

```

/*=====
=====*

| Covance Study Number   : 000000106331          |
|
| Program Name           : d_ADVS.sas             |
|
| Purpose                : Create ADVS dataset    |
|
| Input Data             : SDTM.VS SDTM.SUPPVS ADAM.ADSL          |
|
| Output Data            : ADAM.ADVS              |
|
| Macros Called          : m_printto, m_logchk, m_attrib_adam      |
|
| Originally Performed by : kpothuri              |
|
| Date                   : 25March2015            |
|
|                         |
|=====
=====|

| Modification History          |
|-----|
| Modified by                   : kpothuri          |
| Modification Date            : 6/29/2015          |
| Modification Description     : remove leading zeros from basec and avalc
|
+=====
=====*/

options validvarname=upcase;

libname adam "&base2/datasets/adam/cleaned_adam";

libname sdtm "/cvn/projects/prj/data/000000106331/datasets/sdtm/sdtmx";

%m_printto(route=YES);

```

```
*****,
```

```
* bring in ADSL ;
```

```
*****,
```

```
data adsl;
```

```
    set adam.adsl;
```

```
    drop studyid;
```

```
run;
```

```
proc sort data=adsl; by usubjid; run;
```

```
*****,
```

```
* pick up SUPPVS      ;
```

```
*****,
```

```
data suppvs;
```

```
    set sdtm.suppvs;
```

```
    /*PECLSIG*/
```

```
    if QNAM="SMOK15P" then do;
```

```
        SMOK15P=QVAL;
```

```
        vsseq=input(idvarval,best.);
```

```
        output;
```

```
    end;
```

```
    keep usubjid idvarval SMOK15P vsseq;
```

```
run;
```

```
*****,
```

```
* bring in VS ;
```

```
*****,
```

```
proc sort data = sdtm.vs out = vs; by usubjid vsseq; run; /*10,674*/
```

```
%macro paramcd (PECD=, PAR=, PARAM=);
```

```
    if vstestcd("&PECD" then do;
```

```
        PARAMN=&PAR;
```

```
        PARAMCD="&PARAM";
```

```
        PARAM=vstest;
```

```
        aval=vsstresn;
```

```
        avalc=vsstresc;
```

```
        avalu=vsstresu;
```

```
        output;
```

```
    end;
```

```
%mend paramcd;
```

```
%macro paramcd1 (PECD=, PAR=, PARAM=);
```

```
    if vstestcd("&PECD" and VSPOS in ("SUPINE", "SITTING") then do;
```

```
        PARAMN=&PAR;
```

```
        PARAMCD="&PARAM";
```

```
        PARAM=vstest;
```

```
        aval=vsstresn;
```

```
        avalc=vsstresc;
```

```
        avalu=vsstresu;
```

```
        output;
```

```

        end;

%mend paramcd1;

data vs1;

length AVALCAT1 $40 paramcd $8;

    set vs;

    %paramcd1 (PECD=SYSBP, PAR=1, PARAM=SYSBP);

    %paramcd1 (PECD=DIABP, PAR=2, PARAM=DIABP);

    %paramcd1 (PECD=PULSE, PAR=3, PARAM=PULSE);

    %paramcd1 (PECD=RESP, PAR=4, PARAM=RESP);

    %paramcd (PECD=HEIGHT, PAR=14, PARAM=HEIGHT);

    %paramcd (PECD=WEIGHT, PAR=15, PARAM=WEIGHT);

    %paramcd (PECD=WSTCIR, PAR=26, PARAM=WSTCIR);

    %paramcd (PECD=VSALL, PAR=21, PARAM=VSALL);

    if VSTESTCD='BMI' then do;

        paramcd='BMI';

        param=vstest;

        paramn=16;

        if 0 < VSSTRESN < 18.5 then AVALCAT1 = "Underweight";

        if 18.5 <= VSSTRESN < 25 then AVALCAT1 = "Normal weight";

        if 25 <= VSSTRESN < 30 then AVALCAT1 = "Overweight";

        if VSSTRESN >= 30 then AVALCAT1 = "Obese";

        aval=vsstresn;

        avalc=vsstresc;

        avalu=vsstresu;

