

```

%_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_phyex.sas;
%put NOTE: Purpose              : Summary of Physical Examination of Body
Systems - Safety Population;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADSL ADAM.ADPE;
%put NOTE: Output               : t_15_2_6_17(phyex);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_aobyrne;
%put NOTE: Creation Date        : 2014-03-06;
%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  AOB        1) Column headers amended;
%put NOTE: 21Sep2014  KB         2) Amended clinical significance;
%put NOTE: 21Sep2014  KB         3) Added baseline timepoint and
footnote;
%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_17(phyex);

/* 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 ;
*****;

/*Bring in ADSL for N values in column headers*/
data adsl1;
    set adam.adsl;
    where saffl = 'Y' and enr1fl='Y';
    if missing(trtseqa) then delete;
    if index(trtseqa,'Expos') then delete;
    headorder1=trtseqa;
    headtext1=trtseqa;
    output;
    trtseqa=99;
    headorder1=99;
    trtseqa='Overall Safety';
    headtext1='Overall Safety';
    output;
run;

data dumtrts; /*Use this to output any columns for which N=0*/
    attrib headtext1 length=$200.
            headorder1 length=8.;
    headorder1=1;
    headtext1='THS 2.2 Menthol - mCC';
    output;
    headorder1=2;
    headtext1='mCC - THS 2.2 Menthol';
    output;
    headorder1=3;
    headtext1='THS 2.2 Menthol - NRT gum';
    output;
    headorder1=4;
    headtext1='NRT gum - THS 2.2 Menthol';
    output;
    headorder1=5;
    headtext1='Enrolled not randomized';
    output;
run;

proc sort data=adsl1 out=adsl1_x; by headorder1 headtext1 usubjid; run;

data adsl;
    merge adsl1_x dumtrts;
    by headorder1 headtext1;
run;

```

```

proc sort data=adsl1 nodupkey out=adslcoll; by headorder1 headtext1
usubjid; run;

data adslcol;
    merge adslcoll(in=a) dumtrts;
    by headorder1 headtext1;
    if a;
run;

proc freq data=adslcol noprint;
    table headorder1*headtext1/ out =tot(drop=percent);
run;

data tot2;
    merge dumtrts(in=a) tot(in=b);
    by headorder1 headtext1;
    if a and not b then count=0;
    call symput('trt' || compress(put(headorder1,best.)),
compress(count));
run;

/*Bring in ADPE*/

data adpe;
    set adam.adpe(where=(anl01fl='Y' and saffl='Y'));
    attrib result length=$100.;

    pespidn=input(pespid,best.);

/* 3) START KB 21Sep2014 */
IF ABLFL='Y' THEN DO;
    AVISIT='Baseline';
    AVISITN=100;
END;
IF AVISITN LE 100 AND AVISIT NE 'Baseline' THEN DELETE;
/* 3) END KB 21Sep2014 */

/*Create numeric values for ordering of results*/

    if avalc='Normal' then do;
        aval=1;
        result='Normal - n (%)';
    end;
    else if avalc='Abnormal' then do;
        if peclsig=/'N'/'NCS' then do; /* 2) KB 21Sep2014 */
            aval=2;
/*
            result='Abnormal non-clinically relevant ? n (%)'; */
            RESULT='Abnormal non-clinically significant - n (%)';
/* 2) KB 21Sep2014 */
            end;
        else if peclsig=/'Y'/'CS' then do; /* 2) KB 21Sep2014 */
            aval=3;

```

```

/*          result= 'Abnormal clinically relevant  n (%)'; */
          RESULT= 'Abnormal clinically significant - n (%)';  /*
2) KB 21Sep2014 */
          end;
          else put "WA" "RNING: Unexpected value of avalc " subjid=
avalc= ;
          end;
          else if avalc='Not Examined' then do;
              aval=4;
              result='Not Examined - n (%)';
          end;

run;

data adpel(drop = trtsega trtsegan);
    set adpe;
    attrib headtext1 length =$200.
              headorder1 length=8.;

    if missing(trtsegan) then delete;
    if index(trtsega,'Expos') then delete;

    headorder1=trtsegan;
    headtext1=trtsega;
    output;
    headorder1=99;
    headtext1='Overall Safety';
    output;
run;

proc sort data=adpel;
    by headorder1 headtext1;
run;

proc freq data=adpel;
    table headorder1*headtext1*paramn*param*avisitn*avisit*PECLSIG /
noprnt out=nvals(drop=percent); /* 2) KB 21Sep2014 */
run;

data nvals1;
    set nvals;
    result='n';
    aval=0.5;
run;

/*Create an output with all parameters so these can be merged onto
treatment sequences with no results*/
proc sort data=adpel nodupkey out=paramlist1(keep=paramn param aval
result);
    by paramn param aval result;
run;

/*Merge the parameters onto every treatment sequence*/
data paramlist;

