APPENDIX I

SAS CODE

1. CREATE POST-STRATIFICATION VARIABLES

```
*****************************************************************************
*
* PROGRAM: POSTRAT.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8687-200)
* PURPOSE: SAS LOAD THE POST-STRATIFICATION DATA (AS OF 9/30/1999) - FORM A.
* WRITTEN: 02/16/2000 BY KEITH RATHBUN
*
* INPUTS:  1) VASMATHM.TXT - POST-STRATIFICATION RAW DATA SET - FORM A
*
* OUTPUTS: 1) POSTRAT1.SD2 - POST-STRATIFICATION SAS DATA SET - FORM A PART 1
*          2) POSTRAT2.SD2 - POST-STRATIFICATION SAS DATA SET - FORM A PART 2
*          3) POSTRAT.SD2  - POST-STRATIFICATION SAS DATA SET - FORM A ALL
*
* NOTES:
*
     * 1)  ACV is only one digit, however, it is stored in a 3-digit field.
* 2)  Eligibility Flag is a 'Y' or 'N' indicating the person is eligible
*     to receive care in the MHS.  This is based on DMIS Medical Eligibility
*     not in ('0' '3' '8') or DMIS Beneficiary Category not in ('0' '9')
*     being equal to 'Y'.
* 3)  TSP Eligibility Flag is a 'Y' or 'N'.  'Y' indicates that the person
*     lived in a TSP eligible ZIP code, was Non-Active Duty, and 65+ years
*     of age.  The TSP Site for Non-Enrollees would then indicate which
*     TSP site the person lived in.  So to not cause confusion between this
*     field and the TSP HCFA Site Code field I just used site names here.
*     They are SANANTONIO (BAMC & Wilford Hall), SANDIEGO, COSPRINGS
*     (Evans & Academy), DOVER, KEESLER, MADIGAN, and REYSHEP (Reynolds &
*     Sheppard).
* 4)  TSP Eligibility Flag was also set to 'Y' for all current TSP enrollees
*     as indicated in the TSP Member Months file.
* 5)  For TSP enrollees, the Enrollment DMIS ID will indicate the MTF within
*     the TSP Site that the beneficiary is enrolled to, otherwise this field
*     is as coded on the EBC file.
* 6)  PCM Derived is either 'MIL' or 'CIV' (again to avoid confusion with
*     the 00 & 01 in Primary Care Manager Code).  This field was code 'CIV'
*     for the following Enrollment DMIS IDs: 7901-7916, 8001-8036, 6901-6912.
*     All others were coded as 'MIL'.
* 7)  As requested this does not include any beneficiaries below the age of 18.
* 8)  All files were merged together based on the common fields of Sponsor
*     SSN and DEERS Dependent Suffix.
* 9)  DMIS Derived ZIP Code contains the Unit ZIP code of an Active Duty
*     person and the Residence ZIP code of Dependents.  In some cases where
*     a ZIP code is missing, DMIS attempts to assign one based on other
*     available information.  For example if a Spouse's ZIP is unknown, it
*     may attempt to assign the ZIP code based on a known residence ZIP of
*     the Sponsor.
* 10) DMIS Catchment/NonCatchment ID is then appended based on the DMIS
*     Derived ZIP Code.
* 11) The SAS file is split into two parts for CD storage purposes.
*     (total observations = 10,564,222)
*
*****************************************************************************
;
LIBNAME IN   "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT  "D:\KEITH\DOD99-8676\DATA";

OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

DATA OUT.POSTRAT1 OUT.POSTRAT2 OUT.POSTRAT;
   INFILE "D:\KEITH\DOD99-8676\DATA\VASMATHM.TXT"
          RECFM=V LRECL=999 LENGTH=OLEN MISSOVER;
   LENGTH ZTSPNE $10;
   INPUT  @001 SSN        $Char9.  /* Sponsor SSN                        */
               FILLER     $Char1.
               FSN        $Char1.  /* Family Sequence Number             */
               FILLER     $Char1.
               ZTYPE      $Char1.  /* Record Type                        */
               FILLER     $Char1.
               DDS        $Char2.  /* DEERS Dependent Suffix             */
               FILLER     $Char1.
               ZRZIP      $Char5.  /* Residence ZIP Code                 */
               FILLER     $Char1.
```

```
            ZRCATCH    $Char4.  /* Residence Hospital Catchment       */
            FILLER     $Char1.
            ZUCATCH    $Char4.  /* Unit Hospital Catchment            */
            FILLER     $Char1.
            ZSTATUS    $Char1.  /* Status of Sponsor                  */
            FILLER     $Char1.
            ZACFLAG    $Char3.  /* Alternate Care System History Flag */
            FILLER     $Char1.
            ZAGE            3.  /* Age as of 30 September 1999        */
            FILLER     $Char1.
            ZMELIG     $Char1.  /* DMIS Medical Eligibility           */
            FILLER     $Char1.
            ZBENE      $Char1.  /* DMIS Beneficiary Category          */
            FILLER     $Char1.
            ZCATCHID   $Char4.  /* DMIS Catchment/NonCatchment ID     */
            FILLER     $Char1.
            ZDZIP      $Char5.  /* DMIS Derived ZIP Code              */
            FILLER     $Char1.
            ZACV       $Char3.  /* Alternate Care Value               */
            FILLER     $Char1.
            ZPCM       $Char2.  /* Primary Care Manager Code          */
            FILLER     $Char1.
            ZENRID     $Char4.  /* Enrollment DMIS ID                 */
            FILLER     $Char1.
            ZTSPSITE   $Char2.  /* TSP HCFA Site Code                 */
            FILLER     $Char1.
            ZPCMD      $Char3.  /* PCM Derived                        */
            FILLER     $Char1.
            ZELIG      $Char1.  /* Eligibility Flag                   */
            FILLER     $Char1.
            ZTSPELIG   $Char1.  /* TSP Eligibility Flag               */
            FILLER     $Char1.
            ZTSPNE     $        /* TSP Site for Non-Enrollees         */
        ;
   ************************************************************************
   * Construct SSNSMPL as SSN & FSN & DDS
   ************************************************************************;
   LENGTH SSNSMPL $12;
   SSNSMPL = SSN || FSN || DDS;

   LABEL SSNSMPL  = 'SSNSMPL - SSN & FSN & DDS          '
         SSN      = 'Sponsor SSN                        '
         FSN      = 'Family Sequence Number             '
         ZTYPE    = 'Record Type                        '
         DDS      = 'DEERS Dependent Suffix             '
         ZRZIP    = 'Residence ZIP Code                 '
         ZRCATCH  = 'Residence Hospital Catchment       '
         ZUCATCH  = 'Unit Hospital Catchment            '
         ZSTATUS  = 'Status of Sponsor                  '
         ZACFLAG  = 'Alternate Care System History Flag '
         ZAGE     = 'Age as of 30 September 1999        '
         ZMELIG   = 'DMIS Medical Eligibility           '
         ZBENE    = 'DMIS Beneficiary Category          '
         ZCATCHID = 'DMIS Catchment/NonCatchment ID     '
         ZDZIP    = 'DMIS Derived ZIP Code              '
         ZACV     = 'Alternate Care Value               '
         ZPCM     = 'Primary Care Manager Code          '
         ZENRID   = 'Enrollment DMIS ID                 '
         ZTSPSITE = 'TSP HCFA Site Code                 '
         ZPCMD    = 'PCM Derived                        '
         ZELIG    = 'Eligibility Flag                   '
         ZTSPELIG = 'TSP Eligibility Flag               '
         ZTSPNE   = 'TSP Site for Non-Enrollees         '
    ;
    DROP FILLER;
    IF _N_ LE 5000000 THEN OUTPUT OUT.POSTRAT1;
    ELSE OUTPUT OUT.POSTRAT2;
    OUTPUT OUT.POSTRAT;
RUN;

TITLE1 "1999 DOD Health Survey Post-Stratification (8687-200)";
TITLE2 "Program Name: POSTRAT.SAS By Keith Rathbun";
TITLE3 "Program Outputs: POSTRAT1/2.SD2";
```

```
PROC SORT DATA=OUT.POSTRAT; BY SSNSMPL; RUN;
PROC CONTENTS DATA=OUT.POSTRAT; RUN;

PROC FREQ DATA=OUT.POSTRAT;
   TABLES FSN
          ZTYPE
          DDS
          ZRCATCH
          ZUCATCH
          ZSTATUS
          ZACFLAG
          ZAGE
          ZMELIG
          ZBENE
          ZCATCHID
          ZACV
          ZPCM
          ZENRID
          ZTSPSITE
          ZPCMD
          ZELIG
          ZTSPELIG
          ZTSPNE
   /MISSING LIST;
RUN;
```

```
*****************************************************************************
*
* PROGRAM:  IMPORTA.SAS
* TASK:     DOD HEALTH CARE SURVEY ANALYSIS (8687-100)
* PURPOSE:  Import the 1999 DoD HCSDB final adult (Form A) Dataset
* WRITTEN:  03/22/2000 BY KEITH RATHBUN
* SEQUENCE: First program run
*
* INPUTS:  1) HARET02.XPT - 1999 DOD Health Survey Data - Adult (Form A)
*                           (SAS Transport Format)
*
* OUTPUTS: 1) IMPORTA.SD2 - 1999 DOD Health Survey Data - Adult (Form A)
*                           (SAS SD2 Format)
*
*****************************************************************************
*;
%LET FILENAME = HARET02;
LIBNAME XPT XPORT "D:\KEITH\DOD99-8676\DATA\&filename..xpt";
LIBNAME    SURVEY "D:\KEITH\DOD99-8676\DATA";
LIBNAME   LIBRARY "D:\KEITH\DOD99-8676\DATA\FMTLIBA";
OPTIONS PS=79 LS=132 COMPRESS=YES NOCENTER;

*****************************************************************************
* Import the 1999 DoD HCSDB final adult (Form A) Dataset and produce
* summary report of CONTENTS and final status.
*****************************************************************************;
PROC COPY IN=XPT OUT=SURVEY; RUN;

*****************************************************************************
* Rename the 1999 DoD HCSDB final adult (Form A) Dataset for consistency
* with program naming convention
*****************************************************************************;
PROC DATASETS LIBRARY=SURVEY MEMTYPE=DATA;
   CHANGE HARET02=IMPORTA;
RUN;

TITLE1 "8687:  1999 DOD Health Survey Data - Final Form A (Adult)";
PROC CONTENTS DATA=SURVEY.IMPORTA; RUN;

TITLE2 "Unformatted Dump";
PROC PRINT DATA=SURVEY.IMPORTA(OBS=50);
   FORMAT _ALL_;
RUN;

TITLE2 "FREQ of Final Status";
PROC FREQ  DATA=SURVEY.IMPORTA;
   TABLE FLAG_FIN FLAG_DUP /MISSING LIST;
RUN;
```

```
*************************************************************************
*
* PROGRAM:  CONVAR1A.SAS
* TASK:     1999 DOD HEALTH CARE SURVEY ANALYSIS (8676-100)
* PURPOSE:  Construct KEY Post-Stratification variables
*
* WRITTEN:  02/22/2000 BY DON JANG
*
* MODIFIED: 02/23/2000 BY KEITH RATHBUN - Consolidate the calculation of
*           all post-stratification variables into this program.  Update
*           code as required.
*
* INPUTS:   1) POSTRAT.SD2 - Post-Stratification dataset - FORM A
*
* OUTPUTS:  2) CONVAR1A.SD2 - Constructed Variables FILE1A
*
* INCLUDES: 1) CACSMPL.INC - Colapsing rules for catchment area
*
*************************************************************************
*;
LIBNAME  IN  "D:\KEITH\DOD99-8676\DATA";
LIBNAME  OUT "D:\KEITH\DOD99-8676\DATA";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

DATA OUT.CONVAR1A;
   SET IN.POSTRAT;

   LENGTH BFGROUPP 3.
          AGEGROUP 3.
          ENGROUPP 3.
   ;
   ************************************************************************
   * Calculate BFGROUPP
   ************************************************************************;
   IF ZTYPE = 'S' THEN DO;
      IF ZSTATUS IN ('A','B','J','N','P','V') THEN BFGROUPP = 1;
      ELSE IF ZAGE LE 64 THEN BFGROUPP = 3;
      ELSE IF ZAGE GT 64 THEN BFGROUPP = 4;
   END;
   ELSE IF ZTYPE = 'T' THEN DO;
      IF ZSTATUS IN ('A','B','J','N','P','V') AND (ZAGE LE 64) THEN BFGROUPP = 2;
      ELSE IF ZAGE LE 64 THEN BFGROUPP = 3;
      ELSE IF ZAGE GT 64 THEN BFGROUPP = 4;
   END;

   ************************************************************************
   * Calculate AGEGROUP
   ************************************************************************;
   IF      ZAGE LT '020' THEN AGEGROUP = 1;
   ELSE IF ZAGE LE '029' THEN AGEGROUP = 2;
   ELSE IF ZAGE LE '039' THEN AGEGROUP = 3;
   ELSE IF ZAGE LE '064' THEN AGEGROUP = 4;
   ELSE IF ZAGE GE '065' THEN AGEGROUP = 5;
```

```
************************************************************************
* Calculate ENGROUPP
************************************************************************;
IF ZAGE LT 65 AND ZPCMD = 'MIL' THEN ENGROUPP = 1;
ELSE IF ZAGE LT 65 AND ZPCMD = 'CIV' THEN ENGROUPP = 2;
ELSE IF ZACV ='D' THEN ENGROUPP = 1;
ELSE ENGROUPP = 3;

************************************************************************
* Assign CACSMPL
************************************************************************;
IF ENGROUPP = 1 THEN CELLP = ZENRID*1;
ELSE CELLP = ZCATCHID*1;
%INCLUDE "D:\KEITH\DOD99-8676\SAS\CACSMPL.INC";

LABEL  BFGROUPP = 'Beneficiary Group as of Sept. 30, 1999'
       AGEGROUP = 'Age Group as of Sept. 30, 1999'
       ENGROUPP = 'Enrollment Group as of Sept. 30, 1999'
       CACSMPL  = 'CACSMPL - Catchment Area'
       CELLP    = 'Catchment Area Post-Cell'
;
************************************************************************
* Assign senior prime to beneficiaries in subvention demo areas.
************************************************************************;
IF ZTSPELIG = 'Y' THEN DO;
   IF ZTSPNE = "COSPRINGS" THEN DO;
 IF ZCATCHID = 32 THEN CACSMPL = 32;
 ELSE CACSMPL = 33;
   END;
   IF ZTSPNE = "DOVER" THEN CACSMPL = 36;
   IF ZTSPNE = "KEESLER"  THEN CACSMPL = 73;
   IF ZTSPNE = "MADIGAN" THEN CACSMPL = 125;
   IF ZTSPNE = "REYSHEP" THEN CACSMPL = ZCATCHID;
   IF ZTSPNE = "SANANTONIO" THEN DO;
 IF ZCATCHID = 117 THEN CACSMPL = 117;
 ELSE CACSMPL = 109;
   END;
   IF ZTSPNE = "SANDIEGO" THEN CACSMPL = 29;

   IF ZTSPSITE = "P8" THEN DO;
 IF ZCATCHID = 33 THEN CACSMPL = 33;
 ELSE CACSMPL = 32;
   END;
   IF ZTSPSITE = "P1" THEN CACSMPL = 36;
   IF ZTSPSITE = "P4" THEN CACSMPL = 73;
   IF ZTSPSITE = "PA" THEN CACSMPL = 125;
   IF ZTSPSITE = "P6" THEN DO;
 IF ZCATCHID IN (98, 113, 117) THEN CACSMPL = ZCATCHID;
 ELSE CACSMPL = 109;
   END;
   IF ZTSPSITE = "P9" THEN CACSMPL = 29;
END;

************************************************************************
* Assign Post-Stratification Identification
************************************************************************;
IF ENGROUPP IN(1,2) THEN TVAR = 1;
ELSE TVAR = 3; DROP TVAR;
IF BFGROUPP IN(2,3) THEN PBGROUP = 2;
ELSE PBGROUP = BFGROUPP; DROP PBGROUP;
IF CACSMPL > 9900 AND BFGROUPP IN (1,2,3) THEN PBGROUP = 1;

IF CACSMPL IN (1,4,15,35,50,58,69,74,96,100,118,122,250,326) and TVAR = 3 then do;
   PBGROUP = 2;
   TVAR = 1;
END;
IF CACSMPL IN (19,62,79,84,112,806) and TVAR = 3 and PBGROUP = 1 then
   TVAR = 1;
IF CACSMPL IN (36,46,51,76,85,107,119,128) and TVAR = 3 and PBGROUP = 2 then
   TVAR = 1;
IF CACSMPL IN (638,806) and PBGROUP = 4 then PBGROUP = 2;
IF CACSMPL = 9913 then TVAR = 3;
IF CACSMPL = 9914 then PBGROUP = 1;
```

```
   IF CACSMPL = 9999 and PBGROUP = 2 then PBGROUP = 1;

   ************************************************************************
   * Further collapsing for email dated 4/5/2000 (final weighting)
   ************************************************************************;
   IF PBGROUP = 1 THEN TVAR = 1;
   IF CACSMPL in(606, 633) AND TVAR = 3 AND PBGROUP = 4 THEN PBGROUP = 2;
   IF CACSMPL IN(616, 638) AND TVAR = 1 AND PBGROUP = 2 THEN TVAR = 3;
   IF CACSMPL = 806 AND TVAR = 3 AND PBGROUP = 2 THEN TVAR = 1;
   IF CACSMPL = 9913 THEN DO;
      PBGROUP = 1;
      TVAR = 1;
   END;

   POSTSTR = CACSMPL*1000 + TVAR*100 + PBGROUP*10;
RUN;
TITLE1 "1999 DOD Health Survey Post-Stratification Assignments (8687-100)";
TITLE2 "Program Name: CONVAR1A.SAS By Don Jang and Keith Rathbun";
TITLE3 "Program Output: CONVAR1A.SD2";

PROC CONTENTS; RUN;

PROC FREQ;
WHERE ZTSPELIG = "Y" AND ZELIG = "Y";
TABLES ZTSPNE*ZCATCHID ZTSPSITE*ZCATCHID
   /MISSING LIST;
RUN;

*****************************************************************************
*
* PROGRAM: POSTRATA.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8687-100)
* PURPOSE: MERGE THE POST-STRATIFICATION DATA (AS OF 9/30/1999) WITH
*          ENTIRE FORM A SAMPLE
* WRITTEN: 02/24/2000 BY KEITH RATHBUN
*
* INPUTS:  1) CONVAR1A.SD2 - Post-Stratification file with constructed
*             variables for weighting and response rates - FORM A
*          2) XWALK.SD2    - CROSS-WALK FILE
*          3) IMPORTA.SD2 - FORM A Questionnaire/Sample File
*
* OUTPUTS: 1) POSTRATA.SD2 - Post-Stratification file merged with
*             FORM A Questionnaire/Sample File
*
* INCLUDES: 1) CONSVAR0.SAS - Construct XREGION and CONUS
*
*****************************************************************************
;
LIBNAME IN1 "F:\";
LIBNAME IN2 "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY "D:\KEITH\DOD99-8676\DATA\FMTLIBA";

OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

*****************************************************************************
* SORT the XWALK file by SSNSMPL for merge purposes
*****************************************************************************;
PROC SORT DATA=IN1.XWALK OUT=XWALK; BY SSNSMPL; RUN;

*****************************************************************************
* SORT the Post-Stratification file by SSNSMPL for merge purposes
*****************************************************************************;
PROC SORT DATA=IN2.CONVAR1A OUT=CONVAR1A; BY SSNSMPL; RUN;

*****************************************************************************
* MERGE the Post-Stratification file with the XWALK file by SSNSMPL to
* attach the MPRID for merging with the Form A Questionnaire/Sample file
*****************************************************************************;
DATA MERGED;
   MERGE XWALK(IN=IN1)
         CONVAR1A(IN=IN2);
   BY SSNSMPL;
```

```
   IF IN1 AND IN2;
RUN;

*************************************************************************
* SORT the MERGED file by MPRID so that it can be merged with the
* Form A Questionnaire/Sample file
*************************************************************************;
PROC SORT DATA=MERGED; BY MPRID; RUN;

*************************************************************************
* SORT the Form A Questionnaire/Sample file by MPRID
*************************************************************************;
PROC SORT DATA=IN2.IMPORTA OUT=IMPORTA; BY MPRID; RUN;

*************************************************************************
* MERGE the Post-Stratification file with the Form A Questionnaire/Sample
* file by MPRID
*************************************************************************;
DATA OUT.POSTRATA;
   MERGE MERGED(IN=IN1)
         IMPORTA(IN=IN2);
   BY MPRID;
   LENGTH XREGION  3.
          CONUS    3. ;
   LABEL  XREGION  = 'XREGION - Region'
          CONUS    = 'CONUS - CONUS/OCONUS Indicator'
          POSTSTR  = 'Post-Stratification Identifier';

   DROP SSN SSNSMPL;
   *************************************************************************
   * Calculate missing Post-Stratification values (i.e. No corresponding
   * SSNSMPL) from the original sampling variables.
   *************************************************************************;
   IF CACSMPL  = . THEN CACSMPL  = STESMPL;
   IF BFGROUPP = . THEN BFGROUPP = BFGROUP;
   IF ENGROUPP = . THEN ENGROUPP = ENGROUP;
   IF AGEGROUP = . THEN DO;
      IF      AGE LT 20 THEN AGEGROUP = 1;
      ELSE IF AGE LE 29 THEN AGEGROUP = 2;
      ELSE IF AGE LE 39 THEN AGEGROUP = 3;
      ELSE IF AGE LE 64 THEN AGEGROUP = 4;
      ELSE IF AGE GE 65 THEN AGEGROUP = 5;
   END;

   *************************************************************************
   * Assign Post-Stratification Identification to fix missing values.
   *************************************************************************;
   IF ENGROUPP IN(1,2) THEN TVAR = 1;
   ELSE TVAR = 3; DROP TVAR;
   IF BFGROUPP IN(2,3) THEN PBGROUP = 2;
   ELSE PBGROUP = BFGROUPP; DROP PBGROUP;
   IF CACSMPL > 9900 AND BFGROUPP IN (1,2,3) THEN PBGROUP = 1;

   IF CACSMPL = 31 THEN CACSMPL = 32;
   IF CACSMPL = 449 THEN CACSMPL = 9999;

   IF CACSMPL IN (1,4,15,35,50,58,69,74,96,100,118,122,250,326) and TVAR = 3 then do;
      PBGROUP = 2;
      TVAR = 1;
   END;
   IF CACSMPL IN (19,62,79,84,112,806) and TVAR = 3 and PBGROUP = 1 then
      TVAR = 1;
   IF CACSMPL IN (36,46,51,76,85,107,119,128) and TVAR = 3 and PBGROUP = 2 then
      TVAR = 1;
   IF CACSMPL IN (638,806) and PBGROUP = 4 then PBGROUP = 2;
   IF CACSMPL = 9913 then TVAR = 3;
   IF CACSMPL = 9914 then PBGROUP = 1;
   IF CACSMPL = 9999 and PBGROUP = 2 then PBGROUP = 1;

   *************************************************************************
   * Further collapsing for email dated 4/5/2000 (final weighting)
   *************************************************************************;
   IF PBGROUP = 1 THEN TVAR = 1;
```

```
   IF CACSMPL in(606, 633) AND TVAR = 3 AND PBGROUP = 4 THEN PBGROUP = 2;
   IF CACSMPL IN(616, 638) AND TVAR = 1 AND PBGROUP = 2 THEN TVAR = 3;
   IF CACSMPL = 806 AND TVAR = 3 AND PBGROUP = 2 THEN TVAR = 1;
   IF CACSMPL = 9913 THEN DO;
      PBGROUP = 1;
      TVAR = 1;
   END;

   POSTSTR = CACSMPL*1000 + TVAR*100 + PBGROUP*10;

   ************************************************************************
   * Construct XREGION and CONUS.
   ************************************************************************;
   %INCLUDE "D:\KEITH\DOD99-8676\SAS\CONSVAR0.SAS";

   ************************************************************************
   * KEEP every sampled beneficiary.
   ************************************************************************;
   IF IN2;
RUN;

TITLE1 "1999 DOD Health Survey Post-Stratification (8687-100)";
TITLE2 "Program Name: POSTRATA.SAS By Keith Rathbun";
TITLE3 "Program Outputs: POSTRATA.SD2";
PROC CONTENTS; RUN;

PROC FREQ;
TABLES POSTSTR
       BFGROUP BFGROUPP BFGROUP*BFGROUPP
       ENGROUP ENGROUPP ENGROUP*ENGROUPP
       XREGION CONUS    XREGION*CONUS
       XREGION*CACSMPL
   /MISSING LIST;
RUN;
```

