

```

%_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-JP;
%put NOTE: Program Name        : t_cohb3.sas;
%put NOTE: Purpose              : table of blood cohb by nicotine level;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADSL ADAM.ADBX;
%put NOTE: Output               : t_15_2_4_8_1_2(cohb);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_aobyrne;
%put NOTE: Creation Date        : 2014-08-08;
%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:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

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

%let tflno=T_15_02_04_08_01_02(cohb);

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

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 values for column headers*/

```

```

data adsl;
    set adam.adsl(where=(pprotfl='Y'));
    if analgrln=1 then do;
        if index(trt01a,'THS 2.2 ') or index(trt02a,'THS 2.2 ') then
trtord=4;
        output;
        if index(trt01a,'mCC') or index(trt02a,'mCC') then trtord=5;
        output;
    end;
    else if analgrln=2 then do;
        if index(trt01a,'THS 2.2 ') or index(trt02a,'THS 2.2 ') then
trtord=10;
        output;
        if index(trt01a,'NRT') or index(trt02a,'NRT') then trtord=6;
        output;
    end;
    else if missing(analgrln) then delete;
run;

proc sort data=adsl nodupkey out=adsl1;
    by analgrln analgr1 trtord nicogrln nicogr1 subjid;
run;

proc freq data=adsl1(where=(not missing(trtord))) noprint;
    table analgrln*analgr1*trtord*nicogrln*nicogr1/ out
=tot(drop=percent rename=(count=total));
run;

data tot2;
    set tot;
    call
symput('gender' || strip(put(trtord,best.)) || left(compress(put(nicogrln,bes
t.))), strip(put(total,best.)));
run;

proc freq data=adsl1(where=(not missing(trtord))) noprint;
    table analgrln*analgr1*trtord/ out =tottrt(drop=percent
rename=(count=total));
run;

data tottrt2;
    set tottrt;
    call symput('trt' || strip(put(trtord,best.)), strip(put(total,best.)));
run;

/*Bring in appropriate data from ADBX*/
data adbx;
    set adam.adbx(where=(anl02fl='Y' and pprotfl ='Y' and
paramcd='CARBXHGB'));
run;

data adbx_orig;
    set adbx;

```

```

        statval=aval;
        if nicogrln=. then delete;
run;

proc sort data=adbx_orig nodupkey out=nicotine;
    by usubjid analgrln trtan nicogrln nicogr1;
run;

proc freq data=nicotine noprint;
    tables analgrln*trtan*nicogrln*nicogr1 / out=totals;
run;

data _null_;
    set totals;

    if analgrln=2 and trtan=4 then trtan=10;
run;

proc sort data=adbx_orig;
    by analgrln analgr1 trtan trta atptn atpt nicogrln nicogr1;
run;

proc means data=adbx_orig noprint;
    var statval;
    by analgrln analgr1 trtan trta atptn atpt nicogrln nicogr1;
    output out=results01 n=n1 mean=mean1 std=std1 median=median1 min=min1
max=max1 q1=q1 q3=q3;
run;

data results02;
    set results01;
    attrib meansd minmax n median quart length=$20.;

    n = left(compress(put(n1,8.)));
    if not missing(median1) then median =
left(compress(put(median1,8.2)));
    if not missing(mean1) and not missing(std1) then meansd =
left(compress(put(round(mean1,0.01),8.2))) || ' (' ||
left(compress(strip(put(0.001*ceil(std1/0.001),8.3)))) || ')';
    if not missing(min1) and not missing(max1) then minmax =
left(compress(put(min1,8.1))) || ', ' || left(compress(put(max1,8.1)));
    if not missing(q1) and not missing(q3) then quart =
strip(strip(put(0.01*floor(q1/0.01),8.2)) || ', ' ||
strip(put(0.01*ceil(q3/0.01),8.2)));

    drop n1 mean1 std1 median1 min1 max1 q1 q3;
run;

%macro outrtf(blankn=, halfblnk=);

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

/*Obtain subjects with values BLOQ*/