```

```

        output;

    end;

run;

/*PARAMCD - DBMI only when BMI is not present*/

proc sort data=vs1; by usubjid visitnum vstptnum vsdtc; run;

data sub_WH;

    set vs1 (where=(VSTESTCD="WEIGHT" and WGHT ne .) rename=(VSSTRESN=WGHT));

    by usubjid visitnum vstptnum vsdtc;

run;

proc sort data=vs1; by usubjid; run;

data sub_HG;

    set vs1 (where=(VSTESTCD="HEIGHT" and HGHT ne .) rename=(VSSTRESN=HGHT));

    by usubjid;

    keep usubjid HGHT;

run;

proc sort data = vs1(where=(vstestcd = "BMI" and vsstresn ne .)) out = bmi1(keep = usubjid visitnum
vstptnum vsdtc);

    by usubjid visitnum vstptnum vsdtc;

run;

data bmi2;

    merge bmi1(in=a) sub_WH(in=b);

    by usubjid visitnum vstptnum vsdtc;

    if b and not a; *Keep records where weight recorded but BMI not derived;

run;

```

```

data bmi3;

merge bmi2(in=a) sub_HG(in=b);

by usubjid;

if a and b; *Can only derive if weight and height are present;

run;

data DBMI;

length avalcat1 $40;

set bmi3;

paramcd = 'DBMI';

param = "Body Mass Index (Derived)";

paramn = 22;

aval = (WGHT/((HGHT/100)**2));

avalc = put(aval,5.2);

valu = "kg/m2";

vsseq = .;

paramtyp="Derived";

if 0 < AVAL < 18.5 then AVALCAT1 = "Underweight";

if 18.5 <= AVAL < 25 then AVALCAT1 = "Normal weight";

if 25 <= AVAL < 30 then AVALCAT1 = "Overweight";

if AVAL >= 30 then AVALCAT1 = "Obese";

run; /*69*/

data vs2;

```

```

set vs1 DBMI;

/*PARCAT1*/

if paramcd in ('SYSBP', 'DIABP', 'WEIGHT', 'WSTCIR') then do;

    PARCAT1="Risk Markers";

    PARCAT1N=1;

end;

else do;

    PARCAT1="";

    PARCAT1N=.;

end;

if length(vsdtc) gt 10 then vsdtk_1=dhms(input(substr(vsdtc,1,10),yymmdd10.)
,0,0,input(substr(vsdtc,12),time5.));

format vsdtk_1 datetime13.;

else if length(vsdtc) = 10 then vsdtk_1=dhms(input(substr(vsdtc,1,10),yymmdd10.) ,0,0,0);

format vsdtk_1 datetime13.;

run; /*10,743*/

*****
* Combine VS and SUPPVS data *;
*****

proc sort data=vs2; by usubjid vsseq; run;

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

data vs2a;

length paramcd $8;

```

```

merge vs2 suppv;

by usubjid vsseq;

run; /*10,743*/

proc sort data=vs2a; by usubjid paramcd visitnum vsdte_1; run; /*10,743*/

data vs3;

length paramcd $8;

set vs2a;

by usubjid paramcd visitnum vsdte_1;

/*ADTM*/

ADTM=vsdte;

/*ADT, ATM*/

if length(ADTM)=10 then ADT=input(ADTM,yyymmdd10.);

else if length(ADTM) gt 10 then ADT=input(substr(ADTM,1,10),yyymmdd10.);

format ADT date9.;

if length(ADTM) gt 10 then ATM=input(substr(ADTM,12),time5.);

format ATM time5.;

run;

data vs4;* test;

length AVISIT $40;

merge adsl vs3(in=a);

```


by usubjid;

if a;

/*ADAY*/

if not missing (adt) and not missing (trtsdt) then aday = adt - trtsdt + 1;

/*ATPT*/

ATPT=VSTPT;

ATPTN=VSTPTNUM;

*Visits;

if COMPLFL = "Y" then do;

 AVISIT = VISIT;

