

```

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_2ADQSNd.sas;
%put NOTE: Purpose : create ADQSNd dataset;
%put NOTE: ;
%put NOTE: Input Data : STDLIB.ADQSNd SDTM.QS ADAM.ADSL;
%put NOTE: Output : ADAM.ADQSNd;
%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) Amend assessment windows;
%put NOTE: 04May2014 KB 2) Amended sorting by key variables;
%put NOTE: 04May2014 KB 3) Amended pull out of FTND data;
%put NOTE: 04May2014 KB 4) Amended format of PARCAT1 and
PARCAT2;
%put NOTE: 04May2014 KB 5) Amended ABLFL to only populate for
MNWS;
%put NOTE: 04May2014 KB 6) Amended AWLO and AWHI to populate
for all MNWS questions;
%put NOTE: 04May2014 KB 7) Added DEVN and DEVWC;
%put NOTE: 04May2014 KB 8) Added EPOCH to keep statement;
%put NOTE: 04May2014 KB 9) Dropped AVALU;
%put NOTE: 04May2014 KB 10) Added check for QSSTAT to
derivation of result;
%put NOTE: 09May2014 KB 11) Amended windows for MNWS and
amended ANL01FL;
%put NOTE: 09May2014 KB 12) Amended ANL01FL;
%put NOTE: 01Jun2014 KB 13) Dropped ASTM as not required;
%put NOTE: 02Jun2014 KB 14) Amended AWLO and AWRANGE to match
windows in SAP;
%put NOTE: 03Jun2014 KB 15) Added format to AVISITN;
%put NOTE: 27Jul2014 KB 16) Added EXNOTRFL;
%put NOTE: 13Sep2014 KB 17) Amended ABLFL;
%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 cob1
    enrfl scrfl complfl saffl fasfl pprofl randfl trt: trt01:
trt01: dthfl enfl EXNOTRFL exfl fupfl; /* 16) KB 27Jul2014 */
run;

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

data qs;
    set sdtm.qs(where = (qscat in (/*'FAGERSTROM TEST FOR NICOTINE
DEPENDENCE'*/'FAGERSTROM TEST FOR NICOTINE DEPENDENCE QUESTIONNAIRE'
'MINNESOTA NICOTINE DEPENDENCE/WITHDRAWAL SCALE'))); /* 3) KB 04May2014
*/
    format paramcd $8. parcat1 parcat2 /*$100.*/$200. avisit $40.
paramn parcatln parcat2n 8. aval AVISITN best. param $100. avalc $50. adt
date9. /* 4) KB 04May2014 */ /* 15) KB 03Jun2014 */
    adm dtetime13. atm time5. /*ablfl $2.*/; /* 17) KB 13Sep2014 */
    * parameter variables ;
    parcat1 = procase(qscat);
    if qscat = /*'FAGERSTROM TEST FOR NICOTINE DEPENDENCE'*/'FAGERSTROM
TEST FOR NICOTINE DEPENDENCE QUESTIONNAIRE' then parcatln = 1; /* 3) KB
04May2014 */
    else if qscat = 'MINNESOTA NICOTINE DEPENDENCE/WITHDRAWAL SCALE'
then do;
        parcatln = 2;
        if qstestcd in ('MNWS01' 'MNWS02' 'MNWS03' 'MNWS04' 'MNWS05'
'MNWS06' 'MNWS07' 'MNWS08' 'MNWS09') then do;
            parcat2 = 'Validated';
            parcat2n = 1;
        end;
        else if qstestcd in ('MNWS10' 'MNWS11' 'MNWS12' 'MNWS13'
'MNWS14' 'MNWS15') then do;
            parcat2 = 'Unvalidated';
            parcat2n = 2;
        end;
    end;

    paramcd = qstestcd;

