

```

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 : 000000106326;
%put NOTE: Client Protocol ID : ZRHM-PK-05-JP;
%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 _MTOTPER _PERALL _SCRAMBLE;
%put NOTE: ;
%put NOTE: Programmed by : cvn_jhardman;
%put NOTE: Creation Date : 2014-01-02;
%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: 10Jan2014 JMH 1) Added wa-rning to
pick up unexpected values;
%put NOTE: 10Jan2014 JMH 2) Commented out code to
remove un-initialized message in log;
%put NOTE: 10Jan2014 JMH 3) Amended format of
ABLFL to match spec;
%put NOTE: 14Apr2014 KB 4) Added TRT: to keep for ADSL;
%put NOTE: 14Apr2014 KB 5) Amended formats to AVAL ATPTN BASE
CHG;
%put NOTE: 14Apr2014 KB 6) Amended EGTEST for QTcF;
%put NOTE: 14Apr2014 KB 7) Dropped SDTM variables in QTcF
derivation;
%put NOTE: 14Apr2014 KB 8) Set EGDTC EGBLFL and EGDY to blank
for QTcF;
%put NOTE: 14Apr2014 KB 9) Amended DESC;
%put NOTE: 14Apr2014 KB 10) Amended result for QTcF to not be
rounded;
%put NOTE: 14Apr2014 KB 11) Removed AVAL for INTP;
%put NOTE: 14Apr2014 KB 12) Added variables to scramble macro;
%put NOTE: 05Aug2014 KB 13) Added EXNOTRFL;
%put NOTE: 05Aug2014 KB 14) Amended length of EGTEST;
%put NOTE: 05Aug2014 KB 15) Added units on to parameters;
%put NOTE: 05Aug2014 KB 16) Amended format of PARAM and ABLFL;
%put NOTE: 05Aug2014 KB 17) Amended PARAM;
%put NOTE: 05Aug2014 KB 18) Amended format issue;

```

```

%put NOTE: 05Aug2014    KB           19) Amended format of AVAL and amended
for AVAL/AVALC issue;
%put NOTE: 21Sep2014    KB           20) Amended ABLFL;
%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:
        enrfl scrffl saffl randfl trtsdtm trtsdt trtsday trtedtm
trtedt trteday trt01: tr01: trt02: tr02:
        enfl exfl complfl fupfl dthfl anal: trtseq: TRT: EXNOTRFL;
/* 4) KB 14Apr2014 */ /* 13) KB 05Aug2014 */
run;

proc sort data = adsl;
    by usubjid;
run;

*****;
* pick up SUPPEG ;          /*TEMP FIX! No SUPPEG SDTM*/
*****;

/*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.;*/
/*    egseq = input(idvarval,best.);*/
/*run;*/
/**/
/*proc sort data = suppeg2;*/
/*    by usubjid egseq;*/
/*run;*/

*****;

```

```

* calculate QTcF ;
*****;
data qtcf(drop = hr qt);
    merge sdtm.eg(where = (egtestcd = 'HRMEAN') rename = (egstresn =
hr))
        sdtm.eg(where = (egtestcd = 'QTMEAN') rename = (egstresn =
qt));
    by usubjid visitnum;
    LENGTH EGTEST2 $40; /* 14) KB 05Aug2014 */
    egtestcd = 'QTcF';
/*    egtest = 'QTcF - Fridericia Correction Formula';*/
/*    egtest*/EGTEST2 = "QTcF - Fridericia's Correction Formula"; /* 6)
KB 14Apr2014 */ /* 14) KB 05Aug2014 */
    egstresn = /*round*/(qt/((60/hr)**(1/3)))/*,1.)*/*; /* 10) KB
14Apr2014 */
    egstresc = left(trim(put(egstresn,5.)));
    egorres = trim(egstresc);
    egseq=.;

    DROP EGCAT EGPOS EGMETHOD EPOCH EGTEST; /* 7) KB 14Apr2014 */ /* 14)
KB 05Aug2014 */
run;

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

data /*eg*/EGTEST; /* 14) KB 05Aug2014 */
    set sdtm.eg qtcf;
run;

/* 14) START KB 05Aug2014 */
DATA EG;
    SET EGTEST;
    IF EGTEST NE '' AND EGTEST2='' THEN EGTEST2=EGTEST;

    DROP EGTEST;
    RENAME EGTEST2=EGTEST;
RUN;
/* 14) END KB 05Aug2014 */

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

data eg2;
    SET/*merge*/ eg /*suppeg2(where = (not missing(usubjid)))*/; /*TEMP
FIX!!*/
    by usubjid egseq;
    format paramcd $8. param $50. avisit $40. paramn /*aval*/ avisitn
/*atptn*/ 8. avalc desc $200. avalu paramtyp dtype $20. /*ablfl