```

```

data adbx_blq;
    set adbx;
    where bloqfl='Y';

    statval=aval;
run;

proc freq data=adbx_blq noprint;
    table analgrln*analgrl*trtan*trta*atptn*atpt*nicogrln*nicogr1/ out
    =blq(drop=percent);
run;

%let dsid=%sysfunc(open(blq));
%let nsum=%sysfunc(attrn(&dsid.,nobs));
%let rc=%sysfunc(close(&dsid.));

%put "Check " &nsum.;

%if &nsum. lt 1 %then %do;
    proc sort data=adbx nodupkey out=tpts(keep=analgrln analgr1
    trtan trta atptn atpt nicogrln nicogr1);
        by analgrln analgr1 trtan trta atptn atpt nicogrln
    nicogr1;
    run;

    data blq1;
        set tpts;
        attrib blq length=$50.;
        blq='0';
    run;

%end;

%else %do;
    data blq1;
        attrib blq length=$50.;
        merge blq(in=a) tot;
        by analgrln analgr1 trtord;
        if not a then do;
            count=0;
        end;
        percent=count/total*100;

        if count=0 then blq='0';
        else if percent=100 then blq= put(count,3.)||' (100%)' ;
        else blq=put(count,3.)||' ('||put(percent,3.)||'%)' ;
    run;

%end;

/*Obtain the geometric mean*/

data gmean;
    set adbx_orig;
    statvall=statval;
    ln_statvall=log(statvall);

```

```

run;

proc means data=gmean noprint;
  output out=gmean1 mean=mean std=std1 lclm=lci1 uclm=uci1;
  var ln_statvall;
  by analgrln analgrl trtan trta atptn atpt nicogrln nicogr1;
run;

data gmean2;
  set gmean1;
  gmean1=exp(mean);
  gmean=left(compress(put(gmean1,8.2)));
  GCV=COMPRESS(strip(put(0.01*ceil(sqrt(EXP(STD1*STD1)-
1)*100/0.01),8.2)));

  glci=exp(lci1);
  guci=exp(uci1);

  keep analgrln analgrl trtan trta atptn atpt gmean gcv glci guci std1
nicogrln nicogr1;
run;

/*Combine the gmean and BLQ with other stats*/

data results03; /*Create text as required in output*/
  merge results02 gmean2 blq1;
  by analgrln analgrl trtan trta atptn atpt nicogrln nicogr1;
  attrib gmeancv length=$20.;
  if analgrln=2 and trtan=4 then trtan=10;
  if not missing(gcv) then gmeancv=left(trim(gmean)) || ' (' ||
left(trim(gcv))||'%)';
  else gmeancv=left(trim(gmean));
  if not missing(glci) and not missing(guci) then ci =
strip(strip(put(0.01*floor(glci/0.01),8.2))) || ', ' ||
strip(put(0.01*ceil(guci/0.01),8.2)));
  if trtan in(4 5) then trtan=trtan-3;
  trtans=strip(strip(put(trtan,best.)) ||
strip(put(nicogrln,best.)));
run;

proc sort data=results03;
  by analgrln atptn atpt;
run;

proc transpose data=results03 out=results04 prefix=_ name=varname;
  by analgrln atptn atpt;
  var n meansd median minmax ci quart gmeancv blq;
  id trtans;
  idlabel trta;
run;

data results05;
  set results04;
  attrib stat length = $100.;

```

```

if varname='N' then do;
    statord=1;
    stat='n';
end;
    else if varname='BLQ' then do;
        statord=2;
        stat='BLOQ - n (%)';
        end;
        else if varname='GMEANCV' then do;
            statord=3;
            stat='Geometric Mean (CV%)';
            end;
else if varname='CI' then do;
    statord=4;
    stat='95% CI';
end;
else if varname='MEDIAN' then do;
    statord=5;
    stat='Median';
end;
else if varname='QUART' then do;
    statord=6;
    stat='Q25, Q75';
end;
else if varname='MINMAX' then do;
    statord=7;
    stat='Min, Max';
end;
else if varname='MEANSD' then do;
    statord=8;
    stat='Mean (SD)';
end;
drop varname;
run;

data results06;
    set results05;

    if stat='n' then do;
        if missing(_11) then _11='0';
        if missing(_12) then _12='0';
        if missing(_21) then _21='0';
        if missing(_22) then _22='0';
        if missing(_101) then _101='0';
        if missing(_102) then _102='0';
        if missing(_71) then _71='0';
        if missing(_72) then _72='0';
    end;
run;

proc sort data=results06;
    by analgr1n atptn statord;
run;

