

```

options notes nosource;
proc datasets lib=work nolist memtype=data kill; quit;
* macro to save output and log to appropriate areas ;
%_mprintto;
%put NOTE:
=====;
%put NOTE: Covance Study Number : 000000106326;
%put NOTE: Client Protocol ID : ZRHM-PK-05-JP;
%put NOTE: Program Name : d_2ADVS.sas;
%put NOTE: Purpose : create ADVS dataset;
%put NOTE: ;
%put NOTE: Input Data : STDLIB.ADVS SDTM.VS SDTM.SUPPVS;
%put NOTE: Output : ADAM.ADVS;
%put NOTE: Macros Called : _MPRINTTO _MTOTPER _MPERALL _SCRAMBLE;
%put NOTE: ;
%put NOTE: Programmed by : cvn_jhardman;
%put NOTE: Creation Date : 2014-01-03;
%put NOTE: SAS Version : 9.3;
%put NOTE: ;
%put NOTE: == Latest Run
=====;
%put NOTE: Run by : &sysuserid;
%put NOTE: Date/Time :
%sysfunc(putn(%sysfunc(date()),e8601da.))T%sysfunc(putn(%sysfunc(time()),
e86011z.));
%put NOTE: ;
%put NOTE: == Modification History
=====;
%put NOTE: Date Initials No. Reason;
%put NOTE: 14Jan2014 KB 1) Added in a warning for mapping of
VSSTRESC;
%put NOTE: 14Jan2014 KB 2) Added a check for the position of
SYSBP DISBP and PULSE for ANL01FL;
%put NOTE: 14Jan2014 KB 3) Amended due to no longer needed
previous code;
%put NOTE: 14Jan2014 KB 4) Amended PARAMN of VSALL and added in
SYSBP DIABP and PULSE;
%put NOTE: 14Jan2014 KB 5) Amended update 2;
%put NOTE: 15Jan2014 JMH 6) Amended code for the position of
SYSBP DISBP and PULSE for ANL01FL;
%put NOTE: 14Apr2014 KB 7) Added derived BMI;
%put NOTE: 14Apr2014 KB 8) Removed format for VSSEQ;
%put NOTE: 14Apr2014 KB 9) Amended ABLFL;
%put NOTE: 14Apr2014 KB 10) Added PARAMCDs for all vitals
different positions;
%put NOTE: 14Apr2014 KB 11) Amended PARAMs for all vitals;
%put NOTE: 14Apr2014 KB 12) Added TRTSTMF TR01STMF and TR02STMF
to scramble macro;
%put NOTE: 06Aug2014 KB 13) Added EXNOTRFL;
%put NOTE: 06Aug2014 KB 14) Removed format from SMOK15P;
%put NOTE: 06Aug2014 KB 15) Amended change in day 4;
%put NOTE: 06Aug2014 KB 16) Amended format issue;
%put NOTE: 06Aug2014 KB 17) Amended lengths for merge;
%put NOTE: 21Sep2014 KB 18) Amended ABLFL slightly;

```

```

%put NOTE: 21Sep2014    KB                19) Added AVALCAT1 for BMI;
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

*=====;
* START OF PROGRAM CODE                                ;
*=====;
*****;
* bring in ADSL ;
*****;

data adsl;
    set adam.adsl;
    keep studyid usubjid subjid: siteid age sex: race height weightb1
bmi ucpdgr: nicogr: targr:
        enrfl scrffl complfl saffl randfl trt: tr01: tr02: dthfl
enfl exfl fupfl anal: EXNOTRFL; /* 13) KB 06Aug2014 */
run;

proc sort data = adsl;
    by usubjid;
run;

*****;
* pick up SUPPVS ;
*****;

proc transpose data = sdtm.suppvs out = suppvs prefix=v;
    var qval;
    by usubjid idvarval;
    id qnam;
run;

data suppvs2(drop = _: vsmok15p idvarval);
    set suppvs;
/*    format*/ /*vsseq 8.*/*/* smok15p $1.*/* /* 8) KB 14Apr2014 */ /*
14) KB 06Aug2014 */
    vsseq = input(idvarval,best.);
    smok15p = trim(vsmok15p);
run;

proc sort data = suppvs2;
    by usubjid vsseq;
run;

*****;
* bring in VS ;
*****;

proc sort data = sdtm.vs out = vs;
    by usubjid vsseq;