2. IMPLEMENT CODING SCHEME AND CODING TABLES

```
*****************************************************************
* PROGRAM:  CSCHEM99.SAS
* PURPOSE:  APPLY CODING SCHEME TO DATA.
* WRITTEN:  01/03/00 Rankin
* DERIVED FROM:  Cschem98.sas
* NOTE: no SRDATE in 99 survey                                  **
*
* INPUT: C:\MY DOCUMENTS\8687\POSTRATA.SD2
* OUTPUT:C:\MY DOCUMENTS\8687\CSCHM99A.SD2
*****************************************************************;

OPTIONS PS=67 LS=120 PAGENO=1;

LIBNAME LIBRARY 'C:\MY DOCUMENTS\8687\FMTLIB';
LIBNAME HCDATA 'C:\MY DOCUMENTS\8687';
FILENAME GETFMTS 'C:\MY DOCUMENTS\8687';

%LET OUTDATA=CSCHM99A;


/* Vairable names in survey -- become recoded varibles */

%Let varlist1 =
H99001  H99002  H99003  H99004  H99005  H99006  H99007A
H99007B H99007C H99007D H99007E H99007F H99007G H99007H
H99007I H99007J H99007K H99007L H99008  H99009A H99009B
H99010  H99011  H99012  H99013  H99014  H99015  H99017
H99018  H99019A H99019B H99019C H99020  H99021A H99021B
H99022  H99023  H99024  H99025  H99026  H99027  H99028
H99029  H99030  H99031  H99032  H99033  H99034  H99035
H99036  H99037  H99038  H99039A H99039B H99039C H99039D
H99039E H99039F H99039G H99039H H99039I H99039J H99040
H99041  H99042  H99043  H99044  H99045  H99046  H99047
H99048  H99049  H99050  H99051  H99052  H99053  H99054
H99055  H99056  H99057  H99058  H99059  H99060  H99061A
H99061B H99062  H99063  H99064  H99065  H99066  H99067
H99068  H99069  H99070  H99071  H99072  H99073  H99074
H99075  H99076A H99076B H99076C H99076D H99076E H99076F
H99076G H99077  H99078A H99078B H99079A H99079B H99080A
H99080B H99081  H99082A H99082B H99082C H99083  H99084
SRRACEA SRRACEB SRRACEC SRRACED SRRACEE SREDA   SREDB
SREDC   SREDD   SREDE   SREDF   H99087A H99087B H99088
SRSEX;


/*  _O variables are the original values from the survey response */

%Let varlist2 =
H99001_O H99002_O H99003_O H99004_O H99005_O H99006_O H99007AO
H99007BO H99007CO H99007DO H99007EO H99007FO H99007GO H99007HO
H99007IO H99007JO H99007KO H99007LO H99008_O H99009AO H99009BO
H99010_O H99011_O H99012_O H99013_O H99014_O H99015_O H99017_O
H99018_O H99019AO H99019BO H99019CO H99020_O H99021AO H99021BO
H99022_O H99023_O H99024_O H99025_O H99026_O H99027_O H99028_O
H99029_O H99030_O H99031_O H99032_O H99033_O H99034_O H99035_O
H99036_O H99037_O H99038_O H99039AO H99039BO H99039CO H99039DO
H99039EO H99039FO H99039GO H99039HO H99039IO H99039JO H99040_O
H99041_O H99042_O H99043_O H99044_O H99045_O H99046_O H99047_O
H99048_O H99049_O H99050_O H99051_O H99052_O H99053_O H99054_O
H99055_O H99056_O H99057_O H99058_O H99059_O H99060_O H99061AO
H99061BO H99062_O H99063_O H99064_O H99065_O H99066_O H99067_O
H99068_O H99069_O H99070_O H99071_O H99072_O H99073_O H99074_O
H99075_O H99076AO H99076BO H99076CO H99076DO H99076EO H99076FO
H99076GO H99077_O H99078AO H99078BO H99079AO H99079BO H99080AO
H99080BO H99081_O H99082AO H99082BO H99082CO H99083_O H99084_O
SRRACEAO SRRACEBO SRRACECO SRRACEDO SRRACEEO SREDA_O  SREDB_O
SREDC_O  SREDD_O  SREDE_O  SREDF_O  H99087AO H99087BO H99088_O
SRSEX_O;


TITLE 'DoD 1999 Survey Form A';
TITLE2 'Apply Coding Scheme';
```

```
DATA HCDATA.&OUTDATA;

 %INCLUDE GETFMTS(FORMAT_NEW);

 /* label and format statements for original variables */

  SET HCDATA.POSTRATA;

  /* This is a version of the coding scheme and coding tables for the
     1999 HCSDB Form A.
     The following tables outline the coding of screening questions (skip),
     and subsequent items to be answered (or not answered in a series
     following a skip question.)*/

****DRC entered B for all surveys not returned we will skip **;
****around coding scheme for all surveys not returned   **;

 /*  First set up new variables that capture the original values */

  ARRAY RECODE &VARLIST1;
  ARRAY ORIG   &VARLIST2;

   DO I = 1 to DIM(ORIG);
    ORIG(I) = RECODE(I);
   END;
  DROP I;

 IF SCANDATE = .B AND H99001 = .B THEN GO TO NOSURVEY;


/* Note 1  H99001, H99002 -- H99088, person listed on mailing label */

   ARRAY MISVARS &VARLIST1;
    ARRAY NOTSURE H99003 H99005 H99006 H99038;

    N1MISCNT =0;
    N1NSURE = 0;
    N1BLANK = 0;

    DO OVER MISVARS;
       IF MISVARS NE . THEN N1MISCNT+1;
     END;

/** if question 1 is answered and all other responses are missing, **/
/** set n1misnct = 0 **/

  IF N1MISCNT=1 AND H99001 NE .N THEN N1MISCNT=0;

     DO OVER NOTSURE;
      IF NOTSURE NOT IN (2, 3) THEN N1NSURE+1;
     END;

  N1MISCNT=N1MISCNT-N1NSURE; /* don't count these responses  */
                             /* in case of not missing      */
  IF H99001 = 1 THEN N1=1;
  ELSE IF H99001 = 2 THEN DO;

    DO OVER MISVARS;  /* set all variables equal to valid skip */
      MISVARS = .N;
     END;
   N1=2;
    H99001 = 2; /* reset the response to question 1 */
  END;

IF N1 IN (., .A) THEN DO;

  IF N1NSURE + N1BLANK > 0 THEN DO;
   H99001=1;
    N1=3;
```

```
  END;

  ELSE IF N1 = . AND N1NSURE + N1BLANK =0 THEN N1=4;
  ELSE IF N1= .A AND N1NSURE + N1BLANK =0 THEN DO;

    DO OVER MISVARS;  /* set all variables equal to valid skip */
      MISVARS = .N;
     END;
  H99001 = 2;
 END;
END;

DROP N1NSURE N1BLANK N1MISCNT;

IF N1=2 THEN GOTO NOSURVEY;  /* skip the rest of the coding scheme */

/** Note2  -- H99004, H99005: Tricare prime coverage **/

  IF H99004 = 1 AND H99005 = . THEN DO;
   H99005 = .N;
    N2=1; END;
  ELSE IF H99004 = 1 AND H99005 NE . THEN DO;
   H99005 = .C;
    N2=2; END;
  ELSE IF H99004 IN (2, 3, 4) THEN N2=3;
  ELSE IF H99004 = . THEN N2 = 4;
  ELSE IF H99004 = .A AND H99005 NE . THEN N2=5;
  ELSE IF H99004 = .A AND H99005 = . THEN DO;
   H99004=1;
    H99005=.N;
    N2=6; END;

** Note 3 -- smoking H9912 - H9915;

   IF H99012=1 and H99013 IN (3,4) THEN DO;  /* still smoke */

    IF H99014 NE . THEN DO;
       H99014 =.C;
       N3=1; END;
    ELSE IF H99014 = . THEN DO;
      H99014 =.N;
       N3=2; END;
   END;

   ELSE IF H99012=1 AND H99013 = 2 THEN DO;    /* quit */

    IF H99014 IN (1,2) AND H99015 NE . THEN DO;   /* > 1 year ago */
     H99015 =.C;
     N3=3; END;
   ELSE IF H99014 IN (1,2) AND H99015 = . THEN DO;
    H99015 =.N;
    N3=4; END;
   ELSE IF H99014 IN (3, ., .A) THEN  /* < 1 year ago */
    N3=5;
   END;

   ELSE IF H99012=1 AND H99013 IN (1, ., .A) THEN DO;  /* don't know */

    IF H99014 = 2 AND H99015 NE . THEN DO;   /* > 1 year ago */
     H99013=2;
     H99015 =.C;
     N3=6; END;
   ELSE IF H99014 =2 AND H99015 = . THEN DO;
    H99013 =2;
    H99015 =.N;
    N3=7; END;
   ELSE IF H99014 = 3 THEN DO;            /* < 1 year ago */
    H99013 =2;
    N3=8; END;
    ELSE IF H99014 IN (1, ., .A) THEN N3=9;  /* don't know */
   END;
    ELSE IF H99012 IN (2,3,.,.A) AND H99013 IN (3,4) THEN DO; /*never smoke*/
    IF H99014 NE . THEN DO;
```

```
    H99012 =1;
    H99014 =.C;
    N3=10; END;
   ELSE IF H99014 = . THEN DO;
   H99012 =1;
   H99014 =.N;
   N3=11;
  END;
 END;

  ELSE IF H99012 IN (2,3) AND H99013 IN (2,1,.A) THEN DO; /*never smoke*/

   IF H99014 NE . AND H99015 NE . THEN DO;
    H99013 =.C;
    H99014 =.C;
    H99015 =.C;
    N3=12; END;
   ELSE IF H99014 NE . AND H99015=. THEN DO;
   H99013 =.C;
   H99014 =.C;
   H99015 =.N;
   N3=13; END;
   ELSE IF H99014 = . AND H99015 NE . THEN DO;
   H99013 =.C;
   H99014 =.N;
   H99015 =.C;
   N3=14; END;
   ELSE IF H99014 = . AND H99015=. THEN DO;
   H99013 =.C;
   H99014 =.N;
   H99015 =.N;
   N3=15; END;
 END;

 ELSE IF H99012 IN (2,3) AND H99013 = . THEN DO; /*never smoke*/

  IF H99014 NE . AND H99015 NE . THEN DO;
   H99013 =.N;
   H99014 =.C;
   H99015 =.C;
   N3=16; END;
 ELSE IF H99014 NE . AND H99015=. THEN DO;
   H99013 =.N;
   H99014 =.C;
   H99015 =.N;
   N3=17; END;
 ELSE IF H99014 = . AND H99015 NE . THEN DO;
   H99013 =.N;
   H99014 =.N;
   H99015 =.C;
   N3=18; END;
 ELSE IF H99014 = . AND H99015=. THEN DO;
   H99013 =.N;
   H99014 =.N;
   H99015 =.N;
   N3=19; END;
END;

 ELSE IF H99012 IN (.A, .) AND H99013 IN (2,.,.A) THEN DO; /*MRE/blank*/

  IF H99014 = 3 THEN DO;
   H99012 =1;
   N3=20; END;
 ELSE IF H99014 = 2 AND H99015 NE . THEN DO;
   H99012 =1;
   H99015 =.C;
   N3=21; END;
 ELSE IF H99014 = 2 AND H99015 = . THEN DO;
   H99012 =1;
   H99015 =.N;
   N3=22; END;
 ELSE IF H99014 = 1 AND H99015 = . THEN DO;
   H99015 =.N;
```

```
    N3=23; END;
  ELSE IF H99014 = 1 AND H99015 NE . THEN DO;
    H99015 =.C;
    N3=24; END;
  ELSE IF H99014 IN (.,.A) AND H99015 IN (2,3,4,5) THEN DO;
    H99012 =1;
    N3=25; END;
  ELSE IF H99014 IN (.,.A) AND H99015 IN (1,.,.A) THEN N3=26;
 END;
   ELSE IF H99012 IN (.A, .) AND H99013 = 1 THEN DO; /*MRE/blank*/

   IF H99014 NE . AND H99015 NE .  THEN DO;
    H99014 =.C;
    H99015 =.C;
    N3=27; END;
  ELSE IF H99014 NE . AND H99015 = .  THEN DO;
    H99014 =.C;
    H99015 =.N;
    N3=28; END;
  IF H99014 = . AND H99015 = .  THEN DO;
    H99014 =.N;
    H99015 =.N;
    N3=29; END;
  ELSE IF H99014 = . AND H99015 NE .  THEN DO;
    H99014 =.N;
    H99015 =.C;
    N3=30; END;
 END;

**************;
**** Note no self reported age in 1999 ;

/* Note 4 - gender SRSEX, SEX, H99017, H99018 -- H99021B, XSEXA */

  /* 1/21/98 use SRSEX & responses to gender specific questions
     if there is discrepancy between SRSEX and SEX */
  /* set imputed MALE, FMALE based on gender specific questions */

  IF H99017 > 0 THEN MALE=1;     /* prostate */
  ELSE MALE = 0;
  IF H99018 > 0 OR H99019A > 0 OR H99019B > 0 OR H99019C > 0 OR
     H99020 > 0 OR H99021A > 0 or H99021B > 0  THEN FMALE=1;
    /* mammogram/pap smear/PREGNANT*/

  ELSE FMALE = 0;

  IF SRSEX = . OR SRSEX = .A THEN DO;
   IF  (SEX = 'F' AND MALE AND FMALE) THEN DO;
    N4A = 1;
    XSEXA = 2; END;
  ELSE IF (SEX = 'F' AND MALE=0 AND FMALE=0) THEN DO;
   N4A = 2;
   XSEXA = 2; END;
  ELSE IF (SEX = 'M' AND MALE AND FMALE) THEN DO;
   N4A = 3;
   XSEXA = 1; END;
  ELSE IF (SEX = 'M' AND MALE=0 AND FMALE=0) THEN DO;
   N4A = 4;
   XSEXA = 1; END;
  ELSE IF MALE AND NOT FMALE THEN DO;
   N4A = 5;
   XSEXA = 1; END;
  ELSE IF FMALE AND NOT MALE THEN DO;
   N4A = 6;
   XSEXA = 2; END;
  ELSE IF (SEX = 'Z' AND MALE AND FMALE) THEN DO;
   N4A = 7;
   XSEXA = .; END;
  ELSE IF (SEX = 'Z' AND MALE=0 AND FMALE=0) THEN DO;
   N4A = 8;
   XSEXA = .; END;
  END;
  ELSE IF (SRSEX = 1 ) THEN DO;
```

```
   IF MALE AND NOT FMALE THEN DO;
    N4A = 9;
    XSEXA = 1; END;
  ELSE IF NOT MALE AND FMALE THEN DO;
   IF SEX = 'F' THEN DO;
    N4A = 10;
    XSEXA = 2; END;
  ELSE DO;
   N4A = 11;
   XSEXA = 1; END;
 END;
 ELSE IF MALE AND FMALE THEN DO;
  N4A = 12;
  XSEXA = 1; END;
 ELSE IF MALE=0 AND FMALE=0 THEN DO;
  N4A = 13;
  XSEXA = 1; END;
END;
 ELSE IF (SRSEX = 2) THEN DO;
  IF NOT MALE AND FMALE THEN DO;
   N4A = 14;
   XSEXA = 2; END;
 ELSE IF MALE AND NOT FMALE THEN DO;
  IF SEX = 'M' THEN DO;
   N4A = 15;
   XSEXA = 1; END;
 ELSE DO;
   N4A = 16;
   XSEXA = 2; END;
 END;
 ELSE IF MALE AND FMALE THEN DO;
  N4A = 17;
  XSEXA = 2; END;
 ELSE IF MALE=0 AND FMALE=0 THEN DO;
  N4A = 18;
  XSEXA = 2; END;
 END;

/* Note 4b - gender vs prostate */

 IF XSEXA=1 THEN N4B=1;     /* male */
  ELSE IF XSEXA=2 THEN DO;   /* female */
  IF H99017 NE . THEN DO;
   N4B =2;
   H99017=.C; END; /*inconsistent resp*/
  ELSE DO;
   N4B = 3;
   H99017= .N; END;      /* valid skip */
 END;
  ELSE IF XSEXA=. THEN DO;   /*  missing sex  */
   N4B =4;
   H99017=.; END;

/* Note 4c - gender vs mammogram/paps */
/*   REDEFINE FMALE TO LOOK ONLY AT MAMMOGRAM OR PAP SMEAR ENTRIES  */

  ARRAY NOTE4C H99018 H99019A H99019B H99019C H99020 H99021A H99021B;

  IF H99018 NE . OR H99019A NE . OR H99019B NE . OR H99019C NE . OR
     H99020 NE . OR H99021A NE . OR H99021B NE . THEN FMALE=1; /* mammogram or pap smear */
  ELSE FMALE = 0;
  IF XSEXA=1 THEN DO;   /* male */
   IF FMALE=1 THEN DO;
    N4C=1;
      DO OVER NOTE4C;
       IF NOTE4C = . THEN NOTE4C = .N;
        ELSE NOTE4C=.C;
      END;
   END; /* inconsistent response */
 ELSE IF FMALE=0 THEN DO;
  N4C=2;
   DO OVER NOTE4C;
     NOTE4C=.N;
```

```
    END;
  END;   /* valid skip */
 END;

 ELSE IF XSEXA=2 THEN  N4C=3; /* female */
 ELSE IF XSEXA=. THEN DO;     /*  missing sex  */
  N4C=4;
   DO OVER NOTE4C;
     NOTE4C=.;
    END;
 END;

DROP MALE FMALE;


/* Note 5 - breast exam for female 40 or over  */
/* Note 1999 -- no self reported age variable  */
/* Note no dob variable  -- macro not used     */

/***** review age variable  *******/

 IF XSEXA=1 THEN DO;  /* male */
  IF (H99019A = .C OR H99019A = .N)
   AND (H99019B = .C OR H99019B = .N)
   AND (H99019C = .C OR H99019C = .N)
    THEN N5 = 1;
 END;
 ELSE IF XSEXA=2 THEN DO;
  IF H99019A=2 THEN N5=2;         /* female 40 or over */
   ELSE IF H99019A = 1 THEN DO;   /* female < 40 */
     IF H99019B NE . THEN H99019B = .C;
     ELSE H99019B = .N;
     IF H99019C NE . THEN H99019C = .C;
     ELSE H99019C = .N;
     N5=3; END;
    ELSE IF H99019A = .A THEN DO;
     IF H99019B NE . OR H99019C NE . THEN DO;
      H99019A = 2;
      N5 =4; END;
    ELSE IF H99019B = . AND H99019C =  . THEN DO;
     H99019A = 1;
      IF H99019B NE . THEN H99019B = .N;
       ELSE H99019B = .C;
      IF H99019C NE . THEN H99019C = .N;
        ELSE H99019C = .C;
        N5=5;END;
  END;
  ELSE IF H99019A = . THEN DO;
   IF H99019B NE . OR H99019C NE . THEN DO;
     H99019A = 2;
     N5=6; END;
  ELSE IF H99019B = . AND H99019C = . THEN DO;
   IF AGE < 40 THEN DO;
     H99019A = 1;
    N5=7; END;
  ELSE IF AGE >= 40 THEN DO;
   H99019A = 2;
   N5=8; END;
  IF AGE = . THEN N5=9;
  END;
 END;
END;

 ELSE IF XSEXA = . THEN N5=10;


/* Note 6 - gender vs Pregnancy */

 IF XSEXA=1 THEN N6=1;    /* male */
  ELSE IF XSEXA=2 THEN DO;  /* female */
   IF H99020=1 THEN N6=2; /* pregnant */
    ELSE IF H99020=2 THEN DO;
    IF H99021A = . THEN H99021A = .N;
```

```
    ELSE H99021A = .C;
      N6=3; END;
    ELSE IF H99020 = 3 THEN DO;
    IF H99021A = . THEN H99021A = .N;
    ELSE H99021A = .C;
    IF H99021B = . THEN H99021B = .N;
    ELSE H99021B = .C;
     N6=4; END;
    ELSE IF H99020 IN (., .A) THEN DO;
     IF H99021A NE . THEN DO;
      H99020 = 1;
      N6=5; END;
     ELSE IF H99021A = . THEN DO;
      IF H99020 = . THEN N6=6;
      ELSE IF H99020 = .A THEN DO;
        H99020=3;
        H99021A = .N;
     IF H99021B = . THEN H99021B= .N;
     ELSE H99021B = .C;
      N6=7;
     END;
   END;
  END;
 END;
 ELSE IF XSEXA = . THEN N6=8;


/**** Note 7 -- H99022, H99023 -- Personal doctor or nurse **/

  IF H99022 = 1 AND H99023 IN (1, 2, 3, ., .A) THEN N7=1;
  ELSE IF H99022 IN (1, ., .A) AND H99023 IN (.N, -6) THEN DO;
   H99022=2;
   H99023=.C;
   N7=2;
  END;
  ELSE IF H99022 IN (2, ., .A) AND H99023 IN (1, 2, 3) THEN DO;
   H99022=1;
   N7=3;
  END;
  ELSE IF H99022=2 AND H99023 IN (., -6, .N, .A) THEN DO;
   IF H99023 = . THEN  H99023 = .N;
    ELSE H99023 = .C;
     N7=4;
  END;
  ELSE IF H99022 = . AND H99023 = .A THEN DO;
   H99022 = 1;
    N7=5;
  END;
  ELSE IF H99022 = . AND H99023 = . THEN N7=6;
  ELSE IF H99022 = .A AND H99023 = . THEN DO;
       H99022=2;
       H99023=.N;
       N7=7;
   END;

/******* Note 8 - Personal doctor or nurse H99024, H99025  ******/

*** note in test data, -6 was coded as .N  ;
*** check to ensure -6 and .N are equivalent ;

  IF H99024 = 1 AND H99025 NE .N THEN N8=1;
  ELSE IF H99024 IN (1, ., .A) AND H99025 = .N THEN DO;
   H99024=2;
   H99025=.C;
   N8=2; END;
  ELSE IF H99024 IN (2, ., .A) AND ( 0<=H99025<=10 OR H99025 = .A) THEN DO;
   H99024=1;
   N8=3; END;
  ELSE IF H99024 = 2 AND H99025 IN (.N, .) THEN DO;
   N8=4;
     IF H99025 = . THEN H99025 = .N;
      ELSE H99025=.C; END;
  ELSE IF H99024 = . AND H99025 = . THEN N8=5;
```

```
  ELSE IF H99024 = .A AND H99025 = . THEN DO;
   H99024=2;
   H99025=.N;
   N8=6; END;


/******* Note 9 - health care specialist H99026 -- H99031  ******/
/**** Changed 3/21/2000   *******/

 ARRAY NOTE9 H99027-H99031;
 ARRAY NOTE9B H99027 H99029 H99030 H99031;
  N9MARK=0;
  N9NOTNA=0;

 DO OVER NOTE9;
  IF NOTE9 NE . THEN N9MARK+1 ;
 END;

 DO OVER NOTE9B;
  IF NOTE9B NE .N THEN N9NOTNA+1;
 END;

 IF H99028 NOT IN (2, .) THEN N9NOTNA+1;

 IF H99026 = 1 AND (N9MARK = 0 OR (N9MARK>0 AND N9NOTNA>0))
  THEN N9=1;
 ELSE IF H99026 IN (1, ., .A) AND N9MARK>0 AND N9NOTNA=0
  THEN DO;
   H99026=2;
    N9=2;
     DO OVER NOTE9;
      IF NOTE9=. THEN NOTE9=.N;
       ELSE NOTE9=.C;
     END;
    END;
 ELSE IF H99026 IN (2, ., .A) AND N9MARK>0 AND N9NOTNA>0
  THEN DO;
   H99026=1;
    N9=3; END;
 ELSE IF H99026 = 2 AND (N9MARK = 0 OR (N9MARK>0 AND N9NOTNA=0))
  THEN DO;
   N9=4;
     DO OVER NOTE9;
      IF NOTE9=. THEN NOTE9=.N;
       ELSE NOTE9=.C;
     END;
    END;
  ELSE IF H99026 = . AND N9MARK=0 THEN N9=5;
  ELSE IF H99026 = .A AND N9MARK = 0 THEN DO;
   H99026=2;
    N9=6;
     DO OVER NOTE9;
      NOTE9 = .N;
     END;
  END;

 DROP N9MARK N9NOTNA;

/******* Note 10 - health care specialist H99028 -- H99031  ******/
/*** Note -- come back to when test data arrive  ***/;

 ARRAY NOTE10 H99029--H99031;

  N10NMISS=0;
  N10NOTNA=0;

  DO OVER NOTE10;
   IF NOTE10 NE . THEN N10NMISS+1;
    IF NOTE10 NOT IN (., .N) THEN N10NOTNA+1;
  END;

  IF H99028 IN (.N, .C) THEN N10=1;
  ELSE IF H99028 = 1 AND (N10NMISS=0 OR (N10NMISS>0 AND N10NOTNA>0))
```

```
   THEN N10=2;
  ELSE IF H99028 IN (1, .A, .) AND (N10NMISS>0 AND N10NOTNA=0)
   THEN DO;
     H99028 =2;
     N10=3;
      DO OVER NOTE10;
        IF NOTE10 = . THEN NOTE10 = .N;
          ELSE NOTE10 = .C;
      END;
    END;

  ELSE IF H99028 IN (2, ., .A) AND N10NMISS> 0 AND N10NOTNA>0
   THEN DO;
     H99028=1;
     N10=4;
  END;
  ELSE IF H99028 = 2 AND (N10NMISS=0 OR (N10NMISS>0 AND N10NOTNA=0))
   THEN DO;
     N10=5;
      DO OVER NOTE10;
       IF NOTE10=. THEN NOTE10=.N;
         ELSE NOTE10=.C;
      END;
     END;
    ELSE IF H99028 = . AND N10NMISS=0 THEN N10=6;
    ELSE IF H99028 = .A AND N10NMISS=0 THEN DO;
     H99028=2;
     N10=7;
      DO OVER NOTE10;
       NOTE10=.N;
      END;
   END;

DROP N10NMISS N10NOTNA;

/******* Note 11 - phone advice in past 12 months -- H99032--H99034  ******/

 IF H99032=1 AND H99033 NE .N THEN N11=1;
 ELSE IF H99032 IN (1, ., .A) AND H99033 =.N THEN DO;
    H99032=2;
    H99033=.C;
    N11=2; END;
 ELSE IF H99032 IN (2, ., .A) AND
  H99033 NOT IN (.N, .) THEN DO;
   H99032=1;
   N11=3; END;
 ELSE IF H99032 = 2 AND H99033 IN (.N, .) THEN DO;
  IF H99033=. THEN H99033=.N;
   ELSE H99033=.C;
  N11=4; END;
 ELSE IF H99032=. AND H99033=. THEN N11=5;
 ELSE IF H99032=.A AND H99033=. THEN DO;
  H99032=2;
  H99033=.N;
  N11=6; END;

/* Note 12 - seek care in past 12 months --  */

 IF H99034=1 AND H99035 NE .N THEN N12=1;
 ELSE IF H99034 IN (1, ., .A) AND H99035 = .N THEN DO;
  H99034=2;
  H99035=.C;
  N12=2; END;
 ELSE IF H99034 IN (2, ., .A) AND
  H99035 NOT IN (.N, .) THEN DO;
  H99034=1;
  N12=4; END;
 ELSE IF H99034 = 2 AND H99035 IN (.N, .) THEN DO;
  IF H99035 = . THEN H99035=.N;
   ELSE H99035 = .C;
  N12=5; END;
 ELSE IF H99034=. AND H99035=. THEN N12=5;
 ELSE IF H99034=.A AND H99035=. THEN DO;
```

```
  H99034=2;
  H99035=.N;
  N12=6; END;

 /* Note 13 - health care in past 12 months -- H99037 -- H99056 */


  ARRAY NOTE13A H99039A--H99039I;
  ARRAY NOTE13B H99040 H99041;
  ARRAY NOTE13  H99038--H99056;
  MARKED=0;

  DO OVER NOTE13A;
    IF NOTE13A = 2 THEN NOTE13A=.L;
  END;
  DO OVER NOTE13B;
    IF NOTE13B = 1  THEN NOTE13B=.L;
  END;

 IF H99039J = 1 THEN H99039J = .L;
   ELSE IF H99039J = 2 THEN H99039J = .K;


  DO OVER NOTE13;
    IF NOTE13 NE . AND NOTE13 NE .N AND NOTE13 NE .L AND NOTE13 NE .K
      THEN MARKED+1;
  END;

  /* reset .L to numeric valid value */

  IF H99039J = .L THEN H99039J = 1;
    ELSE IF H99039J = .K THEN H99039J = 2;


  DO OVER NOTE13A;
    IF NOTE13A = .L THEN NOTE13A=2;
  END;
  DO OVER NOTE13B;
    IF NOTE13B = .L THEN NOTE13B=1;
  END;
 IF H99037 IN (1,2) THEN N13=1; /* military or civilian */
 ELSE IF H99037=3 THEN DO;   /* neither */
  IF MARKED > 0  THEN DO;
   N13=2;
   DO OVER NOTE13;
    IF NOTE13 IN (., 2)  THEN NOTE13 = .N;
     ELSE NOTE13 = .C;
   END;
  END;
 ELSE DO;
  N13=3;
   DO OVER NOTE13;
    NOTE13=.N;
   END;
 END;
END;

 ELSE IF H99037=. OR H99037=.A THEN DO; /* missing or multiple responses*/
  IF MARKED = 0 THEN DO;
   DO OVER NOTE13;
    NOTE13=.N;
   END;
  H99037=3;
  N13=4; END;
 ELSE N13=5;
END;

DROP MARKED;

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

/* Note 14 - emergency room in past 12 months -- H99038, H99039A--H9939J */
```

```
ARRAY NOTE14 H99039A--H99039I;
   NSKIP=0;
   MARK=0;
   CCNT=0;
  DO OVER NOTE14;
   IF NOTE14=.N THEN NSKIP+1;
   IF NOTE14=2 THEN MARK+1;
   IF NOTE14=.C THEN CCNT+1;
  END;

  IF H99038 IN (.C, .N)  AND CCNT + NSKIP = 9 THEN N14 = 1;

  ELSE IF H99038 = 1 THEN DO;
    IF MARK > 0 THEN DO;
    IF H99039J = 1 THEN H99039J= .C;
     ELSE H99039J = .N;
    N14=2;
   END;
  ELSE IF MARK=0 AND H99039J IN (2, .) THEN N14 = 3;
 END;
 ELSE IF H99038 IN (1, 3, ., .A) AND MARK=0 AND H99039J =1 THEN DO;
  DO OVER NOTE14;
   NOTE14=.N;
  END;
   H99039J=.C;
   N14=4;
 END;
 ELSE IF H99038 IN (2, 3) AND MARK=0 THEN DO;
  DO OVER NOTE14;
   NOTE14=.N;
  END;
   H99039J=.C;
   N14=5;
 END;
 ELSE IF H99038 IN (2, 3, ., .A) THEN DO;
  IF MARK > 0 AND H99039J IN (2, .) THEN DO;
   H99038 = 1;
   H99039J = .N;
   N14=6;
  END;
  ELSE IF H99039J = 1 THEN DO;
   DO OVER NOTE14;
    IF NOTE14=1 THEN NOTE14 = .C;
     ELSE NOTE14=.N;
   END;
    H99039J =.C;
    N14=7;
   END;
 END;

 DROP NSKIP MARK CCNT;

/* Note 15 - submit claims in past 12 months  H99057 -- H99059 */

  ARRAY NOTE15 H99058 H99059;
  NOTMISS=0;
  NOTNA=0;
  NOTMISNA=0;
  DO OVER NOTE15;
    IF NOTE15 NE . THEN NOTMISS+1;    /* check for all missing */
    IF NOTE15 NE .N THEN NOTNA+1;     /* check for all NA */
    IF NOTE15 NOT IN (., .N) THEN NOTMISNA+1; /* missing or NA */
    END;
  /*  NOTMISS=0  =>  ALL ARE BLANK
      NOTNA=0    =>  ALL ARE N/A
      NOTMISNA=0 =>  ALL ARE BLANK OR N/A
      NOTMISNA>0 =>  AT LEAST ONE IS MARKED   */

  IF H99057=1 AND (NOTMISS = 0 OR (NOTMISS>0 AND NOTMISNA > 0))
   THEN N15=1;
  ELSE IF H99057 IN (1, 3, ., .A) AND NOTNA = 0 THEN DO;   /* all NA */
   H99057=2;
    N15=2;
```

```
    DO OVER NOTE15;
      NOTE15 = .C;
     END;
   END;
  ELSE IF H99057 IN (1, ., .A) AND
    NOTMISNA=0 AND NOTMISS > 0 AND NOTNA>0 THEN DO;  /* missing or NA */
     H99057=2;
     N15=3;
       DO OVER NOTE15;
         IF NOTE15=. THEN NOTE15 = .N;
          ELSE NOTE15 = .C;
      END;
   END;
  ELSE IF H99057 IN (2, 3, . .A) AND
    NOTMISNA > 0 THEN DO;        /*   at least one is marked   */
     H99057=1;
     N15=4; END;
  ELSE IF H99057 IN (2, 3) AND
    NOTMISS=0 OR (NOTMISNA=0 AND NOTMISS > 0 AND NOTNA > 0) THEN DO;
      N15=5;     /*miss or NA*/
       DO OVER NOTE15;
         IF NOTE15 = . THEN NOTE15=.N;
          ELSE NOTE15 =.C;
         END;
    END;
  ELSE IF H99057=2 AND NOTNA = 0 THEN DO;   /* all NA */
   N15=6;
     DO OVER NOTE15;
      NOTE15=.C;
     END;
  END;
  ELSE IF H99057=. AND NOTMISS=0 THEN  /* all missing */
   N15=7;
  ELSE IF H99057=.A AND NOTMISS=0 THEN DO; /* all missing */
   H99057=2;
    N15=8;
     DO OVER NOTE15;
      NOTE15 = .N;
     END;
  END;

 DROP NOTMISS NOTNA NOTMISNA;

/* Note 16 - out of pocket money  H99061A, H99061B */

  IF H99061A=1 THEN DO;
   IF H99061B > 0 OR H99061B IN (.A, .I) THEN DO;
    H99061A=2;
    N16=1; END;
   ELSE IF H99061B = 0 THEN DO;
    H99061B=.C;
    N16=2; END;
   ELSE IF H99061B = . THEN DO;
    H99061B=.N;
    N16=3; END;
  END;
  ELSE IF H99061A=2 THEN DO;
   IF H99061B > 0 OR H99061B IN (., .A, .I) THEN N16=4;
    ELSE IF H99061B = 0 THEN DO;
     H99061A=1;
     H99061B=.C;
     N16=5; END;
  END;

/* Note 17 - look for written information  H99062, H99063 */

  IF H99062 = 1 AND H99063 NE .N THEN N17=1;
  ELSE IF H99062 IN (1, ., .A) AND H99063 = .N THEN DO;
   H99062=2;
   H99063=.C;
   N17=2; END;
  ELSE IF H99062 IN (2, ., .A) AND H99063 NOT IN (.N,  .) THEN DO;
   H99062=1;
```

```
   N17=3; END;
  ELSE IF H99062 = 2 AND H99063 IN (.N, .) THEN DO;
   IF H99063= . THEN H99063=.N;
     ELSE H99063 = .C;
   N17=4; END;
  ELSE IF H99062 = . AND H99063 = . THEN N17=5;
  ELSE IF H99062 = .A AND H99063 = . THEN DO;
   H99062=2;
   H99063=.N;
   N17=6; END;

/* Note 18 - look for written information  H99064, H99065 */

  IF H99064 = 1 AND H99065 NE .N THEN N18=1;
  ELSE IF H99064 IN (1, ., .A) AND H99065 = .N THEN DO;
   H99064=2;
   H99065=.C;
   N18=2; END;
  ELSE IF H99064 IN (2, ., .A) AND H99065 NOT IN (., .N) THEN DO;
   H99064=1;
   N18=3; END;
  ELSE IF H99064 = 2 AND H99065 IN (., .N) THEN DO;
   IF H99065=. THEN H99065 =.N;
     ELSE H99065=.C;
   N18=4; END;
  ELSE IF H99064 = . AND H99065 = . THEN N18=5;
  ELSE IF H99064 = .A AND H99065 = . THEN DO;
   H99064=2;
   H99065=.N;
   N18=6; END;

/* Note 19 - paper work in past 12 months -- H99066, H99067*/

  IF H99066 = 1 AND H99067 NE .N THEN N19=1;
  ELSE IF H99066 IN (1, ., .A) AND H99067 = .N THEN DO;
   H99066=2;
   H99067=.C;
   N19=2; END;
  ELSE IF H99066 IN (2, ., .A) AND H99067 NOT IN (., .N) THEN DO;
   H99066=1;
   N19=3; END;
  ELSE IF H99066 = 2 AND H99067 IN (., .N)  THEN DO;
    IF H99067=. THEN H99067 =.N;
      ELSE H99067 = .C;
    N19=4; END;
  ELSE IF H99066 = . AND H99067 = . THEN N19=5;
  ELSE IF H99066 = .A AND H99067 = . THEN DO;
   H99066=2;
   H99067=.N;
   N19=6; END;

/* Note 20 - filed complaint in past 12 months H99068 -- H99070*/

  ARRAY NOTE20 H99069 H99070;
   NOTMISS=0;
   NOTMISNA=0;
  DO OVER NOTE20;
    IF NOTE20 NE . THEN NOTMISS+1;    /* check for all missing */
    IF NOTE20 NE . AND NOTE20 NE .N THEN NOTMISNA+1; /* missing or NA */
    END;
  /*  NOTMISS=0  =>  ALL ARE BLANK
      NOTMISNA=0 =>  ALL ARE BLANK OR N/A
      NOTMISNA>0 =>  AT LEAST ONE IS MARKED   */

  IF H99068=1 AND (NOTMISNA > 0 OR NOTMISS = 0) THEN N20=1;
  ELSE IF H99068 IN (1, ., .A) AND NOTMISNA = 0 AND NOTMISS > 0 THEN DO;
   H99068=2;
   N20=2;
     DO OVER NOTE20;
      IF NOTE20=. THEN NOTE20=.N;
       ELSE NOTE20=.C;
     END;
  END;
```

```
  ELSE IF H99068 IN (2, ., .A) AND NOTMISNA > 0 THEN DO;
   H99068=1;
   N20=3; END;
  ELSE IF H99068 = 2 AND (NOTMISS=0 OR NOTMISNA=0) THEN DO; /*miss or NA*/
    N20=4;
      DO OVER NOTE20;
       IF NOTE20=. THEN NOTE20=.N;
         ELSE NOTE20 =.C;
      END;
   END;
  ELSE IF H99068=. AND NOTMISS=0 THEN  /* all missing */
   N20=5;
  ELSE IF H99068=.A AND NOTMISS=0 THEN DO; /* all missing */
   H99068=2;
   N20=6;
    DO OVER NOTE20;
      IF NOTE20=. THEN NOTE20=.N;
       ELSE NOTE20 =.C;
     END;
    END;
  DROP NOTMISS NOTMISNA;

/*   Note 21  - education.    */

  ARRAY NOTE21 SREDF SREDE SREDD SREDC SREDB SREDA;
  DO IDX = 1 TO 6;
      IF NOTE21(IDX) = 1 THEN GO TO HIGH;
  END;
  SREDHIGH = .;
  GO TO NOTE22;
HIGH:
      SREDHIGH = 7 - IDX;
  DROP IDX;

NOTE22:

/*  Note 22 work days missed -- H99087A, H99087B */

  IF H99087A = 1 THEN DO;
   IF H99087B >=1 OR H99087B IN (.A, .I) THEN DO;
    H99087A=2;
    N22=1; END;
   ELSE IF H99087B = 0 THEN DO;
    H99087B = .C;
    N22=2; END;
   ELSE IF H99087B=. THEN DO;
    H99087B=.N;
    N22=3; END;
   END;
   ELSE IF H99087A=2 THEN DO;
    IF H99087B >=1 OR H99087B IN (.A, .I, .) THEN N22=4;
    ELSE IF H99087B = 0 THEN DO;
     H99087A = 1;
     H99087B = .C;
     N22=5; END;
   END;

/* Note 23 -- enrollment in Tricare Prime H99072, H99073 */

  IF H99072 >=1 OR H99072 =.D THEN DO;
   IF H99073 NE .N THEN DO;
    H99073 = .N;
    N23=1;
   END;
    ELSE IF H99073 = .N THEN N23=2;
  END;
  ELSE IF H99072 = .N THEN DO;
   IF H99073 >=1 OR H99073 IN (., .A, .D) THEN N23=3;
   ELSE IF H99073 = .N THEN DO;
    H99073=5;
    N23=4; END;
  END;
  ELSE IF H99072 IN (., .A) THEN DO;
```

```
   IF H99073 >=1 OR H99073 = .D THEN DO;
    H99072 = .N;
    N23 = 5; END;
  ELSE IF H99073 = .N THEN DO;
     H99073 = .D;
     N23 = 6; END;
  ELSE IF H99073 IN (., .A) THEN N23 = 7;
 END;

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

NOSURVEY:
  /* missing values */
  ARRAY MISS MISS_9 MISS_8 MISS_7 MISS_6 MISS_5 MISS_4 MISS_1 ;
  MISS_TOT = 0;
  DO OVER MISS;
    MISS = 0;
  END;
  ARRAY MISSARAY &VARLIST2;

  DO OVER MISSARAY;
    IF (MISSARAY EQ . ) THEN MISS_9 = MISS_9 + 1;
    ELSE IF (MISSARAY EQ .A) THEN MISS_8 = MISS_8 + 1;
    ELSE IF (MISSARAY EQ .O) THEN MISS_7 = MISS_7 + 1;
    ELSE IF (MISSARAY EQ .N) THEN MISS_6 = MISS_6 + 1;
    ELSE IF (MISSARAY EQ .D) THEN MISS_5 = MISS_5 + 1;
    ELSE IF (MISSARAY EQ .I) THEN MISS_4 = MISS_4 + 1;
    ELSE IF (MISSARAY EQ .C) THEN MISS_1 = MISS_1 + 1;
  END;
  DO OVER MISS;
    MISS_TOT=MISS_TOT + MISS;
  END;

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

 OUTPUT;
RUN;

PROC CONTENTS data=HCDATA.&OUTDATA;
  RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
 TABLES N2 N3 N4A N4B N4C N5-N20
        N22-N23
  / MISSING LIST;
RUN;

PROC FORMAT;
 VALUE GRID
  0 = '0'
  1-9999 = '>=1' ;
 VALUE $GRIDB
  1-5 = '1-5' ;
 VALUE $AGE
  018-039 = '<40'
  040-120 = '>=40';
 VALUE SCALE
  0-10 = '0-10';
 VALUE MARK
  1-6 = 'Marked' ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
 TITLE3 'Frequency Checks - Note 2';
 TABLES N2*H99004_O*H99005_O*H99004*H99005
 /MISSING LIST;
FORMAT _ALL_ ;
FORMAT H99004_O H99004 GRID. ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 3';
TABLES N3*H99013_O*H99014_O*H99015_O*H99013*H99014*H99015
```

```
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 4A';
TABLES N4A*SRSEX*SEX*XSEXA
   / MISSING LIST;
FORMAT _ALL_ ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 4B';
TABLES N4B*XSEXA*H99017_O*H99017
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 4C';
TABLES N4C*XSEXA*H99018_O*H99019AO*H99019BO*H99019CO*H99018
       N4C*XSEXA*H99018_O*H99020_O*H99021AO*H99021BO*H99018

   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 5';
TABLES N5*XSEXA*AGE*H99019AO*H99019BO*H99019CO
   / MISSING LIST;
FORMAT _ALL_;
FORMAT AGE $AGE.  ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 6';
TABLES N6*XSEXA*H99020_O*H99021AO*H99021BO
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 7';
 TABLES N7*H99022_O*H99023_O*H99022*H99023
 /MISSING LIST;
FORMAT _ALL_;
 FORMAT H99023_O H99023 GRID. ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 8';
TABLES N8*H99024_O*H99025_O*H99024*H99025
   / MISSING LIST;
FORMAT H99025_O H99025 SCALE. H99024_O H99024 ;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 9';
TABLES N9*H99026_O*H99027_O*H99026*H99027
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 10';
TABLES N10*H99028_O*H99029_O*H99030_O*H99031_O*H99028*H99029*H99030*H99031
   / MISSING LIST;
FORMAT _ALL_;
FORMAT H99030_O H99030 SCALE. H99029 H99029_O GRID.;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
```

```
TITLE3 'Frequency Checks - Note 11';
TABLES N11*H99032_O*H99033_O*H99032*H99033
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 12';
TABLES N12*H99034_O*H99035_O*H99034*H99035
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 13';
TABLES N13*H99037_O*H99037
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 14';
TABLES N14*H99038_O*H99039AO*H99039A*H99039JO*H99038*H99039J
 N14*H99038_O*H99039BO*H99039B*H99039JO*H99038*H99039J
 N14*H99038_O*H99039CO*H99039C*H99039JO*H99038*H99039J
 N14*H99038_O*H99039DO*H99039D*H99039JO*H99038*H99039J
 N14*H99038_O*H99039EO*H99039E*H99039JO*H99038*H99039J
 N14*H99038_O*H99039FO*H99039F*H99039JO*H99038*H99039J
 N14*H99038_O*H99039GO*H99039G*H99039JO*H99038*H99039J
 N14*H99038_O*H99039HO*H99039H*H99039JO*H99038*H99039J
 N14*H99038_O*H99039IO*H99039I*H99039JO*H99038*H99039J
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 15';
TABLES N15*H99057_O*H99058_O*H99059_O*H99060_O*H99057
   / MISSING LIST;
FORMAT _ALL_;
FORMAT H99058_O H99059_O H99060_O MARK.;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 16';
TABLES N16*H99061AO*H99061BO*H99061A*H99061B
   / MISSING LIST;
FORMAT _ALL_;
FORMAT H99061BO H99061B GRID.;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 17';
TABLES N17*H99062_O*H99063_O*H99062*H99063
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 18';
TABLES N18*H99064_O*H99065_O*H99064*H99065
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 19';
TABLES N19*H99066_O*H99067_O*H99066*H99067
   / MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
```

```
TITLE3 'Frequency Checks - Note 20';
TABLES N20*H99068_O*H99069_O*H99070_O*H99068
   / MISSING LIST;
FORMAT _ALL_;
FORMAT H99069_O H99070_O MARK.;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 21';
TABLES SREDHIGH*SREDF*SREDE*SREDD*SREDC*SREDB*SREDA
/ MISSING LIST;
FORMAT _ALL_;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 22';
 TABLES N22*H99087AO*H99087BO*H99087A*H99087B
 /MISSING LIST;
FORMAT _ALL_ ;
FORMAT H99087B H99087BO GRID.;
RUN;

PROC FREQ DATA=HCDATA.&OUTDATA;
TITLE3 'Frequency Checks - Note 23';
TABLES N23*H99072_O*H99073_O*H99072*H99073
 /MISSING LIST;
FORMAT H99072_O H99073_O H99072 H99073 GRID.;
RUN;

PROC MEANS DATA=HCDATA.&OUTDATA N NMISS MIN MAX SUM MEAN;
TITLE3 'Frequency Checks - Missing Value Totals';
  VAR MISS_TOT MISS_1 MISS_4 MISS_5 MISS_6-MISS_9;
RUN;
```