 AVISITN = VISITNUM;

end;

if find(DISCCAT,"Discontinued", 'I')>0 then do;

 if paramcd in ('SYSBP', 'DIABP', 'PULSE', 'RESP') then do;

 if visit = "DAY 6/DISCHARGE CONFINEMENT" and ADAY not in (6, .) then do;

 AVISIT="DAY " || strip(put(ADAY, best.));

 AVISITN=ADAY+100;

 ATPT="DAY " || strip(put(ADAY, best.));

 ATPTN=ADAY+3;

 end;

 else if visit = "DAY 91/DISCHARGE AMBULATORY" then do;

 if 7<ADAY<31 then do;

```

        AVISIT="DAY 30";

        AVISITN=130;

        ATPT="DAY 30";

        ATPTN=10;

    end;

    else if 32<ADAY<61 then do;

        AVISIT="DAY 60";

        AVISITN=160;

        ATPT="DAY 60";

        ATPTN=11;

    end;

    else do;

        AVISIT=VISIT;

        AVISITN=VISITNUM;

    end;

end;

else do;

    AVISIT=VISIT;

    AVISITN=VISITNUM;

end;

end;

else if paramcd in ('WEIGHT', 'BMI', 'WSTCIR' ) then do;

    if visit = "DAY 91/DISCHARGE AMBULATORY" then do;

        if 7<ADAY<31 then do;

            AVISIT="DAY 30";

```

```
        AVISITN=130;

        ATPT="DAY 30";

        ATPTN=10;

    end;

    else if 32<ADAY<61 then do;

        AVISIT="DAY 60";

        AVISITN=160;

        ATPT="DAY 60";

        ATPTN=11;

    end;

    else do;

        AVISIT=VISIT;

        AVISITN=VISITNUM;

    end;

end;

else do;

    AVISIT=VISIT;

    AVISITN=VISITNUM;

end;

end;

end;
```

```

/*TRT:*/

TRTP=TRT01P;

TRTPN=TRT01PN;

TRTA=TRT01A;

TRTAN=TRT01AN;

run;/*10,743*/

proc sort data=vs4; by usubjid paramcd avisitn vsdte_1; run;/*10,743*/

*ablfl;

proc sort data=vs4; by paramcd usubjid avisitn vsdte_1; run;

data vs_fact vs_fact_1;

    set vs4;

    by paramcd usubjid avisitn vsdte_1;

    if armcd="SMABST" and avisitn=101 and missing(atm) then do;

        ABLFL="";

        output vs_fact;

    end;

    else output vs_fact_1;

run;

data vs_fact_2 vs_fact_3;

    set vs_fact_1;

```

```

    by paramcd usubjid avisitn vsdtc_1;

    if armcd in ("MCC", "THS 2.2M", "SMABST") then do;

        if vsdtc_1<trtsdtm and avisitn <= 101 and vsstat ne "NOT DONE" then do;

            ablfl_="Y";

            output vs_fact_2;

        end;

    end;

    if ablfl_ ne "Y" then output vs_fact_3;

run;

data vs_fact_4;

    set vs_fact_2;

    by paramcd usubjid avisitn vsdtc_1;

    if last.usubjid then ablfl="Y";

run;

data ablfl;

    set vs_fact vs_fact_3 vs_fact_4;

run;


proc sort data=ablfl out=dummy nodupkey; where paramcd in ('WEIGHT' 'WSTCIR'); by usubjid paramcd;
run;


data dummy1;

    set dummy;

    avisit="DAY -2"; avisitn=98; aval=.; avalc=""; dtype="LOCF";

```

```
atpt = "";
atptn = .;
adt = .;
adtm = "";
atm = .;
aday = .;
vsdtc_1=.;
vsseq=.;
vsdtc="";
vsdy=.;
epoch="";
SMOK15P="";
ablfl="";
```

