

```

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_2ADQSSU.sas;
%put NOTE: Purpose : create ADQSSU dataset;
%put NOTE: ;
%put NOTE: Input Data : STDLIB.ADQSSU SDTM.QS ADAM.ADSL;
%put NOTE: Output : ADAM.ADQSSU;
%put NOTE: Macros Called : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by : cvn_smulholl;
%put NOTE: Creation Date : 2013-11-22;
%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: 02Dec2013 SM 1) Remove mean from output data;
%put NOTE: 04May2014 KB 2) Amended format of PARCAT1 and
PARCAT2;
%put NOTE: 04May2014 KB 3) Added EPOCH to keep statement;
%put NOTE: 04May2014 KB 4) Amended sorting by key variables;
%put NOTE: 04May2014 KB 5) Added DEVN and DEVWC;
%put NOTE: 04May2014 KB 6) Commented out baseline and AEOEFL
section as not needed;
%put NOTE: 04May2014 KB 7) Removed condition in windows as
windows should be on all questions;
%put NOTE: 04May2014 KB 8) Amended MEAN;
%put NOTE: 04May2014 KB 9) Added USUBJID to warning;
%put NOTE: 07May2014 KB 10) Amended formats of AWRANGE AVISITN
DEVN;
%put NOTE: 08May2014 KB 11) Reverted update 6;
%put NOTE: 09May2014 KB 12) Amended ANL01FL;
%put NOTE: 14May2014 KB 13) Added PCHG;
%put NOTE: 15May2014 KB 14) Removed ABLFL for non derived
parameters;
%put NOTE: 27Jul2014 KB 15) Added EXNOTRFL;
%put NOTE: 13Sep2014 KB 16) Amended key variables;
%put NOTE: 13Sep2014 KB 17) Amended ABLFL;
%put NOTE: 13Sep2014 KB 18) AMended user warning for results;
%put NOTE: ;
%put NOTE: ;

```

```

%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 ucpdgr1 ucpdgrln nicogr1 nicogrln targr1 targrln cobl
        enrfl scrffl complfl saffl fasfl pprotfl randfl trt: trt01:
trt01: dthfl enfl EXNOTRFL exfl fupfl; /* 15) KB 27Jul2014 */
run;

*****;
* bring in QS ;
*****;

data qs;
    set sdtm.qs(where = (qscat = 'QUESTIONNAIRE ON SMOKING URGES'));
    format paramcd $8. parcat1 parcat2 /*$100.*/$200. avisit $40.
paramn parcatln parcat2n AVISITN 8. aval best. param $100. avalc $50. adt
date9. /* 2) KB 04May2014 */ /* 10) KB 07May2014 */
    adtm datetime13. atm time5. /*ablfl $2.*/; /* 17) KB 13Sep2014 */
    * parameter variables ;
    parcat1 = propcase(qscat);
    parcatln = 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, '.');

    paramn = input(substr(qstestcd,4),best.);

    * analysis variables ;
    if qsstresc='STRONGLY DISAGREE' then aval=1;
    else if qsstresc='DISAGREE' then aval=2;

