

```

%_mprintto;
options notes nosource;
proc datasets lib=work nolist memtype=data kill; quit;
%put NOTE:
=====;
%put NOTE: Covance Study Number : 000000106324;
%put NOTE: Client Protocol ID   : ZRHR-REXC-03-EU;
%put NOTE: Program Name        : t_comp.sas;
%put NOTE: Purpose              : summary of compliance;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADBX;
%put NOTE: Output               : t_15_02_05_01(comp);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_jriley;
%put NOTE: Creation Date        : 2014-08-01;
%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: 05Aug2014   JMH       1) Amended data selection;
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

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

%let tflno=T_15_02_05_01(comp);

/* Standard - leave this */
%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;

/*Column header numbers*/
data adsl;
    set adam.adsl(where=(fasfl='Y'));

```

```

run;

proc sort data=adsl nodupkey out=adsl1;
  by trt01an trt01a subjid;
run;

proc freq data=adsl1(where=(not missing(trt01an))) noprint;
  table trt01an*trt01a/ out =tot(drop=percent rename=(count=total));
run;

data tot2;
  set tot;
  call symput('trt' || compress(put(trt01an,best.)),
compress(total));
run;

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

proc sort data=adam.adbx out=adbx(where=(paramcd='CO' and trta='SA' ));
  by paramcd avisitn atptn;
run;

data adbx1;
  attrib timep length=$100;
  set adbx;
/*   where lbdy in (1,2,5) and atptn in (12,14,20,22);*/ /* 1) JMH
05Aug2014 */
  WHERE AVISITN GT 100 AND AVISITN NE 106; /* 1) JMH 05Aug2014 */
  timep=trim(avisit)||' '||trim(atpt);
  if avalcat1='<=10' then grp=2;
  else if avalcat1='>10' then grp=3;
  else if missing(avalcat1) then grp=4;
run;

proc freq data=adbx1;
  table timep*grp/ noprint out=counts(drop=percent);
run;

proc sort data=adbx1(keep=timep lbdy atptn) out=timep nodupkey;
  by timep lbdy atptn;
run;

data timep1;
  set timep;
  do grp=2 to 4;
  output;
  end;
run;

data counts1;
  merge timep1 counts;
  by timep grp;

```

```

        if missing(count) then count=0;
run;

proc transpose data=counts1 out=countst;
    by timep lbdy atptn;
    id grp;
    var count;
run;

data num;
    set countst;
    _1=sum( _2,_3,_4);
run;

proc transpose data=num out=num2;
    by timep lbdy atptn ;
    var _2 _3 _4;
run;

data fout;
    attrib text2 length=$20. text3 length=$20.;
    set num2 num(in=a);
    per=100*count/&trt3;
    if missing(per) then text3='';
    else if per=100 then text3='(100 %)';
    else if per ge 10 then text3='( ||compress(put(per,8.1))||'%)';
    else if per lt 10 and per > 0 then text3='(
'||compress(put(per,8.1))||'%)';
    else if per = 0 then text3= '';

    if missing(count) or count = 0 then text2='0';
    else text2=compress(put(count,8.));

    if a then do;
        _name_='_1';
        text2=compress(put(_1,8.));
    end;
    rename _name_=ord;
run;

data fout1;
    attrib stat length=$100;
    set fout;
    if ord='_1' then stat='n';
    else if ord='_2' then stat='n (%) <=10ppm';
    else if ord='_3' then stat='n (%) >10ppm';
    else if ord='_4' and count ne 0 then stat='<Missing>';
    else delete;
run;

proc sort data=fout1;
    by lbdy atptn ord;

```

```

run;

proc sql noprint;
    create table table.T_15_02_05_01 as
    select timep, stat, text2, text3
    from fout1
    order by lbdy, atptn, ord;
quit;

data paging;
    set fout1;;
    by lbdy atptn ord;
    flag=1;
    retain page;
    if first.lbdy 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.tl06324 (read) ;
ods results off;
ods rtf toc_data
file="/cvn/projects/prj/data/000000106324/TFL/&TFL_Part./&tflno..rtf"
style=tl06324 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 15.2.5.1    Summary of Compliance as Measured by
Exhaled CO (ppm) in the SA Arm- 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;

```

```

* 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 nowd split = '#' %if
&i=1 %then %do; contents=' ' %end; %else %do; contents='' %end;;;
    column flag page lbdy atptn  ("Timepoint" timep) ord ("Statistic"
stat) ("SA#(N=&trt3)" text2 text3);

```

```

        define flag          / order order = internal noprint;
        define page          / order order = internal noprint;
        define lbdy          / order order=internal noprint;
        define atptn         / order order=internal noprint;
        define timep         / display group style={just=left cellwidth=10cm}
style(header)={just=left}"";
        define ord           / order=internal noprint;
        define stat          / display style={just=left cellwidth=9cm}
style(header)={just=left}"";
        define text2         / display style={just=d cellwidth=0.8cm}
style(header)={just=center}"";
        define text3         / display style={just=d 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 before page / style={protectspecialchars=off};;
    line "&linetop";
endcomp;

```

```

compute before atptn / style={protectspecialchars=off};;
    line " ";
endcomp;

```

```

        compute after page/style={just=left cellwidth=5cm
protectspecialchars=off};
        line "&linebot" ;
        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};
        line "Note: SA = Smoking abstinence.";
        line "Note: Percentages are based on the number of subjects
indicated in the column header (N).";
        line "";
        line "Appendix 15.3.3.2";
        line "Path: &TFLpath." &_blankn.*"\~\~" "(Page &i of &page)";
        line "Program Run: &sysdate &sysuserid Program Status:
&status";
        endcomp;
run;
%end;
ods rtf close;

ods path sashelp.tmplmst (read);

%mend ;

%outrtf(blankn=72, halfblnk=N);
ods listing;
proc printto print = "&table./T_15_02_05_01.lst" new;
run;

proc contents data = table.T_15_02_05_01 varnum;
run;
ods listing close;
proc printto ; run;
*=====;
* END OF PROGRAM CODE ;
*=====;

```