OF STOMACH */
    '435'  = '1' /* PARTIAL GASTRECTOMY WITH ANASTOMOSIS TO ESOPHAGUS */
    '436'  = '1' /* PARTIAL GASTRECTOMY WITH ANASTOMOSIS TO DUODENUM */
    '437'  = '1' /* PARTIAL GASTRECTOMY WITH ANASTOMOSIS TO JEJUNUM */
    '4381' = '1' /* PARTIAL GASTRECTOMY WITH JEJUNA TRANSPOSITION */
    '4389' = '1' /* OTHER PARTIAL GASTRECTOMY */
    '4391' = '1' /* TOTAL GASTRECTOMY WITH INTESTINAL INTERPOSITION */
    '4399' = '1' /* OTHER TOTAL GASTRECTOMY */
    '4400' = '1' /* VAGOTOMY, NOS */
    '4401' = '1' /* TRUNCAL VAGOTOMY */
    '4402' = '1' /* HIGHLY SELECTIVE VAGOTOMY */
    '4403' = '1' /* OTHER SELECTIVE VAGOTOMOY */
    '4411' = '1' /* TRANSABDOMINAL GASTROSCOPY */
    '4415' = '1' /* OPEN BIOPSY OF STOMACH */
    '4421' = '1' /* DILATION OF PYLORUS BY INCISION */
    '4429' = '1' /* OTHER PYLOROPLASTY */
    '4431' = '1' /* HIGH GASTRIC BYPASS */
    '4439' = '1' /* OTHER GASTROENTEROSTOMY */
    '4440' = '1' /* SUTURE OF PEPTIC ULCER, NOS */
    '4441' = '1' /* SUTURE OF GASTRIC ULCER SITE */
    '4442' = '1' /* SUTURE OF DUODENAL ULCER SITE */
    '445'  = '1' /* REVISION OF GASTRIC ANASTOMOSIS */
    '4461' = '1' /* SUTURE OF LACERATION OF STOMACH */
    '4463' = '1' /* CLOSURE OF OTHER GASTRIC FISTULA */
    '4464' = '1' /* GASTROPEXY */
    '4465' = '1' /* ESOPHAGOGASTROPLASTY */
    '4466'  =  '1'  /*  OTHER  PROCEDURES  FOR  CREATION  OF  ESOPHAGOGASTRIC
SPHINCTERIC
                    COMPETENCE */
    '4469' = '1' /* OTHER REPAIR OF STOMACH */
    '4491' = '1' /* LIGATION OF GASTRIC VARICES */
    '4492' = '1' /* INTRAOPERATIVE MANIPULATION OF STOMACH */
    '4500' = '1' /* INCISION OF INTESTINE, NOS */
    '4501' = '1' /* INCISION OF DUODENUM */
    '4502' = '1' /* OTHER INCISION OF SMALL INTESTINE */
    '4503' = '1' /* INCISION OF LARGE INTESTINE */
    '4531' = '1' /* OTHER LOCAL EXCISION OF LESION OF DUODENUM */
    '4532' = '1' /* OTHER DESTRUCTION OF LESION OF DUODENUM */
    '4533' = '1' /* LOCAL EXCISION OF LESION OR TISSUE OF SMALL INTESTINE,
                    EXCEPT DUODENUM */
    '4534' = '1' /* OTHER DESTRUCTION OF LESION OF SMALL INTESTINE, EXCEPT
                    DUODENUM */
    '4541' = '1' /* EXCISION OF LESION OR TISSUE OF LARGE INTESTINE */
    '4549' = '1' /* OTHER DESTRUCTION OF LESION OF LARGE INTESTINE */
    '4550' = '1' /* ISOLATION OF INTESTINAL SEGMENT, NOS */
```

```
'4551' = '1' /* ISOLATION OF SEGMENT OF SMALL INTESTINE */
'4552' = '1' /* ISOLATION OF SEGMENT OF LARGE INTESTINE */
'4561' = '1' /* MULTIPLE SEGMENTAL RESECTION OF SMALL INTESTINE */
'4562' = '1' /* OTHER PARTIAL RESECTION OF SMALL INTESTINE */
'4563' = '1' /* TOTAL REMOVAL OF SMALL INTESTINE */
'4571' = '1' /* MULTIPLE SEGMENTAL RESECTION OF LARGE INTESTINE */
'4572' = '1' /* CESECTOMY */
'4573' = '1' /* RIGHT HEMICOLECTOMY */
'4574' = '1' /* RESECTION OF TRANSVERSE COLON */
```

PSFMTS -- CONTINUED

```
'4575' = '1' /* LEFT HEMICOLECTOMY */
'4576' = '1' /* SIGMOIDECTOMY */
'4579' = '1' /* OTHER PARTIAL EXCISION OF LARGE INTESTINE */
'458'  = '1' /* TOTAL INTRA-ABDOMINAL COLECTOMY */
'4590' = '1' /* INTESTINAL ANASTOMOSIS, NOS */
'4591' = '1' /* SMALL-TO-SMALL INTESTINAL ANASTOMOSIS */
'4592' = '1' /* ANASTOMOSIS OF SMALL INTESTINE TO RECTAL STUMP */
'4593' = '1' /* OTHER SMALL-TO-LARGE INTESTINAL ANASTOMOSIS */
'4594' = '1' /* LARGE-TO-LARGE INTESTINAL ANASTOMOSIS */
'4595' = '1' /* ANASTOMOSIS TO ANUS */
'4601' = '1' /* EXTERIORIZATION OF SMALL INTESTINE */
'4603' = '1' /* EXTERIORIZATION OF LARGE INTESTINE */
'4610' = '1' /* COLOSTOMY, NOS */
'4611' = '1' /* TEMPORARY COLOSTOMY */
'4613' = '1' /* PERMANENT COLOSTOMY */
'4620' = '1' /* ILEOSTOMY, NOS */
'4621' = '1' /* TEMPORARY ILESOSTOMY */
'4622' = '1' /* CONTINENT ILEOSTOMY */
'4623' = '1' /* OTHER PERMANENT ILEOSTOMY */
'4640' = '1' /* REVISION OF INTESTINA STOMA, NOS */
'4641' = '1' /* REVISION OF STOMA OF SMALL INTESTINE */
'4642' = '1' /* REPAIR OF PERICOLOSTOMY HERNIA */
'4643' = '1' /* OTHER REVISION OF STOMA OF LARGE INTESTINE */
'4650' = '1' /* CLOSURE OF INTESTINAL STOMA, NOS */
'4651' = '1' /* CLOSURE OF STOMA OF SMALL INTESTINE */
'4652' = '1' /* CLOSURE OF STOMA OF LARGE INTESTINE */
'4660' = '1' /* FIXATION OF INTESTINE, NOS */
'4661' = '1' /* FIXATION OF SMALL INTESTINE TO ABDOMINAL WALL */
'4662' = '1' /* OTHER FIXATION OF SMALL INTESTINE */
'4663' = '1' /* FIXATION OF LARGE INTESTINE TO ABDOMINAL WALL */
'4664' = '1' /* OTHER FIXATION OF LARGE INTESTINE */
'4672' = '1' /* CLOSURE OF FISTULA OF DUODENUM */
'4674' = '1' /* CLOSURE OF FISTULA OF SMALL INTESTINE, EXCEPT DUODENUM */
'4676' = '1' /* CLOSURE OF FISTULA OF LARGE INTESTINE */
'4680' = '1' /* INTRA-ABDOMINAL MANIPULATION OF INTESTINE, NOS */
'4681' = '1' /* INTRA-ABDOMINAL MANIPULATION OF SMALL INTESTINE */
'4682' = '1' /* INTRA-ABDOMINAL MANIPULATION OF SMALL INTESTINE */
'4691' = '1' /* MYOTOMY OF SIGMOID COLON */
'4692' = '1' /* MYOTOMY OF OTHER PARTS OF COLON */
'4693' = '1' /* REVISION OF ANASTOMOSIS OF SMALL INTESTINE */
'4694' = '1' /* REVISION OF ANASTOMOSIS OF LARGE INTESTINE */
'4699' = '1' /* OTHER OPERATIONS ON INTESTINES */
'4709' = '1' /* OTHER APPENDECTOMY */
'4719' = '1' /* OTHER INCIDENTAL APPENDECTOMY */
'472'  = '1' /* DRAINAGE OF APPENDICEAL ABSCESS */
'4791' = '1' /* APPENDECTOMY */
'4792' = '1' /* CLOSURE OF APPENDICEAL FISTULA */
```

```
    '4799' = '1' /* OTHER */
    '4841' = '1' /* SUBMUCOSAL RESECTION OF RECTUM */
    '4849' = '1' /* OTHER PULL-THROUGH RESECTION OF RECTUM */
    '485'  = '1' /* ABDOMINOPERINEAL RESECTION OF RECTUM */
    '4875' = '1' /* ABDOMINAL PROCTOPEXY */
    '500'  = '1' /* HEPATOTOMY */
    '5012' = '1' /* OPEN BIOPSY OF LIVER */
    '5021' = '1' /* MARSUPIALIZATION OF LESION OF LIVER */
    '5022' = '1' /* PARTIAL HEPATECTOMY */
    '5029' = '1' /* OTHER DESTRUCTION OF LESION OF LIVER */
    '503'  = '1' /* LOBECTOMY OF LIVER */
    '504'  = '1' /* TOTAL HEPATECTOMY */
```

PSFMTS -- CONTINUED

```
    '5051' = '1' /* AUXILIARY LIVER TRANSPLANT */
    '5059' = '1' /* OTHER TRANSPLANT OF LIVER */
    '5069' = '1' /* OTHER REPAIR OF LIVER */
    '5103' = '1' /* OTHER CHOLECYSTOTSTOMY */
    '5104' = '1' /* OTHER CHOLECYSTOSTOMY */
    '5113' = '1' /* OPEN BIOPSY OF GALLBLADDER OR BILE DUCTS */
    '5121' = '1' /* OTHER PARTIAL CHOLECYSTECTOMY */
    '5122' = '1' /* CHOLECYSTECTOMY */
    '5131' = '1' /* ANASTOMOSIS OF GALLBLADDER TO HEPATIC DUCTS */
    '5132' = '1' /* ANASTOMOSIS OF GALLBLADDER TO INTESTINE */
    '5133' = '1' /* ANASTOMOSIS OF GALLBLADDER TO PANCREAS */
    '5134' = '1' /* ANASTOMOSIS OF GALLBLADDER TO STOMACH */
    '5135' = '1' /* OTHER GALLBLADDER ANASTOMOSIS */
    '5136' = '1' /* CHOLEDOCHOENTEROSTOMY */
    '5137' = '1' /* ANASTOMOSIS OF HEPATIC DUCT TO GASTROINTESTINAL TRACT */
    '5139' = '1' /* OTHER BILE DUCT ANASTOMOSIS */
    '5141' = '1' /* COMMON DUCT EXPLORATION FOR REMOVAL OF CALCULUS */
    '5142' = '1' /* COMMON DUCT EXPLORATION FOR RELIEF OF OTHER OBSTRUCTION */
    '5143' = '1' /* INSERTION OF CHOLEDOCHOHEPATIC TUBE FOR DECOMPRESSION */
    '5149' = '1' /* INCISION OF OTHER BILE DUCTS FOR RELIEF OF OBSTRUCTION */
    '5151' = '1' /* EXPLORATION OF COMMON DUCT */
    '5159' = '1' /* INCISION OF OTHER BILE DUCT */
    '5161' = '1' /* EXCISION OF CYSTIC DUCT REMNANT */
    '5162' = '1' /* EXCISION OF AMPULLA OF VATER WITH REIMPLANTATION OF COMMON
                    DUCT */
    '5163' = '1' /* OTHER EXCISION OF COMMON DUCT */
    '5169' = '1' /* EXCISION OF OTHER BILE DUCT */
    '5171' = '1' /* SIMPLE SUTURE OF COMMON BILE DUCT */
    '5172' = '1' /* CHOLEDOCHOPLASTY */
    '5179' = '1' /* REPAIR OF OTHER BILE DUCTS */
    '5181' = '1' /* DILATION OF SPHINCTER OF ODDI */
    '5182' = '1' /* PANCREATIC SPHINCTEROTOMY */
    '5183' = '1' /* PANCREATIC SPHINCTEROPLSTY */
    '5189' = '1' /* OTHER OPERATIONS ON SPHINCTER OF ODDI */
    '5192' = '1' /* CLOSURE OF CHOLECYSTOSTOMY */
    '5193' = '1' /* CLOSURE OF OTHER BILIARY FISTULA */
    '5194' = '1' /* REVISION OF ANASTOMOSIS OF BILIARY TRACT */
    '5195' = '1' /* REMOVAL OF PROSTHETIC DEVICE FROM BILE DUCT */
    '5199' = '1' /* OTHER OPERATIONS ON BILIARY TRACT */
    '5201' = '1' /* DRAINAGE OF PANCREATIC CYST BY CATHETER */
    '5209' = '1' /* OTHER PANCREATOTOMY */
    '5212' = '1' /* OPEN BIOPSY OF PANCREAS */
    '5222' = '1' /* OTHER EXCISION OR DESTRUCTION OF LESION OR TISSUE OF
PANCREAS
```

```
                    OR PANCREATIC DUCT */
    '523'  = '1' /* GINGIVAL AND PERIODONTAL DISEASES */
    '524'  = '1' /* INTERNAL DRAINAGE OF PANCREATIC CYST */
    '5251' = '1' /* PROXIMAL PANCREATECTOMY */
    '5252' = '1' /* DISTAL PANCREATECTOMY */
    '5253' = '1' /* RADIAL SUBTOTAL PANCREATECTOMY */
    '5259' = '1' /* OTHER PARTIAL PANCREATECTOMY */
    '526'  = '1' /* TOTAL PANCREATECTOMY */
    '527'  = '1' /* RADICAL PANCREATICODUODENCECTOMY */
    '5280' = '1' /* PANCREATIC TRANSPLANT, NOS */
    '5281' = '1' /* REIMPLANTATION */
    '5282' = '1' /* HOMOTRANSPLANT OF PANCREAS */
    '5283' = '1' /* HETEROTRANSPLANT OF PANCREAS */
    '5292' = '1' /* CANNULATION OF PANCREATIC DUCT */
    '5295' = '1' /* OTHER REPAIR OF PANCREAS */
    PSFMTS -- CONTINUED

    '5296' = '1' /* ANASTOMOSIS OF PANCREAS */
    '5299' = '1' /* OTHER OPERATIONS ON PANCREAS */
    '5300' = '1' /* UNILATERAL REPAIR OF INGUINAL HERNIA, NOS */
    '5301' = '1' /* REPAIR OF DIRECT INGUINAL HERNIA */
    '5302' = '1' /* REPAIR OF INDIRECT INGUINAL HERNIA */
    '5303' = '1' /* REPAIR OF DIRECT INGUINAL HERNIA */
    '5304'  =  '1'  /*  REPAIR  OF  INDIRECT  INGUINAL  HERNIA  WITH  GRAFT  OR
PROSTHESIS */
    '5305' = '1' /* REAPIR OF INGUINAL HERNIA WITH GRAFT OR PROSTHESIS, NOS */
    '5310' = '1' /* BILATERAL REPAIR OF INGUINAL HERNIA, NOS */
    '5311' = '1' /* BILATERAL REPAIR OF DIRECT INGUINAL YERNIA */
    '5312' = '1' /* BILATERAL REPAIR OF INDIRECT INGUINAL HERNIA */
    '5313' = '1' /* BILATERAL REPAIR OF INGUINAL HERNIA, ONE DIRECT AND ONE
                    INDIRECT */
    '5314' = '1' /* BILATERAL REPAIR OF DIRECT INGUINAL HERNIA WITH GRAFT OR
                    PROSTHESIS */
    '5315' = '1' /* BILATERAL REPAIR OF INDIRECT INGUINAL HERNIA WITH GRAFT OR
                    PROSTHESIS */
    '5316' = '1' /* BILATERAL REPAIR OF INGUINAL HERNIA, ONE DIRECT AND ONE
                    INDIRECT, WITH GRAFT OR PROSTHESIS */
    '5317' = '1' /* BILATERAL INGUINAL HERNIA REPAIR WITH GRAFT OR PROSTHESIS,
                    NOS */
    '5321' = '1' /* UNILATERAL REPAIR OF FEMORAL HERNIA */
    '5329' = '1' /* OTHER UNILATERAL FEMORAL HERNIORHAPHY */
    '5331'  =  '1'  /*  BILATERAL  REPAIR  OF  FEMORAL  HERNIA  WITH  GRAFT  OR
PROSTHESIS */
    '5339' = '1' /* OTHER BILATERAL FEMORAL HERNIORRHAPHY */
    '5341' = '1' /* REPAIR OF UMBILICAL HERNIA WITH PROSTHESIS */
    '5349' = '1' /* OTHER UMBILICAL HERNIORRHAPHY */
    '5351' = '1' /* INCISIONAL HERNIA REPAIR */
    '5359' = '1' /* REPAIR OF OTHER HERNIA OF ANTERIOR ABDOMINAL WALL */
    '5361' = '1' /* INCISIONAL HERNIA REPAIR WITH PROSTHESIS */
    '5369' = '1' /* REPAIR OF OTHER HERNIA OF ANTERIOR ABDOMINAL WALL WITH
                    PROSTHESIS */
    '537'  = '1' /* REPAIR OF DIAPHRAGMATIC HERNIA, ABDOMINAL APPROACH */
    '540'  = '1' /* INCISION OF ABDOMINAL WALL */
    '5411' = '1' /* EXPLORATORY LAPAROTOMY */
    '5419' = '1' /* OTHER LAPAROTOMY */
    '5422' = '1' /* BIOPSY OF ABDOMINAL WALL OR UMBILICUS */
    '5423' = '1' /* BIOPSY OF ABDOMINAL WALL OR UMBILICUS */
```

```
    '543'  = '1' /* EXCISION OR DESTRUCTION OF LESION OR TISSUE OF ABDOMINAL
WALL
                  OR UMBILICUS */
    '544'  = '1' /* EXCISION OR DESTRUCTION OF PERITONEAL TISSUE */
    '5459' = '1' /* OTHER LYSIS OF PERITONEAL ADHESIONS */
    '5463' = '1' /* OTHER SUTURE OF ABDOMINAL WALL */
    '5464' = '1' /* SUTURE OF PERITONEUM */
    '5471' = '1' /* REPAIR OF GASTROSCHISIS */
    '5472' = '1' /* OTHER REPAIRE OF ABDOMINAL WALLS */
    '5473' = '1' /* OTHER REPAIR OF PERITONEUM */
    '5474' = '1' /* OTHER REPAIR OF OMENTUM */
    '5475' = '1' /* OTHER REPAIR OF MESENTERY */
    '5492' = '1' /* REMOVAL OF FOREIGN BODY FROM PERITONEAL CAVITY */
    '5493' = '1' /* CREATION OF CUTANEOPERITONEAL FISTULA */
    '5494' = '1' /* CREATION OF PERITONEOVASCULAR SHUNT */
    '5495' = '1' /* INCISION OF PERITONEUM */
    '5551' = '1' /* NEPHROURETERECTOMY */
    '5552' = '1' /* NEPHRECTOMY OF REMAINING KIDNEY */
    '5553' = '1' /* REMOVAL OF TRANSPLANTED OR REGECTED KDINEY */
    PSFMTS -- CONTINUED

    '5554' = '1' /* BILATERAL NEPHRECTOMY */
    '5561' = '1' /* RENAL AUTOTRANSPLANTATION */
    '5569' = '1' /* ULCERATIVE COLITIS, UNSPECIFIED */
    '557'  = '1' /* NEPHROPEXY */
    '5583' = '1' /* CLOSURE OF OTHER FISTULA OF KIDNEY */
    '5584' = '1' /* REDUCTION OF TORSION OF RENAL */
    '5585' = '1' /* SYMPHYSIOTOMY FOR HORESHOE KIDNEY */
    '5586' = '1' /* ANASTOMOSIS OF KIDNEY */
    '5587' = '1' /* CORRECTION OF URETEROPELVIC JUNCTION */
    '5591' = '1' /* DECAPSULATION OF KEDNEY */
    '5597' = '1' /* IMPLANTATION OR REPLACEMENT OF MECHANICAL KIDNEY */
    '5598' = '1' /* REMOVAL OF MECHANICAL KIDNEY */
    '5651' = '1' /* FORMATION OF CUTANEOUS URETEROILEOSTOMY */
    '5652' = '1' /* REVISION OF CUTANEOUS URETER OILEOSTOMY */
    '5661' = '1' /* FORMATION OF OTHER CUTANEOUS URETEROSTOMY */
    '5662' = '1' /* REVISION OF OTHER CUTANEOUS URETEROSTOMY */
    '5671' = '1' /* URINARY DIVERSION TO INTESTINE */
    '5672' = '1' /* REVISION OF URETEROINTESTINAL ANASTOMOSIS */
    '5673' = '1' /* NEPHROCYSTANASTOMOSIS, NO */
    '5674' = '1' /* URETERONEOXYSTOSTOMY */
    '5675' = '1' /* TRANSURETEROURETEROSTOMY */
    '5683' = '1' /* CLOSURE OF URETEROSTOMY */
    '5684' = '1' /* CLOSURE OF OTHER FISTULA OF URETER */
    '5685' = '1' /* URETEROPEXY */
    '5686' = '1' /* REMOVAL OF LIGATURE FORM URETER */
    '5689' = '1' /* OTHER REPAIR OF URETER */
    '5695' = '1' /* LIGATION OF URETER */
    '5771' = '1' /* RADICAL CYSTECTOMY */
    '5779' = '1' /* OTHER TOTAL CYSTECTOMY */
    '5782' = '1' /* CLOSRUE OF CYSTOSTOMY */
    '5787' = '1' /* RECONSTRUCTION OF URNIARY BLADDER */
    '5900' = '1' /* RETROPERITONEAL DISSECTION, NOS */
    '5902' = '1' /* OTHER LYSIS OF PERIRENAL OR PERIURETERAL ADHESIONS */
    '5909' = '1' /* OTHER INCISION OF PERIRENAL OR PERIURETERAL TISSUE */
    '6012' = '1' /* OPEN BIOPSY OF PROSTATE */
    '6014' = '1' /* OPEN BIOPSY OF SEMINAL VESICLES */
    '6015' = '1' /* BIOPSY OF PERIPROSTATIC TISSUE */
```

```
    '603'  = '1' /* SUPRAPUBIC PROSTATECTOMY */
    '604'  = '1' /* RETROPUBIC PROSTATECTOMY */
    '605'  = '1' /* RADICAL PROSTATECTOMY */
    '6061' = '1' /* LOCAL EXCISION OF LESION OF PROSTATE */
    '6072' = '1' /* INCISION OF SEMINAL VESICLE */
    '6073' = '1' /* EXCISION OF SEMINAL VESICLE */
    '6079' = '1' /* OTHER OPERATIONS ON SEMINAL VESICLES */
    '6093' = '1' /* REPAIR OF PROSTATE */
    '6509' = '1' /* OTHER OOPHORECTOMY */
    '6512' = '1' /* OTHER BIOPSY OF OVARY */
    '6521' = '1' /* MARSUPIALIZATION OF OVARIAN CYST */
    '6522' = '1' /* WEDGE RESECTION OF OVARY*/
    '6529' = '1' /* OTHER LOCAL EXCISION OR DESTRUCTION OF OVARY */
    '6539' = '1' /* OTHER UNLILATERAL OOPHORECTOMY */
    '6549' = '1' /* OTHER UNLIATERAL SALPINGOOPHORECTOMY */
    '6551' = '1' /* OTHER REMOVAL OF BOTH OVARIES AT SAME OPERATIVE EPISODE */
    '6552' = '1' /* OTHER REMOVAL OF REMAINING OVARY */
    '6561' = '1' /* OTHER REMOVAL OF BOTH OVARIES AND TUBES AT SAME OPERATIVE
                    EPISODE */
    '6562' = '1' /* OTHER REMOVAL OF REMAINING OVARY AND TUBE */
    '6571' = '1' /* OTHER SIMPLE SUTURE OF OVARY */
    '6572' = '1' /* OTHER REIMPLANTATION OF OVARY */
```

PSFMTS -- CONTINUED

```
    '6573' = '1' /* OTHER SALPINGO OOPHOROPLASTY */
    '6579' = '1' /* OTHER REPAIR OF OVARY */
    '6589' = '1' /* OTHER LYSIS OF ADHESIONS OF AVARY AND FALLOPIAN TUBE */
    '6592' = '1' /* TRANSPLANTATION OF OVARY */
    '6593' = '1' /* MANUAL RUPTURE OF OVARIAN CYST */
    '6594' = '1' /* OVARIAN DENERVATION */
    '6595' = '1' /* RELEASE OF TORSION OF OVARY */
    '6599' = '1' /* OTHER OPERATIONS ON OVARY */
    '6601' = '1' /* SALPINGOTOMY */
    '6602' = '1' /* SALPINGOSTOMY */
    '6631' = '1' /* OTHER BILATERAL LIGATION AND CRUSHING OF FALLOPIAN TUBES
*/
    '6632' = '1' /* OTHER BILATERAL LIGATION AND DIVISION OF FALLOPIAN TUBES
*/
    '6639' = '1' /* OTHER BILATERAL DESTRUCTION OR OCCLUSION OF FALLOPIAN
TUBES */
    '664'  = '1' /* TOTAL UNILATERAL SALPINGECTOMY */
    '6651' = '1' /* REMOVAL OF BOTH FALLOPIAN TUBES AT SAME OPERATIVE EPISODE
*/
    '6652' = '1' /* REMOVAL OF REMAINING FALLOPIAN TUBE */
    '6661' = '1' /* EXCISION OR DESTRUCTION OF LESION OF FALLOPIAN TUBE */
    '6662' = '1' /* SALPINGECTOMY WITH REMOVAL OF TUBAL PREGNANCY */
    '6663' = '1' /* BILATERAL PARTIAL SALPINGECTOMY, NOS */
    '6669' = '1' /* OTHER PARTIAL SALPINGECTOMY */
    '6671' = '1' /* SIMPLE SUTURE OF FALLOPIAN TUBE */
    '6672' = '1' /* SALPINGO-OOPHOROSTOMY*/
    '6673' = '1' /* SALPINGO-SALPINGOSTOMY */
    '6674' = '1' /* SALPING0-UTEROSTOMY */
    '6679' = '1' /* OTHER REPAIR OF FALLOPIAN TUBE */
    '6692' = '1' /* UNILATERAL DESTRUCTION OR OCCLUSION OF FALLOPIAN TUBE */
    '6697' = '1' /* BURYING OF FIMBRIAE IN UTERINE WALL */
    '680'  = '1' /* OTHER INCISION AND EXCISION OF UTERUS */
    '6813' = '1' /* OPEN BIOPSY OF UTERUS */
    '6814' = '1' /* OPEN BIOPSY OF UTERINE LIGAMENTS */
```

```
'683'  = '1' /* SUBTOTAL ABDOMINAL HYSTERECTOMY */
'684'  = '1' /* TOTAL ABDOMINAL HYSTERECTOMY */
'686'  = '1' /* RADICAL ABDOMINAL HYSTERECTOMY */
'688'  = '1' /* PELVIC EVISCERATION */
'6922' = '1' /* OTHER UTERINE SUSPENSION */
'693'  = '1' /* PARACERVICAL UTERINE DENERVATION */
'6941' = '1' /* SUTURE OF LACERATION OF UTERUS */
'6942' = '1' /* CLOSURE OF FISTULA OF UTERUS */
'6949' = '1' /* OTHER REPAIR OF UTERUS */
OTHER  = '0' ;

/* Active drug dependence */
/* ICD-9-CM diagnosis codes: */

VALUE $ADRUGID
'30400' = '1' /* OPIOID TYPE DEPENDENCE-UNSPECIFIED */
'30401' = '1' /* OPIOID TYPE DEPENDENCE-CONTINUOUS */
'30402' = '1' /* OPIOID TYPE DEPENDENCE-EPISODIC */
'30410' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                 DEPENDENCE -UNSPECIFIED */
'30411' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                 DEPENDENCE -CONTINUOUS */
'30412' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                 DEPENDENCE, -EPISODIC */
'30420' = '1' /* COCAINE DEPENDENCE-UNSPECIFIED */
```

PSFMTS -- CONTINUED

```
'30421' = '1' /* COCAINE DEPENDENCE-CONTINUOUS */
'30422' = '1' /* COCAINE DEPENDENCE-EPISODIC */
'30430' = '1' /* CANNABIS DEPENDENCE UNSPECIFIED */
'30431' = '1' /* CANNABIS DEPENDENCE CONTINUOUS */
'30432' = '1' /* CANNABIS DEPENDENCE EPISODIC */
'30440' = '1' /* AMPHETAMINE AND OTHER PSYCHO STIMULANT DEPENDENCE
                 -UNSPECIFIED */
'30441' = '1' /* AMPHETAMINE AND OTHER PSYCHO STIMULANT DEPENDENCE
                 -CONTINUOUS */
'30442' = '1' /* AMPHETAMINE AND OTHER PSYCHO STIMULANT DEPENDENCE
                 -EPISODIC */
'30450' = '1' /* HALLUCINOGEN DEPENDENCE UNSPCIFIED */
'30451' = '1' /* HALLUCINOGEN DEPENDENCE- CONTINUOUS */
'30452' = '1' /* HALLUCINOGEN DEPENDENCE - EPISODIC */
'30460' = '1' /* OTHER SPECIFIED DRUG DEPENDENCE - UNSPECIFIED */
'30461' = '1' /* OTHER SPECIFIED DRUG DEPENDENCE -CONTINUOUS */
'30462' = '1' /* OTHER SPECIFIED DRUG DEPENDENCE -EPISODIC */
'30470' = '1' /* COMBINATIONS OF OPIOID TYPE DRUG WITH ANY OTHER
                 -UNSPECIFIED */
'30471' = '1' /* COMBINATIONS OF OPIOID TYPE DRUG WITH ANY OTHER
                 -CONTINUOUS */
'30472' = '1' /* COMBINATIONS OF OPIOID TYPE DRUG WITH ANY OTHER
                 -EPISODIC */
'30480' = '1' /* COMBINATIONS OF DRUG EXCLUDING OPIOID TYPE DRUG
                 -UNSPECIFIED */
'30481' = '1' /* COMBINATIONS OF DRUG EXCLUDING OPIOID TYPE DRUG
                 -CONTINUOUS */
'30482' = '1' /* COMBINATIONS OF DRUG EXCLUDING OPIOID TYPE DRUG
                 -EPISODIC */
'30490' = '1' /* UNSPECIFIED DRUG DEPENDENCE -UNSPECIFIED */
'30491' = '1' /* UNSPECIFIED DRUG DEPENDENCE -CONTINUOUS */
```

```
    '30492' = '1' /* UNSPECIFIED DRUG DEPENDENCE -EPISODIC */
    OTHER   = '0' ;

    /* Active nondependent abuse of drugs */
    /* ICD-9-CM diagnosis codes: */

    VALUE $ANONDID
    '30500' = '1' /* ALCOHOL ABUSE-UNSPECIFIED */
    '30501' = '1' /* ALCOHOL ABUSE-CONTINUOUS */
    '30502' = '1' /* ALCOHOL ABUSE-EPISODIC */
    '30510' = '1' /* TOBACCO USE DISORDER-UNSPECIFIED */
    '30511' = '1' /* TOBACCO USE DISORDER - CONTINUOUS */
    '30512' = '1' /* TOBACCO USE DISORDER -EPSODIC */
    '30520' = '1' /* CANNABIS ABUSE-UNSPECIFIED */
    '30521' = '1' /* CANNABIS ABUSE-CONTINUOUS */
    '30522' = '1' /* CANNABIS ABUSE-EPISODIC */
    '30530' = '1' /* HALLUCINOGEN ABUSE- UNSPECIFIED */
    '30531' = '1' /* HALLUCINOGEN ABUSE-CONTINUOUS */
    '30532' = '1' /* HALLUCINOGEN ABUSE- EPISODIC */
    '30540' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                     ABUSE-UNSPECIFIED */
    '30541' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                     ABUSE-CONTINUOUS */
    '30542' = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                     ABUSE-EPISODIC */
    '30550' = '1' /* OPIOID ABUSE-UNSPECIFIED */
    '30551' = '1' /* OPIOD ABUSE-CONTINUOUS */
    '30552' = '1' /* OPIOID ABUSE-EPISODIC */
    PSFMTS -- CONTINUED

    '30560' = '1' /* COCAINE ABUSE-UNSPECIFIED */
    '30561' = '1' /* COCAINE ABUSE-CONTINUOUS */
    '30562' = '1' /* COCAINE ABUSE-EPISODIC */
    '30570' = '1' /* AMPHETAMINE OR RELATED ACTING SYMPATHOMIMETIC ABUSE
                     -UNSPECIFIED */
    '30571' = '1' /* AMPHETAMINE OR RELATED ACTING SYMPATHOMIMETIC ABUSE
                     -CONTINUOUS */
    '30572' = '1' /* AMPHETAMINE OR RELATED ACTING SYMPATHOMIMETIC ABUSE
                     - EPISODIC */
    '30580' = '1' /* ANTIDEPRESSANT TYPE ABUSE-UNSPECIFIED */
    '30581' = '1' /* ANTIDEPRESSANT TYPE ABUSE-CONTINUOUS */
    '30582' = '1' /* ANTIDEPRESSANT TYPE ABUSE-EPISODIC */
    '30590' = '1' /* OTHER, MIXED, OR UNSPECIFIED DRUG ABUSE-UNSPECIFIED */
    '30591' = '1' /* OTHER, MIXED, OR UNSPECIFIED DRUG ABUSE- CONTINUOUS */
    '30592' = '1' /* OTHER, MIXED, OR UNSPECIFIED DRUG ABUSE- EPISODIC */
    OTHER   = '0' ;

    /* Acute myocardial infarction */
    /* ICD-9-CM diagnosis codes: */

    VALUE $AMIID
    '41000' = '1' /* AMI OF ANTEROLATERAL WALL – EPISODE OF CARE UNSPECIFIED
*/
    '41001' = '1' /* AMI OF ANTEROLATERAL WALL - INITIAL EPISODE OF CARE */
    '41010' = '1' /* AMI OF OTHER ANTERIOR WALL – EPISODE OF CARE UNSPECIFIED
*/
    '41011' = '1' /* AMI OF OTHER ANTERIOR WALL – INITIAL EPISODE OF CARE */
```

```
    '41020' = '1' /* AMI OF INFEROLATERAL WALL – EPISODE OF CARE UNSPECIFIED
*/
    '41021' = '1' /* AMI OF INFEROLATERAL WALL – INITIAL EPISODE OF CARE */
    '41030' = '1' /* AMI OF INFEROPOSTERIOR WALL – EPISODE OF CARE UNSPECIFIED
*/
    '41031' = '1' /* AMI OF INFEROPOSTERIOR WALL –– INITIAL EPISODE OF CARE */
    '41040' = '1' /* AMI OF INFERIOR WALL - EPISODE OF CARE UNSPECIFIED */
    '41041' = '1' /* AMI OF INFERIOR WALL - INITIAL EPISODE OF CARE */
    '41050' = '1' /* AMI OF OTHER LATERAL WALL - EPISODE OF CARE UNSPECIFIED
*/
    '41051' = '1' /* AMI OF OTHER LATERAL WALL - INITIAL EPISODE OF CARE */
    '41060' = '1' /* AMI TRUE POSTERIOR WALL INFARCTION - EPISODE OF CARE
                     UNSPECIFIED */
    '41061' = '1' /* AMI TRUE POSTERIOR WALL INFARCTION - INITIAL EPISODE OF
                     CARE */
    '41070' = '1' /* AMI SUBENDOCARDIAL INFARCTION - EPISODE OF CARE
                     UNSPECIFIED */
    '41071' = '1' /* AMI SUBENDOCARDIAL INFARCTION - INITIAL EPISODE OF CARE
*/
    '41080' = '1' /* AMI OF OTHER SPECIFIED SITES - EPISODE OF CARE
                     UNSPECIFIED */
    '41081' = '1' /* AMI OF OTHER SPECIFIED SITES - - INITIAL EPISODE OF CARE
*/
    '41090' = '1' /* AMI UNSPECIFIED SITE - EPISODE OF CARE UNSPECIFIED */
    '41091' = '1' /* AMI UNSPECIFIED SITE - INITIAL EPISODE OF CARE */
    OTHER   = '0' ;

    /* Acute Respiratory Failure */;
    /* ICD-9-CM diagnosis codes: */

    VALUE $ACURFID
    '51881' = '1'
    PSFMTS -- CONTINUED

    '51884' = '1'
    OTHER   = '0' ;

    /* Anesthesia complications */
    /* ICD-9-CM diagnosis codes: */

    VALUE $ANES1ID
    '9681'  = '1' /* HALOTHANE */
    '9682'  = '1' /* OTHER GASEOUS ANESTHETICS */
    '9683'  = '1' /* INTRAVENEOUS ANESTHETICS */
    '9684'  = '1' /* OTHER AND UNSPECIFIED GENERAL ANESTHETICS */
    '9687'  = '1' /* SPINAL ANESTHETICS */
    'E8551' = '1' /* OTHER NERVOUS SYSTEM DEPRESSANTS */
    OTHER   = '0' ;

    VALUE $ANES2ID
    'E8763' = '1' /* OTHER AND UNSPECIFIED MISADVENTURES DURING MEDICAL CARE,
                     ENDOTRACHEAL TUBE WRONGLY PLACED DURING ANESTHETIC
                     PROCEDURE */
    'E9381' = '1' /* HALOTHANE */
    'E9382' = '1' /* OTHER GASEOUS ANESTHETICS */
    'E9383' = '1' /* INTRAVENOUS ANESTHETICS */
    'E9384' = '1' /* OTHER AND UNSPECIFIED GENERAL ANESTHETICS */
    'E9385' = '1' /* SURFACE AND INFILTRATION ANESTHETICS */
```

```
'E9386' = '1' /* PERIPHERAL NERVE AND PLEXUS BLOCKING ANESTHETICS */
'E9387' = '1' /* SPINAL ANESTHETICS */
'E9389' = '1' /* OTHER AND UNSPECIFIED LOCAL ANESTHETICS */
OTHER   = '0' ;

/* Anoxic brain injury */
/* ICD-9-CM diagnosis codes: */

VALUE $ANOXIID
'3481'  = '1' /* ANOXIC BRAIN DAMAGE */
OTHER   = '0' ;

/* Birth trauma */
/* ICD-9-CM diagnosis codes: */

VALUE $BIRTIDA
'7670'  = '1' /* SUBDURAL AND CEREBRAL HEMORRHAGE (DUE TO TRAUMA OR TO
                 INTRAPARTUM ANOXIA OR HYPOXIA) */
OTHER   = '0' ;

VALUE $BIRTIDB
'7673'  = '1' /* INJURIES TO SKELETON (EXCLUDES CLAVICLE) */
'7674'  = '1' /* INJURY TO SPINE AND SPINAL CORD */
OTHER   = '0' ;

VALUE $BIRTIDC
'7677'  = '1' /* OTHER CRANIAL AND PERIPHERAL NERVE INJURIES */
'7678'  = '1' /* OTHER SPECIFIED BIRTH TRAUMA */
'7679'  = '1' /* BIRTH TRAUMA, UNSPECIFIED */
OTHER   = '0' ;

/* Bone malignancy */
/* ICD-9-CM diagnosis codes (all 4th and 5th digits) : */

VALUE $BONEMID
```

```
'170'   = '1' /* MALIGNANT NEOPLASM OF BONE AND ARTICULAR CARTILAGE */
'1700'  = '1' /* MAL NEO SKULL/FACE BONE */
'1701'  = '1' /* MALIGNANT NEO MANDIBLE */
'1702'  = '1' /* MALIG NEO VERTEBRAE */
'1703'  = '1' /* MAL NEO RIBS/STERN/CLAV */
'1704'  = '1' /* MAL NEO LONG BONES ARM */
'1705'  = '1' /* MAL NEO BONES WRIST/HAND */
'1706'  = '1' /* MAL NEO PELVIC GIRDLE */
'1707'  = '1' /* MAL NEO LONG BONES LEG */
'1708'  = '1' /* MAL NEO BONES ANKLE/FOOT */
'1709'  = '1' /* MALIG NEOPL BONE NOS */
OTHER   = '0' ;

/* Cancer */
/* ICD-9-CM diagnosis codes( all 4th and 5th digits) : */

VALUE $CANCEID
'140'   = '1' /* MALIGNANT NEOPLASM OF LIP */
'1400'  = '1' /* MAL NEO UPPER VERMILION*/
'1401'  = '1' /* MAL NEO LOWER VERMILION */
'1403'  = '1' /* MAL NEO UPPER LIP, INNER */
```

```
'1404'  = '1' /* MAL NEO LOWER LIP, INNER */
'1405'  = '1' /* MAL NEO LIP, INNER NOS */
'1406'  = '1' /* MAL NEO LIP, COMMISSURE */
'1408'  = '1' /* MAL NEO LIP NEC */
'1409'  = '1' /* MAL NEO LIP/VERMIL NOS */
'141'   = '1' /* MALIGANANT NEOPLASM OF TONGUE */
'1410'  = '1' /* MAL NEO TONGUE BASE */
'1411'  = '1' /* MAL NEO DORSAL TONGUE */
'1412'  = '1' /* MAL NEO TIP/LAT TONGUE */
'1413'  = '1' /* MAL NEO VENTRAL TONGUE */
'1414'  = '1' /* MAL NEO ANT 2/3 TONGUE */
'1415'  = '1' /* MAL NEO TONGUE JUNCTION */
'1416'  = '1' /* MAL NEO LINGUAL TONSIL */
'1418'  = '1' /* MALIG NEO TONGUE NEC */
'1419'  = '1' /* MALIG NEO TONGUE NOS */
'142'   = '1' /* MALIGNANT NEOPLASM OF MAJORITY SALIVARY GLANDS */
'1420'  = '1' /* MALIG NEO PAROTID */
'1421'  = '1' /* MALIG NEO SUBMANDIBULAR */
'1422'  = '1' /* MALIG NEO SUBLINGUAL */
'1428'  = '1' /* MAL NEO MAJ SALIVARY NEC */
'1429'  = '1' /* MAL NEO SALIVARY NOS */
'143'   = '1' /* MALIGNANT NEOPLASM OF GUM */
'1430'  = '1' /* MALIG NEO UPPER GUM */
'1431'  = '1' /* MALIG NEO LOWER GUM */
'1438'  = '1' /* MALIG NEO GUM NEC */
'1439'  = '1' /* MALIG NEO GUM NOS */
'144'   = '1' /* MALIGNANT NEOPLASM OF FLOOR OF MOUTH */
'1440'  = '1' /* MAL NEO ANT FLOOR MOUTH */
'1441'  = '1' /* MAL NEO LAT FLOOR MOUTH */
'1448'  = '1' /* MAL NEO MOUTH FLOOR NEC */
'1449'  = '1' /* MAL NEO MOUTH FLOOR NOS */
'145'   = '1' /* MALIGNANT NEOPLASM OF OTHER AND UNSPECIFIED PARTS OF */
                 MOUTH */
'1450'  = '1' /* MAL NEO CHEEK MUCOSA */
'1451'  = '1' /* MAL NEO MOUTH VESTIBULE */
'1452'  = '1' /* MALIG NEO HARD PALATE */
'1453'  = '1' /* MALIG NEO SOFT PALATE */
'1454'  = '1' /* MALIGNANT NEOPLASM UVULA */
```

PSFMTS -- CONTINUED

```
'1455'  = '1' /* MALIGNANT NEO PALATE NOS */
'1456'  = '1' /* MALIG NEO RETROMOLAR */
'1458'  = '1' /* MALIG NEOPLASM MOUTH NEC */
'1459'  = '1' /* MALIG NEOPLASM MOUTH NOS */
'146'   = '1' /* MALIGNANT NEOPLASM OF OROPHARYNX */
'1460'  = '1' /* MALIGNANT NEOPL TONSIL */
'1461'  = '1' /* MAL NEO TONSILLAR FOSSA */
'1462'  = '1' /*  MAL NEO TONSIL PILLARS */
'1463'  = '1' /* MALIGN NEOPL VALLECULA */
'1464'  = '1' /* MAL NEO ANT EPIGLOTTIS */
'1465'  = '1' /* MAL NEO EPIGLOTTIS JUNCT */
'1466'  = '1' /* MAL NEO LAT OROPHARYNX */
'1467'  = '1' /* MAL NEO POST OROPHARYNX */
'1468'  = '1' /* MAL NEO OROPHARYNX NEC */
'1469'  = '1' /* MALIG NEO OROPHARYNX NOS */
'147'   = '1' /* MALIGNANT NEOPLASM OF NASOPHARYNX */
'1470'  = '1' /* MAL NEO SUPER NASOPHARYN */
'1471'  = '1' /* MAL NEO POST NASOPHARYNX */
```

```
    '1472'  = '1' /* MAL NEO LAT NASOPHARYNX */
    '1473'  = '1' /* MAL NEO ANT NASOPHARYNX */
    '1478'  = '1' /* MAL NEO NASOPHARYNX NEC */
    '1479'  = '1' /* MAL NEO NASOPHARYNX NOS */
    '148'   = '1' /* MALIGNANT NEOPLASM OF HYPOPHARYNX */
    '1480'  = '1' /* MAL NEO POSTCRICOID */
    '1481'  = '1' /* MAL NEO PYRIFORM SINUS */
    '1482'  = '1' /* MAL NEO ARYEPIGLOTT FOLD */
    '1483'  = '1' /* MAL NEO POST HYPOPHARYNX */
    '1488'  = '1' /* MAL NEO HYPOPHARYNX NEC */
    '1489'  = '1' /* MAL NEO HYPOPHARYNX NOS */
    '149'   = '1' /* MALIGNANT NEOPLASM OF OTHER AND ILL-DEFINED SITES WITHIN
                     THE LIP, ORAL CAVITY, AND PHARYNX */
    '1490'  = '1' /* MAL NEO PHARYNX NOS */
    '1491'  = '1' /* MAL NEO WALDEYERS RING */
    '1498'  = '1' /* MAL NEO ORAL/PHARYNX NEC */
    '1499'  = '1' /* MAL NEO OROPHRYN ILL-DEF */
    '150'   = '1' /* MALIGNANT NEOPLASM OF EXOPHAGUS */
    '1500'  = '1' /* MAL NEO CERVICAL ESOPHAG */
    '1501'  = '1' /* MAL NEO THORACIC ESOPHAG */
    '1502'  = '1' /* MAL NEO ABDOMIN ESOPHAG */
    '1503'  = '1' /* MAL NEO UPPER 3RD ESOPH */
    '1504'  = '1' /* MAL NEO MIDDLE 3RD ESOPH */
    '1505'  = '1' /* MAL NEO LOWER 3RD ESOPH */
    '1508'  = '1' /* MAL NEO ESOPHAGUS NEC */
    '1509'  = '1' /* MAL NEO ESOPHAGUS NOS */
    '151'   = '1' /* MALIGNANT NEOPLASM OF STOMACH */
    '1510'  = '1' /* MAL NEO STOMACH CARDIA */
    '1511'  = '1' /* MALIGNANT NEO PYLORUS */
    '1512'  = '1' /* MAL NEO PYLORIC ANTRUM */
    '1513'  = '1' /* MAL NEO STOMACH FUNDUS */
    '1514'  = '1' /* MAL NEO STOMACH BODY */
    '1515'  = '1' /* MAL NEO STOM LESSER CURV */
    '1516'  = '1' /* MAL NEO STOM GREAT CURV */
    '1518'  = '1' /* MALIG NEOPL STOMACH NEC */
    '1519'  = '1' /* MALIG NEOPL STOMACH NOS */
    '152'   = '1' /* MALIGNANT NEOPLASM OF SMALL INTESTINE, INCLUDING DUODENUM
*/
    '1520'  = '1' /* MALIGNANT NEOPL DUODENUM */
    '1521'  = '1' /* MALIGNANT NEOPL JEJUNUM */
    '1522'  = '1' /* MALIGNANT NEOPLASM ILEUM */
    PSFMTS -- CONTINUED

    '1523'  = '1' /* MAL NEO MECKELS DIVERT */
    '1528'  = '1' /* MAL NEO SMALL BOWEL NEC */
    '1529'  = '1' /* MAL NEO SMALL BOWEL NOS */
    '153'   = '1' /* MALIGNANT NEOPLASM OF COLON */
    '1530'  = '1' /* MAL NEO HEPATIC FLEXURE */
    '1531'  = '1' /* MAL NEO TRANSVERSE COLON */
    '1532'  = '1' /* MAL NEO DESCEND COLON */
    '1533'  = '1' /* MAL NEO SIGMOID COLON */
    '1534'  = '1' /* MALIGNANT NEOPLASM CECUM */
    '1535'  = '1' /* MALIGNANT NEO APPENDIX */
    '1536'  = '1' /* MALIG NEO ASCEND COLON */
    '1537'  = '1' /* MAL NEO SPLENIC FLEXURE */
    '1538'  = '1' /* MALIGNANT NEO COLON NEC */
    '1539'  = '1' /* MALIGNANT NEO COLON NOS */
```

```
    '154'   = '1' /* MALIGNANT NEOPLASM OF RECTUM, RECTOSIGMOID JUNCTION, AND
*/
                    ANUS */
    '1540'  = '1' /* MAL NEO RECTOSIGMOID JCT */
    '1541'  = '1' /* MALIGNANT NEOPL RECTUM */
    '1542'  = '1' /* MALIG NEOPL ANAL CANAL */
    '1543'  = '1' /* MALIGNANT NEO ANUS NOS */
    '1548'  = '1' /* MAL NEO RECTUM/ANUS NEC */
    '155'   = '1' /* MALIGNANT NEOPLASM OF LIVER AND INTRAHEPATIC BILE DUCTS
*/
    '1550'  = '1' /* MAL NEO LIVER, PRIMARY */
    '1551'  = '1' /* MAL NEO INTRAHEPAT DUCTS */
    '1552'  = '1' /* MALIGNANT NEO LIVER NOS */
    '156'   = '1' /* MALIGNANT NEOPLASM OF GALLBLADDER AND EXTRAHEPATIC BILE
*/
                    DUCTS */
    '1560'  = '1' /* MALIG NEO GALLBLADDER */
    '1561'  = '1' /* MAL NEO EXTRAHEPAT DUCTS */
    '1562'  = '1' /* MAL NEO AMPULLA OF VATER */
    '1568'  = '1' /* MALIG NEO BILIARY NEC */
    '1569'  = '1' /* MALIG NEO BILIARY NOS */
    '157'   = '1' /* MALIGNANT NEOPLASM OF PANCREAS */
    '1570'  = '1' /* MAL NEO PANCREAS HEAD */
    '1571'  = '1' /* MAL NEO PANCREAS BODY */
    '1572'  = '1' /* MAL NEO PANCREAS TAIL */
    '1573'  = '1' /* MAL NEO PANCREATIC DUCT */
    '1574'  = '1' /* MAL NEO ISLET LANGERHANS */
    '1578'  = '1' /* MALIG NEO PANCREAS NEC */
    '1579'  = '1' /* MALIG NEO PANCREAS NOS */
    '158'   = '1' /* MALIG NEO PERITONEUM */
    '1580'  = '1' /* MAL NEO RETROPERITONEUM */
    '1588'  = '1' /* MAL NEO PERITONEUM NEC */
    '1589'  = '1' /* MAL NEO PERITONEUM NOS */
    '159'   = '1' /* OTH MALIG NEO GI/PERITON */
    '1590'  = '1' /* MALIG NEO INTESTINE NOS */
    '1591'  = '1' /* MALIGNANT NEO SPLEEN NEC */
    '1598'  = '1' /* MAL NEO GI/INTRA-ABD NEC */
    '1599'  = '1' /* MAL NEO GI TRACT ILL-DEF */
    '160'   = '1' /* MAL NEO NASAL CAV/SINUS */
    '1600'  = '1' /* MAL NEO NASAL CAVITIES */
    '1601'  = '1' /* MALIG NEO MIDDLE EAR */
    '1602'  = '1' /* MAL NEO MAXILLARY SINUS */
    '1603'  = '1' /* MAL NEO ETHMOIDAL SINUS */
    '1604'  = '1' /* MALIG NEO FRONTAL SINUS */
    '1605'  = '1' /* MAL NEO SPHENOID SINUS */
    PSFMTS -- CONTINUED

    '1608'  = '1' /* MAL NEO ACCESS SINUS NEC */
    '1609'  = '1' /* MAL NEO ACCESS SINUS NOS */
    '161'   = '1' /* MALIGNANT NEO LARYNX */
    '1610'  = '1' /* MALIGNANT NEO GLOTTIS */
    '1611'  = '1' /* MALIG NEO SUPRAGLOTTIS */
    '1612'  = '1' /* MALIG NEO SUBGLOTTIS */
    '1613'  = '1' /* MAL NEO CARTILAGE LARYNX */
    '1618'  = '1' /* MALIGNANT NEO LARYNX NEC */
    '1619'  = '1' /* MALIGNANT NEO LARYNX NOS */
    '162'   = '1' /* MAL NEO TRACHEA/LUNG */
    '1620'  = '1' /* MALIGNANT NEO TRACHEA */
```

```
'1622'  = '1' /* MALIG NEO MAIN BRONCHUS */
'1623'  = '1' /* MAL NEO UPPER LOBE LUNG */
'1624'  = '1' /* MAL NEO MIDDLE LOBE LUNG */
'1625'  = '1' /* MAL NEO LOWER LOBE LUNG */
'1628'  = '1' /* MAL NEO BRONCH/LUNG NEC */
'1629'  = '1' /* MAL NEO BRONCH/LUNG NOS */
'163'   = '1' /* MALIGNANT NEOPL PLEURA */
'1630'  = '1' /* MAL NEO PARIETAL PLEURA */
'1631'  = '1' /* MAL NEO VISCERAL PLEURA */
'1638'  = '1' /* MALIG NEOPL PLEURA NEC */
'1639'  = '1' /* MALIG NEOPL PLEURA NOS */
'164'   = '1' /* MAL NEO THYMUS/MEDIASTIN */
'1640'  = '1' /* MALIGNANT NEOPL THYMUS */
'1641'  = '1' /* MALIGNANT NEOPL HEART */
'1642'  = '1' /* MAL NEO ANT MEDIASTINUM */
'1643'  = '1' /* MAL NEO POST MEDIASTINUM */
'1648'  = '1' /* MAL NEO MEDIASTINUM NEC */
'1649'  = '1' /* MAL NEO MEDIASTINUM NOS */
'165'   = '1' /* OTH/ILL-DEF MAL NEO RESP */
'1650'  = '1' /* MAL NEO UPPER RESP NOS */
'1658'  = '1' /* MAL NEO THORAX/RESP NEC */
'1659'  = '1' /* MAL NEO RESP SYSTEM NOS */
'170'   = '1' /* MAL NEO BONE/ARTIC CART */
'1700'  = '1' /* MAL NEO SKULL/FACE BONE */
'1701'  = '1' /* MALIGNANT NEO MANDIBLE */
'1702'  = '1' /* MALIG NEO VERTEBRAE */
'1703'  = '1' /* MAL NEO RIBS/STERN/CLAV */
'1704'  = '1' /* MAL NEO LONG BONES ARM */
'1705'  = '1' /* MAL NEO BONES WRIST/HAND */
'1706'  = '1' /* MAL NEO PELVIC GIRDLE */
'1707'  = '1' /* MAL NEO LONG BONES LEG */
'1708'  = '1' /* MAL NEO BONES ANKLE/FOOT */
'1709'  = '1' /* MALIG NEOPL BONE NOS */
'171'   = '1' /* MAL NEO SOFT TISSUE */
'1710'  = '1' /* MAL NEO SOFT TISSUE HEAD */
'1712'  = '1' /* MAL NEO SOFT TISSUE ARM */
'1713'  = '1' /* MAL NEO SOFT TISSUE LEG */
'1714'  = '1' /* MAL NEO SOFT TIS THORAX */
'1715'  = '1' /* MAL NEO SOFT TIS ABDOMEN */
'1716'  = '1' /* MAL NEO SOFT TIS PELVIS */
'1717'  = '1' /* MAL NEOPL TRUNK NOS */
'1718'  = '1' /* MAL NEO SOFT TISSUE NEC */
'1719'  = '1' /* MAL NEO SOFT TISSUE NOS */
'172'   = '1' /* MALIGNANT MELANOMA SKIN */
'1720'  = '1' /* MALIG MELANOMA LIP */
'1721'  = '1' /* MALIG MELANOMA EYELID */
'1722'  = '1' /* MALIG MELANOMA EAR */
'1723'  = '1' /* MAL MELANOM FACE NEC/NOS */
```

PSFMTS -- CONTINUED

```
'1724'  = '1' /* MAL MELANOMA SCALP/NECK */
'1725'  = '1' /* MALIG MELANOMA TRUNK */
'1726'  = '1' /* MALIG MELANOMA ARM */
'1727'  = '1' /* MALIG MELANOMA LEG */
'1728'  = '1' /* MALIG MELANOMA SKIN NEC */
'1729'  = '1' /* MALIG MELANOMA SKIN NOS */
'174'   = '1' /* MALIG NEO FEMALE BREAST */
'1740'  = '1' /* MALIG NEO NIPPLE */
```

```
'1741'  = '1' /* MAL NEO BREAST-CENTRAL */
'1742'  = '1' /* MAL NEO BREAST UP-INNER */
'1743'  = '1' /* MAL NEO BREAST LOW-INNER */
'1744'  = '1' /* MAL NEO BREAST UP-OUTER */
'1745'  = '1' /* MAL NEO BREAST LOW-OUTER */
'1746'  = '1' /* MAL NEO BREAST-AXILLARY */
'1748'  = '1' /* MALIGN NEOPL BREAST NEC */
'1749'  = '1' /* MALIGN NEOPL BREAST NOS */
'175'   = '1' /* MALIG NEO MALE BREAST */
'1750'  = '1' /* MAL NEO MALE NIPPLE */
'1759'  = '1' /* MAL NEO MALE BREAST NEC */
'1760'  = '1' /* SKIN - KAPOSIS SARCOMA */
'1761'  = '1' /* SFT TISUE - KPSIS SRCMA */
'1762'  = '1' /* PALATE - KPSIS SARCOMA */
'1763'  = '1' /* GI SITES - KPSIS SRCOMA */
'1764'  = '1' /* LUNG - KAPOSIS SARCOMA */
'1765'  = '1' /* LYM NDS - KPSIS SARCOMA */
'1768'  = '1' /* SPF STS - KPSIS SARCOMA */
'1769'  = '1' /* KAPOSIS SARCOMA NOS */
'179'   = '1' /* MALIG NEOPL UTERUS NOS */
'180'   = '1' /* MALIG NEOPL CERVIX UTERI */
'1800'  = '1' /* MALIG NEO ENDOCERVIX */
'1801'  = '1' /* MALIG NEO EXOCERVIX */
'1808'  = '1' /* MALIG NEO CERVIX NEC */
'1809'  = '1' /* MAL NEO CERVIX UTERI NOS */
'181'   = '1' /* MALIGNANT NEOPL PLACENTA */
'182'   = '1' /* MALIG NEOPL UTERUS BODY */
'1820'  = '1' /* MALIG NEO CORPUS UTERI */
'1821'  = '1' /* MAL NEO UTERINE ISTHMUS */
'1828'  = '1' /* MAL NEO BODY UTERUS NEC */
'183'   = '1' /* MAL NEO UTERINE ADNEXA */
'1830'  = '1' /* MALIGN NEOPL OVARY */
'1832'  = '1' /* MAL NEO FALLOPIAN TUBE */
'1833'  = '1' /* MAL NEO BROAD LIGAMENT */
'1834'  = '1' /* MALIG NEO PARAMETRIUM */
'1835'  = '1' /* MAL NEO ROUND LIGAMENT */
'1838'  = '1' /* MAL NEO ADNEXA NEC */
'1839'  = '1' /* MAL NEO ADNEXA NOS */
'184'   = '1' /* MAL NEO FEM GEN NEC/NOS */
'1840'  = '1' /* MALIGN NEOPL VAGINA */
'1841'  = '1' /* MAL NEO LABIA MAJORA */
'1842'  = '1' /* MAL NEO LABIA MINORA */
'1843'  = '1' /* MALIGN NEOPL CLITORIS */
'1844'  = '1' /* MALIGN NEOPL VULVA NOS */
'1848'  = '1' /* MAL NEO FEMALE GENIT NEC */
'1849'  = '1' /* MAL NEO FEMALE GENIT NOS */
'185'   = '1' /* MALIGN NEOPL PROSTATE */
'186'   = '1' /* MALIGN NEOPL TESTIS */
'1860'  = '1' /* MAL NEO UNDESCEND TESTIS */
'1869'  = '1' /* MALIG NEO TESTIS NEC */
'187'   = '1' /* MAL NEO MALE GENITAL NEC */
```

PSFMTS -- CONTINUED

```
'1871'  = '1' /* MALIGN NEOPL PREPUCE */
'1872'  = '1' /* MALIG NEO GLANS PENIS */
'1873'  = '1' /* MALIG NEO PENIS BODY */
'1874'  = '1' /* MALIG NEO PENIS NOS */
'1875'  = '1' /* MALIG NEO EPIDIDYMIS */
```

```
'1876'  = '1' /* MAL NEO SPERMATIC CORD */
'1877'  = '1' /* MALIGN NEOPL SCROTUM */
'1878'  = '1' /* MAL NEO MALE GENITAL NEC */
'1879'  = '1' /* MAL NEO MALE GENITAL NOS */
'188'   = '1' /* MALIGN NEOPL BLADDER */
'1880'  = '1' /* MAL NEO BLADDER-TRIGONE */
'1881'  = '1' /* MAL NEO BLADDER-DOME */
'1882'  = '1' /* MAL NEO BLADDER-LATERAL */
'1883'  = '1' /* MAL NEO BLADDER-ANTERIOR */
'1884'  = '1' /* MAL NEO BLADDER-POST */
'1885'  = '1' /* MAL NEO BLADDER NECK */
'1886'  = '1' /* MAL NEO URETERIC ORIFICE */
'1887'  = '1' /* MALIG NEO URACHUS */
'1888'  = '1' /* MALIG NEO BLADDER NEC */
'1889'  = '1' /* MALIG NEO BLADDER NOS */
'189'   = '1' /* MAL NEO URINARY NEC/NOS */
'1890'  = '1' /* MALIG NEOPL KIDNEY */
'1891'  = '1' /* MALIG NEO RENAL PELVIS */
'1892'  = '1' /* MALIGN NEOPL URETER */
'1893'  = '1' /* MALIGN NEOPL URETHRA */
'1894'  = '1' /* MAL NEO PARAURETHRAL */
'1898'  = '1' /* MAL NEO URINARY NEC */
'1899'  = '1' /* MAL NEO URINARY NOS */
'190'   = '1' /* MALIGNANT NEOPLASM EYE */
'1900'  = '1' /* MALIGN NEOPL EYEBALL */
'1901'  = '1' /* MALIGN NEOPL ORBIT */
'1902'  = '1' /* MAL NEO LACRIMAL GLAND */
'1903'  = '1' /* MAL NEO CONJUNCTIVA */
'1904'  = '1' /* MALIGN NEOPL CORNEA */
'1905'  = '1' /* MALIGN NEOPL RETINA */
'1906'  = '1' /* MALIGN NEOPL CHOROID */
'1907'  = '1' /* MAL NEO LACRIMAL DUCT */
'1908'  = '1' /* MALIGN NEOPL EYE NEC */
'1909'  = '1' /* MALIGN NEOPL EYE NOS */
'191'   = '1' /* MALIGNANT NEOPLASM BRAIN */
'1910'  = '1' /* MALIGN NEOPL CEREBRUM */
'1911'  = '1' /* MALIG NEO FRONTAL LOBE */
'1912'  = '1' /* MAL NEO TEMPORAL LOBE */
'1913'  = '1' /* MAL NEO PARIETAL LOBE */
'1914'  = '1' /* MAL NEO OCCIPITAL LOBE */
'1915'  = '1' /* MAL NEO CEREB VENTRICLE */
'1916'  = '1' /* MAL NEO CEREBELLUM NOS */
'1917'  = '1' /* MAL NEO BRAIN STEM */
'1918'  = '1' /* MALIG NEO BRAIN NEC */
'1919'  = '1' /* MALIG NEO BRAIN NOS */
'192'   = '1' /* MAL NEO NERVE NEC/NOS */
'1920'  = '1' /* MAL NEO CRANIAL NERVES */
'1921'  = '1' /* MAL NEO CEREBRAL MENING */
'1922'  = '1' /* MAL NEO SPINAL CORD */
'1923'  = '1' /* MAL NEO SPINAL MENINGES */
'1928'  = '1' /* MAL NEO NERVOUS SYST NEC */
'1929'  = '1' /* MAL NEO NERVOUS SYST NOS */
'193'   = '1' /* MALIGN NEOPL THYROID */
'194'   = '1' /* MAL NEO OTHER ENDOCRINE */
```

PSFMTS -- CONTINUED

```
'1940'  = '1' /* MALIGN NEOPL ADRENAL */
'1941'  = '1' /* MALIG NEO PARATHYROID */
```

```
'1943'  = '1' /* MALIG NEO PITUITARY */
'1944'  = '1' /* MALIGN NEO PINEAL GLAND */
'1945'  = '1' /* MAL NEO CAROTID BODY */
'1946'  = '1' /* MAL NEO PARAGANGLIA NEC */
'1948'  = '1' /* MAL NEO ENDOCRINE NEC */
'1949'  = '1' /* MAL NEO ENDOCRINE NOS */
'195'   = '1' /* MAL NEO OTH/ILL-DEF SITE */
'1950'  = '1' /* MAL NEO HEAD/FACE/NECK */
'1951'  = '1' /* MALIGN NEOPL THORAX */
'1952'  = '1' /* MALIG NEO ABDOMEN */
'1953'  = '1' /* MALIGN NEOPL PELVIS */
'1954'  = '1' /* MALIGN NEOPL ARM */
'1955'  = '1' /* MALIGN NEOPL LEG */
'1958'  = '1' /* MALIG NEO SITE NEC */
'196'   = '1' /* MALIG NEO LYMPH NODES */
'1960'  = '1' /* MAL NEO LYMPH-HEAD/NECK */
'1961'  = '1' /* MAL NEO LYMPH-INTRATHOR */
'1962'  = '1' /* MAL NEO LYMPH INTRA-ABD */
'1963'  = '1' /* MAL NEO LYMPH-AXILLA/ARM */
'1965'  = '1' /* MAL NEO LYMPH-INGUIN/LEG */
'1966'  = '1' /* MAL NEO LYMPH-INTRAPELV */
'1968'  = '1' /* MAL NEO LYMPH NODE-MULT */
'1969'  = '1' /* MAL NEO LYMPH NODE NOS */
'197'   = '1' /* SECONDRY MAL NEO GI/RESP */
'1970'  = '1' /* SECONDARY MALIG NEO LUNG */
'1971'  = '1' /* SEC MAL NEO MEDIASTINUM */
'1972'  = '1' /* SECOND MALIG NEO PLEURA */
'1973'  = '1' /* SEC MALIG NEO RESP NEC */
'1974'  = '1' /* SEC MALIG NEO SM BOWEL */
'1975'  = '1' /* SEC MALIG NEO LG BOWEL */
'1976'  = '1' /* SEC MAL NEO PERITONEUM */
'1977'  = '1' /* SECOND MALIG NEO LIVER */
'1978'  = '1' /* SEC MAL NEO GI NEC */
'198'   = '1' /* SEC MALIG NEO OTH SITES */
'1980'  = '1' /* SECOND MALIG NEO KIDNEY */
'1981'  = '1' /* SEC MALIG NEO URIN NEC */
'1982'  = '1' /* SECONDARY MALIG NEO SKIN */
'1983'  = '1' /* SEC MAL NEO BRAIN/SPINE */
'1984'  = '1' /* SEC MALIG NEO NERVE NEC */
'1985'  = '1' /* SECONDARY MALIG NEO BONE */
'1986'  = '1' /* SECOND MALIG NEO OVARY */
'1987'  = '1' /* SECOND MALIG NEO ADRENAL */
'1988'  = '1' /* OTH SECONDARY MALIG NEO */
'19881' = '1' /* SECOND MALIG NEO BREAST */
'19882' = '1' /* SECOND MALIG NEO GENITAL */
'19889' = '1' /* SECONDARY MALIG NEO NEC */
'199'   = '1' /* MALIGNANT NEOPLASM NOS */
'1990'  = '1' /* MALIG NEO DISSEMINATED */
'1991'  = '1' /* MALIGNANT NEOPLASM NOS */
'200'   = '1' /* LYMPHOSARC/RETICULOSARC */
'2000'  = '1' /* RETICULOSARCOMA */
'20000' = '1' /* RETCLSRC UNSP XTRNDL ORG */
'20001' = '1' /* RETICULOSARCOMA HEAD */
'20002' = '1' /* RETICULOSARCOMA THORAX */
'20003' = '1' /* RETICULOSARCOMA ABDOM */
'20004' = '1' /* RETICULOSARCOMA AXILLA */
'20005' = '1' /* RETICULOSARCOMA INGUIN */
```

PSFMTS -- CONTINUED

```
'20006' = '1' /* RETICULOSARCOMA PELVIC */
'20007' = '1' /* RETICULOSARCOMA SPLEEN */
'20008' = '1' /* RETICULOSARCOMA MULT */
'2001'  = '1' /* LYMPHOSARCOMA */
'20010' = '1' /* LYMPHSRC UNSP XTRNDL ORG */
'20011' = '1' /* LYMPHOSARCOMA HEAD */
'20012' = '1' /* LYMPHOSARCOMA THORAX */
'20013' = '1' /* LYMPHOSARCOMA ABDOM */
'20014' = '1' /* LYMPHOSARCOMA AXILLA */
'20015' = '1' /* LYMPHOSARCOMA INGUIN */
'20016' = '1' /* LYMPHOSARCOMA PELVIC */
'20017' = '1' /* LYMPHOSARCOMA SPLEEN */
'20018' = '1' /* LYMPHOSARCOMA MULT */
'2002'  = '1' /* BURKITTS TUMOR/LYMPHOMA */
'20020' = '1' /* BRKT TMR UNSP XTRNDL ORG */
'20021' = '1' /* BURKITTS TUMOR HEAD */
'20022' = '1' /* BURKITTS TUMOR THORAX */
'20023' = '1' /* BURKITTS TUMOR ABDOM */
'20024' = '1' /* BURKITTS TUMOR AXILLA */
'20025' = '1' /* BURKITTS TUMOR INGUIN */
'20026' = '1' /* BURKITTS TUMOR PELVIC */
'20027' = '1' /* BURKITTS TUMOR SPLEEN */
'20028' = '1' /* BURKITTS TUMOR MULT */
'2008'  = '1' /* MIXED LYMPHOSARCOMA */
'20080' = '1' /* OTH VARN UNSP XTRNDL ORG */
'20081' = '1' /* MIXED LYMPHOSARC HEAD */
'20082' = '1' /* MIXED LYMPHOSARC THORAX */
'20083' = '1' /* MIXED LYMPHOSARC ABDOM */
'20084' = '1' /* MIXED LYMPHOSARC AXILLA */
'20085' = '1' /* MIXED LYMPHOSARC INGUIN */
'20086' = '1' /* MIXED LYMPHOSARC PELVIC */
'20087' = '1' /* MIXED LYMPHOSARC SPLEEN */
'20088' = '1' /* MIXED LYMPHOSARC MULT */
'201'   = '1' /* HODGKINS DISEASE */
'2010'  = '1' /* HODGKINS PARAGRANULOMA */
'20100' = '1' /* HDGK PRG UNSP XTRNDL ORG */
'20101' = '1' /* HODGKINS PARAGRAN HEAD */
'20102' = '1' /* HODGKINS PARAGRAN THORAX */
'20103' = '1' /* HODGKINS PARAGRAN ABDOM */
'20104' = '1' /* HODGKINS PARAGRAN AXILLA */
'20105' = '1' /* HODGKINS PARAGRAN INGUIN */
'20106' = '1' /* HODGKINS PARAGRAN PELVIC */
'20107' = '1' /* HODGKINS PARAGRAN SPLEEN */
'20108' = '1' /* HODGKINS PARAGRAN MULT */
'2011'  = '1' /* HODGKINS GRANULOMA */
'20110' = '1' /* HDGK GRN UNSP XTRNDL ORG */
'20111' = '1' /* HODGKINS GRANULOM HEAD */
'20112' = '1' /* HODGKINS GRANULOM THORAX */
'20113' = '1' /* HODGKINS GRANULOM ABDOM */
'20114' = '1' /* HODGKINS GRANULOM AXILLA */
'20115' = '1' /* HODGKINS GRANULOM INGUIN */
'20116' = '1' /* HODGKINS GRANULOM PELVIC */
'20117' = '1' /* HODGKINS GRANULOM SPLEEN */
'20118' = '1' /* HODGKINS GRANULOM MULT */
'2012'  = '1' /* HODGKINS SARCOMA */
'20120' = '1' /* HDGK SRC UNSP XTRNDL ORG */
'20121' = '1' /* HODGKINS SARCOMA HEAD */
```

```
'20122' = '1' /* HODGKINS SARCOMA THORAX */
'20123' = '1' /* HODGKINS SARCOMA ABDOM */
```
PSFMTS -- CONTINUED

```
'20124' = '1' /* HODGKINS SARCOMA AXILLA */
'20125' = '1' /* HODGKINS SARCOMA INGUIN */
'20126' = '1' /* HODGKINS SARCOMA PELVIC */
'20127' = '1' /* HODGKINS SARCOMA SPLEEN */
'20128' = '1' /* HODGKINS SARCOMA MULT */
'2014'  = '1' /* HODGKINS LYMPH-HISTIOCYT */
'20140' = '1' /* LYM-HST UNSP XTRNDL ORGN */
'20141' = '1' /* HODG LYMPH-HISTIO HEAD */
'20142' = '1' /* HODG LYMPH-HISTIO THORAX */
'20143' = '1' /* HODG LYMPH-HISTIO ABDOM */
'20144' = '1' /* HODG LYMPH-HISTIO AXILLA */
'20145' = '1' /* HODG LYMPH-HISTIO INGUIN */
'20146' = '1' /* HODG LYMPH-HISTIO PELVIC */
'20147' = '1' /* HODG LYMPH-HISTIO SPLEEN */
'20148' = '1' /* HODG LYMPH-HISTIO MULT */
'2015'  = '1' /* HODGKINS NODULAR SCLEROS */
'20150' = '1' /* NDR SCLR UNSP XTRNDL ORG */
'20151' = '1' /* HODG NODUL SCLERO HEAD */
'20152' = '1' /* HODG NODUL SCLERO THORAX */
'20153' = '1' /* HODG NODUL SCLERO ABDOM */
'20154' = '1' /* HODG NODUL SCLERO AXILLA */
'20155' = '1' /* HODG NODUL SCLERO INGUIN */
'20156' = '1' /* HODG NODUL SCLERO PELVIC */
'20157' = '1' /* HODG NODUL SCLERO SPLEEN */
'20158' = '1' /* HODG NODUL SCLERO MULT */
'2016'  = '1' /* HODGKINS MIX CELLULARITY */
'20160' = '1' /* MXD CELR UNSP XTRNDL ORG */
'20161' = '1' /* HODGKINS MIX CELL HEAD */
'20162' = '1' /* HODGKINS MIX CELL THORAX */
'20163' = '1' /* HODGKINS MIX CELL ABDOM */
'20164' = '1' /* HODGKINS MIX CELL AXILLA */
'20165' = '1' /* HODGKINS MIX CELL INGUIN */
'20166' = '1' /* HODGKINS MIX CELL PELVIC */
'20167' = '1' /* HODGKINS MIX CELL SPLEEN */
'20168' = '1' /* HODGKINS MIX CELL MULT */
'2017'  = '1' /* HODG LYMPHOCYTIC DEPLET */
'20170' = '1' /* LYM DPLT UNSP XTRNDL ORG */
'20171' = '1' /* HODG LYMPH DEPLET HEAD */
'20172' = '1' /* HODG LYMPH DEPLET THORAX */
'20173' = '1' /* HODG LYMPH DEPLET ABDOM */
'20174' = '1' /* HODG LYMPH DEPLET AXILLA */
'20175' = '1' /* HODG LYMPH DEPLET INGUIN */
'20176' = '1' /* HODG LYMPH DEPLET PELVIC */
'20177' = '1' /* HODG LYMPH DEPLET SPLEEN */
'20178' = '1' /* HODG LYMPH DEPLET MULT */
'2019'  = '1' /* HODGKINS DISEASE NOS */
'20190' = '1' /* HDGK DIS UNSP XTRNDL ORG */
'20191' = '1' /* HODGKINS DIS NOS HEAD */
'20192' = '1' /* HODGKINS DIS NOS THORAX */
'20193' = '1' /* HODGKINS DIS NOS ABDOM */
'20194' = '1' /* HODGKINS DIS NOS AXILLA */
'20195' = '1' /* HODGKINS DIS NOS INGUIN */
'20196' = '1' /* HODGKINS DIS NOS PELVIC */
'20197' = '1' /* HODGKINS DIS NOS SPLEEN */
```

```
'20198' = '1' /* HODGKINS DIS NOS MULT */
'202'   = '1' /* OTH MAL NEO LYMPH/HISTIO */
'2020'  = '1' /* NODULAR LYMPHOMA */
'20200' = '1' /* NDLR LYM UNSP XTRNDL ORG */
'20201' = '1' /* NODULAR LYMPHOMA HEAD */
```

PSFMTS -- CONTINUED

```
'20202' = '1' /* NODULAR LYMPHOMA THORAX */
'20203' = '1' /* NODULAR LYMPHOMA ABDOM */
'20204' = '1' /* NODULAR LYMPHOMA AXILLA */
'20205' = '1' /* NODULAR LYMPHOMA INGUIN */
'20206' = '1' /* NODULAR LYMPHOMA PELVIC */
'20207' = '1' /* NODULAR LYMPHOMA SPLEEN */
'20208' = '1' /* NODULAR LYMPHOMA MULT */
'2021'  = '1' /* MYCOSIS FUNGOIDES */
'20210' = '1' /* MYCS FNG UNSP XTRNDL ORG */
'20211' = '1' /* MYCOSIS FUNGOIDES HEAD */
'20212' = '1' /* MYCOSIS FUNGOIDES THORAX */
'20213' = '1' /* MYCOSIS FUNGOIDES ABDOM */
'20214' = '1' /* MYCOSIS FUNGOIDES AXILLA */
'20215' = '1' /* MYCOSIS FUNGOIDES INGUIN */
'20216' = '1' /* MYCOSIS FUNGOIDES PELVIC */
'20217' = '1' /* MYCOSIS FUNGOIDES SPLEEN */
'20218' = '1' /* MYCOSIS FUNGOIDES MULT */
'2022'  = '1' /* SEZARYS DISEASE */
'20220' = '1' /* SZRY DIS UNSP XTRNDL ORG */
'20221' = '1' /* SEZARYS DISEASE HEAD */
'20222' = '1' /* SEZARYS DISEASE THORAX */
'20223' = '1' /* SEZARYS DISEASE ABDOM */
'20224' = '1' /* SEZARYS DISEASE AXILLA */
'20225' = '1' /* SEZARYS DISEASE INGUIN */
'20226' = '1' /* SEZARYS DISEASE PELVIC */
'20227' = '1' /* SEZARYS DISEASE SPLEEN */
'20228' = '1' /* SEZARYS DISEASE MULT */
'2023'  = '1' /* MALIGNANT HISTIOCYTOSIS */
'20230' = '1' /* MLG HIST UNSP XTRNDL ORG */
'20231' = '1' /* MAL HISTIOCYTOSIS HEAD */
'20232' = '1' /* MAL HISTIOCYTOSIS THORAX */
'20233' = '1' /* MAL HISTIOCYTOSIS ABDOM */
'20234' = '1' /* MAL HISTIOCYTOSIS AXILLA */
'20235' = '1' /* MAL HISTIOCYTOSIS INGUIN */
'20236' = '1' /* MAL HISTIOCYTOSIS PELVIC */
'20237' = '1' /* MAL HISTIOCYTOSIS SPLEEN */
'20238' = '1' /* MAL HISTIOCYTOSIS MULT */
'2024'  = '1' /* LEUKEM RETICULOENDOTHEL */
'20240' = '1' /* LK RTCTL UNSP XTRNDL ORG */
'20241' = '1' /* HAIRY-CELL LEUKEM HEAD */
'20242' = '1' /* HAIRY-CELL LEUKEM THORAX */
'20243' = '1' /* HAIRY-CELL LEUKEM ABDOM */
'20244' = '1' /* HAIRY-CELL LEUKEM AXILLA */
'20245' = '1' /* HAIRY-CELL LEUKEM INGUIN */
'20246' = '1' /* HAIRY-CELL LEUKEM PELVIC */
'20247' = '1' /* HAIRY-CELL LEUKEM SPLEEN */
'20248' = '1' /* HAIRY-CELL LEUKEM MULT */
'2025'  = '1' /* LETTERER-SIWE DISEASE */
'20250' = '1' /* LTR-SIWE UNSP XTRNDL ORG */
'20251' = '1' /* LETTERER-SIWE DIS HEAD */
'20252' = '1' /* LETTERER-SIWE DIS THORAX */
```

```
'20253' = '1' /* LETTERER-SIWE DIS ABDOM */
'20254' = '1' /* LETTERER-SIWE DIS AXILLA */
'20255' = '1' /* LETTERER-SIWE DIS INGUIN */
'20256' = '1' /* LETTERER-SIWE DIS PELVIC */
'20257' = '1' /* LETTERER-SIWE DIS SPLEEN */
'20258' = '1' /* LETTERER-SIWE DIS MULT */
'2026'  = '1' /* MALIG MAST CELL TUMORS */
'20260' = '1' /* MLG MAST UNSP XTRNDL ORG */
```

PSFMTS -- CONTINUED

```
'20261' = '1' /* MAL MASTOCYTOSIS HEAD */
'20262' = '1' /* MAL MASTOCYTOSIS THORAX */
'20263' = '1' /* MAL MASTOCYTOSIS ABDOM */
'20264' = '1' /* MAL MASTOCYTOSIS AXILLA */
'20265' = '1' /* MAL MASTOCYTOSIS INGUIN */
'20266' = '1' /* MAL MASTOCYTOSIS PELVIC */
'20267' = '1' /* MAL MASTOCYTOSIS SPLEEN */
'20268' = '1' /* MAL MASTOCYTOSIS MULT */
'2028'  = '1' /* LYMPHOMAS NEC */
'20280' = '1' /* OTH LYMP UNSP XTRNDL ORG */
'20281' = '1' /* LYMPHOMAS NEC HEAD */
'20282' = '1' /* LYMPHOMAS NEC THORAX */
'20283' = '1' /* LYMPHOMAS NEC ABDOM */
'20284' = '1' /* LYMPHOMAS NEC AXILLA */
'20285' = '1' /* LYMPHOMAS NEC INGUIN */
'20286' = '1' /* LYMPHOMAS NEC PELVIC */
'20287' = '1' /* LYMPHOMAS NEC SPLEEN */
'20288' = '1' /* LYMPHOMAS NEC MULT */
'2029'  = '1' /* MAL NEO LYM/HIST TIS NEC */
'20290' = '1' /* UNSP LYM UNSP XTRNDL ORG */
'20291' = '1' /* LYMPHOID MAL NEC HEAD */
'20292' = '1' /* LYMPHOID MAL NEC THORAX */
'20293' = '1' /* LYMPHOID MAL NEC ABDOM */
'20294' = '1' /* LYMPHOID MAL NEC AXILLA */
'20295' = '1' /* LYMPHOID MAL NEC INGUIN */
'20296' = '1' /* LYMPHOID MAL NEC PELVIC */
'20297' = '1' /* LYMPHOID MAL NEC SPLEEN */
'20298' = '1' /* LYMPHOID MAL NEC MULT */
'203'   = '1' /* MULTIPLE MYELOMA ET AL */
'2030'  = '1' /* MULTIPLE MYELOMA */
'20300' = '1' /* MULT MYELM W/O REMISSION */
'20301' = '1' /* MULT MYELM W REMISSION */
'2031'  = '1' /* PLASMA CELL LEUKEMIA */
'20310' = '1' /* PLSM CELL LEUK W/O RMSON */
'20311' = '1' /* PLSM CELL LEUK W RMSON */
'2038'  = '1' /* IMMUNOPROLIFERAT NEO NEC */
'20380' = '1' /* OTH IMNPRFL NPL W/O RMSN */
'20381' = '1' /* OTH IMNPRFL NPL W RMSN */
'204'   = '1' /* LYMPHOID LEUKEMIA */
'2040'  = '1' /* ACUTE LYMPHOID LEUKEMIA */
'20400' = '1' /* ACT LYM LEUK W/O RMSION */
'20401' = '1' /* ACT LYM LEUK W RMSION */
'2041'  = '1' /* CHR LYMPHOID LEUKEMIA */
'20410' = '1' /* CHR LYM LEUK W/O RMSION */
'20411' = '1' /* CHR LYM LEUK W RMSION */
'2042'  = '1' /* SUBAC LYMPHOID LEUKEMIA */
'20420' = '1' /* SBAC LYM LEUK W/O RMSION */
'20421' = '1' /* SBAC LYM LEUK W RMSION */
```

```
'2048'  = '1' /* LYMPHOID LEUKEMIA NEC */
'20480' = '1' /* OTH LYM LEUK W/O RMSION */
'20481' = '1' /* OTH LYM LEUK W RMSION */
'2049'  = '1' /* LYMPHOID LEUKEMIA NOS */
'20490' = '1' /* UNS LYM LEUK W/O RMSION */
'20491' = '1' /* UNS LYM LEUK W RMSION */
'205'   = '1' /* MYELOID LEUKEMIA */
'2050'  = '1' /* ACUTE MYELOID LEUKEMIA */
'20500' = '1' /* ACT MYL LEUK W/O RMSION */
'20501' = '1' /* ACT MYL LEUK W RMSION */
'2051'  = '1' /* CHRONIC MYELOID LEUKEMIA */
```
PSFMTS -- CONTINUED

```
'20510' = '1' /* CHR MYL LEUK W/O RMSION */
'20511' = '1' /* CHR MYL LEUK W RMSION */
'2052'  = '1' /* SUBACUT MYELOID LEUKEMIA */
'20520' = '1' /* SBAC MYL LEUK W/O RMSION */
'20521' = '1' /* SBAC MYL LEUK W RMSION */
'2053'  = '1' /* MYELOID SARCOMA */
'20530' = '1' /* MYL SRCOMA W/O RMSION */
'20531' = '1' /* MYL SRCOMA W RMSION */
'2058'  = '1' /* MYELOID LEUKEMIA NEC */
'20580' = '1' /* OTH MYL LEUK W/O RMSION */
'20581' = '1' /* OTH MYL LEUK W RMSION */
'2059'  = '1' /* MYELOID LEUKEMIA NOS */
'20590' = '1' /* UNS MYL LEUK W/O RMSION */
'20591' = '1' /* UNS MYL LEUK W RMSION */
'206'   = '1' /* MONOCYTIC LEUKEMIA */
'2060'  = '1' /* ACUTE MONOCYTIC LEUKEMIA */
'20600' = '1' /* ACT MONO LEUK W/O RMSION */
'20601' = '1' /* ACT MONO LEUK W RMSION */
'2061'  = '1' /* CHR MONOCYTIC LEUKEMIA */
'20610' = '1' /* CHR MONO LEUK W/O RMSION */
'20611' = '1' /* CHR MONO LEUK W RMSION */
'2062'  = '1' /* SUBAC MONOCYTIC LEUKEMIA */
'20620' = '1' /* SBAC MONO LEUK W/O RMSON */
'20621' = '1' /* SBAC MONO LEUK W RMSION */
'2068'  = '1' /* MONOCYTIC LEUKEMIA NEC */
'20680' = '1' /* OTH MONO LEUK W/O RMSION */
'20681' = '1' /* OTH MONO LEUK W RMSION */
'2069'  = '1' /* MONOCYTIC LEUKEMIA NOS */
'20690' = '1' /* UNS MONO LEUK W/O RMSION */
'20691' = '1' /* UNS MONO LEUK W RMSION */
'207'   = '1' /* OTHER SPECIFIED LEUKEMIA */
'2070'  = '1' /* ACUTE ERYTHREMIA */
'20700' = '1' /* ACT ERTH/ERYLK W/O RMSON */
'20701' = '1' /* ACT ERTH/ERYLK W RMSON */
'2071'  = '1' /* CHRONIC ERYTHREMIA */
'20710' = '1' /* CHR ERYTHRM W/O REMISION */
'20711' = '1' /* CHR ERYTHRM W REMISION */
'2072'  = '1' /* MEGAKARYOCYTIC LEUKEMIA */
'20720' = '1' /* MGKRYCYT LEUK W/O RMSION */
'20721' = '1' /* MGKRYCYT LEUK W RMSION */
'2078'  = '1' /* SPECIFIED LEUKEMIA NEC */
'20780' = '1' /* OTH SPF LEUK W/O REMSION */
'20781' = '1' /* OTH SPF LEUK W REMSION */
'208'   = '1' /* LEUKEMIA-UNSPECIF CELL */
'2080'  = '1' /* ACT LEUK UNS CL W/O RMSN */
```

```
    '20800' = '1' /* ACT LEUK UNS CL W/O RMSN */
    '20801' = '1' /* ACT LEUK UNS CL W RMSON */
    '2081'  = '1' /* CHRONIC LEUKEMIA NOS */
    '20810' = '1' /* CHR LEUK UNS CL W/O RMSN */
    '20811' = '1' /* CHR LEUK UNS CL W RMSON */
    '2082'  = '1' /* SUBACUTE LEUKEMIA NOS */
    '20820' = '1' /* SBAC LEUK UNS CL W/O RMS */
    '20821' = '1' /* SBAC LEUK UNS CL W RMSON */
    '2088'  = '1' /* LEUKEMIA-UNSPEC CELL NEC */
    '20880' = '1' /* OTH LEUK UNS CL W/O RMSN */
    '20881' = '1' /* OTH LEUK UNS CL W RMSON */
    '2089'  = '1' /* LEUKEMIA-UNSPEC CELL NOS */
    '20890' = '1' /* LEUKEMIA NOS W/O REMSION */
    '20891' = '1' /* LEUKEMIA NOS W REMISSIO */
```
PSFMTS -- CONTINUED

```
    '2386'  = '1' /* NEOPLASM OF UNCERTAIN BEHAVIOR OF OTHER AND UNSPECIFIED
*/
                  SITES AND TISSUES, PLASMA CELLS */
    '2733'  = '1' /* DISORDERS OF PLASMA PROTEIN METABOLISM-MACROGLOBULINEMIA
*/
    'V1000' = '1' /* GASTROINTESTINAL TRACT, UNSPECIFIED */
    'V1001' = '1' /* TONGUE */
    'V1002' = '1' /* OTHER AND UNSPECIFIED ORAL CAVITY AND PHARYNX */
    'V1003' = '1' /* ESOPHAGUS */
    'V1004' = '1' /* STOMACH */
    'V1005' = '1' /* LARGE INTESTINE */
    'V1006' = '1' /* RECTUM, RECTOSIGMOID JUNCTION, AND ANUS */
    'V1007' = '1' /* LIVER */
    'V1009' = '1' /* OTHER */
    'V1011' = '1' /* BRONCHUS AND LUNG */
    'V1012' = '1' /* TRACHEA */
    'V1020' = '1' /* RESPIRATORY ORGAN, UNSPECIFIED */
    'V1021' = '1' /* LARYNX */
    'V1022' = '1' /* NASAL CAVITIES, MIDDLE EAR, AND ACCESSORY SINUSES */
    'V1029' = '1' /* OTHER */
    'V103'  = '1' /* BREAST */
    'V1040' = '1' /* FEMALE GENITAL ORGAN, UNSPECIFIED */
    'V1041' = '1' /* CERVIX UTERI */
    'V1042' = '1' /* OTHER PARTS OF UTERUS */
    'V1043' = '1' /* OVARY */
    'V1044' = '1' /* OTHER FEMALE GENITAL ORGANS */
    'V1045' = '1' /* MALE GENITAL ORGAN, UNSPECIFIED */
    'V1046' = '1' /* PROSTATE */
    'V1047' = '1' /* TESTIS */
    'V1048' = '1' /* EPIDDIDYMIS */
    'V1049' = '1' /* OTHER MALE GENITAL ORGANS */
    'V1050' = '1' /* URINARY ORGAN, UNSPECIFIED */
    'V1051' = '1' /* BLADDER */
    'V1052' = '1' /* KIDNEY */
    'V1059' = '1' /* OTHER */
    'V1060' = '1' /* LYMPHOSARCOMA AND RETICULOSARCOMA */
    'V1061' = '1' /* HODGKINS DISEASE */
    'V1062' = '1' /* MYELOID LEUKEMIA */
    'V1063' = '1' /* MONOCYTIC LEUKEMIA */
    'V1069' = '1' /* OTHER */
    'V1071' = '1' /* LYMPHOSARCOMA AND RETICULOSARCOMA */
    'V1072' = '1' /* HODGKINS DISEASE */
```

```
    'V1079' = '1' /* OTHER */
    'V1081' = '1' /* BONE */
    'V1082' = '1' /* MALIGNANT MELANOMA OF SKIN */
    'V1083' = '1' /* OPTHER MALIGNANT NEOPLASM OF SKIN */
    'V1084' = '1' /* EYE */
    'V1085' = '1' /* BRAIN */
    'V1086' = '1' /* OTHER PARTS OF NERVOUS SYSTEM */
    'V1087' = '1' /* THYROID */
    'V1088' = '1' /* OTHER ENDOCRINE GLANDS AND RELATED STRUCTURES */
    'V1089' = '1' /* OTHER */
    'V109'  = '1' /* UNSPECIFIED PERSONAL HISTORY OF MALIGNANT NEOPLASM */
    OTHER   = '0' ;

    /* Cancer */
    /* Diagnostic Related Groups (DRGS) */

    VALUE $CANCEDR
    PSFMTS -- CONTINUED

    '010' = '1' /* NERVOUS SYSTEM NEOPLASMS WITH CC */
    '011' = '1' /* NERVOUS SYSTEM NEOPLASMS WITHOUT CC */
    '064' = '1' /* EAR, NOSE, MOUTH AND THROAT MALIGNANCY */
    '082' = '1' /* RESPIRATORY NEOPLASMS */
    '172' = '1' /* DIGESTIVE MALIGNANCY WITH CC */
    '173' = '1' /* DIGESTIVE MALIGNANCY WITHOUT CC */
    '199' = '1' /* HEPATOBILIARY DIAGNOSTIC PROCEDURE FOR MALIGNANCY */
    '203' = '1' /* MALIGNANCY OF HEPATOBILIARY SYSTEM OR PANCREAS */
    '239' = '1' /* PATHOLOGICAL FRACTURES AND MUSCULOSKELETAL AND CONNECTIVE
                   TISSUE MALIGNANCY */
    '257' = '1' /* TOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '258' = '1' /* TOTAL MASTECTOMY FOR MALIGNANCY WITHOUT CC */
    '259' = '1' /* SUBTOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '260' = '1' /* SUBTOTAL MASTECTOMY FOR MALIGNANCY WITHOUT CC */
    '274' = '1' /* MALIGNANT BREAST DISORDERS WITH CC */
    '275' = '1' /* MALIGNANT BREAST DISORDERS WITHOUT CC */
    '303' = '1' /* KIDNEY, URETER AND MAJOR BLADDER PROCEDURES FOR NEOPLASM */
    '318' = '1' /* KIDNEY AND URINARY TRACT NEOPLSMS WITH CC */
    '319' = '1' /* KIDNEY AND URINARY TRACT NEOPLASMS WITHOUT CC */
    '338' = '1' /* TESTES PROCEDURES FOR MALIGNANCY */
    '344' = '1' /* OTHER MALE REPRODUCTIVE SYSTEM OR PROCEDURES FOR MALIGNANCY
*/
    '346' = '1' /* MALIGNANCY OF MALE REPRODUCTIVE SYSTEM WITH CC */
    '347' = '1' /* MALIGNANCY OF MALE REPRODUCTIVE SYSTEM WITHOUT CC */
    '354' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONOVARIAN/ADNEXAL
                   MALIGNANCY WITH CC */
    '355' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONOVARIAN/ADNEXAL
                   MALIGNANCY WITHOUT CC */
    '357' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR OVARIAN OR ADNEXAL
                   MALIGNANCY */
    '363' = '1' /* D AND C, CONIZATION AND RADIOIMPLANT FOR MALIGNANCY */
    '367' = '1' /* MALIGNANCY OF FEMALE REPRODUCTIVE SYSTEM WITHOUT CC */
    '400' = '1' /* LYMPHOMA AND LEUKEMIA WITH MAJOR OR PROCEDURES */
    '401' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH OTHER OR PROCEDURE
                   WITH CC */
    '402' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH OTHER OR PROCEDURE
                   WITHOUT CC */
    '403' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH CC */
    '404' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITHOUT CC */
```

```
    '405' = '1' /* ACUTE LEUKEMAI WITHOUT MAJOR OR PROCEDURE, AGE 0-17 */
    '406' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH MAJOR OR PROCEDURES WITH CC */
    '407' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH MAJOR OR PROCEDURE WITHOUT CC */
    '408' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH OTHER OR PROCEDURES */
    '409' = '1' /* RADIOTHERAPY */
    '410' = '1' /* CHEMOTHERAPY WITHOUT ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS
*/
    '411' = '1' /* HISTORY OF MALIGNANCY WITHOUT ENDOSCOPY */
    '412' = '1' /* HISTORY OF MALIGNANCY WITH ENDOSCOPY */
    '413' = '1' /* OTHER MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASM DIAGNOSES WITH CC */
    '414' = '1' /* OTHER MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASM DIAGNOSES WITHOUT CC */
    '473' = '1' /* ACUTE LEUKEMIA WITHOUT MAJOR OR PROCEDURE, AGE GREATER
                   THAN 17 */
    '492' = '1' /* CHEMOTHERAPY WITH ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS */
    OTHER = '0' ;
    PSFMTS -- CONTINUED


    /* Cardiac arrest */
    /* ICD-9-CM diagnosis codes: */

    VALUE $CARDIID
    '4275'  = '1' /* CARDIAC ARREST */
    OTHER   = '0' ;

    /* Cardiac arrhythmia */
    /* ICD-9-CM diagnosis codes: */

    VALUE $CARDRID
    '4260'  = '1' /* ATRIOVENTRICULAR BLOCK, COMPLETE */
    '4270'  = '1' /* PAROXYSMAL SUPRAVENTRICULAR TACHYCARDIA */
    '4271'  = '1' /* PAROSYSMAL VENTRICULAR TACHYCARDIA */
    '4272'  = '1' /* PAROXYSMAL TACHYCARDIA, UNSPECIFIED */
    '42731' = '1' /* ATRIAL FIBRILLATION */
    '42732' = '1' /* ATRIAL FLUTTER */
    '42741' = '1' /* VENTRICULAR FIBRILLATION */
    '42742' = '1' /* VENTRICULAR FLUTTER */
    '4279'  = '1' /* CARDIAC DYSRHYTHMIA */
    OTHER   = '0' ;

    /* Cardiac arrhythmia */
    /* Diagnostic Related Groups (DRGs): */

    VALUE $CARDRDR
    '138' = '1' /* CARDIAC ARRHYTHMIA AND CONDUCTION DISORDERS WITH CC */
    '139' = '1' /* CARDIAC ARRHYTHMIA AND CONDUCTION DISORDERS WITHOUT CC */
    OTHER = '0' ;

    /* Cardiac surgery */
    /* Diagnostic Related Groups (DRGs): */

    VALUE $CARDSDR
    '103' = '1' /* HEART TRANSPLANT */
```

```
    '104' = '1' /* CARDIAC VALVE AND OTHER MAJOR CARDIOTHORACIC PROCEDURES
WITH
                  CARDIAC CATHETERIZATION */
    '105' = '1' /* CARDIAC VALVE AND OTHER MAJOR CARDIOTHORACIC PROCEDURES
                  WITHOUT CARDIAC CATHETERIZATION */
    '106' = '1' /* CORONARY BYPASS WITH PTCA */
    '107' = '1' /* CORONARY BYPASS WITH CARDIAC CATHETERIZATION */
    '108' = '1' /* OTHER CARDIOTHORACIC PROCEDURES */
    '109' = '1' /* CORONARY BYPASS WITHOUT CARDIAC CATHETERIZATION */
    '110' = '1' /* MAJOR CARDIOVASCULAR PROCEDURES WITH CC */
    '111' = '1' /* MAJOR CARDIOVASCULAR PROCEDURES WITHOUT CC */
    OTHER = '0' ;

    /* Cesarean delivery */
    /* Diagnostic related groups (DRGs): */

    VALUE $CESARDR
    '370' = '1' /* CESAREAN SECTION WITH CC */
    '371' = '1' /* CESAREAN SECTION WITHOUT CC */
    OTHER = '0' ;

    /* Coma */
    /* ICD-9-CM diagnosis codes: */
    PSFMTS -- CONTINUED


    VALUE $COMAID
    '2510'    =  '1'  /*  OTHER  DISORDERS  OF  PANCREATIC  INTERNAL  SECRETION,
HYPOGLYCEMIC
                  COMA */
    '5722'   =  '1'  /* LIVER  ABSCESS  AND  SEQUELAE  OF  CHRONIC  LIVER  DISEASE,
HEPATIC
                  COMA */
    '78001' = '1' /* GENERAL SYMPTOMS, ALTERATION OF CONSCIOUSNESS, COMA */
    '25020'  =  '1'  /*  DIABETES  WITH  HYPEROSMOLARITY,  TYPE  2  [NONINSULIN
DEPENDENT
                   TYPE][NIDDM TYPE][ADULT-ONSET] OR UNSPECIFIED TYPE, NOT
STATED
                  AS UNCONTROLLED */
    '25021' = '1' /* DIABETES WITH HYPEROSMOLARITY, TYPE 1 [INSULIN DEPENDENT
TYPE]
                  [NIDDM-TYPE] [JUVENILE TYPE], NOT STATED AS UNCONTROLLED
*/
    '25022' = '1' /* DIABETES WITH HYPEROSMOLARITY, TYPE 2 */
    '25023' = '1' /* DIABETES MELLITUS, DIABETES WITH HYPEROSMOLARITY , TYPE 1
                  [INSULIN DEPENDENT TYPE][NIDMM-TYPE][JUVENILE TYPE]
                  UNCONTROLLED */
    '25030'  =  '1'  /*  DIABETES  WITH  OTHER  COMA,  TYPE  2  NOT  STATED  AS
UNCONTROLLED */
    '25031'  =  '1'  /*  DIABETES  WITH  OTHER  COMA,  TYPE  1  NOT  STATED  AS
UNCONTROLLED */
    '25032' = '1' /* DIABETES MELLITUS, DIABETES WITH OTHER COMA, TYPE 11
                  UNCONTROLLED */
    '25033' = '1' /* DIABETES MELLITUS, DIABETES WITH OTHER COMA, TYPE 1
                  UNCONTROLLED */
    '78003' = '1' /* GENERAL SYMPTOMS, ALTERATION OF CONSCIOUSNESS PERSISTENT
                  VEGETATIVE STATE */
    OTHER   = '0' ;
```

```
    /* Control of post-operative hematoma */
    /* ICD-9-CM procedure codes: */

    VALUE $HEMATIP
    '1809' = '1' /* OTHER INCISION OF EXTERNAL EAR */
    '540'  = '1' /* INCISION OF ABDOMINAL WALL */
    '5412' = '1' /* REOPENING OF RECENT LAPAROTOMY SITE */
    '5919' = '1' /* OTHER INCISION OF PERIVESICLE TISSUE */
    '610'  = '1' /* INCISION AND DRAINAGE OF SCROTUM AND TUNICA AND VAGINALIS
*/
    '6998' = '1' /* OTHER OPERATIONS ON SUPPORTING STRUCTURES OF UTERUS */
    '7014' = '1' /* OTHER VAGINOTOMY */
    '7109' = '1' /* OTHER INCISION OF VULVA AND PERINEUM */
    '7591' = '1' /* EVACUATION OF OBSTETRICAL INCISIONAL HEMATOMA OF PERINEUM
*/
    '7592' = '1' /* EVACUATION OF OTHER HEMATOMA OF VULVA OR VAGINA */
    '8604' = '1' /* OTHER INCISION WITH DRAINAGE OF SKIN AND SUBCUTANEOUS
TISSUE */
    OTHER  = '0' ;

    /* Control of post-operative hemorrhage */
    /* ICD-9-CM procedure codes: */

    VALUE $HEMORIP
```



```
    '287'    =  '1'  /*  CONTROL  OF  HEMORRHAGE  AFTER  TONSILLECTOMY  AND
ADENOIDECTOMY */
    '3880' = '1' /* OTHER SURGICAL OCCLUSION OF UNSPECIFIED SITE */
    '3881' = '1' /* OTHER SURGICAL OCCLUSION OF INTRACRANIAL VESSELS */
    '3882' = '1' /* OTHER SURGICAL OCCLUSION OF OTHER VESSELS OF HEAD AND NECK
*/
    '3883' = '1' /* OTHER SURGICAL OCCLUSION OF UPPER LIMB VESSELS */
    '3884' = '1' /* OTHER SURGICAL OCCLUSION OF AORTA, ABDOMINAL */
    '3885' = '1' /* OTHER SURGICAL OCCLUSION OF THORACIC VESSEL */
    '3886' = '1' /* OTHER SURGICAL OCCLUSION OF ABDOMINAL ARTERIES */
    '3887' = '1' /* OTHER SURGICAL OCCLUSION OF VESSELS ABDOMINAL VEINS */
    '3888' = '1' /* OTHER SURGICAL OCCLUSION OF LOWER LIMB VEINS */
    '3889' = '1' /* OTHER SURGICAL OCCLUSION OF LOWER LIMB VEINS */
    '3941'  =  '1'  /*  CONTROL  OF  HEMORRHAGE  AFTER  TONSILLECTOMY  AND
ADENOIDECTOMY */
    '3998' = '1' /* CONTROL OF HEMORRHAGE NOS */
    '4995' = '1' /* CONTROL OF (POSTOPERATIVE) HEMORRHAGE OF ANUS */
    '5793' = '1' /* CONTROL OF (POSTOPERATIVE HEMORRHAGE OF BLADDER) */
    '6094' = '1' /* CONTROL OF (POSTOPERATIVE) HEMORRHAGE OF PROSTATE */
    OTHER  = '0' ;

    /* Decubitus Ulcer */
    /* ICD-9-CM diagnosis codes: */

    VALUE $DECUBID
    '7070' = '1' /* DECUBITUS ULCER */
    OTHER  = '0' ;

    /* Deep vein thrombosis */
    /* ICD-9-CM diagnosis codes: */
```

```
    VALUE $DEEPVID
    '45111'  =  '1'  /*  PHLEBITIS  AND  THROMBOSIS  OF  FEMORAL  VEIN  (DEEP)
(SUPERFICIAL) *
    /
    '45119' = '1' /* PHLEBITIS AND THROMBOPHLEBITIS - OF DEEP VESSEL OF LOWER
                     EXTREMITIES - OTHER */
    '4512'  = '1' /* PHLEBITIS AND THROMBOPHLEBITIS OF LOWER EXTREMITIES
                     UNSPECIFIED */
    '45181' = '1' /* PHLEBITIS AND THROMBOPHLEBITIS OF ILIAC VEIN */
    '4519'   =  '1'  /*  PHLEBITIS  AND  THROMBOPHLEBITIS  OF  OTHER  SITES  -  OF
UNSPECIFIED
                     SITE */
    '4538'   = '1' /* OTHER VENOUS EMBOLISM AND THROMBOSIS OF OTHER SPECIFIED
VEINS *
    /
    '4539'  = '1' /* OTHER VENOUS EMBOLISM AND THROMBOSIS OF UNSPECIFIED SITE
*/
    OTHER   = '0' ;

    /* Delirium and other psychoses */
    /* ICD-9-CM diagnosis codes (includes all 4th and 5th digits) */

    VALUE $DELIRID
    '290'   = '1' /* SENILE/PRESENILE PSYCHOS */
    '2900'  = '1' /* SENILE DEMENTIA UNCOMP */
    '2901'  = '1' /* PRESENILE DEMENTIA */
    '29010' = '1' /* PRESENILE DEMENTIA */
    '29011' = '1' /* PRESENILE DELIRIUM */
    PSFMTS -- CONTINUED

    '29012' = '1' /* PRESENILE DELUSION */
    '29013' = '1' /* PRESENILE DEPRESSION */
    '2902'  = '1' /* SENILE DELUSION/DEPRESS */
    '29020' = '1' /* SENILE DELUSION */
    '29021' = '1' /* SENILE DEPRESSIVE */
    '2903'  = '1' /* SENILE DELIRIUM */
    '2904'  = '1' /* ARTERIOSCLEROT DEMENTIA */
    '29040' = '1' /* ARTERIOSCLER DEMENT NOS */
    '29041' = '1' /* ARTERIOSCLER DELIRIUM */
    '29042' = '1' /* ARTERIOSCLER DELUSION */
    '29043' = '1' /* ARTERIOSCLER DEPRESSIVE */
    '2908'  = '1' /* SENILE PSYCHOSIS NEC */
    '2909'  = '1' /* SENILE PSYCHOT COND NOS */
    '291'   = '1' /* ALCOHOLIC PSYCHOSES */
    '2910'  = '1' /* DELIRIUM TREMENS */
    '2911'  = '1' /* ALCOHOL AMNESTIC SYND */
    '2912'  = '1' /* ALCOHOLIC DEMENTIA NEC */
    '2913'  = '1' /* ALCOHOL HALLUCINOSIS */
    '2914'  = '1' /* PATHOLOGIC ALCOHOL INTOX */
    '2915'  = '1' /* ALCOHOLIC JEALOUSY */
    '2918'  = '1' /* ALCOHOLIC PSYCHOSIS NEC */
    '29181' = '1' /* ALCOHOL WITHDRAWAL */
    '29189' = '1' /* ALCOHOLIC PSYCHOSIS NEC */
    '2919'  = '1' /* ALCOHOLIC PSYCHOSIS NOS */
    '292'   = '1' /* DRUG PSYCHOSES */
    '2920'  = '1' /* DRUG WITHDRAWAL SYNDROME */
    '2921'  = '1' /* DRUG PARANOID/HALLUCINOS */
```

```
'29211' = '1' /* DRUG PARANOID STATE */
'29212' = '1' /* DRUG HALLUCINOSIS */
'2922'  = '1' /* PATHOLOGIC DRUG INTOX */
'2928'  = '1' /* OTHER DRUG MENTAL DIS */
'29281' = '1' /* DRUG-INDUCED DELIRIUM */
'29282' = '1' /* DRUG-INDUCED DEMENTIA */
'29283' = '1' /* DRUG AMNESTIC SYNDROME */
'29284' = '1' /* DRUG DEPRESSIVE SYNDROME */
'29289' = '1' /* DRUG MENTAL DISORDER NEC */
'2929'  = '1' /* DRUG MENTAL DISORDER NOS */
'293'   = '1' /* TRANSIENT ORG MENTAL DIS */
'2930'  = '1' /* ACUTE DELIRIUM */
'2931'  = '1' /* SUBACUTE DELIRIUM */
'2938'  = '1' /* OTH TRANSIENT ORG MENTAL */
'29381' = '1' /* ORGANIC DELUSIONAL SYND */
'29382' = '1' /* ORGANIC HALLUCINOSIS SYN */
'29383' = '1' /* ORGANIC AFFECTIVE SYND */
'29384' = '1' /* ORGANIC ANXIETY SYNDROME */
'29389' = '1' /* TRANSIENT ORG MENTAL NEC */
'2939'  = '1' /* TRANSIENT ORG MENTAL NOS */
'294'   = '1' /* OTHER ORGANIC PSYCH COND */
'2940'  = '1' /* AMNESTIC SYNDROME */
'2941'  = '1' /* DEMENTIA IN OTH DISEASES */
'29410' = '1' /* DEMENTIA W/O BEHAV DIST */
'29411' = '1' /* DEMENTIA W BEHAVIOR DIST */
'2948'  = '1' /* ORGANIC BRAIN SYND NEC */
'2949'  = '1' /* ORGANIC BRAIN SYND NOS */
'295'   = '1' /* SCHIZOPHRENIC DISORDERS */
'2950'  = '1' /* SIMPLE SCHIZOPHRENIA */
'29500' = '1' /* SIMPL SCHIZOPHREN-UNSPEC */
'29501' = '1' /* SIMPL SCHIZOPHREN-SUBCHR */
'29502' = '1' /* SIMPLE SCHIZOPHREN-CHR */
```

PSFMTS -- CONTINUED

```
'29503' = '1' /* SIMP SCHIZ-SUBCHR/EXACER */
'29504' = '1' /* SIMPL SCHIZO-CHR/EXACERB */
'29505' = '1' /* SIMPL SCHIZOPHREN-REMISS */
'2951'  = '1' /* HEBEPHRENIA */
'29510' = '1' /* HEBEPHRENIA-UNSPEC */
'29511' = '1' /* HEBEPHRENIA-SUBCHRONIC */
'29512' = '1' /* HEBEPHRENIA-CHRONIC */
'29513' = '1' /* HEBEPHREN-SUBCHR/EXACERB */
'29514' = '1' /* HEBEPHRENIA-CHR/EXACERB */
'29515' = '1' /* HEBEPHRENIA-REMISSION */
'2952'  = '1' /* CATATONIC SCHIZOPHRENIA */
'29520' = '1' /* CATATONIA-UNSPEC */
'29521' = '1' /* CATATONIA-SUBCHRONIC */
'29522' = '1' /* CATATONIA-CHRONIC */
'29523' = '1' /* CATATONIA-SUBCHR/EXACERB */
'29524' = '1' /* CATATONIA-CHR/EXACERB */
'29525' = '1' /* CATATONIA-REMISSION */
'2953'  = '1' /* PARANOID SCHIZOPHRENIA */
'29530' = '1' /* PARANOID SCHIZO-UNSPEC */
'29531' = '1' /* PARANOID SCHIZO-SUBCHR */
'29532' = '1' /* PARANOID SCHIZO-CHRONIC */
'29533' = '1' /* PARAN SCHIZO-SUBCHR/EXAC */
'29534' = '1' /* PARAN SCHIZO-CHR/EXACERB */
'29535' = '1' /* PARANOID SCHIZO-REMISS */
```

```
'2954'  = '1' /* AC SCHIZOPHRENIC EPISODE */
'29540' = '1' /* AC SCHIZOPHRENIA-UNSPEC */
'29541' = '1' /* AC SCHIZOPHRENIA-SUBCHR */
'29542' = '1' /* AC SCHIZOPHRENIA-CHR */
'29543' = '1' /* AC SCHIZO-SUBCHR/EXACERB */
'29544' = '1' /* AC SCHIZOPHR-CHR/EXACERB */
'29545' = '1' /* AC SCHIZOPHRENIA-REMISS */
'2955'  = '1' /* LATENT SCHIZOPHRENIA */
'29550' = '1' /* LATENT SCHIZOPHREN-UNSP */
'29551' = '1' /* LAT SCHIZOPHREN-SUBCHR */
'29552' = '1' /* LATENT SCHIZOPHREN-CHR */
'29553' = '1' /* LAT SCHIZO-SUBCHR/EXACER */
'29554' = '1' /* LATENT SCHIZO-CHR/EXACER */
'29555' = '1' /* LAT SCHIZOPHREN-REMISS */
'2956'  = '1' /* RESIDUAL SCHIZOPHRENIA */
'29560' = '1' /* RESID SCHIZOPHREN-UNSP */
'29561' = '1' /* RESID SCHIZOPHREN-SUBCHR */
'29562' = '1' /* RESIDUAL SCHIZOPHREN-CHR */
'29563' = '1' /* RESID SCHIZO-SUBCHR/EXAC */
'29564' = '1' /* RESID SCHIZO-CHR/EXACERB */
'29565' = '1' /* RESID SCHIZOPHREN-REMISS */
'2957'  = '1' /* SCHIZOAFFECTIVE TYPE */
'29570' = '1' /* SCHIZOAFFECTIVE-UNSPEC */
'29571' = '1' /* SCHIZOAFFECTIVE-SUBCHR */
'29572' = '1' /* SCHIZOAFFECTIVE-CHRONIC */
'29573' = '1' /* SCHIZOAFF-SUBCHR/EXACER */
'29574' = '1' /* SCHIZOAFFECT-CHR/EXACER */
'29575' = '1' /* SCHIZOAFFECTIVE-REMISS */
'2958'  = '1' /* SCHIZOPHRENIA NEC */
'29580' = '1' /* SCHIZOPHRENIA NEC-UNSPEC */
'29581' = '1' /* SCHIZOPHRENIA NEC-SUBCHR */
'29582' = '1' /* SCHIZOPHRENIA NEC-CHR */
'29583' = '1' /* SCHIZO NEC-SUBCHR/EXACER */
'29584' = '1' /* SCHIZO NEC-CHR/EXACERB */
'29585' = '1' /* SCHIZOPHRENIA NEC-REMISS */
```

PSFMTS -- CONTINUED

```
'2959'  = '1' /* SCHIZOPHRENIA NOS */
'29590' = '1' /* SCHIZOPHRENIA NOS-UNSPEC */
'29591' = '1' /* SCHIZOPHRENIA NOS-SUBCHR */
'29592' = '1' /* SCHIZOPHRENIA NOS-CHR */
'29593' = '1' /* SCHIZO NOS-SUBCHR/EXACER */
'29594' = '1' /* SCHIZO NOS-CHR/EXACERB */
'29595' = '1' /* SCHIZOPHRENIA NOS-REMISS */
'296'   = '1' /* AFFECTIVE PSYCHOSES */
'2960'  = '1' /* MANIC DIS, SINGL EPISODE */
'29600' = '1' /* MANIC DISORDER-UNSPEC */
'29601' = '1' /* MANIC DISORDER-MILD */
'29602' = '1' /* MANIC DISORDER-MOD */
'29603' = '1' /* MANIC DISORDER-SEVERE */
'29604' = '1' /* MANIC DIS-SEVERE W PSYCH */
'29605' = '1' /* MANIC DIS-PARTIAL REMISS */
'29606' = '1' /* MANIC DIS-FULL REMISSION */
'2961'  = '1' /* MANIC, RECURRENT EPISODE */
'29610' = '1' /* RECUR MANIC DIS-UNSPEC */
'29611' = '1' /* RECUR MANIC DIS-MILD */
'29612' = '1' /* RECUR MANIC DIS-MOD */
'29613' = '1' /* RECUR MANIC DIS-SEVERE */
```

```
'29614' = '1' /* RECUR MANIC-SEV W PSYCHO */
'29615' = '1' /* RECUR MANIC-PART REMISS */
'29616' = '1' /* RECUR MANIC-FULL REMISS */
'2962'  = '1' /* DEPR PSYCH, SINGL EPISOD */
'29620' = '1' /* DEPRESS PSYCHOSIS-UNSPEC */
'29621' = '1' /* DEPRESS PSYCHOSIS-MILD */
'29622' = '1' /* DEPRESSIVE PSYCHOSIS-MOD */
'29623' = '1' /* DEPRESS PSYCHOSIS-SEVERE */
'29624' = '1' /* DEPR PSYCHOS-SEV W PSYCH */
'29625' = '1' /* DEPR PSYCHOS-PART REMISS */
'29626' = '1' /* DEPR PSYCHOS-FULL REMISS */
'2963'  = '1' /* DEPR PSYCH, RECUR EPISOD */
'29630' = '1' /* RECURR DEPR PSYCHOS-UNSP */
'29631' = '1' /* RECURR DEPR PSYCHOS-MILD */
'29632' = '1' /* RECURR DEPR PSYCHOS-MOD */
'29633' = '1' /* RECUR DEPR PSYCH-SEVERE */
'29634' = '1' /* REC DEPR PSYCH-PSYCHOTIC */
'29635' = '1' /* RECUR DEPR PSYC-PART REM */
'29636' = '1' /* RECUR DEPR PSYC-FULL REM */
'2964'  = '1' /* BIPOLAR AFFECTIVE, MANIC */
'29640' = '1' /* BIPOL AFF, MANIC-UNSPEC */
'29641' = '1' /* BIPOLAR AFF, MANIC-MILD */
'29642' = '1' /* BIPOLAR AFFEC, MANIC-MOD */
'29643' = '1' /* BIPOL AFF, MANIC-SEVERE */
'29644' = '1' /* BIPOL MANIC-SEV W PSYCH */
'29645' = '1' /* BIPOL AFF MANIC-PART REM */
'29646' = '1' /* BIPOL AFF MANIC-FULL REM */
'2965'  = '1' /* BIPOLAR AFFECT, DEPRESS */
'29650' = '1' /* BIPOLAR AFF, DEPR-UNSPEC */
'29651' = '1' /* BIPOLAR AFFEC, DEPR-MILD */
'29652' = '1' /* BIPOLAR AFFEC, DEPR-MOD */
'29653' = '1' /* BIPOL AFF, DEPR-SEVERE */
'29654' = '1' /* BIPOL DEPR-SEV W PSYCH */
'29655' = '1' /* BIPOL AFF DEPR-PART REM */
'29656' = '1' /* BIPOL AFF DEPR-FULL REM */
'2966'  = '1' /* BIPOLAR AFFECTIVE, MIXED */
'29660' = '1' /* BIPOL AFF, MIXED-UNSPEC */
'29661' = '1' /* BIPOLAR AFF, MIXED-MILD */
```

PSFMTS -- CONTINUED

```
'29662' = '1' /* BIPOLAR AFFEC, MIXED-MOD */
'29663' = '1' /* BIPOL AFF, MIXED-SEVERE */
'29664' = '1' /* BIPOL MIXED-SEV W PSYCH */
'29665' = '1' /* BIPOL AFF, MIX-PART REM */
'29666' = '1' /* BIPOL AFF, MIX-FULL REM */
'2967'  = '1' /* BIPOLAR AFFECTIVE NOS */
'2968'  = '1' /* MANIC-DEPRESSIVE NEC/NOS */
'29680' = '1' /* MANIC-DEPRESSIVE NOS */
'29681' = '1' /* ATYPICAL MANIC DISORDER */
'29682' = '1' /* ATYPICAL DEPRESSIVE DIS */
'29689' = '1' /* MANIC-DEPRESSIVE NEC */
'2969'  = '1' /* AFFECT PSYCHOSES NEC/NOS */
'29690' = '1' /* AFFECTIVE PSYCHOSIS NOS */
'29699' = '1' /* AFFECTIVE PSYCHOSES NEC */
'297'   = '1' /* PARANOID STATES */
'2970'  = '1' /* PARANOID STATE, SIMPLE */
'2971'  = '1' /* PARANOIA */
'2972'  = '1' /* PARAPHRENIA */
```

```
'2973'  = '1' /* SHARED PARANOID DISORDER */
'2978'  = '1' /* PARANOID STATES NEC */
'2979'  = '1' /* PARANOID STATE NOS */
'298'   = '1' /* OTH NONORGANIC PSYCHOSES */
'2980'  = '1' /* REACT DEPRESS PSYCHOSIS */
'2981'  = '1' /* EXCITATIV TYPE PSYCHOSIS */
'2982'  = '1' /* REACTIVE CONFUSION */
'2983'  = '1' /* ACUTE PARANOID REACTION */
'2984'  = '1' /* PSYCHOGEN PARANOID PSYCH */
'2988'  = '1' /* REACT PSYCHOSIS NEC/NOS */
'2989'  = '1' /* PSYCHOSIS NOS */
'299'   = '1' /* PSYCHOSES OF CHILDHOOD */
'2990'  = '1' /* INFANTILE AUTISM */
'29900' = '1' /* INFANTILE AUTISM-ACTIVE */
'29901' = '1' /* INFANTILE AUTISM-RESID */
'2991'  = '1' /* DISINTEGRATIVE PSYCHOSIS */
'29910' = '1' /* DISINTEGR PSYCH-ACTIVE */
'29911' = '1' /* DISINTEGR PSYCH-RESIDUAL */
'2998'  = '1' /* EARLY CHLD PSYCHOSES NEC */
'29980' = '1' /* CHILD PSYCHOS NEC-ACTIVE */
'29981' = '1' /* CHILD PSYCHOS NEC-RESID */
'2999'  = '1' /* EARLY CHLD PSYCHOSIS NOS */
'29990' = '1' /* CHILD PSYCHOS NOS-ACTIVE */
'29991' = '1' /* CHILD PSYCHOS NOS-RESID */
OTHER   = '0' ;

/* Diabetes */
/* ICD-9-CM diagnosis codes: */

VALUE $DIABEID
'250'   = '1' /* DIABETES MELLITUS */
'2500'  = '1' /* DIABETES MELLITUS WITHOUT MENTION OF COMPLICATION */
'25000' = '1' /* DMII WO CMP NT ST UNCNTR */
'25001' = '1' /* DMI WO CMP NT ST UNCNTRL */
'25002' = '1' /* DMII WO CMP UNCNTRLD */
'25003' = '1' /* DMI WO CMP UNCNTRLD */
'2501'  = '1' /* DIABETES WITH KETOACIDOSIS */
'25010' = '1' /* DMII KETO NT ST UNCNTRLD */
'25011' = '1' /* DMI KETO NT ST UNCNTRLD */
'25012' = '1' /* DMII KETOACD UNCONTROLD */
'25013' = '1' /* DMI KETOACD UNCONTROLD */
```

PSFMTS – CONTINUED

```
'2502'  = '1' /* DIABETES WITH HYPEROSMOLARITY */
'25020' = '1' /* DMII HPRSM NT ST UNCNTRL */
'25021' = '1' /* DMI HPRSM NT ST UNCNTRLD */
'25022' = '1' /* DMII HPROSMLR UNCONTROLD */
'25023' = '1' /* DMI HPROSMLR UNCONTROLD */
'2503'  = '1' /* DIABETES WITH OTHER COMA */
'25030' = '1' /* DMII O CM NT ST UNCNTRLD */
'25031' = '1' /* DMI O CM NT ST UNCNTRLD */
'25032' = '1' /* DMII OTH COMA UNCONTROLD */
'25033' = '1' /* DMI OTH COMA UNCONTROLD */
'2504'  = '1' /* DIABETES WITH RENAL MANIFESTATIONS */
'25040' = '1' /* DMII RENL NT ST UNCNTRLD */
'25041' = '1' /* DMI RENL NT ST UNCNTRLD */
'25042' = '1' /* DMII RENAL UNCNTRLD */
'25043' = '1' /* DMI RENAL UNCNTRLD */
```

```
    '2505'  = '1' /* DIABETES WITH OPHTHALMIC MANIFESTATIONS */
    '25050' = '1' /* DMII OPHTH NT ST UNCNTRL */
    '25051' = '1' /* DMI OPHTH NT ST UNCNTRLD */
    '25052' = '1' /* DMII OPHTH UNCNTRLD */
    '25053' = '1' /* DMI OPHTH UNCNTRLD */
    '2506'  = '1' /* DIABETES WITH NEUROLOGICAL MANIFESTATIONS */
    '25060' = '1' /* DMII NEURO NT ST UNCNTRL */
    '25061' = '1' /* DMI NEURO NT ST UNCNTRLD */
    '25062' = '1' /* DMII NEURO UNCNTRLD */
    '25063' = '1' /* DMI NEURO UNCNTRLD */
    '2507'  = '1' /* DIABETES WITH PERIPHERAL CIRCULATORY DISORDERS */
    '25070' = '1' /* DMII CIRC NT ST UNCNTRLD */
    '25071' = '1' /* DMI CIRC NT ST UNCNTRLD */
    '25072' = '1' /* DMII CIRC UNCNTRLD */
    '25073' = '1' /* DMI CIRC UNCNTRLD */
    '2508'  = '1' /* DIABETES WITH OTHER SPECIFIED MANIFESTATIONS */
    '25080' = '1' /* DMII OTH NT ST UNCNTRLD */
    '25081' = '1' /* DMI OTH NT ST UNCNTRLD */
    '25082' = '1' /* DMII OTH UNCNTRLD */
    '25083' = '1' /* DMI OTH UNCNTRLD */
    '2509'  = '1' /* DIABETES WITH OTHER UNSPECIFIED COMPLICATION */
    '25090' = '1' /* DMII UNSPF NT ST UNCNTRL */
    '25091' = '1' /* DMI UNSPF NT ST UNCNTRLD */
    '25092' = '1' /* DMII UNSPF UNCNTRLD */
    '25093' = '1' /* DMI UNSPF UNCNTRLD */
    OTHER   = '0' ;

    /* Dialysis */
    /* ICD-9-CM procedure codes: */

    VALUE $DIALYIP
    '3995'  = '1'
    '5498'  = '1'
    OTHER   = '0' ;

    /* Foreign body left in during procedure */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FOREIID
    '9984'  = '1' /* FOREIGN BODY ACCIDENTALLY LEFT DURING A PROCEDURE */
    '9987'   = '1' /* ACUTE REACTION TO FOREIGN SUBSTANCE ACCIDENTALLY LEFT
DURING
                    A PROCEDURE */
    'E8710' = '1' /* SURGICAL OPERATION */
```

PSFMTS – CONTINUED

```
    'E8711' = '1' /* INFUSION OR TRANSFUSION */
    'E8712' = '1' /* KIDNEY DIALYSIS OR OTHER PERFUSION */
    'E8713' = '1' /* INJECTION OR VACCINATION */
    'E8714' = '1' /* ENDOSCOPIC EXAMINATION */
    'E8715'  =  '1'  /*  ASPIRATION  OF  FLUID  OR  TISSUE,  PUNCTURE,  AND
CATHETERIZATION */
    'E8716' = '1' /* HEART CATHETERIZATION */
    'E8717' = '1' /* REMOVAL OF CATHETER OR PACKING */
    'E8718' = '1' /* OTHER SPECIFIED PROCEDURES */
    'E8719' = '1' /* UNSPECIFIED PROCEDURE */
    OTHER   = '0' ;
```

```
    /* Gastrointestinal (GI) hemorrhage */
    /* ICD-9-CM diagnosis codes: */

    VALUE $GASTRID
    '4560'  = '1' /* ESOPHAGEAL VARICES WITH BLEEDING */
    '45620' = '1' /* ESOPHAGEAL VARICES IN DISEASES CLASSIFIED ELSEWHERE WITH
                     BLEEDING */
    '5307'  = '1' /* GASTROESOPHAGEAL LACERATION- HEMORRHAGE SYNDROME */
    '53082' = '1' /* ESOPHAGEAL HEMORRHAGE */
    '53100' = '1' /* GASTRIC ULCER ACUTE WITH HEMORRHAGE - WITHOUT MENTION OF
                     OBSTRUCTION */
    '53101' = '1' /* GASTRIC ULCER ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION */
    '53120' = '1' /* GASTRIC ULCER ACUTE WITH HEMORRHAGE AND PERFORATION -
WITHOUT
                     MENTION OF OBSTRUCTION */
    '53121' = '1' /* GASTRIC ULCER, ACUTE WITH HEMORRHAGE AND PERFORATION -
WITH
                     OBSTRUCTION */
    '53140' = '1' /* GASTRIC ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE -
WITHOUT
                     MENTION OF OBSTRUCTION */
    '53141' = '1' /* GASTRIC ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE -
WITH
                     OBSTRUCTION */
    '53160' = '1' /* GASTRIC ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE AND
                     PERFORATION - WITHOUT MENTION OF OBSTRUCTION */
    '53161' = '1' /* GASTRIC ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE AND
                     PERFORATION - WITH OBSTRUCTION */
    '53200' = '1' /* DUODENAL ULCER ACUTE WITH HEMORRHAGE - WITHOUT MENTION OF
                     OBSTRUCTION */
    '53201' = '1' /* DUODENAL ULCER ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION
*/
    '53220' = '1' /* DUODENAL ULCER ACUTE WITH HEMORRHAGE AND PERFORATION -
                     WITHOUT MENTION OF OBSTRUCTION */
    '53221' = '1' /* DUODENAL ULCER ACUTE WITH HEMORRHAGE AND PERFORATION -
                     WITH OBSTRUCTION */
    '53240' = '1' /* DUODENAL ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE -
                     WITHOUT MENTION OF OBSTRUCTION */
    '53241' = '1' /* DUODENAL ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE -
                     WITH OBSTRUCTION */
    '53260' = '1' /* DUODENAL ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE AND
                     PERFORATION – WITHOUT MENTION OF OBSTRUCTION */
    '53261' = '1' /* DUODENAL ULCER CHRONIC OR UNSPECIFIED WITH HEMORRHAGE AND
                     PERFORATION - WITH OBSTRUCTION */
    '53300' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED ACUTE WITH HEMORRHAGE -
                     WITHOUT MENTION OF OBSTRUCTION */
    '53301' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, ACUTE WITH HEMORRHAGE -
```

PSFMTS – CONTINUED

```
                   WITH OBSTRUCTION */


    '53320' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, ACUTE WITH HEMORRHAGE AND
                     PERFORATION - WITHOUT MENTION OF OBSTRUCTION */
    '53321' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, ACUTE WITH HEMORRHAGE AND
                     PERFORATION - WITH OBSTRUCTION */
    '53340' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED CHRONIC OR UNSPECIFIED
WITH
```

```
                    HEMORRHAGE - WITHOUT MENTION OF OBSTRUCTION */
    '53341' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, CHRONIC OR UNSPECIFIED
WITH
                    HEMORRHAGE - WITH OBSTRUCTION */
    '53360' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, CHRONIC OR UNSPECIFIED
WITH
                    HEMORRHAGE   AND   PERFORATION   -WITHOUT   MENTION   OF
OBSTRUCTION */
    '53361' = '1' /* PEPTIC ULCER, SITE UNSPECIFIED, CHRONIC OR UNSPECIFIED
WITH
                    HEMORRHAGE AND PERFORATION - WITH OBSTRUCTION */
    '53400' = '1' /* GASTROJEJUNAL ULCER, ACUTE WITH HEMORRHAGE - WITHOUT
MENTION
                    OF OBSTRUCTION */
    '53401' = '1' /* GASTROJEJUNAL ULCER, ACUTE WITH HEMORRHAGE - WITH
OBSTRUCTION *
    /
    '53420' = '1' /* GASTROJEJUNAL ULCER, ACUTE WITH HEMORRHAGE AND
PERFORATION -
                    WITHOUT MENTION OF OBSTRUCTION */
    '53421' = '1' /* GASTROJEJUNAL ULCER, ACUTE WITH HEMORRHAGE AND
PERFORATION -
                    WITH OBSTRUCTION */
    '53440' = '1' /* GASTROJEJUNAL ULCER, CHRONIC OR UNSPECIFIED WITH
HEMORRHAGE -
                    WITHOUT MENTION OF OBSTRUCTION */
    '53441' = '1' /* GASTROJEJUNAL ULCER, CHRONIC OR UNSPECIFIED WITH
HEMORRHAGE -
                    WITH OBSTRUCTION */
    '53460' = '1' /* GASTROJEJUNAL ULCER, CHRONIC OR UNSPECIFIED WITH
HEMORRHAGE
                    AND PERFORATION - WITHOUT MENTION OF OBSTRUCTION */
    '53461' = '1' /* GASTROJEJUNAL ULCER, CHRONIC OR UNSPECIFIED WITH
HEMORRHAGE
                    AND PERFORATION - WITH OBSTRUCTION */
    '53501' = '1' /* GASTRITIS AND DUODENITIS, ACUTE GASTRITIS WITH HEMORRHAGE
*/
    '53511' = '1' /* GASTRITIS AND DUODENITIS, ATROPHIC GASTRITIS WITH
HEMORRHAGE */
    '53521' = '1' /* GASTRITIS AND DUODENITIS, GASTRIC MUCOSAL HYPERTROPHY,
WITH
                    HEMORRHAGE */
    '53531' = '1' /* GASTRITIS AND DUODENITIS, ALCOHOLIC GASTRITIS, WITH
                    HEMORRHAGE */
    '53541' = '1' /* GASTRITIS AND DUODENITIS, OTHER SPECIFIED GASTRITIS -
WITH
                    HEMORRHAGE */
    '53551' = '1' /* GASTRITIS AND DUODENITIS, UNSPECIFIED GASTRITIS AND
                    GASTRODUODENITIS - WITH HEMORRHAGE */
    '53561' = '1' /* GASTRITIS AND DUODENITIS, DUODENITIS - WITH HEMORRHAGE */
```

PSFMTS -- CONTINUED

```
    '53783' = '1' /* OTHER SPECIFIED DISORDERS OF STOMACH AND DUODENUM,
                     ANGIODYSPLASIA OF STOMACH AND DUODENUM  - WITH HEMORRHAGE
*/
    '56202' = '1' /* DIVERTICULOSIS OF SMALL INTESTINE -WITH HEMORRHAGE */
    '56203' = '1' /* DIVERTICULITIS OF SMALL INTESTINE -WITH HEMORRHAGE */
    '56212' = '1' /* DIVERTICULOSIS OF COLON - WITH HEMORRHAGE */
```

```
    '56213' = '1' /* DIVERTICULITIS OF COLON - WITH HEMORRHAGE */
    '5693'  = '1' /* HEMORRHAGE OF RECTUM AND ANUS */
    '56985' = '1' /* ANGIODYSPLASIA OF INTESTINE - WITH HEMORRHAGE */
    '5780'  = '1' /* GASTROINTESTINAL HEMORRHAGE, HEMATEMESIS */
    '5781'  = '1' /* GASTROINTESTINAL HEMORRHAGE, BLOOD IN STOOL */
    '5789'     =   '1'   /*   GASTROINTESTINAL   HEMORRHAGE,   HEMORRHAGE   OF
GASTROINTESTINAL
                     TRACT, UNSPECIFIED */
    OTHER   = '0' ;

    /* Hemiplegia, paraplegia, or quadriplegia */
    /* ICD-9-CM diagnosis codes: */

    VALUE $HEMIPID
    '3420'  = '1' /* FLACCID HEMIPLEGIA */
    '34200' = '1' /* FLCCD HMIPLGA UNSPF SIDE */
    '34201' = '1' /* FLCCD HMIPLGA DOMNT SIDE */
    '34202' = '1' /* FLCCD HMIPLG NONDMNT SDE */
    '3421'  = '1' /* SPASTIC HEMIPLEGIA */
    '34210' = '1' /* SPSTC HMIPLGA UNSPF SIDE */
    '34211' = '1' /* SPSTC HMIPLGA DOMNT SIDE */
    '34212' = '1' /* SPSTC HMIPLG NONDMNT SDE */
    '3428'  = '1' /* OTHER SPECIFIED HEMIPLEGIA */
    '34280' = '1' /* OT SP HMIPLGA UNSPF SIDE */
    '34281' = '1' /* OT SP HMIPLGA DOMNT SIDE */
    '34282' = '1' /* OT SP HMIPLG NONDMNT SDE */
    '3429'  = '1' /* HEMIPLEGIA, UNSPECIFIED */
    '34290' = '1' /* UNSP HEMIPLGA UNSPF SIDE */
    '34291' = '1' /* UNSP HEMIPLGA DOMNT SIDE */
    '34292' = '1' /* UNSP HMIPLGA NONDMNT SDE */
    '3430'  = '1' /* INFANTILE CEREBRAL PALSY, DIPLEGIC */
    '3431'  = '1' /* INFANTILE CEREBRAL PALSY, HEMIPLEGIC */
    '3432'  = '1' /* INFANTILE CEREBRAL PALSY, QUADRIPLEGIC */
    '3433'  = '1' /* INFANTILE CEREBRAL PALSY, MONOPLEGIC */
    '3434'  = '1' /* INFANTILE CEREBRAL PALSY INFANTILE HEMIPLEGIA */
    '3438'  = '1' /* INFANTILE CEREBRAL PALSY OTHER SPECIFIED INFANTILE
                     CEREBRAL PALSY */
    '3439'  = '1' /* INFANTILE CEREBRAL PALSY, INFANTILE CEREBRAL PALSY,
                     UNSPECIFIED */
    '3440'  = '1' /* QUADRIPLEGIA AND QUADRIPARESIS */
    '34400' = '1' /* QUADRIPLEGIA, UNSPECIFD */
    '34401' = '1' /* QUADRPLG C1-C4, COMPLETE */
    '34402' = '1' /* QUADRPLG C1-C4, INCOMPLT */
    '34403' = '1' /* QUADRPLG C5-C7, COMPLETE */
    '34404' = '1' /* QUADRPLG C5-C7, INCOMPLT */
    '34409' = '1' /* OTHER QUADRIPLEGIA */
    '3441'  = '1' /* PARAPLEGIA */
    '3442'  = '1' /* DIPLEGIA OF UPPER LIMBS */
    '3443'  = '1' /* MONOPLEGIA OF LOWER LIMB */
    '34430' = '1' /* MONPLGA LWR LMB UNSP SDE */
    '34431' = '1' /* MONPLGA LWR LMB DMNT SDE */
    '34432' = '1' /* MNPLG LWR LMB NONDMNT SD */
    '3444'  = '1' /* MONOPLEGIA OF UPPER LIMB */
    PSFMTS -- CONTINUED

    '34440' = '1' /* MONPLGA UPR LMB UNSP SDE */
    '34441' = '1' /* MONPLGA UPR LMB DMNT SDE */
    '34442' = '1' /* MNPLG UPR LMB NONDMNT SD */
```

```
    '3445'  = '1' /* UNSPECIFIED MONOPLEGIA */
    '34460' = '1' /* CAUDA EQUINA SYNDROME, WITHOUT MENTION OF NEUROGENIC
                     BLADDER */
    '34461' = '1' /* CAUDA EQUINA SYNDROME, WITH NEUROGENIC BLADDER */
    '3448'  = '1' /* OTHER SPECIFIED PARALYTIC SYNDROMES */
    '34481' = '1' /* LOCKED-IN STATE */
    '34489' = '1' /* OTH SPCF PARALYTIC SYND */
    '3449'  = '1' /* PARALYSIS, UNSPECIFIED */
    '4382'  = '1' /* HEMIPLEGIA/HEMIPARESIS */
    '43820' = '1' /* LATE EF-HEMPLGA SIDE NOS */
    '43821' = '1' /* LATE EF-HEMPLGA DOM SIDE */
    '43822' = '1' /* LATE EF-HEMIPLGA NON-DOM */
    '4383'  = '1' /* MONOPLEGIA OF UPPER LIMB */
    '43830' = '1' /* LATE EF-MPLGA UP LMB NOS */
    '43831' = '1' /* LATE EF-MPLGA UP LMB DOM */
    '43832' = '1' /* LT EF-MPLGA UPLMB NONDOM */
    '4384'  = '1' /* MONOPLEGIA OF LOWER LIMB */
    '43840' = '1' /* LTE EF-MPLGA LOW LMB NOS */
    '43841' = '1' /* LTE EF-MPLGA LOW LMB DOM */
    '43842' = '1' /* LT EF-MPLGA LOWLMB NONDM */
    '4385'  = '1' /* OTHER PARALYTIC SYNDROME */
    '43850' = '1' /* LT EF OTH PARAL SIDE NOS */
    '43851' = '1' /* LT EF OTH PARAL DOM SIDE */
    '43852' = '1' /* LT EF OTH PARALS NON-DOM */
    '43853' = '1' /* LT EF OTH PARALS-BILAT */
    OTHER   = '0' ;

    /* Hemorrhage */
    /* ICD-9-CM diagnosis codes: */

    VALUE $HEMORID
    '2851'  = '1' /* ACUTE POSTHEMORRHAGIC ANEMIA */
    '4590'  = '1' /* OTHER DISORDERS OF CIRCULATORY SYSTEM, HEMORRHAGE,
                     UNSPECIFIED */
    '9582'  = '1' /* CERTAIN EARLY COMPLICATIONS OF TRAUMA, SECONDARY AND
                     RECURRENT HEMORRHAGE */
    '99811' = '1' /* HEMORRHAGE COMPLICATING A PROCEDURE */
    OTHER   = '0' ;

    /* Hip fracture */
    /* ICD-9-CM diagnosis codes: (includes all 5th digits) */

    VALUE $HIPFXID
    '820'   = '1' /* FRACTURE NECK OF FEMUR */
    '8200'  = '1' /* FRACTURE OF NECK OF FEMUR- TRANSCERVICAL FRACTURE, CLOSED
*/
    '82000' = '1' /* FX FEMUR INTRCAPS NOS-CL */
    '82001' = '1' /* FX UP FEMUR EPIPHY-CLOS */
    '82002' = '1' /* FX FEMUR, MIDCERVIC-CLOS */
    '82003' = '1' /* FX BASE FEMORAL NCK-CLOS */
    '82009' = '1' /* FX FEMUR INTRCAPS NEC-CL */
    '8201'   = '1' /* FRACTURE OF NECK OF FEMUR- TRANSCERVICAL FRACTURE, OPEN
*/
    '82010' = '1' /* FX FEMUR INTRCAP NOS-OPN */
    '82011' = '1' /* FX UP FEMUR EPIPHY-OPEN */
    '82012' = '1' /* FX FEMUR, MIDCERVIC-OPEN */
```

PSFMTS -- CONTINUED

```
    '82013' = '1' /* FX BASE FEMORAL NCK-OPEN */
    '82019' = '1' /* FX FEMUR INTRCAP NEC-OPN */
    '8202'   = '1' /* FRACTURE OF NECK OF FEMUR- PERTROCHANTERIC FRACTURE,
CLOSED */
    '82020' = '1' /* TROCHANTERIC FX NOS-CLOS */
    '82021' = '1' /* INTERTROCHANTERIC FX-CL */
    '82022' = '1' /* SUBTROCHANTERIC FX-CLOSE */
    '8203'  = '1' /* FRACTURE OF NECK OF FEMUR- PERTROCHANTERIC FRACTURE, OPEN
*/
    '82030' = '1' /* TROCHANTERIC FX NOS-OPEN */
    '82031' = '1' /* INTERTROCHANTERIC FX-OPN */
    '82032' = '1' /* SUBTROCHANTERIC FX-OPEN */
    '8208'  = '1' /* UNSPECIFIED PART OF NECK OF FEMUR, CLOSED */
    '8209'  = '1' /* UNSPECIFIED PART OF NECK OF FEMUR, OPEN */
    OTHER   = '0' ;

    /* Iatrogenic pneumothorax */;
    /* ICD-9-CM diagnosis codes: */;

    VALUE $IATROID
    '5121'  = '1'
    OTHER   = '0' ;

    /* Immunocompromised */
    /* ICD-9-CM diagnosis codes (includes all 4th and 5th digits) */

    VALUE $IMMUNID
    '042'   = '1' /* HUMAN IMMUNODEFICIENCY VIRUS DISEASE */
    '1363'  = '1' /* PNEUMOCYSTOSIS */
    '2790'  = '1' /* DEFICIENCY OF HUMORAL IMMUNITY */
    '27900' = '1' /* HYPOGAMMAGLOBULINEM NOS */
    '27901' = '1' /* SELECTIVE IGA IMMUNODEF */
    '27902' = '1' /* SELECTIVE IGM IMMUNODEF */
    '27903' = '1' /* SELECTIVE IG DEFIC NEC */
    '27904' = '1' /* CONG HYPOGAMMAGLOBULINEM */
    '27905' = '1' /* IMMUNODEFIC W HYPER-IGM */
    '27906' = '1' /* COMMON VARIABL IMMUNODEF */
    '27909' = '1' /* HUMORAL IMMUNITY DEF NEC */
    '2791'  = '1' /* DEFICIENCY OF CELL-MEDIATED IMMUNITY */
    '27910' = '1' /* IMMUNDEF T-CELL DEF NOS */
    '27911' = '1' /* DIGEORGES SYNDROME */
    '27912' = '1' /* WISKOTT-ALDRICH SYNDROME */
    '27913' = '1' /* NEZELOFS SYNDROME */
    '27919' = '1' /* DEFIC CELL IMMUNITY NOS */
    '2792'  = '1' /* COMBINED IMMUNITY DEFICIENCY */
    '2793'  = '1' /* UNSPECIFIED IMMUNITY DEFICIENCY */
    '2794'  = '1' /* AUTOIMMUNE DISEASE, NOT ELSEWHERE CLASSIFIED */
    '2798'  = '1' /* OTHER SPECIFIED DISORDERS INVOLVING THE IMMUNE MECHANISM
*/
    '2799'  = '1' /* UNSPECIFIED DISORDER OF IMMUNE MECHANISM */
    '9968'  = '1' /* COMPLICATIONS OF TRANSPLANTED ORGAN */
    '99680' = '1' /* COMP ORGAN TRANSPLNT NOS */
    '99681' = '1' /* COMPL KIDNEY TRANSPLANT */
    '99682' = '1' /* COMPL LIVER TRANSPLANT */
    '99683' = '1' /* COMPL HEART TRANSPLANT */
    '99684' = '1' /* COMPL LUNG TRANSPLANT */
    '99685' = '1' /* COMPL MARROW TRANSPLANT */
    '99686' = '1' /* COMPL PANCREAS TRANSPLNT */
```

```
'99687' = '1' /* COMP INTESTINE TRANSPLNT */
```
PSFMTS -- CONTINUED

```
'99689' = '1' /* COMP OTH ORGAN TRANSPLNT */
'V420'  = '1' /* KIDNEY REPLACED BY TRANSPLANT */
'V421'  = '1' /* HEART REPLACED BY TRANSPLANT */
'V426'  = '1' /* LUNG REPLACED BY TRANSPLANT */
'V427'  = '1' /* LIVER REPLACED BY TRANSPLANT */
'V4281' = '1' /* BONE MARROW SPECIFIED BY TRANSPLANT */
'V4282' = '1' /* PERIPHERAL STEM CELLS REPLACED BY TRANSPLANT */
'V4283' = '1' /* PANCREAS REPLACED BY TRANSPLANT */
'V4284' = '1' /* INTESTINES REPLACE BY TRANSPLANT */
'V4289' = '1' /* OTHER REPLACED BY TRANSPLANT */
OTHER   = '0' ;

/* Immunocompromised */
/* ICD-9-CM procedure codes (includes 4th and 5th digits:) */

VALUE $IMMUNIP
'335'  = '1' /* LUNG TRANSPLANT */
'3350' = '1' /* LUNG TRANSPLANT NOS */
'3351' = '1' /* UNILAT LUNG TRANSPLANT */
'3352' = '1' /* BILAT LUNG TRANSPLANT */
'336'  = '1' /* COMBINED HEART-LUNG TRANSPLANTATION */
'375'  = '1' /* HEART TRANSPLANTATION */
'410'  = '1' /* OPERATIONS ON BONE MAROW AND SPLEEN */
'4100' = '1' /* BONE MARROW TRNSPLNT NOS */
'4101' = '1' /* AUTO BONE MT W/O PURG */
'4102' = '1' /* ALO BONE MARROW TRNSPLNT */
'4103' = '1' /* ALLOGRFT BONE MARROW NOS */
'4104' = '1' /* AUTO HEM STEM CT W/O PUR */
'4105' = '1' /* ALLO HEM STEM CT W/O PUR */
'4106' = '1' /* CORD BLD STEM CELL TRANS */
'4107' = '1' /* AUTO HEM STEM CT W PURG */
'4108' = '1' /* ALLO HEM STEM CT W PURG */
'4109' = '1' /* AUTO BONE MT W PURGING */
'505'  = '1' /* LIVER TRANSPLANT */
'5051' = '1' /* AUXILIARY LIVER TRANSPL */
'5059' = '1' /* LIVER TRANSPLANT NEC */
'5280' = '1' /* PANCREATIC TRANSPLANT, NOS */
'5281' = '1' /* REIMPLANTATION OF PANCREATIC TISSUE */
'5283' = '1' /* HETEROTRANSPLANT OF PANCREAS */
'5285' = '1' /* ALLOTRANSPLANTATION OF CELLS OF ISLETS OF LNGERHANS */
'5286' = '1' /* TRANSPLANTATION OF CELLS OF ISLETS OF LANGERHANS, NOS */
'5569' = '1' /* OTHER KIDNEY TRANSPLANTATION */
OTHER  = '0' ;

/* Immunocompromised */
/* Diagnostic Related Groups (DRGs): */

VALUE $IMMUNDR
'488' = '1' /* HIV WITH EXTENSIVE OR PROCEDURE */
'489' = '1' /* HIV WITH MAJOR RELATED CONDITION */
'490' = '1' /* HIV WITH OR WITHOUT OTHER RELATED CONDITION */
OTHER = '0' ;

/* Indications of current drug abuse */
/* ICD-9-CM diagnosis codes: */
```

```
    VALUE $DRUGAID
    '2910'  = '1' /* ALCOHOL WITHDRAWAL DELIRIUM */
    '2911'  = '1' /* ALCOHOL AMNESTIC SYNDROME */
```
PSFMTS -- CONTINUED

```
    '2912'  = '1' /* OTHER ALCOHOLIC DEMENTIA */
    '2913'  = '1' /* ALCOHOL WITHDRAWAL HALLUCINOSIS */
    '2914'  = '1' /* IDIOSYNCRATIC ALCOHOL INTOXICATION */
    '2915'  = '1' /* ALCOHOL JEALOUSY */
    '2918'  = '1' /* OTHER SPECIFIED ALCOHOLIC PSYCHOSIS */
    '29181' = '1' /* ALCOHOL WITHDRAWAL */
    '2919'  = '1' /* ALCOHOLIC PSYCHOSES */
    '2920'  = '1' /* DRUG WITHDRAWL SYNDROME */
    '29211' = '1' /* DRUG-INDUCED ORGANIC DELUSIONAL SYNDROME */
    '29212' = '1' /* DRUG- INDUCED HALLUCINOSIS */
    '2922'  = '1' /* PATHOLOGICAL DRUG INTOXICATION */
    '29281' = '1' /* DRUG-INDUCED DELIRIUM */
    '29282' = '1' /* DRUG-INDUCED DEMENTIA */
    '29283' = '1' /* DRUG-INDUCED AMNESTIC SYNDROME */
    '29284' = '1' /* DRUG-INDUCED ORGANIC AFFECTIVE SYNDROME */
    '29289' = '1' /* OTHER SPECIFIED DRUG-INDUCED MENTAL DISORDERS */
    '2929'  = '1' /* UNSPECIFIED DRUG-INDUCED MENTAL DISORDER */
    '3030'  = '1' /* ALCOHOL DEPENDENCE SYNDROME */
    '3039'  = '1' /* OTHER AND UNSPECIFIED ALCOHOL DEPENDENCE */
    '3040'  = '1' /* OPIOID TYPE DEPENDENCE */
    '3041'  = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
                     DEPENDENCE */
    '3042'  = '1' /* COCAINE DEPENDENCE*/
    '3043'  = '1' /* CANNABIS DEPENDENCE */
    '3044'  = '1' /* AMPHETAMINE AND OTHER PSYCHOSTIMULANT DEPENDENCE */
    '3045'  = '1' /* HALLUCINOGEN DEPENDENCE */
    '3046'  = '1' /* OTHER SPECIFIED DRUG DEPENDENCE */
    '3047'  = '1' /* COMBINATIONS OF OPIOID TYPE DRUG WITH ANY OTHER */
    '3048'   = '1' /* COMBINATIONS OF DRUG DEPENDENCE EXCLUDING OPIOID TYPE
DRUG */
    '3049'  = '1' /* UNSPECIFIED DRUG DEPENDENCE */
    '3050'  = '1' /* TRIGEMINAL NERVE DISORDERS */
    '3052'  = '1' /* CANNABIS ABUSE */
    '3053'  = '1' /* HALLUCINOGEN ABUSE */
    '3054'   = '1' /* BARBITURATE AND SIMILARLY ACTING SEDATIVE OR HYPNOTIC
ABUSE */
    '3055'  = '1' /* OPIOID ABUSE */
    '3056'  = '1' /* COCAINE ABUSE */
    '3057'  = '1' /* AMPHETAMINE OR RELATED ACTING SYMPATHOMIMETIC ABUSE */
    '3058'  = '1' /* ANTIDEPRESSANT TYPE ABUSE */
    '3059'  = '1' /* OTHER MIXED OR UNSPECIFIED DRUG ABUSE */
    '9800'  = '1' /* ETHYL ALCOHOL */
    '9801'  = '1' /* METHYL ALCOHOL */
    '9802'  = '1' /* ISOPROPYL ALCOHOL */
    '9803'  = '1' /* FUSEL OIL */
    '981'   = '1' /* TOXIC EFFECT OF PETROLEUM PRODUCTS */
    '9820'  = '1' /* BENZENE AND HOMOLOGUES */
    '9821'  = '1' /* CARBON TETRACHLORIDE */
    '9822'  = '1' /* CARBON DISULFIDE */
    '9823'  = '1' /* OTHER CHLORINATED HYDROCARBON SOLVENTS */
    '9824'  = '1' /* NITROGLYCOL */
    '9828'  = '1' /* OTHER NONPETROLEUM-BASED SOLVENTS */
```

```
    '9830'  = '1' /* TOXIC EFFECT OF CORROSIVE AROMATICS */
    '9831'  = '1' /* TOXIC EFFECT OF ACIDS */
    '9832'  = '1' /* TOXIC EFFECT OF CAUSTIC ALKALIDES */
    '9839'  = '1' /* TOXIC EFFECT OF CAUSTIC, UNSPECIFIED */
    '9840'  = '1' /* INORGANIC LEAD COMPOUNDS */
    '9841'  = '1' /* ORGANIC LEAD COMPOUNDS */
    '9848'  = '1' /* OTHER LEAD COMPOUNDS */
```

PSFMTS -- CONTINUED

```
    '9849'  = '1' /* UNSPECIFIED LEAD COMPOUND */
    '9850'  = '1' /* MERCURY AND ITS COMPOUNDS */
    '9851'  = '1' /* ARSENIC AND ITS COMPOUNDS */
    '9852'  = '1' /* MANGANESE AND ITS COMPOUNDS */
    '9853'  = '1' /* BERYLLIUM AND ITS COMPOUNDS */
    '9854'  = '1' /* ANTIMONY AND ITS COMPOUNDS */
    '9855'  = '1' /* CADMIUM AND TIS COMPOUNDS */
    '9856'  = '1' /* CHROMIUM */
    '9858'  = '1' /* OTHER SPECIFIED METALS */
    '9859'  = '1' /* UNSPECIFIED METAL */
    '986'   = '1' /* TOXIC EFFECT OF CARBON MONOXIDE */
    '9870'  = '1' /* LIQUEFIED PETROLEUM GASES */
    '9871'  = '1' /* OTHER HYDROCARBON GAS */
    '9872'  = '1' /* NITROGEN OXIDES */
    '9873'  = '1' /* SULFUR DIOXIDE */
    '9874'  = '1' /* FREON */
    '9875'  = '1' /* LACRIMOGENIC GAS */
    '9876'  = '1' /* CHLORINE GAS */
    '9877'  = '1' /* HYDROCYANIC ACID GAS */
    '9878'  = '1' /* OTHER SPECIFIED GASES, FUMES, OR VAPORS */
    '9879'  = '1' /* UNSPECIFIED GAS, FUME, OR VAPOR */
    '9880'  = '1' /* FISH AND SHELLFISH */
    '9881'  = '1' /* MUSHROOMS */
    '9882'  = '1' /* BERRIES AND OTHER PLANTS */
    '9888'  = '1' /* OTHER SPECIFIED NOXIOUS SUBSTANCES EATEN AS FOOD */
    '9890'  = '1' /* HYDROCYANIC ACID AND CYANIDES */
    '9891'  = '1' /* STRYCHNINE AND SALTS */
    '9892'  = '1' /* CHLORINATED HYDROCARBONS */
    '9893'  = '1' /* ORGANOPHOSPHATE AND CARBAMATE */
    '9894'  = '1' /* OTHER PESTICIDES, NEC */
    '9895'  = '1' /* VENOM */
    '9896'  = '1' /* SOAPS AND DETERGENTS */
    '9897'  = '1' /* AFLATOXIN AND OTHER MYCOTOXIN [FOOD CONTAMINANTS] */
    '9898'  = '1' /* OTHER SUBSTANCES, CHIEFLY NONMEDICIAN AS TO SOURCE */
    '9899'  = '1' /* UNSPECIFIED SUBSTANCE, CHIEFLY NONMEDICIANL AS TO SOURCE
*/
    OTHER   = '0' ;

    /* Infection */
    /* ICD-9-CM diagnosis codes: */

    VALUE $INFECID
    '5400'  = '1' /* ACUTE APPENDICITIS WITH GENERALIZED PERITONITIS */
    '5401'  = '1' /* ACUTE APPENDICITIS WITH PERITONEAL ABSCESS */
    '5409'  = '1' /* ACUTE APPENDICITIS WITHOUT MENTION OF PERITONITIS */
    '541'   = '1' /* APPENDICITIS, UNQUALIFIED */
    '542'   = '1' /* OTHER APPENDICITIS */
    '56201' = '1' /* DIVERTICULITIS OF SMALL INTESTINE (WITHOUT MENTION OF
                     HEMORRHAGE) */
```

```
    '56203' = '1' /* DIVERTICULITIS OF SMALL INTESTINE WITH HEMORRHAGE */
    '56211' = '1' /* DIVERTICULITIS OF COLON (WITHOUT MENTION OF HEMORRHAGE)
*/
    '56213' = '1' /* DIVERTICULITIS OF COLON WITH HEMORRHAGE */
    '566'   = '1' /* ABSCESS OF ANAL AND RECTAL REGIONS */
    '5670'  = '1' /* PERITONITIS IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE
*/
    '5671'  = '1' /* PNEUMOCOCCAL PERITONITIS */
    '5672'  = '1' /* OTHER SUPPURATIVE PERITONITIS */
    '5678'  = '1' /* OTHER SPECIFIED PERITONITIS */
```

PSFMTS -- CONTINUED

```
    '5679'  = '1' /* UNSPECIFIED PERITONITIS */
    '5695'  = '1' /* ABSCESS OF INTESTINE */
    '56961' = '1' /* INFECTION OF COLOSTOMY OR ENTEROSTOMY */
    '5720'  = '1' /* ABSCESS OF LIVER */
    '5721'  = '1' /* PORTAL PYEMIA */
    '57400'  =  '1'  /*  CALCULUS  OF  GALLBLADDER  WITH  ACUTE  CHOLECYSTITIS  -
WITHOUT
                     MENTION OF OBSTRUCTION */
    '57401' = '1' /* CALCULUS OF GALLBLADDER WITH ACUTE CHOLECYSTITS - WITH
                     OBSTRUCTION */
    '57430' = '1' /* CALCULUS OF BILE DUCT WITH ACUTE CHOLECYSTITIS - WITHOUT
                     MENTION OF OBSTRUCTION */
    '57431' = '1' /* CALCULUS OF BILE DUCT WITH ACUTE CHOLECYSTITIS - WITH
                     OBSTRUCTION */
    '57460'  =  '1'  /*  CALCULUS  OF  GALLBLADDER  AND  BILE  DUCT  WITH  ACUTE
CHOLECYSTITIS
                     - WITHOUT MENTION OF OBSTRUCTION */
    '57461'  =  '1'  /*  CALCULUS  OF  GALLBLADDER  AND  BILE  DUCT  WITH  ACUTE
CHOLECYSTITIS
                     -WITH OBSTRUCTION */
    '57480'  =  '1'  /*  CALCULUS  OF  GALLBLADDER  AND  BILE  DUCT  WITH  ACUTE  AND
CHRONIC
                     CHOLECYSTITIS - WITHOUT MENTION OF OBSTRUCTION */
    '57481'  =  '1'  /*  CALCULUS  OF  GALLBLADDER  AND  BILE  DUCT  WITH  ACUTE  AND
CHRONIC
                     CHOLECYSTITIS - WITH OBSTRUCTION */
    '5750'  = '1' /* ACUTE CHOLECYSTITIS */
    '5754'  = '1' /* PERFORATION OF GALLBLADDER */
    '5761'  = '1' /* CHOLANGITIS */
    '5763'  = '1' /* PERFORATION OF BILE DUCT */
    OTHER   = '0' ;

    /* Infection */
    /* Diagnostic Related Groups (DRGs) */

    VALUE $INFECDR
    '020' = '1' /* NERVOUS SYSTEM INFECTION EXCEPT VIRAL MENINGITIS */
    '068' = '1' /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITH CC */
    '069' = '1' /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITHOUT CC */
    '079' = '1' /* RESPIRATORY INFECTIONS AND INFLAMMATIONS, AGE GREATER THAN
                   17 WITH CC */
    '080' = '1' /* RESPIRATORY INF' = '1' /* ECTIONS AND INFLAMMATIONS, AGE
                   GREATER THAN 17 WITHOUT CC */
    '089' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE GREATER THAN 17 WITH CC
*/
```

```
    '090' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE GREATER THAN 17 WITHOUT
CC */
    '126' = '1' /* ACUTE AND SUBACUTE ENDOCARDITIS */
    '238' = '1' /* OSTEOMYELITIS */
    '242' = '1' /* SEPTIC ARTHRITIS */
    '277' = '1' /* CELLULITIS, AGE GREATER THAN 17 WITH CC */
    '278' = '1' /* CELLULITIS, AGE GREATER THAN 17 WITHOUT CC */
    '320' = '1' /* KIDNEY AND URINARY TRACT INFECTIONS, AGE GREATER THAN 17
                   WITH CC */
    '321' = '1' /* KIDNEY AND URINARY TRACT INFECTIONS, AGE GREATER THAN 17
                   WITHOUT CC */
    '368' = '1' /* INFECTIONS OF FEMALE REPRODUCTIVE SYSTEM */
    '416' = '1' /* SEPTICEMIA, AGE GREATER THAN 17 */
    OTHER = '0' ;
```

PSFMTS -- CONTINUED

```
    /* Infection due to medical care */;
    /* ICD-9-CM diagnosis codes: */;

    VALUE $IDTMCID
    '99662' = '1' /* DUE TO OTHER VASCULAR DEVICE, IMPLANT, AND GRAFT */
    '9993'  = '1' /* OTHER INFECTION */
    OTHER   = '0' ;

    /* Instrument assisted delivery */
    /* ICD-9-CM procedure codes: */

    VALUE $INSTRIP
    '720'  = '1' /* LOW FORCEPS OPERATION */
    '721'  = '1' /* LOW FORCEPS OPERATION WITH EPISIOTOMY */
    '722'  = '1' /* MID FORCEPS OPERATION */
    '7221' = '1' /* MID FORCEPS OPERATION WITH EPISIOTOMY */
    '7229' = '1' /* OTHER MID FORCEPS OPERATION */
    '723'  = '1' /* HIGH FORCEPS OPERATION */
    '7231' = '1' /* HIGH FORCEPS OPERATION WITH EPISIOTOMY */
    '7239' = '1' /* OTHER HIGH FORCEPS OPERATION */
    '724'  = '1' /* FORCEPS ROTATION OF FETAL HEAD */
    '7251' = '1' /* PARTIAL BREECH EXTRACTION WITH FORCEPS TO AFTERCOMING HEAD
*/
    '7253' = '1' /* TOTAL BREECH EXTRACTION WITH FORCEPS TO AFTERCOMING HEAD
*/
    '726'  = '1' /* FORCEPS APPLICATION TO AFTERCOMING HEAD */
    '7271' = '1' /* VACUUM EXTRACTION WITH EPISIOTOMY */
    '728'  = '1' /* OTHER SPECIFIED INSTRUMENTAL DELIVERY */
    '729'  = '1' /* UNSPECIFIED INSTRUMENTAL DELIVERY[h1] */
    OTHER  = '0' ;

    /* Liveborn */
    /* Diagnostic Related Groups (DRG): */

    VALUE $LIVEBDR
    '385' = '1' /* NEONATES, DIED OR TRANSFERRED TO ANOTHER ACUTE CARE
FACILITY */
    '386' = '1' /* EXTREME IMMATURITY OR RESPIRATORY DISTRESS SYNDROME OF
NEONATE */
    '387' = '1' /* PREMATURITY WITH MAJOR PROBLEMS */
    '388' = '1' /* PREMATURITY WITHOUT MAJOR PROBLEMS */
```

```
'389' = '1' /* FULL TERM NEONATE WITH MAJOR PROBLEMS */
'390' = '1' /* NEONATE WITH OTHER SIGNIFICANT PROBLEMS */
'391' = '1' /* NORMAL NEWBORN */
OTHER = '0' ;

/* Liveborn */
/* ICD-9-CM diagnosis codes(includes 4th and 5th digits) */
/* Admission type recorded as (4): */

VALUE $LIVEBID
'7640'  = '1' /* LT-FOR-DATES W/O FET MAL */
'76400' = '1' /* LIGHT-FOR-DATES WTNOS */
'76401' = '1' /* LIGHT-FOR-DATES <500G */
'76402' = '1' /* LT-FOR-DATES 500-749G */
'76403' = '1' /* LT-FOR-DATES 750-999G */
'76404' = '1' /* LT-FOR-DATES 1000-1249G */
'76405' = '1' /* LT-FOR-DATES 1250-1499G */
```
PSFMTS -- CONTINUED
```
'76406' = '1' /* LT-FOR-DATES 1500-1749G */
'76407' = '1' /* LT-FOR-DATES 1750-1999G */
'76408' = '1' /* LT-FOR-DATES 2000-2499G */
'76409' = '1' /* LT-FOR-DATES 2500+G */
'7641'  = '1' /* LT-FOR-DATES W FETAL MAL */
'76410' = '1' /* LT-FOR-DATE W/MAL WTNOS */
'76411' = '1' /* LT-FOR-DATE W/MAL <500G */
'76412' = '1' /* LT-DATE W/MAL 500-749G */
'76413' = '1' /* LT-DATE W/MAL 750-999G */
'76414' = '1' /* LT-DATE W/MAL 1000-1249G */
'76415' = '1' /* LT-DATE W/MAL 1250-1499G */
'76416' = '1' /* LT-DATE W/MAL 1500-1749G */
'76417' = '1' /* LT-DATE W/MAL 1750-1999G */
'76418' = '1' /* LT-DATE W/MAL 2000-2499G */
'76419' = '1' /* LT-FOR-DATE W/MAL 2500+G */
'7642'  = '1' /* FET MAL W/O LT-FOR-DATES */
'76420' = '1' /* FETAL MALNUTRITION WTNOS */
'76421' = '1' /* FETAL MALNUTRITION <500G */
'76422' = '1' /* FETAL MALNUTR 500-749G */
'76423' = '1' /* FETAL MAL 750-999G */
'76424' = '1' /* FETAL MAL 1000-1249G */
'76425' = '1' /* FETAL MAL 1250-1499G */
'76426' = '1' /* FETAL MAL 1500-1749G */
'76427' = '1' /* FETAL MALNUTR 1750-1999G */
'76428' = '1' /* FETAL MALNUTR 2000-2499G */
'76429' = '1' /* FETAL MALNUTR 2500+G */
'7649'  = '1' /* FETAL GROWTH RETARD NOS */
'76490' = '1' /* FET GROWTH RETARD WTNOS */
'76491' = '1' /* FET GROWTH RETARD <500G */
'76492' = '1' /* FET GROWTH RET 500-749G */
'76493' = '1' /* FET GROWTH RET 750-999G */
'76494' = '1' /* FET GRWTH RET 1000-1249G */
'76495' = '1' /* FET GRWTH RET 1250-1499G */
'76496' = '1' /* FET GRWTH RET 1500-1749G */
'76497' = '1' /* FET GRWTH RET 1750-1999G */
'76498' = '1' /* FET GRWTH RET 2000-2499G */
'76499' = '1' /* FET GROWTH RET 2500+G */
'7650'  = '1' /* EXTREME IMMATURITY */
'76500' = '1' /* EXTREME IMMATUR WTNOS */
```

```
'76501' = '1' /* EXTREME IMMATUR <500G */
'76502' = '1' /* EXTREME IMMATUR 500-749G */
'76503' = '1' /* EXTREME IMMATUR 750-999G */
'76504' = '1' /* EXTREME IMMAT 1000-1249G */
'76505' = '1' /* EXTREME IMMAT 1250-1499G */
'76506' = '1' /* EXTREME IMMAT 1500-1749G */
'76507' = '1' /* EXTREME IMMAT 1750-1999G */
'76508' = '1' /* EXTREME IMMAT 2000-2499G */
'76509' = '1' /* EXTREME IMMAT 2500+G */
'7651'  = '1' /* OTHER PRETERM INFANTS */
'76510' = '1' /* PRETERM INFANT NEC WTNOS */
'76511' = '1' /* PRETERM NEC <500G */
'76512' = '1' /* PRETERM NEC 500-749G */
'76513' = '1' /* PRETERM NEC 750-999G */
'76514' = '1' /* PRETERM NEC 1000-1249G */
'76515' = '1' /* PRETERM NEC 1250-1499G */
'76516' = '1' /* PRETERM NEC 1500-1749G */
'76517' = '1' /* PRETERM NEC 1750-1999G */
'76518' = '1' /* PRETERM NEC 2000-2499G */
'76519' = '1' /* PRETERM NEC 2500+G */
```

PSFMTS -- CONTINUED

