

```

%_mprintto;
options notes nosource;
proc datasets lib=work nolist memtype=data kill; quit;
%put NOTE:
=====;
%put NOTE: Covance Study Number : 000000106326;
%put NOTE: Client Protocol ID   : ZRHM-PK-05-EU;
%put NOTE: Program Name        : t_ecg.sas;
%put NOTE: Purpose              : table of ecg measurements;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADEG ADAM.ADSL;
%put NOTE: Output               : t_15_2_6_14(eg);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_jriley;
%put NOTE: Creation Date        : 2014-08-07;
%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: 12Aug2014  JR         1) Amended wording of heart rate
dummy;
%put NOTE: 12Aug2014  JR         2) Amended splits in headers;
%put NOTE: 21Sep2014  KB         3) Amended relevant to significant;
%put NOTE: 21Sep2014  KB         4) Amended baseline footnote;
%put NOTE: 23Sep2014  JMH        5) Amended in line with client
comments;
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

*=====;
* START OF PROGRAM CODE                                     ;
*=====;

/* Standard - just change the number to match the listing you're working
on. Also change the letters in the*/
/* bracket, eg ccb = current cigarette brands. Make sure to do this at
the top of the code too. */

      %let tflno=T_15_02_06_14(eg);

/* Standard - leave this */

```

```

%let TFL_Part=%scan(&_SASPROGRAMFILE,-3,%str(/));

/* Standard - leave this */
data _null_;
    tmp="&TFL_Part";
    if tmp not in ("dev" "qc") then call symput("TFL_Part", "prod");
    call symput('TFLpath', compress("&_SASPROGRAMFILE",""));
run;

*****;
* read in data ;
*****;

/*Use ADSL to get N numbers for column headers*/
data adsl;
    set adam.adsl;
    where saffl = 'Y';
    if missing(trtseqa) then delete;
    if index(trtseqa,'Exposed') then delete;
    output;
    trtseqa=99;
    trtseqa='Overall Safety';
    output;
run;

proc freq data=adsl noprint;
    table trtseqa*trtseqa/ out =tot(drop=percent);
run;

data dumtrts; /*Use this to output any columns for which N=0*/
    attrib trtseqa length =$200.
            trtseqa length=8.;
    trtseqa=1;
    trtseqa='THS 2.2 Menthol - mCC';
    output;
    trtseqa=2;
    trtseqa='mCC - THS 2.2 Menthol';
    output;
    trtseqa=3;
    trtseqa='THS 2.2 Menthol - NRT gum';
    output;
    trtseqa=4;
    trtseqa='NRT gum - THS 2.2 Menthol';
    output;
    trtseqa=5;
    trtseqa='Enrolled not randomized';
    output;
run;

data tot2;
    merge tot(in=a) dumtrts(in=b);
    by trtseqa trtseqa;

```

```

        if a or b;
        if b and not a then count=0;
        call symput('trt' || compress(put(trtseqan,best.)),
compress(count));
run;

/*Bring in appropriate data from ADEG*/
data adeg;
    set adam.adeg;
    where saffl = 'Y' and anl01fl='Y';
    if missing(trtseqan) then delete;
    if index(trtseqa,'Exposed') then delete;
    output;
    trtseqan=99;
    trtseqa='Overall Safety';
    output;
run;

/* 4) START KB 21Sep2014 */
DATA ADEG2;
    SET ADEG;

    IF ABLFL='Y' THEN DO;
        AVISIT='Baseline';
        AVISITN=100;
    END;
    IF AVISITN LE 100 AND AVISIT NE 'Baseline' THEN DELETE;
RUN;

/*INTP only as this code will bring out the class variables. Other params
will eb dealt with later*/

proc freq data=/*adeg*/ADEG2(where=(paramcd='INTP')) noprint; /* 4) KB
21Sep2014 */
    table
    trtseqan*trtseqa*avisitn*avisit*aval*avalc*PARAMN*PARAM*EGCLSIG / out
=intp1(drop=percent); /* 3) KB 21Sep2014 */
run;

data intp2;
    merge intp1(in=a) dumtrts(in=b) tot2(rename=(count=total));
    by trtseqan trtseqa;
    if a or b;
    if b and not a then do;
        count = 0;
        avisitn=/*1*/100; /* 4) KB 21Sep2014 */
        avisit=/*'Screening'*/'Baseline'; /* 4) KB 21Sep2014 */
        avalc='Normal';
    end;
    if total ne 0 then percent=count/total*100;
    else percent=0;
run;

