

```

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_2ADEG.sas;
%put NOTE: Purpose              : create ADEG dataset;
%put NOTE: ;
%put NOTE: Input Data           : STDLIB.ADEG SDTM.EG SDTM.SUPPEG;
%put NOTE: Output               : ADAM.ADEG;
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_smulholl;
%put NOTE: Creation Date        : 2013-09-27;
%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: 30Nov2013  SM          1) Correct QTcF calculation to cube
root;
%put NOTE:                                     2) Amend DESC variable
derivation and case;
%put NOTE: 30Apr2014  KB          3) Added EPOCH to keep statement;
%put NOTE: 30Apr2014  KB          4) Amended formats of AVAL ATPTN BASE
CHG;
%put NOTE: 30Apr2014  KB          5) Amended EGTEST for QTcF;
%put NOTE: 30Apr2014  KB          6) Dropped SDTM variables in QTcF
derivation;
%put NOTE: 30Apr2014  KB          7) Set SDTM variables to blank for
QTcF;
%put NOTE: 30Apr2014  KB          8) Removed rounding of QTcF;
%put NOTE: 30Apr2014  KB          9) Removed AVAL for INTP;
%put NOTE: 30Apr2014  KB          10) Amended DESCs;
%put NOTE: 30Apr2014  KB          11) Added EGALL to PARAMCDs;
%put NOTE: 30Apr2014  KB          12) Removed format from EGSEQ;
%put NOTE: 14May2014  KB          13) Added TRTSTMF and TRTETMF to keep
statement;
%put NOTE: 14May2014  KB          14) Amended format of ABLFL;
%put NOTE: 14May2014  KB          15) Amended DESC;
%put NOTE: 23Jul2014  ZUH          16) Amended PARAM labels;
%put NOTE: 27Jul2014  KB          17) Added EXNOTRFL;
%put NOTE: 12Sep2014  KB          18) Removed clinical significance from
AVALC;

```

```

%put NOTE: 12Sep2014    KB          19) Added FASFL & PPROTFL to ADSL keep;
%put NOTE: 12Sep2014    KB          20) Amended SHIFT1 for clinical
significance;
%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 ucpdgr: nicogr: targr: cob1
        enrfl scrffl saffl randfl trtsdtm trtsdt trtsday trtedtm
trtedt trteday trt01: tr01:
        enfl EXNOTRFL exfl complfl fupfl dthfl TRTSTMF TRTETMF FASFL
PPROTFL; /* 13) KB 14May2014 */ /* 17) KB 27Jul2014 */ /* 19) KB
12Sep2014 */
run;

proc sort data = adsl;
    by usubjid;
run;

*****;
* pick up SUPPEG ;
*****;

proc transpose data = sdtm.suppeg out = suppeg(drop = _:);
    var qval;
    by usubjid idvarval;
    id qnam;
    idlabel qlabel;
run;

data suppeg2;
    set suppeg;
    /* format egseq 8.; */ /* 12) KB 30Apr2014 */
    egseq = input(idvarval,best.);
run;

proc sort data = suppeg2;
    by usubjid egseq;
run;

```

```

*****;
* calculate QTcF ;
*****;
data qtcf(drop = hr qt);
  LENGTH EGTEST $50; /* 16) ZUH 2014-07-23 */
  merge sdtm.eg(where = (egtestcd = 'HRMEAN') rename = (egstresn =
hr))
          sdtm.eg(where = (egtestcd = 'QTMEAN') rename = (egstresn =
qt));
  by usubjid visitnum;
  egtestcd = 'QTcF';
/*  egtest = 'QTcF - Fridericia Correction Formula';*/
  EGTEST = "QTcF - Fridericia's Correction Formula"; /* 5) KB
30Apr2014 */
  egstresn = /*round*/(qt/((60/hr)**(1/3))/*,1.)*/*; /* 1) SM
30Nov2013 */ /* 8) KB 30Apr2014 */
  egstresc=compress(put(round(egstresn,
0.000000000000000001),best16.)); /*JM 28JUL2014*/
  egorres = trim(egstresc);
  egseq=.;

  DROP EGCAT EGPOS EGMETHOD EPOCH; /* 6) KB 30Apr2014 */
run;

*****;
* Add to EG;
*****;

data eg;
  set qtcf sdtm.eg;
run;

*****;
* bring in EG ;
*****;
proc sort data = eg;
  by usubjid egseq;
run;

data eg2;
  merge eg suppeg2(where = (not missing(usubjid)));
  by usubjid egseq;
  format paramcd $8. /*param*/ avisit $40. paramn /*aval*/ avisitn
/*atptn*/ 8. avalc desc $200. avalu paramtyp dtype $20. ablfl /*$1.*/$2.
/* 14) KB 14May2014 */
  adt date9. PARAM atpt $50. AVAL best16. ATPTN BEST. ; /* 4)
KB 30Apr2014 */ /* 16) ZUH 2014-07-23 */

  * parameters ;
  paramcd = trim(egtestcd);
  param = trim(egtest);
  if egtestcd = 'HRMEAN' then paramn = 1;
  else if egtestcd = 'PRMEAN' then paramn = 2;
  else if egtestcd = 'QRS DUR' then paramn = 6;