```

```

        else if qsstresc='SOMEWHAT DISAGREE' then aval=3;
        else if qsstresc='DO NOT AGREE OR DISAGREE' then aval=4;
        else if qsstresc='SOMEWHAT AGREE' then aval=5;
        else if qsstresc='AGREE' then aval=6;
        else if qsstresc='STRONGLY AGREE' then aval=7;
        else IF NOT MISSING(QSSTRESC) THEN put 'USER WARN' 'ING: check
response as not in codelist: ' USUBJID= qstestcd = qsstresc = ; /* 9) KB
04May2014 */ /* 18) KB 13Sep2014 */
        avalc = propcase(qsstresc, '.');

        * baseline flag;
/*      ablfl = qsbflfl; */ /* 17) KB 13Sep2014 */

        * visit details ;
        avisit = propcase(visit);
        avisitn = visitnum;

        * dates;
        length qsdtc1 $19.;
        qsdtc1=qsdtc;
        if length(qsdtc) gt 10 then do;
            adtm = input(qsdtc1,e8601dt.);
            adt = datepart(adtm);
            atm = timepart(adtm);
        end;
        else if length(qsdtc) = 10 then adt = input(qsdtc, yymmdd10.);

        keep usubjid qsseq param: parcat: aval: /*ablfl*/ avisit: adt: atm
qsstat qsreasnd qsdtc qsdyc EPOCH; /* 3) KB 04May2014 */ /* 17) KB
13Sep2014 */
run;

* derive factor scores ;

proc sort data = qs;
    by usubjid parcat1n parcat2n avisitn adtm adt atm /*ablfl*/; /* 17)
KB 13Sep2014 */
run;

proc summary data = qs noprint;
    var aval;
    by usubjid parcat1n parcat1 parcat2n parcat2 avisitn avisit adtm
adt atm /*ablfl*/; /* 17) KB 13Sep2014 */
    output      out = mfactor(drop = _:)      mean = mean n = n nmiss =
nmiss;
run;

* derive overall score;
proc sort data = qs;
    by usubjid parcat1n avisitn adtm adt atm /*ablfl*/; /* 17) KB
13Sep2014 */
run;

proc summary data = qs noprint;

```

```

        var aval;
        by usubjid parcat1n parcat1n avisitn avisitn adtm adt atm /*ablfl*/;
/* 17) KB 13Sep2014 */
        output      out = mtotal(drop = _:)      mean = mean n = n nmiss =
nmiss;
run;

data mean(drop = n nmiss);
        set mfactor(in = a) mtotal(in = b);
        format paramcd $8. paramn 8. param $100. paramtyp dtype $10. aval
best. avalc $50.;
        paramtyp = 'DERIVED';
        dtype = 'AVERAGE';
        if nmiss = 0 or (nmiss > 0 and (n/nmiss)*100 > 50) then do;
/*      aval = round(mean,1.);*/
        AVAL=MEAN; /* 8) KB 04May2014 */
        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;
                else do;
                        paramcd = 'QSUTOTAL';
                        paramn = 13;
                        param = 'Total Score';
                end;
        end;
        end;
        else do;
                aval = .;      * > 50% missing;
                avalc = ' ';
        end;
run;

/* 17) START KB 13Sep2014 */
PROC SORT DATA=MEAN(WHERE=(AVISIT IN ('Day -1' 'Day 0' 'Day 1')))
OUT=MEAN3;
        BY USUBJID PARCAT1N PARCAT2N PARAMCD AVISITN ATM;
RUN;

DATA BASELINE;
        SET MEAN3(WHERE=(AVISIT=('Day 1')));

        RENAME ADT=DAY1DT;
        KEEP USUBJID ADT PARAMCD;
RUN;

PROC SORT DATA=BASELINE;
        BY USUBJID PARAMCD;

```

```

RUN;

PROC SORT DATA=MEAN3;
  BY USUBJID PARAMCD;
RUN;

DATA BASELINE2;
  MERGE BASELINE MEAN3;
  BY USUBJID PARAMCD;
RUN;

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

  KEEP USUBJID TRTSDTM TRT01A;
RUN;

DATA BASELINE2A;
  MERGE BASELINE2(IN=A) ADSLTM;
  BY USUBJID;
  IF A;
RUN;

DATA MEAN4;
  SET BASELINE2A;
  ATTRIB ABLFL2 LENGTH=$2.;

  IF TRT01A='SA' THEN DO;
    IF DAY1DT NE . THEN DO;
      IF ADTM<DHMS(DAY1DT,6,30,0) AND
INDEX(UPCASE(AVISIT),'UNSCHED')=0 THEN ABLFL2='Y';
      END;
    ELSE IF DAY1DT EQ . THEN DO;
      IF INDEX(UPCASE(AVISIT),'UNSCHED')=0 THEN ABLFL2='Y';
      END;
    END;
  ELSE IF TRT01A IN ('CC' 'THS 2.2') THEN DO;
    IF ADTM<TRTSDTM AND INDEX(UPCASE(AVISIT),'UNSCHED')=0 THEN
ABLFL2='Y';
    END;
  ELSE IF MISSING(TRT01A) THEN DO;
    IF INDEX(UPCASE(AVISIT),'UNSCHED')=0 THEN ABLFL2='Y';
    END;

  DROP TRT01A TRTSDTM;
RUN;

DATA MEAN4A;
  SET MEAN4;
  WHERE MISSING(DAY1DT);
RUN;

PROC SORT DATA=MEAN4A;

