

```

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 : 000000106324;
%put NOTE: Client Protocol ID   : ZRHR-REXC-03-EU;
%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;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_smulholl;
%put NOTE: Creation Date        : 2013-09-23;
%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: 30Nov2013  SM          1) Amend code to calculate change
from baseline for Day -1 unscheduled obs;
%put NOTE: 03Dec2013  SM          2) Exclude all unscheduled obs from
ANL01FL;
%put NOTE: 01May2014  KB          3) Removed format from VSSEQ;
%put NOTE: 01May2014  KB          4) Added derived BMI;
%put NOTE: 01May2014  KB          5) Amended ABLFL;
%put NOTE: 01May2014  KB          6) Added PARAMCDs for different
positions;
%put NOTE: 01May2014  KB          7) Amended PARAMs for different
position vitals;
%put NOTE: 01May2014  KB          8) Added EPOCH to keep statement;
%put NOTE: 01May2014  KB          9) Amended repeats message in log;
%put NOTE: 10May2014  KB          10) Amended baselines;
%put NOTE: 14May2014  KB          11) Amended sorting by key variables;
%put NOTE: 14May2014  KB          12) Removed format for SMOK15P;
%put NOTE: 14May2014  KB          13) Amended PARAMN for VSALL;
%put NOTE: 14May2014  KB          14) Rounded CHG;
%put NOTE: 14May2014  KB          15) Amended baselines;
%put NOTE: 14May2014  KB          16) Amended ANL01FL for NOT DONE tests;
%put NOTE: 05Jun2014  KB          17) Amended issue with ANL01FL for a
subjects results;
%put NOTE: 27Jul2014  KB          18) Added EXNOTRFL;
%put NOTE: 14Sep2014  KB          19) Added AVALCAT1 for BMI;
%put NOTE: 14Sep2014  KB          20) Added FASFL & PPROTFL to ADSL keep;

```

```

%put NOTE: 14Sep2014    KB                21) Amended ABLFL;
%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: cob1
        enrfl scrffl complfl saffl randfl trt: tr01: dthfl enfl
EXNOTRFL exfl fupfl FASFL PPROTFL; /* 18) KB 27Jul2014 */ /* 20) KB
14Sep2014 */
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.*/; /* 3) KB 01May2014 */ /* 12)
KB 14May2014 */
    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;

/* 4) START KB 01May2014 */
DATA DBMI;
    SET VS(WHERE=(VSTESTCD IN ('HEIGHT' 'WEIGHT') AND VSSTAT NE 'NOT
DONE'));
    FORMAT PARAMCD1 $8.;

    IF VISIT='DAY -2' THEN PARAMCD1='WEIGHT2';
    ELSE IF VISIT='DAY 6/DISCHARGE' THEN PARAMCD1='WEIGHT6';
    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 DBMI2 DBMI6 BEST.;

    DBMI2=WEIGHT2/((HEIGHT/100)**2);
    DBMI6=WEIGHT6/((HEIGHT/100)**2);
RUN;

PROC TRANSPOSE DATA=DBMI3 OUT=DBMI4;
    BY USUBJID;
    VAR DBMI2 DBMI6;
RUN;

DATA DBMI5;
    SET DBMI4;
    FORMAT VSSTRESN BEST. VISITNUM VSTPTNUM 8.;
    LENGTH VISIT $15 VSTPT $33 VSTESTCD $6 VSSTRESC $5;

    IF COL1='' THEN DELETE;

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

    IF VSTESTCD='DBMI2' THEN DO;
        VISIT='DAY -2';
        VISITNUM=98;
        VSTPT=VISIT;
    