```

```

proc sort data=intp2 nodupkey out=trtvis(keep=trtseqan trtsega avisitn
avisit avalc paramn param EGCLSIG); /* 3) KB 21Sep2014 */
    by trtseqan trtsega avisitn avisit;
run;

data dumrows;
set trtvis;
    avalc='Normal';
    output;
/*    avalc='Abnormal, CNR';*/
    AVALC='Abnormal'; /* 3) KB 21Sep2014 */
    EGCLSIG='NCS'; /* 3) KB 21Sep2014 */
    output;
/*    avalc='Abnormal, CR';*/
    AVALC='Abnormal'; /* 3) KB 21Sep2014 */
    EGCLSIG='CS'; /* 3) KB 21Sep2014 */
    output;
run;

proc sort data=dumrows;
    by trtseqan trtsega avisitn avisit avalc EGCLSIG; /* 3) KB
21Sep2014 */
run;

proc sort data=intp2;
    by trtseqan trtsega avisitn avisit avalc EGCLSIG; /* 3) KB
21Sep2014 */
run;

data intp3;
    merge intp2(in=a) dumrows(in=b);
    by trtseqan trtsega avisitn avisit avalc EGCLSIG; /* 3) KB
21Sep2014 */
    if a or b;
    attrib statval statval2 length=$100.
                                count1 length=$3.;
    if b and not a then do;
        count=0;
        percent=0;
    end;
        attrib paramc length = $100.
                                visit length = $100.
                                stat length = $100.;
    paramc=strip(param);
    if trtseqan=5 then do;
        paramc=strip('Interpretation');
        paramn=7;
    end;

        if avisitn=/*1*/100 then do; ord=1;
visit=/*'Screening'*/'Baseline'; end; /* 4) KB 21Sep2014 */
        else if avisitn=101 then do; ord=2; visit='A Single Use Day
1'; end;

```

```

        else if avisitn=103 then do; ord=3; visit='Single Use Day 3';
end;

        if avalc='Normal' then do;
            stat='Normal - n (%)';
            statord=1;
        end;
/*
        else if avalc='Abnormal, CNR' then do;*/
        ELSE IF AVALC='Abnormal' AND EGCLSIG='NCS' THEN DO; /* 3) KB
21Sep2014 */
/*
            stat='Abnormal non-clinically relevant - n (%)';*/
            STAT='Abnormal non-clinically significant - n (%)'; /*
3) KB 21Sep2014 */
            statord=2;
        end;
/*
        else if avalc='Abnormal, CR' then do;*/
/*
            stat='Abnormal clinically relevant - n (%)';*/
        ELSE IF AVALC='Abnormal' AND EGCLSIG='CS' THEN DO; /* 3) KB
21Sep2014 */
            stat='Abnormal clinically significant - n (%)'; /* 3) KB
21Sep2014 */
            statord=3;
        end;

        if count=0 then statval = strip(put(count,best.)) ;

        if count lt 10 then count1=' ' || compress(put(count,best.));
        else count1=strip(put(count,best.));

        count1=trim(count1);

        statval=count1;

        if percent=100 then statval2 = '(100 %)' ;
        else if percent=0 then statval2='';
        else if percent lt 10 then statval2 = '( '
||left(strip(put(round(percent,0.1),5.1))) || '%)';
        else if percent ge 10 then statval2 = '( '
||left(strip(put(round(percent,0.1),5.1))) || '%)';
        run;

        proc sort data=intp3;
            by paramn paramc ord visit statord stat;
        run;

        proc transpose data=intp3 out=intp4t prefix=t;
            by paramn paramc ord visit statord stat;
            var statval;
            id trtseqan;
            idlabel trtseqa;
        run;

        proc transpose data=intp3 out=intp4n prefix=n;