3. CREATE STATUS FLAG FOR RECORD SELECTION

```
*************************************************************************
*
* PROGRAM: SELECTA.SAS
* TASK:    DOD HEALTH CARE SURVEY ANALYSIS (8676-200)
* PURPOSE: ASSIGN FINAL STATUS FOR RECORD SELECTION PURPOSES.
* WRITTEN: 11/09/99 BY KEITH RATHBUN
*
* INPUTS:  1) CSCHM99A.SD2 - 1999 DOD Health Survey Data
*
* OUTPUTS: 1) SELECTA.SD2 - 1999 DOD Health Survey Data w/FNSTATUS
*
*************************************************************************
*;
LIBNAME IN      "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT     "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY "D:\KEITH\DOD99-8676\DATA\FMTLIBA";

OPTIONS PS=79 LS=132 COMPRESS=YES NOCENTER;

PROC SORT DATA=IN.CSCHM99A OUT=TEMPA1; BY MPRID; RUN;

DATA TEMPA2 OUT.DUPSA;
   SET TEMPA1;
   BY MPRID;
   /************************************************************************/
   /** KEY VARIABLES (Total=29)                                           **/
   /************************************************************************/
   ARRAY KEYVAR H99003 H99004 H99005 H99022 H99023 H99024 H99025 H99026
                H99028 H99030 H99032 H99033 H99034 H99035 H99036 H99038
                H99040 H99041 H99044 H99055 H99062 H99064 H99066 H99071
                H99077 H99084
                 ;
   ARRAY Q39(6) H99039A H99039B H99039C H99039D H99039E H99039F;
   ARRAY Q85(5) SRRACEA SRRACEB SRRACEC SRRACED SRRACEE;
   ARRAY Q86(6) SREDA   SREDB   SREDC   SREDD   SREDE   SREDF;

   FLAG39 = 0; DROP FLAG39;
   DO I = 1 TO DIM(Q39);
      IF Q39(I) NOT IN (.,.A,.O,.I,.B) THEN FLAG39 = 1;
   END;

   FLAG85 = 0; DROP FLAG85;
   DO I = 1 TO DIM(Q85);
      IF Q85(I) NOT IN (.,.A,.O,.I,.B) THEN FLAG85 = 1;
   END;

   FLAG86 = 0; DROP FLAG86;
   DO I = 1 TO DIM(Q86);
      IF Q86(I) NOT IN (.,.A,.O,.I,.B) THEN FLAG86 = 1;
   END;

   KEYCOUNT = 0;
   DO I = 1 TO DIM(KEYVAR); DROP I;
      IF KEYVAR(I) NOT IN (.,.A,.O,.I,.B) THEN KEYCOUNT = KEYCOUNT + 1;
   END;
   KEYCOUNT = KEYCOUNT + FLAG39 + FLAG85 + FLAG86;

   /************************************************************************/
   /** SET FLAG FOR DUPLICATES                                            **/
   /************************************************************************/
   LENGTH DUPFLAG $3;
   DUPFLAG = 'NO';
   IF NOT (FIRST.MPRID AND LAST.MPRID) THEN DUPFLAG = 'YES';

   /************************************************************************/
   /** DETERMINE FNSTATUS                                                 **/
   /************************************************************************/
   FNSTATUS = 0;
   IF FLAG_FIN = 1 THEN DO;
      ************************************************************
      **** APPLY THE COMPLETE QUESTIONNAIRE RULE (50% OF KEY   ****
      **** VARIABLES).                                          ****
```

```
      ************************************************************;
      IF KEYCOUNT GT 14 THEN FNSTATUS = 11;
      ELSE FNSTATUS = 12;
   END;
   ELSE IF FLAG_FIN IN(3,6,8,10,11,14,16,21) THEN DO;
      FNSTATUS = 20;
   END;
   ELSE IF FLAG_FIN IN(2,4,5,7,12,13,15) THEN DO;
      FNSTATUS = 30;
   END;
   ELSE IF FLAG_FIN IN(9,17,18,19,20,22) THEN DO;
      IF FLAG_FIN IN (18,19,20) THEN DO;
         FNSTATUS = 42;
      END;
      ELSE DO;
         FNSTATUS = 41;
      END;
   END;

   IF DUPFLAG = 'YES' THEN OUTPUT OUT.DUPSA;
   ELSE OUTPUT TEMPA2;

RUN;

***********************************************************************
* Select the "most complete" questionaire from duplicates and
* SET it back into the non-duplicates file. For now assume the lowest
* FNSTATUS Value is the "most complete".
***********************************************************************
;
PROC SORT DATA=OUT.DUPSA;
BY MPRID FNSTATUS;
RUN;

DATA DEDUPED;
   SET OUT.DUPSA;
   BY MPRID FNSTATUS;
   IF FIRST.MPRID; *KEEP only the first - most complete questionaire;
RUN;

DATA OUT.SELECTA;
   SET TEMPA2 DEDUPED;
   LABEL FNSTATUS = "Final Status"
         DUPFLAG  = "Multiple Response Indicator"
         STRATUM  = "Sampling STRATUM"
         KEYCOUNT = "# Key Questions Answered (Out of 29)"
        ;
RUN;

TITLE1 "1999 DOD Health Survey FNSTATUS assignment (8676-200)";
TITLE2 "Program Name: SELECTA.SAS By Keith Rathbun";
TITLE3 "Program Output: SELECTA.SD2";

PROC CONTENTS DATA=OUT.SELECTA; RUN;

PROC FREQ DATA=OUT.SELECTA;
TABLES FNSTATUS KEYCOUNT FLAG_FIN
       FNSTATUS*KEYCOUNT*FLAG_FIN
   /MISSING LIST;
RUN;
```

