

```

/*=====
Covance Study ID   : COV-000000106331
Program Name       : d_adxp.sas
Purpose            : Program to ADCO dataset
Author             : siva karnati
Date of Creation    : 26MAR2015
Input Data         : SDTM.XP SDTM.SUPPXP,ADAM.ADSL
Output Data        : ADAM.ADXP
Macros Called      : m_printto,m_attrib_adam,m_logchk
=====

Modification History
=====

Modified by        :
Modification Date   :
Modification Description :

=====*/

%M_PRINTTO;
options replace;
libname sdtm "/cvn/projects/prj/data/000000106343/datasets/sdtm/sdtmx";

PROC FORMAT ;

VALUE $ASPR
  "Pre-Randomization Period" =1
  "Confinement Period"=2
  "Ambulatory Period"=3
  "Safety Follow-up Period"=4
;
RUN;

/*ADSL DATA */
DATA ADSL;
  SET ADAM.ADSL;
  KEEP STUDYID USUBJID SUBJID SUBJIDN SITEID AGE SEX SEXN SEXC RACE DTHFL HEIGHT WEIGHTBL BMI
  UCPDGR1 UCPDGR1N /*NICOGR1 NICOGR1N TARGR1 TARGR1N*/ ENRLFL SCRFFL EXFL EXNOTRFL ENFL
  COMPLFL FUPFL /*SAFFL FSAFFL*/ SAFBFL SAFABL FASFL PPROT1FL PPROT2FL PPROT3FL PPROT4FL RANDFL
  TRTSDTM TRTSTMF TRTSDT TRTSDAY TRTEDTM TRTETMF TRTEDT TRTEDAY
  TRT01P TRT01PN TRT01A TRT01AN RANDDT COMPP1FL COMPP2FL COMPP3FL COMPP4FL
;
RUN;

/* SUPPXP */

DATA SUPPXP;
  SET SDTM.SUPPXP(WHERE=(QNAM="XPCLSIG"));
  KEEP USUBJID IDVARVAL QVAL QNAM;
RUN;

DATA XP1;
  SET SDTM.XP;
RUN;

PROC SQL;
  CREATE TABLE XP AS SELECT  A.* ,B.QVAL AS XPCLISG_ FROM XP1 A LEFT JOIN SUPPXP B
    ON A.USUBJID=B.USUBJID AND A.XPSEQ=INPUT(B.IDVARVAL,2.);
  QUIT;

/*DFEVFVC*/
PROC SORT DATA=XP(WHERE=(XPTESTCD='FEV1FVC' AND NOT MISSING (XPSTRESN) AND XPSTAT NE 'NOT DONE')) OUT=XPA NODUPKEY;
  BY USUBJID XPSCAT VISITNUM XPTPT;
RUN;

DATA XP1A;
  SET XPA;
  PRESENT=1;
  KEEP USUBJID XPSCAT PRESENT VISITNUM XPTPT;
RUN;

PROC SORT DATA=XP;
  BY USUBJID XPSCAT VISITNUM XPTPT;
RUN;

DATA XPPRESENT;
  MERGE XP XP1A;
  BY USUBJID XPSCAT VISITNUM XPTPT;

```

```

RUN;

PROC SORT DATA=XPPRESENT;
    BY USUBJID VISITNUM XPTPT;
RUN;

DATA XPPRESENT2;
    SET XPPRESENT;
    BY USUBJID VISITNUM XPTPT;

RUN;

PROC SORT DATA=XPPRESENT2;
    BY USUBJID XPCAT VISITNUM XPTPT;
RUN;

DATA RATIO(DROP = FEV1 FVC PRESENT);
    MERGE XPPRESENT2(WHERE = (XPTTESTCD_ = 'FEV1MEAS' AND PRESENT_ NE 1) RENAME = (XPSTRESN = FEV1 PRESENT=PRESENT_ XPTTESTCD=XPTTESTCD_)
    )
        XPPRESENT2(WHERE = (XPTTESTCD = 'FVCMEAS' AND PRESENT NE 1) RENAME = (XPSTRESN = FVC) KEEP = USUBJID XPCAT VISITNUM XPTTESTCD XPST