```

```

set paramlist1;
attrib headtext1 length =$200.
               headorder1 length=8.;

headorder1=1;
headtext1='THS 2.2 Menthol - mCC';
output;
headorder1=2;
headtext1='mCC - THS 2.2 Menthol';
output;
headorder1=3;
headtext1='THS 2.2 Menthol - NRT gum';
output;
headorder1=4;
headtext1='NRT gum - THS 2.2 Menthol';
output;
headorder1=5;
headtext1='Enrolled not randomized';
output;
headorder1=99;
headtext1='Overall Safety';
output;

run;

proc sort data=paramlist;
  by headorder1 headtext1;
run;

data totparams;
  merge tot2(rename=count=total) paramlist;
  by headorder1 headtext1;
run;

/*Now output all of the parameters for all timepoints*/
data alldata;
  set totparams;
  /* 3) START KB 21Sep2014 */
  /*  avisitn=1;*/
  /*  avisit='Screening          ' */
  /*  output;*/
  avisitn=/*99*/100;
  /*  avisit='Day -1            ' */
  AVISIT='Baseline              '
  /* 3) END KB 21Sep2014 */
  output;
  avisitn=104;
  avisit='Day 4/Discharge';
  output;
run;

proc sort data=adpel;
  by headorder1 headtext1 paramn param avisitn avisit;
run;

proc sort data=alldata nodupkey;

```

```

        by headorder1 headtext1 paramn param avisitn avisit;
run;

/*Merge on the actual data with the dataset of parameters and visits*/
data adpe2;
    merge adpe1(in=a) alldata(in=b);
    by headorder1 headtext1 paramn param avisitn avisit;
    if a or b;
    if b and not a then dumflag=1; /*This flag will allow us to
manipulate values which are in a treatment sequence with no results*/
run;

proc sort data=adpe2;
    by headorder1 headtext1 paramn param avisitn avisit aval;
run;

proc freq data=adpe2;
    table
headorder1*headtext1*paramn*param*avisitn*avisit*aval*result*total*dumfla
g*PECLSIG / noprint out=adpe3(drop=percent); /* 2) KB 21Sep2014 */
run;

data adpe4;
    merge adpe3 nvals1;
    by headorder1 headtext1 paramn param avisitn avisit aval;
    attrib text text2 length=$20. PECLSIG2 LENGTH=$3; /* 2) KB
21Sep2014 */

        PECLSIG2=PECLSIG; /* 2) KB 21Sep2014 */

    if total ne 0 then percent=count / total *100;
    else percent=0;

        /*n value*/
        if missing(count) then text='0';
        else text=put(count,3.);

        /*% value*/
        if missing(percent) or percent=0 then text2='';
        else if percent=100 then text2='(100 %)';
        else if percent ge 10 then text2='(
'||compress(put(percent,8.1))||'%)';
        else if percent lt 10 then text2='(
'||compress(put(percent,8.1))||'%)';

        if dumflag=1 and aval=0.5 then do; /*We want n to =0 for
empty treatment sequences, and then other results to be blank*/
            text='0';
            text2='';
        end;
        else if dumflag=1 and aval ne 0.5 then do;
            text='';
            text2='';
        end;

```

```

end;

if aval=0.5 then text2='';

    RENAME PECLSIG2=PECLSIG; /* 2) KB 21Sep2014 */
    DROP PECLSIG; /* 2) KB 21Sep2014 */
run;

/*This acts as another dummy to merge on all possible results*/
proc sort data=adpe4 nodupkey out=allrows(keep=headorder1 headtext1
paramn param avisitn avisit dumflag); /* 4) KB 21Sep2014 */
    by headorder1 headtext1 paramn param avisitn avisit dumflag;
run;

data allrows2;
    set allrows;
    attrib result length=$100. PECLSIG LENGTH=$3; /* 2) KB 21Sep2014 */

    aval=1;
    result='Normal - n (%)';
    output;
    aval=2;
/*    result='Abnormal non-clinically relevant - n (%)';*/
    RESULT='Abnormal non-clinically significant - n (%)'; /* 2) KB
21Sep2014 */
    PECLSIG='CS'; /* 2) KB 21Sep2014 */
    output;
    aval=3;
/*    result='Abnormal clinically relevant - n (%)';*/
    RESULT='Abnormal clinically significant - n (%)'; /* 2) KB
21Sep2014 */
    PECLSIG='NCS'; /* 2) KB 21Sep2014 */
    output;
run;

proc sort data=adpe4;
    by headorder1 headtext1 paramn param avisitn avisit aval result
PECLSIG dumflag; /* 2) KB 21Sep2014 */
run;

proc sort data=allrows2;
    by headorder1 headtext1 paramn param avisitn avisit aval result
PECLSIG dumflag; /* 2) KB 21Sep2014 */
run;

/*Merge all the possible results onto the actual data*/
data final;
    merge adpe4(in=a) allrows2(in=b);
    by headorder1 headtext1 paramn param avisitn avisit aval result
PECLSIG dumflag; /* 2) KB 21Sep2014 */
    if a or b;
    if b and not a then do;
        if dumflag=1 then do;
            text='';

