

```

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

| Covance Study Number   : 000000106331          |
|
| Program Name           : d_adpe.sas             |
|
| Purpose                : Create ADPE dataset    |
|
| Input Data             : SDTM.PE SDTM.SUPPPE ADAM.ADSL          |
|
| Output Data            : ADAM.ADPE              |
|
| Macros Called          : m_printto, m_logchk, m_attrib_adam      |
|
| Originally Performed by : kpothuri              |
|
| Date                   : 16March2015            |
|
|                       |
|=====
=====|

| Modification History          |
|-----|
| Modified by                  : kpothuri          |
|
| Modification Date           : 5/26/15            |
|
| Modification Description    : ABLFL derivation, ANL01FL flagged at baseline, shift1 derivation
|
+=====
=====*/

options validvarname=upcase; options symbolgen mprint mlogic;

options missing="";

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 SUPPPE      ;
```

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

```
data SUPPPE;
```

```
    set sdtm.supppe;
```

```
/*PECLSIG*/
```

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

```
    PECLSIG=QVAL;
```

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

```
    output;
```

```
end;
```

```
keep usubjid idvarval PECLSIG peseq;  
  
run;
```

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

```
* bring in PE ;
```

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

```
proc sort data=sdtm.pe out=pe; /*24,589*/
```

```
by usubjid peseq;
```

```
run;
```

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

```
if PETESTCD="&PECD" then do;
```

```
if PESTAT='NOT DONE' then AVALC='Not Examined';
```

```
else if PESTRESC='NORMAL' then AVALC='Normal';
```

```
else if not missing (PESTRESC) then AVALC='Abnormal';
```

```
PARAMN=&PAR;
```

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

```
PARAM=PETEST;
```

```
output;
```

```
end;
```

```
%mend paramcd;
```

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

```
if PETESTCD="&PECD" and &COND then do;
```

```

        if PESTAT='NOT DONE' then AVALC='Not Examined';

            else if PESTRESC='NORMAL' then AVALC='Normal';

            else if not missing (PESTRESC) then AVALC='Abnormal';

        PARAMN=&PAR;

        PARAMCD="&PARAM";

        PARAM=PETEST;

        output;

    end;

%mend paramcd1;


%macro paramcd2 (PECD=, PAR=, PARAM=);

    if PETESTCD="&PECD" and SCAN(PESTRESC,2,"-") not in ('EXTREMITIES', 'LYMPH NODES',
'LYMPHATIC SYSTEM', 'VASCULAR') then do;

        if PESTAT='NOT DONE' then AVALC='Not Examined';

            else if PESTRESC='NORMAL' then AVALC='Normal';

            else if not missing (PESTRESC) then AVALC='Abnormal';

        PARAMN=&PAR;

        PARAMCD="&PARAM";

        PARAM=PETEST;

        output;

    end;

%mend paramcd2;


data PE1;

    set PE;

    %paramcd (PECD=GAPPEAR, PAR=1, PARAM=GAPPEAR);

```

```

%paramcd (PECD=HEENT, PAR=2, PARAM=HEENT);

%paramcd (PECD=THYROID, PAR=3, PARAM=THYROID);

%paramcd (PECD=HEART, PAR=4, PARAM=HEART);

%paramcd1 (PECD=CHEST, PAR=5, PARAM=CHEST, COND=%str(PECAT ne 'CHEST X-RAY'));

%paramcd (PECD=LUNGS, PAR=6, PARAM=LUNGS);

%paramcd (PECD=BACK, PAR=11, PARAM=BACK);

%paramcd (PECD=CVS, PAR=8, PARAM=CVS);

%paramcd (PECD=GASTRO, PAR=7, PARAM=GASTRO);

%paramcd (PECD=NEURO, PAR=9, PARAM=NEURO);

%paramcd (PECD=ABDOMEN, PAR=13, PARAM=ABDOMEN);

%paramcd (PECD=SKIN, PAR=10, PARAM=SKIN);

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

%paramcd (PECD=MUSCULO, PAR=12, PARAM=MUSCULO);

%paramcd1 (PECD=OTHER, PAR=16, PARAM=OTHEXTRM, COND=%str(SCAN(PESTRESC,2,"-") =
"EXTREMITIES"));

%paramcd1 (PECD=OTHER, PAR=17, PARAM=OTHLYMN, COND=%str(SCAN(PESTRESC,2,"-") =
"LYMPH NODES"));

%paramcd1 (PECD=OTHER, PAR=18, PARAM=OTHLYMS, COND=%str(SCAN(PESTRESC,2,"-") =
"LYMPHATIC SYSTEM"));

%paramcd1 (PECD=OTHER, PAR=19, PARAM=OTHVASC, COND=%str(SCAN(PESTRESC,2,"-") =
"VASCULAR"));

%paramcd2 (PECD=OTHER, PAR=15, PARAM=OTHER);

if PECAT='CHEST X-RAY' then do;

    PARAMN=20;

    PARAMCD='CHESTX';

    PARAM='Chest X-ray';