```

```

        by paramn paramc ord visit statord stat;
        var statval2;
        id trtseqan;
        idlabel trtsega;
run;

data intp4;
    merge intp4t intp4n;
    by paramn paramc ord visit statord stat;
run;
/*End of INTP, will set on with rest of data later*/

/*Now this will create the stats for all params except INTP*/

data adeg_orig; /*This is for the actual values so aval will be used as
the analysis variable*/
    set /*adeg*/ADEG2(where=(paramcd ne 'INTP')); /* 4) KB 21Sep2014 */
    if avisitn=/*1*/100 then ord=1; /*Screening*/ /* 4) KB 21Sep2014
*/
    else if avisitn=101 then ord=2; /*Day 1*/
    else if avisitn=103 then ord=4; /*Day 3*/
    statval=aval;
run;

data adeg_chg; /*This is for the changes from baseline so chg will be
used as the analysis variable*/
    set /*adeg*/ADEG2(where=(avisitn in(101 103) and paramcd ne
'INTP')); /*Only keep days after baseline*/ /* 4) KB 21Sep2014 */
    if avisitn=101 then ord=3; /*Change from Baseline to Day 1*/
    else if avisitn=103 then ord=5; /*Change from Baseline to Day 3*/
    statval=chg;
run;

data adeg_all;
    set adeg_orig adeg_chg;
run;

proc sort data=adeg_all;
    by trtseqan trtsega;
run;

data all;
    merge adeg_all(in=a) dumtrts(in=b);
    by trtseqan trtsega;
    if a or b;
    if b and not a then statval='';
run;

proc sort data=all;
    by trtseqan trtsega paramn ord param avalu avisit;
run;

proc univariate data=all noprint;

```

```

var statval;
  by trtsega trtsega paramn ord param avalu avisit;
  output out=results01 n=n1 mean=mean1 std=std1 median=med1 min=min1
max=max1;
run;

data results02;
  set results01;
  attrib meansd length=$100.
                                minmax length=$100.
                                n          length=$100.
                                median   length=$100.;

                                n          = left(compress(put(n1,8.)));
                                if not missing(med1) then median =
left(compress(put(med1,8.1)));
                                if not missing(mean1) and not missing(std1) then meansd
= left(compress(put(mean1,8.1))) || ' (' ||
compress(put(0.01*ceil(std1/0.01),8.2)) || ')';
                                if not missing(min1) and not missing(max1) then minmax
= left(compress(put(min1,8.))) || ', ' || left(compress(put(max1,8.)));

                                avalu=lowercase(avalu);
                                pos=notalnum(param,1);
                                if index(param,'Bazett') or index(param,'Fridericia')
then pos+3;

                                param1=substr(param,1,pos)||lowercase(substr(param,pos+1));
                                drop param;
                                rename param1=param;
                                drop n1 mean1 std1 med1 min1 max1;
run;

data results03; /*Create text as required in output*/
  set results02;
  attrib paramc length = $100.
                                visit   length = $100.;

                                if ord=1 then visit=avisit;
                                else if ord=2 then visit=avisit;
                                else if ord=3 then visit='Change from Screening on Day
1';

                                else if ord=4 then visit=avisit;
                                else if ord=5 then visit='Change from Screening on Day
3';

                                paramc=strip(param);

                                /*This bit of code jst populates the variables of dummy
columns to avoid problems with the transpose*/
                                if missing(ord) and missing(paramn) then do;
                                    ord=1;
                                    visit=/'Screening'/'Baseline'; /* 4) KB
21Sep2014 */

```

```

        paramn=1;
        paramc=/'Summary (mean) heart rate
(beats/min)'/Heart rate (beats/min)'; /* 1) JR 12Aug2014 */
        end;
    run;

proc sort data=results03;
by paramn paramc ord visit;
run;

proc transpose data=results03 out=results04 prefix=t name=varname;
    by paramn paramc ord visit;
    var n meansd median minmax;
    id trtsean;
    idlabel trtsega;
run;

data results05;
    set results04;
    attrib stat length = $100.;
    if varname='N' then do; statord=1; stat='n'; end;
    else if varname='MEANSD' then do; statord=2; stat='Mean
(SD)'; end;
    else if varname='MEDIAN' then do; statord=3; stat='Median';
end;
    else if varname='MINMAX' then do; statord=4; stat='Min, Max';
end;

    drop varname;
run;

data results06;
    set results05;
    if stat='n' then do;
        if missing(t1) then t1='0';
        if missing(t2) then t2='0';
        if missing(t3) then t3='0';
        if missing(t4) then t4='0';
        if missing(t5) then t5='0';
        if missing(t99) then t99='0';
    end;
run;

/*Now combine the stats with the classification results*/
data allresults;
    set results06 intp4(in=a);
    if a then do;
        if missing(t1) then t1='0';
        if missing(t2) then t2='0';
        if missing(t3) then t3='0';
        if missing(t4) then t4='0';
        if missing(t5) then t5='0';
        if missing(t99) then t99='0';
    end;