RESN PRESENT XPTPT);
    BY USUBJID XPCAT VISITNUM XPTPT;
    XPTTESTCD = 'DFEVFVC';
    XPTEST = 'Ratio between FEV1/FVC (Derived)';
    XPSTRESN = ROUND((FEV1 / FVC),0.01);
    XPSTRESC = LEFT(TRIM(PUT(XPSTRESN,5.2)));
    XPORRES = TRIM(XPSTRESC);
    XPORRESU = ' ';
    XPSTRESU = 'RATIO';
    XPSEQ=.;

RUN;
/**/
/* PROC SORT DATA=XP1A OUT=XP_FEVMEAS(KEEP=USUBJID XPCAT VISIT XPSCAT VISITNUM XPSTRESN XPDTX XPTPTNUM XPTTESTCD RENAME=(XPSTRESN=XPS
TRESN_FEVMEAS XPTTESTCD=XPTTESTCD_));*/
/*BY USUBJID XPSCAT VISITNUM XPDTX;*/
/*WHERE XPTTESTCD="FEV1MEAS" AND XPSTRESN NE .;*/
/*RUN;*/
/**/
/**/
/* PROC SORT DATA=XP1A OUT=XP_FVCMEAS(KEEP=USUBJID XPSCAT VISITNUM XPSTRESN XPDTX/*XPTPTNUM XPTTESTCD RENAME=(XPSTRESN=XPSTRESN_FVCME
AS ));*/
/*BY USUBJID XPSCAT VISITNUM XPDTX ;*/
/*WHERE XPTTESTCD="FVCMEAS" AND XPSTRESN NE .;*/
/*RUN;*/
/**/
/**/
/*DATA FEVFVC;*/
/*MERGE XP_FEVMEAS XP_FVCMEAS;*/
/*BY USUBJID XPSCAT VISITNUM XPDTX;*/
/*RATIO=ROUND((XPSTRESN_FEVMEAS/XPSTRESN_FVCMEAS),0.01);*/
/*FORMAT PARAMCD $8. PARAM $80. AVALU PARAMTYP DTYPE $20.    ;*/
/*LENGTH PARAMN 8.;*/
/* */
/*PARAMCD="DFEVFVC";*/
/*PARAM="Ratio between FEV1/FVC (Derived)";*/
/*PARAMTYP="DERIVED";*/
/*DTYPE="RATIO";*/
/*AVALU="RATIO";*/
/*PARAMN=20;*/
/*AVAL=RATIO;*/
/*AVALC=PUT(AVAL,5.2);*/
/*DROP XPSTRESN: XPTTESTCD: RATIO;*/
/*RUN;*/;
%MACRO PAR(PARAMCD= ,PARAM= ,PARAMN= ,AVAL= ,AVALC= ,AVALU=);
    IF PARAMCD=&PARAMCD. THEN DO;
        PARAM=&PARAM.;
        PARAMN=&PARAMN.;
        AVAL=&AVAL.;
        AVALC=&AVALC.;
        AVALU=&AVALU.;
    END;
%MEND;
DATA XP2;
    SET XP(RENAME=(EPOCH=EPOCH_)) RATIO(RENAME=(EPOCH=EPOCH_));
    LENGTH PARAMCD $8. PARAM $80. AVALC $200. XPCLSIG $3. ABLFL $1. PARCAT1 PARCAT2 AVISIT $40. ATPT $27. EPOCH $23. PARAMTYP
DTYPE $20.;
    LENGTH PARAMN AVAL ATPTN AVISITN 8. ;

```

```

IF STRIP(XPTESTCD) IN("MEF25_75") THEN XPTESTCD=COMPRESS(XPTESTCD,,"ka");
IF XPSCAT="WITHOUT SHORT ACTING BRONCHODILATOR" THEN DO;
    PARAMCD=STRIP(COMPRESS(XPTESTCD,"1"));

END;
ELSE IF XPSCAT="WITH SHORT ACTING BRONCHODILATOR" THEN DO;
    IF XPTESTCD NOT IN("DFEVFVC","XPALL") THEN PARAMCD="W"||STRIP(COMPRESS(XPTESTCD,"1"));
    ELSE IF XPTESTCD IN ("DFEVFVC","XPALL") THEN PARAMCD=STRIP(COMPRESS(XPTESTCD,"1"));

END;
ELSE PARAMCD=STRIP(COMPRESS(XPTESTCD,"1"));