```

```

        text2='';
    end;
    else do;
        text='0';
        text2='';
    end;
end;
drop total count percent;
run;

proc sort data=final;
    by paramn param avisitn avisit aval result;
run;

/*Transpose the n results*/
proc transpose data=final out=tfinal_n prefix=n;
    by paramn param avisitn avisit aval result;
    id headorder1;
    idlabel headtext1;
    var text;
run;

/*Transpose the percentages*/
proc transpose data=final out=tfinal_p prefix=p;
    by paramn param avisitn avisit aval result;
    id headorder1;
    idlabel headtext1;
    var text2;
run;

/*Combine the n and percentage results*/
data tfinal;
    merge tfinal_n tfinal_p;
    by paramn param avisitn avisit aval result;

    if aval=0.5 then do;
        if missing(n1) then n1='0';
        if missing(n2) then n2='0';
        if missing(n3) then n3='0';
        if missing(n4) then n4='0';
        if missing(n5) then n5='0';
        if missing(n99) then n99='0';
    end;

    flag=1;
    param1=upcase(substr(param,1,1))||lowercase(substr(param,2));
    drop param;
    rename param1=param;

    if index(avisit,'/') then avisit=tranwrd(avisit,'/',' ');
run;

proc sql noprint;
    create table table.t_15_02_06_17 as

```



```

        select param, avisit, result, n1, p1, n2, p2, n3, p3, n4, p4, n5,
p5, n99, p99
        from tfinal
        order by paramn, avisitn, aval;
quit;

proc sort data=tfinal;
    by paramn avisitn aval;
run;

data paging;
    set tfinal;
    by paramn avisitn aval;

    if first.paramn or (first.avisitn and ln>5) then ln=1;
    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;
%let linebot2 = \brdrb\brdrs\brdrw15;

/* 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;

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

```

```

/* Amend title as needed */
_firtitl="Table 15.2.6.17 Summary of Physical Examination of
Body Systems - Safety Population";
_upcas=(length(_firtitl)-
length(compress(_firtitl,'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 proclabel = ' ';
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 nowd split = '#' %if &i=1 %then
%do; contents=' ' %end; %else %do; contents=' ' %end;;
    column flag page paramn ("Body System" param) avisitn ("Study
Day"avisit) aval ("Statistic" result)
("Sequence &linebot." ("THS 2.2 Menthol#- mCC#(N=&trt1)"/*"THS 2.2
Menthol -#mCC#(N=&trt1)"/ n1 p1) ("mCC -#THS 2.2 Menthol#(N=&trt2)" n2
p2) /* 1 AOB 12Aug2014 */
("THS 2.2 Menthol -#NRT gum#(N=&trt3)" n3 p3) ("NRT gum#- THS 2.2
Menthol#(N=&trt4)" /*"NRT gum -#THS 2.2 Menthol#(N=&trt4)"/ n4 p4) /* 1)
AOB 12Aug2014 */
("Enrolled Not#Randomized#(N=&trt5)" n5 p5)) ("Overall#Safety#(N=&trt99)"
n99 p99) ;

    define flag          / order order = internal noprint;
    define page          / order order = internal noprint;
    define paramn        / order order = internal noprint;
    define avisitn       / order order = internal noprint;
    define aval          / order order = internal noprint;
    define param         / group style={just=left cellwidth=2.2cm} "";
    define avisit        / group style={just=left cellwidth=1.8cm}
"";
    define result        / display style={just=left cellwidth=3cm}
"";
    define n1-n5         / display style={just=d cellwidth=0.5cm} "";
    define n99           / display style={just=d cellwidth=0.5cm} "";
    define p1            / display style={just=d cellwidth=1.2cm} "";
    define p2            / display style={just=d cellwidth=1.2cm} "";
    define p3            / display style={just=d cellwidth=1.2cm} "";
    define p4            / display style={just=d cellwidth=1.2cm} "";
    define p5            / display style={just=d cellwidth=1.3cm} "";

```

```

define p99 / display style={just=d cellwidth=1.2cm} "";

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

break after page / page;

compute after avisitn;
  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 all 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 the last available time point prior
to the product test (THS 2.2 Menthol or NRT gum) at Admission (Day -1).
'; /* 3) KB 21Sep2014 */
  line "";
  line "Appendix 15.3.6.10";
  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_17.LST" new;
run;

proc contents data = table.t_15_02_06_17 varnum;

```

```
run;
ods listing close;
proc printto ; run;
*=====;
*  END OF PROGRAM CODE                      ;
*=====;
```