```

```

run;

/* 7) START KB 14Apr2014 */
DATA DBMI;
    SET VS(WHERE=(VSTESTCD IN ('HEIGHT' 'WEIGHT') AND VSSTAT NE 'NOT
DONE'));
    FORMAT PARAMCD1 $8.;

    IF VISIT='DAY -1' THEN PARAMCD1='WEIGHT1';
    ELSE IF VISIT=/'DAY 4'/'DAY 4/DISCHARGE' THEN PARAMCD1='WEIGHT4';
/* 15) KB 06Aug2014 */
    ELSE PARAMCD1=VSTESTCD;

    KEEP USUBJID PARAMCD1 VISIT VSSTRESN ;
RUN;

PROC SORT DATA=DBMI;
    BY USUBJID;
RUN;

PROC TRANSPOSE DATA=DBMI OUT=DBMI2(DROP=_:);
    BY USUBJID;
    VAR VSSTRESN;
    ID PARAMCD1;
RUN;

DATA DBMI3;
    SET DBMI2;
    FORMAT DBMI1 DBMI4 BEST.;

    DBMI1=WEIGHT1/((HEIGHT/100)**2);
    DBMI4=WEIGHT4/((HEIGHT/100)**2);
RUN;

PROC TRANSPOSE DATA=DBMI3 OUT=DBMI4;
    BY USUBJID;
    VAR DBMI1 DBMI4;
RUN;

DATA DBMI5;
    SET DBMI4;
    FORMAT VSSTRESN BEST. VISITNUM VSTPTNUM 8.;
    LENGTH VISIT VSTPT /*$40*/$15 VSTESTCD /*$8*/$6 VSSTRESC /*$200*/$5;
/* 17) KB 06Aug2014 */

    IF COL1='' THEN DELETE;

    VSTESTCD=_NAME_;
    VSSTRESN=PUT(COL1,8.1);
    VSSTRESC=COMPRESS(PUT(VSSTRESN,8.1));

    IF VSTESTCD='DBMI1' THEN DO;
        VISIT='DAY -1';
        VISITNUM=99;

```

```

        VSTPT=VISIT;
        VSTPTNUM=1;
    END;
    ELSE IF VSTESTCD='DBMI4' THEN DO;
        VISIT=/'DAY 4'/'DAY 4/DISCHARGE'; /* 15) KB 06Aug2014 */
        VISITNUM=104;
        VSTPT=VISIT;
        VSTPTNUM=6;
    END;

    VSTESTCD='DBMI';

    DROP _NAME_ COL1;
RUN;

DATA TIMES;
    SET SDTM.VS (WHERE=(VSTESTCD='WEIGHT' AND VISIT IN ('DAY -1' /'DAY
4'/'DAY 4/DISCHARGE') AND VSSTAT NE 'NOT DONE')); /* 15) KB 06Aug2014
*/

    KEEP USUBJID VISIT VSDTC;
RUN;

DATA DBMI5A;
    MERGE DBMI5 TIMES;
    BY USUBJID VISIT;
RUN;

DATA DBMI6;
    SET VS DBMI5A;
RUN;

PROC SORT DATA=DBMI6;
    BY USUBJID VSSEQ;
RUN;
/* 7) END KB 14Apr2014 */

data vs2;
    merge /*vs*/DBMI6 suppv2; /* 7) KB 14Apr2014 */
    by usubjid vsseq;
    format avisitn paramn 8. aval atptn best. paramcd $8. param avisit
$40. atpt $50. AVALCAT1 $40. /* 19) KB 21Sep2014 */
        avalc $200. avalu $20. /*ablfl $1.*/ adt date9. atm time5.
adtm datetime13. PARAMTYP DTYPE $10.; /* 7) KB 14Apr2014 */ /* 9) KB
14Apr2014 */
        * parameters ;
        if vstestcd not in ('WEIGHT' 'HEIGHT' 'BMI' /*'VSALL'*/ 'DBMI')
then do; /* 7) KB 14Apr2014 */ /* 10) KB 14Apr2014 */
            paramcd = compress(substr(vspos,1,2) || trim(vstestcd));
            param = propcase(trim(vspos) || ' ' ||trim(vstest));
        end;
    else do;
        paramcd = left(trim(vstestcd));
        param = propcase(trim(vstest));
    end;

```

```

        end;
        IF PARAMCD='DBMI' THEN PARAM='Body Mass Index (Derived)'; /* 7) KB
14Apr2014 */

```

```

/* 11) START KB 14Apr2014 */
    IF PARAMCD='SUVSALL' THEN PARAM='All Supine Vital Signs';
    ELSE IF PARAMCD='SIVSALL' THEN PARAM='All Sitting Vital Signs';
    ELSE IF PARAMCD='STVSALL' THEN PARAM='All Standing Vital Signs';
    ELSE IF PARAMCD='VSALL' THEN PARAM=STRIP(PARAM);
/* 11) END KB 14Apr2014 */

