

```

/*=====
| Covance Study Number      : 000000106343
| Program Name              : d_adqssu.sas
| Purpose                   : create ADQSSU
| Input Data                : ADAM.ADSL,SDTM.QS
| Output Data               : ADAM.ADQSSU
|
| Macros Called             : m_printto, m_logchk, m_attrib_adam
|
| Originally Performed by   : Keerthi Pothuri
| Date                     : 27Mar2015
|
+=====
|
| Modification History      :
|
| Programmer                :
| Date                     :
| Reason for Change         :
+=====*/
options notes source source2 nofullstimer validvarname=upcase missing=' ';

libname sdtm "/cvn/projects/prj/data/000000106343/datasets/sdtm/sdtmx";

proc datasets lib=work nolist memtype=data kill; quit;
* macro to save output and log to appropriate areas ;
%m_printto(route=YES);

*=====;
* START OF PROGRAM CODE
*=====;
*****;
* bring in ADSL ;
*****;
data adsl;
  set adam.adsl;
  drop studyid;
  *keep studyid usubjid subjid: siteid age sex: race height weightb1 bmi ucpdgr1 ucpdgr1n nicogr1 nicogr1n targr1 targr1n
  /*cobl*/ enrlfl scrffl complfl saffl pprotfl fasfl randfl trt: dthfl enfl exfl fupfl EXNOTRFL RANDDT;
run; /*670 - 07*/ /*659*/

*****;
* bring in QS ;
*****;
data qs (drop=paramn1 paramn2);
  set sdtm.qs (where = (qscat = 'QUESTIONNAIRE ON SMOKING URGES'));
  if QSTESTCD ne "QSALL";

* parameter variables ;
parcat1 = propcase(qscat);
parcat1n = 1;

if qstestcd in ('QSU01' 'QSU03' 'QSU06' 'QSU07' 'QSU10') then do;
  parcat2 = 'Factor 1 - Reward';
  parcat2n = 1;
end;
else if qstestcd in ('QSU02' 'QSU04' 'QSU05' 'QSU08' 'QSU09') then do;
  parcat2 = 'Factor 2 - Relief';
  parcat2n = 2;
end;

paramcd = qstestcd;
param = propcase(qstest, '.');

paramn1 = substr(qstestcd,4);
if paramcd ne 'QSU10' then paramn2=trim(left(tranwrd(paramn1,'0','')));
paramn = input(paramn2, best.);
if paramn = . then paramn = 10;

* analysis variables ;
if UPCASE(QSSTRESC)='STRONGLY DISAGREE' then aval=1;
else if UPCASE(QSSTRESC)='DISAGREE' then aval=2;
else if UPCASE(QSSTRESC)='SOMEWHAT DISAGREE' then aval=3;
else if UPCASE(QSSTRESC)='DO NOT AGREE OR DISAGREE' then aval=4;
else if UPCASE(QSSTRESC)='SOMEWHAT AGREE' then aval=5;
else if UPCASE(QSSTRESC)='AGREE' then aval=6;
else if UPCASE(QSSTRESC)='STRONGLY AGREE' then aval=7;
*else put 'USER WARN' 'ING: check response as not in codelist: ' qstestcd = qsstresc ;

```

```

AVALC = propcase(qsstresc, '.');

qsdte_1=dhms(input(substr(qsdte,1,10),ymmdd10.) ,0,0,input(substr(qsdte,12),time5.));
format qsdte_1 datetime13.;

run; /*16,458*/ /*20,230 - 07*/

proc sort data=qs; by usubjid paramcd visitnum qsdte_1; run;
data qs1;
set qs;
by usubjid paramcd visitnum qsdte_1;

/*ADTM*/
if last.visitnum then ADTM=qsdte_1;
format ADTM datetime13.;

/*ADT, ATM*/
ADT=datepart(ADTM);
format ADT date9.;
ATM=timepart(ADTM);
format ATM time5.;
run;

proc sort data=adsl out=adsl_1 (keep=usubjid complfl disccat trtsdt); by usubjid complfl disccat trtsdt; run;
data qs_adsl;
length AVISIT $40;
merge qs1(in=a) adsl_1;
by usubjid;
if a;

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

*Visits;
if COMPLFL = "Y" then do;
AVISIT = VISIT;
AVISITN = VISITNUM;
end;
if find(DISCCAT,"Discontinued", 'I')>0 then do;
if visit = "DAY 6/DISCHARGE CONFINEMENT" and ADAY not in (6, .) then do;
AVISIT="DAY "||strip(put(ADAY, best.));
AVISITN=ADAY+100;
end;
else if visit = "DAY 91/DISCHARGE AMBULATORY" then do;
if 7<ADAY<31 then do;
AVISIT="DAY 30";
AVISITN=130;
end;
else if 32<ADAY<61 then do;
AVISIT="DAY 60";
AVISITN=160;
end;
else do;
AVISIT=VISIT;
AVISITN=VISITNUM;
end;
end;
else do;
AVISIT=VISIT;
AVISITN=VISITNUM;
end;
end;

run; /*20,230 - 07*/ /*16,458*/

* derive factor scores ;
proc sort data = qs_adsl;
by studyid usubjid qscat parcat1n parcat1 parcat2n parcat2 epoch avisitn avisit adtm adt atm qsdte qsd;
run;

proc summary data = qs_adsl noprint;
var aval;
by studyid usubjid qscat parcat1n parcat1 parcat2n parcat2 epoch avisitn avisit adtm adt atm qsdte qsd;
output out = mfactor(drop = _) mean = mean n = n nmiss = nmiss;
run; /*4046 - 07*/

* derive overall score;
proc sort data = qs_adsl;
by studyid usubjid qscat parcat1n parcat1 epoch avisitn avisit adtm adt atm qsdte qsd;

