

```

%_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_nic.sas;
%put NOTE: Purpose              : table of PK parameters of Nicotine;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADPP ADAM.ADSL;
%put NOTE: Output               : t_15_2_4_5(nic);
%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: 11Aug2014   JMH       1) Removed datasetep ADPP2;
%put NOTE: 23Sep2014   JR        2) Checked for missing values;
%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_05(nic);

%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=7;
        output;
    end;
    else if missing(analgrln) then delete;
run;

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

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

data tot2;
    set tot;
    call symput('trt' || compress(put(trtord,best.)), compress(total));
run;

/*Bring in appropriate data from ADpp*/
data adpp;
    set adam.adpp;
    where paramn in(1 2 3 4 5 6 7 8 9) AND PPROTFL='Y' AND ANL01FL='Y'
AND PPSTAT NE 'NOT DONE'; /* 2) JR 23Sep2014 */

run;

data supppp;
    set sdtm.supppp(where=(idvar='PPGRPID' and qnam='HLFLGE' and
qval='1'));
    format avisit $40.;

    if idvarval='DAY_1' then avisit='Day 1';
    else if idvarval='DAY_3' then avisit='Day 3';

    usubjid2=put(usubjid,$22.);
    drop usubjid;
    rename usubjid2=usubjid;

```

```

        keep usubjid2 qval avisit;
RUN;

proc sort data=adpp;
    by usubjid avisit;
run;

/*data adpp2;*/ /* 1) JMH 11Aug2014 */
/*    merge adpp(in=a) supppp;*/
/*    by usubjid avisit;*/
/*    if a;*/
/**/
/*    if qval='1' and paramn in (3 4 8 9) then delete;*/
/*run;*/

data adpp_orig;
    set ADPP/*adpp2*/; /* 1) JMH 11Aug2014 */

    if analgr1n=2 and trtan=1 then trtord=10;
    else trtord=trtan;

    statval=aval;
run;

proc sort data=adpp_orig;
    by analgr1n analgr1 trtord trta paramn param;
run;

proc means data=adpp_orig noprint;
    var statval;
    by analgr1n analgr1 trtord trta TRTAN paramn param;
    output out=results02 n=n1 NMISS=MISS1 mean=mean1 std=std1
    median=median1 min=min1 max=max1 q1=q1 q3=q3; /* 2) JR 23Sep2014 */
run;

data results03;
    set results02;
    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(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;

/*Obtain the geometric mean*/

data gmean;
    set adpp_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 trtord trta trtan paramn param;
run;

data gmean2;
    set gmean1;
    gmean1=exp(mean);

    gmean=left(compress(put(gmean1,8.2)));
    gcv=compress(put(0.01*ceil((sqrt(exp(std1*std1)-
1)*100)/0.01),8.2));
    gcv1=sqrt(exp(std1*std1)-1)*100;

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

    keep analgrln analgrl trtord trta TRTAN paramn param gmean gcv glci
guci std1;
run;

/*Combine the Gmean and BLQ with other stats*/
data results04;
    merge results03 gmean2;
    attrib gmeancv length=$20.;
    by analgrln analgrl trtord trta paramn param;
    if analgrln=2 and trtan=4 then trtan=10;
    if not missing(gcv) then gmeancv=left(compress(gmean)|| ' (' ||
compress(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;
run;

proc sort data=results04;
    by paramn param;
run;

```

```

proc transpose data=results04 out=results05 prefix=_ name=varname;
  by paramn param;
  var n meansd median minmax ci quart gmeancv;
  id trtan;
  idlabel trta;
run;

```

```

data results06;
  set results05;
  attrib stat length = $100.;

  if varname='N' then do;
    statord=1;
    stat='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 results07;
  set results06;

  if stat='n' then do;
    if missing(_1) then _1='0';
    if missing(_2) then _2='0';
    if missing(_10) then _10='0';
    if missing(_7) then _7='0';
  end;

```

```

run;