```

```

        BY USUBJID PARCAT1 PARCAT2 PARAMCD AVISITN AVISIT;
RUN;

DATA MEAN4B;
    SET MEAN4A;
    BY USUBJID PARCAT1 PARCAT2 PARAMCD AVISITN AVISIT;

    IF LAST.PARAMCD THEN ABLFL3='Y';

    KEEP USUBJID PARCAT1 PARCAT2 PARAMCD AVISITN AVISIT ABLFL3;
RUN;

PROC SORT DATA=MEAN4;
    BY USUBJID PARCAT1 PARCAT2 PARAMCD AVISITN AVISIT;
RUN;

DATA MEAN4C;
    MERGE MEAN4 MEAN4B;
    BY USUBJID PARCAT1 PARCAT2 PARAMCD AVISITN AVISIT;

    IF NOT MISSING(ABLFL3) THEN ABLFL2=ABLFL3;
RUN;

PROC SORT DATA=MEAN4C (WHERE=(ABLFL2='Y')) OUT=MEAN5;
    BY USUBJID PARAMCD AVISITN ATM;
RUN;

DATA MEAN6;
    SET MEAN5;
    BY USUBJID PARAMCD AVISITN ATM;
    FORMAT ABLFL $2.;

    IF LAST.PARAMCD AND LAST.ATM THEN ABLFL='Y';
    DROP ABLFL2 ABLFL3 DAY1DT;
RUN;

PROC SORT DATA=MEAN6;
    BY USUBJID PARCAT1N PARCAT2N PARAMCD AVISITN ATM;
RUN;

PROC SORT DATA=MEAN;
    BY USUBJID PARCAT1N PARCAT2N PARAMCD AVISITN ATM;
RUN;

DATA MEAN7;
    MERGE MEAN MEAN6;
    BY USUBJID PARCAT1N PARCAT2N PARAMCD AVISITN ATM;
RUN;
/* 17) END KB 13Sep2014 */

* set together ;

data qs2;

```

```

        set qs /*mean*/MEAN7; /* 17) KB 13Sep2014 */
run;

*****;
* change from baseline ;
*****;
/* 11) START KB 08May2014 */
/* 6) START KB 04May2014 */
proc sort data = qs2;
    by usubjid paramn;
run;

data base(keep = usubjid paramn base: bvis);
    set qs2(where = (ablfl = 'Y' AND PARAMN IN (11 12 13))); * per SAP
only calculate changes from baseline for factors and overall ; /* 11) KB
08May2014 */
    format base best. basec $50.;
    base = aval;
    basec = avalc;
    bvis = avisitn;
run;

data change(drop = bvis);
    merge qs2 base;
    by usubjid paramn;
    format chg PCHG best.; /* 13) KB 14May2014 */
    if avisitn gt bvis then DO; /* 13) KB 14May2014 */
        chg = aval - base;
        /* 13) START KB 14May2014 */
        IF BASE NE 0 THEN DO;
            PCHG=(CHG/BASE)*100;
        END;
        ELSE DO;
            PCHG=(CHG/1)*100;
        END;
    END;
    /* 13) END KB 14May2014 */
run;

proc sort data = change;
    by usubjid paramn avisitn;
run;
/**/
/*PROC SORT DATA=QS2;*/
/*    BY USUBJID PARAMN AVISITN;*/
/*RUN;*/
/* 11) END KB 08May2014 */
/* 6) END KB 04May2014 */

data change2;
    set change/*QS2*/; /* 6) KB 04May2014 */ /* 11) KB 08May2014 */
    by usubjid paramn avisitn;
    format anl01fl $2.;