```

```

param = propcase(qstest, '.');

paramn = input(substr(qstestcd, 5), best.);
if parcatln = 1 then paramn = paramn + 18;

* analysis variables ;
if qscat = /*'FAGERSTROM TEST FOR NICOTINE DEPENDENCE'*/'FAGERSTROM
TEST FOR NICOTINE DEPENDENCE QUESTIONNAIRE' then do; /* 3) KB 04May2014
*/
    if qstestcd = 'FTND01' then do;
        if qsstresc = 'WITHIN 5 MINUTES' then aval = 3;
        else if qsstresc = '6-30 MINUTES' then aval = 2;
        else if qsstresc = '31-60 MINUTES' then aval = 1;
        else if qsstresc = 'AFTER 60 MINUTES' then aval = 0;
        else IF QSSTAT NE 'NOT DONE' THEN put 'USER WARN' 'ING:
check response as not in codelist: ' usubjid = qstestcd = qsstresc = ; /*
10) KB 04May2014 */
        end;
    else if qstestcd in ('FTND02' 'FTND05' 'FTND06') then do;
        if qsstresc = 'YES' then aval = 1;
        else if qsstresc = 'NO' then aval = 0;
        else IF QSSTAT NE 'NOT DONE' THEN put 'USER WARN' 'ING:
check response as not in codelist: ' usubjid = qstestcd = qsstresc = ; /*
10) KB 04May2014 */
        end;
    else if qstestcd = 'FTND03' then do;
        if qsstresc = 'THE FIRST IN THE MORNING' then aval = 1;
        else if qsstresc = 'ANY OTHER' then aval = 0;
        else IF QSSTAT NE 'NOT DONE' THEN put 'USER WARN' 'ING:
check response as not in codelist: ' usubjid = qstestcd = qsstresc = ; /*
10) KB 04May2014 */
        end;
    else if qstestcd = 'FTND04' then do;
        if qsstresc = '10 OR LESS' then aval = 0;
        else if qsstresc = '11-20' then aval = 1;
        else if qsstresc = '21-30' then aval = 2;
        else if qsstresc = '31 OR MORE' then aval = 3;
        else IF QSSTAT NE 'NOT DONE' THEN put 'USER WARN' 'ING:
check response as not in codelist: ' usubjid = qstestcd = qsstresc = ; /*
10) KB 04May2014 */
        end;
    end;
    else if qscat = 'MINNESOTA NICOTINE DEPENDENCE/WITHDRAWAL SCALE'
then do;
        if qsstresc = 'NONE' then aval = 0;
        else if qsstresc = 'SLIGHT' then aval = 1;
        else if qsstresc = 'MILD' then aval = 2;
        else if qsstresc = 'MODERATE' then aval = 3;
        else if qsstresc = 'SEVERE' then aval = 4;
    end;

avalc = propcase(qsstresc, '.');

* baseline flag;

```

```

/* 17) START KB 13Sep2014 */
/* IF QSCAT NE 'FAGERSTROM TEST FOR NICOTINE DEPENDENCE QUESTIONNAIRE'
THEN DO; *//* 5) KB 04May2014 */
/*      ablfl = qsblfl;*/
/*END;*/ /* 5) KB 04May2014 */
/* 17) END KB 13Sep2014 */

```

```

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

```

```

* dates;
length qsdctl $19.;
qsdctl=qsdtc;
if length(qsdtc) gt 10 then do;
    adtm = input(qsdctl,e8601dt.);
    adt = datepart(adtm);
    atm = timepart(adtm);
end;
else if length(qsdtc) = 10 then adt = input(qsdtc,yyymmdd10.);

```

```

keep usubjid qsseq param: parcat: aval: /*ablfl*/ avisit: adt: atm
qsstat qsreasnd qsdtc qsdyc EPOCH; /* 8) KB 04May2014 */ /* 17) KB
13Sep2014 */
run;

```

```

/* 17) START KB 13Sep2014 */
DATA SV;
    SET SDTM.SV(WHERE=(VISIT=('DAY 1')));
    FORMAT DAY DATE9.;

    DAY=INPUT(SCAN(SVSTDTC,1,'T'),YYMMDD10.);
    KEEP USUBJID DAY;
RUN;

```

```

DATA ABLFL;
    SET QS(WHERE=(AVISIT IN ('Screening' 'Day -2' 'Day -1' 'Day 0' 'Day
1') AND QSSTAT NE 'NOT DONE' AND PARCAT1 NE 'Fagerstrom Test For Nicotine
Dependence Questionnaire'));

```

```

    KEEP USUBJID PARAMCD AVISIT ADTM;
RUN;

```

```

PROC SORT DATA=ABLFL;
    BY USUBJID;
RUN;

```

```

DATA ABLFL2;
    MERGE ABLFL(IN=A) SV;
    BY USUBJID;
    IF A;
RUN;

```

```

DATA ADSLTM;

```

```

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

        KEEP USUBJID TRTSDTM TRT01A;
RUN;

DATA ABLFL2A;
    MERGE ABLFL2 (IN=A) ADSLTM;
    BY USUBJID;
    IF A;
RUN;

PROC SORT DATA=ABLFL2A;
    BY USUBJID PARAMCD ADTM;
RUN;

DATA ABLFL3;
    SET ABLFL2A;

    IF TRT01A='SA' THEN DO;
        IF DAY NE . THEN DO;
            IF ADTM<DHMS(DAY,6,30,0) AND
INDEX(UPCASE(AVISIT), 'UNSCHED')=0 THEN ABLFL2='Y';
            END;
        ELSE IF DAY 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;
RUN;

PROC SORT DATA=ABLFL3 (WHERE=(ABLFL2='Y')) OUT=ABLFL4;
    BY USUBJID PARAMCD ADTM;
RUN;

DATA ABLFL5 (WHERE=(ABLFL='Y')) ;
    SET ABLFL4;
    BY USUBJID PARAMCD ADTM;
    FORMAT ABLFL $2.;

    IF LAST.PARAMCD THEN ABLFL='Y';

    KEEP USUBJID PARAMCD AVISIT ABLFL;
RUN;

PROC SORT DATA=ABLFL5;
    BY USUBJID PARAMCD AVISIT;
RUN;