```

```

        VSTPTNUM=1;
    END;
    ELSE IF VSTESTCD='DBMI6' THEN DO;
        VISIT='DAY 6/DISCHARGE';
        VISITNUM=106;
        VSTPT=VISIT;
        VSTPTNUM=9;
    END;

    VSTESTCD='DBMI';

    DROP _NAME_ COL1;
RUN;

DATA TIMES;
    SET SDTM.VS (WHERE=(VSTESTCD='WEIGHT' AND VISIT IN ('DAY -2' 'DAY
6/DISCHARGE')) AND VSSTAT NE 'NOT DONE'));

    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;
/* 4) END KB 01May2014 */

data vs2;
    merge /*vs*/DBMI6 suppv2; /* 4) KB 01May2014 */
    by usubjid vsseq;
    format avisitn paramn 8. aval atptn best. paramcd $8. param avisit
$40. atpt $50. AVALCAT1 $40. /* 19) KB 14Sep2014 */
        avalc $200. avalu $20. /*ABLFL $1.*//*ablfl $1.*/* adt
date9. atm time5. adtm datetime13. PARAMTYP DTYPE $10.; /* 4) KB
01May2014 */ /* 5) KB 01May2014 */ /* 10) KB 10May2014 */ /* 15) KB
14May2014 */
        * parameters ;
    if vstestcd not in ('WEIGHT' 'HEIGHT' 'BMI' /*'VSALL'*/'DBMI') then
do; /* 4) KB 01May2014 */
        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;

```

```

/* 7) START KB 01May2014 */
  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);
/* 7) END KB 01May2014 */

  IF PARAMCD='DBMI' THEN PARAM='Body Mass Index (Derived)'; /* 4) KB
01May2014 */
    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;
    else if paramcd = 'VSALL' then paramn = /*18*/21; /* 13) KB
14May2014 */
      ELSE IF PARAMCD='DBMI' THEN PARAMN=22; /* 4) KB 01May2014 */
      ELSE IF PARAMCD='SUVSALL' THEN PARAMN=23; /* 6) KB 01May2014 */
      ELSE IF PARAMCD='SIVSALL' THEN PARAMN=24; /* 6) KB 01May2014 */
      ELSE IF PARAMCD='STVSALL' THEN PARAMN=25; /* 6) KB 01May2014 */
      else put 'USER WARN' 'ING: check parameter and position as not
classified: ' usubjid = paramcd =;

/* 4) START KB 01May2014 */
  IF PARAMCD='DBMI' THEN DO;
    PARAMTYP='DERIVED';
    DTYPE='FUNCTION';
  END;
/* 4) END KB 01May2014 */

  * analysis values ;
  aval = vsstresn;
  avalu = trim(vsstresu);
  IF PARAMCD='DBMI' THEN AVALU='kg/m2'; /* 4) KB 01May2014 */
  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;
end;
else avalc = trim(vsstresc);

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

* baseline flag;
/*    ablfl = compress(vsbfl);*/ /* 5) KB 01May2014 */
/* 15) start KB 14May2014 */
/* 10) START KB 10May2014 */
/*    IF PARAMCD IN ('WEIGHT' 'DBMI') AND AVISIT='Day -2' THEN
ABLFL='Y';*/
/*    ELSE IF PARAMCD IN ('BMI' 'HEIGHT') AND AVISIT='Screening' THEN
ABLFL='Y';*/
/*    ELSE IF AVISIT='Day 0' THEN ABLFL='Y';*/
/* 10) END KB 10May2014 */
/* 15) end KB 14May2014 */

* analysis date and time;
length vsdtc1 $19.;
vsdtc1=vsdtc;
if length(vsdtc) gt 10 then adtm = input(vsdtc1,e8601dt.);
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,ymmdd10.);

    IF PARAMCD='DBMI' THEN VSDTC='';    /* 4) KB 01May2014 */

/* 19) START KB 14Sep2014 */
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 14Sep2014 */