```

```

        * determine if any unscheduled;
        if index(upcase(avisit),'UNSCHEDULED') or paramcd = 'QSALL' then
anl01fl = ' ';
        else if last.avisitn and first.avisitn = 0 then anl01fl = ' ';
        else anl01fl = 'Y';
        if anl01fl = ' ' then put 'Check reason for exclusion from
analysis: ' usubjid = param = avisit = ;
run;

* end of exposure flag;
/* 6) START KB 04May2014 */
/*proc sort data = change2 out = eos(where = (not missing(avalc)));*/
/*  by usubjid paramn avisitn adtm;*/
/*run;*/
**/
/*data eos2(keep = usubjid paramn avisitn aeofl);*/
/*  set eos;*/
/*  format aeofl $2.; */
/*  by usubjid paramn;*/
/*  if last.paramn then do;*/
/*      aeofl = 'Y';*/
/*      output;*/
/*  end;*/
/*run;*/
/* 6) END KB 04May2014 */

*merge back onto data;
data qs3;
    /*merge*/SET change2 /*eos2*/; /* 6) KB 04May2014 */
/*  by usubjid paramn avisitn;*/ /* 6) KB 04May2014 */
run;

/* 6) START KB 04May2014 */
PROC SORT DATA=QS3;
    BY USUBJID;
RUN;
/* 6) END KB 04May2014 */

*****;
* Combine ADSL and QS data *;
*****;
* treatment period;
%_mtotper;

data slqssu(drop = trt01: tr01: MEAN); /* 1) SM 02Dec2013 */
    merge adsl qs3(in = a);
    by usubjid;
    if a;          * only include subjects with data ;
    format aperiod trtan trtpn aday 8. trta trtp $40. aperiodc $10.;
    aday = adt - trtsdt + 1;
    * allocate tretament and period;
    %_mperall(dvar1 = adtm, dvar2 = adt);
    aperiodc = 'Period ' || put(aperiod,1.);

```



```

        * only summarise FAS population;
        if fasfl = 'N' then anl01fl = ' ';
        if param not in ('Reward','Relief','Total Score') then anl01fl = '
';
run;

*****;
*Determine assessment window;
*****;

proc sort data = slqssu out = qswind(where = (not(fasfl = 'N' /*or (param
not in ('Reward','Relief','Total Score'))*/))); * as assessed above; /*
7) KB 04May2014 */
    by subjidn aday;
run;

data window;
    set qswind;
    by subjidn aday;
    format awlo awhi time5. awrange /*$20.*/$50.; /* 10) KB 07May2014
*/
    awlo = '20:00't;
    awhi = '23:00't;
    awrange = strip(put(awlo,time5.))||'-'||strip(put(awhi,time5.));
/* 12) START KB 09May2014 */
/*    oldanl = anl01fl;*/
/*    if not (awlo <= atm <= awhi) then anl01fl = ' ';*/
/**/
/*    if oldanl ne anl01fl then put 'USER WARN' 'ING: excluded from
summaries as outside assessment window: ' subjidn = aday = atm = awrange
=;*/
/* 12) END KB 09May2014 */
    keep subjidn aday atm awrange awlo awhi anl01fl paramn;
run;

* add back to data;
proc sort data = window;
    by subjidn aday paramn;
run;

proc sort data = slqssu;
    by subjidn aday paramn;
run;

data slqssu2;
    merge slqssu window;
    by subjidn aday paramn;
run;

/* 5) START KB 04May2014 */
DATA SLQSSU3;
    SET SLQSSU2;
    FORMAT DEVWC $10. DEVN BEST.; /* 10) KB 07May2014 */

```

```

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=''; /* 14) KB 15May2014
*/

/* DROP ABLFL;*/ /* 11) KB 08May2014 */
RUN;
/* 5) END KB 04May2014 */

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

options replace;

data adqssu;
  set stdlib.adqssu /*slqssu2*/SLQSSU3; /* 5) KB 04May2014 */
run;

proc sort data = adqssu out = adam.adqssu(label = 'Smoking Urges Analysis
Dataset');
/* by usubjid avisitn parcat1n parcat2n paramn;*/
/* BY USUBJID AVISITN;*/ /* 4) KB 04May2014 */
  BY USUBJID AVISITN PARAMCD; /* 16) KB 13Sep2014 */
run;

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

```