output;

```
avisit="DAY 91/DISCHARGE AMBULATORY"; avisitn=191; aval=.; avalc=""; dtype="LOCF";
```

```
atpt = "";
atptn = .;
adt = .;
adtm = "";
atm = .;
aday = .;
vsdtc_1=.;
vsseq=.;
vsdtc="";
vsdy=.;
```

```
    epoch="";  
    SMOK15P="";  
    ablfl="";  
output;  
run;  
  
data dummy2;  
    set dummy1;  
    keep usubjid paramcd avisit avisitn dtype;  
run;  
  
proc sort data=ablfl out=vs4_;  
where paramcd in ('WEIGHT' 'WSTCIR');  
by usubjid paramcd avisit avisitn;  
run;  
  
data vs5a;  
    merge vs4_(in=in1) dummy2(in=in2);  
    by usubjid paramcd avisit avisitn;  
    if in1=0 and in2; *keeping only missing avisits;  
run;  
  
data vs6a;  
    merge dummy1 vs5a(keep=usubjid paramcd avisit avisitn in=in5);  
    by usubjid paramcd avisit avisitn;  
    if in5; *merging back with intermediate dataset;
```

```
run;
```

```
proc sort data=ablfl out=dummy_p nodupkey; where paramcd in ('SYSBP', 'DIABP'); by usubjid paramcd;  
run;
```

```
data dummy1_p;
```

```
set dummy_p;
```

```
avisitn=100;avisit="DAY 0";dtype="LOCF";AVAL=.;AVALC="";
```

```
atpt = "";
```

```
atptn = .;
```

```
adt = .;
```

```
adtm = "";
```

```
atm = .;
```

```
aday = .;
```

```
vsdtc_1=.;
```

```
vsseq=.;
```

```
vsdtc="";
```

```
vsdy=.;
```

```
epoch="";
```

```
SMOK15P="";
```

```
ablfl="";
```

```
output;
```

```
avisitn=106;avisit="DAY 6/DISCHARGE CONFINEMENT";dtype="LOCF";AVAL=.;AVALC="";
```

```
atpt = "";
```

```
atptn = .;
```

```
adt = .;
```


adtm = "";

atm = .;

aday = .;

vsdtc_1=.;

vsseq=.;

vsdtc=""

vsdy=.;

epoch=""

SMOK15P=""

ablfl=""

output;

avisitn=130;avisit="DAY 30";dtype="LOCF";AVAL=.;AVALC=""

atpt = ""

atptn = .;

adt = .;

adtm = ""

atm = .;

aday = .;

vsdtc_1=.;

vsseq=.;

vsdtc=""

vsdy=.;

epoch=""

SMOK15P=""

ablfl=""

output;

avisitn=160;avisit="DAY 60";dtype="LOCF";AVAL=.;AVALC="";

atpt = "";

atptn = .;

adt = .;

adtm = "";

atm = .;

aday = .;

vsdtc_1=.

vsseq=.

vsdtc="";

vsdy=.

epoch="";

SMOK15P="";

ablfl="";

output;

avisitn=191;avisit="DAY 91/DISCHARGE AMBULATORY";dtype="LOCF";AVAL=.;AVALC="";

atpt = "";

atptn = .;

adt = .;

adtm = "";

atm = .;

aday = .;

vsdtc_1=.

vsseq=.

```

vsdtc="";

vsdy=.;

epoch="";

SMOK15P="";

ablfl="";

output;

run;


data dummy2_p;

    set dummy1_p;

        keep usubjid paramcd avisit avisitn dtype;

run;


proc sort data=ablfl out=vs4_p;

where paramcd in ('SYSBP', 'DIABP');

by usubjid paramcd avisit avisitn;

run;


proc sort data=dummy2_p; by usubjid paramcd avisit avisitn; run;

data vs5a_p;

    merge vs4_p(in=in1) dummy2_p(in=in2);

    by usubjid paramcd avisit avisitn;

    if in1=0 and in2; *keeping only missing avisits;

run;

```

```

proc sort data=dummy1_p; by usubjid paramcd avisit avisitn; run;

data vs6a_p;

  merge dummy1_p vs5a_p(keep=usubjid paramcd avisit avisitn in=in5);

  by usubjid paramcd avisit avisitn;

  if in5; *merging back with intermediate dataset;

run; /*136*/

```

```

proc sort data=ablfl; by usubjid paramcd avisit avisitn; run;

data dat_sch dat_unsch;

  set ablfl;

  if index(avisit, "UNSCHEDULED")=0 then output dat_sch;

  else output dat_unsch;

run;