%PAR(PARAMCD="FEVMEAS",PARAM="Best measured FEV1 value",PARAMN=15,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FVCMEAS",PARAM="Best measured FVC value",PARAMN=13,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="WBRONCHO",PARAM="Name of bronchodilator (with bronchodilator)",PARAMN=1,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU)
;
%PAR(PARAMCD="WDOSE",PARAM="Dose (with bronchodilator)",PARAMN=2,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FEVFVC",PARAM="Calculated ratio between FEV1/FVC",PARAMN=18,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FEVPCT",PARAM="Percent of predicted FEV1 value",PARAMN=17,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FVCPCT",PARAM="Percent of predicted FVC value",PARAMN=14,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FEVPRED",PARAM="Predicted FEV1 value",PARAMN=16,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="FVCPRED",PARAM="Predicted FVC value",PARAMN=12,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="INTP",PARAM="Interpretation",PARAMN=19,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="WFEVMEAS",PARAM="Best measured FEV1 value (with bronchodilator)",PARAMN=6,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRES
U);
%PAR(PARAMCD="WFVCMESAS",PARAM="Best measured FVC value (with bronchodilator)",PARAMN=4,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU
);
%PAR(PARAMCD="WFEVFVC",PARAM="Calculated ratio between FEV1/FVC (with bronchodilator)",PARAMN=9,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU
=XPSTRESU);
%PAR(PARAMCD="WFEVPCT",PARAM="Percent of predicted FEV1 value (with bronchodilator)",PARAMN=8,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=X
PSTRESU);
%PAR(PARAMCD="WFVCPCT",PARAM="Percent of predicted FVC value (with bronchodilator)",PARAMN=5,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XP
STRESU);
%PAR(PARAMCD="WFEVPRED",PARAM="Predicted FEV1 value (with bronchodilator)",PARAMN=7,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="WFVCPRED",PARAM="Predicted FVC value (with bronchodilator)",PARAMN=3,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="WINTP",PARAM="Interpretation (with bronchodilator)",PARAMN=10,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="XPALL",PARAM="All Spirometry examinations",PARAMN=.,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="DFEVFVC",PARAM="Ratio between FEV1/FVC (Derived)",PARAMN=20,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="MEF",PARAM="Mid expiratory flow 25-75",PARAMN=21,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="WMEF",PARAM="Mid expiratory flow 25-75 (with bronchodilator)",PARAMN=11,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU)