4. FINAL RECORD SELECTION AND REMOVE CONFIDENTIAL DATA

```
*****************************************************************************
*
*  PROJECT:  1999 DOD Health Care Survey (8687-100)
*  PROGRAM:  SANITIZA.SAS
*  PURPOSE:  Create a file containing all records selected for analysis
*            (completed questionaire = FNSTATUS = 11) and delete
*            confidential fields from the file.
*
*    INPUT:  1) SELECTA.SD2 - All sampled adults - Sampling, post-
*               stratification, and item response variables plus recodes.
*
*   OUTPUT:  2) SANITIZA.SD2 - Same as above except with confidential
*               fields removed and only with FNSTATUS = 11
*
* SEQUENCE:  After record selection but before the constructed variables
*
*    NOTES:  1) Drop all sensitive and all unnecessary variables.  Keep
*               only the observations with FNSTATUS = 11.
*
*****************************************************************************;
LIBNAME        IN 'D:\KEITH\DOD99-8676\DATA\';
LIBNAME       OUT 'D:\KEITH\DOD99-8676\DATA';
LIBNAME   LIBRARY 'D:\KEITH\DOD99-8676\DATA\FMTLIBA';

OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

DATA OUT.SANITIZA;
   SET IN.SELECTA (DROP= ADISPO
                         AGE
                         AGESMPL
                         ARVDATE
                         ARVDAY
                         ARVMO
                         ARVYEAR
                         BATCH
                         BFGROUP
                         BGCSMPL
                         DUPRET
                         DUPRET2
                         ELGCDE
                         ENGROUP
                         ENLSMPL
                         ENR_DMIS
                         FLAG_DUP
                         GEOCELL
                         INRECNO
                         LITHO
                         MAILING
                         MAILTYP
                         MISC
                         PCMCODE
                         PRVCDE
                         RADRDMIS
                         RADRREG
                         RADRSTE
                         RADRZIP
                         RCLINIC
                         RECTYP
                         RETCOUNT
                         RETPROC
                         SADRREG
                         SADRSTE
                         SADRZIP
                         SCANDATE
                         SERIAL
                         STATUS
                         STESMPL
                         TSTBATCH
                         UADRREG
                         UADRSTE
                         UADRZIP
                         UCLINIC
```

```
                        UDMIS
                        ZDZIP
                        ZRZIP);
   WHERE FNSTATUS=11;
RUN;

TITLE1 "1999 DOD Health Survey Post-Stratification (8687-100)";
TITLE2 "Program Name: SANITIZA.SAS";
TITLE3 "Program Outputs: SANITIZA.SD2";
TITLE4 "Preliminary 1999 Adult Data - Sanitized";

PROC CONTENTS DATA=OUT.SANITIZA POSITION; RUN;
```