```

```

        if paramcd = 'SUSYSBP' then paramn = 1;
        else if paramcd = 'SUDIABP' then paramn = 2;
        else if paramcd = 'SUPULSE' then paramn = 3;
        else if paramcd = 'SURESP' then paramn = 4;
        else if paramcd = 'INTP' then paramn = 13;
        else if paramcd = 'HEIGHT' then paramn = 14;
        else if paramcd = 'WEIGHT' then paramn = 15;
        else if paramcd = 'BMI' then paramn = 16;
        else if paramcd = 'INTPCM' then paramn = 17;
        else if paramcd = 'SISYSBP' then paramn = 5;
        else if paramcd = 'SIDIABP' then paramn = 6;
        else if paramcd = 'SIPULSE' then paramn = 7;
        else if paramcd = 'SIRESP' then paramn = 8;
        else if paramcd = 'STSYSBP' then paramn = 9;
        else if paramcd = 'STDIABP' then paramn = 10;
        else if paramcd = 'STPULSE' then paramn = 11;
        else if paramcd = 'STRESP' then paramn = 12;

```

```

/* 4) START KB 14Jan2014 */
    ELSE IF PARAMCD='SYSBP' THEN PARAMN=18;
    ELSE IF PARAMCD='DIABP' THEN PARAMN=19;
    ELSE IF PARAMCD='PULSE' THEN PARAMN=20;
        else if paramcd = 'VSALL' then paramn = 21/*18*/;

```

```

/* 4) END KB 14Jan2014 */
    ELSE IF PARAMCD='DBMI' THEN PARAMN=22; /* 7) KB 14Apr2014 */
    ELSE IF PARAMCD='SUVSALL' THEN PARAMN=23; /* 10) KB 14Apr2014 */
    ELSE IF PARAMCD='SIVSALL' THEN PARAMN=24; /* 10) KB 14Apr2014 */
    ELSE IF PARAMCD='STVSALL' THEN PARAMN=25; /* 10) KB 14Apr2014 */

```

```

        else put 'USER WARN' 'ING: check parameter and position as not
classified: ' usubjid = paramcd =;

```

```

/* 7) START KB 14Apr2014 */
    IF PARAMCD='DBMI' THEN DO;
        PARAMTYP='DERIVED';
        DTYPE='FUNCTION';
    END;

```

```

/* 7) END KB 14Apr2014 */

```

```

        * analysis values ;
        aval = vsstresn;
        avalu = trim(vsstresu);
        IF PARAMCD='DBMI' THEN AVALU='kg/m2'; /* 7) KB 14Apr2014 */

```

```

        if vstestcd = 'INTP' then do;