```

```

$1.*//*$2.*// /*$1.*// /*3) JMH 10Jan2014*/ /* 16) KB 05Aug2014 */ /* 16)
KB 05Aug2014 */ /* 20) KB 21Sep2014 */
      adt date9. atpt $50. AVAL BEST16. ATPTN BEST.; /* 5) KB
14Apr2014 */ /* 19) KB 05Aug2014 */

```

```

      * parameters ;
      paramcd = trim(egtestcd);
      IF PARAMCD IN ("HRMEAN" "PRMEAN" "QRSDUR" "QTMEAN") THEN PARAM =
STRIP(SCAN(EGTEST,2,' ')); /* 17) KB 05Aug2014 */
      ELSE PARAM=TRIM(EGTEST); /* 17) KB 05Aug2014 */
/*      param = trim(egtest);*/

```

```

      if egtestcd = 'HRMEAN' then paramn = 1;
      else if egtestcd = 'PRMEAN' then paramn = 2;
      else if egtestcd = 'QRSDUR' 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 put 'USER WARN' 'ING: unidentified parameters: ' egtestcd =
egtest =;

```

```

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

```

```

      * analysis variables;
/* 19) START KB 05Aug2014 */
      IF PARAMTYP='DERIVED' THEN DO;
          AVAL=ROUND(EGSTRESN,0.000000000000000001);
          AVALC=STRIP(PUT(AVAL,BEST16.));
      END;
      ELSE DO;
          aval = egstresn;
          avalc = trim(egstresc);
      END;
/* 19) END KB 05Aug2014 */
      if egstresc = "NORMAL" and egtestcd = "INTP" then do; avalc =
"Normal"; /*aval = 1;*/ end; /* 11) KB 14Apr2014 */
      else if index(egstresc,"ABNORMAL") and egtestcd = "INTP" then do;
          desclen = index(egstresc,'-');
          desc = propcase(substr(egstresc,desclen+1),'.');
          desc = tranwrd(desc,' st ',' ST ');
          DESC = tranwrd(desc,'St ','ST ');
          DESC = tranwrd(desc,'flat t ','flat T ');
          DESC = tranwrd(desc,'flat t ','flat T ');
          DESC = tranwrd(desc,'t wave ','T wave ');
          desc = tranwrd(desc,' st-t ',' ST-T ');
          desc = tranwrd(desc,' stt ',' STT ');
          desc = tranwrd(desc,' v1 ',' V1 ');
          desc = tranwrd(desc,' pr ',' PR ');
          desc = tranwrd(desc,' pr ',' PR ');