```



```

        flag=1;
run;

data labels;
    set allresults;
    attrib t1 label = "THS 2.2 Menthol$ - mCC $(N=&trt1)" /* 2)
JR 12Aug2014 */
                                t2 label = "mCC -$THS 2.2
Menthol$(N=&trt2) "
                                t3 label = "THS 2.2 Menthol -$NRT gum
$(N=&trt3) "
                                t4 label = "NRT gum$- THS 2.2
Menthol$(N=&trt4) " /* 2) JR 12Aug2014 */
                                t5 label = "Enrolled
Not$Randomized$(N=&trt5) "
                                t99 label = "Overall$Safety$(N=&trt99) "
                                n1 label = "THS 2.2 Menthol - mCC %"
                                n2 label = "mCC - THS 2.2 Menthol$ %"
                                n3 label = "THS 2.2 Menthol - NRT gum %"
                                n4 label = "NRT gum - THS 2.2 Menthol %"
                                n5 label = "Enrolled Not Randomized %"
                                n99 label = "Overall Safety %";

    if index(visit,'/') then visit=tranwrd(visit,'/', '/' );
    if index(visit,'Screening') then
visit=tranwrd(visit,'Screening','Baseline');
/*    if index(visit,'from Baseline') then
visit=tranwrd(visit,'Baseline','baseline'); */ /* 5) JMH 23Sep2014 */

run;

proc sql noprint;
    create table table.T_15_02_06_14 as
    select paramc, visit, stat, t1, n1, t2, n2, t3, n3, t4, n4, t5, n5,
t99, n99
    from labels
    order by paramn, ord, statord;
quit;

proc sort data=labels;
    by paramn ord statord;
run;

data paging;
    set labels;
    by paramn ord statord;
    if /*first.paramn or*/ (first.ord and ln ge 8) or (paramn=7
and FIRST.ORD AND ln gt /*2*/4) then ln=1; /*Amend to look presentable,
and avoid page overflows*/ /* 5) JMH 23Sep2014 */ /* 6) JR 24Sep2014 */
    else ln+1;
    if ln=1 then page+1;

```

```

        call symput("page",compress(put(page,best.)));
run;

/* Standard - leave this */
options number nodate orientation=landscape papersize=&p_pgsz missing='
';
ods escapechar='$';
%let linetop = \brdrt\brdrs\brdrw30; * needs to be 1.5pt so calculated
in twips (1/20 pt) ;
%let linebot = \brdrb\brdrs\brdrw30;
/* Standard - macro for paging */
%macro outrtf(blankn=, halfblnk=);

%if &halfblnk=N %then %let halfblnk=;
%else %if &halfblnk=Y %then %let halfblnk=\~;

ods path stdlib.tl06326 (read) ;
ods results off;
ods rtf toc_data
file="/cvn/projects/prj/data/000000106326/TFL/&TFL_Part./&tflno..rtf"
style=tl06326 startpage=yes headery=1440 footery=1440 ;
ods noproctitle;
%do i=1 %to &page;

title ;
footnote;
%let wd=0;
ods proclabel = ' ';

data comp;
    set paging end=eof;
    where page=&i;

    call symput('paramn',paramn);

    /* Amend title as needed */
    _firtitl="Table 15.2.6.14 Summary of ECG Measurements -
Safety Population";
    _upcas=(length("Path: &TFLpath.")-
length(compress("Path:&TFLpath.",'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
    len=&blankn.-length("(page &i of &page)");
    if eof then do;
        call symput('_FSRTITL', trim(left(_firtitl)));
        call symput('_blankn', compress(put(len,best.)));
    end;
    drop _firtitl _upcas len;
run;

ods listing close;