```
'766'   = '1' /* LONG GESTAT/HIGH BIRTHWT */
'7660'  = '1' /* EXCEPTIONALLY LARGE BABY */
'7661'  = '1' /* HEAVY-FOR-DATE INFAN NEC */
'7662'  = '1' /* POST-TERM INFANT NOS */
'767'   = '1' /* BIRTH TRAUMA */
'7670'  = '1' /* CEREBRAL HEM AT BIRTH */
'7671'  = '1' /* SCALP INJURY AT BIRTH */
'7672'  = '1' /* CLAVICLE FX AT BIRTH */
'7673'  = '1' /* BONE INJURY NEC AT BIRTH */
'7674'  = '1' /* SPINAL CORD INJ AT BIRTH */
'7675'  = '1' /* FACIAL NERVE INJ-BIRTH */
'7676'  = '1' /* BRACH PLEXUS INJ-BIRTH */
'7677'  = '1' /* NERVE INJ NEC AT BIRTH */
'7678'  = '1' /* BIRTH TRAUMA NEC */
'7679'  = '1' /* BIRTH TRAUMA NOS */
'768'   = '1' /* INTRAUTERINE ASPHYXIA */
'7680'  = '1' /* FETAL DEATH-ANOXIA NOS */
'7681'  = '1' /* FET DEATH-ANOXIA DUR LAB */
'7682'  = '1' /* FET DISTRESS BEFOR LABOR */
'7683'  = '1' /* FETAL DISTRESS DUR LABOR */
'7684'  = '1' /* FETAL DISTRESS NOS */
'7685'  = '1' /* SEVERE BIRTH ASPHYXIA */
'7686'  = '1' /* MILD/MOD BIRTH ASPHYXIA */
'7689'  = '1' /* BIRTH ASPHYXIA NOS */
'769'   = '1' /* RESPIRATORY DISTRESS SYN */
'770'   = '1' /* OTH NB RESPIRATORY COND */
'7700'  = '1' /* CONGENITAL PNEUMONIA */
'7701'  = '1' /* MECONIUM ASPIRATN SYNDRM */
'7702'  = '1' /* NB INTERSTIT EMPHYSEMA */
'7703'  = '1' /* NB PULMONARY HEMORRHAGE */
'7704'  = '1' /* PRIMARY ATELECTASIS */
'7705'  = '1' /* NB ATELECTASIS NEC/NOS */
'7706'  = '1' /* NB TRANSITORY TACHYPNEA */
'7707'  = '1' /* PERINATAL CHR RESP DIS */
'7708'  = '1' /* POST-BIRTH RESP PROB NEC */
'7709'  = '1' /* NB RESPIRATORY COND NOS */
```

```
'771'   = '1' /* PERINATAL INFECTION */
'7710'  = '1' /* CONGENITAL RUBELLA */
'7711'  = '1' /* CONG CYTOMEGALOVIRUS INF */
'7712'  = '1' /* CONGENITAL INFEC NEC */
'7713'  = '1' /* TETANUS NEONATORUM */
'7714'  = '1' /* OMPHALITIS OF NEWBORN */
'7715'  = '1' /* NEONATAL INFEC MASTITIS */
'7716'  = '1' /* NEONATAL CONJUNCTIVITIS */
'7717'  = '1' /* NEONATAL CANDIDA INFECT */
'7718'  = '1' /* PERINATAL INFECTION NEC */
'772'   = '1' /* FETAL/NEONATAL HEMORRHAG */
'7720'  = '1' /* FETAL BLOOD LOSS NEC */
'7721'  = '1' /* NB INTRAVENTRICULAR HEM */
'7722'  = '1' /* NB SUBARACHNOID HEMORR */
'7723'  = '1' /* POST-BIRTH UMBIL HEMORR */
'7724'  = '1' /* NB GI HEMORRHAGE */
'7725'  = '1' /* NB ADRENAL HEMORRHAGE */
'7726'  = '1' /* NB CUTANEOUS HEMORRHAGE */
'7728'  = '1' /* NEONATAL HEMORRHAGE NEC */
'7729'  = '1' /* NEONATAL HEMORRHAGE NOS */
'773'   = '1' /* NB HEMOLYT DIS:ISOIMMU */
'7730'  = '1' /* NB HEMOLYT DIS:RH ISOIMM */
'7731'  = '1' /* NB HEMOLYT DIS-ABO ISOIM */
```

PSFMTS -- CONTINUED

```
'7732'  = '1' /* NB HEMOLYT DIS-ISOIM NEC */
'7733'  = '1' /* HYDROPS FETALIS:ISOIMM */
'7734'  = '1' /* NB KERNICTERUS:ISOIMMUN */
'7735'  = '1' /* NB LATE ANEMIA:ISOIMMUN */
'774'   = '1' /* OTHER PERINATAL JAUNDICE */
'7740'  = '1' /* PERINAT JAUND-HERED ANEM */
'7741'  = '1' /* PERINAT JAUND:HEMOLYSIS */
'7742'  = '1' /* NEONAT JAUND PRETERM DEL */
'7743'  = '1' /* OTH DELAY CONJUGAT JAUND */
'77430' = '1' /* DELAY CONJUGAT JAUND NOS */
'77431' = '1' /* NEONAT JAUND IN OTH DIS */
'77439' = '1' /* DELAY CONJUGAT JAUND NEC */
'7744'  = '1' /* FETAL/NEONATAL HEPATITIS */
'7745'  = '1' /* PERINATAL JAUNDICE NEC */
'7746'  = '1' /* FETAL/NEONATAL JAUND NOS */
'7747'  = '1' /* NB KERNICTERUS */
'775'   = '1' /* NB ENDOCRIN/METABOL DIS */
'7750'  = '1' /* INFANT DIABET MOTHER SYN */
'7751'  = '1' /* NEONAT DIABETES MELLITUS */
'7752'  = '1' /* NEONAT MYASTHENIA GRAVIS */
'7753'  = '1' /* NEONATAL THYROTOXICOSIS */
'7754'  = '1' /* HYPOCALCEM/HYPOMAGNES NB */
'7755'  = '1' /* NEONATAL DEHYDRATION */
'7756'  = '1' /* NEONATAL HYPOGLYCEMIA */
'7757'  = '1' /* LATE METAB ACIDOSIS NB */
'7758'  = '1' /* TRANSIENT MET DIS NB NEC */
'7759'  = '1' /* TRANSIENT MET DIS NB NOS */
'776'   = '1' /* HEMATOLOGICAL DIS OF NB */
'7760'  = '1' /* NB HEMORRHAGIC DISEASE */
'7761'  = '1' /* NEONATAL THROMBOCYTOPEN */
'7762'  = '1' /* DISSEM INTRAVASC COAG NB */
'7763'  = '1' /* OTH NEONATAL COAG DIS */
'7764'  = '1' /* POLYCYTHEMIA NEONATORUM */
```

```
'7765'  = '1' /* CONGENITAL ANEMIA */
'7766'  = '1' /* ANEMIA OF PREMATURITY */
'7767'  = '1' /* NEONATAL NEUTROPENIA */
'7768'  = '1' /* TRANSIENT HEMAT DIS NEC */
'7769'  = '1' /* NB HEMATOLOGICAL DIS NOS */
'777'   = '1' /* PERINATAL GI SYSTEM DIS */
'7771'  = '1' /* MECONIUM OBSTRUCTION */
'7772'  = '1' /* INTEST OBST-INSPISS MILK */
'7773'  = '1' /* SWALLOWED BLOOD SYNDROME */
'7774'  = '1' /* TRANSITORY ILEUS OF NB */
'7775'  = '1' /* NECROT ENTEROCOLITIS NB */
'7776'  = '1' /* PERINATAL INTEST PERFOR */
'7778'  = '1' /* PERINAT GI SYS DIS NEC */
'7779'  = '1' /* PERINAT GI SYS DIS NOS */
'778'   = '1' /* INTEGUMENT/TEMP COND NB */
'7780'  = '1' /* HYDROPS FETALIS NO ISOIM */
'7781'  = '1' /* SCLEREMA NEONATORUM */
'7782'  = '1' /* NB COLD INJURY SYNDROME */
'7783'  = '1' /* NB HYPOTHERMIA NEC */
'7784'  = '1' /* NB TEMP REGULAT DIS NEC */
'7785'  = '1' /* EDEMA OF NEWBORN NEC/NOS */
'7786'  = '1' /* CONGENITAL HYDROCELE */
'7787'  = '1' /* NB BREAST ENGORGEMENT */
'7788'  = '1' /* NB INTEGUMENT COND NEC */
'7789'  = '1' /* NB INTEGUMENT COND NOS */
'779'   = '1' /* OTH PERINATAL CONDITION */
```

PSFMTS -- CONTINUED

```
'7790'  = '1' /* CONVULSIONS IN NEWBORN */
'7791'  = '1' /* NB CEREB IRRIT NEC/NOS */
'7792'  = '1' /* CNS DYSFUNCTION SYN NB */
'7793'  = '1' /* NB FEEDING PROBLEMS */
'7794'  = '1' /* NB DRUG REACTION/INTOXIC */
'7795'  = '1' /* NB DRUG WITHDRAWAL SYNDR */
'7796'  = '1' /* TERMINATION OF PREGNANCY */
'7798'  = '1' /* PERINATAL CONDITION NEC */
'7799'  = '1' /* PERINATAL CONDITION NOS */
'V30'   = '1' /* SINGLE LIVEBORN */
'V300'  = '1' /* SINGLE LIVEBORN-IN HOSP */
'V3000' = '1' /* SINGLE LB IN-HOSP W/O CS */
'V3001' = '1' /* SINGLE LB IN-HOSP W CS */
'V301'  = '1' /* SINGL LIVEBRN-BEFORE ADM */
'V302'  = '1' /* SINGLE LIVEBORN-NONHOSP */
'V31'   = '1' /* TWIN, MATE LIVEBORN */
'V310'  = '1' /* TWIN, MATE LB-IN HOSP */
'V3100' = '1' /* TWIN-MATE LB-HOSP W/O CS */
'V3101' = '1' /* TWIN-MATE LB-IN HOS W CS */
'V311'  = '1' /* TWIN, MATE LB-BEFORE ADM */
'V312'  = '1' /* TWIN, MATE LB-NONHOSP */
'V32'   = '1' /* TWIN, MATE STILLBORN */
'V320'  = '1' /* TWIN, MATE SB-IN HOSP */
'V3200' = '1' /* TWIN-MATE SB-HOSP W/O CS */
'V3201' = '1' /* TWIN-MATE SB-HOSP W CS */
'V321'  = '1' /* TWIN, MATE SB-BEFORE ADM */
'V322'  = '1' /* TWIN, MATE SB-NONHOSP */
'V33'   = '1' /* TWIN NOS */
'V330'  = '1' /* TWIN NOS-IN HOSPITAL */
'V3300' = '1' /* TWIN-NOS-IN HOSP W/O CS */
```

```
    'V3301' = '1' /* TWIN-NOS-IN HOSP W CS */
    'V331'  = '1' /* TWIN NOS-BEFORE ADMISSN */
    'V332'  = '1' /* TWIN NOS-NONHOSP */
    'V34'   = '1' /* OTH MULT BIRTH, ALL LIVE */
    'V340'  = '1' /* OTH MULTIPLE NB-IN HOSP */
    'V3400' = '1' /* OTH MULT LB-HOSP W/O CS */
    'V3401' = '1' /* OTH MULT LB-IN HOSP W CS */
    'V341'  = '1' /* OTH MULT NB-BEFORE ADM */
    'V342'  = '1' /* OTH MULTIPLE NB-NONHOSP */
    'V35'   = '1' /* OTH MULT BIRTH, ALL SB */
    'V350'  = '1' /* OTH MULTIPLE SB-IN HOSP */
    'V3500' = '1' /* OTH MULT SB-HOSP W/O CS */
    'V3501' = '1' /* OTH MULT SB-IN HOSP W CS */
    'V351'  = '1' /* OTH MULT SB-BEFORE ADM */
    'V352'  = '1' /* OTH MULTIPLE SB-NONHOSP */
    'V36'   = '1' /* OTH MULT BRTH, SOME LIVE */
    'V360'  = '1' /* MULTIPLE NB/SB-IN HOSP */
    'V3600' = '1' /* MULT LB/SB-IN HOS W/O CS */
    'V3601' = '1' /* MULT LB/SB-IN HOSP W CS */
    'V361'  = '1' /* MULT NB/SB-BEFORE ADM */
    'V362'  = '1' /* MULTIPLE NB/SB-NONHOSP */
    'V37'   = '1' /* OTH/UNSPEC MULTIPLE BRTH */
    'V370'  = '1' /* MULT BIRTH NOS-IN HOSP */
    'V3700' = '1' /* MULT BRTH NOS-HOS W/O CS */
    'V3701' = '1' /* MULT BIRTH NOS-HOSP W CS */
    'V371'  = '1' /* MULT BRTH NOS-BEFORE ADM */
    'V372'  = '1' /* MULT BIRTH NOS-NONHOSP */
    'V39'   = '1' /* LIVEBORN NOS */
    'V390'  = '1' /* LIVEBORN NOS-IN HOSP */
    PSFMTS -- CONTINUED

    'V3900' = '1' /* LIVEBORN NOS-HOSP W/O CS */
    'V3901' = '1' /* LIVEBORN NOS-HOSP W CS */
    'V391'  = '1' /* LIVEBORN NOS-BEFORE ADM */
    'V392'  = '1' /* LIVEBORN NOS-NONHOSP */
    OTHER   = '0' ;

    /* Low mortality */
    /* Diagnostic Related Groups DRGs */

    VALUE $LOWMODR
    '015' = '1' /* TRANSIENT ISCHEMIC ATTACK AND PRECEREBRAL OCCLUSIONS */
    '021' = '1' /* VIRAL MENINGITIS */
    '026' = '1' /* SEIZURE AND HEADACHE, AGE 0-17 */
    '030' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE 0-
17 */
    '031' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITH CC */
    '032' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITHOUT CC */
    '033' = '1' /* CONCUSSION, AGE 0-17 */
    '036' = '1' /* RETINAL PROCEDURES */
    '037' = '1' /* ORBITAL PROCEDURES */
    '042' = '1' /* INTRAOCULAR PROCEDURES */
    '044' = '1' /* ACUTE MAJOR EYE INFECTIONS */
    '045' = '1' /* NEUROLOGICAL EYE DISORDERS */
    '050' = '1' /* SIALOADENECTOMY */
    '052' = '1' /* CLEFT LIP AND PALATE REPAIR */
    '053' = '1' /* SINUS AND MASTOID PROCEDURES, AGE GREATER THAN 17 */
    '055' = '1' /* MISCELLANEOUS EAR, NOSE, MOUTH AND THROAT PROCEDURES */
```

```
    '057'  =  '1'  /*  TONSILLECTOMY  AND  ADENOIDECTOMY  PROCEDURES  EXCEPT
TONSILLECTOMY
                  AND/OR ADENOIDECTOMY ONLY, AGE GREATER THAN 17 */
    '060' = '1' /* TONSILLECTOMY AND/OR ADENOIDECTOMY ONLY, AGE 0-17 */
    '062' = '1' /* MYRINGOTOMY WITH TUBE INSERTION, AGE 0-17 */
    '063' = '1' /* OTHER EAR, NOSE, MOUTH AND THROAT OR PROCEDURES */
    '065' = '1' /* DYSEQUILIBRIUM */
    '068' = '1' /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITH CC */
    '070' = '1' /* OTITIS MEDIA AND URI, AGE 0-17 */
    '071' = '1' /* LARYNGOTRACHEITIS */
    '074' = '1' /* OTHER EAR, NOSE, MOUTH AND THROAT DIAGNOSES, AGE 0-17 */
    '091' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE 0-17 */
    '096' = '1' /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITH CC */
    '097' = '1' /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITHOUT CC */
    '098' = '1' /* BRONCHITIS AND ASTHMA, AGE 0-17 */
    '125' = '1' /* CIRCULATORY DISORDERS EXCEPT ACUTE MYOCARDIAL INFARCTION
WITH
                  CARDIAC CATHETERIZATION WITHOUT COMPLEX DIAGNOSIS */
    '134' = '1' /* HYPERTENSION */
    '140' = '1' /* ANGINA PECTORIS */
    '141' = '1' /* SYNCOPE AND COLLAPSE WITH CC */
    '142' = '1' /* SYNCOPE AND COLLAPSE WITHOUT CC */
    '143' = '1' /* CHEST PAIN */
    '156' = '1' /* STOMACH, ESOPHAGEAL AND DUODENAL PROCEDURES, AGE 0-17 */
    '163' = '1' /* HERNIA PROCEDURES, AGE 0-17 */
    '166' = '1' /* APPENDECTOMY WITHOUT COMPLICATED PRINCIPAL DIAGNOSIS WITH
CC */
    '167'  =  '1'  /*  APPENDECTOMY  WITHOUT  COMPLICATED  PRINCIPAL  DIAGNOSIS
WITHOUT CC *
    /
    '184' = '1' /* ESOPHAGITIS, GASTROENTERITIS AND MISCELLANEOUS DIGESTIVE
                  DISORDERS, AGE 0-17 */
```

PSFMTS -- CONTINUED

```
    '190' = '1' /* OTHER DIGESTIVE SYSTEM DIAGNOSES, AGE 0-17 */
    '212' = '1' /* HIP AND FEMUR PROCEDURES EXCEPT MAJOR JOINT PROCEDURES, AGE
                  0-17 */
    '218' = '1' /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT AND
                  FEMUR, AGE GREATER THAN 17 WITH CC */
    '219' = '1' /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT AND
                  FEMUR, AGE GREATER THAN 17 WITHOUT CC */
    '220' = '1' /* LOWER EXTREMITY AND HUMEROUS PROCEDURES EXCEPT HIP, FOOT
AND
                  FEMUR, AGE 0-17 */
    '223' = '1' /* MAJOR SHOULDER, ELBOW PROCEDURES OR OTHER UPPER EXTREMITY
                  PROCEDURES WITH CC */
    '224' = '1' /* SHOULDER, ELBOW OR FOREARM PROCEDURES EXCEPT MAJOR JOINT
                  PROCEDURES WITHOUT CC */
    '225' = '1' /* FOOT PROCEDURES */
    '228' = '1' /* MAJOR THUMB OR JOINT PROCEDURES OR OTHER HAND OR WRIST
                  PROCEDURES WITH CC */
    '229' = '1' /* HAND OR WRIST PROCEDURES EXCEPT MAJOR JOINT PROCEDURES
                  WITHOUT CC */
    '232' = '1' /* ARTHROSCOPY */
    '237' = '1' /* SPRAINS, STRAINS AND DISLOCATIONS OF HIP, PELVIS AND THIGH
*/
    '243' = '1' /* MEDICAL BACK PROBLEMS */
    '246' = '1' /* NONSPECIFIC ARTHROPATHIES */
```

```
    '252' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF FOREARM,
                   HAND AND FOOT, AGE 0-17 */
    '255' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF UPPER ARM
                   AND LOWER LEG EXCEPT FOOT, AGE 0-17 */
    '257' = '1' /* TOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '258' = '1' /* TOTAL MASTECTOMY FOR MALIGNNACY WITHOUT CC */
    '261' = '1' /* BREAST PROCEDURE FOR NONMALIGNANCY EXCEPT BIOPSY AND LOCAL
                   EXCISION */
    '262' = '1' /* BREAST BIOPSY AND LOCAL EXCISION OF NONMALIGNANCY */
    '267' = '1' /* PERIANAL AND PILONICAL PROCEDURES */
    '279' = '1' /* CELLULITIS, AGE 0-17 */
    '282' = '1' /* TRAUMA TO SKIN, SUBCUTANEOUS TISSUE AND BREAST, AGE 0-17 */
    '289' = '1' /* PARATHYROID PROCEDURES */
    '290' = '1' /* THYROID PROCEDURES */
    '293' = '1' /* OTHER ENDOCRINE, NUTRITIONAL AND METABOLIC OR PROCEDURES
                   WITHOUT CC */
    '295' = '1' /* DIABETES, AGE 0-35 */
    '298' = '1' /* NUTRITIONAL AND MISCELLANEOUS METABOLIC DISORDERS, AGE
                   GREATER THAN 17 WITHOUT CC */
    '317' = '1' /* ADMISSION FOR RENAL DIALYSIS */
    '322' = '1' /* KIDNEY AND URINARY TRACT INFECTION, AGE 0-17 */
    '323' = '1' /* URINARY STONES WITH CC AND/OR ESW LITHOTRIPSY */
    '324' = '1' /* URINARY STONES WITHOUT CC */
    '333' = '1' /* OTHER KIDNEY AND URINARY TRACT DIAGNOSES, AGE 0-17 */
    '334' = '1' /* MAJOR MALE PELVIC PROCEDURES WITH CC */
    '335' = '1' /* MAJOR MALE PELVIC PROCEDURES WITHOUT CC */
    '336' = '1' /* TRANSURETHRAL PROSTATECTOMY WITH CC */
    '337' = '1' /* TRANSURETHRAL PROSTATECTOMY WITHOUT CC */
    '351' = '1' /* STERILIZATION, MALE */
    '356' = '1' /* FEMALE REPRODUCTION SYSTEM RECONCSTRUCTIVE PROCEDURES */
    '358' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITH CC */
    '359' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITHOUT CC
*/
    '360' = '1' /* VAGINA, CERVIX AND VULVA PROCEDURES */
    '361' = '1' /* LAPAROSCOPY AND INCISIONAL TUBAL INTERRUPTION */
```

PSFMTS -- CONTINUED

```
    '362' = '1' /* ENDOSCOPIC TUBAL INTERRUPTION */
    '364' = '1' /* D AND C, CONIZATION EXCEPT FOR MALIGNANCY */
    '369' = '1' /* MENSTRUAL AND OTHER FEMALE REPRODUCTIVE SYSTEM DISORDERS */
    '370' = '1' /* CESAREAN SCTION WITH CC */
    '371' = '1' /* CESAREAN SECTION WITHOUT CC */
    '372' = '1' /* VAGINAL DELIVERY WITH COMPLICATING DIAGNOSES */
    '373' = '1' /* VAGINAL DELIVERY WITHOUT COMPLICATING DIAGNOSES */
    '374' = '1' /* VAGINAL DELIVERY WITH STERILIZATION AND/OR D AND C */
    '375' = '1' /* VAGINAL DELIVERY WITH OR PROCEDURE EXCEPT STERILIZATION
                   AND/OR D AND C */
    '377' = '1' /* POSTPARTUM AND POSTABORTION DIAGNOSES WITH OR PROCEDURE */
    '378' = '1' /* ECTOPIC PREGNANCY */
    '379' = '1' /* THREATENED ABORTION */
    '380' = '1' /* ABORTION WITHOUT D AND C */
    '381' = '1' /* ABORTION WITH D AND C, ASPIRATION CURETTAGE OR HYTEROTOMY
*/
    '382' = '1' /* FALSE LABOR */
    '383' = '1' /* OTHER ANTEPARTUM DIAGNOSES WITH MEDICAL COMPLICATIONS */
    '384' = '1' /* OTHER ANTEPARTUM DIAGNOSES WITHOUT MEDICAL COMPLICATIONS */
    '386' = '1' /* EXTREME IMMATURITY OR RESPIRATORY DISTRESS SYNDROME OF
NEONATE */
```

```
    '387' = '1' /* PREMATURITY WITH MAJOR PROBLEMS */
    '388' = '1' /* PREMATURITY WITHOUT MAJOR PROBLEMS */
    '390' = '1' /* NEONATE WITH OTHER SIGNIFICANT PROBLEMS */
    '391' = '1' /* NORMAL NEWBORN */
    '393' = '1' /* SPLENECTOMY, AGE 0-17 */
    '396' = '1' /* RED BLOOD CELL DISORDERS, AGE 0-17 */
    '421' = '1' /* VIRAL ILLNESS, AGE GREATER THAN 17 */
    '422' = '1' /* VIRAL ILLNESS AND FEVER OF UNKNOWN ORIGIN, AGE 0-17 */
    '425' = '1' /* ACUTE ADJUSTMENT REACTIONS AND DISTURBANCES OF PSYCHOSOCIAL
                   DYSFUNCTION */
    '426' = '1' /* DEPRESSIVE NEUROSES */
    '427' = '1' /* NEUROSIES EXCEPT DEPRESSIVE */
    '428' = '1' /* DISORDERS OF PERSONALITY AND IMPULSE CONTROL */
    '431' = '1' /* CHILDHOOD MENTAL DISORDERS */
    '432' = '1' /* OTHER MENTAL DISORDER DIAGNOSES */
    '434' = '1' /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                   SYMPTOMATIC TREATMENT WITH CC */
    '435' = '1' /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                   SYMPTOMATIC TREATMENT WITHOUT CC */
    '436' = '1' /* ALCOHOL/DRUG DEPENDENCE WITH REHABILITATION THERAPY */
    '439' = '1' /* SKIN GRAFTS FOR INJURIES */
    '441' = '1' /* HAND PROCEDURES FOR INJURIES */
    '446' = '1' /* TRAUMATIC INJURY, AGE 0-17 */
    '448' = '1' /* ALLERGIC REACTIONS, AGE 0-17 */
    '451' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE 0-17 */
    '491' = '1' /* MAJOR JOINT AND LIMB REATTACHMENT PROCEUDRES OF UPPER
EXTREMITY *
    /
    '499' = '1' /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITH CC */
    '500' = '1' /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITHOUT CC */
    OTHER = '0' ;

    VALUE $LOWM2DR
    '015' = 'M'  /* TRANSIENT ISCHEMIC ATTACK AND PRECEREBRAL OCCLUSIONS */
    '021' = 'M'  /* VIRAL MENINGITIS */
    '026' = 'PM' /* SEIZURE AND HEADACHE, AGE 0-17 */
    '030' = 'M'  /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE 0-
17 */
```

PSFMTS -- CONTINUED

```
    '031' = 'M'  /* CONCUSSION, AGE GREATER THAN 17 WITH CC */
    '032' = 'M'  /* CONCUSSION, AGE GREATER THAN 17 WITHOUT CC */
    '033' = 'PM' /* CONCUSSION, AGE 0-17 */
    '036' = 'S'  /* RETINAL PROCEDURES */
    '037' = 'S'  /* ORBITAL PROCEDURES */
    '042' = 'S'  /* INTRAOCULAR PROCEDURES */
    '044' = 'M'  /* ACUTE MAJOR EYE INFECTIONS */
    '045' = 'M'  /* NEUROLOGICAL EYE DISORDERS */
    '050' = 'S'  /* SIALOADENECTOMY */
    '052' = 'S'  /* CLEFT LIP AND PALATE REPAIR */
    '053' = 'S'  /* SINUS AND MASTOID PROCEDURES, AGE GREATER THAN 17 */
    '055' = 'S'  /* MISCELLANEOUS EAR, NOSE, MOUTH AND THROAT PROCEDURES */
    '057' = 'S'    /* TONSILLECTOMY AND ADENOIDECTOMY PROCEDURES EXCEPT
TONSILLECTOMY
                   AND/OR ADENOIDECTOMY ONLY, AGE GREATER THAN 17 */
    '060' = 'PS' /* TONSILLECTOMY AND/OR ADENOIDECTOMY ONLY, AGE 0-17 */
    '062' = 'PS' /* MYRINGOTOMY WITH TUBE INSERTION, AGE 0-17 */
    '063' = 'S'  /* OTHER EAR, NOSE, MOUTH AND THROAT OR PROCEDURES */
```

```
    '065' = 'M'  /* DYSEQUILIBRIUM */
    '068' = 'M'  /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITH CC */
    '070' = 'PM' /* OTITIS MEDIA AND URI, AGE 0-17 */
    '071' = 'M'  /* LARYNGOTRACHEITIS */
    '074' = 'PM' /* OTHER EAR, NOSE, MOUTH AND THROAT DIAGNOSES, AGE 0-17 */
    '091' = 'PM' /* SIMPLE PNEUMONIA AND PLEURISY, AGE 0-17 */
    '096' = 'M'  /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITH CC */
    '097' = 'M'  /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITHOUT CC */
    '098' = 'PM' /* BRONCHITIS AND ASTHMA, AGE 0-17 */
    '125' = 'M'  /* CIRCULATORY DISORDERS EXCEPT ACUTE MYOCARDIAL INFARCTION
WITH
                    CARDIAC CATHETERIZATION WITHOUT COMPLEX DIAGNOSIS */
    '134' = 'M'  /* HYPERTENSION */
    '140' = 'M'  /* ANGINA PECTORIS */
    '141' = 'M'  /* SYNCOPE AND COLLAPSE WITH CC */
    '142' = 'M'  /* SYNCOPE AND COLLAPSE WITHOUT CC */
    '143' = 'M'  /* CHEST PAIN */
    '156' = 'PS' /* STOMACH, ESOPHAGEAL AND DUODENAL PROCEDURES, AGE 0-17 */
    '163' = 'PS' /* HERNIA PROCEDURES, AGE 0-17 */
    '166' = 'S'  /* APPENDECTOMY WITHOUT COMPLICATED PRINCIPAL DIAGNOSIS WITH
CC */
    '167' = 'S'   /* APPENDECTOMY WITHOUT COMPLICATED PRINCIPAL DIAGNOSIS
WITHOUT CC
    */
    '184' = 'PM' /* ESOPHAGITIS, GASTROENTERITIS AND MISCELLANEOUS DIGESTIVE
                    DISORDERS, AGE 0-17 */
    '190' = 'PM' /* OTHER DIGESTIVE SYSTEM DIAGNOSES, AGE 0-17 */
    '212' = 'PS' /* HIP AND FEMUR PROCEDURES EXCEPT MAJOR JOINT PROCEDURES,
AGE
                    0-17 */
    '218' = 'S'  /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT
AND
                    FEMUR, AGE GREATER THAN 17 WITH CC */
    '219' = 'S'  /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT
AND
                    FEMUR, AGE GREATER THAN 17 WITHOUT CC */
    '220' = 'PS' /* LOWER EXTREMITY AND HUMEROUS PROCEDURES EXCEPT HIP, FOOT
AND
                    FEMUR, AGE 0-17 */
    '223' = 'S'  /* MAJOR SHOULDER, ELBOW PROCEDURES OR OTHER UPPER EXTREMITY
                    PROCEDURES WITH CC */
```

PSFMTS -- CONTINUED

```
    '224' = 'S'  /* SHOULDER, ELBOW OR FOREARM PROCEDURES EXCEPT MAJOR JOINT
                    PROCEDURES WITHOUT CC */
    '225' = 'S'  /* FOOT PROCEDURES */
    '228' = 'S'  /* MAJOR THUMB OR JOINT PROCEDURES OR OTHER HAND OR WRIST
                    PROCEDURES WITH CC */
    '229' = 'S'  /* HAND OR WRIST PROCEDURES EXCEPT MAJOR JOINT PROCEDURES
                    WITHOUT CC */
    '232' = 'S'  /* ARTHROSCOPY */
    '237' = 'M'  /* SPRAINS, STRAINS AND DISLOCATIONS OF HIP, PELVIS AND THIGH
*/
    '243' = 'M'  /* MEDICAL BACK PROBLEMS */
    '246' = 'M'  /* NONSPECIFIC ARTHROPATHIES */
    '252' = 'PM' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF FOREARM,
                    HAND AND FOOT, AGE 0-17 */
    '255' = 'PM' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF UPPER ARM
```

```
                    AND LOWER LEG EXCEPT FOOT, AGE 0-17 */
    '257' = 'S'  /* TOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '258' = 'S'  /* TOTAL MASTECTOMY FOR MALIGNNACY WITHOUT CC */
    '261' = 'S'  /* BREAST PROCEDURE FOR NONMALIGNANCY EXCEPT BIOPSY AND LOCAL
                    EXCISION */
    '262' = 'S'  /* BREAST BIOPSY AND LOCAL EXCISION OF NONMALIGNANCY */
    '267' = 'S'  /* PERIANAL AND PILONICAL PROCEDURES */
    '279' = 'PM' /* CELLULITIS, AGE 0-17 */
    '282' = 'PM' /* TRAUMA TO SKIN, SUBCUTANEOUS TISSUE AND BREAST, AGE 0-17
*/
    '289' = 'S'  /* PARATHYROID PROCEDURES */
    '290' = 'S'  /* THYROID PROCEDURES */
    '293' = 'S'  /* OTHER ENDOCRINE, NUTRITIONAL AND METABOLIC OR PROCEDURES
                    WITHOUT CC */
    '295' = 'M'  /* DIABETES, AGE 0-35 */
    '298' = 'PM' /* NUTRITIONAL AND MISCELLANEOUS METABOLIC DISORDERS, AGE
                    GREATER THAN 17 WITHOUT CC */
    '317' = 'M'  /* ADMISSION FOR RENAL DIALYSIS */
    '322' = 'PM' /* KIDNEY AND URINARY TRACT INFECTION, AGE 0-17 */
    '323' = 'M'  /* URINARY STONES WITH CC AND/OR ESW LITHOTRIPSY */
    '324' = 'M'  /* URINARY STONES WITHOUT CC */
    '333' = 'PM' /* OTHER KIDNEY AND URINARY TRACT DIAGNOSES, AGE 0-17 */
    '334' = 'S'  /* MAJOR MALE PELVIC PROCEDURES WITH CC */
    '335' = 'S'  /* MAJOR MALE PELVIC PROCEDURES WITHOUT CC */
    '336' = 'S'  /* TRANSURETHRAL PROSTATECTOMY WITH CC */
    '337' = 'S'  /* TRANSURETHRAL PROSTATECTOMY WITHOUT CC */
    '351' = 'M'  /* STERILIZATION, MALE */
    '356' = 'S'  /* FEMALE REPRODUCTION SYSTEM RECONCSTRUCTIVE PROCEDURES */
    '358' = 'S'  /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITH CC */
    '359' = 'S'  /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITHOUT CC
*/
    '360' = 'S'  /* VAGINA, CERVIX AND VULVA PROCEDURES */
    '361' = 'S'  /* LAPAROSCOPY AND INCISIONAL TUBAL INTERRUPTION */
    '362' = 'S'  /* ENDOSCOPIC TUBAL INTERRUPTION */
    '364' = 'S'  /* D AND C, CONIZATION EXCEPT FOR MALIGNANCY */
    '369' = 'M'  /* MENSTRUAL AND OTHER FEMALE REPRODUCTIVE SYSTEM DISORDERS
*/
    '370' = 'O'  /* CESAREAN SCTION WITH CC */
    '371' = 'O'  /* CESAREAN SECTION WITHOUT CC */
    '372' = 'O'  /* VAGINAL DELIVERY WITH COMPLICATING DIAGNOSES */
    '373' = 'O'  /* VAGINAL DELIVERY WITHOUT COMPLICATING DIAGNOSES */
    '374' = 'O'  /* VAGINAL DELIVERY WITH STERILIZATION AND/OR D AND C */
    '375' = 'O'  /* VAGINAL DELIVERY WITH OR PROCEDURE EXCEPT STERILIZATION
                    AND OR D AND C */
```

PSFMTS -- CONTINUED

```
    '377' = 'O'  /* POSTPARTUM AND POSTABORTION DIAGNOSES WITH OR PROCEDURE */
    '378' = 'O'  /* ECTOPIC PREGNANCY */
    '379' = 'O'  /* THREATENED ABORTION */
    '380' = 'O'  /* ABORTION WITHOUT D AND C */
    '381' = 'O'  /* ABORTION WITH D AND C, ASPIRATION CURETTAGE OR HYTEROTOMY
*/
    '382' = 'O'  /* FALSE LABOR */
    '383' = 'O'  /* OTHER ANTEPARTUM DIAGNOSES WITH MEDICAL COMPLICATIONS */
    '384' = 'O'  /* OTHER ANTEPARTUM DIAGNOSES WITHOUT MEDICAL COMPLICATIONS
*/
    '386' = 'N'  /* EXTREME IMMATURITY OR RESPIRATORY DISTRESS SYNDROME OF
NEONATE *
```

```
    /
    '387' = 'N'  /* PREMATURITY WITH MAJOR PROBLEMS */
    '388' = 'N'  /* PREMATURITY WITHOUT MAJOR PROBLEMS */
    '390' = 'N'  /* NEONATE WITH OTHER SIGNIFICANT PROBLEMS */
    '391' = 'N'  /* NORMAL NEWBORN */
    '393' = 'PS' /* SPLENECTOMY, AGE 0-17 */
    '396' = 'PM' /* RED BLOOD CELL DISORDERS, AGE 0-17 */
    '421' = 'M'  /* VIRAL ILLNESS, AGE GREATER THAN 17 */
    '422' = 'PM' /* VIRAL ILLNESS AND FEVER OF UNKNOWN ORIGIN, AGE 0-17 */
    '425'  =  'Y'    /*  ACUTE  ADJUSTMENT  REACTIONS  AND  DISTURBANCES  OF
PSYCHOSOCIAL
                   DYSFUNCTION */
    '426' = 'Y'  /* DEPRESSIVE NEUROSES */
    '427' = 'Y'  /* NEUROSIES EXCEPT DEPRESSIVE */
    '428' = 'Y'  /* DISORDERS OF PERSONALITY AND IMPULSE CONTROL */
    '431' = 'Y'  /* CHILDHOOD MENTAL DISORDERS */
    '432' = 'Y'  /* OTHER MENTAL DISORDER DIAGNOSES */
    '434' = 'Y'  /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                    SYMPTOMATIC TREATMENT WITH CC */
    '435' = 'Y'  /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                    SYMPTOMATIC TREATMENT WITHOUT CC */
    '436' = 'Y'  /* ALCOHOL/DRUG DEPENDENCE WITH REHABILITATION THERAPY */
    '439' = 'S'  /* SKIN GRAFTS FOR INJURIES */
    '441' = 'S'  /* HAND PROCEDURES FOR INJURIES */
    '446' = 'PM' /* TRAUMATIC INJURY, AGE 0-17 */
    '448' = 'PM' /* ALLERGIC REACTIONS, AGE 0-17 */
    '451' = 'PM' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE 0-17 */
    '491'  =  'S'   /*  MAJOR  JOINT  AND  LIMB  REATTACHMENT  PROCEUDRES  OF  UPPER
EXTREMITY
    */
    '499' = 'S'  /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITH CC */
    '500' = 'S'  /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITHOUT CC
*/
    OTHER = '0' ;

    /* Lung or pleural biopsy */
    /* ICD-9-CM Procedure codes: */

    VALUE $LUNGBIP
    '3326' = '1' /* CLOSED [PERCUTANEOUS] [NEEDLE] BIOPSY OF LUNG */
    '3328' = '1' /* OPEN BIOPSY OF LUNG */
    '3424' = '1' /* PLEURAL BIOPSY */
    OTHER  = '0' ;

    /* Lymphoid malignancy */
    /* ICD-9-CM diagnosis codes includes 4th and 5th digits: */
```

PSFMTS -- CONTINUED

```
    VALUE $LYMPHID
    '200'   = '1' /* LYMPHOSARCOMA AND RETICULOSARCOMA */
    '2000'  = '1' /* RETICULOSARCOMA */
    '20000' = '1' /* RETCLSRC UNSP XTRNDL ORG */
    '20001' = '1' /* RETICULOSARCOMA HEAD */
    '20002' = '1' /* RETICULOSARCOMA THORAX */
    '20003' = '1' /* RETICULOSARCOMA ABDOM */
    '20004' = '1' /* RETICULOSARCOMA AXILLA */
    '20005' = '1' /* RETICULOSARCOMA INGUIN */
```

```
'20006' = '1' /* RETICULOSARCOMA PELVIC */
'20007' = '1' /* RETICULOSARCOMA SPLEEN */
'20008' = '1' /* RETICULOSARCOMA MULT */
'2001'  = '1' /* LYMPHOSARCOMA */
'20010' = '1' /* LYMPHSRC UNSP XTRNDL ORG */
'20011' = '1' /* LYMPHOSARCOMA HEAD */
'20012' = '1' /* LYMPHOSARCOMA THORAX */
'20013' = '1' /* LYMPHOSARCOMA ABDOM */
'20014' = '1' /* LYMPHOSARCOMA AXILLA */
'20015' = '1' /* LYMPHOSARCOMA INGUIN */
'20016' = '1' /* LYMPHOSARCOMA PELVIC */
'20017' = '1' /* LYMPHOSARCOMA SPLEEN */
'20018' = '1' /* LYMPHOSARCOMA MULT */
'2002'  = '1' /* BURKITTS TUMOR/LYMPHOMA */
'20020' = '1' /* BRKT TMR UNSP XTRNDL ORG */
'20021' = '1' /* BURKITTS TUMOR HEAD */
'20022' = '1' /* BURKITTS TUMOR THORAX */
'20023' = '1' /* BURKITTS TUMOR ABDOM */
'20024' = '1' /* BURKITTS TUMOR AXILLA */
'20025' = '1' /* BURKITTS TUMOR INGUIN */
'20026' = '1' /* BURKITTS TUMOR PELVIC */
'20027' = '1' /* BURKITTS TUMOR SPLEEN */
'20028' = '1' /* BURKITTS TUMOR MULT */
'2008'  = '1' /* MIXED LYMPHOSARCOMA */
'20080' = '1' /* OTH VARN UNSP XTRNDL ORG */
'20081' = '1' /* MIXED LYMPHOSARC HEAD */
'20082' = '1' /* MIXED LYMPHOSARC THORAX */
'20083' = '1' /* MIXED LYMPHOSARC ABDOM */
'20084' = '1' /* MIXED LYMPHOSARC AXILLA */
'20085' = '1' /* MIXED LYMPHOSARC INGUIN */
'20086' = '1' /* MIXED LYMPHOSARC PELVIC */
'20087' = '1' /* MIXED LYMPHOSARC SPLEEN */
'20088' = '1' /* MIXED LYMPHOSARC MULT */
'201'   = '1' /* HODGKINS DISEASE */
'2010'  = '1' /* HODGKINS PARAGRANULOMA */
'20100' = '1' /* HDGK PRG UNSP XTRNDL ORG */
'20101' = '1' /* HODGKINS PARAGRAN HEAD */
'20102' = '1' /* HODGKINS PARAGRAN THORAX */
'20103' = '1' /* HODGKINS PARAGRAN ABDOM */
'20104' = '1' /* HODGKINS PARAGRAN AXILLA */
'20105' = '1' /* HODGKINS PARAGRAN INGUIN */
'20106' = '1' /* HODGKINS PARAGRAN PELVIC */
'20107' = '1' /* HODGKINS PARAGRAN SPLEEN */
'20108' = '1' /* HODGKINS PARAGRAN MULT */
'2011'  = '1' /* HODGKINS GRANULOMA */
'20110' = '1' /* HDGK GRN UNSP XTRNDL ORG */
'20111' = '1' /* HODGKINS GRANULOM HEAD */
'20112' = '1' /* HODGKINS GRANULOM THORAX */
'20113' = '1' /* HODGKINS GRANULOM ABDOM */
'20114' = '1' /* HODGKINS GRANULOM AXILLA */
```

PSFMTS -- CONTINUED

```
'20115' = '1' /* HODGKINS GRANULOM INGUIN */
'20116' = '1' /* HODGKINS GRANULOM PELVIC */
'20117' = '1' /* HODGKINS GRANULOM SPLEEN */
'20118' = '1' /* HODGKINS GRANULOM MULT */
'2012'  = '1' /* HODGKINS SARCOMA */
'20120' = '1' /* HDGK SRC UNSP XTRNDL ORG */
```

```
'20121' = '1' /* HODGKINS SARCOMA HEAD */
'20122' = '1' /* HODGKINS SARCOMA THORAX */
'20123' = '1' /* HODGKINS SARCOMA ABDOM */
'20124' = '1' /* HODGKINS SARCOMA AXILLA */
'20125' = '1' /* HODGKINS SARCOMA INGUIN */
'20126' = '1' /* HODGKINS SARCOMA PELVIC */
'20127' = '1' /* HODGKINS SARCOMA SPLEEN */
'20128' = '1' /* HODGKINS SARCOMA MULT */
'2014'  = '1' /* HODGKINS LYMPH-HISTIOCYT */
'20140' = '1' /* LYM-HST UNSP XTRNDL ORGN */
'20141' = '1' /* HODG LYMPH-HISTIO HEAD */
'20142' = '1' /* HODG LYMPH-HISTIO THORAX */
'20143' = '1' /* HODG LYMPH-HISTIO ABDOM */
'20144' = '1' /* HODG LYMPH-HISTIO AXILLA */
'20145' = '1' /* HODG LYMPH-HISTIO INGUIN */
'20146' = '1' /* HODG LYMPH-HISTIO PELVIC */
'20147' = '1' /* HODG LYMPH-HISTIO SPLEEN */
'20148' = '1' /* HODG LYMPH-HISTIO MULT */
'2015'  = '1' /* HODGKINS NODULAR SCLEROS */
'20150' = '1' /* NDR SCLR UNSP XTRNDL ORG */
'20151' = '1' /* HODG NODUL SCLERO HEAD */
'20152' = '1' /* HODG NODUL SCLERO THORAX */
'20153' = '1' /* HODG NODUL SCLERO ABDOM */
'20154' = '1' /* HODG NODUL SCLERO AXILLA */
'20155' = '1' /* HODG NODUL SCLERO INGUIN */
'20156' = '1' /* HODG NODUL SCLERO PELVIC */
'20157' = '1' /* HODG NODUL SCLERO SPLEEN */
'20158' = '1' /* HODG NODUL SCLERO MULT */
'2016'  = '1' /* HODGKINS MIX CELLULARITY */
'20160' = '1' /* MXD CELR UNSP XTRNDL ORG */
'20161' = '1' /* HODGKINS MIX CELL HEAD */
'20162' = '1' /* HODGKINS MIX CELL THORAX */
'20163' = '1' /* HODGKINS MIX CELL ABDOM */
'20164' = '1' /* HODGKINS MIX CELL AXILLA */
'20165' = '1' /* HODGKINS MIX CELL INGUIN */
'20166' = '1' /* HODGKINS MIX CELL PELVIC */
'20167' = '1' /* HODGKINS MIX CELL SPLEEN */
'20168' = '1' /* HODGKINS MIX CELL MULT */
'2017'  = '1' /* HODG LYMPHOCYTIC DEPLET */
'20170' = '1' /* LYM DPLT UNSP XTRNDL ORG */
'20171' = '1' /* HODG LYMPH DEPLET HEAD */
'20172' = '1' /* HODG LYMPH DEPLET THORAX */
'20173' = '1' /* HODG LYMPH DEPLET ABDOM */
'20174' = '1' /* HODG LYMPH DEPLET AXILLA */
'20175' = '1' /* HODG LYMPH DEPLET INGUIN */
'20176' = '1' /* HODG LYMPH DEPLET PELVIC */
'20177' = '1' /* HODG LYMPH DEPLET SPLEEN */
'20178' = '1' /* HODG LYMPH DEPLET MULT */
'2019'  = '1' /* HODGKINS DISEASE NOS */
'20190' = '1' /* HDGK DIS UNSP XTRNDL ORG */
'20191' = '1' /* HODGKINS DIS NOS HEAD */
'20192' = '1' /* HODGKINS DIS NOS THORAX */
'20193' = '1' /* HODGKINS DIS NOS ABDOM */
```

PSFMTS -- CONTINUED

```
'20194' = '1' /* HODGKINS DIS NOS AXILLA */
'20195' = '1' /* HODGKINS DIS NOS INGUIN */
'20196' = '1' /* HODGKINS DIS NOS PELVIC */
```

```
    '20197' = '1' /* HODGKINS DIS NOS SPLEEN */
    '20198' = '1' /* HODGKINS DIS NOS MULT */
    '202'      '1' /* OTHER MALIGNANT NEOPLASMS OF LYMPHOID AND HISTIOCYTIC
TISSUE */
    '2020'  = '1' /* NODULAR LYMPHOMA */
    '20200' = '1' /* NDLR LYM UNSP XTRNDL ORG */
    '20201' = '1' /* NODULAR LYMPHOMA HEAD */
    '20202' = '1' /* NODULAR LYMPHOMA THORAX */
    '20203' = '1' /* NODULAR LYMPHOMA ABDOM */
    '20204' = '1' /* NODULAR LYMPHOMA AXILLA */
    '20205' = '1' /* NODULAR LYMPHOMA INGUIN */
    '20206' = '1' /* NODULAR LYMPHOMA PELVIC */
    '20207' = '1' /* NODULAR LYMPHOMA SPLEEN */
    '20208' = '1' /* NODULAR LYMPHOMA MULT */
    '2021'  = '1' /* MYCOSIS FUNGOIDES */
    '20210' = '1' /* MYCS FNG UNSP XTRNDL ORG */
    '20211' = '1' /* MYCOSIS FUNGOIDES HEAD */
    '20212' = '1' /* MYCOSIS FUNGOIDES THORAX */
    '20213' = '1' /* MYCOSIS FUNGOIDES ABDOM */
    '20214' = '1' /* MYCOSIS FUNGOIDES AXILLA */
    '20215' = '1' /* MYCOSIS FUNGOIDES INGUIN */
    '20216' = '1' /* MYCOSIS FUNGOIDES PELVIC */
    '20217' = '1' /* MYCOSIS FUNGOIDES SPLEEN */
    '20218' = '1' /* MYCOSIS FUNGOIDES MULT */
    '2022'  = '1' /* SEZARYS DISEASE */
    '20220' = '1' /* SZRY DIS UNSP XTRNDL ORG */
    '20221' = '1' /* SEZARYS DISEASE HEAD */
    '20222' = '1' /* SEZARYS DISEASE THORAX */
    '20223' = '1' /* SEZARYS DISEASE ABDOM */
    '20224' = '1' /* SEZARYS DISEASE AXILLA */
    '20225' = '1' /* SEZARYS DISEASE INGUIN */
    '20226' = '1' /* SEZARYS DISEASE PELVIC */
    '20227' = '1' /* SEZARYS DISEASE SPLEEN */
    '20228' = '1' /* SEZARYS DISEASE MULT */
    '2023'  = '1' /* MALIGNANT HISTIOCYTOSIS */
    '20230' = '1' /* MLG HIST UNSP XTRNDL ORG */
    '20231' = '1' /* MAL HISTIOCYTOSIS HEAD */
    '20232' = '1' /* MAL HISTIOCYTOSIS THORAX */
    '20233' = '1' /* MAL HISTIOCYTOSIS ABDOM */
    '20234' = '1' /* MAL HISTIOCYTOSIS AXILLA */
    '20235' = '1' /* MAL HISTIOCYTOSIS INGUIN */
    '20236' = '1' /* MAL HISTIOCYTOSIS PELVIC */
    '20237' = '1' /* MAL HISTIOCYTOSIS SPLEEN */
    '20238' = '1' /* MAL HISTIOCYTOSIS MULT */
    '2024'  = '1' /* LEUKEM RETICULOENDOTHEL */
    '20240' = '1' /* LK RTCTL UNSP XTRNDL ORG */
    '20241' = '1' /* HAIRY-CELL LEUKEM HEAD */
    '20242' = '1' /* HAIRY-CELL LEUKEM THORAX */
    '20243' = '1' /* HAIRY-CELL LEUKEM ABDOM */
    '20244' = '1' /* HAIRY-CELL LEUKEM AXILLA */
    '20245' = '1' /* HAIRY-CELL LEUKEM INGUIN */
    '20246' = '1' /* HAIRY-CELL LEUKEM PELVIC */
    '20247' = '1' /* HAIRY-CELL LEUKEM SPLEEN */
    '20248' = '1' /* HAIRY-CELL LEUKEM MULT */
    '2025'  = '1' /* LETTERER-SIWE DISEASE */
    '20250' = '1' /* LTR-SIWE UNSP XTRNDL ORG */
```

PSFMTS -- CONTINUED

```
'20251' = '1' /* LETTERER-SIWE DIS HEAD */
'20252' = '1' /* LETTERER-SIWE DIS THORAX */
'20253' = '1' /* LETTERER-SIWE DIS ABDOM */
'20254' = '1' /* LETTERER-SIWE DIS AXILLA */
'20255' = '1' /* LETTERER-SIWE DIS INGUIN */
'20256' = '1' /* LETTERER-SIWE DIS PELVIC */
'20257' = '1' /* LETTERER-SIWE DIS SPLEEN */
'20258' = '1' /* LETTERER-SIWE DIS MULT */
'2026'  = '1' /* MALIG MAST CELL TUMORS */
'20260' = '1' /* MLG MAST UNSP XTRNDL ORG */
'20261' = '1' /* MAL MASTOCYTOSIS HEAD */
'20262' = '1' /* MAL MASTOCYTOSIS THORAX */
'20263' = '1' /* MAL MASTOCYTOSIS ABDOM */
'20264' = '1' /* MAL MASTOCYTOSIS AXILLA */
'20265' = '1' /* MAL MASTOCYTOSIS INGUIN */
'20266' = '1' /* MAL MASTOCYTOSIS PELVIC */
'20267' = '1' /* MAL MASTOCYTOSIS SPLEEN */
'20268' = '1' /* MAL MASTOCYTOSIS MULT */
'2028'  = '1' /* LYMPHOMAS NEC */
'20280' = '1' /* OTH LYMP UNSP XTRNDL ORG */
'20281' = '1' /* LYMPHOMAS NEC HEAD */
'20282' = '1' /* LYMPHOMAS NEC THORAX */
'20283' = '1' /* LYMPHOMAS NEC ABDOM */
'20284' = '1' /* LYMPHOMAS NEC AXILLA */
'20285' = '1' /* LYMPHOMAS NEC INGUIN */
'20286' = '1' /* LYMPHOMAS NEC PELVIC */
'20287' = '1' /* LYMPHOMAS NEC SPLEEN */
'20288' = '1' /* LYMPHOMAS NEC MULT */
'2029'  = '1' /* MAL NEO LYM/HIST TIS NEC */
'20290' = '1' /* UNSP LYM UNSP XTRNDL ORG */
'20291' = '1' /* LYMPHOID MAL NEC HEAD */
'20292' = '1' /* LYMPHOID MAL NEC THORAX */
'20293' = '1' /* LYMPHOID MAL NEC ABDOM */
'20294' = '1' /* LYMPHOID MAL NEC AXILLA */
'20295' = '1' /* LYMPHOID MAL NEC INGUIN */
'20296' = '1' /* LYMPHOID MAL NEC PELVIC */
'20297' = '1' /* LYMPHOID MAL NEC SPLEEN */
'20298' = '1' /* LYMPHOID MAL NEC MULT */
'203'   = '1' /* MULTIPLE MYELOMA AN DIMMUNOPROLIFERATIVE NEOPLASMS */
'2030'  = '1' /* MULTIPLE MYELOMA */
'20300' = '1' /* MULT MYELM W/O REMISSION */
'20301' = '1' /* MULT MYELM W REMISSION */
'2031'  = '1' /* PLASMA CELL LEUKEMIA */
'20310' = '1' /* PLSM CELL LEUK W/O RMSON */
'20311' = '1' /* PLSM CELL LEUK W RMSON */
'2038'  = '1' /* IMMUNOPROLIFERAT NEO NEC */
'20380' = '1' /* OTH IMNPRFL NPL W/O RMSN */
'20381' = '1' /* OTH IMNPRFL NPL W RMSN */
'204'   = '1' /* LYMPHOID LEUKEMIA */
'2040'  = '1' /* ACUTE LYMPHOID LEUKEMIA */
'20400' = '1' /* ACT LYM LEUK W/O RMSION */
'20401' = '1' /* ACT LYM LEUK W RMSION */
'2041'  = '1' /* CHR LYMPHOID LEUKEMIA */
'20410' = '1' /* CHR LYM LEUK W/O RMSION */
'20411' = '1' /* CHR LYM LEUK W RMSION */
'2042'  = '1' /* SUBAC LYMPHOID LEUKEMIA */
'20420' = '1' /* SBAC LYM LEUK W/O RMSION */
'20421' = '1' /* SBAC LYM LEUK W RMSION */
```

```
'2048'  = '1' /* LYMPHOID LEUKEMIA NEC */
```
PSFMTS -- CONTINUED

```
'20480' = '1' /* OTH LYM LEUK W/O RMSION */
'20481' = '1' /* OTH LYM LEUK W RMSION */
'2049'  = '1' /* LYMPHOID LEUKEMIA NOS */
'20490' = '1' /* UNS LYM LEUK W/O RMSION */
'20491' = '1' /* UNS LYM LEUK W RMSION */
'205'   = '1' /* MYELOID LEUKEMIA */
'2050'  = '1' /* ACUTE MYELOID LEUKEMIA */
'20500' = '1' /* ACT MYL LEUK W/O RMSION */
'20501' = '1' /* ACT MYL LEUK W RMSION */
'2051'  = '1' /* CHRONIC MYELOID LEUKEMIA */
'20510' = '1' /* CHR MYL LEUK W/O RMSION */
'20511' = '1' /* CHR MYL LEUK W RMSION */
'2052'  = '1' /* SUBACUT MYELOID LEUKEMIA */
'20520' = '1' /* SBAC MYL LEUK W/O RMSION */
'20521' = '1' /* SBAC MYL LEUK W RMSION */
'2053'  = '1' /* MYELOID SARCOMA */
'20530' = '1' /* MYL SRCOMA W/O RMSION */
'20531' = '1' /* MYL SRCOMA W RMSION */
'2058'  = '1' /* MYELOID LEUKEMIA NEC */
'20580' = '1' /* OTH MYL LEUK W/O RMSION */
'20581' = '1' /* OTH MYL LEUK W RMSION */
'2059'  = '1' /* MYELOID LEUKEMIA NOS */
'20590' = '1' /* UNS MYL LEUK W/O RMSION */
'20591' = '1' /* UNS MYL LEUK W RMSION */
'206'   = '1' /* MONOCYTIC LEUKEMIA */
'2060'  = '1' /* ACUTE MONOCYTIC LEUKEMIA */
'20600' = '1' /* ACT MONO LEUK W/O RMSION */
'20601' = '1' /* ACT MONO LEUK W RMSION */
'2061'  = '1' /* CHR MONOCYTIC LEUKEMIA */
'20610' = '1' /* CHR MONO LEUK W/O RMSION */
'20611' = '1' /* CHR MONO LEUK W RMSION */
'2062'  = '1' /* SUBAC MONOCYTIC LEUKEMIA */
'20620' = '1' /* SBAC MONO LEUK W/O RMSON */
'20621' = '1' /* SBAC MONO LEUK W RMSION */
'2068'  = '1' /* MONOCYTIC LEUKEMIA NEC */
'20680' = '1' /* OTH MONO LEUK W/O RMSION */
'20681' = '1' /* OTH MONO LEUK W RMSION */
'2069'  = '1' /* MONOCYTIC LEUKEMIA NOS */
'20690' = '1' /* UNS MONO LEUK W/O RMSION */
'20691' = '1' /* UNS MONO LEUK W RMSION */
'207'   = '1' /* OTHER SPECIFIED LEUKEMIA */
'2070'  = '1' /* ACUTE ERYTHREMIA */
'20700' = '1' /* ACT ERTH/ERYLK W/O RMSON */
'20701' = '1' /* ACT ERTH/ERYLK W RMSON */
'2071'  = '1' /* CHRONIC ERYTHREMIA */
'20710' = '1' /* CHR ERYTHRM W/O REMISION */
'20711' = '1' /* CHR ERYTHRM W REMISION */
'2072'  = '1' /* MEGAKARYOCYTIC LEUKEMIA */
'20720' = '1' /* MGKRYCYT LEUK W/O RMSION */
'20721' = '1' /* MGKRYCYT LEUK W RMSION */
'2078'  = '1' /* SPECIFIED LEUKEMIA NEC */
'20780' = '1' /* OTH SPF LEUK W/O REMSION */
'20781' = '1' /* OTH SPF LEUK W REMSION */
'208'   = '1' /* LEUKEMIA OF UNSPECIFIED CELL TYPE */
'2080'  = '1' /* ACT LEUK UNS CL W/O RMSN */
```

```
    '20800' = '1' /* ACT LEUK UNS CL W/O RMSN */
    '20801' = '1' /* ACT LEUK UNS CL W RMSON */
    '2081'  = '1' /* CHRONIC LEUKEMIA NOS */
    '20810' = '1' /* CHR LEUK UNS CL W/O RMSN */
```
PSFMTS -- CONTINUED
```
    '20811' = '1' /* CHR LEUK UNS CL W RMSON */
    '2082'  = '1' /* SUBACUTE LEUKEMIA NOS */
    '20820' = '1' /* SBAC LEUK UNS CL W/O RMS */
    '20821' = '1' /* SBAC LEUK UNS CL W RMSON */
    '2088'  = '1' /* LEUKEMIA-UNSPEC CELL NEC */
    '20880' = '1' /* OTH LEUK UNS CL W/O RMSN */
    '20881' = '1' /* OTH LEUK UNS CL W RMSON */
    '2089'  = '1' /* LEUKEMIA-UNSPEC CELL NOS */
    '20890' = '1' /* LEUKEMIA NOS W/O REMSION */
    '20891' = '1' /* LEUKEMIA NOS W REMISSION */
    OTHER   = '0' ;

    /* Medical */
    /* Diagnostic Related Groups (DRGs): */

    VALUE $MEDICDR
    '009' = '1' /* SPINAL DISORDERS AND INJURIES */
    '010' = '1' /* NERVOUS SYSTEM NEOPLASMS WITH CC */
    '011' = '1' /* NERVOUS SYSTEM NEOPLASMS WITH CC */
    '012' = '1' /* DEGENERATIVE NERVOUS SYSTEM DISORDERS */
    '013' = '1' /* MULTIPLE SCLEROSIS AND CEREBELLAR ATAXIA */
    '014'  =  '1'  /*  SPECIFIC  CEREBROVASCULAR  DISORDERS  EXCEPT  TRANSIENT
ISCHEMIC
                   ATTACK */
    '015' = '1' /* TRANSIENT ISCHEMIC ATTACK AND PRECEREBRAL OCCLUSIONS */
    '016' = '1' /* NONSPECIFIC CEREBROVASCULAR DISORDERS WITH CC */
    '017' = '1' /* NONSPECIFIC CEREBROVASCULAR DISORDERS WITHOUT CC */
    '018' = '1' /* CRANIAL AND PERIPHERAL NERVE DISORDERS WITH CC */
    '019' = '1' /* CRANIAL AND PERIPHERAL NERVE DISORDERS WITHOUT CC */
    '020' = '1' /* NERVOUS SYSTEM INFECTION EXCEPT VIRAL MENINGITIS */
    '021' = '1' /* VIRAL MENINGITIS */
    '022' = '1' /* HYPERTENSIVE ENCEPHALOPATHY */
    '023' = '1' /* NONTRAUMATIC STUPOR AND COMA */
    '024' = '1' /* SEIZURE AND HEADACHE, AGE GREATER THAN 17 WITH CC */
    '025' = '1' /* SEIZURE AND HEADACHE, AGE GREATER THAN 17 WITHOUT CC */
    '026' = '1' /* SEIZURE AND HEADACHE, AGE 0-17 */
    '027' = '1' /* TRAUMATIC STUPOR AND COMA, COMA GREATER THAN ONE HOUR */
    '028' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE
GREATER
                   THAN 19 WITH CC */
    '029' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE
GREATER
                   THAN 17 WITHOUT CC */
    '030' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE 0-
17 */
    '031' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITH CC */
    '032' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITHOUT CC */
    '033' = '1' /* CONCUSSION, AGE 0-17 */
    '034' = '1' /* OTHER DISORDERS OF NERVOUS SYSTEM WITH CC */
    '035' = '1' /* OTHER DISORDERS OF NERVOUS SYSTEM WITHOUT CC */
    '043' = '1' /* HYPHEMA */
    '044' = '1' /* ACUTE MAJOR EYE INFECTIONS */
```

```
    '045' = '1' /* NEUROLOGICAL EYE DISORDERS */
    '046' = '1' /* OTHER DISORDERS OF THE EYE, AGE GREATER THAN 17 WITH CC */
    '047' = '1' /* OTHER DISORDER OF THE EYE, AGE GREATER THAN 17 WITHOUT CC
*/
    '048' = '1' /* OTHER DISORDERS OF THE EYE, AGE 0-17 */
    '064' = '1' /* EAR, NOSE, MOUTH AND THROAT MALIGNANCY */
    '065' = '1' /* DISEQUILIBRIA */
```

PSFMTS -- CONTINUED

```
    '066' = '1' /* EPISTAXIS */
    '067' = '1' /* EPIGLOTTITIS */
    '068' = '1' /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITH CC */
    '069' = '1' /* OTITIS MEDIA AND URI, AGE GREATER THAN 17 WITHOUT CC */
    '070' = '1' /* OTITIS MEDIA AND URI, AGE 0-17 */
    '071' = '1' /* LARYNGOTRACHEITIS */
    '072' = '1' /* NASAL TRAUMA AND DEFORMITY */
    '073' = '1' /* OTHER EAR, NOSE, MOUTH AND THROAT DIAGNOSES, AGE GREATER
THAN 17
    */
    '074' = '1' /* OTHER EAR, NOSE, MOUTH AND THROAT DIAGNOSES, AGE 0-17 */
    '078' = '1' /* PULMONARY EMBOLISM */
    '079' = '1' /* RESPIRATORY INFECTIONS AND INFLAMMATIONS, AGE GREATER THAN
17
                   WITH CC */
    '080' = '1' /* RESPIRATORY INFECTIONS AND INFLAMMATIONS, AGE GREATER THAN
17
                   WITHOUT CC */
    '081' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE GREATER THAN 17 WITH CC
*/
    '082' = '1' /* RESPIRATORY NEOPLASMS */
    '083' = '1' /* MAJOR CHEST TRAUMA WITH CC */
    '084' = '1' /* MAJOR CHEST TRAUMA WITHOUT CC*/
    '085' = '1' /* PLEURAL EFFUSION WITH CC */
    '086' = '1' /* PLEURAL EFFUSION WITHOUT CC */
    '087' = '1' /* PULMONARY EDEMA AND RESPIRATORY FAILURE */
    '088' = '1' /* CHRONIC OBSTRUCTIVE PULMONARY DISEASE */
    '089' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE GREATER THAN 17 WITH CC
*/
    '090' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE GREATER THAN 17 WITHOUT
CC */
    '091' = '1' /* SIMPLE PNEUMONIA AND PLEURISY, AGE 0-17 */
    '092' = '1' /* INTERSTITIAL LUNG DISEASE WITH CC */
    '093' = '1' /* INTERSTITIAL LUNG DISEASE WITHOUT CC */
    '094' = '1' /* PNEUMOTHORAX WITH CC */
    '095' = '1' /* PNEUMOTHORAX WITHOUT CC */
    '096' = '1' /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITH CC */
    '097' = '1' /* BRONCHITIS AND ASTHMA, AGE GREATER THAN 17 WITHOUT CC */
    '098' = '1' /* BRONCHITIS AND ASTHMA, AGE 0-17 */
    '099' = '1' /* RESPIRATORY SIGNS AND SYMPTOMS WITH CC */
    '100' = '1' /* RESPIRATORY SIGNS AND SYMPTOMS WITHOUT CC */
    '101' = '1' /* OTHER RESPIRATORY SYSTEM DIAGNOSES WITH CC */
    '102' = '1' /* OTHER RESPIRATORY SYSTEM DIAGNOSES WITHOUT CC */
    '121' = '1' /* CIRCULATORY DISORDERS WITH ACUTE MYOCARDIAL INFARCTION AND
                   MAJOR COMPLICATION, DISCHARGED ALIVE */
    '122' = '1' /* CIRCULATORY DISORDERS WITH ACUTE MYOCARDIAL INFARCTION AND
                   MAJOR COMPLICATION, DISCHARGED ALIVE */
    '123' = '1' /* CIRCULATORY DISORDERS WITH ACUTE MYOCARDIAL INFARCTION,
                   EXPIRED */
```

```
    '124' = '1' /* CIRCULATORY DISORDERS EXCEPT ACUTE MYOCARDIAL INFARCTION
                   WITH CARDIAC CATHETERIZATION AND COMPLEX DIAGNOSIS */
    '125' = '1' /* CIRCULATORY DISORDERS EXCEPT ACUTE MYOCARDIAL INFARCTION
                   WITH CARDIAC CATHETERIZATION WITHOUT COMPLEX DIAGNOSIS */
    '126' = '1' /* ACUTE AND SUB ACUTE ENDOCARDITIS */
    '127' = '1' /* HEART FAILURE AND SHOCK */
    '128' = '1' /* DEEP VEIN THROMBOPHLEBITIS */
    '129' = '1' /* CARDIAC ARREST, UNEXPLAINED */
    '130' = '1' /* PERIPHERAL VASCULAR DISORDERS WITH CC */
    '131' = '1' /* PERIPHERAL VASCULAR DISORDERS WITHOUT CC */
```

PSFMTS -- CONTINUED

```
    '132' = '1' /* ATHEROSCLEROSIS WITH CC */
    '133' = '1' /* ATHEROSCLEROSIS WITHOUT CC */
    '134' = '1' /* HYPERTENSION */
    '135' = '1' /* CARDIAC CONGENITAL AND VALVULAR DISORDERS, AGE GREATER
                   THAN 17 WITH CC */
    '136' = '1' /* CARDIAC CONGENITAL AND VALVULAR DISORDERS, AGE GREATER
                   THAN 17 WITHOUT CC */
    '137' = '1' /* CARDIAC CONGENITAL AND VALVULAR DISORDERS, AGE GREATER
                   THAN 17 WITHOUT CC */
    '138' = '1' /* CARDIAC ARRHYTHMIA AND CONDUCTION DISORDERS WITH CC */
    '139' = '1' /* CARDIAC ARRHYTHMIA AND CONDUCTION DISORDERS WITHOUT CC */
    '140' = '1' /* ANGINA PECTORIS*/
    '141' = '1' /* SYNCOPE AND COLLAPSE WITH CC */
    '142' = '1' /* SYNCOPE AND COLLAPSE WITHOUT CC */
    '143' = '1' /* CHEST PAIN */
    '144' = '1' /* OTHER CIRCULATORY SYSTEM DIAGNOSES WITH CC*/
    '145' = '1' /* OTHER CIRCULATORY SYSTEM DIAGNOSES WITHOUT CC */
    '172' = '1' /* DIGESTIVE MALIGNANCY WITH CC */
    '173' = '1' /* DIGESTIVE MALIGNANCY WITHOUT CC */
    '174' = '1' /* GI HEMORRHAGE WITH CC*/
    '175' = '1' /* GI HEMORRHAGE WITHOUT CC */
    '176' = '1' /* COMPLICATED PEPTIC ULCER */
    '177' = '1' /* UNCOMPLICATED PEPTIC ULCER WITH CC */
    '178' = '1' /* UNCOMPLICATED PEPTIC ULCER WITHOUT CC*/
    '179' = '1' /* INFLAMMATORY BOWEL DISEASE */
    '180' = '1' /* GI OBSTRUCTION WITH CC */
    '181' = '1' /* GI OBSTRUCTION WITHOUT CC */
    '182' = '1' /* ESOPHAGITIS, GASTROENTERITIS AND MISCELLANEOUS DIGESTIVE
                   DISORDERS, AGE GREATER THAN 17 WITH CC */
    '183' = '1' /* ESOPHAGITIS, GASTROENTERITIS AND MISCELLANEOUS DIGESTIVE
                   DISORDERS, AGE GREATER THAN 17 WITHOUT CC */
    '184' = '1' /* ESOPHAGITIS, GASTROENTERITIS AND MISCELLANESOU DIGESTIVE
                   DISORDERS, AGE 0-17 */
    '185'  =  '1'  /*  DENTAL  AND  ORAL  DISEASES  EXCEPT  EXTRACTIONS  AND
RESTORATIONS,
                   AGE GREATER THAN 17 */
    '186'  =  '1'  /*  DENTAL  AND  ORAL  DISEASED  EXCEPT  EXTRACTIONS  AND
RESTORATIONS,
                   AGE 0-17 */
    '187' = '1' /* DENTAL EXTRACTIONS AND RESTORATIONS */
    '188' = '1' /* OTHER DIGESTIVE SYSTEM DIAGNOSES, AGE GREATER THAN 17
                   WITHOUT CC */
    '189' = '1' /* OTHER DIGESTIVE SYSTEM DIAGNOSES, AGE GREATER THAN 17
                   WITHOUT CC */
    '190' = '1' /* OTHER DIGESTIVE SYSTEM DIAGNOSES, AGE 0-17 */
    '202' = '1' /* CIRRHOSIS AND ALCOHOLIC HEPATITIS */
```

```
    '203' = '1' /* MALIGNANCY OF HEPATOBILIARY SYSTEM OR PANCREAS */
    '204' = '1' /* DISORDERS OF PANCREAS EXCEPT MALIGNANCY */
    '205'  =  '1'  /*  DISORDERS  OF  LIVER  EXCEPT  MALIGNANCY,  CIRRHOSIS  AND
ALCOHOLIC
                   HEPATITIS WITH CC */
    '206'  =  '1'  /*  DISORDERS  OF  LIVER  EXCEPT  MALIGNANCY,  CIRRHOSIS  AND
ALCOHOLIC
                   HEPATITIS WITHOUT CC */
    '207' = '1' /* DISORDERS OF THE BILIARY TRACT WITH CC */
    '208' = '1' /* DISORDERS OF THE BILIARY TRACT WITHOUT CC */
    '235' = '1' /* FRACTURES OF FEMUR */
    '236' = '1' /* FRACTURES OF HIP AND PELVIS */
```

PSFMTS -- CONTINUED

```
    '237' = '1' /* SPRAINS, STRAINS AND DISLOCATIONS OF HIP, PELVIS AND THIGH
*/
    '238' = '1' /* OSTEOMYELITIS */
    '239' = '1' /* PATHOLOGICAL FRACTURES AND MUSCULOSKELETAL AND CONNECTIVE
                   TISSUE MALIGNANCY */
    '240' = '1' /* CONNECTIVE TISSUE DISORDERS WITH CC */
    '241' = '1' /* CONNECTIVE TISSUE DISORDERS WITHOUT CC */
    '242' = '1' /* SEPTIC ARTHRITIS */
    '243' = '1' /* MEDICAL BACK PROBLEMS */
    '244' = '1' /* BONE DISEASES AND SPECIFIC ARTHROPATHIES WITH CC */
    '245' = '1' /* BONE DISEASES AND SPECIFIC ARTHROPATHIES WITHOUT CC */
    '246' = '1' /* NONSPECIFIC ARTHROPATHIES */
    '247' = '1' /* SIGNS AND SYMPTOMS OF MUSCULOSKELETAL SYSTEM AND CONNECTIVE
                   TISSUE */
    '248' = '1' /* TENDONITIS, MYOSITIS AND BURSITIS */
    '249' = '1' /* AFTERCARE, MUSCULOSKELETAL SYSTEM AND CONNECTIVE TISSUE */
    '250' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF FOREARM,
                   HAND AND FOOT, AGE GREATER THAN 17 WITH CC */
    '251' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF FOREARM,
                   HAND AND FOOT, AGE GREATER THAN 17 WITH CC */
    '252' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF FOREARM,
                   HAND AND FOOT, AGE 0-17 */
    '253' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF UPPER ARM
                   AND LOWER LEG EXCEPT FOOT, AGE GREATER THAN 17 WITH CC */
    '254' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF UPPER ARM
                    AND LOWER LEG EXCEPT FOOT, AGE GREATER THAN 17 WITHOUT CC
*/
    '255' = '1' /* FRACTURES, SPRAINS, STRAINS AND DISLOCATIONS OF UPPER ARM
                   AND LOWER LEG EXCEPT FOOT, AGE 0-17 */
    '256'  =  '1'  /*  OTHER  MUSCULOSKELETAL  SYSTEM  AND  CONNECTIVE  TISSUE
DIAGNOSES */
    '271' = '1' /* SKIN ULCERS */
    '272' = '1' /* MAJOR SKIN DISORDERS WITH CC */
    '273' = '1' /* MAJOR SKIN DISORDERS WITHOUT CC */
    '274' = '1' /* MALIGNANT BREAST DISORDERS WITH CC */
    '275' = '1' /* MALIGNANT BREAST DISORDERS WITHOUT CC */
    '276' = '1' /* NONMALIGNANT BREAST DISORDERS */
    '277' = '1' /* CELLULITIS, AGE GREATER THAN 17 WITHOUT CC */
    '278' = '1' /* CELLULTIIS, AGE GREATER THAN 17 WITHOUT CC */
    '279' = '1' /* CELLULITIS, AGE 0-17 */
    '280' = '1' /* TRAUMA TO SKIN, SUBCUTANEOUS TISSUE AND BREAST, AGE GREATER
                   THAN 17 WITH CC */
    '281' = '1' /* TRAUMA TO SKIN, SUBCUTANEOUS TISSUE AND BREAST, AGE GREATER
                   THAN 17 WITH CC */
    '282' = '1' /* TRAUMA TO SKIN, SUBCUTANEOUS TISSUE AND BREAST, AGE 0-17 */
    '283' = '1' /* MINOR SKIN DISORDERS WITH CC */
    '284' = '1' /* MINOR SKIN DISORDERS WITHOUT CC */
    '294' = '1' /* DIABETES, AGE GREATER THAN 35 */
    '295' = '1' /* DIABETES, AGE 0-35 */
    '296'  =  '1'  /*  NUTRITIONAL  AND  MISCELLANEOUS  METABOLIC  DISORDERS,  AGE
GREATER
                   THAN 17 WITH CC */
    '297'  =  '1'  /*  NUTRITIONAL  AND  MISCELLANEOUS  METABOLIC  DISORDERS,  AGE
GREATER
                   THAN 17 WITHOUT CC */
```

```
    '298' = '1' /* NUTRITIONAL AND MISCELLANEOUS METABOLIC DISORDERS, AGE 0-17
*/
    '299' = '1' /* INBORN ERRORS OF METABOLISM */
    '300' = '1' /* ENDOCRINE DISORDERS WITH CC */
```
PSFMTS -- CONTINUED

```
    '301' = '1' /* ENDOCRINE DISORDERS WITHOUT CC */
    '316' = '1' /* RENAL FAILURE */
    '317' = '1' /* ADMISSION FOR RENAL DIALYSIS */
    '318' = '1' /* KIDNEY AND URINARY TRACT NEOPLASMS WITH CC */
    '319' = '1' /* KIDNEY AND URINARY TRACT NEOPLASMS WITHOUT CC */
    '320' = '1' /* KIDNEY AND URINARY TRACT INFECTIONS, AGE GREATER THAN 17
                   WITH CC */
    '321' = '1' /* KIDNEY AND URINARY TRACT INFECTIONS, AGE GREATER THAN 17
                   WITHOUT CC */
    '322' = '1' /* KIDNEY AND URINARY TRACT INFECTION, AGE 0-17 */
    '323' = '1' /* URINARY STONES WITH CC AND/ OR ESW LITHOTRIPSY */
    '324' = '1' /* URINARY STONES WITHOUT CC */
    '325' = '1' /* KIDNEY AND URINARY TRACT SIGNS AND SYMPTOMS, AGE GREATER
                   THAN 17 WITH CC */
    '326' = '1' /* KIDNEY AND URINARY TRACT SIGNS AND SYMPTOMS, AGE GREATER
                   THAN 17 WITHOUT CC */
    '327' = '1' /* KIDNEY AND URINARY TRACT SIGNS AND SYMPTOMS, AGE 0-17 */
    '328' = '1' /* URETHRAL STRICTURE, AGE GREATER THAN 17 WITH CC */
    '329' = '1' /* URETHRAL STRICTURE, AGE GREATER THAN 17 WITHOUT CC */
    '330' = '1' /* URETHRAL STRICTURE, AGE AGE 0-17 */
    '331' = '1' /* OTHER KIDNEY AND URINARY TRACT DIAGNOSES, AGE GREATER THAN
                   17 WITH CC */
    '332' = '1' /* OTHER KIDNEY AND URINARY TRACT DIAGNOSES, AGE GREATER THAN
                   17 WITHOUT CC */
    '333' = '1' /* OTHER KIDNEY AND URINARY TRACT DIAGNOSES, AGE 0-17 */
    '346' = '1' /* MALIGNANCY OF MALE REPRODUCTIVE SYSTEM WITH CC */
    '347' = '1' /* MALIGNANCY OF MALE REPRODUCTIVE SYSTEM WITHOUT CC */
    '348' = '1' /* BENIGN PROSTATIC HYPERTROPHY WITH CC */
    '349' = '1' /* BENIGN PROSTATIC HYPERTROPHY */
    '350' = '1' /* INFLAMMATION OF THE MALE REPRODUCTIVE SYSTEM */
    '351' = '1' /* STERILIZATION, MALE */
    '352' = '1' /* OTHER MALE REPRODUCTIVE SYSTEM DIAGNOSES */
    '366' = '1' /* MALIGNANCY OF FEMALE REPRODUCTIVE SYSTEM WITH CC */
    '367' = '1' /* MALIGNANCY OF FEMALE REPRODUCTIVE SYSTEM WITHOUT CC */
    '368' = '1' /* INFECTIONS OF FEMALE REPRODUCTIVE SYSTEM */
    '369' = '1' /* MENSTRUAL AND OTHER FEMALE REPRODUCTIVE SYSTEM DISORDERS */
    '372' = '1' /* VAGINAL DELIVERY WITH COMPLICATING DIAGNOSES */
    '373' = '1' /* VAGINAL DELIVERY WITHOUT COMPLICATING DIAGNOSES */
    '376' = '1' /* POSTPARTUM AND POSTABORTION DIAGNOSES WITHOUT OR PROCEDURE
*/
    '378' = '1' /* ENTOPIC PREGNANCY */
    '379' = '1' /* THREATENED ABORTION */
    '380' = '1' /* ABORTION WITHOUT D AND G */
    '382' = '1' /* FALSE LABOR */
    '383' = '1' /* OTHER ANTEPARTUM DIAGNOSES WITH MEDICAL COMPLICATIONS */
    '384' = '1' /* OTHER ANTEPARTUM DIAGNOSES WITHOUT MEDICAL COMPLICATIONS */
    '395' = '1' /* RED BLOOD CELL DISORDERS, AGE GREATER THAN 17 */
    '396' = '1' /* RED BLOOD CELL DISORDERS, AGE 0-17 */
    '397' = '1' /* COAGULATION DISORDERS */
    '398' = '1' /* RETICULOENDOTHELIAL AND IMMUNITY DISORDERS WITH CC */
    '399' = '1' /* RETICULOENDOTHELIAL AND IMMUNITY DISORDERS WITHOUT CC */
    '403' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH CC */
```

```
    '404' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITHOUT CC */
    '405' = '1' /* ACUTE LEUKEMIA WITHOUT MAJOR OR PROCEDURE, AGE 0-17 */
    '409' = '1' /* RADIOTHERAPY */
    '410' = '1' /* CHEMOTHERAPY WITHOUT ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS
*/
    '411' = '1' /* HISTORY OF MALIGNANCY WITHOUT ENDOSCOPY */
    '412' = '1' /* HISTORY OF MALIGNANCY WITH ENDOSCOPY */
```

PSFMTS -- CONTINUED

```
    '413' = '1' /* OTHER MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASM DIAGNOSES WITH CC */
    '414' = '1' /* OTHER MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASM DIAGNOSES WITHOUT CC */
    '416' = '1' /* SEPTICEMIA, AGE GREATER THAN 17 */
    '417' = '1' /* SEPTICEMIA, AGE 0-17 */
    '418' = '1' /* POSTOPERATIVE AND POSTTRAUMATIC INFECTIONS */
    '419' = '1' /* FEVER OF UNKNOWN ORIGIN, AGE GREATER THAN 17 WITH CC */
    '420' = '1' /* FEVER OF UNKNOWN ORIGIN, AGE GREATER THAN 17 WITHOUT CC */
    '421' = '1' /* VIRAL ILLNESS, AGE GREATER THAN 17 */
    '422' = '1' /* VIRAL ILLNESS AND FEVER OF UNKNOWN ORIGIN, AGE 0-17 */
    '423' = '1' /* OTHER INFECTIOUS AND PARASITIC DISEASES DIAGNOSES */
    '425' = '1' /* ACUTE ADJUSTMENT REACTIONS AND DISTURBANCES OF PSYCHOSOCIAL
                   DYSFUNCTION */
    '426' = '1' /* DEPRESSIVE NEUROSES */
    '427' = '1' /* NEUROSES EXCEPT DEPRESSIVE */
    '428' = '1' /* DISORDERS OF PERSONALITY AND IMPULSE CONTROL */
    '429' = '1' /* ORGANIC DISTURBANCES AND MENTAL RETARDATION */
    '430' = '1' /* PSYCHOSES */
    '431' = '1' /* CHILDHOOD MENTAL DISORDERS */
    '432' = '1' /* OTHER MENTAL DISORDER DIAGNOSES */
    '433' = '1' /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, LEFT AGAINST MEDICAL
ADVICE */
    '434' = '1' /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                   SYMPTOMATIC TREATMENT WITH CC */
    '435' = '1' /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, DETOXIFICATION OR OTHER
                   SYMPTOMATIC TREATMENT WITH CC */
    '436' = '1' /* ALCOHOL/DRUG DEPENDENCE WITH REHABILITATION THERAPY */
    '437' = '1' /* ALCOHOL DRUG DEPENDENCE WITH COMBINED REHABILITATION AND
                   DETOXIFICATION THERAPY */
    '444' = '1' /* TRAUMATIC INJURY, AGE GREATER THAN 17 WITH CC */
    '445' = '1' /* TRAUMATIC INJURY, AGE GREATER THAN 17 WITHOUT CC */
    '446' = '1' /* TRAUMATIC INJURY, AGE 0-17 */
    '447' = '1' /* ALLERGIC REACTIONS, AGE GREATER THAN 17 */
    '448' = '1' /* ALLERGIC REACTIONS, AGE 0-17 */
    '449' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE GREATER THAN 17
                   WITH CC */
    '450' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE GREATER THAN 17
                   WITHOUT CC */
    '451' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE 0-17 */
    '452' = '1' /* COMPLICATIONS OF TREATMENT WITH CC */
    '453' = '1' /* COMPLICATIONS OF TREATMENT WITHOUT CC */
    '454' = '1' /* OTHER INJURY, POISONING AND TOXIC EFFECT DIAGNOSES WITH CC
*/
    '455' = '1' /* OTHER INJURY, POISONING AND TOXIC EFFECT DIAGNOSES WITHOUT
CC */
    '456' = '1' /* NO LONGER VALID */
    '457' = '1' /* NO LONGER VALID */
    '460' = '1' /* NO LONGER VALID */
```

```
    '462' = '1' /* REHABILITATION */
    '463' = '1' /* SIGNS AND SYMPTOMS WITH CC */
    '464' = '1' /* SIGNS AND SYMPTOMS WITHOUT CC */
    '465' = '1' /* AFTERCARE WITH HISTORY OF MALIGNANCY AS SECONDARY DIAGNOSIS
*/
    '466' = '1' /* AFTERCARE WITHOUT HISTORY OF MALIGNANCY AS SECONDARY
DIAGNOSIS */
    '467' = '1' /* OTHER FACTORS INFLUENCING HEALTH STATUS */
    '473' = '1' /* ACUTE LEUKEMIA WITHOUT MAJOR OR PROCEDURE, AGE GREATER THAN
17 */
```
PSFMTS -- CONTINUED
```
    '474' = '1' /* NO LONGER VALID */
    '475' = '1' /* RESPIRATORY SYSTEM DIAGNOSIS WITH VENTILATOR SUPPORT */
    '487' = '1' /* OTHER MULTIPLE SIGNIFICANT TRAUMA */
    '489' = '1' /* HIV WITH MAJOR RELATED CONDITION */
    '490' = '1' /* HIV WITH OR WITHOUT OTHER RELATED CONDITION */
    '492' = '1' /* CHEMOTHERAPY WITH ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS */
    OTHER  = '0' ;

    /* Metastatic cancer */
    /* ICD-9-CM diagnosis codes: */

    VALUE $METACID
    '196'    = '1' /* SECONDARY AND UNSPECIFIED MALIGNANT NEOPLASM OF LYMPH
NODES */
    '1960'  = '1' /* MAL NEO LYMPH-HEAD/NECK */
    '1961'  = '1' /* MAL NEO LYMPH-INTRATHOR */
    '1962'  = '1' /* MAL NEO LYMPH INTRA-ABD */
    '1963'  = '1' /* MAL NEO LYMPH-AXILLA/ARM */
    '1965'  = '1' /* MAL NEO LYMPH-INGUIN/LEG */
    '1966'  = '1' /* MAL NEO LYMPH-INTRAPELV */
    '1968'  = '1' /* MAL NEO LYMPH NODE-MULT */
    '1969'  = '1' /* MAL NEO LYMPH NODE NOS */
    '197'   = '1' /* SECONDARY MALIGNANT NEOPLASM OF RESPIRATORY AND DIGESTIVE
                     SYSTEMS */
    '1970'  = '1' /* SECONDARY MALIG NEO LUNG */
    '1971'  = '1' /* SEC MAL NEO MEDIASTINUM */
    '1972'  = '1' /* SECOND MALIG NEO PLEURA */
    '1973'  = '1' /* SEC MALIG NEO RESP NEC */
    '1974'  = '1' /* SEC MALIG NEO SM BOWEL */
    '1975'  = '1' /* SEC MALIG NEO LG BOWEL */
    '1976'  = '1' /* SEC MAL NEO PERITONEUM */
    '1977'  = '1' /* SECOND MALIG NEO LIVER */
    '1978'  = '1' /* SEC MAL NEO GI NEC */
    '198'   = '1' /* SECONDARY MALIGNANT NEOPLASM OF OTHER SPECIFIED SITES */
    '1980'  = '1' /* SECOND MALIG NEO KIDNEY */
    '1981'  = '1' /* SEC MALIG NEO URIN NEC */
    '1982'  = '1' /* SECONDARY MALIG NEO SKIN */
    '1983'  = '1' /* SEC MAL NEO BRAIN/SPINE */
    '1984'  = '1' /* SEC MALIG NEO NERVE NEC */
    '1985'  = '1' /* SECONDARY MALIG NEO BONE */
    '1986'  = '1' /* SECOND MALIG NEO OVARY */
    '1987'  = '1' /* SECOND MALIG NEO ADRENAL */
    '1988'  = '1' /* OTH SECONDARY MALIG NEO */
    '19881' = '1' /* SECOND MALIG NEO BREAST */
    '19882' = '1' /* SECOND MALIG NEO GENITAL */
    '19889' = '1' /* SECONDARY MALIG NEO NEC */
```

```
    '1990'  = '1' /* MALIGNANT NEOPLASM WITHOUT SPECIFICATION OF SITE,
                     DISSEMINATED */
    OTHER   = '0' ;

    /* Obstetric trauma */
    /* ICD-9-CM diagnosis codes: */

    VALUE $OBTRAID
    '66430' = '1' /* TRAUMA TO PERINEUM AND VULVA DURING DELIVERY, FOURTH-
                     DEGREE PERINEAL LACERATION */
    '66431' = '1' /* TRAUMA TO PERINEUM AND VULVA DURING DELIVERY, FOURTH-
                     DEGREE PERINEAL LACERATION */
    '66434' = '1' /* TRAUMA TO PERINEUM AND VULVA DURING DELIVERY, FOURTH-
```

PSFMTS -- CONTINUED

```
                     DEGREE PERINEAL LACERATION */
    '66530' = '1' /* OTHER OBSTETRICAL TRAUMA, LACERATION OF CERVIX */
    '66531' = '1' /* OTHER OBSTETRICAL TRAUMA, LACERATION OF CERVIX */
    '66534' = '1' /* OTHER OBSTETRICAL TRAUMA, LACERATION OF CERVIX */
    '66540' = '1' /* OTHER OBSTETRICAL TRAUMA, HIGH VAGINAL LACERATIONS */
    '66541' = '1' /* OTHER OBSTETRICAL TRAUMA, HIGH VAGINAL LACERATIONS */
    '66544' = '1' /* OTHER OBSTETRICAL TRAUMA, HIGH VAGINAL LACERATIONS */
    '66550' = '1' /* OTHER OBSTETRICAL TRAUMA, OTHER INJURY TO PELVIC ORGANS
*/
    '66551' = '1' /* OTHER OBSTETRICAL TRAUMA, OTHER INJURY TO PELVIC ORGANS
*/
    '66554' = '1' /* OTHER OBSTETRICAL TRAUMA, OTHER INJURY TO PELVIC ORGANS
*/
    OTHER   = '0' ;

    /* Obstetric trauma */
    /* ICD-9-CM procedure codes: */

    VALUE $OBTRAIP
    '755'  = '1' /* REPAIR OB LACERAT UTERUS */
    '7550' = '1' /* REPAIR OF CURRENT OBSTETRIC LACERATIONS UTERUS */
    '7551' = '1' /* REPAIR OF CURRENT OBSTETRIC LACERATIONS OF CERVIX */
    '7552' = '1' /* REPAIR OF CURRENT OBSTETRIC LACERATIONS OF CORPUS UTERI */
    '7561' = '1' /* REPAIR OF CURRENT OBSTETRIC LACERATION OF BLADDER AND
URETHRA */
    '7562' = '1' /* REPARI OF CURRENT OBSTETRIC LACERATION OF RECTUM AND
                    SPHINCTER ANI */
    OTHER  = '0' ;

    /* Osteogenesis imperfecta */;
    /* ICD-9-CM diagnosis codes: */

    VALUE $OSTEOID
    '75651' = '1'
    OTHER   = '0' ;

    /* Physiologic and metabolic derangements */
    /* ICD-9-CM diagnosis codes: */

    VALUE $PHYSIDA
    '2501'  = '1' /* DIABETES W KETOACIDOSIS */
    '25010' = '1' /* TYPE 2, OR UNSPECIFIED TYPE, NOT STATED AS UNCONTROLLED
*/
```

```
    '25011' = '1' /* TYPE 1 NOT STATED AS UNCONTROLLED */
    '25012' = '1' /* TYPE 2 OR UNSPECIFIED TYPE, UNCONTROLLED */
    '25013' = '1' /* TYPE 1 UNCONTROLLED */
    '2502'  = '1' /* DIAB W HYPEROSMOLAR COMA */
    '25020' = '1' /* TYPE 2, OR UNSPECIFIED TYPE, NOT STATED AS UNCONTROLLED
*/
    '25021' = '1' /* TYPE 1 NOT STATED AS UNCONTROLLED */
    '25022' = '1' /* TYPE 2 OR UNSPECIFIED TYPE, UNCONTROLLED */
    '25023' = '1' /* TYPE 1 UNCONTROLLED */
    '2503'  = '1' /* DIABETES WITH COMA NEC */
    '25030' = '1' /* TYPE 2, OR UNSPECIFIED TYPE, NOT STATED AS UNCONTROLLED
*/
    '25031' = '1' /* TYPE 1 NOT STATED AS UNCONTROLLED */
    '25032' = '1' /* TYPE 2 OR UNSPECIFIED TYPE, UNCONTROLLED */
    '25033' = '1' /* TYPE 1 UNCONTROLLED */
    OTHER   = '0' ;
    PSFMTS -- CONTINUED

    VALUE $PHYSIDB
    '584'   = '1' /* ACUTE RENAL FAILURE */
    '5845'  = '1' /* WITH LESION OF TUBULAR NECROSIS */
    '5846'  = '1' /* WITH LESION OF RENAL CORTICAL NECROSIS */
    '5847'  = '1' /* WITH LESION OF RENAL MEDULLARY [PAPILLARY] NECROSIS */
    '5848'  = '1' /* WITH OTHER SPECIFIED PATHOLOGICAL LESION IN KIDNEY */
    '5849'  = '1' /* ACUTE RENAL FAILURE, UNSPECIFIED */
    OTHER   = '0' ;

    /* Poisoning */
    /* ICD-9-CM diagnosis codes ( includes 4th and 5th digits): */

    VALUE $POISOID
    '960'   = '1' /* POISONING BY ANTIBIOTICS */
    '9600'  = '1' /* POISONING-PENICILLINS */
    '9601'  = '1' /* POIS-ANTIFUNGAL ANTIBIOT */
    '9602'  = '1' /* POISON-CHLORAMPHENICOL */
    '9603'  = '1' /* POIS-ERYTHROMYC/MACROLID */
    '9604'  = '1' /* POISONING-TETRACYCLINE */
    '9605'  = '1' /* POIS-CEPHALOSPORIN GROUP */
    '9606'  = '1' /* POIS-ANTIMYCOBAC ANTIBIO */
    '9607'  = '1' /* POIS-ANTINEOP ANTIBIOTIC */
    '9608'  = '1' /* POISONING-ANTIBIOTIC NEC */
    '9609'  = '1' /* POISONING-ANTIBIOTIC NOS */
    '961'   = '1' /* POISONING BY OTHER ANTI-INFECTIVES */
    '9610'  = '1' /* POISONING-SULFONAMIDES */
    '9611'  = '1' /* POIS-ARSENIC ANTI-INFEC */
    '9612'  = '1' /* POIS-HEAV MET ANTI-INFEC */
    '9613'  = '1' /* POIS-QUINOLINE/HYDROXYQU */
    '9614'  = '1' /* POISONING-ANTIMALARIALS */
    '9615'  = '1' /* POIS-ANTIPROTOZ DRUG NEC */
    '9616'  = '1' /* POISONING-ANTHELMINTICS */
    '9617'  = '1' /* POISONING-ANTIVIRAL DRUG */
    '9618'  = '1' /* POIS-ANTIMYCOBAC DRG NEC */
    '9619'  = '1' /* POIS-ANTI-INFECT NEC/NOS */
    '962'   = '1' /* POISONING BY HORMONES AND SYNTHETIC SUBSTITUTES */
    '9620'  = '1' /* POIS-CORTICOSTEROIDS */
    '9621'  = '1' /* POISONING-ANDROGENS */
    '9622'  = '1' /* POISONING-OVARIAN HORMON */
    '9623'  = '1' /* POISON-INSULIN/ANTIDIAB */
```

```
    '9624'  = '1' /* POIS-ANT PITUITARY HORM */
    '9625'  = '1' /* POIS-POST PITUITARY HORM */
    '9626'  = '1' /* POISONING-PARATHYROIDS */
    '9627'  = '1' /* POISONING-THYROID/DERIV */
    '9628'  = '1' /* POISON-ANTITHYROID AGENT */
    '9629'  = '1' /* POISONING HORMON NEC/NOS */
    '963'   = '1' /* POISONING BY PRIMARILY SYSTEMIC AGENTS */
    '9630'  = '1' /* POIS-ANTIALLRG/ANTIEMET */
    '9631'  = '1' /* POIS-ANTINEOPL/IMMUNOSUP */
    '9632'  = '1' /* POISONING-ACIDIFYING AGT */
    '9633'  = '1' /* POISONING-ALKALIZING AGT */
    '9634'  = '1' /* POISONING-ENZYMES NEC */
    '9635'  = '1' /* POISONING-VITAMINS NEC */
    '9638'  = '1' /* POISONING-SYSTEM AGT NEC */
    '9639'  = '1' /* POISONING-SYSTEM AGT NOS */
    '964'      =  '1'  /*  POISONING  BY  AGENTS  PRIMARILY  AFFECTING  BLOOD
CONSTITUENTS */
    '9640'  = '1' /* POISONING-IRON/COMPOUNDS */
    '9641'  = '1' /* POISON-LIVER/ANTIANEMICS */
```

PSFMTS -- CONTINUED

```
    '9642'  = '1' /* POISONING-ANTICOAGULANTS */
    '9643'  = '1' /* POISONING-VITAMIN K */
    '9644'  = '1' /* POISON-FIBRINOLYSIS AGNT */
    '9645'  = '1' /* POISONING-COAGULANTS */
    '9646'  = '1' /* POISONING-GAMMA GLOBULIN */
    '9647'  = '1' /* POISONING-BLOOD PRODUCT */
    '9648'  = '1' /* POISONING-BLOOD AGT NEC */
    '9649'  = '1' /* POISONING-BLOOD AGT NOS */
    '965'       =  '1'  /*  POISONOING  BY  ANALGESICS,  ANTIPYRETICS,  AND
ANTIRHEUMATICS */
    '9650'  = '1' /* POISONING-OPIATES */
    '96500' = '1' /* POISONING-OPIUM NOS */
    '96501' = '1' /* POISONING-HEROIN */
    '96502' = '1' /* POISONING-METHADONE */
    '96509' = '1' /* POISONING-OPIATES NEC */
    '9651'  = '1' /* POISONING-SALICYLATES */
    '9654'  = '1' /* POIS-AROM ANALGESICS NEC */
    '9655'  = '1' /* POISONING-PYRAZOLE DERIV */
    '9656'  = '1' /* POISONING-ANTIRHEUMATICS */
    '96561' = '1' /* POIS-PROPIONIC ACID DERV */
    '96569' = '1' /* POISON-ANTIRHEUMATIC NEC */
    '9657'  = '1' /* POIS-NO-NARC ANALGES NEC */
    '9658'  = '1' /* POIS-ANALGES/ANTIPYR NEC */
    '9659'  = '1' /* POIS-ANALGES/ANTIPYR NOS */
    '966'   = '1' /* POISONING BY ANTICONVULSANTS AND ANTI-PARKINSONISM DRUGS
*/
    '9660'  = '1' /* POISON-OXAZOLIDINE DERIV */
    '9661'  = '1' /* POISON-HYDANTOIN DERIVAT */
    '9662'  = '1' /* POISONING-SUCCINIMIDES */
    '9663'  = '1' /* POIS-ANTICONVUL NEC/NOS */
    '9664'  = '1' /* POIS-ANTI-PARKINSON DRUG */
    '967'   = '1' /* POISONING BY SEDATIVES AND HYPNOTICS */
    '9670'  = '1' /* POISONING-BARBITURATES */
    '9671'  = '1' /* POISONING-CHLORAL HYDRAT */
    '9672'  = '1' /* POISONING-PARALDEHYDE */
    '9673'  = '1' /* POISONING-BROMINE COMPND */
    '9674'  = '1' /* POISONING-METHAQUALONE */
```

```
    '9675'  = '1' /* POISONING-GLUTETHIMIDE */
    '9676'  = '1' /* POISON-MIX SEDATIVE NEC */
    '9678'  = '1' /* POIS-SEDATIVE/HYPNOT NEC */
    '9679'  = '1' /* POIS-SEDATIVE/HYPNOT NOS */
    '968'   = '1' /* POISONING BY OTHER CENTRAL NERVOUS SYSTEM DEPRESSANTS AND
                     ANESTHETICS */
    '9680'  = '1' /* POIS-CNS MUSCLE DEPRESS */
    '9681'  = '1' /* POISONING-HALOTHANE */
    '9682'  = '1' /* POISON-GAS ANESTHET NEC */
    '9683'  = '1' /* POISON-INTRAVEN ANESTHET */
    '9684'  = '1' /* POIS-GEN ANESTH NEC/NOS */
    '9685'  = '1' /* POIS-TOPIC/INFILT ANESTH */
    '9686'  = '1' /* POIS-NERVE/PLEX-BLK ANES */
    '9687'  = '1' /* POISON-SPINAL ANESTHETIC */
    '9689'  = '1' /* POIS-LOCAL ANEST NEC/NOS */
    '969'   = '1' /* POISONING BY PSYCHOTROPIC AGENTS */
    '9690'  = '1' /* POISONING-ANTIDEPRESSANT */
    '9691'  = '1' /* POIS-PHENOTHIAZINE TRANQ */
    '9692'  = '1' /* POIS-BUTYROPHENONE TRANQ */
    '9693'  = '1' /* POISON-ANTIPSYCHOTIC NEC */
    '9694'  = '1' /* POIS-BENZODIAZEPINE TRAN */
    '9695'  = '1' /* POISON-TRANQUILIZER NEC */
   PSFMTS -- CONTINUED

    '9696'  = '1' /* POISONING-HALLUCINOGENS */
    '9697'  = '1' /* POISON-PSYCHOSTIMULANTS */
    '9698'  = '1' /* POISON-PSYCHOTROPIC NEC */
    '9699'  = '1' /* POISON-PSYCHOTROPIC NOS */
    '970'   = '1' /* POISONING BY CENTRAL NERVOUS SYSTEM STIMULANTS */
    '9700'  = '1' /* POISONING-ANALEPTICS */
    '9701'  = '1' /* POISON-OPIATE ANTAGONIST */
    '9708'  = '1' /* POIS-CNS STIMULANTS NEC */
    '9709'  = '1' /* POIS-CNS STIMULANT NOS */
    '971'   = '1' /* POISONING BY DRUGS PRIMARILY AFFECTING THE AUTONOMIC
                     NERVOUS SYSTEM */
    '9710'  = '1' /* POIS-PARASYMPATHOMIMETIC */
    '9711'  = '1' /* POIS-PARASYMPATHOLYTICS */
    '9712'  = '1' /* POISON-SYMPATHOMIMETICS */
    '9713'  = '1' /* POISONING-SYMPATHOLYTICS */
    '9719'  = '1' /* POIS-AUTONOMIC AGENT NOS */
    '972'      =  '1'  /*  POISONING  BY  AGENTS  PRIMARILY  AFFECTING  THE
CARDIOVASCULAR
                     SYSTEM */
    '9720'  = '1' /* POIS-CARD RHYTHM REGULAT */
    '9721'  = '1' /* POISONING-CARDIOTONICS */
    '9722'  = '1' /* POISONING-ANTILIPEMICS */
    '9723'  = '1' /* POIS-GANGLION BLOCK AGT */
    '9724'  = '1' /* POIS-CORONARY VASODILAT */
    '9725'  = '1' /* POISON-VASODILATOR NEC */
    '9726'  = '1' /* POIS-ANTIHYPERTEN AGENT */
    '9727'  = '1' /* POISON-ANTIVARICOSE DRUG */
    '9728'  = '1' /* POISON-CAPILLARY ACT AGT */
    '9729'  = '1' /* POIS-CARDIOVASC AGT NEC */
    '973'      =  '1'  /*  POISOING  BY  AGENTS  PRIMARILY  AFFECTING  THE
GASTROINTESTINAL
                     SYSTEM */
    '9730'  = '1' /* POIS-ANTACID/ANTIGASTRIC */
    '9731'  = '1' /* POIS-IRRITANT CATHARTICS */
```

```
    '9732'  = '1' /* POIS-EMOLLIENT CATHARTIC */
    '9733'  = '1' /* POISONING-CATHARTIC NEC */
    '9734'  = '1' /* POISONING-DIGESTANTS */
    '9735'  = '1' /* POISONING-ANTIDIARRH AGT */
    '9736'  = '1' /* POISONING-EMETICS */
    '9738'  = '1' /* POISONING-GI AGENTS NEC */
    '9739'  = '1' /* POISONING-GI AGENT NOS */
    '974'    = '1' /* POISONSING BY WATER, MINERAL, AND URIC ACID METABOLSIM
DRUGS */
    '9740'  = '1' /* POIS-MERCURIAL DIURETICS */
    '9741'  = '1' /* POIS-PURINE DIURETICS */
    '9742'  = '1' /* POIS-H2CO3 ANHYDRA INHIB */
    '9743'  = '1' /* POISONING-SALURETICS */
    '9744'  = '1' /* POISONING-DIURETICS NEC */
    '9745'  = '1' /* POIS-ELECTRO/CAL/WAT AGT */
    '9746'  = '1' /* POISON-MINERAL SALTS NEC */
    '9747'  = '1' /* POIS-URIC ACID METABOL */
    '975'   = '1' /* POISONING BY AGENTS PRIMARILY ACTING ON THE SMOOTH AND
                     SKELETAL MUSCLES AND RESPIRATORY SYSTEM */
    '9750'  = '1' /* POISONING-OXYTOCIC AGENT */
    '9751'  = '1' /* POIS-SMOOTH MUSCLE RELAX */
    '9752'  = '1' /* POIS-SKELET MUSCLE RELAX */
    '9753'  = '1' /* POISON-MUSCLE AGENT NEC */
    '9754'  = '1' /* POISONING-ANTITUSSIVES */
    '9755'  = '1' /* POISONING-EXPECTORANTS */
```

PSFMTS -- CONTINUED

```
    '9756'  = '1' /* POIS-ANTI-COLD DRUGS */
    '9757'  = '1' /* POISONING-ANTIASTHMATICS */
    '9758'  = '1' /* POIS-RESPIR DRUG NEC/NOS */
    '976'   = '1' /* POISONING BY AGENTS PRIMARILY AFFECTING SKIN AND MUCOUS
                     MEMBRANE, OPTHAMOLOGICAL, OTORHINOLARYNCOLOGICAL AND
                     DENTAL DRUGS */
    '9760'  = '1' /* POIS-LOCAL ANTI-INFECT */
    '9761'  = '1' /* POISONING-ANTIPRURITICS */
    '9762'  = '1' /* POIS-LOC ASTRING/DETERG */
    '9763'  = '1' /* POIS-EMOL/DEMUL/PROTECT */
    '9764'  = '1' /* POISON-HAIR/SCALP PREP */
    '9765'  = '1' /* POIS-EYE ANTI-INFEC/DRUG */
    '9766'  = '1' /* POISON-ENT PREPARATION */
    '9767'  = '1' /* POIS-TOPICAL DENTAL DRUG */
    '9768'  = '1' /* POIS-SKIN/MEMBR AGNT NEC */
    '9769'  = '1' /* POIS-SKIN/MEMBR AGNT NOS */
    '977'   = '1' /* POISONING BY OTHER AND UNSPECIFIED DRUGS AND MEDICNAL
                     SUBSTANCES */
    '9770'  = '1' /* POISONING-DIETETICS */
    '9771'  = '1' /* POISON-LIPOTROPIC DRUGS */
    '9772'  = '1' /* POISONING-ANTIDOTES NEC */
    '9773'  = '1' /* POISON-ALCOHOL DETERRENT */
    '9774'  = '1' /* POIS-PHARMACEUT EXCIPIEN */
    '9778'  = '1' /* POISON-MEDICINAL AGT NEC */
    '9779'  = '1' /* POISON-MEDICINAL AGT NOS */
    '978'   = '1' /* POISONING BY BACTERIAL VACCINES */
    '9780'  = '1' /* POISONING-BCG VACCINE */
    '9781'  = '1' /* POIS-TYPH/PARATYPH VACC */
    '9782'  = '1' /* POISONING-CHOLERA VACCIN */
    '9783'  = '1' /* POISONING-PLAGUE VACCINE */
    '9784'  = '1' /* POISONING-TETANUS VACCIN */
```

```
'9785'  = '1' /* POIS-DIPHTHERIA VACCINE */
'9786'  = '1' /* POIS-PERTUSSIS VACCINE */
'9788'  = '1' /* POIS-BACT VACCIN NEC/NOS */
'9789'  = '1' /* POIS-MIX BACTER VACCINES */
'979'   = '1' /* POISONING BY OTHER VACCINES AND BIOLOGICAL SUBSTANCES */
'9790'  = '1' /* POISON-SMALLPOX VACCINE */
'9791'  = '1' /* POISON-RABIES VACCINE */
'9792'  = '1' /* POISON-TYPHUS VACCINE */
'9793'  = '1' /* POIS-YELLOW FEVER VACCIN */
'9794'  = '1' /* POISONING-MEASLES VACCIN */
'9795'  = '1' /* POIS-POLIOMYELIT VACCINE */
'9796'  = '1' /* POIS-VIRAL/RICK VACC NEC */
'9797'  = '1' /* POISONING-MIXED VACCINE */
'9799'  = '1' /* POIS-VACCINE/BIOLOG NEC */
'E850'  = '1' /* ACC POISON-ANALGESIC */
'E8500' = '1' /* ACC POISON-HEROIN */
'E8501' = '1' /* ACC POISON-METHADONE */
'E8502' = '1' /* ACC POISON-OPIATES NEC */
'E8503' = '1' /* ACC POISON-SALICYLATES */
'E8504' = '1' /* ACC POISON-AROM ANALGESC */
'E8505' = '1' /* ACC POISON-PYRAZOLE DERV */
'E8506' = '1' /* ACC POISON-ANTIRHEUMATIC */
'E8507' = '1' /* ACC POISON-NONNARC ANALG */
'E8508' = '1' /* ACC POISON-ANALGESIC NEC */
'E8509' = '1' /* ACC POISON-ANALGESIC NOS */
'E851'  = '1' /* ACC POISON-BARBITURATES */
'E852'  = '1' /* ACC POISON-SEDATIVES NEC */
'E8520' = '1' /* ACC POISN-CHLORL HYDRATE */
PSFMTS -- CONTINUED

'E8521' = '1' /* ACC POISON-PARALDEHYDE */
'E8522' = '1' /* ACC POISON-BROMINE CMPND */
'E8523' = '1' /* ACC POISON-METHAQUALONE */
'E8524' = '1' /* ACC POISON-GLUTETHIMIDE */
'E8525' = '1' /* ACC POISON-MIX SEDTV NEC */
'E8528' = '1' /* ACC POISON-SEDATIVES NEC */
'E8529' = '1' /* ACC POISON-SEDATIVES NOS */
'E853'  = '1' /* ACC POISON-TRANQUILIZERS */
'E8530' = '1' /* ACC POIS-PHENTHIAZ TRANQ */
'E8531' = '1' /* ACC POIS-BUTYRPHEN TRANQ */
'E8532' = '1' /* ACC POISN-BENZDIAZ TRANQ */
'E8538' = '1' /* ACC POISN-TRANQUILZR NEC */
'E8539' = '1' /* ACC POISN-TRANQUILZR NOS */
'E854'  = '1' /* ACC POISN-PSYCHTROPC NEC */
'E8540' = '1' /* ACC POISON-ANTIDEPRESSNT */
'E8541' = '1' /* ACC POISON-HALLUCINOGENS */
'E8542' = '1' /* ACC POISN-PSYCHSTIMULANT */
'E8543' = '1' /* ACC POISON-CNS STIMULANT */
'E8548' = '1' /* ACC POISN PSYCHOTROP NEC */
'E855'  = '1' /* ACC POISON-CNS DRUGS NEC */
'E8550' = '1' /* ACC POISN-ANTICONVULSANT */
'E8551' = '1' /* ACC POISN-CNS DEPRES NEC */
'E8552' = '1' /* ACC POISN-LOCAL ANESTHET */
'E8553' = '1' /* ACC POISON-CHOLINERGICS */
'E8554' = '1' /* ACC POISN-ANTICHOLINERG */
'E8555' = '1' /* ACC POISON-ADRENERGICS */
'E8556' = '1' /* ACC POISN-SYMPATHOLYTICS */
'E8558' = '1' /* ACC POISON-CNS DRUG NEC */
```

```
'E8559' = '1' /* ACC POISON-CNS DRUG NOS */
'E856'  = '1' /* ACC POISON-ANTIBIOTICS */
'E857'  = '1' /* ACC POIS-OTH ANTI-INFECT */
'E858'  = '1' /* ACC POISONING-DRUGS NEC */
'E8580' = '1' /* ACC POISON-HORMONES */
'E8581' = '1' /* ACC POISN-SYSTEMIC AGENT */
'E8582' = '1' /* ACC POISON-BLOOD AGENT */
'E8583' = '1' /* ACC POISN-CARDIOVASC AGT */
'E8584' = '1' /* ACC POISON-GI AGENT */
'E8585' = '1' /* ACC POISN-METABOL AGNT */
'E8586' = '1' /* ACC POISN-MUSCL/RESP AGT */
'E8587' = '1' /* ACC POISN-SKIN/EENT AGNT */
'E8588' = '1' /* ACC POISONING-DRUG NEC */
'E8589' = '1' /* ACC POISONING-DRUG NOS */
'E860'  = '1' /* ACC POISON-ALCOHOL NEC */
'E8600' = '1' /* ACC POISN-ALCOHOL BEVRAG */
'E8601' = '1' /* ACC POISON-ETHYL ALCOHOL */
'E8602' = '1' /* ACC POISN-METHYL ALCOHOL */
'E8603' = '1' /* ACC POISN-ISOPROPYL ALC */
'E8604' = '1' /* ACC POISON-FUSEL OIL */
'E8608' = '1' /* ACC POISON-ALCOHOL NEC */
'E8609' = '1' /* ACC POISON-ALCOHOL NOS */
'E861'  = '1' /* ACC POISN-CLEANSING AGNT */
'E8610' = '1' /* ACC POIS-SYNTH DETERGENT */
'E8611' = '1' /* ACC POISON-SOAP PRODUCTS */
'E8612' = '1' /* ACC POISON-POLISHES */
'E8613' = '1' /* ACC POISON-CLEANSER NEC */
'E8614' = '1' /* ACC POISON-DISINFECTANTS */
'E8615' = '1' /* ACC POISON-LEAD PAINTS */
'E8616' = '1' /* ACC POISON-PAINTS NEC */
'E8619' = '1' /* ACC POISON-CLEANSER NOS */
```

PSFMTS -- CONTINUED

```
'E862'  = '1' /* ACC POIS-PETRLEUM/SOLVNT */
'E8620' = '1' /* ACC POISN-PETROL SOLVENT */
'E8621' = '1' /* ACC POISN-PETROLEUM FUEL */
'E8622' = '1' /* ACC POIS-LUBRICATING OIL */
'E8623' = '1' /* ACC POIS-PETROLEUM SOLID */
'E8624' = '1' /* ACC POISN-SOLVENTS NEC */
'E8629' = '1' /* ACC POISN-SOLVENT NOS */
'E863'  = '1' /* ACC POIS-AGRICUL AGT NEC */
'E8630' = '1' /* ACC POIS-CHLORINE PESTIC */
'E8631' = '1' /* ACC POIS-PHOSPH PESTICID */
'E8632' = '1' /* ACC POISON-CARBAMATES */
'E8633' = '1' /* ACC POISN-MIXED PESTICID */
'E8634' = '1' /* ACC POISON-PESTICIDE NEC */
'E8635' = '1' /* ACC POISON-HERBICIDES */
'E8636' = '1' /* ACC POISON-FUNGICIDES */
'E8637' = '1' /* ACC POISON-RODENTICIDES */
'E8638' = '1' /* ACC POISON-FUMIGANTS */
'E8639' = '1' /* ACC POIS-AGRCULT NEC/NOS */
'E864'  = '1' /* ACC POISON-CAUSTIC NEC */
'E8640' = '1' /* ACC POIS-CORROSIV AROMAT */
'E8641' = '1' /* ACC POISON-ACIDS */
'E8642' = '1' /* ACC POISN-CAUSTIC ALKALI */
'E8643' = '1' /* ACC POISON-CAUSTIC NEC */
'E8644' = '1' /* ACC POISON-CAUSTIC NOS */
'E865'  = '1' /* ACC POISON-FOOD & PLANTS */
```

```
'E8650' = '1' /* ACC POISON-MEAT */
'E8651' = '1' /* ACC POISON-SHELLFISH */
'E8652' = '1' /* ACC POISON-FISH NEC */
'E8653' = '1' /* ACC POISON-BERRIES/SEEDS */
'E8654' = '1' /* ACC POISON-PLANTS NEC */
'E8655' = '1' /* ACC POISON-MUSHROOMS */
'E8658' = '1' /* ACC POISON-FOOD NEC */
'E8659' = '1' /* ACC POISN-FOOD/PLANT NOS */
'E866'  = '1' /* ACC POISN-SOLID/LIQ NEC */
'E8660' = '1' /* ACC POISONING-LEAD */
'E8661' = '1' /* ACC POISONING-MERCURY */
'E8662' = '1' /* ACC POISONING-ANTIMONY */
'E8663' = '1' /* ACC POISONING-ARSENIC */
'E8664' = '1' /* ACC POISON-METALS NEC */
'E8665' = '1' /* ACC POISON-PLANT FOOD */
'E8666' = '1' /* ACC POISON-GLUES */
'E8667' = '1' /* ACC POISON-COSMETICS */
'E8668' = '1' /* ACC POIS-SOLID/LIQ NEC */
'E8669' = '1' /* ACC POIS-SOLID/LIQ NOS */
'E867'  = '1' /* ACC POISON-PIPED GAS */
'E868'  = '1' /* ACC POIS-UTIL GAS/CO NEC */
'E8680' = '1' /* ACC POIS-LIQ PETROL GAS */
'E8681' = '1' /* ACC POIS-UTL GAS NEC/NOS */
'E8682' = '1' /* ACC POISON-EXHAUST GAS */
'E8683' = '1' /* ACC POIS-CO/DOMESTC FUEL */
'E8688' = '1' /* ACC POIS-CARBN MONOX NEC */
'E8689' = '1' /* ACC POIS-CARBN MONOX NOS */
'E869'  = '1' /* ACC POISON-GAS/VAPOR NEC */
'E8690' = '1' /* ACC POISN-NITROGEN OXIDE */
'E8691' = '1' /* ACC POISN-SULFUR DIOXIDE */
'E8692' = '1' /* ACC POISON-FREON */
'E8693' = '1' /* ACC POISON-TEAR GAS */
'E8694' = '1' /* SCNDHND TBCCO SMOKE */
'E8698' = '1' /* ACC POISON-GAS/VAPOR NEC */
```

PSFMTS -- CONTINUED

```
'E8699' = '1' /* ACC POISON-GAS/VAPOR NOS */
'E951'  = '1' /* POISON-UTILITY GAS */
'E9510' = '1' /* POISON-PIPED GAS */
'E9511' = '1' /* POISON-GAS IN CONTAINER */
'E9518' = '1' /* POISON-UTILITY GAS NEC */
'E952'  = '1' /* POISON-GAS/VAPOR NEC */
'E9520' = '1' /* POISON-EXHAUST GAS */
'E9521' = '1' /* POISON-CO NEC */
'E9528' = '1' /* POISON-GAS/VAPOR NEC */
'E9529' = '1' /* POISON-GAS/VAPOR NOS */
'E962'  = '1' /* ASSAULT-POISONING */
'E9620' = '1' /* ASSAULT-POIS W MEDIC AGT */
'E9621' = '1' /* ASSAULT-POIS W SOLID/LIQ */
'E9622' = '1' /* ASSAULT-POIS W GAS/VAPOR */
'E9629' = '1' /* ASSAULT-POISONING NOS */
'E980'  = '1' /* UNDETERM POIS-SOL/LIQUID */
'E9800' = '1' /* UNDETERM POIS-ANALGESICS */
'E9801' = '1' /* UNDETERM POIS-BARBITURAT */
'E9802' = '1' /* UNDET POIS-SED/HYPN NEC */
'E9803' = '1' /* UNDETERM POIS-PSYCHOTROP */
'E9804' = '1' /* UNDET POIS-MED AGNT NEC */
'E9805' = '1' /* UNDET POIS-MED AGNT NOS */
```

```
'E9806' = '1' /* UNDET POIS-CORROS/CAUST */
'E9807' = '1' /* UNDET POIS-AGRICULT AGNT */
'E9808' = '1' /* UNDETER POIS-ARSENIC */
'E9809' = '1' /* UNDETER POIS-SOL/LIQ NEC */
'E981'  = '1' /* UNDETER POIS-UTILITY GAS */
'E9810' = '1' /* UNDETER POIS-PIPED GAS */
'E9811' = '1' /* UNDET POIS-CONTAINER GAS */
'E9818' = '1' /* UNDET POIS-UTIL GAS NEC */
'E982'  = '1' /* UNDETERM POISON-GAS NEC */
'E9820' = '1' /* UNDETER POIS-EXHAUST GAS */
'E9821' = '1' /* UNDETERMIN POISON-CO NEC */
'E9828' = '1' /* UNDET POIS-GAS/VAPOR NEC */
'E9829' = '1' /* UNDET POIS-GAS/VAPOR NOS */
OTHER   = '0' ;

/* Post-operative hematoma */
/* ICD-9-CM diagnosis code: */

VALUE $POHMAID
'99812' = '1' /* HEMATOMA COMPLICATING A PROCEDURE */
OTHER   = '0' ;

/* Post-operative hemorrhage */
/* ICD-9-CM diagnosis code: */

VALUE $POHMRID
'99811' = '1' /* HEMORRHAGE COMPLICATING A PROCEDURE */
OTHER   = '0' ;

/* Preterm infant */
/* ICD-9-CM diagnosis codes: */

VALUE $PRETEID
'76501' = '1' /* EXTREME IMMATURITY */
'76502' = '1' /* EXTREME IMMATURITY */
'76503' = '1' /* EXTREME IMMATURITY */
'76504' = '1' /* EXTREME IMMATURITY */
PSFMTS -- CONTINUED

'76505' = '1' /* EXTREME IMMATURITY */
'76506' = '1' /* EXTREME IMMATURITY */
'76507' = '1' /* EXTREME IMMATURITY */
'76508' = '1' /* EXTREME IMMATURITY */
'76511' = '1' /* OTHER PRETERM INFANTS */
'76512' = '1' /* OTHER PRETERM INFANTS */
'76513' = '1' /* OTHER PRETERM INFANTS */
'76514' = '1' /* OTHER PRETERM INFANTS */
'76515' = '1' /* OTHER PRETERM INFANTS */
'76516' = '1' /* OTHER PRETERM INFANTS */
'76517' = '1' /* OTHER PRETERM INFANTS */
'76518' = '1' /* OTHER PRETERM INFANTS */
OTHER   = '0' ;

/* Pulmonary embolism */
/* ICD-9-CM diagnosis codes: */

VALUE $PULMOID
'41511' = '1' /* IAGTROGENIC PULMONARY EMBOLISM AND INFARCTION */
```

```
    '41519' = '1' /* OTHER */
    OTHER   = '0' ;

    /* Reclosure of postoperative disruption of abdominal wall */;
    /* ICD-9-CM procedure code: */

    VALUE $RECLOIP
    '5461'  = '1'
    OTHER   = '0' ;

    /* Seizure */
    /* ICD-9-CM diagnosis codes: */


    VALUE $SEIZUID
    '34500' = '1' /* GENERALIZED NONCONVULSIVE EPILEPSY - WITHOUT MENTION OF
                     INTRACTABLE EPILEPSY */
    '34501' = '1' /* GENERALIZED NONCONVULSIVE EPILEPSY - WITH INTRACTABLE
                     EPILEPSY */
    '34510' = '1' /* GENERALIZED CONVULSIVE EPILEPSY - WITHOUT MENTION OF
                     INTRACTABLE EPILEPSY */
    '34511'  =  '1'  /*  GENERALIZED  CONVULSIVE  EPILEPSY  -  WITH  INTRACTABLE
EPILEPSY */
    '3452'  = '1' /* EPILEPSY-PETIT MAL STATUS */
    '3453'  = '1' /* EPILEPSY-GRAND MAL STATUS */
    '34540' = '1' /* PARTIAL EPILEPSY, WITH IMPAIRMENT OF CONSCIOUSNESS - WITH
                     INTRACTABLE EPILEPSY */
    '34541'  =  '1'  /*  PARTIAL  EPILEPSY,  WITH  IMPAIRMENT  OF  CONSCIOUSNESS  -
WITHOUT
                     MENTION OF INTRACTABLE EPILEPSY */
    '34550' = '1' /* PARTIAL EPILEPSY, WITHOUT MENTION OF IMPAIRMENT OF
                     CONSCIOUSNESS, - WITHOUT MENTION OF INTRACTABLE EPILEPSY
*/
    '34551' = '1' /* PARTIAL EPILEPSY, WITHOUT MENTION OF IMPAIRMENT OF
                     CONSCIOUSNESS-WITH INTRACTABLE EPILEPSY */
    '34560'  =  '1'  /*  INFANTILE  SPASMS  -  WITHOUT  MENTION  OF  INTRACTABLE
EPILEPSY */
    '34561' = '1' /* INFANTILE SPASMS - WITH INTRACTABLE EPILEPSY */
    '34570'  =  '1'  /*  EPILEPSIA  PARTIALIS  CONTINUA  -  WITHOUT  MENTION  OF
INTRACTABLE
    PSFMTS -- CONTINUED

                     EPILEPSY */
    '34571' = '1' /* EPILEPSIA PARTIALIS CONTINUA - WITH INTRACTABLE EPILEPSY
    */
    '34580' = '1' /* OTHER FORMS OF EPILEPSY - WITHOUT MENTION OF INTRACTABLE
                     EPILEPSY */
    '34581' = '1' /* OTHER FORMS OF EPILEPSY - WITH INTRACTABLE EPILEPSY */
    '34590' = '1' /* EPILEPSY, UNSPECIFIED - WITHOUT MENTION OF INTRACTABLE
                     EPILEPSY */
    '34591' = '1' /* EPILEPSY, UNSPECIFIED - WITH INTRACTABLE EPILEPSY */
    '78031' = '1' /* FEBRILE CONVULSIONS */
    '78039' = '1' /* OTHER CONVULSIONS */
    '7803'  = '1' /* CONVULSIONS (OLD CODE NO LONGER VALID) */
    OTHER   = '0' ;

    /* Self inflicted injury */
    /* ICD-9-CM diagnosis codes: */
```

```
    VALUE $SELFIID
    'E9500' = '1' /* ANALGESICS, ANTIPYRETICS, AND ANTIRHEUMATICS */
    'E9501' = '1' /* BARBITURATES */
    'E9502' = '1' /* OTHER SEDATIVE AND HYPNOTICS */
    'E9503' = '1' /* TRANQUILIZIERS AND OTHER PSYCHOTROPIC AGENTS */
    'E9504' = '1' /* OTHER SPECIFIED DRUGS AND MEDICINAL SUBSTANCES */
    'E9505' = '1' /* UNSPECIFIED DRUG OR MEDICINAL SUBSTANCE */
    'E9506'   =   '1'   /*   AGRICULTURAL   AND   HORTICULTURAL   CHEMICAL   AND
PHARMACEUTICAL
                   PREPARATIONS OTHER THAN PLANT FOODS AND FERTILIZERS */
    'E9507' = '1' /* CORROSIVE AND CAUSTIC SUBSTANCES */
    'E9508' = '1' /* ARSENIC AND ITS COMPOUNDS */
    'E9509' = '1' /* OTHER AND UNSPECIFIED SOLID AND LIQUID SUBSTANCES */
    'E9510' = '1' /* GAS DISTRIBUTED BY PIPELINE */
    'E9511' = '1' /* LIQUEFIED PETROLEUM GAS DISTRIBUTED IN MOBILE CONTAINERS
*/
    'E9518' = '1' /* OTHERS UTILITY GASES */
    'E9520' = '1' /* MOTOR VEHICLE EXHAUST GAS */
    'E9521' = '1' /* OTHER CARBON MONOXIDE */
    'E9528' = '1' /* OTHER SPECIFIED GASES AND VAPORS */
    'E9529' = '1' /* UNSPECIFIED GASES AND VAPORS */
    'E9530' = '1' /* HANGING */
    'E9531' = '1' /* SUFFOCATION BY PLASTIC BAG */
    'E9538' = '1' /* OTHER SPECIFIED MEANS */
    'E954'    =  '1'  /*  SUICIDE  AND  SELF-INFLICTED  INJURY  BY  SUBMERSION
[DROWNING] */
    'E9550' = '1' /* HANDGUN */
    'E9551' = '1' /* SHOTGUN */
    'E9552' = '1' /* HUNTING RIFLE */
    'E9553' = '1' /* MILITARY FIREARMS */
    'E9554' = '1' /* OTHER AND UNSPECIFIED FIREARMS */
    'E9555' = '1' /* EXPLOSIVES */
    'E9559' = '1' /* UNSPECIFIED */
    'E956'  = '1' /* SUICIDE AND SELF INFLICTED INJURY BY CUTTING AND PIERCING
                   INSTRUMENT */
    'E9570' = '1' /* RESIDENTIAL PREMISES */
    'E9571' = '1' /* OTHER MAN-MADE STRUCTURES */
    'E9572' = '1' /* NATURAL SITES */
    'E9573' = '1' /* UNSPECIFIED */
    'E9580' = '1' /* JUMPING OR LYING BEFORE MOVING OBJECT */
    'E9581' = '1' /* BURNS, FIRE */
    'E9582' = '1' /* SCALD */
```

PSFMTS -- CONTINUED

```
    'E9583' = '1' /* EXTREMES OF COLD */
    'E9584' = '1' /* ELECTROCUTION */
    'E9585' = '1' /* CRASHING OF MOTOR VEHICLE */
    'E9586' = '1' /* CRASHING OF AIRCRAFT */
    'E9587' = '1' /* CAUSTIC SUBSTANCES EXCEPT POISONING */
    'E9588' = '1' /* OTHER SPECIFIED MEANS */
    'E9589' = '1' /* UNSPECIFIED MEANS */
    OTHER   = '0' ;

    /* Septicemia */
    /* ICD-9-CM diagnosis codes: */

    VALUE $SEPTIID
```

```
    '0380'  = '1' /* STREPTOCOCCAL SEPTICEMIA */
    '0381'  = '1' /* STAPHYLOCOCCAL SEPTICEMIA */
    '03810' = '1' /* STAPHYLOCOCCAL SEPTICEMIA, UNSPECIFIED */
    '03811' = '1' /* STAPHYLOCOCCUS AUREUS SEPTICEMIA */
    '03819' = '1' /* OTHER STAPHYLOCOCCAL SEPTICEMIA */
    '0382'    =  '1'  /*  PNEUMOCOCCAL  SEPTICEMIA  (STREPTOCOCCUS  PNEUMONIAE
SEPTICEMIA) *
    /
    '0383'  = '1' /* SEPTICEMIA DUE TO ANAEROBES */
    '03840' = '1' /* GRAM-NEGATIVE ORGANISM, UNSPECIFIED */
    '03841' = '1' /* HEMOPHILUS INFLUENZAE */
    '03842' = '1' /* ESCHERICHIA COLI */
    '03843' = '1' /* PSEUDOMONAS */
    '03844' = '1' /* SERRATIA */
    '03849' = '1' /* SEPTICEMIA DUE TO OTHER GRAM-NEGATIVE ORGANISMS */
    '0388'  = '1' /* OTHER SPECIFIED SEPTICEMIAS */
    '0389'  = '1' /* UNSPECIFIED SEPTICEMIA */
    OTHER   = '0' ;

    /* Shock */
    /* ICD-9-CM diagnosis codes: */

    VALUE $SHOCKID
    '78550' = '1' /* SHOCK, UNSPECIFIED */
    '78551' = '1' /* CARDIOGENIC SHOCK */
    '78559' = '1' /* OTHER */
    OTHER   = '0' ;

    /* Stroke */
    /* ICD-9-CM diagnosis codes: */

    VALUE $STROKID
    '430'   = '1' /* SUBARACHNOID HEMORRHAGE */
    '431'   = '1' /* INTRACEREBRAL HEMORRHAGE */
    '432'   = '1' /* INTRACRANIAL HEM NEC/NOS */
    '4320'  = '1' /* NONTRAUMATIC EXTRADURAL HEMORRHAGE */
    '4321'  = '1' /* SUBDURAL HEMORRHAGE */
    '4329'  = '1' /* UNSPECIFIED INTRACRANIAL HEMORRHAGE */
    '43301' = '1' /* BASILAR ARTERY, WITH CEREBRAL INFARCTION */
    '43311' = '1' /* CAROTID ARTERY, WITH CEREBRAL INFARCTION */
    '43321' = '1' /* VERTEBRAL ARTERY - WITH CEREBRAL INFARCTION */
    '43331' = '1' /* MULTIPLE AND BILATERAL WITH CEREBRAL INFARCTION */
    '43381' = '1' /* OTHER SPECIFIED PRECEBRAL ARTERY WITH CEREBRAL INFARCTION
*/
    '43391'  =  '1'  /*  OCCLUSION  AND  STENOSIS  OF  PRECEREBRAL  ARTERIES,
UNSPECIFIED
    PSFMTS -- CONTINUED

                    PRECEREBRAL ARTERY WITH CEREBRAL INFARCTION */
    '43401' = '1' /* CEREBRAL THROMBOSIS - WITH CEREBRAL INFARCTION */
    '43411' = '1' /* CEREBRAL EMBOLISM - WITH CEREBRAL INFARCTION */
    '43491' = '1' /* CEREBRAL ARTERY OCCLUSION, UNSPECIFIED - WITH CEREBRAL
                    INFARCTION */
    '436'   = '1' /* ACUTE, BUT ILL-DEFINED CEREBROVASCULAR DISEASE */
    OTHER   = '0' ;

    /* Surgical */
    /* Diagnostic Related Groups (DRGs): */
```

```
    VALUE $SURGIDR
    '001' = '1' /* CRANIOTOMY, AGE GREATER THAN 17 EXCEPT FOR TRAUMA */
    '002' = '1' /* CRANIOTOMY FOR TRAUMA, AGE GREATER THAN 17 */
    '003' = '1' /* CRANIOTOMY, AGE 0-17 */
    '004' = '1' /* SPINAL PROCEDURES */
    '005' = '1' /* EXTRACRANIAL VASCULAR PROCEDURES */
    '006' = '1' /* CARPAL TUNNEL RELEASE */
    '007' = '1' /* PERIPHERAL AND CRANIAL NERVE AND OTHER NERVOUS SYSTEM
                   PROCEDURES WITH CC */
    '008' = '1' /* PERIPHERAL AND CRANIAL NERVE AND OTHER NERVOUS SYSTEM
                   PROCEDURES WITHOUT CC */
    '036' = '1' /* RETINAL PROCEDURES */
    '037' = '1' /* ORBITAL PROCEDURES */
    '038' = '1' /* PRIMARY IRIS PROCEDURES */
    '039' = '1' /* LENS PROCEDURES WITH OR WITHOUT VITRECTOMY */
    '040' = '1' /* EXTRAOCULAR PROCEDURES EXCEPT ORBIT, AGE GREATER THAN 17 */
    '041' = '1' /* EXTRAOCULAR PROCEDURES EXCEPT ORBIT, AGE 0-17 */
    '042' = '1' /* INTRAOCULAR PROCEDURES EXCEPT RETINA, IRIS AND LENS */
    '049' = '1' /* MAJOR HEAD AND NECK PROCEDURES */
    '050' = '1' /* SIALOADENECTOMY */
    '051' = '1' /* SALIVARY GLAND PROCEDURES EXCEPT SIALOADENECTOMY */
    '052' = '1' /* CLEFT LIP AND PALATE REPAIR */
    '053' = '1' /* SINUS AND MASTOID PROCEDURES, AGE 0-17 */
    '054' = '1' /* SINUS AND MASTOID PROCEDURES, AGE 0-17 */
    '055' = '1' /* MISCELLANEOUS EAR, NOSE, MOUTH AND THROAT PROCEDURES */
    '056' = '1' /* RHINOPLASTY */
    '057'  =  '1'  /*  TONSILLECTOMY  AND  ADENOIDECTOMY  PROCEDURES  EXCEPT
TONSILLECTOMY
                   AND/OR ADENOIDECTOMY ONLY, AGE GREATER THAN 17 */
    '058'  =  '1'  /*  TONSILLECTOMY  AND  ADNOIDECTOMY  PROCEDURES  EXCEPT
TONSILLECTOMY
                   AND/OR ADENOIDECTOMY ONLY, AGE 0-17 */
    '059' = '1' /* TONSILLECTOMY AND/OR ADENOIDECTOMY ONLY, AGE GREATER THAN
17 */
    '060' = '1' /* TONSILLECTOMY AND/OR ADENOIDECTOMY ONLY, AGE GREATER THAN
17 */
    '061' = '1' /* MYRINGOTOMY WITH TUBE INSERTION, AGE GREATER THAN 17 */
    '062' = '1' /* MYRINGOTOMY WITH TUBE INSERTION, AGE 0-17 */
    '063' = '1' /* OTHER EAR, NOSE, MOUTH AND THROAT OR PROCEDURES */
    '075' = '1' /* MAJOR CHEST PROCEDURES */
    '076' = '1' /* OTHER RESPIRATORY SYSTEM OR PROCEDURES WITH CC */
    '077' = '1' /* OTHER RESPIRATORY SYSTEM OR PROCEDURES WITHOUT CC */
    '103' = '1' /* HEART TRANSPLANT */
    '104'  =  '1'  /*  CARDIAC  VALVE  AND  OTHER  MAJOR  CARDIOTHORACIC  PROCEDURES
WITH
                   CARDIAC CATHETERIZATION */
    '105' = '1' /* CARDIAC VALVE AND OTHER MAJOR CARDIOTHORACIC PROCEDURES
                   WITHOUT CARDIAC CATHETERIZATION */
```

PSFMTS -- CONTINUED

```
    '106' = '1' /* CORONARY BYPASS WITH PTCA */
    '107' = '1' /* CORONARY BYPASS WITH CARDIAC CATHETERIZATION */
    '108' = '1' /* OTHER CARDIOTHORACIC PROCEDURES */
    '109' = '1' /* CORONARY BYPASS WITHOUT CARDIAC CATHETERIZATION */
    '110' = '1' /* MAJOR CARDIOVASCULAR PROCEDURES WITH CC */
    '111' = '1' /* MAJOR CARDIOVASCULAR PROCEDURES WITH CC */
    '112' = '1' /* PERCUTANEOUS CARDIOVASCULAR PROCEDURES */
```

```
    '113' = '1' /* AMPUTATION FOR CIRCULATORY SYSTEM DISORDERS EXCEPT UPPER
LIMB
                   AND TOE */
    '114' = '1' /* UPPER LIMB AND TOES AMPUTATION FOR CIRCULATORY SITE */
    '115' = '1' /* PERMANENT CARDIAC PACEMAKER IMPLANT WITH ACUTE MYOCARDIAL
                   INFARCTION, HEART FAILURE OR SHOCK OR ACID LEAD OR
                   GENERATOR PROCEDURE */
    '116' = '1' /* OTHER PERMANENT CARDIAC PACEMAKER IMPLANT OR PTCA WITH
                   CORONARY ARTERIAL STENT */
    '117' = '1' /* CARDIAC PACEMAKER REVISION EXCEPT DEVICE REPLACEMENT */
    '118' = '1' /* CARDIAC PACEMAKER DEVICE REPLACEMENT */
    '119' = '1' /* VEIN LIGATION AND STRIPPING */
    '120' = '1' /* OTHER CIRCULATORY SYSTEM OR PROCEDURES */
    '146' = '1' /* RECTAL RESECTION WITH CC */
    '147' = '1' /* RECTAL RESECTION WITHOUT CC */
    '148' = '1' /* MAJOR SMALL AND LARGE BOWEL PROCEDURES WITH CC */
    '149' = '1' /* MAJOR SMALL AND LARGE BOWEL PROCEDURES WITHOUT CC */
    '150' = '1' /* PERITONEAL ADHESIOLYSIS WITH CC */
    '151' = '1' /* PERITONEAL ADHESIOLYSIS WITHOUT CC */
    '152' = '1' /* MINOR SMALL AND LARGE BOWEL PROCEDURES WITH CC */
    '153' = '1' /* MINOR SMALL AND LARGE BOWEL PROCEDURES WITHOUT CC */
    '154' = '1' /* STOMACH, ESOPHAGEAL AND DUODENAL PROCEDURES, AGE GREATER
                   THAN 17 WITH CC */
    '155' = '1' /* STOMACH, ESOPHAGEAL AND DUODENAL PROCEDURES, AGE GREATER
                   THAN 17 WIHOUT CC*/
    '156' = '1' /* STOMACH, ESOPHAGEAL AND DUODENAL PROCEDURES, AGE 0-17 */
    '157' = '1' /* ANAL AND STOMAL PROCEDURES WITH CC */
    '158' = '1' /* ANAL AND STOMAL PROCEDURES WITHOUT CC */
    '159' = '1' /* HERMIA PROCEDURES EXCEPT INGUINAL AND FEMORAL , AGE GREATER
                   THAN 17 WITHOUT CC */
    '160' = '1' /* HERNIA PROCEDURES EXCEPT INGUINAL AND FEMORAL, AGE GREATER
                   THAN 17 WITHOUT CC */
    '161' = '1' /* INGUINAL AND FEMORAL HERNIA PROCEDURES, AGE GREATER TAN 17
                   WITH CC */
    '162' = '1' /* INGUINAL AND FEMORAL HERNIA PROCEDURES, AGE GREATER TAN 17
                   WITHOUT CC */
    '163' = '1' /* HERNIA PROCEDURES, AGE 0-17 */
    '164' = '1' /* APPENDECTOMY WITH COMPLICATED PRINCIPAL DIAGNOSIS WITH CC
*/
    '165' = '1' /* APPENDECTOMY WITH COMPLICATED PRINCIPAL DIAGNOSIS WIHTOUT
CC */
    '166' = '1' /* APPENDECTOMY WITHOUT COMPLICATED PRINCIPAL IAGNOSIS WITH CC
*/
    '167' = '1' /* APPENDECTOMY WITHOUT COMPLICATED PRINCIPAL DIAGNOSIS
                   WITHOUT CC */
    '168' = '1' /* MOUTH PROCEDURES WITH CC */
    '169' = '1' /* MOUTH PROCEDURES WITHOUT CC */
    '170' = '1' /* OTHER DIGESTIVE SYSTEM OR PROCEDURES WITH CC */
    '171' = '1' /* OTHER DIGESTIVE SYSTEM OR PROCEDURES WITHOUT CC */
    '191' = '1' /* PANCREAS, LIVER AND SHUNT PROCEDURES WITH CC */
    '192' = '1' /* PANCREAS, LIVER AND SHUNT PROCEDURES WITHOUT CC */
```

PSFMTS -- CONTINUED

```
    '193' = '1' /* BILIARY TRACT PROCEDURES EXCEPT ONLY CHOLECYSTECTOIMY WITH
OR
                   WITHOUT COMMON DUCT EXPLORATION WITH CC */
```

```
    '194' = '1' /* BILIARY TRACT PROCEDURES EXCEPT ONLY CHOLECYSTECTOMY WITH
OR
                   WITHOUT COMMON DUCT EXPLORATION WITHOUT CC */
    '195' = '1' /* CHOLECYSTECTOMY WITH COMMON DUCT EXPLORATION WITH CC */
    '196' = '1' /* CHOLECYSTECTOMY WITH COMMON DUCT EXPLORATION WITHOUT CC */
    '197' = '1' /* CHOLECYSTECTOMY EXCEPT BY LAPAROSCOPE WITHOUT COMMON DUCT
                   EXPLORATION WITH CC */
    '198' = '1' /* CHOLECYSTECTOMY EXCEPT BY LAPAROSCOPE WITHOUT COMMON DUCT
                   EXPORTATION WITHOUT CC */
    '199' = '1' /* HEPATOBILIARY DIAGNOSTIC PROCEDURE FOR MALIGNANCY */
    '200' = '1' /* HEPATOBILIARY DIAGNOSTIC PROCEDURE FOR NONMALIGNANCY */
    '201' = '1' /* OTHER HEPATOBILIARY OR PANCREAS OR PROCEDURES */
    '209' = '1' /* MAJOR JOINT AND LIMB REATTACHMENT PROCEDURES OF LOWER
                   EXTREMITY */
    '210' = '1' /* HIP AND FEMUR PROCEDURES EXCEPT MAJOR JOINT PROCEDURES,
                   AGE GREATER THAN 17 WITH CC */
    '211' = '1' /* HIP AND FEMUR PROCEDURES EXCEPT MAJOR JOINT PROCEDURES,
                   AGE GREATER THAN 17 WITHOUT CC */
    '212' = '1' /* HIP AND FEMUR PROCEDURES EXCEPT MAJOR JOINT PROCEDURE,
                   AGE 0-17 */
    '213' = '1' /* AMPUTATION FOR MUSCULOSKELETAL SYSTEM AND CONNECTIVE TISSUE
                   DISORDERS */
    '214' = '1' /* NO LONGER VALID */
    '215' = '1' /* NO LONGER VALID */
    '216' = '1' /* BIOPSIES OF MUSCULOSKELETAL SYSTEM AND CONNECTIVE TISSUE */
    '217'  =  '1'  /*  WOUND  DEBRIDEMENT  AND  SKIN  GRAFT  EXCEPT  HAND  FOR
MUSCULOSKELETAL
                   AND CONNECTIVE TISSUE DISORDERS */
    '218' = '1' /* LOWER EXTREMITY AND HUMERUS PROCEURES EXCEPT HIP, FOOT AND
                   FEMUR, AGE GREATER THAN 17 WITH CC */
    '219' = '1' /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT AND
                   FEMUR, AGE GREATER THAN 17 WITHOUT CC */
    '220' = '1' /* LOWER EXTREMITY AND HUMERUS PROCEDURES EXCEPT HIP, FOOT AND
                   FEMUR, AGE 0-17 */
    '221' = '1' /* NO LONGER VALID */
    '222' = '1' /* NO LONGER VALID */
    '223' = '1' /* MAJOR SHOULDER/ELBOW PROCEDURES OR OTHER UPPER EXTREMITY
                   PROCEDURES WITH CC */
    '224' = '1' /* SHOULDER, ELBOW OR FOREARM PROCEDURES EXCEPT MAJOR JOINT
                   PROCEDURES WITHOUT CC */
    '225' = '1' /* FOOT PROCEDURES*/
    '226' = '1' /* SOFT TISSUE PROCEDURES WITH CC */
    '227' = '1' /* SOFT TISSUE PROCEDURES WITHOUT CC */
    '228' = '1' /* MAJOR THUMB OR JOINT PROCEDURES OR OTHER HAND OR WRIST
                   PROCEDURES WITH CC */
    '229' = '1' /* HAND OR WRIST PROCEDURES EXCEPT MAJOR JOINT PROCEDURES
                   WITHOUT CC */
    '230' = '1' /* LOCAL EXCISION AND REMOVAL OF INTERNAL FIXATION DEVICES
                   OF HIP AND FEMUR */
    '231' = '1' /* LOCAL EXCISION AND REMOVAL OF INTERNAL FIXATION DEVICES
                   EXCEPT HIP AND FEMUR */
    '232' = '1' /* ARTHROSCOPY */
    '233' = '1' /* OTHER MUSCULOSKELETAL SYSTEM AND CONNECTIVE TISSUE OR
                   PROCEDURES WITH CC */
    '234' = '1' /* OTHER MUSCULOSKELETAL SYSTEM AND CONNECTIVE TISSUE OR
                   PROCEDURES WITHOUT CC */
```

PSFMTS -- CONTINUED

```
    '257' = '1' /* TOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '258' = '1' /* TOTAL MASTECTOMY FOR MALIGNANCY WITHOUT CC */
    '259' = '1' /* SUBTOTAL MASTECTOMY FOR MALIGNANCY WITH CC */
    '260' = '1' /* SUBTOTAL MASTECTOMY FOR MALIGNANCY WITHOUT CC */
    '261' = '1' /* BREAST PROCEDURE FOR NONMALIGNANCY EXCEPT BIOPSY AND
                   LOCAL EXCISION */
    '262' = '1' /* BREAST BIOPSY AND LOCAL EXCISION FOR NONMALIGNANCY */
    '263' = '1' /* SKIN GRAFT AND/OR DEBRIDEMENT FOR SKIN ULCER OR
                   CELLULITIS WITH CC */
    '264' = '1' /* SKIN GRAFT AND OR DEBRIDEMENT FOR SKIN ULCER OR
                   CELLULITIS WITHOUT CC */
    '265' = '1' /* SKIN GRAFT AND OR DEBRIDEMENT EXCEPT FOR SKIN ULCER OR
                   CELLULITIS WITH CC */
    '266' = '1' /* SKIN GRAFT AND/OR DEBRIDEMENT EXCEPT FOR SKIN ULCER OR
                   CELLUTLITIES WITHOUT CC */
    '267' = '1' /* PERIANAL AND PILONIDAL PROCEDURES */
    '268' = '1' /* SKIN, SUBCUTANEOUS TISSUE AND BREAST PLASTIC PROCEDURES */
    '269' = '1' /* OTHER SKIN, SUBCUTANEOUS TISSUE AND BREAST PROCEDURES
                   WITH CC */
    '270' = '1' /* OTHER SKIN, SUBCUTANEOUS TISSUE AND BREAST PROCEDURS
                   WITHOUT CC */
    '285' = '1' /* AMPUTAETION OF LOWER LIMB FOR ENDOCRINE, NUTRITIONAL AND
                   METABOLIC DISORDERS */
    '286' = '1' /* ADRENAL AND PITUITARY PROCEDURES */
    '287' = '1' /* SKIN GRAFTS AND WOUND DEBRIDEMENTS FOR ENDOCRINE,
NUTRITIONAL
                   AND METABOLIC DISORDERS */
    '288' = '1' /* OR PROCEDURES FOR OBESITY */
    '289' = '1' /* PARATHYROID PROCEDURES */
    '290' = '1' /* THYROID PROCEDURES */
    '291' = '1' /* THYROGLOSSAL PROCEDURES */
    '292' = '1' /* OTHER ENDOCRINE, NUTRITIONAL AND METABOLIC OR PROCEDURES
                   WITH CC */
    '293' = '1' /* OTHER ENDOCRINE, NUTRITIONAL AND METABOLIC OR PROCEDURES
                   WITHOUT CC */
    '302' = '1' /* KIDNEY TRANSPLANT */
    '303' = '1' /* KIDNEY, URETER AND MAJOR BLADDER PROCEDURES FOR NEOPLASM */
    '304' = '1' /* KIDNEY, URETER AND MAJOR BLADDER PROCEDURES FOR
                   NONNEOPLASMS WITH CC */
    '305' = '1' /* KIDNEY, URETER AND MAJOR BLADDER PROCEDURES FOR
                   NONEOPLSMS WITHOUT CC */
    '306' = '1' /* PROSTATECTOMY WITH CC */
    '307' = '1' /* PROSTATECTOMY WITHOUT CC */
    '308' = '1' /* MINOR BLADDER PROCEDURES WITH CC */
    '309' = '1' /* MINOR BLADDER PROCEDURES WITHOUT CC */
    '310' = '1' /* TRANSURETHRAL PROCEDURES WITH CC */
    '311' = '1' /* TRANSURETHRAL PROCEDURES WITHOUT CC */
    '312' = '1' /* URETHRAL PROCEDURES, AGE GREATER THAN 17 WITH CC */
    '313' = '1' /* URETHRAL PROCEDURES, AGE GREATER THAN 17 WITHOUT CC */
    '314' = '1' /* URETHRAL PROCEDURES, AGE 0-17 */
    '315' = '1' /* OTHER KIDNEY AND URINARY TRACT OR PROCEDURES */
    '334' = '1' /* MAJOR MALE PELVIC PROCEDURES WITH CC */
    '335' = '1' /* MAJOR MALE PELVIC PROCEDURES WITHOUT CC */
    '336' = '1' /* TRANSURETHRAL PROSTATECTOMY WITH CC */
    '337' = '1' /* TRANSURETHRAL PROSTATECTOMY WITHOUT CC */
    '338' = '1' /* TESTES PROCEDURES FOR MALIGNANCY */
    '339' = '1' /* TESTES PROCEDURES FOR NONMALIGNANCY, AGE GREATER THAN 17 */
    '340' = '1' /* TESTES PROCEDURES FOR NONMALIGNANCY, AGE 0-17 */
```

```
    '341' = '1' /* PENIS PROCEDURES */
```
PSFMTS -- CONTINUED
```
    '342' = '1' /* CIRCUMCISION, AGE GREATER THAN 17 */
    '343' = '1' /* CIRCUMCISION, AGE 0-17 */
    '344' = '1' /* OTHER MALE REPRODUCTIVE SYSTEM OR PROCEDURES FOR MALIGNANCY
*/
    '345' = '1' /* OTHER MALE REPRODUCTIVE SYSTEM OR PROCEDURES EXCEPT FOR
                   MALIGNANCY */
    '353' = '1' /* PELVIC EVISCERATION, RADICAL HYSTERECTOMY AND RADICAL
                   VULVECTOMY */
    '354' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONOVARIAN/ADNEXAL
                   MALIGNANCY WITH CC */
    '355' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONOVARIAN/ADNEXA
                   PROCEDURES FOR NONOVARIAN/ADNEXAL MALIGNANCY WITHOUT CC */
    '356' = '1' /* FEMALE REPRODUCTIVE SYSTEM RECONSTRUCTIVE PROCEDURES */
    '357' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR OVARIAN OR ADNEXAL
                   MALIGNANCY */
    '358' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITH CC */
    '359' = '1' /* UTERINE AND ADNEXA PROCEDURES FOR NONMALIGNANCY WITHOUT CC
*/
    '360' = '1' /* VAGINA, CERVIX AND VULVA PROCEDURES */
    '361' = '1' /* LAPAROSCOPY AND INCISIONAL TUBAL INTERRUPTION */
    '362' = '1' /* ENDOSCOPIC TUBAL INTERRUPTION */
    '363' = '1' /* D AND C, CONIZATION AND RADIOIMPLANT FOR MALIGNANCY */
    '364' = '1' /* D AND C, CONIZATION EXCEPT FOR MALIGNANCY */
    '365' = '1' /* OTHER FEMALE REPRODUCTIVE SYSTEM OR PROCEDURES */
    '370' = '1' /* CESAREAN SECTION WITH CC */
    '371' = '1' /* CESAREAN SECTION WITHOUT CC */
    '374' = '1' /* VAGINAL DELIVERY WITH STERILIZATION AND/OR D AND C */
    '375' = '1' /* VAGINAL DELIVERY WITH OR PROCEDURE EXCEPT STERILIZATION
                   AND/OR D AND C */
    '377' = '1' /* POSTPARTUM AND POSTABORTION DIAGNOSES WITH OR PROCEDURE */
    '381' = '1' /* ABORTION WITH D AND C ASPIRATION CURETTAGE OR HYSTERECTOMY
*/
    '392' = '1' /* SPLENECTOMY, AGE GREATER THAN 17 */
    '393' = '1' /* SPLENECTOMY, AGE 0-17 */
    '394' = '1' /* OTHER OR PROCEDURES OF THE BLOOD AND BLOOD-FORMING ORGANS
*/
    '400' = '1' /* LYMPHOMA AND LEUKEMIA WITH MAJOR OR PROCEDURES */
    '401' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH OTHER OR PROCEDURE
                   WITH CC */
    '402' = '1' /* LYMPHOMA AND NONACUTE LEUKEMIA WITH OTHER OR PROCEDURE
                   WITHOUT CC */
    '406' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH MAJOR OR PROCEDURES WITH CC */
    '407' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH MAJOR OR PROCEDURES WITHOUT CC */
    '408' = '1' /* MYELOPROLIFERATIVE DISORDERS OR POORLY DIFFERENTIATED
                   NEOPLASMS WITH OTHER OR PROCEDURES */
    '415' = '1' /* OR PROCEDURE FOR INFECTIOUS AND PARASITIC DISEASES */
    '424' = '1' /* OR PROCEDURES WITH PRINCIPAL DIAGNOSIS OF MENTAL ILLNESS */
    '439' = '1' /* SKIN GRAFTS FOR INJURIES */
    '440' = '1' /* WOUND DEBRIDEMENTS FOR INJURIES */
    '441' = '1' /* WOUND HAND PROCEDURES FOR INJURIES */
    '442' = '1' /* OTHER OR PROCEDURES FOR INJURIES WITH CC */
    '443' = '1' /* OTHER OR PROCEDURES FOR INJURIES */
    '458' = '1' /* NO LONGER VALID */
```

```
    '459' = '1' /* NO LONGER VALID */
    '461' = '1' /* OR PROCEDURES WITH DIAGNOSES OF OTHER CONTACT WITH HEALTH
                   SERVICES */
    '468' = '1' /* EXTENSIVE OR PROCEDURE UNRELATED TO PRINCIPAL DIAGNOSIS */
```
PSFMTS -- CONTINUED
```
    '471' = '1' /* BILATERAL OR MULTIPLE MAJOR JOINT PROCEDURES OF LOWER
                   EXTREMITY */
    '472' = '1' /* NO LONGER VALID */
    '476' = '1' /* PROSTATIC OR PROCEDURE UNRELATED TO PRINCIPAL DIAGNOSIS */
    '477' = '1' /* NONEXTENSIVE OR PROCEDURE UNRELATED TO PRINCIPAL DIAGNOSIS
*/
    '478' = '1' /* OTHER VASCULAR PROCEDURES WITH CC */
    '479' = '1' /* OTHER VASCULAR PROCEDURES WITHOUT CC */
    '480' = '1' /* LIVER TRANSPLANT */
    '481' = '1' /* BONE MARROW TRANSPLANT */
    '482' = '1' /* TRACHEOSTOMY FOR FACE, MOUTH AND NECK DIAGNOSES */
    '483' = '1' /* TRACHEOSTOMY EXCEPT FOR FACE, MOUTH AND NECK DIAGNOSES */
    '484' = '1' /* CRANIOTOMY FOR MULTIPLE SIGNIFICANT TRAUMA */
    '485' = '1' /* LIMB REATTACHMENT, HIP AND FEMUR PROCEDURES FOR MULTIPLE
                   SIGNIFICANT TRAUMA */
    '486' = '1' /* OTHER OR PROCEDURES FOR MULTIPLE SIGNIFICANT TRAUMA */
    '488' = '1' /* HIV WITH EXTENSIVE OR PROCEDURE */
    '491' = '1' /* MAJOR JOINT AND LIMB REATTACHMENT PROCEDURES OF UPPER
                   EXTREMITY */
    '493'  =  '1'  /*  LAPAROSCOPIC  CHOLECYSTECTOMY  WITHOUT  COMMON  DUCT
EXPLORATION
                   WITH CC */
    '494'  =  '1'  /*  LAPAROSCOPIC  CHOLECYSTECTOMY  WITHOUT  COMMON  DUCT
EXPLORATION
                   WITHOUT CC */
    '495' = '1' /* LUNG TRANSPLANT */
    '496' = '1' /* COMBINED ANTERIOR/POSTERIOR SPINAL FUSION */
    '497' = '1' /* SPINAL FUSION WITH CC */
    '498' = '1' /* SPINAL FUSION WITHOUT CC */
    '499' = '1' /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITH CC */
    '500' = '1' /* BACK AND NECK PROCEDURES EXCEPT SPINAL FUSION WITHOUT CC */
    '501' = '1' /* KNEE PROCEDURES WITH PRINCIPAL DIAGNOSIS OF INFECTION,
                   WITH CC */
    '502' = '1' /* KNEE PROCEDURES WITH PRINCIPAL DIAGNOSIS OF INFECTION,
                   WITHOUT CC */
    '503' = '1' /* KNEE PROCEDURES WITHOUT PRINCIPAL DIAGNOSIS OF INFECTION */
    OTHER = '0' ;

    /* Syncope */
    /* ICD-9-CM diagnosis codes: */

    VALUE $SYNCOID
    '7802'  = '1' /* SYNCOPE AND COLLAPSE */
    OTHER   = '0' ;

    /* Tecbnical difficulty */
    /* ICD-9-CM diagnosis codes: */

    VALUE $TECHNID
    '9982'  = '1' /* ACCIDENTAL PUNCTURE OR LACERATION DURING A PROCEDURE */
    'E870'  = '1' /* ACCIDENTAL CUT, PUNCTURE, PERFORATION, OR HEMORRHAGE
                     DURING MEDICAL CARE */
```

```
    'E8700' = '1' /* ACCIDENTAL CUT, PUNCTURE, PERFORATION, OR HEMORRHAGE
                     DURING MEDICAL CARE, SURGICAL OPERATION */
    'E8701' = '1' /* INFUSION OR TRANSFUSION */
    'E8702' = '1' /* KIDNEY DIALYSIS OR OTHER PERFUSION */
    'E8703' = '1' /* INJECTION OR VACCINATION */
    'E8704' = '1' /* ENDOSCOPIC EXAMINATION */
```
PSFMTS -- CONTINUED
```
    'E8705'  =  '1'  /*  ASPIRATION  OF  FLUID  OR  TISSUE,  PUNCTURE,  AND
CATHETERIZATION */
    'E8706' = '1' /* HEART CATHETERIZATION */
    'E8707' = '1' /* ADMINISTRATION OF ENEMA */
    'E8708' = '1' /* OTHER SPECIFIED MEDICAL CARE */
    'E8709' = '1' /* ACCIDENTAL CUT, PUNCTURE, PERFORATION, OR HEMORRHAGE
DURING
                     MEDICAL CARE, UNSPECIFIED MEDICAL CARE */
    OTHER   = '0' ;

    /* Thoracic surgery */
    /* ICD-9-CM procedure codes: */

    VALUE $THORAIP
    '3121' = '1' /* MEDIASTINAL TRACHEOSTOMY */
    '3145' = '1' /* OPEN BIOPSY OF LARYNX OR TRACHEA */
    '3173' = '1' /* CLOSURE OF OTHER FISTULA OF TRACHEA */
    '3179' = '1' /* OTHER REPAIR AND PLASTIC OPERATIONS ON TRACHEA */
    '3199' = '1' /* OTHER OPERATIONS ON TRACHEA */
    '3209' = '1' /* OTHER LOCAL EXCISION OR DESTRUCTION OF LESION OR TISSUE OF
                    BRONCHUS */
    '321'  = '1' /* OTHER EXCISION OF BRONCHUS */
    '3221' = '1' /* PLICATION OF EMPHYSEMATIOUS BLEB */
    '3222' = '1' /* LUNG VOLUME REDUCTION SURGERY */
    '3228' = '1' /* ENDOSCOPIC EXCISION OR DETRUCTION OF LESION OR TISSUE OF
LUNG */
    '3229' = '1' /* OTHER LOCAL EXCISION OR DESTRUCTION OF LESION OR TISSUE OF
                    LUNG */
    '323'  = '1' /* SEGMENTAL RESECTION OF LUNG */
    '324'  = '1' /* LOBECTOMY OF LUNG */
    '325'  = '1' /* COMPLETE PNEUMONECTOMY */
    '326'  = '1' /* RADICAL DISSECTION OF THORACIC STRUCTURES */
    '329'  = '1' /* OTHER EXCISION OF LUNG */
    '330'  = '1' /* INCISION OF BRONCHUS */
    '331'  = '1' /* INCISION OF LUNG */
    '3325' = '1' /* OPEN BIOPSY OF BRONCHUS */
    '3326' = '1' /* CLOSE [PERCUTANEOUS][NEEDLE] BIOPSY OF LUNG */
    '3327' = '1' /* CLOSED ENDOSCOPIC BIOPSY OF LUNG */
    '3328' = '1' /* OPEN BIOPSY OF LUNG */
    '3331' = '1' /* DESTRUCTION OF PHRENIC NERVE FOR COLLAPSE OF LUNG (NO
LONGER
                    PERFORMED) */
    '3332' = '1' /* ARTIFICAL PNEUMOTHORAX FOR COLLAPSE OF LUNG */
    '3334' = '1' /* THORACOPLASTY */
    '3339' = '1' /* OTHER SURGICAL COLLAPSE OF LUNG */
    '3341' = '1' /* SUTURE OF LACERATION OF BRONCHUS */
    '3342' = '1' /* CLOSURE OF BRONCHIAL FISTULA */
    '3343' = '1' /* CLOSURE OF LACERATION OF LUNG */
    '3348' = '1' /* OTHER REPAIR OAND PLASTIC OPERATIONS ON BRONCHUS */
    '3349' = '1' /* OTHER REPAIR AND PLASTIC OPERATIONS ON LUNG */
```

```
    '3350' = '1' /* LUNG TRANSPLANTATION, NOS */
    '3351' = '1' /* UNILATERAL LUNG TRANSPLANTATION */
    '3352' = '1' /* BILATERAL LUNG TRANSPLANTATION */
    '336'  = '1' /* COMBINED HEART-LUNG TRANSPLANTATION */
    '3392' = '1' /* LIGATION OF BRONCHUS */
    '3393' = '1' /* PUNCTURE OF LUNG */
    '3398' = '1' /* OTHER OPERATIONS ON BRONCHUS */
    '3399' = '1' /* OTHER OPERATIONS ON LUNG */
    '3329' = '1' /* OTHER DIAGNOSTIC PROCEDURE ON LUNG AND BRONCHUS */
    '3333' = '1' /* PNEUMOPERITONEUM FOR COLLAPSE OF LUNG */
```

PSFMTS -- CONTINUED

```
    '3401' = '1' /* INCISION OF CHEST WALL */
    '3402' = '1' /* EXPLORATORY THORACOTOMY */
    '3403' = '1' /* REOPENING OF RECENT THORACOTOMY SITE */
    '3405' = '1' /* CREATION OF PLEUROPERITONEAL SHUNT */
    '3409' = '1' /* OTHER INCISION OF PLEURA */
    '341'  = '1' /* INCISION OF MEDIASTINUM */
    '3421' = '1' /* TRANSPLEURAL THORACOSOCOPY */
    '3422' = '1' /* MEDIASTINOSCOPY */
    '3423' = '1' /* BIOPSY OF CHEST WALL */
    '3424' = '1' /* PLEURAL BIOPSY */
    '3425' = '1' /* CLOSED [PERCUTANEOUS][NEEDLE] BIOPSY OF MEDIASTINUM */
    '3426' = '1' /* OPEN BIOPSY OF MEDIASTINUM */
    '3427' = '1' /* BIOPSY OF DIAPHRAGM */
    '3428' = '1' /* OTHER DIAGNOSTIC PROCEDURES ON CHEST WALL, PLEURA, AND
                    DIAPHRAGM */
    '3429' = '1' /* OTHER DIAGNOSTIC PROCEDURES ON MEDIASTINUM */
    '343'  = '1' /* EXCISION OR DESTRUCTION OF LESION OR TISSUE OF MEDIASTINUM
*/
    '344'  = '1' /* EXCISION OR DESTRUCTION OF LESION OF CHEST WALL */
    '345'  = '1' /* PLEURECTOMY */
    '3451' = '1' /* DECORTICATION OF LUNG */
    '3459' = '1' /* OTHER EXCISION OF PLEURA */
    '3471' = '1' /* SUTURE OF LACERATION OF CHEST WALL */
    '3472' = '1' /* CLOSURE OF THORACOSTOMY */
    '3473' = '1' /* CLOSURE OF OTHER FISTULA OF THORAX */
    '3474' = '1' /* REPAIR OF PECTUS DEFORMITY */
    '3479' = '1' /* OTHER REPAIR OF CHEST WALL */
    '3481' = '1' /* EXCISION OF LESION OR TISSUE OF DIAPHRAGM */
    '3482' = '1' /* SUTURE OF LACERATION OF DIAPHRAGM */
    '3483' = '1' /* CLOSURE OF FISTULA OF DIAPHRAGM */
    '3484' = '1' /* OTHER REPAIR OF DIAPHRAGM */
    '3485' = '1' /* IMPLANTATION OF DIAPHRAGMATIC PACEMAKER */
    '3489' = '1' /* OTHER OPERATIONS ON DIAPHRAGM */
    '3493' = '1' /* REPAIR OF PLEURA */
    '3499' = '1' /* OTHER */
    '4061' = '1' /* CANNULATION OF THORACIC DUCT */
    '4062' = '1' /* FISTULIZATION OF THORACIC DUCT */
    '4063' = '1' /* CLOSURE OF FISTULA OF THORACIC DUCT */
    '4064' = '1' /* LIGATION OF THORACIC DUCT */
    '4069' = '1' /* OTHER OPERATIONS ON THORACIC DUCT */
    '4201' = '1' /* INCISION OF ESOPHAGEAL WEB */
    '4209' = '1' /* OTHER INCISION OF EXOPHAGUS */
    '4210' = '1' /* ESOPHAGOSTOMY, NOS */
    '4211' = '1' /* CERVICAL ESOPHAGOSTOMY */
    '4212' = '1' /* EXTERIORIZATION OF ESOPHAGEAL POUCH */
    '4219' = '1' /* OTHER EXTERNAL FISTULIZATION OF ESOPHAGUS */
```

```
    '4221' = '1' /* OPERATIVE ESOPHAGOSCOPY BY INCISION */
    '4225' = '1' /* OPEN BIOPSY OF ESOPHAGUS */
    '4231' = '1' /* LOCAL EXCISION OF ESOPHAGEAL DIVERTICULUM */
    '4232' = '1' /* LOCAL EXCISION OF OTHER LESION OR TISSUE OF ESOPHAGUS */
    '4239' = '1' /* OTHER DESTRUCTION OF LESION OR TISSUE OF ESOPHAGUS */
    '4240' = '1' /* ESOPHAGECTOMY, NOS */
    '4241' = '1' /* PARTIAL ESOPHAGECTOMY */
    '4242' = '1' /* TOTAL ESOPHAGECTOMY */
    '4251' = '1' /* INTRATHORACIC ESOPHAGOESOPHAGOSTOMY */
    '4252' = '1' /* INTRATHORACIC ESOPHAGOGASTROSTOMY */
    '4253' = '1' /* INTRATHORACIC ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    SMALL BOWEL */
    '4254' = '1' /* OTHER INTRATHORACIC ESOPHAGOENTEROSOTOMY */
    PSFMTS -- CONTINUED