5. CONSTRUCTED VARIABLES FOR ANALYSIS

```
*******************************************************************************
* PROGRAM:   CONSVAR2.SAS
* WRITTEN:   2/3/99 BY KELLY WHITE
* UPDATED:   2/29/2000 BY NATALIE JUSTH
* PURPOSE:   TO CREATE 5 INDEPENDENT VARIABLES: XENRLLMT, XENR_PCM, XINS_COV
*            XQENROLL,  XBNFGRP
*            1 INDEPENDENT VARIABLE ALREADY CREATED FROM DEERS-BFGROUPP
*            TO CREATE 24 DEPENDENT VARIABLES: KENRINTN, KDISENRL, KBGPRB1,
*            KBGPRB2, KMILWAT1, KCIVWAT1, KMILOFFC, KCIVOFFC, KMILOP99,
*            KCIVOP99, HP_PRNTL, HP_MAMOG, HP_PAP,
*            HP_BP, HP_FLU, HP_PROS
* INPUT:     J:\8687\DATA1999\AFINAL\SANITIZA.SD2
* OUTPUT:    J:\8687\DATA1999\AFINAL\CONSVAR2.SD2
*******************************************************************************
*;

LIBNAME IN 'J:\8687\DATA1999\AFINAL';
LIBNAME IN4'J:\8574\DATA1998\FINAL';
LIBNAME LIBRARY 'J:\8687\data1999\AFINAL\fmtlib';
OPTIONS PS=79 LS=132 ERRORS=2 CENTER ;
TITLE1 '1999 Health Care Survey of DoD Beneficiaries Study - Adult Form A';
TITLE2 'CREATE CONSTRUCTED & OUTCOME MEASURE VARIABLES';

DATA IN.CONVARA2 (KEEP = XENRLLMT XENR_PCM XINS_COV XQENROLL XREGION ENGROUPP
                         BFGROUPP XBNFGRP KENRINTN KDISENRL
                         KMILWAT1 KCIVWAT1 KMILOFFC
                         KCIVOFFC KBGPRB1 KBGPRB2
                         KMILOP99 KCIVOP99
                         HP_PRNTL HP_MAMOG HP_PAP HP_BP HP_FLU HP_PROS MPRID)
       CONSVAR2;
   SET IN.SANITIZA;


  LABEL
       XENRLLMT    =  "Enrollment in TRICARE Prime"
       XENR_PCM    =  "Enrollment by PCM type"
       XINS_COV    =  "Insurance Coverage"
       XQENROLL    =  "Enrllmnt accordng to questionre rspnse"
       XREGION     =  "Health Care regions "
       XBNFGRP     =  "Constructed Beneficiary Group"
       KENRINTN    =  "Intention to enroll         "
       KDISENRL    =  "Intention to disenroll      "
       KMILWAT1    =  "Wait <4 wks for well patient visit-Mil"
       KCIVWAT1    =  "Wait <4 wks for well patient visit-Civ"
       KMILOFFC    =  "Office wait of 30 min or more-Mil"
       KCIVOFFC    =  "Office wait of 30 min or more-Civ"
       KBGPRB1     =  "Big problem getting referrals to spclst"
       KBGPRB2     =  "Big problem getting neccessary care"
       KMILOP99    =  "Outpatient visits to Military facility"
       KCIVOP99    =  "Outpatient visits to Civilian facility"
       HP_PRNTL    =  "Prgnt in lst yr, received cre 1st trimstr"
       HP_MAMOG    =  "Women 50>=, mammography in pst 2 yrs"
       HP_PAP      =  "All women, pap smear in lst 3 yrs"
       HP_BP       =  "Bld prsre chck in lst 2 yrs,know rslts"
       HP_FLU      =  "65 and older, flu shot in lst 12 mnths"
       HP_PROS     =  "Men 50>=, prostrate exam in lst 12 mnths" ;


  FORMAT
        ENGROUPP       ENLSMP.
        XENRLLMT       ENROLL.
        XENR_PCM       PCM.
        XINS_COV       INSURE.
        XQENROLL       PCM.
        XREGION        CREG.
        XBNFGRP        XBGC_S.
        KMIL99         HAYNN.
        KCIV99         HAYNN.
        KENRINTN       HAYNN.
        KDISENRL       HAYNN.
        KMILWAT1       HAYNN.
```

```
        KCIVWAT1       HAYNN.
        KMILOFFC       HAYNN.
        KCIVOFFC       HAYNN.
        KBGPRB1        HAYNN.
        KBGPRB2        HAYNN.
        KMILOP99       HAGRID.
        KCIVOP99       HAGRID.
        KPRSCPTN       HAYNN.
        HP_PRNTL       PRNTL.
        HP_MAMOG       HAYNN.
        HP_PAP         HAYNN.
        HP_BP          HAYNN2_.
        HP_FLU         HAYNN.
        HP_PROS        HAYNN.
        HP_GP          HAYNN.
        HP_CHOL        HAYNN.
        HP_SMOKE       HAYNN.
        HP_BRST        HAYNN.
        BFGROUPP       XBGC_S. ;



/* CREATE INDEPENDENT VARIABLES */


/* XENRLLMT--ENROLLMENT STATUS */
IF 18 <= ZAGE  < 65 THEN DO;
 IF BFGROUPP = 1 THEN XENRLLMT = 1;             /*   Active duty (<65) */
 ELSE IF ENGROUPP IN ( 1, 2)THEN XENRLLMT = 2;  /*   Non-active duty enrolled (<65)*/
 ELSE IF ENGROUPP =3 THEN XENRLLMT = 3;         /*   Not Enrolled (<65)*/
END;
ELSE IF ZAGE  > = 65
  AND ENGROUPP = 3 THEN XENRLLMT = 4;           /*   Not Enrolled (65+)*/

/* XENR_PCM--ENROLLMENT BY PCM TYPE */
IF 18 <= ZAGE  < 65 THEN DO;
 IF BFGROUPP = 1 THEN XENR_PCM = 1;        /*   Active duty (<65)        */
 ELSE IF ENGROUPP = 1 THEN XENR_PCM = 2;   /*   Enrolled (<65) - mil PCM */
 ELSE IF ENGROUPP = 2 THEN XENR_PCM = 3;   /*   Enrolled (<65) - civ PCM */
 ELSE IF ENGROUPP =3  THEN XENR_PCM = 4;   /*   Not Enrolled (<65)       */
END;
 ELSE IF ZAGE  > = 65
 AND ENGROUPP = 3 THEN XENR_PCM = 5;       /*  Not Enrolled (65+)       */

/* XINS_COV--INSURANCE COVERAGE */
 IF XENRLLMT = 1 THEN XINS_COV =1;                                /* Prime <65-Active Duty */
  ELSE IF 18 <= ZAGE  < 65 AND H99002 IN (1,2) THEN XINS_COV = 2; /* Prime <65-Non-active Duty */
  ELSE IF H99002 = 3 THEN XINS_COV = 3;                           /* Standard/Extra */
  ELSE IF H99002 = 5 THEN XINS_COV = 4;                           /* Medicare */
  ELSE IF H99002 IN (4,6,7) THEN XINS_COV = 5;                    /* Other Insurance */

/* XQENROLL--ENROLLMENT ACCORDING TO QUESTIONNAIRE RESPONSES */
/* USING DEERS SAMPLING VALUES  */
     IF H99005 = 3 THEN DO;
         IF ENGROUPP = 1 THEN XQENROLL = 2;     /* Enrolled (<65) - Mil PCM */
         ELSE IF ENGROUPP = 2 THEN XQENROLL = 3;/* Enrolled (<65) - Civ PCM */
     END;
     ELSE IF H99003 = 2 THEN DO;
      IF 18 <= ZAGE  < 65 THEN XQENROLL = 4;    /* Not enrolled (<65)   */
      ELSE IF ZAGE  >= 65 THEN XQENROLL = 5;    /* Not enrolled (>=65) */
     END;


 /* XBNFGRP-Beneficiary Group that excludes those 65 and over-Active Duty
           and Family Members of Active Duty */
   IF ZAGE  >= 65 AND BFGROUPP IN (1,2) THEN XBNFGRP = .;
   ELSE XBNFGRP = BFGROUPP;



/* CREATE DEPENDENT VARIABLES */
```

```
/* KENRINTN--INTENTION TO ENROLL */
   IF H99073  IN (4, 5) THEN KENRINTN = 1;                    /* Yes */
      ELSE IF H99073  IN (1, 2, 3, -5) THEN KENRINTN = 2;    /* No  */

/* KDISENRL--INTENTION TO DISNEROLL */
   IF H99072  IN (4, 5) THEN KDISENRL = 1;                    /* Yes */
      ELSE IF H99072  IN (1, 2, 3, -5) THEN KDISENRL = 2;    /* No  */

/* KMILWAT1--WAIT LESS THAN 4 WEEKS FOR WELL PATIENT VISIT AT MIL FACILITIES
   KCIVWAT1--WAIT LESS THAN 4 WEEKS FOR WELL PATIENT VISIT AT CIV FACILITIES*/
   IF H99037  = 1 THEN DO;                           /* Military */
      IF H99042  IN (1, 2) THEN KMILWAT1 = 1;        /*   Yes */
      ELSE IF H99042  IN (3,4) THEN KMILWAT1 = 2;    /*   No  */
   END;
      ELSE IF H99037  = 2 THEN DO;                       /*   Civilian */
           IF H99042  IN (1, 2) THEN KCIVWAT1 = 1;      /*   Yes */
           ELSE IF H99042  IN (3,4) THEN KCIVWAT1 = 2;  /*   No  */
      END;

/* KMILOFFC--OFFICE WAIT OF 30 MINUTES OR MORE AT MILITARY FACILITES
   KCIVOFFC--OFFICE WAIT OF 30 MINUTES OR MORE AT CIVILIAN FACILITES */
   IF H99037  = 1 THEN DO;                                 /* Military */
      IF H99046  IN (3,4) THEN KMILOFFC = 1;               /*  Yes */
      ELSE IF H99046  IN (1,2) THEN KMILOFFC = 2;          /*  No  */
   END;
      ELSE IF H99037  = 2 THEN DO;                         /*   Civilian */
           IF H99046  IN (3,4) THEN KCIVOFFC = 1;          /* Yes */
           ELSE IF H99046  IN (1,2) THEN KCIVOFFC = 2;     /* No */
      END;

/* KBGPRB1--BIG PROBLEM GETTING REFERRALS TO SPECIALISTS */
   IF H99027  =1 THEN KBGPRB1 =1;                            /* YES */
      ELSE IF H99027  IN (2,3) THEN KBGPRB1 =2;              /* NO  */

/* KBGPRB2--BIG PROBLEM GETTING NECESSARY CARE */
   IF H99047  =1 THEN KBGPRB2 =1;                          /* YES */
       ELSE IF H99047  IN (2,3) THEN KBGPRB2 =2;           /* NO  */

/* KMILOP99--OUTPATIENT VISITS TO MILITARY FACILITY
   KCIVOP99--OUTPATIENT VISITS TO CIVILIAN FACILITY */
   IF H99037  = 3 THEN KMILOP99=1;
      ELSE KMILOP99 = H99040 ;
   IF H99037  = 3 THEN KCIVOP99=1;
      ELSE KCIVOP99 = H99041 ;


/* HP_PRNTL--IF PREGNANT LAST YEAR, RECEIVED PRENATAL CARE IN 1ST TRIMESTER */
    IF H99020  IN (1,2) THEN DO;                                        /*   Pregnant in last 12
months   */
      IF H99021B = 4 THEN HP_PRNTL = 1;                            /*   Yes   */
         ELSE IF (H99021A = 1 AND H99021B = 1) THEN HP_PRNTL = .;    /* <3 months pregnant now */
         ELSE IF H99021B IN (1,2,3) THEN HP_PRNTL = 2;               /*   No */
   END;

/* HP_MAMOG--FOR WOMEN AGE 50 AND OVER, HAD MAMMOGRAM W/IN PAST 2 YEARS */
   IF XSEXA = 2 AND ZAGE  >= 50 THEN DO;
      IF H99019B IN (5, 4) THEN HP_MAMOG = 1;            /* Yes */
      ELSE IF H99019B IN (1, 2, 3) THEN HP_MAMOG = 2;    /* No */
   END;

/* HP_PAP--FOR ALL WOMEN, HAD PAP SMEAR IN LAST 3 YEARS */
   IF XSEXA = 2 THEN DO;
      IF H99018  IN (4, 5) THEN HP_PAP = 1;            /* Yes */
      ELSE IF H99018  IN (1, 2, 3) THEN HP_PAP = 2;    /* No  */
   END;

/* HP_BP--HAD BLOOD PRESSURE SCREENING IN LAST 2 YEARS AND KNOW RESULT */
   IF H99009A IN (2,3) AND H99009B IN (1, 2) THEN HP_BP = 1; /* Yes */
      ELSE IF H99009A = 1 THEN HP_BP = 2;                    /*    No */
      ELSE IF H99009A < 0 OR H99009B < 0 THEN HP_BP = .;     /* Unknown */
      ELSE HP_BP = 2;         /*    No */

/* HP_FLU--FOR PERSON AGE 65 OR OVER, HAD FLU SHOT IN LAST 12 MONTHS */
```

```
   IF ZAGE  >= 65 THEN DO;
      IF H99011 = 4 THEN HP_FLU = 1;                      /*   Yes */
      ELSE IF H99011 IN (1, 2, 3) THEN HP_FLU = 2;        /*   No  */
   END;

/* HP_PROS--FOR MEN AGE 50 AND OVER, HAD PROSTRATE EXAM W/IN PAST 12 MONTHS */
   IF XSEXA = 1 AND ZAGE  >= 50 THEN DO;
      IF H99017  = 5 THEN HP_PROS = 1;                     /*   Yes */
      ELSE IF H99017  IN (1, 2, 3, 4) THEN HP_PROS = 2;   /*   No   */
   END;

/* HP_GP--EXCEPT WHEN SICK OR PREGNANT, GENERAL PHYSICAL EXAM W/IN PAST 12 MONTHS */
   IF H99008 = 5 THEN HP_GP = 1;                    /*   Yes    */
   ELSE IF H99008 IN (1, 2, 3, 4) THEN HP_GP = 2;  /*   No     */

/* HP_CHOL--HAD CHOLESTEROL SCREENING IN PAST 5 YEARS */
   IF H99010 IN (3, 4, 5) THEN HP_CHOL = 1;         /*    Yes   */
   ELSE IF H99010 IN (1, 2) THEN HP_CHOL = 2;       /*    No    */

/* HP_SMOKE--ADVISED TO QUIT SMOKING IN PAST 12 MONTHS */
   IF H99015 IN (2, 3, 4, 5) THEN HP_SMOKE = 1;     /*   Yes   */
   ELSE IF H99015 = 1 THEN HP_SMOKE = 1;            /*   No    */

/* HP_BRST--BREAST EXAM IN PAST 12 MONTHS */
   IF XSEXA=2 AND ZAGE >= 40 THEN DO;
    IF H99019C = 5 THEN HP_BRST = 1;                   /*   Yes   */
    ELSE IF H99019C IN (1, 2, 3, 4) THEN HP_BRST = 2;  /*   No    */
   END;
 RUN;

/* CHECK NEW 1999 VARIABLES */
PROC FREQ DATA=CONSVAR2;
 TABLES XENRLLMT XENR_PCM XINS_COV XQENROLL BFGROUPP XBNFGRP XREGION
        KENRINTN KDISENRL KBGPRB1 KBGPRB2
        KMILWAT1 KCIVWAT1 KMILOFFC KCIVOFFC
        KMILOP99 KCIVOP99 HP_PRNTL HP_MAMOG HP_PAP HP_BP HP_FLU
        HP_PROS  HP_GP HP_CHOL HP_SMOKE HP_BRST
        / MISSING LIST;
 TITLE3 'ONE WAY FREQUENCIES ON ALL NEW 1999 CONSTRUCTED VARIABLES';
 RUN;

/* CROSSTABS TO CHECK CREATION OF NEW VARIABLES */
   PROC FORMAT;        /* COLLAPSE AGE FOR CROSSTABS */
     VALUE AGE
             18 -< 65 = "LESS THAN 65"
             65-HIGH = "65 OR OLDER"
             .O      = "Out of range err"
             .       = "Missing/unknown" ;
             RUN ;

   PROC FREQ DATA=CONSVAR2;
    TABLES ZAGE *BFGROUPP*ENGROUPP*XENRLLMT
           ZAGE *BFGROUPP*ENGROUPP*XENR_PCM
           ZAGE *XENRLLMT*H99002*XINS_COV
           ZAGE *H99003*H99005*ENGROUPP*XQENROLL
           XENR_PCM*XQENROLL   /* compare enrollment by pcm & questionnaire */
           XREGION*CACSMPL
           ZAGE *BFGROUPP*XBNFGRP
           H99073 *KENRINTN
           H99072 *KDISENRL
           H99037 *H99042 *KMILWAT1*KCIVWAT1
           H99037 *H99046 *KMILOFFC*KCIVOFFC
           H99027 *KBGPRB1
           H99047 *KBGPRB2
           H99040 *H99037 *KMILOP99
           H99037 *H99041 *KCIVOP99
           H99020 *H99021A*H99021B*HP_PRNTL
           XSEXA*H99018 *HP_PAP
           H99009A*H99009B*HP_BP
           ZAGE *H99011*HP_FLU
           H99008*HP_GP;
           H99010*HP_CHOL;
           H99015*H99012*H99013*H99014*HP_SMOKE;
```

```
        ZAGE*XSEXA*HP_BRST;
        / MISSING LIST;
        FORMAT ZAGE  AGE.;
  TITLE3 'CROSSTABS ON ALL NEW VARIABLES';
         RUN;

PROC FREQ DATA=CONSVAR2;
   tables XREGION*CACSMPL
        / MISSING LIST;
   format _all_;
run;
  PROC FORMAT ;  /* COLLAPSE FOR MAMMOGRAPHY AND PROSTRATE XTAB*/
     VALUE AGE2_
          18 - 49 = "LESS THAN 50"
          50-HIGH = "50 OR OLDER"
          .O      = "Out of range err"
          .       = "Missing/unknown" ;
          RUN ;

      PROC FREQ DATA=CONSVAR2;
          TABLES XSEXA*ZAGE *H99019B*HP_MAMOG
                 XSEXA*H99017 *ZAGE *HP_PROS
                 /MISSING LIST;
                 FORMAT ZAGE  AGE2_.;
      RUN;

PROC CONTENTS DATA =IN.CONVARA2;
RUN;
```