```

```

data set_extra;

  set dat_sch vs6a_p vs6a;

  by usubjid paramcd avisit avisitn; *set back extra avisits that are missing;

run; /*10,567 - without unsch*/

```

```

proc sort data=set_extra; by usubjid paramcd avisitn adtm aval; run;

data LOCF;

retain tempvar tempvar2 tempvar_1 tempvar_2 tempvar_4 tempvar_5

tempvar_1a tempvar_2a tempvar_4a tempvar_5a tempvar_6 tempvar_6a tempvar_7 tempvar_7a
tempvar_8 tempvar_8a 0;

retain tempvar1 tempvar3 tempvar4 tempvar4a tempvar_3 tempvar_3a tempvar_9 tempvar_9a
tempvar_10 tempvar_10a

```

```
tempvar_11 tempvar_11a;
```

```
set set_extra;
```

```
by usubjid paramcd avisitn adtm aval;
```

```
if AVAL ^= . and AVALC ^= "" then do;
```

```
tempvar=AVAL;
```

```
tempvar1=AVALC;
```

```
tempvar4=atpt;
```

```
tempvar_1=atptn;
```

```
tempvar_2=adt;
```

```
tempvar_3=adtm;
```

```
tempvar_4=atm;
```

```
tempvar_5=aday;
```

```
tempvar_6=vsdtc_1;
```

```
tempvar_7=vsseq;
```

```
tempvar_9=vsdtc;
```

```
tempvar_8=vsdy;
```

```
tempvar_10=epoch;
```

```
tempvar_11=SMOK15P;
```

```
end;
```

```
/*DTYPE*/
```

```
if AVAL = . and AVALC = "" and AVISIT in ('DAY 0', 'DAY 6/DISCHARGE CONFINEMENT', 'DAY 30',  
'DAY 60', 'DAY 91/DISCHARGE AMBULATORY') and
```

```
PARAMCD in ('SYSBP', 'DIABP') then do;
```

```
AVAL=tempvar;
```

```
AVALC=tempvar1;
```

```
    atpt=tempvar4;
    atptn=tempvar_1;
    adt=tempvar_2;
    adtm=tempvar_3;
    atm=tempvar_4;
    aday=tempvar_5;
    vsdte_1=tempvar_6;
    vsseq=tempvar_7;
    vsdte=tempvar_9;
    vsdy=tempvar_8;
    epoch=tempvar_10;
    SMOK15P=tempvar_11;
end;
```

```
if AVAL ^= . and AVALC ^= "" then do;
```

```
    tempvar2=AVAL;
    tempvar3=AVALC;
    tempvar4a=atpt;
    tempvar_1a=atptn;
    tempvar_2a=adt;
    tempvar_3a=adtm;
    tempvar_4a=atm;
    tempvar_5a=aday;
    tempvar_6a=vsdte_1;
    tempvar_7a=vsseq;
```

```

tempvar_9a=vsdtc;

tempvar_8a=vsdy;

tempvar_10a=epoch;

tempvar_11a=SMOK15P;

end;

if AVAL = . and AVALC = "" and AVISIT in ('DAY -2', 'DAY 91/DISCHARGE AMBULATORY') and
PARAMCD in ('WEIGHT', 'WSTCIR') then do;

    AVAL=tempvar2;

    AVALC=tempvar3;

    atpt=tempvar4a;

    atptn=tempvar_1a;

    adt=tempvar_2a;

    adtm=tempvar_3a;

    atm=tempvar_4a;

    aday=tempvar_5a;

    vsdtc_1=tempvar_6a;

    vsseq=tempvar_7a;

    vsdtc=tempvar_9a;

    vsdy=tempvar_8a;

    epoch=tempvar_10a;

    SMOK15P=tempvar_11a;

end;

run;/*10,567*/

data tot;