```

```

run;

proc summary data = qs_ads1 noprint;
  var aval;
  by studyid usubjid qscat parcat1n parcat1 epoch avisitn avisit adtm adt atm qsdte qsd;
  output out = mtotal(drop = _) mean = mean n = n nmiss = nmiss;
run; /*2023 - 07*/

data mean (drop = n nmiss mean);
length param $25;
set mfactor mtotal;

paramtyp = 'DERIVED';
dtype = 'AVERAGE';

if n>2 then do;
  aval = mean;
  avalc = strip(put(aval,best.));
  if parcat2n = 1 then do;
    paramcd = 'QSUFAC1';
    paramn = 11;
    param = 'Reward';
  end;
  else if parcat2n = 2 then do;
    paramcd = 'QSUFAC2';
    paramn = 12;
    param = 'Relief';
  end;
end;

if n>5 then do;
  paramcd = 'QSUTOTAL';
  paramn = 13;
  param = 'Total Score';
  parcat2='Total Score';
  parcat2n=3;
end;

if paramcd = "" then delete;
run; /*6069 - 07*/ /*4,941*/

data qs2 (drop=complf1 disccat trtsdt);
set qs_ads1 mean;
run; /*26,299 - 07*/ /*21,399*/

proc sort data=qs2; by usubjid; run;
data qs3;
merge ads1 qs2(in=a);
by usubjid;
if a;

/*ADAY - calculating for new paramcds*/
if not missing (adt) and not missing (trtsdt) then aday = adt - trtsdt + 1;

/*TRT:*/
TRTP=TRT01P;
TRTPN=TRT01PN;
TRTA=TRT01A;
TRTAN=TRT01AN;

/*Aperiod, Aperiodc*/
/* aperiod=1;*/
/* if not missing(aperiod) then do;*/
/* aperiodc = 'Period ' || put(aperiod,1.);*/
/* end;*/
run;

/*ABLFL for new paramcds*/
proc sort data=qs3; by paramcd usubjid avisitn adtm; run;
data qs_fact qs_fact_1;
set qs3;
by paramcd usubjid avisitn adtm;

if paramcd in ("QSUFAC1", "QSUFAC2", "QSUTOTAL") and armcd="SMABST" and avisit="DAY 1" and missing(atm) then do;
  ABLFL="";
  output qs_fact; /*0*/
end;
else output qs_fact_1; /*21,399*/
run;