6. CONSTRUCTION OF HEALTH STATUS VARIABLES

```
/*****************************************************************************
Project: 1999 Health Care Survey of DoD Beneficiaries Study
Program: CONVARA3.SAS
Purpose: The purpose of this program is to create the SF-12 (Health Status)
         variables as outlined in the Database Prep Plan, SF-12 Manual,
         Variables created here are used for defining
         rows in the deliverable tables.  Variables created include:

         USE S* VARIABLES TO COMPUTE AVERAGE OF SCALES (NONMISSING VALUES):
         SF12PF, SF12RP, SF12RE, SF12VT, SF12MH, SF12SF, SF12BP, SF12GH,
         SFHCHG (AS IN 1994-5)

         Create health status variables based on norms for general US
         population by age group: KMID_H, KMID_MH

         FOLLOWING CODE IN SF-12 MANUAL, COMPUTE PHYSICAL HEALTH SUMMARY
         MEASURE (SF12PCS) AND MENTAL HEALTH SUMMARY MEASURE (SF12MCS).
         INTERMEDIATE VARIABLES CREATED HERE HAVE THE SAME NAMES AS THOSE
         IN SF-12 MANUAL AND ARE BASED ON THE NON-SCALED VERSIONS OF ITEMS
         1-8.

Date     : 1/26/99
Modified :11/10/98 by Daisy Ewell to correct detected errors from 1997's
          program
           03/22/99 by Daisy Ewell to eliminate the top and bottom
              quartile health status variables
           03/01/00 by Natalie Justh - update for 1999 survey
Author   : Kelly White
Input    : J:\8687\DATA1999\AFINAL\SANITIZA.SD2
Output   : J:\8687\DATA1999\AFINAL\CONSVAR3.SD2
*****************************************************************************/
LIBNAME IN 'J:\8687\DATA1999\AFINAL';
LIBNAME LIBRARY 'J:\8687\data1999\AFINAL\fmtlib';
OPTIONS PS=79 LS=132 ERRORS=2 ;
TITLE 'CREATE SF12 HEALTH STATUS VARIABLES';

DATA IN.CONVARA3 (KEEP=SF12PCS SF12MCS MPRID KMID_H KMID_MH)
   CONSVAR3;
   SET IN.SANITIZA;



 LENGTH   KMID_H KMID_MH 3;

/* PER SF12 MANUAL, CALCULATE PHYSICAL & MENTAL HEALTH SUMMARY VARS -
   REVERSE SCORE 5 VARS THEN CREATE BINARY VARIABLES */
/* NOTE:ALL RS99* VARS AND THE BINARY INDICATOR VARS ARE TEMPORARY --
   NOT DELIVERED */

  LENGTH RS99078A RS99078B RS99083 RS99081 RS99082C
       PF02_1 PF02_2 PF04_1 PF04_2 RP2_1 RP3_1 BP2_1 BP2_2 BP2_3 BP2_4
       GH1_1 GH1_2 GH1_3 GH1_4 VT2_1 VT2_2 VT2_3 VT2_4 VT2_5 SF2_1 SF2_2
       SF2_3 SF2_4 RE2_1 RE3_1 MH3_1 MH3_2 MH3_3 MH3_4 MH3_5 MH4_1 MH4_2
       MH4_3 MH4_4 MH4_5 3 ;

/* REVERSE SCORE 5 VARIABLES SO THAT HIGHER SCORE INDICATES BETTER HEALTH */
  RS99078A=4-H99078A;
  RS99078B=4-H99078B;
  RS99081 =6-H99081;
  RS99082C=7-H99082C;

  select (h99083);
        when (5) rs99083 = 1;
        when (4) rs99083 = 2;
        when (3) rs99083 = 3;
        when (2) rs99083 = 4;
        when (1) rs99083 = 5;
        otherwise rs99083 = .;
  end;
/* THE FOLLOWING CODE COMES FROM SF-12 MANUAL */
/* CREATE BINARY INDICATOR VARIABLES FROM ITEM RESPONSE CHOICES */
  PF02_1 = .;
```

```
if RS99078A < 0 then PF02_1 = .; else
if RS99078A = 1 then PF02_1 = 1; else PF02_1 = 0;

PF02_2 = .;
if RS99078A < 0 then PF02_2 = .; else
if RS99078A = 2 then PF02_2 = 1; else PF02_2 = 0;

PF04_1 = .;
if RS99078B < 0 then PF04_1 = .; else
if RS99078B = 1 then PF04_1 = 1; else PF04_1 = 0;

PF04_2 = .;
if RS99078B < 0 then PF04_2 = .; else
if RS99078B = 2 then PF04_2 = 1; else PF04_2 = 0;

RP2_1 = .;
if H99079A < 0 then RP2_1 = .; else
if H99079A = 1 then RP2_1 = 1; else RP2_1 = 0;

RP3_1 = .;
if H99079B < 0 then RP3_1 = .; else
if H99079B = 1 then RP3_1 = 1; else RP3_1 = 0;

BP2_1 = .;
if RS99081 < 0 then BP2_1 = .; else
if RS99081 = 1 then BP2_1 = 1; else BP2_1 = 0;

BP2_2 = .;
if RS99081 < 0 then BP2_2 = .; else
if RS99081 = 2 then BP2_2 = 1; else BP2_2 = 0;

BP2_3 = .;
if RS99081 < 0 then BP2_3 = .; else
if RS99081 = 3 then BP2_3 = 1; else BP2_3 = 0;

BP2_4 = .;
if RS99081 < 0 then BP2_4 = .; else
if RS99081 = 4 then BP2_4 = 1; else BP2_4 = 0;

GH1_1 = .;
if H99077 < 0 then GH1_1 = .; else
if H99077 = 1 then GH1_1 = 1; else GH1_1 = 0;

GH1_2 = .;
if H99077 < 0 then GH1_2 = .; else
if H99077 = 2 then GH1_2 = 1; else GH1_2 = 0;

GH1_3 = .;
if H99077 < 0 then GH1_3 = .; else
if H99077 = 3 then GH1_3 = 1; else GH1_3 = 0;

GH1_4 = .;
if H99077 < 0 then GH1_4 = .; else
if H99077 = 4 then GH1_4 = 1; else GH1_4 = 0;

VT2_1 = .;
if H99082B < 0 then VT2_1 = .; else
if H99082B = 1 then VT2_1 = 1; else VT2_1 = 0;

VT2_2 = .;
if H99082B < 0 then VT2_2 = .; else
if H99082B = 2 then VT2_2 = 1; else VT2_2 = 0;

VT2_3 = .;
if H99082B < 0 then VT2_3 = .; else
if H99082B = 3 then VT2_3 = 1; else VT2_3 = 0;

VT2_4 = .;
if H99082B < 0 then VT2_4 = .; else
if H99082B = 4 then VT2_4 = 1; else VT2_4 = 0;

VT2_5 = .;
if H99082B < 0 then VT2_5 = .; else
```

```
  if H99082B = 5 then VT2_5 = 1; else VT2_5 = 0;

  SF2_1 = .;
  if RS99083 < 0 then SF2_1 = .; else
  if RS99083 = 1 then SF2_1 = 1; else SF2_1 = 0;

  SF2_2 = .;
  if RS99083 < 0 then SF2_2 = .; else
  if RS99083 = 2 then SF2_2 = 1; else SF2_2 = 0;

  SF2_3 = .;
  if RS99083 < 0 then SF2_3 = .; else
  if RS99083 = 3 then SF2_3 = 1; else SF2_3 = 0;

  SF2_4 =.;
  if RS99083 < 0 then  SF2_4 =.; ELSE
  if RS99083 = 4 THEN  SF2_4= 1; ELSE SF2_4 = 0;

  RE2_1 = .;
  IF H99080A < 0 then RE2_1 = .; ELSE
  IF H99080A = 1 THEN RE2_1 = 1; ELSE RE2_1 = 0;

  RE3_1 = .;
  IF H99080B < 0 then RE3_1 = .; ELSE
  IF H99080B = 1 THEN RE3_1 = 1; ELSE RE3_1 = 0;

  MH3_1 = .;
  IF H99082A < 0 then MH3_1 = .; ELSE
  IF H99082A = 1 THEN MH3_1 = 1; ELSE MH3_1 = 0;

  MH3_2 = .;
  IF H99082A < 0 then MH3_2 = .; ELSE
  IF H99082A = 2 THEN MH3_2 = 1; ELSE MH3_2 = 0;

  MH3_3 = .;
  IF H99082A < 0 then MH3_3 = .; ELSE
  IF H99082A = 3 THEN MH3_3 = 1; ELSE MH3_3 = 0;

  MH3_4 = .;
  IF H99082A < 0 then MH3_4 = .; ELSE
  IF H99082A = 4 THEN MH3_4 = 1; ELSE MH3_4 = 0;

  MH3_5 = .;
  IF H99082A < 0 then MH3_5 = .; ELSE
  IF H99082A = 5 THEN MH3_5 = 1; ELSE MH3_5 = 0;

  MH4_1 = .;
  IF RS99082C < 0 then  MH4_1 = .; ELSE
  IF RS99082C = 1 THEN  MH4_1 = 1; ELSE MH4_1 = 0;

  MH4_2 = .;
  IF RS99082C < 0 then MH4_2 = .; ELSE
  IF RS99082C = 2 THEN MH4_2 = 1; ELSE MH4_2 = 0;

  MH4_3 = .;
  IF RS99082C < 0 then MH4_3 = .; ELSE
  IF RS99082C = 3 THEN MH4_3  = 1;ELSE MH4_3 = 0;

  MH4_4 = .;
  IF RS99082C < 0 then MH4_4 = .; ELSE
  IF RS99082C = 4 THEN MH4_4  = 1; ELSE MH4_4 = 0;

  MH4_5 = .;
  IF RS99082C < 0 then MH4_5 = .; ELSE
  IF RS99082C = 5 THEN MH4_5 = 1; ELSE MH4_5 = 0;

/*************************************************************
WEIGHTING AND AGGREGATION OF INDICATOR VARIABLES USING
PHYSICAL AND MENTAL REGRESSION WEIGHTS.
RAWPCS12 & RAWMCS12 ARE TEMPORARY VARIABLES.
 *************************************************************/
  RAWPCS12 = (-7.23216*PF02_1) + (-3.45555*PF02_2) +
      (-6.24397*PF04_1) + (-2.7357*PF04_2) + (-4.61617*RP2_1) +
```

```
      (-5.51747*RP3_1) + (-11.25544*BP2_1) + (-8.38063*BP2_2) +
      (-6.50522*BP2_3) + (-3.80130*BP2_4) + (-8.37399*GH1_1) +
      (-5.56461*GH1_2) + (-3.02396*GH1_3) + (-1.31872*GH1_4) +
      (-2.44706*VT2_1) + (-2.02168*VT2_2) + (-1.6185*VT2_3)  +
      (-1.14387*VT2_4) + (-0.42251*VT2_5) + (-0.33682*SF2_1) +
      (-0.94342*SF2_2) + (-0.18043*SF2_3) + (0.11038*SF2_4)  +
      (3.04365*RE2_1) + (2.32091*RE3_1)  + (3.46638*MH3_1) +
      (2.90426*MH3_2) + (2.37241*MH3_3)  + (1.36689*MH3_4) +
      (0.66514*MH3_5) + (4.61446*MH4_1)  + (3.41593*MH4_2) +
      (2.34247*MH4_3) + (1.28044*MH4_4)  + (0.41188*MH4_5);

  RAWMCS12 = (3.93115*PF02_1) + (1.8684*PF02_2) +
      (2.68282*PF04_1) + (1.43103*PF04_2) + (1.4406*RP2_1)  +
      (1.66968*RP3_1)  + (1.48619*BP2_1)  +  (1.76691*BP2_2)  +
      (1.49384*BP2_3)  + (0.90384*BP2_4)  +  (-1.71175*GH1_1) +
      (-0.16891*GH1_2) + (0.03482*GH1_3)  +  (-0.06064*GH1_4) +
      (-6.02409*VT2_1) + (-4.88962*VT2_2) +  (-3.29805*VT2_3) +
      (-1.65178*VT2_4) + (-0.92057*VT2_5) +  (-6.29724*SF2_1) +
      (-8.26066*SF2_2) + (-5.63286*SF2_3) +  (-3.13896*SF2_4) +
      (-6.82672*RE2_1) + (-5.69921*RE3_1) +  (-10.19085*MH3_1) +
      (-7.92717*MH3_2) + (-6.31121*MH3_3) +  (-4.09842*MH3_4) +
      (-1.94949*MH3_5) + (-16.15395*MH4_1) +  (-10.77911*MH4_2) +
      (-8.09914*MH4_3) + (-4.59055*MH4_4) + (-1.95934*MH4_5);

/*****************************************************************
NORM-BASED STANDARDIZATION OF SCALE SCORES
*****************************************************************/
  SF12PCS = RAWPCS12 + 56.57706;
  SF12MCS = RAWMCS12 + 60.75781;

/*****************************************************************
BINARY MEASURES BASED ON HEALTH STATUS, AGE OF RESPONDENT, &
NORMS FOR GENERAL U.S. POPULATION
*****************************************************************/

****BELOW MEDIAN PHYSICAL HEALTH****;
IF ZAGE < 18 OR SF12PCS = . THEN KMID_H = .;
ELSE IF 18 <= ZAGE <= 34 AND SF12PCS < 55.18 THEN KMID_H = 1;
ELSE IF 35 <= ZAGE <= 44 AND SF12PCS < 54.30 THEN KMID_H = 1;
ELSE IF 45 <= ZAGE <= 54 AND SF12PCS < 52.76 THEN KMID_H = 1;
ELSE IF 55 <= ZAGE <= 64 AND SF12PCS < 50.22 THEN KMID_H = 1;
ELSE IF 65 <= ZAGE <= 74 AND SF12PCS < 46.36 THEN KMID_H = 1;
ELSE IF ZAGE >= 75 AND SF12PCS < 38.68 THEN KMID_H = 1;
ELSE KMID_H = 2;


****BELOW MEDIAN MENTAL HEALTH****;
IF ZAGE < 18 OR SF12MCS = . THEN KMID_MH = .;
ELSE IF 18 <= ZAGE <= 34 AND SF12MCS < 51.81 THEN KMID_MH = 1;
ELSE IF 35 <= ZAGE <= 44 AND SF12MCS < 52.24 THEN KMID_MH = 1;
ELSE IF 45 <= ZAGE <= 54 AND SF12MCS < 53.30 THEN KMID_MH = 1;
ELSE IF 55 <= ZAGE <= 64 AND SF12MCS < 53.14 THEN KMID_MH = 1;
ELSE IF 65 <= ZAGE <= 74 AND SF12MCS < 55.31 THEN KMID_MH = 1;
ELSE IF ZAGE >= 75 AND SF12MCS < 53.53 THEN KMID_MH = 1;
ELSE KMID_MH = 2;


  LABEL
     PF02_1 = "SF12 Binary indicator var - Q106A val 1"
     PF02_2 = "SF12 Binary indicator var - Q106A val 2"
     PF04_1 = "SF12 Binary indicator var - Q106B val 1"
     PF04_2 = "SF12 Binary indicator var - Q106B val 2"
     RP2_1 = "SF12 Binary indicator var - Q107A"
     RP3_1 = "SF12 Binary indicator var - Q107B"
     BP2_1 = "SF12 Binary indicator var - Q109 val 1"
     BP2_2 = "SF12 Binary indicator var - Q109 val 2"
     BP2_3 = "SF12 Binary indicator var - Q109 val 3"
     BP2_4 = "SF12 Binary indicator var - Q109 val 4"
     GH1_1 = "SF12 Binary indicator var - Q105 val 1"
     GH1_2 = "SF12 Binary indicator var - Q105 val 2"
     GH1_3 = "SF12 Binary indicator var - Q105 val 3"
     GH1_4 = "SF12 Binary indicator var - Q105 val 4"
     VT2_1 = "SF12 Binary indicator var - Q110B val 1"
```

```
     VT2_2 = "SF12 Binary indicator var - Q110B val 2"
     VT2_3 = "SF12 Binary indicator var - Q110B val 3"
     VT2_4 = "SF12 Binary indicator var - Q110B val 4"
     VT2_5 = "SF12 Binary indicator var - Q110B val 5"
     SF2_1 = "SF12 Binary indicator var - Q111  val 1"
     SF2_2 = "SF12 Binary indicator var - Q111  val 2"
     SF2_3 = "SF12 Binary indicator var - Q111  val 3"
     SF2_4 = "SF12 Binary indicator var - Q111  val 4"
     RE2_1 = "SF12 Binary indicator var - Q108A"
     RE3_1 = "SF12 Binary indicator var - Q108B"
     MH3_1 = "SF12 Binary indicator var - Q110A val 1"
     MH3_2 = "SF12 Binary indicator var - Q110A val 2"
     MH3_3 = "SF12 Binary indicator var - Q110A val 3"
     MH3_4 = "SF12 Binary indicator var - Q110A val 4"
     MH3_5 = "SF12 Binary indicator var - Q110A val 5"
     MH4_1 = "SF12 Binary indicator var - Q110C val 1"
     MH4_2 = "SF12 Binary indicator var - Q110C val 2"
     MH4_3 = "SF12 Binary indicator var - Q110C val 3"
     MH4_4 = "SF12 Binary indicator var - Q110C val 4"
     MH4_5 = "SF12 Binary indicator var - Q110C val 5"
     RS99078A = "Reverse score for SF12 calc - Q078A"
     RS99078B = "Reverse score for SF12 calc - Q078B"
     RS99083 = "Reverse score for SF12 calc - Q083"
     RS99081 = "Reverse score for SF12 calc - Q081"
     RS99082C = "Reverse score for SF12 calc - Q082C"
     RAWPCS12 = "Uses physical regression weights"
     RAWMCS12 = "Uses mental regression weights"
     SF12PCS = "Physical Health Summary"
     SF12MCS = "Mental Health Summary"
     KMID_H   = 'Below Median Physical Health'
     KMID_MH  = 'Below Median Mental Health'
     ;

  FORMAT
          KMID_H   HAYNN.
          KMID_MH  HAYNN.  ;

  RUN;


PROC FREQ DATA=CONSVAR3;
FORMAT _ALL_;
TABLES
   PF02_1*RS99078A*H99078A PF02_2*RS99078A*H99078A
   PF04_1*RS99078B*H99078B PF04_2*RS99078B*H99078B
   RP2_1*H99079A RP3_1*H99079B
   (BP2_1 BP2_2 BP2_3 BP2_4)*RS99081*H99081
   (GH1_1 GH1_2 GH1_3 GH1_4)*H99077
   (VT2_1 VT2_2 VT2_3 VT2_4 VT2_5)*H99082B
   (SF2_1 SF2_2 SF2_3 SF2_4)*RS99083*H99083
   RE2_1*H99080A RE3_1*H99080B
   (MH3_1 MH3_2 MH3_3 MH3_4 MH3_5)*H99082A
   (MH4_1 MH4_2 MH4_3 MH4_4 MH4_5)*RS99082C*H99082C
   KMID_H KMID_MH
   /MISSING LIST;
  RUN;


PROC MEANS;
 VAR SF12PCS SF12MCS;

PROC CORR;
 VAR SF12PCS;
 WITH SF12MCS;
 RUN;

DATA MISSING  (KEEP=H99078A H99078B H99083 H99081 H99082C RS99078A
              RS99078B RS99083 RS99081 RS99082c);
   SET CONSVAR3;
   IF RS99078A= .;

 PROC PRINT;
 TITLE 'OBSERVATIONS WITH MISSING DATA VALUES GENERATED';
```

```
 RUN;

PROC CONTENTS DATA=IN.CONVARA3;
  RUN;
```

7. MERGE CONSTRUCTED VARIABLES ONTO DATA FILE

```
*******************************************************************************
* PROGRAM:   MERGE.SAS
* WRITTEN:   1/28/99 BY KELLY WHITE
* MODIFIED:  3/1/00 BY NATALIE JUSTH
* PURPOSE:   TO MERGE FINAL FILES TOGETHER AND REORDER BY VARIABLE TYPE
*            To reorder variables within the record use a
*             LENGTH statement before the SET statement.
*             Make sure that MPRID is the first variable in the
*             record followed by:
*                            1)  other sampling variables
*                            2)  DEERS variables
*                            3)  Post-stratification vars
*                            4)  questionnaire responses
*                            5)  DRC variables
*                            6)  recoded questionnaire responses
*                            7)  coding scheme flags
*                            8)  constructed variables
*                            9)  weights  (NOT AVAILABLE FOR PRELIMINARY DATA)
* INPUT:     J:\8687\DATA1999\AFINAL\SANITIZA.SD2
* OUTPUT:    J:\8687\DATA1999\AFINAL\HCS99A_4.SD2
*******************************************************************************
*;
LIBNAME IN 'J:\8687\DATA1999\AFINAL';
LIBNAME LIBRARY 'J:\8687\DATA1999\AFINAL\FMTLIB';
OPTIONS PS=75 LS=111 ERRORS=2 COMPRESS=YES;

PROC SORT DATA=IN.CONVARA3 OUT=CONSVAR3;
  BY MPRID;
  RUN;


PROC SORT DATA=IN.CONVARA2 OUT=CONSVAR2;
BY MPRID;
RUN;

PROC SORT DATA=IN.SANITIZA OUT=SANITIZA;
BY MPRID;
RUN;

DATA WEIGH03A(KEEP=MPRID BWT99);
  SET IN.WEIGH03A;
RUN;

PROC SORT DATA=WEIGH03A;
BY MPRID;
RUN;

DATA IN.HCS99A_4( DROP=  H99001_O H99002_O H99003_O H99004_O H99005_O H99006_O
                     H99007AO H99007BO H99007CO H99007DO H99007EO H99007FO H99007GO
                     H99007HO H99007IO H99007JO H99007KO H99007LO H99008_O H99009AO
                     H99009BO H99010_O H99011_O H99012_O H99013_O H99014_O H99015_O
                     H99017_O H99018_O H99019AO H99019BO H99019CO H99020_O H99021AO
                     H99021BO H99022_O H99023_O H99024_O H99025_O H99026_O H99027_O
                     H99028_O H99029_O H99030_O H99031_O H99032_O H99033_O H99034_O
                     H99035_O H99036_O H99037_O H99038_O H99039AO H99039BO H99039CO
                     H99039DO H99039EO H99039FO H99039GO H99039HO H99039IO H99039JO
                     H99040_O H99041_O H99042_O H99043_O H99044_O H99045_O H99046_O
                     H99047_O H99048_O H99049_O H99050_O H99051_O H99052_O H99053_O
                     H99054_O H99055_O H99056_O H99057_O H99058_O H99059_O H99060_O
                     H99061AO H99061BO H99062_O H99063_O H99064_O H99065_O H99066_O
                     H99067_O H99068_O H99069_O H99070_O H99071_O H99072_O H99073_O
                     H99074_O H99075_O H99076AO H99076BO H99076CO H99076DO H99076EO
                     H99076FO H99076GO H99077_O H99078AO H99078BO H99079AO H99079BO
                     H99080AO H99080BO H99081_O H99082AO H99082BO H99082CO H99083_O
                     H99084_O H99087AO H99087BO H99088_O SREDA_O SREDB_O SREDC_O SREDD_O
                     SREDE_O SREDF_O SRSEX_O SRRACEAO SRRACEBO SRRACECO SRRACEDO
                     SRRACEEO ) ;
   MERGE SANITIZA(in=hcsdb) CONSVAR2 CONSVAR3 /*WEIGH03A CONSVAR4 IN.FINAL02A */;
   BY MPRID;
   if hcsdb;
```

```
  FORMAT
     CACSMPL    CAC.
     XSEXA      HASEX.
     SEXSMPL    HASEX.
     AGEGROUP   AGESMPL.
     GROUP      $GROUP.
     MISS_1     HAMISS.
     MISS_4     HAMISS.
     MISS_5     HAMISS.
     MISS_6     HAMISS.
     MISS_7     HAMISS.
     MISS_8     HAMISS.
     MISS_9     HAMISS.
     MISS_TOT   HAMISS.
     MPCSMPL    MPCSMPL.
     MSTATUS    $MSTATUS.
     RACE       $RACE.
     SEX        $SEX.
     SREDHIGH   HAEDUC.
     SVC        $SVC.
     SVCSMPL    SVCSMPL.;

 RUN;

PROC CONTENTS DATA=IN.HCS99A_4;
RUN;
```