;
%PAR(PARAMCD="DLCO",PARAM="Diffusion Capacity for Lung CO",PARAMN=51,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="DLCOVA",PARAM="Diffusion Capacity for Lung CO to Alveol",PARAMN=52,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="KCO",PARAM="Rate Constant of CO",PARAMN=54,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="IC",PARAM="Inspiratory Capacity",PARAMN=61,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="RV",PARAM="Residual Volume",PARAMN=63,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="TLC",PARAM="Total Lung Capacity",PARAMN=64,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);
%PAR(PARAMCD="VC",PARAM="Vital Capacity",PARAMN=65,AVAL=XPSTRESN,AVALC=XPSTRESC,AVALU=XPSTRESU);

IF PARAMCD="DFEVFVC" THEN DO;
    PARAMTYP="DERIVED";
    DTYPE="RATIO";
END;
IF XPTESTCD="INTP" AND INDEX(XPSTRESC,"ABNORMAL")>0 THEN DESC=STRIP(SCAN(XPSTRESC,2,"-"));
XPCLSIG=STRIP(XPCLISG_);
ABLFL=XPBLFL;
PARCAT1=STRIP(XPCAT);
PARCAT2=STRIP(XPSCAT);
ATPT=STRIP(XPTPT);
ATPTN=XPTPTNUM;
AVISIT=STRIP(VISIT);
/*IF INDEX(AVISIT,"UNSCHEDULED")>0 THEN AVISITN=XPDY;*/
/* ELSE*/ AVISITN=VISITNUM;
EPOCH =STRIP(EPOCH_);

IF PARAMCD="XPALL" THEN DO;
    IF XPCAT="LUNG CAPACITY" THEN PARAMN=30;
    ELSE IF XPCAT="GAS TRANSFER" THEN PARAMN=60;
    ELSE IF XPCAT="LUNG VOLUME MEASUREMENT" THEN PARAMN=70;

END;

KEEP USUBJID XPSEQ PARAMCD PARAM PARAMN AVAL AVALC AVALU XPCAT XPDTG
XPSTRF XPENRF DESC ABLFL PARCAT1 PARCAT2 XPMETHOD XPCLSIG XPSTAT XPREASND ATPT ATPTN AVISIT AVISITN XPDTG XPDY EPOCH DTYPE PARAMTY
P;

RUN;

```

```

/*BASE LINE DERIVATION*/
PROC SORT DATA=XP2; BY USUBJID; RUN;
DATA XP2A(DROP=ABLFL);;
MERGE XP2(IN=A) ADSL(KEEP=USUBJID TRTSDTM TRT01P IN=B);
BY USUBJID;
IF A;

IF LENGTH(XPDTCT)>10 THEN ADTM1=DHMS(INPUT(SCAN(XPDTCT,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(XPDTCT,2,'T'),TIME5.)),MINUTE(INPUT(SCAN(XPDTCT,3,'T'),TIME5.)),0);
ELSE IF LENGTH(XPDTCT)=10 THEN ADTM1=DHMS(INPUT(XPDTCT,YYMMDD10.),0,0,0);
FORMAT ADTM1 DATETIME13.;
RUN;
DATA XP2A1 XPBASE;
SET XP2A;
IF AVALC NE " " AND ADTM1<=TRTSDTM THEN OUTPUT XPBASE ;
ELSE OUTPUT XP2A1;
RUN;

PROC SORT DATA=XPBASE; BY USUBJID PARAMCD ADTM1;RUN;

DATA XPBASE1;
SET XPBASE;
LENGTH ABLFL $1.;
BY USUBJID PARAMCD ADTM1;