    '4255' = '1' /* INTRATHORACIC ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                   COLON */
    '4256' = '1' /* OTHER INTRATHORACIC ESOPHAGOCOLOSTOMY */
    '4258'  =  '1'  /*  INTRATHORACIC  ESOPHAGEAL  ANASTOMOSIS  WITH  OTHER
INTERPOSITION */
    '4259' = '1' /* OTHER INTRATHORACIC ANASTOMOSIS OF ESOPHAGUS */
    '4261' = '1' /* ANTESTERNAL ESOPHAGOESOPHAGOSTOMY */
    '4262' = '1' /* ANTESTERNAL ESOPHAGOGASTROSTOMY */
    '4263' = '1' /* ANTESTERNAL ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    SMALL BOWEL */
    '4264' = '1' /* OTHER ANTESTERNAL ESOPHAGOENTEROSTOMY */
    '4265' = '1' /* ANTESTERNAL ESOPHAGEAL ANASTOMOSIS WITH INTERPOSITION OF
                    COLON */
    '4266' = '1' /* OTHER ANTESTERNAL ESOPHAGOCOLOSTOMY */
    '4268'  =  '1'  /*  OTHER  ANTESTERNAL  ESOPHAGEAL  ANASTOMOSIS  WITH
INTERPOSISION */
    '4269' = '1' /* OTHER ANTESTERNAL ANASTOMOSIS OF ESOPHAGUS */
    '427'  = '1' /* ESOPHAGOMYOTOMY */
    '4281' = '1' /* INSERTION OF PERMANENT TUBE INTO ESOPHAGUS */
    '4282' = '1' /* SUTURE OF LACERATION OF ESOPHAGUS */
    '4283' = '1' /* CLOSURE OF ESOPHAGOSTOMY */
    '4284' = '1' /* REPAIR OF ESOPHAGEAL FISTULA, NEC */
    '4285' = '1' /* REPAIR OF ESOPHAGEAL STRICTURE */
    '4286' = '1' /* PRODUCTION OF SUBCUTANEOUS TUNNEL WITHOUT ESOPHAGEAL
                    ANASTOMOSIS */
    '4287' = '1' /* OTHER GRAFT OF ESOPHAGUS */
    '4289' = '1' /* OTHER REPAIR OF ESOPHAGUS */
    '4465' = '1' /* ESOPHAGOGASTROPLASTY */
    '4466'  =  '1'  /*  OTHER  PROCEDURES  FOR  CREATION  OF  ESOPHAGOGASTRIC
SPHINTERIC
                     COMPETENCE */
    '8104' = '1' /* DORSAL AND DORSO-LUMBAR FUSION, ANTERIOR TECHNIQUE */
    OTHER  = '0' ;

    /* Transfusion reaction */
    /* ICD-9-CM diagnosis codes: */

    VALUE $TRANFID
    '9996'  = '1' /* ABO INCOMPATIBILITY REACTION */
    '9997'  = '1' /* RH INCOMPATIBILITY REACTION */
    'E8760' = '1' /* MISMATCHED BLOOD IN TRANSFUSION */
    OTHER   = '0' ;
```

```
/* Trauma */
/* ICD-9-CM diagnosis codes ( includes 4th and 5th digits): */

VALUE $TRAUMID
'800'   = '1' /* SKULL VAULT FRACTURE */
'8000'  = '1' /* CLOSED SKULL VAULT FX */
'80000' = '1' /* CLOSED SKULL VAULT FX */
'80001' = '1' /* CL SKULL VLT FX W/O COMA */
'80002' = '1' /* CL SKULL VLT FX-BRF COMA */
'80003' = '1' /* CL SKULL VLT FX-MOD COMA */
'80004' = '1' /* CL SKL VLT FX-PROLN COMA */
'80005' = '1' /* CL SKUL VLT FX-DEEP COMA */
'80006' = '1' /* CL SKULL VLT FX-COMA NOS */
'80009' = '1' /* CL SKL VLT FX-CONCUS NOS */
'8001'  = '1' /* CL SKL VLT FX/CEREBR LAC */
'80010' = '1' /* CL SKL VLT FX/CEREBR LAC */
```
PSFMTS -- CONTINUED
```
'80011' = '1' /* CL SKULL VLT FX W/O COMA */
'80012' = '1' /* CL SKULL VLT FX-BRF COMA */
'80013' = '1' /* CL SKULL VLT FX-MOD COMA */
'80014' = '1' /* CL SKL VLT FX-PROLN COMA */
'80015' = '1' /* CL SKUL VLT FX-DEEP COMA */
'80016' = '1' /* CL SKULL VLT FX-COMA NOS */
'80019' = '1' /* CL SKL VLT FX-CONCUS NOS */
'8002'  = '1' /* CL SKL VLT FX/MENING HEM */
'80020' = '1' /* CL SKL VLT FX/MENING HEM */
'80021' = '1' /* CL SKULL VLT FX W/O COMA */
'80022' = '1' /* CL SKULL VLT FX-BRF COMA */
'80023' = '1' /* CL SKULL VLT FX-MOD COMA */
'80024' = '1' /* CL SKL VLT FX-PROLN COMA */
'80025' = '1' /* CL SKUL VLT FX-DEEP COMA */
'80026' = '1' /* CL SKULL VLT FX-COMA NOS */
'80029' = '1' /* CL SKL VLT FX-CONCUS NOS */
'8003'  = '1' /* CL SKULL VLT FX/HEM NEC */
'80030' = '1' /* CL SKULL VLT FX/HEM NEC */
'80031' = '1' /* CL SKULL VLT FX W/O COMA */
'80032' = '1' /* CL SKULL VLT FX-BRF COMA */
'80033' = '1' /* CL SKULL VLT FX-MOD COMA */
'80034' = '1' /* CL SKL VLT FX-PROLN COMA */
'80035' = '1' /* CL SKUL VLT FX-DEEP COMA */
'80036' = '1' /* CL SKULL VLT FX-COMA NOS */
'80039' = '1' /* CL SKL VLT FX-CONCUS NOS */
'8004'  = '1' /* CL SKL VLT FX/BR INJ NEC */
'80040' = '1' /* CL SKL VLT FX/BR INJ NEC */
'80041' = '1' /* CL SKULL VLT FX W/O COMA */
'80042' = '1' /* CL SKULL VLT FX-BRF COMA */
'80043' = '1' /* CL SKULL VLT FX-MOD COMA */
'80044' = '1' /* CL SKL VLT FX-PROLN COMA */
'80045' = '1' /* CL SKUL VLT FX-DEEP COMA */
'80046' = '1' /* CL SKULL VLT FX-COMA NOS */
'80049' = '1' /* CL SKL VLT FX-CONCUS NOS */
'8005'  = '1' /* OPN SKULL VAULT FRACTURE */
'80050' = '1' /* OPN SKULL VAULT FRACTURE */
'80051' = '1' /* OPN SKUL VLT FX W/O COMA */
'80052' = '1' /* OPN SKUL VLT FX-BRF COMA */
'80053' = '1' /* OPN SKUL VLT FX-MOD COMA */
'80054' = '1' /* OPN SKL VLT FX-PROLN COM */
```

```
'80055' = '1' /* OPN SKL VLT FX-DEEP COMA */
'80056' = '1' /* OPN SKUL VLT FX-COMA NOS */
'80059' = '1' /* OP SKL VLT FX-CONCUS NOS */
'8006'  = '1' /* OPN SKL VLT FX/CEREB LAC */
'80060' = '1' /* OPN SKL VLT FX/CEREB LAC */
'80061' = '1' /* OPN SKUL VLT FX W/O COMA */
'80062' = '1' /* OPN SKUL VLT FX-BRF COMA */
'80063' = '1' /* OPN SKUL VLT FX-MOD COMA */
'80064' = '1' /* OPN SKL VLT FX-PROLN COM */
'80065' = '1' /* OPN SKL VLT FX-DEEP COMA */
'80066' = '1' /* OPN SKUL VLT FX-COMA NOS */
'80069' = '1' /* OP SKL VLT FX-CONCUS NOS */
'8007'  = '1' /* OPN SKL VLT FX/MENIN HEM */
'80070' = '1' /* OPN SKL VLT FX/MENIN HEM */
'80071' = '1' /* OPN SKUL VLT FX W/O COMA */
'80072' = '1' /* OPN SKUL VLT FX-BRF COMA */
'80073' = '1' /* OPN SKUL VLT FX-MOD COMA */
'80074' = '1' /* OPN SKL VLT FX-PROLN COM */
'80075' = '1' /* OPN SKL VLT FX-DEEP COMA */
```
PSFMTS -- CONTINUED

```
'80076' = '1' /* OPN SKUL VLT FX-COMA NOS */
'80079' = '1' /* OP SKL VLT FX-CONCUS NOS */
'8008'  = '1' /* OPN SKULL VLT FX/HEM NEC */
'80080' = '1' /* OPN SKULL VLT FX/HEM NEC */
'80081' = '1' /* OPN SKUL VLT FX W/O COMA */
'80082' = '1' /* OPN SKUL VLT FX-BRF COMA */
'80083' = '1' /* OPN SKUL VLT FX-MOD COMA */
'80084' = '1' /* OPN SKL VLT FX-PROLN COM */

'80085' = '1' /* OPN SKL VLT FX-DEEP COMA */
'80086' = '1' /* OPN SKUL VLT FX-COMA NOS */
'80089' = '1' /* OP SKL VLT FX-CONCUS NOS */
'8009'  = '1' /* OP SKL VLT FX/BR INJ NEC */
'80090' = '1' /* OP SKL VLT FX/BR INJ NEC */
'80091' = '1' /* OPN SKUL VLT FX W/O COMA */
'80092' = '1' /* OPN SKUL VLT FX-BRF COMA */
'80093' = '1' /* OPN SKUL VLT FX-MOD COMA */
'80094' = '1' /* OPN SKL VLT FX-PROLN COM */
'80095' = '1' /* OP SKUL VLT FX-DEEP COMA */
'80096' = '1' /* OPN SKUL VLT FX-COMA NOS */
'80099' = '1' /* OP SKL VLT FX-CONCUS NOS */
'801'   = '1' /* SKULL BASE FRACTURE */
'8010'  = '1' /* CLOS SKULL BASE FRACTURE */
'80100' = '1' /* CLOS SKULL BASE FRACTURE */
'80101' = '1' /* CL SKUL BASE FX W/O COMA */
'80102' = '1' /* CL SKUL BASE FX-BRF COMA */
'80103' = '1' /* CL SKUL BASE FX-MOD COMA */
'80104' = '1' /* CL SKL BASE FX-PROL COMA */
'80105' = '1' /* CL SKL BASE FX-DEEP COMA */
'80106' = '1' /* CL SKUL BASE FX-COMA NOS */
'80109' = '1' /* CL SKULL BASE FX-CONCUSS */
'8011'  = '1' /* CL SKL BASE FX/CEREB LAC */
'80110' = '1' /* CL SKL BASE FX/CEREB LAC */
'80111' = '1' /* CL SKUL BASE FX W/O COMA */
'80112' = '1' /* CL SKUL BASE FX-BRF COMA */
'80113' = '1' /* CL SKUL BASE FX-MOD COMA */
'80114' = '1' /* CL SKL BASE FX-PROL COMA */
```

```
'80115' = '1' /* CL SKL BASE FX-DEEP COMA */
'80116' = '1' /* CL SKUL BASE FX-COMA NOS */
'80119' = '1' /* CL SKULL BASE FX-CONCUSS */
'8012'  = '1' /* CL SKL BASE FX/MENIN HEM */
'80120' = '1' /* CL SKL BASE FX/MENIN HEM */
'80121' = '1' /* CL SKUL BASE FX W/O COMA */
'80122' = '1' /* CL SKUL BASE FX/BRF COMA */
'80123' = '1' /* CL SKUL BASE FX-MOD COMA */
'80124' = '1' /* CL SKL BASE FX-PROL COMA */
'80125' = '1' /* CL SKL BASE FX-DEEP COMA */
'80126' = '1' /* CL SKUL BASE FX-COMA NOS */
'80129' = '1' /* CL SKULL BASE FX-CONCUSS */
'8013'  = '1' /* CL SKULL BASE FX/HEM NEC */
'80130' = '1' /* CL SKULL BASE FX/HEM NEC */
'80131' = '1' /* CL SKUL BASE FX W/O COMA */
'80132' = '1' /* CL SKUL BASE FX-BRF COMA */
'80133' = '1' /* CL SKUL BASE FX-MOD COMA */
'80134' = '1' /* CL SKL BASE FX-PROL COMA */
'80135' = '1' /* CL SKL BASE FX-DEEP COMA */
'80136' = '1' /* CL SKUL BASE FX-COMA NOS */
'80139' = '1' /* CL SKULL BASE FX-CONCUSS */
'8014'  = '1' /* CL SK BASE FX/BR INJ NEC */
```

PSFMTS -- CONTINUED

```
'80140' = '1' /* CL SK BASE FX/BR INJ NEC */
'80141' = '1' /* CL SKUL BASE FX W/O COMA */
'80142' = '1' /* CL SKUL BASE FX-BRF COMA */
'80143' = '1' /* CL SKUL BASE FX-MOD COMA */
'80144' = '1' /* CL SKL BASE FX-PROL COMA */
'80145' = '1' /* CL SKL BASE FX-DEEP COMA */
'80146' = '1' /* CL SKUL BASE FX-COMA NOS */
'80149' = '1' /* CL SKULL BASE FX-CONCUSS */
'8015'  = '1' /* OPEN SKULL BASE FRACTURE */
'80150' = '1' /* OPEN SKULL BASE FRACTURE */
'80151' = '1' /* OPN SKL BASE FX W/O COMA */
'80152' = '1' /* OPN SKL BASE FX-BRF COMA */
'80153' = '1' /* OPN SKL BASE FX-MOD COMA */
'80154' = '1' /* OP SKL BASE FX-PROL COMA */
'80155' = '1' /* OP SKL BASE FX-DEEP COMA */
'80156' = '1' /* OPN SKL BASE FX-COMA NOS */
'80159' = '1' /* OPN SKUL BASE FX-CONCUSS */
'8016'  = '1' /* OP SKL BASE FX/CEREB LAC */
'80160' = '1' /* OP SKL BASE FX/CEREB LAC */
'80161' = '1' /* OPN SKL BASE FX W/O COMA */
'80162' = '1' /* OPN SKL BASE FX-BRF COMA */
'80163' = '1' /* OPN SKL BASE FX-MOD COMA */
'80164' = '1' /* OP SKL BASE FX-PROL COMA */
'80165' = '1' /* OP SKL BASE FX-DEEP COMA */
'80166' = '1' /* OPN SKL BASE FX-COMA NOS */
'80169' = '1' /* OPN SKUL BASE FX-CONCUSS */
'8017'  = '1' /* OP SKL BASE FX/MENIN HEM */
'80170' = '1' /* OP SKL BASE FX/MENIN HEM */
'80171' = '1' /* OPN SKL BASE FX W/O COMA */
'80172' = '1' /* OPN SKL BASE FX-BRF COMA */
'80173' = '1' /* OPN SKL BASE FX-MOD COMA */
'80174' = '1' /* OP SKL BASE FX-PROL COMA */
'80175' = '1' /* OP SKL BASE FX-DEEP COMA */
'80176' = '1' /* OPN SKL BASE FX-COMA NOS */
```

```
'80179' = '1' /* OPN SKUL BASE FX-CONCUSS */
'8018'  = '1' /* OPN SKUL BASE FX/HEM NEC */
'80180' = '1' /* OPN SKUL BASE FX/HEM NEC */
'80181' = '1' /* OPN SKL BASE FX W/O COMA */
'80182' = '1' /* OPN SKL BASE FX-BRF COMA */
'80183' = '1' /* OPN SKL BASE FX-MOD COMA */
'80184' = '1' /* OP SKL BASE FX-PROL COMA */
'80185' = '1' /* OP SKL BASE FX-DEEP COMA */
'80186' = '1' /* OPN SKL BASE FX-COMA NOS */
'80189' = '1' /* OPN SKUL BASE FX-CONCUSS */
'8019'  = '1' /* OP SK BASE FX/BR INJ NEC */
'80190' = '1' /* OP SK BASE FX/BR INJ NEC */
'80191' = '1' /* OP SKUL BASE FX W/O COMA */
'80192' = '1' /* OPN SKL BASE FX-BRF COMA */
'80193' = '1' /* OPN SKL BASE FX-MOD COMA */
'80194' = '1' /* OP SKL BASE FX-PROL COMA */
'80195' = '1' /* OP SKL BASE FX-DEEP COMA */
'80196' = '1' /* OPN SKL BASE FX-COMA NOS */
'80199' = '1' /* OPN SKUL BASE FX-CONCUSS */
'802'   = '1' /* FRACTURE OF FACE BONES */
'8020'  = '1' /* NASAL BONE FX-CLOSED */
'8021'  = '1' /* NASAL BONE FX-OPEN */
'8022'  = '1' /* MANDIBLE FRACTURE-CLOSED */
'80220' = '1' /* MANDIBLE FX NOS-CLOSED */
'80221' = '1' /* FX CONDYL PROC MANDIB-CL */
```

PSFMTS -- CONTINUED

```
'80222' = '1' /* SUBCONDYLAR FX MANDIB-CL */
'80223' = '1' /* FX CORON PROC MANDIB-CL */
'80224' = '1' /* FX RAMUS NOS-CLOSED */
'80225' = '1' /* FX ANGLE OF JAW-CLOSED */
'80226' = '1' /* FX SYMPHY MANDIB BODY-CL */
'80227' = '1' /* FX ALVEOLAR BORD MAND-CL */
'80228' = '1' /* FX MANDIBLE BODY NEC-CL */
'80229' = '1' /* MULT FX MANDIBLE-CLOSED */
'8023'  = '1' /* MANDIBLE FRACTURE-OPEN */
'80230' = '1' /* MANDIBLE FX NOS-OPEN */
'80231' = '1' /* FX CONDYL PROC MAND-OPEN */
'80232' = '1' /* SUBCONDYL FX MANDIB-OPEN */
'80233' = '1' /* FX CORON PROC MANDIB-OPN */
'80234' = '1' /* FX RAMUS NOS-OPEN */
'80235' = '1' /* FX ANGLE OF JAW-OPEN */
'80236' = '1' /* FX SYMPHY MANDIB BDY-OPN */
'80237' = '1' /* FX ALV BORD MAND BDY-OPN */
'80238' = '1' /* FX MANDIBLE BODY NEC-OPN */
'80239' = '1' /* MULT FX MANDIBLE-OPEN */
'8024'  = '1' /* FX MALAR/MAXILLARY-CLOSE */
'8025'  = '1' /* FX MALAR/MAXILLARY-OPEN */
'8026'  = '1' /* FX ORBITAL FLOOR-CLOSED */
'8027'  = '1' /* FX ORBITAL FLOOR-OPEN */
'8028'  = '1' /* FX FACIAL BONE NEC-CLOSE */
'8029'  = '1' /* FX FACIAL BONE NEC-OPEN */
'803'   = '1' /* OTHER SKULL FRACTURE */
'8030'  = '1' /* CLOSE SKULL FRACTURE NEC */
'80300' = '1' /* CLOSE SKULL FRACTURE NEC */
'80301' = '1' /* CL SKULL FX NEC W/O COMA */
'80302' = '1' /* CL SKULL FX NEC-BRF COMA */
'80303' = '1' /* CL SKULL FX NEC-MOD COMA */
```

```
'80304' = '1' /* CL SKL FX NEC-PROLN COMA */
'80305' = '1' /* CL SKUL FX NEC-DEEP COMA */
'80306' = '1' /* CL SKULL FX NEC-COMA NOS */
'80309' = '1' /* CL SKULL FX NEC-CONCUSS */
'8031'  = '1' /* CL SKL FX NEC/CEREBR LAC */
'80310' = '1' /* CL SKL FX NEC/CEREBR LAC */
'80311' = '1' /* CL SKULL FX NEC W/O COMA */
'80312' = '1' /* CL SKULL FX NEC-BRF COMA */
'80313' = '1' /* CL SKULL FX NEC-MOD COMA */
'80314' = '1' /* CL SKL FX NEC-PROLN COMA */
'80315' = '1' /* CL SKUL FX NEC-DEEP COMA */
'80316' = '1' /* CL SKULL FX NEC-COMA NOS */
'80319' = '1' /* CL SKULL FX NEC-CONCUSS */
'8032'  = '1' /* CL SKL FX NEC/MENING HEM */
'80320' = '1' /* CL SKL FX NEC/MENING HEM */
'80321' = '1' /* CL SKULL FX NEC W/O COMA */
'80322' = '1' /* CL SKULL FX NEC-BRF COMA */
'80323' = '1' /* CL SKULL FX NEC-MOD COMA */
'80324' = '1' /* CL SKL FX NEC-PROLN COMA */
'80325' = '1' /* CL SKUL FX NEC-DEEP COMA */
'80326' = '1' /* CL SKULL FX NEC-COMA NOS */
'80329' = '1' /* CL SKULL FX NEC-CONCUSS */
'8033'  = '1' /* CL SKULL FX NEC/HEM NEC */
'80330' = '1' /* CL SKULL FX NEC/HEM NEC */
'80331' = '1' /* CL SKULL FX NEC W/O COMA */
'80332' = '1' /* CL SKULL FX NEC-BRF COMA */
'80333' = '1' /* CL SKULL FX NEC-MOD COMA */
'80334' = '1' /* CL SKL FX NEC-PROLN COMA */
```

PSFMTS -- CONTINUED

```
'80335' = '1' /* CL SKUL FX NEC-DEEP COMA */
'80336' = '1' /* CL SKULL FX NEC-COMA NOS */
'80339' = '1' /* CL SKULL FX NEC-CONCUSS */
'8034'  = '1' /* CL SKL FX NEC/BR INJ NEC */
'80340' = '1' /* CL SKL FX NEC/BR INJ NEC */
'80341' = '1' /* CL SKULL FX NEC W/O COMA */
'80342' = '1' /* CL SKULL FX NEC-BRF COMA */
'80343' = '1' /* CL SKULL FX NEC-MOD COMA */
'80344' = '1' /* CL SKL FX NEC-PROLN COMA */
'80345' = '1' /* CL SKUL FX NEC-DEEP COMA */
'80346' = '1' /* CL SKULL FX NEC-COMA NOS */
'80349' = '1' /* CL SKULL FX NEC-CONCUSS */
'8035'  = '1' /* OPEN SKULL FRACTURE NEC */
'80350' = '1' /* OPEN SKULL FRACTURE NEC */
'80351' = '1' /* OPN SKUL FX NEC W/O COMA */
'80352' = '1' /* OPN SKUL FX NEC-BRF COMA */
'80353' = '1' /* OPN SKUL FX NEC-MOD COMA */
'80354' = '1' /* OPN SKL FX NEC-PROL COMA */
'80355' = '1' /* OPN SKL FX NEC-DEEP COMA */
'80356' = '1' /* OPN SKUL FX NEC-COMA NOS */
'80359' = '1' /* OPN SKULL FX NEC-CONCUSS */
'8036'  = '1' /* OPN SKL FX NEC/CEREB LAC */
'80360' = '1' /* OPN SKL FX NEC/CEREB LAC */
'80361' = '1' /* OPN SKUL FX NEC W/O COMA */
'80362' = '1' /* OPN SKUL FX NEC-BRF COMA */
'80363' = '1' /* OPN SKUL FX NEC-MOD COMA */
'80364' = '1' /* OPN SKL FX NEC-PROLN COM */
'80365' = '1' /* OPN SKL FX NEC-DEEP COMA */
```

```
'80366' = '1' /* OPN SKUL FX NEC-COMA NOS */
'80369' = '1' /* OPN SKULL FX NEC-CONCUSS */
'8037'  = '1' /* OPN SKL FX NEC/MENIN HEM */
'80370' = '1' /* OPN SKL FX NEC/MENIN HEM */
'80371' = '1' /* OPN SKUL FX NEC W/O COMA */
'80372' = '1' /* OPN SKUL FX NEC-BRF COMA */
'80373' = '1' /* OPN SKUL FX NEC-MOD COMA */
'80374' = '1' /* OPN SKL FX NEC-PROL COMA */
'80375' = '1' /* OPN SKL FX NEC-DEEP COMA */
'80376' = '1' /* OPN SKUL FX NEC-COMA NOS */
'80379' = '1' /* OPN SKULL FX NEC-CONCUSS */
'8038'  = '1' /* OPN SKULL FX NEC/HEM NEC */
'80380' = '1' /* OPN SKULL FX NEC/HEM NEC */
'80381' = '1' /* OPN SKUL FX NEC W/O COMA */
'80382' = '1' /* OPN SKUL FX NEC-BRF COMA */
'80383' = '1' /* OPN SKUL FX NEC-MOD COMA */
'80384' = '1' /* OPN SKL FX NEC-PROL COMA */
'80385' = '1' /* OPN SKL FX NEC-DEEP COMA */
'80386' = '1' /* OPN SKUL FX NEC-COMA NOS */
'80389' = '1' /* OPN SKULL FX NEC-CONCUSS */
'8039'  = '1' /* OP SKL FX NEC/BR INJ NEC */
'80390' = '1' /* OP SKL FX NEC/BR INJ NEC */
'80391' = '1' /* OPN SKUL FX NEC W/O COMA */
'80392' = '1' /* OPN SKUL FX NEC-BRF COMA */
'80393' = '1' /* OPN SKUL FX NEC-MOD COMA */
'80394' = '1' /* OPN SKL FX NEC-PROL COMA */
'80395' = '1' /* OPN SKL FX NEC-DEEP COMA */
'80396' = '1' /* OPN SKUL FX NEC-COMA NOS */
'80399' = '1' /* OPN SKULL FX NEC-CONCUSS */
'804'   = '1' /* MULT FX SKULL W OTH BONE */
'8040'  = '1' /* CL SKUL FX W OTH BONE FX */
```

PSFMTS -- CONTINUED

```
'80400' = '1' /* CL SKUL FX W OTH BONE FX */
'80401' = '1' /* CL SKL W OTH FX W/O COMA */
'80402' = '1' /* CL SKL W OTH FX-BRF COMA */
'80403' = '1' /* CL SKL W OTH FX-MOD COMA */
'80404' = '1' /* CL SKL/OTH FX-PROLN COMA */
'80405' = '1' /* CL SKUL/OTH FX-DEEP COMA */
'80406' = '1' /* CL SKL W OTH FX-COMA NOS */
'80409' = '1' /* CL SKUL W OTH FX-CONCUSS */
'8041'  = '1' /* CL SK W OTH FX/CEREB LAC */
'80410' = '1' /* CL SK W OTH FX/CEREB LAC */
'80411' = '1' /* CL SKL W OTH FX W/O COMA */
'80412' = '1' /* CL SKL W OTH FX-BRF COMA */
'80413' = '1' /* CL SKL W OTH FX-MOD COMA */
'80414' = '1' /* CL SKL/OTH FX-PROLN COMA */
'80415' = '1' /* CL SKUL/OTH FX-DEEP COMA */
'80416' = '1' /* CL SKL W OTH FX-COMA NOS */
'80419' = '1' /* CL SKUL W OTH FX-CONCUSS */
'8042'  = '1' /* CL SKL/OTH FX/MENING HEM */
'80420' = '1' /* CL SKL/OTH FX/MENING HEM */
'80421' = '1' /* CL SKL W OTH FX W/O COMA */
'80422' = '1' /* CL SKL W OTH FX-BRF COMA */
'80423' = '1' /* CL SKL W OTH FX-MOD COMA */
'80424' = '1' /* CL SKL/OTH FX-PROLN COMA */
'80425' = '1' /* CL SKUL/OTH FX-DEEP COMA */
'80426' = '1' /* CL SKL W OTH FX-COMA NOS */
```

```
'80429' = '1' /* CL SKUL W OTH FX-CONCUSS */
'8043'  = '1' /* CL SKUL W OTH FX/HEM NEC */
'80430' = '1' /* CL SKUL W OTH FX/HEM NEC */
'80431' = '1' /* CL SKL W OTH FX W/O COMA */
'80432' = '1' /* CL SKL W OTH FX-BRF COMA */
'80433' = '1' /* CL SKL W OTH FX-MOD COMA */
'80434' = '1' /* CL SKL/OTH FX-PROLN COMA */
'80435' = '1' /* CL SKUL/OTH FX-DEEP COMA */
'80436' = '1' /* CL SKL W OTH FX-COMA NOS */
'80439' = '1' /* CL SKUL W OTH FX-CONCUSS */
'8044'  = '1' /* CL SKL/OTH FX/BR INJ NEC */
'80440' = '1' /* CL SKL/OTH FX/BR INJ NEC */
'80441' = '1' /* CL SKL W OTH FX W/O COMA */
'80442' = '1' /* CL SKL W OTH FX-BRF COMA */
'80443' = '1' /* CL SKL W OTH FX-MOD COMA */
'80444' = '1' /* CL SKL/OTH FX-PROLN COMA */
'80445' = '1' /* CL SKUL/OTH FX-DEEP COMA */
'80446' = '1' /* CL SKL W OTH FX-COMA NOS */
'80449' = '1' /* CL SKUL W OTH FX-CONCUSS */
'8045'  = '1' /* OPN SKULL FX/OTH BONE FX */
'80450' = '1' /* OPN SKULL FX/OTH BONE FX */
'80451' = '1' /* OPN SKUL/OTH FX W/O COMA */
'80452' = '1' /* OPN SKUL/OTH FX-BRF COMA */
'80453' = '1' /* OPN SKUL/OTH FX-MOD COMA */
'80454' = '1' /* OPN SKL/OTH FX-PROL COMA */
'80455' = '1' /* OPN SKL/OTH FX-DEEP COMA */
'80456' = '1' /* OPN SKUL/OTH FX-COMA NOS */
'80459' = '1' /* OPN SKULL/OTH FX-CONCUSS */
'8046'  = '1' /* OPN SKL/OTH FX/CEREB LAC */
'80460' = '1' /* OPN SKL/OTH FX/CEREB LAC */
'80461' = '1' /* OPN SKUL/OTH FX W/O COMA */
'80462' = '1' /* OPN SKUL/OTH FX-BRF COMA */
'80463' = '1' /* OPN SKUL/OTH FX-MOD COMA */
'80464' = '1' /* OPN SKL/OTH FX-PROL COMA */
```

PSFMTS -- CONTINUED

```
'80465' = '1' /* OPN SKL/OTH FX-DEEP COMA */
'80466' = '1' /* OPN SKUL/OTH FX-COMA NOS */
'80469' = '1' /* OPN SKULL/OTH FX-CONCUSS */
'8047'  = '1' /* OPN SKL/OTH FX/MENIN HEM */
'80470' = '1' /* OPN SKL/OTH FX/MENIN HEM */
'80471' = '1' /* OPN SKUL/OTH FX W/O COMA */
'80472' = '1' /* OPN SKUL/OTH FX-BRF COMA */
'80473' = '1' /* OPN SKUL/OTH FX-MOD COMA */
'80474' = '1' /* OPN SKL/OTH FX-PROL COMA */
'80475' = '1' /* OPN SKL/OTH FX-DEEP COMA */
'80476' = '1' /* OPN SKUL/OTH FX-COMA NOS */
'80479' = '1' /* OPN SKULL/OTH FX-CONCUSS */
'8048'  = '1' /* OPN SKL W OTH FX/HEM NEC */
'80480' = '1' /* OPN SKL W OTH FX/HEM NEC */
'80481' = '1' /* OPN SKUL/OTH FX W/O COMA */
'80482' = '1' /* OPN SKUL/OTH FX-BRF COMA */
'80483' = '1' /* OPN SKUL/OTH FX-MOD COMA */
'80484' = '1' /* OPN SKL/OTH FX-PROL COMA */
'80485' = '1' /* OPN SKL/OTH FX-DEEP COMA */
'80486' = '1' /* OPN SKUL/OTH FX-COMA NOS */
'80489' = '1' /* OPN SKULL/OTH FX-CONCUSS */
'8049'  = '1' /* OP SKL/OTH FX/BR INJ NEC */
```

```
'80490' = '1' /* OP SKL/OTH FX/BR INJ NEC */
'80491' = '1' /* OPN SKUL/OTH FX W/O COMA */
'80492' = '1' /* OPN SKUL/OTH FX-BRF COMA */
'80493' = '1' /* OPN SKUL/OTH FX-MOD COMA */
'80494' = '1' /* OPN SKL/OTH FX-PROL COMA */
'80495' = '1' /* OPN SKL/OTH FX-DEEP COMA */
'80496' = '1' /* OPN SKUL/OTH FX-COMA NOS */
'80499' = '1' /* OPN SKULL/OTH FX-CONCUSS */
'805'   = '1' /* VERTEBRL FX W/O CORD INJ */
'8050'  = '1' /* FX CERVICAL VERTEBRA-CL */
'80500' = '1' /* FX CERVICAL VERT NOS-CL */
'80501' = '1' /* FX C1 VERTEBRA-CLOSED */
'80502' = '1' /* FX C2 VERTEBRA-CLOSED */
'80503' = '1' /* FX C3 VERTEBRA-CLOSED */
'80504' = '1' /* FX C4 VERTEBRA-CLOSED */
'80505' = '1' /* FX C5 VERTEBRA-CLOSED */
'80506' = '1' /* FX C6 VERTEBRA-CLOSED */
'80507' = '1' /* FX C7 VERTEBRA-CLOSED */
'80508' = '1' /* FX MULT CERVICAL VERT-CL */
'8051'  = '1' /* FX CERVICAL VERTEBRA-OPN */
'80510' = '1' /* FX CERVICAL VERT NOS-OPN */
'80511' = '1' /* FX C1 VERTEBRA-OPEN */
'80512' = '1' /* FX C2 VERTEBRA-OPEN */
'80513' = '1' /* FX C3 VERTEBRA-OPEN */
'80514' = '1' /* FX C4 VERTEBRA-OPEN */
'80515' = '1' /* FX C5 VERTEBRA-OPEN */
'80516' = '1' /* FX C6 VERTEBRA-OPEN */
'80517' = '1' /* FX C7 VERTEBRA-OPEN */
'80518' = '1' /* FX MLT CERVICAL VERT-OPN */
'8052'  = '1' /* FX DORSAL VERTEBRA-CLOSE */
'8053'  = '1' /* FX DORSAL VERTEBRA-OPEN */
'8054'  = '1' /* FX LUMBAR VERTEBRA-CLOSE */
'8055'  = '1' /* FX LUMBAR VERTEBRA-OPEN */
'8056'  = '1' /* FX SACRUM/COCCYX-CLOSED */
'8057'  = '1' /* FX SACRUM/COCCYX-OPEN */
'8058'  = '1' /* VERTEBRAL FX NOS-CLOSED */
'8059'  = '1' /* VERTEBRAL FX NOS-OPEN */
```

PSFMTS -- CONTINUED

```
'806'   = '1' /* VERTEBRAL FX W CORD INJ */
'8060'  = '1' /* CLOS CERV FX W CORD INJ */
'80600' = '1' /* C1-C4 FX-CL/CORD INJ NOS */
'80601' = '1' /* C1-C4 FX-CL/COM CORD LES */
'80602' = '1' /* C1-C4 FX-CL/ANT CORD SYN */
'80603' = '1' /* C1-C4 FX-CL/CEN CORD SYN */
'80604' = '1' /* C1-C4 FX-CL/CORD INJ NEC */
'80605' = '1' /* C5-C7 FX-CL/CORD INJ NOS */
'80606' = '1' /* C5-C7 FX-CL/COM CORD LES */
'80607' = '1' /* C5-C7 FX-CL/ANT CORD SYN */
'80608' = '1' /* C5-C7 FX-CL/CEN CORD SYN */
'80609' = '1' /* C5-C7 FX-CL/CORD INJ NEC */
'8061'  = '1' /* OPEN CERV FX W CORD INJ */
'80610' = '1' /* C1-C4 FX-OP/CORD INJ NOS */
'80611' = '1' /* C1-C4 FX-OP/COM CORD LES */
'80612' = '1' /* C1-C4 FX-OP/ANT CORD SYN */
'80613' = '1' /* C1-C4 FX-OP/CEN CORD SYN */
'80614' = '1' /* C1-C4 FX-OP/CORD INJ NEC */
'80615' = '1' /* C5-C7 FX-OP/CORD INJ NOS */
```

```
'80616' = '1' /* C5-C7 FX-OP/COM CORD LES */
'80617' = '1' /* C5-C7 FX-OP/ANT CORD SYN */
'80618' = '1' /* C5-C7 FX-OP/CEN CORD SYN */
'80619' = '1' /* C5-C7 FX-OP/CORD INJ NEC */
'8062'  = '1' /* CL DORSAL FX W CORD INJ */
'80620' = '1' /* T1-T6 FX-CL/CORD INJ NOS */
'80621' = '1' /* T1-T6 FX-CL/COM CORD LES */
'80622' = '1' /* T1-T6 FX-CL/ANT CORD SYN */
'80623' = '1' /* T1-T6 FX-CL/CEN CORD SYN */
'80624' = '1' /* T1-T6 FX-CL/CORD INJ NEC */
'80625' = '1' /* T7-T12 FX-CL/CRD INJ NOS */
'80626' = '1' /* T7-T12 FX-CL/COM CRD LES */
'80627' = '1' /* T7-T12 FX-CL/ANT CRD SYN */
'80628' = '1' /* T7-T12 FX-CL/CEN CRD SYN */
'80629' = '1' /* T7-T12 FX-CL/CRD INJ NEC */
'8063'  = '1' /* OPN DORSAL FX W CORD INJ */
'80630' = '1' /* T1-T6 FX-OP/CORD INJ NOS */
'80631' = '1' /* T1-T6 FX-OP/COM CORD LES */
'80632' = '1' /* T1-T6 FX-OP/ANT CORD SYN */
'80633' = '1' /* T1-T6 FX-OP/CEN CORD SYN */
'80634' = '1' /* T1-T6 FX-OP/CORD INJ NEC */
'80635' = '1' /* T7-T12 FX-OP/CRD INJ NOS */
'80636' = '1' /* T7-T12 FX-OP/COM CRD LES */
'80637' = '1' /* T7-T12 FX-OP/ANT CRD SYN */
'80638' = '1' /* T7-T12 FX-OP/CEN CRD SYN */
'80639' = '1' /* T7-T12 FX-OP/CRD INJ NEC */
'8064'  = '1' /* CL LUMBAR FX W CORD INJ */
'8065'  = '1' /* OPN LUMBAR FX W CORD INJ */
'8066'  = '1' /* FX SACRUM-CL W CORD INJ */
'80660' = '1' /* FX SACRUM-CL/CRD INJ NOS */
'80661' = '1' /* FX SACR-CL/CAUDA EQU LES */
'80662' = '1' /* FX SACR-CL/CAUDA INJ NEC */
'80669' = '1' /* FX SACRUM-CL/CRD INJ NEC */
'8067'  = '1' /* FX SACRUM-OPN W CORD INJ */
'80670' = '1' /* FX SACRUM-OP/CRD INJ NOS */
'80671' = '1' /* FX SACR-OP/CAUDA EQU LES */
'80672' = '1' /* FX SACR-OP/CAUDA INJ NEC */
'80679' = '1' /* FX SACRUM-OP/CRD INJ NEC */
'8068'  = '1' /* VERT FX NOS-CL W CRD INJ */
'8069'  = '1' /* VERT FX NOS-OP W CRD INJ */
```

PSFMTS -- CONTINUED

```
'807'   = '1' /* FX RIB/STERN/LARYN/TRACH */
'8070'  = '1' /* FRACTURE OF RIB-CLOSED */
'80700' = '1' /* FRACTURE RIB NOS-CLOSED */
'80701' = '1' /* FRACTURE ONE RIB-CLOSED */
'80702' = '1' /* FRACTURE TWO RIBS-CLOSED */
'80703' = '1' /* FRACTURE THREE RIBS-CLOS */
'80704' = '1' /* FRACTURE FOUR RIBS-CLOSE */
'80705' = '1' /* FRACTURE FIVE RIBS-CLOSE */
'80706' = '1' /* FRACTURE SIX RIBS-CLOSED */
'80707' = '1' /* FRACTURE SEVEN RIBS-CLOS */
'80708' = '1' /* FX EIGHT/MORE RIB-CLOSED */
'80709' = '1' /* FX MULT RIBS NOS-CLOSED */
'8071'  = '1' /* FRACTURE OF RIB-OPEN */
'80710' = '1' /* FRACTURE RIB NOS-OPEN */
'80711' = '1' /* FRACTURE ONE RIB-OPEN */
'80712' = '1' /* FRACTURE TWO RIBS-OPEN */
```

```
'80713' = '1' /* FRACTURE THREE RIBS-OPEN */
'80714' = '1' /* FRACTURE FOUR RIBS-OPEN */
'80715' = '1' /* FRACTURE FIVE RIBS-OPEN */
'80716' = '1' /* FRACTURE SIX RIBS-OPEN */
'80717' = '1' /* FRACTURE SEVEN RIBS-OPEN */
'80718' = '1' /* FX EIGHT/MORE RIBS-OPEN */
'80719' = '1' /* FX MULT RIBS NOS-OPEN */
'8072'  = '1' /* FRACTURE OF STERNUM-CLOS */
'8073'  = '1' /* FRACTURE OF STERNUM-OPEN */
'8074'  = '1' /* FLAIL CHEST */
'8075'  = '1' /* FX LARYNX/TRACHEA-CLOSED */
'8076'  = '1' /* FX LARYNX/TRACHEA-OPEN */
'808'   = '1' /* PELVIC FRACTURE */
'8080'  = '1' /* FRACTURE ACETABULUM-CLOS */
'8081'  = '1' /* FRACTURE ACETABULUM-OPEN */
'8082'  = '1' /* FRACTURE OF PUBIS-CLOSED */
'8083'  = '1' /* FRACTURE OF PUBIS-OPEN */
'8084'  = '1' /* OTH PELVIC FRACTURE-CLOS */
'80841' = '1' /* FRACTURE OF ILIUM-CLOSED */
'80842' = '1' /* FRACTURE ISCHIUM-CLOSED */
'80843' = '1' /* PELV FX-CLOS/PELV DISRUP */
'80849' = '1' /* PELVIC FRACTURE NEC-CLOS */
'8085'  = '1' /* OTH PELVIC FRACTURE-OPEN */
'80851' = '1' /* FRACTURE OF ILIUM-OPEN */
'80852' = '1' /* FRACTURE OF ISCHIUM-OPEN */
'80853' = '1' /* PELV FX-OPEN/PELV DISRUP */
'80859' = '1' /* PELVIC FRACTURE NEC-OPEN */
'8088'  = '1' /* PELVIC FRACTURE NOS-CLOS */
'8089'  = '1' /* PELVIC FRACTURE NOS-OPEN */
'809'   = '1' /* FRACTURE OF TRUNK BONES */
'8090'  = '1' /* FRACTURE TRUNK BONE-CLOS */
'8091'  = '1' /* FRACTURE TRUNK BONE-OPEN */
'810'   = '1' /* CLAVICLE FRACTURE */
'8100'  = '1' /* FRACTURE CLAVICLE-CLOSED */
'81000' = '1' /* FX CLAVICLE NOS-CLOSED */
'81001' = '1' /* FX CLAVICL, STERN END-CL */
'81002' = '1' /* FX CLAVICLE SHAFT-CLOSED */
'81003' = '1' /* FX CLAVICL, ACROM END-CL */
'8101'  = '1' /* FRACTURE CLAVICLE-OPEN */
'81010' = '1' /* FX CLAVICLE NOS-OPEN */
'81011' = '1' /* FX CLAVIC, STERN END-OPN */
'81012' = '1' /* FX CLAVICLE SHAFT-OPEN */
'81013' = '1' /* FX CLAVIC, ACROM END-OPN */
```

PSFMTS -- CONTINUED

```
'811'   = '1' /* SCAPULA FRACTURE */
'8110'  = '1' /* FRACTURE SCAPULA-CLOSED */
'81100' = '1' /* FX SCAPULA NOS-CLOSED */
'81101' = '1' /* FX SCAPUL, ACROM PROC-CL */
'81102' = '1' /* FX SCAPUL, CORAC PROC-CL */
'81103' = '1' /* FX SCAP, GLEN CAV/NCK-CL */
'81109' = '1' /* FX SCAPULA NEC-CLOSED */
'8111'  = '1' /* FRACTURE OF SCAPULA-OPEN */
'81110' = '1' /* FX SCAPULA NOS-OPEN */
'81111' = '1' /* FX SCAPUL, ACROM PROC-OP */
'81112' = '1' /* FX SCAPUL, CORAC PROC-OP */
'81113' = '1' /* FX SCAP, GLEN CAV/NCK-OP */
'81119' = '1' /* FX SCAPULA NEC-OPEN */
```

```
'812'   = '1' /* HUMERUS FRACTURE */
'8120'  = '1' /* FX UPPER HUMERUS-CLOSED */
'81200' = '1' /* FX UP END HUMERUS NOS-CL */
'81201' = '1' /* FX SURG NCK HUMERUS-CLOS */
'81202' = '1' /* FX ANATOM NCK HUMERUS-CL */
'81203' = '1' /* FX GR TUBEROS HUMERUS-CL */
'81209' = '1' /* FX UPPER HUMERUS NEC-CL */
'8121'  = '1' /* FX UPPER HUMERUS-OPEN */
'81210' = '1' /* FX UPPER HUMERUS NOS-OPN */
'81211' = '1' /* FX SURG NECK HUMERUS-OPN */
'81212' = '1' /* FX ANAT NECK HUMERUS-OPN */
'81213' = '1' /* FX GR TUBEROS HUMER-OPEN */
'81219' = '1' /* FX UPPER HUMERUS NEC-OPN */
'8122'  = '1' /* FX HUMERUS SHAFT/NOS-CL */
'81220' = '1' /* FX HUMERUS NOS-CLOSED */
'81221' = '1' /* FX HUMERUS SHAFT-CLOSED */
'8123'  = '1' /* FX HUMERUS SHAFT/NOS-OPN */
'81230' = '1' /* FX HUMERUS NOS-OPEN */
'81231' = '1' /* FX HUMERUS SHAFT-OPEN */
'8124'  = '1' /* FX LOWER HUMERUS-CLOSED */
'81240' = '1' /* FX LOWER HUMERUS NOS-CL */
'81241' = '1' /* SUPRCONDYL FX HUMERUS-CL */
'81242' = '1' /* FX HUMER, LAT CONDYL-CL */
'81243' = '1' /* FX HUMER, MED CONDYL-CL */
'81244' = '1' /* FX HUMER, CONDYL NOS-CL */
'81249' = '1' /* FX LOWER HUMERUS NEC-CL */
'8125'  = '1' /* FX LOWER HUMERUS-OPEN */
'81250' = '1' /* FX LOWER HUMER NOS-OPEN */
'81251' = '1' /* SUPRACONDYL FX HUMER-OPN */
'81252' = '1' /* FX HUMER, LAT CONDYL-OPN */
'81253' = '1' /* FX HUMER, MED CONDYL-OPN */
'81254' = '1' /* FX HUMER, CONDYL NOS-OPN */
'81259' = '1' /* FX LOWER HUMER NEC-OPEN */
'813'   = '1' /* RADIUS & ULNA FRACTURE */
'8130'  = '1' /* FX UPPER RADIUS/ULNA-CL */
'81300' = '1' /* FX UPPER FOREARM NOS-CL */
'81301' = '1' /* FX OLECRAN PROC ULNA-CL */
'81302' = '1' /* FX CORONOID PROC ULNA-CL */
'81303' = '1' /* MONTEGGIA'S FX-CLOSED */
'81304' = '1' /* FX UPPER ULNA NEC/NOS-CL */
'81305' = '1' /* FX RADIUS HEAD-CLOSED */
'81306' = '1' /* FX RADIUS NECK-CLOSED */
'81307' = '1' /* FX UP RADIUS NEC/NOS-CL */
'81308' = '1' /* FX UP RADIUS W ULNA-CLOS */
'8131'  = '1' /* FX UPPER RADIUS/ULNA-OPN */
'81310' = '1' /* FX UPPER FOREARM NOS-OPN */
```

PSFMTS -- CONTINUED

```
'81311' = '1' /* FX OLECRAN PROC ULNA-OPN */
'81312' = '1' /* FX CORONOID PRO ULNA-OPN */
'81313' = '1' /* MONTEGGIA'S FX-OPEN */
'81314' = '1' /* FX UP ULNA NEC/NOS-OPEN */
'81315' = '1' /* FX RADIUS HEAD-OPEN */
'81316' = '1' /* FX RADIUS NECK-OPEN */
'81317' = '1' /* FX UP RADIUS NEC/NOS-OPN */
'81318' = '1' /* FX UP RADIUS W ULNA-OPEN */
'8132'  = '1' /* FX RADIUS/ULNA SHAFT-CL */
'81320' = '1' /* FX SHAFT FOREARM NOS-CL */
```

```
'81321' = '1' /* FX RADIUS SHAFT-CLOSED */
'81322' = '1' /* FX ULNA SHAFT-CLOSED */
'81323' = '1' /* FX SHAFT RAD W ULNA-CLOS */
'8133'  = '1' /* FX RADIUS/ULNA SHAFT-OPN */
'81330' = '1' /* FX SHAFT FOREARM NOS-OPN */
'81331' = '1' /* FX RADIUS SHAFT-OPEN */
'81332' = '1' /* FX ULNA SHAFT-OPEN */
'81333' = '1' /* FX SHAFT RAD W ULNA-OPEN */
'8134'  = '1' /* FX LOWER RADIUS/ULNA-CL */
'81340' = '1' /* FX LOWER FOREARM NOS-CL */
'81341' = '1' /* COLLES' FRACTURE-CLOSED */
'81342' = '1' /* FX DISTAL RADIUS NEC-CL */
'81343' = '1' /* FX DISTAL ULNA-CLOSED */
'81344' = '1' /* FX LOW RADIUS W ULNA-CL */
'8135'  = '1' /* FX LOW RADIUS W/ULNA-OPN */
'81350' = '1' /* FX LOWER FOREARM NOS-OPN */
'81351' = '1' /* COLLES' FRACTURE-OPEN */
'81352' = '1' /* FX DISTAL RADIUS NEC-OPN */
'81353' = '1' /* FX DISTAL ULNA-OPEN */
'81354' = '1' /* FX LOW RADIUS W ULNA-OPN */
'8138'  = '1' /* FX RADIUS/ULNA NOS-CLOSE */
'81380' = '1' /* FX FOREARM NOS-CLOSED */
'81381' = '1' /* FX RADIUS NOS-CLOSED */
'81382' = '1' /* FRACTURE ULNA NOS-CLOSED */
'81383' = '1' /* FX RADIUS W ULNA NOS-CL */
'8139'  = '1' /* FX RADIUS/ULNA NOS-OPEN */
'81390' = '1' /* FX FOREARM NOS-OPEN */
'81391' = '1' /* FRACTURE RADIUS NOS-OPEN */
'81392' = '1' /* FRACTURE ULNA NOS-OPEN */
'81393' = '1' /* FX RADIUS W ULNA NOS-OPN */
'814'   = '1' /* CARPAL FRACTURE */
'8140'  = '1' /* FRACTURE CARPAL BONE-CL */
'81400' = '1' /* FX CARPAL BONE NOS-CLOSE */
'81401' = '1' /* FX NAVICULAR, WRIST-CLOS */
'81402' = '1' /* FX LUNATE, WRIST-CLOSED */
'81403' = '1' /* FX TRIQUETRAL, WRIST-CL */
'81404' = '1' /* FX PISIFORM-CLOSED */
'81405' = '1' /* FX TRAPEZIUM BONE-CLOSED */
'81406' = '1' /* FX TRAPEZOID BONE-CLOSED */
'81407' = '1' /* FX CAPITATE BONE-CLOSED */
'81408' = '1' /* FX HAMATE BONE-CLOSED */
'81409' = '1' /* FX CARPAL BONE NEC-CLOSE */
'8141'  = '1' /* FRACTURE CARPAL BONE-OPN */
'81410' = '1' /* FX CARPAL BONE NOS-OPEN */
'81411' = '1' /* FX NAVICULAR, WRIST-OPEN */
'81412' = '1' /* FX LUNATE, WRIST-OPEN */
'81413' = '1' /* FX TRIQUETRAL, WRIST-OPN */
'81414' = '1' /* FX PISIFORM-OPEN */
'81415' = '1' /* FX TRAPEZIUM BONE-OPEN */
```

PSFMTS -- CONTINUED

```
'81416' = '1' /* FX TRAPEZOID BONE-OPEN */
'81417' = '1' /* FX CAPITATE BONE-OPEN */
'81418' = '1' /* FX HAMATE BONE-OPEN */
'81419' = '1' /* FX CARPAL BONE NEC-OPEN */
'815'   = '1' /* METACARPAL FRACTURE */
'8150'  = '1' /* FRACTURE METACARPAL-CLOS */
'81500' = '1' /* FX METACARPAL NOS-CLOSED */
```

```
'81501' = '1' /* FX 1ST METACARP BASE-CL */
'81502' = '1' /* FX METACARP BASE NEC-CL */
'81503' = '1' /* FX METACARPAL SHAFT-CLOS */
'81504' = '1' /* FX METACARPAL NECK-CLOSE */
'81509' = '1' /* MULT FX METACARPUS-CLOSE */
'8151'  = '1' /* FRACTURE METACARPAL-OPEN */
'81510' = '1' /* FX METACARPAL NOS-OPEN */
'81511' = '1' /* FX 1ST METACARP BASE-OPN */
'81512' = '1' /* FX METACARP BASE NEC-OPN */
'81513' = '1' /* FX METACARPAL SHAFT-OPEN */
'81514' = '1' /* FX METACARPAL NECK-OPEN */
'81519' = '1' /* MULT FX METACARPUS-OPEN */
'817'   = '1' /* MULTIPLE HAND FRACTURES */
'8170'  = '1' /* MULTIPLE FX HAND-CLOSED */
'8171'  = '1' /* MULTIPLE FX HAND-OPEN */
'818'   = '1' /* FRACTURE ARM MULT/NOS */
'8180'  = '1' /* FX ARM MULT/NOS-CLOSED */
'8181'  = '1' /* FX ARM MULT/NOS-OPEN */
'819'   = '1' /* FX ARMS W RIB/STERNUM */
'8190'  = '1' /* FX ARMS W RIB/STERNUM-CL */
'8191'  = '1' /* FX ARMS W RIB/STERN-OPEN */
'820'   = '1' /* FRACTURE NECK OF FEMUR */
'8200'  = '1' /* TRANSCERV FX FEMUR-CLOSE */
'82000' = '1' /* FX FEMUR INTRCAPS NOS-CL */
'82001' = '1' /* FX UP FEMUR EPIPHY-CLOS */
'82002' = '1' /* FX FEMUR, MIDCERVIC-CLOS */
'82003' = '1' /* FX BASE FEMORAL NCK-CLOS */
'82009' = '1' /* FX FEMUR INTRCAPS NEC-CL */
'8201'  = '1' /* TRANSCERV FX FEMUR-OPEN */
'82010' = '1' /* FX FEMUR INTRCAP NOS-OPN */
'82011' = '1' /* FX UP FEMUR EPIPHY-OPEN */
'82012' = '1' /* FX FEMUR, MIDCERVIC-OPEN */
'82013' = '1' /* FX BASE FEMORAL NCK-OPEN */
'82019' = '1' /* FX FEMUR INTRCAP NEC-OPN */
'8202'  = '1' /* PERTROCHANTERIC FX-CLOS */
'82020' = '1' /* TROCHANTERIC FX NOS-CLOS */
'82021' = '1' /* INTERTROCHANTERIC FX-CL */
'82022' = '1' /* SUBTROCHANTERIC FX-CLOSE */
'8203'  = '1' /* PERTROCHANTERIC FX-OPEN */
'82030' = '1' /* TROCHANTERIC FX NOS-OPEN */
'82031' = '1' /* INTERTROCHANTERIC FX-OPN */
'82032' = '1' /* SUBTROCHANTERIC FX-OPEN */
'8208'  = '1' /* FX NECK OF FEMUR NOS-CL */
'8209'  = '1' /* FX NECK OF FEMUR NOS-OPN */
'821'   = '1' /* OTHER FEMORAL FRACTURE */
'8210'  = '1' /* FX FEMUR SHAFT/NOS-CLOSE */
'82100' = '1' /* FX FEMUR NOS-CLOSED */
'82101' = '1' /* FX FEMUR SHAFT-CLOSED */
'8211'  = '1' /* FX FEMUR SHAFT/NOS-OPEN */
'82110' = '1' /* FX FEMUR NOS-OPEN */
'82111' = '1' /* FX FEMUR SHAFT-OPEN */
'8212'  = '1' /* FX LOWER END FEMUR-CLOSE */
```

PSFMTS -- CONTINUED

```
'82120' = '1' /* FX LOW END FEMUR NOS-CL */
'82121' = '1' /* FX FEMORAL CONDYLE-CLOSE */
'82122' = '1' /* FX LOW FEMUR EPIPHY-CLOS */
'82123' = '1' /* SUPRACONDYL FX FEMUR-CL */
```

```
'82129' = '1' /* FX LOW END FEMUR NEC-CL */
'8213'  = '1' /* FX LOWER END FEMUR-OPEN */
'82130' = '1' /* FX LOW END FEMUR NOS-OPN */
'82131' = '1' /* FX FEMORAL CONDYLE-OPEN */
'82132' = '1' /* FX LOW FEMUR EPIPHY-OPEN */
'82133' = '1' /* SUPRACONDYL FX FEMUR-OPN */
'82139' = '1' /* FX LOW END FEMUR NEC-OPN */
'822'   = '1' /* PATELLA FRACTURE */
'8220'  = '1' /* FRACTURE PATELLA-CLOSED */
'8221'  = '1' /* FRACTURE PATELLA-OPEN */
'823'   = '1' /* TIBIA & FIBULA FRACTURE */
'8230'  = '1' /* FX UPPER TIBIA/FIBULA-CL */
'82300' = '1' /* FX UPPER END TIBIA-CLOSE */
'82301' = '1' /* FX UPPER END FIBULA-CLOS */
'82302' = '1' /* FX UP TIBIA W FIBULA-CL */
'8231'  = '1' /* FX UPPER TIBIA/FIBULA-OP */
'82310' = '1' /* FX UPPER END TIBIA-OPEN */
'82311' = '1' /* FX UPPER END FIBULA-OPEN */
'82312' = '1' /* FX UP TIBIA W FIBULA-OPN */
'8232'  = '1' /* FX SHAFT TIB/FIB-CLOSED */
'82320' = '1' /* FX SHAFT TIBIA-CLOSED */
'82321' = '1' /* FX SHAFT FIBULA-CLOSED */
'82322' = '1' /* FX SHAFT FIB W TIB-CLOS */
'8233'  = '1' /* FX TIBIA/FIBULA SHAFT-OP */
'82330' = '1' /* FX TIBIA SHAFT-OPEN */
'82331' = '1' /* FX FIBULA SHAFT-OPEN */
'82332' = '1' /* FX SHAFT TIBIA W FIB-OPN */
'8238'  = '1' /* FX TIBIA/FIBULA NOS-CLOS */
'82380' = '1' /* FX TIBIA NOS-CLOSED */
'82381' = '1' /* FX FIBULA NOS-CLOSED */
'82382' = '1' /* FX TIBIA W FIBULA NOS-CL */
'8239'  = '1' /* FX TIBIA/FIBULA NOS-OPEN */
'82390' = '1' /* FX TIBIA NOS-OPEN */
'82391' = '1' /* FX FIBULA NOS-OPEN */
'82392' = '1' /* FX TIBIA W FIB NOS-OPEN */
'824'   = '1' /* ANKLE FRACTURE */
'8240'  = '1' /* FX MEDIAL MALLEOLUS-CLOS */
'8241'  = '1' /* FX MEDIAL MALLEOLUS-OPEN */
'8242'  = '1' /* FX LATERAL MALLEOLUS-CL */
'8243'  = '1' /* FX LATERAL MALLEOLUS-OPN */
'8244'  = '1' /* FX BIMALLEOLAR-CLOSED */
'8245'  = '1' /* FX BIMALLEOLAR-OPEN */
'8246'  = '1' /* FX TRIMALLEOLAR-CLOSED */
'8247'  = '1' /* FX TRIMALLEOLAR-OPEN */
'8248'  = '1' /* FX ANKLE NOS-CLOSED */
'8249'  = '1' /* FX ANKLE NOS-OPEN */
'825'   = '1' /* FX OF TARSAL/METATARSAL */
'8250'  = '1' /* FRACTURE CALCANEUS-CLOSE */
'8251'  = '1' /* FRACTURE CALCANEUS-OPEN */
'8252'  = '1' /* FX TARSL/METATARS NEC-CL */
'82520' = '1' /* FX FOOT BONE NOS-CLOSED */
'82521' = '1' /* FX ASTRAGALUS-CLOSED */
'82522' = '1' /* FX NAVICULAR, FOOT-CLOS */
'82523' = '1' /* FX CUBOID-CLOSED */
'82524' = '1' /* FX CUNEIFORM, FOOT-CLOS */
PSFMTS -- CONTINUED

'82525' = '1' /* FX METATARSAL-CLOSED */
```

```
'82529' = '1' /* FX FOOT BONE NEC-CLOSED */
'8253'  = '1' /* FX TARS/METATARS NEC-OPN */
'82530' = '1' /* FX FOOT BONE NOS-OPEN */
'82531' = '1' /* FX ASTRAGALUS-OPEN */
'82532' = '1' /* FX NAVICULAR, FOOT-OPEN */
'82533' = '1' /* FX CUBOID-OPEN */
'82534' = '1' /* FX CUNEIFORM, FOOT-OPEN */
'82535' = '1' /* FX METATARSAL-OPEN */
'82539' = '1' /* FX FOOT BONE NEC-OPEN */
'827'   = '1' /* LOWER LIMB FRACTURE NEC */
'8270'  = '1' /* FX LOWER LIMB NEC-CLOSED */
'8271'  = '1' /* FX LOWER LIMB NEC-OPEN */
'828'   = '1' /* FX LEGS W ARM/RIB */
'8280'  = '1' /* FX LEGS W ARM/RIB-CLOSED */
'8281'  = '1' /* FX LEGS W ARM/RIB-OPEN */
'829'   = '1' /* FRACTURE NOS */
'8290'  = '1' /* FRACTURE NOS-CLOSED */
'8291'  = '1' /* FRACTURE NOS-OPEN */
'830'   = '1' /* JAW DISLOCATION */
'8300'  = '1' /* DISLOCATION JAW-CLOSED */
'8301'  = '1' /* DISLOCATION JAW-OPEN */
'831'   = '1' /* SHOULDER DISLOCATION */
'8310'  = '1' /* DISLOCAT SHOULDER-CLOSED */
'83100' = '1' /* DISLOC SHOULDER NOS-CLOS */
'83101' = '1' /* ANT DISLOC HUMERUS-CLOSE */
'83102' = '1' /* POST DISLOC HUMERUS-CLOS */
'83103' = '1' /* INFER DISLOC HUMERUS-CL */
'83104' = '1' /* DISLOC ACROMIOCLAVIC-CL */
'83109' = '1' /* DISLOC SHOULDER NEC-CLOS */
'8311'  = '1' /* DISLOCAT SHOULDER-OPEN */
'83110' = '1' /* DISLOC SHOULDER NOS-OPEN */
'83111' = '1' /* ANT DISLOC HUMERUS-OPEN */
'83112' = '1' /* POST DISLOC HUMERUS-OPEN */
'83113' = '1' /* INFER DISLOC HUMERUS-OPN */
'83114' = '1' /* DISLOC ACROMIOCLAVIC-OPN */
'83119' = '1' /* DISLOC SHOULDER NEC-OPEN */
'832'   = '1' /* ELBOW DISLOCATION */
'8320'  = '1' /* DISLOCATION ELBOW-CLOSED */
'83200' = '1' /* DISLOCAT ELBOW NOS-CLOSE */
'83201' = '1' /* ANT DISLOC ELBOW-CLOSED */
'83202' = '1' /* POST DISLOC ELBOW-CLOSED */
'83203' = '1' /* MED DISLOC ELBOW-CLOSED */
'83204' = '1' /* LAT DISLOC ELBOW-CLOSED */
'83209' = '1' /* DISLOCAT ELBOW NEC-CLOSE */
'8321'  = '1' /* DISLOCATION ELBOW-OPEN */
'83210' = '1' /* DISLOCAT ELBOW NOS-OPEN */
'83211' = '1' /* ANT DISLOC ELBOW-OPEN */
'83212' = '1' /* POST DISLOC ELBOW-OPEN */
'83213' = '1' /* MED DISLOC ELBOW-OPEN */
'83214' = '1' /* LAT DISLOCAT ELBOW-OPEN */
'83219' = '1' /* DISLOCAT ELBOW NEC-OPEN */
'833'   = '1' /* WRIST DISLOCATION */
'8330'  = '1' /* DISLOCATION WRIST-CLOSED */
'83300' = '1' /* DISLOC WRIST NOS-CLOSED */
'83301' = '1' /* DISLOC DIST RADIOULN-CL */
'83302' = '1' /* DISLOC RADIOCARPAL-CLOS */
'83303' = '1' /* DISLOCA MIDCARPAL-CLOSED */
'83304' = '1' /* DISLOC CARPOMETACARP-CL */
```

PSFMTS -- CONTINUED

```
'83305' = '1' /* DISLOC METACARPAL-CLOSED */
'83309' = '1' /* DISLOC WRIST NEC-CLOSED */
'8331'  = '1' /* DISLOCATION WRIST-OPEN */
'83310' = '1' /* DISLOCAT WRIST NOS-OPEN */
'83311' = '1' /* DISLOC DIST RADIOULN-OPN */
'83312' = '1' /* DISLOC RADIOCARPAL-OPEN */
'83313' = '1' /* DISLOCAT MIDCARPAL-OPEN */
'83314' = '1' /* DISLOC CARPOMETACARP-OPN */
'83315' = '1' /* DISLOCAT METACARPAL-OPEN */
'83319' = '1' /* DISLOCAT WRIST NEC-OPEN */
'835'   = '1' /* DISLOCATION OF HIP */
'8350'  = '1' /* DISLOCATION HIP-CLOSED */
'83500' = '1' /* DISLOCAT HIP NOS-CLOSED */
'83501' = '1' /* POSTERIOR DISLOC HIP-CL */
'83502' = '1' /* OBTURATOR DISLOC HIP-CL */
'83503' = '1' /* ANT DISLOC HIP NEC-CLOS */
'8351'  = '1' /* DISLOCATION HIP-OPEN */
'83510' = '1' /* DISLOCATION HIP NOS-OPEN */
'83511' = '1' /* POSTERIOR DISLOC HIP-OPN */
'83512' = '1' /* OBTURATOR DISLOC HIP-OPN */
'83513' = '1' /* ANT DISLOC HIP NEC-OPEN */
'836'   = '1' /* DISLOCATION OF KNEE */
'8360'  = '1' /* TEAR MED MENISC KNEE-CUR */
'8361'  = '1' /* TEAR LAT MENISC KNEE-CUR */
'8362'  = '1' /* TEAR MENISCUS NEC-CURREN */
'8363'  = '1' /* DISLOCAT PATELLA-CLOSED */
'8364'  = '1' /* DISLOCATION PATELLA-OPEN */
'8365'  = '1' /* OTH DISLOCAT KNEE-CLOSED */
'83650' = '1' /* DISLOCAT KNEE NOS-CLOSED */
'83651' = '1' /* ANT DISLOC PROX TIBIA-CL */
'83652' = '1' /* POST DISL PROX TIBIA-CL */
'83653' = '1' /* MED DISLOC PROX TIBIA-CL */
'83654' = '1' /* LAT DISLOC PROX TIBIA-CL */
'83659' = '1' /* DISLOCAT KNEE NEC-CLOSED */
'8366'  = '1' /* OTH DISLOCAT KNEE-OPEN */
'83660' = '1' /* DISLOCAT KNEE NOS-OPEN */
'83661' = '1' /* ANT DISL PROX TIBIA-OPEN */
'83662' = '1' /* POST DISL PROX TIBIA-OPN */
'83663' = '1' /* MED DISL PROX TIBIA-OPEN */
'83664' = '1' /* LAT DISL PROX TIBIA-OPEN */
'83669' = '1' /* DISLOCAT KNEE NEC-OPEN */
'837'   = '1' /* DISLOCATION OF ANKLE */
'8370'  = '1' /* DISLOCATION ANKLE-CLOSED */
'8371'  = '1' /* DISLOCATION ANKLE-OPEN */
'838'   = '1' /* DISLOCATION OF FOOT */
'8380'  = '1' /* DISLOCATION FOOT-CLOSED */
'83800' = '1' /* DISLOCAT FOOT NOS-CLOSED */
'83801' = '1' /* DISLOC TARSAL NOS-CLOSED */
'83802' = '1' /* DISLOC MIDTARSAL-CLOSED */
'83803' = '1' /* DISLOC TARSOMETATARS-CL */
'83804' = '1' /* DISLOC METATARSAL NOS-CL */
'83805' = '1' /* DISL METATARSOPHALANG-CL */
'83806' = '1' /* DISL INTERPHALAN FOOT-CL */
'83809' = '1' /* DISLOCAT FOOT NEC-CLOSED */
'8381'  = '1' /* DISLOCATION FOOT-OPEN */
'83810' = '1' /* DISLOCAT FOOT NOS-OPEN */
```

```
'83811' = '1' /* DISLOC TARSAL NOS-OPEN */
'83812' = '1' /* DISLOC MIDTARSAL-OPEN */
'83813' = '1' /* DISL TARSOMETATARSAL-OPN */
```
PSFMTS -- CONTINUED

```
'83814' = '1' /* DISL METATARSAL NOS-OPEN */
'83815' = '1' /* DISLOC METATARSOPHAL-OPN */
'83816' = '1' /* DIS INTERPHALAN FOOT-OPN */
'83819' = '1' /* DISLOCAT FOOT NEC-OPEN */
'839'   = '1' /* DISLOCATION NEC */
'8390'  = '1' /* DISLOC CERVICAL VERT-CL */
'83900' = '1' /* DISLOC CERV VERT NOS-CL */
'83901' = '1' /* DISLOC 1ST CERV VERT-CL */
'83902' = '1' /* DISLOC 2ND CERV VERT-CL */
'83903' = '1' /* DISLOC 3RD CERV VERT-CL */
'83904' = '1' /* DISLOC 4TH CERV VERT-CL */
'83905' = '1' /* DISLOC 5TH CERV VERT-CL */
'83906' = '1' /* DISLOC 6TH CERV VERT-CL */
'83907' = '1' /* DISLOC 7TH CERV VERT-CL */
'83908' = '1' /* DISLOC MULT CERV VERT-CL */
'8391'  = '1' /* DISLOC CERVICAL VERT-OPN */
'83910' = '1' /* DISLOC CERV VERT NOS-OPN */
'83911' = '1' /* DISLOC LST CERV VERT-OPN */
'83912' = '1' /* DISLOC 2ND CERV VERT-OPN */
'83913' = '1' /* DISLOC 3RD CERV VERT-OPN */
'83914' = '1' /* DISLOC 4TH CERV VERT-OPN */
'83915' = '1' /* DISLOC 5TH CERV VERT-OPN */
'83916' = '1' /* DISLOC 6TH CERV VERT-OPN */
'83917' = '1' /* DISLOC 7TH CERV VERT-OPN */
'83918' = '1' /* DISLOC MLT CERV VERT-OPN */
'8392'  = '1' /* DISLOC THOR/LUMB VERT-CL */
'83920' = '1' /* DISLOCAT LUMBAR VERT-CL */
'83921' = '1' /* DISLOC THORACIC VERT-CL */
'8393'  = '1' /* DISL THORA/LUMB VERT-OPN */
'83930' = '1' /* DISLOCAT LUMBAR VERT-OPN */
'83931' = '1' /* DISLOC THORACIC VERT-OPN */
'8394'  = '1' /* OTH DISLOCAT VERTEBRA-CL */
'83940' = '1' /* DISLOCAT VERTEBRA NOS-CL */
'83941' = '1' /* DISLOCAT COCCYX-CLOSED */
'83942' = '1' /* DISLOCAT SACRUM-CLOSED */
'83949' = '1' /* DISLOCAT VERTEBRA NEC-CL */
'8395'  = '1' /* OTH DISLOC VERTEBRA-OPEN */
'83950' = '1' /* DISLOC VERTEBRA NOS-OPEN */
'83951' = '1' /* DISLOCAT COCCYX-OPEN */
'83952' = '1' /* DISLOCAT SACRUM-OPEN */
'83959' = '1' /* DISLOC VERTEBRA NEC-OPEN */
'8396'  = '1' /* DISLOCAT OTH SITE-CLOSED */
'83961' = '1' /* DISLOCAT STERNUM-CLOSED */
'83969' = '1' /* DISLOCAT SITE NEC-CLOSED */
'8397'  = '1' /* DISLOCAT OTH SITE-OPEN */
'83971' = '1' /* DISLOCATION STERNUM-OPEN */
'83979' = '1' /* DISLOCAT SITE NEC-OPEN */
'8398'  = '1' /* DISLOCATION NEC-CLOSED */
'8399'  = '1' /* DISLOCATION NEC-OPEN */
'850'   = '1' /* CONCUSSION */
'8500'  = '1' /* CONCUSSION W/O COMA */
'8501'  = '1' /* CONCUSSION-BRIEF COMA */
'8502'  = '1' /* CONCUSSION-MODERATE COMA */
```

```
'8503'  = '1' /* CONCUSSION-PROLONG COMA */
'8504'  = '1' /* CONCUSSION-DEEP COMA */
'8505'  = '1' /* CONCUSSION W COMA NOS */
'8509'  = '1' /* CONCUSSION NOS */
'851'   = '1' /* CEREBRAL LACER/CONTUSION */
'8510'  = '1' /* CORTEX CONTUSION */
```

PSFMTS -- CONTINUED

```
'85100' = '1' /* CEREBRAL CORTX CONTUSION */
'85101' = '1' /* CORTEX CONTUSION-NO COMA */
'85102' = '1' /* CORTEX CONTUS-BRIEF COMA */
'85103' = '1' /* CORTEX CONTUS-MOD COMA */
'85104' = '1' /* CORTX CONTUS-PROLNG COMA */
'85105' = '1' /* CORTEX CONTUS-DEEP COMA */
'85106' = '1' /* CORTEX CONTUS-COMA NOS */
'85109' = '1' /* CORTEX CONTUS-CONCUS NOS */
'8511'  = '1' /* CORTEX CONTUSION/OPN WND */
'85110' = '1' /* CORTEX CONTUSION/OPN WND */
'85111' = '1' /* OPN CORTX CONTUS-NO COMA */
'85112' = '1' /* OPN CORT CONTUS-BRF COMA */
'85113' = '1' /* OPN CORT CONTUS-MOD COMA */
'85114' = '1' /* OPN CORT CONTU-PROL COMA */
'85115' = '1' /* OPN CORT CONTU-DEEP COMA */
'85116' = '1' /* OPN CORT CONTUS-COMA NOS */
'85119' = '1' /* OPN CORTX CONTUS-CONCUSS */
'8512'  = '1' /* CEREBRAL CORTEX LACERAT */
'85120' = '1' /* CEREBRAL CORTEX LACERAT */
'85121' = '1' /* CORTEX LACERAT W/O COMA */
'85122' = '1' /* CORTEX LACERA-BRIEF COMA */
'85123' = '1' /* CORTEX LACERAT-MOD COMA */
'85124' = '1' /* CORTEX LACERAT-PROL COMA */
'85125' = '1' /* CORTEX LACERAT-DEEP COMA */
'85126' = '1' /* CORTEX LACERAT-COMA NOS */
'85129' = '1' /* CORTEX LACERAT-CONCUSS */
'8513'  = '1' /* CORTEX LACER W OPN WOUND */
'85130' = '1' /* CORTEX LACER W OPN WOUND */
'85131' = '1' /* OPN CORTEX LACER-NO COMA */
'85132' = '1' /* OPN CORTX LAC-BRIEF COMA */
'85133' = '1' /* OPN CORTX LACER-MOD COMA */
'85134' = '1' /* OPN CORTX LAC-PROLN COMA */
'85135' = '1' /* OPN CORTEX LAC-DEEP COMA */
'85136' = '1' /* OPN CORTX LACER-COMA NOS */
'85139' = '1' /* OPN CORTX LACER-CONCUSS */
'8514'  = '1' /* CEREBEL/BRAIN STM CONTUS */
'85140' = '1' /* CEREBEL/BRAIN STM CONTUS */
'85141' = '1' /* CEREBELL CONTUS W/O COMA */
'85142' = '1' /* CEREBELL CONTUS-BRF COMA */
'85143' = '1' /* CEREBELL CONTUS-MOD COMA */
'85144' = '1' /* CEREBEL CONTUS-PROL COMA */
'85145' = '1' /* CEREBEL CONTUS-DEEP COMA */
'85146' = '1' /* CEREBELL CONTUS-COMA NOS */
'85149' = '1' /* CEREBELL CONTUS-CONCUSS */
'8515'  = '1' /* CEREBEL CONTUS W OPN WND */
'85150' = '1' /* CEREBEL CONTUS W OPN WND */
'85151' = '1' /* OPN CEREBE CONT W/O COMA */
'85152' = '1' /* OPN CEREBE CONT-BRF COMA */
'85153' = '1' /* OPN CEREBE CONT-MOD COMA */
'85154' = '1' /* OPN CEREBE CONT-PROL COM */
```

```
'85155' = '1' /* OPN CEREBE CONT-DEEP COM */
'85156' = '1' /* OPN CEREBE CONT-COMA NOS */
'85159' = '1' /* OPN CEREBEL CONT-CONCUSS */
'8516'  = '1' /* CEREBEL/BRAIN STEM LACER */
'85160' = '1' /* CEREBEL/BRAIN STEM LACER */
'85161' = '1' /* CEREBEL LACERAT W/O COMA */
'85162' = '1' /* CEREBEL LACER-BRIEF COMA */
'85163' = '1' /* CEREBEL LACERAT-MOD COMA */
'85164' = '1' /* CEREBEL LACER-PROLN COMA */
```

PSFMTS -- CONTINUED

```
'85165' = '1' /* CEREBELL LACER-DEEP COMA */
'85166' = '1' /* CEREBEL LACERAT-COMA NOS */
'85169' = '1' /* CEREBEL LACER-CONCUSSION */
'8517'  = '1' /* CEREBEL LACER W OPEN WND */
'85170' = '1' /* CEREBEL LACER W OPEN WND */
'85171' = '1' /* OPN CEREBEL LAC W/O COMA */
'85172' = '1' /* OPN CEREBEL LAC-BRF COMA */
'85173' = '1' /* OPN CEREBEL LAC-MOD COMA */
'85174' = '1' /* OPN CEREBE LAC-PROL COMA */
'85175' = '1' /* OPN CEREBE LAC-DEEP COMA */
'85176' = '1' /* OPN CEREBEL LAC-COMA NOS */
'85179' = '1' /* OPN CEREBELL LAC-CONCUSS */
'8518'  = '1' /* CEREBRAL LACERATION NEC */
'85180' = '1' /* BRAIN LACERATION NEC */
'85181' = '1' /* BRAIN LACER NEC W/O COMA */
'85182' = '1' /* BRAIN LAC NEC-BRIEF COMA */
'85183' = '1' /* BRAIN LACER NEC-MOD COMA */
'85184' = '1' /* BRAIN LAC NEC-PROLN COMA */
'85185' = '1' /* BRAIN LAC NEC-DEEP COMA */
'85186' = '1' /* BRAIN LACER NEC-COMA NOS */
'85189' = '1' /* BRAIN LACER NEC-CONCUSS */
'8519'  = '1' /* BRAIN LAC NEC W OPEN WND */
'85190' = '1' /* BRAIN LAC NEC W OPEN WND */
'85191' = '1' /* OPN BRAIN LACER W/O COMA */
'85192' = '1' /* OPN BRAIN LAC-BRIEF COMA */
'85193' = '1' /* OPN BRAIN LACER-MOD COMA */
'85194' = '1' /* OPN BRAIN LAC-PROLN COMA */
'85195' = '1' /* OPEN BRAIN LAC-DEEP COMA */
'85196' = '1' /* OPN BRAIN LACER-COMA NOS */
'85199' = '1' /* OPEN BRAIN LACER-CONCUSS */
'852'   = '1' /* MENINGEAL HEM FOLLOW INJ */
'8520'  = '1' /* TRAUM SUBARACHNOID HEM */
'85200' = '1' /* TRAUM SUBARACHNOID HEM */
'85201' = '1' /* SUBARACHNOID HEM-NO COMA */
'85202' = '1' /* SUBARACH HEM-BRIEF COMA */
'85203' = '1' /* SUBARACH HEM-MOD COMA */
'85204' = '1' /* SUBARACH HEM-PROLNG COMA */
'85205' = '1' /* SUBARACH HEM-DEEP COMA */
'85206' = '1' /* SUBARACH HEM-COMA NOS */
'85209' = '1' /* SUBARACH HEM-CONCUSSION */
'8521'  = '1' /* SUBARACH HEM W OPN WOUND */
'85210' = '1' /* SUBARACH HEM W OPN WOUND */
'85211' = '1' /* OPN SUBARACH HEM-NO COMA */
'85212' = '1' /* OP SUBARACH HEM-BRF COMA */
'85213' = '1' /* OP SUBARACH HEM-MOD COMA */
'85214' = '1' /* OP SUBARACH HEM-PROL COM */
'85215' = '1' /* OP SUBARACH HEM-DEEP COM */
```

```
'85216' = '1' /* OP SUBARACH HEM-COMA NOS */
'85219' = '1' /* OPN SUBARACH HEM-CONCUSS */
'8522'  = '1' /* TRAUMATIC SUBDURAL HEM */
'85220' = '1' /* TRAUMATIC SUBDURAL HEM */
'85221' = '1' /* SUBDURAL HEM W/O COMA */
'85222' = '1' /* SUBDURAL HEM-BRIEF COMA */
'85223' = '1' /* SUBDURAL HEMORR-MOD COMA */
'85224' = '1' /* SUBDURAL HEM-PROLNG COMA */
'85225' = '1' /* SUBDURAL HEM-DEEP COMA */
'85226' = '1' /* SUBDURAL HEMORR-COMA NOS */
'85229' = '1' /* SUBDURAL HEM-CONCUSSION */
'8523'  = '1' /* SUBDURAL HEM W OPN WOUND */
```
PSFMTS -- CONTINUED

```
'85230' = '1' /* SUBDURAL HEM W OPN WOUND */
'85231' = '1' /* OPEN SUBDUR HEM W/O COMA */
'85232' = '1' /* OPN SUBDUR HEM-BRF COMA */
'85233' = '1' /* OPN SUBDUR HEM-MOD COMA */
'85234' = '1' /* OPN SUBDUR HEM-PROL COMA */
'85235' = '1' /* OPN SUBDUR HEM-DEEP COMA */
'85236' = '1' /* OPN SUBDUR HEM-COMA NOS */
'85239' = '1' /* OPN SUBDUR HEM-CONCUSS */
'8524'  = '1' /* TRAUMATIC EXTRADURAL HEM */
'85240' = '1' /* TRAUMATIC EXTRADURAL HEM */
'85241' = '1' /* EXTRADURAL HEM W/O COMA */
'85242' = '1' /* EXTRADUR HEM-BRIEF COMA */
'85243' = '1' /* EXTRADURAL HEM-MOD COMA */
'85244' = '1' /* EXTRADUR HEM-PROLN COMA */
'85245' = '1' /* EXTRADURAL HEM-DEEP COMA */
'85246' = '1' /* EXTRADURAL HEM-COMA NOS */
'85249' = '1' /* EXTADURAL HEM-CONCUSS */
'8525'  = '1' /* EXTRADURAL HEM W OPN WND */
'85250' = '1' /* EXTRADURAL HEM W OPN WND */
'85251' = '1' /* EXTRADURAL HEMOR-NO COMA */
'85252' = '1' /* EXTRADUR HEM-BRIEF COMA */
'85253' = '1' /* EXTRADURAL HEM-MOD COMA */
'85254' = '1' /* EXTRADUR HEM-PROLN COMA */
'85255' = '1' /* EXTRADUR HEM-DEEP COMA */
'85256' = '1' /* EXTRADURAL HEM-COMA NOS */
'85259' = '1' /* EXTRADURAL HEM-CONCUSS */
'853'   = '1' /* OTH TRAUMATIC BRAIN HEM */
'8530'  = '1' /* TRAUMATIC BRAIN HEM NEC */
'85300' = '1' /* TRAUMATIC BRAIN HEM NEC */
'85301' = '1' /* BRAIN HEM NEC W/O COMA */
'85302' = '1' /* BRAIN HEM NEC-BRIEF COMA */
'85303' = '1' /* BRAIN HEM NEC-MOD COMA */
'85304' = '1' /* BRAIN HEM NEC-PROLN COMA */
'85305' = '1' /* BRAIN HEM NEC-DEEP COMA */
'85306' = '1' /* BRAIN HEM NEC-COMA NOS */
'85309' = '1' /* BRAIN HEM NEC-CONCUSSION */
'8531'  = '1' /* BRAIN HEM NEC W OPN WND */
'85310' = '1' /* BRAIN HEM NEC W OPN WND */
'85311' = '1' /* BRAIN HEM OPN W/O COMA */
'85312' = '1' /* BRAIN HEM OPN-BRF COMA */
'85313' = '1' /* BRAIN HEM OPEN-MOD COMA */
'85314' = '1' /* BRAIN HEM OPN-PROLN COMA */
'85315' = '1' /* BRAIN HEM OPEN-DEEP COMA */
'85316' = '1' /* BRAIN HEM OPEN-COMA NOS */
```

```
'85319' = '1' /* BRAIN HEM OPN-CONCUSSION */
'854'   = '1' /* OTHER BRAIN INJURY */
'8540'  = '1' /* BRAIN INJURY NEC */
'85400' = '1' /* BRAIN INJURY NEC */
'85401' = '1' /* BRAIN INJURY NEC-NO COMA */
'85402' = '1' /* BRAIN INJ NEC-BRIEF COMA */
'85403' = '1' /* BRAIN INJ NEC-MOD COMA */
'85404' = '1' /* BRAIN INJ NEC-PROLN COMA */
'85405' = '1' /* BRAIN INJ NEC-DEEP COMA */
'85406' = '1' /* BRAIN INJ NEC-COMA NOS */
'85409' = '1' /* BRAIN INJ NEC-CONCUSSION */
'8541'  = '1' /* BRAIN INJ NEC W OPN WND */
'85410' = '1' /* BRAIN INJURY W OPN WND */
'85411' = '1' /* OPN BRAIN INJ W/O COMA */
'85412' = '1' /* OPN BRAIN INJ-BRIEF COMA */
```

PSFMTS -- CONTINUED

```
'85413' = '1' /* OPN BRAIN INJ-MOD COMA */
'85414' = '1' /* OPN BRAIN INJ-PROLN COMA */
'85415' = '1' /* OPN BRAIN INJ-DEEP COMA */
'85416' = '1' /* OPEN BRAIN INJ-COMA NOS */
'85419' = '1' /* OPN BRAIN INJ-CONCUSSION */
'860'   = '1' /* TRAUM PNEUMOHEMOTHORAX */
'8600'  = '1' /* TRAUM PNEUMOTHORAX-CLOSE */
'8601'  = '1' /* TRAUM PNEUMOTHORAX-OPEN */
'8602'  = '1' /* TRAUM HEMOTHORAX-CLOSED */
'8603'  = '1' /* TRAUM HEMOTHORAX-OPEN */
'8604'  = '1' /* TRAUM PNEUMOHEMOTHOR-CL */
'8605'  = '1' /* TRAUM PNEUMOHEMOTHOR-OPN */
'861'   = '1' /* HEART & LUNG INJURY */
'8610'  = '1' /* HEART INJURY-CLOSED */
'86100' = '1' /* HEART INJURY NOS-CLOSED */
'86101' = '1' /* HEART CONTUSION-CLOSED */
'86102' = '1' /* HEART LACERATION-CLOSED */
'86103' = '1' /* HEART CHAMBER LACERAT-CL */
'8611'  = '1' /* HEART INJURY-OPEN */
'86110' = '1' /* HEART INJURY NOS-OPEN */
'86111' = '1' /* HEART CONTUSION-OPEN */
'86112' = '1' /* HEART LACERATION-OPEN */
'86113' = '1' /* HEART CHAMBER LACER-OPN */
'8612'  = '1' /* LUNG INJURY-CLOSED */
'86120' = '1' /* LUNG INJURY NOS-CLOSED */
'86121' = '1' /* LUNG CONTUSION-CLOSED */
'86122' = '1' /* LUNG LACERATION-CLOSED */
'8613'  = '1' /* LUNG INJURY-OPEN */
'86130' = '1' /* LUNG INJURY NOS-OPEN */
'86131' = '1' /* LUNG CONTUSION-OPEN */
'86132' = '1' /* LUNG LACERATION-OPEN */
'862'   = '1' /* INTRATHORACIC INJURY NEC */
'8620'  = '1' /* DIAPHRAGM INJURY-CLOSED */
'8621'  = '1' /* DIAPHRAGM INJURY-OPEN */
'8622'  = '1' /* OTH INTRATHORACIC INJ-CL */
'86221' = '1' /* BRONCHUS INJURY-CLOSED */
'86222' = '1' /* ESOPHAGUS INJURY-CLOSED */
'86229' = '1' /* INTRATHORACIC INJ NEC-CL */
'8623'  = '1' /* OTH INTRATHORAC INJ-OPEN */
'86231' = '1' /* BRONCHUS INJURY-OPEN */
'86232' = '1' /* ESOPHAGUS INJURY-OPEN */
```

```
'86239' = '1' /* INTRATHORAC INJ NEC-OPEN */
'8628'  = '1' /* INTRATHORACIC INJ NOS-CL */
'8629'  = '1' /* INTRATHORAC INJ NOS-OPEN */
'863'   = '1' /* GI TRACT INJURY */
'8630'  = '1' /* STOMACH INJURY-CLOSED */
'8631'  = '1' /* STOMACH INJURY-OPEN */
'8632'  = '1' /* SMALL INTESTINE INJ-CLOS */
'86320' = '1' /* SMALL INTEST INJ NOS-CL */
'86321' = '1' /* DUODENUM INJURY-CLOSED */
'86329' = '1' /* SMALL INTEST INJ NEC-CL */
'8633'  = '1' /* SMALL INTESTINE INJ-OPEN */
'86330' = '1' /* SMALL INTEST INJ NOS-OPN */
'86331' = '1' /* DUODENUM INJURY-OPEN */
'86339' = '1' /* SMALL INTEST INJ NEC-OPN */
'8634'  = '1' /* COLON INJURY-CLOSED */
'86340' = '1' /* COLON INJURY NOS-CLOSED */
'86341' = '1' /* ASCENDING COLON INJ-CLOS */
'86342' = '1' /* TRANSVERSE COLON INJ-CL */
```
PSFMTS -- CONTINUED
```
'86343' = '1' /* DESCENDING COLON INJ-CL */
'86344' = '1' /* SIGMOID COLON INJ-CLOSED */
'86345' = '1' /* RECTUM INJURY-CLOSED */
'86346' = '1' /* COLON INJ MULT SITE-CLOS */
'86349' = '1' /* COLON INJURY NEC-CLOSED */
'8635'  = '1' /* COLON INJURY-OPEN */
'86350' = '1' /* COLON INJURY NOS-OPEN */
'86351' = '1' /* ASCENDING COLON INJ-OPEN */
'86352' = '1' /* TRANSVERSE COLON INJ-OPN */
'86353' = '1' /* DESCENDING COLON INJ-OPN */
'86354' = '1' /* SIGMOID COLON INJ-OPEN */
'86355' = '1' /* RECTUM INJURY-OPEN */
'86356' = '1' /* COLON INJ MULT SITE-OPEN */
'86359' = '1' /* COLON INJURY NEC-OPEN */
'8638'  = '1' /* GI INJURY NEC-CLOSED */
'86380' = '1' /* GI INJURY NOS-CLOSED */
'86381' = '1' /* PANCREAS, HEAD INJ-CLOSE */
'86382' = '1' /* PANCREAS, BODY INJ-CLOSE */
'86383' = '1' /* PANCREAS, TAIL INJ-CLOSE */
'86384' = '1' /* PANCREAS INJURY NOS-CLOS */
'86385' = '1' /* APPENDIX INJURY-CLOSED */
'86389' = '1' /* GI INJURY NEC-CLOSED */
'8639'  = '1' /* GI INJURY NEC-OPEN */
'86390' = '1' /* GI INJURY NOS-OPEN */
'86391' = '1' /* PANCREAS, HEAD INJ-OPEN */
'86392' = '1' /* PANCREAS, BODY INJ-OPEN */
'86393' = '1' /* PANCREAS, TAIL INJ-OPEN */
'86394' = '1' /* PANCREAS INJURY NOS-OPEN */
'86395' = '1' /* APPENDIX INJURY-OPEN */
'86399' = '1' /* GI INJURY NEC-OPEN */
'864'   = '1' /* LIVER INJURY */
'8640'  = '1' /* LIVER INJURY-CLOSED */
'86400' = '1' /* LIVER INJURY NOS-CLOSED */
'86401' = '1' /* LIVER HEMATOMA/CONTUSION */
'86402' = '1' /* LIVER LACERATION, MINOR */
'86403' = '1' /* LIVER LACERATION, MOD */
'86404' = '1' /* LIVER LACERATION, MAJOR */
'86405' = '1' /* LIVER LACERAT UNSPCF CLS */
```

```
'86409' = '1' /* LIVER INJURY NEC-CLOSED */
'8641'  = '1' /* LIVER INJURY-OPEN */
'86410' = '1' /* LIVER INJURY NOS-OPEN */
'86411' = '1' /* LIVER HEMATOM/CONTUS-OPN */
'86412' = '1' /* LIVER LACERAT, MINOR-OPN */
'86413' = '1' /* LIVER LACERAT, MOD-OPEN */
'86414' = '1' /* LIVER LACERAT, MAJOR-OPN */
'86415' = '1' /* LIVER LACERAT UNSPCF OPN */
'86419' = '1' /* LIVER INJURY NEC-OPEN */
'865'   = '1' /* SPLEEN INJURY */
'8650'  = '1' /* SPLEEN INJURY-CLOSED */
'86500' = '1' /* SPLEEN INJURY NOS-CLOSED */
'86501' = '1' /* SPLEEN HEMATOMA-CLOSED */
'86502' = '1' /* SPLEEN CAPSULAR TEAR */
'86503' = '1' /* SPLEEN PARENCHYMA LACER */
'86504' = '1' /* SPLEEN DISRUPTION-CLOS */
'86509' = '1' /* SPLEEN INJURY NEC-CLOSED */
'8651'  = '1' /* SPLEEN INJURY-OPEN */
'86510' = '1' /* SPLEEN INJURY NOS-OPEN */
'86511' = '1' /* SPLEEN HEMATOMA-OPEN */
'86512' = '1' /* SPLEEN CAPSULAR TEAR-OPN */
```
PSFMTS -- CONTINUED

```
'86513' = '1' /* SPLEEN PARNCHYM LAC-OPN */
'86514' = '1' /* SPLEEN DISRUPTION-OPEN */
'86519' = '1' /* SPLEEN INJURY NEC-OPEN */
'866'   = '1' /* KIDNEY INJURY */
'8660'  = '1' /* KIDNEY INJURY-CLOSED */
'86600' = '1' /* KIDNEY INJURY NOS-CLOSED */
'86601' = '1' /* KIDNEY HEMATOMA-CLOSED */
'86602' = '1' /* KIDNEY LACERATION-CLOSED */
'86603' = '1' /* KIDNEY DISRUPTION-CLOSED */
'8661'  = '1' /* KIDNEY INJURY-OPEN */
'86610' = '1' /* KIDNEY INJURY NOS-OPEN */
'86611' = '1' /* KIDNEY HEMATOMA-OPEN */
'86612' = '1' /* KIDNEY LACERATION-OPEN */
'86613' = '1' /* KIDNEY DISRUPTION-OPEN */
'867'   = '1' /* PELVIC ORGAN INJURY */
'8670'  = '1' /* BLADDER/URETHRA INJ-CLOS */
'8671'  = '1' /* BLADDER/URETHRA INJ-OPEN */
'8672'  = '1' /* URETER INJURY-CLOSED */
'8673'  = '1' /* URETER INJURY-OPEN */
'8674'  = '1' /* UTERUS INJURY-CLOSED */
'8675'  = '1' /* UTERUS INJURY-OPEN */
'8676'  = '1' /* PELVIC ORGAN INJ NEC-CL */
'8677'  = '1' /* PELVIC ORGAN INJ NEC-OPN */
'8678'  = '1' /* PELVIC ORGAN INJ NOS-CL */
'8679'  = '1' /* PELVIC ORGAN INJ NOS-OPN */
'868'   = '1' /* OTH INTRA-ABDOMINAL INJ */
'8680'  = '1' /* OTH INTRA-ABD INJ-CLOSED */
'86800' = '1' /* INTRA-ABDOM INJ NOS-CLOS */
'86801' = '1' /* ADRENAL GLAND INJURY-CL */
'86802' = '1' /* BILIARY TRACT INJURY-CL */
'86803' = '1' /* PERITONEUM INJURY-CLOSED */
'86804' = '1' /* RETROPERITONEUM INJ-CL */
'86809' = '1' /* INTRA-ABDOM INJ NEC-CLOS */
'8681'  = '1' /* OTH INTRA-ABD INJ-OPEN */
'86810' = '1' /* INTRA-ABDOM INJ NOS-OPEN */
```

```
'86811' = '1' /* ADRENAL GLAND INJURY-OPN */
'86812' = '1' /* BILIARY TRACT INJURY-OPN */
'86813' = '1' /* PERITONEUM INJURY-OPEN */
'86814' = '1' /* RETROPERITONEUM INJ-OPEN */
'86819' = '1' /* INTRA-ABDOM INJ NEC-OPEN */
'869'   = '1' /* INTERNAL INJURY NOS */
'8690'  = '1' /* INTERNAL INJ NOS-CLOSED */
'8691'  = '1' /* INTERNAL INJURY NOS-OPEN */
'870'   = '1' /* OCULAR ADNEXA OPEN WOUND */
'8700'  = '1' /* LAC EYELID SKN/PERIOCULR */
'8701'  = '1' /* FULL-THICKNES LAC EYELID */
'8702'  = '1' /* LAC EYELID INV LACRM PAS */
'8703'  = '1' /* PENETR WND ORBIT W/O FB */
'8704'  = '1' /* PENETRAT WND ORBIT W FB */
'8708'  = '1' /* OPN WND OCULAR ADNEX NEC */
'8709'  = '1' /* OPN WND OCULAR ADNEX NOS */
'871'   = '1' /* EYEBALL OPEN WOUND */
'8710'  = '1' /* OCULAR LAC W/O PROLAPSE */
'8711'  = '1' /* OCULAR LACERA W PROLAPSE */
'8712'  = '1' /* RUPTURE EYE W TISSU LOSS */
'8713'  = '1' /* AVULSION OF EYE */
'8714'  = '1' /* LACERATION OF EYE NOS */
'8715'  = '1' /* PENETRAT MAGNET FB EYE */
'8716'  = '1' /* PENETRAT FB NEC EYE */
```

PSFMTS -- CONTINUED

```
'8717'  = '1' /* OCULAR PENETRATION NOS */
'8719'  = '1' /* OPN WOUND OF EYEBALL NOS */
'872'   = '1' /* OPEN WOUND OF EAR */
'8720'  = '1' /* OPEN WOUND EXTERNAL EAR */
'87200' = '1' /* OPN WOUND EXTERN EAR NOS */
'87201' = '1' /* OPEN WOUND OF AURICLE */
'87202' = '1' /* OPN WOUND AUDITORY CANAL */
'8721'  = '1' /* OPEN WOUND EXT EAR-COMPL */
'87210' = '1' /* OPN WND EX EAR NOS-COMPL */
'87211' = '1' /* OPEN WOUND AURICLE-COMPL */
'87212' = '1' /* OPEN WND AUD CANAL-COMPL */
'8726'  = '1' /* OPEN WOUND OF EAR NEC */
'87261' = '1' /* OPEN WOUND OF EAR DRUM */
'87262' = '1' /* OPEN WOUND OF OSSICLES */
'87263' = '1' /* OPEN WND EUSTACHIAN TUBE */
'87264' = '1' /* OPEN WOUND OF COCHLEA */
'87269' = '1' /* OPEN WOUND OF EAR NEC */
'8727'  = '1' /* OPEN WOUND EAR NEC-COMPL */
'87271' = '1' /* OPEN WND EAR DRUM-COMPL */
'87272' = '1' /* OPEN WND OSSICLES-COMPL */
'87273' = '1' /* OPN WND EUSTACH TB-COMPL */
'87274' = '1' /* OPEN WOUND COCHLEA-COMPL */
'87279' = '1' /* OPEN WOUND EAR NEC-COMPL */
'8728'  = '1' /* OPEN WOUND OF EAR NOS */
'8729'  = '1' /* OPEN WOUND EAR NOS-COMPL */
'873'   = '1' /* OTHER OPEN WOUND OF HEAD */
'8730'  = '1' /* OPEN WOUND OF SCALP */
'8731'  = '1' /* OPEN WOUND SCALP-COMPL */
'8732'  = '1' /* OPEN WOUND OF NOSE */
'87320' = '1' /* OPEN WOUND OF NOSE NOS */
'87321' = '1' /* OPEN WOUND NASAL SEPTUM */
'87322' = '1' /* OPEN WOUND NASAL CAVITY */
```

```
'87323' = '1' /* OPEN WOUND NASAL SINUS */
'87329' = '1' /* MULT OPEN WOUND NOSE */
'8733'  = '1' /* OPEN WOUND NOSE-COMPL */
'87330' = '1' /* OPEN WND NOSE NOS-COMPL */
'87331' = '1' /* OPN WND NAS SEPTUM-COMPL */
'87332' = '1' /* OPEN WND NASAL CAV-COMPL */
'87333' = '1' /* OPEN WND NAS SINUS-COMPL */
'87339' = '1' /* MULT OPEN WND NOSE-COMPL */
'8734'  = '1' /* OPEN WOUND OF FACE */
'87340' = '1' /* OPEN WOUND OF FACE NOS */
'87341' = '1' /* OPEN WOUND OF CHEEK */
'87342' = '1' /* OPEN WOUND OF FOREHEAD */
'87343' = '1' /* OPEN WOUND OF LIP */
'87344' = '1' /* OPEN WOUND OF JAW */
'87349' = '1' /* OPEN WOUND OF FACE NEC */
'8735'  = '1' /* OPEN WOUND FACE-COMPL */
'87350' = '1' /* OPEN WND FACE NOS-COMPL */
'87351' = '1' /* OPEN WOUND CHEEK-COMPL */
'87352' = '1' /* OPEN WND FOREHEAD-COMPL */
'87353' = '1' /* OPEN WOUND LIP-COMPLICAT */
'87354' = '1' /* OPEN WOUND JAW-COMPLICAT */
'87359' = '1' /* OPEN WND FACE NEC-COMPL */
'8736'  = '1' /* OPN WOUND INTERNAL MOUTH */
'87360' = '1' /* OPEN WOUND OF MOUTH NOS */
'87361' = '1' /* OPEN WOUND BUCCAL MUCOSA */
'87362' = '1' /* OPEN WOUND OF GUM */
'87363' = '1' /* BROKEN TOOTH */
```

PSFMTS -- CONTINUED

```
'87364' = '1' /* OPN WND TONGUE/MOUTH FLR */
'87365' = '1' /* OPEN WOUND OF PALATE */
'87369' = '1' /* OPEN WOUND MOUTH NEC */
'8737'  = '1' /* OPEN WND INT MOUTH-COMPL */
'87370' = '1' /* OPEN WND MOUTH NOS-COMPL */
'87371' = '1' /* OPN WND BUC MUCOSA-COMPL */
'87372' = '1' /* OPEN WOUND GUM-COMPL */
'87373' = '1' /* BROKEN TOOTH-COMPLICATED */
'87374' = '1' /* OPEN WOUND TONGUE-COMPL */
'87375' = '1' /* OPEN WOUND PALATE-COMPL */
'87379' = '1' /* OPEN WND MOUTH NOS-COMPL */
'8738'  = '1' /* OPEN WOUND OF HEAD NEC */
'8739'  = '1' /* OPEN WND HEAD NEC-COMPL */
'874'   = '1' /* OPEN WOUND OF NECK */
'8740'  = '1' /* OPN WOUND LARYNX/TRACHEA */
'87400' = '1' /* OPN WND LARYNX W TRACHEA */
'87401' = '1' /* OPEN WOUND OF LARYNX */
'87402' = '1' /* OPEN WOUND OF TRACHEA */
'8741'  = '1' /* OPN WND LARYN/TRACH-COMP */
'87410' = '1' /* OPN WND LARY W TRAC-COMP */
'87411' = '1' /* OPEN WOUND LARYNX-COMPL */
'87412' = '1' /* OPEN WOUND TRACHEA-COMPL */
'8742'  = '1' /* OPEN WOUND THYROID GLAND */
'8743'  = '1' /* OPEN WOUND THYROID-COMPL */
'8744'  = '1' /* OPEN WOUND OF PHARYNX */
'8745'  = '1' /* OPEN WOUND PHARYNX-COMPL */
'8748'  = '1' /* OPEN WOUND OF NECK NEC */
'8749'  = '1' /* OPN WOUND NECK NEC-COMPL */
'875'   = '1' /* OPEN WOUND OF CHEST */
```

```
'8750'  = '1' /* OPEN WOUND OF CHEST */
'8751'  = '1' /* OPEN WOUND CHEST-COMPL */
'876'   = '1' /* OPEN WOUND OF BACK */
'8760'  = '1' /* OPEN WOUND OF BACK */
'8761'  = '1' /* OPEN WOUND BACK-COMPL */
'877'   = '1' /* OPEN WOUND OF BUTTOCK */
'8770'  = '1' /* OPEN WOUND OF BUTTOCK */
'8771'  = '1' /* OPEN WOUND BUTTOCK-COMPL */
'878'   = '1' /* OPEN WOUND GENITAL ORGAN */
'8780'  = '1' /* OPEN WOUND OF PENIS */
'8781'  = '1' /* OPEN WOUND PENIS-COMPL */
'8782'  = '1' /* OPN WOUND SCROTUM/TESTES */
'8783'  = '1' /* OPN WND SCROT/TEST-COMPL */
'8784'  = '1' /* OPEN WOUND OF VULVA */
'8785'  = '1' /* OPEN WOUND VULVA-COMPL */
'8786'  = '1' /* OPEN WOUND OF VAGINA */
'8787'  = '1' /* OPEN WOUND VAGINA-COMPL */
'8788'  = '1' /* OPEN WOUND GENITAL NEC */
'8789'  = '1' /* OPN WND GENITAL NEC-COMP */
'879'   = '1' /* OPEN WOUND SITE NEC */
'8790'  = '1' /* OPEN WOUND OF BREAST */
'8791'  = '1' /* OPEN WOUND BREAST-COMPL */
'8792'  = '1' /* OPN WND ANTERIOR ABDOMEN */
'8793'  = '1' /* OPN WND ANT ABDOMEN-COMP */
'8794'  = '1' /* OPN WND LATERAL ABDOMEN */
'8795'  = '1' /* OPN WND LAT ABDOMEN-COMP */
'8796'  = '1' /* OPEN WOUND OF TRUNK NEC */
'8797'  = '1' /* OPEN WND TRUNK NEC-COMPL */
'8798'  = '1' /* OPEN WOUND SITE NOS */
'8799'  = '1' /* OPN WOUND SITE NOS-COMPL */
```

PSFMTS -- CONTINUED

```
'880'   = '1' /* OPN WND SHOULDR/UPPR ARM */
'8800'  = '1' /* OPN WND SHOULDR/UPPR ARM */
'88000' = '1' /* OPEN WOUND OF SHOULDER */
'88001' = '1' /* OPEN WOUND OF SCAPULA */
'88002' = '1' /* OPEN WOUND OF AXILLA */
'88003' = '1' /* OPEN WOUND OF UPPER ARM */
'88009' = '1' /* MULT OPEN WOUND SHOULDER */
'8801'  = '1' /* OPN WND SHLD/UP ARM-COMP */
'88010' = '1' /* OPEN WND SHOULDER-COMPL */
'88011' = '1' /* OPEN WOUND SCAPULA-COMPL */
'88012' = '1' /* OPEN WOUND AXILLA-COMPL */
'88013' = '1' /* OPEN WND UPPER ARM-COMPL */
'88019' = '1' /* MULT OPN WND SHOULD-COMP */
'8802'  = '1' /* OPN WND SHLD/UP ARM-TEND */
'88020' = '1' /* OPN WND SHOULDR W TENDON */
'88021' = '1' /* OPN WND SCAPULA W TENDON */
'88022' = '1' /* OPEN WND AXILLA W TENDON */
'88023' = '1' /* OPEN WND UP ARM W TENDON */
'88029' = '1' /* MLT OPN WND SHLDR W TEND */
'881'   = '1' /* OPEN WOUND OF LOWER ARM */
'8810'  = '1' /* OPEN WOUND OF LOWER ARM */
'88100' = '1' /* OPEN WOUND OF FOREARM */
'88101' = '1' /* OPEN WOUND OF ELBOW */
'88102' = '1' /* OPEN WOUND OF WRIST */
'8811'  = '1' /* OPEN WOUND LOW ARM-COMPL */
'88110' = '1' /* OPEN WOUND FOREARM-COMPL */
```

```
'88111' = '1' /* OPEN WOUND ELBOW-COMPLIC */
'88112' = '1' /* OPEN WOUND WRIST-COMPLIC */
'8812'  = '1' /* OPN WND LOW ARM W TENDON */
'88120' = '1' /* OPEN WND FOREARM W TENDN */
'88121' = '1' /* OPN WOUND ELBOW W TENDON */
'88122' = '1' /* OPN WOUND WRIST W TENDON */
'882'   = '1' /* OPEN WOUND OF HAND */
'8820'  = '1' /* OPEN WOUND OF HAND */
'8821'  = '1' /* OPN WOUND HAND-COMPLICAT */
'8822'  = '1' /* OPEN WOUND HAND W TENDON */
'884'   = '1' /* OPEN WOUND ARM MULT/NOS */
'8840'  = '1' /* OPEN WOUND ARM MULT/NOS */
'8841'  = '1' /* OPEN WOUND ARM NOS-COMPL */
'8842'  = '1' /* OPN WND ARM NOS W TENDON */
'887'   = '1' /* TRAUMATIC AMPUT ARM/HAND */
'8870'  = '1' /* AMPUT BELOW ELB, UNILAT */
'8871'  = '1' /* AMP BELOW ELB, UNIL-COMP */
'8872'  = '1' /* AMPUT ABV ELBOW, UNILAT */
'8873'  = '1' /* AMPUT ABV ELB, UNIL-COMP */
'8874'  = '1' /* AMPUTAT ARM, UNILAT NOS */
'8875'  = '1' /* AMPUT ARM, UNIL NOS-COMP */
'8876'  = '1' /* AMPUTATION ARM, BILAT */
'8877'  = '1' /* AMPUTAT ARM, BILAT-COMPL */
'890'   = '1' /* OPEN WOUND OF HIP/THIGH */
'8900'  = '1' /* OPEN WOUND OF HIP/THIGH */
'8901'  = '1' /* OPEN WND HIP/THIGH-COMPL */
'8902'  = '1' /* OPN WND HIP/THIGH W TEND */
'891'   = '1' /* OPEN WND KNEE/LEG/ANKLE */
'8910'  = '1' /* OPEN WND KNEE/LEG/ANKLE */
'8911'  = '1' /* OPEN WND KNEE/LEG-COMPL */
'8912'  = '1' /* OPN WND KNEE/LEG W TENDN */
'892'   = '1' /* OPEN WOUND OF FOOT */
'8920'  = '1' /* OPEN WOUND OF FOOT */
```

PSFMTS -- CONTINUED

```
'8921'  = '1' /* OPEN WOUND FOOT-COMPL */
'8922'  = '1' /* OPEN WOUND FOOT W TENDON */
'894'   = '1' /* OPEN WOUND OF LEG NEC */
'8940'  = '1' /* OPEN WOUND OF LEG NEC */
'8941'  = '1' /* OPEN WOUND LEG NEC-COMPL */
'8942'  = '1' /* OPN WND LEG NEC W TENDON */
'896'   = '1' /* TRAUMATIC AMPUTAT FOOT */
'8960'  = '1' /* AMPUTATION FOOT, UNILAT */
'8961'  = '1' /* AMPUT FOOT, UNILAT-COMPL */
'8962'  = '1' /* AMPUTATION FOOT, BILAT */
'8963'  = '1' /* AMPUTAT FOOT, BILAT-COMP */
'897'   = '1' /* TRAUMATIC AMPUTATION LEG */
'8970'  = '1' /* AMPUT BELOW KNEE, UNILAT */
'8971'  = '1' /* AMPUTAT BK, UNILAT-COMPL */
'8972'  = '1' /* AMPUT ABOVE KNEE, UNILAT */
'8973'  = '1' /* AMPUT ABV KN, UNIL-COMPL */
'8974'  = '1' /* AMPUTAT LEG, UNILAT NOS */
'8975'  = '1' /* AMPUT LEG, UNIL NOS-COMP */
'8976'  = '1' /* AMPUTATION LEG, BILAT */
'8977'  = '1' /* AMPUTAT LEG, BILAT-COMPL */
'900'   = '1' /* HEAD/NECK VESSEL INJURY */
'9000'  = '1' /* CAROTID ARTERY INJURY */
'90000' = '1' /* INJUR CAROTID ARTERY NOS */
```

```
'90001' = '1' /* INJ COMMON CAROTID ARTER */
'90002' = '1' /* INJ EXTERNAL CAROTID ART */
'90003' = '1' /* INJ INTERNAL CAROTID ART */
'9001'  = '1' /* INJ INTERNL JUGULAR VEIN */
'9008'  = '1' /* INJ HEAD/NECK VESSEL NEC */
'90081' = '1' /* INJ EXTERN JUGULAR VEIN */
'90082' = '1' /* INJ MLT HEAD/NECK VESSEL */
'90089' = '1' /* INJ HEAD/NECK VESSEL NEC */
'9009'  = '1' /* INJ HEAD/NECK VESSEL NOS */
'901'   = '1' /* THORAC BLOOD VESSEL INJ */
'9010'  = '1' /* INJURY THORACIC AORTA */
'9011'  = '1' /* INJ INNOMIN/SUBCLAV ART */
'9012'  = '1' /* INJ SUPERIOR VENA CAVA */
'9013'  = '1' /* INJ INNOMIN/SUBCLAV VEIN */
'9014'  = '1' /* INJURY PULMONARY VESSEL */
'90140' = '1' /* INJ PULMONARY VESSEL NOS */
'90141' = '1' /* INJURY PULMONARY ARTERY */
'90142' = '1' /* INJURY PULMONARY VEIN */
'9018'  = '1' /* INJ THORACIC VESSEL NEC */
'90181' = '1' /* INJ INTERCOSTAL ART/VEIN */
'90182' = '1' /* INJ INT MAMMARY ART/VEIN */
'90183' = '1' /* INJ MULT THORACIC VESSEL */
'90189' = '1' /* INJ THORACIC VESSEL NEC */
'9019'  = '1' /* INJ THORACIC VESSEL NOS */
'902'   = '1' /* ABDOM/PELVIS VESSEL INJ */
'9020'  = '1' /* INJURY ABDOMINAL AORTA */
'9021'  = '1' /* INJ INFERIOR VENA CAVA */
'90210' = '1' /* INJ INFER VENA CAVA NOS */
'90211' = '1' /* INJURY HEPATIC VEINS */
'90219' = '1' /* INJ INFER VENA CAVA NEC */
'9022'  = '1' /* INJ CELIAC/MESENTER ART */
'90220' = '1' /* INJ CELIAC/MESEN ART NOS */
'90221' = '1' /* INJURY GASTRIC ARTERY */
'90222' = '1' /* INJURY HEPATIC ARTERY */
'90223' = '1' /* INJURY SPLENIC ARTERY */
'90224' = '1' /* INJURY CELIAC AXIS NEC */
```

PSFMTS -- CONTINUED

```
'90225' = '1' /* INJ SUPER MESENTERIC ART */
'90226' = '1' /* INJ BRNCH SUP MESENT ART */
'90227' = '1' /* INJ INFER MESENTERIC ART */
'90229' = '1' /* INJ MESENTERIC VESS NEC */
'9023'  = '1' /* PORTAL/SPLENIC VEIN INJ */
'90231' = '1' /* INJ SUPERIOR MESENT VEIN */
'90232' = '1' /* INJ INFERIOR MESENT VEIN */
'90233' = '1' /* INJURY PORTAL VEIN */
'90234' = '1' /* INJURY SPLENIC VEIN */
'90239' = '1' /* INJ PORT/SPLEN VESS NEC */
'9024'  = '1' /* RENAL VESSEL INJURY */
'90240' = '1' /* INJURY RENAL VESSEL NOS */
'90241' = '1' /* INJURY RENAL ARTERY */
'90242' = '1' /* INJURY RENAL VEIN */
'90249' = '1' /* INJURY RENAL VESSEL NEC */
'9025'  = '1' /* ILIAC VESSEL INJURY */
'90250' = '1' /* INJURY ILIAC VESSEL NOS */
'90251' = '1' /* INJ HYPOGASTRIC ARTERY */
'90252' = '1' /* INJURY HYPOGASTRIC VEIN */
'90253' = '1' /* INJURY ILIAC ARTERY */
```

```
'90254' = '1' /* INJURY ILIAC VEIN */
'90255' = '1' /* INJURY UTERINE ARTERY */
'90256' = '1' /* INJURY UTERINE VEIN */
'90259' = '1' /* INJURY ILIAC VESSEL NEC */
'9028'  = '1' /* INJ ABDOMINAL VESSEL NEC */
'90281' = '1' /* INJURY OVARIAN ARTERY */
'90282' = '1' /* INJURY OVARIAN VEIN */
'90287' = '1' /* INJ MULT ABD/PELV VESSEL */
'90289' = '1' /* INJ ABDOMINAL VESSEL NEC */
'9029'  = '1' /* INJ ABDOMINAL VESSEL NOS */
'903'   = '1' /* ARM BLOOD VESSEL INJURY */
'9030'  = '1' /* INJURY AXILLARY VESSELS */
'90300' = '1' /* INJ AXILLARY VESSEL NOS */
'90301' = '1' /* INJURY AXILLARY ARTERY */
'90302' = '1' /* INJURY AXILLARY VEIN */
'9031'  = '1' /* INJURY BRACHIAL VESSELS */
'9032'  = '1' /* INJURY RADIAL VESSELS */
'9033'  = '1' /* INJURY ULNAR VESSELS */
'9034'  = '1' /* INJURY PALMAR ARTERY */
'9035'  = '1' /* INJURY FINGER VESSELS */
'9038'  = '1' /* INJURY ARM VESSELS NEC */
'9039'  = '1' /* INJURY ARM VESSEL NOS */
'904'   = '1' /* BLOOD VESSEL INJURY NEC */
'9040'  = '1' /* INJ COMMON FEMORAL ARTER */
'9041'  = '1' /* INJ SUPERFIC FEMORAL ART */
'9042'  = '1' /* INJURY FEMORAL VEIN */
'9043'  = '1' /* INJURY SAPHENOUS VEIN */
'9044'  = '1' /* INJURY POPLITEAL VESSEL */
'90440' = '1' /* INJ POPLITEAL VESSEL NOS */
'90441' = '1' /* INJURY POPLITEAL ARTERY */
'90442' = '1' /* INJURY POPLITEAL VEIN */
'9045'  = '1' /* INJURY TIBIAL VESSELS */
'90450' = '1' /* INJURY TIBIAL VESSEL NOS */
'90451' = '1' /* INJ ANTER TIBIAL ARTERY */
'90452' = '1' /* INJ ANTERIOR TIBIAL VEIN */
'90453' = '1' /* INJ POST TIBIAL ARTERY */
'90454' = '1' /* INJ POST TIBIAL VEIN */
'9046'  = '1' /* INJ DEEP PLANTAR VESSEL */
'9047'  = '1' /* INJURY LEG VESSELS NEC */
```

PSFMTS -- CONTINUED

```
'9048'  = '1' /* INJURY LEG VESSEL NOS */
'9049'  = '1' /* BLOOD VESSEL INJURY NOS */
'925'   = '1' /* CRUSH INJ FACE/SCALP/NCK */
'9251'  = '1' /* CRUSH INJ FACE SCALP */
'9252'  = '1' /* CRUSH INJ NECK */
'926'   = '1' /* CRUSHING INJURY OF TRUNK */
'9260'  = '1' /* CRUSH INJ EXT GENITALIA */
'9261'  = '1' /* OTH TRUNK CRUSHING INJ */
'92611' = '1' /* CRUSHING INJURY BACK */
'92612' = '1' /* CRUSHING INJURY BUTTOCK */
'92619' = '1' /* CRUSHING INJ TRUNK NEC */
'9268'  = '1' /* MULT CRUSHING INJ TRUNK */
'9269'  = '1' /* CRUSHING INJ TRUNK NOS */
'927'   = '1' /* CRUSHING INJ UPPER LIMB */
'9270'  = '1' /* CRUSH INJ SHOULDER & ARM */
'92700' = '1' /* CRUSH INJ SHOULDER REG */
'92701' = '1' /* CRUSH INJ SCAPUL REGION */
```

```
'92702' = '1' /* CRUSH INJ AXILLARY REG */
'92703' = '1' /* CRUSHING INJ UPPER ARM */
'92709' = '1' /* CRUSH INJ SHOULDER & ARM */
'9271'  = '1' /* CRUSH INJ ELBOW/FOREARM */
'92710' = '1' /* CRUSHING INJURY FOREARM */
'92711' = '1' /* CRUSHING INJURY ELBOW */
'9272'  = '1' /* CRUSHING INJ WRIST/HAND */
'92720' = '1' /* CRUSHING INJURY OF HAND */
'92721' = '1' /* CRUSHING INJURY OF WRIST */
'9273'  = '1' /* CRUSHING INJURY FINGER */
'9278'  = '1' /* MULT CRUSHING INJURY ARM */
'9279'  = '1' /* CRUSHING INJURY ARM NOS */
'928'   = '1' /* CRUSHING INJURY OF LEG */
'9280'  = '1' /* CRUSHING INJ HIP/THIGH */
'92800' = '1' /* CRUSHING INJURY THIGH */
'92801' = '1' /* CRUSHING INJURY HIP */
'9281'  = '1' /* CRUSH INJ KNEE/LOWER LEG */
'92810' = '1' /* CRUSHING INJ LOWER LEG */
'92811' = '1' /* CRUSHING INJURY KNEE */
'9282'  = '1' /* CRUSHING INJ ANKLE/FOOT */
'92820' = '1' /* CRUSHING INJURY FOOT */
'92821' = '1' /* CRUSHING INJURY ANKLE */
'9283'  = '1' /* CRUSHING INJURY TOE */
'9288'  = '1' /* MULT CRUSHING INJURY LEG */
'9289'  = '1' /* CRUSHING INJURY LEG NOS */
'929'   = '1' /* CRUSH INJ MULT/UNSP SITE */
'9290'  = '1' /* CRUSH INJ MULT SITE NEC */
'9299'  = '1' /* CRUSHING INJURY NOS */
'940'   = '1' /* BURN OF EYE & ADNEXA */
'9400'  = '1' /* CHEMICAL BURN PERIOCULAR */
'9401'  = '1' /* BURN PERIOCULAR AREA NEC */
'9402'  = '1' /* ALKAL BURN CORNEA/CONJUN */
'9403'  = '1' /* ACID BURN CORNEA/CONJUNC */
'9404'  = '1' /* BURN CORNEA/CONJUNCT NEC */
'9405'  = '1' /* BURN W EYEBALL DESTRUCT */
'9409'  = '1' /* BURN EYE & ADNEXA NOS */
'941'   = '1' /* BURN OF HEAD/FACE/NECK */
'9410'  = '1' /* BURN NOS HEAD/FACE/NECK */
'94100' = '1' /* BURN NOS HEAD-UNSPEC */
'94101' = '1' /* BURN NOS EAR */
'94102' = '1' /* BURN NOS EYE */
'94103' = '1' /* BURN NOS LIP */
```

PSFMTS -- CONTINUED

```
'94104' = '1' /* BURN NOS CHIN */
'94105' = '1' /* BURN NOS NOSE */
'94106' = '1' /* BURN NOS SCALP */
'94107' = '1' /* BURN NOS FACE NEC */
'94108' = '1' /* BURN NOS NECK */
'94109' = '1' /* BURN NOS HEAD-MULT */
'9411'  = '1' /* 1ST DEGREE BURN HEAD */
'94110' = '1' /* 1ST DEG BURN HEAD NOS */
'94111' = '1' /* 1ST DEG BURN EAR */
'94112' = '1' /* 1ST DEG BURN EYE */
'94113' = '1' /* 1ST DEG BURN LIP */
'94114' = '1' /* 1ST DEG BURN CHIN */
'94115' = '1' /* 1ST DEG BURN NOSE */
'94116' = '1' /* 1ST DEG BURN SCALP */
```

```
'94117' = '1' /* 1ST DEG BURN FACE NEC */
'94118' = '1' /* 1ST DEG BURN NECK */
'94119' = '1' /* 1ST DEG BURN HEAD-MULT */
'9412'  = '1' /* 2ND DEGREE BURN HEAD */
'94120' = '1' /* 2ND DEG BURN HEAD NOS */
'94121' = '1' /* 2ND DEG BURN EAR */
'94122' = '1' /* 2ND DEG BURN EYE */
'94123' = '1' /* 2ND DEG BURN LIP */
'94124' = '1' /* 2ND DEG BURN CHIN */
'94125' = '1' /* 2ND DEG BURN NOSE */
'94126' = '1' /* 2ND DEG BURN SCALP */
'94127' = '1' /* 2ND DEG BURN FACE NEC */
'94128' = '1' /* 2ND DEG BURN NECK */
'94129' = '1' /* 2ND DEG BURN HEAD-MULT */
'9413'  = '1' /* 3RD DEGREE BURN HEAD */
'94130' = '1' /* 3RD DEG BURN HEAD NOS */
'94131' = '1' /* 3RD DEG BURN EAR */
'94132' = '1' /* 3RD DEG BURN EYE */
'94133' = '1' /* 3RD DEG BURN LIP */
'94134' = '1' /* 3RD DEG BURN CHIN */
'94135' = '1' /* 3RD DEG BURN NOSE */
'94136' = '1' /* 3RD DEG BURN SCALP */
'94137' = '1' /* 3RD DEG BURN FACE NEC */
'94138' = '1' /* 3RD DEG BURN NECK */
'94139' = '1' /* 3RD DEG BURN HEAD-MULT */
'9414'  = '1' /* DEEP 3RD DEG BURN HEAD */
'94140' = '1' /* DEEP 3 DEG BURN HEAD NOS */
'94141' = '1' /* DEEP 3RD DEG BURN EAR */
'94142' = '1' /* DEEP 3RD DEG BURN EYE */
'94143' = '1' /* DEEP 3RD DEG BURN LIP */
'94144' = '1' /* DEEP 3RD DEG BURN CHIN */
'94145' = '1' /* DEEP 3RD DEG BURN NOSE */
'94146' = '1' /* DEEP 3RD DEG BURN SCALP */
'94147' = '1' /* DEEP 3RD BURN FACE NEC */
'94148' = '1' /* DEEP 3RD DEG BURN NECK */
'94149' = '1' /* DEEP 3 DEG BRN HEAD-MULT */
'9415'  = '1' /* 3RD DEG BURN W LOSS-HEAD */
'94150' = '1' /* 3RD BURN W LOSS-HEAD NOS */
'94151' = '1' /* 3RD DEG BURN W LOSS-EAR */
'94152' = '1' /* 3RD DEG BURN W LOSS-EYE */
'94153' = '1' /* 3RD DEG BURN W LOSS-LIP */
'94154' = '1' /* 3RD DEG BURN W LOSS-CHIN */
'94155' = '1' /* 3RD DEG BURN W LOSS-NOSE */
'94156' = '1' /* 3RD DEG BRN W LOSS-SCALP */
'94157' = '1' /* 3RD BURN W LOSS-FACE NEC */
```

PSFMTS -- CONTINUED

```
'94158' = '1' /* 3RD DEG BURN W LOSS-NECK */
'94159' = '1' /* 3RD BRN W LOSS-HEAD MULT */
'942'   = '1' /* BURN OF TRUNK */
'9420'  = '1' /* BURN NOS TRUNK */
'94200' = '1' /* BURN NOS TRUNK-UNSPEC */
'94201' = '1' /* BURN NOS BREAST */
'94202' = '1' /* BURN NOS CHEST WALL */
'94203' = '1' /* BURN NOS ABDOMINAL WALL */
'94204' = '1' /* BURN NOS BACK */
'94205' = '1' /* BURN NOS GENITALIA */
'94209' = '1' /* BURN NOS TRUNK NEC */
```

```
'9421'  = '1' /* 1ST DEGREE BURN TRUNK */
'94210' = '1' /* 1ST DEG BURN TRUNK NOS */
'94211' = '1' /* 1ST DEG BURN BREAST */
'94212' = '1' /* 1ST DEG BURN CHEST WALL */
'94213' = '1' /* 1ST DEG BURN ABDOMN WALL */
'94214' = '1' /* 1ST DEG BURN BACK */
'94215' = '1' /* 1ST DEG BURN GENITALIA */
'94219' = '1' /* 1ST DEG BURN TRUNK NEC */
'9422'  = '1' /* 2ND DEGREE BURN TRUNK */
'94220' = '1' /* 2ND DEG BURN TRUNK NOS */
'94221' = '1' /* 2ND DEG BURN BREAST */
'94222' = '1' /* 2ND DEG BURN CHEST WALL */
'94223' = '1' /* 2ND DEG BURN ABDOMN WALL */
'94224' = '1' /* 2ND DEG BURN BACK */
'94225' = '1' /* 2ND DEG BURN GENITALIA */
'94229' = '1' /* 2ND DEG BURN TRUNK NEC */
'9423'  = '1' /* 3RD DEGREE BURN TRUNK */
'94230' = '1' /* 3RD DEG BURN TRUNK NOS */
'94231' = '1' /* 3RD DEG BURN BREAST */
'94232' = '1' /* 3RD DEG BURN CHEST WALL */
'94233' = '1' /* 3RD DEG BURN ABDOMN WALL */
'94234' = '1' /* 3RD DEG BURN BACK */
'94235' = '1' /* 3RD DEG BURN GENITALIA */
'94239' = '1' /* 3RD DEG BURN TRUNK NEC */
'9424'  = '1' /* DEEP 3RD DEG BURN TRUNK */
'94240' = '1' /* DEEP 3RD BURN TRUNK NOS */
'94241' = '1' /* DEEP 3RD DEG BURN BREAST */
'94242' = '1' /* DEEP 3RD BURN CHEST WALL */
'94243' = '1' /* DEEP 3RD BURN ABDOM WALL */
'94244' = '1' /* DEEP 3RD DEG BURN BACK */
'94245' = '1' /* DEEP 3RD BURN GENITALIA */
'94249' = '1' /* DEEP 3RD BURN TRUNK NEC */
'9425'  = '1' /* 3 DEG BURN W LOSS-TRUNK */
'94250' = '1' /* 3RD BRN W LOSS-TRUNK NOS */
'94251' = '1' /* 3RD BURN W LOSS-BREAST */
'94252' = '1' /* 3RD BRN W LOSS-CHEST WLL */
'94253' = '1' /* 3RD BRN W LOSS-ABDOM WLL */
'94254' = '1' /* 3RD DEG BURN W LOSS-BACK */
'94255' = '1' /* 3RD BRN W LOSS-GENITALIA */
'94259' = '1' /* 3RD BRN W LOSS-TRUNK NEC */
'943'   = '1' /* BURN OF ARM */
'9430'  = '1' /* BURN NOS ARM */
'94300' = '1' /* BURN NOS ARM-UNSPEC */
'94301' = '1' /* BURN NOS FOREARM */
'94302' = '1' /* BURN NOS ELBOW */
'94303' = '1' /* BURN NOS UPPER ARM */
'94304' = '1' /* BURN NOS AXILLA */
'94305' = '1' /* BURN NOS SHOULDER */
```

PSFMTS -- CONTINUED

```
'94306' = '1' /* BURN NOS SCAPULA */
'94309' = '1' /* BURN NOS ARM-MULTIPLE */
'9431'  = '1' /* 1ST DEGREE BURN ARM */
'94310' = '1' /* 1ST DEG BURN ARM NOS */
'94311' = '1' /* 1ST DEG BURN FOREARM */
'94312' = '1' /* 1ST DEG BURN ELBOW */
'94313' = '1' /* 1ST DEG BURN UPPER ARM */
'94314' = '1' /* 1ST DEG BURN AXILLA */
```

```
'94315' = '1' /* 1ST DEG BURN SHOULDER */
'94316' = '1' /* 1ST DEG BURN SCAPULA */
'94319' = '1' /* 1ST DEG BURN ARM-MULT */
'9432'  = '1' /* 2ND DEGREE BURN ARM */
'94320' = '1' /* 2ND DEG BURN ARM NOS */
'94321' = '1' /* 2ND DEG BURN FOREARM */
'94322' = '1' /* 2ND DEG BURN ELBOW */
'94323' = '1' /* 2ND DEG BURN UPPER ARM */
'94324' = '1' /* 2ND DEG BURN AXILLA */
'94325' = '1' /* 2ND DEG BURN SHOULDER */
'94326' = '1' /* 2ND DEG BURN SCAPULA */
'94329' = '1' /* 2ND DEG BURN ARM-MULT */
'9433'  = '1' /* 3RD DEGREE BURN ARM */
'94330' = '1' /* 3RD DEG BURN ARM NOS */
'94331' = '1' /* 3RD DEG BURN FOREARM */
'94332' = '1' /* 3RD DEG BURN ELBOW */
'94333' = '1' /* 3RD DEG BURN UPPER ARM */
'94334' = '1' /* 3RD DEG BURN AXILLA */
'94335' = '1' /* 3RD DEG BURN SHOULDER */
'94336' = '1' /* 3RD DEG BURN SCAPULA */
'94339' = '1' /* 3RD DEG BURN ARM-MULT */
'9434'  = '1' /* DEEP 3RD DEG BURN ARM */
'94340' = '1' /* DEEP 3 DEG BURN ARM NOS */
'94341' = '1' /* DEEP 3 DEG BURN FOREARM */
'94342' = '1' /* DEEP 3 DEG BURN ELBOW */
'94343' = '1' /* DEEP 3 DEG BRN UPPER ARM */
'94344' = '1' /* DEEP 3 DEG BURN AXILLA */
'94345' = '1' /* DEEP 3 DEG BURN SHOULDER */
'94346' = '1' /* DEEP 3 DEG BURN SCAPULA */
'94349' = '1' /* DEEP 3 DEG BURN ARM-MULT */
'9435'  = '1' /* 3RD DEG BURN W LOSS-ARM */
'94350' = '1' /* 3RD BURN W LOSS-ARM NOS */
'94351' = '1' /* 3RD BURN W LOSS-FOREARM */
'94352' = '1' /* 3RD BURN W LOSS-ELBOW */
'94353' = '1' /* 3RD BRN W LOSS-UPPER ARM */
'94354' = '1' /* 3RD BURN W LOSS-AXILLA */
'94355' = '1' /* 3RD BURN W LOSS-SHOULDER */
'94356' = '1' /* 3RD BURN W LOSS-SCAPULA */
'94359' = '1' /* 3RD BURN W LOSS ARM-MULT */
'944'   = '1' /* BURN OF HAND & WRIST */
'9440'  = '1' /* BURN NOS HAND & WRIST */
'94400' = '1' /* BURN NOS HAND-UNSPEC */
'94401' = '1' /* BURN NOS FINGER */
'94402' = '1' /* BURN NOS THUMB */
'94403' = '1' /* BURN NOS MULT FINGERS */
'94404' = '1' /* BURN NOS FINGER W THUMB */
'94405' = '1' /* BURN NOS PALM */
'94406' = '1' /* BURN NOS BACK OF HAND */
'94407' = '1' /* BURN NOS WRIST */
'94408' = '1' /* BURN NOS HAND-MULTIPLE */
'9441'  = '1' /* 1ST DEGREE BURN HAND */
```

PSFMTS -- CONTINUED

```
'94410' = '1' /* 1ST DEG BURN HAND NOS */
'94411' = '1' /* 1ST DEG BURN FINGER */
'94412' = '1' /* 1ST DEG BURN THUMB */
'94413' = '1' /* 1ST DEG BURN MULT FINGER */
'94414' = '1' /* 1 DEG BURN FINGR W THUMB */
```

```
'94415' = '1' /* 1ST DEG BURN PALM */
'94416' = '1' /* 1 DEG BURN BACK OF HAND */
'94417' = '1' /* 1ST DEG BURN WRIST */
'94418' = '1' /* 1ST DEG BURN HAND-MULT */
'9442'  = '1' /* 2ND DEGREE BURN HAND */
'94420' = '1' /* 2ND DEG BURN HAND NOS */
'94421' = '1' /* 2ND DEG BURN FINGER */
'94422' = '1' /* 2ND DEG BURN THUMB */
'94423' = '1' /* 2ND DEG BURN MULT FINGER */
'94424' = '1' /* 2 DEG BURN FINGR W THUMB */
'94425' = '1' /* 2ND DEG BURN PALM */
'94426' = '1' /* 2 DEG BURN BACK OF HAND */
'94427' = '1' /* 2ND DEG BURN WRIST */
'94428' = '1' /* 2ND DEG BURN HAND-MULT */
'9443'  = '1' /* 3RD DEGREE BURN HAND */
'94430' = '1' /* 3RD DEG BURN HAND NOS */
'94431' = '1' /* 3RD DEG BURN FINGER */
'94432' = '1' /* 3RD DEG BURN THUMB */
'94433' = '1' /* 3RD DEG BURN MULT FINGER */
'94434' = '1' /* 3 DEG BURN FINGR W THUMB */
'94435' = '1' /* 3RD DEG BURN PALM */
'94436' = '1' /* 3 DEG BURN BACK OF HAND */
'94437' = '1' /* 3RD DEG BURN WRIST */
'94438' = '1' /* 3RD DEG BURN HAND-MULT */
'9444'  = '1' /* DEEP 3RD DEG BURN HAND */
'94440' = '1' /* DEEP 3 DEG BRN HAND NOS */
'94441' = '1' /* DEEP 3 DEG BURN FINGER */
'94442' = '1' /* DEEP 3 DEG BURN THUMB */
'94443' = '1' /* DEEP 3RD BRN MULT FINGER */
'94444' = '1' /* DEEP 3RD BRN FNGR W THMB */
'94445' = '1' /* DEEP 3 DEG BURN PALM */
'94446' = '1' /* DEEP 3RD BRN BACK OF HND */
'94447' = '1' /* DEEP 3 DEG BURN WRIST */
'94448' = '1' /* DEEP 3 DEG BRN HAND-MULT */
'9445'  = '1' /* 3RD DEG BURN W LOSS-HAND */
'94450' = '1' /* 3RD BRN W LOSS-HAND NOS */
'94451' = '1' /* 3RD BURN W LOSS-FINGER */
'94452' = '1' /* 3RD BURN W LOSS-THUMB */
'94453' = '1' /* 3RD BRN W LOSS-MULT FNGR */
'94454' = '1' /* 3RD BRN W LOSS-FNGR/THMB */
'94455' = '1' /* 3RD BURN W LOSS-PALM */
'94456' = '1' /* 3RD BRN W LOSS-BK OF HND */
'94457' = '1' /* 3RD BURN W LOSS-WRIST */
'94458' = '1' /* 3RD BRN W LOSS HAND-MULT */
'945'   = '1' /* BURN OF LEG */
'9450'  = '1' /* BURN NOS LEG */
'94500' = '1' /* BURN NOS LEG-UNSPEC */
'94501' = '1' /* BURN NOS TOE */
'94502' = '1' /* BURN NOS FOOT */
'94503' = '1' /* BURN NOS ANKLE */
'94504' = '1' /* BURN NOS LOWER LEG */
'94505' = '1' /* BURN NOS KNEE */
'94506' = '1' /* BURN NOS THIGH */
'94509' = '1' /* BURN NOS LEG-MULTIPLE */
```

PSFMTS -- CONTINUED

```
'9451'  = '1' /* 1ST DEGREE BURN LEG */
'94510' = '1' /* 1ST DEG BURN LEG NOS */
```

```
'94511' = '1' /* 1ST DEG BURN TOE */
'94512' = '1' /* 1ST DEG BURN FOOT */
'94513' = '1' /* 1ST DEG BURN ANKLE */
'94514' = '1' /* 1ST DEG BURN LOWER LEG */
'94515' = '1' /* 1ST DEG BURN KNEE */
'94516' = '1' /* 1ST DEG BURN THIGH */
'94519' = '1' /* 1ST DEG BURN LEG-MULT */
'9452'  = '1' /* 2ND DEGREE BURN LEG */
'94520' = '1' /* 2ND DEG BURN LEG NOS */
'94521' = '1' /* 2ND DEG BURN TOE */
'94522' = '1' /* 2ND DEG BURN FOOT */
'94523' = '1' /* 2ND DEG BURN ANKLE */
'94524' = '1' /* 2ND DEG BURN LOWER LEG */
'94525' = '1' /* 2ND DEG BURN KNEE */
'94526' = '1' /* 2ND DEG BURN THIGH */
'94529' = '1' /* 2ND DEG BURN LEG-MULT */
'9453'  = '1' /* 3RD DEGREE BURN LEG */
'94530' = '1' /* 3RD DEG BURN LEG NOS */
'94531' = '1' /* 3RD DEG BURN TOE */
'94532' = '1' /* 3RD DEG BURN FOOT */
'94533' = '1' /* 3RD DEG BURN ANKLE */
'94534' = '1' /* 3RD DEG BURN LOW LEG */
'94535' = '1' /* 3RD DEG BURN KNEE */
'94536' = '1' /* 3RD DEG BURN THIGH */
'94539' = '1' /* 3RD DEG BURN LEG-MULT */
'9454'  = '1' /* DEEP 3RD DEGREE BURN LEG */
'94540' = '1' /* DEEP 3RD DEG BRN LEG NOS */
'94541' = '1' /* DEEP 3RD DEG BURN TOE */
'94542' = '1' /* DEEP 3RD DEG BURN FOOT */
'94543' = '1' /* DEEP 3RD DEG BURN ANKLE */
'94544' = '1' /* DEEP 3RD DEG BRN LOW LEG */
'94545' = '1' /* DEEP 3RD DEG BURN KNEE */
'94546' = '1' /* DEEP 3RD DEG BURN THIGH */
'94549' = '1' /* DEEP 3 DEG BURN LEG-MULT */
'9455'  = '1' /* 3RD DEG BURN W LOSS-LEG */
'94550' = '1' /* 3 DEG BRN W LOSS-LEG NOS */
'94551' = '1' /* 3 DEG BURN W LOSS-TOE */
'94552' = '1' /* 3 DEG BURN W LOSS-FOOT */
'94553' = '1' /* 3 DEG BURN W LOSS-ANKLE */
'94554' = '1' /* 3 DEG BRN W LOSS-LOW LEG */
'94555' = '1' /* 3 DEG BURN W LOSS-KNEE */
'94556' = '1' /* 3 DEG BURN W LOSS-THIGH */
'94559' = '1' /* 3 DEG BRN W LOSS LEG-MLT */
'946'   = '1' /* BURN OF MULTIPLE SITE */
'9460'  = '1' /* BURN NOS MULTIPLE SITE */
'9461'  = '1' /* 1ST DEG BURN MULT SITE */
'9462'  = '1' /* 2ND DEG BURN MULT SITE */
'9463'  = '1' /* 3RD DEG BURN MULT SITE */
'9464'  = '1' /* DEEP 3 DEG BRN MULT SITE */
'9465'  = '1' /* 3RD BRN W LOSS-MULT SITE */
'947'   = '1' /* BURN OF INTERNAL ORGANS */
'9470'  = '1' /* BURN OF MOUTH & PHARYNX */
'9471'  = '1' /* BURN LARYNX/TRACHEA/LUNG */
'9472'  = '1' /* BURN OF ESOPHAGUS */
'9473'  = '1' /* BURN OF GI TRACT */
'9474'  = '1' /* BURN OF VAGINA & UTERUS */
'9478'  = '1' /* BURN INTERNAL ORGAN NEC */
```

PSFMTS -- CONTINUED

```
'9479'  = '1' /* BURN INTERNAL ORGAN NOS */
'948'   = '1' /* BURN BY % BODY SURFACE */
'9480'  = '1' /* BODY SURFACE BURN < 10% */
'94800' = '1' /* BDY BRN < 10%/3D DEG NOS */
'9481'  = '1' /* 10-19% BODY SURFACE BURN */
'94810' = '1' /* 10-19% BDY BRN/3 DEG NOS */
'94811' = '1' /* 10-19% BDY BRN/10-19% 3D */
'9482'  = '1' /* 20-29% BODY SURFACE BURN */
'94820' = '1' /* 20-29% BDY BRN/3 DEG NOS */
'94821' = '1' /* 20-29% BDY BRN/10-19% 3D */
'94822' = '1' /* 20-29% BDY BRN/20-29% 3D */
'9483'  = '1' /* 30-39% BODY SURFACE BURN */
'94830' = '1' /* 30-39% BDY BRN/3 DEG NOS */
'94831' = '1' /* 30-39% BDY BRN/10-19% 3D */
'94832' = '1' /* 30-39% BDY BRN/20-29% 3D */
'94833' = '1' /* 30-39% BDY BRN/30-39% 3D */
'9484'  = '1' /* 40-49% BODY SURFACE BURN */
'94840' = '1' /* 40-49% BDY BRN/3 DEG NOS */
'94841' = '1' /* 40-49% BDY BRN/10-19% 3D */
'94842' = '1' /* 40-49% BDY BRN/20-29% 3D */
'94843' = '1' /* 40-49% BDY BRN/30-39% 3D */
'94844' = '1' /* 40-49% BDY BRN/40-49% 3D */
'9485'  = '1' /* 50-59% BODY SURFACE BURN */
'94850' = '1' /* 50-59% BDY BRN/3 DEG NOS */
'94851' = '1' /* 50-59% BDY BRN/10-19% 3D */
'94852' = '1' /* 50-59% BDY BRN/20-29% 3D */
'94853' = '1' /* 50-59% BDY BRN/30-39% 3D */
'94854' = '1' /* 50-59% BDY BRN/40-49% 3D */
'94855' = '1' /* 50-59% BDY BRN/50-59% 3D */
'9486'  = '1' /* 60-69% BODY SURFACE BURN */
'94860' = '1' /* 60-69% BDY BRN/3 DEG NOS */
'94861' = '1' /* 60-69% BDY BRN/10-19% 3D */
'94862' = '1' /* 60-69% BDY BRN/20-29% 3D */
'94863' = '1' /* 60-69% BDY BRN/30-39% 3D */
'94864' = '1' /* 60-69% BDY BRN/40-49% 3D */
'94865' = '1' /* 60-69% BDY BRN/50-59% 3D */
'94866' = '1' /* 60-69% BDY BRN/60-69% 3D */
'9487'  = '1' /* 70-79% BODY SURFACE BURN */
'94870' = '1' /* 70-79% BDY BRN/3 DEG NOS */
'94871' = '1' /* 70-79% BDY BRN/10-19% 3D */
'94872' = '1' /* 70-79% BDY BRN/20-29% 3D */
'94873' = '1' /* 70-79% BDY BRN/30-39% 3D */
'94874' = '1' /* 70-79% BDY BRN/40-49% 3D */
'94875' = '1' /* 70-79% BDY BRN/50-59% 3D */
'94876' = '1' /* 70-79% BDY BRN/60-69% 3D */
'94877' = '1' /* 70-79% BDY BRN/70-79% 3D */
'9488'  = '1' /* 80-89% BODY SURFACE BURN */
'94880' = '1' /* 80-89% BDY BRN/3 DEG NOS */
'94881' = '1' /* 80-89% BDY BRN/10-19% 3D */
'94882' = '1' /* 80-89% BDY BRN/20-29% 3D */
'94883' = '1' /* 80-89% BDY BRN/30-39% 3D */
'94884' = '1' /* 80-89% BDY BRN/40-49% 3D */
'94885' = '1' /* 80-89% BDY BRN/50-59% 3D */
'94886' = '1' /* 80-89% BDY BRN/60-69% 3D */
'94887' = '1' /* 80-89% BDY BRN/70-79% 3D */
'94888' = '1' /* 80-89% BDY BRN/80-89% 3D */
'9489'  = '1' /* 90% OR MORE BDY SURF BRN */
```

```
'94890' = '1' /* 90% + BDY BRN/3D DEG NOS */
'94891' = '1' /* 90% + BDY BRN/10-19% 3RD */
```
PSFMTS -- CONTINUED

```
'94892' = '1' /* 90% + BDY BRN/20-29% 3RD */
'94893' = '1' /* 90% + BDY BRN/30-39% 3RD */
'94894' = '1' /* 90% + BDY BRN/40-49% 3RD */
'94895' = '1' /* 90% + BDY BRN/50-59% 3RD */
'94896' = '1' /* 90% + BDY BRN/60-69% 3RD */
'94897' = '1' /* 90% + BDY BRN/70-79% 3RD */
'94898' = '1' /* 90% + BDY BRN/80-89% 3RD */
'94899' = '1' /* 90% + BDY BRN/90% + 3RD */
'949'   = '1' /* BURN UNSPECIFIED */
'9490'  = '1' /* BURN NOS */
'9491'  = '1' /* 1ST DEGREE BURN NOS */
'9492'  = '1' /* 2ND DEGREE BURN NOS */
'9493'  = '1' /* 3RD DEGREE BURN NOS */
'9494'  = '1' /* DEEP 3RD DEG BURN NOS */
'9495'  = '1' /* 3RD BURN W LOSS-SITE NOS */
'952'   = '1' /* SPINAL CORD INJ W/O FX */
'9520'  = '1' /* CERVICAL SPINAL CORD INJ */
'95200' = '1' /* C1-C4 SPIN CORD INJ NOS */
'95201' = '1' /* COMPLETE LES CORD/C1-C4 */
'95202' = '1' /* ANTERIOR CORD SYND/C1-C4 */
'95203' = '1' /* CENTRAL CORD SYND/C1-C4 */
'95204' = '1' /* C1-C4 SPIN CORD INJ NEC */
'95205' = '1' /* C5-C7 SPIN CORD INJ NOS */
'95206' = '1' /* COMPLETE LES CORD/C5-C7 */
'95207' = '1' /* ANTERIOR CORD SYND/C5-C7 */
'95208' = '1' /* CENTRAL CORD SYND/C5-C7 */
'95209' = '1' /* C5-C7 SPIN CORD INJ NEC */
'9521'  = '1' /* DORSAL SPINAL CORD INJUR */
'95210' = '1' /* T1-T6 SPIN CORD INJ NOS */
'95211' = '1' /* COMPLETE LES CORD/T1-T6 */
'95212' = '1' /* ANTERIOR CORD SYND/T1-T6 */
'95213' = '1' /* CENTRAL CORD SYND/T1-T6 */
'95214' = '1' /* T1-T6 SPIN CORD INJ NEC */
'95215' = '1' /* T7-T12 SPIN CORD INJ NOS */
'95216' = '1' /* COMPLETE LES CORD/T7-T12 */
'95217' = '1' /* ANTERIOR CORD SYN/T7-T12 */
'95218' = '1' /* CENTRAL CORD SYN/T7-T12 */
'95219' = '1' /* T7-T12 SPIN CORD INJ NEC */
'9522'  = '1' /* LUMBAR SPINAL CORD INJUR */
'9523'  = '1' /* SACRAL SPINAL CORD INJUR */
'9524'  = '1' /* CAUDA EQUINA INJURY */
'9528'  = '1' /* SPIN CORD INJ-MULT SITE */
'9529'  = '1' /* SPINAL CORD INJURY NOS */
'953'   = '1' /* INJ NERVE ROOT/SPIN PLEX */
'9530'  = '1' /* CERVICAL ROOT INJURY */
'9531'  = '1' /* DORSAL ROOT INJURY */
'9532'  = '1' /* LUMBAR ROOT INJURY */
'9533'  = '1' /* SACRAL ROOT INJURY */
'9534'  = '1' /* BRACHIAL PLEXUS INJURY */
'9535'  = '1' /* LUMBOSACRAL PLEX INJURY */
'9538'  = '1' /* MULT NERVE ROOT/PLEX INJ */
'9539'  = '1' /* INJ NERVE ROOT/PLEX NOS */
'958'   = '1' /* EARLY COMPLIC OF TRAUMA */
'9580'  = '1' /* AIR EMBOLISM */
```

```
'9581'  = '1' /* FAT EMBOLISM */
'9582'  = '1' /* SECONDARY/RECUR HEMORR */
'9583'  = '1' /* POSTTRAUM WND INFEC NEC */
'9584'  = '1' /* TRAUMATIC SHOCK */
'9585'  = '1' /* TRAUMATIC ANURIA */
```

PSFMTS -- CONTINUED

```
'9586'  = '1' /* VOLKMANN'S ISCH CONTRACT */
'9587'  = '1' /* TRAUM SUBCUTAN EMPHYSEMA */
'9588'  = '1' /* EARLY COMPLIC TRAUMA NEC */
'E800'  = '1' /* RR COLLISION NOS */
'E8000' = '1' /* RR COLLISION NOS-EMPLOY */
'E8001' = '1' /* RR COLL NOS-PASSENGER */
'E8002' = '1' /* RR COLL NOS-PEDESTRIAN */
'E8003' = '1' /* RR COLL NOS-PED CYCLIST */
'E8008' = '1' /* RR COLL NOS-PERSON NEC */
'E8009' = '1' /* RR COLL NOS-PERSON NOS */
'E801'  = '1' /* RR COLL W OTH OBJECT */
'E8010' = '1' /* RR COLL W OTH OBJ-EMPLOY */
'E8011' = '1' /* RR COLL W OTH OBJ-PASNGR */
'E8012' = '1' /* RR COLL W OTH OBJ-PEDEST */
'E8013' = '1' /* RR COLL W OTH OBJ-CYCL */
'E8018' = '1' /* RR COL W OTH OBJ-PER NEC */
'E8019' = '1' /* RR COL W OTH OBJ-PER NOS */
'E802'  = '1' /* RR ACCIDENT W DERAILMENT */
'E8020' = '1' /* RR ACC W DERAIL-EMPLOYEE */
'E8021' = '1' /* RR ACC W DERAIL-PASSENG */
'E8022' = '1' /* RR ACC W DERAIL-PEDEST */
'E8023' = '1' /* RR ACC W DERAIL-PED CYCL */
'E8028' = '1' /* RR ACC W DERAIL-PERS NEC */
'E8029' = '1' /* RR ACC W DERAIL-PERS NOS */
'E803'  = '1' /* RR ACCIDENT W EXPLOSION */
'E8030' = '1' /* RR ACC W EXPLOSION-EMPL */
'E8031' = '1' /* RR ACC W EXPLOS-PASNGR */
'E8032' = '1' /* RR ACC W EXPLOS-PEDEST */
'E8033' = '1' /* RR ACC W EXPLOS-PED CYCL */
'E8038' = '1' /* RR ACC W EXPLOS-PERS NEC */
'E8039' = '1' /* RR ACC W EXPLOS-PERS NOS */
'E804'  = '1' /* FALL ON/FROM TRAIN */
'E8040' = '1' /* FALL ON/FROM TRAIN-EMPL */
'E8041' = '1' /* FALL FROM TRAIN-PASSENGR */
'E8042' = '1' /* FALL FROM TRAIN-PEDEST */
'E8043' = '1' /* FALL FROM TRAIN-PED CYCL */
'E8048' = '1' /* FALL FROM TRAIN-PERS NEC */
'E8049' = '1' /* FALL FROM TRAIN-PERS NOS */
'E805'  = '1' /* HIT BY ROLLING STOCK */
'E8050' = '1' /* HIT BY TRAIN-EMPLOYEE */
'E8051' = '1' /* HIT BY TRAIN-PASSENGER */
'E8052' = '1' /* HIT BY TRAIN-PEDESTRIAN */
'E8053' = '1' /* HIT BY TRAIN-PED CYCLIST */
'E8058' = '1' /* HIT BY TRAIN-PERSON NEC */
'E8059' = '1' /* HIT BY TRAIN-PERSON NOS */
'E806'  = '1' /* RAILWAY ACCIDENT NEC */
'E8060' = '1' /* RR ACC NEC-EMPLOYEE */
'E8061' = '1' /* RR ACC NEC-PASSENGER */
'E8062' = '1' /* RR ACC NEC-PEDESTRIAN */
'E8063' = '1' /* RR ACC NEC-PED CYCLIST */
'E8068' = '1' /* RR ACC NEC-PERSON NEC */
```

```
'E8069' = '1' /* RR ACC NEC-PERSON NOS */
'E807'  = '1' /* RAILWAY ACCIDENT NOS */
'E8070' = '1' /* RR ACCIDENT NOS-EMPLOYEE */
'E8071' = '1' /* RR ACC NOS-PASSENGER */
'E8072' = '1' /* RR ACC NOS-PEDESTRIAN */
'E8073' = '1' /* RR ACC NOS-PED CYCLIST */
'E8078' = '1' /* RR ACC NOS-PERSON NEC */
'E8079' = '1' /* RR ACC NOS-PERSON NOS */
```

PSFMTS -- CONTINUED

```
'E810'  = '1' /* VEHICLE-TRAIN COLLISION */
'E8100' = '1' /* MV-TRAIN COLL-DRIVER */
'E8101' = '1' /* MV-TRAIN COLL-PASNGR */
'E8102' = '1' /* MV-TRAIN COLL-MOTORCYCL */
'E8103' = '1' /* MV-TRAIN COLL-MCYCL PSGR */
'E8104' = '1' /* MV-TRAIN COLL-ST CAR */
'E8105' = '1' /* MV-TRAIN COLL-ANIM RID */
'E8106' = '1' /* MV-TRAIN COLL-PED CYCL */
'E8107' = '1' /* MV-TRAIN COLL-PEDEST */
'E8108' = '1' /* MV-TRAIN COLL-PERS NEC */
'E8109' = '1' /* MV-TRAIN COLL-PERS NOS */
'E811'  = '1' /* MV REENTRANT COLLISION */
'E8110' = '1' /* REENTRANT MV COLL-DRIVER */
'E8111' = '1' /* REENTRANT MV COLL-PASNGR */
'E8112' = '1' /* REENTRANT COLL-MOTCYCL */
'E8113' = '1' /* REENTRANT COLL-MCYC PSGR */
'E8114' = '1' /* REENTRANT COLL-ST CAR */
'E8115' = '1' /* REENTRANT COLL-ANIM RID */
'E8116' = '1' /* REENTRANT COLL-PED CYCL */
'E8117' = '1' /* REENTRANT COLL-PEDEST */
'E8118' = '1' /* REENTRANT COLL-PERS NEC */
'E8119' = '1' /* REENTRANT COLL-PERS NOS */
'E812'  = '1' /* MOT VEHIC COLLISION NOS */
'E8120' = '1' /* MV COLLISION NOS-DRIVER */
'E8121' = '1' /* MV COLLISION NOS-PASNGR */
'E8122' = '1' /* MV COLLIS NOS-MOTORCYCL */
'E8123' = '1' /* MV COLL NOS-MCYCL PSNGR */
'E8124' = '1' /* MV COLLISION NOS-ST CAR */
'E8125' = '1' /* MV COLL NOS-ANIM RID */
'E8126' = '1' /* MV COLL NOS-PED CYCL */
'E8127' = '1' /* MV COLLISION NOS-PEDEST */
'E8128' = '1' /* MV COLLIS NOS-PERS NEC */
'E8129' = '1' /* MV COLLIS NOS-PERS NOS */
'E813'  = '1' /* MV-OTH VEHICLE COLLISION */
'E8130' = '1' /* MV-OTH VEH COLL-DRIVER */
'E8131' = '1' /* MV-OTH VEH COLL-PASNGR */
'E8132' = '1' /* MV-OTH VEH COLL-MOTCYCL */
'E8133' = '1' /* MV-OTH VEH COLL-MCYC PSG */
'E8134' = '1' /* MV-OTH VEH COLL-ST CAR */
'E8135' = '1' /* MV-OTH VEH COLL-ANIM RID */
'E8136' = '1' /* MV-OTH VEH COLL-PED CYCL */
'E8137' = '1' /* MV-OTH VEH COLL-PEDEST */
'E8138' = '1' /* MV-OTH VEH COLL-PERS NEC */
'E8139' = '1' /* MV-OTH VEH COLL-PERS NOS */
'E814'  = '1' /* MV COLLISION W PEDEST */
'E8140' = '1' /* MV COLL W PEDEST-DRIVER */
'E8141' = '1' /* MV COLL W PEDEST-PASNGR */
'E8142' = '1' /* MV COLL W PEDEST-MOTCYCL */
```

```
'E8143' = '1' /* MV COLL W PED-MCYCL PSGR */
'E8144' = '1' /* MV COLL W PEDEST-ST CAR */
'E8145' = '1' /* MV COLL W PED-ANIM RID */
'E8146' = '1' /* MV COLL W PED-PED CYCL */
'E8147' = '1' /* MV COLL W PEDEST-PEDEST */
'E8148' = '1' /* MV COLL W PEDES-PERS NEC */
'E8149' = '1' /* MV COLL W PEDES-PERS NOS */
'E815'  = '1' /* MV COLLISION W OTH OBJ */
'E8150' = '1' /* MV COLL W OTH OBJ-DRIVER */
'E8151' = '1' /* MV COLL W OTH OBJ-PASNGR */
'E8152' = '1' /* MV COLL W OTH OBJ-MOCYCL */
```

PSFMTS -- CONTINUED

```
'E8153' = '1' /* MV COLL W OBJ-MCYCL PSGR */
'E8154' = '1' /* MV COLL W OBJ-ST CAR */
'E8155' = '1' /* MV COLL W OBJ-ANIM RIDER */
'E8156' = '1' /* MV COLL W OBJ-PED CYCL */
'E8157' = '1' /* MV COLL W OBJ-PEDEST */
'E8158' = '1' /* MV COLL W OBJ-PERS NEC */
'E8159' = '1' /* MV COLL W OBJ-PERS NOS */
'E816'  = '1' /* LOSS OF CONTROL MV ACC */
'E8160' = '1' /* LOSS CONTROL MV ACC-DRIV */
'E8161' = '1' /* LOSS CONTROL MV ACC-PSGR */
'E8162' = '1' /* LOSS CONTROL MV-MOCYCL */
'E8163' = '1' /* LOSS CONTROL MV-MCYC PSG */
'E8164' = '1' /* LOSS CONT MV ACC-ST CAR */
'E8165' = '1' /* LOSS CONT MV-ANIM RIDER */
'E8166' = '1' /* LOSS CONTROL MV-PED CYCL */
'E8167' = '1' /* LOSS CONTROL MV-PEDEST */
'E8168' = '1' /* LOSS CONTROL MV-PERS NEC */
'E8169' = '1' /* LOSS CONTROL MV-PERS NOS */
'E817'  = '1' /* MV TRAF ACC-BOARD/ALIGHT */
'E8170' = '1' /* MV ACC BOARD/ALIGHT-DRIV */
'E8171' = '1' /* MV ACC BOARD/ALIGHT-PSGR */
'E8172' = '1' /* MV BOARD/ALIGHT-MOTCYCL */
'E8173' = '1' /* MV BRD/ALIGHT-MCYCL PSGR */
'E8174' = '1' /* MV ACC BRD/ALIGHT-ST CAR */
'E8175' = '1' /* MV BRD/ALIGHT-ANIM RIDER */
'E8176' = '1' /* MV BRD/ALIGHT-PED CYCL */
'E8177' = '1' /* MV BRD/ALIGHT-PEDESTRIAN */
'E8178' = '1' /* MV BOARD/ALIGHT-PERS NEC */
'E8179' = '1' /* MV BOARD/ALIGHT-PERS NOS */
'E818'  = '1' /* MV TRAFFIC ACCIDENT NEC */
'E8180' = '1' /* MV TRAFF ACC NEC-DRIVER */
'E8181' = '1' /* MV TRAFF ACC NEC-PASNGR */
'E8182' = '1' /* MV TRAFF ACC NEC-MOCYCL */
'E8183' = '1' /* MV TRAFF ACC-MCYCL PSGR */
'E8184' = '1' /* MV TRAFF ACC NEC-ST CAR */
'E8185' = '1' /* MV TRAFF ACC-ANIM RIDER */
'E8186' = '1' /* MV TRAFF ACC-PED CYC */
'E8187' = '1' /* MV TRAFF ACC NEC-PEDEST */
'E8188' = '1' /* MV TRAFF ACC-PERS NEC */
'E8189' = '1' /* MV TRAFF ACC-PERS NOS */
'E819'  = '1' /* MV TRAFFIC ACCIDENT NOS */
'E8190' = '1' /* TRAFFIC ACC NOS-DRIVER */
'E8191' = '1' /* TRAFFIC ACC NOS-PASNGR */
'E8192' = '1' /* TRAFFIC ACC NOS-MOTCYCL */
'E8193' = '1' /* TRAFF ACC NOS-MCYCL PSGR */
```

```
'E8194' = '1' /* TRAFFIC ACC NOS-ST CAR */
'E8195' = '1' /* TRAFF ACC NOS-ANIM RIDER */
'E8196' = '1' /* TRAFFIC ACC NOS-PED CYCL */
'E8197' = '1' /* TRAFFIC ACC NOS-PEDEST */
'E8198' = '1' /* TRAFFIC ACC NOS-PERS NEC */
'E8199' = '1' /* TRAFFIC ACC NOS-PERS NOS */
'E820'  = '1' /* SNOW VEHICLE ACCIDENT */
'E8200' = '1' /* SNOW VEH ACC-DRIVER */
'E8201' = '1' /* SNOW VEH ACC-PASNGR */
'E8202' = '1' /* SNOW VEH ACC-MOTORCYCL */
'E8203' = '1' /* SNOW VEH ACC-MCYCL PSGR */
'E8204' = '1' /* SNOW VEH ACC-ST CAR */
'E8205' = '1' /* SNOW VEH ACC-ANIM RIDER */
'E8206' = '1' /* SNOW VEH ACC-PED CYCL */
```

PSFMTS -- CONTINUED

```
'E8207' = '1' /* SNOW VEH ACC-PEDEST */
'E8208' = '1' /* SNOW VEH ACC-PERS NEC */
'E8209' = '1' /* SNOW VEH ACC-PERS NOS */
'E821'  = '1' /* OTH OFF-RD MV N-TRAF ACC */
'E8210' = '1' /* OTH OFF-ROAD MV ACC-DRIV */
'E8211' = '1' /* OTH OFF-ROAD MV ACC-PSGR */
'E8212' = '1' /* OTH OFF-ROAD MV-MOCYCL */
'E8213' = '1' /* OTH OFF-ROAD MV-MCYC PSG */
'E8214' = '1' /* OTH OFF-ROAD MV-ST CAR */
'E8215' = '1' /* OTH OFF-ROAD MV-ANIM RID */
'E8216' = '1' /* OTH OFF-ROAD MV-PED CYCL */
'E8217' = '1' /* OTH OFF-ROAD MV-PEDEST */
'E8218' = '1' /* OTH OFF-ROAD MV-PERS NEC */
'E8219' = '1' /* OTH OFF-ROAD MV-PERS NOS */
'E822'  = '1' /* OTH MV COLL W MOVING OBJ */
'E8220' = '1' /* OTH COLL W MOV OBJ-DRIV */
'E8221' = '1' /* OTH COLL W MOV OBJ-PSGR */
'E8222' = '1' /* OTH COLL MOV OBJ-MOCYCL */
'E8223' = '1' /* OTH COLL MOV OBJ-CYC PSG */
'E8224' = '1' /* OTH COLL MOV OBJ-ST CAR */
'E8225' = '1' /* OTH COLL MOV OBJ-RIDER */
'E8226' = '1' /* OTH COLL MOV OBJ-PED CYC */
'E8227' = '1' /* OTH COLL MOV OBJ-PEDEST */
'E8228' = '1' /* OTH COLL MOV OBJ-PER NEC */
'E8229' = '1' /* OTH COLL MOV OBJ-PER NOS */
'E823'  = '1' /* OTH MV COLL W STNDNG OBJ */
'E8230' = '1' /* OTH COLL STNDNG OBJ-DRIV */
'E8231' = '1' /* OTH COLL STNDNG OBJ-PSGR */
'E8232' = '1' /* OTH COLL STND OBJ-MOCYCL */
'E8233' = '1' /* OTH COLL STN OBJ-CYC PSG */
'E8234' = '1' /* OTH COLL STND OBJ-ST CAR */
'E8235' = '1' /* OTH COLL STND OBJ-RIDER */
'E8236' = '1' /* OTH COLL STN OBJ-PED CYC */
'E8237' = '1' /* OTH COLL STND OBJ-PEDEST */
'E8238' = '1' /* OTH COLL STN OBJ-PER NEC */
'E8239' = '1' /* OTH COL-STND-OBJ-PER NOS */
'E824'  = '1' /* MV N-TRAFF BOARD/ALIGHT */
'E8240' = '1' /* N-TRAF BOARD/ALIGHT-DRIV */
'E8241' = '1' /* N-TRAF BOARD/ALIGHT-PSGR */
'E8242' = '1' /* N-TRAF BRD/ALIGHT-MOCYCL */
'E8243' = '1' /* N-TRAF BRD/ALIT-MCYC PSG */
'E8244' = '1' /* N-TRAF BRD/ALIT-ST CAR */
```

```
'E8245' = '1' /* N-TRAF BRD/ALIT-ANIM RID */
'E8246' = '1' /* N-TRAF BRD/ALIT-PED CYCL */
'E8247' = '1' /* N-TRAF BRD/ALIT-PEDEST */
'E8248' = '1' /* N-TRAF BRD/ALIT-PERS NEC */
'E8249' = '1' /* N-TRAF BRD/ALIT-PERS NOS */
'E825'  = '1' /* MV N-TRAFFIC ACC NEC/NOS */
'E8250' = '1' /* MV N-TRAFF ACC NEC-DRIV */
'E8251' = '1' /* MV N-TRAFF NEC/NOS-PSGR */
'E8252' = '1' /* MV N-TRAF ACC NEC-MOCYCL */
'E8253' = '1' /* MV N-TRAFF NEC-MCYC PSGR */
'E8254' = '1' /* MV N-TRAFF NEC-ST CAR */
'E8255' = '1' /* MV N-TRAF NEC-ANIM RIDER */
'E8256' = '1' /* MV N-TRAFF NEC-PED CYCL */
'E8257' = '1' /* MV N-TRAFF NEC-PEDEST */
'E8258' = '1' /* MV N-TRAFF NEC-PERS NEC */
'E8259' = '1' /* MV N-TRAFF NEC-PERS NOS */
'E826'  = '1' /* PEDAL CYCLE ACCIDENT */
```

PSFMTS -- CONTINUED