```

```

        if vsstresc = "NORMAL" then do;
            avalc = propcase(vsstresc);
            aval = 1;
        end;
        else if vsstresc = "ABNORMAL CLINICALLY NOT RELEVANT" then
do;
            avalc = "Abnormal, CNR";
            aval = 2;
        end;
        else if vsstresc = "ABNORMAL CLINICALLY RELEVANT" then do;
            avalc = "Abnormal, CR";
            aval = 3;
        end;
        ELSE PUT "WARN" "ING: Check V SSTRESC for VSTESTCD: " VSTESTCD=
VSSTRESC=; /* 1) KB 14Jan2014 */
        end;
        else avalc = trim(vsstresc);

        * visit information ;
        avisitn = visitnum;
        avisit = propcase(visit);
        atpt = propcase(vstpt);
        atptn = vstptnum;

        * baseline flag;
/*      ablfl = compress(vsblfl);*/ /* 9) KB 14Apr2014 */

        * analysis date and time;
/*      if length(vsdtc) gt 10 then adtm = input(vsdtc,e8601dt.);*/
        IF LENGTH(VSDTC) GT 10 THEN ADTM =
DHMS(INPUT(SCAN(VSDTC,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(VSDTC,2,'T'),TIME
5.)),MINUTE(INPUT(SCAN(VSDTC,2,'T'),TIME5.)),0); /* 16) KB 06Aug2014 */
        if not missing(adtm) then do;
            adt = datepart(adtm);
            atm = timepart(adtm);
        end;
        else if missing(adtm) and not missing(vsdtc) and length(vsdtc) = 10
then adt = input(vsdtc,yyymmdd10.);

        IF PARAMCD='DBMI' THEN VSDTC=''; /* 7) KB 14Apr2014 */

/* 19) START KB 21Sep2014 */
        IF INDEX(PARAMCD,'BMI') THEN DO;
            IF 0 < AVAL < 18.5 THEN DO;
                AVALCAT1 = 'Underweight';
            END;
            ELSE IF 18.5 <= AVAL < 25 THEN DO;
                AVALCAT1 = 'Normal weight';
            END;
            ELSE IF 25 <= AVAL < 30 THEN DO;
                AVALCAT1 = 'Overweight';
            END;
            ELSE IF AVAL >= 30 THEN DO;
                AVALCAT1 = 'Obese';
            END;
        END;
    
```

```

        END;
        ELSE IF NOT MISSING(AVAL) THEN PUT 'USER WARN' 'ING BMI
unclassified: ' USUBJID= AVAL=;
        END;
/* 19) END KB 21Sep2014 */

        keep usubjid vsseq param: aval: smok15p vsreasnd vsstat avisit:
        vsdtc vsdy /*ablf1*/ adt atm adtm visitnum atpt: epoch DTYPE;
/* 7) KB 14Apr2014 */ /* 9) KB 14Apr2014 */
run;

/* 9) START KB 14Apr2014 */
DATA ADSL2;
    SET ADAM.ADSL;
    FORMAT TESTDTM DATETIME13.;

    IF DTESTDTM=PTESTDTM=. THEN DELETE;

    TESTDTM=MIN(DTESTDTM,PTESTDTM);

    KEEP USUBJID TESTDTM;
RUN;

PROC SORT DATA=VS2;
    BY USUBJID;
RUN;

DATA VS2A;
    MERGE VS2 ADSL2;
    BY USUBJID;
RUN;

DATA VS2B;
    SET VS2A;
        WHERE VSSTAT NE 'NOT DONE' AND
INDEX(UPCASE(AVISIT),'UNSCHEDULED')=0 AND INDEX(PARAMCD,'VSALL')=0; /*
18) KB 21Sep2014 */

    IF ADTM=. AND ADT NE . THEN DO;
        IF ADT<DATEPART(TESTDTM) THEN TESTBASE='Y';
    END;
    ELSE IF ADTM NE . THEN DO;
        IF ADTM<TESTDTM THEN TESTBASE='Y';
    END;

/*    IF TESTDTM=. AND AVISIT='Day -1' THEN TESTBASE='Y';*/ /* 18) KB
21Sep2014 */
    IF TESTDTM=. /*AND PARAMCD IN ('BMI' 'HEIGHT')*/ AND AVISIT/*=*/ IN
('Screening' 'Day -1') THEN TESTBASE='Y'; /* 18) KB 21Sep2014 */
RUN;

PROC SORT DATA=VS2B(WHERE=(TESTBASE='Y')) OUT=VS2C;
    BY USUBJID PARAMCD AVISITN;
RUN;

```

```

DATA VS2D;
  SET VS2C;
  BY USUBJID PARAMCD AVISITN;
  FORMAT ABLFL $1.;

  IF LAST.PARAMCD AND LAST.AVISITN THEN ABLFL='Y';
RUN;

DATA VS2E;
  SET VS2D(WHERE=(ABLFL='Y')) ;

  KEEP USUBJID PARAMCD AVISITN ABLFL;
RUN;

PROC SORT DATA=VS2;
  BY USUBJID PARAMCD AVISITN;
RUN;

PROC SORT DATA=VS2E;
  BY USUBJID PARAMCD AVISITN;
RUN;

DATA VSBASES;
  MERGE VS2 VS2E;
  BY USUBJID PARAMCD AVISITN;
RUN;
/* 9) END KB 14Apr2014 */

* check labelling of multiple unschedules in one timepoint area ;
/*proc sort data=vs2;*/
/*  by avisitn atptn;*/
/*run;*/
/**/
/*proc sort data=vs2 out = uns(where = (index(lowercase(atpt),'unsched'))
keep = avisitn atpt atptn) nodupkey;*/
/*  by avisitn atptn;*/
/*run;*/
/**/
/*data vs2a(drop = count atpt);*/
/*  set uns;*/
/*  by avisitn;*/
/*  if first.avisitn then count=1;*/
/*  else count+1;*/
/*  if count > 1 then do;*/
/*      newatpt = trim(atpt) || ' ' || put(count,1.); */
/*      output;*/
/*  end;*/
/*run;*/

* put back onto data ;
/*data vs2b;*/
/*  merge vs2 vs2a;*/
/*  by avisitn atptn;*/