IF LAST.PARAMCD THEN ABLFL="Y";
RUN;
DATA XP2(DROP= TRTSDTM TRT01P);
SET XP2A1 XPBASE1;
RUN;
PROC SORT DATA=XP2 ; BY USUBJID XPCAT PARAMN ;RUN;

DATA XP_BASEA;
SET XP2(WHERE=(ABLFL="Y"));
LENGTH AVALC XPCL_BASE $200.;
BASE=AVAL;
BASEC=AVALC;
BASEVISIT=AVISITN;
BASEDTC=XPDTCT;
FLAG=1;
XPCLSIG_BASE=XPCLSIG;
IF PROPCASE(STRIIP(SCAN(BASEC,1,"-"))="Abnormal" and xpclsig_base ne " " then XPCL_BASE=PROPCASE(STRIIP(SCAN(BASEC,1,"-")))||"|"||"
" ||STRIIP(XPCLSIG_BASE);
ELSE XPCL_BASE=PROPCASE(STRIIP(SCAN(BASEC,1,"-")));

KEEP USUBJID BASE: PARAMN XPCAT FLAG XPCLSIG_BASE XPCL_BASE;
RUN;
PROC SORT DATA=XP2 ; BY USUBJID PARAMN ;RUN;
PROC SORT DATA=XP_BASEA;BY USUBJID PARAMN;RUN;
DATA XP3A;
MERGE XP2 XP_BASEA(DROP=XPCAT);
BY USUBJID PARAMN;
LENGTH SHIFT1 $50.;
LENGTH CHG 8.;
IF AVISITN >= BASEVISIT THEN DO;
IF NMISSED(AVAL,BASE)=0 THEN CHG=AVAL-BASE;
END;
IF AVISITN > BASEVISIT THEN DO;
IF PARAMCD IN ("INTP","WINTP") THEN DO;
IF CMISSED(AVALC,XPCL_BASE)=0 AND /*PROPCASE(STRIIP(SCAN(AVALC,1,"-"))) NE STRIP(XPCL_BASE) AND*/ NOT ANYDIGIT(SCAN(AVALC,1,"-"))
THEN DO;
IF XPCLSIG NE " " THEN SHIFT1=/*PROPCASE(STRIIP(SCAN(BASEC,1,"-")))*STRIIP(XPCL_BASE)||" "||"to"||" "||PROPCASE(STRIIP(SCAN(AVALC,1,"-")))||" "||STRIIP(XPCLSIG));
ELSE IF XPCLSIG EQ " " THEN SHIFT1=/*PROPCASE(STRIIP(SCAN(BASEC,1,"-")))*STRIIP(XPCL_BASE)||" "||"to"||" "||PROPCASE(STRIIP(SCAN(AVALC,1,"-")));
END;
END;
END;
* IF BASEVISIT > AVISITN AND ABLFL = " " THEN BASEC=" ";
DROP BASEVISIT;

RUN;

DATA XP3;
SET XP3A;
IF LENGTH(BASEDTC)>10 THEN BDTM1=DHMS(INPUT(SCAN(BASEDTC,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(BASEDTC,2,'T'),TIME5.)),MINUTE(INPUT(SCAN(BASEDTC,3,'T'),TIME5.)),0);
ELSE IF LENGTH(BASEDTC)=10 THEN ADTM1=DHMS(INPUT(BASEDTC,YYMMDD10.),0,0,0);

```

```
IF NMISS(ADTM1,BDTM1)=0 AND (ADTM1)<(BDTM1) THEN DO;
BASEC=" ";
SHIFT1=" ";
END;
FORMAT BDTM1 DATETIME13.;
RUN;
```

```
PROC SORT DATA=XP3;BY USUBJID PARAMCD AVISITN XPDTC;RUN;
```

```
PROC SORT DATA=XP3 OUT=ANL;
BY USUBJID PARAMCD AVISITN XPDTC;
WHERE AVAL NE . OR AVALC NE " " /*AND (INDEX(AVISIT, "UNSCHEDULED")) = 0*/;
RUN;
```

```
DATA ANL1(KEEP=USUBJID PARAMCD AVISITN XPDTC ANL01FL);
SET ANL;
LENGTH ANL01FL $2.;
BY USUBJID PARAMCD AVISITN XPDTC;
IF FIRST.AVISITN;
ANL01FL = "Y";
RUN;
```

```
PROC SORT DATA=ANL1;BY USUBJID PARAMCD AVISITN XPDTC; RUN;
DATA XP4;
MERGE XP3(IN=A) ANL1;
BY USUBJID PARAMCD AVISITN XPDTC;
IF A;
RUN;