```
'E8260' = '1' /* PEDAL CYCLE ACC-PEDEST */
'E8261' = '1' /* PED CYCL ACC-PED CYCLIST */
'E8262' = '1' /* PED CYCLE ACC-ANIM RIDER */
'E8263' = '1' /* PED CYC ACC-OCC ANIM VEH */
'E8264' = '1' /* PED CYCLE ACC-OCC ST CAR */
'E8268' = '1' /* PED CYCLE ACC-PERS NEC */
'E8269' = '1' /* PED CYCLE ACC-PERS NOS */
'E827'  = '1' /* ANIMAL DRAWN VEHICLE ACC */
'E8270' = '1' /* ANIMAL DRAWN VEH-PEDEST */
'E8272' = '1' /* ANIM DRAWN VEH-ANIM RID */
'E8273' = '1' /* ANIMAL DRAWN VEH-OCCUPAN */
'E8274' = '1' /* ANIM DRAWN-OCC ST CAR */
'E8278' = '1' /* ANIM DRAWN VEH-PERS NEC */
'E8279' = '1' /* ANIM DRAWN VEH-PERS NOS */
'E828'  = '1' /* RIDDEN ANIMAL ACCIDENT */
'E8280' = '1' /* RIDDEN ANIMAL ACC-PEDEST */
'E8282' = '1' /* RIDDEN ANIMAL ACC-RIDER */
'E8284' = '1' /* RIDDEN ANIMAL ACC-ST CAR */
'E8288' = '1' /* RIDDEN ANIM ACC-PERS NEC */
'E8289' = '1' /* RIDDEN ANIM ACC-PERS NOS */
'E829'  = '1' /* OTHER ROAD VEHICLE ACC */
'E8290' = '1' /* OTH ROAD VEH ACC-PEDEST */
'E8294' = '1' /* OTH RD VEH ACC-ST CAR */
'E8298' = '1' /* OTH RD VEH ACC-PERS NEC */
'E8299' = '1' /* OTH RD VEH ACC-PERS NOS */
'E830'  = '1' /* BOAT ACC W SUBMERSION */
'E8300' = '1' /* BOAT ACC W SUBMERS-UNPOW */
'E8301' = '1' /* BOAT ACC W SUBMERS-POWER */
'E8302' = '1' /* BOAT ACC W SUBMERS-CREW */
'E8303' = '1' /* BOAT ACC W SUBMERS-PSGR */
'E8304' = '1' /* BOAT SUBMERS-WATER SKIER */
'E8305' = '1' /* BOAT SUBMERS-SWIMMER */
'E8306' = '1' /* BOAT SUBMERS-DOCKERS */
'E8308' = '1' /* BOAT SUBMERS-PERS NEC */
'E8309' = '1' /* BOAT SUBMERS-PERS NOS */
'E831'  = '1' /* BOAT ACC W OTH INJURY */
'E8310' = '1' /* BOAT ACC INJ NEC-UNPOWER */
'E8311' = '1' /* BOAT ACC INJ NEC-POWER */
'E8312' = '1' /* BOAT ACC INJ NEC-CREW */
```

```
'E8313' = '1' /* BOAT ACC INJ NEC-PASSENG */
'E8314' = '1' /* BOAT ACC INJ NEC-SKIER */
'E8315' = '1' /* BOAT ACC INJ NEC-SWIM */
'E8316' = '1' /* BOAT ACC INJ NEC-DOCKER */
'E8318' = '1' /* BOAT INJ NEC-PERSON NEC */
'E8319' = '1' /* BOAT INJ NEC-PERSON NOS */
'E832'  = '1' /* BOAT ACC W SUBMERS NEC */
'E8320' = '1' /* SUBMERS NEC-UNPOW BOAT */
'E8321' = '1' /* SUBMERS NEC-POWER BOAT */
'E8322' = '1' /* SUBMERS NEC-CREW */
'E8323' = '1' /* SUBMERS NEC-PASSENGER */
'E8324' = '1' /* SUBMERS NEC-WATER SKIER */
'E8325' = '1' /* SUBMERS NEC-SWIMMER */
'E8326' = '1' /* SUBMERS NEC-DOCKER */
'E8328' = '1' /* SUBMERS NEC-PERSON NEC */
'E8329' = '1' /* SUBMERS NEC-PERSON NOS */
'E833'  = '1' /* WATERCRAFT STAIRWAY FALL */
'E8330' = '1' /* W/CRAFT STAIR FALL-UNPOW */
'E8331' = '1' /* W/CRAFT STAIR FALL-POWER */
'E8332' = '1' /* WTRCRAFT STAIR FALL-CREW */
```

PSFMTS -- CONTINUED

```
'E8333' = '1' /* WTRCRAFT STAIR FALL-PSGR */
'E8334' = '1' /* W/CRAFT STAIR FALL-SKIER */
'E8335' = '1' /* W/CRAFT STAIR FALL-SWIM */
'E8336' = '1' /* W/CRF STAIR FALL-DOCKER */
'E8338' = '1' /* W/CRF STAIR FALL-PER NEC */
'E8339' = '1' /* W/CRF STAIR FALL-PER NOS */
'E834'  = '1' /* FALL IN WATERCRAFT NEC */
'E8340' = '1' /* W/CRAFT FALL NEC-UNPOW */
'E8341' = '1' /* W/CRAFT FALL NEC-POWER */
'E8342' = '1' /* WATERCRAFT FALL NEC-CREW */
'E8343' = '1' /* WTRCRAFT FALL NEC-PASNGR */
'E8344' = '1' /* W/CRAFT FALL NEC-SKIER */
'E8345' = '1' /* W/CRAFT FALL NEC-SWIM */
'E8346' = '1' /* WTRCRAFT FALL NEC-DOCKER */
'E8348' = '1' /* W/CRFT FALL NEC-PERS NEC */
'E8349' = '1' /* W/CRFT FALL NEC-PERS NOS */
'E835'  = '1' /* FALL IN WATERCRAFT NOS */
'E8350' = '1' /* W/CRAFT FALL NOS-UNPOW */
'E8351' = '1' /* W/CRAFT FALL NOS-POWER */
'E8352' = '1' /* WTRCRAFT FALL NOS-CREW */
'E8353' = '1' /* WTRCRAFT FALL NOS-PASNGR */
'E8354' = '1' /* W/CRAFT FALL NOS-SKIER */
'E8355' = '1' /* W/CRAFT FALL NOS-SWIM */
'E8356' = '1' /* WTRCRAFT FALL NOS-DOCKER */
'E8358' = '1' /* W/CRFT FALL NOS-PERS NEC */
'E8359' = '1' /* W/CRFT FALL NOS-PERS NOS */
'E836'  = '1' /* WATERCRAFT MACHINERY ACC */
'E8360' = '1' /* MACHINE ACC-UNPOW BOAT */
'E8361' = '1' /* MACH ACC-OCC POWER BOAT */
'E8362' = '1' /* MACHINERY ACCIDENT-CREW */
'E8363' = '1' /* MACHINERY ACC-PASNGR */
'E8364' = '1' /* MACHINE ACCIDENT-SKIER */
'E8365' = '1' /* MACHINE ACCIDENT-SWIM */
'E8366' = '1' /* MACHINERY ACC-DOCKER */
'E8368' = '1' /* MACHINERY ACC-PERS NEC */
'E8369' = '1' /* MACHINERY ACC-PERS NOS */
```

```
'E837'  = '1' /* WATERCRAFT EXPLOSION */
'E8370' = '1' /* EXPLOSION-OCC UNPOW BOAT */
'E8371' = '1' /* EXPLOSION-OCC POWER BOAT */
'E8372' = '1' /* WATERCRAFT EXPLOS-CREW */
'E8373' = '1' /* WATERCRAFT EXPLOS-PASNGR */
'E8374' = '1' /* WATERCRAFT EXPLOS-SKIER */
'E8375' = '1' /* WATERCRAFT EXPLOS-SWIM */
'E8376' = '1' /* WATERCRAFT EXPLOS-DOCKER */
'E8378' = '1' /* WATERCRAFT EXPL-PERS NEC */
'E8379' = '1' /* WATERCRAFT EXPL-PERS NOS */
'E838'  = '1' /* WATERCRAFT ACC NEC/NOS */
'E8380' = '1' /* WATERCRAFT ACC NEC-UNPOW */
'E8381' = '1' /* WATERCRAFT ACC NEC-POWER */
'E8382' = '1' /* WATERCRAFT ACC NEC-CREW */
'E8383' = '1' /* WATERCRFT ACC NEC-PASNGR */
'E8384' = '1' /* WATERCRAFT ACC NEC-SKIER */
'E8385' = '1' /* WATRCRFT ACC NEC-SWIMMER */
'E8386' = '1' /* WATERCRFT ACC NEC-DOCKER */
'E8388' = '1' /* WTRCRFT ACC NEC-PERS NEC */
'E8389' = '1' /* WTRCRFT ACC NEC-PERS NOS */
'E840'  = '1' /* TAKEOFF/LANDING ACCIDENT */
'E8400' = '1' /* TK OFF/LAND-SPCRFT */
'E8401' = '1' /* TK OFF/LAND-MILIT CRAFT */
```

PSFMTS -- CONTINUED

```
'E8402' = '1' /* TK OFF/LAND-CREW AIRCRFT */
'E8403' = '1' /* TK OFF/LAND-PSNG AIRCRFT */
'E8404' = '1' /* TK OFF/LAND-COMM CRF NEC */
'E8405' = '1' /* TK OFF/LAND-AIRCRAFT NEC */
'E8406' = '1' /* TK OFF/LAND-UNP AIRCRFT */
'E8407' = '1' /* TK OFF/LAND-PARACHUTIST */
'E8408' = '1' /* TK OFF/LAND-GROUND CREW */
'E8409' = '1' /* TK OFF/LAND-PERS NEC */
'E841'  = '1' /* POWERED AIRCRAFT ACC NEC */
'E8410' = '1' /* POW AIRCRAFT ACC-SPCRFT */
'E8411' = '1' /* POWER AIRCRAFT ACC-MILIT */
'E8412' = '1' /* POWER AIRCRAFT ACC-CREW */
'E8413' = '1' /* POWER AIRCRAFT ACC-PSNGR */
'E8414' = '1' /* AIRCRAF ACC-OCC COMM NEC */
'E8415' = '1' /* OTH POWERED AIRCRAFT ACC */
'E8416' = '1' /* POW AIRC ACC-UNP AIRCR */
'E8417' = '1' /* AIRCRAFT ACC-PARACHUTIST */
'E8418' = '1' /* AIRCRAFT ACC-GROUND CREW */
'E8419' = '1' /* AIRCRFT ACC NOS-PERS NEC */
'E842'  = '1' /* UNPOWERED AIRCRAFT ACC */
'E8426' = '1' /* UNPOWER AIRCRAFT ACC-OCC */
'E8427' = '1' /* UNPOW AIRCRF ACC-CHUTIST */
'E8428' = '1' /* UNPOW AIRCRF ACC-GR CREW */
'E8429' = '1' /* UNPOW AIRCRF ACC-PER NEC */
'E843'  = '1' /* FALL IN/ON/FROM AIRCRAFT */
'E8430' = '1' /* FALL-OCC SPACECRAFT */
'E8431' = '1' /* FALL-MILIT AIRCRAFT OCCP */
'E8432' = '1' /* FALL-CREW COMM AIRCRAFT */
'E8433' = '1' /* FALL-PSNG COMM AIRCRAFT */
'E8434' = '1' /* FALL-OCC COMM AIRCRF NEC */
'E8435' = '1' /* FALL-OCCUP OTH AIRCRAFT */
'E8436' = '1' /* FALL-OCC UNPOWER AIRCRAF */
'E8437' = '1' /* FALL-PARACHUTIST */
```

```
'E8438' = '1' /* AIRCRFT FALL-GROUND CREW */
'E8439' = '1' /* AIRCRAFT FALL-PERSON NEC */
'E844'  = '1' /* AIR TRANSPORT ACC NEC */
'E8440' = '1' /* AIRCRFT ACC NEC-SPCRFT */
'E8441' = '1' /* AIRCRFT ACC NEC-MILITARY */
'E8442' = '1' /* AIRCRFT ACC NEC-CREW */
'E8443' = '1' /* AIRCRFT ACC NEC-PASNGR */
'E8444' = '1' /* AIRCRFT ACC NEC-COMM NEC */
'E8445' = '1' /* AIRCRFT ACC NEC-OCCP NEC */
'E8446' = '1' /* AIRCR ACC NEC-UNP AIRCR */
'E8447' = '1' /* AIRCRFT ACC-PARACHUTIST */
'E8448' = '1' /* AIRCRFT ACC NEC-GRD CREW */
'E8449' = '1' /* AIRCRFT ACC NEC-PERS NEC */
'E845'  = '1' /* SPACECRAFT ACCIDENT */
'E8450' = '1' /* SPACECRAFT ACC-OCCUPANT */
'E8458' = '1' /* SPACECRAFT ACC-GRND CREW */
'E8459' = '1' /* SPACECRAFT ACC-PERS NEC */
'E846'  = '1' /* INDUS VEH ACC ON PREMISE */
'E847'  = '1' /* CABL CAR ACC NOT ON RAIL */
'E848'  = '1' /* OTH VEHICLE ACC NEC */
'E849'  = '1' /* PLACE OF OCCURRENCE */
'E8490' = '1' /* ACCIDENT IN HOME */
'E8491' = '1' /* ACCIDENT ON FARM */
'E8492' = '1' /* ACCIDENT IN MINE/QUARRY */
'E8493' = '1' /* ACC ON INDUSTR PREMISES */
'E8494' = '1' /* ACCID IN RECREATION AREA */
```

PSFMTS -- CONTINUED

```
'E8495' = '1' /* ACCID ON STREET/HIGHWAY */
'E8496' = '1' /* ACCIDENT IN PUBLIC BLDG */
'E8497' = '1' /* ACCID IN RESIDENT INSTIT */
'E8498' = '1' /* ACCIDENT IN PLACE NEC */
'E8499' = '1' /* ACCIDENT IN PLACE NOS */
'E880'  = '1' /* FALL ON STAIRS OR STEPS */
'E8800' = '1' /* FALL ON ESCALATOR */
'E8801' = '1' /* FALL ON SIDEWALK CURB */
'E8809' = '1' /* FALL ON STAIR/STEP NEC */
'E881'  = '1' /* FALL-LADDER OR SCAFFOLD */
'E8810' = '1' /* FALL FROM LADDER */
'E8811' = '1' /* FALL FROM SCAFFOLDING */
'E882'  = '1' /* FALL FROM BUILDING */
'E883'  = '1' /* FALL INTO HOLE */
'E8830' = '1' /* DIVING ACCIDENT */
'E8831' = '1' /* FALL INTO WELL */
'E8832' = '1' /* FALL INTO STORM DRAIN */
'E8839' = '1' /* FALL INTO OTHER HOLE */
'E884'  = '1' /* OTH FALL-1 LEVEL TO OTH */
'E8840' = '1' /* FALL FROM PLAYGRND EQUIP */
'E8841' = '1' /* FALL FROM CLIFF */
'E8842' = '1' /* FALL FROM CHAIR */
'E8843' = '1' /* FALL FROM WHEELCHAIR */
'E8844' = '1' /* FALL FROM BED */
'E8845' = '1' /* FALL FROM FURNITURE NEC */
'E8846' = '1' /* FALL FROM COMMODE */
'E8849' = '1' /* FALL-1 LEVEL TO OTH NEC */
'E885'  = '1' /* FALL ON LEVEL-TRIPPING */
'E8851' = '1' /* FALL FROM ROLLER SKATES */
'E8852' = '1' /* FALL FROM SKATEBOARD */
```

```
'E8853' = '1' /* FALL FROM SKIS */
'E8854' = '1' /* FALL FROM SNOWBOARD */
'E8859' = '1' /* FALL FROM SLIPPING NEC */
'E886'  = '1' /* FALL ON LEVEL-PUSHED */
'E8860' = '1' /* FALL IN SPORTS */
'E8869' = '1' /* FALL ON LEVEL NEC/NOS */
'E887'  = '1' /* FRACTURE, CAUSE NOS */
'E888'  = '1' /* FALL NEC & NOS */
'E890'  = '1' /* FIRE IN PRIVATE DWELLING */
'E8900' = '1' /* PRIVAT DWELL FIRE-EXPLOS */
'E8901' = '1' /* PRIV DWEL FIRE-PVC FUMES */
'E8902' = '1' /* PRIV DWEL FIRE-FUMES NOS */
'E8903' = '1' /* PRIV DWEL FIRE-BURNING */
'E8908' = '1' /* PRIV DWEL FIRE-ACCID NEC */
'E8909' = '1' /* PRIV DWEL FIRE-ACCID NOS */
'E891'  = '1' /* FIRE IN OTHER BUILDING */
'E8910' = '1' /* FIRE IN BLDG-EXPLOSION */
'E8911' = '1' /* FIRE IN BLDG-PVC FUMES */
'E8912' = '1' /* FIRE IN BLDG-FUMES NOS */
'E8913' = '1' /* FIRE IN BLDG-BURNING */
'E8918' = '1' /* FIRE IN BLDG-ACCID NEC */
'E8919' = '1' /* FIRE IN BLDG-ACCID NOS */
'E892'  = '1' /* FIRE NOT IN BUILDING */
'E893'  = '1' /* CLOTHING FIRE ACCIDENT */
'E8930' = '1' /* CLOTHING FIRE-PRIV DWELL */
'E8931' = '1' /* CLOTHING FIRE-BLDG NEC */
'E8932' = '1' /* CLOTHING FIRE NOT IN BLD */
'E8938' = '1' /* CLOTHING FIRE NEC */
'E8939' = '1' /* CLOTHING FIRE NOS */
```

PSFMTS -- CONTINUED

```
'E894'  = '1' /* FIRE-HIGHLY INFLAM MATER */
'E895'  = '1' /* BURN ACC IN PRIVAT DWELL */
'E896'  = '1' /* BURN ACC IN BLDG NEC */
'E897'  = '1' /* BURN ACC NOT IN BLDG */
'E898'  = '1' /* OTHER FIRE ACCIDENT */
'E8980' = '1' /* BURNING BEDCLOTHES */
'E8981' = '1' /* FIRE ACCIDENT NEC */
'E899'  = '1' /* FIRE ACCIDENT NOS */
'E910'  = '1' /* ACCIDENTAL DROWNING */
'E9100' = '1' /* WATER-SKIING ACCIDENT */
'E9101' = '1' /* SKIN/SCUBA DIVING ACC */
'E9102' = '1' /* SWIMMING ACCIDENT NOS */
'E9103' = '1' /* SWIMMING/DIVING ACC NEC */
'E9104' = '1' /* DROWNING IN BATHTUB */
'E9108' = '1' /* ACCIDENTAL DROWNING NEC */
'E9109' = '1' /* ACCIDENTAL DROWNING NOS */
'E913'  = '1' /* ACC MECHANICAL SUFFOCAT */
'E9130' = '1' /* SUFFOCAT IN BED/CRADLE */
'E9131' = '1' /* SUFFOCATION-PLASTIC BAG */
'E9132' = '1' /* SUFFOCATION-LACK OF AIR */
'E9133' = '1' /* CAVE-IN NOS */
'E9138' = '1' /* SUFFOCATION NEC */
'E9139' = '1' /* SUFFOCATION NOS */
'E914'  = '1' /* FB ENTERING EYE */
'E915'  = '1' /* FB ENTERING OTH ORIFICE */
'E916'  = '1' /* STRUCK BY FALLING OBJECT */
'E917'  = '1' /* STRUCK BY OBJECT/PERSON */
```

```
'E9170' = '1' /* STRUCK IN SPORTS */
'E9171' = '1' /* CROWD ACCIDENT */
'E9172' = '1' /* STRUCK IN RUNNING WATER */
'E9179' = '1' /* STRUCK BY OBJ/PERSON NEC */
'E918'  = '1' /* CAUGHT BETWEEN OBJECTS */
'E919'  = '1' /* MACHINERY ACCIDENTS */
'E9190' = '1' /* MACHINE ACCID-AGRICULT */
'E9191' = '1' /* MACHINE ACCID-MINING */
'E9192' = '1' /* LIFTING MACHINE ACCIDENT */
'E9193' = '1' /* METALWORKING MACHINE ACC */
'E9194' = '1' /* WOODWORKING MACHINE ACC */
'E9195' = '1' /* PRIME MOVER MACHINE ACC */
'E9196' = '1' /* TRANSMISSION MACHINE ACC */
'E9197' = '1' /* EARTH MOVING MACHINE ACC */
'E9198' = '1' /* MACHINERY ACCIDENT NEC */
'E9199' = '1' /* MACHINERY ACCIDENT NOS */
'E920'  = '1' /* CUTTING INSTRUMENT ACCID */
'E9200' = '1' /* ACC-POWERED LAWN MOWER */
'E9201' = '1' /* ACC-POWER HAND TOOL NEC */
'E9202' = '1' /* ACC-POWER HOUSE APPLIANC */
'E9203' = '1' /* KNIFE/SWORD/DAGGER ACC */
'E9204' = '1' /* ACCID-OTHER HAND TOOLS */
'E9205' = '1' /* ACC-HYPODERMIC NEEDLE */
'E9208' = '1' /* ACC-CUTTING INSTRUM NEC */
'E9209' = '1' /* ACC-CUTTING INSTRUM NOS */
'E921'  = '1' /* PRESSUR VESSEL EXPLOSION */
'E9210' = '1' /* BOILER EXPLOSION */
'E9211' = '1' /* GAS CYLINDER EXPLOSION */
'E9218' = '1' /* PRESS VESSEL EXPLOS NEC */
'E9219' = '1' /* PRESS VESSEL EXPLOS NOS */
'E922'  = '1' /* FIREARM ACCIDENTS */
'E9220' = '1' /* HANDGUN ACCIDENT */
```

PSFMTS -- CONTINUED

```
'E9221' = '1' /* SHOTGUN ACCIDENT */
'E9222' = '1' /* HUNTING RIFLE ACCIDENT */
'E9223' = '1' /* MILITARY FIREARM ACCID */
'E9224' = '1' /* ACCIDENT - AIR GUN */
'E9228' = '1' /* FIREARM ACCIDENT NEC */
'E9229' = '1' /* FIREARM ACCIDENT NOS */
'E923'  = '1' /* ACCIDENT D/T EXPLOSIVES */
'E9230' = '1' /* FIREWORKS ACCIDENT */
'E9231' = '1' /* BLASTING MATERIALS ACCID */
'E9232' = '1' /* EXPLOSIVE GASES ACCIDENT */
'E9238' = '1' /* EXPLOSIVES ACCIDENT NEC */
'E9239' = '1' /* EXPLOSIVES ACCIDENT NOS */
'E924'  = '1' /* HOT SUBSTANCE ACCIDENT */
'E9240' = '1' /* ACC-HOT LIQUID & STEAM */
'E9241' = '1' /* ACCID-CAUSTIC SUBSTANCE */
'E9242' = '1' /* ACC-HOT TAP WATER */
'E9248' = '1' /* HOT SUBSTANCE ACCID NEC */
'E9249' = '1' /* HOT SUBSTANCE ACCID NOS */
'E925'  = '1' /* ELECTRIC CURRENT ACCID */
'E9250' = '1' /* DOMESTIC WIRING ACCIDENT */
'E9251' = '1' /* ELECTR POWER GENERAT ACC */
'E9252' = '1' /* INDUST WIRING/MACHIN ACC */
'E9258' = '1' /* ELECTRIC CURRENT ACC NEC */
'E9259' = '1' /* ELECTRIC CURRENT ACC NOS */
```

```
'E926'  = '1' /* RADIATION EXPOSURE ACCID */
'E9260' = '1' /* RADIOFREQ RADIAT EXPOSUR */
'E9261' = '1' /* INFRA-RED APPL RAD EXOS */
'E9262' = '1' /* VIS/ULTRAVIOL LGHT EXPOS */
'E9263' = '1' /* X-RAY/GAMMA RAY EXPOSURE */
'E9264' = '1' /* LASER EXPOSURE */
'E9265' = '1' /* RADIOACT ISOTOPE EXPOSUR */
'E9268' = '1' /* RADIATION EXPOSURE NEC */
'E9269' = '1' /* RADIATION EXPOSURE NOS */
'E927'  = '1' /* ACCID FROM OVEREXERTION */
'E928'  = '1' /* ACCIDENTS NEC & NOS */
'E9280' = '1' /* ACC D/T WEIGHTLESS ENVIR */
'E9281' = '1' /* EXPOSURE TO NOISE */
'E9282' = '1' /* EXPOSURE TO VIBRATION */
'E9283' = '1' /* HUMAN BITE - ACCIDENTAL */
'E9288' = '1' /* ACCIDENT NEC */
'E9289' = '1' /* ACCIDENT NOS */
'E960'  = '1' /* FIGHT, BRAWL, RAPE */
'E9600' = '1' /* UNARMED FIGHT OR BRAWL */
'E9601' = '1' /* RAPE */
'E961'  = '1' /* ASSAULT-CORROSIV/CAUST */
'E962'  = '1' /* ASSAULT-POISONING */
'E9620' = '1' /* ASSAULT-POIS W MEDIC AGT */
'E9621' = '1' /* ASSAULT-POIS W SOLID/LIQ */
'E9622' = '1' /* ASSAULT-POIS W GAS/VAPOR */
'E9629' = '1' /* ASSAULT-POISONING NOS */
'E963'  = '1' /* ASSAULT-HANGING/STRANGUL */
'E964'  = '1' /* ASSAULT-SUBMERSION */
'E965'  = '1' /* ASSAULT-FIREARM/EXPLOSIV */
'E9650' = '1' /* ASSAULT-HANDGUN */
'E9651' = '1' /* ASSAULT-SHOTGUN */
'E9652' = '1' /* ASSAULT-HUNTING RIFLE */
'E9653' = '1' /* ASSAULT-MILITARY WEAPON */
'E9654' = '1' /* ASSAULT-FIREARM NEC */
'E9655' = '1' /* ASSAULT-ANTIPERSON BOMB */
```

PSFMTS -- CONTINUED

```
'E9656' = '1' /* ASSAULT-GASOLINE BOMB */
'E9657' = '1' /* ASSAULT-LETTER BOMB */
'E9658' = '1' /* ASSAULT-EXPLOSIVE NEC */
'E9659' = '1' /* ASSAULT-EXPLOSIVE NOS */
'E966'  = '1' /* ASSAULT-CUTTING INSTR */
'E967'  = '1' /* CHILD BATTERING/MALTREAT */
'E9670' = '1' /* ABUSE BY FTHER/STPFTH/BF */
'E9671' = '1' /* CHILD ABUSE BY PERS NEC */
'E9672' = '1' /* ABUSE BY MTHER/STPMTH/GF */
'E9673' = '1' /* BATTER BY SPOUSE/PARTNER */
'E9674' = '1' /* BATTERING BY CHILD */
'E9675' = '1' /* BATTERING BY SIBLING */
'E9676' = '1' /* BATTERING BY GRANDPARENT */
'E9677' = '1' /* BATTER BY OTHER RELATIVE */
'E9678' = '1' /* BATTER BY NON-RELATIVE */
'E9679' = '1' /* CHILD ABUSE NOS */
'E968'  = '1' /* ASSAULT NEC & NOS */
'E9680' = '1' /* ASSAULT-FIRE */
'E9681' = '1' /* ASSLT-PUSH FROM HI PLACE */
'E9682' = '1' /* ASSAULT-STRIKING W OBJ */
'E9683' = '1' /* ASSAULT-HOT LIQUID */
```

```
'E9684' = '1' /* ASSAULT-CRIMINAL NEGLECT */
'E9685' = '1' /* ASSLT-TRANSPORT VEHICLE */
'E9686' = '1' /* ASSAULT - AIR GUN */
'E9687' = '1' /* HUMAN BITE - ASSAULT */
'E9688' = '1' /* ASSAULT NEC */
'E9689' = '1' /* ASSAULT NOS */
'E969'  = '1' /* LATE EFFECT ASSAULT */
'E970'  = '1' /* LEGAL INTERVENT-FIREARM */
'E971'  = '1' /* LEGAL INTERVENT-EXPLOSIV */
'E972'  = '1' /* LEGAL INTERVENT-GAS */
'E973'  = '1' /* LEGAL INTERVEN-BLUNT OBJ */
'E974'  = '1' /* LEGAL INTERVEN-CUT INSTR */
'E975'  = '1' /* LEGAL INTERVENTION NEC */
'E976'  = '1' /* LEGAL INTERVENTION NOS */
'E977'  = '1' /* LATE EFF-LEGAL INTERVENT */
'E978'  = '1' /* LEGAL EXECUTION */
'E980'  = '1' /* UNDETERM POIS-SOL/LIQUID */
'E9800' = '1' /* UNDETERM POIS-ANALGESICS */
'E9801' = '1' /* UNDETERM POIS-BARBITURAT */
'E9802' = '1' /* UNDET POIS-SED/HYPN NEC */
'E9803' = '1' /* UNDETERM POIS-PSYCHOTROP */
'E9804' = '1' /* UNDET POIS-MED AGNT NEC */
'E9805' = '1' /* UNDET POIS-MED AGNT NOS */
'E9806' = '1' /* UNDET POIS-CORROS/CAUST */
'E9807' = '1' /* UNDET POIS-AGRICULT AGNT */
'E9808' = '1' /* UNDETER POIS-ARSENIC */
'E9809' = '1' /* UNDETER POIS-SOL/LIQ NEC */
'E981'  = '1' /* UNDETER POIS-UTILITY GAS */
'E9810' = '1' /* UNDETER POIS-PIPED GAS */
'E9811' = '1' /* UNDET POIS-CONTAINER GAS */
'E9818' = '1' /* UNDET POIS-UTIL GAS NEC */
'E982'  = '1' /* UNDETERM POISON-GAS NEC */
'E9820' = '1' /* UNDETER POIS-EXHAUST GAS */
'E9821' = '1' /* UNDETERMIN POISON-CO NEC */
'E9828' = '1' /* UNDET POIS-GAS/VAPOR NEC */
'E9829' = '1' /* UNDET POIS-GAS/VAPOR NOS */
'E983'  = '1' /* UNDETERM CIRC-SUFFOCATN */
'E9830' = '1' /* UNDETERMIN CIRC-HANGING */
```

PSFMTS -- CONTINUED

```
'E9831' = '1' /* UNDET CIRC-SUF PLAST BAG */
'E9838' = '1' /* UNDET CIRC-SUFFOCATE NEC */
'E9839' = '1' /* UNDET CIRC-SUFFOCATE NOS */
'E984'  = '1' /* UNDETERM CIRC-SUBMERSION */
'E985'  = '1' /* UNDET CIRC-FIREARM/EXPL */
'E9850' = '1' /* UNDETERMIN CIRC-HANDGUN */
'E9851' = '1' /* UNDETERMIN CIRC-SHOTGUN */
'E9852' = '1' /* UNDET CIRC-HUNTING RIFLE */
'E9853' = '1' /* UNDET CIRC-MILITARY ARMS */
'E9854' = '1' /* UNDETER CIRC-FIREARM NEC */
'E9855' = '1' /* UNDETERM CIRC-EXPLOSIVE */
'E9856' = '1' /* UNDETRMINE ACCID-AIR GUN */
'E986'  = '1' /* UNDET CIRC-CUT INSTRUMNT */
'E987'  = '1' /* UNDET CIRC-FALL HI PLACE */
'E9870' = '1' /* UNDET CIRC-FALL RESIDENC */
'E9871' = '1' /* UNDET FALL STRUCTURE NEC */
'E9872' = '1' /* UNDET FALL NATURAL SITE */
'E9879' = '1' /* UNDET CIRC-FALL SITE NOS */
```

```
    'E988'  = '1' /* UNDETERMINED CIRCUM NEC */
    'E9880' = '1' /* UNDETERM CIRC-MOVING OBJ */
    'E9881' = '1' /* UNDETERM CIRC-BURN, FIRE */
    'E9882' = '1' /* UNDETERM CIRC-SCALD */
    'E9883' = '1' /* UNDETERM CIRC-EXTRM COLD */
    'E9884' = '1' /* UNDETERM CIRC-ELECTROCUT */
    'E9885' = '1' /* UNDET CIRC-MOT VEH CRASH */
    'E9886' = '1' /* UNDET CIRC-AIRCRFT CRASH */
    'E9887' = '1' /* UNDET CIRC-CAUSTIC SUBST */
    'E9888' = '1' /* UNDETERMIN CIRCUMST NEC */
    'E9889' = '1' /* UNDETERMIN CIRCUMST NOS */
    'E989'  = '1' /* LATE EFF INJ-UNDET CIRC */
    'E990'  = '1' /* WAR INJURY FIRE/CONFLAG */
    'E9900' = '1' /* WAR INJ GASOLINE BOMB */
    'E9909' = '1' /* WAR INJURY FIRE NEC */
    'E991'  = '1' /* WAR INJ BULLETS/FRAGM */
    'E9910' = '1' /* WAR INJ RUBBER BULLET */
    'E9911' = '1' /* WAR INJURY PELLETS */
    'E9912' = '1' /* WAR INJURY BULLET NEC */
    'E9913' = '1' /* WAR INJ ANTIPERSON BOMB */
    'E9919' = '1' /* WAR INJ FRAGMENTS NEC */
    'E992'  = '1' /* WAR INJ MARINE EXPLOS */
    'E993'  = '1' /* WAR INJURY EXPLOS NEC */
    'E994'  = '1' /* WAR INJ AIRCRFT DESTRUC */
    'E995'  = '1' /* WAR INJUR-CONVEN WAR NEC */
    'E996'  = '1' /* WAR INJ NUCLEAR WEAPONS */
    'E997'  = '1' /* WAR INJ-UNCONVEN WAR NEC */
    'E9970' = '1' /* WAR INJURY LASERS */
    'E9971' = '1' /* WAR INJURY BIOL WARFARE */
    'E9972' = '1' /* WAR INJURY GAS/FUM/CHEM */
    'E9978' = '1' /* WAR INJ-UNCONVEN WAR NEC */
    'E9979' = '1' /* WAR INJ-UNCONVEN WAR NOS */
    'E998'  = '1' /* WAR INJ POST WAR OPERAT */
    'E999'  = '1' /* LATE EFF OF WAR INJURY */
    OTHER   = '0' ;

    /* Trauma */
    /* Diagnostic Related Groups (DRGs): */

    VALUE $TRAUMDR
    '002' = '1' /* CRANIOTOMY FOR TRAUMA, AGE GREATER THAN 17 */
    PSFMTS -- CONTINUED

    '027' = '1' /* TRAUMATIC STUPOR AND COMA, COMA GREATER THAN ONE HOUR */
    '028' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE
GREATER
                  THAN 17 WITH CC */
    '029' = '1' /* TRAUMATIC STUPOR AND COMA, COMA LESS THAN ONE HOUR, AGE
GREATER
                  THAN 17 WITHOUT CC */
    '031' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITHOUT CC */
    '032' = '1' /* CONCUSSION, AGE GREATER THAN 17 WITHOUT CC */
    '072' = '1' /* NASAL TRAUMA AND DEFORMITY */
    '083' = '1' /* MAJOR CHEST TRAUMA WITH CC */
    '084' = '1' /* MAJOR CHEST TRAUMA WITHOUT CC */
    '235' = '1' /* FRACTURES OF FEMUR */
    '236' = '1' /* FRACTURE OF HIP AND PELVIS */
```

```
    '237' = '1' /* SPRAINS, STRAINS AND DISLOCATIONS OF HIP, PELVIS AND THIGH
*/
    '440' = '1' /* WOUND DEBRIDEMENTS FOR INJURIES */
    '441' = '1' /* HAND PROCEDURES FOR INJURIES */
    '442' = '1' /* OTHER OR PROCEDURES FOR INJURIES WITH CC */
    '443' = '1' /* OHER OR PROCEDURES FOR INJRUIES WITHOUT CC */
    '444' = '1' /* AUMATIC INJRY, AGE GREATER THAN 17 WITH CC */
    '445' = '1' /* AUMATIC INJRY, AGE GREATER THAN 17 WITHOUT CC */
    '446' = '1' /* AUMATIC INJURY, AGE 0-17 */
    '447' = '1' /* LERFIC REACTIONS, AGE GREATER THAN 17 */
    '448' = '1' /* ALLERGIC REACTIONS, AGE 0-17 */
    '449' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE GREATER THAN 17
                   WITH CC */
    '450' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE GREATER THAN 17
                   WITHOUT CC */
    '451' = '1' /* POISONING AND TOXIC EFFECTS OF DRUGS, AGE 0-17 */
    '452' = '1' /* COMPLICATIONS OF TREATMENT WITH CC */
    '453' = '1' /* MPLICATIONS OF TREATMENT WITHOUT CC */
    '454' = '1' /* OTHER INJURY, POISONING AND TOXIC EFFECT DIAGNOSES WITH CC
*/
    '455' = '1' /* OTHER INJURY, POISONING AND TOXIC EFFECT DIAGNOSES WITHOUT
CC */
    '460' = '1' /* NO LONGER VALID */
    '484' = '1' /* CRANIOTOMY FOR MULTIPLE SIGNIFICANT TRAUMA */
    '485' = '1' /* LIMB REATTACHMENT, HIP AND FEMUR PROCEDURES FOR MULTIPLE
                   SIGNIFICANT TRAUMA */
    '486' = '1' /* OTHER OR PROCEDURES FOR MULTIPLE SIGNIFICANT TRAUMA */
    '487' = '1' /* OTHER MULTIPLE SIGNIFICANT TRAUMAS */
    '491' = '1' /* MAJOR JOINT AND LIMB REATTACHMENT PROCEDURES OF UPPER
                   EXTREMITY */
    OTHER = '0' ;

    /* Vaginal delivery */
    /* Diagnostic Related Groups (DRGs): */

    VALUE $VAGINDR
    '372' = '1' /* VAGINAL DELIVERY WITH COMPLICATING DIAGNOSES */
    '373' = '1' /* VAGINAL DELIVERY WITHOUT COMPLICATING DIAGNOSES */
    '374' = '1' /* VAGINAL DELIVERY WITH STERILIZATION AND/OR D AND C */
    '375' = '1' /* VAGINAL DELIVERY WITH OR PROCEDURE EXCEPT STERILIZATION
                   ANDOR D AND C */
    OTHER = '0' ;

    /* Interuption of vena cava */;
    /* ICD-9-CM procedure code: */;

    PSFMTS -- CONTINUED

    VALUE $VENACIP
    '387'  = '1' /* INTERUPTION OF VENA CAVA */
    OTHER  = '0' ;
    /* FTR-FAILURE TO RESCUE */

    /* FTR-Acute Renal Failure */
    /* ICD-9-CM diagnosis codes (all 4th and 5th digits included): */

    VALUE $FTR1DX
    '584'   = '1' /* ACUTE RENAL FAILURE */
```

```
    '5845'  = '1' /* ACUTE RENAL FAILURE - W LESION OF TUBULAR NECROSIS */
    '5846'  = '1' /* ACUTE RENAL FAILURE - W LESION OF RENAL CORTICAL NECROSIS
*/
    '5847'   = '1'  /*  ACUTE  RENAL  FAILURE  -  W  LESION  OF  RENAL  MEDULLARY
NECROSIS */
    '5848'   = '1'  /*  ACUTE  RENAL  FAILURE  -  W  OTHER  SPECIFIED  PATHOLOGICAL
LESION */
    '5849'  = '1' /* ACUTE RENAL FAILURE - UNSPECIFIED */
    OTHER   = '0' ;

    /* FTR-DVT PE */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FTR2DX
    '41511' = '1' /* IATROGENIC PULMONARY EMBOLISM AND INFARCTION */
    '41519' = '1' /* OTHER */
    '45111' = '1' /* FEMORAL VEIN (DEEP) (SUPERFICIAL) */
    '45119' = '1' /* OTHER */
    '4512'  = '1' /* LOWER EXTREMITIES */
    '45181' = '1' /* ILIAC VEIN */
    '4519'  = '1' /* UNSPECIFIED SITE */
    '4538'  = '1' /* OTHER VENOUS EMBOLISM AND THROMBOSIS OF OTHER SPECIFIED
                     VEINS */
    '4539'  = '1' /* OTHER VENOUS EMBOLISM AND THROMBOSIS OF UNSPECIFIED SITE
*/
    OTHER   = '0' ;

    /* FTR-Pneumonia */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FTR3DX
    '482'   = '1' /* OTHER BACTERIAL PNEUMONIA */
    '4820'  = '1' /* K. PNEUMONIAE PNEUMONIA */
    '4821'  = '1' /* PSEUDOMONAL PNEUMONIA */
    '4822'  = '1' /* H.INFLUENZAE PNEUMONIA */
    '4823'  = '1' /* STREPTOCOCCAL PNEUMONIA */
    '48230' = '1' /* STREPTOCOCCAL PNEUMN NOS */
    '48231' = '1' /* PNEUMONIA STRPTOCOCCUS A */
    '48232' = '1' /* PNEUMONIA STRPTOCOCCUS B */
    '48239' = '1' /* PNEUMONIA OTH STREP */
    '4824'  = '1' /* STAPHYLOCOCCAL PNEUMONIA */
    '48240' = '1' /* STAPHYLOCOCCAL PNEU NOS */
    '48241' = '1' /* STAPH AUREUS PNEUMONIA */
    '48249' = '1' /* STAPH PNEUMONIA NEC */
    '4828'  = '1' /* BACTERIAL PNEUMONIA NEC */
    '48281' = '1' /* PNEUMONIA ANAEROBES */
    '48282' = '1' /* PNEUMONIA E COLI */
    '48283' = '1' /* PNEUMO OTH GRM-NEG BACT */
    PSFMTS -- CONTINUED

    '48284' = '1' /* LEGIONNAIRES  DISEASE */
    '48289' = '1' /* PNEUMONIA OTH SPCF BACT */
    '4829'  = '1' /* BACTERIAL PNEUMONIA NOS */
    '485'   = '1' /* BRONCHOPNEUMONIA, ORGANISM UNSPECIFIED */
    '486'   = '1' /* PNEUMONIA, ORGANISM UNSPECIFIED */
    '5070'  = '1' /* DUE TO INHALATION OF FOOD OR VOMITUS */
    '514'   = '1' /* PULMONARY CONGESTION AND HYPOSTASIS */
    OTHER   = '0' ;
```

```
    /* Exclude */
    /* CD-9-CM diagnosis codes: */

    VALUE $FTR3EXA
    '480'   = '1' /* VIRAL PNEUMONIA */
    '4800'  = '1' /* VIRAL PNEUMONIA */
    '4801'  = '1' /* VIRAL PNEUMONIA */
    '4802'  = '1' /* VIRAL PNEUMONIA */
    '4808'  = '1' /* VIRAL PNEUMONIA */
    '4809'  = '1' /* VIRAL PNEUMONIA */
    '481'      =  '1'  /*  PNEUMOCOCCAL  PNEUMONIA  [STREPTOCOCCUS  PNEUMONIAE
PNEUMONIA] */
    '482'   = '1' /* OTHER BACTERIAL PNEUMONIA */
    '4820'  = '1' /* OTHER BACTERIAL PNEUMONIA */
    '4821'  = '1' /* OTHER BACTERIAL PNEUMONIA */
    '4822'  = '1' /* OTHER BACTERIAL PNEUMONIA */
    '4823'  = '1' /* OTHER BACTERIAL PNEUMONIA */
    '48230' = '1' /* OTHER BACTERIAL PNEUMONIA */
    '48231' = '1' /* OTHER BACTERIAL PNEUMONIA */
    '48232' = '1' /* OTHER BACTERIAL PNEUMONIA */
    '48239' = '1' /* OTHER BACTERIAL PNEUMONIA */
    '483'   = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '4830'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '4831'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '4838'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '484'   = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4841'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4843'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4845'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4846'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4847'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4848'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '485'   = '1' /* BRONCHOPNEUMONIA, ORGANISM UNSPECIFIED */
    '486'   = '1' /* PNEUMONIA, ORGANISM UNSPECIFIED */
    '487'   = '1' /* INFLUENZA */
    '4870'  = '1' /* INFLUENZA */
    '4871'  = '1' /* INFLUENZA */
    '4878'  = '1' /* INFLUENZA */
    '5070'  = '1' /* DUE TO INHALATION OF FOOD OR VOMITUS */
    '514'   = '1' /* PULMONARY CONGESTION AND HYPOSTASIS */
    '9973'  = '1' /* RESPIRATORY COMPLICATIONS */
    OTHER   = '0' ;

    VALUE $FTR3EXB
    '480'   = '1' /* VIRAL PNEUMONIA */
    '4800'  = '1' /* VIRAL PNEUMONIA */
    '4801'  = '1' /* VIRAL PNEUMONIA */
    '4802'  = '1' /* VIRAL PNEUMONIA */
    '4808'  = '1' /* VIRAL PNEUMONIA */
    '4809'  = '1' /* VIRAL PNEUMONIA */
    PSFMTS -- CONTINUED

    '481'      =  '1'  /*  PNEUMOCOCCAL  PNEUMONIA  [STREPTOCOCCUS  PNEUMONIAE
PNEUMONIA] */
    '483'   = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '4830'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '4831'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
```

```
    '4838'  = '1' /* PNEUMONIA DUE TO OTHER SPECIFIED ORGANISM */
    '484'   = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4841'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4843'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4845'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4846'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4847'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '4848'  = '1' /* PNEUMONIA IN INFECTIOUS DISEASES CLASSIFIED ELSEWHERE */
    '487'   = '1' /* INFLUENZA */
    '4870'  = '1' /* INFLUENZA */
    '4871'  = '1' /* INFLUENZA */
    '4878'  = '1' /* INFLUENZA */
    OTHER   = '0' ;

    /* FTR-Sepsis */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FTR4DX
    '038'   = '1' /* SEPTICEMIA */
    '0380'  = '1' /* STREPTOCOCCAL SEPTICEMIA */
    '0381 ' = '1' /* STAPHYLOCOCCAL SEPTICEMIA */
    '03810' = '1' /* STAPHYLOCOCCAL SEPTICEMIA, UNSPECIFIED */
    '03811' = '1' /* STAPHYLOCOCCUS AUREAU SEPTICEMIA */
    '03819' = '1' /* OTHER STAPHYLOCOCCAL SEPTICEMIA */
    '0382'    =  '1'  /*  PNEUMOCOCCAL  SEPTICEMIA  [STREPTOCOCCUS  PNEUMONIAE
SEPTICEMIA] *
    /
    '0383'  = '1' /* SEPTICEMIA DUE TO ANAEROBES */
    '03840' = '1' /* SEPTICEMIA DUE TO GRAM NEGATIVE ORGANISM, UNSPECIFIED */
    '03841' = '1' /* HEMOPHILUS INFLUENZE [H. INFLUENZAE] */
    '03842' = '1' /* ESCHERICHIA COLI [E COLI] */
    '03843' = '1' /* PSEUDOMONAS */
    '03844' = '1' /* SERRATIA */
    '03849' = '1' /* OTHER */
    '0388'  = '1' /* OTHER SPECIFIED SEPTICEMIAS */
    '0389'  = '1' /* UNSPECIFIED SEPTICEMIA */
    '7907'  = '1' /* OTHER NONSPESIFIC FINDINGS ON EXAMINATION OF BLOOD */
    OTHER   = '0' ;

    /* FTR-Shock or cardiac arrest */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FTR5DX
    '4275'  = '1' /* CARDIAC ARREST */
    '6691'  = '1' /* SHOCK DURING OR FOLLOWING LABOR AND DELIVERY */
    '66910' = '1' /* OBSTETRIC SHOCK-UNSPEC */
    '66911' = '1' /* OBSTETRIC SHOCK-DELIVER */
    '66912' = '1' /* OBSTET SHOCK-DELIV W P/P */
    '66913' = '1' /* OBSTETRIC SHOCK-ANTEPAR */
    '66914' = '1' /* OBSTETRIC SHOCK-POSTPART */
    '7855'  = '1' /* SHOCK WITHOUT MENTION OF TRAUMA */
    '78550' = '1' /* SHOCK, UNSPECIFIED */
    '78551' = '1' /* CARDIOGENIC SHOCK */
    '78559' = '1' /* SHOCK WITHOUT MENTION OF TRAUMA- OTHER */
    PSFMTS -- CONTINUED

    '7991'  = '1' /* RESPIRATORY ARREST */
    '9950'  = '1' /* OTHER ANAPHYLACTIC SHOCK */
```

```
    '9954'  = '1' /* SHOCK DUE TO ANESTHESIA */
    '9980'  = '1' /* POSTOPERATIVE SHOCK */
    '9994'  = '1' /* ANAPHYLACTIC SHOCK DUE TO SERUM */
    OTHER   = '0' ;

    /* ICD-9-CM procedure codes: */

    VALUE $FTR5PR
    '9393' = '1' /* NONMECHANICAL METHODS OF RESUSCITATION */
    '9960' = '1' /* CARDIOPULMONARY RESUSCITATION, NOS */
    '9963' = '1' /* CLOSED CHEST CARDIAC MASSAGE */
    OTHER  = '0' ;

    /* FTR-GI hemorrhage-acute ulcer */
    /* ICD-9-CM diagnosis codes: */

    VALUE $FTR6DX
    '4560'  = '1' /* ESOPHAGEAL VARICES WITH BLEEDING */
    '53130' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION –
                     WITHOUT MENTION OF OBSTRUCTION */
    '53131' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION - WITH
                     OBSTRUCTION */
    '53190' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
                     HEMORRHAGE   OR   PERFORATION   -   WITHOUT   MENTION   OF
OBSTRUCTION */
    '53191' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
                     HEMORRHAGE OR PERFORATION - WITH OBSTRUCTION */
    '53230' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION -
WTHOUT
                     MENTION OF OBSTRUCTION */
    '53231' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION - WITH
                     OBSTRUCTION */
    '53290' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
                     HEMORRHAGE   OR   PERFORATION   -   WITHOUT   MENTION   OF
OBSTRUCTION */
    '53291' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
                     HEMORRHAGE OR PERFORATION - WITH OBSTRUCTION */
    '53330' = '1' /* SITE UNSPECIFIED ACUTE WITHOUT MENTION OF HEMORRHAGE AND
                     PERFFORATION - WITHOUT MENTION OF OBSTRUCTION */
    '53331' = '1' /* SITE UNSPECIFIED ACUTE WITHOUT MENTION OF HEMORRHAGE AND
                     PERFORATION - WITH OBSTRUCTION */
    '53390' = '1' /* SITE UNSPECIFIED UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT
                     MENTION OF HEMORRHAGE OR PERFORATION - WITHOUT MENTION OF
OBSTR
    UCTION */
    '53391' = '1' /* SITE UNSPECIFIED UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT
                      MENTION OF HEMORRHAGE OR PERFORATION - WITH OBSTRUCTION
*/
    '53430' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION -
WITHOUT
                     MENTION OF OBSTRUCTION */
    '53431' = '1' /* ACUTE WITHOUT MENTION OF HEMORRHAGE OR PERFORATION - WITH
                     OBSTRUCTION */
    '53490' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
HEMORRHAGE
                     OR PERFORATION - WITHOUT MENTION OF OBSTRUCTION */
```

PSFMTS -- CONTINUED

```
    '53491' = '1' /* UNSPECIFIED AS ACUTE OR CHRONIC, WITHOUT MENTION OF
HEMORRHAGE
                     OR PERFORATION - WITH OBSTRUCTION */
    '5307'  = '1' /* GASTROESOPHAGEAL LACERATION-HEMORRHAGE SYNDROME */
    '53082' = '1' /* ESOPHAGEAL HEMORRHAGE */
    '53100' = '1' /* ACUTE WITH HEMORRHAGE - WITHOUT MENTION OF OBSTRUCTION */
    '53101' = '1' /* ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION */
    '53110' = '1' /* ACUTE WITH PERFORATION  - WITHOUT MENTION OF OBSTRUCTION
*/
    '53111' = '1' /* ACUTE WITH PERFORATION - WITH OBSTRUCTION */
    '53120' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION  - WITHOUT MENTION
OF
                     OBSTRUCTION */
    '53121' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION - WITH OBSTRUCTION
*/
    '53200' = '1' /* ACUTE WITH HEMORRHAGE - WITHOUT MENTION OF OBSTRUCTION */
    '53201' = '1' /* ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION */
    '53210' = '1' /* ACUTE WITH PERFORATION  - WITHOUT MENTION OF OBSTRUCTION
*/
    '53211' = '1' /* ACUTE WITH PERFORATION - WITH OBSTRUCTION */
    '53220' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION  - WITHOUT MENTION
OF
                     OBSTRUCTION */
    '53221' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION - WITH OBSTRUCTION
*/
    '53300' = '1' /* SITE UNSPECIFIED ACUTE WITH HEMORRHAGE - WITHOUT MENTION
OF
                     OBSTRUCTION */
    '53301' = '1' /* SITE UNSPECIFIED ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION
*/
    '53310' = '1' /* SITE UNSPECIFIED ACUTE WITH PERFORATION - WITHOUT MENTION
OF
                     OBSTRUCTION */
    '53311' = '1' /* SITE UNSPECIFIED ACUTE WITH -PERFORATION WITH OBSTRUCTION
*/
    '53320' = '1' /* SITE UNSPECIFIED ACTE WITH HEMORRHAGE AND PERFORATION -
                     WITHOUT MENTION OF OBSTRUCTION */
    '53321' = '1' /* SITE UNSPECIFIED ACUTE WITH HEMORRHAGE AND PERFORATION -
                     WITHOUT MENTION OF OBSTRUCTION */
    '53400' = '1' /* ACUTE WITH HEMORRHAGE -WITHOUT MENTION OF OBSTRUCTION */
    '53401' = '1' /* ACUTE WITH HEMORRHAGE - WITH OBSTRUCTION */
    '53410' = '1' /* ACUTE WITH PERFORATION  - WITHOUT MENTION OF OBSTRUCTION
*/
    '53411' = '1' /* ACUTE WITH PERFORATION – WITH OBSTRUCTION */
    '53420' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION  - WITHOUT MENTION
OF
                     OBSTRUCTION */
    '53421' = '1' /* ACUTE WITH HEMORRHAGE AND PERFORATION - WITH OBSTRUCTION
*/
    '53501' = '1' /* ACUTE GASTRITIS - WITH HEMORRHAGE */
    '53511' = '1' /* ATROPHIC GASTRITIS - WITH HEMORRHAGE */
    '53521' = '1' /* GASTRIC MUCOSAL HYPERTROPHY -WITH HEMORRHAGE */
    '53531' = '1' /* ALCOHOLIC GASTRITIS - WITH HEMORRHAGE */
    '53541' = '1' /* OTHER SPECIFIED GASTRITIS - WITH HEMORRHAGE */
    '53551'  =  '1'  /*  UNSPECIFIED  GASTRITIS  AND  GASTRODUODENITIS  -  WITH
HEMORRHAGE */
    '53561' = '1' /* DUODENITIS - WITH HEMORRHAGE */
    '53783' = '1' /* ANGIODYSPLASIA OF STOMACH AND DUODENUM WITH HEMORRHAGE */
```

```
'54620' = '1' /* ESOPHAGEAL VARICES IN DISEASES CLASSIFIED ELSEWHERE WITH
                 BLEEDING */
```
PSFMTS -- CONTINUED

```
'56202' = '1' /* DIVERTICULOSIS OF SMALL INTESTINE WITH HEMORRHAGE */
'56203' = '1' /* DIVERTICULITIS OF SMALL INTESTINE WITH HEMORRHAGE */
'56212' = '1' /* DIVERTICULOSIS OF COLON WITH HEMORRHAGE */
'56213' = '1' /* DIVERICULITIS OF COLON WITH HEMORRHAGE */
'5693'  = '1' /* HEMORRHAGE OF RECTUM AND ANUS */
'56985' = '1' /* ANGIODYSPLASIA OF INTESTINE WITH HEMORRHAGE */
'5780'  = '1' /* HEMATEMESIS */
'5781'  = '1' /* BLOOD IN STOOL */
'5789'  = '1' /* HEMORRHAGE OF GASTROINTESTINAL TRACT, UNSPECIFIED */
OTHER   = '0' ;

/* Exclude */
/* ICD-9-CM diagnosis codes: */

VALUE $FTR6EX
'2800' = '1' /* SECONDARY TO BLOOD LOSS [CHRONIC] */
'2851' = '1' /* ACUTE POSTHEMORRHAGIC ANEMIA */
OTHER  = '0' ;

 * ------------------------------------------------- ;
 * ---   AREA FORMATS     ------------------------- ;
 * ---   Translates FIPS State and County Code (SSCCC)   ;
 * ---   to MSA (Metropolitan Statistical Area)          ;
 * ------------------------------------------------- ;

VALUE AREA
1000  = '01000'
1001  = '5240 '
1003  = '5160 '
1005  = '01005'
1007  = '01007'
1009  = '1000 '
1011  = '01011'
1013  = '01013'
1015  = '0450 '
1017  = '01017'
1019  = '01019'
1021  = '01021'
1023  = '01023'
1025  = '01025'
1027  = '01027'
1029  = '01029'
1031  = '01031'
1033  = '2650 '
1035  = '01035'
1037  = '01037'
1039  = '01039'
1041  = '01041'
1043  = '01043'
1045  = '2180 '
1047  = '01047'
1049  = '01049'
1051  = '5240 '
1053  = '01053'
```

```
1055  = '2880 '
1057  = '01057'
1059  = '01059'
1061  = '01061'
1063  = '01063'
```
PSFMTS -- CONTINUED

```
1065  = '01065'
1067  = '01067'
1069  = '2180 '
1071  = '01071'
1073  = '1000 '
1075  = '01075'
1077  = '2650 '
1079  = '2030 '
1081  = '0580 '
1083  = '3440 '
1085  = '01085'
1087  = '01087'
1089  = '3440 '
1091  = '01091'
1093  = '01093'
1095  = '01095'
1097  = '5160 '
1099  = '01099'
1101  = '5240 '
1103  = '2030 '
1105  = '01105'
1107  = '01107'
1109  = '01109'
1111  = '01111'
1113  = '1800 '
1115  = '1000 '
1117  = '1000 '
1119  = '01119'
1121  = '01121'
1123  = '01123'
1125  = '8600 '
1127  = '01127'
1129  = '01129'
1131  = '01131'
1133  = '01133'
2000  = '02000'
2013  = '02013'
2016  = '02016'
2020  = '0380 '
2050  = '02050'
2060  = '02060'
2068  = '02068'
2070  = '02070'
2090  = '02090'
2100  = '02100'
2110  = '02110'
2122  = '02122'
2130  = '02130'
2150  = '02150'
2164  = '02164'
2170  = '02170'
```

```
2180  = '02180'
2185  = '02185'
2188  = '02188'
2201  = '02201'
2220  = '02220'
2231  = '02231'
2232  = '02232'
2240  = '02240'
```
PSFMTS -- CONTINUED

```
2261  = '02261'
2270  = '02270'
2280  = '02280'
2282  = '02282'
2290  = '02290'
4000  = '04000'
4001  = '04001'
4003  = '04003'
4005  = '2620 '
4007  = '04007'
4009  = '04009'
4011  = '04011'
4012  = '04012'
4013  = '6200 '
4015  = '4120 '
4017  = '04017'
4019  = '8520 '
4021  = '6200 '
4023  = '04023'
4025  = '04025'
4027  = '9360 '
5000  = '05000'
5001  = '05001'
5003  = '05003'
5005  = '05005'
5007  = '2580 '
5009  = '05009'
5011  = '05011'
5013  = '05013'
5015  = '05015'
5017  = '05017'
5019  = '05019'
5021  = '05021'
5023  = '05023'
5025  = '05025'
5027  = '05027'
5029  = '05029'
5031  = '3700 '
5033  = '2720 '
5035  = '4920 '
5037  = '05037'
5039  = '05039'
5041  = '05041'
5043  = '05043'
5045  = '4400 '
5047  = '05047'
5049  = '05049'
5051  = '05051'
```

```
5053  = '05053'
5055  = '05055'
5057  = '05057'
5059  = '05059'
5061  = '05061'
5063  = '05063'
5065  = '05065'
5067  = '05067'
5069  = '6240 '
5071  = '05071'
5073  = '05073'
```
PSFMTS -- CONTINUED

```
5075  = '05075'
5077  = '05077'
5079  = '05079'
5081  = '05081'
5083  = '05083'
5085  = '4400 '
5087  = '05087'
5089  = '05089'
5091  = '8360 '
5093  = '05093'
5095  = '05095'
5097  = '05097'
5099  = '05099'
5101  = '05101'
5103  = '05103'
5105  = '05105'
5107  = '05107'
5109  = '05109'
5111  = '05111'
5113  = '05113'
5115  = '05115'
5117  = '05117'
5119  = '4400 '
5121  = '05121'
5123  = '05123'
5125  = '4400 '
5127  = '05127'
5129  = '05129'
5131  = '2720 '
5133  = '05133'
5135  = '05135'
5137  = '05137'
5139  = '05139'
5141  = '05141'
5143  = '2580 '
5145  = '05145'
5147  = '05147'
5149  = '05149'
6000  = '06000'
6001  = '7362 '
6003  = '06003'
6005  = '06005'
6007  = '1620 '
6009  = '06009'
6011  = '06011'
```

```
6013  = '7362 '
6015  = '06015'
6017  = '6922 '
6019  = '2840 '
6021  = '06021'
6023  = '06023'
6025  = '06025'
6027  = '06027'
6029  = '0680 '
6031  = '06031'
6033  = '06033'
6035  = '06035'
6037  = '4472 '
6039  = '2840 '
```
PSFMTS -- CONTINUED

```
6041  = '7362 '
6043  = '06043'
6045  = '06045'
6047  = '4940 '
6049  = '06049'
6051  = '06051'
6053  = '7120 '
6055  = '7362 '
6057  = '06057'
6059  = '4472 '
6061  = '6922 '
6063  = '06063'
6065  = '4472 '
6067  = '6922 '
6069  = '06069'
6071  = '4472 '
6073  = '7320 '
6075  = '7362 '
6077  = '8120 '
6079  = '7460 '
6081  = '7362 '
6083  = '7480 '
6085  = '7362 '
6087  = '7362 '
6089  = '6690 '
6091  = '06091'
6093  = '06093'
6095  = '7362 '
6097  = '7362 '
6099  = '5170 '
6101  = '9340 '
6103  = '06103'
6105  = '06105'
6107  = '8780 '
6109  = '06109'
6111  = '4472 '
6113  = '6922 '
6115  = '9340 '
8000  = '08000'
8001  = '2082 '
8003  = '08003'
8005  = '2082 '
```

```
8007  = '08007'
8009  = '08009'
8011  = '08011'
8013  = '2082 '
8015  = '08015'
8017  = '08017'
8019  = '08019'
8021  = '08021'
8023  = '08023'
8025  = '08025'
8027  = '08027'
8029  = '08029'
8031  = '2082 '
8033  = '08033'
8035  = '2082 '
8037  = '08037'
8039  = '08039'
```

PSFMTS -- CONTINUED

```
8041  = '1720 '
8043  = '08043'
8045  = '08045'
8047  = '08047'
8049  = '08049'
8051  = '08051'
8053  = '08053'
8055  = '08055'
8057  = '08057'
8059  = '2082 '
8061  = '08061'
8063  = '08063'
8065  = '08065'
8067  = '08067'
8069  = '2670 '
8071  = '08071'
8073  = '08073'
8075  = '08075'
8077  = '2995 '
8079  = '08079'
8081  = '08081'
8083  = '08083'
8085  = '08085'
8087  = '08087'
8089  = '08089'
8091  = '08091'
8093  = '08093'
8095  = '08095'
8097  = '08097'
8099  = '08099'
8101  = '6560 '
8103  = '08103'
8105  = '08105'
8107  = '08107'
8109  = '08109'
8111  = '08111'
8113  = '08113'
8115  = '08115'
8117  = '08117'
```

```
8119  = '08119'
8121  = '08121'
8123  = '2082 '
8125  = '08125'
9000  = '09000'
9001  = '5483 '
9003  = '3283 '
9005  = '09005'
9007  = '3283 '
9009  = '5483 '
9011  = '5523 '
9013  = '3283 '
9015  = '09015'
10000 = '10000'
10001 = '2190 '
10003 = '6162 '
10005 = '10005'
11000 = '11000'
11001 = '8872 '
12000 = '12000'
```
PSFMTS -- CONTINUED

```
12001 = '2900 '
12003 = '12003'
12005 = '6015 '
12007 = '12007'
12009 = '4900 '
12011 = '4992 '
12013 = '12013'
12015 = '6580 '
12017 = '12017'
12019 = '3600 '
12021 = '5345 '
12023 = '12023'
12025 = '4992 '
12027 = '12027'
12029 = '12029'
12031 = '3600 '
12033 = '6080 '
12035 = '2020 '
12037 = '12037'
12039 = '8240 '
12041 = '12041'
12043 = '12043'
12045 = '12045'
12047 = '12047'
12049 = '12049'
12051 = '12051'
12053 = '8280 '
12055 = '12055'
12057 = '8280 '
12059 = '12059'
12061 = '12061'
12063 = '12063'
12065 = '12065'
12067 = '12067'
12069 = '5960 '
12071 = '2700 '
```

```
12073 = '8240 '
12075 = '12075'
12077 = '12077'
12079 = '12079'
12081 = '7510 '
12083 = '5790 '
12085 = '2710 '
12086 = '4992 '
12087 = '12087'
12089 = '3600 '
12091 = '2750 '
12093 = '12093'
12095 = '5960 '
12097 = '5960 '
12099 = '8960 '
12101 = '8280 '
12103 = '8280 '
12105 = '3980 '
12107 = '12107'
12109 = '3600 '
12111 = '2710 '
12113 = '6080 '
12115 = '7510 '
```
PSFMTS -- CONTINUED

```
12117 = '5960 '
12119 = '12119'
12121 = '12121'
12123 = '12123'
12125 = '12125'
12127 = '2020 '
12129 = '12129'
12131 = '12131'
12133 = '12133'
13000 = '13000'
13001 = '13001'
13003 = '13003'
13005 = '13005'
13007 = '13007'
13009 = '13009'
13011 = '13011'
13013 = '0520 '
13015 = '0520 '
13017 = '13017'
13019 = '13019'
13021 = '4680 '
13023 = '13023'
13025 = '13025'
13027 = '13027'
13029 = '7520 '
13031 = '13031'
13033 = '13033'
13035 = '13035'
13037 = '13037'
13039 = '13039'
13043 = '13043'
13045 = '0520 '
13047 = '1560 '
```

```
13049 = '13049'
13051 = '7520 '
13053 = '1800 '
13055 = '13055'
13057 = '0520 '
13059 = '0500 '
13061 = '13061'
13063 = '0520 '
13065 = '13065'
13067 = '0520 '
13069 = '13069'
13071 = '13071'
13073 = '0600 '
13075 = '13075'
13077 = '0520 '
13079 = '13079'
13081 = '13081'
13083 = '1560 '
13085 = '13085'
13087 = '13087'
13089 = '0520 '
13091 = '13091'
13093 = '13093'
13095 = '0120 '
13097 = '0520 '
13099 = '13099'
```
PSFMTS -- CONTINUED

```
13101 = '13101'
13103 = '7520 '
13105 = '13105'
13107 = '13107'
13109 = '13109'
13111 = '13111'
13113 = '0520 '
13115 = '13115'
13117 = '0520 '
13119 = '13119'
13121 = '0520 '
13123 = '13123'
13125 = '13125'
13127 = '13127'
13129 = '13129'
13131 = '13131'
13133 = '13133'
13135 = '0520 '
13137 = '13137'
13139 = '13139'
13141 = '13141'
13143 = '13143'
13145 = '1800 '
13147 = '13147'
13149 = '13149'
13151 = '0520 '
13153 = '4680 '
13155 = '13155'
13157 = '13157'
13159 = '13159'
```

```
13161 = '13161'
13163 = '13163'
13165 = '13165'
13167 = '13167'
13169 = '4680 '
13171 = '13171'
13173 = '13173'
13175 = '13175'
13177 = '0120 '
13179 = '13179'
13181 = '13181'
13183 = '13183'
13185 = '13185'
13187 = '13187'
13189 = '0600 '
13191 = '13191'
13193 = '13193'
13195 = '0500 '
13197 = '13197'
13199 = '13199'
13201 = '13201'
13205 = '13205'
13207 = '13207'
13209 = '13209'
13211 = '13211'
13213 = '13213'
13215 = '1800 '
13217 = '0520 '
13219 = '0500 '
```

PSFMTS -- CONTINUED

```
13221 = '13221'
13223 = '0520 '
13225 = '4680 '
13227 = '0520 '
13229 = '13229'
13231 = '13231'
13233 = '13233'
13235 = '13235'
13237 = '13237'
13239 = '13239'
13241 = '13241'
13243 = '13243'
13245 = '0600 '
13247 = '0520 '
13249 = '13249'
13251 = '13251'
13253 = '13253'
13255 = '0520 '
13257 = '13257'
13259 = '13259'
13261 = '13261'
13263 = '13263'
13265 = '13265'
13267 = '13267'
13269 = '13269'
13271 = '13271'
13273 = '13273'
```

```
13275 = '13275'
13277 = '13277'
13279 = '13279'
13281 = '13281'
13283 = '13283'
13285 = '13285'
13287 = '13287'
13289 = '4680 '
13291 = '13291'
13293 = '13293'
13295 = '1560 '
13297 = '0520 '
13299 = '13299'
13301 = '13301'
13303 = '13303'
13305 = '13305'
13307 = '13307'
13309 = '13309'
13311 = '13311'
13313 = '13313'
13315 = '13315'
13317 = '13317'
13319 = '13319'
13321 = '13321'
15000 = '15000'
15001 = '15001'
15003 = '3320 '
15005 = '15005'
15007 = '15007'
15009 = '15009'
16000 = '16000'
16001 = '1080 '
```
PSFMTS -- CONTINUED

```
16003 = '16003'
16005 = '6340 '
16007 = '16007'
16009 = '16009'
16011 = '16011'
16013 = '16013'
16015 = '16015'
16017 = '16017'
16019 = '16019'
16021 = '16021'
16023 = '16023'
16025 = '16025'
16027 = '1080 '
16029 = '16029'
16031 = '16031'
16033 = '16033'
16035 = '16035'
16037 = '16037'
16039 = '16039'
16041 = '16041'
16043 = '16043'
16045 = '16045'
16047 = '16047'
16049 = '16049'
```

```
16051 = '16051'
16053 = '16053'
16055 = '16055'
16057 = '16057'
16059 = '16059'
16061 = '16061'
16063 = '16063'
16065 = '16065'
16067 = '16067'
16069 = '16069'
16071 = '16071'
16073 = '16073'
16075 = '16075'
16077 = '16077'
16079 = '16079'
16081 = '16081'
16083 = '16083'
16085 = '16085'
16087 = '16087'
17000 = '17000'
17001 = '17001'
17003 = '17003'
17005 = '17005'
17007 = '6880 '
17009 = '17009'
17011 = '17011'
17013 = '17013'
17015 = '17015'
17017 = '17017'
17019 = '1400 '
17021 = '17021'
17023 = '17023'
17025 = '17025'
17027 = '7040 '
17029 = '17029'
```
PSFMTS -- CONTINUED

```
17031 = '1602 '
17033 = '17033'
17035 = '17035'
17037 = '1602 '
17039 = '17039'
17041 = '17041'
17043 = '1602 '
17045 = '17045'
17047 = '17047'
17049 = '17049'
17051 = '17051'
17053 = '17053'
17055 = '17055'
17057 = '17057'
17059 = '17059'
17061 = '17061'
17063 = '1602 '
17065 = '17065'
17067 = '17067'
17069 = '17069'
17071 = '17071'
```

```
17073 = '1960 '
17075 = '17075'
17077 = '17077'
17079 = '17079'
17081 = '17081'
17083 = '7040 '
17085 = '17085'
17087 = '17087'
17089 = '1602 '
17091 = '1602 '
17093 = '1602 '
17095 = '17095'
17097 = '1602 '
17099 = '17099'
17101 = '17101'
17103 = '17103'
17105 = '17105'
17107 = '17107'
17109 = '17109'
17111 = '1602 '
17113 = '1040 '
17115 = '2040 '
17117 = '17117'
17119 = '7040 '
17121 = '17121'
17123 = '17123'
17125 = '17125'
17127 = '17127'
17129 = '7880 '
17131 = '17131'
17133 = '7040 '
17135 = '17135'
17137 = '17137'
17139 = '17139'
17141 = '6880 '
17143 = '6120 '
17145 = '17145'
17147 = '17147'
```
PSFMTS -- CONTINUED

```
17149 = '17149'
17151 = '17151'
17153 = '17153'
17155 = '17155'
17157 = '17157'
17159 = '17159'
17161 = '1960 '
17163 = '7040 '
17165 = '17165'
17167 = '7880 '
17169 = '17169'
17171 = '17171'
17173 = '17173'
17175 = '17175'
17177 = '17177'
17179 = '6120 '
17181 = '17181'
17183 = '17183'
```

```
17185 = '17185'
17187 = '17187'
17189 = '17189'
17191 = '17191'
17193 = '17193'
17195 = '17195'
17197 = '1602 '
17199 = '17199'
17201 = '6880 '
17203 = '6120 '
18000 = '18000'
18001 = '2760 '
18003 = '2760 '
18005 = '18005'
18007 = '18007'
18009 = '18009'
18011 = '3480 '
18013 = '18013'
18015 = '18015'
18017 = '18017'
18019 = '4520 '
18021 = '8320 '
18023 = '3920 '
18025 = '18025'
18027 = '18027'
18029 = '1642 '
18031 = '18031'
18033 = '2760 '
18035 = '5280 '
18037 = '18037'
18039 = '2330 '
18041 = '18041'
18043 = '4520 '
18045 = '18045'
18047 = '18047'
18049 = '18049'
18051 = '18051'
18053 = '18053'
18055 = '18055'
18057 = '3480 '
18059 = '3480 '
```
PSFMTS -- CONTINUED

```
18061 = '4520 '
18063 = '3480 '
18065 = '18065'
18067 = '3850 '
18069 = '2760 '
18071 = '18071'
18073 = '18073'
18075 = '18075'
18077 = '18077'
18079 = '18079'
18081 = '3480 '
18083 = '18083'
18085 = '18085'
18087 = '18087'
18089 = '1602 '
```

18091 = '18091'
18093 = '18093'
18095 = '3480 '
18097 = '3480 '
18099 = '18099'
18101 = '18101'
18103 = '18103'
18105 = '1020 '
18107 = '18107'
18109 = '3480 '
18111 = '18111'
18113 = '18113'
18115 = '1642 '
18117 = '18117'
18119 = '18119'
18121 = '18121'
18123 = '18123'
18125 = '18125'
18127 = '1602 '
18129 = '2440 '
18131 = '18131'
18133 = '18133'
18135 = '18135'
18137 = '18137'
18139 = '18139'
18141 = '7800 '
18143 = '4520 '
18145 = '3480 '
18147 = '18147'
18149 = '18149'
18151 = '18151'
18153 = '18153'
18155 = '18155'
18157 = '3920 '
18159 = '3850 '
18161 = '18161'
18163 = '2440 '
18165 = '8320 '
18167 = '8320 '
18169 = '18169'
18171 = '18171'
18173 = '2440 '
18175 = '18175'
18177 = '18177'
PSFMTS -- CONTINUED

18179 = '2760 '
18181 = '18181'
18183 = '2760 '
19000 = '19000'
19001 = '19001'
19003 = '19003'
19005 = '19005'
19007 = '19007'
19009 = '19009'
19011 = '19011'
19013 = '8920 '
19015 = '19015'

```
19017 = '19017'
19019 = '19019'
19021 = '19021'
19023 = '19023'
19025 = '19025'
19027 = '19027'
19029 = '19029'
19031 = '19031'
19033 = '19033'
19035 = '19035'
19037 = '19037'
19039 = '19039'
19041 = '19041'
19043 = '19043'
19045 = '19045'
19047 = '19047'
19049 = '2120 '
19051 = '19051'
19053 = '19053'
19055 = '19055'
19057 = '19057'
19059 = '19059'
19061 = '2200 '
19063 = '19063'
19065 = '19065'
19067 = '19067'
19069 = '19069'
19071 = '19071'
19073 = '19073'
19075 = '19075'
19077 = '19077'
19079 = '19079'
19081 = '19081'
19083 = '19083'
19085 = '19085'
19087 = '19087'
19089 = '19089'
19091 = '19091'
19093 = '19093'
19095 = '19095'
19097 = '19097'
19099 = '19099'
19101 = '19101'
19103 = '3500 '
19105 = '19105'
19107 = '19107'
19109 = '19109'
```
PSFMTS -- CONTINUED

```
19111 = '19111'
19113 = '1360 '
19115 = '19115'
19117 = '19117'
19119 = '19119'
19121 = '19121'
19123 = '19123'
19125 = '19125'
19127 = '19127'
```

19129 = '19129'
19131 = '19131'
19133 = '19133'
19135 = '19135'
19137 = '19137'
19139 = '19139'
19141 = '19141'
19143 = '19143'
19145 = '19145'
19147 = '19147'
19149 = '19149'
19151 = '19151'
19153 = '2120 '
19155 = '5920 '
19157 = '19157'
19159 = '19159'
19161 = '19161'
19163 = '1960 '
19165 = '19165'
19167 = '19167'
19169 = '19169'
19171 = '19171'
19173 = '19173'
19175 = '19175'
19177 = '19177'
19179 = '19179'
19181 = '2120 '
19183 = '19183'
19185 = '19185'
19187 = '19187'
19189 = '19189'
19191 = '19191'
19193 = '7720 '
19195 = '19195'
19197 = '19197'
20000 = '20000'
20001 = '20001'
20003 = '20003'
20005 = '20005'
20007 = '20007'
20009 = '20009'
20011 = '20011'
20013 = '20013'
20015 = '9040 '
20017 = '20017'
20019 = '20019'
20021 = '20021'
20023 = '20023'
20025 = '20025'
20027 = '20027'

20029 = '20029'
20031 = '20031'
20033 = '20033'
20035 = '20035'
20037 = '20037'
20039 = '20039'

```
20041 = '20041'
20043 = '20043'
20045 = '4150 '
20047 = '20047'
20049 = '20049'
20051 = '20051'
20053 = '20053'
20055 = '20055'
20057 = '20057'
20059 = '20059'
20061 = '20061'
20063 = '20063'
20065 = '20065'
20067 = '20067'
20069 = '20069'
20071 = '20071'
20073 = '20073'
20075 = '20075'
20077 = '20077'
20079 = '9040 '
20081 = '20081'
20083 = '20083'
20085 = '20085'
20087 = '20087'
20089 = '20089'
20091 = '3760 '
20093 = '20093'
20095 = '20095'
20097 = '20097'
20099 = '20099'
20101 = '20101'
20103 = '3760 '
20105 = '20105'
20107 = '20107'
20109 = '20109'
20111 = '20111'
20113 = '20113'
20115 = '20115'
20117 = '20117'
20119 = '20119'
20121 = '3760 '
20123 = '20123'
20125 = '20125'
20127 = '20127'
20129 = '20129'
20131 = '20131'
20133 = '20133'
20135 = '20135'
20137 = '20137'
20139 = '20139'
20141 = '20141'
20143 = '20143'
20145 = '20145'
```
PSFMTS -- CONTINUED

```
20147 = '20147'
20149 = '20149'
20151 = '20151'
```

20153 = '20153'
20155 = '20155'
20157 = '20157'
20159 = '20159'
20161 = '20161'
20163 = '20163'
20165 = '20165'
20167 = '20167'
20169 = '20169'
20171 = '20171'
20173 = '9040 '
20175 = '20175'
20177 = '8440 '
20179 = '20179'
20181 = '20181'
20183 = '20183'
20185 = '20185'
20187 = '20187'
20189 = '20189'
20191 = '20191'
20193 = '20193'
20195 = '20195'
20197 = '20197'
20199 = '20199'
20201 = '20201'
20203 = '20203'
20205 = '20205'
20207 = '20207'
20209 = '3760 '
21000 = '21000'
21001 = '21001'
21003 = '21003'
21005 = '21005'
21007 = '21007'
21009 = '21009'
21011 = '21011'
21013 = '21013'
21015 = '1642 '
21017 = '4280 '
21019 = '3400 '
21021 = '21021'
21023 = '21023'
21025 = '21025'
21027 = '21027'
21029 = '4520 '
21031 = '21031'
21033 = '21033'
21035 = '21035'
21037 = '1642 '
21039 = '21039'
21041 = '21041'
21043 = '3400 '
21045 = '21045'
21047 = '1660 '
21049 = '4280 '
21051 = '21051'
PSFMTS -- CONTINUED

21053 = '21053'
21055 = '21055'
21057 = '21057'
21059 = '5990 '
21061 = '21061'
21063 = '21063'
21065 = '21065'
21067 = '4280 '
21069 = '21069'
21071 = '21071'
21073 = '21073'
21075 = '21075'
21077 = '1642 '
21079 = '21079'
21081 = '1642 '
21083 = '21083'
21085 = '21085'
21087 = '21087'
21089 = '3400 '
21091 = '21091'
21093 = '21093'
21095 = '21095'
21097 = '21097'
21099 = '21099'
21101 = '2440 '
21103 = '21103'
21105 = '21105'
21107 = '21107'
21109 = '21109'
21111 = '4520 '
21113 = '4280 '
21115 = '21115'
21117 = '1642 '
21119 = '21119'
21121 = '21121'
21123 = '21123'
21125 = '21125'
21127 = '21127'
21129 = '21129'
21131 = '21131'
21133 = '21133'
21135 = '21135'
21137 = '21137'
21139 = '21139'
21141 = '21141'
21143 = '21143'
21145 = '21145'
21147 = '21147'
21149 = '21149'
21151 = '4280 '
21153 = '21153'
21155 = '21155'
21157 = '21157'
21159 = '21159'
21161 = '21161'
21163 = '21163'
21165 = '21165'
21167 = '21167'

21169 = '21169'
PSFMTS -- CONTINUED

21171 = '21171'
21173 = '21173'
21175 = '21175'
21177 = '21177'
21179 = '21179'
21181 = '21181'
21183 = '21183'
21185 = '4520 '
21187 = '21187'
21189 = '21189'
21191 = '1642 '
21193 = '21193'
21195 = '21195'
21197 = '21197'
21199 = '21199'
21201 = '21201'
21203 = '21203'
21205 = '21205'
21207 = '21207'
21209 = '4280 '
21211 = '21211'
21213 = '21213'
21215 = '21215'
21217 = '21217'
21219 = '21219'
21221 = '21221'
21223 = '21223'
21225 = '21225'
21227 = '21227'
21229 = '21229'
21231 = '21231'
21233 = '21233'
21235 = '21235'
21237 = '21237'
21239 = '4280 '
22000 = '22000'
22001 = '3880 '
22003 = '22003'
22005 = '0760 '
22007 = '22007'
22009 = '22009'
22011 = '22011'
22013 = '22013'
22015 = '7680 '
22017 = '7680 '
22019 = '3960 '
22021 = '22021'
22023 = '22023'
22025 = '22025'
22027 = '22027'
22029 = '22029'
22031 = '22031'
22033 = '0760 '
22035 = '22035'
22037 = '22037'

22039 = '22039'
22041 = '22041'
22043 = '22043'
22045 = '22045'
PSFMTS -- CONTINUED

22047 = '22047'
22049 = '22049'
22051 = '5560 '
22053 = '22053'
22055 = '3880 '
22057 = '3350 '
22059 = '22059'
22061 = '22061'
22063 = '0760 '
22065 = '22065'
22067 = '22067'
22069 = '22069'
22071 = '5560 '
22073 = '5200 '
22075 = '5560 '
22077 = '22077'
22079 = '0220 '
22081 = '22081'
22083 = '22083'
22085 = '22085'
22087 = '5560 '
22089 = '5560 '
22091 = '22091'
22093 = '5560 '
22095 = '5560 '
22097 = '3880 '
22099 = '3880 '
22101 = '22101'
22103 = '5560 '
22105 = '22105'
22107 = '22107'
22109 = '3350 '
22111 = '22111'
22113 = '22113'
22115 = '22115'
22117 = '22117'
22119 = '7680 '
22121 = '0760 '
22123 = '22123'
22125 = '22125'
22127 = '22127'
23000 = '23000'
23001 = '4243 '
23003 = '23003'
23005 = '6403 '
23007 = '23007'
23009 = '23009'
23011 = '23011'
23013 = '23013'
23015 = '23015'
23017 = '23017'
23019 = '0733 '

```
23021 = '23021'
23023 = '23023'
23025 = '23025'
23027 = '23027'
23029 = '23029'
23031 = '23031'
24000 = '24000'
```
PSFMTS -- CONTINUED

```
24001 = '1900 '
24003 = '8872 '
24005 = '8872 '
24009 = '8872 '
24011 = '24011'
24013 = '8872 '
24015 = '6162 '
24017 = '8872 '
24019 = '24019'
24021 = '8872 '
24023 = '24023'
24025 = '8872 '
24027 = '8872 '
24029 = '24029'
24031 = '8872 '
24033 = '8872 '
24035 = '8872 '
24037 = '24037'
24039 = '24039'
24041 = '24041'
24043 = '8872 '
24045 = '24045'
24047 = '24047'
24510 = '8872 '
25000 = '25000'
25001 = '0743 '
25003 = '6323 '
25005 = '1123 '
25007 = '25007'
25009 = '1123 '
25011 = '25011'
25013 = '8003 '
25015 = '8003 '
25017 = '1123 '
25019 = '25019'
25021 = '1123 '
25023 = '1123 '
25025 = '1123 '
25027 = '1123 '
26000 = '26000'
26001 = '26001'
26003 = '26003'
26005 = '3000 '
26007 = '26007'
26009 = '26009'
26011 = '26011'
26013 = '26013'
26015 = '26015'
26017 = '6960 '
```

```
26019 = '26019'
26021 = '0870 '
26023 = '26023'
26025 = '3720 '
26027 = '26027'
26029 = '26029'
26031 = '26031'
26033 = '26033'
26035 = '26035'
26037 = '4040 '
```
PSFMTS -- CONTINUED

```
26039 = '26039'
26041 = '26041'
26043 = '26043'
26045 = '4040 '
26047 = '26047'
26049 = '2162 '
26051 = '26051'
26053 = '26053'
26055 = '26055'
26057 = '26057'
26059 = '26059'
26061 = '26061'
26063 = '26063'
26065 = '4040 '
26067 = '26067'
26069 = '26069'
26071 = '26071'
26073 = '26073'
26075 = '3520 '
26077 = '3720 '
26079 = '26079'
26081 = '3000 '
26083 = '26083'
26085 = '26085'
26087 = '2162 '
26089 = '26089'
26091 = '2162 '
26093 = '2162 '
26095 = '26095'
26097 = '26097'
26099 = '2162 '
26101 = '26101'
26103 = '26103'
26105 = '26105'
26107 = '26107'
26109 = '26109'
26111 = '6960 '
26113 = '26113'
26115 = '2162 '
26117 = '26117'
26119 = '26119'
26121 = '3000 '
26123 = '26123'
26125 = '2162 '
26127 = '26127'
26129 = '26129'
```

```
26131 = '26131'
26133 = '26133'
26135 = '26135'
26137 = '26137'
26139 = '3000 '
26141 = '26141'
26143 = '26143'
26145 = '6960 '
26147 = '2162 '
26149 = '26149'
26151 = '26151'
26153 = '26153'
26155 = '26155'
```
PSFMTS -- CONTINUED

```
26157 = '26157'
26159 = '3720 '
26161 = '2162 '
26163 = '2162 '
26165 = '26165'
27000 = '27000'
27001 = '27001'
27003 = '5120 '
27005 = '27005'
27007 = '27007'
27009 = '6980 '
27011 = '27011'
27013 = '27013'
27015 = '27015'
27017 = '27017'
27019 = '5120 '
27021 = '27021'
27023 = '27023'
27025 = '5120 '
27027 = '2520 '
27029 = '27029'
27031 = '27031'
27033 = '27033'
27035 = '27035'
27037 = '5120 '
27039 = '27039'
27041 = '27041'
27043 = '27043'
27045 = '27045'
27047 = '27047'
27049 = '27049'
27051 = '27051'
27053 = '5120 '
27055 = '3870 '
27057 = '27057'
27059 = '5120 '
27061 = '27061'
27063 = '27063'
27065 = '27065'
27067 = '27067'
27069 = '27069'
27071 = '27071'
27073 = '27073'
```

```
27075 = '27075'
27077 = '27077'
27079 = '27079'
27081 = '27081'
27083 = '27083'
27085 = '27085'
27087 = '27087'
27089 = '27089'
27091 = '27091'
27093 = '27093'
27095 = '27095'
27097 = '27097'
27099 = '27099'
27101 = '27101'
27103 = '27103'
27105 = '27105'
```

```
27107 = '27107'
27109 = '6820 '
27111 = '27111'
27113 = '27113'
27115 = '27115'
27117 = '27117'
27119 = '2985 '
27121 = '27121'
27123 = '5120 '
27125 = '27125'
27127 = '27127'
27129 = '27129'
27131 = '27131'
27133 = '27133'
27135 = '27135'
27137 = '2240 '
27139 = '5120 '
27141 = '5120 '
27143 = '27143'
27145 = '6980 '
27147 = '27147'
27149 = '27149'
27151 = '27151'
27153 = '27153'
27155 = '27155'
27157 = '27157'
27159 = '27159'
27161 = '27161'
27163 = '5120 '
27165 = '27165'
27167 = '27167'
27169 = '27169'
27171 = '5120 '
27173 = '27173'
28000 = '28000'
28001 = '28001'
28003 = '28003'
28005 = '28005'
28007 = '28007'
28009 = '28009'
```

```
28011 = '28011'
28013 = '28013'
28015 = '28015'
28017 = '28017'
28019 = '28019'
28021 = '28021'
28023 = '28023'
28025 = '28025'
28027 = '28027'
28029 = '28029'
28031 = '28031'
28033 = '4920 '
28035 = '3285 '
28037 = '28037'
28039 = '28039'
28041 = '28041'
28043 = '28043'
28045 = '0920 '
28047 = '0920 '
```
PSFMTS -- CONTINUED

```
28049 = '3560 '
28051 = '28051'
28053 = '28053'
28055 = '28055'
28057 = '28057'
28059 = '0920 '
28061 = '28061'
28063 = '28063'
28065 = '28065'
28067 = '28067'
28069 = '28069'
28071 = '28071'
28073 = '3285 '
28075 = '28075'
28077 = '28077'
28079 = '28079'
28081 = '28081'
28083 = '28083'
28085 = '28085'
28087 = '28087'
28089 = '3560 '
28091 = '28091'
28093 = '28093'
28095 = '28095'
28097 = '28097'
28099 = '28099'
28101 = '28101'
28103 = '28103'
28105 = '28105'
28107 = '28107'
28109 = '28109'
28111 = '28111'
28113 = '28113'
28115 = '28115'
28117 = '28117'
28119 = '28119'
28121 = '3560 '
```

28123 = '28123'
28125 = '28125'
28127 = '28127'
28129 = '28129'
28131 = '28131'
28133 = '28133'
28135 = '28135'
28137 = '28137'
28139 = '28139'
28141 = '28141'
28143 = '28143'
28145 = '28145'
28147 = '28147'
28149 = '28149'
28151 = '28151'
28153 = '28153'
28155 = '28155'
28157 = '28157'
28159 = '28159'
28161 = '28161'
28163 = '28163'
29000 = '29000'
PSFMTS -- CONTINUED

29001 = '29001'
29003 = '7000 '
29005 = '29005'
29007 = '29007'
29009 = '29009'
29011 = '29011'
29013 = '29013'
29015 = '29015'
29017 = '29017'
29019 = '1740 '
29021 = '7000 '
29023 = '29023'
29025 = '29025'
29027 = '29027'
29029 = '29029'
29031 = '29031'
29033 = '29033'
29035 = '29035'
29037 = '3760 '
29039 = '29039'
29041 = '29041'
29043 = '7920 '
29045 = '29045'
29047 = '3760 '
29049 = '3760 '
29051 = '29051'
29053 = '29053'
29055 = '7040 '
29057 = '29057'
29059 = '29059'
29061 = '29061'
29063 = '29063'
29065 = '29065'
29067 = '29067'

```
29069 = '29069'
29071 = '7040 '
29073 = '29073'
29075 = '29075'
29077 = '7920 '
29079 = '29079'
29081 = '29081'
29083 = '29083'
29085 = '29085'
29087 = '29087'
29089 = '29089'
29091 = '29091'
29093 = '29093'
29095 = '3760 '
29097 = '3710 '
29099 = '7040 '
29101 = '29101'
29103 = '29103'
29105 = '29105'
29107 = '3760 '
29109 = '29109'
29111 = '29111'
29113 = '7040 '
29115 = '29115'
29117 = '29117'
```

PSFMTS -- CONTINUED

```
29119 = '29119'
29121 = '29121'
29123 = '29123'
29125 = '29125'
29127 = '29127'
29129 = '29129'
29131 = '29131'
29133 = '29133'
29135 = '29135'
29137 = '29137'
29139 = '29139'
29141 = '29141'
29143 = '29143'
29145 = '3710 '
29147 = '29147'
29149 = '29149'
29151 = '29151'
29153 = '29153'
29155 = '29155'
29157 = '29157'
29159 = '29159'
29161 = '29161'
29163 = '29163'
29165 = '3760 '
29167 = '29167'
29169 = '29169'
29171 = '29171'
29173 = '29173'
29175 = '29175'
29177 = '3760 '
29179 = '29179'
```

```
29181 = '29181'
29183 = '7040 '
29185 = '29185'
29186 = '29186'
29187 = '29187'
29189 = '7040 '
29195 = '29195'
29197 = '29197'
29199 = '29199'
29201 = '29201'
29203 = '29203'
29205 = '29205'
29207 = '29207'
29209 = '29209'
29211 = '29211'
29213 = '29213'
29215 = '29215'
29217 = '29217'
29219 = '7040 '
29221 = '29221'
29223 = '29223'
29225 = '7920 '
29227 = '29227'
29229 = '29229'
29510 = '7040 '
30000 = '30000'
30001 = '30001'
30003 = '30003'
```
PSFMTS -- CONTINUED

```
30005 = '30005'
30007 = '30007'
30009 = '30009'
30011 = '30011'
30013 = '3040 '
30015 = '30015'
30017 = '30017'
30019 = '30019'
30021 = '30021'
30023 = '30023'
30025 = '30025'
30027 = '30027'
30029 = '30029'
30031 = '30031'
30033 = '30033'
30035 = '30035'
30037 = '30037'
30039 = '30039'
30041 = '30041'
30043 = '30043'
30045 = '30045'
30047 = '30047'
30049 = '30049'
30051 = '30051'
30053 = '30053'
30055 = '30055'
30057 = '30057'
30059 = '30059'
```

```
30061 = '30061'
30063 = '5140 '
30065 = '30065'
30067 = '30067'
30069 = '30069'
30071 = '30071'
30073 = '30073'
30075 = '30075'
30077 = '30077'
30079 = '30079'
30081 = '30081'
30083 = '30083'
30085 = '30085'
30087 = '30087'
30089 = '30089'
30091 = '30091'
30093 = '30093'
30095 = '30095'
30097 = '30097'
30099 = '30099'
30101 = '30101'
30103 = '30103'
30105 = '30105'
30107 = '30107'
30109 = '30109'
30111 = '0880 '
30113 = '30113'
31000 = '31000'
31001 = '31001'
31003 = '31003'
31005 = '31005'
```

PSFMTS -- CONTINUED

```
31007 = '31007'
31009 = '31009'
31011 = '31011'
31013 = '31013'
31015 = '31015'
31017 = '31017'
31019 = '31019'
31021 = '31021'
31023 = '31023'
31025 = '5920 '
31027 = '31027'
31029 = '31029'
31031 = '31031'
31033 = '31033'
31035 = '31035'
31037 = '31037'
31039 = '31039'
31041 = '31041'
31043 = '7720 '
31045 = '31045'
31047 = '31047'
31049 = '31049'
31051 = '31051'
31053 = '31053'
31055 = '5920 '
```

```
31057 = '31057'
31059 = '31059'
31061 = '31061'
31063 = '31063'
31065 = '31065'
31067 = '31067'
31069 = '31069'
31071 = '31071'
31073 = '31073'
31075 = '31075'
31077 = '31077'
31079 = '31079'
31081 = '31081'
31083 = '31083'
31085 = '31085'
31087 = '31087'
31089 = '31089'
31091 = '31091'
31093 = '31093'
31095 = '31095'
31097 = '31097'
31099 = '31099'
31101 = '31101'
31103 = '31103'
31105 = '31105'
31107 = '31107'
31109 = '4360 '
31111 = '31111'
31113 = '31113'
31115 = '31115'
31117 = '31117'
31119 = '31119'
31121 = '31121'
31123 = '31123'
```
PSFMTS -- CONTINUED

```
31125 = '31125'
31127 = '31127'
31129 = '31129'
31131 = '31131'
31133 = '31133'
31135 = '31135'
31137 = '31137'
31139 = '31139'
31141 = '31141'
31143 = '31143'
31145 = '31145'
31147 = '31147'
31149 = '31149'
31151 = '31151'
31153 = '5920 '
31155 = '31155'
31157 = '31157'
31159 = '31159'
31161 = '31161'
31163 = '31163'
31165 = '31165'
31167 = '31167'
```

```
31169 = '31169'
31171 = '31171'
31173 = '31173'
31175 = '31175'
31177 = '5920 '
31179 = '31179'
31181 = '31181'
31183 = '31183'
31185 = '31185'
32000 = '32000'
32001 = '32001'
32003 = '4120 '
32005 = '32005'
32007 = '32007'
32009 = '32009'
32011 = '32011'
32013 = '32013'
32015 = '32015'
32017 = '32017'
32019 = '32019'
32021 = '32021'
32023 = '4120 '
32027 = '32027'
32029 = '32029'
32031 = '6720 '
32033 = '32033'
32510 = '32510'
33000 = '33000'
33001 = '33001'
33003 = '33003'
33005 = '33005'
33007 = '33007'
33009 = '33009'
33011 = '1123 '
33013 = '33013'
33015 = '1123 '
33017 = '1123 '
```

PSFMTS -- CONTINUED

```
33019 = '33019'
34000 = '34000'
34001 = '6162 '
34003 = '5602 '
34005 = '6162 '
34007 = '6162 '
34009 = '6162 '
34011 = '6162 '
34013 = '5602 '
34015 = '6162 '
34017 = '5602 '
34019 = '5602 '
34021 = '5602 '
34023 = '5602 '
34025 = '5602 '
34027 = '5602 '
34029 = '5602 '
34031 = '5602 '
34033 = '6162 '
```

```
34035 = '5602 '
34037 = '5602 '
34039 = '5602 '
34041 = '5602 '
35000 = '35000'
35001 = '0200 '
35003 = '35003'
35005 = '35005'
35006 = '35006'
35007 = '35007'
35009 = '35009'
35011 = '35011'
35013 = '4100 '
35015 = '35015'
35017 = '35017'
35019 = '35019'
35021 = '35021'
35023 = '35023'
35025 = '35025'
35027 = '35027'
35028 = '7490 '
35029 = '35029'
35031 = '35031'
35033 = '35033'
35035 = '35035'
35037 = '35037'
35039 = '35039'
35041 = '35041'
35043 = '0200 '
35045 = '35045'
35047 = '35047'
35049 = '7490 '
35051 = '35051'
35053 = '35053'
35055 = '35055'
35057 = '35057'
35059 = '35059'
35061 = '0200 '
36000 = '36000'
36001 = '0160 '
```
PSFMTS -- CONTINUED

```
36003 = '36003'
36005 = '5602 '
36007 = '0960 '
36009 = '36009'
36011 = '8160 '
36013 = '3610 '
36015 = '2335 '
36017 = '36017'
36019 = '36019'
36021 = '36021'
36023 = '36023'
36025 = '36025'
36027 = '5602 '
36029 = '1280 '
36031 = '36031'
36033 = '36033'
```

```
36035 = '36035'
36037 = '6840 '
36039 = '36039'
36041 = '36041'
36043 = '8680 '
36045 = '36045'
36047 = '5602 '
36049 = '36049'
36051 = '6840 '
36053 = '8160 '
36055 = '6840 '
36057 = '0160 '
36059 = '5602 '
36061 = '5602 '
36063 = '1280 '
36065 = '8680 '
36067 = '8160 '
36069 = '6840 '
36071 = '5602 '
36073 = '6840 '
36075 = '8160 '
36077 = '36077'
36079 = '5602 '
36081 = '5602 '
36083 = '0160 '
36085 = '5602 '
36087 = '5602 '
36089 = '36089'
36091 = '0160 '
36093 = '0160 '
36095 = '0160 '
36097 = '36097'
36099 = '36099'
36101 = '36101'
36103 = '5602 '
36105 = '36105'
36107 = '0960 '
36109 = '36109'
36111 = '36111'
36113 = '2975 '
36115 = '2975 '
36117 = '6840 '
36119 = '5602 '
```

PSFMTS -- CONTINUED

```
36121 = '36121'
36123 = '36123'
37000 = '37000'
37001 = '3120 '
37003 = '3290 '
37005 = '37005'
37007 = '37007'
37009 = '37009'
37011 = '37011'
37013 = '37013'
37015 = '37015'
37017 = '37017'
37019 = '9200 '
```

```
37021 = '0480 '
37023 = '3290 '
37025 = '1520 '
37027 = '3290 '
37029 = '37029'
37031 = '37031'
37033 = '37033'
37035 = '3290 '
37037 = '6640 '
37039 = '37039'
37041 = '37041'
37043 = '37043'
37045 = '37045'
37047 = '37047'
37049 = '37049'
37051 = '2560 '
37053 = '5720 '
37055 = '37055'
37057 = '3120 '
37059 = '3120 '
37061 = '37061'
37063 = '6640 '
37065 = '6895 '
37067 = '3120 '
37069 = '6640 '
37071 = '1520 '
37073 = '37073'
37075 = '37075'
37077 = '37077'
37079 = '37079'
37081 = '3120 '
37083 = '37083'
37085 = '37085'
37087 = '37087'
37089 = '37089'
37091 = '37091'
37093 = '37093'
37095 = '37095'
37097 = '37097'
37099 = '37099'
37101 = '6640 '
37103 = '37103'
37105 = '37105'
37107 = '37107'
37109 = '1520 '
37111 = '37111'
```

PSFMTS -- CONTINUED

```
37113 = '37113'
37115 = '0480 '
37117 = '37117'
37119 = '1520 '
37121 = '37121'
37123 = '37123'
37125 = '37125'
37127 = '6895 '
37129 = '9200 '
37131 = '37131'
```

```
37133 = '3605 '
37135 = '6640 '
37137 = '37137'
37139 = '37139'
37141 = '37141'
37143 = '37143'
37145 = '37145'
37147 = '3150 '
37149 = '37149'
37151 = '3120 '
37153 = '37153'
37155 = '37155'
37157 = '37157'
37159 = '1520 '
37161 = '37161'
37163 = '37163'
37165 = '37165'
37167 = '37167'
37169 = '3120 '
37171 = '37171'
37173 = '37173'
37175 = '37175'
37177 = '37177'
37179 = '1520 '
37181 = '37181'
37183 = '6640 '
37185 = '37185'
37187 = '37187'
37189 = '37189'
37191 = '2980 '
37193 = '37193'
37195 = '37195'
37197 = '3120 '
37199 = '37199'
38000 = '38000'
38001 = '38001'
38003 = '38003'
38005 = '38005'
38007 = '38007'
38009 = '38009'
38011 = '38011'
38013 = '38013'
38015 = '1010 '
38017 = '2520 '
38019 = '38019'
38021 = '38021'
38023 = '38023'
38025 = '38025'
38027 = '38027'
```

PSFMTS -- CONTINUED

```
38029 = '38029'
38031 = '38031'
38033 = '38033'
38035 = '2985 '
38037 = '38037'
38039 = '38039'
38041 = '38041'
```

38043 = '38043'
38045 = '38045'
38047 = '38047'
38049 = '38049'
38051 = '38051'
38053 = '38053'
38055 = '38055'
38057 = '38057'
38059 = '1010 '
38061 = '38061'
38063 = '38063'
38065 = '38065'
38067 = '38067'
38069 = '38069'
38071 = '38071'
38073 = '38073'
38075 = '38075'
38077 = '38077'
38079 = '38079'
38081 = '38081'
38083 = '38083'
38085 = '38085'
38087 = '38087'
38089 = '38089'
38091 = '38091'
38093 = '38093'
38095 = '38095'
38097 = '38097'
38099 = '38099'
38101 = '38101'
38103 = '38103'
38105 = '38105'
39000 = '39000'
39001 = '39001'
39003 = '4320 '
39005 = '39005'
39007 = '1692 '
39009 = '39009'
39011 = '4320 '
39013 = '9000 '
39015 = '1642 '
39017 = '1642 '
39019 = '1320 '
39021 = '39021'
39023 = '2000 '
39025 = '1642 '
39027 = '39027'
39029 = '9320 '
39031 = '39031'
39033 = '4800 '
39035 = '1692 '
39037 = '39037'

PSFMTS -- CONTINUED

39039 = '39039'
39041 = '1840 '
39043 = '39043'
39045 = '1840 '

```
39047 = '39047'
39049 = '1840 '
39051 = '8400 '
39053 = '39053'
39055 = '1692 '
39057 = '2000 '
39059 = '39059'
39061 = '1642 '
39063 = '39063'
39065 = '39065'
39067 = '39067'
39069 = '39069'
39071 = '39071'
39073 = '39073'
39075 = '39075'
39077 = '39077'
39079 = '39079'
39081 = '8080 '
39083 = '39083'
39085 = '1692 '
39087 = '3400 '
39089 = '1840 '
39091 = '39091'
39093 = '1692 '
39095 = '8400 '
39097 = '1840 '
39099 = '9320 '
39101 = '39101'
39103 = '1692 '
39105 = '39105'
39107 = '39107'
39109 = '2000 '
39111 = '39111'
39113 = '2000 '
39115 = '39115'
39117 = '39117'
39119 = '39119'
39121 = '39121'
39123 = '39123'
39125 = '39125'
39127 = '39127'
39129 = '1840 '
39131 = '39131'
39133 = '1692 '
39135 = '39135'
39137 = '39137'
39139 = '4800 '
39141 = '39141'
39143 = '39143'
39145 = '39145'
39147 = '39147'
39149 = '39149'
39151 = '1320 '
39153 = '1692 '
39155 = '9320 '
```

PSFMTS -- CONTINUED

```
39157 = '39157'
```

39159 = '39159'
39161 = '39161'
39163 = '39163'
39165 = '1642 '
39167 = '6020 '
39169 = '39169'
39171 = '39171'
39173 = '8400 '
39175 = '39175'
40000 = '40000'
40001 = '40001'
40003 = '40003'
40005 = '40005'
40007 = '40007'
40009 = '40009'
40011 = '40011'
40013 = '40013'
40015 = '40015'
40017 = '5880 '
40019 = '40019'
40021 = '40021'
40023 = '40023'
40025 = '40025'
40027 = '5880 '
40029 = '40029'
40031 = '4200 '
40033 = '40033'
40035 = '40035'
40037 = '8560 '
40039 = '40039'
40041 = '40041'
40043 = '40043'
40045 = '40045'
40047 = '2340 '
40049 = '40049'
40051 = '40051'
40053 = '40053'
40055 = '40055'
40057 = '40057'
40059 = '40059'
40061 = '40061'
40063 = '40063'
40065 = '40065'
40067 = '40067'
40069 = '40069'
40071 = '40071'
40073 = '40073'
40075 = '40075'
40077 = '40077'
40079 = '40079'
40081 = '40081'
40083 = '5880 '
40085 = '40085'
40087 = '5880 '
40089 = '40089'
40091 = '40091'
40093 = '40093'
40095 = '40095'

PSFMTS -- CONTINUED

```
40097 = '40097'
40099 = '40099'
40101 = '40101'
40103 = '40103'
40105 = '40105'
40107 = '40107'
40109 = '5880 '
40111 = '40111'
40113 = '8560 '
40115 = '40115'
40117 = '40117'
40119 = '40119'
40121 = '40121'
40123 = '40123'
40125 = '5880 '
40127 = '40127'
40129 = '40129'
40131 = '8560 '
40133 = '40133'
40135 = '2720 '
40137 = '40137'
40139 = '40139'
40141 = '40141'
40143 = '8560 '
40145 = '8560 '
40147 = '40147'
40149 = '40149'
40151 = '40151'
40153 = '40153'
41000 = '41000'
41001 = '41001'
41003 = '1890 '
41005 = '6442 '
41007 = '41007'
41009 = '6442 '
41011 = '41011'
41013 = '41013'
41015 = '41015'
41017 = '41017'
41019 = '41019'
41021 = '41021'
41023 = '41023'
41025 = '41025'
41027 = '41027'
41029 = '4890 '
41031 = '41031'
41033 = '41033'
41035 = '41035'
41037 = '41037'
41039 = '2400 '
41041 = '41041'
41043 = '41043'
41045 = '41045'
41047 = '6442 '
41049 = '41049'
41051 = '6442 '
```

```
41053 = '6442 '
41055 = '41055'
41057 = '41057'
```
PSFMTS -- CONTINUED

```
41059 = '41059'
41061 = '41061'
41063 = '41063'
41065 = '41065'
41067 = '6442 '
41069 = '41069'
41071 = '6442 '
42000 = '42000'
42001 = '42001'
42003 = '6280 '
42005 = '42005'
42007 = '6280 '
42009 = '42009'
42011 = '6680 '
42013 = '0280 '
42015 = '42015'
42017 = '6162 '
42019 = '6280 '
42021 = '3680 '
42023 = '42023'
42025 = '0240 '
42027 = '8050 '
42029 = '6162 '
42031 = '42031'
42033 = '42033'
42035 = '42035'
42037 = '7560 '
42039 = '42039'
42041 = '3240 '
42043 = '3240 '
42045 = '6162 '
42047 = '42047'
42049 = '2360 '
42051 = '6280 '
42053 = '42053'
42055 = '42055'
42057 = '42057'
42059 = '42059'
42061 = '42061'
42063 = '42063'
42065 = '42065'
42067 = '42067'
42069 = '7560 '
42071 = '4000 '
42073 = '42073'
42075 = '3240 '
42077 = '0240 '
42079 = '7560 '
42081 = '9140 '
42083 = '42083'
42085 = '7610 '
42087 = '42087'
42089 = '42089'
```

```
42091 = '6162 '
42093 = '42093'
42095 = '0240 '
42097 = '42097'
42099 = '3240 '
42101 = '6162 '
```
PSFMTS -- CONTINUED

```
42103 = '5602 '
42105 = '42105'
42107 = '42107'
42109 = '42109'
42111 = '3680 '
42113 = '42113'
42115 = '42115'
42117 = '42117'
42119 = '42119'
42121 = '42121'
42123 = '42123'
42125 = '6280 '
42127 = '42127'
42129 = '6280 '
42131 = '7560 '
42133 = '9280 '
44000 = '44000'
44001 = '6483 '
44003 = '6483 '
44005 = '44005'
44007 = '6483 '
44009 = '6483 '
45000 = '45000'
45001 = '45001'
45003 = '0600 '
45005 = '45005'
45007 = '3160 '
45009 = '45009'
45011 = '45011'
45013 = '45013'
45015 = '1440 '
45017 = '45017'
45019 = '1440 '
45021 = '3160 '
45023 = '45023'
45025 = '45025'
45027 = '45027'
45029 = '45029'
45031 = '45031'
45033 = '45033'
45035 = '1440 '
45037 = '0600 '
45039 = '45039'
45041 = '2655 '
45043 = '45043'
45045 = '3160 '
45047 = '45047'
45049 = '45049'
45051 = '5330 '
45053 = '45053'
```

45055 = '45055'
45057 = '45057'
45059 = '45059'
45061 = '45061'
45063 = '1760 '
45065 = '45065'
45067 = '45067'
45069 = '45069'
45071 = '45071'
PSFMTS -- CONTINUED

45073 = '45073'
45075 = '45075'
45077 = '3160 '
45079 = '1760 '
45081 = '45081'
45083 = '3160 '
45085 = '8140 '
45087 = '45087'
45089 = '45089'
45091 = '1520 '
46000 = '46000'
46003 = '46003'
46005 = '46005'
46007 = '46007'
46009 = '46009'
46011 = '46011'
46013 = '46013'
46015 = '46015'
46017 = '46017'
46019 = '46019'
46021 = '46021'
46023 = '46023'
46025 = '46025'
46027 = '46027'
46029 = '46029'
46031 = '46031'
46033 = '46033'
46035 = '46035'
46037 = '46037'
46039 = '46039'
46041 = '46041'
46043 = '46043'
46045 = '46045'
46047 = '46047'
46049 = '46049'
46051 = '46051'
46053 = '46053'
46055 = '46055'
46057 = '46057'
46059 = '46059'
46061 = '46061'
46063 = '46063'
46065 = '46065'
46067 = '46067'
46069 = '46069'
46071 = '46071'
46073 = '46073'

46075 = '46075'
46077 = '46077'
46079 = '46079'
46081 = '46081'
46083 = '7760 '
46085 = '46085'
46087 = '46087'
46089 = '46089'
46091 = '46091'
46093 = '46093'
46095 = '46095'
46097 = '46097'

PSFMTS -- CONTINUED

46099 = '7760 '
46101 = '46101'
46103 = '6660 '
46105 = '46105'
46107 = '46107'
46109 = '46109'
46111 = '46111'
46113 = '46113'
46115 = '46115'
46117 = '46117'
46119 = '46119'
46121 = '46121'
46123 = '46123'
46125 = '46125'
46127 = '46127'
46129 = '46129'
46135 = '46135'
46137 = '46137'
47000 = '47000'
47001 = '3840 '
47003 = '47003'
47005 = '47005'
47007 = '47007'
47009 = '3840 '
47011 = '47011'
47013 = '47013'
47015 = '47015'
47017 = '47017'
47019 = '3660 '
47021 = '5360 '
47023 = '3580 '
47025 = '47025'
47027 = '47027'
47029 = '47029'
47031 = '47031'
47033 = '47033'
47035 = '47035'
47037 = '5360 '
47039 = '47039'
47041 = '47041'
47043 = '5360 '
47045 = '47045'
47047 = '4920 '
47049 = '47049'

47051 = '47051'
47053 = '47053'
47055 = '47055'
47057 = '47057'
47059 = '47059'
47061 = '47061'
47063 = '47063'
47065 = '1560 '
47067 = '47067'
47069 = '47069'
47071 = '47071'
47073 = '3660 '
47075 = '47075'
47077 = '47077'
47079 = '47079'

PSFMTS -- CONTINUED

47081 = '47081'
47083 = '47083'
47085 = '47085'
47087 = '47087'
47089 = '47089'
47091 = '47091'
47093 = '3840 '
47095 = '47095'
47097 = '47097'
47099 = '47099'
47101 = '47101'
47103 = '47103'
47105 = '3840 '
47107 = '47107'
47109 = '47109'
47111 = '47111'
47113 = '3580 '
47115 = '1560 '
47117 = '47117'
47119 = '47119'
47121 = '47121'
47123 = '47123'
47125 = '1660 '
47127 = '47127'
47129 = '47129'
47131 = '47131'
47133 = '47133'
47135 = '47135'
47137 = '47137'
47139 = '47139'
47141 = '47141'
47143 = '47143'
47145 = '47145'
47147 = '5360 '
47149 = '5360 '
47151 = '47151'
47153 = '47153'
47155 = '3840 '
47157 = '4920 '
47159 = '47159'
47161 = '47161'

```
47163 = '3660 '
47165 = '5360 '
47167 = '4920 '
47169 = '47169'
47171 = '3660 '
47173 = '3840 '
47175 = '47175'
47177 = '47177'
47179 = '3660 '
47181 = '47181'
47183 = '47183'
47185 = '47185'
47187 = '5360 '
47189 = '5360 '
48000 = '48000'
48001 = '48001'
48003 = '48003'
48005 = '48005'
```
PSFMTS -- CONTINUED

```
48007 = '48007'
48009 = '9080 '
48011 = '48011'
48013 = '48013'
48015 = '48015'
48017 = '48017'
48019 = '48019'
48021 = '0640 '
48023 = '48023'
48025 = '48025'
48027 = '3810 '
48029 = '7240 '
48031 = '48031'
48033 = '48033'
48035 = '48035'
48037 = '8360 '
48039 = '3362 '
48041 = '1260 '
48043 = '48043'
48045 = '48045'
48047 = '48047'
48049 = '48049'
48051 = '48051'
48053 = '48053'
48055 = '0640 '
48057 = '48057'
48059 = '48059'
48061 = '1240 '
48063 = '48063'
48065 = '48065'
48067 = '48067'
48069 = '48069'
48071 = '3362 '
48073 = '48073'
48075 = '48075'
48077 = '48077'
48079 = '48079'
48081 = '48081'
```

```
48083 = '48083'
48085 = '1922 '
48087 = '48087'
48089 = '48089'
48091 = '7240 '
48093 = '48093'
48095 = '48095'
48097 = '48097'
48099 = '3810 '
48101 = '48101'
48103 = '48103'
48105 = '48105'
48107 = '48107'
48109 = '48109'
48111 = '48111'
48113 = '1922 '
48115 = '48115'
48117 = '48117'
48119 = '48119'
48121 = '1922 '
48123 = '48123'
```
PSFMTS -- CONTINUED

```
48125 = '48125'
48127 = '48127'
48129 = '48129'
48131 = '48131'
48133 = '48133'
48135 = '5800 '
48137 = '48137'
48139 = '1922 '
48141 = '2320 '
48143 = '48143'
48145 = '48145'
48147 = '48147'
48149 = '48149'
48151 = '48151'
48153 = '48153'
48155 = '48155'
48157 = '3362 '
48159 = '48159'
48161 = '48161'
48163 = '48163'
48165 = '48165'
48167 = '3362 '
48169 = '48169'
48171 = '48171'
48173 = '48173'
48175 = '48175'
48177 = '48177'
48179 = '48179'
48181 = '7640 '
48183 = '4420 '
48185 = '48185'
48187 = '7240 '
48189 = '48189'
48191 = '48191'
48193 = '48193'
```

48195 = '48195'
48197 = '48197'
48199 = '0840 '
48201 = '3362 '
48203 = '4420 '
48205 = '48205'
48207 = '48207'
48209 = '0640 '
48211 = '48211'
48213 = '1922 '
48215 = '4880 '
48217 = '48217'
48219 = '48219'
48221 = '1922 '
48223 = '48223'
48225 = '48225'
48227 = '48227'
48229 = '48229'
48231 = '1922 '
48233 = '48233'
48235 = '48235'
48237 = '48237'
48239 = '48239'
48241 = '48241'
PSFMTS -- CONTINUED

48243 = '48243'
48245 = '0840 '
48247 = '48247'
48249 = '48249'
48251 = '1922 '
48253 = '48253'
48255 = '48255'
48257 = '1922 '
48259 = '48259'
48261 = '48261'
48263 = '48263'
48265 = '48265'
48267 = '48267'
48269 = '48269'
48271 = '48271'
48273 = '48273'
48275 = '48275'
48277 = '48277'
48279 = '48279'
48281 = '48281'
48283 = '48283'
48285 = '48285'
48287 = '48287'
48289 = '48289'
48291 = '3362 '
48293 = '48293'
48295 = '48295'
48297 = '48297'
48299 = '48299'
48301 = '48301'
48303 = '4600 '
48305 = '48305'

```
48307 = '48307'
48309 = '8800 '
48311 = '48311'
48313 = '48313'
48315 = '48315'
48317 = '48317'
48319 = '48319'
48321 = '48321'
48323 = '48323'
48325 = '48325'
48327 = '48327'
48329 = '5800 '
48331 = '48331'
48333 = '48333'
48335 = '48335'
48337 = '48337'
48339 = '3362 '
48341 = '48341'
48343 = '48343'
48345 = '48345'
48347 = '48347'
48349 = '48349'
48351 = '48351'
48353 = '48353'
48355 = '1880 '
48357 = '48357'
48359 = '48359'
```
PSFMTS -- CONTINUED

```
48361 = '0840 '
48363 = '48363'
48365 = '48365'
48367 = '1922 '
48369 = '48369'
48371 = '48371'
48373 = '48373'
48375 = '0320 '
48377 = '48377'
48379 = '48379'
48381 = '0320 '
48383 = '48383'
48385 = '48385'
48387 = '48387'
48389 = '48389'
48391 = '48391'
48393 = '48393'
48395 = '48395'
48397 = '1922 '
48399 = '48399'
48401 = '48401'
48403 = '48403'
48405 = '48405'
48407 = '48407'
48409 = '1880 '
48411 = '48411'
48413 = '48413'
48415 = '48415'
48417 = '48417'
```

48419 = '48419'
48421 = '48421'
48423 = '8640 '
48425 = '48425'
48427 = '48427'
48429 = '48429'
48431 = '48431'
48433 = '48433'
48435 = '48435'
48437 = '48437'
48439 = '1922 '
48441 = '0040 '
48443 = '48443'
48445 = '48445'
48447 = '48447'
48449 = '48449'
48451 = '7200 '
48453 = '0640 '
48455 = '48455'
48457 = '48457'
48459 = '4420 '
48461 = '48461'
48463 = '48463'
48465 = '48465'
48467 = '48467'
48469 = '8750 '
48471 = '48471'
48473 = '3362 '
48475 = '48475'
48477 = '48477'
PSFMTS -- CONTINUED

48479 = '4080 '
48481 = '48481'
48483 = '48483'
48485 = '9080 '
48487 = '48487'
48489 = '48489'
48491 = '0640 '
48493 = '7240 '
48495 = '48495'
48497 = '48497'
48499 = '48499'
48501 = '48501'
48503 = '48503'
48505 = '48505'
48507 = '48507'
49000 = '49000'
49001 = '49001'
49003 = '49003'
49005 = '49005'
49007 = '49007'
49009 = '49009'
49011 = '7160 '
49013 = '49013'
49015 = '49015'
49017 = '49017'
49019 = '49019'

```
49021 = '49021'
49023 = '49023'
49025 = '2620 '
49027 = '49027'
49029 = '49029'
49031 = '49031'
49033 = '49033'
49035 = '7160 '
49037 = '49037'
49039 = '49039'
49041 = '49041'
49043 = '49043'
49045 = '49045'
49047 = '49047'
49049 = '6520 '
49051 = '49051'
49053 = '49053'
49055 = '49055'
49057 = '7160 '
50000 = '50000'
50001 = '50001'
50003 = '50003'
50005 = '50005'
50007 = '1303 '
50009 = '50009'
50011 = '1303 '
50013 = '1303 '
50015 = '50015'
50017 = '50017'
50019 = '50019'
50021 = '50021'
50023 = '50023'
50025 = '50025'
```

PSFMTS -- CONTINUED

```
50027 = '50027'
51000 = '51000'
51001 = '51001'
51003 = '1540 '
51005 = '51005'
51007 = '51007'
51009 = '4640 '
51011 = '51011'
51013 = '8872 '
51015 = '51015'
51017 = '51017'
51019 = '4640 '
51021 = '51021'
51023 = '6800 '
51025 = '51025'
51027 = '51027'
51029 = '51029'
51031 = '4640 '
51033 = '51033'
51035 = '51035'
51036 = '6760 '
51037 = '51037'
51041 = '6760 '
```

```
51043 = '8872 '
51045 = '51045'
51047 = '8872 '
51049 = '51049'
51051 = '51051'
51053 = '6760 '
51057 = '51057'
51059 = '8872 '
51061 = '8872 '
51063 = '51063'
51065 = '1540 '
51067 = '51067'
51069 = '51069'
51071 = '51071'
51073 = '5720 '
51075 = '6760 '
51077 = '51077'
51079 = '1540 '
51081 = '51081'
51083 = '51083'
51085 = '6760 '
51087 = '6760 '
51089 = '51089'
51091 = '51091'
51093 = '5720 '
51095 = '5720 '
51097 = '51097'
51099 = '8872 '
51101 = '51101'
51103 = '51103'
51105 = '51105'
51107 = '8872 '
51109 = '51109'
51111 = '51111'
51113 = '51113'
51115 = '5720 '
```
PSFMTS -- CONTINUED

```
51117 = '51117'
51119 = '51119'
51121 = '51121'
51125 = '51125'
51127 = '6760 '
51131 = '51131'
51133 = '51133'
51135 = '51135'
51137 = '51137'
51139 = '51139'
51141 = '51141'
51143 = '1950 '
51145 = '6760 '
51147 = '51147'
51149 = '6760 '
51153 = '8872 '
51155 = '51155'
51157 = '51157'
51159 = '51159'
51161 = '6800 '
```

```
51163 = '51163'
51165 = '51165'
51167 = '51167'
51169 = '3660 '
51171 = '51171'
51173 = '51173'
51175 = '51175'
51177 = '8872 '
51179 = '8872 '
51181 = '51181'
51183 = '51183'
51185 = '51185'
51187 = '8872 '
51191 = '3660 '
51193 = '51193'
51195 = '51195'
51197 = '51197'
51199 = '5720 '
51510 = '8872 '
51515 = '4640 '
51520 = '3660 '
51530 = '51530'
51540 = '1540 '
51550 = '5720 '
51560 = '51560'
51570 = '6760 '
51580 = '51580'
51590 = '1950 '
51595 = '51595'
51600 = '8872 '
51610 = '8872 '
51620 = '51620'
51630 = '8872 '
51640 = '51640'
51650 = '5720 '
51660 = '51660'
51670 = '6760 '
51678 = '51678'
51680 = '4640 '
```

PSFMTS -- CONTINUED

```
51683 = '8872 '
51685 = '8872 '
51690 = '51690'
51700 = '5720 '
51710 = '5720 '
51720 = '51720'
51730 = '6760 '
51735 = '5720 '
51740 = '5720 '
51750 = '51750'
51760 = '6760 '
51770 = '6800 '
51775 = '6800 '
51780 = '51780'
51790 = '51790'
51800 = '5720 '
51810 = '5720 '
```

```
51820 = '51820'
51830 = '5720 '
51840 = '51840'
53000 = '53000'
53001 = '53001'
53003 = '53003'
53005 = '6740 '
53007 = '53007'
53009 = '53009'
53011 = '6442 '
53013 = '53013'
53015 = '53015'
53017 = '53017'
53019 = '53019'
53021 = '6740 '
53023 = '53023'
53025 = '53025'
53027 = '53027'
53029 = '7602 '
53031 = '53031'
53033 = '7602 '
53035 = '7602 '
53037 = '53037'
53039 = '53039'
53041 = '53041'
53043 = '53043'
53045 = '53045'
53047 = '53047'
53049 = '53049'
53051 = '53051'
53053 = '7602 '
53055 = '53055'
53057 = '53057'
53059 = '53059'
53061 = '7602 '
53063 = '7840 '
53065 = '53065'
53067 = '7602 '
53069 = '53069'
53071 = '53071'
53073 = '0860 '
53075 = '53075'
```

PSFMTS -- CONTINUED

```
53077 = '9260 '
54000 = '54000'
54001 = '54001'
54003 = '8872 '
54005 = '54005'
54007 = '54007'
54009 = '8080 '
54011 = '3400 '
54013 = '54013'
54015 = '54015'
54017 = '54017'
54019 = '54019'
54021 = '54021'
54023 = '54023'
```

```
54025 = '54025'
54027 = '54027'
54029 = '8080 '
54031 = '54031'
54033 = '54033'
54035 = '54035'
54037 = '8872 '
54039 = '1480 '
54041 = '54041'
54043 = '54043'
54045 = '54045'
54047 = '54047'
54049 = '54049'
54051 = '9000 '
54053 = '54053'
54055 = '54055'
54057 = '1900 '
54059 = '54059'
54061 = '54061'
54063 = '54063'
54065 = '54065'
54067 = '54067'
54069 = '9000 '
54071 = '54071'
54073 = '54073'
54075 = '54075'
54077 = '54077'
54079 = '1480 '
54081 = '54081'
54083 = '54083'
54085 = '54085'
54087 = '54087'
54089 = '54089'
54091 = '54091'
54093 = '54093'
54095 = '54095'
54097 = '54097'
54099 = '3400 '
54101 = '54101'
54103 = '54103'
54105 = '54105'
54107 = '6020 '
54109 = '54109'
55000 = '55000'
55001 = '55001'
```
PSFMTS -- CONTINUED

```
55003 = '55003'
55005 = '55005'
55007 = '55007'
55009 = '3080 '
55011 = '55011'
55013 = '55013'
55015 = '0460 '
55017 = '2290 '
55019 = '55019'
55021 = '55021'
55023 = '55023'
```

```
55025 = '4720 '
55027 = '55027'
55029 = '55029'
55031 = '2240 '
55033 = '55033'
55035 = '2290 '
55037 = '55037'
55039 = '55039'
55041 = '55041'
55043 = '55043'
55045 = '55045'
55047 = '55047'
55049 = '55049'
55051 = '55051'
55053 = '55053'
55055 = '55055'
55057 = '55057'
55059 = '1602 '
55061 = '55061'
55063 = '3870 '
55065 = '55065'
55067 = '55067'
55069 = '55069'
55071 = '55071'
55073 = '8940 '
55075 = '55075'
55077 = '55077'
55078 = '55078'
55079 = '5082 '
55081 = '55081'
55083 = '55083'
55085 = '55085'
55087 = '0460 '
55089 = '5082 '
55091 = '55091'
55093 = '5120 '
55095 = '55095'
55097 = '55097'
55099 = '55099'
55101 = '5082 '
55103 = '55103'
55105 = '3620 '
55107 = '55107'
55109 = '5120 '
55111 = '55111'
55113 = '55113'
55115 = '55115'
55117 = '7620 '
```

PSFMTS -- CONTINUED

```
55119 = '55119'
55121 = '55121'
55123 = '55123'
55125 = '55125'
55127 = '55127'
55129 = '55129'
55131 = '5082 '
55133 = '5082 '
```

```
    55135 = '55135'
    55137 = '55137'
    55139 = '0460 '
    55141 = '55141'
    56000 = '56000'
    56001 = '56001'
    56003 = '56003'
    56005 = '56005'
    56007 = '56007'
    56009 = '56009'
    56011 = '56011'
    56013 = '56013'
    56015 = '56015'
    56017 = '56017'
    56019 = '56019'
    56021 = '1580 '
    56023 = '56023'
    56025 = '1350 '
    56027 = '56027'
    56029 = '56029'
    56031 = '56031'
    56033 = '56033'
    56035 = '56035'
    56037 = '56037'
    56039 = '56039'
    56041 = '56041'
    56043 = '56043'
    56045 = '56045'
    OTHER = '     '
    ;

    /*=============================================================================
====;
    *  Title:
    *          Format Library for Patient Safety Indicators
    *
    *  Description:
    *          COVARIATE FORMAT LIB FOR AHRQ PATIENT SAFETY INDICATORS;
    *
    *                 >>>  VERSION 2.1  <<<
    *
    *============================================================================
===;*/

     /* ------------------------------------------------------------- */
     /* ---   FORMAT FOR AGE STRATIFIERS                          --- */
     /* ---   MODULE 3. ASSIGNS ARRAY INDEX FOR AGE CATEGORIES    --- */
     /* ------------------------------------------------------------- */

    VALUE AGEFMT
    low - 4 = 1
    PSFMTS -- CONTINUED

     5 -  9 = 2
    10 - 14 = 3
    15 - 17 = 4
    18 - 24 = 5
    25 - 29 = 6
```

```
30 - 34 = 7
35 - 39 = 8
40 - 44 = 9
45 - 49 = 10
50 - 54 = 11
55 - 59 = 12
60 - 64 = 13
65 - 69 = 14
70 - 74 = 15
75 - 79 = 16
80 - 84 = 17
85 - high = 18;

 /* ---------------------------------------------------------------- */
 /* ---   FORMATS FOR DRG STRATIFIERS                            --- */
 /* ---   MODULE 3. ASSIGNS ARRAY INDEX FOR DRG CATEGORIES       --- */
 /* ---------------------------------------------------------------- */

VALUE $DRGFMT
'001' = 1   /* CRANIOTOMY AGE >17 EXCEPT FOR TRAUMA */
'002' = 2   /* CRANIOTOMY FOR TRAUMA AGE >17 */
'003' = 3   /* CRANIOTOMY AGE 0-17 */
'004' = 4   /* SPINAL PROCEDURES */
'005' = 5   /* EXTRACRANIAL VASCULAR PROCEDURES */
'006' = 6   /* CARPAL TUNNEL RELEASE */
'007' = 7   /* PERIPH & CRANIAL NERVE & OTHER NERV SYST PROC W CC */
'008' = 7   /* PERIPH & CRANIAL NERVE & OTHER NERV SYST PROC W/O CC */
'009' = 8   /* SPINAL DISORDERS & INJURIES */
'010' = 9   /* NERVOUS SYSTEM NEOPLASMS W CC */
'011' = 9   /* NERVOUS SYSTEM NEOPLASMS W/O CC */
'012' = 10  /* DEGENERATIVE NERVOUS SYSTEM DISORDERS */
'013' = 11  /* MULTIPLE SCLEROSIS & CEREBELLAR ATAXIA */
'014' = 12  /* SPECIFIC CEREBROVASCULAR DISORDERS EXCEPT TIA */
'015' = 13  /* TRANSIENT ISCHEMIC ATTACK & PRECEREBRAL OCCLUSIONS */
'016' = 14  /* NONSPECIFIC CEREBROVASCULAR DISORDERS W CC */
'017' = 14  /* NONSPECIFIC CEREBROVASCULAR DISORDERS W/O CC */
'018' = 15  /* CRANIAL & PERIPHERAL NERVE DISORDERS W CC */
'019' = 15  /* CRANIAL & PERIPHERAL NERVE DISORDERS W/O CC */
'020' = 16  /* NERVOUS SYSTEM INFECTION EXCEPT VIRAL MENINGITIS */
'021' = 17  /* VIRAL MENINGITIS */
'022' = 18  /* HYPERTENSIVE ENCEPHALOPATHY */
'023' = 19  /* NONTRAUMATIC STUPOR & COMA */
'024' = 20  /* SEIZURE & HEADACHE AGE >17 W CC */
'025' = 20  /* SEIZURE & HEADACHE AGE >17 W/O CC */
'026' = 21  /* SEIZURE & HEADACHE AGE 0-17 */
'027' = 22  /* TRAUMATIC STUPOR & COMA, COMA >1 HR */
'028' = 23  /* TRAUMATIC STUPOR & COMA, COMA <1 HR AGE >17 W CC */
'029' = 23  /* TRAUMATIC STUPOR & COMA, COMA <1 HR AGE >17 W/O CC */
'030' = 24  /* TRAUMATIC STUPOR & COMA, COMA <1 HR AGE 0-17 */
'031' = 25  /* CONCUSSION AGE >17 W CC */
'032' = 25  /* CONCUSSION AGE >17 W/O CC */
'033' = 26  /* CONCUSSION AGE 0-17 */
'034' = 27  /* OTHER DISORDERS OF NERVOUS SYSTEM W CC */
'035' = 27  /* OTHER DISORDERS OF NERVOUS SYSTEM W/O CC */
PSFMTS -- CONTINUED

'036' = 28  /* RETINAL PROCEDURES */
'037' = 29  /* ORBITAL PROCEDURES */
```

```
    '038' = 30  /* PRIMARY IRIS PROCEDURES */
    '039' = 31  /* LENS PROCEDURES WITH OR WITHOUT VITRECTOMY */
    '040' = 32  /* EXTRAOCULAR PROCEDURES EXCEPT ORBIT AGE >17 */
    '041' = 33  /* EXTRAOCULAR PROCEDURES EXCEPT ORBIT AGE 0-17 */
    '042' = 34  /* INTRAOCULAR PROCEDURES EXCEPT RETINA, IRIS & LENS */
    '043' = 35  /* HYPHEMA */
    '044' = 36  /* ACUTE MAJOR EYE INFECTIONS */
    '045' = 37  /* NEUROLOGICAL EYE DISORDERS */
    '046' = 38  /* OTHER DISORDERS OF THE EYE AGE >17 W CC */
    '047' = 38  /* OTHER DISORDERS OF THE EYE AGE >17 W/O CC */
    '048' = 39  /* OTHER DISORDERS OF THE EYE AGE 0-17 */
    '049' = 40  /* MAJOR HEAD & NECK PROCEDURES */
    '050' = 41  /* SIALOADENECTOMY */
    '051' = 42  /* SALIVARY GLAND PROCEDURES EXCEPT SIALOADENECTOMY */
    '052' = 43  /* CLEFT LIP & PALATE REPAIR */
    '053' = 44  /* SINUS & MASTOID PROCEDURES AGE >17 */
    '054' = 45  /* SINUS & MASTOID PROCEDURES AGE 0-17 */
    '055' = 46  /* MISCELLANEOUS EAR, NOSE, MOUTH & THROAT PROCEDURES */
    '056' = 47  /* RHINOPLASTY */
    '057' = 48  /* T&A PROC, EXCEPT TONSILLECTOMY &/OR ADENOIDECTOMY ONLY, AGE
>17 *
    /
    '058' = 49  /* T&A PROC, EXCEPT TONSILLECTOMY &/OR ADENOIDECTOMY ONLY, AGE
0-17
    */
    '059' = 50  /* TONSILLECTOMY &/OR ADENOIDECTOMY ONLY, AGE >17 */
    '060' = 51  /* TONSILLECTOMY &/OR ADENOIDECTOMY ONLY, AGE 0-17 */
    '061' = 52  /* MYRINGOTOMY W TUBE INSERTION AGE >17 */
    '062' = 53  /* MYRINGOTOMY W TUBE INSERTION AGE 0-17 */
    '063' = 54  /* OTHER EAR, NOSE, MOUTH & THROAT O.R. PROCEDURES */
    '064' = 55  /* EAR, NOSE, MOUTH & THROAT MALIGNANCY */
    '065' = 56  /* DYSEQUILIBRIUM */
    '066' = 57  /* EPISTAXIS */
    '067' = 58  /* EPIGLOTTITIS */
    '068' = 59  /* OTITIS MEDIA & URI AGE >17 W CC */
    '069' = 59  /* OTITIS MEDIA & URI AGE >17 W/O CC */
    '070' = 60  /* OTITIS MEDIA & URI AGE 0-17 */
    '071' = 61  /* LARYNGOTRACHEITIS */
    '072' = 62  /* NASAL TRAUMA & DEFORMITY */
    '073' = 63  /* OTHER EAR, NOSE, MOUTH & THROAT DIAGNOSES AGE >17 */
    '074' = 64  /* OTHER EAR, NOSE, MOUTH & THROAT DIAGNOSES AGE 0-17 */
    '075' = 65  /* MAJOR CHEST PROCEDURES */
    '076' = 66  /* OTHER RESP SYSTEM O.R. PROCEDURES W CC */
    '077' = 66  /* OTHER RESP SYSTEM O.R. PROCEDURES W/O CC */
    '078' = 67  /* PULMONARY EMBOLISM */
    '079' = 68  /* RESPIRATORY INFECTIONS & INFLAMMATIONS AGE >17 W CC */
    '080' = 68  /* RESPIRATORY INFECTIONS & INFLAMMATIONS AGE >17 W/O CC */
    '081' = 69  /* RESPIRATORY INFECTIONS & INFLAMMATIONS AGE 0-17 */
    '082' = 70  /* RESPIRATORY NEOPLASMS */
    '083' = 71  /* MAJOR CHEST TRAUMA W CC */
    '084' = 71  /* MAJOR CHEST TRAUMA W/O CC */
    '085' = 72  /* PLEURAL EFFUSION W CC */
    '086' = 72  /* PLEURAL EFFUSION W/O CC */
    '087' = 73  /* PULMONARY EDEMA & RESPIRATORY FAILURE */
    '088' = 74  /* CHRONIC OBSTRUCTIVE PULMONARY DISEASE */
    '089' = 75  /* SIMPLE PNEUMONIA & PLEURISY AGE >17 W CC */
    '090' = 75  /* SIMPLE PNEUMONIA & PLEURISY AGE >17 W/O CC */
```

PSFMTS -- CONTINUED

```
    '091' = 76  /* SIMPLE PNEUMONIA & PLEURISY AGE 0-17 */
    '092' = 77  /* INTERSTITIAL LUNG DISEASE W CC */
    '093' = 77  /* INTERSTITIAL LUNG DISEASE W/O CC */
    '094' = 78  /* PNEUMOTHORAX W CC */
    '095' = 78  /* PNEUMOTHORAX W/O CC */
    '096' = 79  /* BRONCHITIS & ASTHMA AGE >17 W CC */
    '097' = 79  /* BRONCHITIS & ASTHMA AGE >17 W/O CC */
    '098' = 80  /* BRONCHITIS & ASTHMA AGE 0-17 */
    '099' = 81  /* RESPIRATORY SIGNS & SYMPTOMS W CC */
    '100' = 81  /* RESPIRATORY SIGNS & SYMPTOMS W/O CC */
    '101' = 82  /* OTHER RESPIRATORY SYSTEM DIAGNOSES W CC */
    '102' = 82  /* OTHER RESPIRATORY SYSTEM DIAGNOSES W/O CC */
    '103' = 83  /* HEART TRANSPLANT */
    '104' = 84  /* CARDIAC VALVE & OTHER MAJOR CARDIOTHORACIC PROC W CARDIAC
CATH */
    '105' = 85  /* CARDIAC VALVE & OTHER MAJOR CARDIOTHORACIC PROC W/O CARDIAC
CATH
    */
    '106' = 86  /* CORONARY BYPASS W PTCA */
    '107' = 87  /* CORONARY BYPASS W CARDIAC CATH */
    '108' = 88  /* OTHER CARDIOTHORACIC PROCEDURES */
    '109' = 87  /* CORONARY BYPASS W/O CARDIAC CATH */
    '110' = 89  /* MAJOR CARDIOVASCULAR PROCEDURES W CC */
    '111' = 89  /* MAJOR CARDIOVASCULAR PROCEDURES W/O CC */
    '112' = 90  /* PERCUTANEOUS CARDIOVASCULAR PROCEDURES */
    '113' = 91  /* AMPUTATION FOR CIRC SYSTEM DISORDERS EXCEPT UPPER LIMB &
TOE */
    '114' = 92  /* UPPER LIMB & TOE AMPUTATION FOR CIRC SYSTEM DISORDERS */
    '115' = 93  /* PRM CARD PACEM IMPL W AMI,HRT FAIL OR SHK,OR AICD LEAD OR
GNRTR P
    ROC */
    '116' = 94  /* OTH PERM CARD PACEMAK IMPL OR PTCA W CORONARY ARTERY STENT
IMPLNT
     */
    '117' = 95  /* CARDIAC PACEMAKER REVISION EXCEPT DEVICE REPLACEMENT */
    '118' = 96  /* CARDIAC PACEMAKER DEVICE REPLACEMENT */
    '119' = 97  /* VEIN LIGATION & STRIPPING */
    '120' = 98  /* OTHER CIRCULATORY SYSTEM O.R. PROCEDURES */
    '121' = 99  /* CIRCULATORY DISORDERS W AMI & MAJOR COMP, DISCHARGED ALIVE
*/
    '122' = 99  /* CIRCULATORY DISORDERS W AMI W/O MAJOR COMP, DISCHARGED
ALIVE */
    '123' = 100 /* CIRCULATORY DISORDERS W AMI, EXPIRED */
    '124' = 101 /* CIRCULATORY DISORDERS EXCEPT AMI, W CARD CATH & COMPLEX
DIAG */
    '125' = 101 /* CIRCULATORY DISORDERS EXCEPT AMI, W CARD CATH W/O COMPLEX
DIAG */
    '126' = 102 /* ACUTE & SUBACUTE ENDOCARDITIS */
    '127' = 103 /* HEART FAILURE & SHOCK */
    '128' = 104 /* DEEP VEIN THROMBOPHLEBITIS */
    '129' = 105 /* CARDIAC ARREST, UNEXPLAINED */
    '130' = 106 /* PERIPHERAL VASCULAR DISORDERS W CC */
    '131' = 106 /* PERIPHERAL VASCULAR DISORDERS W/O CC */
    '132' = 107 /* ATHEROSCLEROSIS W CC */
    '133' = 107 /* ATHEROSCLEROSIS W/O CC */
    '134' = 108 /* HYPERTENSION */
    '135' = 109 /* CARDIAC CONGENITAL & VALVULAR DISORDERS AGE >17 W CC */
```

```
    '136' = 109 /* CARDIAC CONGENITAL & VALVULAR DISORDERS AGE >17 W/O CC */
    '137' = 110 /* CARDIAC CONGENITAL & VALVULAR DISORDERS AGE 0-17 */
```
PSFMTS -- CONTINUED

```
    '138' = 111 /* CARDIAC ARRHYTHMIA & CONDUCTION DISORDERS W CC */
    '139' = 111 /* CARDIAC ARRHYTHMIA & CONDUCTION DISORDERS W/O CC */
    '140' = 112 /* ANGINA PECTORIS */
    '141' = 113 /* SYNCOPE & COLLAPSE W CC */
    '142' = 113 /* SYNCOPE & COLLAPSE W/O CC */
    '143' = 114 /* CHEST PAIN */
    '144' = 115 /* OTHER CIRCULATORY SYSTEM DIAGNOSES W CC */
    '145' = 115 /* OTHER CIRCULATORY SYSTEM DIAGNOSES W/O CC */
    '146' = 116 /* RECTAL RESECTION W CC */
    '147' = 116 /* RECTAL RESECTION W/O CC */
    '148' = 117 /* MAJOR SMALL & LARGE BOWEL PROCEDURES W CC */
    '149' = 117 /* MAJOR SMALL & LARGE BOWEL PROCEDURES W/O CC */
    '150' = 118 /* PERITONEAL ADHESIOLYSIS W CC */
    '151' = 118 /* PERITONEAL ADHESIOLYSIS W/O CC */
    '152' = 119 /* MINOR SMALL & LARGE BOWEL PROCEDURES W CC */
    '153' = 119 /* MINOR SMALL & LARGE BOWEL PROCEDURES W/O CC */
    '154' = 120 /* STOMACH, ESOPHAGEAL & DUODENAL PROCEDURES AGE >17 W CC */
    '155' = 120 /* STOMACH, ESOPHAGEAL & DUODENAL PROCEDURES AGE >17 W/O CC */
    '156' = 121 /* STOMACH, ESOPHAGEAL & DUODENAL PROCEDURES AGE 0-17 */
    '157' = 122 /* ANAL & STOMAL PROCEDURES W CC */
    '158' = 122 /* ANAL & STOMAL PROCEDURES W/O CC */
    '159' = 123 /* HERNIA PROCEDURES EXCEPT INGUINAL & FEMORAL AGE >17 W CC */
    '160' = 123 /* HERNIA PROCEDURES EXCEPT INGUINAL & FEMORAL AGE >17 W/O CC
*/
    '161' = 124 /* INGUINAL & FEMORAL HERNIA PROCEDURES AGE >17 W CC */
    '162' = 124 /* INGUINAL & FEMORAL HERNIA PROCEDURES AGE >17 W/O CC */
    '163' = 125 /* HERNIA PROCEDURES AGE 0-17 */
    '164' = 126 /* APPENDECTOMY W COMPLICATED PRINCIPAL DIAG W CC */
    '165' = 126 /* APPENDECTOMY W COMPLICATED PRINCIPAL DIAG W/O CC */
    '166' = 127 /* APPENDECTOMY W/O COMPLICATED PRINCIPAL DIAG W CC */
    '167' = 127 /* APPENDECTOMY W/O COMPLICATED PRINCIPAL DIAG W/O CC */
    '168' = 128 /* MOUTH PROCEDURES W CC */
    '169' = 128 /* MOUTH PROCEDURES W/O CC */
    '170' = 129 /* OTHER DIGESTIVE SYSTEM O.R. PROCEDURES W CC */
    '171' = 129 /* OTHER DIGESTIVE SYSTEM O.R. PROCEDURES W/O CC */
    '172' = 130 /* DIGESTIVE MALIGNANCY W CC */
    '173' = 130 /* DIGESTIVE MALIGNANCY W/O CC */
    '174' = 131 /* G.I. HEMORRHAGE W CC */
    '175' = 131 /* G.I. HEMORRHAGE W/O CC */
    '176' = 132 /* COMPLICATED PEPTIC ULCER */
    '177' = 133 /* UNCOMPLICATED PEPTIC ULCER W CC */
    '178' = 133 /* UNCOMPLICATED PEPTIC ULCER W/O CC */
    '179' = 134 /* INFLAMMATORY BOWEL DISEASE */
    '180' = 135 /* G.I. OBSTRUCTION W CC */
    '181' = 135 /* G.I. OBSTRUCTION W/O CC */
    '182' = 136 /* ESOPHAGITIS, GASTROENT & MISC DIGEST DISORDERS AGE >17 W CC
*/
    '183' = 136 /* ESOPHAGITIS, GASTROENT & MISC DIGEST DISORDERS AGE >17 W/O
CC */
    '184' = 137 /* ESOPHAGITIS, GASTROENT & MISC DIGEST DISORDERS AGE 0-17 */
    '185' = 138 /* DENTAL & ORAL DIS EXCEPT EXTRACTIONS & RESTORATIONS, AGE
>17 */
    '186' = 139 /* DENTAL & ORAL DIS EXCEPT EXTRACTIONS & RESTORATIONS, AGE 0-
17 */
```

```
    '187' = 140 /* DENTAL EXTRACTIONS & RESTORATIONS */
    '188' = 141 /* OTHER DIGESTIVE SYSTEM DIAGNOSES AGE >17 W CC */
    '189' = 141 /* OTHER DIGESTIVE SYSTEM DIAGNOSES AGE >17 W/O CC */
    '190' = 142 /* OTHER DIGESTIVE SYSTEM DIAGNOSES AGE 0-17 */
    '191' = 143 /* PANCREAS, LIVER & SHUNT PROCEDURES W CC */
```
PSFMTS -- CONTINUED

```
    '192' = 143 /* PANCREAS, LIVER & SHUNT PROCEDURES W/O CC */
    '193' = 144 /* BILIARY TRACT PROC EXCEPT ONLY CHOLECYST W OR W/O C.D.E. W
CC */
    '194' = 144 /* BILIARY TRACT PROC EXCEPT ONLY CHOLECYST W OR W/O C.D.E.
W/O CC *
    /
    '195' = 145 /* CHOLECYSTECTOMY W C.D.E. W CC */
    '196' = 145 /* CHOLECYSTECTOMY W C.D.E. W/O CC */
    '197' = 146 /* CHOLECYSTECTOMY EXCEPT BY LAPAROSCOPE W/O C.D.E. W CC */
    '198' = 146 /* CHOLECYSTECTOMY EXCEPT BY LAPAROSCOPE W/O C.D.E. W/O CC */
    '199' = 147 /* HEPATOBILIARY DIAGNOSTIC PROCEDURE FOR MALIGNANCY */
    '200' = 148 /* HEPATOBILIARY DIAGNOSTIC PROCEDURE FOR NON-MALIGNANCY */
    '201' = 149 /* OTHER HEPATOBILIARY OR PANCREAS O.R. PROCEDURES */
    '202' = 150 /* CIRRHOSIS & ALCOHOLIC HEPATITIS */
    '203' = 151 /* MALIGNANCY OF HEPATOBILIARY SYSTEM OR PANCREAS */
    '204' = 152 /* DISORDERS OF PANCREAS EXCEPT MALIGNANCY */
    '205' = 153 /* DISORDERS OF LIVER EXCEPT MALIG,CIRR,ALC HEPA W CC */
    '206' = 153 /* DISORDERS OF LIVER EXCEPT MALIG,CIRR,ALC HEPA W/O CC */
    '207' = 154 /* DISORDERS OF THE BILIARY TRACT W CC */
    '208' = 154 /* DISORDERS OF THE BILIARY TRACT W/O CC */
    '209' = 155 /* MAJOR JOINT & LIMB REATTACHMENT PROCEDURES OF LOWER
EXTREMITY */
    '210' = 156 /* HIP & FEMUR PROCEDURES EXCEPT MAJOR JOINT AGE >17 W CC */
    '211' = 156 /* HIP & FEMUR PROCEDURES EXCEPT MAJOR JOINT AGE >17 W/O CC */
    '212' = 157 /* HIP & FEMUR PROCEDURES EXCEPT MAJOR JOINT AGE 0-17 */
    '213' = 158 /* AMPUTATION FOR MUSCULOSKELETAL SYSTEM & CONN TISSUE
DISORDERS */
    '214' = 0   /* INVALID */
    '215' = 0   /* INVALID */
    '216' = 159 /* BIOPSIES OF MUSCULOSKELETAL SYSTEM & CONNECTIVE TISSUE */
    '217' = 160 /* WND DEBRID & SKN GRFT EXCEPT HAND,FOR MUSCSKELET & CONN
TISS DIS
    */
    '218' = 161 /* LOWER EXTREM & HUMER PROC EXCEPT HIP,FOOT,FEMUR AGE >17 W
CC */
    '219' = 161 /* LOWER EXTREM & HUMER PROC EXCEPT HIP,FOOT,FEMUR AGE >17 W/O
CC */
    '220' = 162 /* LOWER EXTREM & HUMER PROC EXCEPT HIP,FOOT,FEMUR AGE 0-17 */
    '221' = 0   /* INVALID */
    '222' = 0   /* INVALID */
    '223' = 163 /* MAJOR SHOULDER/ELBOW PROC, OR OTHER UPPER EXTREMITY PROC W
CC */
    '224' = 163 /* SHOULDER,ELBOW OR FOREARM PROC,EXC MAJOR JOINT PROC, W/O CC
*/
    '225' = 164 /* FOOT PROCEDURES */
    '226' = 165 /* SOFT TISSUE PROCEDURES W CC */
    '227' = 165 /* SOFT TISSUE PROCEDURES W/O CC */
    '228' = 166 /* MAJOR THUMB OR JOINT PROC,OR OTH HAND OR WRIST PROC W CC */
    '229' = 166 /* HAND OR WRIST PROC, EXCEPT MAJOR JOINT PROC, W/O CC */
    '230' = 167 /* LOCAL EXCISION & REMOVAL OF INT FIX DEVICES OF HIP & FEMUR
*/
```

```
    '231' = 168 /* LOCAL EXCISION & REMOVAL OF INT FIX DEVICES EXCEPT HIP &
FEMUR */
    '232' = 169 /* ARTHROSCOPY */
    '233' = 170 /* OTHER MUSCULOSKELET SYS & CONN TISS O.R. PROC W CC */
    '234' = 170 /* OTHER MUSCULOSKELET SYS & CONN TISS O.R. PROC W/O CC */
    '235' = 171 /* FRACTURES OF FEMUR */
    '236' = 172 /* FRACTURES OF HIP & PELVIS */
    '237' = 173 /* SPRAINS, STRAINS, & DISLOCATIONS OF HIP, PELVIS & THIGH */
```

PSFMTS -- CONTINUED

```
    '238' = 174 /* OSTEOMYELITIS */
    '239' = 175 /* PATHOLOGICAL FRACTURES & MUSCULOSKELETAL & CONN TISS
MALIGNANCY *
    /
    '240' = 176 /* CONNECTIVE TISSUE DISORDERS W CC */
    '241' = 176 /* CONNECTIVE TISSUE DISORDERS W/O CC */
    '242' = 177 /* SEPTIC ARTHRITIS */
    '243' = 178 /* MEDICAL BACK PROBLEMS */
    '244' = 179 /* BONE DISEASES & SPECIFIC ARTHROPATHIES W CC */
    '245' = 179 /* BONE DISEASES & SPECIFIC ARTHROPATHIES W/O CC */
    '246' = 180 /* NON-SPECIFIC ARTHROPATHIES */
    '247' = 181 /* SIGNS & SYMPTOMS OF MUSCULOSKELETAL SYSTEM & CONN TISSUE */
    '248' = 182 /* TENDONITIS, MYOSITIS & BURSITIS */
    '249' = 183 /* AFTERCARE, MUSCULOSKELETAL SYSTEM & CONNECTIVE TISSUE */
    '250' = 184 /* FX, SPRN, STRN & DISL OF FOREARM, HAND, FOOT AGE >17 W CC
*/
    '251' = 184 /* FX, SPRN, STRN & DISL OF FOREARM, HAND, FOOT AGE >17 W/O CC
*/
    '252' = 185 /* FX, SPRN, STRN & DISL OF FOREARM, HAND, FOOT AGE 0-17 */
    '253' = 186 /* FX, SPRN, STRN & DISL OF UPARM,LOWLEG EX FOOT AGE >17 W CC
*/
    '254' = 186 /* FX, SPRN, STRN & DISL OF UPARM,LOWLEG EX FOOT AGE >17 W/O
CC */
    '255' = 187 /* FX, SPRN, STRN & DISL OF UPARM,LOWLEG EX FOOT AGE 0-17 */
    '256' = 188 /* OTHER MUSCULOSKELETAL SYSTEM & CONNECTIVE TISSUE DIAGNOSES
*/
    '257' = 189 /* TOTAL MASTECTOMY FOR MALIGNANCY W CC */
    '258' = 189 /* TOTAL MASTECTOMY FOR MALIGNANCY W/O CC */
    '259' = 190 /* SUBTOTAL MASTECTOMY FOR MALIGNANCY W CC */
    '260' = 190 /* SUBTOTAL MASTECTOMY FOR MALIGNANCY W/O CC */
    '261' = 191 /* BREAST PROC FOR NON-MALIGNANCY EXCEPT BIOPSY & LOCAL
EXCISION */
    '262' = 192 /* BREAST BIOPSY & LOCAL EXCISION FOR NON-MALIGNANCY */
    '263' = 193 /* SKIN GRAFT &/OR DEBRID FOR SKN ULCER OR CELLULITIS W CC */
    '264' = 193 /* SKIN GRAFT &/OR DEBRID FOR SKN ULCER OR CELLULITIS W/O CC
*/
    '265' = 194 /* SKIN GRAFT &/OR DEBRID EXCEPT FOR SKIN ULCER OR CELLULITIS
W CC *
    /
    '266' = 194 /* SKIN GRAFT &/OR DEBRID EXCEPT FOR SKIN ULCER OR CELLULITIS
W/O CC
     */
    '267' = 195 /* PERIANAL & PILONIDAL PROCEDURES */
    '268' = 196 /* SKIN, SUBCUTANEOUS TISSUE & BREAST PLASTIC PROCEDURES */
    '269' = 197 /* OTHER SKIN, SUBCUT TISS & BREAST PROC W CC */
    '270' = 197 /* OTHER SKIN, SUBCUT TISS & BREAST PROC W/O CC */
    '271' = 198 /* SKIN ULCERS */
    '272' = 199 /* MAJOR SKIN DISORDERS W CC */
```

```
    '273' = 199 /* MAJOR SKIN DISORDERS W/O CC */
    '274' = 200 /* MALIGNANT BREAST DISORDERS W CC */
    '275' = 200 /* MALIGNANT BREAST DISORDERS W/O CC */
    '276' = 201 /* NON-MALIGANT BREAST DISORDERS */
    '277' = 202 /* CELLULITIS AGE >17 W CC */
    '278' = 202 /* CELLULITIS AGE >17 W/O CC */
    '279' = 203 /* CELLULITIS AGE 0-17 */
    '280' = 204 /* TRAUMA TO THE SKIN, SUBCUT TISS & BREAST AGE >17 W CC */
    '281' = 204 /* TRAUMA TO THE SKIN, SUBCUT TISS & BREAST AGE >17 W/O CC */
    '282' = 205 /* TRAUMA TO THE SKIN, SUBCUT TISS & BREAST AGE 0-17 */
    '283' = 206 /* MINOR SKIN DISORDERS W CC */
```
PSFMTS -- CONTINUED

```
    '284' = 206 /* MINOR SKIN DISORDERS W/O CC */
    '285' = 207 /* AMPUTAT OF LOWER LIMB FOR ENDOCRINE,NUTRIT,& METABOL
DISORDERS */
    '286' = 208 /* ADRENAL & PITUITARY PROCEDURES */
    '287' = 209 /* SKIN GRAFTS & WOUND DEBRID FOR ENDOC, NUTRIT & METAB
DISORDERS */
    '288' = 210 /* O.R. PROCEDURES FOR OBESITY */
    '289' = 211 /* PARATHYROID PROCEDURES */
    '290' = 212 /* THYROID PROCEDURES */
    '291' = 213 /* THYROGLOSSAL PROCEDURES */
    '292' = 214 /* OTHER ENDOCRINE, NUTRIT & METAB O.R. PROC W CC */
    '293' = 214 /* OTHER ENDOCRINE, NUTRIT & METAB O.R. PROC W/O CC */
    '294' = 215 /* DIABETES AGE >35 */
    '295' = 216 /* DIABETES AGE 0-35 */
    '296' = 217 /* NUTRITIONAL & MISC METABOLIC DISORDERS AGE >17 W CC */
    '297' = 217 /* NUTRITIONAL & MISC METABOLIC DISORDERS AGE >17 W/O CC */
    '298' = 218 /* NUTRITIONAL & MISC METABOLIC DISORDERS AGE 0-17 */
    '299' = 219 /* INBORN ERRORS OF METABOLISM */
    '300' = 220 /* ENDOCRINE DISORDERS W CC */
    '301' = 220 /* ENDOCRINE DISORDERS W/O CC */
    '302' = 221 /* KIDNEY TRANSPLANT */
    '303' = 222 /* KIDNEY,URETER & MAJOR BLADDER PROCEDURES FOR NEOPLASM */
    '304' = 223 /* KIDNEY,URETER & MAJOR BLADDER PROC FOR NON-NEOPL W CC */
    '305' = 223 /* KIDNEY,URETER & MAJOR BLADDER PROC FOR NON-NEOPL W/O CC */
    '306' = 224 /* PROSTATECTOMY W CC */
    '307' = 224 /* PROSTATECTOMY W/O CC */
    '308' = 225 /* MINOR BLADDER PROCEDURES W CC */
    '309' = 225 /* MINOR BLADDER PROCEDURES W/O CC */
    '310' = 226 /* TRANSURETHRAL PROCEDURES W CC */
    '311' = 226 /* TRANSURETHRAL PROCEDURES W/O CC */
    '312' = 227 /* URETHRAL PROCEDURES, AGE >17 W CC */
    '313' = 227 /* URETHRAL PROCEDURES, AGE >17 W/O CC */
    '314' = 228 /* URETHRAL PROCEDURES, AGE 0-17 */
    '315' = 229 /* OTHER KIDNEY & URINARY TRACT O.R. PROCEDURES */
    '316' = 230 /* RENAL FAILURE */
    '317' = 231 /* ADMIT FOR RENAL DIALYSIS */
    '318' = 232 /* KIDNEY & URINARY TRACT NEOPLASMS W CC */
    '319' = 232 /* KIDNEY & URINARY TRACT NEOPLASMS W/O CC */
    '320' = 233 /* KIDNEY & URINARY TRACT INFECTIONS AGE >17 W CC */
    '321' = 233 /* KIDNEY & URINARY TRACT INFECTIONS AGE >17 W/O CC */
    '322' = 234 /* KIDNEY & URINARY TRACT INFECTIONS AGE 0-17 */
    '323' = 235 /* URINARY STONES W CC, &/OR ESW LITHOTRIPSY */
    '324' = 235 /* URINARY STONES W/O CC */
    '325' = 236 /* KIDNEY & URINARY TRACT SIGNS & SYMPTOMS AGE >17 W CC */
    '326' = 236 /* KIDNEY & URINARY TRACT SIGNS & SYMPTOMS AGE >17 W/O CC */
```

```
    '327' = 237 /* KIDNEY & URINARY TRACT SIGNS & SYMPTOMS AGE 0-17 */
    '328' = 238 /* URETHRAL STRICTURE AGE >17 W CC */
    '329' = 238 /* URETHRAL STRICTURE AGE >17 W/O CC */
    '330' = 239 /* URETHRAL STRICTURE AGE 0-17 */
    '331' = 240 /* OTHER KIDNEY & URINARY TRACT DIAGNOSES AGE >17 W CC */
    '332' = 240 /* OTHER KIDNEY & URINARY TRACT DIAGNOSES AGE >17 W/O CC */
    '333' = 241 /* OTHER KIDNEY & URINARY TRACT DIAGNOSES AGE 0-17 */
    '334' = 242 /* MAJOR MALE PELVIC PROCEDURES W CC */
    '335' = 242 /* MAJOR MALE PELVIC PROCEDURES W/O CC */
    '336' = 243 /* TRANSURETHRAL PROSTATECTOMY W CC */
    '337' = 243 /* TRANSURETHRAL PROSTATECTOMY W/O CC */
    '338' = 244 /* TESTES PROCEDURES, FOR MALIGNANCY */
    '339' = 245 /* TESTES PROCEDURES, NON-MALIGNANCY AGE >17 */
    '340' = 246 /* TESTES PROCEDURES, NON-MALIGNANCY AGE 0-17 */
```
PSFMTS -- CONTINUED

```
    '341' = 247 /* PENIS PROCEDURES */
    '342' = 248 /* CIRCUMCISION AGE >17 */
    '343' = 249 /* CIRCUMCISION AGE 0-17 */
    '344' = 250 /* OTHER MALE REPRODUCTIVE SYSTEM O.R. PROCEDURES FOR
MALIGNANCY */
    '345' = 251 /* OTHER MALE REPRODUCTIVE SYSTEM O.R. PROC EXCEPT FOR
MALIGNANCY */
    '346' = 252 /* MALIGNANCY, MALE REPRODUCTIVE SYSTEM, W CC */
    '347' = 252 /* MALIGNANCY, MALE REPRODUCTIVE SYSTEM, W/O CC */
    '348' = 253 /* BENIGN PROSTATIC HYPERTROPHY W CC */
    '349' = 253 /* BENIGN PROSTATIC HYPERTROPHY W/O CC */
    '350' = 254 /* INFLAMMATION OF THE MALE REPRODUCTIVE SYSTEM */
    '351' = 255 /* STERILIZATION, MALE */
    '352' = 256 /* OTHER MALE REPRODUCTIVE SYSTEM DIAGNOSES */
    '353' = 257 /* PELVIC EVISCERATION, RADICAL HYSTERECTOMY & RADICAL
VULVECTOMY */
    '354' = 258 /* UTERINE,ADNEXA PROC FOR NON-OVARIAN/ADNEXAL MALIG W CC */
    '355' = 258 /* UTERINE,ADNEXA PROC FOR NON-OVARIAN/ADNEXAL MALIG W/O CC */
    '356' = 259 /* FEMALE REPRODUCTIVE SYSTEM RECONSTRUCTIVE PROCEDURES */
    '357' = 260 /* UTERINE & ADNEXA PROC FOR OVARIAN OR ADNEXAL MALIGNANCY */
    '358' = 261 /* UTERINE & ADNEXA PROC FOR NON-MALIGNANCY W CC */
    '359' = 261 /* UTERINE & ADNEXA PROC FOR NON-MALIGNANCY W/O CC */
    '360' = 262 /* VAGINA, CERVIX & VULVA PROCEDURES */
    '361' = 263 /* LAPAROSCOPY & INCISIONAL TUBAL INTERRUPTION */
    '362' = 264 /* ENDOSCOPIC TUBAL INTERRUPTION */
    '363' = 265 /* D&C, CONIZATION & RADIO-IMPLANT, FOR MALIGNANCY */
    '364' = 266 /* D&C, CONIZATION EXCEPT FOR MALIGNANCY */
    '365' = 267 /* OTHER FEMALE REPRODUCTIVE SYSTEM O.R. PROCEDURES */
    '366' = 268 /* MALIGNANCY, FEMALE REPRODUCTIVE SYSTEM W CC */
    '367' = 268 /* MALIGNANCY, FEMALE REPRODUCTIVE SYSTEM W/O CC */
    '368' = 269 /* INFECTIONS, FEMALE REPRODUCTIVE SYSTEM */
    '369' = 270 /* MENSTRUAL & OTHER FEMALE REPRODUCTIVE SYSTEM DISORDERS */
    '370' = 271 /* CESAREAN SECTION W CC */
    '371' = 271 /* CESAREAN SECTION W/O CC */
    '372' = 272 /* VAGINAL DELIVERY W COMPLICATING DIAGNOSES */
    '373' = 273 /* VAGINAL DELIVERY W/O COMPLICATING DIAGNOSES */
    '374' = 274 /* VAGINAL DELIVERY W STERILIZATION &/OR D&C */
    '375' = 275 /* VAGINAL DELIVERY W O.R. PROC EXCEPT STERIL &/OR D&C */
    '376' = 276 /* POSTPARTUM & POST ABORTION DIAGNOSES W/O O.R. PROCEDURE */
    '377' = 277 /* POSTPARTUM & POST ABORTION DIAGNOSES W O.R. PROCEDURE */
    '378' = 278 /* ECTOPIC PREGNANCY */
    '379' = 279 /* THREATENED ABORTION */
```

```
    '380' = 280 /* ABORTION W/O D&C */
    '381' = 281 /* ABORTION W D&C, ASPIRATION CURETTAGE OR HYSTEROTOMY */
    '382' = 282 /* FALSE LABOR */
    '383' = 283 /* OTHER ANTEPARTUM DIAGNOSES W MEDICAL COMPLICATIONS */
    '384' = 284 /* OTHER ANTEPARTUM DIAGNOSES W/O MEDICAL COMPLICATIONS */
    '385' = 285 /* NEONATES, DIED OR TRANSFERRED TO ANOTHER ACUTE CARE
FACILITY */
    '386' = 286 /* EXTREME IMMATURITY OR RESPIRATORY DISTRESS SYNDROME,
NEONATE */
    '387' = 287 /* PREMATURITY W MAJOR PROBLEMS */
    '388' = 288 /* PREMATURITY W/O MAJOR PROBLEMS */
    '389' = 289 /* FULL TERM NEONATE W MAJOR PROBLEMS */
    '390' = 290 /* NEONATE W OTHER SIGNIFICANT PROBLEMS */
    '391' = 291 /* NORMAL NEWBORN */
    '392' = 292 /* SPLENECTOMY AGE >17 */
    '393' = 293 /* SPLENECTOMY AGE 0-17 */
```

PSFMTS -- CONTINUED

```
    '394' = 294 /* OTHER O.R. PROCEDURES OF THE BLOOD AND BLOOD FORMING ORGANS
*/
    '395' = 295 /* RED BLOOD CELL DISORDERS AGE >17 */
    '396' = 296 /* RED BLOOD CELL DISORDERS AGE 0-17 */
    '397' = 297 /* COAGULATION DISORDERS */
    '398' = 298 /* RETICULOENDOTHELIAL & IMMUNITY DISORDERS W CC */
    '399' = 298 /* RETICULOENDOTHELIAL & IMMUNITY DISORDERS W/O CC */
    '400' = 299 /* LYMPHOMA & LEUKEMIA W MAJOR O.R. PROCEDURE */
    '401' = 300 /* LYMPHOMA & NON-ACUTE LEUKEMIA W OTHER O.R. PROC W CC */
    '402' = 300 /* LYMPHOMA & NON-ACUTE LEUKEMIA W OTHER O.R. PROC W/O CC */
    '403' = 301 /* LYMPHOMA & NON-ACUTE LEUKEMIA W CC */
    '404' = 301 /* LYMPHOMA & NON-ACUTE LEUKEMIA W/O CC */
    '405' = 302 /* ACUTE LEUKEMIA W/O MAJOR O.R. PROCEDURE AGE 0-17 */
    '406' = 303 /* MYELOPROLIF DISORD OR POORLY DIFF NEOPL W MAJ O.R.PROC W CC
*/
    '407' = 303 /* MYELOPROLIF DISORD OR POORLY DIFF NEOPL W MAJ O.R.PROC W/O
CC */
    '408' = 304 /* MYELOPROLIF DISORD OR POORLY DIFF NEOPL W OTHER O.R.PROC */
    '409' = 305 /* RADIOTHERAPY */
    '410' = 306 /* CHEMOTHERAPY W/O ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS */
    '411' = 307 /* HISTORY OF MALIGNANCY W/O ENDOSCOPY */
    '412' = 308 /* HISTORY OF MALIGNANCY W ENDOSCOPY */
    '413' = 309 /* OTHER MYELOPROLIF DIS OR POORLY DIFF NEOPL DIAG W CC */
    '414' = 309 /* OTHER MYELOPROLIF DIS OR POORLY DIFF NEOPL DIAG W/O CC */
    '415' = 310 /* O.R. PROCEDURE FOR INFECTIOUS & PARASITIC DISEASES */
    '416' = 311 /* SEPTICEMIA AGE >17 */
    '417' = 312 /* SEPTICEMIA AGE 0-17 */
    '418' = 313 /* POSTOPERATIVE & POST-TRAUMATIC INFECTIONS */
    '419' = 314 /* FEVER OF UNKNOWN ORIGIN AGE >17 W CC */
    '420' = 314 /* FEVER OF UNKNOWN ORIGIN AGE >17 W/O CC */
    '421' = 315 /* VIRAL ILLNESS AGE >17 */
    '422' = 316 /* VIRAL ILLNESS & FEVER OF UNKNOWN ORIGIN AGE 0-17 */
    '423' = 317 /* OTHER INFECTIOUS & PARASITIC DISEASES DIAGNOSES */
    '424' = 318 /* O.R. PROCEDURE W PRINCIPAL DIAGNOSES OF MENTAL ILLNESS */
    '425' = 319 /* ACUTE ADJUSTMENT REACTION & PSYCHOSOCIAL DYSFUNCTION */
    '426' = 320 /* DEPRESSIVE NEUROSES */
    '427' = 321 /* NEUROSES EXCEPT DEPRESSIVE */
    '428' = 322 /* DISORDERS OF PERSONALITY & IMPULSE CONTROL */
    '429' = 323 /* ORGANIC DISTURBANCES & MENTAL RETARDATION */
    '430' = 324 /* PSYCHOSES */
```