```



```

/*run;*/

/*data vs2c(drop = newatpt);*/
/*    set vs2b;*/
/*    if not missing(newatpt) then atpt = trim(newatpt);*/
/*run;*/

*****;
* Calculate changes from baseline (D-1) ;
*****;
proc sort data = /*VS2*//*vs2c*/VSBASES; /* 3) KB 14Jan2014 */ /* 9) KB
14Apr2014 */
    by usubjid paramn avisitn;
run;

* baseline ;
data base;
    set /*VS2*//*vs2c*/VSBASES(where = (ablfl = 'Y')); * check SDTM.VS
has VSBLFL correct to SAP ; /* 3) KB 14Jan2014 */ /* 9) KB 14Apr2014 */
    format base best. basec $200.;
    base = aval;
    basec = avalc;
    bvis = atptn; * keep to make sure only calculate change after
baseline ;

    keep usubjid paramn base basec bvis;
run;

* change ;
data change(drop = bvis);
    merge /*VS2*//*vs2c*/VSBASES base; /* 3) KB 14Jan2014 */ /* 9) KB
14Apr2014 */
    by usubjid paramn;
    format chg best.;
    if atptn gt bvis then chg = aval - base;
run;

proc sort data = change;
    by usubjid paramn avisitn;
run;

data change2;
    set change;
    by usubjid paramn avisitn;
    format anl01fl $2.;

    * determine if any unscheduled;
    if index(upcase(avisit), 'UNSCHEDULED') OR
index(upcase(atpt), 'UNSCHEDULED') or paramcd /*=*/IN ('VSALL' 'SUVSALL'
'SIVSALL' 'STVSALL') then anl01fl = ' '; /* 10) KB 14Apr2014 */
    else if last.avisitn and first.avisitn = 0 /*AND
INDEX(PARAMCD, 'SU')=0*/ then anl01fl = ' '; /* 2) KB 14Jan2014 */
    else anl01fl = 'Y';

```

```

/*IF PARAMCD NOT IN ('SUSYSBP' 'SUDIABP' 'SUPULSE') THEN
ANL01FL='';*/ /* 5) KB 14Jan2014 */
      IF INDEX(PARAMCD,"SYSBP") OR INDEX(PARAMCD,"DIABP") OR
INDEX(PARAMCD,"PULSE") THEN DO; /*6) JMH 15Jan2014*/
      IF SUBSTR(PARAMCD,1,2) NE "SU" THEN ANL01FL='';
      END;

      if anl01fl = ' ' then put 'Check reason for exclusion from
analysis: ' usubjid = param = avisit = atpt =;
run;

*****;
* Combine ADSL and VS data *;
*****;
* find period;
*_mtotper;

data slvs(drop = trt01p: trt01a: trt02p: trt02a: visitnum astday);
  merge adsl change2(in = a);
  by usubjid;
  if a;      * only include subject level data in vital signs ;
  format aperiod trtan trtpn aday astday 8. trta trtp $40. aperiodc
$8.;
  aday = adt - trtsdt + 1;
  astday = adt - trtsdt + 1;
  * allocate period/treatment;
  if astday in (0 1) then aperiod=1;
  else if astday in (2 3) then aperiod=2;
  *_mperall(dvar1 = adtm, dvar2 = adt);
  if not missing(aperiod) then do;
    aperiodc = 'Period ' || put(aperiod, 1.);
  end;
run;

*****;
* create output dataset ;
*****;

options replace;

data advs;
  set stdlib.advs slvs;
run;

proc sort data = advs out = adam.advs(label = 'Vital Signs Analysis
Dataset');
  by usubjid avisitn paramcd;
run;

options noreplace;

*_scramble(set=advs,id=usubjid subjid subjidn age sex sexc sexn race
dthfl height weightbl bmi ucpdgr1 ucpdgrln nicogr1

```

```

        nicogrln targr1 targrln analgr1 analgrln, dates=trtsdtm
trtsdt trtsday trtedtm trtedt trteday tr01sdt tr01stm tr01sdm tr01edt
tr01etm tr01edtm
        tr02sdt tr02stm tr02sdm tr02edt tr02etm tr02edtm,
        nullc=trtp trta trtseqp trtseqa TRTSTMF TR01STMF
TR02STMF, nulln=trtpn trtan trtseqpn trtseqan, nullcc=/*4*/7, nullnc=4);
/* 12) KB 14Apr2014 */

proc printto; run;
*=====;
* END OF PROGRAM CODE;
*=====;

```