```

```

PROC SORT DATA=QS;
  BY USUBJID PARAMCD AVISIT;
RUN;

DATA QSA;
  MERGE QS ABLFL5;
  BY USUBJID PARAMCD AVISIT;
RUN;
/* 17) END KB 13Sep2014 */

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

proc summary data = /*qs*/QSA(where = (parcat2n ne 2)) noprint; *
  exclude unvalidated MNWS scores 10-15; /* 17) KB 13Sep2014 */
  var aval;
  by usubjid parcat1n parcat1 parcat2n parcat2 avisitn avisit adtm
  adt atm ablfl;
  output      out = mfactor(drop = _:)      sum = sum n = n nmiss =
  nmiss;
run;

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

proc summary data = /*qs*/QSA(where = (parcat1n = 2)) noprint; /* 17) KB
13Sep2014 */
  var aval;
  by usubjid parcat1n parcat1 avisitn avisit adtm adt atm ablfl;
  output      out = mttotal(drop = _:)      sum = sum n = n nmiss =
  nmiss;
run;

data mean(drop = n nmiss);
  set mfactor(in = a) mttotal(in = b);
  format paramcd $8. paramn 8. param $100. paramtyp dtype $10. aval
  best. avalc avalcat1 $50.;
  paramtyp = 'DERIVED';
  dtype = 'SUM';
  if parcat1n = 2 then do;    * mnws scores ;
    if nmiss = 0 or (nmiss > 0 and (n/nmiss)*100 > 50) then do;
      aval = sum;
      avalc = strip(put(aval,best.));
      if parcat2n = 1 then do;
        paramcd = 'MNWRWDS1';
        paramn = 17;
        param = 'Total Score 1';
      end;
    else do;

```

```

        paramcd = 'MNWRWDS2';
        paramn = 18;
        param = 'Total Score 2';
    end;
end;
else do;
    aval = .;          * > 50% missing;
    avalc = ' ';
end;
end;
else do;
    paramcd = 'FTNDSC';
    paramn = 25;
    param = 'Fagerstrom Score';
    if nmiss = 0 then do;
        aval = sum;
        avalc = strip(put(aval,best.));

        if 0 le aval le 3 then avalcat1 = 'Mild';
        else if 4 le aval le 6 then avalcat1 = 'Moderate';
        else if 7 le aval le 10 then avalcat1 = 'Severe';
        else put 'USER WARN' 'ING: check score as outside required
values: ' usubjid = aval = ;
    end;
    else do;
        aval = .;
        avalc = ' ';
        avalcat1 = ' ';
    end;
end;
run;

* set together ;

data qs2;
    set /*qs*/QSA mean; /* 17) KB 13Sep2014 */
run;

*****;
* change from baseline ;
*****;
proc sort data = qs2;
    by usubjid paramn;
run;

data base(keep = usubjid paramn base: bvis);
    set qs2(where = (ablfl = 'Y'));
    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.;
  if avisitn gt bvis then do;
    chg = aval - base;
    if base = 0 then pchg = ((aval - base) / 1) * 100;
    else pchg = ((aval - base) / base ) * 100;
  end;
run;

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

data change2;
  set change;
  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;
proc sort data = change2 out = eos(where = (not missing(avalc) and
parcatln = 2)); * mnws only ;
  by usubjid paramn avisitn adtm;
run;

data eos2(keep = usubjid paramn avisitn aeoeffl);
  set eos;
  format aeoeffl $2.;
  by usubjid paramn;
  if last.paramn then do;
    aeoeffl = 'Y';
    output;
  end;
run;

*merge back onto data;
data qs3;
  merge change2 eos2;
  by usubjid paramn avisitn;
run;

*****;
* Combine ADSL and QS data *;