```
    '431' = 325 /* CHILDHOOD MENTAL DISORDERS */
    '432' = 326 /* OTHER MENTAL DISORDER DIAGNOSES */
    '433' = 327 /* ALCOHOL/DRUG ABUSE OR DEPENDENCE, LEFT AMA */
    '434' = 328 /* ALC/DRUG ABUSE OR DEPEND, DETOX OR OTH SYMPT TREAT W CC */
    '435' = 328 /* ALC/DRUG ABUSE OR DEPEND, DETOX OR OTH SYMPT TREAT W/O CC
*/
    '436' = 329 /* ALC/DRUG DEPENDENCE W REHABILITATION THERAPY */
    '437' = 330 /* ALC/DRUG DEPENDENCE, COMBINED REHAB & DETOX THERAPY */
    '438' = 0   /* INVALID */
    '439' = 331 /* SKIN GRAFTS FOR INJURIES */
    '440' = 332 /* WOUND DEBRIDEMENTS FOR INJURIES */
    '441' = 333 /* HAND PROCEDURES FOR INJURIES */
    '442' = 334 /* OTHER O.R. PROCEDURES FOR INJURIES W CC */
    '443' = 334 /* OTHER O.R. PROCEDURES FOR INJURIES W/O CC */
    '444' = 335 /* TRAUMATIC INJURY AGE >17 W CC */
    '445' = 335 /* TRAUMATIC INJURY AGE >17 W/O CC */
    '446' = 336 /* TRAUMATIC INJURY AGE 0-17 */
    '447' = 337 /* ALLERGIC REACTIONS AGE >17 */
    '448' = 338 /* ALLERGIC REACTIONS AGE 0-17 */
    '449' = 339 /* POISONING & TOXIC EFFECTS OF DRUGS AGE >17 W CC */
```

PSFMTS -- CONTINUED

```
    '450' = 339 /* POISONING & TOXIC EFFECTS OF DRUGS AGE >17 W/O CC */
    '451' = 340 /* POISONING & TOXIC EFFECTS OF DRUGS AGE 0-17 */
    '452' = 341 /* COMPLICATIONS OF TREATMENT W CC */
    '453' = 341 /* COMPLICATIONS OF TREATMENT W/O CC */
    '454' = 342 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W CC */
    '455' = 342 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '456' = 343 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '457' = 344 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '458' = 345 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '459' = 346 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '460' = 347 /* OTHER INJURY, POISONING & TOXIC EFFECT DIAG W/O CC */
    '461' = 348 /* O.R. PROC W DIAGNOSES OF OTHER CONTACT W HEALTH SERVICES */
    '462' = 349 /* REHABILITATION */
    '463' = 350 /* SIGNS & SYMPTOMS W CC */
    '464' = 350 /* SIGNS & SYMPTOMS W/O CC */
    '465' = 351 /* AFTERCARE W HISTORY OF MALIGNANCY AS SECONDARY DIAGNOSIS */
    '466' = 352 /* AFTERCARE W/O HISTORY OF MALIGNANCY AS SECONDARY DIAGNOSIS
*/
    '467' = 353 /* OTHER FACTORS INFLUENCING HEALTH STATUS */
    '468' = 0   /* EXTENSIVE O.R. PROCEDURE UNRELATED TO PRINCIPAL DIAGNOSIS
*/
    '469' = 0   /* PRINCIPAL DIAGNOSIS INVALID AS DISCHARGE DIAGNOSIS */
    '470' = 0   /* UNGROUPABLE */
    '471' = 354 /* BILATERAL OR MULTIPLE MAJOR JOINT PROCS OF LOWER EXTREMITY
*/
    '472' = 355 /* BILATERAL OR MULTIPLE MAJOR JOINT PROCS OF LOWER EXTREMITY
*/
    '473' = 356 /* ACUTE LEUKEMIA W/O MAJOR O.R. PROCEDURE AGE >17 */
    '474' = 0   /* INVALID */
    '475' = 357 /* RESPIRATORY SYSTEM DIAGNOSIS WITH VENTILATOR SUPPORT */
    '476' = 0   /* PROSTATIC O.R. PROCEDURE UNRELATED TO PRINCIPAL DIAGNOSIS
*/
    '477' = 0     /* NON-EXTENSIVE O.R. PROCEDURE UNRELATED TO PRINCIPAL
DIAGNOSIS */
    '478' = 358 /* OTHER VASCULAR PROCEDURES W CC */
    '479' = 358 /* OTHER VASCULAR PROCEDURES W/O CC */
```

```
    '480' = 374 /* LIVER TRANSPLANT */
    '481' = 374 /* BONE MARROW TRANSPLANT */
    '482' = 0   /* TRACHEOSTOMY FOR FACE,MOUTH & NECK DIAGNOSES */
    '483' = 0   /* TRACHEOSTOMY EXCEPT FOR FACE,MOUTH & NECK DIAGNOSES */
    '484' = 359 /* CRANIOTOMY FOR MULTIPLE SIGNIFICANT TRAUMA */
    '485'  =  360  /*  LIMB  REATTACHMENT,  HIP  AND  FEMUR  PROC  FOR  MULTIPLE
SIGNIFICANT TR
    AUMA */
    '486' = 361 /* OTHER O.R. PROCEDURES FOR MULTIPLE SIGNIFICANT TRAUMA */
    '487' = 362 /* OTHER MULTIPLE SIGNIFICANT TRAUMA */
    '488' = 363 /* HIV W EXTENSIVE O.R. PROCEDURE */
    '489' = 364 /* HIV W MAJOR RELATED CONDITION */
    '490' = 365 /* HIV W OR W/O OTHER RELATED CONDITION */
    '491'  =  366  /*  MAJOR  JOINT  &  LIMB  REATTACHMENT  PROCEDURES  OF  UPPER
EXTREMITY */
    '492' = 367 /* CHEMOTHERAPY W ACUTE LEUKEMIA AS SECONDARY DIAGNOSIS */
    '493' = 368 /* LAPAROSCOPIC CHOLECYSTECTOMY W/O C.D.E. W CC */
    '494' = 368 /* LAPAROSCOPIC CHOLECYSTECTOMY W/O C.D.E. W/O CC */
    '495' = 369 /* LUNG TRANSPLANT */
    '496' = 370 /* COMBINED ANTERIOR/POSTERIOR SPINAL FUSION */
    '497' = 371 /* SPINAL FUSION W CC */
    '498' = 371 /* SPINAL FUSION W/O CC */
    '499' = 372 /* BACK & NECK PROCEDURES EXCEPT SPINAL FUSION W CC */
    PSFMTS -- CONTINUED

    '500' = 372 /* BACK & NECK PROCEDURES EXCEPT SPINAL FUSION W/O CC */
    '501' = 373 /* KNEE PROCEDURES W PDX OF INFECTION W CC */
    '502' = 373 /* KNEE PROCEDURES W PDX OF INFECTION W/O CC */
    '503' = 374 /* KNEE PROCEDURES W/O PDX OF INFECTION */
    '504' = 0   /* EXTENSIVE 3RD DEGREE BURNS W SKIN GRAFT */
    '505' = 0   /* EXTENSIVE 3RD DEGREE BURNS W/O SKIN GRAFT */
    '506' = 0   /* FULL THICKNESS BURN W SKIN GRAFT OR INHAL INJ W CC OR SIG
TRAUMA
    */
    '507' = 0   /* FULL THICKNESS BURN W SKIN GRFT OR INHAL INJ W/O CC OR SIG
TRAUMA
     */
    '508' = 0   /* FULL THICKNESS BURN W/O SKIN GRFT OR INHAL INJ W CC OR SIG
TRAUMA
     */
    '509' = 0   /* FULL THICKNESS BURN W/O SKIN GRFT OR INH INJ W/O CC OR SIG
TRAUMA
     */
    '510' = 0   /* NON-EXTENSIVE BURNS W CC OR SIGNIFICANT TRAUMA */
    '511' = 0   /* NON-EXTENSIVE BURNS W/O CC OR SIGNIFICANT TRAUMA */
    OTHER   = 0;

     /* ------------------------------------------------------------- */
     /* ---   FORMATS FOR COMORBIDITY STRATIFIERS                 --- */
     /* ------------------------------------------------------------- */

       VALUE $RCOMFMT
          "39891",
          "4280 ",
          "4281 ",
          "4289 " = "CHF"   /* Congestive heart failure */
          "40201",
          "40211",
```

```
      "40291",
      "40401",
      "40411",
      "40491" = "HTNCHF"   /* Hypertension+Congestive heart failure */
      "40403",
      "40413",
      "40493" = "HCHFRF"   /* Hypertension+CHF+Renal Failure */
      "42610",
      "42611",
      "42613",
      "4262 ",
      "4263 ",
      "4264 ",
      "42650",
      "42651",
      "42652",
      "42653",
      "4266 ",
      "4267 ",
      "42681",
      "42689",
      "4270 ",
      "4272 ",
      "42731",
      "42760",
      "42761",
```

PSFMTS -- CONTINUED

```
      "42769",
      "42781",
      "42789",
      "4279 ",
      "7850 ",
      "V450 ",
      "V4500",
      "V4501",
      "V4502",
      "V4509",
      "V533 ",
      "V5331",
      "V5332",
      "V5339" = "ARYTHM" /* Cardiac arrhythmias */
      "09320",
      "09321",
      "09322",
      "09323",
      "09324",
      "3940 ",
      "3941 ",
      "3942 ",
      "3949 ",
      "3950 ",
      "3951 ",
      "3952 ",
      "3959 ",
      "3960 ",
      "3961 ",
      "3962 ",
```

```
      "3963 ",
      "3968 ",
      "3969 ",
      "3970 ",
      "3971 ",
      "4240 ",
      "4241 ",
      "4242 ",
      "4243 ",
      "42490",
      "42491",
      "42499",
      "7463 ",
      "7464 ",
      "7465 ",
      "7466 ",
      "V422 ",
      "V433 " = "VALVE"     /* Valvular disease */
      "4160 ",
      "4161 ",
      "4168 ",
      "4169 ",
      "4179 " = "PULMCIRC"  /* Pulmonary circulation disorder */
      "4400 ",
      "4401 ",
      "4402 ",
      "44020",
      "44021",
      "44022",
```

PSFMTS -- CONTINUED

```
      "44023",
      "44024",
      "44029",
      "44030",
      "44031",
      "44032",
      "4408 ",
      "4409 ",
      "44100",
      "44101",
      "44102",
      "44103",
      "4411 ",
      "4412 ",
      "4413 ",
      "4414 ",
      "4415 ",
      "4416 ",
      "4417 ",
      "4419 ",
      "4431 ",
      "44381",
      "44389",
      "4439 ",
      "4471 ",
      "5571 ",
      "5579 ",
```

```
      "V434 " = "PERIVASC"  /* Peripheral vascular disorder */
      "4011 ",
      "4019 ",
      "64200",
      "64201",
      "64202",
      "64203",
      "64204" = "HTN"       /* Hypertension uncomplicated */
      "4010 ",
      "40200",
      "40210",
      "40290",
      "40300",
      "40310",
      "40390",
      "40400",
      "40410",
      "40490",
      "40501",
      "40509",
      "40511",
      "40519",
      "40591",
      "40599",
      "64210",
      "64211",
      "64212",
      "64213",
      "64214",
      "64220",
      "64221",
      "64222",
```

PSFMTS -- CONTINUED

```
      "64223",
      "64224",
      "64270",
      "64271",
      "64272",
      "64273",
      "64274",
      "64290",
      "64291",
      "64292",
      "64293",
      "64294" = "HTNCX"     /* Hypertension, complicated */
      "3420 ",
      "34200",
      "34201",
      "34202",
      "3421 ",
      "34210",
      "34211",
      "34212",
      "3428 ",
      "34280",
      "34281",
      "34282",
```

```
      "3429 ",
      "34290",
      "34291",
      "34292",
      "3430 ",
      "3431 ",
      "3432 ",
      "3433 ",
      "3434 ",
      "3438 ",
      "3439 ",
      "3440 ",
      "34400",
      "34401",
      "34402",
      "34403",
      "34404",
      "34409",
      "3441 ",
      "3442 ",
      "3443 ",
      "34430",
      "34431",
      "34432",
      "3444 ",
      "34440",
      "34441",
      "34442",
      "3445 ",
      "3446 ",
      "34460",
      "34461",
      "3448 ",
      "34481",
      "34489",
```

PSFMTS -- CONTINUED

```
      "3449 ",
      "43820",
      "43821",
      "43822",
      "43830",
      "43831",
      "43832",
      "43840",
      "43841",
      "43842",
      "43850",
      "43851",
      "43852",
      "43853" = "PARA"   /* Paralysis */
      "3300 ",
      "3301 ",
      "3302 ",
      "3303 ",
      "3308 ",
      "3309 ",
      "3310 ",
```

```
      "3311 ",
      "3312 ",
      "3313 ",
      "3314 ",
      "3317 ",
      "33181",
      "33189",
      "3319 ",
      "3320 ",
      "3334 ",
      "3335 ",
      "3340 ",
      "3341 ",
      "3342 ",
      "3343 ",
      "3344 ",
      "3348 ",
      "3349 ",
      "3350 ",
      "3351 ",
      "33510",
      "33511",
      "33519",
      "3352 ",
      "33520",
      "33521",
      "33522",
      "33523",
      "33524",
      "33529",
      "3358 ",
      "3359 ",
      "340  ",
      "3411 ",
      "3418 ",
      "3419 ",
      "34500",
      "34501",
```

PSFMTS -- CONTINUED

```
      "34510",
      "34511",
      "3452 ",
      "3453 ",
      "34540",
      "34541",
      "34550",
      "34551",
      "34560",
      "34561",
      "34570",
      "34571",
      "34580",
      "34581",
      "34590",
      "34591",
      "3481 ",
      "3483 ",
```

```
      "7803 ",
      "78039",
      "7843 " = "NEURO"  /* Other neurological */
      "490  ",
      "4910 ",
      "4911 ",
      "49120",
      "49121",
      "4918 ",
      "4919 ",
      "4920 ",
      "4928 ",
      "49300",
      "49301",
      "49302",
      "49310",
      "49311",
      "49312",
      "49320",
      "49321",
      "49322",
      "49390",
      "49391",
      "49392",
      "494  ",
      "4940 ",
      "4941 ",
      "4950 ",
      "4951 ",
      "4952 ",
      "4953 ",
      "4954 ",
      "4955 ",
      "4956 ",
      "4957 ",
      "4958 ",
      "4959 ",
      "496  ",
      "500  ",
      "501  ",
      "502  ",
PSFMTS -- CONTINUED

      "503  ",
      "504  ",
      "505  ",
      "5064 " = "CHRNLUNG"  /* Chronic pulmonary disease */
      "25000",
      "25001",
      "25002",
      "25003",
      "25010",
      "25011",
      "25012",
      "25013",
      "25020",
      "25021",
      "25022",
```

```
      "25023",
      "25030",
      "25031",
      "25032",
      "25033",
      "64800",
      "64801",
      "64802",
      "64803",
      "64804" = "DM"      /* Diabetes */
      "25040",
      "25041",
      "25042",
      "25043",
      "25050",
      "25051",
      "25052",
      "25053",
      "25060",
      "25061",
      "25062",
      "25063",
      "25070",
      "25071",
      "25072",
      "25073",
      "25080",
      "25081",
      "25082",
      "25083",
      "25090",
      "25091",
      "25092",
      "25093" = "DMCX"     /* Diabetes, complicated */
      "7751 " = "DMNB"     /* Diabetes, newborn */
      "243  ",
      "2440 ",
      "2441 ",
      "2442 ",
      "2448 ",
      "2449 " = "HYPOTHY"  /* Hypothyroidism */
      "585  ",
      "586  ",
      "V420 ",
PSFMTS -- CONTINUED

      "V451 ",
      "V560 ",
      "V561 ",
      "V562 ",
      "V5631",
      "V5632",
      "V568 " = "RENLFAIL" /* Renal failure */
      "40301",
      "40311",
      "40391",
      "40402",
      "40412",
```

```
      "40492" = "HTNRF"    /* Hypertension+Renal failure */
      "07022",
      "07023",
      "07032",
      "07033",
      "07044",
      "07054",
      "4560 ",
      "4561 ",
      "45620",
      "45621",
      "5710 ",
      "5712 ",
      "5713 ",
      "57140",
      "57141",
      "57149",
      "5715 ",
      "5716 ",
      "5718 ",
      "5719 ",
      "5723 ",
      "5728 ",
      "V427 " = "LIVER"  /* Liver disease */
      "53170",
      "53171",
      "53190",
      "53191",
      "53270",
      "53271",
      "53290",
      "53291",
      "53370",
      "53371",
      "53390",
      "53391",
      "53470",
      "53471",
      "53490",
      "53491" = "ULCER"    /* Peptic ulcer disease including bleeding */
      "042  ",
      "0420 ",
      "0421 ",
      "0422 ",
      "0429 ",
      "0430 ",
      "0431 ",
PSFMTS -- CONTINUED

      "0432 ",
      "0433 ",
      "0439 ",
      "0440 ",
      "0449 " = "AIDS"     /* HIV and AIDS */
      "20000",
      "20001",
      "20002",
      "20003",
```

"20004",
"20005",
"20006",
"20007",
"20008",
"20010",
"20011",
"20012",
"20013",
"20014",
"20015",
"20016",
"20017",
"20018",
"20020",
"20021",
"20022",
"20023",
"20024",
"20025",
"20026",
"20027",
"20028",
"20080",
"20081",
"20082",
"20083",
"20084",
"20085",
"20086",
"20087",
"20088",
"20100",
"20101",
"20102",
"20103",
"20104",
"20105",
"20106",
"20107",
"20108",
"20110",
"20111",
"20112",
"20113",
"20114",
"20115",
"20116",
"20117",
"20118",

PSFMTS -- CONTINUED

"20120",
"20121",
"20122",
"20123",
"20124",
"20125",

"20126",
"20127",
"20128",
"20140",
"20141",
"20142",
"20143",
"20144",
"20145",
"20146",
"20147",
"20148",
"20150",
"20151",
"20152",
"20153",
"20154",
"20155",
"20156",
"20157",
"20158",
"20160",
"20161",
"20162",
"20163",
"20164",
"20165",
"20166",
"20167",
"20168",
"20170",
"20171",
"20172",
"20173",
"20174",
"20175",
"20176",
"20177",
"20178",
"20190",
"20191",
"20192",
"20193",
"20194",
"20195",
"20196",
"20197",
"20198",
"20200",
"20201",
"20202",
"20203",
"20204",

PSFMTS -- CONTINUED

"20205",
"20206",
"20207",

```
      "20208",
      "20210",
      "20211",
      "20212",
      "20213",
      "20214",
      "20215",
      "20216",
      "20217",
      "20218",
      "20220",
      "20221",
      "20222",
      "20223",
      "20224",
      "20225",
      "20226",
      "20227",
      "20228",
      "20230",
      "20231",
      "20232",
      "20233",
      "20234",
      "20235",
      "20236",
      "20237",
      "20238",
      "20250",
      "20251",
      "20252",
      "20253",
      "20254",
      "20255",
      "20256",
      "20257",
      "20258",
      "20260",
      "20261",
      "20262",
      "20263",
      "20264",
      "20265",
      "20266",
      "20267",
      "20268",
      "20280",
      "20281",
      "20282",
      "20283",
      "20284",
      "20285",
      "20286",
      "20287",
      "20288",
      "20290",
```

PSFMTS -- CONTINUED

```
"20291",
"20292",
"20293",
"20294",
"20295",
"20296",
"20297",
"20298",
"2030 ",
"20300",
"20301",
"2038 ",
"20380",
"20381",
"2386 ",
"2733 ",
"V1071",
"V1072",
"V1079" = "LYMPH"  /* Lymphoma */
"1960 ",
"1961 ",
"1962 ",
"1963 ",
"1965 ",
"1966 ",
"1968 ",
"1969 ",
"1970 ",
"1971 ",
"1972 ",
"1973 ",
"1974 ",
"1975 ",
"1976 ",
"1977 ",
"1978 ",
"1980 ",
"1981 ",
"1982 ",
"1983 ",
"1984 ",
"1985 ",
"1986 ",
"1987 ",
"19881",
"19882",
"19889",
"1990 ",
"1991 " = "METS"   /* Metastatic cancer */
"1400 ",
"1401 ",
"1403 ",
"1404 ",
"1405 ",
"1406 ",
"1408 ",
"1409 ",
"1410 ",
```

"1411 ",

PSFMTS -- CONTINUED

"1412 ",
"1413 ",
"1414 ",
"1415 ",
"1416 ",
"1418 ",
"1419 ",
"1420 ",
"1421 ",
"1422 ",
"1428 ",
"1429 ",
"1430 ",
"1431 ",
"1438 ",
"1439 ",
"1440 ",
"1441 ",
"1448 ",
"1449 ",
"1450 ",
"1451 ",
"1452 ",
"1453 ",
"1454 ",
"1455 ",
"1456 ",
"1458 ",
"1459 ",
"1460 ",
"1461 ",
"1462 ",
"1463 ",
"1464 ",
"1465 ",
"1466 ",
"1467 ",
"1468 ",
"1469 ",
"1470 ",
"1471 ",
"1472 ",
"1473 ",
"1478 ",
"1479 ",
"1480 ",
"1481 ",
"1482 ",
"1483 ",
"1488 ",
"1489 ",
"1490 ",
"1491 ",
"1498 ",
"1499 ",

```
       "1500 ",
       "1501 ",
       "1502 ",
       "1503 ",
PSFMTS -- CONTINUED

       "1504 ",
       "1505 ",
       "1508 ",
       "1509 ",
       "1510 ",
       "1511 ",
       "1512 ",
       "1513 ",
       "1514 ",
       "1515 ",
       "1516 ",
       "1518 ",
       "1519 ",
       "1520 ",
       "1521 ",
       "1522 ",
       "1523 ",
       "1528 ",
       "1529 ",
       "1530 ",
       "1531 ",
       "1532 ",
       "1533 ",
       "1534 ",
       "1535 ",
       "1536 ",
       "1537 ",
       "1538 ",
       "1539 ",
       "1540 ",
       "1541 ",
       "1542 ",
       "1543 ",
       "1548 ",
       "1550 ",
       "1551 ",
       "1552 ",
       "1560 ",
       "1561 ",
       "1562 ",
       "1568 ",
       "1569 ",
       "1570 ",
       "1571 ",
       "1572 ",
       "1573 ",
       "1574 ",
       "1578 ",
       "1579 ",
       "1580 ",
       "1588 ",
       "1589 ",
```

```
      "1590 ",
      "1591 ",
      "1598 ",
      "1599 ",
      "1600 ",
      "1601 ",
      "1602 ",
PSFMTS -- CONTINUED

      "1603 ",
      "1604 ",
      "1605 ",
      "1608 ",
      "1609 ",
      "1610 ",
      "1611 ",
      "1612 ",
      "1613 ",
      "1614 ",
      "1615 ",
      "1616 ",
      "1617 ",
      "1618 ",
      "1619 ",
      "1620 ",
      "1622 ",
      "1623 ",
      "1624 ",
      "1625 ",
      "1628 ",
      "1629 ",
      "1630 ",
      "1631 ",
      "1638 ",
      "1639 ",
      "1640 ",
      "1641 ",
      "1642 ",
      "1643 ",
      "1648 ",
      "1649 ",
      "1650 ",
      "1658 ",
      "1659 ",
      "1700 ",
      "1701 ",
      "1702 ",
      "1703 ",
      "1704 ",
      "1705 ",
      "1706 ",
      "1707 ",
      "1708 ",
      "1709 ",
      "1710 ",
      "1712 ",
      "1713 ",
      "1714 ",
```

```
      "1715 ",
      "1716 ",
      "1717 ",
      "1718 ",
      "1719 ",
      "1720 ",
      "1721 ",
      "1722 ",
      "1723 ",
      "1724 ",
```
PSFMTS -- CONTINUED
```
      "1725 ",
      "1726 ",
      "1727 ",
      "1728 ",
      "1729 ",
      "1740 ",
      "1741 ",
      "1742 ",
      "1743 ",
      "1744 ",
      "1745 ",
      "1746 ",
      "1748 ",
      "1749 ",
      "1750 ",
      "1759 ",
      "179  ",
      "1800 ",
      "1801 ",
      "1808 ",
      "1809 ",
      "181  ",
      "1820 ",
      "1821 ",
      "1828 ",
      "1830 ",
      "1832 ",
      "1833 ",
      "1834 ",
      "1835 ",
      "1838 ",
      "1839 ",
      "1840 ",
      "1841 ",
      "1842 ",
      "1843 ",
      "1844 ",
      "1848 ",
      "1849 ",
      "185  ",
      "1860 ",
      "1869 ",
      "1871 ",
      "1872 ",
      "1873 ",
      "1874 ",
```

```
      "1875 ",
      "1876 ",
      "1877 ",
      "1878 ",
      "1879 ",
      "1880 ",
      "1881 ",
      "1882 ",
      "1883 ",
      "1884 ",
      "1885 ",
      "1886 ",
      "1887 ",
```

PSFMTS -- CONTINUED

```
      "1888 ",
      "1889 ",
      "1890 ",
      "1891 ",
      "1892 ",
      "1893 ",
      "1894 ",
      "1898 ",
      "1899 ",
      "1900 ",
      "1901 ",
      "1902 ",
      "1903 ",
      "1904 ",
      "1905 ",
      "1906 ",
      "1907 ",
      "1908 ",
      "1909 ",
      "1910 ",
      "1911 ",
      "1912 ",
      "1913 ",
      "1914 ",
      "1915 ",
      "1916 ",
      "1917 ",
      "1918 ",
      "1919 ",
      "1920 ",
      "1921 ",
      "1922 ",
      "1923 ",
      "1928 ",
      "1929 ",
      "193  ",
      "1940 ",
      "1941 ",
      "1943 ",
      "1944 ",
      "1945 ",
      "1946 ",
      "1948 ",
```

```
        "1949 ",
        "1950 ",
        "1951 ",
        "1952 ",
        "1953 ",
        "1954 ",
        "1955 ",
        "1958 ",
        "V1000",
        "V1001",
        "V1002",
        "V1003",
        "V1004",
        "V1005",
        "V1006",
        "V1007",
```

PSFMTS -- CONTINUED

```
        "V1009",
        "V1011",
        "V1012",
        "V1020",
        "V1021",
        "V1022",
        "V1029",
        "V103 ",
        "V1040",
        "V1041",
        "V1042",
        "V1043",
        "V1044",
        "V1045",
        "V1046",
        "V1047",
        "V1049",
        "V1050",
        "V1051",
        "V1052",
        "V1059",
        "V1081",
        "V1082",
        "V1083",
        "V1084",
        "V1085",
        "V1086",
        "V1087",
        "V1088",
        "V1089",
        "V109 " = "TUMOR"  /* Solid tumor without metastasis */
        "7010 ",
        "7100 ",
        "7101 ",
        "7102 ",
        "7103 ",
        "7104 ",
        "7105 ",
        "7108 ",
        "7109 ",
```

```
          "7140 ",
          "7141 ",
          "7142 ",
          "71430",
          "71431",
          "71432",
          "71433",
          "7144 ",
          "71481",
          "71489",
          "7149 ",
          "7200 ",
          "7201 ",
          "7202 ",
          "72081",
          "72089",
          "7209 ",
           "725  " = "ARTH" /* Rheumatoid arthritis/collagen vascular diseases
*/
```

PSFMTS -- CONTINUED

```
          "2860 ",
          "2861 ",
          "2862 ",
          "2863 ",
          "2864 ",
          "2865 ",
          "2866 ",
          "2867 ",
          "2869 ",
          "2871 ",
          "2873 ",
          "2874 ",
          "2875 " = "COAG"     /* Coagulopathy */
          "2780 ",
          "27800",
          "27801" = "OBESE"    /* Obesity      */
          "260  ",
          "261  ",
          "262  ",
          "2630 ",
          "2631 ",
          "2632 ",
          "2638 ",
          "2639 ",
          "7832 ",
          "78321",
          "78322" = "WGHTLOSS" /* Weight loss */
          "2760 ",
          "2761 ",
          "2762 ",
          "2763 ",
          "2764 ",
          "2765 ",
          "2766 ",
          "2767 ",
          "2768 ",
          "2769 " = "LYTES"    /* Fluid and elctrolyte disorders */
```

```
      "2800 ",
      "64820",
      "64821",
      "64822",
      "64823",
      "64824" = "BLDLOSS"  /* Blood loss anemia */
      "2801 ",
      "2808 ",
      "2809 ",
      "2810 ",
      "2811 ",
      "2812 ",
      "2813 ",
      "2814 ",
      "2818 ",
      "2819 ",
      "28521",
      "28522",
      "28529",
      "2859 " = "ANEMDEF"  /* Deficiency anemias */
      "2910 ",
      "2911 ",
```

PSFMTS -- CONTINUED

```
      "2912 ",
      "2913 ",
      "2915 ",
      "2918 ",
      "29181",
      "29189",
      "2919 ",
      "30300",
      "30301",
      "30302",
      "30303",
      "30390",
      "30391",
      "30392",
      "30393",
      "30500",
      "30501",
      "30502",
      "30503"= "ALCOHOL"  /* Alcohol abuse */
      "2920 ",
      "29282",
      "29283",
      "29284",
      "29289",
      "2929 ",
      "30400",
      "30401",
      "30402",
      "30403",
      "30410",
      "30411",
      "30412",
      "30413",
      "30420",
```

"30421",
"30422",
"30423",
"30430",
"30431",
"30432",
"30433",
"30440",
"30441",
"30442",
"30443",
"30450",
"30451",
"30452",
"30453",
"30460",
"30461",
"30462",
"30463",
"30470",
"30471",
"30472",
"30473",
"30480",
"30481",

PSFMTS -- CONTINUED

"30482",
"30483",
"30490",
"30491",
"30492",
"30493",
"30520",
"30521",
"30522",
"30523",
"30530",
"30531",
"30532",
"30533",
"30540",
"30541",
"30542",
"30543",
"30550",
"30551",
"30552",
"30553",
"30560",
"30561",
"30562",
"30563",
"30570",
"30571",
"30572",
"30573",
"30580",

```
        "30581",
        "30582",
        "30583",
        "30590",
        "30591",
        "30592",
        "30593",
        "64830",
        "64831",
        "64832",
        "64833",
        "64834" = "DRUG"     /* Drug abuse */
        "29500",
        "29501",
        "29502",
        "29503",
        "29504",
        "29505",
        "29510",
        "29511",
        "29512",
        "29513",
        "29514",
        "29515",
        "29520",
        "29521",
        "29522",
        "29523",
```

PSFMTS -- CONTINUED

```
        "29524",
        "29525",
        "29530",
        "29531",
        "29532",
        "29533",
        "29534",
        "29535",
        "29540",
        "29541",
        "29542",
        "29543",
        "29544",
        "29545",
        "29550",
        "29551",
        "29552",
        "29553",
        "29554",
        "29555",
        "29560",
        "29561",
        "29562",
        "29563",
        "29564",
        "29565",
        "29570",
        "29571",
```

"29572",
"29573",
"29574",
"29575",
"29580",
"29581",
"29582",
"29583",
"29584",
"29585",
"29590",
"29591",
"29592",
"29593",
"29594",
"29595",
"29600",
"29601",
"29602",
"29603",
"29604",
"29605",
"29606",
"29610",
"29611",
"29612",
"29613",
"29614",
"29615",
"29616",
"29620",

PSFMTS -- CONTINUED

"29621",
"29622",
"29623",
"29624",
"29625",
"29626",
"29630",
"29631",
"29632",
"29633",
"29634",
"29635",
"29636",
"29640",
"29641",
"29642",
"29643",
"29644",
"29645",
"29646",
"29650",
"29651",
"29652",
"29653",
"29654",

```
      "29655",
      "29656",
      "29660",
      "29661",
      "29662",
      "29663",
      "29664",
      "29665",
      "29666",
      "2967 ",
      "29680",
      "29681",
      "29682",
      "29689",
      "29690",
      "29699",
      "2970 ",
      "2971 ",
      "2972 ",
      "2973 ",
      "2978 ",
      "2979 ",
      "2980 ",
      "2981 ",
      "2982 ",
      "2983 ",
      "2984 ",
      "2988 ",
      "2989 ",
      "29910",
      "29911" = "PSYCH"    /* Psychoses */
      "3004 ",
      "30112",
      "3090 ",
```

PSFMTS -- CONTINUED

```
      "3091 ",
      "311  " = "DEPRESS"  /* Depression */
    ;


    /** DRG Formats  **/

    VALUE CARDDRG                       /* Cardiac */
        103,104,105,106,107,108,109,110,111,
        112,115,116,117,118,120,121,122,
        123,124,125,126,127,129,132,133,134,
        135,136,137,138,139,140,141,142,
        143 = "YES" ;

     VALUE PERIDRG                      /* Peripheral vascular */
        130,131 = "YES" ;

     VALUE RENALDRG                     /* Renal */
        302,303,304,305,315,316,317,318,319,
        320,321,322,323,324,325,326,327,328,
        329,330,331,332,333 = "YES" ;
```

```
        VALUE CEREDRG                       /* Cerebrovascular */
           5,14,15,16,17 = "YES" ;

        VALUE PULMDRG                       /* COPD asthma */
           88,96,97,98 = "YES" ;

        VALUE  DIABDRG                      /* Diabetes */
           294,295     = "YES" ;

        VALUE HYPODRG                       /* Thyroid endocrine */
           290,300,301 = "YES";

          VALUE RENFDRG                              /* Kidney transp, renal
fail/dialysis */
           302,316,317  = "YES" ;

        VALUE LIVERDRG                      /* Liver */
           199,200,201,202,205,206,207,208 = "YES" ;

        VALUE ULCEDRG                       /* GI hemorrhage or ulcer */
           174,175,176,177,178  = "YES"  ;

        VALUE HIVDRG                        /* Human immunodeficiency virus */
           488,489,490  =  "YES" ;

        VALUE LEUKDRG                       /* Leukemia/lymphoma */
           400,401,402,403,404,405,406,407,408,
           409,410,411,412,413,414,473,492     = "YES"  ;

        VALUE CANCDRG                       /* Cancer, lymphoma */
           10,11,64,82,172,173,199,203,239,
           257,258,259,260,274,275,303,318,319,
           338,344,346,347,354,355,357,363,366,
           367,406,407,408,409,410,411,412,413,
           414 = "YES"   ;

        VALUE ARTHDRG                       /* Connective tissue */
   PSFMTS -- CONTINUED

           240,241 = "YES"  ;

        VALUE NUTRDRG                       /* Nutrition/metabolic */
           296,297,298 = "YES"  ;

        VALUE ANEMDRG                       /* Anemia */
           395,396 = "YES";

        VALUE ALCDRG                        /* Alcohol drug */
           433,434,435,436,437 = "YES" ;

    /* ------------------------------------------------------------- */
    /* ---   INDICATOR SPECIFIC AGE, DRG AND COMORBIDITY STRATIFIERS - */
    /* ------------------------------------------------------------- */


   VALUE A1FMT
   18 - 24  = '1'
   25 - 29  = '2'
```

```
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
50 - 54  = '7'
55 - 59  = '8'
60 - 64  = '9'
65 - 69  = '10'
70 - 74  = '0'
75 - 79  = '11'
80 - 84  = '12'
OTHER  = '13'
;

VALUE A2FMT
15 - 17  = '1'
18 - 24  = '0'
25 - 29  = '2'
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
50 - 54  = '7'
OTHER  = '8'
;

VALUE A3FMT
30 - 34  = '1'
35 - 39  = '2'
40 - 44  = '3'
45 - 49  = '4'
50 - 54  = '5'
55 - 59  = '6'
60 - 64  = '7'
65 - 69  = '8'
70 - 74  = '9'
75 - 79  = '0'
80 - 84  = '10'
85 - high  = '11'
OTHER  = '12'
PSFMTS -- CONTINUED

;

VALUE A4FMT
40 - 44  = '1'
45 - 49  = '2'
50 - 54  = '3'
55 - 59  = '4'
60 - 64  = '5'
65 - 69  = '6'
70 - 74  = '7'
75 - 79  = '8'
80 - 84  = '9'
85 - high  = '0'
OTHER  = '10'
;
```

```
VALUE A5FMT
18 - 24  = '1'
25 - 29  = '2'
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
50 - 54  = '7'
55 - 59  = '8'
60 - 64  = '9'
65 - 69  = '10'
70 - 74  = '11'
75 - 79  = '0'
80 - 84  = '12'
85 - high  = '13'
OTHER  = '14'
;

VALUE A6FMT
18 - 24  = '1'
25 - 29  = '2'
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
50 - 54  = '7'
55 - 59  = '8'
60 - 64  = '9'
65 - 69  = '10'
70 - 74  = '11'
75 - 79  = '0'
80 - 84  = '12'
85 - high  = '13'
OTHER  = '14'
;

VALUE A7FMT
18 - 24  = '1'
25 - 29  = '2'
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
```

```
50 - 54  = '7'
55 - 59  = '8'
60 - 64  = '9'
65 - 69  = '10'
70 - 74  = '11'
75 - 79  = '0'
80 - 84  = '12'
85 - high  = '13'
OTHER  = '14'
;

VALUE A8FMT
55 - 59  = '1'
```

```
60 - 64  = '2'
65 - 69  = '3'
70 - 74  = '0'
75 - 79  = '4'
80 - 84  = '5'
85 - high  = '6'
OTHER  = '7'
;

VALUE A9FMT
18 - 24  = '1'
25 - 29  = '2'
30 - 34  = '3'
35 - 39  = '4'
40 - 44  = '5'
45 - 49  = '6'
50 - 54  = '7'
55 - 59  = '8'
60 - 64  = '9'
65 - 69  = '0'
70 - 74  = '10'
75 - 79  = '11'
80 - 84  = '12'
85 - high  = '13'
OTHER  = '14'
;

VALUE A10FMT
30 - 34  = '1'
35 - 39  = '2'
40 - 44  = '3'
45 - 49  = '4'
50 - 54  = '5'
55 - 59  = '6'
60 - 64  = '7'
65 - 69  = '8'
70 - 74  = '0'
75 - 79  = '9'
80 - 84  = '10'
85 - high  = '11'
OTHER  = '12'
;

VALUE A11FMT
25 - 29  = '1'
30 - 34  = '2'
```
PSFMTS -- CONTINUED
```
35 - 39  = '3'
40 - 44  = '4'
45 - 49  = '5'
50 - 54  = '6'
55 - 59  = '7'
60 - 64  = '8'
65 - 69  = '9'
70 - 74  = '0'
75 - 79  = '10'
80 - 84  = '11'
```

```
OTHER  = '12'
;

VALUE A12FMT
30 - 34  = '1'
35 - 39  = '2'
40 - 44  = '3'
45 - 49  = '4'
50 - 54  = '5'
55 - 59  = '6'
60 - 64  = '7'
65 - 69  = '8'
70 - 74  = '0'
75 - 79  = '9'
80 - 84  = '10'
85 - high  = '11'
OTHER  = '12'
;

VALUE A13FMT
35 - 39  = '1'
40 - 44  = '2'
45 - 49  = '3'
50 - 54  = '4'
55 - 59  = '5'
60 - 64  = '6'
65 - 69  = '7'
70 - 74  = '0'
75 - 79  = '8'
80 - 84  = '9'
85 - high  = '10'
OTHER  = '11'
;

VALUE A14FMT
30 - 34  = '1'
35 - 39  = '2'
40 - 44  = '3'
45 - 49  = '0'
50 - 54  = '4'
55 - 59  = '5'
60 - 64  = '6'
65 - 69  = '7'
70 - 74  = '8'
75 - 79  = '9'
80 - 84  = '10'
85 - high  = '11'
OTHER  = '12'
;
```

PSFMTS -- CONTINUED

```
VALUE A15FMT
30 - 34  = '1'
35 - 39  = '2'
40 - 44  = '3'
45 - 49  = '4'
50 - 54  = '5'
```

```
55 - 59  = '6'
60 - 64  = '7'
65 - 69  = '8'
70 - 74  = '9'
75 - 79  = '0'
80 - 84  = '10'
85 - high  = '11'
OTHER  = '12'
;

VALUE A17FMT
OTHER  = '0'
;

VALUE A18FMT
18 - 24  = '0'
25 - 29  = '1'
30 - 34  = '2'
35 - 39  = '3'
OTHER  = '4'
;

VALUE A19FMT
18 - 24  = '0'
25 - 29  = '1'
30 - 34  = '2'
35 - 39  = '3'
OTHER  = '4'
;

VALUE A20FMT
18 - 24  = '1'
25 - 29  = '2'
30 - 34  = '0'
35 - 39  = '3'
OTHER  = '4'
;

VALUE D1FMT
0 = '1'
5 = '2'
65 = '3'
89 = '4'
90 = '5'
117 = '6'
120 = '7'
123 = '8'
155 = '9'
156 = '10'
161 = '11'
163 = '12'
189 = '13'
```
PSFMTS -- CONTINUED

```
242 = '14'
243 = '15'
259 = '16'
261 = '17'
```

```
271 = '0'
334 = '18'
358 = '19'
368 = '20'
OTHER  = '21'
;

VALUE D2FMT
13 = '1'
17 = '2'
21 = '3'
59 = '4'
76 = '5'
79 = '6'
101 = '7'
108 = '8'
112 = '9'
113 = '10'
114 = '11'
127 = '12'
178 = '13'
216 = '14'
218 = '15'
235 = '16'
243 = '17'
261 = '18'
271 = '0'
272 = '19'
315 = '20'
319 = '21'
320 = '22'
328 = '23'
OTHER  = '24'
;

VALUE D3FMT
0 = '1'
1 = '2'
2 = '3'
4 = '4'
5 = '5'
7 = '6'
9 = '7'
10 = '8'
11 = '9'
12 = '10'
13 = '11'
14 = '12'
15 = '13'
16 = '14'
20 = '15'
22 = '16'
23 = '17'
27 = '18'
65 = '19'
```
PSFMTS -- CONTINUED

```
66 = '20'
```

67 = '21'
68 = '22'
70 = '23'
71 = '24'
72 = '25'
73 = '26'
74 = '27'
75 = '28'
77 = '29'
78 = '30'
79 = '31'
81 = '32'
82 = '33'
84 = '34'
85 = '35'
86 = '36'
87 = '37'
88 = '38'
89 = '39'
90 = '40'
91 = '41'
92 = '42'
93 = '43'
94 = '44'
98 = '45'
99 = '46'
100 = '47'
101 = '48'
102 = '49'
103 = '50'
104 = '51'
106 = '52'
107 = '53'
109 = '54'
111 = '55'
112 = '56'
113 = '57'
114 = '58'
115 = '59'
116 = '60'
117 = '61'
118 = '62'
120 = '63'
122 = '64'
123 = '65'
129 = '66'
130 = '67'
131 = '68'
132 = '69'
133 = '70'
134 = '71'
135 = '72'
136 = '73'
141 = '74'
143 = '75'
144 = '76'
145 = '77'
146 = '78'

PSFMTS -- CONTINUED

```
150 = '79'
151 = '80'
152 = '81'
153 = '82'
154 = '83'
155 = '84'
156 = '85'
158 = '86'
159 = '87'
160 = '88'
161 = '89'
164 = '90'
165 = '91'
168 = '92'
170 = '93'
171 = '94'
172 = '95'
174 = '96'
175 = '97'
176 = '98'
178 = '99'
179 = '100'
181 = '101'
182 = '102'
183 = '103'
186 = '104'
188 = '105'
207 = '106'
209 = '107'
214 = '108'
215 = '109'
217 = '110'
220 = '111'
222 = '112'
223 = '113'
224 = '114'
225 = '115'
226 = '116'
229 = '117'
230 = '118'
232 = '119'
233 = '120'
240 = '121'
243 = '122'
268 = '123'
269 = '124'
295 = '125'
297 = '126'
298 = '127'
299 = '128'
300 = '129'
301 = '130'
306 = '131'
309 = '132'
310 = '133'
311 = '134'
```

```
313 = '135'
314 = '136'
317 = '137'
```
PSFMTS -- CONTINUED

```
319 = '138'
320 = '139'
323 = '140'
324 = '0'
328 = '141'
332 = '142'
334 = '143'
339 = '144'
341 = '145'
345 = '146'
348 = '147'
349 = '148'
350 = '149'
352 = '150'
353 = '151'
354 = '152'
356 = '153'
357 = '154'
358 = '155'
360 = '156'
361 = '157'
362 = '158'
364 = '159'
365 = '160'
368 = '161'
374 = '162'
OTHER  = '163'
;

VALUE D4FMT
0 = '1'
1 = '2'
2 = '3'
3 = '4'
4 = '5'
5 = '6'
7 = '7'
9 = '8'
10 = '9'
12 = '10'
13 = '11'
14 = '12'
15 = '13'
16 = '14'
19 = '15'
20 = '16'
22 = '17'
23 = '18'
27 = '19'
55 = '20'
63 = '21'
65 = '22'
66 = '23'
```

67 = '24'
68 = '25'
70 = '26'
72 = '27'
73 = '28'
74 = '29'
PSFMTS -- CONTINUED

75 = '30'
77 = '31'
81 = '32'
82 = '33'
84 = '34'
85 = '35'
86 = '36'
87 = '37'
88 = '38'
89 = '39'
90 = '40'
91 = '41'
93 = '42'
94 = '43'
98 = '44'
101 = '45'
102 = '46'
103 = '0'
105 = '47'
106 = '48'
107 = '49'
109 = '50'
111 = '51'
112 = '52'
113 = '53'
115 = '54'
116 = '55'
117 = '56'
118 = '57'
119 = '58'
120 = '59'
122 = '60'
123 = '61'
124 = '62'
126 = '63'
129 = '64'
130 = '65'
131 = '66'
132 = '67'
134 = '68'
135 = '69'
136 = '70'
141 = '71'
143 = '72'
144 = '73'
145 = '74'
146 = '75'
150 = '76'
151 = '77'
152 = '78'

153 = '79'
154 = '80'
155 = '81'
156 = '82'
158 = '83'
159 = '84'
160 = '85'
161 = '86'
165 = '87'
PSFMTS -- CONTINUED

170 = '88'
172 = '89'
174 = '90'
175 = '91'
176 = '92'
178 = '93'
179 = '94'
182 = '95'
183 = '96'
186 = '97'
193 = '98'
197 = '99'
198 = '100'
199 = '101'
200 = '102'
202 = '103'
204 = '104'
207 = '105'
209 = '106'
210 = '107'
214 = '108'
215 = '109'
216 = '110'
217 = '111'
218 = '112'
220 = '113'
221 = '114'
222 = '115'
223 = '116'
224 = '117'
225 = '118'
226 = '119'
229 = '120'
230 = '121'
232 = '122'
233 = '123'
236 = '124'
240 = '125'
242 = '126'
243 = '127'
252 = '128'
258 = '129'
260 = '130'
261 = '131'
268 = '132'
271 = '133'
292 = '134'

```
295 = '135'
297 = '136'
298 = '137'
299 = '138'
300 = '139'
301 = '140'
303 = '141'
304 = '142'
306 = '143'
309 = '144'
310 = '145'
311 = '146'
```
PSFMTS -- CONTINUED

```
312 = '147'
313 = '148'
314 = '149'
315 = '150'
317 = '151'
319 = '152'
323 = '153'
324 = '154'
328 = '155'
334 = '156'
339 = '157'
341 = '158'
342 = '159'
349 = '160'
350 = '161'
352 = '162'
353 = '163'
354 = '164'
356 = '165'
357 = '166'
358 = '167'
360 = '168'
361 = '169'
362 = '170'
363 = '171'
364 = '172'
365 = '173'
367 = '174'
368 = '175'
374 = '176'
OTHER  = '177'
;

VALUE D5FMT
0 = '1'
86 = '2'
89 = '3'
90 = '4'
117 = '5'
120 = '6'
155 = '7'
261 = '8'
271 = '0'
310 = '9'
```

```
334 = '10'
358 = '11'
OTHER  = '12'
;

VALUE D6FMT
0 = '1'
1 = '2'
4 = '3'
5 = '4'
7 = '5'
12 = '6'
65 = '7'
66 = '8'
67 = '9'
```
PSFMTS -- CONTINUED
```
68 = '10'
70 = '11'
72 = '12'
73 = '13'
74 = '14'
75 = '15'
82 = '16'
90 = '17'
91 = '18'
93 = '19'
94 = '20'
98 = '21'
99 = '22'
100 = '23'
101 = '24'
103 = '0'
106 = '25'
107 = '26'
111 = '27'
115 = '28'
116 = '29'
117 = '30'
118 = '31'
119 = '32'
120 = '33'
123 = '34'
124 = '35'
126 = '36'
127 = '37'
129 = '38'
130 = '39'
131 = '40'
134 = '41'
135 = '42'
136 = '43'
141 = '44'
143 = '45'
144 = '46'
145 = '47'
146 = '48'
150 = '49'
```

```
151 = '50'
152 = '51'
153 = '52'
155 = '53'
160 = '54'
175 = '55'
189 = '56'
208 = '57'
210 = '58'
215 = '59'
217 = '60'
222 = '61'
223 = '62'
229 = '63'
230 = '64'
233 = '65'
242 = '66'
257 = '67'
```
PSFMTS -- CONTINUED

```
258 = '68'
260 = '69'
261 = '70'
271 = '71'
299 = '72'
300 = '73'
301 = '74'
303 = '75'
306 = '76'
310 = '77'
311 = '78'
313 = '79'
349 = '80'
352 = '81'
356 = '82'
357 = '83'
358 = '84'
364 = '85'
368 = '86'
374 = '87'
OTHER  = '88'
;

VALUE D7FMT
0 = '1'
1 = '2'
2 = '3'
3 = '4'
4 = '5'
5 = '6'
7 = '7'
10 = '8'
12 = '9'
13 = '10'
15 = '11'
16 = '12'
20 = '13'
23 = '14'
```

```
27 = '15'
65 = '16'
66 = '17'
67 = '18'
68 = '19'
73 = '20'
74 = '21'
75 = '22'
79 = '23'
84 = '24'
85 = '25'
86 = '26'
87 = '27'
88 = '28'
89 = '29'
90 = '30'
91 = '31'
93 = '32'
94 = '33'
98 = '34'
99 = '35'
```
PSFMTS -- CONTINUED

```
100 = '36'
101 = '37'
102 = '38'
103 = '39'
106 = '40'
107 = '41'
111 = '42'
112 = '43'
113 = '44'
114 = '45'
115 = '46'
117 = '47'
118 = '48'
120 = '49'
121 = '50'
126 = '51'
129 = '52'
131 = '53'
134 = '54'
135 = '55'
136 = '56'
137 = '57'
141 = '58'
143 = '59'
146 = '60'
150 = '61'
152 = '62'
153 = '63'
154 = '64'
155 = '65'
156 = '66'
158 = '67'
160 = '68'
170 = '69'
174 = '70'
```

```
176 = '71'
178 = '72'
193 = '73'
197 = '74'
202 = '75'
214 = '76'
215 = '77'
216 = '78'
217 = '79'
218 = '80'
219 = '81'
220 = '82'
223 = '83'
229 = '84'
230 = '85'
233 = '86'
240 = '87'
261 = '88'
271 = '89'
272 = '90'
283 = '91'
295 = '92'
297 = '93'
298 = '94'
```

PSFMTS -- CONTINUED

```
310 = '95'
311 = '96'
312 = '97'
313 = '98'
314 = '99'
317 = '100'
324 = '0'
328 = '101'
332 = '102'
334 = '103'
339 = '104'
341 = '105'
348 = '106'
349 = '107'
352 = '108'
357 = '109'
358 = '110'
361 = '111'
368 = '112'
OTHER  = '113'
;

VALUE D8FMT
0 = '1'
66 = '2'
90 = '0'
94 = '3'
117 = '4'
214 = '5'
229 = '6'
310 = '7'
348 = '8'
```

```
358 = '9'
OTHER  = '10'
;

VALUE D9FMT
0 = '0'
1 = '1'
5 = '2'
7 = '3'
41 = '4'
48 = '5'
49 = '6'
65 = '7'
84 = '8'
85 = '9'
86 = '10'
87 = '11'
88 = '12'
89 = '13'
90 = '14'
91 = '15'
94 = '16'
98 = '17'
116 = '18'
117 = '19'
118 = '20'
120 = '21'
```
PSFMTS -- CONTINUED

```
122 = '22'
123 = '23'
124 = '24'
129 = '25'
143 = '26'
146 = '27'
155 = '28'
156 = '29'
160 = '30'
189 = '31'
193 = '32'
194 = '33'
196 = '34'
197 = '35'
212 = '36'
221 = '37'
222 = '38'
223 = '39'
225 = '40'
229 = '41'
242 = '42'
243 = '43'
258 = '44'
259 = '45'
261 = '46'
262 = '47'
292 = '48'
299 = '49'
310 = '50'
```

```
332 = '51'
334 = '52'
358 = '53'
361 = '54'
368 = '55'
374 = '56'
OTHER  = '57'
;

VALUE D10FMT
0 = '1'
65 = '2'
85 = '3'
86 = '4'
87 = '5'
89 = '6'
90 = '7'
91 = '8'
98 = '9'
117 = '10'
155 = '0'
221 = '11'
222 = '12'
223 = '13'
229 = '14'
310 = '15'
358 = '16'
374 = '17'
OTHER  = '18'
;
```
PSFMTS -- CONTINUED

```
VALUE D11FMT
0 = '1'
1 = '2'
5 = '3'
7 = '4'
116 = '5'
117 = '6'
118 = '7'
120 = '8'
123 = '9'
129 = '10'
143 = '11'
144 = '12'
146 = '13'
155 = '14'
156 = '15'
160 = '16'
193 = '17'
222 = '18'
223 = '19'
229 = '20'
242 = '21'
258 = '22'
260 = '23'
261 = '0'
```

```
299 = '24'
303 = '25'
310 = '26'
368 = '27'
371 = '28'
374 = '29'
OTHER  = '30'
;

VALUE D12FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
65 = '6'
66 = '7'
84 = '8'
85 = '9'
86 = '10'
87 = '11'
88 = '12'
89 = '13'
90 = '14'
91 = '15'
93 = '16'
94 = '17'
98 = '18'
116 = '19'
117 = '20'
118 = '21'
119 = '22'
```
PSFMTS -- CONTINUED
```
120 = '23'
122 = '24'
123 = '25'
124 = '26'
126 = '27'
127 = '28'
129 = '29'
143 = '30'
145 = '31'
146 = '32'
155 = '33'
156 = '34'
160 = '35'
161 = '36'
163 = '37'
165 = '38'
168 = '39'
170 = '40'
189 = '41'
190 = '42'
193 = '43'
194 = '44'
197 = '45'
```

```
222 = '46'
223 = '47'
224 = '48'
225 = '49'
226 = '50'
229 = '51'
242 = '52'
243 = '53'
258 = '54'
259 = '55'
260 = '56'
261 = '57'
262 = '58'
299 = '59'
300 = '60'
310 = '61'
332 = '62'
334 = '63'
348 = '64'
354 = '65'
358 = '66'
361 = '67'
368 = '68'
371 = '69'
372 = '70'
374 = '71'
OTHER  = '72'
;

VALUE D13FMT
0 = '1'
1 = '2'
5 = '3'
7 = '4'
65 = '5'
66 = '6'
```
PSFMTS -- CONTINUED
```
84 = '7'
85 = '8'
86 = '9'
87 = '10'
88 = '11'
89 = '12'
90 = '13'
91 = '14'
94 = '15'
98 = '16'
117 = '17'
118 = '18'
120 = '19'
129 = '20'
143 = '21'
146 = '22'
155 = '0'
156 = '23'
158 = '24'
160 = '25'
```

```
193 = '26'
197 = '27'
223 = '28'
229 = '29'
243 = '30'
261 = '31'
310 = '32'
334 = '33'
348 = '34'
358 = '35'
368 = '36'
OTHER  = '37'
;

VALUE D14FMT
0 = '1'
89 = '2'
117 = '3'
118 = '4'
120 = '5'
126 = '6'
146 = '7'
222 = '8'
258 = '9'
261 = '0'
310 = '10'
334 = '11'
OTHER  = '12'
;

VALUE D15FMT
0 = '1'
1 = '2'
4 = '3'
5 = '4'
7 = '5'
12 = '6'
41 = '7'
44 = '8'
```
PSFMTS -- CONTINUED
```
54 = '9'
65 = '10'
66 = '11'
68 = '12'
75 = '13'
78 = '14'
84 = '15'
85 = '16'
86 = '17'
87 = '18'
88 = '19'
89 = '20'
90 = '21'
91 = '22'
93 = '23'
94 = '24'
98 = '25'
```

99 = '26'
101 = '27'
103 = '0'
106 = '28'
115 = '29'
116 = '30'
117 = '31'
118 = '32'
119 = '33'
120 = '34'
121 = '35'
122 = '36'
123 = '37'
124 = '38'
126 = '39'
127 = '40'
129 = '41'
130 = '42'
131 = '43'
135 = '44'
136 = '45'
141 = '46'
143 = '47'
144 = '48'
145 = '49'
146 = '50'
149 = '51'
152 = '52'
154 = '53'
155 = '54'
156 = '55'
160 = '56'
161 = '57'
165 = '58'
168 = '59'
170 = '60'
178 = '61'
197 = '62'
208 = '63'
210 = '64'
212 = '65'
221 = '66'

PSFMTS -- CONTINUED

222 = '67'
223 = '68'
225 = '69'
226 = '70'
229 = '71'
230 = '72'
235 = '73'
240 = '74'
242 = '75'
243 = '76'
247 = '77'
257 = '78'
258 = '79'
259 = '80'

```
260 = '81'
261 = '82'
262 = '83'
263 = '84'
266 = '85'
267 = '86'
292 = '87'
299 = '88'
300 = '89'
303 = '90'
304 = '91'
310 = '92'
311 = '93'
313 = '94'
332 = '95'
334 = '96'
341 = '97'
348 = '98'
349 = '99'
352 = '100'
357 = '101'
358 = '102'
361 = '103'
368 = '104'
371 = '105'
372 = '106'
374 = '107'
OTHER  = '108'
;

VALUE D17FMT
OTHER  = '0'
;

VALUE D18FMT
OTHER  = '0'
;

VALUE D19FMT
OTHER  = '0'
;

VALUE D20FMT
OTHER  = '0'
;
```
PSFMTS -- CONTINUED

```
VALUE C1FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
17 = '8'
```

```
18 = '9'
22 = '10'
OTHER  = '11'
;

VALUE C2FMT
0 = '0'
5 = '1'
8 = '2'
22 = '3'
OTHER  = '4'
;

VALUE C3FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
11 = '8'
17 = '9'
18 = '10'
23 = '11'
24 = '12'
OTHER  = '13'
;

VALUE C4FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
6 = '5'
7 = '6'
8 = '7'
9 = '8'
10 = '9'
12 = '10'
14 = '11'
17 = '12'
18 = '13'
23 = '14'
OTHER  = '15'
;
```
PSFMTS -- CONTINUED

```
VALUE C5FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
8 = '5'
```

```
9 = '6'
18 = '7'
OTHER  = '8'
;

VALUE C6FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
11 = '8'
18 = '9'
22 = '10'
23 = '11'
OTHER  = '12'
;

VALUE C7FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
22 = '8'
23 = '9'
24 = '10'
OTHER  = '11'
;

VALUE C8FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
11 = '8'
18 = '9'
OTHER  = '10'
;

VALUE C9FMT
```

```
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
```

```
7 = '5'
8 = '6'
9 = '7'
11 = '8'
17 = '9'
18 = '10'
OTHER  = '11'
;

VALUE C10FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
10 = '8'
12 = '9'
23 = '10'
OTHER  = '11'
;

VALUE C11FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
8 = '5'
9 = '6'
11 = '7'
17 = '8'
18 = '9'
OTHER  = '10'
;

VALUE C12FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
11 = '8'
17 = '9'
18 = '10'
OTHER  = '11'
;

VALUE C13FMT
```
PSFMTS -- CONTINUED
```
0 = '0'
1 = '1'
```

```
2 = '2'
4 = '3'
5 = '4'
8 = '5'
9 = '6'
11 = '7'
23 = '8'
OTHER  = '9'
;

VALUE C14FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
7 = '5'
8 = '6'
9 = '7'
17 = '8'
18 = '9'
23 = '10'
OTHER  = '11'
;

VALUE C15FMT
0 = '0'
1 = '1'
2 = '2'
4 = '3'
5 = '4'
6 = '5'
7 = '6'
8 = '7'
9 = '8'
11 = '9'
18 = '10'
23 = '11'
24 = '12'
OTHER  = '13'
;

VALUE C17FMT
OTHER  = '0'
;

VALUE C18FMT
OTHER  = '0'
;

VALUE C19FMT
OTHER  = '0'
;

VALUE C20FMT
OTHER  = '0'
;
```

```
RUN;
```

MRGB1

```
//TRA0P2 JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR92,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR92,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1002.DENOMB92,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR93,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR93,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1002.DENOMB93,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR94,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR94,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1002.DENOMB94,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR95,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR95,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1002.DENOMB95,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
```

```
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
MRGB1 -- CONTINUED

//       PARM='SYSPARM="96"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR96,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR96,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1002.DENOMB96,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR97,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR97,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1002.DENOMB97,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR98,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR98,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR99,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR99,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
```

```
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR00,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR00,DISP=SHR
MRGB1 -- CONTINUED

//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1002.DENOMB00,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR01,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS1.MEDPAR01,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1002.DENOMB01,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(MRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
```

MRGDNM

```
OPTION LS=132 PS=60;
%LET YR=&SYSPARM;

PROC SORT DATA=IN3.DENOMB&YR.(KEEP=Y&YR.MDST
Y&YR.HICNO Y&YR.CDIV Y&YR.CREG Y&YR.DUAL Y&YR.FEEJ Y&YR.M13
Y&YR.M14 Y&YR.RACE Y&YR.STATE Y&YR.URBN RENAME=(Y&YR.HICNO=XREF))
OUT=TEMP.DENOM;
BY XREF;
RUN;

DATA DENOM;
  MERGE TEMP.DENOM (IN=A)
        IN1.FINDR&YR (IN=B)
        ;
  BY XREF;
  IF A AND B;
RUN;

PROC SORT DATA=DENOM NODUPKEY;
  BY HICNO;
RUN;

PROC SORT DATA=IN1.FINDR&YR OUT=FIND;
   BY HICNO;
RUN;

DATA DENOM;
  MERGE DENOM(IN=A)
        FIND(IN=B)
        ;
  BY HICNO;
  IF A AND B;
RUN;

PROC DATASETS NOLIST;
  DELETE FIND;
RUN;

PROC SORT DATA=DENOM;
   BY XREF;
RUN;

DATA OUT1.PS1;
  MERGE IN2.PS1(IN=A RENAME=(KEY=XREF))
        DENOM(IN=B)
        ;
  BY XREF;
  IF A AND B;
RUN;
```

PS2B1

```
//TRA0PS2 JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR92,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR93,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR94,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR95,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="96"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR96,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR97,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
```

```
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR98,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR99,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR00,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.MRGDNM.MEDPAR01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//*         UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PSI(PSSAS2),
//          DISP=(SHR,PASS,KEEP)
//*
```

COMBINE

```
//TRA0CM JOB (4AGDCH77010),'NHK',CLASS=K,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR92,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR93,DISP=SHR
//IN3    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR94,DISP=SHR
//IN4    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR95,DISP=SHR
//IN5    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR96,DISP=SHR
//IN6    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR97,DISP=SHR
//IN7    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR98,DISP=SHR
//IN8    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR99,DISP=SHR
//IN9    DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR00,DISP=SHR
//IN10   DD DSN=TRA0.@MSQ0000.SAS.PSSAS2.MEDPAR01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.COMBINE.SUM9201,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(10000,1000),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

  /* START: assign macro variables to specify begin and end year
     of summary PSI data */

%let begyr=1992;
%let endyr=2001;
%let file=SUM9201; /* should match the last node of the output DSN */

  /* END: assign macro variables to specify begin and end year
     of summary PSI data */

  /* START: macro program to merge summary PSI data */

%macro process;

  /* start: create macro variables to handle multiple years' data */

%let k=1;
%do i=&begyr %to &endyr;
  %if &i<2000 %then %let x&k=%eval(&i-1900);
  %else %let x&k=0%eval(&i-2000);
  %let k=%eval(&k+1);
%end;
%let tot=%eval(&k-1);

  /* end: create macro variables to handle multiple years' data */

  /* start: create macro variables to handle multiple PSIs */

%do i=1 %to 16;
  %if &i<10 %then %let y&i=0%eval(&i);
  %else %let y&i=%eval(&i);
%end;

  /* end: create macro variables to handle multiple PSIs */

  /* start: data step to merge the summary files */
```

```
data out1.&file.;
  merge
  %do i=1 %to &tot;
```
COMBINE -- CONTINUED
```
    in&i..ps2 (keep=tpps01-tpps16 ppps01-ppps16 opps01-opps16
    y&&x&i..dual y&&x&i..mdst y&&x&i..race y&&x&i..urbn y&&x&i..state
    y&&x&i..creg agecat sexcat _type_
    rename=(y&&x&i..dual=dual y&&x&i..mdst=mdst y&&x&i..race=race
    y&&x&i..urbn=urbn y&&x&i..state=state y&&x&i..creg=creg
    _type_=type&&x&i
    %do j=1 %to 16;
        tpps&&y&j=tpps&&y&j..&&x&i
        ppps&&y&j=ppps&&y&j..&&x&i
        opps&&y&j=opps&&y&j..&&x&i
    %end;))
  %end;;
  by agecat sexcat creg state urbn race mdst dual;
  maxtp=max(of type92-type99, of type00-type&&x&tot);
  mintp=min(of type92-type99, of type00-type&&x&tot);
  if maxtp ne mintp then put type92= type93= type94= type95=
  type96= type97= type98= type99= type00= type01=;
  _type_=maxtp;
  drop maxtp mintp;
  label agecat='Age Group      '
        sexcat='Gender         '
        creg  ='Census Region  '
        urbn  ='Urban/Rural    '
        race  ='Race           '
        mdst  ='Reason for Medicare Eligibility'
        dual  ='Medicare Buy-in'
        state ='State          '
        ;
run;

  /* end: data step to merge the summary files */

%mend;
  /* END: macro program to merge summary PSI data */

%process;  /* call the macro to merge the files */

  /* START: proc steps to examine the data */

  title 'combined summary file';
  title2 "begin year: &begyr. end year: &endyr.";
  proc contents data=out1.&file.;
  run;

  title3 'first 10 observation';
  proc print data=out1.&file.(obs=10);
  run;

  /* END: proc steps to examine the data */

  /* START: steps to examine the _type_ variable */

  proc sort data=out1.&file.(keep=agecat sexcat creg state
```

```
                    urbn race mdst dual _type_) nodupkey out=temp;
      by _type_;
      run;

      title 'values of the _type_ variable marking all possible interactions
of';
```

```
      title2 'age groups, gender, census region, state, urban/rual, race,';
      title3 'Medicare eligibility, and Medicaid buy-in status';

      proc print data=temp;
      run;

      /* END: steps to examine the _type_ variable */
```

OBSERVE

```
//TRA0OB JOB (4AGDCH77010),'NHK',CLASS=K,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.COMBINE.SUM9201,DISP=SHR
//REPORT DD DSN=TRA0.@MSQ0000.HTML.OBSERVE.PDS(SUM9201),
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(100,100,3),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

proc format;
value $st
'01'='AL' '02'='AK' '03'='AZ' '04'='AR' '05'='CA'
'06'='CO' '07'='CT' '08'='DE' '09'='DC' '10'='FL'
'11'='GA' '12'='HI' '13'='ID' '14'='IL' '15'='IN'
'16'='IA' '17'='KS' '18'='KY' '19'='LA' '20'='ME'
'21'='MD' '22'='MA' '23'='MI' '24'='MN' '25'='MS'
'26'='MO' '27'='MT' '28'='NE' '29'='NV' '30'='NH'
'31'='NJ' '32'='NM' '33'='NY' '34'='NC' '35'='ND'
'36'='OH' '37'='OK' '38'='OR' '39'='PA' '40'='RQ'
'41'='RI' '42'='SC' '43'='SD' '44'='TN' '45'='TX'
'46'='UT' '47'='VT' '48'='VI' '49'='VA' '50'='WA'
'51'='WV' '52'='WI' '53'='WY'
;

value $age
'1'='0-54'
'2'='55-64'
'3'='65-69'
'4'='70-74'
'5'='75-79'
'6'='80-84'
'7'='85-89'
'8'='90-94'
'9'='95+'
;

value $sex
'M'='Male'
'F'='Female'
;

value $race
'1'='White'
'2'='African American'
'3'='Other'
'0'='Unknown'
;

value $mdst
'1'='Aged without ESRD'
'2'='Aged with ESRD'
'3'='Disabled without ESRD'
'4'='Disabled with ESRD'
'5'='ESRD only'
;
```

```
    value $creg
    '0'='Puero Rico and Vergin Island'
    '1'='Northeast'
    OBSERVE -- CONTINUED

    '2'='Midwest'
    '3'='South'
    '4'='West'
    ;

    value $urbn
    '0'='Rural'
    '1'='Urban'
    ;

    value $dual
    '0'='Not Buy-in'
    '1'='Buy-in'
    ;
    run;

      /* START: assign macro variables to specify begin and end year
         of summary PSI data */

    %let begyr=1992;
    %let endyr=2001;
    %let file=SUM9201; /* should match the last node of the output DSN */

      /* END: assign macro variables to specify begin and end year
         of summary PSI data */

      /* START: macro program to create the list of observed outcome variables
*/

    %macro varlist;
        %do j=1 %to &tot;
           opps&&y&i..&&x&j
        %end;;

        label
        %do j=1 %to &tot;
           opps&&y&i..&&x&j="&&z&j"
        %end;;

        format
        %do j=1 %to &tot;
           opps&&y&i..&&x&j 13.8
        %end;;
    %mend;

       /* END: macro program to create the list of observed outcome variables
*/


       /* START: macro program to produce the observed rates */

    %macro observe;
```

```
   /* start: create macro variables to handle multiple years' data */

%let k=1;
%do i=&begyr %to &endyr;
  %if &i<2000 %then %let x&k=%eval(&i-1900);
  %else %let x&k=0%eval(&i-2000);
```
OBSERVE -- CONTINUED
```
  %let z&k=&i;
  %let k=%eval(&k+1);
%end;
%let tot=%eval(&k-1);

   /* end: create macro variables to handle multiple years' data */

   /* start: create macro variables to handle multiple PSIs */

%do i=1 %to 16;
  %if &i<10 %then %let y&i=0%eval(&i);
  %else %let y&i=%eval(&i);
%end;

   /* end: create macro variables to handle multiple PSIs */

   /* start: create macro variables for the PSI labels */

%let t1=Indicator # 01: COMPLICATIONS OF ANESTHESIA;
%let t2=Indicator # 02: DEATH IN LOW MORTALITY DRGS;
%let t3=Indicator # 03: DECUBITUS ULCER;
%let t4=Indicator # 04: FAILURE TO RESCUE;
%let t5=Indicator # 05: FOREIGN BODY LEFT IN DURING PROCEDURE;
%let t6=Indicator # 06: IATROGENIC PNEUMOTHORAX;
%let t7=Indicator # 07: INFECTION DUE TO MEDICAL CARE;
%let t8=Indicator # 08: POST-OPERATIVE HIP FRACTURE;
%let t9=Indicator # 09: POST-OPERATIVE HEMORRHAGE OR HEMATOMA;
%let t10=Indicator # 10: POST-OP PHYSIOLOGIC AND METABOLIC DERANGEMENTS;
%let t11=Indicator # 11: POST-OPERATIVE RESPIRATORY FAILURE;
%let  t12=Indicator  #  12:  POST-OP  PULMONARY  EMBOLISM  OR  DEEP  VEIN
THROMBOSIS;
%let t13=Indicator # 13: POST-OPERATIVE SEPSIS;
%let t14=Indicator # 14: POST-OPERATIVE WOUND DEHISCENCE;
%let t15=Indicator # 15: TECHNICAL DIFFICULTY WITH PROCEDURE;
%let t16=Indicator # 16: TRANSFUSION REACTION;

   /* end: create macro variables for the PSI labels */

   /* start: data steps to read obs with relevent _type_ */

data &file;
  set in1.&file(where=(_type_ in (0,1,2,4,8,16,64,128)));
run;

   /* end: data steps to read obs with relevent _type_ */

   /* start: use SAS ODS to output the reports in HTML format */

ods html file=report;
```

```
%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'Overall US';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=0)) label;
    var %varlist;
  run;
%end;
```

OBSERVE -- CONTINUED

```
%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By Age';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=128)) label;
    var agecat %varlist;
    format agecat $age.;
  run;
%end;


%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By Sex';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=64)) label;
    var sexcat %varlist;
    format sexcat $sex.;
  run;
%end;

%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By Medicaid Buy-in Status';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=1)) label;
    var dual %varlist;
    format dual $dual.;
  run;
%end;

%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By Reason for Entitlement';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=2)) label;
    var mdst %varlist;
    format mdst $mdst.;
  run;
%end;

%do i=1 %to 16;
```

```
  title 'Un-adjusted Rates';
  title2 'By Race';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=4)) label;
    var race %varlist;
    format race $race.;
  run;
%end;

%do i=1 %to 16;
  title 'Un-adjusted Rates';
```
OBSERVE -- CONTINUED
```
  title2 'By Urban/Rural Status';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=8)) label;
    var urbn %varlist;
    format urbn $urbn.;
  run;
%end;

%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By State';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=16)) label;
    var state %varlist;
    format state $st.;
  run;
%end;

%do i=1 %to 16;
  title 'Un-adjusted Rates';
  title2 'By Region';
  title3 "&&t&i";

  proc print noobs data=&file.(where=(_type_=32)) label;
    var creg %varlist;
    format creg $creg.;
  run;
%end;

ods html close;

   /* end: use SAS ODS to output the reports in HTML format */

%mend;

   /* END: macro program to produce the observed rates */

%observe;    /* call macro to produce the observed rates */
```

ADJUST

```
//TRA0AD JOB (4AGDCH77010),'NHK',CLASS=M,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.COMBINE.SUM9201,DISP=SHR
//REPORT DD DSN=TRA0.@MSQ0000.HTML.ADJUST.PDS(SUM9201),
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(100,100,3),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

proc format;
value $st
'01'='AL' '02'='AK' '03'='AZ' '04'='AR' '05'='CA'
'06'='CO' '07'='CT' '08'='DE' '09'='DC' '10'='FL'
'11'='GA' '12'='HI' '13'='ID' '14'='IL' '15'='IN'
'16'='IA' '17'='KS' '18'='KY' '19'='LA' '20'='ME'
'21'='MD' '22'='MA' '23'='MI' '24'='MN' '25'='MS'
'26'='MO' '27'='MT' '28'='NE' '29'='NV' '30'='NH'
'31'='NJ' '32'='NM' '33'='NY' '34'='NC' '35'='ND'
'36'='OH' '37'='OK' '38'='OR' '39'='PA' '40'='RQ'
'41'='RI' '42'='SC' '43'='SD' '44'='TN' '45'='TX'
'46'='UT' '47'='VT' '48'='VI' '49'='VA' '50'='WA'
'51'='WV' '52'='WI' '53'='WY'
;

value $age
'1'='0-54'
'2'='55-64'
'3'='65-69'
'4'='70-74'
'5'='75-79'
'6'='80-84'
'7'='85-89'
'8'='90-94'
'9'='95+'
;

value $sex
'1'='Male'
'2'='Female'
;

value $race
'1'='White'
'2'='African American'
'3'='Other'
'0'='Unknown'
;

value $mdst
'1'='Aged without ESRD'
'2'='Aged with ESRD'
'3'='Disabled without ESRD'
'4'='Disabled with ESRD'
'5'='ESRD only'
;
```

```
    value $creg
    '0'='Puero Rico and Vergin Island'
    '1'='Northeast'
    ADJUST -- CONTINUED

    '2'='Midwest'
    '3'='South'
    '4'='West'
    ;

    value $urbn
    '0'='Rural'
    '1'='Urban'
    ;

    value $dual
    '0'='Not Buy-in'
    '1'='Buy-in'
    ;
    run;

      /*  START: data step to create files for age-sex adjustments */
      /*  1999 age-sex populations are hard coded */
      /*  Data Source: D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99 */

    data totp(keep=pop) /* total MQMS population */
         sexp(keep=sexcat pop) /* standard MQMS population by sex */
         agep(keep=agecat pop) /* standard MQMS population by age-groups */
          agesexp(keep=agecat sexcat pop) /* standard MQMS populations by age-
sex */
          adjasp(keep=agecat sexcat pop); /* revised poplns for state by age-
sex */
      input
      @1  agecat   $1.
      @11 sexcat   $1.
      @19 pop       9.
      @;

      if _n_=1 then output totp;
      else if _n_<=3 then output sexp;
      else if _n_<=12 then output agep;
      else if _n_<=30 then output agesexp;
      else output adjasp;
      cards;
                       31471880
              1        13607588
              2        17864292
    1                   2986590
    2                   1862336
    3                   7008112
    4                   6717803
    5                   5646458
    6                   3785275
    7                   2172280
    8                    898468
    9                    394558
    1         1         1739165
    1         2         1247425
```

```
2         1        1013695
2         2         848641
3         1        3262650
3         2        3745462
4         1        2957419
4         2        3760384
```
ADJUST -- CONTINUED

```
5         1        2299257
5         2        3347201
6         1        1375789
6         2        2409486
7         1         657825
7         2        1514455
8         1         215105
8         2         683363
9         1          86683
9         2         307875
1         1        2752860
1         2        2096066
2         1        6220069
2         2        7505846
3         1        3675046
3         2        5756687
4         1         959613
4         2        2505693
;
run;


  /* END: data step to create files for age-sex adjustments */


  /* START: assign macro variables to specify begin and end year
     of summary PSI data */

%let begyr=1992;
%let endyr=2001;
%let file=SUM9201; /* should match the last node of the input DSN */

  /* END: assign macro variables to specify begin and end year
     of summary PSI data */

  /* START: macro program to create the list of adjusted outcome variables
*/

%macro varlist1;
  %do i=1 %to &tot;
    %do j=1 %to 16;
        apps&&y&j..&&x&i
    %end;
  %end;;
%mend;

%macro varlist2;
    %do j=1 %to &tot;
       apps&&y&i..&&x&j
    %end;;
```

```
        label
        %do j=1 %to &tot;
           apps&&y&i..&&x&j="&&z&j"
        %end;;

        format
        %do j=1 %to &tot;
           apps&&y&i..&&x&j 13.8
```
ADJUST -- CONTINUED
```
        %end;;
    %mend;

       /* END: macro program to create the list of adjusted outcome variables
*/

       /* START: macro program to produce the adjusted rates */

    %macro adjust;

       /* start: create macro variables to handle multiple years' data */

    %let k=1;
    %do i=&begyr %to &endyr;
      %if &i<2000 %then %let x&k=%eval(&i-1900);
      %else %let x&k=0%eval(&i-2000);
      %let z&k=&i;
      %let k=%eval(&k+1);
    %end;
    %let tot=%eval(&k-1);

       /* end: create macro variables to handle multiple years' data */

       /* start: create macro variables to handle multiple PSIs */

    %do i=1 %to 16;
      %if &i<10 %then %let y&i=0%eval(&i);
      %else %let y&i=%eval(&i);
    %end;

       /* end: create macro variables to handle multiple PSIs */

       /* start: create macro variables for the PSI labels */

    %let t1=Indicator # 01: COMPLICATIONS OF ANESTHESIA;
    %let t2=Indicator # 02: DEATH IN LOW MORTALITY DRGS;
    %let t3=Indicator # 03: DECUBITUS ULCER;
    %let t4=Indicator # 04: FAILURE TO RESCUE;
    %let t5=Indicator # 05: FOREIGN BODY LEFT IN DURING PROCEDURE;
    %let t6=Indicator # 06: IATROGENIC PNEUMOTHORAX;
    %let t7=Indicator # 07: INFECTION DUE TO MEDICAL CARE;
    %let t8=Indicator # 08: POST-OPERATIVE HIP FRACTURE;
    %let t9=Indicator # 09: POST-OPERATIVE HEMORRHAGE OR HEMATOMA;
    %let t10=Indicator # 10: POST-OP PHYSIOLOGIC AND METABOLIC DERANGEMENTS;
    %let t11=Indicator # 11: POST-OPERATIVE RESPIRATORY FAILURE;
    %let  t12=Indicator  #  12:  POST-OP  PULMONARY  EMBOLISM  OR  DEEP  VEIN
THROMBOSIS;
```

```
%let t13=Indicator # 13: POST-OPERATIVE SEPSIS;
%let t14=Indicator # 14: POST-OPERATIVE WOUND DEHISCENCE;
%let t15=Indicator # 15: TECHNICAL DIFFICULTY WITH PROCEDURE;
%let t16=Indicator # 16: TRANSFUSION REACTION;

   /* end: create macro variables for the PSIs label       */

   /* start: create summary files of observed rates with age-sex
             interactions, and age-sex interactions with other
             individual characteristics */

   /* two separate datasets are created -- 1) without state-age-sex
ADJUST -- CONTINUED

      interactions and 2) with state-age-sex interactions. */

data obsrate1(where=(_type_ ne 208)) /* without state-age-sex ints */
     obsrate2(where=(_type_=208));  /* with state-age-sex interactions */
  set in1.&file(keep=agecat sexcat creg state urbn race mdst dual _type_
  %do i=1 %to &tot;
    %do j=1 %to 16;
        tpps&&y&j..&&x&i
        ppps&&y&j..&&x&i
        opps&&y&j..&&x&i
    %end;
  %end;);

   /* end: create summary files of observed rates with age-sex
           interactions, and age-sex interactions with other
           individual characteristics */

  if _type_ in (192,193,194,196,200,208,224);

   /* start: recode sex to match the sex in agesexp and adjasp files */

  if sexcat='M' then sexcat='1';
  else if sexcat='F' then sexcat='2';

   /* end: recode sex to match the sex in agesexp and adjasp files */

run;

   /* end: create files of observed rates with age-sex interactions */

   /* start: construct ratios of sex-specific populn to overall populn */

data sexp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set sexp;
  prop=pop/totpop;
run;

   /* end: construct ratios of sex-specific populn to overall populn */

   /* start: construct ratios of age-specific populn to overall populn */

data agep;
  if _n_=1 then set totp(rename=(pop=totpop));
```

```
  set agep;
  prop=pop/totpop;
run;

   /* end: construct ratios of age-specific populn to overall populn */

data agesexp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set agesexp;
  prop=pop/totpop;
run;

data adjasp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set adjasp;
```
ADJUST -- CONTINUED
```
  prop=pop/totpop;
run;

   /* end: construct ratios of age-sex-specific populn to overall populn
*/

   /*  start:  fold  some  age  categories  in  the  state-specific  summary
dataset,
            drop the observed rates */

data obsrate2;
  set obsrate2(rename=(agecat=age));
  if age in ('1','2') then agecat='1';
  else if age in ('3','4') then agecat='2';
  else if age in ('5','6') then agecat='3';
  else if age in ('7','8','9') then agecat='4';
  drop
  %do i=1 %to &tot;
    %do j=1 %to 16;
        opps&&y&j..&&x&i
    %end;
  %end;;
run;

   /* end: fold some age categories in the state-specific summary dataset,
           drop the observed rates */

    /* start: sum the numerators and denominators in the state-specific
              summary file based on the revised age categories. Create
              new summary file */

proc sort data=obsrate2;
  by state agecat sexcat;
run;

proc means noprint data=obsrate2;
  var
  %do i=1 %to &tot;
    %do j=1 %to 16;
        tpps&&y&j..&&x&i ppps&&y&j..&&x&i
    %end;
```

```
      %end;;
      by state agecat sexcat;
      output out=obsrate2(drop=_:)
             sum=
             ;
    run;

        /* end: sum the numerators and denominators in the state-specific
                summary file based on the revised age categories. Create
                new summary file */

        /* start: calculate the new observed and adjusted rates based on
revised
                  age-sex categories in the new state-specific summary file.
*/

    proc sort data=obsrate2;
      by agecat sexcat;
    ADJUST -- CONTINUED

    run;

    data adjrate2;
      merge obsrate2(in=a)
            adjasp(in=b)
            ;
      by agecat sexcat;
      if a;
      %do i=1 %to &tot;
        %do j=1 %to 16;
            opps&&y&j..&&x&i=tpps&&y&j..&&x&i / ppps&&y&j..&&x&i; /* obs rate
*/
            apps&&y&j..&&x&i=opps&&y&j..&&x&i * prop;             /* adj rate
*/
        %end;
      %end;
    run;

        /* end: calculate the new observed rates based on revised age-sex
                categories in the new state-specific summary file. */

        /* start: calculate adjusted rates based on standard MQMS age-sex
           categories in the non-state-specific file. */

    proc sort data=obsrate1;
      by agecat sexcat;
    run;

    data adjrate1;
      merge obsrate1(in=a)
            agesexp(in=b)
            ;
      by agecat sexcat;
      if a;
      %do i=1 %to &tot;
        %do j=1 %to 16;
            apps&&y&j..&&x&i=opps&&y&j..&&x&i * prop;
        %end;
```

```
  %end;
run;

   /* end: calculate adjusted rates based on standard MQMS age-sex
       categories in the non-state-specific file. */

   /* start: use SAS ODS to output the reports in HTML format */

ods html file=report;

   /* start: summarize the adjusted rate with obs containing age-sex-only
             interactions to create file with age-sex-adjusted rates
             for overall US. print the adjusted rates for each PSI */

proc means noprint data=adjrate1(where=(_type_=192));
  var %varlist1;
  output out=all(drop=_:)
         sum=
         ;
run;
```

ADJUST -- CONTINUED

```
%do i=1 %to 16;

  title 'Age-sex-adjusted Rates';
  title2 'Overall US';
  title3 "&&t&i";

  proc print noobs data=all label;
     var %varlist2;
  run;
%end;

   /* end: summarize the adjusted rate with obs containing age-sex-only
             interactions to create file with age-sex-adjusted rates
             for overall US. print the adjusted rates for each PSI */

   /* start: summarize the adjusted rate with obs containing age-sex-only
             interactions by age categories to create file with sex-
adjusted
              rates for age categories. print the adjusted rates for each
PSI */

proc sort data=obsrate1(where=(_type_ =192)) out=age;
  by sexcat;
run;

data age;
  merge age(in=a)
        sexp(in=b)
        ;
  by sexcat;
  if a;
  %do i=1 %to &tot;
    %do j=1 %to 16;
        apps&&y&j..&&x&i=opps&&y&j..&&x&i * prop;
    %end;
```

```
  %end;
run;

proc sort data=age;
  by agecat;
run;

proc means noprint data=age;
  var %varlist1;
  by agecat;
  output out=age(drop=_:)
         sum=
         ;
run;

%do i=1 %to 16;

  title 'Sex-adjusted Rates';
  title2 'By Age';
  title3 "&&t&i";

  proc print noobs data=age label;
    var agecat %varlist2;
    format agecat $age.;
```
ADJUST -- CONTINUED
```
  run;
%end;

   /* end: summarize the adjusted rate with obs containing age-sex-only
           interactions by age categories to create file with sex-adjusted
           rates for age categories. print the adjusted rates for each PSI
*/

   /* start: summarize the adjusted rate with obs containing age-sex-only
             interactions by sex categories to create file with age-
adjusted
              rates for sex categories. print the adjusted rates for each
PSI */

proc sort data=obsrate1(where=(_type_ =192)) out=sex;
  by agecat;
run;

data sex;
  merge sex(in=a)
        agep(in=b)
        ;
  by agecat;
  if a;
  %do i=1 %to &tot;
    %do j=1 %to 16;
        apps&&y&j..&&x&i=opps&&y&j..&&x&i * prop;
    %end;
  %end;
run;

proc sort data=sex;
```

```
      by sexcat;
    run;

    proc means noprint data=sex;
      var %varlist1;
      by sexcat;
      output out=sex(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;

      title 'Age-adjusted Rates';
      title2 'By Sex';
      title3 "&&t&i";

      proc print noobs data=sex label;
        var sexcat %varlist2;
        format sexcat $sex.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing age-sex-only
               interactions by sex categories to create file with age-adjusted
               rates for sex categories. print the adjusted rates for each PSI
*/
```

ADJUST -- CONTINUED

```
       /* start: summarize the adjusted rate with obs containing dual-age-sex
                 interactions by dual categories to create file with age-sex-
adjustd
                  rates for dual categories. print the adjusted rates for each
PSI */

    proc sort data=adjrate1(where=(_type_=193)) out=dual;
      by dual agecat sexcat;
    run;

    proc means noprint data=dual;
      var %varlist1;
      by dual;
      output out=dual(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;

      title 'Age-sex-adjusted Rates';
      title2 'By Medicaid Buy-in Status';
      title3 "&&t&i";

      proc print noobs data=dual label;
        var dual %varlist2;
        format dual $dual.;
      run;
```

```
    %end;

       /* end: summarize the adjusted rate with obs containing dual-age-sex
                interactions by dual categories to create file with age-sex-
adjusted
               rates for dual categories. print the adjusted rates for each
PSI */


       /* start: summarize the adjusted rate with obs containing mdst-age-sex
                 interactions by mdst categories to create file with age-sex-
adjustd
                 rates for mdst categories. print the adjusted rates for each
PSI */


    proc sort data=adjrate1(where=(_type_=194)) out=mdst;
      by mdst agecat sexcat;
    run;

    proc means noprint data=mdst;
      var %varlist1;
      by mdst;
      output out=mdst(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      title 'Age-sex-adjusted Rates';
    ADJUST -- CONTINUED

      title2 'By Reason for Entitlement';
      title3 "&&t&i";

      proc print noobs data=mdst label;
        var mdst %varlist2;
        format mdst $mdst.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing mdst-age-sex
               interactions by mdst categories to create file with age-sex-
adjusted
                rates for mdst categories. print the adjusted rates for each
PSI */


       /* start: summarize the adjusted rate with obs containing race-age-sex
                 interactions by race categories to create file with age-sex-
adjustd
                 rates for race categories. print the adjusted rates for each
PSI */

    proc sort data=adjrate1(where=(_type_=196)) out=race;
      by race agecat sexcat;
    run;
```

```
    proc means noprint data=race;
      var %varlist1;
      by race;
      output out=race(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      title 'Age-sex-adjusted Rates';
      title2 'By Race';
      title3 "&&t&i";

      proc print noobs data=race label;
        var race %varlist2;
        format race $race.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing race-age-sex
                interactions by race categories to create file with age-sex-
adjusted
                rates for race categories. print the adjusted rates for each
PSI */

       /* start: summarize the adjusted rate with obs containing urbn-age-sex
                 interactions by urbn categories to create file with
                 age-sex-adjusted rates for urbn categories.  print the
                 adjusted rates for each PSI */

    proc sort data=adjrate1(where=(_type_=200)) out=urbn;
      by urbn agecat sexcat;
    run;
```
ADJUST -- CONTINUED

```
    proc means noprint data=urbn;
      var %varlist1;
      by urbn;
      output out=urbn(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      title 'Age-sex-adjusted Rates';
      title2 'By Urban/Rural Status';
      title3 "&&t&i";

      proc print noobs data=urbn label;
        var urbn %varlist2;
        format urbn $urbn.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing urbn-age-sex
                interactions by urbn categories to create file with age-sex-
adjusted
```

```
                rates for urbn categories. print the adjusted rates for each
PSI */

       /* start: summarize the adjusted rate with obs containing state-age-sex
                 interactions by states to create file with age-sex-adjusted
                  rates for the states. print the adjusted rates for each PSI
*/

    proc sort data=adjrate2 out=state;
      by state agecat sexcat;
    run;

    proc means noprint data=state;
      var %varlist1;
      by state;
      output out=state(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      title 'Age-sex-adjusted Rates';
      title2 'By State';
      title3 "&&t&i";

      proc print noobs data=state label;
        var state %varlist2;
        format state $st.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing state-age-sex
               interactions by states to create file with age-sex-adjusted
               rates for the states. print the adjusted rates for each PSI */

       /* start: summarize the adjusted rate with obs containing creg-age-sex
    ADJUST -- CONTINUED

                interactions by creg categories to create file with age-sex-
adjustd
                  rates for creg categories. print the adjusted rates for each
PSI */

    proc sort data=adjrate1(where=(_type_=224)) out=creg;
      by creg agecat sexcat;
    run;

    proc means noprint data=creg;
      var %varlist1;
      by creg;
      output out=creg(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      title 'Age-sex-adjusted Rates';
      title2 'By Region';
```

```
      title3 "&&t&i";

      proc print noobs data=creg label;
        var creg %varlist2;
        format creg $creg.;
      run;
    %end;

       /* end: summarize the adjusted rate with obs containing creg-age-sex
                interactions by creg categories to create file with age-sex-
adjusted
               rates for creg categories. print the adjusted rates for each
PSI */

    ods html close;

       /* end: use SAS ODS to output the reports in HTML format */

    %mend;
       /* END: macro program to produce the adjusted rates */
    %adjust /* call macro to produce adjusted rates */
```

PROGRAMS –

PREVENTION QUALITY INDICATORS (PQI)

TRA0.@MSQ0000.PQI(PQA1B1)

```
//TRA0QA1 JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1992"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR92,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1993"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR93,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1994"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR94,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1995"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR95,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1996"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR96,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
```

```
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1997"'
```
TRA0.@MSQ0000.PQI(PQA1B1) -- CONTINUED

```
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR97,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1998"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR98,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="1999"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR99,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="2000"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR00,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='9000,3000',
//       PARM='SYSPARM="2001"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PSIDAT.MEDPAR01,DISP=SHR
//CONT   DD DSN=TRA0.@MSQ0000.PQI(CONTROL),DISP=SHR
//FMTS   DD DSN=TRA0.@MSQ0000.PQI(PQFMTS),DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA1),
//          DISP=(SHR,PASS,KEEP)
```

TRA0.@MSQ0000.PQI(PQSASA1)

```
 *=============================================================;
 *  Title:  PROGRAM A1  ASSIGN AHRQ PREVENTION QUALITY INDICATORS
 *          TO INPATIENT RECORDS
 *
 *  Description:;
 *         ASSIGNS THE OUTCOME OF INTEREST AND POPULATION AT RISK
 *         INDICATORS TO INPATIENT RECORDS;
 *
 *                >>>  VERSION 2.1  <<<
 *
 *  USER NOTE:  Make sure you have created format library
 *                    (using FORMAT.SAS)
 *                 BEFORE running this program.
 *
 *=============================================================;
 *FILENAME CONTROL 'D:\Files\8908\PQIs\TEST\CONTROL.SAS';
 *<===USER MUST modify;

 %INCLUDE CONT;
 %INCLUDE FMTS;

OPTIONS COMPRESS=YES;

TITLE2 'PROGRAM A1';
TITLE3 'AHRQ PREVENTION QUALITY INDICATORS: ASSIGN TO INPATIENT DATA';

* ------------------------------------------------------------- ;
* --- PREVENTION QUALITY INDICATOR NAMING CONVENTION:        --- ;
* --- THE FIRST LETTER IDENTIFIES THE PREVENTION QUALITY     --- ;
* --- INDICATOR AS ONE OF THE FOLLOWING:
              (O) OBSERVED RATES
              (R) RISK-ADJUSTED RATES
              (S) SMOOTHED RATES
              (T) NUMERATOR ("TOP")
              (P) POPULATION ("POP")
* --- THE SECOND LETTER REFERS TO THE INDICATOR              --- ;
* --- SUBTYPE, (A)MBULATORY CARE SENSITIVE CONDITION. THE    --- ;
* --- NEXT TWO CHARACTERS ARE ALWAYS 'PQ'. THE LAST TWO      --- ;
* --- DIGITS ARE THE INDICATOR NUMBER (WITHIN THAT SUBTYPE). --- ;
* ------------------------------------------------------------- ;

DATA OUT1.&outfile1.
               (KEEP=KEY SEXCAT AGE AGECAT RACECAT HOSPSTCO
                     HOSPID DRG MDC AREA
                     TAPQ01-TAPQ16);
 SET IN1.&infile1.
               (KEEP=KEY DRG MDC SEX AGE HOSPSTCO
                     HOSPID ASOURCE DISP RACE
                     DX1-DX&NDX. PR1-PR&NPR.);

 * ------------------------------------------------------------- ;
 * --- DIAGNOSIS AND PROCEDURE MACROS -------------------------- ;
 * ------------------------------------------------------------- ;

 %MACRO MDX(FMT);
```

```
 (%DO I = 1 %TO &NDX.-1;
  (PUT(DX&I.,&FMT.) = '1') OR
  %END;
```
TRA0.@MSQ0000.PQI(PQSASA1) -- CONTINUED

```
  (PUT(DX&NDX.,&FMT.) = '1'))

 %MEND;

 %MACRO MDX1(FMT);

 ((PUT(DX1,&FMT.) = '1'))

 %MEND;

 %MACRO MDX2(FMT);

 (%DO I = 2 %TO &NDX.-1;
  (PUT(DX&I.,&FMT.) = '1') OR
  %END;
  (PUT(DX&NDX.,&FMT.) = '1'))

 %MEND;

 %MACRO MPR(FMT);

 (%DO I = 1 %TO &NPR.-1;
  (PUT(PR&I.,&FMT.) = '1') OR
  %END;
  (PUT(PR&NPR.,&FMT.) = '1'))

 %MEND;

 * ------------------------------------------------------------- ;
 * --- DELETE RECORDS WITH MISSING VALUES FOR AGE OR SEX--------- ;
 * ------------------------------------------------------------- ;
 IF SEX LT 0 THEN DELETE;
 IF AGE LT 0 THEN DELETE;

 * ------------------------------------------------------------- ;
 * --- DEFINE GEOGRAPHIC AREAS ---------------------------------- ;
 * ------------------------------------------------------------- ;
 ATTRIB AREA LENGTH=$5
   LABEL='AREA ASSIGNED FROM HOSPSTCO';

 AREA = HOSPSTCO;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: RACE CATEGORY ------------------------- ;
 * ------------------------------------------------------------- ;
 ATTRIB RACECAT LENGTH=$1
   LABEL='(W)hite (B)lack (H)ispanic (O)ther';

 SELECT (RACE);
   WHEN (1)  RACECAT = 'W';
   WHEN (2)  RACECAT = 'B';
   WHEN (3)  RACECAT = 'H';
   OTHERWISE RACECAT = 'O';
```

```
 END; * SELECT RACE ;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: AGE CATEGORY  ----------------------- ;
 * ------------------------------------------------------------- ;
 ATTRIB AGECAT LENGTH=$1
```
TRA0.@MSQ0000.PQI(PQSASA1) -- CONTINUED
```
   LABEL='AGE CATEGORIES';

  SELECT;
    WHEN (0  <= AGE < 55)  AGECAT = '1';
    WHEN (55 <= AGE < 65)  AGECAT = '2';
    WHEN (65 <= AGE < 70)  AGECAT = '3';
    WHEN (70 <= AGE < 75)  AGECAT = '4';
    WHEN (75 <= AGE < 80)  AGECAT = '5';
    WHEN (80 <= AGE < 85)  AGECAT = '6';
    WHEN (85 <= AGE < 90)  AGECAT = '7';
    WHEN (90 <= AGE < 95)  AGECAT = '8';
    WHEN (95 <= AGE     )  AGECAT = '9';
  END; * SELECT AGE ;

 * ------------------------------------------------------------- ;
 * --- DEFINE STRATIFIER: SEX CATEGORY  ----------------------- ;
 * ------------------------------------------------------------- ;
 ATTRIB SEXCAT LENGTH=$1
   LABEL='(M)ale (F)emale';

 SELECT (SEX);
   WHEN (1)  SEXCAT = 'M';
   WHEN (2)  SEXCAT = 'F';
 END; * SELECT SEX ;


 * ------------------------------------------------------------- ;
 * --- DEFINE AREA LEVEL ACSC INDICATORS ---------------------- ;
 * ------------------------------------------------------------- ;

 LENGTH TAPQ01 TAPQ02 TAPQ03 TAPQ04 TAPQ05
        TAPQ06 TAPQ07 TAPQ08 TAPQ09 TAPQ10
        TAPQ11 TAPQ12 TAPQ13 TAPQ14 TAPQ15
        TAPQ16
          3 ;

 LABEL
 TAPQ01 = 'DIABETES SHORT TRM COMPLICATN (Numeratr)'
 TAPQ02 = 'PERFORATED APPENDIX           (Numeratr)'
 TAPQ03 = 'DIABETES LONG TERM COMPLICATN (Numeratr)'
 TAPQ04 = 'PEDIATRIC ASTHMA              (Numeratr)'
 TAPQ05 = 'COPD                          (Numeratr)'
 TAPQ06 = 'PEDIATRIC GASTROENTERITIS     (Numeratr)'
 TAPQ07 = 'HYPERTENSION                  (Numeratr)'
 TAPQ08 = 'CONGESTIVE HEART FAILURE      (Numeratr)'
 TAPQ09 = 'LOW BIRTH WEIGHT              (Numeratr)'
 TAPQ10 = 'DEHYDRATION                   (Numeratr)'
 TAPQ11 = 'BACTERIAL PNEUMONIA           (Numeratr)'
 TAPQ12 = 'URINARY INFECTION             (Numeratr)'
 TAPQ13 = 'ANGINA                        (Numeratr)'
```

```
 TAPQ14 = 'DIABETES UNCONTROLLED         (Numeratr)'
 TAPQ15 = 'ADULT ASTHMA                  (Numeratr)'
 TAPQ16 = 'LOWER EXTREMITY AMPUTATION    (Numeratr)'
        ;


 * ------------------------------------------------------------- ;
 * --- CONSTRUCT AREA LEVEL ACSC INDICATORS --------------------- ;
 * ------------------------------------------------------------- ;
```

TRA0.@MSQ0000.PQI(PQSASA1) -- CONTINUED

```
   * --- DIABETES SHORT TERM COMPLICATION  --- ;

   IF MDC NOTIN (14,15) AND (AGE >= 18) AND

      %MDX1($ACDIASD.) THEN

        TAPQ01 = 1;

   * --- PERFORATED APPENDIX               --- ;

   IF MDC NOTIN (14,15) AND %MDX($ACSAP2D.) THEN DO;

      TAPQ02 = 0;

        IF %MDX($ACSAPPD.) THEN TAPQ02 = 1;

   END;

   * --- DIABETES LONG TERM COMPLICATION   --- ;

   IF MDC NOTIN (14,15) AND (AGE >= 18) AND

      %MDX1($ACDIALD.) THEN

        TAPQ03 = 1;

   * --- PEDIATRIC ASTHMA                  --- ;

   IF MDC NOTIN (14,15) AND (0 <= AGE < 18) AND

      %MDX1($ACSASTD.) THEN

        TAPQ04 = 1;

   * --- COPD                              --- ;

   IF MDC NOTIN (14,15) AND (AGE >= 18) AND

      (%MDX1($ACCOPDD.) OR

       (%MDX1($ACCPD2D.) AND %MDX2($ACCOPDD.))

      ) THEN

        TAPQ05 = 1;
```

```
   * --- PEDIATRIC GASTROENTERITIS         --- ;
   IF MDC NOTIN (14,15) AND (0 <= AGE < 18) AND
      %MDX1($ACPGASD.) THEN
        TAPQ06 = 1;
   * --- HYPERTENSION                      --- ;
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
```

```
      %MDX1($ACSHYPD.) AND NOT %MPR($ACSHYPP.) THEN
        TAPQ07 = 1;
   * --- CONGESTIVE HEART FAILURE          --- ;
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
      %MDX1($MRTCHFD.) AND NOT %MPR($MRTCHFP.) THEN
      TAPQ08 = 1;
   * --- LOW BIRTH WEIGHT                  --- ;
   IF MDC IN (15) THEN DO;
      TAPQ09 = 0;
      IF %MDX($ACSLBWD.) THEN TAPQ09 = 1;
   END;
   * --- DEHYDRATION                       --- ;
   IF MDC NOTIN (14,15) AND %MDX1($ACSDEHD.) THEN
      TAPQ10 = 1;
   * --- BACTERIAL PNEUMONIA               --- ;
   IF MDC NOTIN (14,15) AND %MDX1($ACSBACD.) AND
      NOT %MDX($ACSBA2D.)  THEN
      TAPQ11 = 1;
   * --- URINARY INFECTION                 --- ;
   IF MDC NOTIN (14,15) AND %MDX1($ACSUTID.) THEN
      TAPQ12 = 1;
   * --- ANGINA                            --- ;
```

```
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
      %MDX1($ACSANGD.) AND NOT %MPR($NONBLKP.) THEN
         TAPQ13 = 1;
   * --- DIABETES UNCONTROLLED             --- ;
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
      %MDX1($ACDIAUD.) THEN
         TAPQ14 = 1;
   * --- ADULT ASTHMA                      --- ;
```

TRA0.@MSQ0000.PQI(PQSASA1) -- CONTINUED

```
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
      %MDX1($ACSASTD.) THEN
         TAPQ15 = 1;
   * --- LOWER EXTREMITY AMPUTATION       --- ;
   IF MDC NOTIN (14,15) AND (AGE >= 18) AND
      %MPR($ACSLEAP.) AND
      %MDX($ACSLEAD.) AND NOT %MDX($ACLEA2D.) THEN
         TAPQ16 = 1;

 * ------------------------------------------------------------- ;
 * --- EXCLUDE TRANSFERS ------------------------------------- ;
 * ------------------------------------------------------------- ;

 * --- TRANSFER FROM ANOTHER ---------------- ;
 IF (2 <= ASOURCE <= 3) THEN DO;
   TAPQ01 = .;
   TAPQ02 = .;
   TAPQ03 = .;
   TAPQ04 = .;
   TAPQ05 = .;
   TAPQ06 = .;
   TAPQ07 = .;
   TAPQ08 = .;
   TAPQ09 = .;
   TAPQ10 = .;
   TAPQ11 = .;
   TAPQ12 = .;
   TAPQ13 = .;
   TAPQ14 = .;
   TAPQ15 = .;
   TAPQ16 = .;
 END;
```

```
RUN;

* ------------------------------- ;
PROC MEANS DATA=OUT1.&OUTFILE1. N NMISS MIN MAX MEAN SUM;
RUN;

PROC CONTENTS DATA=OUT1.&OUTFILE1. POSITION;
RUN;
```

TRA0.@MSQ0000.PQI(PQMRGB1)

```
//TRA0MRG JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR92,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR92,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1002.DENOMB92,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR93,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR93,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1002.DENOMB93,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR94,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR94,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1002.DENOMB94,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR95,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR95,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1002.DENOMB95,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
```

```
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
```
TRA0.@MSQ0000.PQI(PQMRGB1) -- CONTINUED
```
//       PARM='SYSPARM="96"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR96,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR96,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1002.DENOMB96,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR97,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR97,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1002.DENOMB97,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR98,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR98,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR99,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR99,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
```

```
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR00,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR00,DISP=SHR
```

```
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1002.DENOMB00,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='180000,90000',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR01,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA1.MEDPAR01,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1002.DENOMB01,DISP=SHR
//TEMP   DD DSN=TRA0.@MSQ0000.TEMP,
//          DISP=(NEW,DELETE,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2003/365)
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=MCART,LABEL=(1,SL,EXPDT=2007/365)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQMRGDNM),
//          DISP=(SHR,PASS,KEEP)
//*
```

TRA0.@MSQ0000.PQI(PQMRGDNM)

```
  OPTION LS=132 PS=60;
  %LET YR=&SYSPARM;

  PROC SORT DATA=IN3.DENOMB&YR.(KEEP=Y&YR.MDST
  Y&YR.HICNO Y&YR.CDIV Y&YR.CREG Y&YR.DUAL Y&YR.FEEJ Y&YR.M13
  Y&YR.M14 Y&YR.RACE Y&YR.STATE Y&YR.URBN RENAME=(Y&YR.HICNO=XREF))
  OUT=TEMP.DENOM;
  BY XREF;
  RUN;

  DATA DENOM;
    MERGE TEMP.DENOM (IN=A)
          IN1.FINDR&YR (IN=B)
          ;
    BY XREF;
    IF A AND B;
  RUN;

  PROC SORT DATA=DENOM NODUPKEY;
    BY HICNO;
  RUN;

  PROC SORT DATA=IN1.FINDR&YR OUT=FIND;
     BY HICNO;
  RUN;

  DATA DENOM;
    MERGE DENOM(IN=A)
          FIND(IN=B)
          ;
    BY HICNO;
    IF A AND B;
  RUN;

  PROC DATASETS NOLIST;
    DELETE FIND;
  RUN;

  PROC SORT DATA=DENOM;
     BY XREF;
  RUN;

  DATA OUT1.PS1;
    MERGE IN2.PQA1(IN=A RENAME=(KEY=XREF))
          DENOM(IN=B)
          ;
    BY XREF;
    IF A AND B;
  RUN;
```

TRA0.@MSQ0000.PQI(PQSUMB1)

```
//TRA0SUM JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1002.DENOMB92,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1002.DENOMB93,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1002.DENOMB94,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1002.DENOMB95,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="96"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1002.DENOMB96,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1002.DENOMB97,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
```

```
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98,DISP=SHR
```


```
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1002.DENOMB00,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='18000,9000',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1002.DENOMB01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(500,500),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQM4DNM),
//          DISP=(SHR,PASS,KEEP)
//*
```

TRA0.@MSQ0000.PQI(PQM4DNM)

```
 OPTIONS COMPRESS=YES LS=132 PS=60;
 %LET YR=&SYSPARM;

 TITLE2 'PROGRAM: PQM4DNM';
 TITLE3 'POPULATION COUNTS BY STRATA';

  * ------------------------------------------------------------- ;
  * --- COUNTS OF INDIVIDUALS IN THE BASE ANALYTIC DENOM FILE ---- ;
  * ------------------------------------------------------------- ;

 PROC SUMMARY DATA=IN1.DENOMB&YR.(WHERE=(Y&YR.M14='1'));
 CLASS Y&YR.FAGE Y&YR.SEX Y&YR.CREG Y&YR.STATE Y&YR.URBN Y&YR.RACE
       Y&YR.MDST Y&YR.DUAL;
 VAR   Y&YR.CAGE;
 OUTPUT OUT=POP&YR.(DROP=_:)
        N=CNT&YR.
        ;
 LABEL Y&YR.CAGE='POPULATION COUNT'; /* LABEL THE COUNT VAR */
 RUN;

 DATA POP&YR.;
   SET POP&YR.(RENAME=(Y&YR.FAGE=AGECAT)); /* TO MATCH MEDPAR
                                              VARIABLE NAME */
   IF Y&YR.SEX='1' THEN SEXCAT='M';      /* TO MATCH MEDPAR SEX */
   ELSE IF Y&YR.SEX='2' THEN SEXCAT='F';
   DROP Y&YR.SEX;
 RUN;

 PROC SORT DATA=POP&YR. OUT=OUT1.POP&YR.;
    BY AGECAT SEXCAT Y&YR.CREG Y&YR.STATE Y&YR.URBN Y&YR.RACE
       Y&YR.MDST Y&YR.DUAL;
 RUN;

 PROC CONTENTS;
 RUN;

 PROC PRINT DATA=OUT1.POP&YR.(OBS=50);
 RUN;
```

TRA0.@MSQ0000.PQI(PQA2B1)

```
//TRA0QA2 JOB (MSQ00000000),'NHK',CLASS=H,MSGCLASS=Q
//*
//JS010  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="92"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR92,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR92,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR92,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS020  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="93"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR93,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR93,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR93,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS030  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="94"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR94,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR94,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR94,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS040  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="95"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR95,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR95,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR95,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS050  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="96"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR96,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR96,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR96,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS060  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="97"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR97,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR97,DISP=SHR
```

```
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR97,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
```
TRA0.@MSQ0000.PQI(PQA2B1) -- CONTINUED

```
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS070  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="98"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR98,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR98,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR98,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS080  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="99"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR99,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR99,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR99,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS090  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="00"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR00,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR00,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR00,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
//*
//JS100  EXEC SAS8,OPTIONS='NONEWS',WORK='80000,40000',
//       PARM='SYSPARM="01"'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQMRGDNM.MEDPAR01,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQM4DNM.MEDPAR01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR01,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(1000,1000),RLSE)
//SYSIN  DD DSN=TRA0.@MSQ0000.PQI(PQSASA2),
//          DISP=(SHR,PASS,KEEP)
```

TRA0.@MSQ0000.PQI(PQSASA2)

```
*===================================================================
*  Title:  PROGRAM A2  CALCULATE OBSERVED AREA RATES FOR AHRQ
*          PREVENTION QUALITY INDICATORS
*
*  Description:
*         USES PROC SUMMARY TO CALCULATE OBSERVED RATES FOR
*         PREVENTION QUALITY INDICATORS ACROSS STRATIFIERS:
*
*                 >>>  VERSION 2.1  <<<
*
*===================================================================;

OPTIONS PS=60 LS=132 COMPRESS=YES;
%LET YR=&SYSPARM;

 TITLE2 'PROGRAM A2';
 TITLE3 'AHRQ PREVENTION QUALITY INDICATORS: CALCULATE OBSERVED RATES';
 run;

 * ------------------------------------------------------------- ;
 * --- Calculate the Numerators (T prefix) for the observed rates ;
 * --- and the Denominators (P prefix) for those Prevention       ;
 * --- Quality Indicators that are based on the hospitalized      ;
 * --- population.                                                ;
 * ------------------------------------------------------------- ;

 PROC   SUMMARY DATA=IN1.PS1(WHERE=(Y&YR.M14='1'));
 CLASS  AGECAT SEXCAT Y&YR.CREG Y&YR.STATE Y&YR.URBN Y&YR.RACE
        Y&YR.MDST Y&YR.DUAL;
 VAR    TAPQ01-TAPQ16;
 OUTPUT OUT=PQSUM
        SUM(TAPQ01-TAPQ16)=TAPQ01-TAPQ16
          N(TAPQ02 TAPQ09)=PAPQ02 PAPQ09 ;
 RUN;

 PROC   SORT DATA=PQSUM;
 BY     AGECAT SEXCAT Y&YR.CREG Y&YR.STATE Y&YR.URBN Y&YR.RACE
        Y&YR.MDST Y&YR.DUAL;
 RUN;

 * ------------------------------------------------------------- ;
 * --- ADD THE AREA POPULATION DENOMINATORS --------------------- ;
 * ------------------------------------------------------------- ;


 DATA  OUT1.PQA2;
 MERGE PQSUM(IN=A)
       IN2.POP&YR.(IN=B)
       ;
 BY    AGECAT SEXCAT Y&YR.CREG Y&YR.STATE Y&YR.URBN Y&YR.RACE
       Y&YR.MDST Y&YR.DUAL;
 IF A;

 ARRAY  PPQ{16}  PAPQ01-PAPQ16;

 DO J = 1 to 16;
```

```
   IF J NOTIN(2,9) THEN PPQ{J} = CNT&YR.;
END;
```

TRA0.@MSQ0000.PQI(PQSASA2) -- CONTINUED

```
ARRAY  TPQ{16}  TAPQ01-TAPQ16 ;
ARRAY  OPQ{16}  OAPQ01-OAPQ16 ;

DO J = 1 to 16;
   IF TPQ{J} GT 0 AND PPQ{J} GT 0 THEN OPQ{J} = TPQ{J} / PPQ{J};
   ELSE IF PPQ{J} GT 0 THEN OPQ{J} = 0 ;
END;

LABEL
CNT&YR = 'AREA POPULATION                          '
TAPQ01 = 'DIABETES SHORT TRM COMPLICATN (Numeratr)'
TAPQ02 = 'PERFORATED APPENDIX           (Numeratr)'
TAPQ03 = 'DIABETES LONG TERM COMPLICATN (Numeratr)'
TAPQ04 = 'PEDIATRIC ASTHMA              (Numeratr)'
TAPQ05 = 'COPD                          (Numeratr)'
TAPQ06 = 'PEDIATRIC GASTROENTERITIS     (Numeratr)'
TAPQ07 = 'HYPERTENSION                  (Numeratr)'
TAPQ08 = 'CONGESTIVE HEART FAILURE      (Numeratr)'
TAPQ09 = 'LOW BIRTH WEIGHT              (Numeratr)'
TAPQ10 = 'DEHYDRATION                   (Numeratr)'
TAPQ11 = 'BACTERIAL PNEUMONIA           (Numeratr)'
TAPQ12 = 'URINARY INFECTION             (Numeratr)'
TAPQ13 = 'ANGINA                        (Numeratr)'
TAPQ14 = 'DIABETES UNCONTROLLED         (Numeratr)'
TAPQ15 = 'ADULT ASTHMA                  (Numeratr)'
TAPQ16 = 'LOWER EXTREMITY AMPUTATION    (Numeratr)'
PAPQ01 = 'DIABETES SHORT TRM CMPLICTN(Denominator)'
PAPQ02 = 'PERFORATED APPENDIX        (Denominator)'
PAPQ03 = 'DIABETES LONG TRM CMPLICTN (Denominator)'
PAPQ04 = 'PEDIATRIC ASTHMA           (Denominator)'
PAPQ05 = 'COPD                       (Denominator)'
PAPQ06 = 'PEDIATRIC GASTROENTERITIS  (Denominator)'
PAPQ07 = 'HYPERTENSION               (Denominator)'
PAPQ08 = 'CONGESTIVE HEART FAILURE   (Denominator)'
PAPQ09 = 'LOW BIRTH WEIGHT           (Denominator)'
PAPQ10 = 'DEHYDRATION                (Denominator)'
PAPQ11 = 'BACTERIAL PNEUMONIA        (Denominator)'
PAPQ12 = 'URINARY INFECTION          (Denominator)'
PAPQ13 = 'ANGINA                     (Denominator)'
PAPQ14 = 'DIABETES UNCONTROLLED      (Denominator)'
PAPQ15 = 'ADULT ASTHMA               (Denominator)'
PAPQ16 = 'LOWER EXTREMITY AMPUTATION (Denominator)'
OAPQ01 = 'DIABETES SHORT TRM COMPLICATN (Observed)'
OAPQ02 = 'PERFORATED APPENDIX           (Observed)'
OAPQ03 = 'DIABETES LONG TERM COMPLICATN (Observed)'
OAPQ04 = 'PEDIATRIC ASTHMA              (Observed)'
OAPQ05 = 'COPD                          (Observed)'
OAPQ06 = 'PEDIATRIC GASTROENTERITIS     (Observed)'
OAPQ07 = 'HYPERTENSION                  (Observed)'
OAPQ08 = 'CONGESTIVE HEART FAILURE      (Observed)'
OAPQ09 = 'LOW BIRTH WEIGHT              (Observed)'
OAPQ10 = 'DEHYDRATION                   (Observed)'
OAPQ11 = 'BACTERIAL PNEUMONIA           (Observed)'
```

```
 OAPQ12 = 'URINARY INFECTION            (Observed)'
 OAPQ13 = 'ANGINA                       (Observed)'
 OAPQ14 = 'DIABETES UNCONTROLLED        (Observed)'
 OAPQ15 = 'ADULT ASTHMA                 (Observed)'
 OAPQ16 = 'LOWER EXTREMITY AMPUTATION   (Observed)'
 _TYPE_ = 'Stratification level                   '
```
TRA0.@MSQ0000.PQI(PQSASA2) -- CONTINUED

```
       ;
 DROP J _FREQ_;

 RUN;

 * ------------------------------------------------------------- ;
 * --- CONTENTS OF AREA OBSERVED MEANS FILE -------------------- ;
 * ------------------------------------------------------------- ;

 PROC CONTENTS DATA=OUT1.PQA2 POSITION;
 RUN;

 PROC MEANS DATA=OUT1.PQA2 N NMISS MIN MAX MEAN SUM;
 RUN;

 * ------------------------------------------------------------- ;
 * --- PRINT AREA OBSERVED MEANS FILE -------------------------- ;
 * ------------------------------------------------------------- ;

 PROC PRINT DATA=OUT1.PQA2(OBS=24);
 RUN;
```

TRA0.@MSQ0000.PQI(PQCOMBN)

```
//TRA0QM JOB (4AGDCH77010),'NHK',CLASS=K,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR92,DISP=SHR
//IN2    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR93,DISP=SHR
//IN3    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR94,DISP=SHR
//IN4    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR95,DISP=SHR
//IN5    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR96,DISP=SHR
//IN6    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR97,DISP=SHR
//IN7    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR98,DISP=SHR
//IN8    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR99,DISP=SHR
//IN9    DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR00,DISP=SHR
//IN10   DD DSN=TRA0.@MSQ0000.SAS.PQSASA2.MEDPAR01,DISP=SHR
//OUT1   DD DSN=TRA0.@MSQ0000.SAS.PQCOMBN.SUM9201,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(TRK,(10000,1000),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

  /* START: assign macro variables to specify begin and end year
     of summary PQI data */

%let begyr=1992;
%let endyr=2001;
%let file=SUM9201; /* should match the last node of the output DSN */

  /* END: assign macro variables to specify begin and end year
     of summary PQI data */

  /* START: macro program to merge summary PQI data */

%macro process;

  /* start: create macro variables to handle multiple years' data */

%let k=1;
%do i=&begyr %to &endyr;
  %if &i<2000 %then %let x&k=%eval(&i-1900);
  %else %let x&k=0%eval(&i-2000);
  %let k=%eval(&k+1);
%end;
%let tot=%eval(&k-1);

  /* end: create macro variables to handle multiple years' data */

  /* start: create macro variables to handle multiple PQIs */

%do i=1 %to 16;
  %if &i<10 %then %let y&i=0%eval(&i);
  %else %let y&i=%eval(&i);
%end;

  /* end: create macro variables to handle multiple PQIs */

  /* start: data step to merge the summary files */
```

```
data out1.&file.;
  merge
  %do i=1 %to &tot;
```
TRA0.@MSQ0000.PQI(PQCOMBN) -- CONTINUED

```
    in&i..pqa2 (keep=tapq01-tapq16 papq01-papq16 oapq01-oapq16
    y&&x&i..dual y&&x&i..mdst y&&x&i..race y&&x&i..urbn y&&x&i..state
    y&&x&i..creg agecat sexcat _type_
    rename=(y&&x&i..dual=dual y&&x&i..mdst=mdst y&&x&i..race=race
    y&&x&i..urbn=urbn y&&x&i..state=state y&&x&i..creg=creg
    _type_=type&&x&i
    %do j=1 %to 16;
        tapq&&y&j=tapq&&y&j..&&x&i
        papq&&y&j=papq&&y&j..&&x&i
        oapq&&y&j=oapq&&y&j..&&x&i
    %end;))
  %end;;
  by agecat sexcat creg state urbn race mdst dual;
  maxtp=max(of type92-type99, of type00-type&&x&tot);
  mintp=min(of type92-type99, of type00-type&&x&tot);
  if maxtp ne mintp then put type92= type93= type94= type95=
  type96= type97= type98= type99= type00= type01=;
  _type_=maxtp;
  drop maxtp mintp;
  label agecat='Age Group      '
        sexcat='Gender         '
        creg  ='Census Region  '
        urbn  ='Urban/Rural    '
        race  ='Race           '
        mdst  ='Reason for Medicare Eligibility'
        dual  ='Medicare Buy-in'
        state ='State          '
        ;
run;

  /* end: data step to merge the summary files */

%mend;
  /* END: macro program to merge summary PQI data */

%process;  /* call the macro to merge the files */

  /* START: proc steps to examine the data */

  title 'combined summary file';
  title2 "begin year: &begyr. end year: &endyr.";
  proc contents data=out1.&file.;
  run;

  title3 'first 10 observation';
  proc print data=out1.&file.(obs=10);
  run;

  /* END: proc steps to examine the data */

  /* START: steps to examine the _type_ variable */

  proc sort data=out1.&file.(keep=agecat sexcat creg state
```

```
                    urbn race mdst dual _type_) nodupkey out=temp;
      by _type_;
      run;

       title 'values of the _type_ variable marking all possible interactions
of';
```

```
      title2 'age groups, gender, census region, state, urban/rual, race,';
      title3 'Medicare eligibility, and Medicaid buy-in status';

      proc print data=temp;
      run;

      /* END: steps to examine the _type_ variable */
```

TRA0.@MSQ0000.PQI(PQOBSERV)

```
//TRA0OB JOB (4AGDCH77010),'NHK',CLASS=M,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQCOMBN.SUM9201,DISP=SHR
//REPORT DD DSN=TRA0.@MSQ0000.HTML.PQOBSERV.PDS(SUM9201),
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(100,100,3),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

proc format;
value $st
'01'='AL' '02'='AK' '03'='AZ' '04'='AR' '05'='CA'
'06'='CO' '07'='CT' '08'='DE' '09'='DC' '10'='FL'
'11'='GA' '12'='HI' '13'='ID' '14'='IL' '15'='IN'
'16'='IA' '17'='KS' '18'='KY' '19'='LA' '20'='ME'
'21'='MD' '22'='MA' '23'='MI' '24'='MN' '25'='MS'
'26'='MO' '27'='MT' '28'='NE' '29'='NV' '30'='NH'
'31'='NJ' '32'='NM' '33'='NY' '34'='NC' '35'='ND'
'36'='OH' '37'='OK' '38'='OR' '39'='PA' '40'='RQ'
'41'='RI' '42'='SC' '43'='SD' '44'='TN' '45'='TX'
'46'='UT' '47'='VT' '48'='VI' '49'='VA' '50'='WA'
'51'='WV' '52'='WI' '53'='WY'
;

value $age
'1'='0-54'
'2'='55-64'
'3'='65-69'
'4'='70-74'
'5'='75-79'
'6'='80-84'
'7'='85-89'
'8'='90-94'
'9'='95+'
;

value $sex
'M'='Male'
'F'='Female'
;

value $race
'1'='White'
'2'='African American'
'3'='Other'
'0'='Unknown'
;

value $mdst
'1'='Aged without ESRD'
'2'='Aged with ESRD'
'3'='Disabled without ESRD'
'4'='Disabled with ESRD'
'5'='ESRD only'
;
```

```
    value $creg
    '0'='Puero Rico and Vergin Island'
    '1'='Northeast'
```

```
    '2'='Midwest'
    '3'='South'
    '4'='West'
    ;

    value $urbn
    '0'='Rural'
    '1'='Urban'
    ;

    value $dual
    '0'='Not Buy-in'
    '1'='Buy-in'
    ;
    run;

      /* START: assign macro variables to specify begin and end year
         of summary PSI data */

    %let begyr=1992;
    %let endyr=2001;
    %let file=SUM9201; /* should match the last node of the output DSN */

      /* END: assign macro variables to specify begin and end year
         of summary PSI data */

      /* START: macro program to create the list of observed outcome variables
*/

    %macro varlist;
        %do j=1 %to &tot;
            oapq&&y&i..&&x&j
        %end;;

        label
        %do j=1 %to &tot;
            oapq&&y&i..&&x&j="&&z&j"
        %end;;

        format
        %do j=1 %to &tot;
            oapq&&y&i..&&x&j 13.8
        %end;;
    %mend;

       /* END: macro program to create the list of observed outcome variables
*/


       /* START: macro program to produce the observed rates */

    %macro observe;
```

```
   /* start: create macro variables to handle multiple years' data */

%let k=1;
%do i=&begyr %to &endyr;
  %if &i<2000 %then %let x&k=%eval(&i-1900);
  %else %let x&k=0%eval(&i-2000);
```
TRA0.@MSQ0000.PQI(PQOBSERV) -- CONTINUED
```
  %let z&k=&i;
  %let k=%eval(&k+1);
%end;
%let tot=%eval(&k-1);

   /* end: create macro variables to handle multiple years' data */

   /* start: create macro variables to handle multiple PSIs */

%do i=1 %to 16;
  %if &i<10 %then %let y&i=0%eval(&i);
  %else %let y&i=%eval(&i);
%end;

   /* end: create macro variables to handle multiple PSIs */

   /* start: create macro variables for the PSI labels */

%let t1=Indicator # 01: DIABETES SHORT TERM COMPLICATION;
%let t2=Indicator # 02: PERFORATED APPENDIX;
%let t3=Indicator # 03: DIABETES LONG TERM COMPLICATION;
%let t4=Indicator # 04: PEDIATRIC ASTHMA;
%let t5=Indicator # 05: COPD;
%let t6=Indicator # 06: PEDIATRIC GASTROENTERITIS;
%let t7=Indicator # 07: HYPERTENSION;
%let t8=Indicator # 08: CONGESTIVE HEART FAILURE;
%let t9=Indicator # 09: LOW BIRTH WEIGHT;
%let t10=Indicator # 10: DEHYDRATION;
%let t11=Indicator # 11: BACTERIAL PNEUMONIA;
%let t12=Indicator # 12: URINARY INFECTION;
%let t13=Indicator # 13: ANGINA;
%let t14=Indicator # 14: DIABETES UNCONTROLLED;
%let t15=Indicator # 15: ADULT ASTHMA;
%let t16=Indicator # 16: LOWER EXTREMITY AMPUTATION;

   /* end: create macro variables for the PSI labels */

   /* start: data steps to read obs with relevent _type_ */

data &file;
  set in1.&file(where=(_type_ in (0,1,2,4,8,16,64,128)));
run;

   /* end: data steps to read obs with relevent _type_ */

   /* start: use SAS ODS to output the reports in HTML format */

ods html file=report;

%do i=1 %to 16;
```

```
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'Overall US';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=0)) label;
       var %varlist;
     run;
  %end;
```
TRA0.@MSQ0000.PQI(PQOBSERV) -- CONTINUED
```
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Age';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=128)) label;
       var agecat %varlist;
       format agecat $age.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Sex';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=64)) label;
        var sexcat %varlist;
        format sexcat $sex.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Medicaid Buy-in Status';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=1)) label;
       var dual %varlist;
       format dual $dual.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Reason for Entitlement';
     title3 "&&t&i";
```

```
     proc print noobs data=&file.(where=(_type_=2)) label;
       var mdst %varlist;
       format mdst $mdst.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Race';
     title3 "&&t&i";
```
TRA0.@MSQ0000.PQI(PQOBSERV) -- CONTINUED

```
     proc print noobs data=&file.(where=(_type_=4)) label;
        var race %varlist;
        format race $race.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Urban/Rural Status';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=8)) label;
       var urbn %varlist;
       format urbn $urbn.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By State';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=16)) label;
       var state %varlist;
       format state $st.;
     run;
  %end;
%end;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Un-adjusted Rates';
     title2 'By Region';
     title3 "&&t&i";

     proc print noobs data=&file.(where=(_type_=32)) label;
       var creg %varlist;
       format creg $creg.;
     run;
```

```
  %end;
%end;

ods html close;

   /* end: use SAS ODS to output the reports in HTML format */

%mend;

   /* END: macro program to produce the observed rates */

%observe;    /* call macro to produce the observed rates */
```

TRA0.@MSQ0000.PQI(PQADJUST)

```
//TRA0AD JOB (4AGDCH77010),'NHK',CLASS=M,MSGCLASS=Q
// EXEC SAS8,OPTIONS='NONEWS',WORK='8000,4000'
//IN1    DD DSN=TRA0.@MSQ0000.SAS.PQCOMBN.SUM9201,DISP=SHR
//REPORT DD DSN=TRA0.@MSQ0000.HTML.PQADJUST.PDS(SUM9201),
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(100,100,3),RLSE)
//SYSIN  DD *

options nocenter compress=yes;

proc format;
value $st
'01'='AL' '02'='AK' '03'='AZ' '04'='AR' '05'='CA'
'06'='CO' '07'='CT' '08'='DE' '09'='DC' '10'='FL'
'11'='GA' '12'='HI' '13'='ID' '14'='IL' '15'='IN'
'16'='IA' '17'='KS' '18'='KY' '19'='LA' '20'='ME'
'21'='MD' '22'='MA' '23'='MI' '24'='MN' '25'='MS'
'26'='MO' '27'='MT' '28'='NE' '29'='NV' '30'='NH'
'31'='NJ' '32'='NM' '33'='NY' '34'='NC' '35'='ND'
'36'='OH' '37'='OK' '38'='OR' '39'='PA' '40'='RQ'
'41'='RI' '42'='SC' '43'='SD' '44'='TN' '45'='TX'
'46'='UT' '47'='VT' '48'='VI' '49'='VA' '50'='WA'
'51'='WV' '52'='WI' '53'='WY'
;

value $age
'1'='0-54'
'2'='55-64'
'3'='65-69'
'4'='70-74'
'5'='75-79'
'6'='80-84'
'7'='85-89'
'8'='90-94'
'9'='95+'
;

value $sex
'1'='Male'
'2'='Female'
;

value $race
'1'='White'
'2'='African American'
'3'='Other'
'0'='Unknown'
;

value $mdst
'1'='Aged without ESRD'
'2'='Aged with ESRD'
'3'='Disabled without ESRD'
'4'='Disabled with ESRD'
'5'='ESRD only'
;
```

```
    value $creg
    '0'='Puero Rico and Vergin Island'
    '1'='Northeast'
    TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

    '2'='Midwest'
    '3'='South'
    '4'='West'
    ;

    value $urbn
    '0'='Rural'
    '1'='Urban'
    ;

    value $dual
    '0'='Not Buy-in'
    '1'='Buy-in'
    ;
    run;

      /*  START: data step to create files for age-sex adjustments */
      /*  1999 age-sex populations are hard coded */
      /*  Data Source: D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99 */

    data totp(keep=pop) /* total MQMS population */
         sexp(keep=sexcat pop) /* standard MQMS population by sex */
         agep(keep=agecat pop) /* standard MQMS population by age-groups */
         agesexp(keep=agecat sexcat pop) /* standard MQMS populations by age-
sex */
          adjasp(keep=agecat sexcat pop); /* revised poplns for state by age-
sex */
      input
      @1  agecat   $1.
      @11 sexcat   $1.
      @19 pop       9.
      @;

      if _n_=1 then output totp;
      else if _n_<=3 then output sexp;
      else if _n_<=12 then output agep;
      else if _n_<=30 then output agesexp;
      else output adjasp;
      cards;
                       31471880
              1        13607588
              2        17864292
    1                   2986590
    2                   1862336
    3                   7008112
    4                   6717803
    5                   5646458
    6                   3785275
    7                   2172280
    8                    898468
    9                    394558
    1         1         1739165
    1         2         1247425
```

```
2         1        1013695
2         2         848641
3         1        3262650
3         2        3745462
4         1        2957419
4         2        3760384
```

```
5         1        2299257
5         2        3347201
6         1        1375789
6         2        2409486
7         1         657825
7         2        1514455
8         1         215105
8         2         683363
9         1          86683
9         2         307875
1         1        2752860
1         2        2096066
2         1        6220069
2         2        7505846
3         1        3675046
3         2        5756687
4         1         959613
4         2        2505693
;
run;


  /* END: data step to create files for age-sex adjustments */


  /* START: assign macro variables to specify begin and end year
     of summary PSI data */

%let begyr=1992;
%let endyr=2001;
%let file=SUM9201; /* should match the last node of the input DSN */

  /* END: assign macro variables to specify begin and end year
     of summary PSI data */

  /* START: macro program to create the list of adjusted outcome variables
*/

%macro varlist1;
  %do i=1 %to &tot;
    %do j=1 %to 16;
        aapq&&y&j..&&x&i
    %end;
  %end;;
%mend;

%macro varlist2;
    %do j=1 %to &tot;
        aapq&&y&i..&&x&j
    %end;;
```

```
        label
        %do j=1 %to &tot;
            aapq&&y&i..&&x&j="&&z&j"
        %end;;

        format
        %do j=1 %to &tot;
            aapq&&y&i..&&x&j 13.8
```

```

        %end;;
    %mend;

       /* END: macro program to create the list of adjusted outcome variables
*/

       /* START: macro program to produce the adjusted rates */

    %macro adjust;

       /* start: create macro variables to handle multiple years' data */

    %let k=1;
    %do i=&begyr %to &endyr;
      %if &i<2000 %then %let x&k=%eval(&i-1900);
      %else %let x&k=0%eval(&i-2000);
      %let z&k=&i;
      %let k=%eval(&k+1);
    %end;
    %let tot=%eval(&k-1);

       /* end: create macro variables to handle multiple years' data */

       /* start: create macro variables to handle multiple PSIs */

    %do i=1 %to 16;
      %if &i<10 %then %let y&i=0%eval(&i);
      %else %let y&i=%eval(&i);
    %end;

       /* end: create macro variables to handle multiple PSIs */

       /* start: create macro variables for the PSI labels */

    %let t1=Indicator # 01: DIABETES SHORT TERM COMPLICATION;
    %let t2=Indicator # 02: PERFORATED APPENDIX;
    %let t3=Indicator # 03: DIABETES LONG TERM COMPLICATION;
    %let t4=Indicator # 04: PEDIATRIC ASTHMA;
    %let t5=Indicator # 05: COPD;
    %let t6=Indicator # 06: PEDIATRIC GASTROENTERITIS;
    %let t7=Indicator # 07: HYPERTENSION;
    %let t8=Indicator # 08: CONGESTIVE HEART FAILURE;
    %let t9=Indicator # 09: LOW BIRTH WEIGHT;
    %let t10=Indicator # 10: DEHYDRATION;
    %let t11=Indicator # 11: BACTERIAL PNEUMONIA;
    %let t12=Indicator # 12: URINARY INFECTION;
    %let t13=Indicator # 13: ANGINA;
```

```
%let t14=Indicator # 14: DIABETES UNCONTROLLED;
%let t15=Indicator # 15: ADULT ASTHMA;
%let t16=Indicator # 16: LOWER EXTREMITY AMPUTATION;

   /* end: create macro variables for the PSIs label       */

   /* start: create summary files of observed rates with age-sex
             interactions, and age-sex interactions with other
             individual characteristics */

   /* two separate datasets are created -- 1) without state-age-sex
      interactions and 2) with state-age-sex interactions. */
```
TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

```
data obsrate1(where=(_type_ ne 208)) /* without state-age-sex ints */
     obsrate2(where=(_type_=208));  /* with state-age-sex interactions */
  set in1.&file(keep=agecat sexcat creg state urbn race mdst dual _type_
  %do i=1 %to &tot;
    %do j=1 %to 16;
        tapq&&y&j..&&x&i
        papq&&y&j..&&x&i
        oapq&&y&j..&&x&i
    %end;
  %end;);

   /* end: create summary files of observed rates with age-sex
           interactions, and age-sex interactions with other
           individual characteristics */

  if _type_ in (192,193,194,196,200,208,224);

   /* start: recode sex to match the sex in agesexp and adjasp files */

  if sexcat='M' then sexcat='1';
  else if sexcat='F' then sexcat='2';

   /* end: recode sex to match the sex in agesexp and adjasp files */

run;

   /* end: create files of observed rates with age-sex interactions */

   /* start: construct ratios of sex-specific populn to overall populn */

data sexp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set sexp;
  prop=pop/totpop;
run;

   /* end: construct ratios of sex-specific populn to overall populn */

   /* start: construct ratios of age-specific populn to overall populn */

data agep;
  if _n_=1 then set totp(rename=(pop=totpop));
  set agep;
```

```
  prop=pop/totpop;
run;

   /* end: construct ratios of age-specific populn to overall populn */

data agesexp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set agesexp;
  prop=pop/totpop;
run;

data adjasp;
  if _n_=1 then set totp(rename=(pop=totpop));
  set adjasp;
  prop=pop/totpop;
```

```
run;

   /* end: construct ratios of age-sex-specific populn to overall populn
*/

   /* start: fold some age categories in the state-specific summary
dataset,
              drop the observed rates */

data obsrate2;
  set obsrate2(rename=(agecat=age));
  if age in ('1','2') then agecat='1';
  else if age in ('3','4') then agecat='2';
  else if age in ('5','6') then agecat='3';
  else if age in ('7','8','9') then agecat='4';
  drop
  %do i=1 %to &tot;
    %do j=1 %to 16;
        oapq&&y&j..&&x&i
    %end;
  %end;;
run;

   /* end: fold some age categories in the state-specific summary dataset,
           drop the observed rates */

    /* start: sum the numerators and denominators in the state-specific
              summary file based on the revised age categories. Create
              new summary file */

proc sort data=obsrate2;
  by state agecat sexcat;
run;

proc means noprint data=obsrate2;
  var
  %do i=1 %to &tot;
    %do j=1 %to 16;
        tapq&&y&j..&&x&i papq&&y&j..&&x&i
    %end;
  %end;;
```

```
      by state agecat sexcat;
      output out=obsrate2(drop=_:)
             sum=
             ;
   run;

       /* end: sum the numerators and denominators in the state-specific
               summary file based on the revised age categories. Create
               new summary file */

       /* start: calculate the new observed and adjusted rates based on
revised
                 age-sex categories in the new state-specific summary file.
*/

   proc sort data=obsrate2;
     by agecat sexcat;
   run;
```

TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

```
   data adjrate2;
     merge obsrate2(in=a)
           adjasp(in=b)
           ;
     by agecat sexcat;
     if a;
     %do i=1 %to &tot;
       %do j=1 %to 16;
           oapq&&y&j..&&x&i=tapq&&y&j..&&x&i / papq&&y&j..&&x&i; /* obs rate
*/
           aapq&&y&j..&&x&i=oapq&&y&j..&&x&i * prop;             /* adj rate
*/
       %end;
     %end;
   run;

       /* end: calculate the new observed rates based on revised age-sex
               categories in the new state-specific summary file. */

       /* start: calculate adjusted rates based on standard MQMS age-sex
          categories in the non-state-specific file. */

   proc sort data=obsrate1;
     by agecat sexcat;
   run;

   data adjrate1;
     merge obsrate1(in=a)
           agesexp(in=b)
           ;
     by agecat sexcat;
     if a;
     %do i=1 %to &tot;
       %do j=1 %to 16;
           aapq&&y&j..&&x&i=oapq&&y&j..&&x&i * prop;
       %end;
     %end;
```

```
    run;

       /* end: calculate adjusted rates based on standard MQMS age-sex
           categories in the non-state-specific file. */

       /* start: use SAS ODS to output the reports in HTML format */

    ods html file=report;

       /* start: summarize the adjusted rate with obs containing age-sex-only
                 interactions to create file with age-sex-adjusted rates
                 for overall US. print the adjusted rates for each PSI */

    proc means noprint data=adjrate1(where=(_type_=192));
      var %varlist1;
      output out=all(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
    TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'Overall US';
         title3 "&&t&i";

         proc print noobs data=all label;
            var %varlist2;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing age-sex-only
                 interactions to create file with age-sex-adjusted rates
                 for overall US. print the adjusted rates for each PSI */

       /* start: summarize the adjusted rate with obs containing age-sex-only
                 interactions  by  age  categories  to  create  file  with  sex-
adjusted
                  rates for age categories. print the adjusted rates for each
PSI */

    proc sort data=obsrate1(where=(_type_ =192)) out=age;
      by sexcat;
    run;

    data age;
      merge age(in=a)
            sexp(in=b)
            ;
      by sexcat;
      if a;
      %do i=1 %to &tot;
        %do j=1 %to 16;
            aapq&&y&j..&&x&i=oapq&&y&j..&&x&i * prop;
        %end;
```

```
  %end;
run;

proc sort data=age;
  by agecat;
run;

proc means noprint data=age;
  var %varlist1;
  by agecat;
  output out=age(drop=_:)
         sum=
         ;
run;

%do i=1 %to 16;
  %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
     title 'Sex-adjusted Rates';
     title2 'By Age';
     title3 "&&t&i";

     proc print noobs data=age label;
       var agecat %varlist2;
       format agecat $age.;
```



```
     run;
  %end;
%end;

   /* end: summarize the adjusted rate with obs containing age-sex-only
           interactions by age categories to create file with sex-adjusted
           rates for age categories. print the adjusted rates for each PSI
*/

   /* start: summarize the adjusted rate with obs containing age-sex-only
             interactions by sex categories to create file with age-
adjusted
              rates for sex categories. print the adjusted rates for each
PSI */

proc sort data=obsrate1(where=(_type_ =192)) out=sex;
  by agecat;
run;

data sex;
  merge sex(in=a)
        agep(in=b)
        ;
  by agecat;
  if a;
  %do i=1 %to &tot;
    %do j=1 %to 16;
        aapq&&y&j..&&x&i=oapq&&y&j..&&x&i * prop;
    %end;
  %end;
run;
```

```
    proc sort data=sex;
      by sexcat;
    run;

    proc means noprint data=sex;
      var %varlist1;
      by sexcat;
      output out=sex(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-adjusted Rates';
         title2 'By Sex';
         title3 "&&t&i";

         proc print noobs data=sex label;
           var sexcat %varlist2;
           format sexcat $sex.;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing age-sex-only
               interactions by sex categories to create file with age-adjusted
```

```
               rates for sex categories. print the adjusted rates for each PSI
*/

       /* start: summarize the adjusted rate with obs containing dual-age-sex
                  interactions by dual categories to create file with age-sex-
adjustd
                  rates for dual categories. print the adjusted rates for each
PSI */

    proc sort data=adjrate1(where=(_type_=193)) out=dual;
      by dual agecat sexcat;
    run;

    proc means noprint data=dual;
      var %varlist1;
      by dual;
      output out=dual(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By Medicaid Buy-in Status';
         title3 "&&t&i";

         proc print noobs data=dual label;
           var dual %varlist2;
```

```
          format dual $dual.;
        run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing dual-age-sex
               interactions by dual categories to create file with age-sex-
adjusted
                rates for dual categories. print the adjusted rates for each
PSI */


       /* start: summarize the adjusted rate with obs containing mdst-age-sex
                interactions by mdst categories to create file with age-sex-
adjustd
                 rates for mdst categories. print the adjusted rates for each
PSI */


    proc sort data=adjrate1(where=(_type_=194)) out=mdst;
      by mdst agecat sexcat;
    run;

    proc means noprint data=mdst;
      var %varlist1;
      by mdst;
      output out=mdst(drop=_:)
             sum=
             ;
    run;
    TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED


    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By Reason for Entitlement';
         title3 "&&t&i";

         proc print noobs data=mdst label;
           var mdst %varlist2;
           format mdst $mdst.;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing mdst-age-sex
               interactions by mdst categories to create file with age-sex-
adjusted
                rates for mdst categories. print the adjusted rates for each
PSI */


       /* start: summarize the adjusted rate with obs containing race-age-sex
                interactions by race categories to create file with age-sex-
adjustd
                 rates for race categories. print the adjusted rates for each
PSI */
```

```
    proc sort data=adjrate1(where=(_type_=196)) out=race;
      by race agecat sexcat;
    run;

    proc means noprint data=race;
      var %varlist1;
      by race;
      output out=race(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By Race';
         title3 "&&t&i";

         proc print noobs data=race label;
           var race %varlist2;
           format race $race.;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing race-age-sex
                interactions by race categories to create file with age-sex-
adjusted
               rates for race categories. print the adjusted rates for each
PSI */

       /* start: summarize the adjusted rate with obs containing urbn-age-sex
    TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

                interactions by urbn categories to create file with
                age-sex-adjusted rates for urbn categories.  print the
                adjusted rates for each PSI */

    proc sort data=adjrate1(where=(_type_=200)) out=urbn;
      by urbn agecat sexcat;
    run;

    proc means noprint data=urbn;
      var %varlist1;
      by urbn;
      output out=urbn(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By Urban/Rural Status';
         title3 "&&t&i";

         proc print noobs data=urbn label;
```

```
          var urbn %varlist2;
          format urbn $urbn.;
        run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing urbn-age-sex
               interactions by urbn categories to create file with age-sex-
adjusted
               rates for urbn categories. print the adjusted rates for each
PSI */

       /* start: summarize the adjusted rate with obs containing state-age-sex
                 interactions by states to create file with age-sex-adjusted
                  rates for the states. print the adjusted rates for each PSI
*/

    proc sort data=adjrate2 out=state;
      by state agecat sexcat;
    run;

    proc means noprint data=state;
      var %varlist1;
      by state;
      output out=state(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By State';
         title3 "&&t&i";

         proc print noobs data=state label;
    TRA0.@MSQ0000.PQI(PQADJUST) -- CONTINUED

           var state %varlist2;
           format state $st.;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing state-age-sex
               interactions by states to create file with age-sex-adjusted
               rates for the states. print the adjusted rates for each PSI */

       /* start: summarize the adjusted rate with obs containing creg-age-sex
                 interactions by creg categories to create file with age-sex-
adjustd
                 rates for creg categories. print the adjusted rates for each
PSI */

    proc sort data=adjrate1(where=(_type_=224)) out=creg;
      by creg agecat sexcat;
    run;
```

```
    proc means noprint data=creg;
      var %varlist1;
      by creg;
      output out=creg(drop=_:)
             sum=
             ;
    run;

    %do i=1 %to 16;
      %if &i ne 4 and &i ne 6 and &i ne 9 %then %do;
         title 'Age-sex-adjusted Rates';
         title2 'By Region';
         title3 "&&t&i";

         proc print noobs data=creg label;
           var creg %varlist2;
           format creg $creg.;
         run;
      %end;
    %end;

       /* end: summarize the adjusted rate with obs containing creg-age-sex
               interactions by creg categories to create file with age-sex-
adjusted
               rates for creg categories. print the adjusted rates for each
PSI */

    ods html close;

       /* end: use SAS ODS to output the reports in HTML format */

    %mend;
       /* END: macro program to produce the adjusted rates */
    %adjust /* call macro to produce adjusted rates */
```

PROGRAMS –

PROGRESSION OF DISEASE MEASURES

D#MSQ.MSQ00M7.PGMLIB99(A001HH)

```
//WF17A01H JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS WHEN RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       IN1 = FILE NAME OF FLAT HHA FILE FROM DESY EXTRACTION
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/23/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL
//* INPUTS    :IN1
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :READ HH FILE INTO SAS FILE, GENERATE CLAIMS LEVEL FILE
//*            AND FILE WITH UNDUPLICATED HICNO FOR CROSS-REFERENCE
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A001HH.HH99
  DELETE D#MSQ.@MSQ00M7.SAS.A001HH.HHID99
//*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(900,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0008819.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A001HH.HH99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(900,300),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.A001HH.HHID99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(200,50),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;

*********************************************************************;
*  FULLNAME:  FI HHA CLAIMS RECORD                                  ;
*  VERSION 'I' SAS COPYLIB STATEMENTS                               ;
*  MEMBER NAME: UTLIHHA                                             ;
*  CREATED BY MARVELLA BUNCH 11/01/2000, modified by Hong Chen      ;
*  UPDATED BY MARVELLA BUNCH 03/01/2001                             ;

*********************************************************************;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. Reads in the flat HH file, keep only necessary
 *         variables for identifying diabetes later, and write out
```

```
 *         a SAS file.
 ********************************************************************;
DATA out1.HH&YEAR (keep=hicno DGNSCD01-DGNSCD10 RVCNTR01-RVCNTR45
D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

               HCPSCD01-HCPSCD45
               FROM_DT THRU_DT );
  INFILE IN1 MISSOVER            ;
    LENGTH REC_LEN                              4;
    LENGTH ACRTN_NM                             3;
    LENGTH LINK_NUM                             5;
    LENGTH SGMT_CNT                             2;
    LENGTH SGMT_NUM                             2;
    LENGTH LINECNT                              3;
    LENGTH SGMTLINE                             2;
    LENGTH HHEDCNT                              2;
    LENGTH HHPATCNT                             2;
    LENGTH HHMCOCNT                             2;
    LENGTH HHPLANNT                             2;
    LENGTH HHDEMCNT                             2;
    LENGTH HHDGNCNT                             2;
    LENGTH HHCONCNT                             2;
    LENGTH HHOCRCNT                             2;
    LENGTH HHSPNCNT                             2;
    LENGTH HHVALCNT                             2;
    LENGTH HHREVCNT                             2;
    LENGTH VISITCNT                             3;
    ARRAY EDITIND{13}                      $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                      $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                     $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                      $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                      $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                        $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                        $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                       $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                      $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                      $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                      $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                      $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                      $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                        $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                       $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                       $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                       $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                      $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                      $    5 DGNSCD01-DGNSCD10;
    ARRAY CONDIND{30}                      $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                     $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                     $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                     $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                      $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                      $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                      $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                     $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                     $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                       $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                       $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                           8 VALAMT01-VALAMT36;
```

```
    ARRAY REVIND{45}                          $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                          $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                        $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                        $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                        $    5 RVNS301-RVNS345;
D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

    ARRAY REVANSI4{45}                        $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                        $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                        $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                        $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                        $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                        $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                        $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                         $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                        $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                        $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                          $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                          $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                          $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                          $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                             4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                             8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                             8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                             8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                              8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                             8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                             8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                             8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                             8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                             8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                             8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                             8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                               8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                             8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                             8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                        $    1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_HHA_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                    PD3.
                @4  REC_LVL                    $CHAR1.
                @5  RIC_CD                     $CHAR1.
                @6  MQA_RIC                    $CHAR1.
                @7  CLM_TYPE                   $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                    @9  CAN                        $CHAR9.

                    @9  HICNO                      $CHAR11.

                    @18  EQ_BIC                    $CHAR2.
                @20  BIC                       $CHAR2.
                @22  ST_SGMT                   $CHAR1.
                @23  STATE_CD                  $CHAR2.
                @25  FROM_DT                   $CHAR8.
```

```
                @33  THRU_DT                    $CHAR8.
                @41  WKLY_DT                    $CHAR8.
                @49  ACRTN_DT                   $CHAR8.
                @57  ACRTN_NM                   PD2.
                @59  CLM_CNTL                   $CHAR23.
                @82  ORIGCNTL                   $CHAR23.
                @105  QUERY_CD                   $CHAR1.
                @106  PROVIDER                   $CHAR6.
                @112  DAILY_DT                   $CHAR8.
D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

                @120  LINK_NUM                   PD5.
                @125  SGMT_CNT                   2.
                @127  SGMT_NUM                   2.
                @129  LINECNT                    3.
                @132  SGMTLINE                   2.
              /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                     $CHAR1.
                @135  TRANS_CD                   $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                   $CHAR1.
                    @137  TYPESRVC                   $CHAR1.
                @138  FREQ_CD                    $CHAR1.
                @139  _F000001                   $CHAR1.
                @140  MQAQUERY                   $CHAR1.
                @141  DISP_CD                    $CHAR2.
                @143  EDITDISP                   $CHAR2.
                @145  BIC_MDFY                   $CHAR1.
                @146  CNTY_CD                    $CHAR3.
                @149  RCPT_DT                    $CHAR8.
                @157  SCHLD_DT                   $CHAR8.
                @165  FRWRD_DT                   $CHAR8.
                @173  FI_NUM                     $CHAR5.
                @178  ASGN_NUM                   $CHAR8.
                @186  FIBATCH                    $CHAR4.
                @190  BENE_ZIP                   $CHAR9.
                @199  SEX                        $CHAR1.
                @200  RACE                       $CHAR1.
                @201  BENE_DOB                   $CHAR8.
                @209  MS_CD                      $CHAR2.
                @211  SURNAME                    $CHAR6.
                @217  FRSTINIT                   $CHAR1.
                @218  MDL_INIT                   $CHAR1.
                @219  CWFLOCCD                   $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
                @242  ACTIONCD                   $CHAR1.
                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
```

```
                    @269  AT_NPI                      $CHAR10.
                    @279  AT_SRNM                     $CHAR6.
                    @285  AT_GVNNM                    $CHAR1.
                    @286  AT_MDL                      $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                     $CHAR6.
                    @293  OP_NPI                      $CHAR10.
                    @303  OP_SRNM                     $CHAR6.
                    @309  OP_GVN                      $CHAR1.
                    @310  OP_MDL                      $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @311  OT_UPIN                     $CHAR6.
```

D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

```
                    @317  OT_NPI                      $CHAR10.
                    @327  OT_SRNM                     $CHAR6.
                    @333  OT_GVN                      $CHAR1.
                    @334  OT_MDL                      $CHAR1.
                @335  MDCD_PRV                    $CHAR13.
                @348  MDCDINFO                    $CHAR4.
                @352  MCOPDSW                     $CHAR1.
                @353  AUTHRZTN                    $CHAR18.
                @371  PTNTCNTL                    $CHAR20.
                @391  MDCL_REC                    $CHAR17.
                @408  PRO_CNTL                    $CHAR12.
                @420  PRO_DT                      $CHAR8.
                @428  STUS_CD                     $CHAR2.
                @430  DGNS_E                      $CHAR5.
                @435  _F000003                    $CHAR1.
                @436  PPS_IND                     $CHAR1.
                @437  TOT_CHRG                    PD6.2
                @443  _F000004                    $CHAR50.
            @493  HHEDCNT                     2.
            @495  HHPATCNT                    2.
            @497  HHMCOCNT                    1.
            @498  HHPLANNT                    1.
            @499  HHDEMCNT                    1.
            @500  HHDGNCNT                    2.
            @502  _F000005                    $CHAR2.
            @504  HHCONCNT                    2.
            @506  HHOCRCNT                    2.
            @508  HHSPNCNT                    2.
            @510  HHVALCNT                    2.
            @512  HHREVCNT                    2.
            @514  _F000006                    $CHAR4.
            /*  FI_HHA_CLM_SPECF_I_GRP  */
                @518  LUPAIND                     $CHAR1.
                @519  HHA_RFRL                    $CHAR1.
                @520  VISITCNT                    PD2.
                @522  QLFYFROM                    $CHAR8.
                @530  QLFYTHRU                    $CHAR8.
                @538  DSCHRGDT                    $CHAR8.
                @546  HHSTRTDT                    $CHAR8.
                @554  _F000007                    $CHAR16.
        /*  FI_HHA_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO HHEDCNT;
        INPUT
```

```
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}               $CHAR1.
                EDIT_CD{_I1_}               $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO HHPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}              $CHAR1.
                PATCHCD{_I1_}               $CHAR2.
                PATCHDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHMCOCNT;
        INPUT
```

D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

```
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}                $CHAR1.
                MCONUM{_I1_}                $CHAR5.
                MCOOPTN{_I1_}               $CHAR1.
                MCOEFFDT{_I1_}              $CHAR8.
                MCOTRMDT{_I1_}              $CHAR8.
                MCOPLNID{_I1_}              $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO HHPLANNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}              $CHAR1.
                PLANIDCD{_I1_}              $CHAR1.
                PLANID{_I1_}                $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO HHDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}               $CHAR1.
                DEMONUM{_I1_}               $CHAR2.
                DEMOTXT{_I1_}               $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO HHDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}               $CHAR1.
                DGNS_CD{_I1_}               $CHAR5.
                _F000008                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO HHCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO HHOCRCNT;
```

```
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHVALCNT;
        INPUT
```

D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

```
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO HHREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
```

```
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000009                    $CHAR50.
        @;
    END;
    INPUT       EOR                         $CHAR3.@;


   /* Release trailing @ */
   input;

    DROP _I1_ ;
    DROP _F000001-_F000009;

  /******Creates labels for the variables on the SAS file.******/
```
D#MSQ.MSQ00M7.PGMLIB99(A001HH) -- CONTINUED

```
LABEL

HICNO        =     "BENEFICIARY CLAIM#"
FROM_DT      =     "HHA FROM DATE"
THRU_DT      =     "HHA THROUGH DATE"

DGNSCD01        =     "DIAGNOSIS CODE_1"
DGNSCD02        =     "DIAGNOSIS CODE_2"
DGNSCD03        =     "DIAGNOSIS CODE_3"
DGNSCD04        =     "DIAGNOSIS CODE_4"
DGNSCD05        =     "DIAGNOSIS CODE_5"
DGNSCD06        =     "DIAGNOSIS CODE_6"
DGNSCD07        =     "DIAGNOSIS CODE_7"
DGNSCD08        =     "DIAGNOSIS CODE_8"
DGNSCD09        =     "DIAGNOSIS CODE_9"
DGNSCD10        =     "DIAGNOSIS CODE_10"

RVCNTR01       =     "REVENUE CENTER CODE_1"
HCPSCD01       =     "HCPCS CODE_1"
;

RUN;


 ******************************************************************;
 * STEP 2. Proc contents and proc print on the SAS claims files.  *
 ******************************************************************;
PROC CONTENTS DATA=out1.HH&YEAR DIRECTORY ;
TITLE "HHA Y&YEAR: CLAIMS FILE WITH SELECTED VARIABLES ";
RUN;

PROC PRINT DATA=out1.HH&YEAR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR HHA FILE";
RUN;
```

```
 ********************************************************************;
 * STEP 3. CREATE FILE WITH HICNO ONLY FOR FINDER FILE LATER
 *         and proc print on the SAS ID files.
 ********************************************************************;
PROC SORT NODUPKEY DATA=out1.HH&YEAR (KEEP=HICNO) out=OUT2.HHID&YEAR;
     BY HICNO;
RUN;

PROC PRINT DATA=OUT2.HHID&YEAR (OBS=10);
TITLE "HHA HICNO Y&YEAR: PRINT OUT 10 OBS ";
RUN;



ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A001IP)

```
//WF17A01I JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       IN1 = FILE NAME OF FLAT INPATIENT FILE FROM DESY EXTRACTION
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/21/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :READ IP FILE INTO SAS FILE, GENERATE CLAIMS LEVEL FILE
//*            AND FILE WITH UNDUPLICATED HICNO FOR CROSS-REFERENCE
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A001IP.IP99
  DELETE D#MSQ.@MSQ00M7.SAS.A001IP.IPID99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(400,200))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0008817.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A001IP.IP99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.A001IP.IPID99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(200,50),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *


options nocenter pageno=1 ps=55 ls=80 symbolgen ;

*********************************************************************;
*  FULL_NAME:  FI INPATIENT SNF CLAIMS RECORD                       ;
*  VERSION 'I' SAS COPYLIB STATEMENT                                ;
*  MEMBER NAME: UTLIIPSN                                            ;
*  CREATED BY  MARVELLA BUNCH 11/01/2000                            ;
*  UPDATED BY  MARVELLA BUNCH 03/01/2001                            ;
*********************************************************************;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. Reads in the flat Inpatient file, keep only necessary
```

```
 *         variables for identifying diabetes later, and write out
 *         a SAS file.
 ********************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

```
DATA out1.IP&YEAR (keep=hicno DGNSCD01-DGNSCD10
                PRCDRCD1-PRCDRCD6
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT );
  INFILE IN1 MISSOVER;
    LENGTH REC_LEN                                4;
    LENGTH ACRTN_NM                               3;
    LENGTH LINK_NUM                               5;
    LENGTH SGMT_CNT                               2;
    LENGTH SGMT_NUM                               2;
    LENGTH LINECNT                                3;
    LENGTH SGMTLINE                               2;
    LENGTH BENE_DOB                               5;
    LENGTH IPEDCNT                                2;
    LENGTH IPPATCNT                               2;
    LENGTH IPMCOCNT                               2;
    LENGTH IPPLNCNT                               2;
    LENGTH IPDEMCNT                               2;
    LENGTH IPDGNCNT                               2;
    LENGTH IPPRCNT                                2;
    LENGTH IPCONCNT                               2;
    LENGTH IPOCRCNT                               2;
    LENGTH IPSPNCNT                               2;
    LENGTH IPVALCNT                               2;
    LENGTH IPREVCNT                               2;
    LENGTH GRC_DAY                                2;
    LENGTH UTIL_DAY                               3;
    LENGTH CR_DAY                                 3;
    LENGTH COIN_DAY                               3;
    LENGTH COYR1DAY                               3;
    LENGTH COYR2DAY                               3;
    LENGTH LRD_USE                                3;
    LENGTH NUTILDAY                               4;
    LENGTH PSYCHDAY                               3;
    LENGTH BLDFRNSH                               3;
    LENGTH BLD_RPLC                               3;
    LENGTH BLDNRPLC                               3;
    LENGTH BLDDEDPT                               3;
    ARRAY EDITIND{13}                        $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                        $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                       $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                        $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                        $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                          $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                          $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                         $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                        $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                        $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                        $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                        $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                        $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                          $   14 PLANID1-PLANID3;
```

```
    ARRAY DEMOIND{5}                          $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                          $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                          $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                         $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                         $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDRIND{6}                         $    1 PRCDRND1-PRCDRND6;
D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

    ARRAY PRCDR_CD{6}                         $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                         $    8 PRCDRDT1-PRCDRDT6;
    ARRAY CONDIND{30}                         $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                        $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                        $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                        $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                         $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                         $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                         $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                        $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                        $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                          $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                          $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                              8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                          $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                          $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                        $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                        $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                        $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                        $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                        $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                        $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                        $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                        $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                        $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                        $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                         $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                        $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                        $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                          $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                          $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                          $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                          $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                             4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                             8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                             8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                             8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                              8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                             8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                             8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                             8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                             8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                             8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                             8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                             8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                               8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                             8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                             8 RVNCVR01-RVNCVR45;
```

```
    ARRAY REVDEDCD{45}                          $    1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_IP_SNF_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                       PD3.
                @4  REC_LVL                       $CHAR1.
                @5  RIC_CD                        $CHAR1.
                @6  MQA_RIC                       $CHAR1.
                @7  CLM_TYPE                      $CHAR2.
```

D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

```
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                    @9  CAN                          $CHAR9.
                    @9  HICNO                        $CHAR11.