```

```

        if PESTRESC='NORMAL' then AVALC='Normal';

            else if not missing (PESTRESC) then AVALC='Abnormal';

        output;

    end;

    else if petestcd='PEALL' then do;

        PARAMN=99;

        PARAMCD='PEALL';

        PARAM='All Physical Examination Tests';

        output;

    end;

run;

data pe2;

length DESC $200;

set pe1;

/*DESC*/

if PESTRESC='NORMAL' then DESC = "";

    else DESC=left(trim(propcase(tranwrd(pestresc,'ABNORMAL - ',''))));

*Dates;

if not missing(pedtc) then adt = input(pedtc,yymmdd10.);

    format adt date9.;

run;

```

```

*****;

* Combine PE and SUPPPE data *;

*****;

proc sort data=pe2; by usubjid pseq; run;

proc sort data=supppe; by usubjid pseq; run;

data pe_supp (drop=PESPID PECLSIG rename=(PESPID_=PESPID PECLSIG_=PECLSIG));

length PESPID_ $2 PECLSIG_ $3;

    merge pe2 supppe;

    by usubjid pseq;

    PESPID_=PESPID;

    PECLSIG_=PECLSIG;

run;


data pe3;

length AVISIT $40;

    merge adsl pe_supp(in=a);

    by usubjid;

    if a;

/*ADAY*/

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

*Visits;

    if find(DISCCAT,"Discontinued", 'l')>0 and visit = "DAY 91/DISCHARGE AMBULATORY" then do;

        if 7<ADAY<31 then do;

```

```

        AVISIT="DAY 30";

        AVISITN=130;

        ATPT="";

        ATPTN=.;

    end;

    else if 32<ADAY<61 then do;

        AVISIT="DAY 60";

        AVISITN=160;

        ATPT="";

        ATPTN=.;

    end;

    else do;

        AVISIT=VISIT;

        AVISITN=VISITNUM;

    end;

end;

else do;

    AVISIT=VISIT;

    AVISITN=VISITNUM;

end;

/*TRT:*/

TRTP=TRT01P;

TRTPN=TRT01PN;

TRTA=TRT01A;

```



```

        TRTAN=TRT01AN;

run;

*ablfl;

proc sort data=pe3; by paramcd usubjid avisitn adt; run;

data base1 base2 base2_;

    set pe3;

    by paramcd usubjid avisitn adt;

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

        if not missing(adl) and adl<trtsdt and PESTAT ne "NOT DONE" and
INDEX(UPCASE(AVISIT),'UNSCHEDULED') = 0 then do;

            ablfl_="Y";

            output base1;

        end;

        else output base2;

    end;

    else do;

        output base2_;

    end;

run;

data base3;

    set base1;

    by paramcd usubjid avisitn adt;

    if last.usubjid then ablfl="Y";

run;

```

```

data ablfl;

    set base2 base2_ base3;

run;

/* Obtain Baselines*/

data BASE (rename=(adt=adt_ avalc=basec));

    set ablfl;

    where ABLFL='Y' and INDEX(UPCASE(AVISIT),'UNSCHEDULED') = 0;

    keep usubjid paramcd adt avalc;

run;

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

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

data new;

    merge ablfl BASE;

    by usubjid paramcd;

run;

data pe4;

    set new;

    if ADT<ADT_ then BASEC="";

    else if pestat="NOT DONE" then BASEC="";

run;

PROC SORT DATA=pe4; BY USUBJID PARAMN AVISITN ADT; RUN;

data pe5 pe6;

    set pe4;

```

```

        if pestat="NOT DONE" then output pe5;
            else output pe6;
run;

DATA pe7;

    SET pe6;

    BY USUBJID PARAMN AVISITN ADT;

    /*ANL01FL*/

    if adt_ ne . and adt>=adt_ and last.avisitn then ANL01FL = "Y";
        else if TRTSDT ne . and adt>=TRTSDT and last.avisitn then ANL01FL = "Y";

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

RUN;

/*Shift1*/

data shift (rename=(peclsig=clig_base));

    set pe7;

    where ablfl="Y";

    keep usubjid paramcd peclsig;

run;

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

proc sort data=pe7; by usubjid paramcd; run;

data shift_1;

    merge shift pe7;

```

```

        by usubjid paramcd;

run;

data shift_2;

    set shift_1;

    if ADT > ADT_ and avalc ne "" and basec ne "" then do;

        if basec="Abnormal" and avalc="Abnormal" then shift1 = trim(basec) || ', ' || trim(PECLSIG)
        || ' to ' || trim(avalc) || ', ' || trim(PECLSIG);

        else if clig_base ne "" and basec="Abnormal" then shift1 = trim(basec) || ', ' ||
        trim(clig_base) || ' to ' || trim(avalc);

        else if avalc="Normal" and basec="Normal" then shift1 = trim(basec) || ' to '
        || trim(avalc);

        else if not missing(PECLSIG) then shift1 = trim(basec) || ' to ' || trim(avalc) || ', ' ||
        trim(PECLSIG);

    end;

run;

data comb;

    set pe5 shift_2;

run;

*****;

* create output dataset ;

*****;

*options replace;

data ADPE;

```

```
set comb;

AVAL = .;

BASE = .;

AVISIT=propcase(avisit);

run;


%m_attrib_adam(dset=ADPE);


proc sort data=ADPE out=adam.adpe(label = 'Physical Examination Analysis Dataset');

    BY USUBJID AVISITN PARAMCD PESPID;

run;


%m_logchk;
```