```

```

data labels;
set results07;
  attrib _1 label = "THS 2.2 Menthol$(N=&trt4)"
         _2 label = "mCC$(N=&trt5)"
         _10 label = "THS 2.2 Menthol$(N=&trt10)"
         _7 label = "NRT gum$(N=&trt7)"
         param label= "Formatted parameter"
         param1 label= "Unformatted parameter"
  paramc pfoot length=$200.;

  param1=param;

  if paramn=3 then do;
    paramc="AUC${sub (0-\uc8\u8734 infinity)} (h*ng/mL)";
    pfoot="AUC${sub (0-\uc8\u8734 infinity)} = Area under the
plasma concentration-time curve from start of product use extrapolated to
infinite time.";
    end;
  else if paramn=2 then do;
    paramc="AUC${sub (0-last)} (h*ng/mL)";
    pfoot="AUC${sub (0-last)} = Area under the plasma
concentration-time curve from start of product use to the time of the
last quantifiable concentration.";
    end;
  else if paramn=5 then do;
    paramc="AUC${sub (0-t')} (h*ng/mL)" ;
    PFOOT="AUC${sub (0-t')} = Area under the plasma
concentration-time curve from start of product use to the subject-
specific t'.";
    END;
  else if paramn=4 then do;
    paramc="% AUC${sub extrap} (%)";
    pfoot="% AUC${sub extrap} = AUC that is due to extrapolation
from tlast to infinity.";
    paramc=tranwrd(paramc,'% ','%');
    pfoot=tranwrd(pfoot,'% ','%');
    end;
  else if paramn=1 then do;
    paramc= "C${sub max} (ng/mL)";
    pfoot="C${sub max} = Maximum observed plasma concentration.";
    end;
  else if paramn=7 then do;
    paramc="t${sub last} (h)" ;
    pfoot="t${sub last} = Time of last quantifiable
concentration.";
    end;
  else if paramn=6 then do;
    paramc="t${sub max} (min)" ;
    pfoot="t${sub max} = Time of maximum observed plasma
concentration.";
    end;
  else if paramn=9 then do;
    paramc= "\uc6\u955 lambda${sub z} (1/h)";

```

```

        pfoot="\uc6\u955 lambda${sub z} = Terminal elimination rate
constant, estimated by linear regression analysis of the natural log-
transformed concentration-time data.";
        end;
    else if paramn=8 then do;
        paramc="t${sub 1/2} (h)";
        PFOOT="t${sub 1/2} = Terminal elimination half-life.";
        end;
    else put "WA" "RNING: Unexpected value of paramn " paramn= param=;

flag=1;

if paramn in(6 7) and statord not in(1 5 6 7) then delete;

run;

proc sql noprint;
    create table table.T_15_02_04_05 as
    select paramc, paraml, stat, _1, _2, _10, _7
    from labels
    order by paramn, statord;
quit;

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

%macro outrtf(blankn=, halfblnk=);

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

data paging;
    set labels;
    by paramn statord;
    if first.paramn 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.)));
run;

options number nodate orientation=landscape papersize=&p_pgsize 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.tl106326 (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;
%let geo=0;

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

    /* Amend title as needed */
    _firtitl="Table 15.2.4.5 Descriptive Statistics of Pharmacokinetic
Parameters of Nicotine - PK Population";
    _uppas=(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;
    call symput("pfoot",pfoot);
    if statord in(3 4 8) then call symput('geo',1);
    drop _firtitl _uppas len;
run;

ods listing close;
ods proclabel = ' ';
* 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 %then %do; contents=' ' %end; %else %do; contents='' %end;;
    column flag page paramn paramc statord stat ("Group-1 PK &linebot"
_1 _2) ("Group-2 PK &linebot" _10 _7);

        define flag          / order order = internal noprint;
        define page          / order order = internal noprint;
    define paramn            / order order=internal noprint;
    define paramc            / group style={just=left cellwidth=2cm}
style(header)={just=center} "Variable";
    define statord          / order order = internal noprint;

```



```

        define stat          / display style={just=left cellwidth=2cm}
style(header)={just=center} "Statistic";
        define _1           / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        define _2           / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        define _10          / display style={just=c cellwidth=1.5cm}
style(header)={just=center};
        define _7           / display style={just=c cellwidth=1.5cm}
style(header)={just=center};

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

break after page / page;

compute after paramn;
        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 ' ';
        line 'Appendix 15.3.3.1';
        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_05.lst" new;
run;

```

```
proc contents data = table.T_15_02_04_05 varnum;
run;
ods listing close;
proc printto ; run;
*=====;
* END OF PROGRAM CODE ;
*=====;
```