                    @18  EQ_BIC                      $CHAR2.
                @20  BIC                            $CHAR2.
                @22  ST_SGMT                        $CHAR1.
                @23  STATE_CD                       $CHAR2.
                @25  FROM_DT                        $CHAR8.
                @33  THRU_DT                        $CHAR8.
                @41  WKLY_DT                        $CHAR8.
                @49  ACRTN_DT                       $CHAR8.
                @57  ACRTN_NM                       PD2.
                @59  CLM_CNTL                       $CHAR23.
                @82  ORIGCNTL                       $CHAR23.
                @105  QUERY_CD                       $CHAR1.
                @106  PROVIDER                       $CHAR6.
                @112  DAILY_DT                       $CHAR8.
                @120  LINK_NUM                       PD5.
                @125  SGMT_CNT                       2.
                @127  SGMT_NUM                       2.
                @129  LINECNT                        3.
                @132  SGMTLINE                       2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                         $CHAR1.
                @135  TRANS_CD                       $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                       $CHAR1.
                    @137  TYPESRVC                       $CHAR1.
                @138  FREQ_CD                        $CHAR1.
                @139  _F000001                       $CHAR1.
                @140  MQAQUERY                       $CHAR1.
                @141  DISP_CD                        $CHAR2.
                @143  EDITDISP                       $CHAR2.
                @145  BIC_MDFY                       $CHAR1.
                @146  CNTY_CD                        $CHAR3.
                @149  RCPT_DT                        $CHAR8.
                @157  SCHLD_DT                       $CHAR8.
                @165  FRWRD_DT                       $CHAR8.
                @173  FI_NUM                         $CHAR5.
                @178  ASGN_NUM                       $CHAR8.
                @186  FIBATCH                        $CHAR4.
                @190  BENE_ZIP                       $CHAR9.
                @199  SEX                            $CHAR1.
                @200  RACE                           $CHAR1.
                @201  BENE_DOB                       8.
```

```
                @209  MS_CD                        $CHAR2.
                @211  SURNAME                      $CHAR6.
                @217  FRSTINIT                     $CHAR1.
                @218  MDL_INIT                     $CHAR1.
                @219  CWFLOCCD                     $CHAR1.
                @220  PDGNS_CD                     $CHAR5.
                @225  _F000002                     $CHAR1.
                @226  NOPAY_CD                     $CHAR1.
                @227  TRTMT_CD                     $CHAR1.
                @228  PMT_AMT                      PD6.2
                @234  PRPAYAMT                     PD6.2
                @240  PRPAY_CD                     $CHAR1.
D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

                @241  CANCELCD                     $CHAR1.
                @242  ACTIONCD                     $CHAR1.
                @243  APRVL_DT                     $CHAR8.
                @251  PRSTATE                      $CHAR2.
                @253  ORGNPINM                     $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                      $CHAR6.
                    @269  AT_NPI                       $CHAR10.
                    @279  AT_SRNM                      $CHAR6.
                    @285  AT_GVNNM                     $CHAR1.
                    @286  AT_MDL                       $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                      $CHAR6.
                    @293  OP_NPI                       $CHAR10.
                    @303  OP_SRNM                      $CHAR6.
                    @309  OP_GVN                       $CHAR1.
                    @310  OP_MDL                       $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @351  OT_UPIN                      $CHAR6.
                    @357  OT_NPI                       $CHAR10.
                    @327  OT_SRNM                      $CHAR6.
                    @333  OT_GVN                       $CHAR1.
                    @334  OT_MDL                       $CHAR1.
                @335  MDCD_PRV                     $CHAR13.
                @348  MDCDINFO                     $CHAR4.
                @352  MCOPDSW                      $CHAR1.
                @353  AUTHRZTN                     $CHAR18.
                @371  PTNTCNTL                     $CHAR20.
                @391  MDCL_REC                     $CHAR17.
                @408  PRO_CNTL                     $CHAR12.
                @420  PRO_DT                       $CHAR8.
                @428  STUS_CD                      $CHAR2.
                @430  DGNS_E                       $CHAR5.
                @435  _F000003                     $CHAR1.
                @436  PPS_IND                      $CHAR1.
                @437  TOT_CHRG                     PD6.2
                @443  _F000004                     $CHAR50.
            @493  IPEDCNT                   2.
            @495  IPPATCNT                  2.
            @497  IPMCOCNT                  1.
            @498  IPPLNCNT                  1.
            @499  IPDEMCNT                  1.
            @500  IPDGNCNT                  2.
            @502  IPPRCNT                   2.
```

```
            @504  IPCONCNT                    2.
            @506  IPOCRCNT                    2.
            @508  IPSPNCNT                    2.
            @510  IPVALCNT                    2.
            @512  IPREVCNT                    2.
            @514  _F000005                    $CHAR4.
            /*  FI_IP_SNF_CLM_SPECF_I_GRP  */
                @518  ADMSN_DT                    $CHAR8.
                @526  TYPE_ADM                    $CHAR1.
                @527  SRC_ADMS                    $CHAR1.
                @528  AD_DGNS                     $CHAR5.
                @533  _F000006                    $CHAR1.
                @534  PTNTSTUS                    $CHAR1.
                @535  APRVL_CD                    $CHAR1.
                @536  PRO_FROM                    $CHAR8.
```



```
                @544  PRO_THRU                    $CHAR8.
                @552  GRC_DAY                     1.
                @553  PER_DIEM                    PD6.2
                @559  DED_AMT                     PD6.2
                @565  COIN_AMT                    PD6.2
                @571  BLDDEDAM                    PD6.2
                @577  BLDTCHRG                    PD6.2
                @583  BLDNCHRG                    PD6.2
                @589  PCCHGAMT                    PD6.2
                @595  NCCHGAMT                    PD6.2
                @601  TDEDAMT                     PD6.2
                @607  PPS_CPTL                    PD6.2
                @613  CPTL_HSP                    PD6.2
                @619  CPTL_FSP                    PD6.2
                @625  CPTLOUTL                    PD6.2
                @631  DISP_SHR                    PD6.2
                @637  IME_AMT                     PD6.2
                @643  CPTL_EXP                    PD6.2
                @649  HLDHRMLS                    PD6.2
                @655  DSCHFRCT                    PD3.4
                @658  DRGWTAMT                    PD4.4
                @662  UTIL_DAY                    PD2.
                @664  CR_DAY                      PD2.
                @666  COIN_DAY                    PD2.
                @668  COYR1DAY                    PD2.
                @670  COYR1AMT                    PD6.2
                @676  COYR2DAY                    PD2.
                @678  COYR2AMT                    PD6.2
                @684  LRD_USE                     PD2.
                @686  NUTILDAY                    PD3.
                @689  PSYCHDAY                    PD2.
                @691  BLDFRNSH                    PD2.
                @693  BLD_RPLC                    PD2.
                @695  BLDNRPLC                    PD2.
                @697  BLDDEDPT                    PD2.
                @699  QLFYFROM                    $CHAR8.
                @707  QLFYTHRU                    $CHAR8.
                @715  NCOVFROM                    $CHAR8.
                @723  NCOVTHRU                    $CHAR8.
                @731  GURPMTDT                    $CHAR8.
                @739  URNTCDT                     $CHAR8.
```

```
                @747  CARETHRU                    $CHAR8.
                @755  EXHST_DT                   $CHAR8.
                @763  DSCHRGDT                   $CHAR8.
                @771  DRG_CD                     $CHAR3.
                @774  OUTLR_CD                   $CHAR1.
                @775  OUTLRPMT                   PD6.2
                @781  KRON_IND                   $CHAR1.
                @782  _F000007                   $CHAR24.
        /*  FI_IP_SNF_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO IPEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}              $CHAR1.
                EDIT_CD{_I1_}              $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO IPPATCNT;
```

D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

```
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}             $CHAR1.
                PATCHCD{_I1_}              $CHAR2.
                PATCHDT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}               $CHAR1.
                MCONUM{_I1_}               $CHAR5.
                MCOOPTN{_I1_}              $CHAR1.
                MCOEFFDT{_I1_}             $CHAR8.
                MCOTRMDT{_I1_}             $CHAR8.
                MCOPLNID{_I1_}             $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}             $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}              $CHAR1.
                DEMONUM{_I1_}              $CHAR2.
                DEMOTXT{_I1_}              $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO IPDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}              $CHAR1.
```

```
                DGNS_CD{_I1_}               $CHAR5.
                _F000008                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO IPPRCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
                PRCDRIND{_I1_}              $CHAR1.
                PRCDR_CD{_I1_}              $CHAR4.
                _F000009                    $CHAR3.
                PRCDR_DT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO IPOCRCNT;
```

D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

```
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO IPREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
```

```
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000010                    $CHAR50.
        @;
    END;
    INPUT
        EOR                          $CHAR3.  @;
   /* Release trailing @ */
   input;

    DROP _I1_;
    DROP _F000001-_F000010;

 /******Creates labels for the variables on the SAS file.******/
LABEL

HICNO        =     "BENEFICIARY CLAIM#"
FROM_DT      =     "INPT FROM DATE"
THRU_DT      =     "INPT DISCHARGE DATE"

DGNSCD01        =     "DIAGNOSIS CODE_1"
DGNSCD02        =     "DIAGNOSIS CODE_2"
DGNSCD03        =     "DIAGNOSIS CODE_3"
DGNSCD04        =     "DIAGNOSIS CODE_4"
DGNSCD05        =     "DIAGNOSIS CODE_5"
DGNSCD06        =     "DIAGNOSIS CODE_6"
DGNSCD07        =     "DIAGNOSIS CODE_7"
DGNSCD08        =     "DIAGNOSIS CODE_8"
DGNSCD09        =     "DIAGNOSIS CODE_9"
```

```
DGNSCD10        =     "DIAGNOSIS CODE_10"

PRCDRCD1       =     "ICD-9 PROCEDURE CODE_1"
PRCDRCD2       =     "ICD-9 PROCEDURE CODE_2"
PRCDRCD3       =     "ICD-9 PROCEDURE CODE_3"
PRCDRCD4       =     "ICD-9 PROCEDURE CODE_4"
PRCDRCD5       =     "ICD-9 PROCEDURE CODE_5"
PRCDRCD6       =     "ICD-9 PROCEDURE CODE_6"

RVCNTR01       =     "REVENUE CENTER CODE_1"
HCPSCD01       =     "HCPCS CODE_1"
;

RUN;


 ******************************************************************;
 * STEP 2. Proc contents and proc print on the SAS claims files.
 ******************************************************************;
PROC CONTENTS DATA=out1.IP&YEAR DIRECTORY ;
TITLE "INPATIENT Y&YEAR: CLAIMS FILE WITH SELECTED VARIABLES ";
RUN;

PROC PRINT DATA=out1.IP&YEAR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR INPATIENT FILE";
RUN;
```

D#MSQ.MSQ00M7.PGMLIB99(A001IP) -- CONTINUED

```
 ******************************************************************;
 * STEP 3. CREATE FILE WITH HICNO ONLY FOR FINDER FILE LATER
 *         and proc print on the SAS ID files.
 ******************************************************************;
PROC SORT NODUPKEY DATA=out1.IP&YEAR (KEEP=HICNO) out=OUT2.IPID&YEAR;
     BY HICNO;
RUN;

PROC PRINT DATA=OUT2.IPID&YEAR (OBS=10);
TITLE "INPATIENT HICNO Y&YEAR: PRINT OUT 10 OBS ";
RUN;



ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A001OP)

```
//WF17A01O JOB (@MSQ0000,WF17),HONG,CLASS=H,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS WHEN RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       IN1 = FILE NAME OF FLAT OP FILE FROM DESY EXTRACTION
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/22/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :READ OP FILE INTO SAS FILE, GENERATE CLAIMS LEVEL FILE
//*            AND FILE WITH UNDUPLICATED HICNO FOR CROSS-REFERENCE
//*********************************************************************
/*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A001OP.OP99
  DELETE D#MSQ.@MSQ00M7.SAS.A001OP.OPID99
/*
//STEP1  EXEC SAS8
//WORK   DD UNIT=WORKDA,SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0008815.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A001OP.OP99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,300),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.A001OP.OPID99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

DATA out1.OP&YEAR (keep=hicno DGNSCD01-DGNSCD10 PRCDRCD1-PRCDRCD6
                RVCNTR01-RVCNTR45 HCPSCD01-HCPSCD45
                FROM_DT THRU_DT
          );

  INFILE IN1 MISSOVER           ;

*********************************************************************;
*  FULLNAME:  FI OUTPATIENT CLAIMS RECORD                           ;
*  VERSION 'I' SAS COPYLIB STATEMENTS                               ;
```

```
*  MEMBER NAME: UTLIOUTP                                                ;
*  CREATED BY MARVELLA BUNCH 11/01/2000                                 ;
*  UPDATED BY MARVELLA BUNCH 03/01/2001                                 ;
D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

*********************************************************************;

    LENGTH REC_LEN                                4;
    LENGTH ACRTN_NM                               3;
    LENGTH LINK_NUM                               5;
    LENGTH SGMT_CNT                               2;
    LENGTH SGMT_NUM                               2;
    LENGTH LINECNT                                3;
    LENGTH SGMTLINE                               2;
    LENGTH OPEDCNT                                2;
    LENGTH OPPATCNT                               2;
    LENGTH OPMCOCNT                               2;
    LENGTH OPPLNCNT                               2;
    LENGTH OPDEMCNT                               2;
    LENGTH OPDGNCNT                               2;
    LENGTH OPPRCCNT                               2;
    LENGTH OPCONCNT                               2;
    LENGTH OPOCRCNT                               2;
    LENGTH OPSPNCNT                               2;
    LENGTH OPVALCNT                               2;
    LENGTH OPREVCNT                               2;
    LENGTH BLDFRNSH                               3;
    LENGTH BLD_RPLC                               3;
    LENGTH BLDNRPLC                               3;
    LENGTH BLDDEDPT                               3;
    ARRAY EDITIND{13}                        $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                        $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                       $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                        $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                        $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                          $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                          $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                         $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                        $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                        $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                        $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                        $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                        $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                          $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                         $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                         $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                         $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                        $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                        $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDRIND{6}                        $    1 PRCDRND1-PRCDRND6;
    ARRAY PRCDR_CD{6}                        $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                        $    8 PRCDRDT1-PRCDRDT6;
    ARRAY CONDIND{30}                        $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                       $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                       $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                       $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                        $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                        $    1 SPNND01-SPNND10;
```

```
    ARRAY SPAN_CD{10}                          $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                         $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                         $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                           $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                           $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                               8 VALAMT01-VALAMT36;
```

D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

```
    ARRAY REVIND{45}                           $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                           $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                         $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                         $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                         $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                         $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                         $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                         $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                         $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                         $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                         $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                         $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                          $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                         $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                         $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                           $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                           $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                           $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                           $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                              4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                              8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                              8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                              8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                               8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                              8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                              8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                              8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                              8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                              8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                              8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                              8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                                8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                              8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                              8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                         $    1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_OP_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                 @1  REC_LEN                    PD3.

                @4  REC_LVL                    $CHAR1.
                @5  RIC_CD                     $CHAR1.
                @6  MQA_RIC                    $CHAR1.
                @7  CLM_TYPE                   $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                @9  HICNO                      $CHAR11.
                    @59  CAN                        $CHAR9.
```

```
                    @68  EQ_BIC                    $CHAR2.
                @20  BIC                        $CHAR2.
                @22  ST_SGMT                    $CHAR1.
                @23  STATE_CD                   $CHAR2.
                @25  FROM_DT                    $CHAR8.
                @33  THRU_DT                    $CHAR8.
                @41  WKLY_DT                    $CHAR8.
                @49  ACRTN_DT                   $CHAR8.
```

D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

```
                @57  ACRTN_NM                   PD2.
                @59  CLM_CNTL                   $CHAR23.
                @82  ORIGCNTL                   $CHAR23.
                @105  QUERY_CD                   $CHAR1.
                @106  PROVIDER                   $CHAR6.
                @112  DAILY_DT                   $CHAR8.
                @120  LINK_NUM                   PD5.
                @125  SGMT_CNT                   2.
                @127  SGMT_NUM                   2.
                @129  LINECNT                    3.
                @132  SGMTLINE                   2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                     $CHAR1.
                @135  TRANS_CD                   $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                   $CHAR1.
                    @137  TYPESRVC                   $CHAR1.
                @138  FREQ_CD                    $CHAR1.
                @139  _F000001                   $CHAR1.
                @140  MQAQUERY                   $CHAR1.
                @141  DISP_CD                    $CHAR2.
                @143  EDITDISP                   $CHAR2.
                @145  BIC_MDFY                   $CHAR1.
                @146  CNTY_CD                    $CHAR3.
                @149  RCPT_DT                    $CHAR8.
                @157  SCHLD_DT                   $CHAR8.
                @165  FRWRD_DT                   $CHAR8.
                @173  FI_NUM                     $CHAR5.
                @178  ASGN_NUM                   $CHAR8.
                @186  FIBATCH                    $CHAR4.
                @190  BENE_ZIP                   $CHAR9.
                @199  SEX                        $CHAR1.
                @200  RACE                       $CHAR1.
                @201  BENE_DOB                   $CHAR8.
                @209  MS_CD                      $CHAR2.
                @211  SURNAME                    $CHAR6.
                @217  FRSTINIT                   $CHAR1.
                @218  MDL_INIT                   $CHAR1.
                @219  CWFLOCCD                   $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
```

```
                @242  ACTIONCD                    $CHAR1.
                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
                    @269  AT_NPI                     $CHAR10.
                    @279  AT_SRNM                    $CHAR6.
                    @285  AT_GVNNM                   $CHAR1.
                    @286  AT_MDL                     $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                    $CHAR6.
D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

                    @293  OP_NPI                     $CHAR10.
                    @303  OP_SRNM                    $CHAR6.
                    @309  OP_GVN                     $CHAR1.
                    @310  OP_MDL                     $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @311  OT_UPIN                    $CHAR6.
                    @317  OT_NPI                     $CHAR10.
                    @327  OT_SRNM                    $CHAR6.
                    @333  OT_GVN                     $CHAR1.
                    @334  OT_MDL                     $CHAR1.
                @335  MDCD_PRV                   $CHAR13.
                @348  MDCDINFO                   $CHAR4.
                @352  MCOPDSW                    $CHAR1.
                @353  AUTHRZTN                   $CHAR18.
                @371  PTNTCNTL                   $CHAR20.
                @491  MDCL_REC                   $CHAR17.
                @408  PRO_CNTL                   $CHAR12.
                @420  PRO_DT                     $CHAR8.
                @428  STUS_CD                    $CHAR2.
                @430  DGNS_E                     $CHAR5.
                @435  _F000003                   $CHAR1.
                @436  PPS_IND                    $CHAR1.
                @437  TOT_CHRG                   PD6.2
                @443  _F000004                   $CHAR50.
            @493  OPEDCNT                    2.
            @495  OPPATCNT                   2.
            @497  OPMCOCNT                   1.
            @498  OPPLNCNT                   1.
            @499  OPDEMCNT                   1.
            @500  OPDGNCNT                   2.
            @502  OPPRCCNT                   2.
            @504  OPCONCNT                   2.
            @506  OPOCRCNT                   2.
            @508  OPSPNCNT                   2.
            @510  OPVALCNT                   2.
            @512  OPREVCNT                   2.
            @514  _F000005                   $CHAR4.
            /*  FI_OP_CLM_SPECF_I_GRP  */
                @518  OPSRVTYP                   $CHAR1.
                @519  OP_RFRL                    $CHAR1.
                @520  BLDDEDAM                   PD6.2
                @526  PTB_DED                    PD6.2
                @532  PTB_COIN                   PD6.2
                @538  PCCHGAMT                   PD6.2
```

```
                @544  INTRMDED                    PD6.2
                @550  PRVDRPMT                   PD6.2
                @556  BENEPMT                    PD6.2
                @562  BLDFRNSH                   PD2.
                @564  BLD_RPLC                   PD2.
                @566  BLDNRPLC                   PD2.
                @568  BLDDEDPT                   PD2.
                @570  TRANTYPE                   $CHAR1.
                @571  ESRDMTHD                   $CHAR1.
                @572  _F000006                   $CHAR24.

        /*  FI_OP_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO OPEDCNT;
        INPUT
```

D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

```
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}               $CHAR1.
                EDIT_CD{_I1_}               $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO OPPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}              $CHAR1.
                PATCHCD{_I1_}               $CHAR2.
                PATCHDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}                $CHAR1.
                MCONUM{_I1_}                $CHAR5.
                MCOOPTN{_I1_}               $CHAR1.
                MCOEFFDT{_I1_}              $CHAR8.
                MCOTRMDT{_I1_}              $CHAR8.
                MCOPLNID{_I1_}              $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO OPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}              $CHAR1.
                PLANIDCD{_I1_}              $CHAR1.
                PLANID{_I1_}                $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO OPDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}               $CHAR1.
                DEMONUM{_I1_}               $CHAR2.
                DEMOTXT{_I1_}               $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO OPDGNCNT;
```

```
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}               $CHAR1.
                DGNS_CD{_I1_}               $CHAR5.
                _F000007                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO OPPRCCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
                PRCDRIND{_I1_}              $CHAR1.
                PRCDR_CD{_I1_}              $CHAR4.
                _F000008                    $CHAR3.
                PRCDR_DT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPCONCNT;
        INPUT
```



```
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO OPOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO OPREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
```

```
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000009                    $CHAR50.
        @;
    END;
    INPUT
        EOR                         $CHAR3.
    @;
    /* Release trailing @ */
    input;

    DROP _I1_;
    DROP _F000001-_F000009;

 /******Creates labels for the variables on the SAS file.******/
LABEL

HICNO        =     "BENEFICIARY CLAIM#"
FROM_DT      =     "OPT FROM DATE"
THRU_DT      =     "OPT THROUGH DATE"

DGNSCD01        =     "DIAGNOSIS CODE_1"
DGNSCD02        =     "DIAGNOSIS CODE_2"
DGNSCD03        =     "DIAGNOSIS CODE_3"
DGNSCD04        =     "DIAGNOSIS CODE_4"
DGNSCD05        =     "DIAGNOSIS CODE_5"
```

```
DGNSCD06        =     "DIAGNOSIS CODE_6"
DGNSCD07        =     "DIAGNOSIS CODE_7"
DGNSCD08        =     "DIAGNOSIS CODE_8"
DGNSCD09        =     "DIAGNOSIS CODE_9"
DGNSCD10        =     "DIAGNOSIS CODE_10"
PRCDRCD1       =     "ICD-9 PROCEDURE CODE_1"
PRCDRCD2       =     "ICD-9 PROCEDURE CODE_2"
PRCDRCD3       =     "ICD-9 PROCEDURE CODE_3"
PRCDRCD4       =     "ICD-9 PROCEDURE CODE_4"
PRCDRCD5       =     "ICD-9 PROCEDURE CODE_5"
PRCDRCD6       =     "ICD-9 PROCEDURE CODE_6"

RVCNTR01       =     "REVENUE CENTER CODE_1"
HCPSCD01       =     "HCPCS CODE_1"
;

run;



 ********************************************************************;
 * STEP 2. Proc contents and proc print on the SAS claims files.
 ********************************************************************;
PROC CONTENTS DATA=out1.OP&YEAR DIRECTORY ;
D#MSQ.MSQ00M7.PGMLIB99(A001OP) -- CONTINUED

TITLE ''OUTPATIENT Y&YEAR: CLAIMS FILE WITH SELECTED VARIABLES '';
RUN;

PROC PRINT DATA=out1.OP&YEAR (OBS=10);
TITLE ''PRINT OUT 10 OBS FOR Y&YEAR OUTPATIENT FILE'';
RUN;

 ********************************************************************;
 * STEP 3. CREATE FILE WITH HICNO ONLY FOR FINDER FILE LATER
 *         and proc print on the SAS ID files.
 ********************************************************************;
PROC SORT NODUPKEY DATA=out1.OP&YEAR (KEEP=HICNO) out=OUT2.OPID&YEAR;
     BY HICNO;
RUN;

PROC PRINT DATA=OUT2.OPID&YEAR (OBS=10);
TITLE ''OUTPATIENT HICNO Y&YEAR: PRINT OUT 10 OBS '';
RUN;



ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A001PS)

```
//WF17A01P JOB (@MSQ0000,WF17),HONG,CLASS=H,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       IN1 = FILE NAME OF FLAT OP FILE FROM DESY EXTRACTION
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/24/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :READ PS FILE INTO SAS FILE, GENERATE CLAIMS LEVEL FILE
//*            AND FILE WITH UNDUPLICATED HICNO FOR CROSS-REFERENCE
//*********************************************************************
/*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A001PS.PS99
  DELETE D#MSQ.@MSQ00M7.SAS.A001PS.PSID99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0008810.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A001PS.PS99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(900,100),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.A001PS.PSID99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(400,50),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


*****************************************************************;
***MEMBER  SIPBVB   CREATED 07/7/2000   UPDATED 02/08/2001       ;
***SAS VERSION I INPUT STATEMENT FOR CARRIER PART B (RIC O) CLAIMS;
***THIS VERSION OUTPUTS 1 OBSERVATION FOR EACH LINE ITEM         ;
***FOR EFFICIENCY:                                              ;
***  KEEP STATEMENT SHOULD BE MODIFIED TO SPECIFY ONLY          ;
***  DATA ELEMENTS REQUIRED FOR OUTPUT FILE                     ;
*****************************************************************;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ****************************************************************;
 * STEP 1. Reads in the flat P/S file, keep only necessary
```

```
 *         variables for identifying diabetes later, and write out
 *         a SAS file.
 ********************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED

```
DATA out1.PS&YEAR (keep=hicno DGNSCD01-DGNSCD04 LINEDGNS
                HCPSCD01
                FROM_DT THRU_DT
           );

  INFILE IN1 MISSOVER           ;
    *LENGTH REC_LEN                              4;
    *LENGTH ACRTN_NM                             3;
    *LENGTH BLDFRNSH                             3;
    *LENGTH BLD_DED                              3;
    LENGTH CEDCNT                               2;
    LENGTH CPATCNT                              2;
    LENGTH CMCOCNT                              2;
    LENGTH CPLNCNT                              2;
    LENGTH CDEMCNT                              2;
    LENGTH CDGNCNT                               2;
    LENGTH CLINECNT                              2;
 ***ADD KEEP STATEMENT HERE...;
*--------------------------------------------------------------------*
|    INPUT NEAR-LINE SEQ CWFB RECORDS, FIXED & LINE-ITEMS, VER 'I'    |
*--------------------------------------------------------------------;
    INPUT
        /*  CARR_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                /*
                @51  REC_LEN                    PD3.
                @54  REC_LVL                    $CHAR1.
                @55  RIC_CD                     $CHAR1.
                @56  MQA_RIC                    $CHAR1.
                @57  CLM_TYPE                   $CHAR2.
                */
            /*  CARR_DMERC_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                @59  HICNO                       $CHAR11.

                /*
                @70  BIC                        $CHAR2.
                @72  ST_SGMT                    $CHAR1.
                @73  STATE_CD                   $CHAR2.
                */
                @75  FROM_DT                    $CHAR8.
                @83  THRU_DT                    $CHAR8.
                /*
                @91  WKLY_DT                    $CHAR8.
                @99  ACRTN_DT                   $CHAR8.
                @107  ACRTN_NM                   PD2.
                @109  CARRCNTL                   $CHAR15.
                @124  _F000001                   $CHAR38.
                @162  DAILY_DT                   $CHAR8.
                @170  LINK_NUM                   PD5.
                @175  SGMT_CNT                   2.
                @177  SGMT_NUM                   2.
```

```
                @179  LINECNT                    3.
                @182  SGMTLINE                  2.
                */
             /*  CARR_DMERC_CLM_CMN_1_I_GRP  */
                /*
                @184  _F000002                  $CHAR5.
```

D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED

```
                @189  ENTRY_CD                  $CHAR1.
                @190  _F000003                  $CHAR1.
                @191  DISP_CD                   $CHAR2.
                @193  EDITDISP                  $CHAR2.
                @195  BIC_MDFY                  $CHAR1.
                @196  CNTY_CD                   $CHAR3.
                @199  RCPT_DT                   $CHAR8.
                @207  SCHLD_DT                  $CHAR8.
                @215  FRWRD_DT                  $CHAR8.
                @223  CARR_NUM                  $CHAR5.
                @228  _F000004                  $CHAR8.
                @236  FIBATCH                   $CHAR4.
                @240  BENE_ZIP                  $CHAR9.
                @249  SEX                       $CHAR1.
                @250  RACE                      $CHAR1.
                @251  BENE_DOB                  $CHAR8.
                @259  MS_CD                     $CHAR2.
                @261  SURNAME                   $CHAR6.
                @267  FRSTINIT                  $CHAR1.
                @268  MDL_INIT                  $CHAR1.
                @269  CWFLOCCD                  $CHAR1.
                @270  PDGNS_CD                  $CHAR5.
                @275  _F000005                  $CHAR1.
                @276  PMTDNLCD                  $CHAR1.
                @277  TRTMT_CD                  $CHAR1.
                */
                @278  PMT_AMT                   PD6.2
                /*
                @284  PRPAYAMT                  PD6.2
                @290  _F000006                  $CHAR1.
                @291  RFR_UPIN                  $CHAR6.
                @297  RFR_NPI                   $CHAR10.
                @307  ASGMNTCD                  $CHAR1.
                @308  PROV_PMT                  PD6.2
                @314  BENE_PMT                  PD6.2
                @320  BENEPAID                  PD6.2
                @326  SBMTCHRG                  PD6.2
                @332  ALOWCHRG                  PD6.2
                @338  DEDAPPLY                  PD6.2
                @344  HCPCS_YR                  1.
                @345  MCOOVRRD                  $CHAR1.
                @346  HOSPOVRD                  $CHAR1.
                @347  _F000007                  $CHAR31.
                */
             /*  CARR_SPECFC_GRP  */
                /*
                @378  RFR_PRFL                  $CHAR14.
                @392  CPO_PROV                  $CHAR6.
                @398  CPO_NPI                   $CHAR10.
                @408  BLDFRNSH                  PD2.
```

```
               @410  BLD_DED                     PD2.
               */
            @412  CEDCNT                      2.
            @414  CPATCNT                     2.
            @416  CMCOCNT                     1.
            @417  CPLNCNT                     1.
            @418  CDEMCNT                     1.
            @419  CDGNCNT                     1.
            @420  CLINECNT                    2.
```
D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED
```
            @422  _F000008                    $CHAR4.

        /*  CARR_CLM_VAR_I_GRP  */
     @;
   IF CEDCNT GT 0 THEN DO;

   ARRAY EDITIND{*} $ 1 EDTIND01-EDTIND13;
   ARRAY EDIT_CD{*} $ 4 EDT_CD01-EDT_CD13;
    DO _I_=1 TO CEDCNT;
      INPUT
         EDITIND{_I_} $CHAR1.
         EDIT_CD{_I_} $CHAR4.
         @;
    END;
   END;
   IF CPATCNT GT 0 THEN DO;
   ARRAY PATCHIND{*} $ 1 PTCHND01-PTCHND99;
   ARRAY PATCHCD{*}  $ 2 PTCHCD01-PTCHCD99;
   ARRAY PATCHDT{*}  $ 8 PTCHDT01-PTCHDT99;
    DO _I_ = 1 TO CPATCNT;
      INPUT
         PATCHIND{_I_} $CHAR1.
         PATCHCD {_I_} $CHAR2.
         PATCHDT{_I_}  $CHAR8.
         @;
    END;
   END;
   IF CMCOCNT GT 0 THEN DO;
   ARRAY MCOIND{*}   $ 1   MCOIND1-MCOIND2;
   ARRAY MCONUM{*}   $ 5   MCONUM1-MCONUM2;
   ARRAY MCOOPTN{*}  $ 1   MCOOPTN1-MCOOPTN2;
   ARRAY MCOEFFDT{*} $ 8   MCEFFDT1-MCEFFDT2;
   ARRAY MCOTRMDT{*} $ 8   MCTRMDT1-MCTRMDT2;
   ARRAY MCOPLNID{*} $ 14   MCPLNID1-MCPLNID2;
    DO _I_ = 1 TO CMCOCNT;
      INPUT
         MCOIND{_I_}    $CHAR1.
         MCONUM{_I_}    $CHAR5.
         MCOOPTN{_I_}   $CHAR1.
         MCOEFFDT{_I_}  $CHAR8.
         MCOTRMDT{_I_}  $CHAR8.
         MCOPLNID{_I_}  $CHAR14.
         @;
    END;
   END;
   IF CPLNCNT GT 0 THEN DO;
   ARRAY PLANIDIN {*} $ 1 PLNDNID1-PLNDNID3;
```

```
  ARRAY PLANIDCD{*}  $ 1 PLNDCD1-PLNDCD3;
  ARRAY PLANID{*} $ 14 PLANID1-PLANID3;
    DO _I1_ = 1 TO CPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN {_I1_}            $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
  END;
  IF CDEMCNT GT 0 THEN DO;
```
D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED
```
  ARRAY DEMOIND{*} $ 1  DEMOIND1-DEMOIND5;
  ARRAY DEMONUM{*} $ 2  DEMONUM1-DEMONUM5;
  ARRAY DEMOTXT{*} $ 15 DEMOTXT1-DEMOTXT5;
   DO _I_ = 1 TO CDEMCNT;
     INPUT
        DEMOIND{_I_} $CHAR1.
        DEMONUM{_I_} $CHAR2.
        DEMOTXT{_I_} $CHAR15.
        @;
   END;
  END;
  IF CDGNCNT GT 0 THEN DO;
  ARRAY DGNSIND{*} $ 1 DGNSIND1-DGNSIND4;
  ARRAY DGNS_CD{*} $ 5 DGNSCD01-DGNSCD04;
   DO _I_ = 1 TO CDGNCNT;
     INPUT DGNSIND{_I_} $CHAR1.
           DGNS_CD{_I_} $CHAR5.
           _F000009      $CHAR1.
           @;
   END;
  END;
  DO _I_ = 1 TO CLINECNT;
     INPUT
        LINEIND   $CHAR1.
        PRF_PRFL  $CHAR10.
        PRF_UPIN  $CHAR6.
        PRF_NPI   $CHAR10.
        PRGRPNPI  $CHAR10.
        PRV_TYPE  $CHAR1.
        TAX_NUM   $CHAR10.
        PRVSTATE  $CHAR2.
        PROVZIP   $CHAR9.
        HCFASPCL  $CHAR2.
        CARRSPCL  $CHAR2.
        PRTCPTG   $CHAR1.
        ASTNT_CD  $CHAR1.
        SRVC_CNT  PD2.
        TYPSRVCB  $CHAR1.
        PTYPESRV  $CHAR2.
        PLCSRVC   $CHAR2.
        LCLTY_CD  $CHAR2.
        EXPNSDT1  $CHAR8.
        EXPNSDT2  $CHAR8.
        HCPSCD01  $CHAR5.
```

```
        MDFR_CD1  $CHAR2.
        MDFR_CD2  $CHAR2.
        BETOS     $CHAR3.
        LINE_IDE  $CHAR7.
        NDC_CD    $CHAR11.
        LINEPMT   PD6.2
        LBENPMT   PD6.2
        LPRVPMT   PD6.2
        LDEDAMT   PD6.2
        LPRPAYCD  $CHAR1.
        LPRPDAMT  PD6.2
        COINAMT   PD6.2
        LLMTAMT   PD6.2
        LINT_AMT  PD6.2
        PRPYALOW  PD6.2
```

D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED

```
        PNLTYAMT  PD6.2
        LBLD_DED  PD2.
        LSBMTCHG  PD6.2
        LALOWCHG  PD6.2
        LAB_NUM   $CHAR10.
        LAB_AMT   PD6.2
        PRCNGIND  $CHAR1.
        PMTINDSW  $CHAR1.
        DED_SW    $CHAR1.
        PMTINDCD  $CHAR1.
        MTUS_CNT  PD2.
        MTUS_IND  $CHAR1.
        LINEDGNS  $CHAR5.
        _F000010  $CHAR1.
        ANSTHUNT  PD2.
        CLIAALRT  $CHAR1.
        DCMTN_CD  $CHAR1.
        DMEST_DT  $CHAR8.
        DME_PURC  PD6.2
        NCSTY_MO  PD2.
        _F000011  $CHAR50.
        @;
 **OUTPUT EACH LINE ITEM AS AN OBSERVATION;
        OUTPUT;
  END;
  INPUT
    EOR    $CHAR3. ;
  DROP _F000001-_F000011  _I_ ;

 /******Creates labels for the variables on the SAS file.******/

LABEL

HICNO        =     "BENEFICIARY CLAIM#"
FROM_DT      =     "PART B CARRIER FROM DATE"
THRU_DT      =     "PART B CARRIER THROUGH DATE"
DGNSCD01     =     "DIAGNOSIS CODE_1"
DGNSCD02     =     "DIAGNOSIS CODE_2"
DGNSCD03     =     "DIAGNOSIS CODE_3"
DGNSCD04     =     "DIAGNOSIS CODE_4"
LINEDGNS     =     "LINE ITEM DIAGNOSIS CODE"
```

```
HCPSCD01     =     "HCPCS CODE"
;
run;



 ************************************************************************;
 * STEP 2. Proc contents and proc print on the SAS claims files.
 ************************************************************************;
PROC CONTENTS DATA=out1.PS&YEAR DIRECTORY ;
TITLE "P/S Y&YEAR: CLAIMS FILE WITH SELECTED VARIABLES ";
RUN;

PROC PRINT DATA=out1.PS&YEAR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR P/S FILE";
RUN;

 ************************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(A001PS) -- CONTINUED

```
 * STEP 3. CREATE FILE WITH HICNO ONLY FOR FINDER FILE LATER
 *         and proc print on the SAS ID files.
 ************************************************************************;
PROC SORT NODUPKEY DATA=out1.PS&YEAR (KEEP=HICNO) out=OUT2.PSID&YEAR;
     BY HICNO;
RUN;

PROC PRINT DATA=OUT2.PSID&YEAR (OBS=10);
TITLE "P/S HICNO Y&YEAR: PRINT OUT 10 OBS ";
RUN;



ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A001SN)

```
//WF17A01S JOB (@MSQ0000,WF17),HONG,CLASS=D,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       IN1 = FILE NAME OF FLAT SNF FILE FROM DESY EXTRACTION
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/21/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :READ SNF FILE INTO SAS FILE, GENERATE CLAIMS LEVEL FILE
//*            AND FILE WITH UNDUPLICATED HICNO FOR CROSS-REFERENCE
//*********************************************************************
/*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A001SN.SN99
  DELETE D#MSQ.@MSQ00M7.SAS.A001SN.SNID99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(400,200))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009774.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A001SN.SN99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.A001SN.SNID99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(200,50),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;

*********************************************************************;
*  FULL_NAME:  FI  SNF CLAIMS RECORD                                 ;
*  VERSION 'I' SAS COPYLIB STATEMENT                                  ;
*  MEMBER NAME: UTLIIPSN                                              ;
*  CREATED BY  MARVELLA BUNCH 11/01/2000                              ;
*  UPDATED BY  MARVELLA BUNCH 03/01/2001                              ;
*********************************************************************;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. Reads in the flat SNF file, keep only necessary
 *         variables for identifying diabetes later, and write out
 *         a SAS file.
```

```
 ********************************************************************;
DATA out1.SN&YEAR (keep=hicno DGNSCD01-DGNSCD10
                PRCDRCD1-PRCDRCD6
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT );
  INFILE IN1 MISSOVER;
    LENGTH REC_LEN                                4;
    LENGTH ACRTN_NM                               3;
    LENGTH LINK_NUM                               5;
    LENGTH SGMT_CNT                               2;
    LENGTH SGMT_NUM                               2;
    LENGTH LINECNT                                3;
    LENGTH SGMTLINE                               2;
    LENGTH BENE_DOB                               5;
    LENGTH IPEDCNT                                2;
    LENGTH IPPATCNT                               2;
    LENGTH IPMCOCNT                               2;
    LENGTH IPPLNCNT                               2;
    LENGTH IPDEMCNT                               2;
    LENGTH IPDGNCNT                               2;
    LENGTH IPPRCNT                                2;
    LENGTH IPCONCNT                               2;
    LENGTH IPOCRCNT                               2;
    LENGTH IPSPNCNT                               2;
    LENGTH IPVALCNT                               2;
    LENGTH IPREVCNT                               2;
    LENGTH GRC_DAY                                2;
    LENGTH UTIL_DAY                               3;
    LENGTH CR_DAY                                 3;
    LENGTH COIN_DAY                               3;
    LENGTH COYR1DAY                               3;
    LENGTH COYR2DAY                               3;
    LENGTH LRD_USE                                3;
    LENGTH NUTILDAY                               4;
    LENGTH PSYCHDAY                               3;
    LENGTH BLDFRNSH                               3;
    LENGTH BLD_RPLC                               3;
    LENGTH BLDNRPLC                               3;
    LENGTH BLDDEDPT                               3;
    ARRAY EDITIND{13}                        $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                        $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                       $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                        $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                        $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                          $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                          $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                         $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                        $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                        $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                        $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                        $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                        $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                          $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                         $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                         $    2 DEMONUM1-DEMONUM5;
```

```
    ARRAY DEMOTXT{5}                          $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                         $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                         $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDRIND{6}                         $    1 PRCDRND1-PRCDRND6;
    ARRAY PRCDR_CD{6}                         $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                         $    8 PRCDRDT1-PRCDRDT6;
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

    ARRAY CONDIND{30}                         $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                        $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                        $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                        $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                         $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                         $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                         $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                        $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                        $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                          $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                          $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                              8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                          $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                          $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                        $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                        $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                        $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                        $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                        $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                        $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                        $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                        $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                        $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                        $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                         $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                        $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                        $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                          $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                          $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                          $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                          $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                             4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                             8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                             8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                             8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                              8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                             8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                             8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                             8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                             8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                             8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                             8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                             8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                               8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                             8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                             8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                        $    1 RVDDCD01-RVDDCD45;
    INPUT
```

```
        /*  FI_IP_SNF_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                     PD3.
                @4  REC_LVL                     $CHAR1.
                @5  RIC_CD                      $CHAR1.
                @6  MQA_RIC                     $CHAR1.
                @7  CLM_TYPE                    $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

```
                    @9  CAN                        $CHAR9.
                    @9  HICNO                      $CHAR11.

                    @18  EQ_BIC                    $CHAR2.
                @20  BIC                          $CHAR2.
                @22  ST_SGMT                      $CHAR1.
                @23  STATE_CD                     $CHAR2.
                @25  FROM_DT                      $CHAR8.
                @33  THRU_DT                      $CHAR8.
                @41  WKLY_DT                      $CHAR8.
                @49  ACRTN_DT                     $CHAR8.
                @57  ACRTN_NM                     PD2.
                @59  CLM_CNTL                     $CHAR23.
                @82  ORIGCNTL                     $CHAR23.
                @105  QUERY_CD                     $CHAR1.
                @106  PROVIDER                     $CHAR6.
                @112  DAILY_DT                     $CHAR8.
                @120  LINK_NUM                     PD5.
                @125  SGMT_CNT                     2.
                @127  SGMT_NUM                     2.
                @129  LINECNT                      3.
                @132  SGMTLINE                     2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                       $CHAR1.
                @135  TRANS_CD                     $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                     $CHAR1.
                    @137  TYPESRVC                     $CHAR1.
                @138  FREQ_CD                      $CHAR1.
                @139  _F000001                     $CHAR1.
                @140  MQAQUERY                     $CHAR1.
                @141  DISP_CD                      $CHAR2.
                @143  EDITDISP                     $CHAR2.
                @145  BIC_MDFY                     $CHAR1.
                @146  CNTY_CD                      $CHAR3.
                @149  RCPT_DT                      $CHAR8.
                @157  SCHLD_DT                     $CHAR8.
                @165  FRWRD_DT                     $CHAR8.
                @173  FI_NUM                       $CHAR5.
                @178  ASGN_NUM                     $CHAR8.
                @186  FIBATCH                      $CHAR4.
                @190  BENE_ZIP                     $CHAR9.
                @199  SEX                          $CHAR1.
                @200  RACE                         $CHAR1.
                @201  BENE_DOB                     8.
                @209  MS_CD                        $CHAR2.
                @211  SURNAME                      $CHAR6.
```

```
                @217  FRSTINIT                    $CHAR1.
                @218  MDL_INIT                   $CHAR1.
                @219  CWFLOCCD                   $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
                @242  ACTIONCD                   $CHAR1.
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
                    @269  AT_NPI                     $CHAR10.
                    @279  AT_SRNM                    $CHAR6.
                    @285  AT_GVNNM                   $CHAR1.
                    @286  AT_MDL                     $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                    $CHAR6.
                    @293  OP_NPI                     $CHAR10.
                    @303  OP_SRNM                    $CHAR6.
                    @309  OP_GVN                     $CHAR1.
                    @310  OP_MDL                     $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @351  OT_UPIN                    $CHAR6.
                    @357  OT_NPI                     $CHAR10.
                    @327  OT_SRNM                    $CHAR6.
                    @333  OT_GVN                     $CHAR1.
                    @334  OT_MDL                     $CHAR1.
                @335  MDCD_PRV                   $CHAR13.
                @348  MDCDINFO                   $CHAR4.
                @352  MCOPDSW                    $CHAR1.
                @353  AUTHRZTN                   $CHAR18.
                @371  PTNTCNTL                   $CHAR20.
                @391  MDCL_REC                   $CHAR17.
                @408  PRO_CNTL                   $CHAR12.
                @420  PRO_DT                     $CHAR8.
                @428  STUS_CD                    $CHAR2.
                @430  DGNS_E                     $CHAR5.
                @435  _F000003                   $CHAR1.
                @436  PPS_IND                    $CHAR1.
                @437  TOT_CHRG                   PD6.2
                @443  _F000004                   $CHAR50.
            @493  IPEDCNT                    2.
            @495  IPPATCNT                   2.
            @497  IPMCOCNT                   1.
            @498  IPPLNCNT                   1.
            @499  IPDEMCNT                   1.
            @500  IPDGNCNT                   2.
            @502  IPPRCNT                    2.
            @504  IPCONCNT                   2.
            @506  IPOCRCNT                   2.
```

```
            @508  IPSPNCNT                    2.
            @510  IPVALCNT                    2.
            @512  IPREVCNT                    2.
            @514  _F000005                    $CHAR4.
            /*  FI_IP_SNF_CLM_SPECF_I_GRP  */
                @518  ADMSN_DT                    $CHAR8.
                @526  TYPE_ADM                    $CHAR1.
                @527  SRC_ADMS                    $CHAR1.
                @528  AD_DGNS                     $CHAR5.
                @533  _F000006                    $CHAR1.
                @534  PTNTSTUS                    $CHAR1.
                @535  APRVL_CD                    $CHAR1.
                @536  PRO_FROM                    $CHAR8.
                @544  PRO_THRU                    $CHAR8.
                @552  GRC_DAY                     1.
```

D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

```
                @553  PER_DIEM                    PD6.2
                @559  DED_AMT                     PD6.2
                @565  COIN_AMT                    PD6.2
                @571  BLDDEDAM                    PD6.2
                @577  BLDTCHRG                    PD6.2
                @583  BLDNCHRG                    PD6.2
                @589  PCCHGAMT                    PD6.2
                @595  NCCHGAMT                    PD6.2
                @601  TDEDAMT                     PD6.2
                @607  PPS_CPTL                    PD6.2
                @613  CPTL_HSP                    PD6.2
                @619  CPTL_FSP                    PD6.2
                @625  CPTLOUTL                    PD6.2
                @631  DISP_SHR                    PD6.2
                @637  IME_AMT                     PD6.2
                @643  CPTL_EXP                    PD6.2
                @649  HLDHRMLS                    PD6.2
                @655  DSCHFRCT                    PD3.4
                @658  DRGWTAMT                    PD4.4
                @662  UTIL_DAY                    PD2.
                @664  CR_DAY                      PD2.
                @666  COIN_DAY                    PD2.
                @668  COYR1DAY                    PD2.
                @670  COYR1AMT                    PD6.2
                @676  COYR2DAY                    PD2.
                @678  COYR2AMT                    PD6.2
                @684  LRD_USE                     PD2.
                @686  NUTILDAY                    PD3.
                @689  PSYCHDAY                    PD2.
                @691  BLDFRNSH                    PD2.
                @693  BLD_RPLC                    PD2.
                @695  BLDNRPLC                    PD2.
                @697  BLDDEDPT                    PD2.
                @699  QLFYFROM                    $CHAR8.
                @707  QLFYTHRU                    $CHAR8.
                @715  NCOVFROM                    $CHAR8.
                @723  NCOVTHRU                    $CHAR8.
                @731  GURPMTDT                    $CHAR8.
                @739  URNTCDT                     $CHAR8.
                @747  CARETHRU                    $CHAR8.
                @755  EXHST_DT                    $CHAR8.
```

```
                @763  DSCHRGDT                    $CHAR8.
                @771  DRG_CD                     $CHAR3.
                @774  OUTLR_CD                   $CHAR1.
                @775  OUTLRPMT                   PD6.2
                @781  KRON_IND                   $CHAR1.
                @782  _F000007                   $CHAR24.
        /*  FI_IP_SNF_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO IPEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}              $CHAR1.
                EDIT_CD{_I1_}              $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO IPPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

```
                PATCHIND{_I1_}             $CHAR1.
                PATCHCD{_I1_}              $CHAR2.
                PATCHDT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}               $CHAR1.
                MCONUM{_I1_}               $CHAR5.
                MCOOPTN{_I1_}              $CHAR1.
                MCOEFFDT{_I1_}             $CHAR8.
                MCOTRMDT{_I1_}             $CHAR8.
                MCOPLNID{_I1_}             $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}             $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}              $CHAR1.
                DEMONUM{_I1_}              $CHAR2.
                DEMOTXT{_I1_}              $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO IPDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}              $CHAR1.
                DGNS_CD{_I1_}              $CHAR5.
                _F000008                   $CHAR1.
```

```
        @;
    END;
    DO _I1_ = 1 TO IPPRCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
                PRCDRIND{_I1_}              $CHAR1.
                PRCDR_CD{_I1_}              $CHAR4.
                _F000009                    $CHAR3.
                PRCDR_DT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO IPOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO IPREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
```

```
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED

                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000010                    $CHAR50.
        @;
    END;
    INPUT
        EOR                         $CHAR3.  @;
   /* Release trailing @ */
   input;

    DROP _I1_;
    DROP _F000001-_F000010;

 /******Creates labels for the variables on the SAS file.******/
LABEL

HICNO        =     "BENEFICIARY CLAIM#"
FROM_DT      =     "SNF FROM DATE"
THRU_DT      =     "SNF DISCHARGE DATE"
DGNSCD01        =     "DIAGNOSIS CODE_1"
DGNSCD02        =     "DIAGNOSIS CODE_2"
DGNSCD03        =     "DIAGNOSIS CODE_3"
DGNSCD04        =     "DIAGNOSIS CODE_4"
DGNSCD05        =     "DIAGNOSIS CODE_5"
DGNSCD06        =     "DIAGNOSIS CODE_6"
DGNSCD07        =     "DIAGNOSIS CODE_7"
DGNSCD08        =     "DIAGNOSIS CODE_8"
DGNSCD09        =     "DIAGNOSIS CODE_9"
DGNSCD10        =     "DIAGNOSIS CODE_10"
PRCDRCD1       =     "ICD-9 PROCEDURE CODE_1"
PRCDRCD2       =     "ICD-9 PROCEDURE CODE_2"
```

```
PRCDRCD3       =     "ICD-9 PROCEDURE CODE_3"
PRCDRCD4       =     "ICD-9 PROCEDURE CODE_4"
PRCDRCD5       =     "ICD-9 PROCEDURE CODE_5"
PRCDRCD6       =     "ICD-9 PROCEDURE CODE_6"

RVCNTR01       =     "REVENUE CENTER CODE_1"
HCPSCD01       =     "HCPCS CODE_1"
;

RUN;

 ********************************************************************;
 * STEP 2. Proc contents and proc print on the SAS claims files.
 ********************************************************************;
PROC CONTENTS DATA=out1.SN&YEAR DIRECTORY ;
TITLE ''SNF Y&YEAR: CLAIMS FILE WITH SELECTED VARIABLES '';
RUN;

PROC PRINT DATA=out1.SN&YEAR (OBS=10);
TITLE ''PRINT OUT 10 OBS FOR Y&YEAR SNF FILE'';
RUN;


 ********************************************************************;
 * STEP 3. CREATE FILE WITH HICNO ONLY FOR FINDER FILE LATER
 *         and proc print on the SAS ID files.
 ********************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(A001SN) -- CONTINUED
```
PROC SORT NODUPKEY DATA=out1.SN&YEAR (KEEP=HICNO) out=OUT2.SNID&YEAR;
     BY HICNO;
RUN;

PROC PRINT DATA=OUT2.SNID&YEAR (OBS=10);
TITLE ''SKILLED NURSING HICNO Y&YEAR: PRINT OUT 10 OBS'';
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A002)

```
//WF17A002  JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
/*JBS BIND SYSS0H5
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS WHEN RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/24/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL
//* INPUTS    :IN1, IN2, IN3, IN4, IN5
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN FIVE A001 FILES AND GENERATE A FLAT FINDER FILE
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASETS EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.FLAT.A002.FINDER99
//*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(400,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001HH.HHID99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A001IP.IPID99,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ00M7.SAS.A001OP.OPID99,DISP=SHR
//IN4    DD DSN=D#MSQ.@MSQ00M7.SAS.A001PS.PSID99,DISP=SHR
//IN5    DD DSN=D#MSQ.@MSQ00M7.SAS.A001SN.SNID99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.FLAT.A002.FINDER99,
//       DISP=(NEW,CATLG,KEEP),SPACE=(CYL,(400,50),RLSE),
//       DCB=(RECFM=FB,LRECL=14,BLKSIZE=23464)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

*********************************************************************;
* STEP 1. Reads in the five SAS files (HH, IP, OP, PS, and SN)
*         and sorts the dataset by HICNO deleting all observations
*         with duplicate keys.
*********************************************************************;

DATA ALL;
SET IN1.HHID&YEAR
    IN2.IPID&YEAR
    IN3.OPID&YEAR
    IN4.PSID&YEAR
```

```
    IN5.SNID&YEAR
;
RUN;
```
D#MSQ.MSQ00M7.PGMLIB99(A002) -- CONTINUED

```
PROC SORT NODUPKEY DATA=ALL;
     BY HICNO;
RUN;

*******************************************************************;
* STEP 2.  Creates a flat finder file that will be used
*     for cross-reference
*******************************************************************;

DATA _NULL_;
     SET ALL;
     JUNK='   ';
     FILE OUT1;
     PUT   @1  HICNO   $CHAR11.
           @12 JUNK    $CHAR3.;
RUN;

PROC PRINT U DATA=ALL (OBS=10);
     TITLE '*** FINDER IS CREATED ***';
RUN;



ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(A003)

```
//WF17A003  JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
/*JBS BIND SYSS0H5
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS WHEN RUNNING THIS *****
//*       IN1  = REPLACE "WF17.@0000000.DIAB99.OUTPUT.NODUPE" WITH
//*              CROSS-REFERENCE FILE OUTPUTED BY DSAF
//*       99   = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*****************************************************************
//*****************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/12/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN A CROSS-REFERENCE FILE AND GENERATE A SAS FILE
//*****************************************************************
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASETS EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.A003.XREF99
/*
//STEP1  EXEC SAS8
//WORK   DD UNIT=WORKDA,SPACE=(CYL,(900,300))
//X37IGN DD DUMMY
//IN1    DD DSN=WF17.@0000000.DIAB99.OUTPUT.NODUPE,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,
//       DISP=(NEW,CATLG,KEEP),SPACE=(CYL,(500,50),RLSE),
//       DATACLAS=DC9DAY
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

*****************************************************************;
* STEP 1. Reads in the flat cross-reference file
*****************************************************************;

DATA TEMP (DROP=FILLER);
  INFILE IN1 MISSOVER ;
    INPUT
                @1  hicno                     $CHAR11.

                @15 filler                    $CHAR2.
                @17 xref                      $CHAR11.
;
  LABEL HICNO = "ASSIGNED UNIQUE BENEFICIARY HIC #"
        XREF = "HISTORICAL BENEFICIARY HIC #";
```

```
    run;

    ******************************************************************;
    D#MSQ.MSQ00M7.PGMLIB99(A003) -- CONTINUED

    * STEP 2. Sorts the data by XREF and outputs it as a SAS file
    ******************************************************************;

    PROC SORT NODUPKEY DATA=TEMP  OUT=OUT1.XREF&YEAR;
         BY XREF;
    RUN;

    ******************************************************************;
    * STEP 3. Proc contents and proc print on the SAS claims files.
    ******************************************************************;
    PROC CONTENTS DATA=out1.XREF&YEAR DIRECTORY ;
    TITLE   "CROSS-REFERENCE   Y&YEAR:   CROSS-REFERENCE   FILE   WITH   SELECTED
VARIABLES ";
    RUN;

    PROC PRINT DATA=out1.XREF&YEAR (OBS=10);
    TITLE "PRINT OUT 10 OBS FOR Y&YEAR CROSS-REFERENCE FILE";
    RUN;




    ******************************************************************;
    * STEP 4. OUTPUT UNIQUE CLAIMANTS ONLY INTO A TEMP DATASET
    ******************************************************************;
    PROC SORT NODUPKEY DATA=OUT1.XREF&YEAR OUT=TEMP2;
         BY HICNO;
    RUN;

    ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004HH)

```
//WF17B04H JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/28/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT2
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*             AND IDENTIFY NON-ACUTE FACE-TO-FACE CLAIMS
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.B004HH.DB99HH2
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001HH.HH99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B004HH.DB99HH2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort HHA file
 ********************************************************************;
PROC SORT DATA=IN1.HH&YEAR(keep=hicno DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT )
OUT=HH&YEAR;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge HHA file with cross-reference file to assign
```

```
 *         unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
D#MSQ.MSQ00M7.PGMLIB99(B004HH) -- CONTINUED

  MERGE HH&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=HH&YEAR;
RUN;


 ********************************************************************;
 * STEP 3. sort HHA file again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. Identify non-acute face-to-face claims, two of which are
 *         required to qualify claimants as diabetes.
 ********************************************************************;
DATA  OUT2.DB&YEAR.HH2 (KEEP=HICNO DXOUT NONACUTE FROM_DT THRU_DT);
  SET TEMP2;
    ARRAY DGNSCD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;

label
      dxout="diagnosis exclusion code"
      nonacute="non-acute face-to-face"
      from_dt="HHA from date"
      thru_dt="HHA through date";


dxout=0;
nonacute=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 10 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;

 /*** FLAG NON-ACUTE FACE-TO-FACE CLAIMS ***/
DO I = 1 TO 45 UNTIL (NONACUTE=1);
IF SUBSTR(REV_CNTR{I},1,3) IN ('010','011','012','013',
   '014','015','016','020','021','022','045','072','080',
   '049','050','051','052','053','055','056','057','058',
   '059','065','066','076', '082','083','084','085','088',
   '092','094','096') OR
   REV_CNTR{I} IN ('0981','0987', '0972','0973','0974','0975',
   '0976','0977','0978','0979','0982','0983','0984','0985',
   '0986','0988','0989')
```

```
 THEN NONACUTE=1;
END;


IF NONACUTE=0 THEN DO;
DO I = 1 TO 45 UNTIL (NONACUTE=1);
```
D#MSQ.MSQ00M7.PGMLIB99(B004HH) -- CONTINUED
```

IF HCPCS_CD{I} IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
   '99262','99263','99291','99292','99281','99282','99283','99284',
   '99285','99286','99287','99288','99356','99357','99201','99202',
   '99203','99204','99205','99211','99212','99213','99214','99215',
   '99217','99218','99219','99220','99241','99242','99243','99244',
   '99245','99271','99272','99273','99274','99275','99354','99355',
   '99381','99382','99383','99384','99385','99386','99387','99391',
   '99392','99393','99394','99395','99396','99397','99401','99402',
   '99403','99404','99411','99412','99420','99421','99422','99423',
   '99424','99425','99426','99427','99428','99429','99341','99342',
   '99343','99347','99348','99349','99350','99351','99352','99353',
   '99499','92002','92003','92004','92005','92006','92007','92008',
   '92009','92010','92011','92012','92013','92014','99301','99302',
   '99303','99311','99312','99313','99321','99322','99323')
THEN NONACUTE=1;
END;
END;

  /*** OUTPUT NON-ACUTE CLAIMS ***/
IF NONACUTE=1 THEN OUTPUT OUT2.DB&YEAR.HH2;
RUN;



 ********************************************************************;
 * STEP 5. Proc contents and proc print on the SAS non-acute claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT2.DB&YEAR.HH2 DIRECTORY ;
TITLE "HHA Y&YEAR: NON-ACUTE CLAIMS FILE ";
RUN;

PROC PRINT DATA=OUT2.DB&YEAR.HH2 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR HHA NONACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004HH_CORRECT)

```
//WF17B04H JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       00= REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :07/22/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT2
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*             AND IDENTIFY NON-ACUTE FACE-TO-FACE CLAIMS
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.B004HH.DB00HH2
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001HH.HH00,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF00,DISP=SHR
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B004HH.DB00HH2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=00;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort HHA file
 ********************************************************************;
PROC SORT DATA=IN1.HH&YEAR(keep=hicno DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT )
OUT=HH&YEAR;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge HHA file with cross-reference file to assign
```

```
 *         unique hicno to each claimant
 ******************************************************************;
DATA TEMP2 (DROP=XREF);


  MERGE HH&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=HH&YEAR;
RUN;


 ******************************************************************;
 * STEP 3. sort HHA file again by the assigned unique HICNO
 ******************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ******************************************************************;
 * STEP 4. Identify non-acute face-to-face claims, two of which are
 *         required to qualify claimants as diabetes.
 ******************************************************************;
DATA  OUT2.DB&YEAR.HH2 (KEEP=HICNO DXOUT NONACUTE FROM_DT THRU_DT);
  SET TEMP2;
    ARRAY DGNSCD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;

label
      dxout="diagnosis exclusion code"
      nonacute="non-acute face-to-face"
      from_dt="HHA from date"
      thru_dt="HHA through date";


dxout=0;
nonacute=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 10 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;

 /*** FLAG NON-ACUTE FACE-TO-FACE CLAIMS ***/
DO I = 1 TO 45 UNTIL (NONACUTE=1);
IF SUBSTR(REV_CNTR{I},1,3) IN ('010','011','012','013',
   '014','015','016','020','021','022','045','072','080',
   '049','050','051','052','053','055','056','057','058',
   '059','065','066','076', '082','083','084','085','088',
   '092','094','096') OR
   REV_CNTR{I} IN ('0981','0987', '0972','0973','0974','0975',
   '0976','0977','0978','0979','0982','0983','0984','0985',
   '0986','0988','0989')
```

```
 THEN NONACUTE=1;
END;


IF NONACUTE=0 THEN DO;
DO I = 1 TO 45 UNTIL (NONACUTE=1);



IF HCPCS_CD{I} IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
   '99262','99263','99291','99292','99281','99282','99283','99284',
   '99285','99288','99356','99357','99201','99202',
   '99203','99204','99205','99211','99212','99213','99214','99215',
   '99217','99218','99219','99220','99241','99242','99243','99244',
   '99245','99271','99272','99273','99274','99275','99354','99355',
   '99381','99382','99383','99384','99385','99386','99387','99391',
   '99392','99393','99394','99395','99396','99397','99401','99402',
   '99403','99404','99411','99412','99420','99429','99341','99342',
   '99343','99347','99348','99349','99350','99351','99352','99353',
   '99499','92002','92004','92012','92014','99301','99302',
   '99303','99311','99312','99313','99321','99322','99323')
THEN NONACUTE=1;
END;
END;

  /*** OUTPUT NON-ACUTE CLAIMS ***/
IF NONACUTE=1 THEN OUTPUT OUT2.DB&YEAR.HH2;
RUN;



 ********************************************************************;
 * STEP 5. Proc contents and proc print on the SAS non-acute claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT2.DB&YEAR.HH2 DIRECTORY ;
TITLE "HHA Y&YEAR: NON-ACUTE CLAIMS FILE ";
RUN;

PROC PRINT DATA=OUT2.DB&YEAR.HH2 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR HHA NONACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004IP)

```
//WF17B04I JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/25/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT1
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*            AND STRATIFY CLAIMS INTO ACUTE VS. NON-ACUTE
//*            FACE-TO-FACE CLAIMS: IP HOSPITAL IS CONSIDERED ACUTE
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.B004IP.DB99IP1
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001IP.IP99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.B004IP.DB99IP1,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


%LET YEAR=99;  *** YEAR OF PRODUCTION;


 ********************************************************************;
 * STEP 1. sort Inpatient file
 ********************************************************************;
PROC SORT DATA=IN1.IP&YEAR(keep=hicno DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT )
OUT=IP&YEAR;
    BY HICNO;
RUN;
```

```
 ********************************************************************;
 * STEP 2. merge Inpatient file with cross-reference file to assign
           unique hicno to each claimant
D#MSQ.MSQ00M7.PGMLIB99(B004IP) -- CONTINUED

 ********************************************************************;
DATA TEMP2 (DROP=XREF);
  MERGE IP&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=IP&YEAR;
RUN;


 ********************************************************************;
 * STEP 3. sort Inpatient file again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. stratify claims into acute vs. non-acute face-to-face claims.
           One acute is required to qualify claimants as diabetes, two
           non-acute claims are required to qualify claimants as diabetes
 ********************************************************************;
DATA OUT1.DB&YEAR.IP1 (KEEP=HICNO DXOUT ACUTE)
;
  SET TEMP2;
    ARRAY DGNSCD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;

label
      dxout="diagnosis exclusion code"
      acute="acute face-to-face"

acute=0;
dxout=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 10 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;


  /*** FLAG ACUTE AND/OR NON-ACUTE FACE-TO-FACE CLAIMS ***/
DO I = 1 TO 45 UNTIL (ACUTE=1);
IF SUBSTR(REV_CNTR{I},1,3) IN ('010','011','012','013',
   '014','015','016','020','021','022','045','072','080',
   '049','050','051','052',
   '053','055','056','057','058','059','065','066','076',
   '082','083','084','085','088','092','094','096') OR
   REV_CNTR{I} IN ('0981','0987',
```

```
   '0972','0973','0974','0975',
   '0976','0977','0978','0979','0982','0983','0984','0985',
   '0986','0988','0989')
THEN ACUTE=1;
END;
```

D#MSQ.MSQ00M7.PGMLIB99(B004IP) -- CONTINUED

```
IF ACUTE=0 THEN DO;
DO I = 1 TO 45 UNTIL (ACUTE=1);

IF HCPCS_CD{I} IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
   '99262','99263','99291','99292','99281','99282','99283','99284',
   '99285','99286','99287','99288','99356','99357',
   '99201','99202','99203','99204','99205','99211','99212','99213',
   '99214','99215','99217','99218','99219','99220','99241','99242',
   '99243','99244','99245','99271','99272','99273','99274','99275',
   '99354','99355','99381','99382','99383','99384','99385','99386',
   '99387','99391','99392','99393','99394','99395','99396','99397',
   '99401','99402','99403','99404','99411','99412','99420','99421',
   '99422','99423','99424','99425','99426','99427','99428','99429',
   '99341','99342','99343','99347','99348','99349','99350','99351',
   '99352','99353','99499','92002','92003','92004','92005','92006',
   '92007','92008','92009','92010','92011','92012','92013','92014',
   '99301','99302','99303','99311','99312','99313','99321','99322',
   '99323')
THEN ACUTE=1;
END;
END;

  /*** OUTPUT CLAIMS BASED ON ACUTE VS. NON-ACUTE ***/
IF ACUTE=1 THEN OUTPUT OUT1.DB&YEAR.IP1;
RUN;


 ********************************************************************;
 * STEP 5. Proc contents and proc print on the SAS claims files.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.DB&YEAR.IP1 DIRECTORY ;
TITLE "INPATIENT Y&YEAR: ACUTE CLAIMS FILE";
RUN;

PROC PRINT DATA=OUT1.DB&YEAR.IP1 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR INPATIENT ACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004OP)

```
//WF17B04O JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/29/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*            AND STRATIFY CLAIMS INTO ACUTE VS. NON-ACUTE
//*            FACE-TO-FACE CLAIMS
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP1
  DELETE D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP2
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001OP.OP99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP1,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort Outpatient file
 ********************************************************************;
PROC SORT DATA=IN1.OP&YEAR(keep=hicno DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT )
OUT=OP&YEAR;
    BY HICNO;
```

```
RUN;
```

D#MSQ.MSQ00M7.PGMLIB99(B004OP) -- CONTINUED

```
 ********************************************************************;
 * STEP 2. merge Outpatient file with cross-reference file to assign
           unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
  MERGE OP&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=OP&YEAR;
RUN;


 ********************************************************************;
 * STEP 3. sort Outpatient file again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. stratify claims into acute vs. non-acute face-to-face claims.
           One acute is required to qualify claimants as diabetes, two
           non-acute claims are required to qualify claimants as diabetes
 ********************************************************************;
DATA OUT1.DB&YEAR.OP1 (KEEP=HICNO DXOUT ACUTE)
     OUT2.DB&YEAR.OP2 (KEEP=HICNO DXOUT NONACUTE FROM_DT THRU_DT);
  SET TEMP2;
    ARRAY DGNSCD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;

label
      dxout="diagnosis exclusion code"
      acute="acute face-to-face"
      nonacute="non-acute face-to-face";

acute=0;
dxout=0;
nonacute=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 10 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;


  /*** FLAG ACUTE AND NON-ACUTE FACE-TO-FACE CLAIMS ***/
  /*** IN OP FILE, ER IS CONSIDERED ACUTE           ***/
DO I = 1 TO 45 UNTIL (ACUTE=1);
```

```
IF SUBSTR(REV_CNTR{I},1,3) IN ('045') OR
   REV_CNTR{I} IN ('0981')
THEN ACUTE=1;
ELSE IF
   SUBSTR(REV_CNTR{I},1,3) IN ('010','011','012','013',
   '014','015','016','020','021','022','072','080',
```
D#MSQ.MSQ00M7.PGMLIB99(B004OP) -- CONTINUED
```
   '049','050','051','052',
   '053','055','056','057','058','059','065','066','076',
   '082','083','084','085','088','092','094','096') OR
   REV_CNTR{I} IN ('0987','0972','0973','0974','0975',
   '0976','0977','0978','0979','0982','0983','0984','0985',
   '0986','0988','0989')
THEN NONACUTE=1;
END;


IF ACUTE=0 THEN DO;
DO I = 1 TO 45 UNTIL (ACUTE=1);

IF HCPCS_CD{I} IN ('99281','99282','99283','99284',
   '99285','99286','99287','99288')
THEN ACUTE=1;
ELSE IF
   HCPCS_CD{I} IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
   '99262','99263','99291','99292','99356','99357',
   '99201','99202','99203','99204','99205','99211','99212','99213',
   '99214','99215','99217','99218','99219','99220','99241','99242',
   '99243','99244','99245','99271','99272','99273','99274','99275',
   '99354','99355','99381','99382','99383','99384','99385','99386',
   '99387','99391','99392','99393','99394','99395','99396','99397',
   '99401','99402','99403','99404','99411','99412','99420','99421',
   '99422','99423','99424','99425','99426','99427','99428','99429',
   '99341','99342','99343','99347','99348','99349','99350','99351',
   '99352','99353','99499','92002','92003','92004','92005','92006',
   '92007','92008','92009','92010','92011','92012','92013','92014',
   '99301','99302','99303','99311','99312','99313','99321','99322',
   '99323')
THEN NONACUTE=1;
END;
END;

  /*** OUTPUT CLAIMS BASED ON ACUTE VS. NON-ACUTE ***/
IF ACUTE=1 THEN OUTPUT OUT1.DB&YEAR.OP1;
ELSE IF NONACUTE=1 THEN OUTPUT OUT2.DB&YEAR.OP2;

RUN;


 ******************************************************************;
 * STEP 5. Proc contents and proc print on the SAS claims files.
 ******************************************************************;
PROC CONTENTS DATA=OUT1.DB&YEAR.OP1 DIRECTORY ;
TITLE "OUTPATIENT Y&YEAR: ACUTE CLAIMS FILE";
RUN;
```

```
PROC PRINT DATA=OUT1.DB&YEAR.OP1 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR OUTPATIENT ACUTE CLAIMS";
RUN;

PROC CONTENTS DATA=OUT2.DB&YEAR.OP2 DIRECTORY ;
TITLE "OUTPATIENT Y&YEAR: NON-ACUTE CLAIMS FILE ";
RUN;

PROC PRINT DATA=OUT2.DB&YEAR.OP2 (OBS=10);
```
D#MSQ.MSQ00M7.PGMLIB99(B004OP) -- CONTINUED
```
TITLE "PRINT OUT 10 OBS FOR Y&YEAR OUTPATIENT NONACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004PS)

```
//WF17B04P JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/28/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*            AND STRATIFY CLAIMS INTO ACUTE VS. NON-ACUTE
//*            FACE-TO-FACE CLAIMS
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS1
    DELETE D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS2
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(1000,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001PS.PS99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS1,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(800,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort PART B physician  file
 ********************************************************************;
PROC SORT DATA=IN1.PS&YEAR(keep=hicno DGNSCD01-DGNSCD04 LINEDGNS
                HCPSCD01
                FROM_DT THRU_DT)
OUT=PS&YEAR;
    BY HICNO;
```

```
RUN;
```

D#MSQ.MSQ00M7.PGMLIB99(B004PS) -- CONTINUED

```
 ******************************************************************;
  * STEP 2. merge PART B physician  file with cross-reference file to
assign
            unique hicno to each claimant
 ******************************************************************;
DATA TEMP2 (DROP=XREF);
  MERGE PS&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=PS&YEAR;
RUN;


 ******************************************************************;
 * STEP 3. sort Part B physician file again by the assigned unique HICNO
 ******************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ******************************************************************;
 * STEP 4. stratify claims into acute vs. non-acute face-to-face claims.
           One acute is required to qualify claimants as diabetes, two
           non-acute claims are required to qualify claimants as diabetes
 ******************************************************************;
DATA OUT1.DB&YEAR.PS1 (KEEP=HICNO DXOUT ACUTE)
     OUT2.DB&YEAR.PS2 (KEEP=HICNO DXOUT NONACUTE FROM_DT THRU_DT);
  SET TEMP2;
    ARRAY DGNSCD{5}          $    5 LINEDGNS DGNSCD01-DGNSCD04;

label
      dxout="diagnosis exclusion code"
      acute="acute face-to-face"
      nonacute="non-acute face-to-face";

acute=0;
dxout=0;
nonacute=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 5 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;


  /*** FLAG ACUTE AND NON-ACUTE FACE-TO-FACE CLAIMS ***/
  /*** IP HOSPITAL AND ER SERVICES ARE CONSIDERED ACUTE ***/
IF HCPSCD01 IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
```

```
   '99262','99263','99291','99292','99281','99282','99283','99284',
   '99285','99286','99287','99288','99356','99357')
THEN ACUTE=1;
ELSE IF
   HCPSCD01 IN (
'99201','99202','99203','99204','99205','99211','99212','99213',
```
D#MSQ.MSQ00M7.PGMLIB99(B004PS) -- CONTINUED

```
'99214','99215','99217','99218','99219','99220','99241','99242',
'99243','99244','99245','99271','99272','99273','99274','99275',
'99354','99355','99381','99382','99383','99384','99385','99386',
'99387','99391','99392','99393','99394','99395','99396','99397',
'99401','99402','99403','99404','99411','99412','99420','99421',
'99422','99423','99424','99425','99426','99427','99428','99429',
'99341','99342','99343','99347','99348','99349','99350','99351',
'99352','99353','99499','92002','92003','92004','92005','92006',
'92007','92008','92009','92010','92011','92012','92013','92014',
'99301','99302','99303','99311','99312','99313','99321','99322',
'99323')
THEN NONACUTE=1;

  /*** OUTPUT CLAIMS BASED ON ACUTE VS. NON-ACUTE ***/
IF ACUTE=1 THEN OUTPUT OUT1.DB&YEAR.PS1;
ELSE IF NONACUTE=1 THEN OUTPUT OUT2.DB&YEAR.PS2;

RUN;


 ******************************************************************;
 * STEP 5. Proc contents and proc print on the SAS claims files.
 ******************************************************************;
PROC CONTENTS DATA=OUT1.DB&YEAR.PS1 DIRECTORY ;
TITLE "PART B  Y&YEAR: ACUTE CLAIMS FILE";
RUN;

PROC PRINT DATA=OUT1.DB&YEAR.PS1 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR PART B  ACUTE CLAIMS";
RUN;

PROC CONTENTS DATA=OUT2.DB&YEAR.PS2 DIRECTORY ;
TITLE "PART B  Y&YEAR: NON-ACUTE CLAIMS FILE ";
RUN;

PROC PRINT DATA=OUT2.DB&YEAR.PS2 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR PART B  NONACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B004SN)

```
//WF17B04S JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :05/28/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL
//* INPUTS    :IN1, IN2
//* OUTPUTS   :OUT2
//* PURPOSE   :ASSIGN UNIQUE HICNO TO CLAIMS FOR THE SAME CLAIMANT,
//*             AND IDENTIFY NON-ACUTE FACE-TO-FACE CLAIMS
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.B004SN.DB99SN2
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A001SN.SN99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B004SN.DB99SN2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort SNF file
 ********************************************************************;
PROC SORT DATA=IN1.SN&YEAR(keep=hicno DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT )
OUT=SN&YEAR;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge SNF file with cross-reference file to assign
```

```
          unique hicno to each claimant
 ******************************************************************;
DATA TEMP2 (DROP=XREF);
D#MSQ.MSQ00M7.PGMLIB99(B004SN) -- CONTINUED

  MERGE SN&YEAR          (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR    (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;

PROC DELETE DATA=SN&YEAR;
RUN;


 ******************************************************************;
 * STEP 3. sort SNF file again by the assigned unique HICNO
 ******************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ******************************************************************;
 * STEP 4. IDENTIFY NON-ACUTE FACE-TO-FACE CLAIMS, TWO OF WHICH ARE
 *         REQUIRED TO QUALIFY CLAIMANTS AS DIABETES.
 ******************************************************************;
DATA  OUT2.DB&YEAR.SN2 (KEEP=HICNO DXOUT NONACUTE FROM_DT THRU_DT);
  SET TEMP2;
    ARRAY DGNSCD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;

label
      dxout=" diagnosis exclusion code"
      nonacute=" non-acute face-to-face"
      from_dt="SNF from date"
      thru_dt="SNF discharge date";

nonacute=0;
dxout=0;

  /*** FLAG PREGNANCY RELATED DIABETES FOR EXCLUSION ***/
DO I = 1 TO 10 UNTIL (DXOUT=1);
IF SUBSTR(DGNSCD{I},1,4) in ('6488','6480') THEN DXOUT=1;
END;

 /*** FLAG NON-ACUTE FACE-TO-FACE CLAIMS ***/
DO I = 1 TO 45 UNTIL (NONACUTE=1);
IF SUBSTR(REV_CNTR{I},1,3) IN ('010','011','012','013',
   '014','015','016','020','021','022','045','072','080',
   '049','050','051','052','053','055','056','057','058',
   '059','065','066','076', '082','083','084','085','088',
   '092','094','096') OR
   REV_CNTR{I} IN ('0981','0987', '0972','0973','0974','0975',
   '0976','0977','0978','0979','0982','0983','0984','0985',
   '0986','0988','0989')
 THEN NONACUTE=1;
```

```
END;


IF NONACUTE=0 THEN DO;
DO I = 1 TO 45 UNTIL (NONACUTE=1);
```

D#MSQ.MSQ00M7.PGMLIB99(B004SN) -- CONTINUED

```
IF HCPCS_CD{I} IN ('99221','99222','99223','99231','99232','99233',
   '99238','99239','99251','99252','99253','99254','99255','99261',
   '99262','99263','99291','99292','99281','99282','99283','99284',
   '99285','99286','99287','99288','99356','99357','99201','99202',
   '99203','99204','99205','99211','99212','99213','99214','99215',
   '99217','99218','99219','99220','99241','99242','99243','99244',
   '99245','99271','99272','99273','99274','99275','99354','99355',
   '99381','99382','99383','99384','99385','99386','99387','99391',
   '99392','99393','99394','99395','99396','99397','99401','99402',
   '99403','99404','99411','99412','99420','99421','99422','99423',
   '99424','99425','99426','99427','99428','99429','99341','99342',
   '99343','99347','99348','99349','99350','99351','99352','99353',
   '99499','92002','92003','92004','92005','92006','92007','92008',
   '92009','92010','92011','92012','92013','92014','99301','99302',
   '99303','99311','99312','99313','99321','99322','99323')
THEN NONACUTE=1;
END;
END;

  /*** OUTPUT NON-ACUTE CLAIMS ***/


IF NONACUTE=1 THEN OUTPUT OUT2.DB&YEAR.SN2;
RUN;

 ********************************************************************;
 * STEP 5. Proc contents and proc print on the SAS non-acute claims file.
 ********************************************************************;


PROC CONTENTS DATA=OUT2.DB&YEAR.SN2 DIRECTORY ;
TITLE "SNF  Y&YEAR: NON-ACUTE CLAIMS FILE ";
RUN;

PROC PRINT DATA=OUT2.DB&YEAR.SN2 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR SNF NONACUTE CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(B005)

```
//WF17B005 JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/13/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2, IN3, IN4, IN5, IN6, IN7, IN8, IN9
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :CONSOLIDATE ALL ACUTE AND NON-ACUTE CLAIMS FROM IP, SNF,
//*            OP, HHA, PS. IDENTIFY DIABETES COHORT USING CRITERIA: (1)
//*            ONE ACUTE FACE-TO-FACE, OR (2)TWO NON-ACUTE FACE-TO-FACE
//*            AT LEAST SEVEN DAYS APART
//*********************************************************************
//*
//**IF THE OLD DATASETS EXIST
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.FLAT.B005.DIAB99
    DELETE D#MSQ.@MSQ00M7.SAS.B005.DIABET99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.B004IP.DB99IP1,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP1,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS1,DISP=SHR
//IN4    DD DSN=D#MSQ.@MSQ00M7.SAS.B004SN.DB99SN2,DISP=SHR
//IN5    DD DSN=D#MSQ.@MSQ00M7.SAS.B004OP.DB99OP2,DISP=SHR
//IN6    DD DSN=D#MSQ.@MSQ00M7.SAS.B004HH.DB99HH2,DISP=SHR
//IN7    DD DSN=D#MSQ.@MSQ00M7.SAS.B004PS.DB99PS2,DISP=SHR
//IN8   DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR
//IN9   DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.FLAT.B005.DIAB99,
//       DISP=(NEW,CATLG,KEEP),SPACE=(CYL,(400,50),RLSE),
//       DCB=(RECFM=FB,LRECL=14,BLKSIZE=23464)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.B005.DIABET99,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen ;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ****************************************************************;
```

```
     * STEP 1. combine acute claims files and output one hicno for each
     *         claimant
     ************************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(B005) -- CONTINUED

```
    DATA ACUTE (KEEP=HICNO DXOUT2);
      SET IN1.DB&YEAR.IP1
          IN2.DB&YEAR.OP1
          IN3.DB&YEAR.PS1
    ;
    BY HICNO;
    ATTRIB DXOUT2  LENGTH=2 LABEL='EXCLUSION FLAG FOR PREGNANCY RELATED DX';
    RETAIN DXOUT2 ;
    IF FIRST.HICNO THEN DO;
       DXOUT2=0;
    END;
    IF DXOUT=1 THEN DXOUT2=1;
    IF LAST.HICNO THEN OUTPUT;
    RUN;


     ************************************************************************;
     * STEP 2. combine all non-acute claims files into one file
     ************************************************************************;
    DATA TEMP1 (DROP= NONACUTE FROM_DT THRU_DT);
      SET IN4.DB&YEAR.SN2 (IN=X)
          IN5.DB&YEAR.OP2 (IN=Y)
          IN6.DB&YEAR.HH2 (IN=Z)
          IN7.DB&YEAR.PS2 (IN=A)
    ;
    BY HICNO;
       FROM_DT2=INPUT(FROM_DT, YYMMDD8.);
       THRU_DT2=INPUT(THRU_DT, YYMMDD8.);
    RUN;

     ************************************************************************;
     * STEP 3. sort combine non-acute claims file
     ************************************************************************;
    PROC SORT DATA=TEMP1;
        BY HICNO  THRU_DT2;
    RUN;



     ************************************************************************;
      * STEP 4. apply two claims seven days apart criterion and output one
hicno
     *         each for those claimants who meet the criterion
     ************************************************************************;
    DATA NONACUTE (KEEP=HICNO DXOUT2);
      SET TEMP1;
    BY HICNO;
    ATTRIB DXOUT2  LENGTH=2 LABEL='EXCLUSION FLAG FOR PREGNANCY RELATED DX';
    RETAIN DXOUT2 datef;
    IF FIRST.HICNO THEN DO;
       DXOUT2=0;
       datef= THRU_DT2;
    END;
```

```
    IF DXOUT=1 THEN DXOUT2=1;

    IF LAST.HICNO THEN DO;
      datel= THRU_DT2;
      diff = datel - datef;
    if diff >=7 then output;
    D#MSQ.MSQ00M7.PGMLIB99(B005) -- CONTINUED

    END;
    RUN;

    PROC DELETE DATA=TEMP1;
    RUN;

     ********************************************************************;
     * STEP 5. combine acute diabetes hicno with non-acute diabetes hicno
     *         to get tentative diabetes cohort
     ********************************************************************;
    DATA DIABETES;
      SET ACUTE
          NONACUTE
    ;
    BY HICNO;
    RETAIN DXOUT3;
    IF FIRST.HICNO THEN DO;
       DXOUT3=0;
    END;
    IF DXOUT2=1 THEN DXOUT3=1;
      /***OUTPUT ONE HICNO FOR EACH CLAIMANT ***/
    IF LAST.HICNO THEN OUTPUT;
    RUN;

    PROC DELETE DATA=ACUTE NONACUTE;
    RUN;


     ********************************************************************;
     * STEP 6. merge cross-reference file with Base Denominator File in case
     *         hicno changes, before merge with tentative diabetes cohort
     ********************************************************************;
    DATA TEMP2 (KEEP=HICNO );
      MERGE IN8.DENOMB&YEAR   (IN=X KEEP= Y&YEAR.HICNO Y&YEAR.M7
                               WHERE=(Y&YEAR.M7='1')
RENAME=(Y&YEAR.HICNO=XREF))
            IN9.XREF&YEAR     (IN=Y KEEP=HICNO XREF );
      BY XREF;
    IF X AND Y;
    RUN;


     ********************************************************************;
     * STEP 7. sort by hicno, before merge with tentative diabetes cohort
     ********************************************************************;
    PROC SORT NODUPKEY DATA=TEMP2;
    BY HICNO;
    RUN;

     ********************************************************************;
```

```
 * STEP 8. merge tentative diabetes cohort with Base Denominator File
 *         who are eligible for M7 (diabetes) selection to get final
 *         diabetes cohort
 ********************************************************************;
DATA DIAB;
  MERGE TEMP2    (IN=X)
        DIABETES (IN=Y)
;
BY HICNO;
```
D#MSQ.MSQ00M7.PGMLIB99(B005) -- CONTINUED
```

IF X AND Y;
RUN;

PROC DELETE DATA=TEMP2 ;
RUN;


PROC FREQ DATA=DIAB;
TABLE DXOUT3;
TITLE  'HOW MANY CLAIMANTS ARE FLAGGED';
TITLE1 'FOR EXCLUSION?';
RUN;

********************************************************************;
* STEP 9.  sort xref file by hicno, then merge diabetes cohort file with
*          it to get all their historical hicno for use in creating
*          finder file
********************************************************************;
PROC SORT DATA=IN9.XREF&YEAR (KEEP=HICNO XREF) OUT=XREF;
BY HICNO;
RUN;

DATA DIAB2 (KEEP=XREF )
     CHECK;
  /***ONLY OUTPUT THOSE HICNO WHICH ARE NOT FLAGGED FOR EXCLUSION ***/
  MERGE DIAB               (IN=X WHERE=(DXOUT3=0))
        XREF               (IN=Y  );
  BY HICNO;
IF X AND Y THEN OUTPUT DIAB2;
ELSE IF X THEN OUTPUT CHECK;  ***THIS SHOULD NEVER HAPPEN;
RUN;


********************************************************************;
* STEP 10.  Creates a flat finder file that will be used
*           to pull all claims using DESY
********************************************************************;

DATA _NULL_;
     SET DIAB2 ;
     JUNK='   ';
     FILE OUT1;
     PUT   @1  XREF    $CHAR11.
           @12 JUNK    $CHAR3.;
RUN;

PROC PRINT U DATA=DIAB (OBS=10);
```

```
     TITLE '*** PASS2: FINDER IS CREATED ***';
RUN;


*******************************************************************;
* STEP 11.  Creates a SAS file of tentative diabetes cohort in case
*           need it in the future
*******************************************************************;
DATA OUT2.DIABET&YEAR (KEEP=HICNO XREF )
     CHECK;
  /***ONLY OUTPUT THOSE HICNO WHICH ARE NOT FLAGGED FOR EXCLUSION ***/
  MERGE DIABETES               (IN=X WHERE=(DXOUT3=0))
```
D#MSQ.MSQ00M7.PGMLIB99(C005) -- CONTINUED
```
        XREF                   (IN=Y  );
  BY HICNO;
IF X AND Y THEN OUTPUT OUT2.DIABET&YEAR;
ELSE IF X THEN OUTPUT CHECK;  ***THIS SHOULD NEVER HAPPEN;
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM)

```
//WF1706D JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1 = REPLACE FILE NAME WITH FILE NAME OUTPUTTED FROM DESY
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/13/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN DME RAW CLAIMS DATA AND GENERATE FLAGS FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES ON THE FLY
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C006DM.Y99DMCLM
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(900,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009817.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006DM.Y99DMCLM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ********************************************************************;
 * STEP 1. read in DME claims
            note: this input statement is an old version. For the latest
version
           of files generated by DESY, need to add 50 spaces to each
variable.
           See 1992-1998 or 2000-2001 for example.
 ********************************************************************;

*************************************************************;
***MEMBER $SDMVI   CREATED 07/8/2000  UPDATED ON 2/08/2001       ;
```

```
***SAS VERSION I INPUT STATEMENT FOR DMERC   (RIC M) CLAIMS       ;
***THIS VERSION OUTPUTS 1 OBSERVATION FOR EACH LINE ITEM          ;
***FOR EFFICIENCY:                                                ;
```
D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
***  KEEP STATEMENT SHOULD BE MODIFIED TO SPECIFY ONLY            ;
***  DATA ELEMENTS REQUIRED FOR OUTPUT FILE                       ;
*****************************************************************;

DATA TEMP1 (keep=hicno
                 DGNSCD01-DGNSCD04
                 LINEDGNS
                 HCPSCD01
                 FROM_DT THRU_DT PMT_AMT LINEPMT HCFASPCL );
  INFILE IN1 STOPOVER;

    *LENGTH REC_LEN                                  4;
    *LENGTH ACRTN_NM                                 3;
    LENGTH DEDCNT                                   2;
    LENGTH DPATCNT                                  2;
    LENGTH DMCOCNT                                  2;
    LENGTH DPLNCNT                                  2;
    LENGTH DDEMCNT                                  2;
    LENGTH DDGNCNT                                  2;
    LENGTH DLINECNT                                 2;
    INPUT
        /*  DMERC_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                /*
                @1  REC_LEN                    PD3.
                @4  REC_LVL                    $CHAR1.
                @5  RIC_CD                     $CHAR1.
                @6  MQA_RIC                    $CHAR1.
                @7  CLM_TYPE                   $CHAR2.
                */
            /*  CARR_DMERC_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                @9  HICNO                       $CHAR11.

                @20  BIC                        $CHAR2.
                @22  ST_SGMT                    $CHAR1.
                @23  STATE_CD                   $CHAR2.
                @25  FROM_DT                    YYMMDD8.
                @33  THRU_DT                    YYMMDD8.
                /*
                @41  WKLY_DT                    $CHAR8.
                @49  ACRTN_DT                   $CHAR8.
                @57  ACRTN_NM                   PD2.
                @59  CARRCNTL                   $CHAR15.
                @74  _F000001                   $CHAR38.
                @112  DAILY_DT                  $CHAR8.
                @120  LINK_NUM                  PD5.
                @125  SGMT_CNT                   2.
                @127  SGMT_NUM                   2.
                @129  LINECNT                    3.
                @132  SGMTLINE                   2.
                */
            /*  CARR_DMERC_CLM_CMN_2_I_GRP  */
```

```
                /*
                @134  _F000002                  $CHAR5.
                @139  ENTRY_CD                  $CHAR1.
                @140  _F000003                  $CHAR1.
                @141  DISP_CD                   $CHAR2.
                @143  EDITDISP                  $CHAR2.
```
D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
                @145  BIC_MDFY                  $CHAR1.
                @146  CNTY_CD                   $CHAR3.
                @149  RCPT_DT                   $CHAR8.
                @157  SCHLD_DT                  $CHAR8.
                @165  FRWRD_DT                  $CHAR8.
                @173  CARR_NUM                  $CHAR5.
                @178  _F000004                  $CHAR8.
                @186  FIBATCH                   $CHAR4.
                @190  BENE_ZIP                  $CHAR9.
                @199  SEX                       $CHAR1.
                @200  RACE                      $CHAR1.
                @201  BENE_DOB                  $CHAR8.
                @209  MS_CD                     $CHAR2.
                @211  SURNAME                   $CHAR6.
                @217  FRSTINIT                  $CHAR1.
                @218  MDL_INIT                  $CHAR1.
                @219  CWFLOCCD                  $CHAR1.
                @220  PDGNS_CD                  $CHAR5.
                @225  _F000005                  $CHAR1.
                @226  PMTDNLCD                  $CHAR1.
                @227  TRTMT_CD                  $CHAR1.
                */
                @228  PMT_AMT                   PD6.2
                /*
                @234  PRPAYAMT                  PD6.2
                @240  _F000006                  $CHAR1.
                @241  ORD_UPIN                  $CHAR6.
                @247  ORD_NPI                   $CHAR10.
                @257  ASGMNTCD                  $CHAR1.
                @258  PROV_PMT                  PD6.2
                @264  BENE_PMT                  PD6.2
                @270  BENEPAID                  PD6.2
                @276  SBMTCHRG                  PD6.2
                @282  ALOWCHRG                  PD6.2
                @288  DEDAPPLY                  PD6.2
                @294  HCPCS_YR                  1.
                @295  MCOOVRRD                  $CHAR1.
                @296  HOSPOVRD                  $CHAR1.
                @297  _F000007                  $CHAR31.
                */
            @328  DEDCNT                    2.
            @330  DPATCNT                   2.
            @332  DMCOCNT                   1.
            @333  DPLNCNT                   1.
            @334  DDEMCNT                   1.
            @335  DDGNCNT                   1.
            @336  DLINECNT                  2.
            @338  _F000008                  $CHAR4.
        /*  DMERC_CLM_VAR_I_GRP  */
    @;
```

```
   IF DEDCNT GT 0 THEN DO;
    ARRAY EDITIND{*}                          $    1 EDTIND01-EDTIND13;
    ARRAY EDIT_CD{*}                          $    4 EDT_CD01-EDT_CD13;
    DO _I_ = 1 TO DEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I_}             $CHAR1.
                EDIT_CD{_I_}             $CHAR4.
        @;
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
    END;
   END;
   IF DPATCNT GT 0 THEN DO;
    ARRAY PATCHIND{*}                         $    1 PTCHND01-PTCHND99;
    ARRAY PATCHCD{*}                          $    2 PTCHCD01-PTCHCD99;
    ARRAY PATCHDT{*}                          $    8 PTCHDT01-PTCHDT99;
    DO _I_ = 1 TO DPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I_}            $CHAR1.
                PATCHCD{_I_}             $CHAR2.
                PATCHDT{_I_}             $CHAR8.
        @;
    END;
   END;
   IF DMCOCNT GT 0 THEN DO;
    ARRAY MCOIND{*}                           $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{*}                           $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{*}                          $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{*}                         $    8 MCEFFDT1-MCEFFDT2;
    ARRAY MCOTRMDT{*}                         $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{*}                         $   14 MCPLNID1-MCPLNID2;
    DO _I_ = 1 TO DMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I_}              $CHAR1.
                MCONUM{_I_}              $CHAR5.
                MCOOPTN{_I_}             $CHAR1.
                MCOEFFDT{_I_}            $CHAR8.
                MCOTRMDT{_I_}            $CHAR8.
                MCOPLNID{_I_}            $CHAR14.
        @;
    END;
   END;
   IF DPLNCNT GT 0 THEN DO ;
    ARRAY PLANIDIN{*}                         $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{*}                         $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{*}                           $   14 PLANID1-PLANID3;
    DO _I_  = 1 TO DPLNCNT;
        INPUT
            /*  CLM_PAYERID_I_GRP  */
                PLANIDIN{_I_}            $CHAR1.
                PLANIDCD{_I_}            $CHAR1.
                PLANID  {_I_}            $CHAR14.
        @;
    END;
   END;
```

```
   IF DDEMCNT GT 0 THEN DO;
    ARRAY DEMOIND{*}                          $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{*}                          $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{*}                          $   15 DEMOTXT1-DEMOTXT5;
    DO _I_ = 1 TO DDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I_}             $CHAR1.
                DEMONUM{_I_}             $CHAR2.
                DEMOTXT{_I_}             $CHAR15.
        @;
    END;
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
   END;
   IF DDGNCNT GT 0 THEN DO;
    ARRAY DGNSIND{4}                          $    1 DGNSIND1-DGNSIND4;
    ARRAY DGNS_CD{4}                          $    5 DGNSCD01-DGNSCD04;
    DO _I_ = 1 TO DDGNCNT;
        INPUT
            /*  CARR_CLM_DGNS_I_GRP  */
                DGNSIND{_I_}             $CHAR1.
                DGNS_CD{_I_}             $CHAR5.
_F000009                $CHAR1.
        @;
    END;
   END;
    DO _I1_ = 1 TO DLINECNT;
        INPUT
            /*  DMERC_LINE_I_GRP  */
                LINEIND                     $CHAR1.
                SUPLRNUM                    $CHAR10.
                SUP_NPI                     $CHAR10.
                PRCNG_ST                    $CHAR2.
                PRVSTATE                    $CHAR2.
                SUP_TYPE                    $CHAR1.
                TAX_NUM                     $CHAR10.
                HCFASPCL                    $CHAR2.
                PRTCPTG                     $CHAR1.
                SRVC_CNT                    PD2.
                TYPSRVCB                    $CHAR1.
                PLCSRVC                     $CHAR2.
                EXPNSDT1                    $CHAR8.
                EXPNSDT2                    $CHAR8.
                HCPSCD01                    $CHAR5.
                MDFR_CD1                    $CHAR2.
                MDFR_CD2                    $CHAR2.
                MDFR_CD3                    $CHAR2.
                MDFR_CD4                    $CHAR2.
                BETOS                       $CHAR3.
                LINE_IDE                    $CHAR7.
                NOC_TXT                     $CHAR14.
                NDC_CD                      $CHAR11.
                LINEPMT                     PD6.2
                LBENPMT                     PD6.2
                LPRVPMT                     PD6.2
                LDEDAMT                     PD6.2
                LPRPAYCD                    $CHAR1.
```

```
                  LPRPDAMT                    PD6.2
                  COINAMT                     PD6.2
                  LINT_AMT                    PD6.2
                  PRPYALOW                    PD6.2
                  PNLTYAMT                    PD6.2
                  LSBMTCHG                    PD6.2
                  LALOWCHG                    PD6.2
                  SCRNSVGS                    PD6.2
                  DME_PURC                    PD6.2
                  PRCNGIND                    $CHAR1.
                  PMTINDSW                    $CHAR1.
                  DED_SW                      $CHAR1.
                  PMTINDCD                    $CHAR1.
                  DME_UNIT                    PD4.
                  UNIT_IND                    $CHAR1.
D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

                  LINEDGNS                    $CHAR5.
                  _F000010                    $CHAR1.
                  DCMTN_CD                    $CHAR1.
                  SUSP_IND                    $CHAR4.
                  RSLT_IND                    $CHAR1.
                  WAIVERSW                    $CHAR1.
                  DCSN_IND                    $CHAR1.
                  _F000011                    $CHAR50.
        @;
     *OUTPUT EACH LINE AS AN OBSERVATION;
        OUTPUT;
    END;
    INPUT
       EOR                        $CHAR3.;
    DROP _F000001-_F000011 _I_ ;
run;


 ********************************************************************;
 * STEP 2. generate process and progression of disease measure flags
 ********************************************************************;
DATA OUT1.Y&YEAR.DMCLM (keep=HICNO FROM_DT THRU_DT HCFASPCL LINEPMT
      i_hgl
      i_lipid1 i_lip3_1-i_lip3_3
      i_self   i_educ1          i_eye1
      i_urine2 i_foot1 i_shoe

      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar i_gla2 i_ede i_las
      i_ptc
);
  SET TEMP1;

   attrib i_hgl     length=$1 label='Hgl A1c Test';
   attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
    attrib i_lip3_1  length=$1 label='Lipid Profiling definition 3: part
1';
```

```
       attrib i_lip3_2  length=$1 label='Lipid Profiling definition 3: part
2';
       attrib i_lip3_3  length=$1 label='Lipid Profiling definition 3: part
3';

      attrib i_self  length=$1 label='Self-Monitoring';
      attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
      attrib i_eye1   length=$1 label='Eye Examination 1';

      attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
      attrib i_foot1  length=$1 label='Podiatry visits 1';
      attrib i_shoe    length=$1 label='Therapeutic Shoes';

      attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
      attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
      attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
      attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
      attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
      attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
      attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
      attrib i_led5  length=$1 label='diabetes w periph cir dis';
      attrib i_led6  length=$1 label='Lower limb amputations';
      attrib i_led7  length=$1 label='Incision and drainage below facia';
      attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
      attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

      attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
      attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
      attrib i_neph    length=$1 label='Nephropathy';
      attrib i_ret1    length=$1 label='Retinopathy 1';
      attrib i_blind  length=$1 label='Blindness';
      attrib i_catar  length=$1 label='Diabetic cataract';

      attrib i_gla2   length=$1 label='Glaucoma 2';
      attrib i_ede    length=$1 label='Macular Edema';
      attrib i_las    length=$1 label='Laser treatment';
      attrib  i_ptc       length=$1  label='Progression  to  PTCA,  including
stents';

      attrib hicno    length=$11 label='Health insurance claim number';
      attrib from_dt  length=8   label='Claim from date';
      attrib thru_dt  length=8   label='Claim through date';
      attrib HCFASPCL length=$2  label='Line item HCFA specialty code';
      attrib linepmt  length=8   label='Line item payment amount';

      /* Array for input variables: LINEDGNS is primary DX  */
       ARRAY DGNS_CD{5}                                $ 5 LINEDGNS DGNSCD01-
DGNSCD04;


      attrib tmp      length=$2  label='Temporary';

      /* Initialize indicators */
      array inds {*} $
```

```
      i_hgl
      i_lipid1 i_lip3_1-i_lip3_3
      i_self   i_educ1  i_eye1
      i_urine2 i_foot1  i_shoe

      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc
;
   do _i=1 to dim(inds);
      inds{_i} = '0';
   end;

   ******** (1) For process of disease ************************;

   /* CPT code for Hgl A1c Test*/
         tmp = put(HCPSCD01, $hgl.);
         if tmp = '1' then i_hgl = '1';
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
   /* CPT code  for lipid profiling*/
         /* i_lipid1 */
         tmp = put(HCPSCD01, $lipid1f.);
         if tmp = '1' then i_lipid1 = '1';

         /* i_lipid3: 1-3 on the same day */
         tmp = put(HCPSCD01, $lipid3f.);
         if tmp = '1' then i_lip3_1 = '1';
         ELSE if tmp = '2' then i_lip3_2 = '1';
         ELSE if tmp = '3' then i_lip3_3 = '1';


   /* CPT: Self monitoring  */
         tmp = put(HCPSCD01, $monitor.);
         if tmp = '1' then i_self = '1';
         ELSE if tmp = '2' then i_self = '1';


   /* diabetes education */
   /* educ1: CPT */
         tmp = put(HCPSCD01, $educ1f.);
         if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/

   /* Eye examination */
      /*Eye1: CPT */
         tmp = put(HCPSCD01, $eye1f.);
         if tmp = '1' then i_eye1 = '1';       *don't need specialty code;
         ELSE   if   tmp   =   '2'   then   i_eye1_2   =   '1';*w   provider
specialty'18','41;
         if HCFASPCL in ('18','41') and i_eye1_2 = '1' then i_eye1='1';


   /* CPT: Microalbumin Screening /Quantitative Urine Protein */
```

```
            /* urine2:  */
            tmp = put(HCPSCD01, $urine2f.);
            if tmp = '1' then i_urine2 = '1';

      /* DX for foot1: Podiatry visits */
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $foot.);
            if tmp = '1' and HCFASPCL ='48' then i_foot1 = '1';
      end;

      /* CPT: Therapeutic Shoes */
            tmp = put(HCPSCD01, $shoe.);
            if tmp = '1' then i_shoe = '1';


      ******** (2) For progression of disease ************************;

      /* Lower Extremity Diseases               */
      /* separate flags for principal or secondary DX for '1'*/
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $lower.);
            if i=1 then do;
              if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
            end;
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
            else do;
              if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
            end;
      end;

      /* Lower1: DX for category 2 to 5  */
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $lower1f.);
            if i=1 then do;
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;

            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;


      /* Lower2: CPT for category 6 to 9 for Physician only? */
            tmp = put(HCPSCD01, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
```

```
      ELSE if tmp = '2' then i_led7 = '1';
          * incision and drainage of bone cortex No 8*;
      ELSE if tmp = '3' then i_led8 = '1';
          * skin and subcutaneous debridement for gangrene No 9*;
      ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;


*********  Final indicator for Low Extremety Disease *****;
   if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                        then i_led9 = '1';


/* DX: Neurological complications */
/* Autonomic Neuropathy */
do i =1 to 5;
      tmp = put(DGNS_CD{i}, $neuro.);
      if i=1 then do;
        if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
      end;

      else do;
        if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
      end;
end;


/* DX: Nephropathy   */
do i =1 to 5;
```

D#MSQ.MSQ00M7.PGMLIB99(C006DM) -- CONTINUED

```
      tmp = put(DGNS_CD{i}, $nephro.);
      if tmp = '2' then i_neph = '1';
 end;

/* Diabetic Eye Diseases   */
do i =1 to 5;
     /* No 1: retinopathy DX  */
      tmp = put(DGNS_CD{i}, $retin1f.);
      if tmp = '2' then i_ret1 = '1';


     /* No 2: blidness DX  */
      tmp = put(DGNS_CD{i}, $blind.);
      if tmp = '1' then i_blind = '1';

     /* No 3: Diabetic cataract DX   */
      tmp = put(DGNS_CD{i}, $catar.);
      if tmp = '1' then i_catar = '1';

     /* No 4:  Glaucoma DX  */
      tmp = put(DGNS_CD{i}, $glauco.);
      if tmp = '1' then i_gla2 = '1';          /*glauco2*/

     /* No 5:  Macular Edema DX  */
      tmp = put(DGNS_CD{i}, $edema.);
      if tmp = '1' then i_ede = '1';
end;
```

```
   /* No 6: Laser treatment CPT    */
   /* Laser CPT: for physician  */
         tmp = put(HCPSCD01, $laser1f.);
         if tmp = '1' then i_las = '1';

   /* PTCA: CPT code */
         tmp = put(HCPSCD01, $PTCA2F.);
         if tmp = '1' then i_ptc = '1';
RUN;



 ********************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.DMCLM DIRECTORY ;
TITLE "DME Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.DMCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR DME CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH)

```
//WF1706H JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1  = REPLACE FILE NAME WITH FILE NAME OUTPUTTED FROM DESY
//*       99   = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/05/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN RAW CLAIMS DATA AND GENERATE FLAGS FOR PROCESS AND
//*            PROGRESSION OF DISEASE MEASURES ON THE FLY
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C006HH.Y99HHCLM
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009819.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006HH.Y99HHCLM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ********************************************************************;
 * STEP 1. read in HHA claims
            note: this input statement is an old version. For the latest
version
          of files generated by DESY, need to add 50 spaces to each
variable.
          See 1992-1998 or 2000-2001 for example.
 ********************************************************************;
*********************************************************************;
*  FULLNAME:  FI HHA CLAIMS RECORD                                   ;
*  VERSION 'I' SAS COPYLIB STATEMENTS                                ;
```

```
*  MEMBER NAME: UTLIHHA                                                    ;
*  CREATED BY MARVELLA BUNCH 11/01/2000, modified by Hong Chen             ;
*  UPDATED BY MARVELLA BUNCH 03/01/2001                                    ;
```
D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
*********************************************************************;
DATA TEMP10  (keep=hicno STUS_CD
                DGNSCD01-DGNSCD10
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                FROM_DT THRU_DT PMT_AMT );
  INFILE IN1 MISSOVER           ;
    LENGTH REC_LEN                              4;
    LENGTH ACRTN_NM                             3;
    LENGTH LINK_NUM                             5;
    LENGTH SGMT_CNT                             2;
    LENGTH SGMT_NUM                             2;
    LENGTH LINECNT                              3;
    LENGTH SGMTLINE                             2;
    LENGTH HHEDCNT                              2;
    LENGTH HHPATCNT                             2;
    LENGTH HHMCOCNT                             2;
    LENGTH HHPLANNT                             2;
    LENGTH HHDEMCNT                             2;
    LENGTH HHDGNCNT                             2;
    LENGTH HHCONCNT                             2;
    LENGTH HHOCRCNT                             2;
    LENGTH HHSPNCNT                             2;
    LENGTH HHVALCNT                             2;
    LENGTH HHREVCNT                             2;
    LENGTH VISITCNT                             3;
    ARRAY EDITIND{13}                       $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                       $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                      $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                       $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                       $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                         $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                         $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                        $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                       $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                       $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                       $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                       $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                       $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                         $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                        $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                        $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                        $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                       $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                       $    5 DGNSCD01-DGNSCD10;
    ARRAY CONDIND{30}                       $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                      $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                      $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                      $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                       $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                       $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                       $    2 SPANCD01-SPANCD10;
```

```
    ARRAY SPANFROM{10}                          $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                          $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                            $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                            $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                                8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                            $    1 REVIND01-REVIND45;
D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

    ARRAY REV_CNTR{45}                          $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                            $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                          $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                          $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                          $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                          $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                          $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                          $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                          $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                          $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                          $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                          $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                          $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                           $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                          $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                          $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                            $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                            $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                            $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                            $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                               4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                               8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                               8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                               8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                                8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                               8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                               8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                               8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                               8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                               8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                               8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                               8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                                 8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                               8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                               8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                          $    1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_HHA_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                   PD3.
                @4  REC_LVL                   $CHAR1.
                @5  RIC_CD                    $CHAR1.
                @6  MQA_RIC                   $CHAR1.
                @7  CLM_TYPE                  $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                    @9  CAN                        $CHAR9.

                    @9  HICNO                      $CHAR11.
```

```
                    @18  EQ_BIC                        $CHAR2.
                @20  BIC                          $CHAR2.
                @22  ST_SGMT                      $CHAR1.
                @23  STATE_CD                     $CHAR2.
                @25  FROM_DT                      YYMMDD8.

                @33  THRU_DT                      YYMMDD8.

                @41  WKLY_DT                      $CHAR8.
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
                @49  ACRTN_DT                     $CHAR8.
                @57  ACRTN_NM                     PD2.
                @59  CLM_CNTL                     $CHAR23.
                @82  ORIGCNTL                     $CHAR23.
                @105  QUERY_CD                     $CHAR1.
                @106  PROVIDER                     $CHAR6.
                @112  DAILY_DT                     $CHAR8.
                @120  LINK_NUM                     PD5.
                @125  SGMT_CNT                     2.
                @127  SGMT_NUM                     2.
                @129  LINECNT                      3.
                @132  SGMTLINE                     2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                       $CHAR1.
                @135  TRANS_CD                     $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                     $CHAR1.
                    @137  TYPESRVC                     $CHAR1.
                @138  FREQ_CD                      $CHAR1.
                @139  _F000001                     $CHAR1.
                @140  MQAQUERY                     $CHAR1.
                @141  DISP_CD                      $CHAR2.
                @143  EDITDISP                     $CHAR2.
                @145  BIC_MDFY                     $CHAR1.
                @146  CNTY_CD                      $CHAR3.
                @149  RCPT_DT                      $CHAR8.
                @157  SCHLD_DT                     $CHAR8.
                @165  FRWRD_DT                     $CHAR8.
                @173  FI_NUM                       $CHAR5.
                @178  ASGN_NUM                     $CHAR8.
                @186  FIBATCH                      $CHAR4.
                @190  BENE_ZIP                     $CHAR9.
                @199  SEX                          $CHAR1.
                @200  RACE                         $CHAR1.
                @201  BENE_DOB                     $CHAR8.
                @209  MS_CD                        $CHAR2.
                @211  SURNAME                      $CHAR6.
                @217  FRSTINIT                     $CHAR1.
                @218  MDL_INIT                     $CHAR1.
                @219  CWFLOCCD                     $CHAR1.
                @220  PDGNS_CD                     $CHAR5.
                @225  _F000002                     $CHAR1.
                @226  NOPAY_CD                     $CHAR1.
                @227  TRTMT_CD                     $CHAR1.
                @228  PMT_AMT                      PD6.2
                @234  PRPAYAMT                     PD6.2
                @240  PRPAY_CD                     $CHAR1.
```

```
                @241  CANCELCD                    $CHAR1.
                @242  ACTIONCD                    $CHAR1.
                @243  APRVL_DT                    $CHAR8.
                @251  PRSTATE                     $CHAR2.
                @253  ORGNPINM                    $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                     $CHAR6.
                    @269  AT_NPI                      $CHAR10.
                    @279  AT_SRNM                     $CHAR6.
                    @285  AT_GVNNM                    $CHAR1.
                    @286  AT_MDL                      $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
                    @287  OP_UPIN                     $CHAR6.
                    @293  OP_NPI                      $CHAR10.
                    @303  OP_SRNM                     $CHAR6.
                    @309  OP_GVN                      $CHAR1.
                    @310  OP_MDL                      $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @311  OT_UPIN                     $CHAR6.
                    @317  OT_NPI                      $CHAR10.
                    @327  OT_SRNM                     $CHAR6.
                    @333  OT_GVN                      $CHAR1.
                    @334  OT_MDL                      $CHAR1.
                @335  MDCD_PRV                    $CHAR13.
                @348  MDCDINFO                    $CHAR4.
                @352  MCOPDSW                     $CHAR1.
                @353  AUTHRZTN                    $CHAR18.
                @371  PTNTCNTL                    $CHAR20.
                @391  MDCL_REC                    $CHAR17.
                @408  PRO_CNTL                    $CHAR12.
                @420  PRO_DT                      $CHAR8.
                @428  STUS_CD                     $CHAR2.
                @430  DGNS_E                      $CHAR5.
                @435  _F000003                    $CHAR1.
                @436  PPS_IND                     $CHAR1.
                @437  TOT_CHRG                    PD6.2
                @443  _F000004                    $CHAR50.
            @493  HHEDCNT                     2.
            @495  HHPATCNT                    2.
            @497  HHMCOCNT                    1.
            @498  HHPLANNT                    1.
            @499  HHDEMCNT                    1.
            @500  HHDGNCNT                    2.
            @502  _F000005                    $CHAR2.
            @504  HHCONCNT                    2.
            @506  HHOCRCNT                    2.
            @508  HHSPNCNT                    2.
            @510  HHVALCNT                    2.
            @512  HHREVCNT                    2.
            @514  _F000006                    $CHAR4.
            /*  FI_HHA_CLM_SPECF_I_GRP  */
                @518  LUPAIND                     $CHAR1.
                @519  HHA_RFRL                    $CHAR1.
                @520  VISITCNT                    PD2.
                @522  QLFYFROM                    $CHAR8.
                @530  QLFYTHRU                    $CHAR8.
```

```
                @538  DSCHRGDT                    $CHAR8.
                @546  HHSTRTDT                   $CHAR8.
                @554  _F000007                   $CHAR16.
        /*  FI_HHA_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO HHEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}              $CHAR1.
                EDIT_CD{_I1_}              $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO HHPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
                PATCHIND{_I1_}             $CHAR1.
                PATCHCD{_I1_}              $CHAR2.
                PATCHDT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}               $CHAR1.
                MCONUM{_I1_}               $CHAR5.
                MCOOPTN{_I1_}              $CHAR1.
                MCOEFFDT{_I1_}             $CHAR8.
                MCOTRMDT{_I1_}             $CHAR8.
                MCOPLNID{_I1_}             $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO HHPLANNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}             $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO HHDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}              $CHAR1.
                DEMONUM{_I1_}              $CHAR2.
                DEMOTXT{_I1_}              $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO HHDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}              $CHAR1.
                DGNS_CD{_I1_}              $CHAR5.
                _F000008                   $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO HHCONCNT;
```

```
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO HHOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
```



```
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO HHVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO HHREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
```

```
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000009                    $CHAR50.
        @;
    END;
    INPUT       EOR                         $CHAR3.@;
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
   /* Release trailing @ */
   input;

    DROP _I1_ ;
    DROP _F000001-_F000009;
RUN;



 ********************************************************************;
 * STEP 2. generate process and progression of disease measure flags
 ********************************************************************;
DATA OUT1.Y&YEAR.HHCLM (keep=hicno
                STUS_CD
                FROM_DT THRU_DT PMT_AMT
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar i_gla2 i_ede i_las
      i_ptc
);
  SET TEMP10;

   attrib i_hgl     length=$1 label='Hgl A1c Test';
   attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
   attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';

   attrib i_self  length=$1 label='Self-Monitoring';
   attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
   attrib i_educ2  length=$1 label='Diabetes Education: definition 2';
```

```
      attrib i_eye1   length=$1 label='Eye Examination 1';

      attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
      attrib i_shoe    length=$1 label='Therapeutic Shoes';

      attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
      attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
      attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
      attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
      attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
      attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
      attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
      attrib i_led5  length=$1 label='diabetes w periph cir dis';
      attrib i_led6  length=$1 label='Lower limb amputations';
      attrib i_led7  length=$1 label='Incision and drainage below facia';
      attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
      attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

      attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
      attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
      attrib i_neph    length=$1 label='Nephropathy';
      attrib i_ret1    length=$1 label='Retinopathy 1';
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
      attrib i_blind  length=$1 label='Blindness';
      attrib i_catar  length=$1 label='Diabetic cataract';

      attrib i_gla2   length=$1 label='Glaucoma 2';
      attrib i_ede    length=$1 label='Macular Edema';
      attrib i_las    length=$1 label='Laser treatment';

      attrib i_ptc  length=$1 label='Progression to PTCA, including stents';

      attrib hicno    length=$11 label='Health insurance claim number';
      attrib from_dt  length=8   label='Claim from date';
      attrib thru_dt  length=8   label='Claim through date';
      attrib stus_cd  length=$2  label='Patient discharge status code';
      attrib pmt_amt  length=8   label='Claim payment amount';

      /* Array for input variables */
       ARRAY DGNS_CD{10}                        $    5 DGNSCD01-DGNSCD10;

       ARRAY REV_CNTR{45}                       $    4 RVCNTR01-RVCNTR45;
       ARRAY HCPCS_CD{45}                       $    5 HCPSCD01-HCPSCD45;

      attrib tmp      length=$2  label='Temporary';

      /* Initialize indicators */
      array inds {*} $
         i_hgl
         i_lipid1 i_lip3_1-i_lip3_3
         i_self   i_educ1 i_educ2 i_eye1
         i_urine2 i_shoe

         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
         i_led4   i_led5
```

```
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar   i_gla2 i_ede i_las
      i_ptc
;
   do _i=1 to dim(inds);
      inds{_i} = '0';
   end;

   ******** (1) For process of disease ************************;

   do i =1 to 45;
         /* CPT code for Hgl A1c Test*/
         tmp = put(HCPCS_CD{i}, $hgl.);
         if tmp = '1' then i_hgl = '1';

         /* CPT code  for lipid profiling*/
         /* i_lipid1 */
         tmp = put(HCPCS_CD{i}, $lipid1f.);
         if tmp = '1' then i_lipid1 = '1';

         /* i_lipid3: 1-3 on the same day */
         tmp = put(HCPCS_CD{i}, $lipid3f.);
         if tmp = '1' then i_lip3_1 = '1';
         ELSE if tmp = '2' then i_lip3_2 = '1';
         ELSE if tmp = '3' then i_lip3_3 = '1';
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
         /* CPT: Self monitoring  */
         tmp = put(HCPCS_CD{i}, $monitor.);
         if tmp = '1' then i_self = '1';
         ELSE if tmp = '2' then i_self = '1';
   end;
   ********* create final indicator for i_lipid3 ***************;
   IF (i_lip3_1='1' AND i_lip3_2='1' AND i_lip3_3='1' )
                       THEN i_lipid3='1';  else i_lipid3='0';

   /* diabetes education */
   /* educ1: CPT  */
   do i =1 to 45;
         /* i_educ11 */
         tmp = put(HCPCS_CD{i}, $educ1f.);
         if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/
   end;

   /* educ2: revenue code */
   do i =1 to 45;
         /* i_educ11 */
         tmp = put(REV_CNTR{i}, $educ2f.);         /*FOR ALL YR*/
         if tmp = '1' then i_educ2 = '1';
   end;

   do i =1 to 45;
         /* Eye examination: CPT */
         tmp = put(HCPCS_CD{i}, $eye1f.);
         if tmp = '1' then i_eye1 = '1';
```

```
            /* CPT: Microalbumin Screening /Quantitative Urine Protein*/
            /* urine2:  */
            tmp = put(HCPCS_CD{i}, $urine2f.);
            if tmp = '1' then i_urine2 = '1';

            /* CPT: Therapeutic Shoes*/
            tmp = put(HCPCS_CD{i}, $shoe.);
            if tmp = '1' then i_shoe = '1';
      end;


      ******** (2) For progression of disease ************************;

      /* Lower Extremity Diseases               */
      /* separate flags for principal or secondary DX for '1'*/
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $lower.);
            if i=1 then do;
              if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
            end;

            else do;
              if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
            end;
      end;

      /* Lower1: DX for category 2 to 5  */
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $lower1f.);
            if i=1 then do;
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;

            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;


      /* Lower2: CPT for category 6 to 9 */
      do i =1 to 45;
            tmp = put(HCPCS_CD{i}, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
            ELSE if tmp = '2' then i_led7 = '1';
                * incision and drainage of bone cortex No 8*;
            ELSE if tmp = '3' then i_led8 = '1';
```

```
          * skin and subcutaneous debridement for gangrene No 9*;
       ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;
end;

********* Final indicator for Low Extremety Disease *****;
   if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                        then i_led9 = '1';

/* DX: Neurological complications */
/* Autonomic Neuropathy */
do i =1 to 10;
      tmp = put(DGNS_CD{i}, $neuro.);
      if i=1 then do;
        if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
      end;

      else do;
        if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
      end;
end;

/* DX: Nephropathy   */
do i =1 to 10;
      tmp = put(DGNS_CD{i}, $nephro.);
      if tmp = '2' then i_neph = '1';
 end;

/* Diabetic Eye Diseases   */
do i =1 to 10;
     /* No 1: retinopathy DX  */
      tmp = put(DGNS_CD{i}, $retin1f.);
      if tmp = '2' then i_ret1 = '1';
```

D#MSQ.MSQ00M7.PGMLIB99(C006HH) -- CONTINUED

```
     /* No 2: blidness DX  */
      tmp = put(DGNS_CD{i}, $blind.);
      if tmp = '1' then i_blind = '1';

     /* No 3: Diabetic cataract DX   */
      tmp = put(DGNS_CD{i}, $catar.);
      if tmp = '1' then i_catar = '1';

     /* No 4:  Glaucoma DX  */
      tmp = put(DGNS_CD{i}, $glauco.);
      if tmp = '1' then i_gla2 = '1';          /*glauco2*/

     /* No 5:  Macular Edema DX  */
      tmp = put(DGNS_CD{i}, $edema.);
      if tmp = '1' then i_ede = '1';
end;


  /* No 6: Laser treatment CPT    */
  /* Laser CPT  */
  do i =1 to 45;
      tmp = put(HCPCS_CD{i}, $laser1f.);
```

```
         if tmp = '1' then i_las = '1';
     end;

   /* PTCA: CPT code for Progression to PTCA */
     do i =1 to 45;
         tmp = put(HCPCS_CD{i}, $PTCA2F.);
         if tmp = '1' then i_ptc = '1';
     end;
RUN;


 ******************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
 ******************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.HHCLM DIRECTORY ;
TITLE "HHA Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.HHCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR HHA CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP)

```
//WF1706I JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1 = REPLACE FILE NAME WITH FILE NAME OUTPUTTED FROM DESY
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/14/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN INPATIENT RAW CLAIMS DATA AND GENERATE FLAGS FOR
//*            PROCESS AND PROGRESSION OF DISEASE MEASURES ON THE FLY
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C006IP.Y99IPCLM
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009815.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006IP.Y99IPCLM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen OBS=MAX;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ******************************************************************;
 * STEP 1. read in Inpatient Hospital claims
            note: this input statement is an old version. For the latest
version
           of files generated by DESY, need to add 50 spaces to each
variable.
           See 1992-1998 or 2000-2001 for example.
 ******************************************************************;

*******************************************************************;
*  FULL_NAME:  FI INPATIENT SNF CLAIMS RECORD                      ;
```

```
*  VERSION 'I' SAS COPYLIB STATEMENT                                    ;
*  MEMBER NAME: UTLIIPSN                                                ;
*  CREATED BY  MARVELLA BUNCH 11/01/2000                                ;
```
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
*  UPDATED BY  MARVELLA BUNCH 03/01/2001                                ;
*********************************************************************;


DATA TEMP1 (KEEP=HICNO PROVIDER
                 DGNSCD01-DGNSCD10
                 PRCDRCD1-PRCDRCD6
                 RVCNTR01-RVCNTR45
                 HCPSCD01-HCPSCD45
                 STUS_CD PER_DIEM UTIL_DAY COIN_DAY
                 FROM_DT THRU_DT PMT_AMT
                 PTNTSTUS ACC_CHRG ER_CHRG);
  INFILE IN1 MISSOVER;
    LENGTH REC_LEN                              4;
    LENGTH ACRTN_NM                             3;
    LENGTH LINK_NUM                             5;
    LENGTH SGMT_CNT                             2;
    LENGTH SGMT_NUM                             2;
    LENGTH LINECNT                              3;
    LENGTH SGMTLINE                             2;
    LENGTH BENE_DOB                             5;
    LENGTH IPEDCNT                              2;
    LENGTH IPPATCNT                             2;
    LENGTH IPMCOCNT                             2;
    LENGTH IPPLNCNT                             2;
    LENGTH IPDEMCNT                             2;
    LENGTH IPDGNCNT                             2;
    LENGTH IPPRCNT                              2;
    LENGTH IPCONCNT                             2;
    LENGTH IPOCRCNT                             2;
    LENGTH IPSPNCNT                             2;
    LENGTH IPVALCNT                             2;
    LENGTH IPREVCNT                             2;
    LENGTH GRC_DAY                              2;
    LENGTH UTIL_DAY                             3;
    LENGTH CR_DAY                               3;
    LENGTH COIN_DAY                             3;
    LENGTH COYR1DAY                             3;
    LENGTH COYR2DAY                             3;
    LENGTH LRD_USE                              3;
    LENGTH NUTILDAY                             4;
    LENGTH PSYCHDAY                             3;
    LENGTH BLDFRNSH                             3;
    LENGTH BLD_RPLC                             3;
    LENGTH BLDNRPLC                             3;
    LENGTH BLDDEDPT                             3;
    ARRAY EDITIND{13}                       $   1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                       $   4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                      $   1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                       $   2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                       $   8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                         $   1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                         $   5 MCONUM1-MCONUM2;
```

```
    ARRAY MCOOPTN{2}                          $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                         $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                         $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                         $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                         $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                         $    1 PLNDCD1-PLNDCD3;
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

    ARRAY PLANID{3}                           $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                          $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                          $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                          $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                         $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                         $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDRIND{6}                         $    1 PRCDRND1-PRCDRND6;
    ARRAY PRCDR_CD{6}                         $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                         $    8 PRCDRDT1-PRCDRDT6;
    ARRAY CONDIND{30}                         $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                        $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                        $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                        $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                         $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                         $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                         $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                        $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                        $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                          $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                          $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                              8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                          $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                        $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                          $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                        $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                        $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                        $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                        $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                        $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                        $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                        $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                        $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                        $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                        $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                        $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                         $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                        $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                        $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                          $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                          $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                          $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                          $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                             4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                             8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                             8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                             8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                              8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                             8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                             8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                             8 RVMSP201-RVMSP245;
```

```
    ARRAY REVPCCHG{45}                              8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                              8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                              8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                              8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                                8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                              8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                              8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                          $   1 RVDDCD01-RVDDCD45;
    INPUT
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
        /*  FI_IP_SNF_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                     PD3.
                @4  REC_LVL                     $CHAR1.
                @5  RIC_CD                      $CHAR1.
                @6  MQA_RIC                     $CHAR1.
                @7  CLM_TYPE                    $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                    @9  CAN                        $CHAR9.
                    @9  HICNO                      $CHAR11.

                    @18  EQ_BIC                    $CHAR2.
                @20  BIC                        $CHAR2.
                @22  ST_SGMT                    $CHAR1.
                @23  STATE_CD                   $CHAR2.
                @25  FROM_DT                    YYMMDD8.

                @33  THRU_DT                    YYMMDD8.

                @41  WKLY_DT                    $CHAR8.
                @49  ACRTN_DT                   $CHAR8.
                @57  ACRTN_NM                   PD2.
                @59  CLM_CNTL                   $CHAR23.
                @82  ORIGCNTL                   $CHAR23.
                @105  QUERY_CD                   $CHAR1.
                @106  PROVIDER                   $CHAR6.
                @112  DAILY_DT                   $CHAR8.
                @120  LINK_NUM                   PD5.
                @125  SGMT_CNT                   2.
                @127  SGMT_NUM                   2.
                @129  LINECNT                    3.
                @132  SGMTLINE                   2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                     $CHAR1.
                @135  TRANS_CD                   $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                  $CHAR1.
                    @137  TYPESRVC                  $CHAR1.
                @138  FREQ_CD                    $CHAR1.
                @139  _F000001                   $CHAR1.
                @140  MQAQUERY                   $CHAR1.
                @141  DISP_CD                    $CHAR2.
                @143  EDITDISP                   $CHAR2.
                @145  BIC_MDFY                   $CHAR1.
                @146  CNTY_CD                    $CHAR3.
                @149  RCPT_DT                    $CHAR8.
```

```
                @157  SCHLD_DT                    $CHAR8.
                @165  FRWRD_DT                   $CHAR8.
                @173  FI_NUM                     $CHAR5.
                @178  ASGN_NUM                   $CHAR8.
                @186  FIBATCH                    $CHAR4.
                @190  BENE_ZIP                   $CHAR9.
                @199  SEX                        $CHAR1.
                @200  RACE                       $CHAR1.
                @201  BENE_DOB                   8.
                @209  MS_CD                      $CHAR2.
                @211  SURNAME                    $CHAR6.
                @217  FRSTINIT                   $CHAR1.
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
                @218  MDL_INIT                   $CHAR1.
                @219  CWFLOCCD                   $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
                @242  ACTIONCD                   $CHAR1.
                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
                    @269  AT_NPI                     $CHAR10.
                    @279  AT_SRNM                    $CHAR6.
                    @285  AT_GVNNM                   $CHAR1.
                    @286  AT_MDL                     $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                    $CHAR6.
                    @293  OP_NPI                     $CHAR10.
                    @303  OP_SRNM                    $CHAR6.
                    @309  OP_GVN                     $CHAR1.
                    @310  OP_MDL                     $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @351  OT_UPIN                    $CHAR6.
                    @357  OT_NPI                     $CHAR10.
                    @327  OT_SRNM                    $CHAR6.
                    @333  OT_GVN                     $CHAR1.
                    @334  OT_MDL                     $CHAR1.
                @335  MDCD_PRV                   $CHAR13.
                @348  MDCDINFO                   $CHAR4.
                @352  MCOPDSW                    $CHAR1.
                @353  AUTHRZTN                   $CHAR18.
                @371  PTNTCNTL                   $CHAR20.
                @391  MDCL_REC                   $CHAR17.
                @408  PRO_CNTL                   $CHAR12.
                @420  PRO_DT                     $CHAR8.
                @428  STUS_CD                    $CHAR2.
                @430  DGNS_E                     $CHAR5.
                @435  _F000003                   $CHAR1.
                @436  PPS_IND                    $CHAR1.
```

```
            @437  TOT_CHRG                    PD6.2
            @443  _F000004                   $CHAR50.
          @493  IPEDCNT                     2.
          @495  IPPATCNT                    2.
          @497  IPMCOCNT                    1.
          @498  IPPLNCNT                    1.
          @499  IPDEMCNT                    1.
          @500  IPDGNCNT                    2.
          @502  IPPRCNT                     2.
          @504  IPCONCNT                    2.
          @506  IPOCRCNT                    2.
          @508  IPSPNCNT                    2.
          @510  IPVALCNT                    2.
          @512  IPREVCNT                    2.
          @514  _F000005                    $CHAR4.
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
          /*  FI_IP_SNF_CLM_SPECF_I_GRP  */
            @518  ADMSN_DT                    $CHAR8.
            @526  TYPE_ADM                    $CHAR1.
            @527  SRC_ADMS                    $CHAR1.
            @528  AD_DGNS                     $CHAR5.
            @533  _F000006                    $CHAR1.
            @534  PTNTSTUS                    $CHAR1.
            @535  APRVL_CD                    $CHAR1.
            @536  PRO_FROM                    $CHAR8.
            @544  PRO_THRU                    $CHAR8.
            @552  GRC_DAY                     1.
            @553  PER_DIEM                    PD6.2
            @559  DED_AMT                     PD6.2
            @565  COIN_AMT                    PD6.2
            @571  BLDDEDAM                    PD6.2
            @577  BLDTCHRG                    PD6.2
            @583  BLDNCHRG                    PD6.2
            @589  PCCHGAMT                    PD6.2
            @595  NCCHGAMT                    PD6.2
            @601  TDEDAMT                     PD6.2
            @607  PPS_CPTL                    PD6.2
            @613  CPTL_HSP                    PD6.2
            @619  CPTL_FSP                    PD6.2
            @625  CPTLOUTL                    PD6.2
            @631  DISP_SHR                    PD6.2
            @637  IME_AMT                     PD6.2
            @643  CPTL_EXP                    PD6.2
            @649  HLDHRMLS                    PD6.2
            @655  DSCHFRCT                    PD3.4
            @658  DRGWTAMT                    PD4.4
            @662  UTIL_DAY                    PD2.
            @664  CR_DAY                      PD2.
            @666  COIN_DAY                    PD2.
            @668  COYR1DAY                    PD2.
            @670  COYR1AMT                    PD6.2
            @676  COYR2DAY                    PD2.
            @678  COYR2AMT                    PD6.2
            @684  LRD_USE                     PD2.
            @686  NUTILDAY                    PD3.
            @689  PSYCHDAY                    PD2.
            @691  BLDFRNSH                    PD2.
```

```
                @693  BLD_RPLC                    PD2.
                @695  BLDNRPLC                   PD2.
                @697  BLDDEDPT                   PD2.
                @699  QLFYFROM                   $CHAR8.
                @707  QLFYTHRU                   $CHAR8.
                @715  NCOVFROM                   $CHAR8.
                @723  NCOVTHRU                   $CHAR8.
                @731  GURPMTDT                   $CHAR8.
                @739  URNTCDT                    $CHAR8.
                @747  CARETHRU                   $CHAR8.
                @755  EXHST_DT                   $CHAR8.
                @763  DSCHRGDT                   $CHAR8.
                @771  DRG_CD                     $CHAR3.
                @774  OUTLR_CD                   $CHAR1.
                @775  OUTLRPMT                   PD6.2
                @781  KRON_IND                   $CHAR1.
                @782  _F000007                   $CHAR24.
        /*  FI_IP_SNF_CLM_TRLR_I_GRP  */
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

    @;
    DO _I1_ = 1 TO IPEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}              $CHAR1.
                EDIT_CD{_I1_}              $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO IPPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}             $CHAR1.
                PATCHCD{_I1_}              $CHAR2.
                PATCHDT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}               $CHAR1.
                MCONUM{_I1_}               $CHAR5.
                MCOOPTN{_I1_}              $CHAR1.
                MCOEFFDT{_I1_}             $CHAR8.
                MCOTRMDT{_I1_}             $CHAR8.
                MCOPLNID{_I1_}             $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}             $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
```

```
                DEMOIND{_I1_}               $CHAR1.
                DEMONUM{_I1_}               $CHAR2.
                DEMOTXT{_I1_}               $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO IPDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}               $CHAR1.
                DGNS_CD{_I1_}               $CHAR5.
                _F000008                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO IPPRCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
                PRCDRIND{_I1_}              $CHAR1.
                PRCDR_CD{_I1_}              $CHAR4.
                _F000009                    $CHAR3.
                PRCDR_DT{_I1_}              $CHAR8.
        @;
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
    END;
    DO _I1_ = 1 TO IPCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO IPOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO IPREVCNT;
        INPUT
```

```
                /*  CLM_REV_CNTR_I_GRP  */
                    REVIND{_I1_}                $CHAR1.
                    REV_CNTR{_I1_}              $CHAR4.
                    REV_DT{_I1_}                $CHAR8.
                    REVANSI1{_I1_}              $CHAR5.
                    REVANSI2{_I1_}              $CHAR5.
                    REVANSI3{_I1_}              $CHAR5.
                    REVANSI4{_I1_}              $CHAR5.
                    APCHIPPS{_I1_}              $CHAR5.
                    HCPCS_CD{_I1_}              $CHAR5.
                    MDFR_CD1{_I1_}              $CHAR2.
                    MDFR_CD2{_I1_}              $CHAR2.
                    MDFR_CD3{_I1_}              $CHAR2.
                    MDFR_CD4{_I1_}              $CHAR2.
                    MDFR_CD5{_I1_}              $CHAR2.
                    PMTMTHD{_I1_}               $CHAR2.
                    DSCNTIND{_I1_}              $CHAR1.
                    PACKGIND{_I1_}              $CHAR1.
                    PRICNG{_I1_}                $CHAR2.
                    OTAF_1{_I1_}                $CHAR1.
                    OTAF_2{_I1_}                $CHAR1.
                    IDENDC{_I1_}                $CHAR24.
                    REV_UNIT{_I1_}              PD4.
                    REV_RATE{_I1_}              PD6.2
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

                    REVBLOOD{_I1_}              PD6.2
                    REVDCTBL{_I1_}              PD6.2
                    WAGEADJ{_I1_}               PD6.2
                    RDCDCOIN{_I1_}              PD6.2
                    REV_MSP1{_I1_}              PD6.2
                    REV_MSP2{_I1_}              PD6.2
                    REVPCCHG{_I1_}              PD6.2
                    RPRVDPMT{_I1_}              PD6.2
                    RBENEPMT{_I1_}              PD6.2
                    PTNTRESP{_I1_}              PD6.2
                    REVPMT{_I1_}                PD6.2
                    REV_CHRG{_I1_}              PD6.2
                    REV_NCVR{_I1_}              PD6.2
                    REVDEDCD{_I1_}              $CHAR1.
                    _F000010                    $CHAR50.
            @;
        END;
        INPUT
            EOR                         $CHAR3.  @;
       /* Release trailing @ */
       input;

        DROP _I1_;
        DROP _F000001-_F000010;

    *** create total accommodation charge count, total ER service charge
count;
    *** which will be used to identify stand-alone ER claims in next code;
        ACC_CHRG=0;
        ER_CHRG=0;

        DO _I1_ = 1 TO 45;
```

```
          IF (REV_CNTR{_I1_} >='0100' AND REV_CNTR{_I1_} <='0169') THEN
          DO;
               ACC_CHRG=SUM(ACC_CHRG, REV_CHRG{_I1_} GT 0);
          END;

          ELSE               IF               REV_CNTR{_I1_}               IN
('0450','0451','0452','0456','0459','0981')
          THEN DO;
               ER_CHRG=SUM(ER_CHRG, REV_CHRG{_I1_} GT 0);
          END;
        END;
   RUN;


    ********************************************************************;
    * STEP 2. generate process and progression of disease measure flags
    ********************************************************************;
   DATA OUT1.Y&YEAR.IPCLM (keep=hicno provider
               STUS_CD PER_DIEM UTIL_DAY
               FROM_DT THRU_DT PMT_AMT ippmtamt LOS
              PTNTSTUS ACC_CHRG ER_CHRG
         i_hgl
         i_lipid1 i_lip3_1-i_lip3_3
         i_self   i_educ1 i_educ2 i_eye1
         i_urine2 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
         i_led4   i_led5
   D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

         i_led6   i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind  i_catar    i_gla2 i_ede i_las
         i_stk_p  i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
         i_cab    i_ptc i_lea i_byp
         i_er_p   i_er_s
         i_db1-i_db5
   );                    ***lIST OF VARIABLES VARY SLIGHTLY FOR DIFF CLAIMS
TYPES;
     SET TEMP1
   ;
      attrib i_hgl     length=$1 label='Hgl A1c Test';
      attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
       attrib i_lip3_1  length=$1 label='Lipid Profiling definition 3: part
1';
       attrib i_lip3_2  length=$1 label='Lipid Profiling definition 3: part
2';
       attrib i_lip3_3  length=$1 label='Lipid Profiling definition 3: part
3';

      attrib i_self   length=$1 label='Self-Monitoring';
      attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
      attrib i_educ2  length=$1 label='Diabetes Education: definition 2';

      attrib i_eye1    length=$1 label='Eye Examination 1';
      attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
      attrib i_shoe    length=$1 label='Therapeutic Shoes';
```

```
      attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
      attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
      attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
      attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
      attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
      attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
      attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
      attrib i_led5  length=$1 label='diabetes w periph cir disorder';
      attrib i_led6  length=$1 label='Lower limb amputations';
      attrib i_led7  length=$1 label='Incision and drainage below facia';
      attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
      attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

      attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
      attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
      attrib i_neph    length=$1 label='Nephropathy';
      attrib i_ret1    length=$1 label='Retinopathy 1';

      attrib i_blind  length=$1 label='Blindness';
      attrib i_catar  length=$1 label='Diabetic cataract';
      attrib i_gla2   length=$1 label='Glaucoma 2';
      attrib i_ede    length=$1 label='Macular Edema';
      attrib i_las    length=$1 label='Laser treatment';

      attrib  i_stk_p    length=$1  label='Primary  DX:  Hospitalization  for
Stroke';
      attrib i_stk_s  length=$1 label='Secondary DX:Hospitaliza for Stroke';
      attrib i_ami_p  length=$1 label='Primary DX: Hospitalization for AMI';
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
      attrib  i_ami_s   length=$1  label='Secondary  DX:  Hospitalization  for
AMI';
      attrib  i_hrt_p    length=$1  label='Primary  DX:  Hospitali  for  Hrt
Failure';
      attrib i_hrt_s  length=$1 label='Secondary DX: Hospitali Hrt Failure';

      attrib i_cab  length=$1 label='Hospitalization for CABG';
      attrib i_ptc  length=$1 label='Progression to PTCA, including stents';

      attrib i_lea  length=$1 label='Hospitaliz for Lower extremity amputat';
      attrib   i_byp      length=$1   label='Hospitaliz   for   femoral-bypass
procedure';
      attrib i_er_p  length=$1 label='metabolic cris: ER or observ bed 1';
      attrib i_er_s  length=$1 label='metabolic cris: ER or observ bed 2';

      attrib i_db1  length=$1 label='Primary Hosp diabetes w/o complication';
      attrib i_db2  length=$1 label='Primary Hosp diabetes w/ ketoacidosis';
      attrib   i_db3      length=$1   label='Primary   Hosp   diabetes   w/
hyperosmolarit';
      attrib i_db4  length=$1 label='Primary Hosp diabetes w/ other coma';
      attrib  i_db5    length=$1  label='Primary  Hosp  diabetes  w/  other
complica';

      attrib hicno    length=$11 label='Health insurance claim number';
      attrib FROM_DT  length=8   label='Claim from date';
      attrib THRU_DT  length=8   label='Claim through date';
```

```
   attrib stus_cd  length=$2  label='Patient discharge status code';
   attrib provider length=$6  label='Provider number';

   attrib PTNTSTUS length=$1  label='Patient status indicator code';
   attrib ACC_CHRG length=8   label='Positive accommodation charge count';
   attrib ER_CHRG  length=8   label='Positive ER service charge count';

   attrib pmt_amt  length=8   label='Claim payment amount';
   attrib UTIL_DAY length=3   label='Claim utilization days count';
   attrib LOS      length=3   label='Length of stay';
   attrib PER_DIEM length=8   label='Claim pass through per diem amnt';
   attrib ippmtamt length=8   label='Inpatient claim payment amount';

   ippmtamt = sum(pmt_amt, (UTIL_DAY*PER_DIEM), 0);

   /* Array for input variables */
    ARRAY DGNS_CD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDR_CD{6}                          $    4 PRCDRCD1-PRCDRCD6;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;


   /* Length of stay */
   if FROM_DT not = . and THRU_DT not= . then LOS = THRU_DT - FROM_DT;
   IF LOS=0 THEN LOS=1;

   attrib tmp      length=$2  label='Temporary';

   /* Initialize indicators */
   array inds {*} $
      i_hgl
      i_lipid1 i_lip3_1-i_lip3_3
      i_self   i_educ1 i_educ2 i_eye1
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
      i_urine2 i_shoe

      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_stk_p i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab i_ptc i_lea i_byp
      i_er_p i_er_s
      i_db1-i_db5
;
   do _i=1 to dim(inds);
      inds{_i} = '0';
   end;

   ******** (1) For process of disease ************************;
   do i =1 to 45;
        /* CPT code for Hgl A1c Test:*/
         tmp = put(HCPCS_CD{i}, $hgl.);
         if tmp = '1' then i_hgl = '1';

        /* CPT code  for lipid profiling: */
```

```
      /* i_lipid1 */
      tmp = put(HCPCS_CD{i}, $lipid1f.);
      if tmp = '1' then i_lipid1 = '1';

      /* i_lipid3: 1-3 on the same day */
      tmp = put(HCPCS_CD{i}, $lipid3f.);
      if tmp = '1' then i_lip3_1 = '1';
      ELSE if tmp = '2' then i_lip3_2 = '1';
      ELSE if tmp = '3' then i_lip3_3 = '1';

      /* CPT: Self monitoring  */
      tmp = put(HCPCS_CD{i}, $monitor.);
      if tmp = '1' then i_self = '1';       /*physician*/
      ELSE if tmp = '2' then i_self = '1';  /*DME */
   end;


   /* diabetes education */
   /* educ1: CPT  */
   do i =1 to 45;
      /* i_educ11 */
      tmp = put(HCPCS_CD{i}, $educ1f.);
      if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/
   end;

   /* educ2: revenue code  */
   do i =1 to 45;
      /* i_educ11 */
      tmp = put(REV_CNTR{i}, $educ2f.);         /*FOR ALL YR*/
      if tmp = '1' then i_educ2 = '1';
   end;


   do i =1 to 45;
      /* Eye examination */
      /*Eye1: CPT */
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
      tmp = put(HCPCS_CD{i}, $eye1f.);
      if tmp = '1' then i_eye1 = '1';

      /* CPT: Microalbumin Screening /Quantitative Urine Protein */
      /* urine2:  */
      tmp = put(HCPCS_CD{i}, $urine2f.);
      if tmp = '1' then i_urine2 = '1';

      /* CPT: Therapeutic Shoes */
      tmp = put(HCPCS_CD{i}, $shoe.);
      if tmp = '1' then i_shoe = '1';
   end;


   ******** (2) For progression of disease ************************;

   /* Lower Extremity Diseases               */
   /* separate flags for principal or secondary DX for '1'*/
   do i =1 to 10;
      tmp = put(DGNS_CD{i}, $lower.);
```

```
            if i=1 then do;
              if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
            end;

            else do;
              if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
            end;
      end;

      /* Lower1: DX for category 2 to 5  */
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $lower1f.);
            if i=1 then do;
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;

            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;

      /* Lower2: CPT for category 6 to 9  */
      do i =1 to 45;
            tmp = put(HCPCS_CD{i}, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
            ELSE if tmp = '2' then i_led7 = '1';
                * incision and drainage of bone cortex No 8*;
            ELSE if tmp = '3' then i_led8 = '1';
                * skin and subcutaneous debridement for gangrene No 9*;
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
            ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;
      end;

      /* Lower3: ICD procedure code for category 6 to 9  */
         do i =1 to 6;
            tmp = put(PRCDR_CD{i}, $lower3f.);
               /* lower limb amputations No 6*/
            if tmp = '1' then i_led6 = '1';
                /* incision and drainage below facia No 7*/
            else if tmp = '2' then i_led7 = '1';
                /* incision and drainage of bone cortex No 8*/
            else if tmp = '3' then i_led8 = '1';
                /* skin and subcutaneous debridement for gangrene No 9*/
            else if tmp = '4' then i_led_9 = '1'; /*combine w DX 785.4*/
         end;

      ********* Final indicator for Low Extremety Disease *****;
```

```
   if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                        then i_led9 = '1';


/* New ESRD: this will be done in another SAS code using */
/* Base Denominator File based on variable Medicare Status Code--
   have ESRD in current year but no ESRD in previous year  */

/* DX: Neurological complications */
/* Autonomic Neuropathy    */
do i =1 to 10;
      tmp = put(DGNS_CD{i}, $neuro.);
      if i=1 then do;
        if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
      end;

      else do;
        if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
      end;
end;

/* DX: Nephropathy   */
do i =1 to 10;
      tmp = put(DGNS_CD{i}, $nephro.);
      if tmp = '2' then i_neph = '1';
 end;

/* Diabetic Eye Diseases   */
do i =1 to 10;
     /* No 1: retinopathy DX  */
      tmp = put(DGNS_CD{i}, $retin1f.);
      if tmp = '2' then i_ret1 = '1';

     /* No 2: blidness DX  */
      tmp = put(DGNS_CD{i}, $blind.);
      if tmp = '1' then i_blind = '1';

     /* No 3: Diabetic cataract DX   */
      tmp = put(DGNS_CD{i}, $catar.);
      if tmp = '1' then i_catar = '1';

      /* No 4:  Glaucoma DX  */
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

      tmp = put(DGNS_CD{i}, $glauco.);
      if tmp = '1' then i_gla2 = '1';          /*glauco2*/

     /* No 5:  Macular Edema DX  */
      tmp = put(DGNS_CD{i}, $edema.);
      if tmp = '1' then i_ede = '1';
end;


  /* No 6: Laser treatment CPT    */
  /* Laser CPT:   */
  do i =1 to 45;
      tmp = put(HCPCS_CD{i}, $laser1f.);
      if tmp = '1' then i_las = '1';
```

```
        end;

        /* Laser ICD9 treatment procedure:  */
         do i =1 to 6;
            tmp = put(PRCDR_CD{i}, $laser2f.);
            if tmp = '1' then i_las = '1';
         end;

      ***************** Hospitalizations *************************;
       /* ICD-9 diagnosis code for IP file*/
       /* Keep tract of primary and secondary*/
       do i =1 to 10;
            tmp = put(DGNS_CD{i}, $hosp1f.);

            if i=1 then do;
                /* Hospitalization for Stroke */
                if tmp = '1' then i_stk_p = '1';
                /* Hospitalization for Acute Myocardial Infarction*/
                ELSE if tmp = '2' then i_ami_p = '1';
                /* Hospitalization for Heart failure */
                ELSE if tmp = '3' then i_hrt_p = '1';
            end;

            else do;
                /* Hospitalization for Stroke */
                if tmp = '1' then i_stk_s = '1';
                /* Hospitalization for Acute Myocardial Infarction*/
                ELSE if tmp = '2' then i_ami_s = '1';
                /* Hospitalization for Heart failure */
                ELSE if tmp = '3' then i_hrt_s = '1';
            end;
       end;


       /* ICD-9 procedure code */
          do i =1 to 6;
            tmp = put(PRCDR_CD{i}, $hosp2f.);
                /* Hospitalization for CABG  */
            if tmp = '1' then i_cab = '1';
                /* Progresion to PTCA */
            ELSE    if    tmp    =    '2'    then    i_ptc    =    '1';
/*PTCA1*/
                /* Hospitalization for lower extremety amputation */
            ELSE if tmp = '3' then i_lea = '1';
                /* Hospitalization for a femoral-bypass procedure */
```

D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

```
            ELSE if tmp = '4' then i_byp = '1';
         end;


       /* PTCA2 : CPT code  */
         do i =1 to 45;
            tmp = put(HCPCS_CD{i}, $PTCA2F.);
            if tmp = '1' then i_ptc = '1';
         end;
```

```
   /* Admission to an Emergency Department or Observation Bed*/
   /* for Metabolic Crisis, for IP/OPD file */
   *** Diagnosis code for Metabolic Crisis;
   do i =1 to 10;
         tmp = put(DGNS_CD{i}, $er1dx.);
         if i=1 then do;
           if tmp = '1' then i_er_p1 = '1';   /* need RV code*/
         end;
         else do;
           if tmp = '1' then i_er_s1 = '1';
         end;
   end;
   *** Revenue code for Emergency Department or Observation Bed;
   do i =1 to 45;
         tmp = put(REV_CNTR{i}, $er1rv.);
         if tmp = '1' then i_er_2 = '1';
   end;
   ********* Final indicator for Metabolic Crisis *****;
   if i_er_p1 = '1' and i_er_2 = '1' then i_er_p ='1';
   if i_er_s1 = '1' and i_er_2 = '1' then i_er_s ='1';


   /* Hospitalizations with Diabetes as PRINCIPAL Diagnosis*/
   /* ICD-9 diagnosis code for IP file*/
   do i =1 to 1;                       *PRINCIPAL Diagnosis only;
         tmp = put(DGNS_CD{i}, $diab.);
             /* (1) Diabetes without mention of complications */
         if tmp = '1' then i_db1 = '1';
             /* (2) Diabetes with ketoacidosis*/
         ELSE if tmp = '2' then i_db2 = '1';
             /* (3) Diabetes with hyperosmolarity */
         ELSE if tmp = '3' then i_db3 = '1';
             /* (4) Diabetes with other coma */
         ELSE if tmp = '4' then i_db4 = '1';
             /* (5) Diabetes with other complications  */
         ELSE if tmp = '5' then i_db5 = '1';
   end;
RUN;

proc freq;
table i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5 i_led9 /missing;
title 'test i_led9';
run;


 ********************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
D#MSQ.MSQ00M7.PGMLIB99(C006IP) -- CONTINUED

 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.IPCLM DIRECTORY ;
TITLE "IP Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.IPCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR IP CLAIMS";
RUN;
```

```
ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006OP)

```
//WF1706O JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1 = REPLACE FILE NAME WITH FILE NAME OUTPUTTED FROM DESY
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/05/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN OUTPATIENT RAW CLAIMS DATA AND GENERATE FLAGS
//*            FOR PROCESS & PROGRESSION OF DISEASE MEASURES ON THE FLY
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C006OP.Y99OPCLM
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,200))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009820.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006OP.Y99OPCLM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ******************************************************************;
 * STEP 1. read in Outpatient Hospital claims
            note: this input statement is an old version. For the latest
version
          of files generated by DESY, need to add 50 spaces to each
variable.
          See 1992-1998 or 2000-2001 for example.
 ******************************************************************;

*******************************************************************;
*  FULLNAME:  FI OUTPATIENT CLAIMS RECORD                          ;
```

```
*  VERSION 'I' SAS COPYLIB STATEMENTS                                    ;
*  MEMBER NAME: UTLIOUTP                                                 ;
*  CREATED BY MARVELLA BUNCH 11/01/2000                                  ;
```

```
*  UPDATED BY MARVELLA BUNCH 03/01/2001                                  ;
************************************************************************;
DATA TEMP10 (keep=hicno
                DGNSCD01-DGNSCD10
                PRCDRCD1-PRCDRCD6
                RVCNTR01-RVCNTR45
                HCPSCD01-HCPSCD45
                STUS_CD
                FROM_DT THRU_DT PMT_AMT );
  INFILE IN1 STOPOVER;

    LENGTH REC_LEN                                4;
    LENGTH ACRTN_NM                               3;
    LENGTH LINK_NUM                               5;
    LENGTH SGMT_CNT                               2;
    LENGTH SGMT_NUM                               2;
    LENGTH LINECNT                                3;
    LENGTH SGMTLINE                               2;
    LENGTH OPEDCNT                                2;
    LENGTH OPPATCNT                               2;
    LENGTH OPMCOCNT                               2;
    LENGTH OPPLNCNT                               2;
    LENGTH OPDEMCNT                               2;
    LENGTH OPDGNCNT                               2;
    LENGTH OPPRCCNT                               2;
    LENGTH OPCONCNT                               2;
    LENGTH OPOCRCNT                               2;
    LENGTH OPSPNCNT                               2;
    LENGTH OPVALCNT                               2;
    LENGTH OPREVCNT                               2;
    LENGTH BLDFRNSH                               3;
    LENGTH BLD_RPLC                               3;
    LENGTH BLDNRPLC                               3;
    LENGTH BLDDEDPT                               3;
    ARRAY EDITIND{13}                          $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                          $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                         $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                          $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                          $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                            $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                            $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                           $    1 MCOOPTN1-MCOOPTN2;
    ARRAY MCOEFFDT{2}                          $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                          $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                          $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                          $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                          $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                            $   14 PLANID1-PLANID3;
    ARRAY DEMOIND{5}                           $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                           $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                           $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                          $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                          $    5 DGNSCD01-DGNSCD10;
```

```
    ARRAY PRCDRIND{6}                          $    1 PRCDRND1-PRCDRND6;
    ARRAY PRCDR_CD{6}                          $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                          $    8 PRCDRDT1-PRCDRDT6;
    ARRAY CONDIND{30}                          $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                         $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                         $    1 OCRCND01-OCRCND30;
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

    ARRAY OCRNC_CD{30}                         $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                          $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                          $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                          $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                         $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                         $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                           $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                           $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                               8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                           $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                           $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                         $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                         $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                         $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                         $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                         $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                         $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                         $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                         $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                         $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                         $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                          $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                         $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                         $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                           $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                           $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                           $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                           $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                              4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                              8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                              8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                              8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                               8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                              8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                              8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                              8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                              8 RVPCHG01-RVPCHG45;
    ARRAY RPRVDPMT{45}                              8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                              8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                              8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                                8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                              8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                              8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                         $    1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_OP_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                      PD3.
```

```
            @4  REC_LVL                     $CHAR1.
            @5  RIC_CD                      $CHAR1.
            @6  MQA_RIC                     $CHAR1.
            @7  CLM_TYPE                    $CHAR2.
         /*  FI_CLM_LINK_GRP  */
            /*  CLM_LCTR_NUM_GRP  */
            @9  HICNO                       $CHAR11.
                @9  CAN                         $CHAR9.
                @18  EQ_BIC                     $CHAR2.
```

D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

```
            @20  BIC                        $CHAR2.
            @22  ST_SGMT                    $CHAR1.
            @23  STATE_CD                   $CHAR2.

            @25  FROM_DT                    YYMMDD8.