```

```

        else if egtestcd = 'QTMEAN' then paramn = 3;
        else if egtestcd = 'QTCB' then paramn = 4;
        else if egtestcd = 'INTP' then paramn = 7;
        else if egtestcd = 'INTPCM' then paramn = 8;
        else if egtestcd = 'QTCF' then paramn = 5;
    ELSE IF EGTESTCD='EGALL' THEN PARAMN=9; /* 11) KB 30Apr2014 */
        else put 'USER WARN' 'ING: unidentified parameters: ' egtestcd =
egtest =;

/* 16) ZUH 2014-07-23 */
    IF PARAMCD='HRMEAN' THEN PARAM='Heart Rate';
    ELSE IF PARAMCD='PRMEAN' THEN PARAM='PR Duration';
    ELSE IF PARAMCD='QRS DUR' THEN PARAM='QRS Duration';
    ELSE IF PARAMCD='QTMEAN' THEN PARAM='QT Duration';
/* 16) ZUH 2014-07-23 */

    if egtestcd = 'QTCF' then do;
        paramtyp = 'DERIVED';
        dtype = 'FUNCTION';
    end;

    * analysis variables;
    aval = egstresn;
    avalc = trim(egstresc);
    if egstresc = "NORMAL" and egtestcd = "INTP" then do; avalc =
"Normal"; /*aval = 1;*/ end; /* 9) KB 30Apr2014 */
    else if index(egstresc,"ABNORMAL") and egtestcd = "INTP" then do;
        DESCLEN = INDEX(EGSTRESC,'-'); /* 2) SM 30Nov2013 */
        desc = propcase(/*scan*/ SUBSTR(egstresc,DESCLEN+1/*2,'-
'*/),'.'); /* 2) SM 30Nov2013 */
        desc = tranwrd(desc,' st ',' ST ');
        DESC = TRANWRD(DESC,' st-t ',' ST-T '); /* 2) SM 30Nov2013 */
        desc = tranwrd(desc,' stt ',' STT ');
        desc = tranwrd(desc,' v1 ',' V1 ');
        desc = tranwrd(desc,' pr ',' PR ');
        desc = tranwrd(desc,' pr ',' PR ');
        desc = tranwrd(desc,' Rbbb ',' RBBB ');
        DESC = TRANWRD(DESC,' Rbbb ',' RBBB '); /* 2) SM 30Nov2013 */
        desc = tranwrd(desc,' qtcB ',' QTcB ');
        desc = tranwrd(desc,' r ',' R ');
        desc = tranwrd(desc,' rbbb ',' RBBB ');
        desc = tranwrd(desc,' av ',' AV ');
        desc = tranwrd(desc,' pq=pr ',' PQ=PR ');
        desc = tranwrd(desc,' qtc ',' QTc ');
        desc = tranwrd(desc,' qt ',' QT ');
        desc = tranwrd(desc,' rs ',' RS ');
        DESC = TRANWRD(DESC,' rs-r ',' RS-R '); /* 2) SM 30Nov2013 */
        DESC = TRANWRD(DESC,' Ae ',' AE '); /* 2) SM 30Nov2013 */
        DESC = TRANWRD(DESC,' avr ',' AVR '); /* 2) SM 30Nov2013 */
        desc = tranwrd(desc,' ii i iii ',' II I III '); /* 2) SM
30Nov2013 */
/* 10) START KB 30Apr2014 */
        DESC=TRANWRD(DESC,'bradycardiast','bradycardiaST');

```

```

DESC=TRANWRD(DESC,'Nonspecific t abnormality','Nonspecific T
abnormality');
DESC=TRANWRD(DESC,'Supraventricularextrasystolesqtinterval
prolonged','SupraventricularextrasystolesQTinterval prolonged');
DESC = TRANWRD(DESC,'rsr ','rsR ');
DESC = TRANWRD(DESC,'Rsr ','rsR ');
DESC = TRANWRD(DESC,' st-T ',' ST-T ');
DESC = TRANWRD(DESC,' ncs ',' NCS ');
DESC = TRANWRD(DESC,' pvcs ',' PVCs ');
DESC = TRANWRD(DESC," pvc's "," PVC's ");
DESC = TRANWRD(DESC,' avf ',' aVF ');
DESC = TRANWRD(DESC,' pac(s) ',' PAC(s) ');
DESC = TRANWRD(DESC,' avf ',' aVF,');
/* 10) END KB 30Apr2014 */
/* 15) START KB 14May2014 */
DESC=TRANWRD(DESC,'RS-R','rs-R');
DESC=TRANWRD(DESC,'AVR','aVR');
/* 15) END KB 14May2014 */

if egclsig = 'NCS' then do;
/*      avalc = "Abnormal, CNR";      */
      AVALC = "Abnormal";      /* 18) KB 12Sep2014 */
/*      aval = 2; */ /* 9) KB 30Apr2014 */
end;
else if egclsig = 'CS' then do;
/*      avalc = "Abnormal, CR"; */
      AVALC = "Abnormal"; /* 18) KB 12Sep2014 */
/*      aval = 3; */ /* 9) KB 30Apr2014 */
end;
end;

avalu = trim(egstresu);

/* 16) ZUH 2014-07-23 */
IF PARAMCD NE 'HRMEAN' AND AVALU NE '' THEN DO;
PARAM=LEFT(TRIM(PARAM)) || ' (' || LEFT(TRIM(AVALU)) || ')';
END;
ELSE IF PARAMCD EQ 'HRMEAN' AND AVALU NE '' THEN DO;
PARAM=LEFT(TRIM(PARAM)) || '
(' || LEFT(TRIM(SUBSTR(AVALU,1,1)) || LOWCASE(SUBSTR(AVALU,2,8))) || ')';
END;
/* 16) ZUH 2014-07-23 */