        keep usubjid vsseq param: aval: smok15p vsreasnd vsstat avisit:
        vsdtc vsdy /*ablfl*/ /*ABLFL*/ adt atm adtm visitnum atpt:
DTYPE EPOCH; /* 4) KB 01May2014 */ /* 5) KB 01May2014 */ /* 8) KB
01May2014 */ /* 10) KB 10May2014 */ /* 15) KB 14May2014 */
run;
/* 10) START KB 10May2014 */
/* 5) START KB 01May2014 */
/*DATA ADSL2;*/
/*    SET ADAM.ADSL;*/
/*    FORMAT TESTDTM DATETIME13.;*/
/**/
/*    IF DTESTDTM=. THEN DELETE;*/
/**/
/*    TESTDTM=DTESTDTM;*/
/*    */
/*    KEEP USUBJID TESTDTM;*/
/*RUN;*/

/*PROC SORT DATA=VS2;*/
/*    BY USUBJID;*/
/*RUN;*/
/**/
/*DATA VS2A;*/
/*    MERGE VS2 ADSL2;*/
/*    BY USUBJID;*/
/*RUN;*/
/**/
/*DATA VS2B;*/
/*    SET VS2A;*/
/**/
/*    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 PARAMCD IN ('DBMI' 'WEIGHT') AND AVISIT='Day -2' THEN
TESTBASE='Y';*/
/**/
/*    IF TESTDTM=. AND PARAMCD IN ('WEIGHT' 'DBMI') AND AVISIT='Day -2'
THEN TESTBASE='Y';*/
/*    IF TESTDTM=. AND PARAMCD IN ('BMI' 'HEIGHT') AND AVISIT='Screening'
THEN TESTBASE='Y';*/
/*RUN;*/

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

```

```

/*RUN;*/
/**/
/*DATA VS2D;*/
/*    SET VS2C;*/
/*    BY USUBJID PARAMCD AVISITN ATPTN;*/
/*    FORMAT ABLFL $1.;*/
/**/
/*    IF LAST.PARAMCD AND FIRST.ATPTN THEN ABLFL='Y';*/
/*RUN; */
/**/
/*DATA VS2E;*/
/*    SET VS2D(WHERE=(ABLFL='Y')) ;*/
/**/
/*    KEEP USUBJID PARAMCD AVISITN ABLFL ATPTN;*/
/*RUN;*/
/**/
/*PROC SORT DATA=VS2;*/
/*    BY USUBJID PARAMCD AVISITN ATPTN;*/
/*RUN;*/
/**/
/*PROC SORT DATA=VS2E;*/
/*    BY USUBJID PARAMCD AVISITN ATPTN;*/
/*RUN;*/
/**/
/*DATA VSBASES;*/
/*    MERGE VS2 VS2E;*/
/*    BY USUBJID PARAMCD AVISITN ATPTN;*/
/*RUN;*/
/* 5) END KB 01May2014 */
/* 10) END KB 10May2014 */

* 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) ;
*****;
/* 21) START KB 14Sep2014 */
/*PROC SORT DATA=VS2(WHERE=(AVISIT IN ('Screening' 'Day -2' 'Day -1' 'Day
0' 'Day 1') AND INDEX(UPCASE(ATPT),'UNSCHE')=0 and VSSTAT ne 'NOT DONE'))
OUT=VS2A; *//* 19) KB 27Jul2014 */
/*      BY USUBJID;*/
/*RUN;*/
**/
/*DATA VS2DATE;*/
/*      SET VS2(WHERE=(AVISIT ='Day 1')) ;*/
/*      KEEP USUBJID ADT;*/
/*      RENAME ADT=DAY1DT;*/
/*RUN;*/
**/
/*PROC SORT DATA=VS2DATE NODUPKEY;*/
/*      BY USUBJID;*/
/*RUN;*/
**/
/*DATA VS2DATE2;*/
/*      MERGE VS2A VS2DATE;*/
/*      BY USUBJID;*/
**/
/*      IF ADTM NE . THEN DO;*/
/*          IF ADTM<DHMS(DAY1DT,6,29,0) THEN CHECK='Y';*/
/*      END;*/
/*      IF PARAMCD='HEIGHT' AND AVISIT='Screening' THEN CHECK='Y';*/
/*      ELSE IF PARAMCD IN ('DBMI' 'WEIGHT') AND AVISIT='Day -2' THEN
CHECK='Y';*/
**/
/*      IF DAY1DT=. AND AVISIT='Day 0' THEN CHECK='Y';*/
/*RUN;*/
**/
/*DATA BASE;*/
/*      SET VS2DATE2;*/
/*      IF CHECK NE 'Y' THEN DELETE;*/
/*RUN;*/
**/
/*PROC SORT DATA=BASE;*/
/*      BY USUBJID PARAMN DESCENDING ADTM;*/
/*RUN;*/
**/
/*PROC SORT DATA=BASE OUT=BASE2(KEEP=USUBJID PARAMN ADTM CHECK AVISITN
ATPTN) NODUPKEY;*/

```

```

/* BY USUBJID PARAMN; */
/*RUN; */
/**/
/*DATA VS2DATE3 (WHERE=(ABLFL='Y')) ; */