```

```

        set dat_unsch locf;

run; /*10,925*/

*****
* Calculate changes from baseline (Screening) ;
*****

*baseline, change;

data base (rename=(adt=adt_ avalc=basec aval=base));

        set tot;

        where ABLFL='Y';

        keep usubjid paramcd adt avalc aval;

run;

proc sort data=BASE; by usubjid paramcd; run;

proc sort data=tot; by usubjid paramcd; run;

proc sql noprint;

        create table new as select distinct (A.*), b.adt_, b.basec, b.base

        from tot as A left join BASE B

        on A.usubjid=B.usubjid and A.paramcd=B.paramcd;

quit; /*10,928*/

data change;

        set new;

        if ADT<ADT_ then do;

                BASEC="";

```



```

        BASE=.;
end;

else if vsstat="NOT DONE" then do;

        BASEC="";

        BASE=.;

end;


if adt>adt_ then do;

        if BASE >0 then chg = aval - base;

        IF BASE >0 THEN DO;

                PCHG=(CHG/BASE)*100;

        END;

        ELSE if BASE=0 then DO;

                PCHG=(CHG/1)*100;

        END;

end;


/*      Asper, Asperc*/

if AVISITN < 101 then do;

        ASPER=1;

        ASPERC="Pre-Randomization Period";

end;

else if 101<=AVISITN<=106 then do;

        ASPER=2;

        ASPERC="Confinement Period";

```

```
end;  
  
else if 106<AVISITN<=191 then do;  
  
    ASPER=3;  
  
    ASPERC="Ambulatory Period";  
  
end;  
  
else if AVISITN>191 then do;  
  
    ASPER=4;  
  
    ASPERC="Safety Follow-up Period";  
  
end;
```

```
/*APUPER*/
```

```
if 101<=AVISITN<=106 then do;  
  
    APUPER=1;  
  
    APUPERC='Period 1';  
  
end;  
  
else If 106<AVISITN<=131 then do;  
  
    APUPER=2;  
  
    APUPERC='Period 2';  
  
end;  
  
else if 131<AVISITN<=161 then do;  
  
    APUPER=3;  
  
    APUPERC='Period 3';  
  
end;  
  
else if 161<AVISITN<=191 then do;  
  
    APUPER=4;
```

```

        APUPERC='Period 4';

    end;

run;

proc sort data=change; by usubjid paramn avisitn vsdte_1; run;

data vs5 vs6;

    set change;

    if vsstat="NOT DONE" then output vs5;

        else output vs6;

run;

data vs7;

    set vs6;

    by usubjid paramn avisitn vsdte_1;

    /*Anl01fl*/

    if SAFBFL="Y" or SAFAFBFL="Y" then do;

        if adt_ ne . and adt>=adt_ and first.avisitn then ANL01FL = "Y";

            else if ablfl ne "Y" and TRTSDT ne . and adt>=TRTSDT and first.avisitn then ANL01FL =

"Y";

        end;

        if INDEX(UPCASE(AVISIT),'UNSCHEDULED') ^= 0 then ANL01FL="";

run;

data comb (drop=ADTM);

    set vs7 vs5;

run;

```

```

data final_1;

    set comb;

    *for Enrolled not randomized;

    if trtpn in (97,98) then do;

        ASPERC="Pre-Randomization Period";

        ASPER=1;

        APUPER=.;

        APUPERC="";

    end;

run;

*****;

* create output dataset ;

*****;

*options replace;

data ADVS (drop=avalc basec rename=(vsdtc_1=ADTM avals_1=avalc basec_1=basec));

    set final_1;

    AVISIT=propcase(avisit);

    ATPT=propcase(atpt);

    if not missing(avalc) then do;

        if int(input(avalc,best.))<0 and substr(avalc,1,1)="0" then avals_1=substr(avalc,2);

```

```

        else avalc_1=avalc;

    end;

    else avalc_1=avalc;

    if not missing(basec) then do;

        if int(input(basec,best.))<0 and substr(basec,1,1)="0" then basec_1=substr(basec,2);

        else basec_1=basec;

    end;

    else basec_1=basec;

run;

%m_attrib_adam(dset=ADVS);

proc sort data=ADVS out=adam.ADVS(label = 'Vital Signs Analysis Dataset');

    by USUBJID AVISITN ATPTN PARAMCD;

run;

%m_logchk;

```