8. RESPONSE RATE CALCULATIONS

```
*****************************************************************************
*
* PROGRAM: TABLE02.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8676-002)
* PURPOSE: BUILD TABLE 2: RESPONSE RATES BY DOMAIN SUMMARY
*          1999 DOD HEALTH CARE SURVEY FILE.
* WRITTEN: 11/09/99 BY KEITH RATHBUN
*
* INCLUDES: 1) TABLE02.IN1
*           2) TABLE02.IN2
*
*****************************************************************************
*;
LIBNAME IN   "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT  "D:\KEITH\DOD99-8676\DATA";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

%MACRO PROCESS(INPT,FORM,LIB);
LIBNAME LIBRARY &LIB;
*********************************************************************
* Process OVERALL Summary of response rates
*********************************************************************
;
DATA _NULL_;
   SET IN.&INPT END=FINISHED;
   IF _N_ = 1 THEN DO;
      SN    = 0;
      SN1   = 0;
      SN11  = 0;
      SN12  = 0;
      SN2   = 0;
      SN3   = 0;
      SN4   = 0;
      SN41  = 0;
      SN42  = 0;
      WN   = 0;
      WN1  = 0;
      WN11 = 0;
      WN12 = 0;
      WN2  = 0;
      WN3  = 0;
      WN4  = 0;
      WN41 = 0;
      WN42 = 0;
   END;
*****************************************************
* Accumulate group 1 weighted and unweighted counts
*****************************************************
;
SN + 1;
WN + BWT99;
IF FNSTATUS IN(11,12) THEN DO;
   SN1 + 1;
   WN1 + BWT99;
   IF FNSTATUS = 11 THEN DO;
      SN11 + 1;
      WN11 + BWT99;
   END;
   ELSE DO;
      SN12 + 1;
      WN12 + BWT99;
   END;
END;
*****************************************************
* Accumulate group 2 weighted and unweighted counts
*****************************************************
;
ELSE IF FNSTATUS = 20 THEN DO;
   SN2 + 1;
   WN2 + BWT99;
END;
*****************************************************
```

```
* Accumulate group 3 weighted and unweighted counts
****************************************************
;
ELSE IF FNSTATUS = 30 THEN DO;
   SN3 + 1;
   WN3 + BWT99;
END;
****************************************************
* Accumulate group 4 weighted and unweighted counts
****************************************************
;
ELSE IF FNSTATUS IN(41,42) THEN DO;
   SN4 + 1;
   WN4 + BWT99;
   IF FNSTATUS = 42 THEN DO;
      SN42 + 1;
      WN42 + BWT99;
   END;
   ELSE DO;
      SN41 + 1;
      WN41 + BWT99;
   END;
END;

DROP I;
RETAIN
   SN
   SN1
   SN11
   SN12
   SN2
   SN3
   SN4
   SN41
   SN42
   WN
   WN1
   WN11
   WN12
   WN2
   WN3
   WN4
   WN41
   WN42
  ;

   IF FINISHED THEN GO TO FINISHED;

   RETURN;

FINISHED:
   FILE "D:\KEITH\DOD99-8676\RES-RATE\TABLE02&FORM..OUT" LRECL=132;
   PUT; PUT;
   PUT @001 "TABLE 2: OVERALL RESPONSE RATES SUMMARY";
   PUT @001 "WRITTEN BY KEITH RATHBUN, 02-03-2000, TASK: 8676-002";
   PUT;
   PUT "SUMMARY OF GROUP COUNTS: FORM &FORM";
   PUT;
   PUT @060 "UNWEIGHTED COUNT"
       @100 "WEIGHTED COUNT"
       ;
   PUT @050 'FLR'
       @060 'FCR'
       @070 'FRR'
       @090 'FLR'
       @100 'FCR'
       @110 'FRR'
       ;
   %INCLUDE "D:\KEITH\DOD99-8676\SAS\TABLE02.IN2";
RUN;
%MEND PROCESS;

%PROCESS(KEEPERSA,A,"D:\KEITH\DOD99-8676\DATA\FMTLIBA");
```

```
RUN;

*********************************************************************
* Process Single Domain where domain1 is the variable of interest
*********************************************************************
;

%MACRO PROCESS1(DOMAIN1,INPT,FORM,LIB);

LIBNAME LIBRARY &LIB;

PROC SORT DATA=OUT.&INPT OUT=&INPT TAGSORT; BY &DOMAIN1; RUN;

DATA _NULL_;
   SET &INPT;
   BY &DOMAIN1;
   FILE "D:\KEITH\DOD99-8676\RES-RATE\&DOMAIN1..OUT" LRECL=132;
   LENGTH VARNAME1 $8;
   LENGTH VARIABLE $30;
   CALL VNAME(&DOMAIN1,VARNAME1);
   VARIABLE = VARNAME1;
   %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN1';
   IF LAST.&DOMAIN1 THEN DO;
      PUT @001 &DOMAIN1 @;
      %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN2';
   END; * DOMAIN;
RUN;
%MEND PROCESS1;

**********************************************************************
* Process Double Domain where domain1/domain2 are the variables of interest
**********************************************************************
;
%MACRO PROCESS2(DOMAIN1,DOMAIN2,INPT,FORM,LIB);

LIBNAME LIBRARY &LIB;

PROC SORT DATA=OUT.&INPT OUT=&INPT TAGSORT; BY &DOMAIN1 &DOMAIN2; RUN;

DATA _NULL_;
   SET &INPT;
   BY &DOMAIN1 &DOMAIN2;
   FILE "D:\KEITH\DOD99-8676\RES-RATE\&DOMAIN1&DOMAIN2..OUT" LRECL=132;
   LENGTH VARNAME1 $8;
   LENGTH VARNAME2 $8;
   LENGTH VARIABLE $30;
   CALL VNAME(&DOMAIN1,VARNAME1);
   CALL VNAME(&DOMAIN2,VARNAME2);
   VARIABLE = VARNAME1 || " " || VARNAME2;
   %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN1';
   IF LAST.&DOMAIN2 THEN DO;
      PUT @001 &DOMAIN1 @;
      PUT @025 &DOMAIN2 @;
      %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN2';
      SN    = 0;
      SN1   = 0;
      SN11  = 0;
      SN12  = 0;
      SN2   = 0;
      SN3   = 0;
      SN4   = 0;
      SN41  = 0;
      SN42  = 0;
      WN   = 0;
      WN1  = 0;
      WN11 = 0;
      WN12 = 0;
      WN2  = 0;
      WN3  = 0;
      WN4  = 0;
      WN41 = 0;
      WN42 = 0;
   END; * DOMAIN;
```

```
RUN;
%MEND PROCESS2;

*****************************************************************************
* Process Triple Domain where domain1-3 are the variables of interest
*****************************************************************************
;
%MACRO PROCESS3(DOMAIN1,DOMAIN2,DOMAIN3,INPT,FORM,LIB);

LIBNAME LIBRARY &LIB;

PROC SORT DATA=OUT.&INPT OUT=&INPT TAGSORT; BY &DOMAIN1 &DOMAIN2 &DOMAIN3; RUN;

DATA _NULL_;
   SET &INPT;
   BY &DOMAIN1 &DOMAIN2 &DOMAIN3;
   FILE "D:\KEITH\DOD99-8676\RES-RATE\&DOMAIN1&DOMAIN2&DOMAIN3..OUT" LRECL=132;
   LENGTH VARNAME1 $8;
   LENGTH VARNAME2 $8;
   LENGTH VARNAME3 $8;
   LENGTH VARIABLE $30;
   CALL VNAME(&DOMAIN1,VARNAME1);
   CALL VNAME(&DOMAIN2,VARNAME2);
   CALL VNAME(&DOMAIN3,VARNAME3);
   VARIABLE = VARNAME1 || " " || VARNAME2 || " " || VARNAME3;
   %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN1';
   IF LAST.&DOMAIN3 THEN DO;
      PUT @001 &DOMAIN1 @;
      PUT @018 &DOMAIN2 @;
      PUT @038 &DOMAIN3 @;
      %INCLUDE 'D:\KEITH\DOD99-8676\SAS\TABLE02.IN2';
      SN    = 0;
      SN1   = 0;
      SN11  = 0;
      SN12  = 0;
      SN2   = 0;
      SN3   = 0;
      SN4   = 0;
      SN41  = 0;
      SN42  = 0;
      WN   = 0;
      WN1  = 0;
      WN11 = 0;
      WN12 = 0;
      WN2  = 0;
      WN3  = 0;
      WN4  = 0;
      WN41 = 0;
      WN42 = 0;
   END; * DOMAIN;
RUN;
%MEND PROCESS3;

****************************************************
* PROCESS SINGLE DOMAIN TABULATION - FORM A
****************************************************
;
%PROCESS1(CONUS,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS1(CACSMPL,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS1(BFGROUPP,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS1(ENGROUPP,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS1(XREGION,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
RUN;

****************************************************
* PROCESS DOUBLE DOMAIN TABULATION - FORM A
****************************************************
;
```

```
%PROCESS2(XREGION,CACSMPL,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS2(ENGROUPP,BFGROUPP,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS2(BFGROUPP,MPCSMPL,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS2(BFGROUPP,SEXSMPL,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS2(BFGROUPP,SVCSMPL,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
%PROCESS2(BFGROUPP,RACE,KEEPERSA, "FORM A",
         "D:\KEITH\DOD99-8676\DATA\FMTLIBA");
RUN;
```

9. DEVELOPMENT OF WEIGHTS

```
*************************************************************************
*
* PROGRAM: WEIGHT03.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8676-200)
* PURPOSE: BUILD AND ASSIGN ADJUSTED WEIGHTS.
*          WEIGHTS FOR DOD SURVEY.
*          DOD HEALTH CARE SURVEY FILE.
*          REQUESTED BY DON JANG.
* WRITTEN: 11/09/99 BY KEITH RATHBUN
*
* INPUTS:  1) KEEPERSA.SD2 - Reduced/Deduped 1999 DOD survey dataset - Form A
*          2) KEEPERSC.SD2 - Reduced/Deduped 1999 DOD survey dataset - Form C
*
* OUTPUTS: 1) WEIGH03A.SD2 - Form A dataset with adjusted weight (AWT)
*          2) WEIGH03C.SD2 - Form C dataset with adjusted weight (AWT)
*
*************************************************************************
*;
LIBNAME IN       "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT      "D:\KEITH\DOD99-8676\DATA";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

*************************************************************************
* Calculate adjusted weight based on user-specified domains.
*************************************************************************;
%MACRO PROCESS(DOMAIN1,DOMAIN2,DOMAIN3,FORM,INPT);
*************************************************************************
* Collapse enrollment group and beneficiary group for non-response adjustment
*************************************************************************;
DATA PRIME;
   SET IN.&INPT;

   IF ENGROUPP IN(1,2)  THEN PRIME = 1; * Prime;
   ELSE IF ENGROUPP = 3 THEN PRIME = 3; * Not Prime;

   *************************************************************************
   * Collapse beneficiary group (BFGROUPP) to create BENCAT.
   *************************************************************************;
   IF BFGROUPP = 1 THEN BENCAT = 1; * Active Duty;
   ELSE IF BFGROUPP IN (2,3) THEN BENCAT = 2; * Active duty(2) and retirees dependents LT 65 (3);
   ELSE IF BFGROUPP = 4 THEN BENCAT = 4; * 65 or over  ;

   *************************************************************************
   * Collapse all active duty into PRIME enrollees.
   *************************************************************************;
   IF BENCAT = 1 AND PRIME = 3 THEN PRIME = 1;

   IF CACSMPL IN (4,35,58,122,250,606,633,638,806,9913,9914)
              AND BENCAT = 4 THEN BENCAT = 2;

   IF CACSMPL IN (4,15,35,58,69,85,96,100,122,250,326,616,806,9913,9999)
              AND BENCAT = 2 THEN PRIME = 1;

   IF CACSMPL = 9912 and BENCAT = 1 THEN BENCAT = 2;

   IF CACSMPL IN (15,50,74,96,100,326) AND BENCAT = 4 THEN DO;
    BENCAT = 2;
    PRIME = 1;
   END;

   IF CACSMPL IN(36,51,76,107,119,128) AND BENCAT = 2 THEN PRIME = 1;

   IF CACSMPL = 118 AND PRIME = 3 THEN DO;
    BENCAT = 2;
    PRIME = 1;
   END;

   LABEL PRIME  = "Prime Indicator"
         BENCAT = "Benficiary Category";
RUN;

PROC SORT DATA=PRIME(KEEP= MPRID FNSTATUS BWT99 &DOMAIN1 &DOMAIN2 &DOMAIN3 POSTSTR)
```

```
          OUT=&INPT(KEEP= MPRID FNSTATUS BWT99 &DOMAIN1 &DOMAIN2 &DOMAIN3 POSTSTR)
          TAGSORT;
BY &DOMAIN1 &DOMAIN2 &DOMAIN3;
RUN;

*************************************************************************
* Calculate adjustment factor A1 for each cell
*************************************************************************;
DATA CELLSA1  (KEEP=SUMBWT SUMG1-SUMG3 A1 CELLNAME CELLCNT)
     MPRIDSA1 (KEEP=CELLNAME MPRID FNSTATUS BWT99 &DOMAIN1 &DOMAIN2 &DOMAIN3)
    ;
   SET &INPT;
   BY &DOMAIN1 &DOMAIN2 &DOMAIN3;

   LENGTH CELLNAME $25;
   CELLNAME = PUT(&DOMAIN1,5.) || PUT(&DOMAIN2,5.) || PUT(&DOMAIN3,5.);

   IF FIRST.&DOMAIN1 THEN DO;
      CELLCNT = 0;
      SUMBWT  = 0.0;
      SUMG1   = 0.0;
      SUMG2   = 0.0;
      SUMG3   = 0.0;
      A1 = 0.0;
   END;
   CELLCNT + 1;

   ****************************************************
   * Accumulate total weight sum
   ****************************************************;
   SUMBWT + BWT99;
   ****************************************************
   * Accumulate group 1 weight sum
   ****************************************************;
   IF FNSTATUS IN(11,12) THEN SUMG1 + BWT99;
   ****************************************************
   * Accumulate group 2 weight sum
   ****************************************************;
   ELSE IF FNSTATUS = 20 THEN SUMG2 + BWT99;
   ****************************************************
   * Accumulate group 3 weight sum
   ****************************************************;
   ELSE IF FNSTATUS = 30 THEN SUMG3 + BWT99;

   RETAIN SUMBWT SUMG1-SUMG3 A1 CELLNAME CELLCNT MPRID;

   IF LAST.&DOMAIN3 THEN DO;
      A1 = SUMBWT/(SUMG1 + SUMG2 + SUMG3);
      OUTPUT CELLSA1;
      CELLCNT = 0;
      SUMBWT  = 0.0;
      SUMG1   = 0.0;
      SUMG2   = 0.0;
      SUMG3   = 0.0;
   END; * DOMAIN;
   OUTPUT MPRIDSA1;
RUN;

*************************************************************************
* Calculate adjustment factor A2 for each cell
*************************************************************************;
DATA CELLSA2 (KEEP=CELLNAME CELLCNT A1 A2 NUMER DENOM);
   MERGE MPRIDSA1 CELLSA1;
   BY CELLNAME;

   IF FIRST.CELLNAME THEN DO;
      A2 = 0.0;
      NUMER = 0.0;
      DENOM = 0.0;
   END;
   RETAIN NUMER DENOM A2;

   IF FNSTATUS IN(11,12) OR FNSTATUS = 20 THEN NUMER + BWT99*A1;
```

```
   IF FNSTATUS  = 11 THEN DENOM + BWT99*A1;

   IF LAST.CELLNAME THEN DO;
      A2 = NUMER/DENOM;
      OUTPUT CELLSA2;
   END;
RUN;

***************************************************************************
* Calculate Adjusted Weight
***************************************************************************;
DATA WEIGHT03;
   MERGE CELLSA2 MPRIDSA1;
   BY CELLNAME;
   IF FNSTATUS = 11 THEN
      AWT = A1*A2*BWT99;
   ELSE IF FNSTATUS IN(12,20,41,42) THEN
      AWT = 0;
   ELSE IF FNSTATUS = 30 THEN
      AWT = A1*BWT99;
   LABEL AWT = 'Adjusted Weight Factor';
   KEEP MPRID AWT A1 A2;
RUN;

***************************************************************************
* Sort the original data
***************************************************************************;
PROC SORT DATA=&INPT OUT=&INPT TAGSORT; BY MPRID; RUN;

***************************************************************************
* Sort the adjusted weight (AWT) data
***************************************************************************;
PROC SORT DATA=WEIGHT03 TAGSORT; BY MPRID; RUN;

***************************************************************************
* Append adjusted weight (AWT) variable
***************************************************************************;
DATA OUT.WEIGH03&FORM;
   MERGE WEIGHT03 &INPT;
   BY MPRID;
RUN;

TITLE1 "1999 DOD SURVEY OF HEALTH BENES - FORM &FORM";
TITLE2 "PROGRAM: WEIGHT03.SAS, INPUT: KEEPERS&FORM..SD2, OUTPUT: WEIGH03&FORM..SD2";

TITLE3 'DOMAIN TABULATIONS - UNWEIGHTED WHERE FNSTATUS=11';
PROC TABULATE DATA=OUT.WEIGH03&FORM NOSEPS;
CLASS &DOMAIN1 &DOMAIN2 &DOMAIN3;
WHERE FNSTATUS = 11;
TABLE  &DOMAIN2*&DOMAIN3 ALL, F=12.0*(&DOMAIN1 ALL);
RUN;


TITLE3 'DOMAIN TABULATIONS - WEIGHTED BY AWT';
PROC TABULATE DATA=OUT.WEIGH03&FORM NOSEPS;
CLASS &DOMAIN1 &DOMAIN2 &DOMAIN3;
VAR AWT;
TABLE  &DOMAIN2*&DOMAIN3 ALL, SUM*F=12.0*AWT*(&DOMAIN1 ALL);
RUN;

TITLE3 'DOMAIN TABULATIONS - WEIGHTED BY BWT99';
PROC TABULATE DATA=OUT.WEIGH03&FORM NOSEPS;
CLASS &DOMAIN1 &DOMAIN2 &DOMAIN3;
VAR BWT99;
TABLE  &DOMAIN2*&DOMAIN3 ALL, SUM*F=12.0*BWT99*(&DOMAIN1 ALL);
RUN;

TITLE3 'WEIGHT TABULATIONS - UNWEIGHTED WHERE FNSTATUS=11';
PROC SORT DATA=OUT.WEIGH03&FORM OUT=WEIGH03&FORM; BY &DOMAIN1 &DOMAIN2 &DOMAIN3; RUN;

PROC MEANS DATA=WEIGH03&FORM NOPRINT;
BY &DOMAIN1 &DOMAIN2 &DOMAIN3;
WHERE FNSTATUS = 11;
```

```
VAR A1 A2;
OUTPUT OUT=TEMP MEAN=A1MEAN A2MEAN N=NA1 NA2;
RUN;
PROC PRINT; RUN;
%MEND PROCESS;
*************************************************************************
* Calculate adjusted weight based on user-specified parameters.
*************************************************************************;
%PROCESS(BENCAT,CACSMPL,PRIME,A,KEEPERSA);
*%PROCESS(xxxxxxx,xxxxxxx,C,KEEPERSC);
RUN;

*************************************************************************
*
* PROGRAM: WEIGHT04.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8676-200)
* PURPOSE: BUILD AND ASSIGN FINAL WEIGHTS - POST STRATIFICATION - FORM A/C.
*          WEIGHTS FOR DOD SURVEY.
*          DOD HEALTH CARE SURVEY FILE.
*          REQUESTED BY DON JANG.
* WRITTEN: 12/30/99 BY KEITH RATHBUN
*
* INPUTS:   1) WEIGH03A.SD2 - Adjusted Weights file - Form A
*           2) WEIGH03C.SD2 - Adjusted Weights file - Form C
*
* OUTPUTS   1) WEIGH04A.SD2 - Final Weights file - Form A
*
*************************************************************************
*;
LIBNAME IN   "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT  "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY  "D:\KEITH\DOD99-8676\DATA\FMTLIB";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

%MACRO PROCESS(DOMAIN=,FORM=);

************************************************************************
* Sort the adjusted weights file by user-specified domains
************************************************************************;
PROC SORT DATA=IN.WEIGH03&FORM
          OUT=WEIGHT03(KEEP=FNSTATUS MPRID AWT &DOMAIN)
         ;
     BY &DOMAIN;
RUN;

************************************************************************
* Assign cell names and calculate the sum of AWT
************************************************************************;
DATA CELLS  (KEEP=SUMAWT SUMFN11 &DOMAIN)
     MPRIDS (KEEP=MPRID FNSTATUS AWT &DOMAIN)
    ;
   SET WEIGHT03;
   BY &DOMAIN;

   IF FIRST.&DOMAIN THEN DO;
      SUMAWT   = 0.0;
      SUMFN11  = 0;
   END;

   ********************************************************************
   * Accumulate sum of adjusted weight
   ********************************************************************;
   SUMAWT + AWT;

   ********************************************************************
   * COUNT the FNSTATUS = 11 within each DOMAIN
   ********************************************************************;
   IF FNSTATUS = 11 THEN SUMFN11 + 1;

   RETAIN SUMAWT SUMFN11;

   IF LAST.&DOMAIN THEN DO;
      OUTPUT CELLS;
```

```
      SUMAWT  = 0.0;
      SUMFN11  = 0;
   END; * DOMAIN;
   OUTPUT MPRIDS;
RUN;

************************************************************************
* Merge the population counts and calculate the adjusted population (AP)
************************************************************************;
PROC SORT DATA=IN.COUNT&FORM OUT=COUNT&FORM; BY &DOMAIN; RUN;

DATA AP;
   MERGE COUNT&FORM CELLS;
   BY &DOMAIN;
   AP = POP/SUMAWT;
RUN;

************************************************************************
* Merge the adjusted population and calculate the final weight (WRWT99)
************************************************************************;
DATA WEIGHT04;
   MERGE AP(IN=IN1) MPRIDS(IN=IN2);
   BY &DOMAIN;

   IF IN2 THEN DO;
      WRWT99 = AP*AWT;
      OUTPUT;
   END;

   LABEL WRWT99   = 'Final Weight';
   LABEL AP       = 'Adjusted population';
   LABEL POP      = 'DEERS population by CELLNAME for weights';
   LABEL SUMFN11  = 'COUNT of FNSTATUS=11 within CELLNAME';

   KEEP FNSTATUS WRWT99 AWT AP MPRID POP SUMFN11 &DOMAIN;
RUN;

TITLE1 "1999 DOD SURVEY OF HEALTH BENES - FORM &FORM";
TITLE2 "PROGRAM: WEIGHT04.SAS, INPUT: WEIGH03&FORM..SD2, OUTPUT: WEIGH04&FORM..SD2";
TITLE3 "WEIGHT TABULATIONS - UNWEIGHTED";

PROC MEANS DATA=WEIGHT04 NOPRINT;
   VAR POP WRWT99 AP SUMFN11;
   BY &DOMAIN;
   OUTPUT OUT=STATS(KEEP=POSTSTR  DEERSPOP POPCNT APM FN11CNT)
                    SUM= DUMMY1   POPCNT  DUMMY2  DUMMY3
                    MEAN=DUMMY4   DUMMY5  APM     DUMMY6
                    MAX= DEERSPOP DUMMY7  DUMMY8 FN11CNT;
RUN;

PROC PRINT;
   SUM DEERSPOP POPCNT FN11CNT;
RUN;

************************************************************************
* Sort the original data and append the final weight (WRWT99)
************************************************************************;
PROC SORT DATA=IN.WEIGH03&FORM OUT=WEIGHT03 TAGSORT; BY MPRID; RUN;
PROC SORT DATA=WEIGHT04 TAGSORT; BY MPRID; RUN;

DATA OUT.WEIGH04&FORM;
   MERGE WEIGHT03 WEIGHT04;
   BY MPRID;
RUN;

%MEND;

%PROCESS(DOMAIN=POSTSTR,FORM=A);
*%PROCESS(DOMAIN=xxxxxxx,FORM=C);



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