            @33  THRU_DT                    YYMMDD8.
            @41  WKLY_DT                    $CHAR8.
            @49  ACRTN_DT                   $CHAR8.
            @57  ACRTN_NM                   PD2.
            @59  CLM_CNTL                   $CHAR23.
            @82  ORIGCNTL                   $CHAR23.
            @105  QUERY_CD                   $CHAR1.
            @106  PROVIDER                   $CHAR6.
            @112  DAILY_DT                   $CHAR8.
            @120  LINK_NUM                   PD5.
            @125  SGMT_CNT                   2.
            @127  SGMT_NUM                   2.
            @129  LINECNT                    3.
            @132  SGMTLINE                   2.
         /*  FI_CLM_CMN_I_GRP  */
            @134  PE_RIC                     $CHAR1.
            @135  TRANS_CD                   $CHAR1.
            /*  CLM_BILL_TYPE_CD_GRP  */
                @136  FAC_TYPE                   $CHAR1.
                @137  TYPESRVC                   $CHAR1.
            @138  FREQ_CD                    $CHAR1.
            @139  _F000001                   $CHAR1.
            @140  MQAQUERY                   $CHAR1.
            @141  DISP_CD                    $CHAR2.
            @143  EDITDISP                   $CHAR2.
            @145  BIC_MDFY                   $CHAR1.
            @146  CNTY_CD                    $CHAR3.
            @149  RCPT_DT                    $CHAR8.
            @157  SCHLD_DT                   $CHAR8.
            @165  FRWRD_DT                   $CHAR8.
            @173  FI_NUM                     $CHAR5.
            @178  ASGN_NUM                   $CHAR8.
            @186  FIBATCH                    $CHAR4.
            @190  BENE_ZIP                   $CHAR9.
            @199  SEX                        $CHAR1.
            @200  RACE                       $CHAR1.
            @201  BENE_DOB                   $CHAR8.
            @209  MS_CD                      $CHAR2.
            @211  SURNAME                    $CHAR6.
            @217  FRSTINIT                   $CHAR1.
            @218  MDL_INIT                   $CHAR1.
```

```
                @219  CWFLOCCD                    $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
                @242  ACTIONCD                   $CHAR1.
                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
                    @269  AT_NPI                     $CHAR10.
                    @279  AT_SRNM                    $CHAR6.
                    @285  AT_GVNNM                   $CHAR1.
                    @286  AT_MDL                     $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                    $CHAR6.
                    @293  OP_NPI                     $CHAR10.
                    @303  OP_SRNM                    $CHAR6.
                    @309  OP_GVN                     $CHAR1.
                    @310  OP_MDL                     $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @311  OT_UPIN                    $CHAR6.
                    @317  OT_NPI                     $CHAR10.
                    @327  OT_SRNM                    $CHAR6.
                    @333  OT_GVN                     $CHAR1.
                    @334  OT_MDL                     $CHAR1.
                @335  MDCD_PRV                   $CHAR13.
                @348  MDCDINFO                   $CHAR4.
                @352  MCOPDSW                    $CHAR1.
                @353  AUTHRZTN                   $CHAR18.
                @371  PTNTCNTL                   $CHAR20.
                @391  MDCL_REC                   $CHAR17.
                @408  PRO_CNTL                   $CHAR12.
                @420  PRO_DT                     $CHAR8.
                @428  STUS_CD                    $CHAR2.
                @430  DGNS_E                     $CHAR5.
                @435  _F000003                   $CHAR1.
                @436  PPS_IND                    $CHAR1.
                @437  TOT_CHRG                   PD6.2
                @443  _F000004                   $CHAR50.
            @493  OPEDCNT                    2.
            @495  OPPATCNT                   2.
            @497  OPMCOCNT                   1.
            @498  OPPLNCNT                   1.
            @499  OPDEMCNT                   1.
            @500  OPDGNCNT                   2.
            @502  OPPRCCNT                   2.
            @504  OPCONCNT                   2.
            @506  OPOCRCNT                   2.
            @508  OPSPNCNT                   2.
            @510  OPVALCNT                   2.
```

```
            @512  OPREVCNT                    2.
            @514  _F000005                    $CHAR4.
            /*  FI_OP_CLM_SPECF_I_GRP  */
                @518  OPSRVTYP                    $CHAR1.
                @519  OP_RFRL                     $CHAR1.
                @520  BLDDEDAM                    PD6.2
                @526  PTB_DED                     PD6.2
                @532  PTB_COIN                    PD6.2
                @538  PCCHGAMT                    PD6.2
                @544  INTRMDED                    PD6.2
                @550  PRVDRPMT                    PD6.2
                @556  BENEPMT                     PD6.2
                @562  BLDFRNSH                    PD2.
                @564  BLD_RPLC                    PD2.
                @566  BLDNRPLC                    PD2.
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

                @568  BLDDEDPT                    PD2.
                @570  TRANTYPE                    $CHAR1.
                @571  ESRDMTHD                    $CHAR1.
                @572  _F000006                    $CHAR24.
        /*  FI_OP_CLM_TRLR_I_GRP  */
    @;
    DO _I1_ = 1 TO OPEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}             $CHAR1.
                EDIT_CD{_I1_}             $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO OPPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}            $CHAR1.
                PATCHCD{_I1_}             $CHAR2.
                PATCHDT{_I1_}             $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}              $CHAR1.
                MCONUM{_I1_}              $CHAR5.
                MCOOPTN{_I1_}             $CHAR1.
                MCOEFFDT{_I1_}            $CHAR8.
                MCOTRMDT{_I1_}            $CHAR8.
                MCOPLNID{_I1_}            $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO OPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}            $CHAR1.
                PLANIDCD{_I1_}            $CHAR1.
                PLANID{_I1_}              $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO OPDEMCNT;
```

```
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}               $CHAR1.
                DEMONUM{_I1_}               $CHAR2.
                DEMOTXT{_I1_}               $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO OPDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
                DGNSIND{_I1_}               $CHAR1.
                DGNS_CD{_I1_}               $CHAR5.
                _F000007                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO OPPRCCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
```



```
                PRCDRIND{_I1_}              $CHAR1.
                PRCDR_CD{_I1_}              $CHAR4.
                _F000008                    $CHAR3.
                PRCDR_DT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
                CONDIND{_I1_}               $CHAR1.
                RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO OPOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
                OCRNCIND{_I1_}              $CHAR1.
                OCRNC_CD{_I1_}              $CHAR2.
                OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
                SPANIND{_I1_}               $CHAR1.
                SPAN_CD{_I1_}               $CHAR2.
                SPANFROM{_I1_}              $CHAR8.
                SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO OPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
                VALIND{_I1_}                $CHAR1.
                VAL_CD{_I1_}                $CHAR2.
                VAL_AMT{_I1_}               PD6.2
        @;
    END;
```

```
    DO _I1_ = 1 TO OPREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
                REVIND{_I1_}                $CHAR1.
                REV_CNTR{_I1_}              $CHAR4.
                REV_DT{_I1_}                $CHAR8.
                REVANSI1{_I1_}              $CHAR5.
                REVANSI2{_I1_}              $CHAR5.
                REVANSI3{_I1_}              $CHAR5.
                REVANSI4{_I1_}              $CHAR5.
                APCHIPPS{_I1_}              $CHAR5.
                HCPCS_CD{_I1_}              $CHAR5.
                MDFR_CD1{_I1_}              $CHAR2.
                MDFR_CD2{_I1_}              $CHAR2.
                MDFR_CD3{_I1_}              $CHAR2.
                MDFR_CD4{_I1_}              $CHAR2.
                MDFR_CD5{_I1_}              $CHAR2.
                PMTMTHD{_I1_}               $CHAR2.
                DSCNTIND{_I1_}              $CHAR1.
                PACKGIND{_I1_}              $CHAR1.
                PRICNG{_I1_}                $CHAR2.
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

                OTAF_1{_I1_}                $CHAR1.
                OTAF_2{_I1_}                $CHAR1.
                IDENDC{_I1_}                $CHAR24.
                REV_UNIT{_I1_}              PD4.
                REV_RATE{_I1_}              PD6.2
                REVBLOOD{_I1_}              PD6.2
                REVDCTBL{_I1_}              PD6.2
                WAGEADJ{_I1_}               PD6.2
                RDCDCOIN{_I1_}              PD6.2
                REV_MSP1{_I1_}              PD6.2
                REV_MSP2{_I1_}              PD6.2
                REVPCCHG{_I1_}              PD6.2
                RPRVDPMT{_I1_}              PD6.2
                RBENEPMT{_I1_}              PD6.2
                PTNTRESP{_I1_}              PD6.2
                REVPMT{_I1_}                PD6.2
                REV_CHRG{_I1_}              PD6.2
                REV_NCVR{_I1_}              PD6.2
                REVDEDCD{_I1_}              $CHAR1.
                _F000009                    $CHAR50.
        @;
    END;
    INPUT
        EOR                         $CHAR3.
    @;
    /* Release trailing @ */
    input;

    DROP _I1_;
    DROP _F000001-_F000009;
run;


 ********************************************************************;
 * STEP 2. generate process and progression of disease measure flags
```

```
 ********************************************************************;
DATA OUT1.Y&YEAR.OPCLM (keep=hicno
                STUS_CD
                FROM_DT THRU_DT PMT_AMT
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar i_gla2 i_ede i_las
      i_ptc    i_er_p i_er_s
);
  SET TEMP10;

   attrib i_hgl     length=$1 label='Hgl A1c Test';
   attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
   attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';

   attrib i_self  length=$1 label='Self-Monitoring';
   attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
   attrib i_educ2  length=$1 label='Diabetes Education: definition 2';
```



```
   attrib i_eye1   length=$1 label='Eye Examination 1';

   attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
   attrib i_shoe    length=$1 label='Therapeutic Shoes';

   attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
   attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
   attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
   attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
   attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
   attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
   attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
   attrib i_led5  length=$1 label='diabetes w periph cir dis';
   attrib i_led6  length=$1 label='Lower limb amputations';
   attrib i_led7  length=$1 label='Incision and drainage below facia';
   attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
   attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

   attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
   attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
   attrib i_neph    length=$1 label='Nephropathy';
   attrib i_ret1    length=$1 label='Retinopathy 1';
   attrib i_blind  length=$1 label='Blindness';
   attrib i_catar  length=$1 label='Diabetic cataract';

   attrib i_gla2   length=$1 label='Glaucoma 2';
   attrib i_ede    length=$1 label='Macular Edema';
   attrib i_las    length=$1 label='Laser treatment';

   attrib i_ptc  length=$1 label='Progression to PTCA, including stents';
```

```
   attrib i_er_p  length=$1 label='metabolic cris: ER or observ bed 1';
   attrib i_er_s  length=$1 label='metabolic cris: ER or observ bed 2';

   attrib hicno    length=$11 label='Health insurance claim number';
   attrib from_dt  length=8   label='Claim from date';
   attrib thru_dt  length=8   label='Claim through date';
   attrib stus_cd  length=$2  label='Patient discharge status code';
   attrib pmt_amt  length=8   label='Claim payment amount';

   /* Array for input variables */
    ARRAY DGNS_CD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDR_CD{6}                          $    4 PRCDRCD1-PRCDRCD6;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;

   attrib tmp      length=$2  label='Temporary';

   /* Initialize indicators */
   array inds {*} $
      i_hgl
      i_lipid1 i_lip3_1-i_lip3_3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe

      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar    i_gla2 i_ede i_las
      i_ptc    i_er_p i_er_s
;
   do _i=1 to dim(inds);
      inds{_i} = '0';
   end;

   ******** (1) For process of disease ************************;

   do i =1 to 45;
         /* CPT code for Hgl A1c Test*/
         tmp = put(HCPCS_CD{i}, $hgl.);
         if tmp = '1' then i_hgl = '1';

         /* CPT code  for lipid profiling*/
         /* i_lipid1 */
         tmp = put(HCPCS_CD{i}, $lipid1f.);
         if tmp = '1' then i_lipid1 = '1';

         /* i_lipid3: 1-3 on the same day */
         tmp = put(HCPCS_CD{i}, $lipid3f.);
         if tmp = '1' then i_lip3_1 = '1';
         ELSE if tmp = '2' then i_lip3_2 = '1';
         ELSE if tmp = '3' then i_lip3_3 = '1';

         /* CPT: Self monitoring  */
         tmp = put(HCPCS_CD{i}, $monitor.);
         if tmp = '1' then i_self = '1';
```

```
      ELSE if tmp = '2' then i_self = '1';
   end;
   ********* create final indicator for i_lipid3 ***************;
   IF (i_lip3_1='1' AND i_lip3_2='1' AND i_lip3_3='1' )
                       THEN i_lipid3='1';  else i_lipid3='0';

   /* diabetes education */
   /* educ1: CPT  */
   do i =1 to 45;
      /* i_educ11 */
      tmp = put(HCPCS_CD{i}, $educ1f.);
      if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/
   end;

   /* educ2: revenue code */
   do i =1 to 45;
      /* i_educ11 */
      tmp = put(REV_CNTR{i}, $educ2f.);         /*FOR ALL YR*/
      if tmp = '1' then i_educ2 = '1';
   end;

   do i =1 to 45;
      /* Eye examination: CPT */
      tmp = put(HCPCS_CD{i}, $eye1f.);
      if tmp = '1' then i_eye1 = '1';

      /* CPT: Microalbumin Screening /Quantitative Urine Protein*/
      /* urine2:  */
      tmp = put(HCPCS_CD{i}, $urine2f.);
      if tmp = '1' then i_urine2 = '1';
```

D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

```
      /* CPT: Therapeutic Shoes*/
      tmp = put(HCPCS_CD{i}, $shoe.);
      if tmp = '1' then i_shoe = '1';
   end;


   ******** (2) For progression of disease ************************;

   /* Lower Extremity Diseases               */
   /* separate flags for principal or secondary DX for '1'*/
   do i =1 to 10;
      tmp = put(DGNS_CD{i}, $lower.);
      if i=1 then do;
        if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
      end;

      else do;
        if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
      end;
   end;

   /* Lower1: DX for category 2 to 5  */
   do i =1 to 10;
      tmp = put(DGNS_CD{i}, $lower1f.);
      if i=1 then do;
```

```
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;

            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;


      /* Lower2: CPT for category 6 to 9  */
      do i =1 to 45;
            tmp = put(HCPCS_CD{i}, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
            ELSE if tmp = '2' then i_led7 = '1';
                * incision and drainage of bone cortex No 8*;
            ELSE if tmp = '3' then i_led8 = '1';
                * skin and subcutaneous debridement for gangrene No 9*;
            ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;
      end;


      /* Lower3: ICD procedure code for category 6 to 9  */
         do i =1 to 6;
   D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

            tmp = put(PRCDR_CD{i}, $lower3f.);
               /* lower limb amputations No 6*/
            if tmp = '1' then i_led6 = '1';
                /* incision and drainage below facia No 7*/
            else if tmp = '2' then i_led7 = '1';
                /* incision and drainage of bone cortex No 8*/
            else if tmp = '3' then i_led8 = '1';
                /* skin and subcutaneous debridement for gangrene No 9*/
            else if tmp = '4' then i_led_9 = '1'; /*combine w DX 785.4*/
         end;

      ********* Final indicator for Low Extremety Disease *****;
         if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                              then i_led9 = '1';

      /* New ESRD: this will be done in another SAS code using */
      /* Base Denominator File based on variable Medicare Status Code--
         have ESRD in current year but no ESRD in previous year  */


      /* DX: Neurological complications */
      /* Autonomic Neuropathy */
      do i =1 to 10;
```

```
        tmp = put(DGNS_CD{i}, $neuro.);
        if i=1 then do;
          if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
        end;

        else do;
          if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
        end;
    end;


    /* DX: Nephropathy   */
    do i =1 to 10;
        tmp = put(DGNS_CD{i}, $nephro.);
        if tmp = '2' then i_neph = '1';
     end;

    /* Diabetic Eye Diseases   */
    do i =1 to 10;
       /* No 1: retinopathy DX  */
        tmp = put(DGNS_CD{i}, $retin1f.);
        if tmp = '2' then i_ret1 = '1';


       /* No 2: blidness DX  */
        tmp = put(DGNS_CD{i}, $blind.);
        if tmp = '1' then i_blind = '1';

       /* No 3: Diabetic cataract DX   */
        tmp = put(DGNS_CD{i}, $catar.);
        if tmp = '1' then i_catar = '1';

       /* No 4:  Glaucoma DX  */
        tmp = put(DGNS_CD{i}, $glauco.);
        if tmp = '1' then i_gla2 = '1';          /*glauco2*/
```

D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

```
       /* No 5:  Macular Edema DX  */
        tmp = put(DGNS_CD{i}, $edema.);
        if tmp = '1' then i_ede = '1';
    end;


      /* No 6: Laser treatment CPT    */
      /* Laser CPT: for physician  */
      do i =1 to 45;
          tmp = put(HCPCS_CD{i}, $laser1f.);
          if tmp = '1' then i_las = '1';
      end;

      /* Laser treatment ICD9 procedure  */
       do i =1 to 6;
          tmp = put(PRCDR_CD{i}, $laser2f.);
          if tmp = '1' then i_las = '1';
       end;

    /* ICD-9 procedure code for Progression to PTCA */
```

```
      do i =1 to 6;
         tmp = put(PRCDR_CD{i}, $hosp2f.);
         if tmp = '2' then i_ptc = '1';                          /*PTCA1*/
      end;

   /* PTCA2: CPT code for Progression to PTCA */
     do i =1 to 45;
         tmp = put(HCPCS_CD{i}, $PTCA2F.);
         if tmp = '1' then i_ptc = '1';
     end;

   /* Admission to an Emergency Department or Observation Bed*/
   /* for Metabolic Crisis, for IP/OPD file */
   *** Diagnosis code for Metabolic Crisis;
   do i =1 to 10;
         tmp = put(DGNS_CD{i}, $er1dx.);
         if i=1 then do;
           if tmp = '1' then i_er_p1 = '1';   /* need RV code*/
         end;
         else do;
           if tmp = '1' then i_er_s1 = '1';
         end;
   end;
   *** Revenue code for Emergency Department or Observation Bed;
   do i =1 to 45;
         tmp = put(REV_CNTR{i}, $er1rv.);
         if tmp = '1' then i_er_2 = '1';   /* need DX code $er1dx*/
   end;
   ********* Final indicator for Metabolic Crisis *****;
   if i_er_p1 = '1' and i_er_2 = '1' then i_er_p ='1';
   if i_er_s1 = '1' and i_er_2 = '1' then i_er_s ='1';

RUN;


 ********************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.OPCLM DIRECTORY ;
D#MSQ.MSQ00M7.PGMLIB99(C006OP) -- CONTINUED

TITLE "OP Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.OPCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR OP CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006PS)

```
//WF1706P JOB (@MSQ0000,WF17),HONG,CLASS=H,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1 = REPLACE FILE NAME WITH FILE NAME OUTPUTTED FROM DESY
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//********IF OLD OUTPUT DATASET OUT1 EXIST, DELETE THE OLD CATALOGED
//*       DATASETS THROUGH TSO OR RENAME IT BEFORE RERUN THIS CODE
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/06/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN PHYSICIAN RAW CLAIMS DATA AND GENERATE FLAGS FOR
//*            PROCESS AND PROGRESSION OF DISEASE MEASURES ON THE FLY
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C006PS.Y99PSCLM
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009816.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006PS.Y99PSCLM,
//       UNIT=(MCART,,DEFER),LABEL=(,,,RETPD=700),
//       DISP=(NEW,CATLG,DELETE)
//WORK10 DD DSN=WF17.@MSQ0000.SAS2.TEMP10,
//       UNIT=(MCART,,DEFER),
//       DISP=(NEW,DELETE,DELETE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ********************************************************************;
 * STEP 1. read in Part B Carrier (Physician) claims
            note: this input statement is an old version. For the latest
version
          of files generated by DESY, need to add 50 spaces to each
variable.
          See 1992-1998 or 2000-2001 for example.
```

```
 ********************************************************************;

*******************************************************************;
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

***MEMBER  SIPBVB   CREATED 07/7/2000   UPDATED 02/08/2001        ;
***SAS VERSION I INPUT STATEMENT FOR CARRIER PART B (RIC O) CLAIMS;
***THIS VERSION OUTPUTS 1 OBSERVATION FOR EACH LINE ITEM          ;
***FOR EFFICIENCY:                                               ;
***  KEEP STATEMENT SHOULD BE MODIFIED TO SPECIFY ONLY           ;
***  DATA ELEMENTS REQUIRED FOR OUTPUT FILE                      ;
*******************************************************************;

DATA WORK10.TEMP10 (keep=HICNO
               DGNSCD01-DGNSCD04
               LINEDGNS
               HCPSCD01
               FROM_DT THRU_DT HCFASPCL LINEPMT);

  INFILE IN1 MISSOVER           ;
    *LENGTH REC_LEN                              4;
    *LENGTH ACRTN_NM                             3;
    *LENGTH BLDFRNSH                             3;
    *LENGTH BLD_DED                              3;
    LENGTH CEDCNT                               2;
    LENGTH CPATCNT                              2;
    LENGTH CMCOCNT                              2;
    LENGTH CPLNCNT                              2;
    LENGTH CDEMCNT                              2;
    LENGTH CDGNCNT                              2;
    LENGTH CLINECNT                             2;
 ***ADD KEEP STATEMENT HERE...;
*-------------------------------------------------------------------*
|    INPUT NEAR-LINE SEQ CWFB RECORDS, FIXED & LINE-ITEMS, VER 'I'    |
*-------------------------------------------------------------------;
    INPUT
        /*  CARR_CLM_FIX_I_GRP  */
            /*  CLM_REC_IDENT_GRP  */
                /*
                @1  REC_LEN                    PD3.
                @4  REC_LVL                    $CHAR1.
                @5  RIC_CD                     $CHAR1.
                @6  MQA_RIC                    $CHAR1.
                @7  CLM_TYPE                   $CHAR2.
                */
            /*  CARR_DMERC_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                @9  HICNO                       $CHAR11.

                /*
                @20  BIC                        $CHAR2.
                @22  ST_SGMT                    $CHAR1.
                @23  STATE_CD                   $CHAR2.
                */
                @25  FROM_DT                    YYMMDD8.
                @33  THRU_DT                    YYMMDD8.
                /*
                @41  WKLY_DT                    $CHAR8.
```

```
                @49  ACRTN_DT                    $CHAR8.
                @57  ACRTN_NM                    PD2.
                @59  CARRCNTL                    $CHAR15.
                @74  _F000001                    $CHAR38.
                @112  DAILY_DT                    $CHAR8.
                @120  LINK_NUM                    PD5.
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

                @125  SGMT_CNT                    2.
                @127  SGMT_NUM                    2.
                @129  LINECNT                     3.
                @132  SGMTLINE                    2.
                */
             /*  CARR_DMERC_CLM_CMN_1_I_GRP  */
                /*
                @134  _F000002                    $CHAR5.
                @139  ENTRY_CD                    $CHAR1.
                @140  _F000003                    $CHAR1.
                @141  DISP_CD                     $CHAR2.
                @143  EDITDISP                    $CHAR2.
                @145  BIC_MDFY                    $CHAR1.
                @146  CNTY_CD                     $CHAR3.
                @149  RCPT_DT                     $CHAR8.
                @157  SCHLD_DT                    $CHAR8.
                @165  FRWRD_DT                    $CHAR8.
                @173  CARR_NUM                    $CHAR5.
                @178  _F000004                    $CHAR8.
                @186  FIBATCH                     $CHAR4.
                @190  BENE_ZIP                    $CHAR9.
                @199  SEX                         $CHAR1.
                @200  RACE                        $CHAR1.
                @201  BENE_DOB                    $CHAR8.
                @209  MS_CD                       $CHAR2.
                @211  SURNAME                     $CHAR6.
                @217  FRSTINIT                    $CHAR1.
                @218  MDL_INIT                    $CHAR1.
                @219  CWFLOCCD                    $CHAR1.
                @220  PDGNS_CD                    $CHAR5.
                @225  _F000005                    $CHAR1.
                @226  PMTDNLCD                    $CHAR1.
                @227  TRTMT_CD                    $CHAR1.
                */
                @228  PMT_AMT                     PD6.2
                /*
                @234  PRPAYAMT                    PD6.2
                @240  _F000006                    $CHAR1.
                @241  RFR_UPIN                    $CHAR6.
                @247  RFR_NPI                     $CHAR10.
                @257  ASGMNTCD                    $CHAR1.
                @258  PROV_PMT                    PD6.2
                @264  BENE_PMT                    PD6.2
                @270  BENEPAID                    PD6.2
                @276  SBMTCHRG                    PD6.2
                @282  ALOWCHRG                    PD6.2
                @288  DEDAPPLY                    PD6.2
                @294  HCPCS_YR                    1.
                @295  MCOOVRRD                    $CHAR1.
                @296  HOSPOVRD                    $CHAR1.
```

```
                @297  _F000007                    $CHAR31.
                */
            /*  CARR_SPECFC_GRP  */
                /*
                @328  RFR_PRFL                    $CHAR14.
                @342  CPO_PROV                    $CHAR6.
                @348  CPO_NPI                     $CHAR10.
                @358  BLDFRNSH                    PD2.
                @360  BLD_DED                     PD2.
```

D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

```
                */
            @362  CEDCNT                      2.
            @364  CPATCNT                     2.
            @366  CMCOCNT                     1.
            @367  CPLNCNT                     1.
            @368  CDEMCNT                     1.
            @369  CDGNCNT                     1.
            @370  CLINECNT                    2.
            @372  _F000008                    $CHAR4.
        /*  CARR_CLM_VAR_I_GRP  */
    @;
  IF CEDCNT GT 0 THEN DO;

  ARRAY EDITIND{*} $ 1 EDTIND01-EDTIND13;
  ARRAY EDIT_CD{*} $ 4 EDT_CD01-EDT_CD13;
   DO _I_=1 TO CEDCNT;
     INPUT
        EDITIND{_I_} $CHAR1.
        EDIT_CD{_I_} $CHAR4.
        @;
   END;
  END;
  IF CPATCNT GT 0 THEN DO;
  ARRAY PATCHIND{*} $ 1 PTCHND01-PTCHND99;
  ARRAY PATCHCD{*}  $ 2 PTCHCD01-PTCHCD99;
  ARRAY PATCHDT{*}  $ 8 PTCHDT01-PTCHDT99;
   DO _I_ = 1 TO CPATCNT;
     INPUT
        PATCHIND{_I_} $CHAR1.
        PATCHCD {_I_} $CHAR2.
        PATCHDT{_I_}  $CHAR8.
        @;
   END;
  END;
  IF CMCOCNT GT 0 THEN DO;
  ARRAY MCOIND{*}   $ 1   MCOIND1-MCOIND2;
  ARRAY MCONUM{*}   $ 5   MCONUM1-MCONUM2;
  ARRAY MCOOPTN{*}  $ 1   MCOOPTN1-MCOOPTN2;
  ARRAY MCOEFFDT{*} $ 8   MCEFFDT1-MCEFFDT2;
  ARRAY MCOTRMDT{*} $ 8   MCTRMDT1-MCTRMDT2;
  ARRAY MCOPLNID{*} $ 14   MCPLNID1-MCPLNID2;
   DO _I_ = 1 TO CMCOCNT;
     INPUT
        MCOIND{_I_}    $CHAR1.
        MCONUM{_I_}    $CHAR5.
        MCOOPTN{_I_}   $CHAR1.
        MCOEFFDT{_I_}  $CHAR8.
```

```
        MCOTRMDT{_I_}  $CHAR8.
        MCOPLNID{_I_}  $CHAR14.
        @;
     END;
    END;
    IF CPLNCNT GT 0 THEN DO;
    ARRAY PLANIDIN {*} $ 1 PLNDNID1-PLNDNID3;
    ARRAY PLANIDCD{*}  $ 1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{*} $ 14 PLANID1-PLANID3;
      DO _I1_ = 1 TO CPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

                PLANIDIN {_I1_}            $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
      END;
    END;
    IF CDEMCNT GT 0 THEN DO;
    ARRAY DEMOIND{*} $ 1  DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{*} $ 2  DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{*} $ 15 DEMOTXT1-DEMOTXT5;
     DO _I_ = 1 TO CDEMCNT;
       INPUT
        DEMOIND{_I_} $CHAR1.
        DEMONUM{_I_} $CHAR2.
        DEMOTXT{_I_} $CHAR15.
        @;
     END;
    END;
    IF CDGNCNT GT 0 THEN DO;
    ARRAY DGNSIND{*} $ 1 DGNSIND1-DGNSIND4;
    ARRAY DGNS_CD{*} $ 5 DGNSCD01-DGNSCD04;
     DO _I_ = 1 TO CDGNCNT;
       INPUT DGNSIND{_I_} $CHAR1.
             DGNS_CD{_I_} $CHAR5.
             _F000009      $CHAR1.
             @;
     END;
    END;
    DO _I_ = 1 TO CLINECNT;
       INPUT
        LINEIND   $CHAR1.
        PRF_PRFL  $CHAR10.
        PRF_UPIN  $CHAR6.
        PRF_NPI   $CHAR10.
        PRGRPNPI  $CHAR10.
        PRV_TYPE  $CHAR1.
        TAX_NUM   $CHAR10.
        PRVSTATE  $CHAR2.
        PROVZIP   $CHAR9.
        HCFASPCL  $CHAR2.
        CARRSPCL  $CHAR2.
        PRTCPTG   $CHAR1.
        ASTNT_CD  $CHAR1.
        SRVC_CNT  PD2.
```

```
        TYPSRVCB  $CHAR1.
        PTYPESRV  $CHAR2.
        PLCSRVC   $CHAR2.
        LCLTY_CD  $CHAR2.
        EXPNSDT1  $CHAR8.
        EXPNSDT2  $CHAR8.
        HCPSCD01  $CHAR5.
        MDFR_CD1  $CHAR2.
        MDFR_CD2  $CHAR2.
        BETOS     $CHAR3.
        LINE_IDE  $CHAR7.
        NDC_CD    $CHAR11.
        LINEPMT   PD6.2
        LBENPMT   PD6.2
        LPRVPMT   PD6.2
```
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED
```
        LDEDAMT   PD6.2
        LPRPAYCD  $CHAR1.
        LPRPDAMT  PD6.2
        COINAMT   PD6.2
        LLMTAMT   PD6.2
        LINT_AMT  PD6.2
        PRPYALOW  PD6.2
        PNLTYAMT  PD6.2
        LBLD_DED  PD2.
        LSBMTCHG  PD6.2
        LALOWCHG  PD6.2
        LAB_NUM   $CHAR10.
        LAB_AMT   PD6.2
        PRCNGIND  $CHAR1.
        PMTINDSW  $CHAR1.
        DED_SW    $CHAR1.
        PMTINDCD  $CHAR1.
        MTUS_CNT  PD2.
        MTUS_IND  $CHAR1.
        LINEDGNS  $CHAR5.
        _F000010  $CHAR1.
        ANSTHUNT  PD2.
        CLIAALRT  $CHAR1.
        DCMTN_CD  $CHAR1.
        DMEST_DT  $CHAR8.
        DME_PURC  PD6.2
        NCSTY_MO  PD2.
        _F000011  $CHAR50.
        @;
 **OUTPUT EACH LINE ITEM AS AN OBSERVATION;
        OUTPUT;
  END;
  INPUT
    EOR    $CHAR3. ;
  DROP _F000001-_F000011  _I_ ;
run;


 ******************************************************************;
 * STEP 2. generate process and progression of disease measure flags
 ******************************************************************;
```

```
    DATA OUT1.Y&YEAR.PSCLM (keep=HICNO FROM_DT THRU_DT HCFASPCL LINEPMT
          i_hgl
          i_lipid1 i_lip3_1-i_lip3_3
          i_self   i_educ1          i_eye1
          i_urine2 i_foot1 i_shoe

          i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
          i_led4   i_led5
          i_led6   i_led7 i_led8 i_led9
          i_neur_p i_neur_s  i_neph i_ret1
          i_blind  i_catar i_gla2 i_ede i_las
          i_ptc
    );
      SET WORK10.TEMP10;

       attrib i_hgl     length=$1 label='Hgl A1c Test';
       attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
```
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED
```
        attrib i_lip3_1   length=$1 label='Lipid Profiling definition 3: part
1';
        attrib i_lip3_2   length=$1 label='Lipid Profiling definition 3: part
2';
        attrib i_lip3_3   length=$1 label='Lipid Profiling definition 3: part
3';

       attrib i_self  length=$1 label='Self-Monitoring';
       attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
       attrib i_eye1   length=$1 label='Eye Examination 1';

       attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
       attrib i_foot1  length=$1 label='Podiatry visits 1';
       attrib i_shoe    length=$1 label='Therapeutic Shoes';

       attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
       attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
       attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
       attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
       attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
       attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
       attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
       attrib i_led5  length=$1 label='diabetes w periph cir dis';
       attrib i_led6  length=$1 label='Lower limb amputations';
       attrib i_led7  length=$1 label='Incision and drainage below facia';
       attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
       attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

       attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
       attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
       attrib i_neph    length=$1 label='Nephropathy';
       attrib i_ret1    length=$1 label='Retinopathy 1';
       attrib i_blind  length=$1 label='Blindness';
       attrib i_catar  length=$1 label='Diabetic cataract';

       attrib i_gla2   length=$1 label='Glaucoma 2';
       attrib i_ede    length=$1 label='Macular Edema';
```

```
      attrib i_las    length=$1 label='Laser treatment';
      attrib  i_ptc        length=$1  label='Progression  to  PTCA,  including
stents';

      attrib hicno    length=$11 label='Health insurance claim number';
      attrib from_dt  length=8   label='Claim from date';
      attrib thru_dt  length=8   label='Claim through date';
      attrib HCFASPCL length=$2  label='Line item HCFA specialty code';
      attrib linepmt  length=8   label='Line item payment amount';

      /* Array for input variables: LINEDGNS is primary DX  */
       ARRAY DGNS_CD{5}                                $ 5 LINEDGNS DGNSCD01-
DGNSCD04;


      attrib tmp      length=$2  label='Temporary';

      /* Initialize indicators */
      array inds {*} $
         i_hgl
         i_lipid1 i_lip3_1-i_lip3_3
   D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

         i_self   i_educ1  i_eye1
         i_urine2 i_foot1  i_shoe

         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
         i_led4   i_led5
         i_led6   i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc
   ;
      do _i=1 to dim(inds);
         inds{_i} = '0';
      end;

      ******** (1) For process of disease ************************;

      /* CPT code for Hgl A1c Test*/
            tmp = put(HCPSCD01, $hgl.);
            if tmp = '1' then i_hgl = '1';


      /* CPT code  for lipid profiling*/
            /* i_lipid1 */
            tmp = put(HCPSCD01, $lipid1f.);
            if tmp = '1' then i_lipid1 = '1';

            /* i_lipid3: 1-3 on the same day */
            tmp = put(HCPSCD01, $lipid3f.);
            if tmp = '1' then i_lip3_1 = '1';
            ELSE if tmp = '2' then i_lip3_2 = '1';
            ELSE if tmp = '3' then i_lip3_3 = '1';


      /* CPT: Self monitoring  */
            tmp = put(HCPSCD01, $monitor.);
```

```
            if tmp = '1' then i_self = '1';
            ELSE if tmp = '2' then i_self = '1';


      /* diabetes education */
      /* educ1: CPT */
            tmp = put(HCPSCD01, $educ1f.);
            if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/

      /* Eye examination */
         /*Eye1: CPT */
            tmp = put(HCPSCD01, $eye1f.);
            if tmp = '1' then i_eye1 = '1';       *don't need specialty code;
            ELSE   if   tmp   =   '2'   then   i_eye1_2   =   '1';*w   provider
specialty'18','41;
            if HCFASPCL in ('18','41') and i_eye1_2 = '1' then i_eye1='1';


      /* CPT: Microalbumin Screening /Quantitative Urine Protein */
            /* urine2:  */
            tmp = put(HCPSCD01, $urine2f.);
            if tmp = '1' then i_urine2 = '1';

      /* DX for foot1: Podiatry visits */
   D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $foot.);
            if tmp = '1' and HCFASPCL ='48' then i_foot1 = '1';
      end;

      /* CPT: Therapeutic Shoes */
            tmp = put(HCPSCD01, $shoe.);
            if tmp = '1' then i_shoe = '1';


      ******** (2) For progression of disease ************************;

      /* Lower Extremity Diseases               */
      /* separate flags for principal or secondary DX for '1'*/
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $lower.);
            if i=1 then do;
              if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
            end;

            else do;
              if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
            end;
      end;

      /* Lower1: DX for category 2 to 5  */
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $lower1f.);
            if i=1 then do;
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
```

```
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;

            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;


      /* Lower2: CPT for category 6 to 9 for Physician only? */
            tmp = put(HCPSCD01, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
            ELSE if tmp = '2' then i_led7 = '1';
                * incision and drainage of bone cortex No 8*;
            ELSE if tmp = '3' then i_led8 = '1';
                * skin and subcutaneous debridement for gangrene No 9*;
            ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;


      ********* Final indicator for Low Extremety Disease *****;
```
D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED
```
         if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                              then i_led9 = '1';


      /* DX: Neurological complications */
      /* Autonomic Neuropathy */
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $neuro.);
            if i=1 then do;
              if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
            end;

            else do;
              if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
            end;
      end;


      /* DX: Nephropathy   */
      do i =1 to 5;
            tmp = put(DGNS_CD{i}, $nephro.);
            if tmp = '2' then i_neph = '1';
       end;

      /* Diabetic Eye Diseases   */
      do i =1 to 5;
           /* No 1: retinopathy DX  */
            tmp = put(DGNS_CD{i}, $retin1f.);
            if tmp = '2' then i_ret1 = '1';
```

```
        /* No 2: blidness DX  */
         tmp = put(DGNS_CD{i}, $blind.);
         if tmp = '1' then i_blind = '1';

        /* No 3: Diabetic cataract DX   */
         tmp = put(DGNS_CD{i}, $catar.);
         if tmp = '1' then i_catar = '1';

        /* No 4:  Glaucoma DX  */
         tmp = put(DGNS_CD{i}, $glauco.);
         if tmp = '1' then i_gla2 = '1';          /*glauco2*/

        /* No 5:  Macular Edema DX  */
         tmp = put(DGNS_CD{i}, $edema.);
         if tmp = '1' then i_ede = '1';
   end;


   /* No 6: Laser treatment CPT    */
   /* Laser CPT: for physician  */
         tmp = put(HCPSCD01, $laser1f.);
         if tmp = '1' then i_las = '1';

   /* PTCA: CPT code */
         tmp = put(HCPSCD01, $PTCA2F.);
         if tmp = '1' then i_ptc = '1';
RUN;
```

D#MSQ.MSQ00M7.PGMLIB99(C006PS) -- CONTINUED

```
 ********************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.PSCLM DIRECTORY ;
TITLE "PS Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.PSCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR PS CLAIMS";
RUN;


proc freq;
table i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5 i_led9 /missing;
title 'test i_led9';
run;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN)

```
//WF1706S JOB (@MSQ0000,WF17),HONG,CLASS=K,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 3 PARAMETERS BEFORE RUNNING THIS *****
//*       IN1  = REPLACE FILE NAME WITH FILE NAME OUTPUTED FROM DESY
//*       99   = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*******************************************************************
//*******************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/03/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :READ IN RAW CLAIMS DATA AND GENERATE FLAGS FOR PROCESS AND
//*            PROGRESSION OF DISEASE MEASURES ON THE FLY
//*******************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*******************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C006SN.Y99SNCLM
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(900,300))
//X37IGN DD DUMMY
//IN1    DD DSN=P#DSY.@AAA2049.WF17.@-------.R0009818.OUT,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C006SN.Y99SNCLM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(600,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;
%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;


 ******************************************************************;
 * STEP 1. read in SNF claims
            note: this input statement is an old version. For the latest
version
           of files generated by DESY, need to add 50 spaces to each
variable.
           See 1992-1998 or 2000-2001 for example.
 ******************************************************************;

*******************************************************************;
*  FULL_NAME:  FI INPATIENT SNF CLAIMS RECORD                      ;
```

```
*  VERSION 'I' SAS COPYLIB STATEMENT                                    ;
*  MEMBER NAME: UTLIIPSN                                                ;
*  CREATED BY  MARVELLA BUNCH 11/01/2000                                ;
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

*  UPDATED BY  MARVELLA BUNCH 03/01/2001                                ;
*********************************************************************;


DATA TEMP1 (KEEP=HICNO PROVIDER
                 DGNSCD01-DGNSCD10
                 PRCDRCD1-PRCDRCD6
                 RVCNTR01-RVCNTR45
                 HCPSCD01-HCPSCD45
             STUS_CD PER_DIEM UTIL_DAY COIN_DAY
                 FROM_DT THRU_DT PMT_AMT);
  INFILE IN1 MISSOVER;
    LENGTH REC_LEN                                4;
    LENGTH ACRTN_NM                               3;
    LENGTH LINK_NUM                               5;
    LENGTH SGMT_CNT                               2;
    LENGTH SGMT_NUM                               2;
    LENGTH LINECNT                                3;
    LENGTH SGMTLINE                               2;
    LENGTH BENE_DOB                               5;
    LENGTH IPEDCNT                                2;
    LENGTH IPPATCNT                               2;
    LENGTH IPMCOCNT                               2;
    LENGTH IPPLNCNT                               2;
    LENGTH IPDEMCNT                               2;
    LENGTH IPDGNCNT                               2;
    LENGTH IPPRCNT                                2;
    LENGTH IPCONCNT                               2;
    LENGTH IPOCRCNT                               2;
    LENGTH IPSPNCNT                               2;
    LENGTH IPVALCNT                               2;
    LENGTH IPREVCNT                               2;
    LENGTH GRC_DAY                                2;
    LENGTH UTIL_DAY                               3;
    LENGTH CR_DAY                                 3;
    LENGTH COIN_DAY                               3;
    LENGTH COYR1DAY                               3;
    LENGTH COYR2DAY                               3;
    LENGTH LRD_USE                                3;
    LENGTH NUTILDAY                               4;
    LENGTH PSYCHDAY                               3;
    LENGTH BLDFRNSH                               3;
    LENGTH BLD_RPLC                               3;
    LENGTH BLDNRPLC                               3;
    LENGTH BLDDEDPT                               3;
    ARRAY EDITIND{13}                        $    1 EDTND01-EDTND13;
    ARRAY EDIT_CD{13}                        $    4 EDITCD01-EDITCD13;
    ARRAY PATCHIND{30}                       $    1 PTCHND01-PTCHND30;
    ARRAY PATCHCD{30}                        $    2 PTCHCD01-PTCHCD30;
    ARRAY PATCHDT{30}                        $    8 PTCHDT01-PTCHDT30;
    ARRAY MCOIND{2}                          $    1 MCOIND1-MCOIND2;
    ARRAY MCONUM{2}                          $    5 MCONUM1-MCONUM2;
    ARRAY MCOOPTN{2}                         $    1 MCOOPTN1-MCOOPTN2;
```

```
    ARRAY MCOEFFDT{2}                          $    8 MCFFDT1-MCFFDT2;
    ARRAY MCOTRMDT{2}                          $    8 MCTRMDT1-MCTRMDT2;
    ARRAY MCOPLNID{2}                          $   14 MCPLND1-MCPLND2;
    ARRAY PLANIDIN{3}                          $    1 PLNDND1-PLNDND3;
    ARRAY PLANIDCD{3}                          $    1 PLNDCD1-PLNDCD3;
    ARRAY PLANID{3}                            $   14 PLANID1-PLANID3;
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

    ARRAY DEMOIND{5}                           $    1 DEMOIND1-DEMOIND5;
    ARRAY DEMONUM{5}                           $    2 DEMONUM1-DEMONUM5;
    ARRAY DEMOTXT{5}                           $   15 DEMOTXT1-DEMOTXT5;
    ARRAY DGNSIND{10}                          $    1 DGNSND01-DGNSND10;
    ARRAY DGNS_CD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDRIND{6}                          $    1 PRCDRND1-PRCDRND6;
    ARRAY PRCDR_CD{6}                          $    4 PRCDRCD1-PRCDRCD6;
    ARRAY PRCDR_DT{6}                          $    8 PRCDRDT1-PRCDRDT6;
    ARRAY CONDIND{30}                          $    1 CNDND01-CNDND30;
    ARRAY RLT_COND{30}                         $    2 RLTCND01-RLTCND30;
    ARRAY OCRNCIND{30}                         $    1 OCRCND01-OCRCND30;
    ARRAY OCRNC_CD{30}                         $    2 OCRCCD01-OCRCCD30;
    ARRAY OCRNCDT{30}                          $    8 OCRCDT01-OCRCDT30;
    ARRAY SPANIND{10}                          $    1 SPNND01-SPNND10;
    ARRAY SPAN_CD{10}                          $    2 SPANCD01-SPANCD10;
    ARRAY SPANFROM{10}                         $    8 SPNFRM01-SPNFRM10;
    ARRAY SPANTHRU{10}                         $    8 SPNTHR01-SPNTHR10;
    ARRAY VALIND{36}                           $    1 VALIND01-VALIND36;
    ARRAY VAL_CD{36}                           $    2 VAL_CD01-VAL_CD36;
    ARRAY VAL_AMT{36}                               8 VALAMT01-VALAMT36;
    ARRAY REVIND{45}                           $    1 REVIND01-REVIND45;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY REV_DT{45}                           $    8 REV_DT01-REV_DT45;
    ARRAY REVANSI1{45}                         $    5 RVNS101-RVNS145;
    ARRAY REVANSI2{45}                         $    5 RVNS201-RVNS245;
    ARRAY REVANSI3{45}                         $    5 RVNS301-RVNS345;
    ARRAY REVANSI4{45}                         $    5 RVNS401-RVNS445;
    ARRAY APCHIPPS{45}                         $    5 APCPPS01-APCPPS45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;
    ARRAY MDFR_CD1{45}                         $    2 MDFCD101-MDFCD145;
    ARRAY MDFR_CD2{45}                         $    2 MDFCD201-MDFCD245;
    ARRAY MDFR_CD3{45}                         $    2 MDFCD301-MDFCD345;
    ARRAY MDFR_CD4{45}                         $    2 MDFCD401-MDFCD445;
    ARRAY MDFR_CD5{45}                         $    2 MDFCD501-MDFCD545;
    ARRAY PMTMTHD{45}                          $    2 PMTTHD01-PMTTHD45;
    ARRAY DSCNTIND{45}                         $    1 DSCTND01-DSCTND45;
    ARRAY PACKGIND{45}                         $    1 PCKGND01-PCKGND45;
    ARRAY PRICNG{45}                           $    2 PRICNG01-PRICNG45;
    ARRAY OTAF_1{45}                           $    1 OTAF_101-OTAF_145;
    ARRAY OTAF_2{45}                           $    1 OTAF_201-OTAF_245;
    ARRAY IDENDC{45}                           $   24 IDENDC01-IDENDC45;
    ARRAY REV_UNIT{45}                              4 RVUNT01-RVUNT45;
    ARRAY REV_RATE{45}                              8 RVRT01-RVRT45;
    ARRAY REVBLOOD{45}                              8 RVBLD01-RVBLD45;
    ARRAY REVDCTBL{45}                              8 RVDTBL01-RVDTBL45;
    ARRAY WAGEADJ{45}                               8 WGDJ01-WGDJ45;
    ARRAY RDCDCOIN{45}                              8 RDCDCN01-RDCDCN45;
    ARRAY REV_MSP1{45}                              8 RVMSP101-RVMSP145;
    ARRAY REV_MSP2{45}                              8 RVMSP201-RVMSP245;
    ARRAY REVPCCHG{45}                              8 RVPCHG01-RVPCHG45;
```

```
    ARRAY RPRVDPMT{45}                              8 RPRPMT01-RPRPMT45;
    ARRAY RBENEPMT{45}                              8 RBNPMT01-RBNPMT45;
    ARRAY PTNTRESP{45}                              8 PTNRSP01-PTNRSP45;
    ARRAY REVPMT{45}                                8 REVPMT01-REVPMT45;
    ARRAY REV_CHRG{45}                              8 RVCHRG01-RVCHRG45;
    ARRAY REV_NCVR{45}                              8 RVNCVR01-RVNCVR45;
    ARRAY REVDEDCD{45}                           $  1 RVDDCD01-RVDDCD45;
    INPUT
        /*  FI_IP_SNF_CLM_FIX_I_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

```
            /*  CLM_REC_IDENT_GRP  */
                @1  REC_LEN                     PD3.
                @4  REC_LVL                     $CHAR1.
                @5  RIC_CD                      $CHAR1.
                @6  MQA_RIC                     $CHAR1.
                @7  CLM_TYPE                    $CHAR2.
            /*  FI_CLM_LINK_GRP  */
                /*  CLM_LCTR_NUM_GRP  */
                    @9  CAN                        $CHAR9.
                    @9  HICNO                      $CHAR11.

                    @18  EQ_BIC                    $CHAR2.
                @20  BIC                         $CHAR2.
                @22  ST_SGMT                     $CHAR1.
                @23  STATE_CD                    $CHAR2.
                @25  FROM_DT                     YYMMDD8.

                @33  THRU_DT                     YYMMDD8.

                @41  WKLY_DT                     $CHAR8.
                @49  ACRTN_DT                    $CHAR8.
                @57  ACRTN_NM                    PD2.
                @59  CLM_CNTL                    $CHAR23.
                @82  ORIGCNTL                    $CHAR23.
                @105  QUERY_CD                    $CHAR1.
                @106  PROVIDER                    $CHAR6.
                @112  DAILY_DT                    $CHAR8.
                @120  LINK_NUM                    PD5.
                @125  SGMT_CNT                    2.
                @127  SGMT_NUM                    2.
                @129  LINECNT                     3.
                @132  SGMTLINE                    2.
            /*  FI_CLM_CMN_I_GRP  */
                @134  PE_RIC                      $CHAR1.
                @135  TRANS_CD                    $CHAR1.
                /*  CLM_BILL_TYPE_CD_GRP  */
                    @136  FAC_TYPE                    $CHAR1.
                    @137  TYPESRVC                    $CHAR1.
                @138  FREQ_CD                     $CHAR1.
                @139  _F000001                    $CHAR1.
                @140  MQAQUERY                    $CHAR1.
                @141  DISP_CD                     $CHAR2.
                @143  EDITDISP                    $CHAR2.
                @145  BIC_MDFY                    $CHAR1.
                @146  CNTY_CD                     $CHAR3.
                @149  RCPT_DT                     $CHAR8.
                @157  SCHLD_DT                    $CHAR8.
```

```
                @165  FRWRD_DT                    $CHAR8.
                @173  FI_NUM                     $CHAR5.
                @178  ASGN_NUM                   $CHAR8.
                @186  FIBATCH                    $CHAR4.
                @190  BENE_ZIP                   $CHAR9.
                @199  SEX                        $CHAR1.
                @200  RACE                       $CHAR1.
                @201  BENE_DOB                   8.
                @209  MS_CD                      $CHAR2.
                @211  SURNAME                    $CHAR6.
                @217  FRSTINIT                   $CHAR1.
                @218  MDL_INIT                   $CHAR1.
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

                @219  CWFLOCCD                   $CHAR1.
                @220  PDGNS_CD                   $CHAR5.
                @225  _F000002                   $CHAR1.
                @226  NOPAY_CD                   $CHAR1.
                @227  TRTMT_CD                   $CHAR1.
                @228  PMT_AMT                    PD6.2
                @234  PRPAYAMT                   PD6.2
                @240  PRPAY_CD                   $CHAR1.
                @241  CANCELCD                   $CHAR1.
                @242  ACTIONCD                   $CHAR1.
                @243  APRVL_DT                   $CHAR8.
                @251  PRSTATE                    $CHAR2.
                @253  ORGNPINM                   $CHAR10.
                /*  ATNDG_PHYSN_ID_GRP  */
                    @263  AT_UPIN                    $CHAR6.
                    @269  AT_NPI                     $CHAR10.
                    @279  AT_SRNM                    $CHAR6.
                    @285  AT_GVNNM                   $CHAR1.
                    @286  AT_MDL                     $CHAR1.
                /*  OPRTG_PHYSN_ID_GRP  */
                    @287  OP_UPIN                    $CHAR6.
                    @293  OP_NPI                     $CHAR10.
                    @303  OP_SRNM                    $CHAR6.
                    @309  OP_GVN                     $CHAR1.
                    @310  OP_MDL                     $CHAR1.
                /*  OTHR_PHYSN_ID_GRP  */
                    @311  OT_UPIN                    $CHAR6.
                    @317  OT_NPI                     $CHAR10.
                    @327  OT_SRNM                    $CHAR6.
                    @333  OT_GVN                     $CHAR1.
                    @334  OT_MDL                     $CHAR1.
                @335  MDCD_PRV                   $CHAR13.
                @348  MDCDINFO                   $CHAR4.
                @352  MCOPDSW                    $CHAR1.
                @353  AUTHRZTN                   $CHAR18.
                @371  PTNTCNTL                   $CHAR20.
                @391  MDCL_REC                   $CHAR17.
                @408  PRO_CNTL                   $CHAR12.
                @420  PRO_DT                     $CHAR8.
                @428  STUS_CD                    $CHAR2.
                @430  DGNS_E                     $CHAR5.
                @435  _F000003                   $CHAR1.
                @436  PPS_IND                    $CHAR1.
                @437  TOT_CHRG                   PD6.2
```

```
            @443  _F000004                    $CHAR50.
          @493  IPEDCNT                     2.
          @495  IPPATCNT                    2.
          @497  IPMCOCNT                    1.
          @498  IPPLNCNT                    1.
          @499  IPDEMCNT                    1.
          @500  IPDGNCNT                    2.
          @502  IPPRCNT                     2.
          @504  IPCONCNT                    2.
          @506  IPOCRCNT                    2.
          @508  IPSPNCNT                    2.
          @510  IPVALCNT                    2.
          @512  IPREVCNT                    2.
          @514  _F000005                    $CHAR4.
          /*  FI_IP_SNF_CLM_SPECF_I_GRP  */
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

```
            @518  ADMSN_DT                    $CHAR8.
            @526  TYPE_ADM                    $CHAR1.
            @527  SRC_ADMS                    $CHAR1.
            @528  AD_DGNS                     $CHAR5.
            @533  _F000006                    $CHAR1.
            @534  PTNTSTUS                    $CHAR1.
            @535  APRVL_CD                    $CHAR1.
            @536  PRO_FROM                    $CHAR8.
            @544  PRO_THRU                    $CHAR8.
            @552  GRC_DAY                     1.
            @553  PER_DIEM                    PD6.2
            @559  DED_AMT                     PD6.2
            @565  COIN_AMT                    PD6.2
            @571  BLDDEDAM                    PD6.2
            @577  BLDTCHRG                    PD6.2
            @583  BLDNCHRG                    PD6.2
            @589  PCCHGAMT                    PD6.2
            @595  NCCHGAMT                    PD6.2
            @601  TDEDAMT                     PD6.2
            @607  PPS_CPTL                    PD6.2
            @613  CPTL_HSP                    PD6.2
            @619  CPTL_FSP                    PD6.2
            @625  CPTLOUTL                    PD6.2
            @631  DISP_SHR                    PD6.2
            @637  IME_AMT                     PD6.2
            @643  CPTL_EXP                    PD6.2
            @649  HLDHRMLS                    PD6.2
            @655  DSCHFRCT                    PD3.4
            @658  DRGWTAMT                    PD4.4
            @662  UTIL_DAY                    PD2.
            @664  CR_DAY                      PD2.
            @666  COIN_DAY                    PD2.
            @668  COYR1DAY                    PD2.
            @670  COYR1AMT                    PD6.2
            @676  COYR2DAY                    PD2.
            @678  COYR2AMT                    PD6.2
            @684  LRD_USE                     PD2.
            @686  NUTILDAY                    PD3.
            @689  PSYCHDAY                    PD2.
            @691  BLDFRNSH                    PD2.
            @693  BLD_RPLC                    PD2.
```

```
                @695  BLDNRPLC                    PD2.
                @697  BLDDEDPT                   PD2.
                @699  QLFYFROM                   $CHAR8.
                @707  QLFYTHRU                   $CHAR8.
                @715  NCOVFROM                   $CHAR8.
                @723  NCOVTHRU                   $CHAR8.
                @731  GURPMTDT                   $CHAR8.
                @739  URNTCDT                    $CHAR8.
                @747  CARETHRU                   $CHAR8.
                @755  EXHST_DT                   $CHAR8.
                @763  DSCHRGDT                   $CHAR8.
                @771  DRG_CD                     $CHAR3.
                @774  OUTLR_CD                   $CHAR1.
                @775  OUTLRPMT                   PD6.2
                @781  KRON_IND                   $CHAR1.
                @782  _F000007                   $CHAR24.
        /*  FI_IP_SNF_CLM_TRLR_I_GRP  */
    @;
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

```
    DO _I1_ = 1 TO IPEDCNT;
        INPUT
            /*  NCH_EDIT_I_GRP  */
                EDITIND{_I1_}              $CHAR1.
                EDIT_CD{_I1_}              $CHAR4.
        @;
    END;
    DO _I1_ = 1 TO IPPATCNT;
        INPUT
            /*  NCH_PATCH_I_GRP  */
                PATCHIND{_I1_}             $CHAR1.
                PATCHCD{_I1_}              $CHAR2.
                PATCHDT{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPMCOCNT;
        INPUT
            /*  MCO_PRD_I_GRP  */
                MCOIND{_I1_}               $CHAR1.
                MCONUM{_I1_}               $CHAR5.
                MCOOPTN{_I1_}              $CHAR1.
                MCOEFFDT{_I1_}             $CHAR8.
                MCOTRMDT{_I1_}             $CHAR8.
                MCOPLNID{_I1_}             $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPPLNCNT;
        INPUT
            /*  CLM_HLTH_PLANID_I_GRP  */
                PLANIDIN{_I1_}             $CHAR1.
                PLANIDCD{_I1_}             $CHAR1.
                PLANID{_I1_}               $CHAR14.
        @;
    END;
    DO _I1_ = 1 TO IPDEMCNT;
        INPUT
            /*  CLM_DEMO_ID_I_GRP  */
                DEMOIND{_I1_}              $CHAR1.
```

```
            DEMONUM{_I1_}               $CHAR2.
            DEMOTXT{_I1_}               $CHAR15.
        @;
    END;
    DO _I1_ = 1 TO IPDGNCNT;
        INPUT
            /*  CLM_DGNS_I_GRP  */
            DGNSIND{_I1_}               $CHAR1.
            DGNS_CD{_I1_}               $CHAR5.
            _F000008                    $CHAR1.
        @;
    END;
    DO _I1_ = 1 TO IPPRCNT;
        INPUT
            /*  CLM_PRCDR_I_GRP  */
            PRCDRIND{_I1_}              $CHAR1.
            PRCDR_CD{_I1_}              $CHAR4.
            _F000009                    $CHAR3.
            PRCDR_DT{_I1_}              $CHAR8.
        @;
    END;
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

```
    DO _I1_ = 1 TO IPCONCNT;
        INPUT
            /*  CLM_RLT_COND_I_GRP  */
            CONDIND{_I1_}               $CHAR1.
            RLT_COND{_I1_}              $CHAR2.
        @;
    END;
    DO _I1_ = 1 TO IPOCRCNT;
        INPUT
            /*  CLM_RLT_OCRNC_I_GRP  */
            OCRNCIND{_I1_}              $CHAR1.
            OCRNC_CD{_I1_}              $CHAR2.
            OCRNCDT{_I1_}               $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPSPNCNT;
        INPUT
            /*  CLM_OCRNC_SPAN_I_GRP  */
            SPANIND{_I1_}               $CHAR1.
            SPAN_CD{_I1_}               $CHAR2.
            SPANFROM{_I1_}              $CHAR8.
            SPANTHRU{_I1_}              $CHAR8.
        @;
    END;
    DO _I1_ = 1 TO IPVALCNT;
        INPUT
            /*  CLM_VAL_I_GRP  */
            VALIND{_I1_}                $CHAR1.
            VAL_CD{_I1_}                $CHAR2.
            VAL_AMT{_I1_}               PD6.2
        @;
    END;
    DO _I1_ = 1 TO IPREVCNT;
        INPUT
            /*  CLM_REV_CNTR_I_GRP  */
```

```
                REVIND{_I1_}               $CHAR1.
                REV_CNTR{_I1_}             $CHAR4.
                REV_DT{_I1_}               $CHAR8.
                REVANSI1{_I1_}             $CHAR5.
                REVANSI2{_I1_}             $CHAR5.
                REVANSI3{_I1_}             $CHAR5.
                REVANSI4{_I1_}             $CHAR5.
                APCHIPPS{_I1_}             $CHAR5.
                HCPCS_CD{_I1_}             $CHAR5.
                MDFR_CD1{_I1_}             $CHAR2.
                MDFR_CD2{_I1_}             $CHAR2.
                MDFR_CD3{_I1_}             $CHAR2.
                MDFR_CD4{_I1_}             $CHAR2.
                MDFR_CD5{_I1_}             $CHAR2.
                PMTMTHD{_I1_}              $CHAR2.
                DSCNTIND{_I1_}             $CHAR1.
                PACKGIND{_I1_}             $CHAR1.
                PRICNG{_I1_}               $CHAR2.
                OTAF_1{_I1_}               $CHAR1.
                OTAF_2{_I1_}               $CHAR1.
                IDENDC{_I1_}               $CHAR24.
                REV_UNIT{_I1_}             PD4.
                REV_RATE{_I1_}             PD6.2
                REVBLOOD{_I1_}             PD6.2
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

                REVDCTBL{_I1_}             PD6.2
                WAGEADJ{_I1_}              PD6.2
                RDCDCOIN{_I1_}             PD6.2
                REV_MSP1{_I1_}             PD6.2
                REV_MSP2{_I1_}             PD6.2
                REVPCCHG{_I1_}             PD6.2
                RPRVDPMT{_I1_}             PD6.2
                RBENEPMT{_I1_}             PD6.2
                PTNTRESP{_I1_}             PD6.2
                REVPMT{_I1_}               PD6.2
                REV_CHRG{_I1_}             PD6.2
                REV_NCVR{_I1_}             PD6.2
                REVDEDCD{_I1_}             $CHAR1.
                _F000010                   $CHAR50.
        @;
    END;
    INPUT
        EOR                        $CHAR3.  @;
   /* Release trailing @ */
   *input;

    DROP _I1_;
    DROP _F000001-_F000010;
RUN;


 ********************************************************************;
 * STEP 2. generate process and progression of disease measure flags
 ********************************************************************;
DATA OUT1.Y&YEAR.SNCLM (keep=hicno provider
            STUS_CD FROM_DT THRU_DT PMT_AMT
      i_hgl
```

```
          i_lipid1 i_lipid3
          i_self   i_educ1 i_educ2 i_eye1
          i_urine2 i_shoe
          i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
          i_led4   i_led5
          i_led6   i_led7    i_led8 i_led9
          i_neur_p i_neur_s  i_neph i_ret1
          i_blind  i_catar   i_gla2 i_ede i_las
          i_ptc
    );
      SET TEMP1
    ;
       attrib i_hgl     length=$1 label='Hgl A1c Test';
       attrib i_lipid1  length=$1 label='Lipid Profiling: definition 1';
       attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';

       attrib i_self   length=$1 label='Self-Monitoring';
       attrib i_educ1  length=$1 label='Diabetes Education: definition 1';
       attrib i_educ2  length=$1 label='Diabetes Education: definition 2';

       attrib i_eye1    length=$1 label='Eye Examination 1';
       attrib  i_urine2   length=$1  label='Microalbumin  Scrning/Urine  Protein
2';
       attrib i_shoe    length=$1 label='Therapeutic Shoes';

       attrib i_led_p1  length=$1 label='Primary DX: cellulitis of LED';
       attrib i_led_s1  length=$1 label='Secondary DX: cellulitis of LEDs';
```



```
       attrib i_led_p2  length=$1 label='Primary DX: cutaneous gangrene';
       attrib i_led_s2  length=$1 label='Secondary DX: cutaneous gangrene';
       attrib i_led_p3  length=$1 label='Primary DX:osteomyelitis';
       attrib i_led_s3  length=$1 label='Secondary DX:osteomyelitis';
       attrib i_led4  length=$1 label='Arthropathy w/ Nerve Dis';
       attrib i_led5  length=$1 label='diabetes w periph cir disorder';
       attrib i_led6  length=$1 label='Lower limb amputations';
       attrib i_led7  length=$1 label='Incision and drainage below facia';
       attrib i_led8  length=$1 label='Incision and drainage of bone cortex';
       attrib  i_led9    length=$1  label='Skin  and  Subcutan  debride  for
gangrene';

       attrib i_neur_p  length=$1 label='Primary DX: Peripheral neuropathy';
       attrib i_neur_s  length=$1 label='Second DX: Peripheral neuropathy';
       attrib i_neph    length=$1 label='Nephropathy';
       attrib i_ret1    length=$1 label='Retinopathy 1';

       attrib i_blind  length=$1 label='Blindness';
       attrib i_catar  length=$1 label='Diabetic cataract';
       attrib i_gla2   length=$1 label='Glaucoma 2';
       attrib i_ede    length=$1 label='Macular Edema';
       attrib i_las    length=$1 label='Laser treatment';
       attrib i_ptc  length=$1 label='Progression to PTCA, including stents';

       attrib hicno    length=$11 label='Health insurance claim number';
       attrib FROM_DT  length=8   label='Claim from date';
       attrib THRU_DT  length=8   label='Claim through date';
       attrib stus_cd  length=$2  label='Patient discharge status code';
       attrib provider length=$6  label='Provider number';
```

```
   attrib pmt_amt  length=8   label='Claim payment amount';

   /* Array for input variables */
    ARRAY DGNS_CD{10}                          $    5 DGNSCD01-DGNSCD10;
    ARRAY PRCDR_CD{6}                          $    4 PRCDRCD1-PRCDRCD6;
    ARRAY REV_CNTR{45}                         $    4 RVCNTR01-RVCNTR45;
    ARRAY HCPCS_CD{45}                         $    5 HCPSCD01-HCPSCD45;

   attrib tmp      length=$2  label='Temporary';

   /* Initialize indicators */
   array inds {*} $
      i_hgl
      i_lipid1 i_lip3_1-i_lip3_3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc
;
   do _i=1 to dim(inds);
      inds{_i} = '0';
   end;

   ******** (1) For process of disease ************************;
   do i =1 to 45;
        /* CPT code for Hgl A1c Test:*/
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

         tmp = put(HCPCS_CD{i}, $hgl.);
         if tmp = '1' then i_hgl = '1';

        /* CPT code  for lipid profiling: */
         /* i_lipid1 */
         tmp = put(HCPCS_CD{i}, $lipid1f.);
         if tmp = '1' then i_lipid1 = '1';

         /* i_lipid3: 1-3 on the same day */
         tmp = put(HCPCS_CD{i}, $lipid3f.);
         if tmp = '1' then i_lip3_1 = '1';
         ELSE if tmp = '2' then i_lip3_2 = '1';
         ELSE if tmp = '3' then i_lip3_3 = '1';

         /* CPT: Self monitoring  */
         tmp = put(HCPCS_CD{i}, $monitor.);
         if tmp = '1' then i_self = '1';       /*physician*/
         ELSE if tmp = '2' then i_self = '1';  /*DME */
   end;
   ********* create final indicator for i_lipid3 ***************;
   IF (i_lip3_1='1' AND i_lip3_2='1' AND i_lip3_3='1' )
                       THEN i_lipid3='1';  else i_lipid3='0';


   /* diabetes education */
   /* educ1: CPT  */
```

```
      do i =1 to 45;
            /* i_educ11 */
            tmp = put(HCPCS_CD{i}, $educ1f.);
            if tmp = '1' then i_educ1 = '1';          /*FOR 1998 FORWARD*/
      end;

      /* educ2: revenue code  */
      do i =1 to 45;
            /* i_educ11 */
            tmp = put(REV_CNTR{i}, $educ2f.);         /*FOR ALL YR*/
            if tmp = '1' then i_educ2 = '1';
      end;

      do i =1 to 45;
            /* Eye examination */
            /*Eye1: CPT */
            tmp = put(HCPCS_CD{i}, $eye1f.);
            if tmp = '1' then i_eye1 = '1';

            /* CPT: Microalbumin Screening /Quantitative Urine Protein */
            /* urine2:  */
            tmp = put(HCPCS_CD{i}, $urine2f.);
            if tmp = '1' then i_urine2 = '1';

            /* CPT: Therapeutic Shoes */
            tmp = put(HCPCS_CD{i}, $shoe.);
            if tmp = '1' then i_shoe = '1';
      end;


      ******** (2) For progression of disease ************************;

      /* Lower Extremity Diseases               */
   D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

      /* separate flags for principal or secondary DX for '1'*/
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $lower.);
            if i=1 then do;
              if tmp = '1' then i_led_p1 = '1';      /*cellulitis */
            end;

            else do;
              if tmp = '1' then i_led_s1 = '1';      /*cellulitis */
            end;
      end;

      /* Lower1: DX for category 2 to 5  */
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $lower1f.);
            if i=1 then do;
              if tmp = '1' then i_led_p2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_p3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4 = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
```

```
            else do;
              if tmp = '1' then i_led_s2 = '1';      /*cutaneous gangrene*/
              ELSE if tmp = '2' then i_led_s3 = '1'; /*osteomyelitis: */
              ELSE if tmp = '4' then i_led4   = '1'; /*arthropathy w below*/
              ELSE  if  tmp  =  '5'  then  i_led5  =  '1';  /*diab  periph  circu
disor*/
            end;
      end;


      /* Lower2: CPT for category 6 to 9  */
      do i =1 to 45;
            tmp = put(HCPCS_CD{i}, $lower2f.);
                * lower limb amputations No 6*;
            if tmp = '1' then i_led6 = '1';
                * incision and drainage below facia No 7*;
            ELSE if tmp = '2' then i_led7 = '1';
                * incision and drainage of bone cortex No 8*;
            ELSE if tmp = '3' then i_led8 = '1';
                * skin and subcutaneous debridement for gangrene No 9*;
            ELSE if tmp = '4' then i_led_9 = '1'; *combine w DX 785.4*;
      end;

      /* Lower3: ICD procedure code for category 6 to 9  */
         do i =1 to 6;
            tmp = put(PRCDR_CD{i}, $lower3f.);
               /* lower limb amputations No 6*/
            if tmp = '1' then i_led6 = '1';
                /* incision and drainage below facia No 7*/
            else if tmp = '2' then i_led7 = '1';
                /* incision and drainage of bone cortex No 8*/
            else if tmp = '3' then i_led8 = '1';
                /* skin and subcutaneous debridement for gangrene No 9*/
            else if tmp = '4' then i_led_9 = '1'; /*combine w DX 785.4*/
         end;
```

D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

```
         ********* Final indicator for Low Extremety Disease *****;
            if i_led_9 = '1' and (i_led_p2 = '1' or i_led_s2 = '1')
                                 then i_led9 = '1';


      /* DX: Neurological complications */
      /* Autonomic Neuropathy    */
      do i =1 to 10;
            tmp = put(DGNS_CD{i}, $neuro.);
            if i=1 then do;
              if tmp = '1' then i_neur_p = '1';  /*perpheral neuropathy*/
            end;

            else do;
              if tmp = '1' then i_neur_s = '1';  /*perpheral neuropathy*/
            end;
      end;

      /* DX: Nephropathy   */
      do i =1 to 10;
```

```
          tmp = put(DGNS_CD{i}, $nephro.);
          if tmp = '2' then i_neph = '1';
     end;

   /* Diabetic Eye Diseases   */
   do i =1 to 10;
        /* No 1: retinopathy DX  */
          tmp = put(DGNS_CD{i}, $retin1f.);
          if tmp = '2' then i_ret1 = '1';


        /* No 2: blidness DX  */
          tmp = put(DGNS_CD{i}, $blind.);
          if tmp = '1' then i_blind = '1';

        /* No 3: Diabetic cataract DX   */
          tmp = put(DGNS_CD{i}, $catar.);
          if tmp = '1' then i_catar = '1';

        /* No 4:  Glaucoma DX  */
          tmp = put(DGNS_CD{i}, $glauco.);
          if tmp = '1' then i_gla2 = '1';          /*glauco2*/

        /* No 5:  Macular Edema DX  */
          tmp = put(DGNS_CD{i}, $edema.);
          if tmp = '1' then i_ede = '1';
   end;


     /* No 6: Laser treatment CPT    */
     /* Laser CPT:   */
     do i =1 to 45;
         tmp = put(HCPCS_CD{i}, $laser1f.);
         if tmp = '1' then i_las = '1';
     end;

     /* Laser treatment ICD9 procedure:  */
      do i =1 to 6;
         tmp = put(PRCDR_CD{i}, $laser2f.);
D#MSQ.MSQ00M7.PGMLIB99(C006SN) -- CONTINUED

         if tmp = '1' then i_las = '1';
      end;


   /* Progresion to PTCA */
      /* ICD-9 procedure code */
      do i =1 to 6;
         tmp = put(PRCDR_CD{i}, $hosp2f.);
         if tmp = '2' then i_ptc = '1';
      end;

     /* CPT code  */
     do i =1 to 45;
         tmp = put(HCPCS_CD{i}, $PTCA2F.);
         if tmp = '1' then i_ptc = '1';
     end;
```

```
RUN;


 ********************************************************************;
 * STEP 3. Proc contents and proc print on the SAS claims file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.SNCLM DIRECTORY ;
TITLE "SNF Y&YEAR: CLAIMS FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.SNCLM (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR SNF CLAIMS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C007)

```
//WF17007 JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/10/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :(1) DELETE STAND-ALONE ER CLAIMS, DEMONSTRATION CLAIMS,
//*            (2) DELETE ALL OVERLAPPING CLAIMS,
//*            (3) IDENTIFY AND CONSOLIDATE INTERIM CLAIMS,
//*            (4) IDENTIFY AND CONSOLIDATE TRANSFER CLAIMS,
//*            TO GET A STAY-LEVEL INPATIENT FILE.
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C007.Y99IPSTY
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006IP.Y99IPCLM,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C007.Y99IPSTY,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen  ;

%LET YEAR=99;  *** YEAR OF PRODUCTION;


 ********************************************************************;
 * STEP 1. identify and delete stand-alone ER, demonstration claims.
 * Claims that meet all three of the following criteria are considered
 * stand-alone ER claims: 1) MedPAR Discharge Destination equals '20' or
 * MedPAR Discharge Status Code equals 'B', and 2) the sum of all MedPAR
 * Accommodation Charges is equal to 0, and 3) the MedPAR Emergency Room
 * Service Charge is greater than 0.
 * Demonstration Claims are defined as claims with the 4th-6th positions
 * of the MedPAR Provider Identification Number in the 880 to 899 range.
 ********************************************************************;
```

```
data temp1 delete;
  set in1.Y99IPCLM;
if (stus_cd='20' or PTNTSTUS='B') and ACC_CHRG=0 and ER_CHRG>0 then er=1;
```

```
   else er=0;
if ('880' LE (SUBSTR(provider,4,3)) LE '899') then dem=1;
   else dem=0;
if er =1 or dem=1 then output delete;
else output temp1;
run;

******* check: how many stand-alone ER, demonstration claims?;
proc freq data=delete;
table er*dem /missing;
run;


*********************************************************************;
* STEP 2. Sort file by HICNO, admission date, discharge date, and
*         provider ID to facilitate identification of overlapping claims.
*********************************************************************;
proc sort data=temp1;
   by hicno from_dt thru_dt provider;
run;

***** create new variable OBS for use later ***;
data temp2;
  set temp1;
obs= _n_;
run;


*********************************************************************;
* STEP 3. identify all overlapping claims.
* Overlapping claims are those claims where: 1) HICNOs are the same;
* and 2) Admission date is larger than or equal to the admission date
* of previous claim but smaller than the discharge date of previous
* claim.
*********************************************************************;
data temp3 (keep=NEWOBS
      CASENO     /**UNIQUE # FOR A GROUP OF OVERLAPPING CLAIMS**/
      RENAME=(NEWOBS=OBS));
   set temp2 (drop=pmt_amt UTIL_DAY PER_DIEM );

LENGTH hicno2 $ 11 MARK $ 1 ;
RETAIN MARK ' ';

*** define lagged variables***;
hicno2=lag1(hicno);
from_dt2=lag1(from_dt);
thru_dt2=lag1(thru_dt);
obs2=lag1(obs);

***DEFINE CRITERIA FOR OVERLAPPING CLAIMS***;
IF ((hicno2=hicno) AND
    (from_dt2 LE from_dt LT thru_dt2))
THEN DO;
```

```
    IF MARK=' ' THEN DO;
        CASENO+1;
        NEWOBS=OBS2;
        OUTPUT;
        NEWOBS=OBS;
        OUTPUT;
        MARK='1';
    END;
    ELSE IF MARK='1' THEN DO;
        NEWOBS=OBS;
        OUTPUT;
    END;
END;
ELSE MARK=' ';
RUN;

PROC PRINT DATA=TEMP3 (OBS=10);
TITLE2 "PRINT 10 OBS";
RUN;


*******************************************************************;
* STEP 4. describe basic info on these overlapping claims
*******************************************************************;

PROC FREQ DATA=temp3 NOPRINT;
TABLES CASENO / OUT=TEMP3CNT(DROP=PERCENT);
RUN;
PROC MEANS DATA=TEMP3CNT N MIN MAX MEAN STD;
VAR COUNT;
TITLE "YEAR &YEAR OVERLAPPING CLAIMS DELETIONS";
TITLE2 "STATS ARE FOR VARIABLE COUNT:";
TITLE3 "THE # OF CLAIMS IN EACH OVERLAPPING EPISODE";
RUN;


*******************************************************************;
* STEP 5. Merge inpatient file and overlapping claims case file by OBS.
*         Delete overlapping claims.
*******************************************************************;

DATA temp4;
MERGE temp2 (IN=X)
      temp3 (IN=Y KEEP=OBS);
BY OBS;
IF X AND NOT Y;
RUN;

proc delete data=temp1 temp2 temp3 delete;
run;


 *******************************************************************;
 * STEP 6. sort Inpatient claims to facilitate identification of
 *         interim claims.
 *******************************************************************;
proc sort data=temp4;
   by hicno provider from_dt thru_dt;
```

```
run;


 ********************************************************************;
  * STEP 7. identify interim claims: if they have the same provider and
  * the admission date of the claim is equal to, or one day after, the
  * discharge date of the previous claim, and the status code of the
  * previous claim is "still patient.
 ********************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED

```
data TEMP5 (drop=lastfrom lastthru laststus);
   set TEMP4;
   by hicno provider;

   attrib staynum  length=5  label='Inpatient stay number (sequential)';
   attrib lastfrom length=8  label='Claim from date of last claim';
   attrib lastthru length=8  label='Claim through date of last claim';
   attrib laststus length=$2 label='Claim status code of last claim';

   retain staynum 0 lastfrom lastthru laststus;

   /* Only one inpatient claim with given provider, for this claimant */
   if first.provider and last.provider then staynum = staynum + 1;

   else do;
      if first.provider then do;
         staynum = staynum + 1;
         lastfrom = from_dt;
         lastthru = thru_dt;
         laststus = stus_cd;
      end;
      else do;
         /* Part of the same inpatient stay */
         /* will have same staynum */
         if lastfrom <= from_dt <= lastthru then do;

            /* Make "last through date" and "last status code" */
            /* equal to latest through date and status code.   */
            if thru_dt >= lastthru then do;
               lastthru = thru_dt;
               laststus = stus_cd;
            end;
         end;

         /* Abutting claims, with status code of earlier claim */
         /* equal to "still a patient"                         */
         /* will have same staynum */
         else if (from_dt = lastthru + 1) and
            (30 <= input(laststus, 2.) <= 39) then do;
            lastthru = thru_dt;
            laststus = stus_cd;
         end;

         /* Not part of the same inpatient stay */
         /* will have different staynum */
         else do;
            staynum = staynum + 1;
```

```
               lastfrom = from_dt;
               laststhru = thru_dt;
               laststus = stus_cd;
            end;
         end;
      end;
   RUN;


    ********************************************************************;
    * STEP 8. sort Inpatient claims by hicno and staynum
    ********************************************************************;
   D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED

   proc sort data=temp5 OUT=temp6;
      by hicno staynum from_dt thru_dt;
   RUN;

   proc delete data=temp4 temp5;
   run;


    ********************************************************************;
     * STEP 9. collapse interim claims belong to same stay into a single
claim.
     * The admission date comes from the first claim in a series
     * of interim claims and discharge date is the discharge
     * date of the last claim in a series of interim claims.
     * Sum payment variables.
     * All other variable values come from the last discharge claim
     * in a series of interim claims.
    ********************************************************************;
   data temp7 (drop=lastfrom laststhru laststus lastpamt _i
         i_lip3_1-i_lip3_3
         l_hgl
         l_lipid1 l_lipid3
         l_self   l_educ1 l_educ2 l_eye1
         l_urine2 l_shoe
         l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
         l_led4   l_led5
         l_led6   l_led7 l_led8 l_led9
         l_neur_p l_neur_s  l_neph l_ret1
         l_blind  l_catar l_gla2 l_ede l_las
         l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
         l_cab    l_ptc l_lea l_byp
         l_er_p   l_er_s
         l_db1-l_db5
   );
      set temp6;
      by hicno staynum from_dt thru_dt;

      attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';
      /* create i_lipid3, which requires i_lip3_1-i_lip3_3 on same day */
         IF (i_lip3_1='1' AND i_lip3_2='1' AND i_lip3_3='1' )
         THEN i_lipid3='1';  else i_lipid3='0';

      label
         from_dt  = 'From date'
```

```
      thru_dt  = 'Through date'
      stus_cd  = 'Status code'
      ippmtamt = 'Inpatient payment amount';
   length
      lastfrom 8
      lastthru 8
      laststus $ 2
      lastpamt 8

      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED
```
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5 $ 1;
   retain
      lastfrom
      lastthru
      laststus
      lastpamt

      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5;
   array oinds {*} $
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar    i_gla2 i_ede i_las
      i_stk_p  i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab    i_ptc i_lea i_byp
      i_er_p   i_er_s
      i_db1-i_db5
;
```

```
   array linds {*} $
      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5;

   /* if only one claim for this inpatient stay */
   if first.staynum and last.staynum then do;
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED
```
      output;
   end;

   else do;
      if first.staynum then do;
         lastfrom = from_dt;
         lastthru = thru_dt;
         laststus = stus_cd;
         lastpamt = ippmtamt;
         do _i=1 to dim(oinds);
            linds{_i} = oinds{_i};
         end;
      end;
      else do;
         /* Set latest through date and status code */
         /* Take the last status code, even if through dates are equal */
         if thru_dt >= lastthru then do;
            lastthru = thru_dt;
            laststus = stus_cd;
         end;
         /* Sum payment amounts */
         lastpamt = sum(lastpamt, ippmtamt);
         do _i=1 to dim(oinds);
            if oinds{_i} = '1' then linds{_i} = '1';
         end;

         /* Reset the variables for the last claim, which will */
         /* be output to represent the inpatient stay.         */
         if last.staynum then do;
            from_dt = lastfrom;
            thru_dt = lastthru;
            stus_cd = laststus;
            ippmtamt = lastpamt;

            output;
         end;
      end;
   end;
```

```
run;


proc delete data=temp6;
run;

 ********************************************************************;
 * STEP 10. sort Inpatient claims to facilitate identification of
 *          transfer claims.
 ********************************************************************;
proc sort data=temp7;
   by hicno from_dt thru_dt provider;
run;


 ********************************************************************;
 * STEP 11. identify transfer claims. Transfer
 * claims are those claims where: 1) HICNOs are the same, and
 * 2) PROVIDs are not the same, and 3) admission date is equal to
 * the previous discharge date or the previous discharge date plus 1,
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED
```
 * and 4) discharge destination equals 2.
 ********************************************************************;
data TEMP8 (drop=lastfrom lastthru laststus lastprov);
   set TEMP7;
   by hicno ;

   attrib staynum  length=5  label='Inpatient stay number (sequential)';
   attrib lastfrom length=8  label='Claim from date of last claim';
   attrib lastthru length=8  label='Claim through date of last claim';
   attrib laststus length=$2 label='Claim status code of last claim';
   attrib lastprov length=$6 label='Claim provider code of last claim';

   retain staynum 0 lastfrom lastthru laststus lastprov;

   /* Only one inpatient claim for this claimant */
   if first.hicno and last.hicno then staynum = staynum + 1;

   else do;
      if first.hicno then do;
         staynum = staynum + 1;
         lastfrom = from_dt;
         lastthru = thru_dt;
         laststus = stus_cd;
         lastprov = provider;
      end;
      else do;
         /* transfer claim(s) and the claim right before transfer */
         /* will have same staynum */
         if (provider not= lastprov) and
            (lastthru <= from_dt <= (lastthru +1)) and
            (input(laststus, 2.) =2)
         then do;
            /* Make "last through date", "last status code", and */
            /* "last provider code" equal to latest through date,*/
            /* status code, and provider code.   */
```

```
               lastthru = thru_dt;
               laststus = stus_cd;
               lastprov = provider;
         end;

         /* Not part of the inpatient transfer claims */
         /* will have different staynum */
         else do;
            staynum = staynum + 1;
            lastfrom = from_dt;
            lastthru = thru_dt;
            laststus = stus_cd;
            lastprov = provider;
         end;
      end;
   end;
RUN;


 ********************************************************************;
 * STEP 12. sort Inpatient claims by hicno and staynum
 ********************************************************************;
proc sort data=temp8 ;
   by hicno staynum from_dt thru_dt;
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED
```

RUN;

proc delete data=temp7;
run;


 ********************************************************************;
  * STEP 13. collapse transfer claim and the claims right before it into a
  * single record.
  * The admission date comes from the first claim in a series
  * of transfer claims and discharge date is the discharge
  * date of the last claim in a series of transfer claims.
  * Sum payment variables. Calculate new Length of Stay (LOS).
  * All other variable values come from the last discharge claim
  * in a series of transfer claims.
 ********************************************************************;
data OUT1.Y&YEAR.IPSTY (drop=lastfrom lastthru laststus lastpamt _i
                        PTNTSTUS ACC_CHRG ER_CHRG er dem obs
      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5
);
```

```
   set temp8;
   by hicno staynum from_dt thru_dt;

   label
      from_dt  = 'From date'
      thru_dt  = 'Through date'
      stus_cd  = 'Status code'
      ippmtamt = 'Inpatient payment amount';
   length
      lastfrom 8
      lastthru 8
      laststus $ 2
      lastpamt 8

      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5 $ 1;
```
D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED

```
   retain
      lastfrom
      lastthru
      laststus
      lastpamt

      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5;
   array oinds {*} $
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3
      i_led4   i_led5
      i_led6   i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind  i_catar    i_gla2 i_ede i_las
```

```
      i_stk_p  i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab    i_ptc i_lea i_byp
      i_er_p   i_er_s
      i_db1-i_db5
;

   array linds {*} $
      l_hgl
      l_lipid1 l_lipid3
      l_self   l_educ1 l_educ2 l_eye1
      l_urine2 l_shoe
      l_led_p1 l_led_s1 l_led_p2 l_led_s2 l_led_p3 l_led_s3
      l_led4   l_led5
      l_led6   l_led7 l_led8 l_led9
      l_neur_p l_neur_s  l_neph l_ret1
      l_blind  l_catar l_gla2 l_ede l_las
      l_stk_p  l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab    l_ptc l_lea l_byp
      l_er_p   l_er_s
      l_db1-l_db5;

   /* if only one claim for this inpatient stay */
   if first.staynum and last.staynum then do;
      output;
   end;

   else do;
      if first.staynum then do;
         lastfrom = from_dt;
         lastthru = thru_dt;
```

D#MSQ.MSQ00M7.PGMLIB99(C007) -- CONTINUED

```
         laststus = stus_cd;
         lastpamt = ippmtamt;
         do _i=1 to dim(oinds);
            linds{_i} = oinds{_i};
         end;
      end;
      else do;
         /* Set latest through date and status code */
         /* Take the last status code, even if through dates are equal */
         if thru_dt >= lastthru then do;
            lastthru = thru_dt;
            laststus = stus_cd;
         end;
         /* Sum payment amounts */
         lastpamt = sum(lastpamt, ippmtamt);
         do _i=1 to dim(oinds);
            if oinds{_i} = '1' then linds{_i} = '1';
         end;

         /* Reset the variables for the last claim, which will */
         /* be output to represent the inpatient stay.         */
         if last.staynum then do;
            from_dt = lastfrom;
            thru_dt = lastthru;
            stus_cd = laststus;
            if from_dt > 0 and thru_dt > 0 then LOS = thru_dt - from_dt;
```

```
            ippmtamt = lastpamt;

            output;
         end;
      end;
   end;

run;



 ************************************************************************;
 * STEP 14. Proc contents and proc print on the SAS IP stay file.       *;
 ************************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.IPSTY DIRECTORY ;
TITLE "IP Y&YEAR: STAY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.IPSTY (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR IP STAYS";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008DM)

```
//WF1708D JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/21/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY DME FILE FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C008DM.Y99DMPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006DM.Y99DMCLM,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008DM.Y99DMPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ******************************************************************;
 * STEP 1. sort DME file
 ******************************************************************;
PROC SORT DATA=IN1.Y&YEAR.DMCLM (DROP=LINEPMT HCFASPCL)  OUT=TEMP10;
    BY HICNO;
RUN;


 ******************************************************************;
 * STEP 2. merge DME file with cross-reference file to assign
           a unique hicno to each claimant
 ******************************************************************;
DATA TEMP20 (DROP=XREF);
```

```
  MERGE TEMP10        (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
  BY XREF;
```
D#MSQ.MSQ00M7.PGMLIB99(C008DM) -- CONTINUED

```
IF X AND Y;
RUN;


 ********************************************************************;
 * STEP 3. sort DME file again by the assigned unique HICNO
 *         and through service date
 ********************************************************************;
PROC SORT DATA=TEMP20   OUT=TEMP30;
    BY HICNO THRU_DT;
RUN;

PROC DELETE DATA=TEMP10 TEMP20;
RUN;

 ********************************************************************;
 * STEP 4. special treatment for i_lipid3: need to have i_lip3_1-i_lip3_3
 *         performed on the same date
 ********************************************************************;
DATA  TEMP40 (DROP=FROM_DT THRU_DT i_lip3_1-i_lip3_3 l_lip3_1-l_lip3_3);
  SET  TEMP30 (where=(i_lip3_1='1' or i_lip3_2='1' or i_lip3_3='1'));
by HICNO THRU_DT;
   attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';
   i_lipid3 = '0';

   length l_lip3_1-l_lip3_3   $ 1;
   retain l_lip3_1-l_lip3_3 ;

   array oinds {*} $   i_lip3_1-i_lip3_3 ;
   array linds {*} $   l_lip3_1-l_lip3_3 ;

   /* Initialize indicators */
   if first.THRU_DT then do;
      do _i=1 to dim(linds);
        linds{_i} = '0';
      end;
   end;

   /* pass information from i_lip3_1-i_lip3_3 to retain variables*/
   do _i=1 to dim(linds);
       if oinds{_i} ='1' then linds{_i} = '1';
   end;

if last.THRU_DT then do;
    /* create final i_lipid3 */
    if (l_lip3_1='1' and l_lip3_2='1' and l_lip3_3='1' )
       then i_lipid3='1';
end;
run;


 ********************************************************************;
 * STEP 5. create DME person summary file with measure flags
```

```
 ********************************************************************;
data OUT1.Y&YEAR.DMPR (drop= i _i
      i_hgl
      i_lipid1 i_lipid3 i_lip3_1-i_lip3_3
      i_self   i_educ1 i_eye1
      i_urine2 i_foot1 i_shoe
D#MSQ.MSQ00M7.PGMLIB99(C008DM) -- CONTINUED

      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc
);
   set TEMP30 (where=(i_lip3_1 not='1' and i_lip3_2 not='1' and i_lip3_3
not='1'
)
               drop=FROM_DT THRU_DT)
       TEMP40;
   by hicno;

  /* consolidate variables */
  if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

  /* set secondary DX='0' if primary DX ='1' */
   array pp {*} $
      i_led_p1
      i_led_p2
      i_neur_p
;
   array ss {*} $
      i_led_s1
      i_led_s2
      i_neur_s
;
  do i=1 to dim (ss);
       if pp{i} ='1' then ss{i} ='0';
  end;

   /* Person-level indicators */
   attrib p_hgl     length=2 label='Hgl A1c Test';
   attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
   attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';

   attrib p_self   length=2 label='Self-Monitoring';
   attrib p_educ1  length=2 label='Diabetes Education: definition 1';

   attrib p_eye1   length=2 label='Eye Examination 1';
   attrib  p_urine2  length=2  label='Microalbumin  Screening/Urine  Protein
2';
   attrib p_foot1  length=2 label='Podiatry visits 1';
   attrib p_shoe   length=2 label='Therapeutic Shoes';

   attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
   attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
   attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
   attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
```

```
      attrib p_led3    length=2 label='osteomyelitis';
      attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
      attrib p_led5    length=2 label='diabetes w periph cir disorder';
      attrib p_led6  length=2 label='Lower limb amputations';
      attrib p_led7  length=2 label='Incision and drainage below facia';
      attrib p_led8  length=2 label='Incision and drainage of bone cortex';
      attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

      attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
```

D#MSQ.MSQ00M7.PGMLIB99(C008DM) -- CONTINUED

```
      attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
      attrib p_neph    length=2 label='Nephropathy';
      attrib p_ret1    length=2 label='Retinopathy 1';

      attrib p_blind  length=2 label='Blindness';
      attrib p_catar  length=2 label='Diabetic cataract';

      attrib p_gla2   length=2 label='Glaucoma 2';
      attrib p_ede    length=2 label='Macular Edema';
      attrib p_las    length=2 label='Laser treatment';
      attrib p_ptc  length=2 label='Progression to PTCA, including stents';

      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self   length=5 label='Count:Self-Monitoring';
      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';

      attrib c_eye1    length=5 label='Count:Eye Examination 1';
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
      attrib c_foot1   length=5 label='Count:Podiatry visits 1';
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1   length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
      attrib c_led3    length=5 label='Count:osteomyelitis';
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7      length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8      length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9      length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
```

```
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';

      attrib  c_ptc      length=5  label='Count:Progress  to  PTCA,  including
stents';
      attrib c_total length=5 label='Count:total';
```

D#MSQ.MSQ00M7.PGMLIB99(C008DM) -- CONTINUED

```
      /* Array for existing measure flags*/
      array i_vars {*} $
         i_hgl
         i_lipid1 i_lipid3
         i_self   i_educ1 i_eye1
         i_urine2 i_foot1 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
         i_led4   i_led5
         i_led6 i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc
   ;
      /* Array for new person level measure flags*/
      array p_vars {*}
         p_hgl
         p_lipid1 p_lipid3
         p_self   p_educ1 p_eye1
         p_urine2 p_foot1 p_shoe
         p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
         p_led4   p_led5
         p_led6 p_led7 p_led8 p_led9
         p_neur_p p_neur_s  p_neph p_ret1
         p_blind p_catar p_gla2 p_ede p_las
         p_ptc
   ;
      /* Array for new person level count of claims for each measure*/
      array c_vars {*}
         c_hgl
         c_lipid1 c_lipid3
         c_self   c_educ1 c_eye1
         c_urine2 c_foot1 c_shoe
         c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
         c_led4   c_led5
         c_led6 c_led7 c_led8 c_led9
         c_neur_p c_neur_s  c_neph c_ret1
         c_blind c_catar c_gla2 c_ede c_las
         c_ptc
   ;
      retain
         p_hgl
         p_lipid1 p_lipid3
```

```
      p_self   p_educ1 p_eye1
      p_urine2 p_foot1 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc

      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_eye1
      c_urine2 c_foot1 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
```
D#MSQ.MSQ00M7.PGMLIB99(C008DM) -- CONTINUED
```
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc
      c_total
;

   /* Initialize indicators, counts */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
         c_vars{_i} = 0;
      end;

      c_total=0;
   end;

   /* create person-level indicators, counts */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
       end;
   end;

   /* Person-level total counts of claims */
    c_total=c_total + 1;

if last.hicno then output;
run;

proc freq;
table p_led9 /missing;
title 'test p_led9';
run;

 ********************************************************************;
 * STEP 6. Proc contents and proc print on the person summary file.
 ********************************************************************;
```

```
PROC CONTENTS DATA=OUT1.Y&YEAR.DMPR DIRECTORY ;
TITLE "DME Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.DMPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR DME PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008HH)

```
//WF1708H JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/21/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY HHA FILE FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C008HH.Y99HHPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006HH.Y99HHCLM,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008HH.Y99HHPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort HHA file
 ********************************************************************;
PROC SORT DATA=IN1.Y&YEAR.HHCLM (DROP=STUS_CD FROM_DT THRU_DT PMT_AMT )
    OUT=Y&YEAR.HHCLM;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge HHA file with cross-reference file to assign
           unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
```

```
  MERGE Y&YEAR.HHCLM        (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
  BY XREF;
```
D#MSQ.MSQ00M7.PGMLIB99(C008HH) --- CONTINUED

```
IF X AND Y;
RUN;


PROC DELETE DATA=Y&YEAR.HHCLM;
RUN;

 ********************************************************************;
 * STEP 3. sort HHA file again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. create HHA person summary file with measure flags attached
 ********************************************************************;
data OUT1.Y&YEAR.HHPR (drop= i _i
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc
);

   set TEMP2 ;
   by hicNO;

  /* consolidate variables */
  if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

  /* set secondary DX='0' if primary DX ='1' */
   array pp {*} $
      i_led_p1
      i_led_p2
      i_neur_p

;
   array ss {*} $
      i_led_s1
      i_led_s2
      i_neur_s

;
  do i=1 to dim (ss);
       if pp{i} ='1' then ss{i} ='0';
  end;
```

```
      /* Person-level indicators */
      attrib p_hgl     length=2 label='Hgl A1c Test';
      attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
      attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';
```



```
      attrib p_self   length=2 label='Self-Monitoring';
      attrib p_educ1  length=2 label='Diabetes Education: definition 1';
      attrib p_educ2  length=2 label='Diabetes Education: definition 2';

      attrib p_eye1   length=2 label='Eye Examination 1';
       attrib p_urine2  length=2 label='Microalbumin Screening/Urine Protein
2';
      attrib p_shoe    length=2 label='Therapeutic Shoes';

      attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
      attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
      attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
      attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
      attrib p_led3    length=2 label='osteomyelitis';
      attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
      attrib p_led5    length=2 label='diabetes w periph cir disorder';
      attrib p_led6  length=2 label='Lower limb amputations';
      attrib p_led7  length=2 label='Incision and drainage below facia';
      attrib p_led8  length=2 label='Incision and drainage of bone cortex';
      attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

      attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
      attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
      attrib p_neph    length=2 label='Nephropathy';
      attrib p_ret1    length=2 label='Retinopathy 1';

      attrib p_blind  length=2 label='Blindness';
      attrib p_catar  length=2 label='Diabetic cataract';

      attrib p_gla2   length=2 label='Glaucoma 2';
      attrib p_ede    length=2 label='Macular Edema';
      attrib p_las    length=2 label='Laser treatment';

      attrib p_ptc  length=2 label='Progression to PTCA, including stents';


      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self  length=5 label='Count:Self-Monitoring';
      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';
      attrib  c_educ2   length=5  label='Count:Diabetes  Education:  definition
2';

      attrib c_eye1   length=5 label='Count:Eye Examination 1';
```

```
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1   length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
```
D#MSQ.MSQ00M7.PGMLIB99(C008HH) --- CONTINUED
```
      attrib c_led3    length=5 label='Count:osteomyelitis';
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7       length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8       length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9       length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';

      attrib  c_ptc      length=5  label='Count:Progress  to  PTCA,  including
stents';

      attrib c_total length=5 label='Count:total';

      /* Array for existing measure flags*/
      array i_vars {*} $
         i_hgl
         i_lipid1 i_lipid3
         i_self   i_educ1 i_educ2 i_eye1
         i_urine2 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
         i_led4   i_led5
         i_led6 i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc

   ;
      /* Array for new person level measure flags*/
      array p_vars {*}
         p_hgl
         p_lipid1 p_lipid3
         p_self   p_educ1 p_educ2 p_eye1
```

```
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc

;
   /* Array for new person level count of claims for each measure*/
   array c_vars {*}
      c_hgl
      c_lipid1 c_lipid3
```
D#MSQ.MSQ00M7.PGMLIB99(C008HH) --- CONTINUED
```

      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc

;
   retain
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc


      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc

      c_total
;

   /* Initialize indicators, counts */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
```

```
         c_vars{_i} = 0;
      end;

      c_total=0;
   end;

   /* create person-level indicators, counts */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
       end;
   end;

   /* Person-level total counts of claims */
    c_total=c_total + 1;
```
D#MSQ.MSQ00M7.PGMLIB99(C008HH) --- CONTINUED

```
if last.hicno then output;

run;
proc freq;
table p_led9 /missing;
title 'test p_led9';
run;

 ********************************************************************;
 * STEP 5. Proc contents and proc print on the person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.HHPR DIRECTORY ;
TITLE "HHA Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.HHPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR HHA PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008IP)

```
//WF1708I JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/10/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY INPATIENT FILE FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C008IP.Y99IPPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(600,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C007.Y99IPSTY,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008IP.Y99IPPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;


 ********************************************************************;
 * STEP 1. merge Inpatient stay file with cross-reference file to assign
           unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
  MERGE IN1.Y&YEAR.IPSTY    (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
  BY XREF;
IF X AND Y;
RUN;
```

```
 ********************************************************************;
 * STEP 2. sort Inpatient file again by the assigned unique HICNO
 ********************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED

```
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 3. create Inpatient person summary file with measure flags
 ********************************************************************;
data OUT1.Y&YEAR.IPPR (drop= i _i
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_stk_p i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab i_ptc i_lea i_byp
      i_er_p i_er_s
      i_db1-i_db5);

   set TEMP2 ;
   by hicNO;

  /* if thru_dt = from_dt then count as 1 day */
  if (thru_dt-from_dt) =0 then los=1;

  /* consolidate variables */
  if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

  /* set secondary DX='0' if primary DX ='1' */
   array pp {*} $
      i_led_p1
      i_led_p2
      i_neur_p
      i_stk_p
      i_ami_p
      i_hrt_p
      i_er_p
;
   array ss {*} $
      i_led_s1
      i_led_s2
      i_neur_s
      i_stk_s
      i_ami_s
      i_hrt_s
      i_er_s
;
  do i=1 to dim (ss);
       if pp{i} ='1' then ss{i} ='0';
```

```
      end;

       /* Person-level indicators */
       attrib p_hgl     length=2 label='Hgl A1c Test';
       attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
       attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';
```
D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED

```
       attrib p_self   length=2 label='Self-Monitoring';
       attrib p_educ1  length=2 label='Diabetes Education: definition 1';
       attrib p_educ2  length=2 label='Diabetes Education: definition 2';

       attrib p_eye1   length=2 label='Eye Examination 1';
        attrib p_urine2   length=2 label='Microalbumin Screening/Urine Protein
2';
       attrib p_shoe    length=2 label='Therapeutic Shoes';

       attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
       attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
       attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
       attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
       attrib p_led3    length=2 label='osteomyelitis';
       attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
       attrib p_led5    length=2 label='diabetes w periph cir disorder';
       attrib p_led6  length=2 label='Lower limb amputations';
       attrib p_led7  length=2 label='Incision and drainage below facia';
       attrib p_led8  length=2 label='Incision and drainage of bone cortex';
       attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

       attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
       attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
       attrib p_neph    length=2 label='Nephropathy';
       attrib p_ret1    length=2 label='Retinopathy 1';

       attrib p_blind  length=2 label='Blindness';
       attrib p_catar  length=2 label='Diabetic cataract';

       attrib p_gla2   length=2 label='Glaucoma 2';
       attrib p_ede    length=2 label='Macular Edema';
       attrib p_las    length=2 label='Laser treatment';

       attrib  p_stk_p    length=2  label='Primary  DX:  Hospitalization  for
Stroke';
       attrib  p_stk_s    length=2  label='Secondary  DX:  Hospitalization  for
Stroke';
       attrib p_ami_p  length=2 label='Primary DX: Hospitalization for AMI';
       attrib p_ami_s  length=2 label='Secondary DX: Hospitalization for AMI';
       attrib p_hrt_p  length=2 label='Primary DX: Hospitali for Hrt Failure';
       attrib  p_hrt_s    length=2  label='Secondary  DX:  Hospitali  for  Hrt
Failure';

       attrib p_cab  length=2 label='Hospitalization for CABG';
       attrib p_ptc  length=2 label='Progression to PTCA, including stents';
       attrib p_lea  length=2 label='Hospitaliza for Lower extremity amputat';
       attrib   p_byp      length=2   label='Hospitaliza   for   femoral-bypass
procedure';
       attrib p_er_p  length=2 label='metabolic cris: ER or observ bed 1';
```

```
      attrib p_er_s  length=2 label='metabolic cris: ER or observ bed 2';

      attrib p_db1  length=2 label='Primary Hosp diabetes w/o complication';
      attrib p_db2  length=2 label='Primary Hosp diabetes w/ ketoacidosis';
      attrib   p_db3      length=2   label='Primary   Hosp   diabetes   w/
hyperosmolarity';
      attrib p_db4  length=2 label='Primary Hosp diabetes w/ other coma';
      attrib p_db5  length=2 label='Primary Hosp diabetes w/ other complica';
```

D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED

```
      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self  length=5 label='Count:Self-Monitoring';

      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';
      attrib  c_educ2   length=5  label='Count:Diabetes  Education:  definition
2';

      attrib c_eye1   length=5 label='Count:Eye Examination 1';
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1    length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
      attrib c_led3    length=5 label='Count:osteomyelitis';
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7       length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8       length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9       length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';
```

```
      attrib  c_stk_p    length=5  label='Count:Primary  DX:  Hospitaliz  for
Stroke';
      attrib  c_stk_s    length=5  label='Count:Secondary  DX:Hospitaliz  for
Stroke';
      attrib c_ami_p  length=5 label='Count:Primary DX: Hospitaliz for AMI';
      attrib  c_ami_s   length=5  label='Count:Secondary  DX:  Hospitaliz  for
AMI';
      attrib  c_hrt_p    length=5  label='Count:Primary  DX:  Hospit  for  Hrt
Failure';
      attrib  c_hrt_s    length=5  label='Count:Second  DX:  Hospit  for  Hrt
Failure';

      attrib c_cab  length=5 label='Count:Hospitalization for CABG';
```
D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED
```
      attrib  c_ptc    length=5  label='Count:Progress  to  PTCA,  including
stents';
      attrib c_lea  length=5 label='Count:Hospita for Lower extremity amput';
      attrib c_byp  length=5 label='Count:Hospita for femoral-bypass proced';
       attrib c_er_p  length=5 label='Count:metabolic cris: ER or observ bed
1';
       attrib c_er_s  length=5 label='Count:metabolic cris: ER or observ bed
2';

      attrib   c_db1      length=5   label='Count:Primary   Hosp   diab   w/o
complication';
      attrib c_db2  length=5 label='Count:Primary Hosp diab w/ ketoacidosis';
      attrib c_db3  length=5 label='Count:Primary Hosp diab w/ hyperosmolar';
      attrib c_db4  length=5 label='Count:Primary Hosp diab w/ other coma';
      attrib  c_db5    length=5  label='Count:Primary  Hosp  diab  w/  other
complic';

      attrib c_total  length=5 label='Count:Total';

      /* Person-level days (sum of LOS) */

      attrib l_stk_p  length=4 label='Day:Primary DX: Hospita for Stroke';
      attrib l_stk_s  length=4 label='Day:Secondary DX: Hospita for Stroke';
      attrib l_ami_p  length=4 label='Day:Primary DX: Hospita for AMI';
      attrib l_ami_s  length=4 label='Day:Secondary DX: Hospita for AMI';
      attrib  l_hrt_p    length=4  label='Day:Primary  DX:  Hospital  for  Hrt
Failure';
      attrib  l_hrt_s   length=4  label='Day:Secondary  DX:  Hospita  for  Hrt
Failur';

      attrib l_cab  length=4 label='Day:Hospitalization for CABG';
      attrib l_lea  length=4 label='Day:Hospita for Lower extremity amputat';
      attrib   l_byp      length=4   label='Day:Hospita   for   femoral-bypass
procedure';

      attrib l_db1  length=4 label='Day:Primary Hosp diab w/o complication';
      attrib l_db2  length=4 label='Day:Primary Hosp diab w/ ketoacidosis';
      attrib   l_db3       length=4    label='Day:Primary    Hosp    diab    w/
hyperosmolarity';
      attrib l_db4  length=4 label='Day:Primary Hosp diab w/ other coma';
      attrib l_db5  length=4 label='Day:Primary Hosp diab w/ other complica';

      attrib l_total  length=4 label='Day:Total';
```

```
   /* Array for existing measure flags*/
   array i_vars {*} $
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_stk_p i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab i_ptc i_lea i_byp
      i_er_p i_er_s
      i_db1-i_db5
D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED

;
   /* Array for new person level measure flags*/
   array p_vars {*}
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_stk_p p_stk_s p_ami_p p_ami_s  p_hrt_p p_hrt_s
      p_cab p_ptc p_lea p_byp
      p_er_p p_er_s
      p_db1-p_db5
;
   /* Array for new person level count of claims for each measure*/
   array c_vars {*}
      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_stk_p c_stk_s c_ami_p c_ami_s  c_hrt_p c_hrt_s
      c_cab c_ptc c_lea c_byp
      c_er_p c_er_s
      c_db1-c_db5
;
   /* Array for new person level days only for hospitalization measures*/
   array l_vars {*}
      l_stk_p l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab   l_lea l_byp
      l_db1-l_db5
;
   retain
```

```
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_stk_p p_stk_s p_ami_p p_ami_s  p_hrt_p p_hrt_s
      p_cab p_ptc p_lea p_byp
      p_er_p p_er_s
      p_db1-p_db5

      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
```
D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED
```
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_stk_p c_stk_s c_ami_p c_ami_s  c_hrt_p c_hrt_s
      c_cab c_ptc c_lea c_byp
      c_er_p c_er_s
      c_db1-c_db5
      c_total

      l_stk_p l_stk_s l_ami_p l_ami_s  l_hrt_p l_hrt_s
      l_cab   l_lea   l_byp
      l_db1-l_db5
      l_total
;

   /* Initialize indicators, counts, and days */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
         c_vars{_i} = 0;
      end;
      do _i=1 to dim(l_vars);
         l_vars{_i} = 0;
      end;

      c_total=0;
      l_total=0;
   end;

   /* create person-level indicators, counts of measures */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
```

```
       end;
   end;

   /* Array2 for existing measure flags*/
   array i_vars2 {*} $
      i_stk_p i_stk_s i_ami_p i_ami_s  i_hrt_p i_hrt_s
      i_cab i_lea i_byp
      i_db1-i_db5
;

   /* create person-level days only for hospitalization measures*/
   do _i=1 to dim(i_vars2);
       if i_vars2{_i} ='1' then do;
          l_vars{_i}  = l_vars{_i} + los;
       end;
   end;


   /* Person-level total of counts, and days */
    c_total=c_total + 1;
    l_total=l_total + los;
```

D#MSQ.MSQ00M7.PGMLIB99(C008IP) --- CONTINUED

```
if last.hicno then output;

run;


 ********************************************************************;
 * STEP 4. Proc contents and proc print on the person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.IPPR DIRECTORY ;
TITLE "IP Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.IPPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR IP PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008OP)

```
//WF1708O JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/21/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY OUTPATIENT FILE FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C008OP.Y99OPPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006OP.Y99OPCLM,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008OP.Y99OPPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort outpatient file
 ********************************************************************;
PROC SORT DATA=IN1.Y&YEAR.OPCLM (DROP=STUS_CD FROM_DT THRU_DT PMT_AMT )
    OUT=Y&YEAR.OPCLM;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge outpatient file with cross-reference file to assign
           unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
```

```
  MERGE Y&YEAR.OPCLM        (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
  BY XREF;
```
D#MSQ.MSQ00M7.PGMLIB99(C008OP) --- CONTINUED

```
IF X AND Y;
RUN;


PROC DELETE DATA=Y&YEAR.OPCLM;
RUN;

 ********************************************************************;
 * STEP 3. sort outpatient file again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. create outpatient person summary file with measure flags
 ********************************************************************;
data OUT1.Y&YEAR.OPPR (drop= i _i
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc    i_er_p i_er_s
);

   set TEMP2 ;
   by hicNO;

  /* consolidate variables */
  if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

  /* set secondary DX='0' if primary DX ='1' */
   array pp {*} $
      i_led_p1
      i_led_p2
      i_neur_p
      i_er_p
;
   array ss {*} $
      i_led_s1
      i_led_s2
      i_neur_s
      i_er_s
;
  do i=1 to dim (ss);
       if pp{i} ='1' then ss{i} ='0';
  end;
```

```
      /* Person-level indicators */
      attrib p_hgl     length=2 label='Hgl A1c Test';
      attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
      attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';
```



```
      attrib p_self   length=2 label='Self-Monitoring';
      attrib p_educ1  length=2 label='Diabetes Education: definition 1';
      attrib p_educ2  length=2 label='Diabetes Education: definition 2';

      attrib p_eye1   length=2 label='Eye Examination 1';
       attrib p_urine2  length=2 label='Microalbumin Screening/Urine Protein
2';
      attrib p_shoe    length=2 label='Therapeutic Shoes';

      attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
      attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
      attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
      attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
      attrib p_led3    length=2 label='osteomyelitis';
      attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
      attrib p_led5    length=2 label='diabetes w periph cir disorder';
      attrib p_led6  length=2 label='Lower limb amputations';
      attrib p_led7  length=2 label='Incision and drainage below facia';
      attrib p_led8  length=2 label='Incision and drainage of bone cortex';
      attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

      attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
      attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
      attrib p_neph    length=2 label='Nephropathy';
      attrib p_ret1    length=2 label='Retinopathy 1';

      attrib p_blind  length=2 label='Blindness';
      attrib p_catar  length=2 label='Diabetic cataract';

      attrib p_gla2   length=2 label='Glaucoma 2';
      attrib p_ede    length=2 label='Macular Edema';
      attrib p_las    length=2 label='Laser treatment';

      attrib p_ptc  length=2 label='Progression to PTCA, including stents';
      attrib p_er_p  length=2 label='metabolic cris: ER or observ bed 1';
      attrib p_er_s  length=2 label='metabolic cris: ER or observ bed 2';

      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self  length=5 label='Count:Self-Monitoring';
      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';
      attrib  c_educ2   length=5  label='Count:Diabetes  Education:  definition
2';

      attrib c_eye1   length=5 label='Count:Eye Examination 1';
```

```
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1   length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
```


```
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
      attrib c_led3    length=5 label='Count:osteomyelitis';
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7      length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8      length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9      length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';

      attrib  c_ptc      length=5  label='Count:Progress  to  PTCA,  including
stents';
       attrib c_er_p  length=5 label='Count:metabolic cris: ER or observ bed
1';
       attrib c_er_s  length=5 label='Count:metabolic cris: ER or observ bed
2';
      attrib c_total length=5 label='Count:total';

      /* Array for existing measure flags*/
      array i_vars {*} $
         i_hgl
         i_lipid1 i_lipid3
         i_self   i_educ1 i_educ2 i_eye1
         i_urine2 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
         i_led4   i_led5
         i_led6 i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc
         i_er_p i_er_s
   ;
      /* Array for new person level measure flags*/
      array p_vars {*}
```

```
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc
      p_er_p p_er_s
;
```
D#MSQ.MSQ00M7.PGMLIB99(C008OP) --- CONTINUED
```
   /* Array for new person level count of claims for each measure*/
   array c_vars {*}
      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc
      c_er_p c_er_s
;
   retain
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc
      p_er_p p_er_s

      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc
      c_er_p c_er_s
      c_total
;

   /* Initialize indicators, counts */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
```

```
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
         c_vars{_i} = 0;
      end;

      c_total=0;
   end;

   /* create person-level indicators, counts */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
       end;
```

```

   end;

   /* Person-level total counts of claims */
    c_total=c_total + 1;

if last.hicno then output;

run;

proc freq;
table p_led9 /missing;
title 'test p_led9';
run;
 ********************************************************************;
 * STEP 5. Proc contents and proc print on the person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.OPPR DIRECTORY ;
TITLE "OP Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.OPPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR OP PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008PS)

```
//WF1708P JOB (@MSQ0000,WF17),HONG,CLASS=H,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/21/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY PART B PHYSICIAN FILE FOR
//*            PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C008PS.Y99PSPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(900,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006PS.Y99PSCLM,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008PS.Y99PSPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//WORK10 DD DSN=WF17.@MSQ0000.SAS.TEMP10,
//       UNIT=(MCART,,DEFER),
//       DISP=(NEW,DELETE,DELETE)
//WORK20 DD DSN=WF17.@MSQ0000.SAS.TEMP20,
//       UNIT=(MCART,,DEFER),
//       DISP=(NEW,DELETE,DELETE)
//WORK30 DD DSN=WF17.@MSQ0000.SAS.TEMP30,
//       UNIT=(MCART,,DEFER),
//       DISP=(NEW,DELETE,DELETE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort part B physician file
 ********************************************************************;
```

```
    PROC      SORT      DATA=IN1.Y&YEAR.PSCLM      (DROP=LINEPMT      HCFASPCL)
OUT=WORK10.TEMP10;
        BY HICNO;
    RUN;
```
D#MSQ.MSQ00M7.PGMLIB99(C008PS) --- CONTINUED

```
     ********************************************************************;
     * STEP 2. merge part B physician file with cross-reference file to assign
     *         a unique hicno to each claimant
     ********************************************************************;
    DATA WORK20.TEMP20 (DROP=XREF);
      MERGE WORK10.TEMP10        (IN=X RENAME=(HICNO=XREF))
            IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
      BY XREF;
    IF X AND Y;
    RUN;


     ********************************************************************;
     * STEP 3. sort part B physician file again by the assigned unique HICNO
     *         and through service date
     ********************************************************************;
    PROC SORT DATA=WORK20.TEMP20   OUT=WORK30.TEMP30;
        BY HICNO THRU_DT;
    RUN;


     ********************************************************************;
     * STEP 4. special treatment for i_lipid3: need to have i_lip3_1-i_lip3_3
     *         performed on the same date
     ********************************************************************;
    DATA  TEMP40 (DROP=FROM_DT THRU_DT i_lip3_1-i_lip3_3 l_lip3_1-l_lip3_3);
      SET       WORK30.TEMP30   (where=(i_lip3_1='1'    or    i_lip3_2='1'    or
i_lip3_3='1'));
    by HICNO THRU_DT;
       attrib i_lipid3  length=$1 label='Lipid Profiling: definition 3';
       i_lipid3 = '0';

       length l_lip3_1-l_lip3_3   $ 1;
       retain l_lip3_1-l_lip3_3 ;

       array oinds {*} $   i_lip3_1-i_lip3_3 ;
       array linds {*} $   l_lip3_1-l_lip3_3 ;

       /* Initialize indicators */
       if first.THRU_DT then do;
          do _i=1 to dim(linds);
            linds{_i} = '0';
          end;
       end;

       /* pass information from i_lip3_1-i_lip3_3 to retain variables*/
       do _i=1 to dim(linds);
           if oinds{_i} ='1' then linds{_i} = '1';
       end;
```

```
    if last.THRU_DT then do;
        /* create final i_lipid3 */
        if (l_lip3_1='1' and l_lip3_2='1' and l_lip3_3='1' )
           then i_lipid3='1';
    end;
    run;
```

D#MSQ.MSQ00M7.PGMLIB99(C008PS) --- CONTINUED

```
     ********************************************************************;
     * STEP 5. create part B physician person summary file with measure flags
     ********************************************************************;
    data OUT1.Y&YEAR.PSPR (drop= i _i
          i_hgl
          i_lipid1 i_lipid3 i_lip3_1-i_lip3_3
          i_self   i_educ1 i_eye1
          i_urine2 i_foot1 i_shoe
          i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
          i_led4   i_led5
          i_led6 i_led7 i_led8 i_led9
          i_neur_p i_neur_s  i_neph i_ret1
          i_blind i_catar i_gla2 i_ede i_las
          i_ptc
    );
       set  WORK30.TEMP30  (where=(i_lip3_1  not='1'  and  i_lip3_2  not='1'  and
i_lip3_3
    not='1')
                   drop=FROM_DT THRU_DT)
           TEMP40;
       by hicno;

      /* consolidate variables */
      if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

      /* set secondary DX='0' if primary DX ='1' */
       array pp {*} $
          i_led_p1
          i_led_p2
          i_neur_p
    ;
       array ss {*} $
          i_led_s1
          i_led_s2
          i_neur_s
    ;
      do i=1 to dim (ss);
           if pp{i} ='1' then ss{i} ='0';
      end;

       /* Person-level indicators */
       attrib p_hgl     length=2 label='Hgl A1c Test';
       attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
       attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';

       attrib p_self   length=2 label='Self-Monitoring';
       attrib p_educ1  length=2 label='Diabetes Education: definition 1';
```

```
      attrib p_eye1   length=2 label='Eye Examination 1';
      attrib  p_urine2  length=2  label='Microalbumin  Screening/Urine  Protein
2';
      attrib p_foot1  length=2 label='Podiatry visits 1';
      attrib p_shoe   length=2 label='Therapeutic Shoes';

      attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
      attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
      attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
      attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
```

D#MSQ.MSQ00M7.PGMLIB99(C008PS) --- CONTINUED

```
      attrib p_led3    length=2 label='osteomyelitis';
      attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
      attrib p_led5    length=2 label='diabetes w periph cir disorder';
      attrib p_led6  length=2 label='Lower limb amputations';
      attrib p_led7  length=2 label='Incision and drainage below facia';
      attrib p_led8  length=2 label='Incision and drainage of bone cortex';
      attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

      attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
      attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
      attrib p_neph    length=2 label='Nephropathy';
      attrib p_ret1    length=2 label='Retinopathy 1';

      attrib p_blind  length=2 label='Blindness';
      attrib p_catar  length=2 label='Diabetic cataract';

      attrib p_gla2   length=2 label='Glaucoma 2';
      attrib p_ede    length=2 label='Macular Edema';
      attrib p_las    length=2 label='Laser treatment';
      attrib p_ptc  length=2 label='Progression to PTCA, including stents';

      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self   length=5 label='Count:Self-Monitoring';
      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';

      attrib c_eye1    length=5 label='Count:Eye Examination 1';
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
      attrib c_foot1   length=5 label='Count:Podiatry visits 1';
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1   length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
      attrib c_led3    length=5 label='Count:osteomyelitis';
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
```

```
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7       length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8       length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9       length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
```
D#MSQ.MSQ00M7.PGMLIB99(C008PS) --- CONTINUED
```

      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';

      attrib  c_ptc      length=5  label='Count:Progress  to  PTCA,  including
stents';
      attrib c_total length=5 label='Count:total';

      /* Array for existing measure flags*/
      array i_vars {*} $
         i_hgl
         i_lipid1 i_lipid3
         i_self   i_educ1 i_eye1
         i_urine2 i_foot1 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
         i_led4   i_led5
         i_led6 i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc
   ;
      /* Array for new person level measure flags*/
      array p_vars {*}
         p_hgl
         p_lipid1 p_lipid3
         p_self   p_educ1 p_eye1
         p_urine2 p_foot1 p_shoe
         p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
         p_led4   p_led5
         p_led6 p_led7 p_led8 p_led9
         p_neur_p p_neur_s  p_neph p_ret1
         p_blind p_catar p_gla2 p_ede p_las
         p_ptc
   ;
      /* Array for new person level count of claims for each measure*/
      array c_vars {*}
         c_hgl
         c_lipid1 c_lipid3
         c_self   c_educ1 c_eye1
         c_urine2 c_foot1 c_shoe
         c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
```

```
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc
;
   retain
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_eye1
      p_urine2 p_foot1 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
D#MSQ.MSQ00M7.PGMLIB99(C008PS) --- CONTINUED

      p_ptc

      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_eye1
      c_urine2 c_foot1 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc
      c_total
;

   /* Initialize indicators, counts */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
         c_vars{_i} = 0;
      end;

      c_total=0;
   end;

   /* create person-level indicators, counts */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
       end;
   end;

   /* Person-level total counts of claims */
    c_total=c_total + 1;

if last.hicno then output;
run;
```

```
proc freq;
table p_led9 /missing;
title 'test p_led9';
run;

 ********************************************************************;
 * STEP 6. Proc contents and proc print on the person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.PSPR DIRECTORY ;
TITLE "PS Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.PSPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR PS PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C008SN)

```
//WF1708S JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/21/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE PERSON-LEVEL SUMMARY SNF FILE FOR
//*             PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//**STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//**IF THE OLD DATASET EXIST
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C008SN.Y99SNPR
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C006SN.Y99SNCLM,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C008SN.Y99SNPR,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen;


%LET YEAR=99;  *** YEAR OF PRODUCTION;

 ********************************************************************;
 * STEP 1. sort SNF file
 ********************************************************************;
PROC SORT DATA=IN1.Y&YEAR.SNCLM (DROP=STUS_CD FROM_DT THRU_DT PMT_AMT)
    OUT=Y&YEAR.SNCLM;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 2. merge SNF file with cross-reference file to assign
           unique hicno to each claimant
 ********************************************************************;
DATA TEMP2 (DROP=XREF);
```

```
  MERGE Y&YEAR.SNCLM        (IN=X RENAME=(HICNO=XREF))
        IN2.XREF&YEAR       (IN=Y KEEP=HICNO XREF);
  BY XREF;
```
D#MSQ.MSQ00M7.PGMLIB99(C008SN) --- CONTINUED

```
IF X AND Y;
RUN;


PROC DELETE DATA=Y&YEAR.SNCLM;
RUN;

 ********************************************************************;
 * STEP 3. sort SNF again by the assigned unique HICNO
 ********************************************************************;
PROC SORT DATA=TEMP2;
    BY HICNO;
RUN;


 ********************************************************************;
 * STEP 4. create SNF person summary file with measure flags attached
 ********************************************************************;
data OUT1.Y&YEAR.SNPR (drop= i _i
      i_hgl
      i_lipid1 i_lipid3
      i_self   i_educ1 i_educ2 i_eye1
      i_urine2 i_shoe
      i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led_p3 i_led_s3 i_led3
      i_led4   i_led5
      i_led6 i_led7 i_led8 i_led9
      i_neur_p i_neur_s  i_neph i_ret1
      i_blind i_catar i_gla2 i_ede i_las
      i_ptc
);

   set TEMP2 ;
   by hicNO;

  /* consolidate variables */
  if (i_led_s3='1' or i_led_p3='1') then i_led3='1'; else i_led3 ='0';

  /* set secondary DX='0' if primary DX ='1' */
   array pp {*} $
      i_led_p1
      i_led_p2
      i_neur_p

;
   array ss {*} $
      i_led_s1
      i_led_s2
      i_neur_s

;
  do i=1 to dim (ss);
       if pp{i} ='1' then ss{i} ='0';
  end;
```

```
      /* Person-level indicators */
      attrib p_hgl     length=2 label='Hgl A1c Test';
      attrib p_lipid1  length=2 label='Lipid Profiling: definition 1';
      attrib p_lipid3  length=2 label='Lipid Profiling: definition 3';
```



```
      attrib p_self   length=2 label='Self-Monitoring';
      attrib p_educ1  length=2 label='Diabetes Education: definition 1';
      attrib p_educ2  length=2 label='Diabetes Education: definition 2';

      attrib p_eye1   length=2 label='Eye Examination 1';
       attrib p_urine2  length=2 label='Microalbumin Screening/Urine Protein
2';
      attrib p_shoe    length=2 label='Therapeutic Shoes';

      attrib p_led_p1  length=2 label='Primary DX: cellulitis of LED';
      attrib p_led_s1  length=2 label='Secondary DX: cellulitis of LEDs';
      attrib p_led_p2  length=2 label='Primary DX: cutaneous gangrene';
      attrib p_led_s2  length=2 label='Secondary DX: cutaneous gangrene';
      attrib p_led3    length=2 label='osteomyelitis';
      attrib p_led4    length=2 label='Arthropathy w/ Nerve Dis';
      attrib p_led5    length=2 label='diabetes w periph cir disorder';
      attrib p_led6  length=2 label='Lower limb amputations';
      attrib p_led7  length=2 label='Incision and drainage below facia';
      attrib p_led8  length=2 label='Incision and drainage of bone cortex';
      attrib p_led9  length=2 label='Skin and Subcutan debride for gangrene';

      attrib p_neur_p  length=2 label='Primary DX: Peripheral neuropathy';
      attrib p_neur_s  length=2 label='Secondary DX: Peripheral neuropathy';
      attrib p_neph    length=2 label='Nephropathy';
      attrib p_ret1    length=2 label='Retinopathy 1';

      attrib p_blind  length=2 label='Blindness';
      attrib p_catar  length=2 label='Diabetic cataract';

      attrib p_gla2   length=2 label='Glaucoma 2';
      attrib p_ede    length=2 label='Macular Edema';
      attrib p_las    length=2 label='Laser treatment';

      attrib p_ptc  length=2 label='Progression to PTCA, including stents';

      /* Person-level counts */

      attrib c_hgl     length=5 label='Count:Hgl A1c Test';
      attrib c_lipid1  length=5 label='Count:Lipid Profiling: definition 1';
      attrib c_lipid3  length=5 label='Count:Lipid Profiling: definition 3';

      attrib c_self  length=5 label='Count:Self-Monitoring';
      attrib  c_educ1   length=5  label='Count:Diabetes  Education:  definition
1';
      attrib  c_educ2   length=5  label='Count:Diabetes  Education:  definition
2';

      attrib c_eye1   length=5 label='Count:Eye Examination 1';
      attrib   c_urine2     length=5   label='Count:Microalbum   Scrning/Urine
Protein2';
```

```
      attrib c_shoe    length=5 label='Count:Therapeutic Shoes';

      attrib c_led_p1  length=5 label='Count:Primary DX: cellulitis of LED';
      attrib  c_led_s1   length=5  label='Count:Secondary  DX:  cellulitis  of
LEDs';
      attrib c_led_p2  length=5 label='Count:Primary DX: cutaneous gangrene';
      attrib   c_led_s2     length=5   label='Count:Secondary   DX:   cutaneous
gangrene';
      attrib c_led3    length=5 label='Count:osteomyelitis';
```
D#MSQ.MSQ00M7.PGMLIB99(C008SN) --- CONTINUED
```
      attrib c_led4    length=5 label='Count:Arthrop w/ Nerve Dis';
      attrib c_led5    length=5 label='Count:diab w periph cir disorder';
      attrib c_led6    length=5 label='Count:Lower limb amputations';
      attrib  c_led7       length=5  label='Count:Incision  and  drainage  below
facia';
      attrib  c_led8       length=5  label='Count:Incision  and  drain  of  bone
cortex';
      attrib  c_led9       length=5  label='Count:Skin  and  Subcuta  debri  for
gangren';

      attrib c_neur_p  length=5 label='Count:Primary DX: Periph neuropathy';
      attrib   c_neur_s      length=5   label='Count:Secondary   DX:   Periph
neuropathy';
      attrib c_neph    length=5 label='Count:Nephropathy';
      attrib c_ret1    length=5 label='Count:Retinopathy 1';
      attrib c_blind  length=5 label='Count:Blindness';
      attrib c_catar  length=5 label='Count:Diabetic cataract';
      attrib c_gla2   length=5 label='Count:Glaucoma 2';
      attrib c_ede    length=5 label='Count:Macular Edema';
      attrib c_las    length=5 label='Count:Laser treatment';

      attrib  c_ptc      length=5  label='Count:Progress  to  PTCA,  including
stents';


      attrib c_total length=5 label='Count:total';

      /* Array for existing measure flags*/
      array i_vars {*} $
         i_hgl
         i_lipid1 i_lipid3
         i_self   i_educ1 i_educ2 i_eye1
         i_urine2 i_shoe
         i_led_p1 i_led_s1 i_led_p2 i_led_s2 i_led3
         i_led4   i_led5
         i_led6 i_led7 i_led8 i_led9
         i_neur_p i_neur_s  i_neph i_ret1
         i_blind i_catar i_gla2 i_ede i_las
         i_ptc;

      /* Array for new person level measure flags*/
      array p_vars {*}
         p_hgl
         p_lipid1 p_lipid3
         p_self   p_educ1 p_educ2 p_eye1
         p_urine2 p_shoe
         p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
```

```
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc

;
   /* Array for new person level count of claims for each measure*/
   array c_vars {*}
      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
D#MSQ.MSQ00M7.PGMLIB99(C008SN) --- CONTINUED

      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc

;
   retain
      p_hgl
      p_lipid1 p_lipid3
      p_self   p_educ1 p_educ2 p_eye1
      p_urine2 p_shoe
      p_led_p1 p_led_s1 p_led_p2 p_led_s2 p_led3
      p_led4   p_led5
      p_led6 p_led7 p_led8 p_led9
      p_neur_p p_neur_s  p_neph p_ret1
      p_blind p_catar p_gla2 p_ede p_las
      p_ptc


      c_hgl
      c_lipid1 c_lipid3
      c_self   c_educ1 c_educ2 c_eye1
      c_urine2 c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led_s2 c_led3
      c_led4   c_led5
      c_led6 c_led7 c_led8 c_led9
      c_neur_p c_neur_s  c_neph c_ret1
      c_blind c_catar c_gla2 c_ede c_las
      c_ptc

      c_total
;

   /* Initialize indicators, counts */
   if first.hicno then do;
      do _i=1 to dim(p_vars);
         p_vars{_i} = 0;
      end;
      do _i=1 to dim(c_vars);
         c_vars{_i} = 0;
      end;
```

```
      c_total=0;
   end;

   /* create person-level indicators, counts */
   do _i=1 to dim(i_vars);
       if i_vars{_i} ='1' then do;
           p_vars{_i} = 1;
           c_vars{_i} = c_vars{_i} + 1;
       end;
   end;

   /* Person-level total counts of claims */
    c_total=c_total + 1;
```

D#MSQ.MSQ00M7.PGMLIB99(C008SN) --- CONTINUED

```
if last.hicno then output;

run;

proc freq;
table p_led9 /missing;
title 'test p_led9';
run;
 ********************************************************************;
 * STEP 5. Proc contents and proc print on the person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.SNPR DIRECTORY ;
TITLE "SNF Y&YEAR: PERSON SUMMARY FILE WITH MEASURE FLAGS";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.SNPR (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR SNF PERSON SUMMARY FILE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C009)

```
//WF17009 JOB (@MSQ0000,WF17),HONG,CLASS=A,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/18/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2, IN3, IN4, IN5, IN6
//* OUTPUTS   :OUT1
//* PURPOSE   :CREATE INITIAL PERSON-LEVEL SUMMARY FILE BY COMBINING
//*            PERSON SUMMARY FILES OF IP, SNF, OP, HHA, PS, DME FOR
//*            PROCESS AND PROGRESSION OF DISEASE MEASURES
//*********************************************************************
//*
//* IF THE OLD DATASET EXIST
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
    DELETE D#MSQ.@MSQ00M7.SAS.C009.Y99PRALL
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,300))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C008IP.Y99IPPR,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ00M7.SAS.C008SN.Y99SNPR,DISP=SHR
//IN3    DD DSN=D#MSQ.@MSQ00M7.SAS.C008OP.Y99OPPR,DISP=SHR
//IN4    DD DSN=D#MSQ.@MSQ00M7.SAS.C008HH.Y99HHPR,DISP=SHR
//IN5    DD DSN=D#MSQ.@MSQ00M7.SAS.C008PS.Y99PSPR,DISP=SHR
//IN6    DD DSN=D#MSQ.@MSQ00M7.SAS.C008DM.Y99DMPR,DISP=SHR
//IN7    DD DSN=D#MSQ.@MSQ00M7.FLAT.B005.DIAB99,DISP=SHR
//IN8    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C009.Y99PRALL,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nocenter pageno=1 ps=55 ls=80 symbolgen OBS=MAX;

%LET YEAR=99;  *** YEAR OF PRODUCTION;


 ********************************************************************;
 * STEP 1. create final person summary file by combining person summary
 *         files of IP, SNF, OP, HHS, PS, and DME into a single file
 ********************************************************************;

data Y&YEAR.PRALL(keep= y&year.hicno
```

```
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
D#MSQ.MSQ00M7.PGMLIB99(C009) --- CONTINUED

      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5

      c&year.hgl    c&year.lip
      c&year.self   c&year.educ  c&year.eye
      c&year.urin   c&year.foot  c&year.shoe
      c&year.ledp1  c&year.leds1 c&year.led2 c&year.led3
      c&year.led4   c&year.led5
      c&year.led6   c&year.led7  c&year.led8 c&year.led9
      c&year.neph   c&year.ret
      c&year.blind  c&year.catar c&year.gla c&year.ede c&year.las
      c&year.stk_p  c&year.stk_s c&year.ami_p c&year.ami_s
      c&year.hrt_p c&year.hrt_s
      c&year.cab    c&year.ptc   c&year.lea c&year.byp
      c&year.er_p
      c&year.db1-c&year.db5

      c&year.total
);
   set
      IN1.Y&YEAR.IPPR  (in=in1 rename=(hicno=y&year.hicno) drop=p_educ2)
      IN2.Y&YEAR.SNPR  (in=in2 rename=(hicno=y&year.hicno))
      IN3.Y&YEAR.OPPR  (in=in3 rename=(hicno=y&year.hicno))
      IN4.Y&YEAR.HHPR  (in=in4 rename=(hicno=y&year.hicno))
      IN5.Y&YEAR.PSPR  (in=in5 rename=(hicno=y&year.hicno))
      IN6.Y&YEAR.DMPR  (in=in6 rename=(hicno=y&year.hicno))
;
   by y&year.hicno;

    /* reset following as: primary DX="any of primary or secondary DX" */
  if  p_led_s2=1  then do;
      p_led_p2=1;
      c_led_p2=sum(c_led_p2, c_led_s2, 0);
  end;
  if  p_neur_s=1  then do;
      p_neur_p=1;
      c_neur_p=sum(c_neur_p, c_neur_s, 0);
  end;
  if  p_er_s=1    then do;
      p_er_p=1;
      c_er_p=sum(c_er_p, c_er_s, 0);
  end;

  /* reset following as: first indicator="any of those included" */
  if p_lipid3=1 then do;
```

```
        p_lipid1=1;
        c_lipid1=sum(c_lipid1, c_lipid3, 0);
      end;
      if p_educ2=1 then do;
        p_educ1=1;
        c_educ1=sum(c_educ1, c_educ2, 0);
```
D#MSQ.MSQ00M7.PGMLIB99(C009) --- CONTINUED
```

      end;
      if p_neur_p=1 then do;
        p_led4=1;
        c_led4=sum(c_led4, c_neur_p, 0);
      end;


        /* Person-level indicators */
       attrib y&year.hgl   length=2 label='Hgl A1c Test';
       attrib y&year.lip   length=2 label='Lipid Profiling';

       attrib y&year.self  length=2 label='Self-Monitoring';
       attrib y&year.educ  length=2 label='Diabetes Education';

       attrib y&year.eye   length=2 label='Eye Examination';
       attrib  y&year.urin    length=2  label='Microalbumin  Screening/Urine
Protein';
       attrib y&year.foot  length=2 label='Podiatry visit';
       attrib y&year.shoe  length=2 label='Therapeutic Shoes';

       attrib y&year.ledp1  length=2 label='Primary DX: cellulitis of LED';
       attrib y&year.leds1  length=2 label='Secondary DX: cellulitis of LEDs';
       attrib y&year.led2   length=2 label='Cutaneous gangrene';
       attrib y&year.led3   length=2 label='Osteomyelitis';
        attrib y&year.led4    length=2 label='Periph neuropathy or Arthrop w
Nerve Dis
    ';
       attrib  y&year.led5    length=2  label='Diabetes w periph circulatory
disorder';
       attrib y&year.led6   length=2 label='Lower limb amputations';
       attrib  y&year.led7     length=2  label='Incision  and  drainage  below
facia';
       attrib  y&year.led8    length=2  label='Incision and drainage of bone
cortex';
       attrib  y&year.led9    length=2  label='Skin  and  Subcutan  debride  for
gangrene';

       attrib y&year.neph   length=2 label='Nephropathy';
       attrib y&year.ret    length=2 label='Retinopathy';
       attrib y&year.blind  length=2 label='Blindness';
       attrib y&year.catar  length=2 label='Diabetic cataract';

       attrib y&year.gla    length=2 label='Glaucoma';
       attrib y&year.ede    length=2 label='Macular Edema';
       attrib y&year.las    length=2 label='Laser treatment';

       attrib  y&year.ptc    length=2  label='Progression  to  PTCA,  including
stents';
       attrib y&year.er_p  length=2 label='metabolic cris: ER or observ bed';
```

```
        /* indicators for IP only */
       attrib y&year.stk_p   length=2 label='Primary DX: Hospitalization for
Stroke';
      attrib y&year.stk_s   length=2 label='Secondary DX: Hospitalization for
Stroke
    ';
       attrib y&year.ami_p   length=2 label='Primary DX: Hospitalization for
AMI';
```



```
       attrib y&year.ami_s   length=2 label='Secondary DX: Hospitalization for
AMI';
      attrib  y&year.hrt_p   length=2  label='Primary  DX:  Hospitali  for  Hrt
Failure';
       attrib y&year.hrt_s   length=2 label='Secondary DX: Hospitali for Hrt
Failure'
    ;

      attrib y&year.cab   length=2 label='Hospitalization for CABG';
       attrib y&year.lea    length=2  label='Hospitaliza  for  Lower  extremity
amputat';
      attrib  y&year.byp     length=2  label='Hospitaliza  for  femoral-bypass
procedure'
    ;

      attrib   y&year.db1     length=2   label='Primary   DX:Hosp   diab   w/o
complication';
      attrib   y&year.db2     length=2   label='Primary   DX:Hosp   diab   w/
ketoacidosis';
      attrib   y&year.db3     length=2   label='Primary   DX:Hosp   diab   w/
hyperosmolarity';
      attrib y&year.db4  length=2 label='Primary DX:Hosp diab w/ other coma';
      attrib  y&year.db5    length=2  label='Primary  DX:Hosp  diab  w/  other
complica';

        /* Person-level counts  */
      attrib c&year.hgl    length=8 label='Count:Hgl A1c Test';
      attrib c&year.lip    length=8 label='Count:Lipid Profiling';

      attrib c&year.self   length=8 label='Count:Self-Monitoring';
      attrib c&year.educ   length=8 label='Count:Diabetes Education';

      attrib c&year.eye    length=8 label='Count:Eye Examination';
      attrib  c&year.urin     length=8  label='Count:Microalbumin  Screen/Urine
Protein'
    ;
      attrib c&year.foot   length=8 label='Count:Podiatry visit';
      attrib c&year.shoe   length=8 label='Count:Therapeutic Shoes';

       attrib c&year.ledp1   length=8 label='Count:Primary DX: cellulitis of
LED';
       attrib c&year.leds1  length=8 label='Count:Secondary DX: cellulitis of
LEDs';
      attrib c&year.led2   length=8 label='Count:cutaneous gangrene';
      attrib c&year.led3   length=8 label='Count:osteomyelitis';
       attrib c&year.led4    length=8 label='Count:Periph neurop /Arthrop w
Nerve Dis
```

```
    ';
       attrib c&year.led5   length=8 label='Count:diabetes w periph cir dis';
       attrib c&year.led6   length=8 label='Count:Lower limb amputations';
       attrib c&year.led7   length=8 label='Count:Incision and drainage below
facia'
    ;
       attrib c&year.led8   length=8 label='Count:Incision & drainage of bone
cortex
    ';
       attrib c&year.led9   length=8 label='Count:Skin & Subcutan debride for
gangre
    ';
```
D#MSQ.MSQ00M7.PGMLIB99(C009) --- CONTINUED

```
       attrib c&year.neph   length=8 label='Count:Nephropathy';
       attrib c&year.ret    length=8 label='Count:Retinopathy';

       attrib c&year.blind  length=8 label='Count:Blindness';
       attrib c&year.catar  length=8 label='Count:Diabetic cataract';

       attrib c&year.gla    length=8 label='Count:Glaucoma';
       attrib c&year.ede    length=8 label='Count:Macular Edema';
       attrib c&year.las    length=8 label='Count:Laser treatment';

       attrib c&year.ptc    length=8 label='Count:Progress to PTCA, including
stents
    ';
       attrib c&year.er_p   length=8 label='Count:metabolic cris: ER or observ
bed';

         /* Below are admissions: for IP only */
       attrib  c&year.stk_p   length=8  label='Count:Primary  DX:  Hospital  for
Stroke';
       attrib c&year.stk_s  length=8 label='Count:Secondary DX: Hospital for
Stroke'
    ;
       attrib  c&year.ami_p   length=8  label='Count:Primary  DX:  Hospital  for
AMI';
       attrib c&year.ami_s  length=8 label='Count:Secondary DX: Hospital for
AMI';
       attrib c&year.hrt_p  length=8 label='Count:Primary DX: Hospi for Hrt
Failure'
    ;
       attrib c&year.hrt_s  length=8 label='Count:Secondary DX: Hospi for Hrt
Failur
    ';

       attrib c&year.cab  length=8 label='Count:Hospitalization for CABG';
       attrib  c&year.lea   length=8  label='Count:Hospi  for  Lower  extremity
amputat';
       attrib  c&year.byp    length=8  label='Count:Hospi  for  femoral-bypass
procedure';

       attrib   c&year.db1      length=8   label='Count:PDX:Hosp   diab   w/o
complication';
       attrib   c&year.db2      length=8   label='Count:PDX:Hosp   diab   w/
ketoacidosis';
```

```
      attrib   c&year.db3      length=8   label='Count:PDX:Hosp   diab   w/
hyperosmolari';
      attrib c&year.db4  length=8 label='Count:PDX:Hosp diab w/ other coma';
      attrib  c&year.db5    length=8  label='Count:PDX:Hosp  diab  w/  other
complic';

      attrib c&year.total  length=8 label='Count: total';

      /* New Indicators */
      array f_new {*}
         y&year.hgl    y&year.lip
         y&year.self   y&year.educ  y&year.eye
         y&year.urin   y&year.foot  y&year.shoe
         y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
         y&year.led4   y&year.led5
```



```
         y&year.led6   y&year.led7  y&year.led8 y&year.led9
         y&year.neph   y&year.ret
         y&year.blind  y&year.catar y&year.gla y&year.ede y&year.las
         y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
         y&year.hrt_p  y&year.hrt_s
         y&year.cab    y&year.ptc   y&year.lea y&year.byp
         y&year.er_p
         y&year.db1-y&year.db5
   ;


      /* New Counts */
      array t_new {*}
         c&year.hgl    c&year.lip
         c&year.self   c&year.educ  c&year.eye
         c&year.urin   c&year.foot  c&year.shoe
         c&year.ledp1  c&year.leds1 c&year.led2 c&year.led3
         c&year.led4   c&year.led5
         c&year.led6   c&year.led7  c&year.led8 c&year.led9
         c&year.neph   c&year.ret
         c&year.blind  c&year.catar c&year.gla c&year.ede c&year.las
         c&year.stk_p  c&year.stk_s c&year.ami_p c&year.ami_s
         c&year.hrt_p  c&year.hrt_s
         c&year.cab    c&year.ptc   c&year.lea c&year.byp
         c&year.er_p
         c&year.db1-c&year.db5
   ;

      /* Old Indicators */
      array f_vars {*}
         p_hgl    p_lipid1
         p_self   p_educ1  p_eye1
         p_urine2 p_foot1  p_shoe
         p_led_p1 p_led_s1 p_led_p2 p_led3
         p_led4   p_led5
         p_led6   p_led7   p_led8 p_led9
         p_neph   p_ret1
         p_blind  p_catar  p_gla2 p_ede p_las
         p_stk_p  p_stk_s  p_ami_p p_ami_s  p_hrt_p p_hrt_s
         p_cab    p_ptc    p_lea p_byp
         p_er_p
```

```
      p_db1-p_db5
;

   /* Old Counts */
   array t_vars {*}
      c_hgl    c_lipid1
      c_self   c_educ1  c_eye1
      c_urine2 c_foot1  c_shoe
      c_led_p1 c_led_s1 c_led_p2 c_led3
      c_led4   c_led5
      c_led6   c_led7   c_led8 c_led9
      c_neph   c_ret1
      c_blind  c_catar  c_gla2 c_ede c_las
      c_stk_p  c_stk_s  c_ami_p c_ami_s  c_hrt_p c_hrt_s
      c_cab    c_ptc    c_lea c_byp
      c_er_p
      c_db1-c_db5
;
```
D#MSQ.MSQ00M7.PGMLIB99(C009) --- CONTINUED

```


   retain
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5

      c&year.hgl    c&year.lip
      c&year.self   c&year.educ  c&year.eye
      c&year.urin   c&year.foot  c&year.shoe
      c&year.ledp1  c&year.leds1 c&year.led2 c&year.led3
      c&year.led4   c&year.led5
      c&year.led6   c&year.led7  c&year.led8 c&year.led9
      c&year.neph   c&year.ret
      c&year.blind  c&year.catar c&year.gla c&year.ede c&year.las
      c&year.stk_p  c&year.stk_s c&year.ami_p c&year.ami_s
      c&year.hrt_p  c&year.hrt_s
      c&year.cab    c&year.ptc   c&year.lea c&year.byp
      c&year.er_p
      c&year.db1-c&year.db5

      c&year.total
;

   /* Initialize new indicators, counts */
   if first.y&year.hicno then do;
      do _i=1 to dim(f_new);
         f_new{_i} = 0;
      end;
```

```
      do _i=1 to dim(t_new);
         t_new{_i} = 0;
      end;

      c&year.total=0;
   end;

   /* Set new indicators */
   do _i=1 to dim(f_vars);
      if f_vars(_i) =1 then f_new(_i) =1;
   end;

   /* Counts:  */
   do _i=1 to dim(t_vars);
      t_new{_i} = sum(t_new{_i}, t_vars{_i}, 0);
   end;

   /* Totals:  */
   c&year.total=sum(c&year.total, c_total, 0);   *** total count;


   if last.y&year.hicno then do;
      output;           ***** output the last obs as summary;
   end;

run;



*********************************************************************;
* STEP 2. Reads in the flat FINDER file
          Note: from Step 2 to Step 5 are make-up steps of B005.
*********************************************************************;
DATA TEMP ;
  INFILE IN7 MISSOVER ;
    INPUT
                @1  xref                      $CHAR11.
;
run;

*********************************************************************;
* STEP 3. Sorts the data by XREF
*********************************************************************;
PROC SORT NODUPKEY DATA=TEMP  OUT=TEMP2;
     BY XREF;
RUN;


*********************************************************************;
* STEP 4. Merge with cross-reference file to get unique HIC
*********************************************************************;
DATA TEST  (KEEP=HICNO);
  MERGE TEMP2               (IN=X )
        IN8.XREF&YEAR       (IN=Y  KEEP=HICNO XREF) ;
  BY XREF;
IF X AND Y;
RUN;
```

```
    PROC SORT NODUPKEY DATA=TEST  OUT=TEST2;
         BY HICNO;
    RUN;

    ********************************************************************;
    * STEP 5. Merge it with the person-summary file to get rid of extra cases
              that  don't  meet  two  non-acute  face-to-face,  7  days  apart
criteria
    ********************************************************************;
    DATA OUT1.Y&YEAR.PRALL;
      MERGE TEST2         (IN=X rename=(hicno=y&year.hicno))
            Y&YEAR.PRALL  (IN=Y)
    ;
    BY y&year.hicno;
    IF X AND Y;
    RUN;


     ********************************************************************;
     * STEP 6. Proc contents and proc print on the combined person summary
file
     ********************************************************************;
    PROC CONTENTS DATA=OUT1.Y&YEAR.PRALL DIRECTORY ;
    TITLE "Y&YEAR: COMBINED PERSON SUMMARY FILE ";
    D#MSQ.MSQ00M7.PGMLIB99(C009) --- CONTINUED

    RUN;

    PROC PRINT DATA=OUT1.Y&YEAR.PRALL (OBS=10);
    TITLE "PRINT OUT 10 OBS FOR Y&YEAR COMBINED PERSON SUMMARY FILE";
    RUN;


     ********************************************************************;
     * STEP 7. frequency count of each disease measure
     ********************************************************************;
    PROC FREQ DATA=OUT1.Y&YEAR.PRALL;
    TABLE
          y&year.hgl    y&year.lip
          y&year.self   y&year.educ  y&year.eye
          y&year.urin   y&year.foot  y&year.shoe
          y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
          y&year.led4   y&year.led5
          y&year.led6   y&year.led7  y&year.led8 y&year.led9
          y&year.neph   y&year.ret
          y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
          y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
          y&year.hrt_p y&year.hrt_s
          y&year.cab    y&year.ptc   y&year.lea y&year.byp
          y&year.er_p
          y&year.db1-y&year.db5
    /MISSING;
    RUN;

    ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C010)

```
//WF17C10 JOB (@MSQ0000,WF17),HONG,CLASS=H,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       98  = REPLACE WITH TWO-DIGIT YEAR PRIOR TO PRODUCTION YEAR
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/24/02
//* PGMR      :HONG CHEN, MODIFIED BY OLGA KHAVJOU
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :IN1, IN2, IN3, IN4
//* OUTPUTS   :OUT1
//* PURPOSE   :IDENTIFY NEW ESRD CLAIMANTS AND THEN CREATE FINAL PERSON
//*            SUMMARY FILE, AND PROC SUMMARY OF THIS PERSON SUMMARY
//*            FILE IN ORDER TO GENERATE TABLES.
//*********************************************************************
//*
//* IF THE OLD DATASET EXISTS
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
/*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C010.Y99PRAL2
  DELETE D#MSQ.@MSQ00M7.SAS.C010.Y99TBL
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(800,200))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.A003.XREF99,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1002.DENOMB99,DISP=SHR,
//       UNIT=(,,DEFER),FREE=CLOSE
//IN3    DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1002.DENOMB98,DISP=SHR,
//       UNIT=(,,DEFER),FREE=CLOSE
//IN4    DD DSN=D#MSQ.@MSQ00M7.SAS.C009.Y99PRALL,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C010.Y99PRAL2,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(400,200),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.C010.Y99TBL,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(100,100),RLSE)
//SYSOUT DD SYSOUT=*
//SYSIN DD *


options nocenter pageno=1 ps=55 ls=80 symbolgen  ;

%LET YEAR=99;  *** YEAR OF PRODUCTION;
%LET YR_1=98;  *** ONE YEAR PRIOR TO PRODUCTION YEAR;

%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;
```

```
    ********************************************************************;
    * STEP 1. MERGE CROSS-REFERENCE FILE WITH THE PRODUCTION YEAR
    *         DENOMINATOR FILE TO ASSIGN UNIQUE HICNO.
    *         KEEP ONLY THOSE CLAIMANTS ELIGIBLE FOR M7.
    ********************************************************************;
    DATA TEMP1 (DROP=XREF );
      MERGE       IN2.       DENOMB&YEAR       (IN=X       WHERE=(Y&YEAR.M7='1')
RENAME=(Y&YEAR.HICNO=XREF))
            IN1.XREF&YEAR    (IN=Y);
      BY XREF;
    IF X AND Y;
    RUN;

    ********************************************************************;
    * STEP 2. sort by unique HICNO
    ********************************************************************;
    proc sort NODUPKEY data=TEMP1;
       by HICNO;
    run;

    ********************************************************************;
    * STEP 3. MERGE CROSS-REFERENCE FILE WITH THE DENOMINATOR FILE
    *         FOR YEAR PRIOR TO PRODUCTION YEAR TO ASSIGN UNIQUE HICNO.
    ********************************************************************;

    DATA TEMP2 (DROP=XREF);
      MERGE IN3.DENOMB&YR_1   (IN=X RENAME=(Y&YR_1.HICNO=XREF)
                                    keep=y&YR_1.mdst Y&YR_1.HICNO)
            IN1.XREF&YEAR     (IN=Y);
      BY XREF;
    IF X AND Y;
    RUN;

    ********************************************************************;
    * STEP 4. sort by unique HICNO
    ********************************************************************;
    proc sort NODUPKEY data=TEMP2;
       by HICNO;
    run;

    ********************************************************************;
    * STEP 5. MERGE DENOMINATOR FILES FOR TWO YEARS BY ASSIGNED UNIQUE HICNO
    *         KEEP OBSERVATIONS THAT ARE IN THE PRODUCTION YEAR FILE
    ********************************************************************;

    DATA TEMP3;
       MERGE TEMP1 (IN=X)
             TEMP2 (IN=Y);
    BY HICNO;
    IF X ;
    IF  Y THEN den&YR_1=1; else den&YR_1=0;
    RUN;
```

```
**** test: how many in PRODUCTION YEAR but not in THE PRIOR YEAR*****;
PROC FREQ DATA=TEMP3;
TABLE den&YR_1/MISSING;
RUN;

*******************************************************************;
```
D#MSQ.MSQ00M7.PGMLIB99(C010) --- CONTINUED

```
 * STEP 6. MERGE INITIAL PERSON SUMMARY FILE WITH BASE DENOMINATOR FILES
 * TO CREATE FINAL PERSON SUMMARY FILE WITH NEW ESRD AND DEMOGRAPHIC
 * INFORMATION ON BENEFICIARIES
*******************************************************************;

DATA OUT1.Y&YEAR.PRAL2 ;
   MERGE IN4.Y&YEAR.PRALL (in=x)
         TEMP3            (in=y RENAME=(HICNO=Y&YEAR.HICNO));
BY Y&YEAR.HICNO;
if x and y;

**** REASSIGN VARIABLE VALUES TO DEMOGRAPHIC VARIABLES *****;
         AGEGRP = put(Y99FAGE, $AGEGRP.);

         SEX = put(Y99SEX, $GENDER.);
         RACE = put(Y99RACE, $RACEf.);
         STATUS = put(Y99MDST, $STUSF.);
         DUAL = put(Y99DUAL, $DUALf.);

         URBAN = put(Y99URBN, $URBANf.);
         STATE = put(Y99STATE, $STAf.);

         REGION = put(Y99CREG, $CREf.);
         DIVISION = put(Y99CDIV, $CDIf.);

***** IDENTIFY NEW ESRD DIABETES CLAIMANTS *****;
   attrib y&year.esrd  length=2 label='New ESRD';
if y&YR_1.mdst in ('2','4','5') then esrd&YR_1=1; else esrd&YR_1=0;
if y&YEAR.mdst in ('2','4','5') then esrd&YEAR=1; else esrd&YEAR=0;
if esrd&YEAR=1 and esrd&YR_1=0 then y&YEAR.esrd=1; else y&YEAR.esrd=0;

 ***** FOR THOSE MEASURES WITH PRIMARY AND SECONDARY INDICATORS****;
 **** SET SECONDARY INDICATOR = 0 IF PRIMARY INDICATOR = 1****;
   array pp {*}
      y&year.ledp1
      y&year.stk_p
      y&year.ami_p
      y&year.hrt_p
;

   array ss {*}
      y&year.leds1
      y&year.stk_s
      y&year.ami_s
      y&year.hrt_s
;

  do i=1 to dim (ss);
       if pp{i} =1 then ss{i} =0;
  end;
```

```
RUN;

PROC DELETE DATA=TEMP1 TEMP2 TEMP3;
RUN;

proc freq ;
table AGEGRP SEX RACE STATUS DUAL URBAN REGION DIVISION STATE /missing;
D#MSQ.MSQ00M7.PGMLIB99(C010) --- CONTINUED

run;


 ********************************************************************;
 * STEP 7. Proc contents and proc print on the final person summary file.
 ********************************************************************;
PROC CONTENTS DATA=OUT1.Y&YEAR.PRAL2 DIRECTORY ;
TITLE "IP Y&YEAR: FINAL PERSON SUMMARY FILE";
RUN;

PROC PRINT DATA=OUT1.Y&YEAR.PRAL2 (OBS=10);
TITLE "PRINT OUT 10 OBS FOR Y&YEAR FINAL PERSON SUMMARY FILE";
RUN;


********************************************************************;
 * STEP 8. TOTAL SUM OF MEASURES AND SUM BY EACH DEMOGRAPHIC VARIABLE
 ********************************************************************;
******** OUT2.S_&YEAR._*_* IS USED FOR UNADJUSTED COUNT AND RATE;
PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2;

  VAR _NUMERIC_;
  OUTPUT OUT=OUT2.S_&YEAR._0_0 SUM=;

  TITLE1; TITLE2 'TOTAL: FOR ALL CLAIMANTS';
RUN;

PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP;

  OUTPUT OUT=OUT2.S_&YEAR._0_1 SUM=;

  TITLE1; TITLE2 'BY AGE GROUP';

PROC PRINT DATA=OUT2.S_&YEAR._0_1;

PROC CONTENTS DATA=OUT2.S_&YEAR._0_1;



PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS SEX;
  OUTPUT OUT=OUT2.S_&YEAR._0_2 SUM=;
```

```
  TITLE1; TITLE2 'BY GENDER';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS RACE;
  OUTPUT OUT=OUT2.S_&YEAR._0_3 SUM=;

  TITLE1; TITLE2 'BY RACE';
```

D#MSQ.MSQ00M7.PGMLIB99(C010) --- CONTINUED

```
PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS STATUS;

  OUTPUT OUT=OUT2.S_&YEAR._0_4 SUM=;

  TITLE1; TITLE2 'BY STATUS';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS DUAL;
  OUTPUT OUT=OUT2.S_&YEAR._0_5 SUM=;

  TITLE1; TITLE2 'BY DUAL';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS URBAN;

  OUTPUT OUT=OUT2.S_&YEAR._0_6 SUM=;

  TITLE1; TITLE2 'BY URBAN';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS REGION;

  OUTPUT OUT=OUT2.S_&YEAR._0_7 SUM=;

  TITLE1; TITLE2 'BY REGION';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS DIVISION;
```

```
  OUTPUT OUT=OUT2.S_&YEAR._0_8 SUM=;

  TITLE1; TITLE2 'BY DIVISION';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS STATE;

  OUTPUT OUT=OUT2.S_&YEAR._0_9 SUM=;

  TITLE1; TITLE2 'BY STATE';
RUN;
```
D#MSQ.MSQ00M7.PGMLIB99(C010) --- CONTINUED

```
******** OUT2.A_&YEAR._*_* IS USED FOR AGE/SEX ADJUSTED RATE;

PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX;

  OUTPUT OUT=OUT2.A_&YEAR._0_1 SUM=;

  TITLE1; TITLE2 'BY AGEGRP SEX';

PROC PRINT DATA=OUT2.A_&YEAR._0_1;

PROC CONTENTS DATA=OUT2.A_&YEAR._0_1;



PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX RACE;

  OUTPUT OUT=OUT2.A_&YEAR._0_3 SUM=;

  TITLE1; TITLE2 'BY AGEGRP SEX RACE';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX STATUS;

  OUTPUT OUT=OUT2.A_&YEAR._0_4 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX STATUS';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;
```

```
  VAR _NUMERIC_;
  CLASS AGEGRP SEX DUAL;

  OUTPUT OUT=OUT2.A_&YEAR._0_5 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX DUAL';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX URBAN;

  OUTPUT OUT=OUT2.A_&YEAR._0_6 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX URBAN';
```

D#MSQ.MSQ00M7.PGMLIB99(C010) --- CONTINUED

```
PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX REGION;

  OUTPUT OUT=OUT2.A_&YEAR._0_7 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX REGION';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX DIVISION;

  OUTPUT OUT=OUT2.A_&YEAR._0_8 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX DIVISION';


PROC SUMMARY DATA=OUT1.Y&YEAR.PRAL2 nway;

  VAR _NUMERIC_;
  CLASS AGEGRP SEX STATE;

  OUTPUT OUT=OUT2.A_&YEAR._0_9 SUM=;

  TITLE1; TITLE2 'BY  AGEGRP SEX STATE';
RUN;


ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C011)

```
//WF17011 JOB (@MSQ0000,WF17),HONG,CLASS=D,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :06/24/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :INSAS1
//* OUTPUTS   :OUT1, OUT2, OUT3, OUT4
//* PURPOSE   :CREATE TABLES, ADD DESCRIPTION, AND USE ODS TO OUTPUT THEM
//*            AS RTF FILES
//*********************************************************************
//*
//* IF THE OLD DATASET EXISTS
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
/*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C011.RTFP1C99
  DELETE D#MSQ.@MSQ00M7.SAS.C011.RTFP1R99
  DELETE D#MSQ.@MSQ00M7.SAS.C011.RTFP2C99
  DELETE D#MSQ.@MSQ00M7.SAS.C011.RTFP2R99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(300,100))
//X37IGN DD DUMMY
//INSAS1 DD DSN=D#MSQ.@MSQ00M7.SAS.C010.Y99TBL,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C011.RTFP1C99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.C011.RTFP1R99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT3   DD DSN=D#MSQ.@MSQ00M7.SAS.C011.RTFP2C99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT4   DD DSN=D#MSQ.@MSQ00M7.SAS.C011.RTFP2R99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nodate pageno=1 ps=90 ls=132 symbolgen ;
*options nodate pageno=1 ps=100 ls=148 symbolgen;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;
```

```
   ********************************************************************;
   * STEP 1. prepare numerators and denominators for each table
   D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED

   ********************************************************************;
   data part1 part2 part3 part4 part5 part6 part7 part8 part9;
     set  INSAS1.S_&year._0_7 (rename=(region  =group))
          INSAS1.S_&year._0_8 (rename=(division=group))
          INSAS1.S_&year._0_9 (rename=(state   =group))
   ;

      attrib group     length=$6 label='Demographic*Characteristic';

   if group in ('CRE1  ','CDI1  ','STA07 ','STA20 ','STA22 ','STA30 ','STA41
','STA
   47 ')
      then output part1;
   else if group in ('CDI2  ','STA31 ','STA33 ','STA39 ')
      then output part2;
   else if group in ('CRE2   ','CDI3   ','STA14 ','STA15 ','STA23 ','STA36
','STA52 '
   )
      then output part3;
   else if group in ('CDI4   ','STA16 ','STA17 ','STA24 ','STA26 ','STA28
','STA35 '
   ,
      'STA43 ')
      then output part4;
   else if group in ('CRE3   ','CDI5   ','STA08 ','STA09 ','STA10 ','STA11
','STA21 '
   ,'STA34 ',
      'STA42 ','STA49 ','STA51 ')
      then output part5;
   else if group in ('CDI6  ','STA01 ','STA18 ','STA25 ','STA44 ')
      then output part6;
   else if group in ('CDI7  ','STA04 ','STA19 ','STA37 ','STA45 ')
      then output part7;
   else if group in ('CRE4   ','CDI8   ','STA03 ','STA06 ','STA13 ','STA27
','STA29 '
   ,'STA32 ',
      'STA46 ','STA53 ')
      then output part8;
   else if group in ('CDI9  ','STA02 ','STA05 ','STA12 ','STA38 ','STA50 ')
      then output part9;

   run;


   ********************************************************************;
   *  STEP  2.  create  tables  of  count  for  (1)  process  and  outcome,  (2)
progression
   *         of disease measures
   ********************************************************************;
   data count
        rate;
     set INSAS1.S_&year._0_0 (IN=X)
         INSAS1.S_&year._0_1 (rename=(AGEGRP  =group))
```

```
      INSAS1.S_&year._0_2 (rename=(SEX     =group))
      INSAS1.S_&year._0_3 (rename=(RACE    =group))
      INSAS1.S_&year._0_4 (rename=(STATUS  =group))
      INSAS1.S_&year._0_5 (rename=(DUAL    =group))
      INSAS1.S_&year._0_6 (rename=(URBAN   =group))
      part1
```
D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED
```
      part2
      part3
      part4
      part5
      part6
      part7
      part8
      part9
;
     /* Remove previous old labels */
   LABEL
    y&year.hgl     =' '
    y&year.lip     =' '

    y&year.self    =' '
    y&year.educ    =' '

    y&year.eye     =' '
    y&year.urin    =' '
    y&year.foot    =' '
    y&year.shoe    =' '

    y&year.ledp1   =' '
    y&year.leds1   =' '
    y&year.led2    =' '
    y&year.led3    =' '
    y&year.led4    =' '
    y&year.led5    =' '
    y&year.led6    =' '
    y&year.led7    =' '
    y&year.led8    =' '
    y&year.led9    =' '

    y&year.neph    =' '
    y&year.ret     =' '
    y&year.blind   =' '
    y&year.catar   =' '

    y&year.gla     =' '
    y&year.ede     =' '
    y&year.las     =' '

    y&year.ptc    =' '
    y&year.er_p   =' '


    y&year.stk_p   =' '
    y&year.stk_s   =' '
    y&year.ami_p   =' '
    y&year.ami_s   =' '
```

```
    y&year.hrt_p   =' '
    y&year.hrt_s   =' '

    y&year.cab    =' '
    y&year.lea    =' '
    y&year.byp    =' '

    y&year.db1    =' '
    y&year.db2    =' '
```
D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED
```
    y&year.db3    =' '
    y&year.db4    =' '
    y&year.db5    =' '
    y&year.esrd   =' ';

     /* Assign new labels */
   LABEL
    y&year.hgl     ='Hgl A1c Test'
    y&year.lip     ='Lipid Profiling'

    y&year.self    ='Self Monitoring'
    y&year.educ    ='Diabetes Education'

    y&year.eye     ='Eye Examination'
    y&year.urin    ='Microalbumin Screen'
    y&year.foot    ='Podiatry visit'
    y&year.shoe    ='Therapeutic Shoes'

    y&year.ledp1   ='1st DX: cellulitis of LED'
    y&year.leds1   ='2nd DX: cellulitis of LED'
    y&year.led2    ='Cutaneous gangrene'
    y&year.led3    ='Osteomyelitis'
    y&year.led4    ='Periph neurop or Arthrop'
    y&year.led5    ='Diab w periph circul disorder'
    y&year.led6    ='Lower limb amputation'
    y&year.led7    ='Incision /drain below facia'
    y&year.led8    ='Incision /drain of bone cortex'
    y&year.led9    ='Skin/Subcutan debrid for gangr'

    y&year.neph    ='Nephropathy'
    y&year.ret     ='Retinopathy'
    y&year.blind   ='Blindness'
    y&year.catar   ='Diabetic cataract'

    y&year.gla     ='Glaucoma'
    y&year.ede     ='Macular Edema'
    y&year.las     ='Laser treatment'

    y&year.ptc    ='Progress to PTCA'
    y&year.er_p   ='metabolic crisis: ER/observ bed'


    y&year.stk_p   ='1st DX: Hosp for Stroke'
    y&year.stk_s   ='2nd DX: Hosp for Stroke'
    y&year.ami_p   ='1st DX: Hosp for AMI'
    y&year.ami_s   ='2nd DX: Hosp for AMI'
    y&year.hrt_p   ='1st DX: Hosp for Hrt Failure'
```

```
    y&year.hrt_s   ='2nd DX: Hosp for Hrt Failure'

    y&year.cab    ='Hosp for CABG'
    y&year.lea    ='Hosp for Lower extrem amput'
    y&year.byp    ='Hosp for femoral-bypass'

    y&year.db1    ='1st DX: Hosp diab w/o complication'
    y&year.db2    ='1st DX: Hosp diab w/ ketoacidosis'
    y&year.db3    ='1st DX: Hosp diab w/ hyperosmolar'
    y&year.db4    ='1st DX: Hosp diab w/ other coma'
    y&year.db5    ='1st DX: Hosp diab w/ other complic'
    y&year.esrd   ='New ESRD';
```
D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED

```
IF X THEN group='TOTAL ';
KEEP  group   _FREQ_
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;

   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;

  /* create count file*/
output count;

  /* create file for rate per 1000: numerator is           */
  /* count of diabetes with measure, denominator is _FREQ_ */
do j= 1 to dim (list);
      list{j}= (list{j}/_FREQ_) * 1000 ;
end;
```

```
    output rate;

    run;


    PROC PRINT DATA=count LABEL;
    VAR   group _FREQ_ y&year.hgl    y&year.lip
          y&year.self   y&year.educ  y&year.eye
          y&year.urin   y&year.foot  y&year.shoe;
    title 'count: process and outcome measures';
    run;


    ************************************************************************;
```

D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED

```
    *  STEP  3.  create  table  of  count  for  process  and  outcome  of  disease
measures
    ************************************************************************;

    ODS RTF FILE=OUT1 TRANTAB=ASCII;

    TITLE  "M7 PROCESS AND OUTCOME MEASURES (YEAR &YEAR)";
    TITLE2 'COUNT FOR DIABETES CLAIMANTS (BASED ON 5% SAF)';


    PROC PRINT DATA=COUNT  WIDTH=MIN ROUND SPLIT='*';
    VAR   y&year.hgl    y&year.lip
          y&year.self   y&year.educ  y&year.eye
          y&year.urin   y&year.foot  y&year.shoe;
    FORMAT GROUP $GROUPS.;
    FORMAT y&year.hgl    y&year.lip
           y&year.self   y&year.educ  y&year.eye
           y&year.urin   y&year.foot  y&year.shoe   COMMA8.0;
    ID GROUP;
    RUN;

    ODS RTF CLOSE;



    ************************************************************************;
    * STEP 4. create table of rate per 1000 for process and outcome
    *         of disease measures
    ************************************************************************;

    ODS RTF FILE=OUT2 TRANTAB=ASCII;

    TITLE  "M7 PROCESS AND OUTCOME MEASURES (YEAR &YEAR)";
    TITLE2 'RATE PER 1000 DIABETES CLAIMANTS (BASED ON 5% SAF)';
    TITLE3 'UNADJUSTED RATES';


    PROC PRINT DATA=RATE WIDTH=MIN ROUND SPLIT='*';
    VAR   y&year.hgl    y&year.lip
          y&year.self   y&year.educ  y&year.eye
          y&year.urin   y&year.foot  y&year.shoe;
```

```
FORMAT GROUP $GROUPS.;
FORMAT y&year.hgl    y&year.lip
       y&year.self   y&year.educ  y&year.eye
       y&year.urin   y&year.foot  y&year.shoe   COMMA6.2;
ID GROUP;
RUN;

ODS RTF CLOSE;



*********************************************************************;
* STEP 5. create table of count for progression of disease measures
*********************************************************************;

ODS RTF FILE=OUT3 TRANTAB=ASCII;

TITLE  "M7 PROGRESSION OF DISEASE MEASURES (YEAR &YEAR)";
D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED

TITLE2 'COUNT FOR DIABETES CLAIMANTS (BASED ON 5% SAF)';


PROC PRINT DATA=COUNT WIDTH=MIN ROUND SPLIT='*';
VAR   y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp
      ;
FORMAT GROUP $GROUPS.;
FORMAT y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp   COMMA8.0;
ID GROUP;
RUN;

ODS RTF CLOSE;



*********************************************************************;
* STEP 6. create table of rate per 1000 for progression
*         of disease measures
*********************************************************************;
```

```
ODS RTF FILE=OUT4 TRANTAB=ASCII;

TITLE  "M7 PROGRESSION OF DISEASE MEASURES (YEAR &YEAR)";
TITLE2 'RATE PER 1000 DIABETES CLAIMANTS (BASED ON 5% SAF)';
TITLE3 'UNADJUSTED RATES';

PROC PRINT DATA=RATE WIDTH=MIN ROUND SPLIT='*';
VAR   y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp;
FORMAT GROUP $GROUPS.;
FORMAT y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
D#MSQ.MSQ00M7.PGMLIB99(C011) --- CONTINUED

      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp   COMMA6.2;
ID GROUP;
RUN;

ODS RTF CLOSE;

*****************************************************************;
* STEP 7. PROC CONTENTS FOR EACH OF THE COUNT AND RATE FILES
*****************************************************************;
PROC CONTENTS DATA=COUNT  DIRECTORY ;
TITLE "Y&YEAR: FOR PROCESS AND OUTCOME, PROGRESSION OF DISEASE";
RUN;

ENDSAS;
```

D#MSQ.MSQ00M7.PGMLIB99(C012)

```
//WF17012 JOB (@MSQ0000,WF17),HONG,CLASS=D,
//        MSGCLASS=Q,NOTIFY=&SYSUID
//*
//********REPLACE THE FOLLOWING 2 PARAMETERS BEFORE RUNNING THIS *****
//*       99  = REPLACE WITH TWO-DIGIT PRODUCTION YEAR IN ALL PLACES
//*       WF17 = REPLACE WITH ID OF PERSON WHO WILL SUBMIT THE JOB
//*********************************************************************
//*********************************************************************
//* PROJ      :MQMS
//* REQ BY    :TEAM MQMS
//* DATE      :07/16/02
//* PGMR      :HONG CHEN
//* COMPANY   :RTI INTERNATIONAL.
//* INPUTS    :INSAS1, IN2
//* OUTPUTS   :OUT1, OUT2
//* PURPOSE   :CREATE AGE/SEX ADJUSTED RATE TABLES. FOR AGE, ONLY
//*            SEX ADJUSTED; FOR SEX, ONLY AGE ADJUSTED
//*********************************************************************
//*
//* IF THE OLD DATASET EXISTS
//* STEP00 IS TO DELETE THE OLD CATALOGED DATASETS BEFORE RERUN CODE
//*********************************************************************
/*
//STEP00 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DELETE D#MSQ.@MSQ00M7.SAS.C012.RTFP1A99
  DELETE D#MSQ.@MSQ00M7.SAS.C012.RTFP2A99
/*
//STEP1  EXEC SAS8
//WORK   DD SPACE=(CYL,(300,100))
//X37IGN DD DUMMY
//IN1    DD DSN=D#MSQ.@MSQ00M7.SAS.C010.Y99TBL,DISP=SHR
//IN2    DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1004.AGESEX99,DISP=SHR
//OUT1   DD DSN=D#MSQ.@MSQ00M7.SAS.C012.RTFP1A99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//OUT2   DD DSN=D#MSQ.@MSQ00M7.SAS.C012.RTFP2A99,
//       DISP=(NEW,CATLG,DELETE),SPACE=(TRK,(30,10)),
//       DCB=(RECFM=FB,LRECL=80)
//SYSOUT DD SYSOUT=*
//SYSIN DD *

options nodate pageno=1 ps=90 ls=132 symbolgen ;
*options nodate pageno=1 ps=100 ls=148 symbolgen;

%LET YEAR=99;  *** YEAR OF PRODUCTION;

%INCLUDE "D#MSQ.@MSQ00M7.PGMLIB99(FORMAT&YEAR)";   ***SAS formats;



*******************************************************************;
 * STEP 1. CREATE AGE/SEX WEIGHTS USING M1 1999 FILE AGESEX99.
           THIS FILE WILL ALSO BE APPLIED TO OTHER YEAR IN 1992-2000
*******************************************************************;
```

```
DATA WGTA_S WGT_S WGT_A;
  SET IN2.AGESEX99;
```
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

```
**** REASSIGN VARIABLE VALUES TO DEMOGRAPHIC VARIABLES *****;
     AGEGRP = put(Y99FAGE, $AGEGRP.);
     SEX = put(Y99SEX, $GENDER.);

IF _N_=1 THEN TOTAL99=Y99CNTS;
RETAIN TOTAL99;
ASWGT99=Y99CNTS/TOTAL99;
IF AGEGRP NE '' & SEX NE '' THEN OUTPUT WGTA_S;
   ELSE
IF AGEGRP = '' & SEX NE '' THEN OUTPUT WGT_S;
   ELSE
IF AGEGRP NE '' & SEX = '' THEN OUTPUT WGT_A;
RUN;


 PROC PRINT DATA=WGTA_S;
      TITLE "1999 AGE/SEX WEIGHTS";
 RUN;
 PROC PRINT DATA=WGT_S;
      TITLE "1999 SEX WEIGHTS";
 RUN;
 PROC PRINT DATA=WGT_A;
      TITLE "1999 AGE WEIGHTS";
 RUN;


*********************************************************************;
 * STEP 2. CREATE ADJUSTED RATES BY AGE/SEX CELLS FOR EACH SUBGROUP
*********************************************************************;

********** FOR TOTAL **************;
DATA A_&YEAR._0_0 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_1 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
```

```
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_0;
```
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED
```

END;
run;

PROC SUMMARY DATA=A_&YEAR._0_0 nway ;

  VAR _NUMERIC_;
  OUTPUT OUT=A_&YEAR._1_0 SUM=;
  TITLE1; TITLE2 'TOTAL';

PROC PRINT DATA=A_&YEAR._1_0;
  TITLE1; TITLE2 'TOTAL';
RUN;


******** FOR AGEGRP: ONLY SEX ADJUSTED ***********;
PROC SORT DATA=IN1.A_&YEAR._0_1  OUT=TEMP;
BY SEX;
RUN;


DATA A_&YEAR._0_1 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE TEMP         (IN=X)
        WGT_S        (IN=Y  KEEP=SEX ASWGT99);
BY SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_1;
END;
run;
```

```
PROC SUMMARY DATA=A_&YEAR._0_1 nway ;

  VAR _NUMERIC_;
  CLASS AGEGRP;

  OUTPUT OUT=A_&YEAR._1_1 SUM=;
  TITLE1; TITLE2 'AGEGRP';
```

D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

```
PROC PRINT DATA=A_&YEAR._1_1;
  TITLE1; TITLE2 'AGEGRP';
RUN;


******** FOR SEX: ONLY AGEGRP ADJUSTED **********;
DATA A_&YEAR._0_2 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_1         (IN=X )
        WGT_A                    (IN=Y KEEP=AGEGRP ASWGT99);
BY AGEGRP;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_2;
END;
run;

PROC SUMMARY DATA=A_&YEAR._0_2 nway ;

  VAR _NUMERIC_;
  CLASS SEX;
  OUTPUT OUT=A_&YEAR._1_2 SUM=;
  TITLE1; TITLE2 'SEX';


********** FOR RACE **********;
DATA A_&YEAR._0_3 (DROP= _TYPE_ _FREQ_)  ERR;
```

```
  MERGE IN1.A_&YEAR._0_3 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_3;
END;
run;

PROC SUMMARY DATA=A_&YEAR._0_3 nway ;

  VAR _NUMERIC_;
  CLASS RACE;
  OUTPUT OUT=A_&YEAR._1_3 SUM=;
  TITLE1; TITLE2 'RACE';

PROC PRINT DATA=A_&YEAR._1_3;
  TITLE1; TITLE2 'RACE';
RUN;

************ FOR STATUS **********;
DATA A_&YEAR._0_4 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_4 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
```

```
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_4;
END;
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

run;

PROC SUMMARY DATA=A_&YEAR._0_4 nway ;

  VAR _NUMERIC_;
  CLASS STATUS;

  OUTPUT OUT=A_&YEAR._1_4 SUM=;
  TITLE1; TITLE2 'STATUS';

************ FOR DUAL **********;
DATA A_&YEAR._0_5 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_5 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_5;
END;
run;
```

```
PROC SUMMARY DATA=A_&YEAR._0_5 nway ;

  VAR _NUMERIC_;
  CLASS DUAL;
  OUTPUT OUT=A_&YEAR._1_5 SUM=;
  TITLE1; TITLE2 'DUAL';

************ FOR URBAN **********;
DATA A_&YEAR._0_6 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_6 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_6;
END;
run;

PROC SUMMARY DATA=A_&YEAR._0_6 nway ;

  VAR _NUMERIC_;
  CLASS URBAN;

  OUTPUT OUT=A_&YEAR._1_6 SUM=;
  TITLE1; TITLE2 'URBAN';

************ FOR REGION **********;
DATA A_&YEAR._0_7 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_7 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
```

```
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_7;
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

END;
run;

PROC SUMMARY DATA=A_&YEAR._0_7 nway ;

  VAR _NUMERIC_;
  CLASS REGION;

  OUTPUT OUT=A_&YEAR._1_7 SUM=;
  TITLE1; TITLE2 'REGION';

************ FOR DIVISION **********;
DATA A_&YEAR._0_8 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_8 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
```

```
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_8;
END;
run;

PROC SUMMARY DATA=A_&YEAR._0_8 nway ;

  VAR _NUMERIC_;
  CLASS DIVISION;

  OUTPUT OUT=A_&YEAR._1_8 SUM=;
  TITLE1; TITLE2 'DIVISION';

************ FOR STATE **********;
DATA A_&YEAR._0_9 (DROP= _TYPE_ _FREQ_)  ERR;
  MERGE IN1.A_&YEAR._0_9 (IN=X)
        WGTA_S           (IN=Y);
BY AGEGRP SEX;
IF X AND NOT Y THEN OUTPUT ERR;
ELSE IF X AND Y THEN DO;
   /* rate per 1000: numerator is count of each measure flag, */
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

   /* denominator is diabetesdiabetes claimants, i.e., _FREQ_*/
   array list {*}
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
;
  do j= 1 to dim (list);
      list{j}= ASWGT99* ((list{j}/_FREQ_) * 1000) ;
  end;
OUTPUT A_&YEAR._0_9;
END;
run;

PROC SUMMARY DATA=A_&YEAR._0_9 nway ;

  VAR _NUMERIC_;
  CLASS STATE;

  OUTPUT OUT=A_&YEAR._1_9 SUM=;
  TITLE1; TITLE2 'STATE';
RUN;
```

```
   ********************************************************************;
   * STEP 3. prepare rows (subgroups) in proper order for tables
   ********************************************************************;
   data part1 part2 part3 part4 part5 part6 part7 part8 part9;
     set  a_&year._1_7 (rename=(region  =group))
          a_&year._1_8 (rename=(division=group))
          a_&year._1_9 (rename=(state   =group))
   ;

      attrib group     length=$6 label='Demographic*Characteristic';

   if group in ('CRE1  ','CDI1  ','STA07 ','STA20 ','STA22 ','STA30 ','STA41
','STA
   47 ')
      then output part1;
   else if group in ('CDI2  ','STA31 ','STA33 ','STA39 ')
      then output part2;
   else if group in ('CRE2   ','CDI3   ','STA14 ','STA15 ','STA23 ','STA36
','STA52 '
   )
      then output part3;
   else if group in ('CDI4   ','STA16 ','STA17 ','STA24 ','STA26 ','STA28
','STA35 '
   ,
      'STA43 ')
   D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

      then output part4;
   else if group in ('CRE3   ','CDI5   ','STA08 ','STA09 ','STA10 ','STA11
','STA21 '
   ,'STA34 ',
      'STA42 ','STA49 ','STA51 ')
      then output part5;
   else if group in ('CDI6  ','STA01 ','STA18 ','STA25 ','STA44 ')
      then output part6;
   else if group in ('CDI7  ','STA04 ','STA19 ','STA37 ','STA45 ')
      then output part7;
   else if group in ('CRE4   ','CDI8   ','STA03 ','STA06 ','STA13 ','STA27
','STA29 '
   ,'STA32 ',
      'STA46 ','STA53 ')
      then output part8;
   else if group in ('CDI9  ','STA02 ','STA05 ','STA12 ','STA38 ','STA50 ')
      then output part9;

   run;


   ********************************************************************;
   * STEP 4. put all rows together for tables of (1) process and outcome, (2)
progr
   ession
   *         of disease measures
   ********************************************************************;
   data rate;
     set a_&year._1_0 (IN=X)
```

```
      a_&year._1_1 (rename=(AGEGRP  =group))
      a_&year._1_2 (rename=(SEX     =group))
      a_&year._1_3 (rename=(RACE    =group))
      a_&year._1_4 (rename=(STATUS  =group))
      a_&year._1_5 (rename=(DUAL    =group))
      a_&year._1_6 (rename=(URBAN   =group))
      part1
      part2
      part3
      part4
      part5
      part6
      part7
      part8
      part9
;
     /* Remove previous old labels */
   LABEL
    y&year.hgl     =' '
    y&year.lip     =' '
    y&year.self    =' '
    y&year.educ    =' '
    y&year.eye     =' '
    y&year.urin    =' '
    y&year.foot    =' '
    y&year.shoe    =' '
    y&year.ledp1   =' '
    y&year.leds1   =' '
    y&year.led2    =' '
    y&year.led3    =' '
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

    y&year.led4    =' '
    y&year.led5    =' '
    y&year.led6    =' '
    y&year.led7    =' '
    y&year.led8    =' '
    y&year.led9    =' '
    y&year.neph    =' '
    y&year.ret     =' '
    y&year.blind   =' '
    y&year.catar   =' '
    y&year.gla     =' '
    y&year.ede     =' '
    y&year.las     =' '
    y&year.ptc    =' '
    y&year.er_p   =' '
    y&year.stk_p   =' '
    y&year.stk_s   =' '
    y&year.ami_p   =' '
    y&year.ami_s   =' '
    y&year.hrt_p   =' '
    y&year.hrt_s   =' '
    y&year.cab    =' '
    y&year.lea    =' '
    y&year.byp    =' '
    y&year.db1    =' '
    y&year.db2    =' '
```

```
    y&year.db3    =' '
    y&year.db4    =' '
    y&year.db5    =' '
    y&year.esrd   =' ';

     /* Assign new labels */
   LABEL
    y&year.hgl     ='Hgl A1c Test'
    y&year.lip     ='Lipid Profiling'
    y&year.self    ='Self Monitoring'
    y&year.educ    ='Diabetes Education'
    y&year.eye     ='Eye Examination'
    y&year.urin    ='Microalbumin Screen'
    y&year.foot    ='Podiatry visit'
    y&year.shoe    ='Therapeutic Shoes'
    y&year.ledp1   ='1st DX: cellulitis of LED'
    y&year.leds1   ='2nd DX: cellulitis of LED'
    y&year.led2    ='Cutaneous gangrene'
    y&year.led3    ='Osteomyelitis'
    y&year.led4    ='Periph neurop or Arthrop'
    y&year.led5    ='Diab w periph circul disorder'
    y&year.led6    ='Lower limb amputation'
    y&year.led7    ='Incision /drain below facia'
    y&year.led8    ='Incision /drain of bone cortex'
    y&year.led9    ='Skin/Subcutan debrid for gangr'
    y&year.neph    ='Nephropathy'
    y&year.ret     ='Retinopathy'
    y&year.blind   ='Blindness'
    y&year.catar   ='Diabetic cataract'
    y&year.gla     ='Glaucoma'
    y&year.ede     ='Macular Edema'
    y&year.las     ='Laser treatment'
    y&year.ptc    ='Progress to PTCA'
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

    y&year.er_p   ='metabolic crisis: ER/observ bed'
    y&year.stk_p   ='1st DX: Hosp for Stroke'
    y&year.stk_s   ='2nd DX: Hosp for Stroke'
    y&year.ami_p   ='1st DX: Hosp for AMI'
    y&year.ami_s   ='2nd DX: Hosp for AMI'
    y&year.hrt_p   ='1st DX: Hosp for Hrt Failure'
    y&year.hrt_s   ='2nd DX: Hosp for Hrt Failure'
    y&year.cab    ='Hosp for CABG'
    y&year.lea    ='Hosp for Lower extrem amput'
    y&year.byp    ='Hosp for femoral-bypass'
    y&year.db1    ='1st DX: Hosp diab w/o complication'
    y&year.db2    ='1st DX: Hosp diab w/ ketoacidosis'
    y&year.db3    ='1st DX: Hosp diab w/ hyperosmolar'
    y&year.db4    ='1st DX: Hosp diab w/ other coma'
    y&year.db5    ='1st DX: Hosp diab w/ other complic'
    y&year.esrd   ='New ESRD';

IF X THEN group='TOTAL ';
KEEP  group   _FREQ_
      y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe
      y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
```

```
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.neph   y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p y&year.hrt_s
      y&year.cab    y&year.ptc   y&year.lea y&year.byp
      y&year.er_p
      y&year.db1-y&year.db5
      y&year.esrd;
run;


PROC PRINT DATA=rate LABEL;
VAR   group _FREQ_ y&year.hgl    y&year.lip
      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe;
title 'process and outcome measures';
run;



*****************************************************************;
* STEP 5. create rich text format table of age/sex adjusted rate per 1000
*         for process and outcome of disease measures
*****************************************************************;

ODS RTF FILE=OUT1 TRANTAB=ASCII;

TITLE  "M7 PROCESS AND OUTCOME MEASURES (YEAR &YEAR)";
TITLE2 'RATE PER 1000 DIABETES CLAIMANTS (BASED ON 5% SAF)';
TITLE3 'AGE_SEX_ADJUSTED RATES';


PROC PRINT DATA=RATE WIDTH=MIN ROUND SPLIT='*';
VAR   y&year.hgl    y&year.lip
D#MSQ.MSQ00M7.PGMLIB99(C012) --- CONTINUED

      y&year.self   y&year.educ  y&year.eye
      y&year.urin   y&year.foot  y&year.shoe;
FORMAT GROUP $GROUPS.;
FORMAT y&year.hgl    y&year.lip
       y&year.self   y&year.educ  y&year.eye
       y&year.urin   y&year.foot  y&year.shoe   COMMA6.2;
ID GROUP;
RUN;

ODS RTF CLOSE;


*****************************************************************;
* STEP 6. create rich text format table of age/sex adjusted rate per 1000
*         for progression of disease measures
*****************************************************************;

ODS RTF FILE=OUT2 TRANTAB=ASCII;

TITLE  "M7 PROGRESSION OF DISEASE MEASURES (YEAR &YEAR)";
```

```
TITLE2 'RATE PER 1000 DIABETES CLAIMANTS (BASED ON 5% SAF)';
TITLE3 'AGE_SEX_ADJUSTED RATES';