```

```

data labels;
set results06;
  attrib _11 label = "<= 0.6 mg$(N=&gender41)"
         _12 label = "> 0.6 - 1.0 mg$(N=&gender42)"
         _21 label = "<= 0.6 mg$(N=&gender51)"
         _22 label = "> 0.6 - 1.0 mg$(N=&gender52)"
         _101 label = "<= 0.6 mg$(N=&gender101)"
         _102 label = "> 0.6 - 1.0 mg$(N=&gender102)"
         _71 label = "<= 0.6 mg$(N=&gender61)"
         _72 label = "> 0.6 - 1.0 mg$(N=&gender62)"
         atpt label= "Formatted timepoint"
         atpt1 label= "Unformatted timepoint";

         atpt1=atpt;

  if index(atpt,'T0') then atpt=tranwrd(atpt,'T0',"T${sub 0}");

  if index(stat,'BLOQ') and _11=0 and _12=0 and _21=0 and _22=0 then
delete;
  if index(stat, 'BLOQ') and _101=0 and _102=0 and _71=0 and _72=0 then
delete;

run;

proc sql noprint;
  create table table.t_15_02_04_08_01_02 as
  select analgrln, atpt, atpt1, stat, _11, _12, _21, _22, _101, _102,
_71, _72
  from labels
  order by analgrln, atptn, statord;
quit;

data paging;
  set labels;
  by analgrln atptn statord;
  if first.analgrln or first.atptn or ln gt 11 then ln=1; /*Amend to
look presentable, and avoid page overflows*/
  else ln+1;
  if ln=1 then page+1;
  call symput("page",compress(put(page,best.)));
  flag=1;
run;

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;

```

```

ods path stdlib.t106326 (read) ;
ods results off;
ods rtf toc_data
file="/cvn/projects/prj/data/000000106326/TFL/&TFL_Part./&tflno..rtf"
style=t106326 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;

        /* Amend title as needed */
        _firtitl="Table 15.2.4.8.1.2 Descriptive Statistics of Blood COHb
(%) by Nicotine Level - PK 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.)));
            call symput('j',analgrln);
        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;
proc report data = comp missing headline headskip missing nowd split
= '$' %if &i=1 and &j=1 %then %do; contents=' ' %end; %else %do;
contents='' %end;;
    column flag page atptn atpt statord stat
        %if &j=1 %then %do; ("Group-1 PK &linebot" ("THS 2.2 Menthol $
(N=&trt4) &linebot" _11 _12) ("mCC $(N=&trt5) &linebot" _21 _22)) %end;
        %if &j=2 %then %do; ("Group-2 PK &linebot" ("THS 2.2 Menthol $
(N=&trt10) &linebot" _101 _102) ("NRT gum$(N=&trt6) &linebot" _71 _72))
%end;;

    define flag          / order order = internal noprint;
    define page          / order order = internal noprint;
    define atptn         / order order=internal noprint;

```



```

        define atpt          / group style={just=left cellwidth=1.5cm}
style(header)={just=center} "Timepoint";
        define statord       / order order = internal noprint;
        define stat         / display style={just=left cellwidth=1.8cm}
style(header)={just=center} "Statistic";
        %if &j=1 %then %do;
            define _11        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _12        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _21        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _22        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        %end;
        %if &j=2 %then %do;
            define _101       / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _102       / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _71        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
            define _72        / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        %end;

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

break after page / page;

compute after atptn;
    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: Geometric: mean, CV% and 95%
confidence interval (CI) are reported.';

```

```

                                line "Note: T${sub 0} = Time of first product use
at single use day.";
                                %if &nsum. ge 1 %then %do;
                                    line 'Note: LOQ = XX %'; /*Update this value if
required*/
                                %end;
                                    line ' ';
                                    line 'Appendix 15.3.3.4';
                                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_04_08_01_02.lst" new;
run;

proc contents data = table.T_15_02_04_08_01_02 varnum;
run;
ods listing close;
proc printto ; run;
*=====;
* END OF PROGRAM CODE ;
*=====;

```