/* SET BASE2; */
/* FORMAT ABLFL $1.; */
/**/
/* IF CHECK='Y' THEN ABLFL='Y'; */
/* KEEP USUBJID PARAMN ABLFL ADTM AVISITN ATPTN; */
/*RUN; */
/**/
/*PROC SORT DATA=VS2DATE3; */
/* BY USUBJID PARAMN */ /*AVISITN */ /*ATPTN; */ /* 17) KB 15May2014 */
/*RUN; */
/**/
/*PROC SORT DATA=VS2; */
/* BY USUBJID PARAMN */ /*AVISITN */ /*ATPTN; */ /* 17) KB 15May2014 */
/*RUN; */
/**/
/*DATA VS2A; */
/* MERGE VS2 VS2DATE3; */
/* BY USUBJID PARAMN */ /*AVISITN */ /*ATPTN; */ /* 17) KB 15May2014 */
/*RUN; */
/* 15) END KB 14May2014 */
/* 21) END KB 14Sep2014 */

/*proc sort data = */ /*vs2c */ /*VSBASES */ /*VS2; */ /* 5) KB 01May2014 */ /*
10) KB 10May2014 */
/* by usubjid paramn avisitn ATPTN; */ /* 10) KB 10May2014 */
/*run; */

/* 10) START KB 10May2014 */
/*DATA VS2A; */
/* SET VS2 (WHERE=(ABLFL='Y')) ; */
/* BY USUBJID PARAMN AVISITN ATPTN; */
/**/
/* IF NOT (FIRST.AVISITN AND FIRST.ATPTN) THEN ABLFL=''; */
/**/
/* KEEP USUBJID PARAMN AVISITN ATPTN ABLFL; */
/*RUN; */
/**/
/*DATA VS2B; */
/* MERGE VS2 VS2A; */
/* BY USUBJID PARAMN AVISITN ATPTN; */
/*RUN; */
/* 10) END KB 10May2014 */

/* 21) START KB 14Sep2014 */
DATA SV;
SET SDTM.SV (WHERE=(VISIT=('DAY 1')));
FORMAT DAY DATE9.;

DAY=INPUT(SCAN(SVSTDTC,1,'T'),YYMMDD10.);
KEEP USUBJID DAY;

```

```

RUN;

DATA ABLFL;
  SET VS2(WHERE=(AVISIT IN ('Screening' 'Day -2' 'Day -1' 'Day 0' 'Day
1') AND VSSTAT NE 'NOT DONE'));

  KEEP USUBJID PARAMCD AVISIT ATPTN ADTM ATPT;
RUN;

PROC SORT DATA=ABLFL;
  BY USUBJID;
RUN;

DATA ABLFL2;
  MERGE ABLFL(IN=A) SV;
  BY USUBJID;
  IF A;
RUN;

DATA ADSLTM;
  SET ADAM.ADSL;
  WHERE TRT01A IN ('CC' 'THS 2.2' 'SA');

  KEEP USUBJID TRTSDTM TRT01A;
RUN;

DATA ABLFL2A;
  MERGE ABLFL2(IN=A) ADSLTM;
  BY USUBJID;
  IF A;
RUN;

PROC SORT DATA=ABLFL2A;
  BY USUBJID PARAMCD ATPTN ADTM;
RUN;

DATA ABLFL3;
  SET ABLFL2A;

  IF PARAMCD IN ('HEIGHT' 'WEIGHT' 'BMI') AND
INDEX(UPCASE(AVISIT), 'UNSCHED')=0 AND INDEX(UPCASE(ATPT), 'UNSCHED')=0
THEN ABLFL2='Y';
  ELSE DO;
    IF TRT01A='SA' THEN DO;
      IF DAY NE . THEN DO;
        IF ADTM<DHMS(DAY,6,30,0) AND
INDEX(UPCASE(AVISIT), 'UNSCHED')=0 AND INDEX(UPCASE(ATPT), 'UNSCHED')=0
THEN ABLFL2='Y';
        END;
      ELSE IF DAY EQ . THEN DO;
        IF INDEX(UPCASE(AVISIT), 'UNSCHED')=0 AND
INDEX(UPCASE(ATPT), 'UNSCHED')=0 THEN ABLFL2='Y';
        END;
      END;
    END;
  END;

```

```

        ELSE IF TRT01A IN ('CC' 'THS 2.2') THEN DO;
            IF ADTM<TRTSDTM AND INDEX(UPCASE(AVISIT), 'UNSCHED')=0 AND
INDEX(UPCASE(ATPT), 'UNSCHED')=0 THEN ABLFL2='Y';
            END;
        ELSE IF MISSING(TRT01A) THEN DO;
            IF INDEX(UPCASE(AVISIT), 'UNSCHED')=0 AND
INDEX(UPCASE(ATPT), 'UNSCHED')=0 THEN ABLFL2='Y';
            END;
        END;
RUN;

PROC SORT DATA=ABLFL3(WHERE=(ABLFL2='Y')) OUT=ABLFL4;
    BY USUBJID PARAMCD ATPTN ADTM;
RUN;

DATA ABLFL5(WHERE=(ABLFL='Y'));
    SET ABLFL4;
    BY USUBJID PARAMCD ATPTN ADTM;
    FORMAT ABLFL $1.;

    IF LAST.PARAMCD THEN ABLFL='Y';