```

```
PROC SORT DATA=XP4 OUT=LOCF;
BY USUBJID PARAMCD AVISITN;
WHERE (AVAL NE . OR AVALC NE " ") AND ANL01FL = "Y";
RUN;
```

```
DATA LOCF1;
SET LOCF;
LENGTH DTYPE $20.;
BY USUBJID PARAMCD AVISITN;
IF LAST.PARAMCD;
IF AVISITN IN (1,100) THEN DO;
DTYPE="LOCF";
AVISITN=106;
ATPTN = 9;
OUTPUT;
AVISITN=191;
ATPTN = 10;
OUTPUT;
END;
IF AVISITN = 106 THEN DO;
DTYPE="LOCF";
AVISITN=191;
ATPTN = 10;
OUTPUT;
END;
RUN;
```

```
DATA XP5;
SET XP4 LOCF1;
IF DTYPE="LOCF" THEN DO;
IF AVISITN=106 THEN AVISIT="DAY 6/DISCHARGE CONFINEMENT";
ELSE IF AVISITN=191 THEN AVISIT= "DAY 91/DISCHARGE AMBULATORY";
IF ATPTN = 9 THEN ATPT = "DAY 6/DISCHARGE CONFINEMENT";
ELSE IF ATPTN = 10 THEN ATPT = "DAY 91/DISCHARGE AMBULATORY";
END;
IF AVISITN IN (1,100) THEN ORD =1;
IF AVISITN EQ 106 THEN ORD =2;
IF AVISITN EQ 191 THEN ORD=3;
RUN;
```

```
PROC SORT DATA= XP5; BY USUBJID; RUN;
DATA XP6;
MERGE XP5(IN=A) ADSL(KEEP=USUBJID SUBJIDN);
BY USUBJID;
IF A;
RUN;
```

```

PROC SORT DATA=XP6 OUT=XP6A; BY USUBJID PARAMCD DESCENDING ORD;
WHERE ANL01FL ="Y" AND (AVAL NE . OR AVALC NE " ");
RUN;
DATA XP6B;
SET XP6A;
BY USUBJID PARAMCD DESCENDING ORD;
ORD1=LAG(ORD);
SUBJIDN1=LAG(SUBJIDN);
PAR1=LAG(PARAMN);
RUN;
PROC SORT DATA=XP6B;
BY USUBJID PARAMCD ORD;
RUN;
DATA XP6C;
SET XP6B;
LENGTH DTYPE $20.;
IF ORD=1 AND ORD1=3 AND SUBJIDN1=SUBJIDN AND PAR1=PARAMN THEN DO;
AVISITN = 106;
AVISIT = "DAY 6/DISCHARGE CONFINEMENT";
ATPTN = 9;
ATPT = "DAY 6/DISCHARGE CONFINEMENT";
DTYPE = "LOCF";
OUTPUT;
END;
RUN;

DATA XP7(DROP=SUBJIDN);
SET XP5 XP6C;
IF SUBSTR(PARAMCD,1,1)="W" THEN ANL01FL = " ";
AVISIT=PROPCASE(AVISIT);
IF PARAMCD in ('FEVFC' 'FEVMEAS' 'FEVPCT' 'FEVPRED' 'FVCMEAS' 'FVCPCT' 'FVCPRED' 'INTP' 'MEF' ) AND DTYPE="LOCF" THEN DELETE;
IF DTYPE="LOCF" THEN ABLFL=" ";
RUN;
PROC SORT DATA=XP7; BY USUBJID PARAMCD AVISITN; RUN;

/*COMBINE ADSL AND XP4*/

DATA XP8;
MERGE ADSL XP7(IN=B);
BY USUBJID;
IF B;
FORMAT ADTM DATETIME13. ADT DATE9. ;
LENGTH APERIODC $10. TRTP TRTA $40. ASPER 8. ASPERC $40.;
LENGTH ADAY APERIOD TRTPN TRTAN 8.;
IF LENGTH(XPDTN)>10 THEN ADTM=DHMS(INPUT(SCAN(XPDTN,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(XPDTN,2,'T'),TIME5.)),MINUTE(INPUT(SCAN(XPDTN,3,'T'),TIME5.)),0);
ELSE IF LENGTH(XPDTN)=10 THEN ADTM=DHMS(INPUT(XPDTN,YYMMDD10.),0,0,0);
IF ADTM NE . THEN ADT=DATEPART(ADTM);
IF NMISS(ADT,TRTSDT)=0 THEN ADAY=ADT-TRTSDT+1;
IF LENGTH(BASEDTC)>10 THEN