```
*
* PROGRAM: WEIGHT05.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8676-200)
* PURPOSE: BUILD AND ASSIGN JK WEIGHTS - POST STRATIFICATION - FORM A/C.
*          WEIGHTS FOR DOD SURVEY.
*          DOD HEALTH CARE SURVEY FILE.
*          REQUESTED BY DON JANG.
* WRITTEN: 12/30/99 BY KEITH RATHBUN
*
* INPUTS:   1) WEIGH04A.SD2 - Final Weights file - Form A
*           2) WEIGH04C.SD2 - Final Weights file - Form C
*
* OUTPUTS   1) WEIGH05A.SD2 - JackKnife (JK) Weights file - Form A
*           2) WEIGH05C.SD2 - JackKnife (JK) Weights file - Form C
*
*****************************************************************************
*;
LIBNAME IN   "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT  "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY  "D:\KEITH\DOD99-8676\DATA\FMTLIBA";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=0 NOCENTER;

%MACRO PROCESS(DOMAIN1,DOMAIN2,DOMAIN3,DOMAIN4,FORM);

*********************************************************************
* Sort the final weights file by user-specified domains
*********************************************************************;
PROC SORT DATA=IN.WEIGH04&FORM
          OUT=WEIGHT04(KEEP=FNSTATUS MPRID BWT99 &DOMAIN1 &DOMAIN2 &DOMAIN3
                       &DOMAIN4)
         ;
     BY &DOMAIN1 &DOMAIN2 &DOMAIN3;
RUN;

*********************************************************************
* Append SUBSET index (I) to each observation
*********************************************************************;
DATA SUBSETS;
   SET WEIGHT04;
   BY &DOMAIN1 &DOMAIN2 &DOMAIN3;

   IF _N_ = 1 OR MOD(_N_-1,40) = 0 THEN SUBSET = 1;
   ELSE SUBSET + 1;

   RETAIN SUBSET;
   BBWT = BWT99*(40/39);
RUN;

*********************************************************************
*********************************************************************
* Generate JackKnife/replicated weights WRWT01-WRWT40
*********************************************************************
*********************************************************************;
%DO I = 1 %TO 40;

DATA SUBSET;
   SET SUBSETS;
   IF &I = SUBSET THEN DELETE; *Remove the current subset;
RUN;

*********************************************************************
* Calculate adjustment factor A1 for each cell
*********************************************************************;
DATA CELLSA1  (KEEP=SUMBBWT SUMG1-SUMG3 A1 CELLNAME CELLCNT)
     MPRIDSA1 (KEEP=CELLNAME MPRID FNSTATUS BBWT &DOMAIN1 &DOMAIN2 &DOMAIN3
               &DOMAIN4)
    ;
   SET SUBSET;
   BY &DOMAIN1 &DOMAIN2 &DOMAIN3;

   LENGTH CELLNAME $25;
   CELLNAME = PUT(&DOMAIN1,5.) || PUT(&DOMAIN2,5.) || PUT(&DOMAIN3,5.);
```

```
   IF FIRST.&DOMAIN1 THEN DO;
      CELLCNT = 0;
      SUMBBWT = 0.0;
      SUMG1   = 0.0;
      SUMG2   = 0.0;
      SUMG3   = 0.0;
      A1 = 0.0;
   END;
   CELLCNT + 1;

   ****************************************************
   * Accumulate total weight sum
   ****************************************************;
   SUMBBWT + BBWT;
   ****************************************************
   * Accumulate group 1 weight sum
   ****************************************************;
   IF FNSTATUS IN(11,12) THEN SUMG1 + BBWT;
   ****************************************************
   * Accumulate group 2 weight sum
   ****************************************************;
   ELSE IF FNSTATUS = 20 THEN SUMG2 + BBWT;
   ****************************************************
   * Accumulate group 3 weight sum
   ****************************************************;
   ELSE IF FNSTATUS = 30 THEN SUMG3 + BBWT;

   RETAIN SUMBBWT SUMG1-SUMG3 A1 CELLNAME CELLCNT MPRID;

   IF LAST.&DOMAIN3 THEN DO;
      A1 = SUMBBWT/(SUMG1 + SUMG2 + SUMG3);
      OUTPUT CELLSA1;
      CELLCNT = 0;
      SUMBBWT = 0.0;
      SUMG1   = 0.0;
      SUMG2   = 0.0;
      SUMG3   = 0.0;
   END; * DOMAIN;
   OUTPUT MPRIDSA1;
RUN;

*************************************************************************
* Calculate adjustment factor A2 for each cell
*************************************************************************;
DATA CELLSA2 (KEEP=CELLNAME CELLCNT A1 A2 NUMER DENOM);
   MERGE MPRIDSA1 CELLSA1;
   BY CELLNAME;

   IF FIRST.CELLNAME THEN DO;
      A2 = 0.0;
      NUMER = 0.0;
      DENOM = 0.0;
   END;
   RETAIN NUMER DENOM A2;

   IF FNSTATUS IN(11,12) OR FNSTATUS = 20 THEN NUMER + BBWT*A1;
   IF FNSTATUS  = 11 THEN DENOM + BBWT*A1;

   IF LAST.CELLNAME THEN DO;
      A2 = NUMER/DENOM;
      OUTPUT CELLSA2;
   END;
RUN;

*************************************************************************
* Calculate Adjusted Weight
*************************************************************************;
DATA ADJWGT;
   MERGE CELLSA2 MPRIDSA1;
   BY CELLNAME;
   IF FNSTATUS = 11 THEN
      AWT = A1*A2*BBWT;
   ELSE IF FNSTATUS IN(12,20,41,42) THEN
```

```
      AWT = 0;
   ELSE IF FNSTATUS = 30 THEN
      AWT = A1*BBWT;
   KEEP MPRID FNSTATUS AWT BBWT &DOMAIN1 &DOMAIN2 &DOMAIN3 &DOMAIN4;
RUN;

************************************************************************
* Begin final weight code
************************************************************************
* Assign cell names and calculate the sum of AWT
************************************************************************;

PROC SORT DATA=ADJWGT; BY &DOMAIN4; RUN;

DATA CELLS  (KEEP=SUMAWT &DOMAIN4)
     MPRIDS (KEEP=MPRID FNSTATUS AWT &DOMAIN1 &DOMAIN2 &DOMAIN3
            &DOMAIN4)
    ;
   SET ADJWGT;
   BY &DOMAIN4;

   IF FIRST.&DOMAIN4 THEN DO;
      SUMAWT  = 0.0;
   END;

   ***********************************************************************
   * Accumulate sum of adjusted weight
   ***********************************************************************;
   SUMAWT + AWT;

   RETAIN SUMAWT;

   IF LAST.&DOMAIN4 THEN DO;
      OUTPUT CELLS;
      SUMAWT  = 0.0;
   END; * DOMAIN;
   OUTPUT MPRIDS;
RUN;

************************************************************************
* Merge the population counts and calculate the adjusted population (AP)
************************************************************************;
PROC SORT DATA=IN.COUNT&FORM OUT=COUNT&FORM; BY &DOMAIN4; RUN;

DATA AP;
   MERGE COUNT&FORM CELLS;
   BY &DOMAIN4;
   AP = POP/SUMAWT;
RUN;

************************************************************************
* Merge the adjusted population and calculate JackKnife Weights
* (WRWT1-WRWT40)
************************************************************************;
DATA SUBSET&I(KEEP=MPRID SUBSET JKWEIGHT);
   MERGE AP(IN=IN1) MPRIDS(IN=IN2);
   BY &DOMAIN4;

   SUBSET = &I;
   IF IN2 THEN DO;
      JKWEIGHT = AP*AWT;
      OUTPUT;
   END;

RUN;
PROC SORT DATA=SUBSET&I; BY MPRID; RUN;

************************************************************************
************************************************************************
* End of JackKnife/replicated weights WRWT01-WRWT40 assignments
************************************************************************
************************************************************************;
%END;
```

```
************************************************************************
* Combine all of the JackKnife weight subsets by MPRID
************************************************************************;
DATA ALLSETS;
   SET SUBSET1   SUBSET2   SUBSET3   SUBSET4   SUBSET5
       SUBSET6   SUBSET7   SUBSET8   SUBSET9   SUBSET10
       SUBSET11  SUBSET12  SUBSET13  SUBSET14  SUBSET15
       SUBSET16  SUBSET17  SUBSET18  SUBSET19  SUBSET20
       SUBSET21  SUBSET22  SUBSET23  SUBSET24  SUBSET25
       SUBSET26  SUBSET27  SUBSET28  SUBSET29  SUBSET30
       SUBSET31  SUBSET32  SUBSET33  SUBSET34  SUBSET35
       SUBSET36  SUBSET37  SUBSET38  SUBSET39  SUBSET40
   ;
   BY MPRID;
   ARRAY JKWT(40) WRWT1-WRWT40; RETAIN WRWT1-WRWT40;
   IF FIRST.MPRID THEN DO;
      DO I = 1 TO 40; DROP I;
         JKWT(I) = . ;
      END;
   END;
   JKWT(SUBSET) = JKWEIGHT;
   IF LAST.MPRID THEN OUTPUT;
   KEEP MPRID WRWT1-WRWT40;
RUN;

************************************************************************
* Sort the original data, get the final weight (WRWT99), append the
* JackKnife/Replicated weights (WRWT1-WRWT40), and label variables.
************************************************************************;
PROC SORT DATA=IN.WEIGH04&FORM
          OUT=WEIGHT04;
     BY MPRID;
RUN;

DATA OUT.WEIGH05&FORM;
   MERGE WEIGHT04 ALLSETS;
   BY MPRID;
   LABEL
      MPRID  = 'MPR ID Number'
      WRWT1  = 'Replicated/JackKnife Weight 1'
      WRWT2  = 'Replicated/JackKnife Weight 2'
      WRWT3  = 'Replicated/JackKnife Weight 3'
      WRWT4  = 'Replicated/JackKnife Weight 4'
      WRWT5  = 'Replicated/JackKnife Weight 5'
      WRWT6  = 'Replicated/JackKnife Weight 6'
      WRWT7  = 'Replicated/JackKnife Weight 7'
      WRWT8  = 'Replicated/JackKnife Weight 8'
      WRWT9  = 'Replicated/JackKnife Weight 9'
      WRWT10 = 'Replicated/JackKnife Weight 10'
      WRWT11 = 'Replicated/JackKnife Weight 11'
      WRWT12 = 'Replicated/JackKnife Weight 12'
      WRWT13 = 'Replicated/JackKnife Weight 13'
      WRWT14 = 'Replicated/JackKnife Weight 14'
      WRWT15 = 'Replicated/JackKnife Weight 15'
      WRWT16 = 'Replicated/JackKnife Weight 16'
      WRWT17 = 'Replicated/JackKnife Weight 17'
      WRWT18 = 'Replicated/JackKnife Weight 18'
      WRWT19 = 'Replicated/JackKnife Weight 19'
      WRWT20 = 'Replicated/JackKnife Weight 20'
      WRWT21 = 'Replicated/JackKnife Weight 21'
      WRWT22 = 'Replicated/JackKnife Weight 22'
      WRWT23 = 'Replicated/JackKnife Weight 23'
      WRWT24 = 'Replicated/JackKnife Weight 24'
      WRWT25 = 'Replicated/JackKnife Weight 25'
      WRWT26 = 'Replicated/JackKnife Weight 26'
      WRWT27 = 'Replicated/JackKnife Weight 27'
      WRWT28 = 'Replicated/JackKnife Weight 28'
      WRWT29 = 'Replicated/JackKnife Weight 29'
      WRWT30 = 'Replicated/JackKnife Weight 30'
      WRWT31 = 'Replicated/JackKnife Weight 31'
      WRWT32 = 'Replicated/JackKnife Weight 32'
      WRWT33 = 'Replicated/JackKnife Weight 33'
```

```
      WRWT34 = 'Replicated/JackKnife Weight 34'
      WRWT35 = 'Replicated/JackKnife Weight 35'
      WRWT36 = 'Replicated/JackKnife Weight 36'
      WRWT37 = 'Replicated/JackKnife Weight 37'
      WRWT38 = 'Replicated/JackKnife Weight 38'
      WRWT39 = 'Replicated/JackKnife Weight 39'
      WRWT40 = 'Replicated/JackKnife Weight 40'
   ;
RUN;

TITLE1 "1999 DOD Health Survey Final/Replicated Weights (8676-200)";
TITLE2 "Program Name: WEIGHT05.SAS By Keith Rathbun";
TITLE3 "Program Output: WEIGH05&FORM..SD2";

PROC CONTENTS DATA=OUT.WEIGH05&FORM;
PROC MEANS DATA=OUT.WEIGH05&FORM;
VAR  WRWT99 WRWT1-WRWT40;
RUN;

PROC SORT DATA=OUT.WEIGH05&FORM; BY &DOMAIN4; RUN;
PROC MEANS DATA=OUT.WEIGH05&FORM NOPRINT;
   VAR POP WRWT1;
   BY &DOMAIN4;
   OUTPUT OUT=STATS(KEEP=&DOMAIN4 DEERSPOP POPCNT)
                    SUM= DUMMY1   POPCNT
                    MAX= DEERSPOP DUMMY2;
RUN;
PROC PRINT; RUN;
PROC PRINT DATA=OUT.WEIGH05&FORM(OBS=50);
VAR MPRID BWT99 AWT WRWT99 WRWT1-WRWT40;
RUN;
%MEND;

%PROCESS(BENCAT,CACSMPL,PRIME,POSTSTR,A);
*%PROCESS(xxxxxxx,xxxxxxx,xxxxxxx,xxxxx,C);

*************************************************************************
*
* PROGRAM: FINAL02.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8687-100)
* PURPOSE: RECODE MISSING REPLICATED WEIGHTS TO ZERO.
*          DOD HEALTH CARE SURVEY FILE.
* WRITTEN: 04/10/2000 BY KEITH RATHBUN
*
* INPUTS:  1) WEIGH05A.SD2 - Replicated Weights file - Form A
*          2) WEIGH05C.SD2 - Replicated Weights file - Form C
*
* OUTPUTS: 1) FINAL02A.SD2 - Replicated Weights file with recodes - Form A
*          2) FINAL02C.SD2 - Replicated Weights file with recodes - Form C
*
*************************************************************************
*;
LIBNAME IN      "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT     "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY "D:\KEITH\DOD99-8676\DATA\FMTLIBA";
OPTIONS PS=79 LS=132 COMPRESS=YES ERRORS=2 NOCENTER;

%MACRO PROCESS(FORM);

PROC SORT DATA=OUT.WEIGH05&FORM; BY MPRID; RUN;
DATA OUT.FINAL02&FORM;
   SET OUT.WEIGH05&FORM;
   BY MPRID;

   ARRAY WGTS(40) WRWT1-WRWT40;
   DO I = 1 TO 40; DROP I;
      IF WGTS(I) EQ . THEN WGTS(I) = 0;
   END;

   KEEP MPRID BWT99 WRWT99 WRWT1-WRWT40;
RUN;

TITLE1 "1999 DOD SURVEY OF HEALTH BENES - FORM &FORM";
```

```
TITLE2 "PROGRAM: FINAL02.SAS, OUTPUT: FINAL02&FORM..SD2";

PROC CONTENTS DATA=OUT.FINAL02&FORM; RUN;

PROC MEANS DATA=OUT.FINAL02&FORM;
VAR BWT99 WRWT99 WRWT1-WRWT40;
RUN;

%MEND PROCESS;

%PROCESS(A);
*%PROCESS(C);
RUN;
```

10. FINAL MERGE

```
***************************************************************************
*
* PROGRAM: ADDWGTS.SAS
* TASK:    1999 DOD HEALTH CARE SURVEY ANALYSIS (8687-100)
* PURPOSE: MERGE THE FINAL WEIGHTS FILE WITH THE FINAL
*          QUESTIONNAIRE/SAMPLE FILE
*
* WRITTEN: 02/24/2000 BY KEITH RATHBUN
*
* INPUTS:  1) FINAL02A.SD2 - Final/Replicated Weights file - FORM A
*          2) FINAL02C.SD2 - Final/Replicated Weights file - FORM C
*          3) HCS99A_n.SD2 - Final FORM A Questionnaire/Sample File
*          4) HCS99C_n.SD2 - Final FORM C Questionnaire/Sample File
*             where n = final dataset suffix
*
* OUTPUTS: 1) HCS99A_n+1.SD2 - Final FORM A Questionnaire/Sample File
*             combined with Final/Replicated Weights file - FORM A
*          2) HCS99C_n+1.SD2 - Final FORM C Questionnaire/Sample File
*             combined with Final/Replicated Weights file - FORM C
*             where n+1 = final dataset suffix
*
***************************************************************************
;
LIBNAME IN  "D:\KEITH\DOD99-8676\DATA";
LIBNAME OUT "D:\KEITH\DOD99-8676\DATA";
LIBNAME LIBRARY "D:\KEITH\DOD99-8676\DATA\FMTLIBA";

OPTIONS PS=79 LS=132 COMPRESS=YES NOCENTER;

%MACRO PROCESS(DSNI_1=,DSNI_2=,DSNO=);
***************************************************************************
* Merge the final weights file with the final Questionnaire/Sample file
***************************************************************************;
PROC SORT DATA=IN.&DSNI_1 OUT=&DSNI_1; BY MPRID; RUN;
PROC SORT DATA=IN.&DSNI_2 OUT=&DSNI_2; BY MPRID; RUN;

DATA OUT.&DSNO;
   MERGE &DSNI_1(IN=IN1)
         &DSNI_2(IN=IN2);
   BY MPRID;
   IF IN1 AND IN2;
   LABEL KEYCOUNT = "# of Key Questions Answered";
RUN;

TITLE1 "1999 DOD Health Survey Post-Stratification (8687-100)";
TITLE2 "Program Name: ADDWGTS.SAS By Keith Rathbun";
TITLE3 "Program Inputs: &DSNI_1..SD2 -- &DSNI_2..SD2";
TITLE3 "Program Outputs: &DSNO..SD2";
PROC CONTENTS; RUN;

%MEND PROCESS;

%PROCESS(DSNI_1=FINAL02A, DSNI_2=HCS99A_4, DSNO=HCS99A_5)
*%PROCESS(DSNI_1=FINAL02C, DSNI_2=HCS99C_n, DSNO=HCS99C_n+1);
```