```

```

data qs_fact_2 qs_fact_3;
  set qs_fact_1;
  by paramcd usubjid avisitn adtm;
  if armcd in ("MCC", "THS 2.2M", "SMABST") then do;
    if paramcd in ("QSUFAC1", "QSUFAC2", "QSUTOTAL") and adtm<trtsdtm then do;
      ablf1="Y";
      output qs_fact_2; /*942*/
    end;
  end;
  if ablf1 ne "Y" then output qs_fact_3; /*20,457*/
run;
data qs_fact_4;
  set qs_fact_2;
  by paramcd usubjid avisitn adtm;

  if last.usubjid then ablf1="Y";
run;
data ablf1; /*21,399*/
  set qs_fact qs_fact_3 qs_fact_4;
run;

*****;
* Calculate changes from baseline (Screening) ;
*****;
*baseline, change ;
data base (rename=(adt=adt_ avalc=basec aval=base));
  set ablf1;
  where paramcd in ("QSUFAC1", "QSUFAC2", "QSUTOTAL") and ABLFL='Y';

  keep usubjid paramcd adt avalc aval;
run;
proc sort data=BASE; by usubjid paramcd; run;
proc sort data=ablf1; by usubjid paramcd; run;
proc sql noprint;
  create table new as select distinct (A.*), b.adt_, b.basec, b.base
  from ablf1 as A left join BASE B
  on A.usubjid=B.usubjid and A.paramcd=B.paramcd;
quit;
data change;
  set new;

  if ADT<ADT_ then do;
    BASEC=" ";
    BASE=.;
  end;
  else if qsstat="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*/
If . < AVISITN < 101 then ASPER=1;
  Else If 101 <= AVISITN <= 106 then ASPER=2;
  Else if 106 < AVISITN <= 191 then ASPER=3;
  Else if AVISITN > 191 then ASPER=4;

if ASPER=1 then ASPERC = 'Pre-Randomization Period';
if ASPER=2 then ASPERC = 'Confinement Period';
if ASPER=3 then ASPERC = 'Ambulatory Period';
if ASPER=4 then ASPERC = 'Safety Follow-up Period';

/*APUPER*/
  If 101 <= AVISITN <= 106 then APUPER=1;
  Else If 106 < AVISITN <= 131 then APUPER=2;
  Else if 131 < AVISITN <= 161 then APUPER=3;
  Else if 161 < AVISITN <= 191 then APUPER=4;

```

```

If APUPER=1 then APUPERC="Period 1";
If APUPER=2 then APUPERC="Period 2";
If APUPER=3 then APUPERC="Period 3";
If APUPER=4 then APUPERC="Period 4";
run;

proc sort data=change; by usubjid paramn avisitn qsdtc_1; run;
data qs4 qs5;
set change;
if qsstat="NOT DONE" then output qs4;
else output qs5;
run;
data qs6;
set qs5;
by usubjid paramn avisitn qsdtc_1;

/*Anl01f1*/
if randf1="Y" then do;
if paramcd in ("QSUFAC1", "QSUFAC2", "QSUTOTAL") then do;
if adt_ ne . and adt>=adt_ and first.avisitn then ANL01FL = "Y";
else if ablf1 ne "Y" and TRTSDT ne . and adt>=TRTSDT and first.avisitn then ANL01FL = "Y";
end;
end;

if INDEX(UPCASE(AVISIT),'UNSCHEDULED') ^= 0 then ANL01FL='';
run;
data comb;
set qs6 qs4;
run;

data qs7;
set comb;

awlo=dhms(0,20,0,0);
awhi=dhms(0,23,0,0);
awrange=strip(put(awlo,TIME5.))||'-'||strip(put(awhi,TIME5.));
format awlo awhi TIME5.;
run;

proc sort data=qs7; by usubjid; run;

DATA qs8;
SET qs7;

IF QSSTAT NE 'NOT DONE' THEN DO;
IF ATM<AWLO THEN DO;
DEVN=FLOOR((ATM-AWLO)/60);
DEVWC=COMPRESS(PUT(FLOOR((ATM-AWLO)/60),BEST.));
END;
ELSE IF ATM>AWHI THEN DO;
DEVN=CEIL((ATM-AWHI)/60);
DEVWC=COMPRESS(PUT(CEIL((ATM-AWHI)/60),BEST.));
END;
END;

IF NOT MISSING(DEVWC) THEN DO;
IF INDEX(DEVWC,'-')=0 THEN DEVWC=CATS(CATS('+',DEVWC),' min');
ELSE IF INDEX(DEVWC,'-') THEN DEVWC=CATS(DEVWC,' min');
END;

IF ABLFL='Y' AND MISSING(PARAMTYP) THEN ABLFL='';

RUN;

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

*options replace;

data adqssu (drop=AVALC BASEC DEVWC /*EPOCH QSREASND*/ rename=(AVALC_=AVALC BASEC_=BASEC DEVWC_=DEVWC /*QSREASND_=QSREASND EPOCH_=EP
OCH*/));
length AVALC_ BASEC_ $50 DEVWC_ $10 /*EPOCH_ $23 QSREASND_ $154*/;
set qs8;
AVALC_=AVALC;
BASEC_=BASEC;
DEVWC_=DEVWC;
AVISIT=propcase(AVISIT);

```

```
/* EPOCH_=EPOCH;*/
/* QSREASND_=QSREASND;*/
run;

%m_attrib_adam(dset=adqssu);

proc sort data = adqssu out = adam.adqssu(label = 'Smoking Urges Analysis Dataset');
    BY USUBJID AVISITN PARCAT1 PARAMCD;
run;

*options noreplace;

*proc printto; *run;

%m_logchk;
*=====;
* END OF PROGRAM CODE ;
*=====;
```