PROC PRINT DATA=RATE WIDTH=MIN ROUND SPLIT='*';
VAR   y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp;
FORMAT GROUP $GROUPS.;
FORMAT y&year.ledp1  y&year.leds1 y&year.led2 y&year.led3
      y&year.led4   y&year.led5
      y&year.led6   y&year.led7  y&year.led8 y&year.led9
      y&year.esrd   y&year.neph  y&year.ret
      y&year.blind  y&year.catar y&year.gla  y&year.ede y&year.las
      y&year.stk_p  y&year.stk_s y&year.ami_p y&year.ami_s
      y&year.hrt_p  y&year.hrt_s
      y&year.cab    y&year.ptc
      y&year.er_p
      y&year.db1-y&year.db5 y&year.lea y&year.byp   COMMA6.2;
ID GROUP;
RUN;

ODS RTF CLOSE;

*********************************************************************;
* STEP 7. PROC CONTENTS FOR THE INTERMEDIATE SAS RATE FILE
*********************************************************************;
PROC CONTENTS DATA=RATE  DIRECTORY ;
TITLE "Y&YEAR: FOR PROCESS AND OUTCOME, PROGRESSION OF DISEASE";
RUN;

ENDSAS;
```

PROGRAMS –

HIGH-RISK SURGERIES

D#MSQ.@MSQ01.JCLTEST(MSQ#2H10)

```
//L1982H10 JOB (MSQ01,MQMS),'STEP1_92',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ2H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H10D)

```
//MSQ2H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ2H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="92"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC92PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL92PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H10)

```
//L1983H10 JOB (MSQ01,MQMS),'STEP1_93',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ3H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H10D)

```
//MSQ2H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ3H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="93"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC93PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL93PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H10)

```
//L1984H10 JOB (MSQ01,MQMS),'STEP1_94',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ4H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H10D)

```
//MSQ4H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ4H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="94"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC94PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL94PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H10)

```
//L1985H10 JOB (MSQ01,MQMS),'STEP1_95',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ5H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H10D)

```
//MSQ5H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ5H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="95"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC95PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL95PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H10)

```
//L1986H10 JOB (MSQ01,MQMS),'STEP1_96',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ6H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H10D)

```
//MSQ6H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ6H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="96"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC96PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL96PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H10)

```
//L1987H10 JOB (MSQ01,MQMS),'STEP1_97',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ7H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H10D)

```
//MSQ7H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ7H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="97"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC97PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL97PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H10)

```
//L1988H10 JOB (MSQ01,MQMS),'STEP1_98',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ8H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H10D)

```
//MSQ8H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ8H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="98"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC98PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL98PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H10)

```
//L1989H10 JOB (MSQ01,MQMS),'STEP1_99',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQ9H10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H10D)

```
//MSQ9H10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQ9H000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="99"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC99PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL99PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH10)

```
//L198AH10 JOB (MSQ01,MQMS),'STEP1_00',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQAH10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH10D)

```
//MSQAH10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQAH000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="00"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC00PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL00PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH10)

```
//L198BH10 JOB (MSQ01,MQMS),'STEP1_01',CLASS=H,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//JS010   EXEC MSQBH10D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH10D)

```
//MSQBH10D PROC
//*
//PS000  EXEC PGM=IDCAMS
//SYSIN    DD DSN=D#MSQ.@MSQ01.PARMLIB(MSQBH000),
//            DISP=(SHR,PASS,KEEP)
//SYSPRINT  DD SYSOUT=*
//*
//PS010  EXEC SAS8,PARM='SYSPARM="01"'
//TEMP1    DD DSN=&&TEMP1,
//            UNIT=MCART,LABEL=(1,SL,,,RETPD=5)
//M4FLAGS  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4FLAGS,
//            DISP=(SHR,KEEP,KEEP)
//M4OTHER  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4OTHER,
//            DISP=(SHR,KEEP,KEEP)
//OCOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.OC01PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//VLOUT    DD DSN=D#MSQ.@MSQ01.MSQYH102.VL01PX,DISP=(NEW,CATLG,DELETE),
//            UNIT=HCFA80,SPACE=(CYL,(500,300),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH102),
//           DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQYH102)

```
  /*************************************************************/
  /* PROGRAM:        D#MSQ.@MSQ01.PGMLIB(MSQYH102)             */
  /*-----------------------------------------------------------*/
  /* PURPOSE:        IDENTIFY 12 VOLUMES AND 14 PROCEDURES FOR */
  /*                 HIGH RISK SURGERIES FOR MQMS PROJECT(8908-006)*/
  /*                 PRODUCE 1 VLOLUME FILE AND 1 OUTCOME FILE */
  /*                 FOR EACH YEAR                             */
  /*-----------------------------------------------------------*/
  /* SOURCE CODE TO IDENTIFY VOLUME AND OUTCOME FOR EACH PROCEDURE:*/
  /*                 BIRKMEYER ET AL., <<HOSPITAL VOLUME AND   */
  /*                 SURGICAL MORTALITY IN THE UNITED STATES>>,*/
  /*                 NEW ENG J MED, 2002; 346, 1128-1137       */
  /*-----------------------------------------------------------*/
  /* INPUT DATA:     D#MSQ.@MSQ0000.SAS.YYYM4001.M4FLAGS       */
  /*                 D#MSQ.@MSQ0000.SAS.YYYM4001.M4OTHER       */
  /*             !!INPUT DATA IS EVOKED BY PROCLIB(MSQYH102P)  */
  /* OUTPUT DATA:    P#MSQ.@MSQ01.MSQYH102.OCYYPX              */
  /*                       (OUTCOME FILES WHERE  Y=1-9, A, B;  */
  /*                       YY=92-01)                           */
  /*                 P#MSQ.@MSQ01.MSQYH102.VLYYPX              */
  /*                       (VOLUME FILES WHERE Y=1-9, A, B;    */
  /*                       YY=92-01)                           */
  /*             !!OUTPUT FILES ARE PROCESSED BY PROCLIB(MSQYH102P)*/
  /*-----------------------------------------------------------*/
  /* KEY VARIABLES PRODUCED: VOLUME VARIABLE: VLN              */
  /*                               WHERE N=1-7,9,11-14         */
  /*                         OUTCOME VARIABLE OCN              */
  /*                               WHERE N=1-14                */
  /*                         YEAR FLAG: YYYY (1992-2001)       */
  /*                                                           */
  /*-----------------------------------------------------------*/
  /* WRITTEN BY:     LUCY LU                                   */
  /*                 MATHEMATICA POLICY RESEARCH, INC.         */
  /*                 PHONE 202-554-7578                        */
  /*-----------------------------------------------------------*/
  /* DATE:           MAR. 16, 2003                             */
  /*************************************************************/

 OPTIONS NOCENTER;

%LET YR  = &SYSPARM;
%LET P1  = ABD. AORTIC ANEURYSM REPAIR;
%LET P2  = CORONARY ARTERY BYPASS GRAFT;
%LET P3  = CAROTID ENDARTERECTOMY;
%LET P4  = ESOPHAGECTOMY;
%LET P5  = LOWER EXTREMITY BYPASS;
%LET P6  = PANCREATECTOMY;
%LET P7  = AORTIC VAL. REPLACEMENT;
%LET P8  = MITRAL VAL. REPLACEMENT;
%LET P9  = PNEUMONECTOMY;
%LET P10 = LOBECTOMY;
%LET P11 = COLECTOMY;
%LET P12 = CYSTECTOMY;
%LET P13 = GASTRECTOMY;
%LET P14 = NEPHRECTOMY;
```

```
 /*******************************************************/
 /** PROCESSING ALL YEAR FILES                         **/
D#MSQ.@MSQ01.PGMLIB(MSQYH102) -- CONTINUED

 /** INPUT DATASETS ARE CLAIM-LEVEL DATA WITH UNIQUE   **/
 /** ID YYYNOBS WHERE YY=92-01                         **/
 /*******************************************************/


DATA TEMP1.TEMP;
   SET   M4FLAGS.M4FLAGS
         M4OTHER.M4OTHER
         ;


  /***SET UP KEY VOLUME VARIABLES AND OUTCOME VARIABLES***/

%MACRO LAB;

 %DO I=1 %TO 14;
    VL&I=0;
    OC&I=0;
 %END;

 DROP VL8 VL10;
 %MEND;

 %LAB;


  /***DEFINE ESRD PATIENTS, WHO WILL BE EXCLUDED IN LEBY CASE***/
  IF Y&YR.MDST IN ('2','4','5') THEN ESRD=1;
  ELSE ESRD=0;


  /***FLAG YEAR FOR LATER PROGRAMMING CONVENIENCE***/
YEAR=INPUT("&YR",2.)+1900;
IF YEAR < 1992 THEN YEAR +100;


  /********** LABEL VARIABLES **************/
 LABEL
 VL1 = "VOLUME INDICATOR-&P1"
 VL2 = "VOLUME INDICATOR-&P2"
 VL3 = "VOLUME INDICATOR-&P3"
 VL4 = "VOLUME INDICATOR-&P4"
 VL5 = "VOLUME INDICATOR-&P5"
 VL6 = "VOLUME INDICATOR-&P6"
 VL7 = "VOLUME INDICATOR-HEART VALVE REPLACEMENTS"
 VL9 = "VOLUME INDICATOR-PULMONARY RESECTIONS"
 VL11 = "VOLUME INDICATOR-&P11"
 VL12 = "VOLUME INDICATOR-&P12"
 VL13 = "VOLUME INDICATOR-&P13"
 VL14 = "VOLUME INDICATOR-&P14"

 OC1 = "OUTCOME INDICATOR-&P1"
 OC2 = "OUTCOME INDICATOR-&P2"
```

```
 OC3 = "OUTCOME INDICATOR-&P3"
 OC4 = "OUTCOME INDICATOR-&P4"
 OC5 = "OUTCOME INDICATOR-&P5"
 OC6 = "OUTCOME INDICATOR-&P6"
 OC7 = "OUTCOME INDICATOR-&P7"
 OC8 = "OUTCOME INDICATOR-&P8"
```
D#MSQ.@MSQ01.PGMLIB(MSQYH102) -- CONTINUED

```
 OC9 = "OUTCOME INDICATOR-&P9"
 OC10 = "OUTCOME INDICATOR-&P10"
 OC11 = "OUTCOME INDICATOR-&P11"
 OC12 = "OUTCOME INDICATOR-&P12"
 OC13 = "OUTCOME INDICATOR-&P13"
 OC14 = "OUTCOME INDICATOR-&P14"
 ;


  /*******************************************************/
  /***    FOLLOWING CODE THAT IDENTIFYS VOLUME AND      ***/
  /***  OUTCOME PROCEDURES COMES FROM BIRKMEYER ET AL.  ***/
  /*******************************************************/


ARRAY SXRAY    (6) Y&YR.SPR1-Y&YR.SPR6;
ARRAY DXRAY    (10) Y&YR.DIA1-Y&YR.DIA10;


*** DEFINING AAA ***;
SXAIF=0;
SXANAS=0;
SXTAA=0;
DO I=1 TO 6;
  IF SXRAY(I)='3844' THEN VL1=1;
  IF SXRAY(I)='3925' THEN SXAIF=1;
  IF SXRAY(I)='3834' THEN SXANAS=1;  *FRINGE CODE;
  IF SXRAY(I)='3845' THEN SXTAA=1;   *THORACIC ANEURYSM PROCEDURE;
END;

DXAAA=0;    *ABDOMINAL ANEURYSM;
DXTAAA=0;   *THORACOABDOMINAL ANEURYSM;
DXTAA=0;    *THORACIC ANEURYSM;
DXRUP=0;    *RUPTURED ANEURYSM, ANY LOCATION;
DO I=1 TO 10;
  IF DXRAY(I) IN ('4414','4419') THEN DXAAA=1;
  IF DXRAY(I)='4417' THEN DXTAAA=1;
  IF DXRAY(I)='4412' THEN DXTAA=1;
  IF ((DXRAY(I)='4411') OR
      (DXRAY(I)='4413') OR
      (DXRAY(I)='4415') OR
      (DXRAY(I)='4416') OR
      (DXRAY(I)='4410') OR
      ('44100'<=DXRAY(I)<='44103')) THEN DXRUP=1;
END;


IF ((VL1=1 OR SXAIF=1) AND
    (SXTAA=0)) THEN VL1=1;
IF ((VL1=1 OR SXAIF=1) AND
```

```
    (DXAAA=1 OR DXTAAA=1) AND
    (SXTAA=0) AND
    (DXTAA=0) AND
    (DXRUP=0)) THEN OC1=1;
*IF VL1=1 THEN OUTPUT VLOUT.VL&YR.PX1;
*IF OC1=1 THEN OUTPUT OCOUT.OC&YR.PX1;

*** DEFINING CABG ***;
HRTVLV=0;            *ANY VALVE REPAIR-REPLACEMENT;
```
D#MSQ.@MSQ01.PGMLIB(MSQYH102) -- CONTINUED
```
DO I=1 TO 6;
  IF ('3610'<=SXRAY(I)<='3619') THEN VL2=1;
  IF ('3510'<=SXRAY(I)<='3529') THEN HRTVLV=1;
END;

IF ((VL2=1) AND (HRTVLV=0)) THEN OC2=1;
*IF VL2=1 THEN OUTPUT VLOUT.VL&YR.PX2;
*IF OC2=1 THEN OUTPUT OCOUT.OC&YR.PX2;

*** DEFINING CEA ***;
DO I=1 TO 6;
  IF SXRAY(I)='3812' THEN DO;
    VL3=1;
    OC3=1;
  END;
END;
*IF VL3=1 THEN OUTPUT VLOUT.VL&YR.PX3;
*IF OC3=1 THEN OUTPUT OCOUT.OC&YR.PX3;

*** DEFINING ESOPHAGECTOMY ***;
DO J=1 TO 6;
  IF ((SXRAY(J)='4399') OR ('4240'<=SXRAY(J)<='4242')) THEN VL4=1;
END;
DXCA=0;
DO I=1 TO 10;
  IF ('150'<=DXRAY(I)<='1509') THEN DXCA=1;
END;
IF ((VL4=1) AND
    (DXCA=1)) THEN OC4=1;
*IF VL4=1 THEN OUTPUT VLOUT.VL&YR.PX4;
*IF OC4=1 THEN OUTPUT OCOUT.OC&YR.PX4;


*** DEFINING LEBY ***;
SXLEBY=0;
DO I=1 TO 6;
  IF SXRAY(I)='3929' THEN SXLEBY=1;
END;
DXUPPER=0;
DO I=1 TO 10;
  IF DXRAY(I) = '44421' THEN DXUPPER=1;
END;
IF ((SXLEBY=1) AND
    (DXUPPER=0) AND
    (ESRD =0)) THEN DO;
    VL5=1;
    OC5=1;
```

```
END;
*IF VL5=1 THEN OUTPUT VLOUT.VL&YR.PX5;
*IF OC5=1 THEN OUTPUT OCOUT.OC&YR.PX5;


*** DEFINING PANCREATIC RESECTION ***;
DO I=1 TO 6;
  IF (SXRAY(I) IN ('5251','5253','527')) THEN VL6=1;
END;
DXCA=0;
DO I=1 TO 10;
  IF ('152'<=DXRAY(I)<='1529') THEN DXCA=1;
```

```
  IF ('156'<=DXRAY(I)<='1579') THEN DXCA=1;
END;
IF ((VL6=1) AND
    (DXCA=1)) THEN OC6=1;
*IF VL6=1 THEN OUTPUT VLOUT.VL&YR.PX6;
*IF OC6=1 THEN OUTPUT OCOUT.OC&YR.PX6;


*** DEFINING HEART VALVE REPLACEMENTS ***;
DO I=1 TO 6;
  IF ('3520'<=SXRAY(I)<='3528') THEN VL7=1;
                  /*ANY VALVE REPLACEMENT*/
  IF SXRAY(I) IN ('3521','3522') THEN OC7=1;
                  /*AORTIC VALVE REPLACEMENT*/
  IF SXRAY(I) IN ('3523','3524') THEN OC8=1;
                  /*MITRAL VALVE REPLACEMENT*/
END;
*IF VL7=1 THEN OUTPUT VLOUT.VL&YR.PX7;
*IF OC7=1 THEN OUTPUT OCOUT.OC&YR.PX7;
*IF OC8=1 THEN OUTPUT OCOUT.OC&YR.PX8;


*** DEFINING LUNG RESECTIONS ***;
LOBE=0;
PNEUM=0;
DO I=1 TO 6;
  IF SXRAY(I)='324' THEN LOBE=1;
  IF SXRAY(I)='325' THEN PNEUM=1;
END;
IF ((LOBE=1) OR (PNEUM=1)) THEN VL9=1;
DXCA=0;
DO I=1 TO 10;
  IF ('162'<=DXRAY(I)<='1659') THEN DXCA=1;
END;
IF ((DXCA=1) AND (PNEUM=1)) THEN OC9=1;
IF ((DXCA=1) AND (LOBE=1)) THEN OC10=1;

*IF VL9=1 THEN OUTPUT VLOUT.VL&YR.PX9;
*IF OC9=1 THEN OUTPUT OCOUT.OC&YR.PX9;
*IF OC10=1 THEN OUTPUT OCOUT.OC&YR.PX10;

*** DEFINING COLECTOMY ***;
DO I=1 TO 6;
  IF ('4573'<=SXRAY(I)<='4576') THEN VL11=1;
```

```
END;
DXCA=0;
DO I=1 TO 10;
  IF ('153'<=DXRAY(I)<='1540') THEN DXCA=1;
END;
IF ((VL11=1)      AND
    (DXCA=1)) THEN OC11=1;

*IF VL4=11 THEN OUTPUT VLOUT.VL&YR.PX11;
*IF OC4=11 THEN OUTPUT OCOUT.OC&YR.PX11;


*** DEFINING CYSTECTOMY ***;
DO I=1 TO 6;
  IF ('5770'<=SXRAY(I)<='5779') THEN VL12=1;
```

```

END;
DXCA=0;
DO I=1 TO 10;
  IF ('188'<=DXRAY(I)<='1889') THEN DXCA=1;
  IF ('189'<=DXRAY(I)<='1899') THEN DXCA=1;
END;
IF ((VL12=1)     AND
    (DXCA=1)) THEN OC12=1;
*IF VL5=12 THEN OUTPUT VLOUT.VL&YR.PX12;
*IF OC5=12 THEN OUTPUT OCOUT.OC&YR.PX12;


*** DEFINING GASTRECTOMY ***;
DO J=1 TO 6;
  IF ('435'<=SXRAY(J)<='4399') THEN VL13=1;
END;
DXCA=0;
DO I=1 TO 10;
  IF ('151'<=DXRAY(I)<='1519') THEN DXCA=1;
END;
IF ((VL13=1) AND
    (DXCA=1)) THEN OC13=1;
*IF VL7=13 THEN OUTPUT VLOUT.VL&YR.PX13;
*IF OC7=13 THEN OUTPUT OCOUT.OC&YR.PX13;


*** DEFINING NEPHRECTOMY ***;
SXNEPH1=0;
DO I=1 TO 6;
  IF (SXRAY(I) IN ('5551','5552')) THEN VL14=1;
  IF SXRAY(I)='5551' THEN SXNEPH1=1;
END;
DXCA=0;
DO I=1 TO 10;
  IF ('189'<=DXRAY(I)<='1899') THEN DXCA=1;
END;
IF ((SXNEPH1=1)  AND
    (DXCA=1)) THEN OC14=1;
*IF VL9=14 THEN OUTPUT VLOUT.VL&YR.PX14;
*IF OC9=14 THEN OUTPUT OCOUT.OC&YR.PX14;
```

```
RUN;


 /*******************************************************/
 /**            PRODUCE VOLUME FILES BY YEAR           **/
 /**  READ EACH RECORD AND PULL THE ONES WITH VLNN=1    **/
 /*******************************************************/

DATA VLOUT.VL&YR.PX;
  SET TEMP1.TEMP;

 ARRAY VOL VL1 VL2 VL3 VL4 VL5 VL6 VL7
           VL9 VL11 VL12 VL13 VL14
           ;

   /***FOLLOWING CODE READS EACH RECORD. THE ONES THAT MEET THE***/
   /***CRITERIA (VLNN=1) ARE PULLED INTO NEW DATA FILE OCYYPX***/
   /***AND ONES THAT DO NOT MEET THE CRITERIA ARE SKIPED.***/
D#MSQ.@MSQ01.PGMLIB(MSQYH102) -- CONTINUED


 DO I = 1 TO 12;
    IF VOL(I)=1 THEN DO;
    OUTPUT;
    GOTO SKIP;
    END;
 END;

 SKIP:

 RUN;

 /*******************************************************/
 /**           PRODUCE OUTCOME FILES BY YEAR           **/
 /**  READ EACH RECORD AND PULL THE ONES WITH OCNN=1    **/
 /*******************************************************/
DATA OCOUT.OC&YR.PX;
  SET TEMP1.TEMP;

 ARRAY OCM OC1  OC2  OC3  OC4  OC5  OC6  OC7
           OC8  OC9  OC10 OC11 OC12 OC13 OC14
           ;

   /***FOLLOWING CODE READS EACH RECORD. THE ONES THAT MEET THE***/
   /***CRITERIA (OCNN=1) ARE PULLED INTO NEW DATA FILE OCYYPX***/
   /***AND ONES THAT DO NOT MEET THE CRITERIA ARE SKIPED.***/

 DO I=1 TO 14;
    IF OCM(I)=1 THEN DO;
    OUTPUT;
    GOTO SKIP;
    END;
 END;

 SKIP:

 RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#2H20)

```
//L1982H20 JOB (MSQ01,MQMS),'STP2_1_92',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ2H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H20D)

```
//MSQ2H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="92"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL92PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL92ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H20)

```
//L1983H20 JOB (MSQ01,MQMS),'STP2_1_93',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ3H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ3H20D)

```
//MSQ3H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="93"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL93PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL93ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H20)

```
//L1984H20 JOB (MSQ01,MQMS),'STP2_1_94',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ4H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H20D)

```
//MSQ4H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="94"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL94PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL94ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H20)

```
//L1985H20 JOB (MSQ01,MQMS),'STP2_1_95',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ5H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H20D)

```
//MSQ5H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="95"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL95PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL95ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H20)

```
//L1986H20 JOB (MSQ01,MQMS),'STP2_1_96',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ6H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H20D)

```
//MSQ6H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="96"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL96PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL96ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H20)

```
//L1987H20 JOB (MSQ01,MQMS),'STP2_1_97',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ7H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H20D)

```
//MSQ7H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="97"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL97PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL97ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H20)

```
//L1988H20 JOB (MSQ01,MQMS),'STP2_1_98',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ8H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H20D)

```
//MSQ8H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="98"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL98PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL98ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H20)

```
//L1989H20 JOB (MSQ01,MQMS),'STP2_1_99',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ9H20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H20D)

```
//MSQ9H20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="99"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL99PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL99ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH20)

```
//L198AH20 JOB (MSQ01,MQMS),'STP2_1_00',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQAH20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH20D)

```
//MSQAH20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="00"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL00PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL00ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH20)

```
//L198BH20 JOB (MSQ01,MQMS),'STP2_1_01',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQBH20D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH20D)

```
//MSQBH20D PROC
//*
//PS010  EXEC SAS8,PARM='SYSPARM="01"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH102.VL01PX,
//          DISP=(SHR,KEEP,KEEP)
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL01ALL,
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH202),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQYH202)

```
  /*************************************************************/
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH202)               */
  /*-----------------------------------------------------------*/
  /* PURPOSE :     CALCULATE THE TOTAL VOLUME OF 12 SURGICAL   */
  /*               PROCEDURES PERFORMED IN EACH HOSPITAL       */
  /*-----------------------------------------------------------*/
  /* INPUT DATA :  P#MSQ.@MSQ01.MSGYH102.VLYYPX                */
  /*               WHERE YY=92-01, TOTAL 10 INPUT FILES        */
  /*-----------------------------------------------------------*/
  /* OUTPUT DATA : P#MSQ.@MSQ01.MSGYH202.PVLYYALL              */
  /*               WHERE YY=92-01, TOTAL 10 OUTPUT FILES       */
  /*-----------------------------------------------------------*/
  /* KEY VARIABLES PRODUCED: SUMVLN                            */
  /*                         WHERE N=1-7,9,11-13               */
  /*-----------------------------------------------------------*/
  /* WRITTEN BY    LUCY LU                                     */
  /* CONTACT PERSON: ANGELA SCHMITT                            */
  /*                 MATHEMATICA POLICY RESEARCH, INC.         */
  /*                 PHONE 202-484-4517                        */
  /*-----------------------------------------------------------*/
  /* DATE:         APR. 16, 2003                               */
  /*************************************************************/

 OPTIONS NOCENTER NOBYLINE;


 %LET YR=&SYSPARM;

 PROC SUMMARY DATA=IN.VL&YR.PX;
 CLASS Y&YR.PRVID YEAR;
 VAR VL1 VL2 VL3 VL4
     VL5 VL6 VL7 VL9
     VL11 VL12 VL13 VL14
     ;
OUTPUT OUT=OUT.PVL&YR.ALL SUM=
     SUMVL1  SUMVL2  SUMVL3  SUMVL4
     SUMVL5  SUMVL6  SUMVL7  SUMVL9
     SUMVL11 SUMVL12 SUMVL13 SUMVL14
     ;
RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#YH22)

```
//L198YH22 JOB (MSQ01),'STP2_2',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
// EXEC SAS8
//IN92   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL92ALL,DISP=SHR
//IN93   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL93ALL,DISP=SHR
//IN94   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL94ALL,DISP=SHR
//IN95   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL95ALL,DISP=SHR
//IN96   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL96ALL,DISP=SHR
//IN97   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL97ALL,DISP=SHR
//IN98   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL98ALL,DISP=SHR
//IN99   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL99ALL,DISP=SHR
//IN00   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL00ALL,DISP=SHR
//IN01   DD DSN=D#MSQ.@MSQ01.MSQYH202.PVL01ALL,DISP=SHR
//Y94    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT94MEA,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y95    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT95MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y96    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT96MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT97MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y98    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT98MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y99    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT99MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y00    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT00MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT01MEA,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//SYSIN  DD *

  /*************************************************************/
  /* PROGRAM :     D#MSQ.@MSQ01.JCLTEST(MSQ#YH22)              */
  /*-----------------------------------------------------------*/
  /* PURPOSE :     CALCULATE THREE YEAR MOVING AVERAGE VOLUME OF   */
  /*               PROCEDURES PERFORMED IN EACH HOSPITAL           */
  /*-----------------------------------------------------------*/
  /* INPUT DATA :  D#MSQ.@MSQ01.MSGYH202.PVLYYALL              */
  /*               WHERE YY=92-01, TOTAL 10 INTPUT FILES       */
  /*-----------------------------------------------------------*/
  /* OUTPUT DATA : D#MSQ.@MSQ01.MSGYH202.QNTYYMEA              */
  /*               WHERE YY=94-01                              */
  /*-----------------------------------------------------------*/
  /* KEY VARIABLES PRODUCED: MEAN VOLUME FOR 3 YR: MEAYYVN     */
  /*                         WHERE N=1-7, 9, 11-14;  YY=94-01  */
  /*-----------------------------------------------------------*/
```

```
  /* WRITTEN BY:   LUCY LU                                          */
  /* CONTACT PERSON:  ANGELA SCHMITT                                */
  /*               MATHEMATICA POLICY RESEARCH, INC.                */
D#MSQ.@MSQ01.JCLTEST(MSQ#YH22) -- CONTINUED

  /*               PHONE 202-484-4517                               */
  /*---------------------------------------------------------------*/
  /* DATE:         APR. 24, 2003                                    */
  /*****************************************************************/

 OPTIONS NOCENTER MPRINT;

  /*********************************************************/
  /**  PULL THREE YEAR VOLUME FILES EACH TIME AND COMPUTE  **/
  /**  THE THREE YEAR MOVING AVERAGE.                      **/
  /*********************************************************/

%MACRO STP2_2(YR1,YR2,YR3,YRINDX);

 DATA TEMP;
   SET IN&YR1..PVL&YR1.ALL(RENAME=(Y&YR1.PRVID=PRVID))
       IN&YR2..PVL&YR2.ALL(RENAME=(Y&YR2.PRVID=PRVID))
       IN&YR3..PVL&YR3.ALL(RENAME=(Y&YR3.PRVID=PRVID))
       ;

 INDXYR=&YRINDX;

 RUN;


 PROC MEANS DATA=TEMP NWAY NOPRINT;
 CLASS PRVID INDXYR;
 VAR SUMVL1  SUMVL2  SUMVL3  SUMVL4
     SUMVL5  SUMVL6  SUMVL7  SUMVL9
     SUMVL11 SUMVL12 SUMVL13  SUMVL14
     ;
 OUTPUT OUT=Y&YR3..QNT&YR3.MEA(DROP=_FREQ_ _TYPE_) MEAN=;
 RUN;

PROC PRINT DATA=Y&YR3..QNT&YR3.MEA(OBS=10);
RUN;
%MEND;

%STP2_2(92,93,94,1994);
%STP2_2(93,94,95,1995);
%STP2_2(94,95,96,1996);
%STP2_2(95,96,97,1997);
%STP2_2(96,97,98,1998);
%STP2_2(97,98,99,1999);
%STP2_2(98,99,00,2000);
%STP2_2(99,00,01,2001);
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H24)

```
//L1984H24 JOB (MSQ01,MQMS),'STP2_3_94',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ4H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H24D)

```
//MSQ4H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="94"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT94MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV94ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM94ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H24)

```
//L1985H24 JOB (MSQ01,MQMS),'STP2_3_95',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ5H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H24D)

```
//MSQ5H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="95"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT95MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV95ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM95ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H24)

```
//L1986H24 JOB (MSQ01,MQMS),'STP2_3_96',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ6H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H24D)

```
//MSQ6H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="96"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT96MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV96ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM96ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H24)

```
//L1987H24 JOB (MSQ01,MQMS),'STP2_3_97',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ7H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H24D)

```
//MSQ7H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="97"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT97MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV97ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM97ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H24)

```
//L1988H24 JOB (MSQ01,MQMS),'STP2_3_98',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ8H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H24D)

```
//MSQ4H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="98"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT98MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV98ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM98ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H24)

```
//L1989H24 JOB (MSQ01,MQMS),'STP2_3_99',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQ9H24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H24D)

```
//MSQ9H24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="99"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT99MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV99ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM99ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH24)

```
//L198AH24 JOB (MSQ01,MQMS),'STP2_3_00',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQAH24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH24D)

```
//MSQAH24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="00"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT00MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV00ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM00ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH24)

```
//L198BH24 JOB (MSQ01,MQMS),'STP2_3_01',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS024   EXEC MSQBH24D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH24D)

```
//MSQBH24D PROC
//*
//PS024  EXEC SAS8,PARM='SYSPARM="01"'
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH202.QNT01MEA,
//          DISP=(SHR,KEEP,KEEP)
//OUT1   DD DSN=D#MSQ.@MSQ01.MSQYH204.QV01ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//OUT2   DD DSN=D#MSQ.@MSQ01.MSQYH204.QM01ALL,
//          DISP=(OLD,CATLG,DELETE),
//          UNIT=HCFA80,SPACE=(CYL,(100,50),RLSE)
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH204),
//          DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQYH204)

```
  /*************************************************************/
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH204)               */
  /*-----------------------------------------------------------*/
  /* PURPOSE :     CREATINGE FIVE CATEGORICAL GROUPS FOR EACH  */
  /*               PROCEDURE VOLUME FROM THREE-YEAR MOVING AVERAGE.*/
  /*               FIVE CATEGORIES IS DEFINDED BY VOLUMES FROM HIGH*/
  /*               TO LOW. CALCULATING MEAN, MAX, AND MIN FOR EACH */
  /*               CATEGORY.                                   */
  /*-----------------------------------------------------------*/
  /* INPUT DATA :  D#MSQ.@MSQ01.MSGYH202.QNTYYMEA              */
  /*               WHERE YY=94-01, TOTAL 8  INTPUT FILES       */
  /*-----------------------------------------------------------*/
  /* OUTPUT DATA : TWO SETS OF OUTPUT FILES WILL BE PRODUCED   */
  /*           1.  SET ONE--OUT PUT FILES WITH PROVIDER ID FOR */
  /*               STEP 3:                                     */
  /*               DSN=D#MSQ.@MSQ01.MSQYH204.QVYYALL.QNTYYVN   */
  /*               WHERE YY=94-01, N=1-7, 9, 11-14             */
  /*           2.  SET TWO--OUT PUT FILES WITH MEAN, MAX, MIN IN */
  /*               EACH QUINTILE.                              */
  /*               DSN=D#MSQ.@MSQ01.MSQYH204.QMYYALL.MEAYYVN   */
  /*               WHERE YY=94-01, N=1-7, 9, 11-14             */
  /*-----------------------------------------------------------*/
  /* KEY VARIABLES PRODUCED: QNTN WITH VALUE OF 1-5 FOR EACH VOLUME*/
  /*                              WHERE N=1-7,9 11-14          */
  /*-----------------------------------------------------------*/
  /* WRITTEN BY:   LUCY LU                                     */
  /* CONTACT PERSON:  ANGELA SCHMITT                           */
  /*               MATHEMATICA POLICY RESEARCH, INC.           */
  /*               PHONE 202-484-4517                          */
  /*-----------------------------------------------------------*/
  /* DATE:         APR. 30, 2003                               */
  /*************************************************************/

 OPTIONS NOCENTER MPRINT;

%LET YR  = &SYSPARM;


  /* FOLLOWING MACRO CREATES 5 VOLUMES CATEGORIES FROM HIGH TO LOW.*/
  /* SINCE EACH PROCEDURE VOLUME HAS DIFFERENT QUINTILES, THE    */
  /* QUINTILE VARIABLES ARE NAMED QNT1-QNT14 IN COORDINATE WITH  */
  /* THREE YEAR MOVING AVERAGE VOLUME VARIABLES SUMVL1-SUMVL14.  */

 %MACRO QNT(I);

 PROC SORT DATA=IN.QNT&YR.MEA(KEEP=PRVID SUMVL&I) OUT=TEMP;
 BY DESCENDING SUMVL&I;
 RUN;

DATA TEMP;
  SET TEMP;

 IF SUMVL&I>0;

TOTVL&I+SUMVL&I;
```

```
RUN;

PROC SQL;
```
D#MSQ.@MSQ01.PGMLIB(MSQYH204) -- CONTINUED
```
CREATE TABLE TEMP2 AS
SELECT*, SUM(SUMVL&I) AS SUMSURG
FROM TEMP;
QUIT;
RUN;

DATA TEMP3;
  SET TEMP2;

N=ROUND(SUMSURG/5);

IF TOTVL&I<=N THEN QNT&I=1;
ELSE IF N<TOTVL&I<=N*2 THEN QNT&I=2;
ELSE IF N*2<TOTVL&I<=N*3 THEN QNT&I=3;
ELSE IF N*3<TOTVL&I<=N*4 THEN QNT&I=4;
ELSE QNT&I=5;

RUN;
      /* THIS CODE FOR INITIAL QA PURPOSE. WE DON'T NEED IT NOW.
PROC PRINT DATA=TEMP3(OBS=5);
WHERE QNT&I =1;
PROC PRINT DATA=TEMP3(OBS=5);
WHERE QNT&I =3;
RUN; */

  /*FOR THE PROVIDERS WITH SAME PROCEDURE VOLUMES BUT CUT OFF BY*/
  /*DIFFERENT CATEGORIES, WE NEED TO MOVE THOSE PROVIDERS FROM  */
  /*HIGHER CATEGORY TO LOWER CATEGORY SO THAT THE PROVIDERS WITH*/
  /*SAME VOLUMES STAY IN ONE CATEGORY.                          */

 /*CREATE A DUMMY DATA,CHECK FOR PROVIDERS WITH SAME VOL BUT DIFF CAT*/
 DATA TEMP4;
  SET TEMP3(KEEP=SUMVL&I QNT&I);
   *FORMAT SUMVL&I 22.18;

  /*DELETE ONES WITH SAME VOLUMES AND SAME CATEGORIES IN DUMMY DATA*/
 PROC SORT DATA=TEMP4 NODUPKEY;
 BY SUMVL&I QNT&I;
 RUN;


  /*TAKE THE ONES WITH LOWER CATEGORY IF PROVIDERS WITH SAME VOLUMES
    IN TWO OR MORE CATEGORIES*/

 PROC SORT DATA=TEMP4;
 BY DESCENDING SUMVL&I;
 RUN;

 DATA TEMP4;
   SET TEMP4;
   BY DESCENDING SUMVL&I QNT&I;

 IF NOT (FIRST.SUMVL&I AND LAST.SUMVL&I);
```

```
 PROC PRINT;
 TITLE "SAME PROCEDURE VOLUMES BUT IN DIFFERENT CATEGORIES";
 RUN;

 DATA TEMP4;
```
D#MSQ.@MSQ01.PGMLIB(MSQYH204) -- CONTINUED
```
   SET TEMP4;
   BY DESCENDING SUMVL&I QNT&I;

 IF LAST.SUMVL&I;

 NEWSUMVL=SUMVL&I;
 KEEP SUMVL&I QNT&I NEWSUMVL;

 PROC PRINT;
 TITLE "CHANGE THOSE TO LOWER CATEGORIES";

 RUN;

  /*MERGE TWO DATA SETS, CORRECTE CATEGORIES IN TENP3*/

 DATA TEMP5;
  MERGE TEMP3(IN=MAIN)
        TEMP4(IN=TRANS RENAME=(QNT&I=NEWQNT))
        ;
  BY DESCENDING SUMVL&I;

 IF MAIN;
 IF SUMVL&I=NEWSUMVL THEN QNT&I=NEWQNT;

 DROP NEWSUMVL NEWQNT;

 RUN;

 PROC MEANS DATA=TEMP5 MEAN MAX MIN NWAY;
 CLASS QNT&I;
 VAR SUMVL&I;
 OUTPUT OUT=OUT2.MEA&YR.VL&I(DROP=_TYPE_ _FREQ_)
       MAX=MAX&YR.VL&I
       MIN=MIN&YR.VL&I
       MEAN=MEA&YR.VL&I
       ;
 TITLE "YEAR &YR, MEAN, MAX, MIN FROM EACH QUINTILE";
 RUN;

PROC SORT DATA=TEMP5(KEEP=PRVID SUMVL&I QNT&I)
OUT=OUT1.QNT&YR.V&I;
BY PRVID;
RUN;

%MEND;

%QNT(1);
%QNT(2);
%QNT(3);
%QNT(4);
```

```
%QNT(5);
%QNT(6);
%QNT(7);
%QNT(9);
%QNT(11);
%QNT(12);
%QNT(13);
%QNT(14);
```

D#MSQ.@MSQ01.JCLTEST(MSQ#YH30)

```
//L198YH30 JOB (MSQ01,MQMS),'STEP 3 PRE-MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 13 QUINTILE FILES FOR EACH YEAR                 *
//*************************************************************
//JS010   EXEC MSQYH30D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQYH30D)

```
//MSQYH30D PROC
//*
//PS010  EXEC SAS8
//INQNT94  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV94ALL, *1994 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT95  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV95ALL, *1995 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT96  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV96ALL, *1996 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT97  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV97ALL, *1997 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT98  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV98ALL, *1998 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT99  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV99ALL, *1999 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT00  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV00ALL, *2000 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT01  DD DSN=D#MSQ.@MSQ01.MSQYH204.QV01ALL, *2001 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OUTQNT94 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT94ALL, *1994 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT95 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT95ALL, *1995 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT96 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT96ALL, *1996 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT97 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT97ALL, *1997 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT98 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT98ALL, *1998 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT99 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT99ALL, *1999 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT00 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT00ALL, *2000 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//OUTQNT01 DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT01ALL, *2001 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH302),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQYH302)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *CREATE A MACRO LOOP THAT COMBINES THE 13 QUINTLE * **/
/** *FILES PER YEAR INTO 1 FILE PER YEAR              * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: QUINTILE FILES FROM STEP 2         * **/
/** * OUTPUT FILES: 1 QUINTILE FILE PER YEAR          * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
OPTIONS SYMBOLGEN MPRINT;


 /* MERGE THE 12 QUINTILE FILES INTO 1 QUINTILE FILE FOR EACH YR  */
%MACRO MERGE(OUTDS, INDS1, INDS2, INDS3, INDS4, INDS5, INDS6,
             INDS7, INDS9, INDS11, INDS12, INDS13, INDS14, RNMYR);
DATA &OUTDS
(RENAME=(SUMVL1=Y&RNMYR.SUMVL1
         SUMVL2=Y&RNMYR.SUMVL2
         SUMVL3=Y&RNMYR.SUMVL3
         SUMVL4=Y&RNMYR.SUMVL4
         SUMVL5=Y&RNMYR.SUMVL5
         SUMVL6=Y&RNMYR.SUMVL6
         SUMVL7=Y&RNMYR.SUMVL7
         SUMVL9=Y&RNMYR.SUMVL9
         SUMVL11=Y&RNMYR.SUMVL11
         SUMVL12=Y&RNMYR.SUMVL12
         SUMVL13=Y&RNMYR.SUMVL13
         SUMVL14=Y&RNMYR.SUMVL14
         QNT1=Y&RNMYR.QNT1
         QNT2=Y&RNMYR.QNT2
         QNT3=Y&RNMYR.QNT3
         QNT4=Y&RNMYR.QNT4
         QNT5=Y&RNMYR.QNT5
         QNT6=Y&RNMYR.QNT6
         QNT7=Y&RNMYR.QNT7
         QNT9=Y&RNMYR.QNT9
         QNT11=Y&RNMYR.QNT11
         QNT12=Y&RNMYR.QNT12
         QNT13=Y&RNMYR.QNT13
         QNT14=Y&RNMYR.QNT14));

    MERGE &INDS1 &INDS2 &INDS3 &INDS4 &INDS5 &INDS6
          &INDS7 &INDS9 &INDS11 &INDS12 &INDS13 &INDS14;
    BY PRVID;
RUN;

PROC PRINT DATA=&OUTDS(OBS=20); RUN;

%MEND MERGE;
                       /*** 1994 ***/
%MERGE(OUTDS=OUTQNT94.QNT94ALL,
       INDS1=INQNT94.QNT94V1,
```

```
        INDS2=INQNT94.QNT94V2,
        INDS3=INQNT94.QNT94V3,
        INDS4=INQNT94.QNT94V4,
D#MSQ.@MSQ01.PGMLIB(MSQYH302) -- CONTINUED

        INDS5=INQNT94.QNT94V5,
        INDS6=INQNT94.QNT94V6,
        INDS7=INQNT94.QNT94V7,
        INDS9=INQNT94.QNT94V9,
        INDS11=INQNT94.QNT94V11,
        INDS12=INQNT94.QNT94V12,
        INDS13=INQNT94.QNT94V13,
        INDS14=INQNT94.QNT94V14,
        RNMYR=94);
                        /*** 1995 ***/
 %MERGE(OUTDS=OUTQNT95.QNT95ALL,
        INDS1=INQNT95.QNT95V1,
        INDS2=INQNT95.QNT95V2,
        INDS3=INQNT95.QNT95V3,
        INDS4=INQNT95.QNT95V4,
        INDS5=INQNT95.QNT95V5,
        INDS6=INQNT95.QNT95V6,
        INDS7=INQNT95.QNT95V7,
        INDS9=INQNT95.QNT95V9,
        INDS11=INQNT95.QNT95V11,
        INDS12=INQNT95.QNT95V12,
        INDS13=INQNT95.QNT95V13,
        INDS14=INQNT95.QNT95V14,
        RNMYR=95);
                        /*** 1996 ***/
 %MERGE(OUTDS=OUTQNT96.QNT96ALL,
        INDS1=INQNT96.QNT96V1,
        INDS2=INQNT96.QNT96V2,
        INDS3=INQNT96.QNT96V3,
        INDS4=INQNT96.QNT96V4,
        INDS5=INQNT96.QNT96V5,
        INDS6=INQNT96.QNT96V6,
        INDS7=INQNT96.QNT96V7,
        INDS9=INQNT96.QNT96V9,
        INDS11=INQNT96.QNT96V11,
        INDS12=INQNT96.QNT96V12,
        INDS13=INQNT96.QNT96V13,
        INDS14=INQNT96.QNT96V14,
        RNMYR=96);
                        /*** 1997 ***/
 %MERGE(OUTDS=OUTQNT97.QNT97ALL,
        INDS1=INQNT97.QNT97V1,
        INDS2=INQNT97.QNT97V2,
        INDS3=INQNT97.QNT97V3,
        INDS4=INQNT97.QNT97V4,
        INDS5=INQNT97.QNT97V5,
        INDS6=INQNT97.QNT97V6,
        INDS7=INQNT97.QNT97V7,
        INDS9=INQNT97.QNT97V9,
        INDS11=INQNT97.QNT97V11,
        INDS12=INQNT97.QNT97V12,
        INDS13=INQNT97.QNT97V13,
        INDS14=INQNT97.QNT97V14,
```

```
       RNMYR=97);
                        /*** 1998 ***/
 %MERGE(OUTDS=OUTQNT98.QNT98ALL,
       INDS1=INQNT98.QNT98V1,
       INDS2=INQNT98.QNT98V2,
       INDS3=INQNT98.QNT98V3,
```
D#MSQ.@MSQ01.PGMLIB(MSQYH302) -- CONTINUED

```
       INDS4=INQNT98.QNT98V4,
       INDS5=INQNT98.QNT98V5,
       INDS6=INQNT98.QNT98V6,
       INDS7=INQNT98.QNT98V7,
       INDS9=INQNT98.QNT98V9,
       INDS11=INQNT98.QNT98V11,
       INDS12=INQNT98.QNT98V12,
       INDS13=INQNT98.QNT98V13,
       INDS14=INQNT98.QNT98V14,
       RNMYR=98);
                        /*** 1999 ***/
 %MERGE(OUTDS=OUTQNT99.QNT99ALL,
       INDS1=INQNT99.QNT99V1,
       INDS2=INQNT99.QNT99V2,
       INDS3=INQNT99.QNT99V3,
       INDS4=INQNT99.QNT99V4,
       INDS5=INQNT99.QNT99V5,
       INDS6=INQNT99.QNT99V6,
       INDS7=INQNT99.QNT99V7,
       INDS9=INQNT99.QNT99V9,
       INDS11=INQNT99.QNT99V11,
       INDS12=INQNT99.QNT99V12,
       INDS13=INQNT99.QNT99V13,
       INDS14=INQNT99.QNT99V14,
       RNMYR=99);
                        /*** 2000 ***/
 %MERGE(OUTDS=OUTQNT00.QNT00ALL,
       INDS1=INQNT00.QNT00V1,
       INDS2=INQNT00.QNT00V2,
       INDS3=INQNT00.QNT00V3,
       INDS4=INQNT00.QNT00V4,
       INDS5=INQNT00.QNT00V5,
       INDS6=INQNT00.QNT00V6,
       INDS7=INQNT00.QNT00V7,
       INDS9=INQNT00.QNT00V9,
       INDS11=INQNT00.QNT00V11,
       INDS12=INQNT00.QNT00V12,
       INDS13=INQNT00.QNT00V13,
       INDS14=INQNT00.QNT00V14,
       RNMYR=00);
                        /*** 2001 ***/
 %MERGE(OUTDS=OUTQNT01.QNT01ALL,
       INDS1=INQNT01.QNT01V1,
       INDS2=INQNT01.QNT01V2,
       INDS3=INQNT01.QNT01V3,
       INDS4=INQNT01.QNT01V4,
       INDS5=INQNT01.QNT01V5,
       INDS6=INQNT01.QNT01V6,
       INDS7=INQNT01.QNT01V7,
       INDS9=INQNT01.QNT01V9,
```

```
INDS11=INQNT01.QNT01V11,
INDS12=INQNT01.QNT01V12,
INDS13=INQNT01.QNT01V13,
INDS14=INQNT01.QNT01V14,
RNMYR=01);
```

D#MSQ.@MSQ01.JCLTEST(MSQ#2H34)

```
//L1982H34 JOB (MSQ01,MQMS),'1992 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1992 OUTCOMES FILE WITH THE 1994 QUINTILE FILE   *
//*************************************************************
//JS010   EXEC MSQ2H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H34D)

```
//MSQ2H34D PROC
//*
//PS010  EXEC SAS8
//INOC92   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC92PXR, *1992 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT94  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT94ALL, *1994 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC292    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC292ALL, *1992 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(100,50),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ2H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ2H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1992 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1992 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC292.OC292ALL;
    MERGE INOC92.OC92PXR(RENAME=(Y92PRVID=PRVID) IN=A)
          INQNT94.QNT94ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 92 OC2 FILES";
PROC PRINT DATA=OC292.OC292ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H34)

```
//L1983H34 JOB (MSQ01,MQMS),'1993 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1993 OUTCOMES FILE WITH THE 94,95 QNT FILES      *
//*************************************************************
//JS010   EXEC MSQ3H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ3H34D)

```
//MSQ3H34D PROC
//*
//PS010  EXEC SAS8
//INOC93   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC93PXR, *1993 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT94  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT94ALL, *1994 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT95  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT95ALL, *1995 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC293    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC293ALL, *1993 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(250,100),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ3H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ3H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1993 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1993 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC293.OC293ALL;
    MERGE INOC93.OC93PXR(RENAME=(Y93PRVID=PRVID) IN=A)
          INQNT94.QNT94ALL
          INQNT95.QNT95ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 93 OC2 FILES";
PROC PRINT DATA=OC293.OC293ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H34)

```
//L1984H34 JOB (MSQ01,MQMS),'1994 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1994 OUTCOMES FILE WITH THE 94,95,96 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ4H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H34D)

```
//MSQ4H34D PROC
//*
//PS010  EXEC SAS8
//INOC94   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC94PXR, *1994 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT94  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT94ALL, *1994 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT95  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT95ALL, *1995 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT96  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT96ALL, *1996 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC294    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC294ALL, *1994 OUTPUT*
//            DISP=(NEW,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,250),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ4H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ4H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1994 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1994 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC294.OC294ALL;
    MERGE INOC94.OC94PXR(RENAME=(Y94PRVID=PRVID) IN=A)
          INQNT94.QNT94ALL
          INQNT95.QNT95ALL
          INQNT96.QNT96ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 94 OC2 FILES";
PROC PRINT DATA=OC294.OC294ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H34)

```
//L1985H34 JOB (MSQ01,MQMS),'1995 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1995 OUTCOMES FILE WITH THE 95,96,97 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ5H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H34D)

```
//MSQ5H34D PROC
//*
//PS010  EXEC SAS8
//INOC95   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC95PXR, *1995 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT95  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT95ALL, *1995 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT96  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT96ALL, *1996 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT97  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT97ALL, *1997 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC295    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC295ALL, *1995 OUTPUT*
//            DISP=(NEW,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,250),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ5H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ5H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1995 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1995 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC295.OC295ALL;
    MERGE INOC95.OC95PXR(RENAME=(Y95PRVID=PRVID) IN=A)
          INQNT95.QNT95ALL
          INQNT96.QNT96ALL
          INQNT97.QNT97ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 95 OC2 FILES";
PROC PRINT DATA=OC295.OC295ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H34)

```
//L1986H34 JOB (MSQ01,MQMS),'1996 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1996 OUTCOMES FILE WITH THE 96,97,98 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ6H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H34D)

```
//MSQ6H34D PROC
//*
//PS010  EXEC SAS8
//INOC96   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC96PXR, *1996 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT96  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT96ALL, *1996 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT97  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT97ALL, *1997 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT98  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT98ALL, *1998 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC296    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC296ALL, *1996 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,200),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ6H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ6H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1996 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1996 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC296.OC296ALL;
    MERGE INOC96.OC96PXR(RENAME=(Y96PRVID=PRVID) IN=A)
          INQNT96.QNT96ALL
          INQNT97.QNT97ALL
          INQNT98.QNT98ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 96 OC2 FILES";
PROC PRINT DATA=OC296.OC296ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H34)

```
//L1987H34 JOB (MSQ01,MQMS),'1997 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1997 OUTCOMES FILE WITH THE 97,98,99 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ7H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H34D)

```
//MSQ7H34D PROC
//*
//PS010  EXEC SAS8
//INOC97   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC97PXR, *1997 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT97  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT97ALL, *1997 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT98  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT98ALL, *1998 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT99  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT99ALL, *1999 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC297    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC297ALL, *1997 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,200),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ7H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ7H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1997 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1997 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC297.OC297ALL;
    MERGE INOC97.OC97PXR(RENAME=(Y97PRVID=PRVID) IN=A)
          INQNT97.QNT97ALL
          INQNT98.QNT98ALL
          INQNT99.QNT99ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 97 OC2 FILES";
PROC PRINT DATA=OC297.OC297ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H34)

```
//L1988H34 JOB (MSQ01,MQMS),'1998 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1998 OUTCOMES FILE WITH THE 98,99,00 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ8H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H34D)

```
//MSQ8H34D PROC
//*
//PS010  EXEC SAS8
//INOC98   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC98PXR, *1998 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT98  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT98ALL, *1998 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT99  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT99ALL, *1999 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT00  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT00ALL, *2000 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC298    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC298ALL, *1998 OUTPUT*
//            DISP=(NEW,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,250),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ8H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ8H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1998 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1998 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC298.OC298ALL;
    MERGE INOC98.OC98PXR(RENAME=(Y98PRVID=PRVID) IN=A)
          INQNT98.QNT98ALL
          INQNT99.QNT99ALL
          INQNT00.QNT00ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 98 OC2 FILES";
PROC PRINT DATA=OC298.OC298ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H34)

```
//L1989H34 JOB (MSQ01,MQMS),'1999 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 1999 OUTCOMES FILE WITH THE 99,00,01 QNT FILES   *
//*************************************************************
//JS010   EXEC MSQ9H34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H34D)

```
//MSQ9H34D PROC
//*
//PS010  EXEC SAS8
//INOC99   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC99PXR, *1999 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT99  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT99ALL, *1999 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT00  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT00ALL, *2000 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT01  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT01ALL, *2001 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC299    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC299ALL, *1999 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(250,100),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQ9H304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQ9H304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 1999 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 1999 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC299.OC299ALL;
    MERGE INOC99.OC99PXR(RENAME=(Y99PRVID=PRVID) IN=A)
          INQNT99.QNT99ALL
          INQNT00.QNT00ALL
          INQNT01.QNT01ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 99 OC2 FILES";
PROC PRINT DATA=OC299.OC299ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH34)

```
//L198AH34 JOB (MSQ01,MQMS),'2000 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 2000 OUTCOMES FILE WITH THE 00,01 QNT FILES      *
//*************************************************************
//JS010   EXEC MSQAH34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH34D)

```
//MSQAH34D PROC
//*
//PS010  EXEC SAS8
//INOC00   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC00PXR, *2000 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT00  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT00ALL, *2000 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//INQNT01  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT01ALL, *2001 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC200    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC200ALL, *2000 OUTPUT*
//            DISP=(OLD,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,250),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQAH304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQAH304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 2000 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 2000 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC200.OC200ALL;
    MERGE INOC00.OC00PXR(RENAME=(Y00PRVID=PRVID) IN=A)
          INQNT00.QNT00ALL
          INQNT01.QNT01ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 00 OC2 FILES";
PROC PRINT DATA=OC200.OC200ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH34)

```
//L198BH34 JOB (MSQ01,MQMS),'2001 STEP 3 MERGE',
//             CLASS=J,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT MERGES *
//*       THE 2001 OUTCOMES FILE WITH THE 2001 QUINTILE FILE   *
//*************************************************************
//JS010   EXEC MSQBH34D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH34D)

```
//MSQBH34D PROC
//*
//PS010  EXEC SAS8
//INOC01   DD DSN=D#MSQ.@MSQ01.MSQYH102.OC01PXR, *2001 OUTCOMES R*
//            DISP=(SHR,KEEP,KEEP)
//INQNT01  DD DSN=D#MSQ.@MSQ01.MSQYH302.QNT01ALL, *2001 QUINTILES*
//            DISP=(SHR,KEEP,KEEP)
//OC201    DD DSN=D#MSQ.@MSQ01.MSQYH304.OC201ALL, *2001 OUTPUT*
//            DISP=(NEW,CATLG,DELETE),
//            UNIT=PROD,
//            SPACE=(CYL,(450,250),RLSE)
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQBH304),
//            DISP=(SHR,PASS,KEEP)
//*
```

D#MSQ.@MSQ01.PGMLIB(MSQBH304)

```
/** ************************************************ **/
/** *PROGRAMMING FOR PROJECT 8908, TASK 006, STEP 3   * **/
/** *MERGE 2001 OUTCOME AND QUINTILE FILES            * **/
/** *-----------------------------------------------* **/
/** * INPUT FILES: OC FILE (STEP 2), QNT FILE (STEP 3)* **/
/** * OUTPUT FILES: 2001 OC2 FILE                     * **/
/** *-----------------------------------------------* **/
/** *PROGRAMMER: MATTHEW R. GILLINGHAM                * **/
/** *SENIOR PROGRAMMER                               * **/
/** *MATHEMATICA POLICY RESEARCH, INC.                * **/
/** *CONTACT: ANGELA SCHMITT AT 202/484-4517          * **/
/** ************************************************ **/
DATA OC201.OC201ALL;
    MERGE INOC01.OC01PXR(RENAME=(Y01PRVID=PRVID) IN=A)
          INQNT01.QNT01ALL;
    BY PRVID;
    IF A;
RUN;

 TITLE "PRINT OF 01 OC2 FILES";
PROC PRINT DATA=OC201.OC201ALL(OBS=20); RUN;
```

D#MSQ.@MSQ01.JCLTEST(MSQ#2H40)

```
//L1982H40 JOB (MSQ01,MQMS),'STP4_1_92',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ2H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H40D)

```
//MSQ2H40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="92"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026062
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027063
//XRF92    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR92,                     00028064
//            DISP=(SHR,KEEP,KEEP)                                       00029063
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC292ALL,       *1992 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y92,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280067
//          SPACE=(CYL,(600,500),RLSE)                                   00300059
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF92ALL,                         00310071
//          DISP=(NEW,CATLG,DELETE),                                     00320070
//          SPACE=(CYL,(200,150),RLSE)                                   00322065
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00323065
//          DISP=(SHR,PASS,KEEP)                                         00324065
//*                                                                      00325065
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H40)

```
//L1983H40 JOB (MSQ01,MQMS),'STP4_1_93',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ3H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ3H40D)

```
//MSQ3H40D PROC                                                          00010061
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023062
//             PARM='SYSPARM="93"'                                       00025061
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026063
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027064
//XRF93    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR93,                     00028064
//            DISP=(SHR,KEEP,KEEP)                                       00029064
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC293ALL,    *1993 OUTCOMES*       00110061
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y93,                           00270061
//          DISP=(NEW,CATLG,DELETE),                                     00280070
//          SPACE=(CYL,(850,350),RLSE)                                   00291070
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF93ALL,                         00300076
//          DISP=(NEW,CATLG,DELETE),                                     00301074
//          SPACE=(CYL,(200,150),RLSE)                                   00303069
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                           00310072
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H40)

```
//L1984H40 JOB (MSQ01,MQMS),'STP4_1_94',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ4H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H40D)

```
//MSQ4H40D PROC                                                          00010071
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023062
//             PARM='SYSPARM="94"'                                       00025071
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026063
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027064
//XRF94    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR94,                     00028071
//            DISP=(SHR,KEEP,KEEP)                                       00029064
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC294ALL,    *1994 OUTCOMES*       00110071
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y94,                           00270071
//          DISP=(NEW,CATLG,DELETE),                                     00280070
//          SPACE=(CYL,(850,350),RLSE)                                   00291070
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF94ALL,                         00300073
//          DISP=(NEW,CATLG,DELETE),                                     00301073
//          SPACE=(CYL,(200,150),RLSE)                                   00303069
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00310071
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H40)

```
//L1985H40 JOB (MSQ01,MQMS),'STP4_1_95',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ5H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H40D)

```
//MSQ5H40D PROC                                                          00010055
//*                                                                      00050056
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00060056
//             PARM='SYSPARM="95"'                                       00070056
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00080060
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00090061
//XRF95    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR95,    *1995 FINDER*    00100061
//            DISP=(SHR,KEEP,KEEP)                                       00101061
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC295ALL,       *1995 OUTCOMES*    00110055
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y95,                           00270058
//          DISP=(NEW,CATLG,DELETE),                                     00280065
//          SPACE=(CYL,(600,350),RLSE)                                   00290068
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF95ALL,                         00300069
//          DISP=(NEW,CATLG,DELETE),                                     00301068
//          SPACE=(CYL,(200,150),RLSE)                                   00302067
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00310054
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H40)

```
//L1986H40 JOB (MSQ01,MQMS),'STP4_1_96',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ6H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H40D)

```
//MSQ6H40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="96"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026062
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027063
//XRF96    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR96,    *1996 FINDER*    00028063
//            DISP=(SHR,KEEP,KEEP)                                       00029063
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC296ALL,       *1996 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y96,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280067
//          SPACE=(CYL,(500,350),RLSE)                                   00291065
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF96ALL,                         00300071
//          DISP=(NEW,CATLG,DELETE),                                     00301070
//          SPACE=(CYL,(200,150),RLSE)                                   00303064
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00304064
//          DISP=(SHR,PASS,KEEP)                                         00310064
//*                                                                      00320064
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H40)

```
//L1987H40 JOB (MSQ01,MQMS),'STP4_1_97',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ7H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H40D)

```
//MSQ7H40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="97"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026062
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027063
//XRF97    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR97,    *1997 FINDER*    00028063
//            DISP=(SHR,KEEP,KEEP)                                       00029063
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC297ALL,       *1997 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y97,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280067
//          SPACE=(CYL,(500,300),RLSE)                                   00291065
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF97ALL,                         00300071
//          DISP=(NEW,CATLG,DELETE),                                     00301070
//          SPACE=(CYL,(200,150),RLSE)                                   00303064
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00304064
//          DISP=(SHR,PASS,KEEP)                                         00310064
//*                                                                      00320064
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H40)

```
//L1988H40 JOB (MSQ01,MQMS),'STP4_1_98',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ8H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H40D)

```
//MSQ8H40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="98"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026062
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027063
//XRF98    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR98,    *1998 FINDER*    00028063
//            DISP=(SHR,KEEP,KEEP)                                       00029063
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC298ALL,       *1998 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y98,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280066
//          SPACE=(CYL,(600,500),RLSE)                                   00300059
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF98ALL,                         00301068
//          DISP=(NEW,CATLG,DELETE),                                     00302068
//          SPACE=(CYL,(200,150),RLSE)                                   00304064
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00310054
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H40)

```
//L1989H40 JOB (MSQ01,MQMS),'STP4_1_99',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ9H40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H40D)

```
//MSQ9H40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="99"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026062
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027063
//XRF99    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR99,    *1999 FINDER*    00028063
//            DISP=(SHR,KEEP,KEEP)                                       00029063
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC299ALL,       *1999 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y99,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280066
//          SPACE=(CYL,(600,500),RLSE)                                   00300059
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF99ALL,                         00301069
//          DISP=(NEW,CATLG,DELETE),                                     00302069
//          SPACE=(CYL,(200,150),RLSE)                                   00304064
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00310054
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH40)

```
//L198AH40 JOB (MSQ01,MQMS),'STP4_1_00',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQAH40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH40D)

```
//MSQAH40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="00"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026064
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00050065
//XRF00    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR00,    *2000 FINDER*    00060074
//            DISP=(SHR,KEEP,KEEP)                                       00070065
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC200ALL,       *2000 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y00,                           00270060
//          DISP=(OLD,CATLG,DELETE),                                     00280077
//          SPACE=(CYL,(800,500),RLSE)                                   00300072
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF00ALL,                         00301078
//          DISP=(NEW,CATLG,DELETE),                                     00302076
//          SPACE=(CYL,(200,150),RLSE)                                   00304068
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00310054
//          DISP=(SHR,PASS,KEEP)                                         00320052
//*                                                                      00330038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH40)

```
//L198BH40 JOB (MSQ01,MQMS),'STP4_1_01',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQBH40D
//
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH40D)

```
//MSQBH40D PROC                                                          00010060
//*                                                                      00020038
//PS010   EXEC SAS8,OPTIONS='NONEWS',                                    00023055
//             PARM='SYSPARM="01"'                                       00025060
//SAS.WORK DD SPACE=(CYL,(800,800),,,ROUND)                              00026063
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00027064
//XRF01    DD DSN=D#MSQ.@MSQ0000.SAS.M2XREF.FINDR01,    *2001 FINDER*    00028065
//            DISP=(SHR,KEEP,KEEP)                                       00029064
//IN     DD DSN=D#MSQ.@MSQ01.MSQYH304.OC201ALL,       *2001 OUTCOMES*    00110060
//          DISP=(SHR,KEEP,KEEP)                                         00120052
//OUT    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y01,                           00270060
//          DISP=(NEW,CATLG,DELETE),                                     00280070
//          SPACE=(CYL,(600,500),RLSE)                                   00300059
//XRF    DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF01ALL,                         00310072
//          DISP=(NEW,CATLG,DELETE),                                     00320072
//          SPACE=(CYL,(200,150),RLSE)                                   00340066
//SYSIN  DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH402),                          00350066
//          DISP=(SHR,PASS,KEEP)                                         00360066
//*                                                                      00370066
```

D#MSQ.@MSQ01.PGMLIB(MSQYH402)

```
  /*************************************************************/   00010091
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH402)               */   00020092
  /*-----------------------------------------------------------*/   00030091
  /* PURPOSE :     CREATINGE INDEX ADMISSION AND ID THE PROCEDURES */   00040095
  /*-----------------------------------------------------------*/   00050091
  /* INPUT DATA :  D#MSQ.@MSQ01.MSQYH304.OC2YYALL              */   00060092
  /*               WHERE YY=92-01, TOTAL 10 INTPUT FILES       */   00070092
  /*-----------------------------------------------------------*/   00080091
  /* OUTPUT DATA : 14 OUTCOME FILES PER YEAR FROM YEAR 94-01    */   00090092
  /*               DSN=D#MSQ.@MSQ01.MSQYH402.OC3YY             */   00100099
  /*               DATASET OCNYYY,  WHERE YY=92-01, N=1-14     */   00110092
  /*-----------------------------------------------------------*/   00150091
  /* WRITTEN BY:   LUCY LU, ERIC SCHONE                        */   00160092
  /* CONTACT PERSON:  ANGELA SCHMITT                           */   00170091
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   00180091
  /*               PHONE 202-484-4517                          */   00190091
  /*-----------------------------------------------------------*/   00200091
  /* DATE:         JUN. 13, 2003                               */   00210099
  /*************************************************************/   00220091
                                                                    00230091
OPTIONS NOCENTER MPRINT ;                                           00240099
                                                                    00250099
%LET YR  = &SYSPARM;                                                00260091
                                                                    00270091
 /****MERGE W/ FINDER FILE TO GET MASTER HIC NO.****/                00280099
                                                                    00281099
  PROC SORT DATA=IN.OC2&YR.ALL OUT=OC2&YR.ALL;                      00290099
  BY Y&YR.HICNO; RUN;                                               00320099
                                                                    00321599
DATA XRF&YR.ALL;                                                    00322099
    MERGE XRF&YR..FINDR&YR(IN=A                                     00323099
                RENAME=(XREF=Y&YR.HICNO HICNO=MASTRHIC))            00324099
          OC2&YR.ALL(IN=B KEEP=Y&YR.HICNO)                          00325099
          ;                                                         00326099
    BY Y&YR.HICNO;                                                  00327099
                                                                    00328099
 IF A AND B;                                                        00329099
                                                                    00329199
 RUN;                                                               00329299
                                                                    00329399
 /***PICK UP ALL THE RECORDS WITH SAME MASTER HIC***/                00329499
 /*** AND CREATE THE NEW FINDER FILE FOR TASK 6  ***/              00329799
  DATA A;                                                           00329899
    SET XRF&YR.ALL(KEEP=MASTRHIC);                                  00329999
  RUN;                                                              00330099
                                                                    00330199
  PROC SORT DATA=A NODUPKEY;                                        00330299
  BY MASTRHIC; RUN;                                                 00330399
                                                                    00330499
DATA B;                                                             00330599
  SET A;                                                            00330699
                                                                    00330799
FMTNAME="WANTED";                                                   00330899
TYPE="C";                                                           00330999
START=MASTRHIC;                                                     00331099
LABEL="%%";                                                         00331199
```

```
                                                                        00331299
RUN;                                                                    00331399

D#MSQ.@MSQ01.PGMLIB(MSQYH402) -- CONTINUED
                                                                        00331499
PROC FORMAT CNTLIN=B ;                                                  00331599
RUN;                                                                    00331699
                                                                        00331799
                                                                        00332899
 DATA TEMP;                                                             00332999
     SET XRF&YR..FINDR&YR(RENAME=(HICNO=MASTRHIC XREF=Y&YR.HICNO));     00333099
                                                                        00333199
                                                                        00333299
 IF PUT(MASTRHIC,$WANTED.)="%%" THEN OUTPUT;                            00333399
                                                                        00333499
 PROC SORT DATA=TEMP OUT=XRF.XRF&YR.ALL;                                00333599
 BY Y&YR.HICNO;                                                         00333699
 RUN;                                                                   00333799
                                                                        00333899
   /****NOW MERGE NEW FINDER FILE WITH OUTCOME FILE****/                 00333999
 DATA TEMP1.OC2&YR.ALL;                                                 00334099
      MERGE XRF.XRF&YR.ALL(IN=A)                                        00334199
            OC2&YR.ALL(IN=B);                                           00334299
   BY Y&YR.HICNO;                                                       00334399
   IF A AND B;                                                          00334499
 RUN;                                                                   00334599
                                                                        00334699
PROC DATASETS NOLIST;                                                   00334799
DELETE OC2&YR.ALL A XRF&YR.ALL TEMP;                                    00334899
RUN;                                                                    00334999
                                                                        00335099
                                                                        00335199
 /***** CREATE FIRST ADMISSION DATE AND PROCEDURE DATE*****/            00335299
                                                                        00336092
%MACRO ADMDCH(PXFLAG);                                                  00340095
                                                                        00370091
 DATA DS&PXFLAG;                                                        00380091
     SET TEMP1.OC2&YR.ALL;                                              00390099
     WHERE &PXFLAG=1;                                                   00400091
                                                                        00401099
 PROC SORT DATA=DS&PXFLAG OUT=TEMP1.DS&PXFLAG;                          00410099
 BY MASTRHIC Y&YR.SADM;                                                 00410199
 RUN;                                                                   00411099
                                                                        00412099
 DATA OUT.&PXFLAG.Y&YR;                                                 00420098
      SET TEMP1.DS&PXFLAG; BY MASTRHIC;                                 00430099
      IF FIRST.MASTRHIC;         /*FIRST ADMISSION FOR A MBR*/         00440099
                                                                        00441099
        DT&PXFLAG=MDY(12,31,YEAR);  /*DEFAULT PROCEDURE DATE*/         00441199
                                                                        00441299
    %IF %SUBSTR(&PXFLAG,3)=1 %THEN %DO;                                 00442099
                            /*AAA*/                                     00450099
      %DO J = 1 %TO 6;                                                  00451099
       IF Y&YR.SDT&J >0  THEN  DO;                                      00452099
      IF Y&YR.SPR&J. IN('3844','3925','3834','3845')                    00460099
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                     00460199
             THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                    00460299
         END;                                                           00461099
```

```
      %END;                                                              00462099
    %END;                                                                02260099
                                                                         02261099
     %IF %SUBSTR(&PXFLAG,3)=2 %THEN %DO;                                 02262099
                          /*CABG*/                                       02270099
```

D#MSQ.@MSQ01.PGMLIB(MSQYH402) -- CONTINUED

```
       %DO J = 1 %TO 6;                                                  02270199
       IF Y&YR.SDT&J >0  THEN  DO;                                       02270299
        IF ('3610'<=Y&YR.SPR&J.<='3619')                                 02271099
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                      02272099
           THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                       02273099
        END;                                                             02274099
       %END;                                                             02275099
     %END;                                                               02340099
                                                                         02340199
     %IF %SUBSTR(&PXFLAG,3)=3 %THEN %DO;                                 02341099
                          /*CEA*/                                        02350099
       %DO J = 1 %TO 6;                                                  02351099
        IF Y&YR.SDT&J >0  THEN  DO;                                      02352099
        IF Y&YR.SPR&J.='3812'                                            02360099
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                      02361099
           THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                       02362099
        END;                                                             02363099
       %END;                                                             02364099
     %END;                                                               02420099
                                                                         02420199
     %IF %SUBSTR(&PXFLAG,3)=4 %THEN %DO;                                 02420299
                          /*ESOP*/                                       02420399
       %DO J=1 %TO 6;                                                    02420499
        IF Y&YR.SDT&J >0  THEN  DO;                                      02420599
          IF ('4240'<=Y&YR.SPR&J.<='4242' OR Y&YR.SPR&J.='4399')         02420699
             AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                        02420799
          THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                        02420899
        END;                                                             02420999
       %END;                                                             02421099
     %END;                                                               02421999
                                                                         02422099
     %IF %SUBSTR(&PXFLAG,3)=5 %THEN %DO;                                 02422199
                          /*LEBY*/                                       02422299
       %DO J=1 %TO 6;                                                    02422399
        IF Y&YR.SDT&J >0  THEN  DO;                                      02422499
        IF Y&YR.SPR&J.='3929'                                            02422599
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                      02422699
           THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                       02422799
         END;                                                            02422899
        %END;                                                            02422999
     %END;                                                               02423299
                                                                         02423399
     %IF %SUBSTR(&PXFLAG,3)=6 %THEN %DO;                                 02423499
                          /*PANC*/                                       02423599
      %DO J =1 %TO 6;                                                   02423699
       IF Y&YR.SDT&J >0  THEN  DO;                                       02423799
        IF Y&YR.SPR&J. IN('5251','5253','527')                           02423899
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                      02423999
           THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                       02424099
        END;                                                             02424199
```

```
   %END;                                                                02424299
%END;                                                                   02424599
                                                                        02424699
%IF %SUBSTR(&PXFLAG,3)=7 %THEN %DO;                                     02424899
                        /*AVR*/                                         02424999
  %DO J =1 %TO 6;                                                       02425199
   IF Y&YR.SDT&J >0  THEN  DO;                                          02425299
   IF Y&YR.SPR&J IN('3521','3522')                                      02425399
```

D#MSQ.@MSQ01.PGMLIB(MSQYH402) -- CONTINUED

```
            AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                        02425499
        THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                         02425599
    END;                                                                02425699
   %END;                                                                02426199
%END;                                                                   02426299
                                                                        02426399
%IF %SUBSTR(&PXFLAG,3)=8  %THEN %DO;                                    02426499
                        /*MVR*/                                         02426599
  %DO J =1 %TO 6;                                                       02426699
   IF Y&YR.SDT&J >0  THEN  DO;                                          02426799
   IF Y&YR.SPR&J IN('3523','3524')                                      02426899
           AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                         02426999
        THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                         02427099
    END;                                                                02427199
   %END;                                                                02427299
%END;                                                                   02427399
                                                                        02427499
%IF %SUBSTR(&PXFLAG,3)=9 %THEN %DO;                                     02427599
                        /*PNEU*/                                        02427699
  %DO J = 1 %TO 6;                                                      02427799
   IF Y&YR.SDT&J >0  THEN  DO;                                          02427899
   IF Y&YR.SPR&J ='325'                                                 02427999
           AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                         02428099
        THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                         02428199
    END;                                                                02428299
   %END;                                                                02428399
%END;                                                                   02428499
                                                                        02428599
%IF %SUBSTR(&PXFLAG,3)=10 %THEN %DO;                                    02428699
                        /*LOBE*/                                        02428799
   %DO J = 1 %TO 6;                                                     02428899
   IF Y&YR.SDT&J >0  THEN  DO;                                          02428999
   IF Y&YR.SPR&J ='324'                                                 02429099
           AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                         02429199
        THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                         02429299
    END;                                                                02429399
   %END;                                                                02429499
%END;                                                                   02429799
                                                                        02429899
%IF %SUBSTR(&PXFLAG,3)=11 %THEN %DO;                                    02429999
                       /*COLE*/                                         02430099
  %DO J =1 %TO 6;                                                       02430199
   IF Y&YR.SDT&J >0  THEN  DO;                                          02430299
   IF ('4573'<=Y&YR.SPR&J.<='4576')                                     02430399
           AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                         02430499
        THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                         02430599
    END;                                                                02430699
```

```
       %END;                                                          02430799
     %END;                                                            02431099
                                                                      02431199
     %IF %SUBSTR(&PXFLAG,3)=12 %THEN %DO;                             02431299
                          /*CYST*/                                    02431399
       %DO J = 1 %TO 6;                                               02431499
        IF Y&YR.SDT&J >0  THEN  DO;                                   02431599
        IF ('5770'<=Y&YR.SPR&J.<='5779')                              02431699
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                   02431799
            THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                   02431899
        END;                                                          02431999
```

D#MSQ.@MSQ01.PGMLIB(MSQYH402) -- CONTINUED

```
       %END;                                                          02432099
     %END;                                                            02440099
                                                                      02450099
     %IF %SUBSTR(&PXFLAG,3)=13 %THEN %DO;                             02561099
                          /*GAST*/                                    02562099
       %DO J = 1 %TO 6;                                               02562199
        IF Y&YR.SDT&J >0  THEN  DO;                                   02562299
        IF ('435'<=Y&YR.SPR&J.<='4399')                               02563099
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                   02563199
            THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                   02563299
         END;                                                         02563399
        %END;                                                         02563499
     %END;                                                            02569099
                                                                      02570099
     %IF %SUBSTR(&PXFLAG,3)=14 %THEN %DO;                             02641099
                          /*NEPH*/                                    02642099
       %DO J = 1 %TO 6;                                               02642199
        IF Y&YR.SDT&J >0  THEN  DO;                                   02642299
        IF Y&YR.SPR&J. IN('5551')                                     02643099
               AND DATEJUL(Y&YR.SDT&J.) < DT&PXFLAG                   02643199
            THEN DT&PXFLAG  = DATEJUL(Y&YR.SDT&J.);                   02643299
         END;                                                         02643399
        %END;                                                         02643499
    %END;                                                             02649099
                                                                      03370099
                                                                      03470099
 FORMAT DT&PXFLAG  MMDDYY10. ;                                        03471099
                                                                      03480099
 RUN;                                                                 03490062
                                                                      03500191
 PROC DATASETS NOLIST;                                                03500799
 DELETE DS&PXFLAG ;                                                   03500899
 RUN;                                                                 03500998
                                                                      03501098
  /*PRINT 10 RECORDS FOR CHECKING*/                                   03501199
 PROC PRINT DATA=OUT.&PXFLAG.Y&YR.(OBS=10);                           03501399
 RUN;                                                                 03501499
                                                                      03501599
 %MEND ADMDCH;                                                        03501662
 %ADMDCH(PXFLAG=OC1) ;                                                03501792
 %ADMDCH(PXFLAG=OC2) ;                                                03501892
 %ADMDCH(PXFLAG=OC3) ;                                                03501992
 %ADMDCH(PXFLAG=OC4) ;                                                03502092
 %ADMDCH(PXFLAG=OC5) ;                                                03502192
```

```
%ADMDCH(PXFLAG=OC6) ;                                                    03502292
%ADMDCH(PXFLAG=OC7) ;                                                    03502392
%ADMDCH(PXFLAG=OC8) ;                                                    03502492
%ADMDCH(PXFLAG=OC9) ;                                                    03502592
%ADMDCH(PXFLAG=OC10);                                                    03503092
%ADMDCH(PXFLAG=OC11);                                                    03504092
%ADMDCH(PXFLAG=OC12);                                                    03505092
%ADMDCH(PXFLAG=OC13);                                                    03506092
%ADMDCH(PXFLAG=OC14);                                                    03507092
                                                                         03508091
```

D#MSQ.@MSQ01.JCLTEST(MSQ#2H44)

```
//G1V12H44 JOB (MSQ01,MQMS),'STP4_2A_92',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ2H44D
//
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H44)

```
//G1V13H44 JOB (MSQ01,MQMS),'STP4_2A93',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ3H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H44D)

```
//MSQ3H44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="93"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF1,                          00090086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00091086
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF2,                          00092086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00093086
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF3,                          00094086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00095086
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF4,                          00096086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00097086
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF5,                          00098086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00099086
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF6,                          00099186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100086
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF7,                          00110086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120086
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF8,                          00130086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140086
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF9,                          00141086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00142086
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF10,                         00143086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00144086
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF11,                         00145086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00146086
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF12,                         00147086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00148086
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF13,                         00149086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00149186
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.INREF14,                         00150086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF93ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1001.DENOMA93, *1993 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1001.DENOMA94, *1994 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y93,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y93,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H44)

```
//G1V14H44 JOB (MSQ01,MQMS),'STP4_2A94',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ4H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H44D)

```
//MSQ4H44D PROC                                                          00010079
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="94"'                                        00050066
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF1,                   00090090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100090
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF2,                   00110090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120090
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF3,                   00130090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140090
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF4,                   00150090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160090
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF5,                   00170090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180090
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF6,                   00181090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182090
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF7,                   00183090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00184090
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF8,                   00185090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00186090
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF9,                   00187090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00188090
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF10,                  00189090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189190
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF11,                  00189290
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189390
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF12,                  00189490
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189590
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF13,                  00189690
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189790
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY94.INREF14,                  00189890
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189990
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF94ALL,     *NEW FINDER FILE*  00190086
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1001.DENOMA94, *1994 DENOM*     00290038
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1001.DENOMA95, *1995 DENOM*     00310073
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y94,                          00330088
//            DISP=(SHR,KEEP,KEEP)                                       00340086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y94,                          00350086
//            DISP=(NEW,CATLG,DELETE),                                   00360087
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H44)

```
//G1V15H44 JOB (MSQ01,MQMS),'STP4_2A95',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ5H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H44D)

```
//MSQ5H44D PROC                                                          00010081
//*                                                                       00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="95"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF1,                   00081088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00082087
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF2,                   00083088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00084087
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF3,                   00085088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00086087
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF4,                   00087088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00088087
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF5,                   00089088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089187
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF6,                   00089288
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089387
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF7,                   00089488
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089587
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF8,                   00089688
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089787
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF9,                   00090088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100087
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF10,                  00110088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120087
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF11,                  00130088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140087
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF12,                  00150088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160087
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF13,                  00170088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180087
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY95.INREF14,                  00181088
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182087
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF95ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1001.DENOMA95, *1995 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1001.DENOMA96, *1996 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y95,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y95,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                       00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H44)

```
//G1V16H44 JOB (MSQ01,MQMS),'STP4_2A96',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ6H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H44D)

```
//MSQ6H44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="96"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF1,                   00090086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100086
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF2,                   00110086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120086
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF3,                   00130086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140086
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF4,                   00150086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160086
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF5,                   00170086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180086
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF6,                   00180186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180286
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF7,                   00180386
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180486
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF8,                   00180586
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180686
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF9,                   00180986
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00181086
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF10,                  00181186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00181286
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF11,                  00181386
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00181486
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF12,                  00181586
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00181686
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF13,                  00181786
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00181886
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY96.INREF14,                  00181986
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF96ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1001.DENOMA96, *1996 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1001.DENOMA97, *1997 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y96,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y96,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H44)

```
//G1V17H44 JOB (MSQ01,MQMS),'STP4_2A97',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ7H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H44D)

```
//MSQ7H44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="97"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF1,                   00081087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00082086
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF2,                   00083087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00084086
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF3,                   00085087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00086086
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF4,                   00087087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00088086
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF5,                   00089087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089186
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF6,                   00090087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100086
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF7,                   00110087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120086
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF8,                   00121087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00122086
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF9,                   00123087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00124086
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF10,                  00125087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00126086
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF11,                  00127087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00128086
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF12,                  00129087
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00129186
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF13,                  00129287
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00129386
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY97.INREF14,                  00129487
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00129586
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF97ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1001.DENOMA97, *1997 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1001.DENOMA98, *1998 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y97,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y97,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H44)

```
//G1V18H44 JOB (MSQ01,MQMS),'STP4_2A98',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ8H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H44D)

```
//MSQ8H44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="98"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF1,                   00090086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00091086
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF2,                   00092086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00093086
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF3,                   00094086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00095086
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF4,                   00096086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00097086
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF5,                   00098086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00099086
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF6,                   00099186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00099286
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF7,                   00099386
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00099486
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF8,                   00099586
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00099686
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF9,                   00099886
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100086
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF10,                  00110086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120086
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF11,                  00130086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140086
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF12,                  00150086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160086
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF13,                  00170086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180086
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY98.INREF14,                  00181086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF98ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1001.DENOMA98, *1998 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1001.DENOMA99, *1999 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y98,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y98,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H44)

```
//G1V19H44 JOB (MSQ01,MQMS),'STP4_2A99',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ9H44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H44D)

```
//MSQ4H44D PROC                                                          00010079
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="99"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF1,                   00080186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00080286
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF2,                   00080386
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00080486
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF3,                   00080586
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00080686
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF4,                   00080786
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00080886
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF5,                   00080986
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00081086
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF6,                   00081186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00081286
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF7,                   00081386
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00081486
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF8,                   00081586
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00081686
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF9,                   00081886
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00081986
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF10,                  00082086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00083086
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF11,                  00084086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00085086
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF12,                  00086086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00087086
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF13,                  00088086
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00089086
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY99.INREF14,                  00089186
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00090086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF99ALL,     *NEW FINDER FILE*  00190083
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1001.DENOMA99, *1999 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1001.DENOMA00, *2000 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y99,                          00330084
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y99,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360080
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH44)

```
//G1V1AH44 JOB (MSQ01,MQMS),'STP4_2A00',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQAH44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH44D)

```
//MSQAH44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="00"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF1,                   00090090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100090
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF2,                   00110090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120090
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF3,                   00130090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140090
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF4,                   00150090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160090
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF5,                   00170090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180090
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF6,                   00181090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182090
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF7,                   00183090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00184090
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF8,                   00185090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00186090
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF9,                   00187090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00188090
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF10,                  00189090
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189190
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF11,                  00189290
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189390
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF12,                  00189490
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189590
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF13,                  00189690
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189790
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY00.INREF14,                  00189890
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189990
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF00ALL,    *2000 FINDER*       00190087
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1001.DENOMA00, *2000 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300038
//M1DEN01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1001.DENOMA01, *2001 DENOM*     00310081
//            DISP=(SHR,KEEP,KEEP)                                       00320072
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y00,                          00330088
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y00,                          00350086
//            DISP=(NEW,CATLG,DELETE),                                   00360086
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH404),                         00390069
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH44)

```
//G1V1BH44 JOB (MSQ01,MQMS),'STP4_2A01',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQBH44D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH44D)

```
//MSQBH44D PROC                                                          00010081
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="01"'                                        00050081
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080090
//TMP1     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF1,                   00090091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00100091
//TMP2     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF2,                   00110091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00120091
//TMP3     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF3,                   00130091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00140091
//TMP4     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF4,                   00150091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00160091
//TMP5     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF5,                   00170091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00180091
//TMP6     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF6,                   00181091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00182091
//TMP7     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF7,                   00183091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00184091
//TMP8     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF8,                   00185091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00186091
//TMP9     DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF9,                   00187091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00188091
//TMP10    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF10,                  00189091
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189191
//TMP11    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF11,                  00189291
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189391
//TMP12    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF12,                  00189491
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189591
//TMP13    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF13,                  00189691
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189791
//TMP14    DD DSN=D#MSQ.@MSQ01.MSQYH404.TMPY01.INREF14,                  00189891
//            DATACLAS=DC14DAY,SPACE=(CYL,(200,100),RLSE)                00189991
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF01ALL,     *NEW FINDER FILE*  00190088
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//M1DEN01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1001.DENOMA01, *2001 DENOM*     00290081
//            DISP=(SHR,KEEP,KEEP)                                       00300086
//OC1      DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y01,                          00330089
//            DISP=(SHR,KEEP,KEEP)                                       00340045
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y01,                          00350081
//            DISP=(NEW,CATLG,DELETE),                                   00360088
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQBH404),                         00390083
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQYH404)

```
  /*************************************************************/   00010091
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH404)               */   00020098
  /*-----------------------------------------------------------*/   00030091
  /* PURPOSE :     USING FINDER FILE TO GET ALL MEMBERS' HIC NO.*/   00040099
  /*               GET THE DATE OF ENTERY OF HMO PROGRAM       */   00041099
  /*-----------------------------------------------------------*/   00050091
  /* INPUT FILE :  D#MSQ.@MSQ01.MSQYH304.OC3Yyy                */   00060099
  /*               D#MSQ.@MSQ01.MSQYH402.XRFALL(XRFyyALL)      */   00061099
  /*               D#MSQ.@MSQ0000.SAS.YyyM1001.DENOMyy         */   00064099
  /*-----------------------------------------------------------*/   00080091
  /* OUTPUT FILE : D#MSQ.@MSQ01.MSQYH404.OC4Yyy                */   00090099
  /*               DATASET: OCnYyy                             */   00100099
  /*                                WHERE yy=92-01, n=1-14     */   00110099
  /*-----------------------------------------------------------*/   00150091
  /* WRITTEN BY  : LUCY LU, JUNE 17, 2003                      */   00160099
  /* EDITED BY: ERIC SCHONE AND MATTHEW GILLINGHAM 7/10/2003   */   00161099
  /* CONTACT PERSON:  ANGELA SCHMITT                           */   00170091
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   00180091
  /*               PHONE 202-484-4517                          */   00190091
  /*************************************************************/   00220091
                                                                    00230091
OPTIONS NOCENTER MPRINT;                                            00240099
                                                                    00320062
  /*GET CURRENT YEAR AND NEXT YEAR MACRO VARIABLES*/                00321099
                                                                    00330099
%LET YR  = &SYSPARM; /*CREAE FIRST YEAR MACRO VARIABLE*/           00330199
                                                                    00330299
  /*CREATE MACRO VARIABLE YR2 FOR 2ND YEAR*/                        00330499
  /* YEAR 2001 ONLY HAS ONE YEAR*/                                 00330599
                                                                    00330799
DATA _NULL_;                                                        00330899
                                                                    00330999
YEAR=&YR.+1900;                                                     00331099
IF YEAR <1992 THEN YEAR+100;                                        00331199
                                                                    00331299
IF YEAR <2001 THEN YRP=YEAR+1;                                      00331399
ELSE YRP=YEAR;                                                      00331499
Y=SUBSTR(PUT(YRP,4.),3);                                            00331599
CALL SYMPUT('YR2',TRIM(Y));                                         00331699
                                                                    00331799
RUN;                                                                00331899
                                                                    00331999
                                                                    00332099
 PROC SORT DATA=XRF.XRF&YR.ALL OUT=TEMP1.XRF&YR.ALL;                00332199
 BY MASTRHIC;                                                       00332299
 RUN;                                                               00332399
                                                                    00332499
  /*****PROCESS EACH SINGLE OUTCOME FILE****/                       00332599
%MACRO GETHMO(YR2);                                                 00332699
%DO J= 1 %TO 14;                                                    00332799
                                                                    00332899
  /*  SINGLE OUTCOME FILE CREATED IN STEP4.1 MEAGES                 00332999
     WITH DENOMINATOR FILES--NOTE: OC FILE CONTAINS MASTER HIC NO.*/ 00333099
 PROC SORT DATA=OC1.OC&J.Y&YR.(KEEP=MASTRHIC Y&YR.SADM YEAR)        00333199
           OUT=ALL1YR&J;                                            00334099
 BY MASTRHIC;                                                       00334599
```

```
 RUN;                                                                    00335299
                                                                         00335399
   /*MERGE W/ NEW FINDER FILE TO GET ALL HICNO*/                         00335499
D#MSQ.@MSQ01.PGMLIB(MSQYH404) -- CONTINUED

 DATA ALL&J;                                                             00335599
   MERGE ALL1YR&J(IN=A)                                                  00335699
         TEMP1.XRF&YR.ALL(IN=B)                                          00335899
          ;                                                              00335999
   BY MASTRHIC;                                                          00336099
                                                                         00336199
 IF A AND B;                                                             00336299
                                                                         00336399
 PROC SORT DATA=ALL&J OUT=TMP&J..INREF&J;                                00336499
 BY Y&YR.HICNO;                                                          00336599
 RUN;                                                                    00336699
 %END; /*EDIT ES,MG*/                                                   00336799
                                                                         00336899
 /*MERGE THE DATASET CONTAINS 1ST ADM WITH CURRENT YR & NEXT YR DENOM    00336999
   FILES TO GET ALL MEMBERS AND 1ST HMO DATE */                          00337099
 DATA ALL;                                                               00337199
 SET TMP1.INREF1(IN=IN1)                                                 00337299
     TMP2.INREF2(IN=IN2)                                                 00337399
     TMP3.INREF3(IN=IN3)                                                 00337499
     TMP4.INREF4(IN=IN4)                                                 00337599
     TMP5.INREF5(IN=IN5)                                                 00337699
     TMP6.INREF6(IN=IN6)                                                 00337799
     TMP7.INREF7(IN=IN7)                                                 00337899
     TMP8.INREF8(IN=IN8)                                                 00338099
     TMP9.INREF9(IN=IN9)                                                 00339099
     TMP10.INREF10(IN=IN10)                                              00340099
     TMP11.INREF11(IN=IN11)                                              00340199
     TMP12.INREF12(IN=IN12)                                              00340299
     TMP13.INREF13(IN=IN13)                                              00340399
     TMP14.INREF14(IN=IN14);                                             00340499
 BY Y&YR.HICNO;                                                          00340599
 PX=1;                                                                   00340699
 IF IN2 THEN PX=2;                                                       00340799
 IF IN3 THEN PX=3;                                                       00340899
 IF IN4 THEN PX=4;                                                       00340999
 IF IN5 THEN PX=5;                                                       00341099
 IF IN6 THEN PX=6;                                                       00341199
 IF IN7 THEN PX=7;                                                       00341299
 IF IN8 THEN PX=8;                                                       00341399
 IF IN9 THEN PX=9;                                                       00341499
 IF IN10 THEN PX=10;                                                     00341599
 IF IN11 THEN PX=11;                                                     00341699
 IF IN12 THEN PX=12;                                                     00341799
 IF IN13 THEN PX=13;                                                     00341899
 IF IN14 THEN PX=14;                                                     00341999
                                                                         00342199
 DATA HMO;                                                               00342299
     MERGE M1DEN&YR..DENOMA&YR(IN=B KEEP=                                00342399
           Y&YR.HICNO Y&YR.HMO1 Y&YR.HMO2 Y&YR.HMO3 Y&YR.HMO4            00342499
           Y&YR.HMO5 Y&YR.HMO6 Y&YR.HMO7 Y&YR.HMO8 Y&YR.HMO9             00342599
           Y&YR.HMO10 Y&YR.HMO11 Y&YR.HMO12 Y&YR.DETH Y&YR.VDOD          00342699
           RENAME=(Y&YR.HICNO=HICNO))                                    00342799
                                                                         00342899
```

```
          M1DEN&YR2..DENOMA&YR2.(IN=C KEEP=                                  00342999
          Y&YR2.HICNO Y&YR2.HMO1 Y&YR2.HMO2 Y&YR2.HMO3 Y&YR2.HMO4            00343099
          Y&YR2.HMO5 Y&YR2.HMO6 Y&YR2.HMO7 Y&YR2.HMO8 Y&YR2.HMO9             00343199
          Y&YR2.HMO10 Y&YR2.HMO11 Y&YR2.HMO12 Y&YR2.DETH Y&YR2.VDOD          00343299
          RENAME=(Y&YR2.HICNO=HICNO))                                        00343399
                                                                             00343499
```

D#MSQ.@MSQ01.PGMLIB(MSQYH404) -- CONTINUED

```
          ALL(IN=A RENAME=(Y&YR.HICNO=HICNO));                               00343599
     BY HICNO;                                                               00343699
                                                                             00343799
 IF A;                                                                       00343899
                                                                             00343999
                                                                             00344099
      HMOdt=mdy(1,1,2099);                                                   00344199
     %DO i=1 %to 12;                                                         00344299
      if NOT(Y&YR.hmo&i in ('0','')) & hmodt=mdy(1,1,2099)                   00344399
       & &I >= MONTH(Y&YR.SADM) THEN                                         00344499
     HMODT=MDY(&I,1,YEAR);                                                   00344599
     %END;                                                                   00344699
                                                                             00345799
     %DO i=1 %to 12;                                                         00345899
      if NOT(Y&YR2.hmo&i in ('0','')) & hmodt=mdy(1,1,2099)                  00345999
      THEN                                                                   00346099
     HMODT=MDY(&I,1,YEAR+1);                                                 00346199
     %END;                                                                   00346299
                                                                             00346399
IF Y&YR2.VDOD ='V' THEN DO;                                                  00346499
VDOD=Y&YR2.VDOD; DEATH=Y&YR2.DETH;                                           00346599
END;                                                                         00346699
ELSE IF Y&YR.VDOD='V' THEN DO;                                               00346799
VDOD=Y&YR.VDOD; DEATH=Y&YR.DETH;                                             00346899
END;                                                                         00346999
ELSE IF C THEN DEATH=Y&YR2.DETH;                                             00347099
ELSE DEATH=Y&YR.DETH;                                                        00347199
                                                                             00347399
FORMAT HMODT Y&YR.SADM MMDDYY10.;                                            00347499
                                                                             00347599
%DO J=1 %TO 14;                                                              00347699
                                                                             00347799
DATA HMO&J(DROP=PX);                                                         00347899
SET HMO;                                                                     00347999
WHERE PX=&J;                                                                 00348099
                                                                             00348199
PROC PRINT DATA=HMO&J.(OBS=25);                                              00348299
WHERE HMODT NE MDY(1,1,2099) OR DEATH > .;                                   00348399
                                                                             00348499
PROC SORT DATA=HMO&J(KEEP=MASTRHIC HICNO HMODT) OUT=TEMP1.HMO&J;             00348599
BY MASTRHIC HMODT;                                                           00348699
                                                                             00348799
PROC SORT DATA=HMO&J(KEEP=MASTRHIC VDOD DEATH);                              00348899
BY MASTRHIC VDOD DEATH;                                                      00348999
                                                                             00349099
DATA OC&J.Y&YR;                                                              00349199
  SET TEMP1.HMO&J;                                                           00349299
BY MASTRHIC;                                                                 00349399
 IF FIRST.MASTRHIC;           /*FIRST HMO DATE*/                            00349499
RUN;                                                                         00349599
```

```
                                                                      00349699
 DATA HMO&J;                                                          00349799
 SET HMO&J;                                                           00349899
 BY MASTRHIC;                                                         00349999
 IF LAST.MASTRHIC;                                                    00350099
                                                                      00350199
 DATA OUT.OC&J.Y&YR;                                                  00350299
 MERGE OC&J.Y&YR HMO&J;                                               00350399
 BY MASTRHIC;                                                         00350499
D#MSQ.@MSQ01.PGMLIB(MSQYH404) -- CONTINUED

                                                                      00350599
                                                                      00350699
 PROC DATASETS NOLIST;                                                00353099
 DELETE ALL1YR&J ALL&J HMO&J;                                         00354099
 RUN;                                                                 00355099
                                                                      00355899
                                                                      03508091
%END;                                                                 03509099
                                                                      03510099
%MEND;                                                                03511099
%GETHMO(YR2=&YR2);                                                    03520099
```

D#MSQ.@MSQ01.PGMLIB(MSQBH404)

```
  /*************************************************************/   03509099
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQBH404)               */   03510099
  /*-----------------------------------------------------------*/   03520099
  /* PURPOSE :     USING FINDER FILE TO GET ALL MEMBERS' HIC NO.*/   03530099
  /*               GET THE DATE OF ENTERY OF HMO PROGRAM       */   03540099
  /*               NOTE: THIS PROGRAM RUNS FOR 2001 ONLY!!!    */   03541099
  /*-----------------------------------------------------------*/   03550099
  /* INPUT FILE :  D#MSQ.@MSQ01.MSQYH304.OC3Y01                */   03560099
  /*               D#MSQ.@MSQ01.MSQYH402.XRFALL(XRF01ALL)      */   03570099
  /*               D#MSQ.@MSQ0000.SAS.Y01M1001.DENOM01         */   03580099
  /*-----------------------------------------------------------*/   03590099
  /* OUTPUT FILE : D#MSQ.@MSQ01.MSQYH404.OC4Y01                */   03600099
  /*               DATASET: OCnY01                             */   03610099
  /*-----------------------------------------------------------*/   03630099
  /* WRITTEN BY  : LUCY LU, JUNE 17, 2003                      */   03640099
  /* EDITED BY ERIC SCHONE, MATT GILLINGHAM 7/14/2003          */   03641099
  /* CONTACT PERSON:  ANGELA SCHMITT                           */   03650099
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   03660099
  /*               PHONE 202-484-4517                          */   03670099
  /*************************************************************/   03680099
                                                                    03690099
OPTIONS NOCENTER MPRINT;                                            03700099
                                                                    03710099
  /*GET CURRENT YEAR AND NEXT YEAR MACRO VARIABLES*/                03720099
                                                                    03730099
%LET YR  = &SYSPARM; /*CREATE FIRST YEAR MACRO VARIABLE*/           03740099
                                                                    03750099
  /*CREATE MACRO VARIABLE YR2 FOR 2ND YEAR*/                        03760099
  /* YEAR 2001 ONLY HAS ONE YEAR*/                                  03770099
                                                                    03780099
DATA _NULL_;                                                        03790099
                                                                    03800099
YEAR=&YR.+1900;                                                     03810099
IF YEAR <1992 THEN YEAR+100;                                        03820099
                                                                    03830099
IF YEAR <2001 THEN YRP=YEAR+1;                                      03840099
ELSE YRP=YEAR;                                                      03850099
Y=SUBSTR(PUT(YRP,4.),3);                                            03860099
CALL SYMPUT('YR2',TRIM(Y));                                         03870099
                                                                    03880099
RUN;                                                                03890099
                                                                    03900099
                                                                    03910099
 PROC SORT DATA=XRF.XRF&YR.ALL OUT=TEMP1.XRF&YR.ALL;                03920099
 BY MASTRHIC;                                                       03930099
 RUN;                                                               03940099
                                                                    03950099
  /*****PROCESS EACH SINGLE OUTCOME FILE****/                       03960099
%MACRO GETHMO;                                                      03970099
%DO J= 1 %TO 14;                                                    03980099
                                                                    03990099
  /*  SINGLE OUTCOME FILE CREATED IN STEP4.1 MEAGES                 04000099
     WITH DENOMINATOR FILES--NOTE: OC FILE CONTAINS MASTER HIC NO.*/ 04010099
 PROC SORT DATA=OC1.OC&J.Y&YR.(KEEP=MASTRHIC Y&YR.SADM YEAR)        04020099
           OUT=ALL1YR&J;                                            04030099
 BY MASTRHIC;                                                       04040099
```

```
 RUN;                                                                    04090099
                                                                         04100099
   /*MERGE W/ NEW FINDER FILE TO GET ALL HICNO*/                         04110099
D#MSQ.@MSQ01.PGMLIB(MSQBH404) -- CONTINUED

 DATA ALL&J;                                                             04120099
   MERGE ALL1YR&J(IN=A)                                                  04130099
         TEMP1.XRF&YR.ALL(IN=B)                                          04150099
          ;                                                              04160099
   BY MASTRHIC;                                                          04170099
                                                                         04180099
 IF A AND B;                                                             04190099
                                                                         04200099
 PROC SORT DATA=ALL&J OUT=TMP&J..INREF&J;                                04210099
 BY Y&YR.HICNO;                                                          04220099
 RUN;                                                                    04230099
 %END;                                                                   04240099
                                                                         04241099
 /*MERGE THE DATASET CONTAINING 1ST ADM WITH CURRENT YR & NEXT YR DENOM 04250099
   FILES TO GET ALL MEMBERS AND 1ST HMO DATE */                          04260099
 DATA ALL;                                                               04270099
 SET TMP1.INREF1(IN=IN1)                                                 04271099
     TMP2.INREF2(IN=IN2)                                                 04272099
     TMP3.INREF3(IN=IN3)                                                 04273099
     TMP4.INREF4(IN=IN4)                                                 04274099
     TMP5.INREF5(IN=IN5)                                                 04275099
     TMP6.INREF6(IN=IN6)                                                 04276099
     TMP7.INREF7(IN=IN7)                                                 04277099
     TMP8.INREF8(IN=IN8)                                                 04278099
     TMP9.INREF9(IN=IN9)                                                 04279099
     TMP10.INREF10(IN=IN10)                                              04279199
     TMP11.INREF11(IN=IN11)                                              04279299
     TMP12.INREF12(IN=IN12)                                              04279399
     TMP13.INREF13(IN=IN13)                                              04279499
     TMP14.INREF14(IN=IN14);                                             04279599
 BY Y&YR.HICNO;                                                          04279699
 PX=1;                                                                   04279799
 IF IN2 THEN PX=2;                                                       04279899
 IF IN3 THEN PX=3;                                                       04279999
 IF IN4 THEN PX=4;                                                       04280099
 IF IN5 THEN PX=5;                                                       04280199
 IF IN6 THEN PX=6;                                                       04280299
 IF IN7 THEN PX=7;                                                       04280399
 IF IN8 THEN PX=8;                                                       04280499
 IF IN9 THEN PX=9;                                                       04280599
 IF IN10 THEN PX=10;                                                     04280699
 IF IN11 THEN PX=11;                                                     04280799
 IF IN12 THEN PX=12;                                                     04280899
 IF IN13 THEN PX=13;                                                     04280999
 IF IN14 THEN PX=14;                                                     04281099
                                                                         04281199
 DATA HMO;                                                               04282099
     MERGE M1DEN&YR..DENOMA&YR(IN=B KEEP=                                04290099
           Y&YR.HICNO Y&YR.HMO1 Y&YR.HMO2 Y&YR.HMO3 Y&YR.HMO4            04300099
           Y&YR.HMO5 Y&YR.HMO6 Y&YR.HMO7 Y&YR.HMO8 Y&YR.HMO9             04310099
           Y&YR.HMO10 Y&YR.HMO11 Y&YR.HMO12 Y&YR.DETH Y&YR.VDOD          04320099
           RENAME=(Y&YR.HICNO=HICNO))                                    04330099
                                                                         04340099
```

```
          ALL(IN=A RENAME=(Y&YR.HICNO=HICNO));                          04400099
      BY HICNO;                                                         04410099
                                                                        04420099
  IF A;                                                                 04430099
                                                                        04450099
       HMOdt=mdy(1,1,2099);                                             04460099
```

D#MSQ.@MSQ01.PGMLIB(MSQBH404) -- CONTINUED

```
      %DO i=1 %to 12;                                                   04470099
      IF NOT(Y&YR.HMO&I IN ('0','')) & HMODT=MDY(1,1,2099)              04471099
      & &i >= MONTH(Y&YR.SADM) THEN                                     04472099
      HMODT=MDY(&I,1,YEAR);                                             04473099
      %END;                                                             04474099
                                                                        04474199
  IF Y&YR.VDOD='V' THEN DO;                                             04474299
  VDOD=Y&YR.VDOD;                                                       04474399
  END;                                                                  04474499
  DEATH=Y&YR.DETH;                                                      04474599
                                                                        04580099
 FORMAT HMODT Y&YR.SADM  MMDDYY10.;                                     04590099
                                                                        04600099
 %DO J=1 %TO 14;                                                        04601099
 DATA HMO&J(DROP=PX);                                                   04602099
 SET HMO;                                                               04603099
 WHERE PX=&J;                                                           04604099
                                                                        04605099
 PROC PRINT DATA=HMO&J.(OBS=10);                                        04610099
 WHERE HMODT NE MDY(1,1,2099) OR DEATH > .;                             04620099
                                                                        04630099
 PROC SORT DATA=HMO&J(KEEP=MASTRHIC HICNO HMODT) OUT=TEMP1.HMO&J;       04640099
 BY MASTRHIC HMODT;                                                     04650099
                                                                        04651099
 PROC SORT DATA=HMO&J(KEEP=MASTRHIC VDOD DEATH);                        04652099
 BY MASTRHIC VDOD DEATH;                                                04653099
                                                                        04660099
 DATA OC&J.Y&YR;                                                        04670099
   SET TEMP1.HMO&J;                                                     04680099
 BY MASTRHIC;                                                           04690099
  IF FIRST.MASTRHIC;           /*FIRST HMO DATE*/                       04700099
 RUN;                                                                   04710099
                                                                        04711099
 DATA HMO&J;                                                            04712099
 SET HMO&J;                                                             04713099
 BY MASTRHIC;                                                           04714099
 IF LAST.MASTRHIC;                                                      04715099
                                                                        04720099
 DATA OUT.OC&J.Y&YR;                                                    04721099
 MERGE OC&J.Y&YR HMO&J;                                                 04722099
 BY MASTRHIC;                                                           04723099
                                                                        04730099
 PROC DATASETS NOLIST;                                                  04740099
 DELETE ALL1YR&J  ALL&J HMO&J;                                          04750099
 RUN;                                                                   04760099
                                                                        04770099
                                                                        04780099
%END;                                                                   04790099
                                                                        04800099
%MEND;                                                                  04810099
```

%GETHMO; 04820099

D#MSQ.@MSQ01.JCLTEST(MSQ#2H46)

```
//G1V12H46 JOB (MSQ01,MQMS),'STP4_2B_92',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ2H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H46D)

```
//MSQ2H46D PROC                                                          00010086
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="92"'                                        00050080
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF92ALL,     *NEW FINDER FILE* 00160089
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG92  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M4001.M4FLAGS,  *1992 FLAGS*    00180086
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4FLAGS,  *1993 FLAGS*    00182086
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH92  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M4001.M4OTHER,  *1992 OTHER*    00184086
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4OTHER,  *1993 OTHER*    00186086
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y92,                          00188087
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y92,                          00350086
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H46)

```
//L1983H46 JOB (MSQ01,MQMS),'STP4_2B_93',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ3H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ3H46D)

```
//MSQ3H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="93"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF93ALL,    *NEW FINDER FILE*  00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4FLAGS,  *1993 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4FLAGS,  *1994 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M4001.M4OTHER,  *1993 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4OTHER,  *1994 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y93,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y93,                          00350092
//            DISP=(NEW,CATLG,DELETE),                                   00360091
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H46)

```
//G1V14H46 JOB (MSQ01,MQMS),'STP4_2B94',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ4H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H46D)

```
//MSQ4H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="94"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF94ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4FLAGS,  *1994 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4FLAGS,  *1995 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M4001.M4OTHER,  *1994 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4OTHER,  *1995 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y94,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y94,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                       00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H46)

```
//L1985H46 JOB (MSQ01,MQMS),'STP4_2B_95',CLASS=J,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ5H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H46D)

```
//MSQ5H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="95"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF95ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4FLAGS,  *1995 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4FLAGS,  *1996 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M4001.M4OTHER,  *1995 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4OTHER,  *1996 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y95,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y95,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H46)

```
//G1V16H46 JOB (MSQ01,MQMS),'STP4_2B_96',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ6H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H46D)

```
//MSQ6H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="96"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF96ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4FLAGS,  *1996 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4FLAGS,  *1997 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M4001.M4OTHER,  *1996 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4OTHER,  *1997 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y96,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y96,                          00350092
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H46)

```
//L1987H46 JOB (MSQ01,MQMS),'STP4_2B_97',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ7H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H46D)

```
//MSQ7H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="97"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF97ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4FLAGS,  *1997 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4FLAGS,  *1998 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M4001.M4OTHER,  *1997 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4OTHER,  *1998 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y97,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y97,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H46)

```
//G1V18H46 JOB (MSQ01,MQMS),'STP4_2B_98',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ8H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H46D)

```
//MSQ8H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="98"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF98ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4FLAGS,  *1998 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4FLAGS,  *1999 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M4001.M4OTHER,  *1998 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4OTHER,  *1999 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y98,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y98,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H46)

```
//L1989H46 JOB (MSQ01,MQMS),'STP4_2B_99',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQ9H46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H46D)

```
//MSQ9H46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="99"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF99ALL,     *NEW FINDER FILE* 00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4FLAGS,  *1999 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4FLAGS,  *2000 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M4001.M4OTHER,  *1999 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4OTHER,  *2000 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y99,                          00188088
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y99,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360085
//            UNIT=HCFA80,                                               00370046
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH46)

```
//G1V1AH46 JOB (MSQ01,MQMS),'STP4_2B_00',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQAH46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH46D)

```
//MSQAH46D PROC                                                          00010088
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="00"'                                        00050088
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF00ALL,    *NEW FINDER FILE*  00160090
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4FLAGS,  *2000 FLAGS*    00180088
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4FLG01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4FLAGS,  *2001 FLAGS*    00182088
//            DISP=(SHR,KEEP,KEEP)                                       00183086
//M4OTH00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M4001.M4OTHER,  *2000 OTHER*    00184088
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//M4OTH01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4OTHER,  *2001 OTHER*    00186088
//            DISP=(SHR,KEEP,KEEP)                                       00187086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y00,                          00188093
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y00,                          00350095
//            DISP=(NEW,CATLG,DELETE),                                   00360095
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH406),                         00390086
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH46)

```
//L198BH46 JOB (MSQ01,MQMS),'STP4_2B_01',CLASS=K,MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1127358,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB  DD DSN=D#MSQ.@MSQ01.PROCLIB,
//            DISP=(SHR,KEEP,KEEP)
//JS020   EXEC MSQBH46D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH46D)

```
//MSQBH46D PROC                                                          00010089
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="01"'                                        00050089
//SAS.WORK DD SPACE=(CYL,(900,900),,,ROUND)                              00070079
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//TEMP2    DD DSN=&&TEMP2,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00150086
//XRF      DD DSN=D#MSQ.@MSQ01.MSQYH402.XRF01ALL,     *NEW FINDER FILE* 00160096
//            DISP=(SHR,KEEP,KEEP)                                       00170086
//M4FLG01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4FLAGS,  *2001 FLAGS*    00180089
//            DISP=(SHR,KEEP,KEEP)                                       00181086
//M4OTH01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M4001.M4OTHER,  *2001 OTHER*    00184089
//            DISP=(SHR,KEEP,KEEP)                                       00185086
//IN       DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y01,                          00188089
//            DISP=(SHR,KEEP,KEEP)                                       00189086
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y01,                          00350089
//            DISP=(NEW,CATLG,DELETE),                                   00360093
//            SPACE=(CYL,(1000,1000),RLSE)                               00380067
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQBH406),                         00390091
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D@MSQ.MSQ01.PGMLIB(MSQYH406)

```
  /*************************************************************/   00010091
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH406)               */   00020099
  /*-----------------------------------------------------------*/   00030091
  /* PURPOSE :     MERGE DATASET FROM STEP 4.2A WITH CLAIM FILES   */   00040099
  /*               CREATE READMT AND INDEX DEATH VIARIABLES        */   00041099
  /*-----------------------------------------------------------*/   00050091
  /* INPUT FILE :  D#MSQ.@MSQ01.MSQYH404.OC4Yyy                */   00060099
  /*               D#MSQ.@MSQ0000.SAS.YyyM4001.M4FLAGS             */   00062099
  /*               D#MSQ.@MSQ0000.SAS.YyyM4001.M4OTHER             */   00063099
  /*-----------------------------------------------------------*/   00080091
  /* OUTPUT FILE : D#MSQ.@MSQ01.MSQYH406.OC5Yyy                */   00090099
  /*                                WHERE yy=92-01               */   00110099
  /*-----------------------------------------------------------*/   00150091
  /* LUCY LU, APR. 7, 2003                                     */   00160099
  /* CONTACT PERSON:  ANGELA SCHMITT                           */   00170091
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   00180091
  /*               PHONE 202-484-4517                          */   00190091
  /*************************************************************/   00220091
                                                                    00230091
OPTIONS NOCENTER MPRINT /*OBS=100*/;                                00240099
                                                                    00320062
  /*----CREATE MACRO VARIABLES FOR CURRENT YEAR AND NEXT YEAR----*/    00321099
                                                                    00330099
%LET YR  = &SYSPARM; /*FIRST YEAR MACRO VARIABLE*/                  00330199
                                                                    00330299
  /*NEXT YEAR MACRO VARIABLE*/                                      00330499
  /* YEAR 2001 HAS NO NEXT YEAR*/                                   00330599
                                                                    00330799
DATA _NULL_;                                                        00330899
                                                                    00330999
YEAR=&YR.+1900;                                                     00331099
IF YEAR <1992 THEN YEAR+100;                                        00331199
                                                                    00331299
IF YEAR <2001 THEN YRP=YEAR+1;                                      00331399
ELSE YRP=YEAR;                                                      00331499
Y=SUBSTR(PUT(YRP,4.),3);                                            00331599
CALL SYMPUT('YR2',TRIM(Y));                                         00331699
                                                                    00331799
RUN;                                                                00331899
                                                                    00331999
PROC SORT DATA=XRF.XRF&YR.ALL OUT=TEMP1.XRF&YR.ALL;                 00332099
BY MASTRHIC; RUN;                                                   00332299
                                                                    00332399
%MACRO COMB(YR2);                                                   00332499
 DATA ALLYR;                                                        00332599
  SET IN.OC1Y&YR(KEEP=MASTRHIC)                                     00332699
      IN.OC2Y&YR(KEEP=MASTRHIC)                                     00332799
      IN.OC3Y&YR(KEEP=MASTRHIC)                                     00332899
      IN.OC4Y&YR(KEEP=MASTRHIC)                                     00332999
      IN.OC5Y&YR(KEEP=MASTRHIC)                                     00333099
      IN.OC6Y&YR(KEEP=MASTRHIC)                                     00333199
      IN.OC7Y&YR(KEEP=MASTRHIC)                                     00334099
      IN.OC9Y&YR(KEEP=MASTRHIC)                                     00335099
      IN.OC10Y&YR(KEEP=MASTRHIC)                                    00336099
      IN.OC11Y&YR(KEEP=MASTRHIC)                                    00337099
      IN.OC12Y&YR(KEEP=MASTRHIC)                                    00342899
```

```
      IN.OC13Y&YR(KEEP=MASTRHIC)                                          00343399
      IN.OC14Y&YR(KEEP=MASTRHIC)                                          00343499
      ;                                                                   00343599
D@MSQ.MSQ01.PGMLIB(MSQYH406) -- CONTINUED

                                                                          00343699
 PROC SORT DATA=ALLYR NODUPKEY;                                           00343999
 BY MASTRHIC;                                                             00344099
                                                                          00344199
DATA ALL;                                                                 00344299
  MERGE ALLYR(IN=A)                                                       00344399
        TEMP1.XRF&YR.ALL                                                  00344499
        ;                                                                 00344599
  BY MASTRHIC;                                                            00344699
IF A;                                                                     00344799
RUN;                                                                      00344899
                                                                          00345499
  /*CREATE LOOK UP FILE, PICK THE HICNO FROM OUTCOME FILES*/              00345599
PROC SORT DATA=ALL(RENAME=(Y&YR.HICNO=HICNO))                             00345699
OUT=LOOKUP NODUPKEY;                                                      00345799
BY HICNO;                                                                 00345899
RUN;                                                                      00345999
                                                                          00346099
DATA A;                                                                   00346199
  SET LOOKUP;                                                             00346299
                                                                          00346399
FMTNAME="WANTED";                                                         00346499
TYPE="C";                                                                 00346599
START=HICNO;                                                              00346699
LABEL="%%";                                                               00346799
                                                                          00346899
RUN;                                                                      00346999
                                                                          00347099
PROC FORMAT CNTLIN=A ;                                                    00347199
RUN;                                                                      00347299
                                                                          00347399
                                                                          00347499
/*COMBINE THE DATASETS CONTAIN ALL POSSIBLE HICNOS W/ M4 FILES*/          00347599
                                                                          00347699
DATA PICK;                                                                00347799
   SET M4FLG&YR..M4FLAGS                                                  00347899
             (KEEP=Y&YR.HICNO Y&YR.SADM Y&YR.DISCH Y&YR.PRVID             00347999
              Y&YR.DIA1 Y&YR.DIA2 Y&YR.DIA3 Y&YR.DIA4 Y&YR.DIA5           00348099
              Y&YR.DIA6 Y&YR.DIA7 Y&YR.DIA8 Y&YR.DIA9 Y&YR.DIA10          00348199
              Y&YR.SPR1 Y&YR.SPR2 Y&YR.SPR3 Y&YR.SPR4 Y&YR.SPR5           00348299
              Y&YR.SPR6 Y&YR.SDT1 Y&YR.SDT2 Y&YR.SDT3 Y&YR.SDT4           00348399
              Y&YR.SDT5 Y&YR.SDT6 Y&YR.NOBS                               00348499
              rename=(Y&YR.HICNO=HICNO))                                  00348599
       M4OTH&YR..M4OTHER                                                  00348999
             (KEEP=Y&YR.HICNO Y&YR.SADM Y&YR.DISCH Y&YR.PRVID             00349099
              Y&YR.DIA1 Y&YR.DIA2 Y&YR.DIA3 Y&YR.DIA4 Y&YR.DIA5           00349199
              Y&YR.DIA6 Y&YR.DIA7 Y&YR.DIA8 Y&YR.DIA9 Y&YR.DIA10          00349299
              Y&YR.SPR1 Y&YR.SPR2 Y&YR.SPR3 Y&YR.SPR4 Y&YR.SPR5           00349399
              Y&YR.SPR6 Y&YR.SDT1 Y&YR.SDT2 Y&YR.SDT3 Y&YR.SDT4           00349499
              Y&YR.SDT5 Y&YR.SDT6 Y&YR.NOBS                               00349599
              rename=(Y&YR.HICNO=HICNO))                                  00349699
      M4FLG&YR2..M4FLAGS                                                  00350399
             (KEEP=Y&YR2.HICNO Y&YR2.SADM Y&YR2.DISCH Y&YR2.PRVID         00350499
```

```
               Y&YR2.DIA1 Y&YR2.DIA2 Y&YR2.DIA3 Y&YR2.DIA4 Y&YR2.DIA5   00350599
               Y&YR2.DIA6 Y&YR2.DIA7 Y&YR2.DIA8 Y&YR2.DIA9 Y&YR2.DIA10  00350699
               Y&YR2.SPR1 Y&YR2.SPR2 Y&YR2.SPR3 Y&YR2.SPR4 Y&YR2.SPR5   00350799
               Y&YR2.SPR6 Y&YR2.SDT1 Y&YR2.SDT2 Y&YR2.SDT3 Y&YR2.SDT4   00350899
               Y&YR2.SDT5 Y&YR2.SDT6 Y&YR2.NOBS RENAME=(                00350999
               Y&YR2.HICNO=HICNO))                                      00351099
D@MSQ.MSQ01.PGMLIB(MSQYH406) -- CONTINUED

     M4OTH&YR2..M4OTHER                                                 00351799
              (KEEP=Y&YR2.HICNO Y&YR2.SADM Y&YR2.DISCH Y&YR2.PRVID      00351899
               Y&YR2.DIA1 Y&YR2.DIA2 Y&YR2.DIA3 Y&YR2.DIA4 Y&YR2.DIA5   00351999
               Y&YR2.DIA6 Y&YR2.DIA7 Y&YR2.DIA8 Y&YR2.DIA9 Y&YR2.DIA10  00352099
               Y&YR2.SPR1 Y&YR2.SPR2 Y&YR2.SPR3 Y&YR2.SPR4 Y&YR2.SPR5   00352199
               Y&YR2.SPR6 Y&YR2.SDT1 Y&YR2.SDT2 Y&YR2.SDT3 Y&YR2.SDT4   00352299
               Y&YR2.SDT5 Y&YR2.SDT6 Y&YR2.NOBS RENAME=(                00352399
               Y&YR2.HICNO=HICNO))                                      00352499
               ;                                                        00352599
  BY HICNO;                                                             00352799
                                                                        00352999
    IF PUT(HICNO,$WANTED.)="%%" THEN OUTPUT;                            00353099
                                                                        00353199
 DATA TEMP2.PICK;                                                       00353299
   SET PICK;                                                            00353399
                                                                        00353499
  RADMDT=Y&YR.SADM ;                                                    00353599
  IF RADMDT=. THEN RADMDT=Y&YR2.SADM;                                   00353699
  PRVID=Y&YR.PRVID;                                                     00353799
  IF PRVID='' THEN PRVID=Y&YR2.PRVID;                                   00353899
  RUN;                                                                  00353999
                                                                        00354099
                                                                        00356699
  DATA TEMP1.PICK;                                                      00356799
    MERGE TEMP2.PICK LOOKUP;                                            00356899
    BY HICNO;                                                           00356999
                                                                        00357099
  FORMAT Y&YR.SADM Y&YR2.SADM RADMDT MMDDYY10.;                         00357199
                                                                        00357299
  PROC SORT DATA=TEMP1.PICK OUT=OUT.OC5Y&YR; BY MASTRHIC RADMDT;        00357399
                                                                        00357499
 PROC PRINT DATA=OUT.OC5Y&YR(OBS=20);                                   00357599
 RUN;                                                                   00357698
                                                                        00357799
%MEND;                                                                  00357899
                                                                        00358099
%COMB(YR2=&YR2);                                                        00360099
                                                                        03508091
```

D#MSQ.@MSQ01.PGMLIB(MSQYH406)

```
  /*************************************************************/   00010091
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH406)               */   00020099
  /*-----------------------------------------------------------*/   00030091
  /* PURPOSE :     MERGE DATASET FROM STEP 4.2A WITH CLAIM FILES   */   00040099
  /*               CREATE READMT AND INDEX DEATH VIARIABLES        */   00041099
  /*-----------------------------------------------------------*/   00050091
  /* INPUT FILE :  D#MSQ.@MSQ01.MSQYH404.OC4Yyy                */   00060099
  /*               D#MSQ.@MSQ0000.SAS.YyyM4001.M4FLAGS         */   00062099
  /*               D#MSQ.@MSQ0000.SAS.YyyM4001.M4OTHER         */   00063099
  /*-----------------------------------------------------------*/   00080091
  /* OUTPUT FILE : D#MSQ.@MSQ01.MSQYH406.OC5Yyy                */   00090099
  /*                                WHERE yy=92-01             */   00110099
  /*-----------------------------------------------------------*/   00150091
  /* LUCY LU, APR. 7, 2003                                     */   00160099
  /* CONTACT PERSON:  ANGELA SCHMITT                           */   00170091
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   00180091
  /*               PHONE 202-484-4517                          */   00190091
  /*************************************************************/   00220091
                                                                    00230091
OPTIONS NOCENTER MPRINT /*OBS=100*/;                                00240099
                                                                    00320062
  /*----CREATE MACRO VARIABLES FOR CURRENT YEAR AND NEXT YEAR----*/    00321099
                                                                    00330099
%LET YR  = &SYSPARM; /*FIRST YEAR MACRO VARIABLE*/                  00330199
                                                                    00330299
  /*NEXT YEAR MACRO VARIABLE*/                                      00330499
  /* YEAR 2001 HAS NO NEXT YEAR*/                                   00330599
                                                                    00330799
DATA _NULL_;                                                        00330899
                                                                    00330999
YEAR=&YR.+1900;                                                     00331099
IF YEAR <1992 THEN YEAR+100;                                        00331199
                                                                    00331299
IF YEAR <2001 THEN YRP=YEAR+1;                                      00331399
ELSE YRP=YEAR;                                                      00331499
Y=SUBSTR(PUT(YRP,4.),3);                                            00331599
CALL SYMPUT('YR2',TRIM(Y));                                         00331699
                                                                    00331799
RUN;                                                                00331899
                                                                    00331999
PROC SORT DATA=XRF.XRF&YR.ALL OUT=TEMP1.XRF&YR.ALL;                 00332099
BY MASTRHIC; RUN;                                                   00332299
                                                                    00332399
%MACRO COMB;                                                        00332499
 DATA ALLYR;                                                        00332599
  SET IN.OC1Y&YR(KEEP=MASTRHIC)                                     00332699
      IN.OC2Y&YR(KEEP=MASTRHIC)                                     00332799
      IN.OC3Y&YR(KEEP=MASTRHIC)                                     00332899
      IN.OC4Y&YR(KEEP=MASTRHIC)                                     00332999
      IN.OC5Y&YR(KEEP=MASTRHIC)                                     00333099
      IN.OC6Y&YR(KEEP=MASTRHIC)                                     00333199
      IN.OC7Y&YR(KEEP=MASTRHIC)                                     00334099
      IN.OC9Y&YR(KEEP=MASTRHIC)                                     00335099
      IN.OC10Y&YR(KEEP=MASTRHIC)                                    00336099
      IN.OC11Y&YR(KEEP=MASTRHIC)                                    00337099
      IN.OC12Y&YR(KEEP=MASTRHIC)                                    00342899
```

```
      IN.OC13Y&YR(KEEP=MASTRHIC)                                        00343399
      IN.OC14Y&YR(KEEP=MASTRHIC)                                        00343499
      ;                                                                 00343599
D#MSQ.@MSQ01.PGMLIB(MSQYH406) -- CONTINUED
                                                                        00343699
 PROC SORT DATA=ALLYR NODUPKEY;                                         00343999
 BY MASTRHIC;                                                           00344099
                                                                        00344199
DATA ALL;                                                               00344299
  MERGE ALLYR(IN=A)                                                     00344399
        TEMP1.XRF&YR.ALL                                                00344499
        ;                                                               00344599
  BY MASTRHIC;                                                          00344699
IF A;                                                                   00344799
RUN;                                                                    00344899
                                                                        00345499
  /*CREATE LOOK UP FILE, PICK THE HICNO FROM OUTCOME FILES*/            00345599
PROC SORT DATA=ALL(RENAME=(Y&YR.HICNO=HICNO))                           00345699
OUT=LOOKUP NODUPKEY;                                                    00345799
BY HICNO;                                                               00345899
RUN;                                                                    00345999
DATA A;                                                                 00346099
  SET LOOKUP;                                                           00346199
                                                                        00346299
FMTNAME="WANTED";                                                       00346399
TYPE="C";                                                               00346499
START=HICNO;                                                            00346599
LABEL="%%";                                                             00346699
                                                                        00346799
RUN;                                                                    00346899
                                                                        00346999
PROC FORMAT CNTLIN=A ;                                                  00347099
RUN;                                                                    00347199
                                                                        00347299
                                                                        00347399
/*COMBINE THE DATASETS CONTAIN ALL POSSIBLE HICNOS W/ M4 FILES*/        00347499
                                                                        00347599
DATA PICK;                                                              00347699
   SET M4FLG&YR..M4FLAGS                                                00347799
            (KEEP=Y&YR.HICNO Y&YR.SADM Y&YR.DISCH Y&YR.PRVID            00347899
             Y&YR.DIA1 Y&YR.DIA2 Y&YR.DIA3 Y&YR.DIA4 Y&YR.DIA5          00347999
             Y&YR.DIA6 Y&YR.DIA7 Y&YR.DIA8 Y&YR.DIA9 Y&YR.DIA10         00348099
             Y&YR.SPR1 Y&YR.SPR2 Y&YR.SPR3 Y&YR.SPR4 Y&YR.SPR5          00348199
             Y&YR.SPR6 Y&YR.SDT1 Y&YR.SDT2 Y&YR.SDT3 Y&YR.SDT4          00348299
             Y&YR.SDT5 Y&YR.SDT6 Y&YR.NOBS                              00348399
             rename=(Y&YR.HICNO=HICNO))                                 00348499
       M4OTH&YR..M4OTHER                                                00348999
            (KEEP=Y&YR.HICNO Y&YR.SADM Y&YR.DISCH Y&YR.PRVID            00349099
             Y&YR.DIA1 Y&YR.DIA2 Y&YR.DIA3 Y&YR.DIA4 Y&YR.DIA5          00349199
             Y&YR.DIA6 Y&YR.DIA7 Y&YR.DIA8 Y&YR.DIA9 Y&YR.DIA10         00349299
             Y&YR.SPR1 Y&YR.SPR2 Y&YR.SPR3 Y&YR.SPR4 Y&YR.SPR5          00349399
             Y&YR.SPR6 Y&YR.SDT1 Y&YR.SDT2 Y&YR.SDT3 Y&YR.SDT4          00349499
             Y&YR.SDT5 Y&YR.SDT6 Y&YR.NOBS                              00349599
             rename=(Y&YR.HICNO=HICNO))                                 00349699
             ;                                                          00352599
  BY HICNO;                                                             00352799
                                                                        00352999
    IF PUT(HICNO,$WANTED.)="%%" THEN OUTPUT;                            00353099
```

```
                                                                        00353199
 DATA PICK;                                                             00353299
   SET PICK;                                                            00353399
                                                                        00353499
  RADMDT=Y&YR.SADM ;                                                    00353599
  PRVID=Y&YR.PRVID;                                                     00353799
D#MSQ.@MSQ01.PGMLIB(MSQYH406) -- CONTINUED

 RUN;                                                                   00353999
                                                                        00354099
                                                                        00356699
  DATA TEMP1.PICK;                                                      00356799
    MERGE PICK LOOKUP;                                                  00356899
    BY HICNO;                                                           00356999
                                                                        00357099
  FORMAT Y&YR.SADM  RADMDT MMDDYY10.;                                   00357199
                                                                        00357299
  PROC SORT DATA=TEMP1.PICK OUT=OUT.OC5Y&YR; BY MASTRHIC RADMDT;        00357399
                                                                        00357499
 PROC PRINT DATA=OUT.OC5Y&YR(OBS=20);                                   00357599
 RUN;                                                                   00357698
                                                                        00357799
%MEND;                                                                  00357899
                                                                        00358099
%COMB;                                                                  00360099
                                                                        03508091
```

D#MSQ.@MSQ01.JCLTEST(MSQ#2H48)

```
//G1V12H48 JOB (MSQ01,MQMS),'STP4_3_92',
//             CLASS=K,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1992 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ2H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ2H48D)

```
//MSQ2H48D PROC                                                          00010080
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="92"'                                        00050080
//SAS.WORK DD SPACE=(CYL,(500,500),,,ROUND)                              00070077
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y92,                          00190080
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y92,                          00201084
//            DISP=(SHR,KEEP,KEEP)                                       00202084
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y92,                          00210081
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y92,                          00350080
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            UNIT=HCFA80,                                               00370046
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#3H48)

```
//G1V13H48 JOB (MSQ01,MQMS),'STP4_3_93',
//             CLASS=K,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1993 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ3H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ3H48D)

```
//MSQ3H48D PROC                                                          00010082
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="93"'                                        00050082
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070081
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y93,                          00190082
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y93,                          00201084
//            DISP=(SHR,KEEP,KEEP)                                       00202084
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y93,                          00210082
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y93,                          00350086
//            DISP=(NEW,CATLG,DELETE),                                   00360086
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#4H48)

```
//G1V14H48 JOB (MSQ01,MQMS),'STP4_3_94',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1994 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ4H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ4H48D)

```
//MSQ4H48D PROC                                                          00010079
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="94"'                                        00050066
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070081
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y94,                          00190079
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y94,                          00201084
//            DISP=(SHR,KEEP,KEEP)                                       00202084
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y94,                          00210080
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y94,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360084
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#5H48)

```
//G1V15H48 JOB (MSQ01,MQMS),'STP4_3_95',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1995 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ5H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ5H48D)

```
//MSQ5H48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="95"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070085
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y95,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y95,                          00201086
//            DISP=(SHR,KEEP,KEEP)                                       00202086
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y95,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y95,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(1000,800),RLSE)                                00380087
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#6H48)

```
//G1V16H48 JOB (MSQ01,MQMS),'STP4_3_96',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1996 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ6H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ6H48D)

```
//MSQ6H48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="96"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070085
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y96,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y96,                          00201087
//            DISP=(SHR,KEEP,KEEP)                                       00202087
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y96,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y96,                          00350087
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(1000,800),RLSE)                                00380088
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#7H48)

```
//G1V17H48 JOB (MSQ01,MQMS),'STP4_3_97',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1997 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ7H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ7H48D)

```
//MSQ7H48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="97"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1600,900),,,ROUND)                             00070088
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y97,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y97,                          00201086
//            DISP=(SHR,KEEP,KEEP)                                       00202086
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y97,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y97,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(1000,800),RLSE)                                00380087
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#8H48)

```
//G1V18H48 JOB (MSQ01,MQMS),'STP4_3_98',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1998 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ8H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ8H48D)

```
//MSQ8H48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="98"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070085
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y98,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y98,                          00201086
//            DISP=(SHR,KEEP,KEEP)                                       00202086
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y98,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y98,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#9H48)

```
//G1V19H48 JOB (MSQ01,MQMS),'STP4_3_99',
//             CLASS=H,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 1999 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQ9H48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQ9H48D)

```
//MSQ9H48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="99"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070085
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y99,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y99,                          00201086
//            DISP=(SHR,KEEP,KEEP)                                       00202086
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y99,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y99,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#AH48)

```
//G1V1AH48 JOB (MSQ01,MQMS),'STP4_3_00',
//             CLASS=K,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 2000 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQAH48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQAH48D)

```
//MSQAH48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="00"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1400,900),,,ROUND)                             00070090
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y00,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y00,                          00201088
//            DISP=(SHR,KEEP,KEEP)                                       00202088
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y00,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y00,                          00350088
//            DISP=(NEW,CATLG,DELETE),                                   00360089
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390079
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.JCLTEST(MSQ#BH48)

```
//G1V1BH48 JOB (MSQ01,MQMS),'STP4_3_01',
//             CLASS=K,
//             MSGCLASS=Q
//LUPRT OUTPUT WRITER=P1827102,
//             DEFAULT=Y,
//             JESDS=ALL
//*
//PROCLIB   DD DSN=D#MSQ.@MSQ01.PROCLIB,
//             DISP=(SHR,KEEP,KEEP)
//*****************************************************************
//* JS010 EXECUTES A PROC TO EXECUTE A SAS PROGRAM THAT PERFORMS   *
//*       THE 2001 STEP 4 TIME TO EVENT CALCULATIONS               *
//*****************************************************************
//JS010   EXEC MSQBH48D
//
```

D#MSQ.@MSQ01.PROCLIB(MSQBH48D)

```
//MSQBH48D PROC                                                          00010084
//*                                                                      00020038
//PS010  EXEC SAS8,OPTIONS='NONEWS',                                     00030041
//            PARM='SYSPARM="01"'                                        00050084
//SAS.WORK DD SPACE=(CYL,(1000,1000),,,ROUND)                            00070086
//TEMP1    DD DSN=&&TEMP1,UNIT=MCART,LABEL=(1,SL,,,RETPD=5)              00080077
//INOC3    DD DSN=D#MSQ.@MSQ01.MSQYH402.OC3Y01,                          00190084
//            DISP=(SHR,KEEP,KEEP)                                       00200038
//INOC4    DD DSN=D#MSQ.@MSQ01.MSQYH404.OC4Y01,                          00201088
//            DISP=(SHR,KEEP,KEEP)                                       00202088
//INOC5    DD DSN=D#MSQ.@MSQ01.MSQYH406.OC5Y01,                          00210084
//            DISP=(SHR,KEEP,KEEP)                                       00220038
//OUT      DD DSN=D#MSQ.@MSQ01.MSQYH408.OC6Y01,                          00350084
//            DISP=(NEW,CATLG,DELETE),                                   00360083
//            SPACE=(CYL,(800,500),RLSE)                                 00380078
//SYSIN    DD DSN=D#MSQ.@MSQ01.PGMLIB(MSQYH408),                         00390087
//            DISP=(SHR,PASS,KEEP)                                       00400038
//*                                                                      00410038
```

D#MSQ.@MSQ01.PGMLIB(MSQYH408)

```
  /*************************************************************/   00001000
  /* PROGRAM :     D#MSQ.@MSQ01.PGMLIB(MSQYH408)               */   00002000
  /*-----------------------------------------------------------*/   00003000
  /* PURPOSE :     MERGE FILES CREATED FROM MSQYH402 AND MSQYH406. */ 00004000
  /*               COMBINE CURRENT YEAR AND NEXT YEAR FILES, AND   */
  /*               CALCULATE TIME TO EVENT INDICATES.          */   00005001
  /*-----------------------------------------------------------*/   00006000
  /* INPUT FILE :  D#MSQ.@MSQ01.MSQYH402.OC3Yyy(OCnYyy)        */   00007002
  /*               D#MSQ.@MSQ01.MSQYH404.OC4Yyy(OCnYyy)        */   00008002
  /*               D#MSQ.@MSQ01.MSQYH406.OC5Yyy                */   00008002
  /*-----------------------------------------------------------*/   00009300
  /* OUTPUT FILE : D#MSQ.@MSQ01.MSQYH408.OC6Yyy                */   00009402
  /*                                WHERE yy=92-01             */   00009600
  /*-----------------------------------------------------------*/   00009700
  /* ERIC SCHONE, LUCY LU, JULY 7, 2003                        */   00009802
  /* EDITED BY ERIC SCHONE, MATT GILLINGHAM ON 7/14/2003       */
  /* EDITED BY ERIC SCHONE, MATT GILLINGHAM ON 7/28/2003,7/29/2003,*/
  /* CONTACT PERSON:  ANGELA SCHMITT          8/04/2003        */   00009900
  /*               MATHEMATICA POLICY RESEARCH, INC.           */   00010000
  /*               PHONE 202-484-4517                          */   00010100
  /*************************************************************/   00010200
                                                                    00010300
OPTIONS NOCENTER MPRINT /*OBS=100*/;                                00010400
                                                                    00010500
  /*----CREATE TWO MACRO VARIABLES FOR CURRENT YEAR AND NEXT YEAR----*/
                                                                    00010700
%LET YR  = &SYSPARM;        /*FIRST YEAR MACRO VARIABLE*/           00010800
                                                                    00010900
  /*CREATE MACRO VARIABLE YR2 FOR NEXT YEAR*/                       00011000
  /* YEAR 2001 ONLY HAS ONE YEAR*/                                  00011100
                                                                    00011300
DATA _NULL_;                                                        00011400
                                                                    00011500
YEAR=&YR.+1900;                                                     00011600
IF YEAR <1992 THEN YEAR+100;                                        00011700
                                                                    00011800
IF YEAR <2001 THEN YRP=YEAR+1;                                      00011900
ELSE YRP=YEAR;                                                      00012000
Y=SUBSTR(PUT(YRP,4.),3);                                            00012100
CALL SYMPUT('YR2',TRIM(Y));                                         00012200
                                                                    00012300
RUN;                                                                00012400
                                                                    00012500
                                                                    00012800
     /*----------- MAIN PROGRAM  -----------*/                       00012903
                                                                    00013002
 %MACRO EVENTS(YR2);                                                00120003
                                                                    00121002
 %DO I = 1 %TO 14;                                                  00122002
                                                                    00123002
 PROC SORT DATA=INOC3.OC&I.Y&YR(DROP=Y&YR.HICNO)                    00124003
 OUT=C3OC&I; BY MASTRHIC;                                           00124003
 PROC SORT DATA=INOC4.OC&I.Y&YR(DROP=HICNO)
 OUT=C4OC&I; BY MASTRHIC;

 DATA C3C4OC&I;
```

```
   MERGE C3OC&I
         C4OC&I;
   BY MASTRHIC;
```
D#MSQ.@MSQ01.PGMLIB(MSQYH408) -- CONTINUED

```
 DATA READM;                                                            00126003
   MERGE C3C4OC&I (IN=A RENAME=(                                        00127003
                  DTOC&I.=Y&YR.DTOC&I
                  PRVID=Y&YR.IND_PRVID
                  HMODT=Y&YR.HMODT))                                    00128003

         INOC5.OC5Y&YR(KEEP=MASTRHIC PRVID RADMDT                       00128003
                     IN=B) /*7/14/2003*/
                     ;                                                  01560000

     BY MASTRHIC;

   IF A AND B;
   RETAIN Y&YR.IND_RADM;

   RADM_DT=RADMDT;
   Y&YR.IND_PRC_DT=Y&YR.DTOC&I;
   Y&YR.IND_ADM_DT=Y&YR.SADM;
   Y&YR.IND_DCH_DT=Y&YR.SDIS;
   Y&YR.IND_DETH=DEATH; /*7/14/2003*/

    /*RECODE THE PROCEDURE DATE IF THE PROC DATE LATER THAN DCH DATE
      OR EARLIER THAN ADM DATE*/
   IF NOT (Y&YR.IND_ADM_DT <= Y&YR.IND_PRC_DT <=Y&YR.IND_DCH_DT)
   THEN Y&YR.IND_PRC_DT=Y&YR.IND_ADM_DT;
     FORMAT Y&YR.IND_ADM_DT Y&YR.IND_PRC_DT Y&YR.IND_DCH_DT
            Y&YR.IND_DETH MMDDYY10.;
     DROP Y&YR.DTOC&I RADMDT
          ;
    /* CREATE READMISSION INDEX VARIABLE*/

      IF FIRST.MASTRHIC THEN
         Y&YR.IND_RADM=MDY(12,31,YEAR+1);

      IF Y&YR.IND_RADM=MDY(12,31,YEAR+1) AND
           ((RADM_DT>=Y&YR.IND_DCH_DT & PRVID=Y&YR.IND_PRVID)
           OR (RADM_DT>Y&YR.IND_DCH_DT & PRVID NE Y&YR.IND_PRVID))
     THEN Y&YR.IND_RADM=RADM_DT;

       IF LAST.MASTRHIC THEN OUTPUT;
     RUN;

     DATA OC&I;
        SET READM;

      /*TIME FROM INDEX DCH TO RE-ADMISSION---TRD*/
         Y&YR.TRD=Y&YR.IND_RADM-Y&YR.IND_DCH_DT;

      /*TIME FROM INDEX PX DATE TO DEATH---TMP*/
         IF Y&YR.IND_DETH = . THEN
            Y&YR.IND_DETH = MDY(12,31,YEAR+1);
            Y&YR.TMP=Y&YR.IND_DETH-Y&YR.IND_PRC_DT;
```

```
        /*TIME FROM INDEX ADM DATE TO DEATH---TMA*/
             Y&YR.TMA=Y&YR.IND_DETH-Y&YR.IND_ADM_DT;

        /*CALCULATE CENSORING TIME*/
        /*SET UP THE END OF PERIOD AS DEFAULT DATE*/
```
D#MSQ.@MSQ01.PGMLIB(MSQYH408) -- CONTINUED

```
         /*CENSORING DATE---DEFINED AS THE EARLEST DATE OF
          DEATH OR DATE OF HMO ENROLLMENT*/
           CRDDATE= MIN(Y&YR.IND_DETH,Y&YR.HMODT,MDY(1,1,YEAR+1));

      /*FLAG INPATIENT DEATH*/
           Y&YR.IPDTH=0;
           IF Y&YR.IND_DETH NE MDY(12,31,YEAR+1) THEN DO;
              IF Y&YR.IND_DETH<=Y&YR.IND_DCH_DT THEN Y&YR.IPDTH=1;
           END;

      /*TIME FORM INDEX DCH TO INDEX RE-ADM OR DEATH---TRM*/
         Y&YR.TRM=min(Y&YR.IND_radm,Y&YR.ind_deth)-Y&YR.IND_DCH_DT;

     /*-----CALCULATE CENSORING TIME----- */
       /*FROM INDEX DCH TO CENSORING DATE--CRD*/
        Y&YR.CRD=CRDDATE-Y&YR.IND_DCH_DT;

      /*HMO ENROLLMENT DATE---CMPDATE*/
       CMPDATE= MIN(Y&YR.HMODT,MDY(1,1,YEAR+1)); /*7/29/03 - MG*/

    /*FROM INDEX PX DATE TO THE END OF PERIOD--CMP*/
     Y&YR.CMP=MDY(1,1,YEAR+1)-Y&YR.IND_PRC_DT; /*7/29/03 - MG*/

    /*FROM INDEX ADM DATE TO THE END OF PERIOD---CMA*/
     Y&YR.CMA=MDY(1,1,YEAR+1)-Y&YR.IND_ADM_DT; /*7/29/03 - MG*/

    /*TIME FORM THE DATE OF HMO ENROLLMENT TO DCH---CMD*/
     Y&YR.CMD=CMPDATE-Y&YR.IND_DCH_DT;

 LABEL
 Y&YR.TRD  ="TIME FROM INDEX DCH TO READMISSION"                          01100803
 Y&YR.TMP  ="TIME FROM INDEX PX DATE TO DEATH"                            01102403
 Y&YR.TMA  ="TIME FROM INDEX ADM DATE TO DEATH"                           01102103
 CRDDATE   ="CENSORING DATE-EARLEST DATE OF DEATH OR HMO ENROLLMENT"
 Y&YR.TRM  ="TIME FORM INDEX DCH TO INDEX RE-ADM OR DEATH"
 Y&YR.CRD  ="TIME FROM INDEX DCH TO CENSORING DATE"
 Y&YR.CMP  ="TIME FROM INDEX PX DATE TO THE END OF PERIOD"
 Y&YR.CMA  ="TIEM FROM INDEX ADM DATE TO THE END OF PERIOD"
 Y&YR.CMD  ="TIME FORM DISCHARGE TO HMO ENROLLMENT"
 CMPDATE   ="DATE OF ENTRY INTO MEDICARE+CHOICE(HMO)"
 Y&YR.IPDTH="INPATIENT DEATH FLAG"
 Y&YR.IND_RADM  ="INDEX READMISSION DATE"
 ;

 PROC DATASETS NOLIST;
 DELETE C3OC&I C4OC&I C3C4OC&I READM;

 RUN;                                                                     01570000
 %END;                                                                    01770000
```

```
                                                                        01780000
 %MEND EVENTS;                                                          01790000
 %EVENTS(YR2=&YR2);                                                     01930000
                                                                        01940000
 /* COMBINE THE CALCULATIONS BY PROCEDURE INTO ONE DATASET */          01950000
 DATA OUT.OC6Y&YR;                                                      01960000
     SET OC1 OC2 OC3 OC4 OC5                                            01970000
         OC6 OC7 OC8 OC9 OC10                                           01980000
         OC11 OC12 OC13 OC14                                            01990000
D#MSQ.@MSQ01.PGMLIB(MSQYH408) -- CONTINUED

         ;                                                              02000000
                                                                        02060000
 DROP I J ;

 PROC CONTENTS DATA=OUT.OC6Y&YR POSITION;
 RUN;
 PROC MEANS DATA=OUT.OC6Y&YR N NMISS MIN MAX MEAN SUM;
 RUN;

 TITLE "FINALIZED STEP 4.3, YEAR &YR.";                                02070000
 PROC PRINT DATA=OUT.OC6Y&YR.(OBS=20);                                  02080000
 WHERE Y&YR.IND_DETH >.Z AND Y&YR.HMODT NE MDY(1,1,2099);               02090000
 RUN;                                                                   02100000
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG1)

```
//P1H4T601 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(1300,300))
//IN92  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC92PX,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC93PX,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC94PX,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC95PX,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC96PX,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC97PX,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC98PX,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC99PX,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC00PX,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC01PX,DISP=SHR
//Y94    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC94SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y95    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC95SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y96    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC96SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC97SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y98    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC98SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y99    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC99SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y00    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC00SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC01SUM,
//       DISP=(OLD,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//SYSIN  DD *

*******************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ00M4.PGMLIBAG(HRSAG1)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ01.MSQYH102.OC99PX
* OUTPUTS: P1H4.@MSQ0000.SAS.HRSAG1.OC99PX
* PURPOSE: PRODUCE aggregate FILES THAT CONTAIN COUNTS OF DISCHARGES
*          WITH HIGH RISK SURGERIES AND SUM UP FOR 3 YEARS TO
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*******************************************************************;

%MACRO AGG(YY);

%LET VARS=Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG;
```

```
%LET OCS = OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG1) -- CONTINUED

```
 ************************************************
 * AGGREGATE ACROSS ALL STRATIFIERS FOR EACH YEAR
 ************************************************;

  PROC SUMMARY DATA=IN&YY..OC&YY.PX NWAY;
    VAR &OCS;
    CLASS &VARS;
    OUTPUT OUT=OC&YY.PX (DROP=_TYPE_ _FREQ_) SUM=;
  RUN;

%MEND;

%AGG(92);
%AGG(93);
%AGG(94);
%AGG(95);
%AGG(96);
%AGG(97);
%AGG(98);
%AGG(99);
%AGG(00);
%AGG(01);

  /*********************************************************/
  /**  PULL THREE YEAR OC COUNTS AND SUM UP.           **/
  /*********************************************************/

%MACRO STP2_2(YR1,YR2,YR3,YRINDX);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;
%LET OCS = OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;

 DATA TEMP;
   SET OC&YR1.PX
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE ))
       OC&YR2.PX
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE ))
       OC&YR3.PX
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE ))
       ;
 INDXYR=&YRINDX;
 RUN;

 PROC SUMMARY DATA=TEMP NWAY NOPRINT;
 CLASS &NEWVARS INDXYR;
 VAR &OCS;
 OUTPUT OUT=Y&YR3..OC&YR3.SUM(DROP=_FREQ_ _TYPE_) SUM=;
```

```
 RUN;

PROC PRINT DATA=Y&YR3..OC&YR3.SUM(OBS=10);
RUN;
%MEND;
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG1) -- CONTINUED

```
%STP2_2(92,93,94,1994);
%STP2_2(93,94,95,1995);
%STP2_2(94,95,96,1996);
%STP2_2(95,96,97,1997);
%STP2_2(96,97,98,1998);
%STP2_2(97,98,99,1999);
%STP2_2(98,99,00,2000);
%STP2_2(99,00,01,2001);
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG1Y5)

```
//P1H4T601 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(1300,300))
//IN92  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC92PX,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC93PX,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC94PX,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC95PX,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC96PX,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC97PX,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC98PX,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC99PX,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC00PX,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ01.MSQYH102.OC01PX,DISP=SHR
//Y96    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC96SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC97SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y98    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC98SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y99    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC99SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y00    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC00SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH202.OC01SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//SYSIN  DD *

*********************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ00M4.PGMLIBAG(HRSAG1)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ01.MSQYH102.OC99PX
* OUTPUTS: P1H4.@MSQ0000.SAS.HRSAG1.OC99PX
* PURPOSE: PRODUCE aggregate FILES THAT CONTAIN COUNTS OF DISCHARGES
*          WITH HIGH RISK SURGERIES AND SUM UP FOR 5 YEARS TO
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*********************************************************************;

%MACRO AGG(YY);

%LET VARS=Y&YY.FAGE Y&YY.SEX Y&YY.RACE Y&YY.DUAL
          Y&YY.MDST Y&YY.URBN Y&YY.STATE Y&YY.CREG;

%LET OCS = OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;

 ************************************************
 * AGGREGATE ACROSS ALL STRATIFIERS FOR EACH YEAR
 ************************************************;
```

```
  PROC SUMMARY DATA=IN&YY..OC&YY.PX NWAY;
    VAR &OCS;
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG1Y5) -- CONTINUED

```
    CLASS &VARS;
    OUTPUT OUT=OC&YY.PX (DROP=_TYPE_ _FREQ_) SUM=;
  RUN;

%MEND;

%AGG(92);
%AGG(93);
%AGG(94);
%AGG(95);
%AGG(96);
%AGG(97);
%AGG(98);
%AGG(99);
%AGG(00);
%AGG(01);

  /*********************************************************/
  /**  PULL THREE YEAR OC COUNTS AND SUM UP.           **/
  /*********************************************************/

%MACRO STP2_2(YR1,YR2,YR3,YR4,YR5,YRINDX);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;
%LET OCS = OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;

 DATA TEMP;
   SET OC&YR1.PX
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE ))
       OC&YR2.PX
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE ))
       OC&YR3.PX
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE ))
       OC&YR4.PX
       (RENAME=(Y&YR4.FAGE=FAGE Y&YR4.SEX=SEX Y&YR4.RACE=RACE
                Y&YR4.MDST=MDST Y&YR4.DUAL=DUAL Y&YR4.URBN=URBN
                Y&YR4.CREG=CREG Y&YR4.STATE=STATE ))
       OC&YR5.PX
       (RENAME=(Y&YR5.FAGE=FAGE Y&YR5.SEX=SEX Y&YR5.RACE=RACE
                Y&YR5.MDST=MDST Y&YR5.DUAL=DUAL Y&YR5.URBN=URBN
                Y&YR5.CREG=CREG Y&YR5.STATE=STATE ))
       ;
 INDXYR=&YRINDX;
 RUN;

 PROC SUMMARY DATA=TEMP NWAY NOPRINT;
 CLASS &NEWVARS INDXYR;
```

```
 VAR &OCS;
 OUTPUT OUT=Y&YR5..OC&YR5.SUM5(DROP=_FREQ_ _TYPE_) SUM=;
 RUN;

PROC PRINT DATA=Y&YR5..OC&YR5.SUM5(OBS=10);
RUN;
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG1Y5) -- CONTINUED

```
%MEND;

%STP2_2(92,93,94,95,96,1996);
%STP2_2(93,94,95,96,97,1997);
%STP2_2(94,95,96,97,98,1998);
%STP2_2(95,96,97,98,99,1999);
%STP2_2(96,97,98,99,00,2000);
%STP2_2(97,98,99,00,01,2001);
```

P1H4.@MSQ0HRS.PGMLIB(BENESUM)

```
//P1H4T601 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(1300,300))
//IN92  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1AG1.M14BENE,DISP=SHR
//Y94    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN94SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y95    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN95SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y96    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN96SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN97SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y98    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN98SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y99    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN99SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y00    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN00SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN01SUM,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//SYSIN  DD *

*******************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ00M4.PGMLIBAG(HRSAG1)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ01.MSQYH102.OC99PX
* OUTPUTS: P1H4.@MSQ0000.SAS.HRSAG1.OC99PX
* PURPOSE: PRODUCE DENOMINATOR aggregate FILES THAT COMBINE 3 YEARS
*          OF BENEFICIARY COUNTS (FOR DISCHARGE RATE COMPUTATION) BY
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*******************************************************************;

%MACRO STP2_2(YR1,YR2,YR3,YRINDX);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;
```

```
 DATA TEMP;
   SET IN&YR1..M14BENE
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
```
P1H4.@MSQ0HRS.PGMLIB(BENESUM) -- CONTINUED
```
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE Y&YR1.COUNT=BENCOUNT))
       IN&YR2..M14BENE
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE Y&YR2.COUNT=BENCOUNT))
       IN&YR3..M14BENE
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE Y&YR3.COUNT=BENCOUNT))
       ;
 INDXYR=&YRINDX;
 RUN;

 PROC SUMMARY DATA=TEMP NWAY NOPRINT;
 CLASS &NEWVARS INDXYR;
 VAR BENCOUNT;
 OUTPUT OUT=Y&YR3..BEN&YR3.SUM(DROP=_FREQ_ _TYPE_) SUM=;
 RUN;

PROC PRINT DATA=Y&YR3..BEN&YR3.SUM(OBS=10);
RUN;
%MEND;

%STP2_2(92,93,94,1994);
%STP2_2(93,94,95,1995);
%STP2_2(94,95,96,1996);
%STP2_2(95,96,97,1997);
%STP2_2(96,97,98,1998);
%STP2_2(97,98,99,1999);
%STP2_2(98,99,00,2000);
%STP2_2(99,00,01,2001);
```

P1H4.@MSQ0HRS.PGMLIB(BENESUM5)

```
//P1H4T6AG JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(1300,300))
//IN92  DD DSN=D#MSQ.@MSQ0000.SAS.Y92M1AG1.M14BENE,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ0000.SAS.Y93M1AG1.M14BENE,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ0000.SAS.Y94M1AG1.M14BENE,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ0000.SAS.Y95M1AG1.M14BENE,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ0000.SAS.Y96M1AG1.M14BENE,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ0000.SAS.Y97M1AG1.M14BENE,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ0000.SAS.Y98M1AG1.M14BENE,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ0000.SAS.Y99M1AG1.M14BENE,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ0000.SAS.Y00M1AG1.M14BENE,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ0000.SAS.Y01M1AG1.M14BENE,DISP=SHR
//Y96    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN96SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN97SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y98    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN98SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y99    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN99SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y00    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN00SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH202.BEN01SU5,
//       DISP=(NEW,CATLG,DELETE),
//       UNIT=HCFA80,SPACE=(TRK,(150,150),RLSE)
//SYSIN  DD *

*********************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ0HRS.PGMLIB(BENESUM5)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ0000.SAS.YxxM1AG1.M14BENE
* OUTPUTS: D#MSQ.@MSQ01.MSQYH202.BENxxSU5
* PURPOSE: PRODUCE DENOMINATOR aggregate FILES THAT COMBINE 5 YEARS
*          OF BENEFICIARY COUNTS (FOR DISCHARGE RATE COMPUTATION) BY
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*********************************************************************;

%MACRO STP2_2(YR1,YR2,YR3,YR4,YR5,YRINDX);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;

 DATA TEMP;
   SET IN&YR1..M14BENE
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE Y&YR1.COUNT=BENCOUNT))
       IN&YR2..M14BENE
```

```
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE Y&YR2.COUNT=BENCOUNT))
```
P1H4.@MSQ0HRS.PGMLIB(BENESUM5) -- CONTINUED
```
       IN&YR3..M14BENE
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE Y&YR3.COUNT=BENCOUNT))
       IN&YR4..M14BENE
       (RENAME=(Y&YR4.FAGE=FAGE Y&YR4.SEX=SEX Y&YR4.RACE=RACE
                Y&YR4.MDST=MDST Y&YR4.DUAL=DUAL Y&YR4.URBN=URBN
                Y&YR4.CREG=CREG Y&YR4.STATE=STATE Y&YR4.COUNT=BENCOUNT))
       IN&YR5..M14BENE
       (RENAME=(Y&YR5.FAGE=FAGE Y&YR5.SEX=SEX Y&YR5.RACE=RACE
                Y&YR5.MDST=MDST Y&YR5.DUAL=DUAL Y&YR5.URBN=URBN
                Y&YR5.CREG=CREG Y&YR5.STATE=STATE Y&YR5.COUNT=BENCOUNT))
       ;
 INDXYR=&YRINDX;
 RUN;

 PROC SUMMARY DATA=TEMP NWAY NOPRINT;
 CLASS &NEWVARS INDXYR;
 VAR BENCOUNT;
 OUTPUT OUT=Y&YR5..BEN&YR5.SU5(DROP=_FREQ_ _TYPE_) SUM=;
 RUN;

PROC PRINT DATA=Y&YR5..BEN&YR5.SU5(OBS=10);
RUN;
%MEND;

%STP2_2(92,93,94,95,96,1996);
%STP2_2(93,94,95,96,97,1997);
%STP2_2(94,95,96,97,98,1998);
%STP2_2(95,96,97,98,99,1999);
%STP2_2(96,97,98,99,00,2000);
%STP2_2(97,98,99,00,01,2001);
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG2)

```
//P1H4T601 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(2500,500))
//IN92  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y92.OLD,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y93.OLD,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y94.OLD,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y95.OLD,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y96.OLD,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y97.OLD,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y98.OLD,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y99.OLD,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y00.OLD,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y01,DISP=SHR
//Y01    DD DSN=D#MSQ.@MSQ01.MSQYH502.RM01SUM,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(800,500),RLSE)
//SYSIN  DD *
*******************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ0HRS.PGMLIB(HRSAG2)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ01.MSQYH500.OC7Yxx
* OUTPUTS: D#MSQ.@MSQ01.MSQYH502.RMxxSUM
* PURPOSE: PRODUCE aggregate FILES THAT CONTAIN COUNTS OF READMISSIONS
*          & DEATHS FOR HIGH RISK SURGERIES AND SUM UP FOR 3 YEARS TO
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*******************************************************************;

  /*********************************************************/
  /**  PULL THREE YEAR OC COUNTS AND SUM UP.           **/
  /*********************************************************/

%MACRO STP2_2(YR1,YR2,YR3);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;

 DATA TEMP (KEEP=FAGE SEX RACE MDST DUAL URBN CREG STATE
                 OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10
                 OC11 OC12 OC13 OC14
                 READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM
                 Y&YR3.QNT1 Y&YR3.QNT2 Y&YR3.QNT3 Y&YR3.QNT4 Y&YR3.QNT5
                 Y&YR3.QNT6 Y&YR3.QNT7 Y&YR3.QNT9
                 Y&YR3.QNT11 Y&YR3.QNT12 Y&YR3.QNT13 Y&YR3.QNT14);
   SET IN&YR1..OC7Y&YR1
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE ))
       IN&YR2..OC7Y&YR2
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE ))
       IN&YR3..OC7Y&YR3
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
```

```
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE ))
      ;
*INDXYR=&YRINDX;
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG2) -- CONTINUED

```
 RUN;

PROC SUMMARY DATA=tEMP (WHERE=(OC1=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT1;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC1 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC2=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT2;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC2 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC3=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT3;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC3 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC4=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT4;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC4 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC5=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT5;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC5 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC6=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT6;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC6 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC7=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT7;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC7 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC8=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT7;          * VOL7 FOR OC8;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC8 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC9=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT9;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC9 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC10=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT9;                              * VOL9 FOR OC10;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC10 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC11=1)) NWAY;
```

```
CLASS &NEWVARS Y&YR3.QNT11;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC11 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC12=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT12;
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG2) -- CONTINUED
```
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC12 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC13=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT13;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC13 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC14=1)) NWAY;
CLASS &NEWVARS Y&YR3.QNT14;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC14 (DROP=_FREQ_ _TYPE_) SUM=;

DATA OC1; SET OC1;  OCS=1;
DATA OC2; SET OC2;  OCS=2;
DATA OC3; SET OC3;  OCS=3;
DATA OC4; SET OC4;  OCS=4;
DATA OC5; SET OC5;  OCS=5;
DATA OC6; SET OC6;  OCS=6;
DATA OC7; SET OC7;  OCS=7;
DATA OC8; SET OC8;  OCS=8;
DATA OC9; SET OC9;  OCS=9;
DATA OC10; SET OC10;  OCS=10;
DATA OC11; SET OC11;  OCS=11;
DATA OC12; SET OC12;  OCS=12;
DATA OC13; SET OC13;  OCS=13;
DATA OC14; SET OC14;  OCS=14;

DATA Y&YR3..RM&YR3.SUM;
  SET OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;
RUN;

PROC SUMMARY DATA=Y&YR3..RM&YR3.SUM;
CLASS OCS;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=RM&YR3.SUM(DROP=_FREQ_ _TYPE_) SUM=;

PROC PRINT DATA=RM&YR3.SUM ;

%MEND;

*STP2_2(92,93,94);
*STP2_2(93,94,95);
*STP2_2(94,95,96);
*STP2_2(95,96,97);
*STP2_2(96,97,98);
*STP2_2(97,98,99);
*STP2_2(98,99,00);
%STP2_2(99,00,01);
```

P1H4.@MSQ0HRS.PGMLIB(HRSAG2Y5)

```
//P1H4Y597 JOB (MSQ00,ANASYS),P1H4CP,MSGCLASS=Q,CLASS=B,NOTIFY=P1H4
//P1H4 EXEC SAS8
//WORK DD SPACE=(CYL,(2500,500))
//IN92  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y92.OLD,DISP=SHR
//IN93  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y93.OLD,DISP=SHR
//IN94  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y94.OLD,DISP=SHR
//IN95  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y95.OLD,DISP=SHR
//IN96  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y96.OLD,DISP=SHR
//IN97  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y97.OLD,DISP=SHR
//IN98  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y98.OLD,DISP=SHR
//IN99  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y99.OLD,DISP=SHR
//IN00  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y00.OLD,DISP=SHR
//IN01  DD DSN=D#MSQ.@MSQ01.MSQYH500.OC7Y01,DISP=SHR
//Y97    DD DSN=D#MSQ.@MSQ01.MSQYH502.RM97SUM5,
//       DISP=(NEW,CATLG,DELETE),
//       SPACE=(CYL,(800,500),RLSE)
//SYSIN  DD *
*******************************************************************
* PROJ   : MQMS
* PGM    : P1H4.@MSQ0HRS.PGMLIB(HRSAG2)
* PGMR   : c.park
* COMPANY: ANASYS
* INPUTS : D#MSQ.@MSQ01.MSQYH500.OC7Yxx
* OUTPUTS: D#MSQ.@MSQ01.MSQYH502.RMxxSUM5
* PURPOSE: PRODUCE aggregate FILES THAT CONTAIN COUNTS OF READMISSIONS
*          & DEATHS FOR HIGH RISK SURGERIES AND SUM UP FOR 5 YEARS TO
*          level of age, sex, race, dual, Medicare status, urban/rural  ,
*          and beneficiary REGION AND STATE.
*******************************************************************;

  /*********************************************************/
  /**  PULL THREE YEAR OC COUNTS AND SUM UP.           **/
  /*********************************************************/

%MACRO STP2_2(YR1,YR2,YR3,YR4,YR5);

%LET NEWVARS=FAGE SEX RACE DUAL MDST URBN CREG STATE;

 DATA TEMP (KEEP=FAGE SEX RACE MDST DUAL URBN CREG STATE
                 OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10
                 OC11 OC12 OC13 OC14
                 READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM);
   SET IN&YR1..OC7Y&YR1
       (RENAME=(Y&YR1.FAGE=FAGE Y&YR1.SEX=SEX Y&YR1.RACE=RACE
                Y&YR1.MDST=MDST Y&YR1.DUAL=DUAL Y&YR1.URBN=URBN
                Y&YR1.CREG=CREG Y&YR1.STATE=STATE ))
       IN&YR2..OC7Y&YR2
       (RENAME=(Y&YR2.FAGE=FAGE Y&YR2.SEX=SEX Y&YR2.RACE=RACE
                Y&YR2.MDST=MDST Y&YR2.DUAL=DUAL Y&YR2.URBN=URBN
                Y&YR2.CREG=CREG Y&YR2.STATE=STATE ))
       IN&YR3..OC7Y&YR3
       (RENAME=(Y&YR3.FAGE=FAGE Y&YR3.SEX=SEX Y&YR3.RACE=RACE
                Y&YR3.MDST=MDST Y&YR3.DUAL=DUAL Y&YR3.URBN=URBN
                Y&YR3.CREG=CREG Y&YR3.STATE=STATE ))
       IN&YR4..OC7Y&YR4
       (RENAME=(Y&YR4.FAGE=FAGE Y&YR4.SEX=SEX Y&YR4.RACE=RACE
```

```
               Y&YR4.MDST=MDST Y&YR4.DUAL=DUAL Y&YR4.URBN=URBN
               Y&YR4.CREG=CREG Y&YR4.STATE=STATE ))
       IN&YR5..OC7Y&YR5
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG2Y5) -- CONTINUED
```
       (RENAME=(Y&YR5.FAGE=FAGE Y&YR5.SEX=SEX Y&YR5.RACE=RACE
               Y&YR5.MDST=MDST Y&YR5.DUAL=DUAL Y&YR5.URBN=URBN
               Y&YR5.CREG=CREG Y&YR5.STATE=STATE ))
       ;
*INDXYR=&YRINDX;
 RUN;

PROC SUMMARY DATA=tEMP (WHERE=(OC1=1)) NWAY;
CLASS &NEWVARS; *Y&YR3.QNT1;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC1 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC2=1)) NWAY;
CLASS &NEWVARS; *Y&YR3.QNT2;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC2 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC3=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT3;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC3 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC4=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT4;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC4 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC5=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT5;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC5 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC6=1)) NWAY;
CLASS &NEWVARS;  *Y&YR3.QNT6;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC6 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC7=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT7;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC7 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC8=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT7;          * VOL7 FOR OC8;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC8 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC9=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT9;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC9 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC10=1)) NWAY;
```

```
CLASS &NEWVARS; * Y&YR3.QNT9;                              * VOL9 FOR OC10;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC10 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC11=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT11;
```
P1H4.@MSQ0HRS.PGMLIB(HRSAG2Y5) -- CONTINUED
```
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC11 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC12=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT12;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC12 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC13=1)) NWAY;
CLASS &NEWVARS; * Y&YR3.QNT13;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC13 (DROP=_FREQ_ _TYPE_) SUM=;

PROC SUMMARY DATA=tEMP (WHERE=(OC14=1)) NWAY;
CLASS &NEWVARS;  *Y&YR3.QNT14;
VAR READM30 M5DENOM30 READM90 M5DENOM90 DEATH M6DENOM;
OUTPUT OUT=OC14 (DROP=_FREQ_ _TYPE_) SUM=;

DATA OC1; SET OC1;  OCS=1;
DATA OC2; SET OC2;  OCS=2;
DATA OC3; SET OC3;  OCS=3;
DATA OC4; SET OC4;  OCS=4;
DATA OC5; SET OC5;  OCS=5;
DATA OC6; SET OC6;  OCS=6;
DATA OC7; SET OC7;  OCS=7;
DATA OC8; SET OC8;  OCS=8;
DATA OC9; SET OC9;  OCS=9;
DATA OC10; SET OC10;  OCS=10;
DATA OC11; SET OC11;  OCS=11;
DATA OC12; SET OC12;  OCS=12;
DATA OC13; SET OC13;  OCS=13;
DATA OC14; SET OC14;  OCS=14;

DATA Y&YR5..RM&YR5.SUM5;
  SET OC1 OC2 OC3 OC4 OC5 OC6 OC7 OC8 OC9 OC10 OC11 OC12 OC13 OC14;
RUN;

PROC PRINT DATA=Y&YR5..RM&YR5.SUM5 (OBS=20);

%MEND;

*STP2_2(92,93,94,95,96);
%STP2_2(93,94,95,96,97);
*STP2_2(94,95,96,97,98);
*STP2_2(95,96,97,98,99);
*STP2_2(96,97,98,99,00);
*STP2_2(97,98,99,00,01);
```