```

```

desc = tranwrđ(desc,'Rbbb ','RBBB ');
desc = tranwrđ(desc,'Rbbb, ','RBBB, ');
desc = tranwrđ(desc,' qtc b ',' QTc B ');
desc = tranwrđ(desc,' r ',' R ');
desc = tranwrđ(desc,'rbbb ','RBBB ');
desc = tranwrđ(desc,' av ',' AV ');
desc = tranwrđ(desc,' pq=pr',' PQ=PR');
desc = tranwrđ(desc,' qtc ',' QTc ');
DESC = tranwrđ(desc,'qtc ','QTc ');
desc = tranwrđ(desc,' qt ',' QT ');
desc = tranwrđ(desc,' rs ',' RS ');
desc = tranwrđ(desc,' rs-r ',' RS-R ');
desc = tranwrđ(desc,'Ae ','AE ');
desc = tranwrđ(desc,' avr ',' AVR ');
desc = tranwrđ(desc,' ii i iii ',' II I III ');
/* 9) START KB 14Apr2014 */
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,');
/* 9) END KB 14Apr2014 */
/*
if egclsig = 'NCS' then do; */ /*2) JMH 10Jan2014*/ /*TEMP
FIX!!*/
/*
avalc = "Abnormal, CNR"; */
/*
aval = 2; */
/*
end;*/
/*
else if egclsig = 'CS' then do; */
/*
avalc = "Abnormal, CR"; */
/*
aval = 3; */
/*
end;*/
/*
ELSE PUT "WA" "RNING: Check unexpected EGCSLIG value "
SUBJID= EGSLIG=; */ /*1) JMH 10Jan2014*/
end;

valu = trim(egstresu);
/*
ablfl = egblfl;*/ /* 20) KB 21Sep2014 */

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

* dates;

```

```

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

/* 15) START KB 05Aug2014 */
        IF AVALU='BEATS/MIN' THEN PARAM=TRIM(PARAM)||" (Beats/min)";
        ELSE IF NOT MISSING(AVALU) THEN PARAM=TRIM(PARAM)||"
(||COMPRESS(AVALU)||)";
/* 15) END KB 05Aug2014 */

        keep usubjid /*egclsig*/ egseq egcat egmethod egblfl egpos param:
aval: /*ablfl*/ egstat egreasnd visit visitnum avisit: /*2) JMH
10Jan2014*/ /*TEMP FIX!!*/ /* 20) KB 21Sep2014 */
        egdtc egdy adt: paramtyp dtype desc atpt: epoch;
run;

*****;
* Calculate changes from baseline (Screening) ;
*****;
/* 20) START KB 21Sep2014 */
DATA ADSL2;
    SET ADAM.ADSL;
    FORMAT TESTDTM DATETIME16.;

    IF DTESTDTM=PTESTDTM=. THEN DELETE;

    TESTDTM=MIN(DTESTDTM,PTESTDTM);

    KEEP USUBJID TESTDTM;
RUN;

PROC SORT DATA=EG2;
    BY USUBJID;
RUN;

DATA EG2A;
    MERGE EG2 ADSL2;
    BY USUBJID;
RUN;

DATA EG2B;
    SET EG2A;
        WHERE EGSTAT NE 'NOT DONE' AND
INDEX(UPCASE(AVISIT),'UNSCHEDULED')=0 AND INDEX(PARAMCD,'EGALL')=0;

    IF ADTM=. AND ADT NE . THEN DO;
        IF ADT LE DATEPART(TESTDTM) THEN TESTBASE='Y';
    END;
    ELSE IF ADTM NE . THEN DO;
        IF ADTM<TESTDTM THEN TESTBASE='Y';
    END;

    IF TESTDTM=. AND AVISIT IN ('Screening' 'Day -1') THEN TESTBASE='Y';
RUN;