```

```
ablfl = egblfl;
```

```

* visit data;
avisit = propcase(visit);
avisitn = visitnum;
atpt = propcase(egtpt);
atptn = egtptnum;

```

```

* dates;
adt = input(egdtc,yymmdd10.);
adtm=.; * set up missing for period calculations ;

keep usubjid egclsig egseq egcat egmethod egblfl egpos param: aval:
ablfl egstat egreasnd visit visitnum avisit:
egdtc egdy adt: paramtyp dtype desc atpt: EPOCH; /* 3) KB 30Apr2014
*/
run;

*****;
* Calculate changes from baseline (Screening) ;
*****;
proc sort data = eg2;
    by usubjid paramn avisitn;
run;

* baseline ;
data base;
    set eg2(where = (ablfl = 'Y')); * check SDTM.EG has EGBLFL correct
to SAP ;
    format base /*8.*/BEST. basec $200.; /* 4) KB 30Apr2014 */
    base = aval;
    basec = avalc;
    bvis = visitnum; * keep to make sure only calculate change after
baseline ;
    BEGCLSIG=EGCLSIG; /* 20) KB 12Sep2014 */

    keep usubjid paramn base basec bvis BEGCLSIG; /* 20) KB 12Sep2014
*/
run;

* change ;
data change(drop = bvis BEGCLSIG); /* 20) KB 12Sep2014 */
    merge eg2 base;
    by usubjid paramn;
    format chg /*8.*/BEST. shift1 $50.; /* 4) KB 30Apr2014 */
    if avisitn gt bvis then do;
        chg = aval - base;
/* 20) START KB 12Sep2014 */
/*
        if paramcd = 'INTP' then shift1 = trim(basec) || ' to '
||trim(avalc);*/
        IF PARAMCD = 'INTP' AND EGSTAT NE 'NOT DONE' THEN DO;
            IF NOT MISSING(BEGCLSIG) AND NOT MISSING(EGCLSIG) THEN SHIFT1
= TRIM(BASEC) || ', ' || STRIP(BEGCLSIG) || ' to ' ||TRIM(AVALC) || ', '
|| STRIP(EGCLSIG);
            ELSE IF NOT MISSING(BEGCLSIG) AND MISSING(EGCLSIG) THEN
SHIFT1=TRIM(BASEC) || ', ' || STRIP(BEGCLSIG) || ' to ' ||TRIM(AVALC);
            ELSE IF MISSING(BEGCLSIG) AND NOT MISSING(EGCLSIG) THEN
SHIFT1=TRIM(BASEC) || ' to ' ||TRIM(AVALC) || ', ' || STRIP(EGCLSIG);
            ELSE IF MISSING(BEGCLSIG) AND MISSING(EGCLSIG) THEN
SHIFT1=TRIM(BASEC) || ' to ' ||TRIM(AVALC);
        END;

```

```

/* 20) END KB 12Sep2014 */
end;
run;

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

data change2;
  set change;
  by usubjid paramn avisitn;
  * determine if any unscheduled;
  format anl01fl $2.;
  if avisit = 'UNSCHEDULED' or paramcd = 'EGALL' 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 = ;

/* 7) START KB 30Apr2014 */
  IF PARAMCD='QTCF' THEN DO;
    EGDTC='';
    EGDY=.;
    EGBLFL='';
  END;
/* 7) END KB 30Apr2014 */
run;

*****;
* Combine ADSL and EG data *;
*****;
* find periods;
*_mtotper;

data sleg(drop = trt01: tr01: adtm visitnum visit);
  merge adsl change2(in = a);
  by usubjid;
  if a;          * only include subject level data in vital signs ;
  format aperiod trtan trtpn aday 8. trta trtp $40. aperiodc $10.;
  aday = adt - trtsdt + 1;
  * allocate period and treatment according to full and partial dates
;
  *_mperall(dvar1 = adtm, dvar2 = adt);
  aperiodc = 'Period ' || put(aperiod,1.);
run;

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

options replace;

data adeg;

```

```
        set stdlib.adeq sleg;
        label aperiodc = 'Period (C)';
        format base best.;
run;

proc sort data = adeq out = adam.adeq(label = 'ECG Analysis Dataset');
    by USUBJID AVISITN ATPTN PARAMCD;
run;

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