BDTM=DHMS(INPUT(SCAN(BASEDTC,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(BASEDTC,2,'T'),TIME5.)),MINUTE(INPUT(SCAN(BASEDTC,3,'T'),TIME5.)),0);
;
ELSE IF LENGTH(BASEDTC)=10 THEN BDTM=DHMS(INPUT(BASEDTC,YYMMDD10.),0,0,0);
APERIOD=1;
APERIODC="Period 1";
TRTP=STRIP(TRT01P);
TRTPN=TRT01PN;
TRTA=STRIP(TRT01A);
TRTAN=TRT01AN;

IF AVISITN <101 THEN ASPERC="Pre-Randomization Period";
ELSE IF 101<=AVISITN<=106 THEN ASPERC="Confinement Period";
ELSE IF 106<AVISITN<=191 THEN ASPERC="Ambulatory Period";
ELSE IF AVISITN>191 THEN ASPERC="Safety Follow-up Period";
ASPER=INPUT(PUT(ASPERC,$ASPR.),BEST.);

DROP TRT01;;

RUN;
PROC SORT DATA=XP8; BY USUBJID PARAMN PARCAT1 AVISITN;RUN;
DATA XP8A;
SET XP8;
ATM=TIMEPART(ADTM);
ADT=DATEPART(ADTM);
RUN;
PROC SORT DATA=XP8A;BY USUBJID PARAMN AVISITN ADT ATM; RUN;
DATA XP8B;

```

```

SET XP8A(DROP=ANL01FL);
LENGTH ANL01FL $2.;
BY USUBJID PARAMN AVISITN ADT ATM;
IF INDEX(upcase(AVISIT),"UNSCHEDULED") >0 OR PARAMCD="XPALL" OR XPSTAT EQ 'NOT DONE' then ANL01FL =" ";
ELSE IF BDTM1 NE . AND ADT>DATEPART(BDTM1)AND FIRST.AVISITN THEN ANL01FL ="Y";
ELSE IF BDTM1=. AND TRTSDTM NE . AND ADT>DATEPART(TRTSDTM) THEN ANL01FL ="Y";
/* ELSE IF ABLFL NE " " AND BDTM GE TRTSDTM THEN ANL01FL="Y";*/
/* ELSE IF ABLFL EQ " " AND ADTM GT TRTSDTM THEN ANL01FL="Y";*/
ELSE IF ABLFL="Y" THEN ANL01FL="Y" ;
IF SUBSTR(PARAMCD,1,1)="W" THEN ANL01FL="";
/* ELSE IF LAST.PARCAT1 AND FIRST.AVISITN=0 THEN ANL01FL="";*/
/* ELSE ANL01FL="Y";*/;
IF SAFBFL NE "Y" OR SAFAFL NE "Y" THEN ANL01FL=" ";
AVISIT=PROPCASE(AVISIT);
RUN;*/;

```

```

PROC SORT DATA=XP8B OUT=ADXP; BY USUBJID AVISITN ATPTN PARAMCD;RUN;

```

```

%m_attrib_adam(dset=ADXP);
data adxp;
set adxp;
if basec="02.12" then basec="2.12";
if avalc="02.79" then avalc="2.79";
if avalc="02.12" then avalc="2.12";
run;

```

```

DATA ADAM.ADXP(LABEL="Pulmonary Function Analysis Dataset");
SET ADXP;
RUN;

```

```

proc sort data=adxp; by usubjid xpseq; run;
proc sort data=qadam.qadxp out=qadxp; by usubjid xpseq; run;

```

```

PROC COMPARE BASE=ADXP COMPARE=QADXP ; RUN;
%M_LOGCHK;

```

```

/**/
/*data adxp;*/
/*set adam.adxp;*/
/*run;*/
/*data qadxp;*/
/*set qadam.qadxp;*/
/*run;*/
/**/
/**/
/* proc sql;*/
/**/
/*create table prd as select count(*),paramcd from adam.adxp group by paramcd;*/
/*create table qc as select count(*),paramcd from qadam.qadxp group by paramcd;*/
/**/
/**/
/* quit;*/
/* proc compare b=prd c=qc;run;*/

```