```



```

*****;
* treatment period;
*_mtotper;

data slqsnd(drop = trt01: tr01: sum);
  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 for MNWS;
  if fasfl = 'N' and parcatln = 2 then anl01fl = ' ';
  if param not in ('Fagerstrom Score','Total Score 1','Total Score
2') then anl01fl = ' ';
run;

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

* pick up all doses;
data doses;
  set adam.addx(in = a where = (astday ge 0) DROP=AVALU) adam.adex(in
= b where = (astday ge 0) DROP=AVALU) adam.adsv(in = c where = (astday ge
0 /*and trta = 'SA'*/)); /* 1) SM 02Dec2013 */ /* 9) KB 04May2014 */
  format astm time5.;
  /* 1) RE-WRITTEN CODE TO GET CORRECT TIMES PER ARM SM 02Dec2013 */
  IF TRTA = 'SA' AND (1 LE ASTDAY LE 5) THEN ASTM =
/*'22:00'T*/'10:00'T; * NO PRODUCT SMOKED SO 10PM AT LATEST ; /* 11) KB
09May2014 */
  ELSE IF ASTDAY = 6 THEN ASTM = '06:30'T;          * NO PRODUCT
RECORDED AFTER DAY 5 SO PRIOR TO START OF DAY;
  else astm = timepart(astdtm);
  IF MISSING(ASTM) THEN DELETE; /* 1) SM 02Dec2013 */
  keep subjidn astday astm astdt;
run;

proc sort data=doses;
  by subjidn astday astm;
run;

/* 11) START KB 09May2014 */
DATA DOSES1;
  SET DOSES;
  BY SUBJIDN ASTDAY ASTM;

  IF FIRST.ASTDAY;
RUN;
/* 11) END KB 09May2014 */

```

```

data doses2;
    set /*doses*/DOSES1; /* 11) KB 09May2014 */
    by subjidn astday;
    format awrange $50. awhi AWLO time5.; /* 11) KB 09May2014 */
    if first.asterday;

/* 11) START KB 09May2014 */
    AWLO=.'06:00'T*/; /* 14) KB 02Jun2014 */
    IF ASTM>DHMS(0,10,0,0) THEN AWHI='10:00'T;
    ELSE AWHI=ASTM;
/* 11) END KB 09May2014 */

    awrange = /*put(*//*astm*//*AWLO,time5.) ||*/ /*'-22:00'*//*'-
'*/STRIP("<")||STRIP(PUT(AWHI,TIME5.)); * window from time of first
cigarette next day must be prior to 10:00; /* 11) KB 09May2014 */ /* 14)
KB 02Jun2014 */
/*    awhi = '22:00't;*/ /* 11) KB 09May2014 */
    rename asterday = aday /*astm = awlo*/; /* 11) KB 09May2014 */
run;

proc sort data = doses2;
    by subjidn aday;
run;

proc sort data = slqsnd out = qswind(where = (/*param in ('Total Score
1','Total Score 2'*/PARCAT1N=2/*)*/)); * mnws scores only; /* 6) KB
04May2014 */
    by subjidn aday;
run;

data window;
    merge qswind(in = a) doses2(in = b);
    by subjidn aday;
    if a and b;
/* 12) START KB 09May2014 */
/*    oldanl = anl01fl;*/
/*    if datepart(adtm) = astdt then do;*/ * make sure qs taken prior to
first exposure on day;
/*        if*/ /*not*/*/* (atm lt awlo) then anl01fl = ' '*/ /* 11) KB
09May2014 */
/*        else if atm gt awhi then anl01fl = ' '*/;
/*    end;*/

/*    if oldanl ne anl01fl then put 'USER WARN' 'ING: excluded from
summaries as outside assessment window: ' subjidn = aday = adtm =
awrange =;*/
/* 12) END KB 09May2014 */

/*    keep subjidn aday adtm awrange awlo awhi astdt anl01fl paramn;*/
run;

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

```

```

run;

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

data slqsnd2;
    merge slqsnd window(drop = astdt);
    by subjidn aday paramn;
    * not included as causes issues in opencdisc 1.4.1;
    *if parcat1n = 2 then avisit = 'Day ' ||strip(put(aday -1,best.));
    * re SAP section 7.3.4 report as Day prior to day collected;
run;

/* 7) START KB 04May2014 */
DATA SLQSND3;
    SET SLQSND2;
    FORMAT DEVWC $10.;

    IF QSSTAT NE 'NOT DONE' THEN DO;
        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;

    DROP ASTM; /* 13) KB 01Jun2014 */
RUN;
/* 7) END KB 04May2014 */

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

options replace;

data adqsnd;
    set stdlib.adqsnd /*slqsnd2*/SLQSND3; /* 7) KB 04May2014 */
run;

proc sort data = adqsnd out = adam.adqsnd(label = 'Nicotine Dependence
Analysis Dataset');
/*    by usubjid avisitn parcat1n parcat2n paramn;*/
    BY USUBJID AVISITN PARCAT1 PARAMCD; /* 2) KB 04May2014 */
run;

options noreplace;
proc printto; run;
*=====;

```

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