    KEEP USUBJID PARAMCD AVISIT ATPTN ABLFL;
RUN;

PROC SORT DATA=ABLFL5;
    BY USUBJID PARAMCD ATPTN AVISIT;
RUN;

PROC SORT DATA=VS2;
    BY USUBJID PARAMCD ATPTN AVISIT;
RUN;

DATA VS2D;
    MERGE VS2 ABLFL5;
    BY USUBJID PARAMCD ATPTN AVISIT;
RUN;
/* 21) END KB 14Sep2014 */

* baseline ;
data base;
    set /*vs2c*//*VSBASES*//*VS2B*//*VS2A*/VS2D(where = (ablfl = 'Y'));
* check SDTM.VS has VSBLFL correct to SAP ; /* 5) KB 01May2014 */ /* 10)
KB 10May2014 */ /* 15) KB 14May2014 */ /* 21) KB 14Sep2014 */
    format base best. basec $200.;
    base = aval;
    basec = avalc;
    bvis = ATPTN/*visitnum*/;    * keep to make sure only calculate
change after baseline ;/* 1) SM 30Nov2013 */

    keep usubjid paramn base basec bvis VISITNUM; /* 15) KB 14May2014
*/
run;

```

```

/* 9) START KB 01May2014 */
PROC SORT DATA=BASE;
    BY USUBJID PARAMN VISITNUM DESCENDING BVIS; /* 15) KB 14May2014 */
RUN;

DATA BASE2(DROP=VISITNUM); /* 15) KB 14May2014 */
    SET BASE;
    BY USUBJID PARAMN VISITNUM DESCENDING BVIS; /* 15) KB 14May2014 */

    IF NOT /*FIRST.BVIS*/FIRST.PARAMN THEN DELETE; /* 15) KB 14May2014 */
RUN;

PROC SORT DATA=BASE2;
    BY USUBJID PARAMN;
RUN;
/* 9) END KB 01May2014 */

/* 21) START KB 14Sep2014 */
PROC SORT DATA=VS2D;
    BY USUBJID PARAMN;
RUN;
/* 21) END KB 14Sep2014 */

* change ;
data change(drop = bvis);
    merge /*vs2c*//*VSBASES*//*VS2*//*VS2A*/ VS2D /*base*/BASE2; /* 5)
KB 01May2014 */ /* 9) KB 01May2014 */ /* 10) KB 10May2014 */ /* 15) KB
14May2014 */ /* 21) KB 14Sep2014 */
    by usubjid paramn;
    format chg best.;
    if /*avisitn*/ATPTN gt bvis then chg = /*aval - base*/ROUND(AVAL-
BASE,0.00001); /* 1) SM 30Nov2013 */ /* 14) KB 14May2014 */
run;

proc sort data = change;
    by usubjid paramn avisitn ATPTN; /* 17) KB 05Jun2014 */
run;

data change2;
    set change;
    by usubjid paramn avisitn ATPTN; /* 17) KB 05Jun2014 */
    format anl01fl $2.;

    * determine if any unscheduled;
    if INDEX(UPCASE(avisit), /*=*/ 'UNSCHEDULED') OR
INDEX(UPCASE(ATPT), 'UNSCHEDULED') or paramcd /*=*/ IN ('VSALL' 'SUVSALL'
'SIVSALL' 'STVSALL') OR VSSTAT='NOT DONE' then anl01fl = ' '; /* 2) SM
03Dec2013 */ /* 6) KB 01May2014 */ /* 16) KB 14May2014 */
    else if last.ATPTN/*avisitn*/ and first.ATPTN/*avisitn*/ = 0 then
anl01fl = ' '; /* 17) KB 05Jun2014 */
    else anl01fl = 'Y';
    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 = trt01: tr01: visitnum);
    merge adsl change2(in = a);
    by usubjid;
    if a;          * only include subject level data in vital signs ;
    format aperiod trtan trtpn aday 8. trta trtp $40. aperiodc $10.;
    aday = adt - trtsdt + 1;
    * allocate period/treatment;
    *_mperall(dvar1 = adtm, dvar2 = adt);
    aperiodc = 'Period ' || put(aperiod, 1.);
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;*/
    BY USUBJID AVISITN ATPTN PARAMCD; /* 11) KB 14May2014 */
run;

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

```