```

```

* most set up in template others below;
* title arial 12pt bold with 12pt paragraph space below;
* all headers to be arial 11pt bold;
* data arial 10pt;
* headers to be central, text values left aligned and numeric centered
around decimal point;
/* Update with your variables as needed */
proc report data = comp headline headskip missing nowd split = '$' %if
&i=1 %then %do; contents=' ' %end; %else %do; contents='' %end;;
    column flag page %if &paramn=7 %then %do; paramn
('Parameter$(units)' paramc) ord ('Study Day' visit) statord ('Statistic'
stat) ("Sequence &linebot" ("THS 2.2$Menthol$- mCC$(N=&trt1)" t1 n1)
("mCC -$THS 2.2 Menthol$(N=&trt2)" t2 n2) ("THS 2.2 Menthol-$NRT gum
$(N=&trt3)" t3 n3)

("NRT gum$- THS 2.2$Menthol$(N=&trt4)" t4 n4) ("Enrolled
Not$Randomized$(N=&trt5)" t5 n5)) ( "Overall$Safety$(N=&trt99)" t99 n99)
%end;

    %else %do; paramn paramc ord visit statord stat ("Sequence
&linebot" t1 t2 t3 t4 t5) t99 %end;;

        define flag          / order order = internal noprint;
        define page          / order order = internal noprint;
        define paramn        / order order = internal noprint;
        define ord           / order order = internal noprint;
        define statord       / order order = internal noprint;
        %if &paramn ne 7 %then %do;
            define paramc      / group style={just=left
cellwidth=1.5cm}'Parameter$(units)';
            define visit       / group style={just=left
cellwidth=2cm}'Study Day';
            define stat        / display style={just=left
cellwidth=1.6cm}'Statistic';
            define t1          / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define t2          / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define t3          / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define t4          / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define t5          / display style={just=center
cellwidth=2.0cm} style(header)={just=center};
            define t99         / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        %end;
        %else %if &paramn=7 %then %do;
            define paramc      / group style={just=left
cellwidth=1.7cm}'';

```

```

        define visit          / group style={just=left
cellwidth=2cm}'';
        define stat           / display style={just=left
cellwidth=/*2*/2.5cm}''; /* 5) JMH 23Sep2014 */
        define t1             / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define t2             / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define t3             / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define t4             / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define t5             / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define t99            / display style={just=c cellwidth=0.6cm}
style(header)={just=center}'';
        define n1             / display style={just=c cellwidth=1.2cm}
style(header)={just=center}'';
        define n2             / display style={just=c cellwidth=1.2cm}
style(header)={just=center}'';
        define n3             / display style={just=c cellwidth=1.2cm}
style(header)={just=center}'';
        define n4             / display style={just=c cellwidth=1.2cm}
style(header)={just=center}'';
        define n5             / display style={just=c cellwidth=1.2cm}
style(header)={just=center}'';
        define n99            / display style={just=c cellwidth=1.3cm}
style(header)={just=center}'';
        %end;

break before flag / page %if &i=1 %then %do;
contents="%&_fsrtitl" %end; %else %do; contents='' %end;;

break after page / page;

compute after ord;
    line " ";
endcomp;

compute before page / style={protectspecialchars=off};
    line "&linetop";
endcomp;

compute before _page_ / style={just=left protectspecialchars=off};
    line "\b\fs24\sa24&_FSRTITL." ; * \b = bold, \fs24 is font
size 12pt, \sa24 is space after 12pt;
    line "&linebot";
endcomp;

compute after _page_ / style={just=left protectspecialchars=off
PRETEXT="&LINETOP."};
    line 'Note: mCC = menthol conventional cigarettes; NRT gum =
Nicotine Replacement Therapy gum; THS = Tobacco Heating System.';

```

```

        line 'Note: Enrolled Not Randomized refers to all subjects
enrolled but not randomized. Overall Safety refers to enrolled subjects
exposed to THS 2.2 Menthol or NRT gum.';
        line 'Note: Percentages are based on the number of subjects
indicated in the column header (N).';
/*        line 'Note: Baseline is Screening.'; */
        LINE 'Note: Baseline is the last available time point prior
to the product test (THS 2.2 Menthol or NRT gum) at Admission (Day -1).
'; /* 4) KB 21Sep2014 */
        line ' ';
        line 'Appendix 15.3.6.9';
        line "Path: &TFLpath." &_blankn.*"\~\~" "(Page &i of &page)";
        line "Program Run: &sysdate  &sysuserid  Program Status:
&status";
        endcomp;
run;
%end;
ods rtf close;
ods results on;
ods path sashelp.tmplmst (read);

%mend ;

%outrtf(blankn=70, halfblnk=N);
ods listing;
proc printto print = "&table./T_15_02_06_14.lst" new;
run;

proc contents data = table.T_15_02_06_14 varnum;
run;
ods listing close;
proc printto ; run;
*=====;
* END OF PROGRAM CODE ;
*=====;

```