```

```
PROC SORT DATA=EG2B(WHERE=(TESTBASE='Y')) OUT=EG2C;
  BY USUBJID PARAMCD AVISITN;
RUN;
```

```
DATA EG2D;
  SET EG2C;
  BY USUBJID PARAMCD AVISITN;
  FORMAT ABLFL $1.;

  IF LAST.PARAMCD AND LAST.AVISITN THEN ABLFL='Y';
RUN;
```

```
DATA EG2E;
  SET EG2D(WHERE=(ABLFL='Y'));

  KEEP USUBJID PARAMCD AVISITN ABLFL;
RUN;
```

```
PROC SORT DATA=EG2;
  BY USUBJID PARAMCD AVISITN;
RUN;
```

```
PROC SORT DATA=EG2E;
  BY USUBJID PARAMCD AVISITN;
RUN;
```

```
DATA EGBASES;
  MERGE EG2 EG2E;
  BY USUBJID PARAMCD AVISITN;
RUN;
/* 20) END KB 21Sep2014 */
```

```
proc sort data = /*eg2*/EGBASES; /* 20) KB 21Sep2014 */
  by usubjid paramn avisitn;
run;
```

```
* baseline ;
data base;
  set /*eg2*/EGBASES(where = (ablfl = 'Y')); * check SDTM.EG has
EGBLFL correct to SAP ; /* 20) KB 21Sep2014 */
  format base /*8.*/BEST. basec $200.; /* 5) KB 14Apr2014 */
  base = aval;
  basec = avalc;
  bvis = visitnum; * keep to make sure only calculate change after
baseline ;
```

```
  keep usubjid paramn base basec bvis;
run;
```

```
* change ;
data change(drop = bvis);
  merge /*eg2*/EGBASES base; /* 20) KB 21Sep2014 */
  by usubjid paramn;
```

```

        format chg /*8.*/BEST. shift1 $50.; /* 5) KB 14Apr2014 */
        if avisitn gt bvis then do;
            chg = aval - base;
            if paramcd = 'INTP' AND EGSTAT NE 'NOT DONE' then shift1 =
trim(basec) || ' to ' || trim(avalc);
            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 = ;

/*    if length(egdtc) gt 10 then astdtm = input(egdtc,e8601dt.);*/
    IF LENGTH(EGDTC) GT 10 THEN ASTDTM =
DHMS(INPUT(SCAN(EGDTC,1,'T'),YYMMDD10.),HOUR(INPUT(SCAN(EGDTC,2,'T'),TIME
5.)),MINUTE(INPUT(SCAN(EGDTC,2,'T'),TIME5.)),0); /* 18) KB 05Aug2014 */
    adt = input(egdtc,ymmdd10.);

    if not missing(astdtm) then astdt = datepart(astdtm);
    else if length(egdtc) = 10 then astdt = input(egdtc,ymmdd10.);

/* 8) START KB 14Apr2014 */
    IF PARAMCD='QTCF' THEN DO;
        EGDTC='';
        EGDY=.;
        EGBLFL='';
    END;
/* 8) END KB 14Apr2014 */
run;

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

data sleg(drop = trt01: tr01: tr02: trt02: adtm astdt astdtm astday
visitnum visit);
    merge adsl change2(in = a);
    by usubjid;
    if a;          * only include subject level data in vital signs ;
    format aperiod astday trtan trtpn aday 8. trta trtp $40. aperiodc
$8.;

```



```

    aday = adt - trtsdt + 1;
    astday = astdt - trtsdt + 1;
    * allocate period and treatment according to full and partial dates
;
    if astday in (0 1) then aperiod=1;
    else if astday in (2 3) then aperiod=2;
    %_mperall(dvar1 = adtm, dvar2 = adt);

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

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

options replace;

data adeg;
    set stdlib.adeg sleg;
    label aperiodc = 'Period (C)';
run;

proc sort data = adeg out = adam.adeg(label = 'ECG Analysis Dataset');
    by usubjid avisitn paramcd;
run;

options noreplace;

%_scramble(set=adeq, id=USUBJID SUBJID SUBJIDN age sex sexc sexn race
dthfl height weightbl bmi ucpdgr1 ucpdgrln nicogr1
        nicogrln targr1 targrln analgr1 analgrln, dates=trtsdtm
trtsdt trtsday trtedtm trtedt trteday,
        nullc=trtp trta trtseqp trtseqa TRTSTMF, /* 12) KB
14Apr2014 */
        nulln=trtpn trtan trtseqpn trtseqan, nullcc=/*4*/5,
nullnc=4); /* 12) KB 14Apr2014 */

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

```