

```

%_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_cough.sas;
%put NOTE: Purpose              : table of cough assessments;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADQSSYM ADAM.ADSL;
%put NOTE: Output               : t_15_2_6_20(cough);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_kbooth;
%put NOTE: Creation Date        : 2014-06-27;
%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: 25Jul2014  AOB        1)  Paging amended;
%put NOTE: 25Jul2014  AOB        2)  Event zero counts removed;
%put NOTE: 25Jul2014  AOB        3)  Wrapping of Variable column
amended;
%put NOTE: 04Aug2014  JR         4)  Footnote ordering amended;
%put NOTE: 04Aug2014  JR         5)  Amended appendix number;
%put NOTE: 04Aug2014  JR         6)  Amended wrapping;
%put NOTE: 04Aug2014  JR         7)  Removed gap;
%put NOTE: ;
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

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

%let tflno=T_15_02_06_20(cough);

%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 ;
*****;
data ex int courfeq sput;
    set adam.adqssym(where=(saffl='Y' and anl01fl='Y'));
    if missing(trtan) then delete;
    if index(trta,'Expos') then delete;
    if paramn=1 then output ex;
    if paramn=3 and not missing(aval) then output int;
    if paramn=4 and not missing(aval) then output courfeq;
    if paramn=5 and not missing(aval) then output sput;
run;
data ex01;
    set ex;
    if missing(avalc) then avalc='Missing';
run;

proc sort data=ex01;
    by subjid descending avalc;
run;

proc freq data=ex01 noprint;
    tables subjid*trtan*avalc / out=ex02(drop=percent);
run;

proc sort data=ex02;
    by subjid descending avalc;
run;

data ex03;
    set ex02;
    by subjid descending avalc;
    if first.subjid;
run;

proc summary data=ex03;
    class avalc trtan;
    var count;
    output out=ex04
    sum(count)=sum;
run;

data ex04a(where=(avalc ne '' and trtan ne .));
    set ex04(keep=avalc trtan sum);
run;

proc sort data=ex04a;
    by descending avalc;

```

```

run;
proc transpose data=ex04a out=ex_ev02 (drop=_name_ _label_) prefix=ev;
  by descending avalc;
  id trtan;
  var sum;
run;
data ex_ev03;
  set ex_ev02;
  rown=_n_+1;
  variable=put(avalc,$80.);
run;
/*Pull out event numbers for bottom half of table*/
data int_ev;
  set int;
  if missing(aval) then delete;
run;
proc freq data=int_ev;
  table subjid*trtan*avalc*aval / noprint out=int_ev01(drop=percent);
run;
proc sort data=int_ev01;
  by subjid descending aval;
run;

data int_ev01a;
  set int_ev01;
  by subjid descending aval;

  if first.subjid then output;
run;

proc summary data=int_ev01a;
  class aval avalc trtan;
  var count;
  output out=int_ev01b
  sum(count)=sum;
run;

data int_ev01c(where=(avalc ne '' and aval ne . and trtan ne .));
  set int_ev01b(keep=aval avalc trtan sum);
run;

proc sort data=int_ev01c;
  by aval avalc;
run;

proc transpose data=int_ev01c out=int_ev02 (drop=_name_ _label_)
prefix=ev;
  by aval avalc;
  id trtan;
  var sum;
run;
data int_ev03;
  set int_ev02;
  rown=aval+6;

```

```

        variable=put(avalc,$80.);
run;
/*Pull out n numbers for top half*/
proc sort data=ex01 out=ex_sub1;
    by trtan usubjid descending avalc ;
run;

proc sort data=ex_sub1 out=ex_sub2 nodupkey;
    by trtan usubjid descending avalc;
run;

proc sort data=ex_sub2;
    by descending avalc;
run;

proc freq data=ex_sub2 noprint;
    tables trtan*usubjid*avalc / out=ex_sub3(drop=percent
rename=(count=count2));
run;

proc sort data=ex_sub3;
    by trtan usubjid descending avalc;
run;

proc sort data=ex_sub3 nodupkey out=ex_sub4;
    by trtan usubjid ;
run;

proc freq data=ex_sub4;
    table trtan*avalc / noprint out=ex_sub01(drop=percent);
run;

proc sort data=ex_sub01;
    by descending avalc;
run;

proc transpose data=ex_sub01 out=ex_sub02 (drop=_name_ _label_)
prefix=sub;
    by descending avalc;
    id trtan;
    var count;
run;
data ex_sub03;
    set ex_sub02;
    rown=_n_+1;
    variable=put(avalc,$80.);
run;
/*Pull out n numbers for bottom third*/
data int_sub;
    set int(where=(not missing(aval)));
    by subjid;
    retain subcount maxint;
    maxint=max(maxint,aval);
    if first.subjid then do;

```

```

        maxint=aval;
        subcount=subjid;
        end;
    keep subjid aval avalc maxint subcount trtan;
run;
proc sort data=int_sub out=int_sub01;
    by subjid descending aval;
run;
data int_sub02;
    set int_sub01;
    by subjid descending aval;
    if not first.subjid then delete;
run;
proc freq data=int_sub02;
    table trtan*aval*avalc / noprint out=int_sub03;
run;
proc sort data=int_sub03;
    by aval avalc;
run;
proc transpose data=int_sub03 out=int_sub04 (drop=_name_ _label_)
    prefix=sub;
    by aval avalc;
    id trtan;
    var count;
run;
data int_sub05;
    set int_sub04;
    length variable $80.;
    rown=aval+6;
    variable=put(avalc,$50.);
run;

/*FREQUENCY OF COUGH*/
data courfeq2;
    set courfeq;
    if missing(avalc) then avalc='Missing';
run;

proc sort data=courfeq2;
    by subjid descending aval;
run;

proc freq data=courfeq2 noprint;
    tables subjid*trtan*aval*avalc / out=courfeq3(drop=percent);
run;

proc sort data=courfeq3;
    by subjid descending aval;
run;

data courfeq4;
    set courfeq3;
    by subjid descending aval;
    if first.subjid;

```

```

run;

proc summary data=courfeq4;
    class aval avalc trtan;
    var count;
    output out=courfeq5
    sum(count)=sum;
run;

data courfeq6(where=(avalc ne '' and trtan ne . and aval ne .));
    set courfeq5(keep=avalc trtan sum aval);
run;

proc sort data=courfeq6;
    by aval avalc;
run;

proc transpose data=courfeq6 out=courfeq7 (drop=_name_ _label_)
    prefix=ev;
    by aval avalc;
    id trtan;
    var sum;
run;

data courfeq8; /* THIS IS NUMBER OF EVENTS FOR SOUGH FREQUENCY */
    set courfeq7;
    rown=aval+13;
    variable=put(avalc,$80.);
run;

data courfeqsub;
    set courfeq(where=(not missing(aval)));
    by subjid;
    retain subcount maxint;
    maxint=max(maxint,aval);
    if first.subjid then do;
        maxint=aval;
        subcount=subjid;
    end;
    keep subjid aval avalc maxint subcount trtan;
run;

proc sort data=courfeqsub out=courfeqsub1;
    by subjid descending aval;
run;

data courfeqsub2;
    set courfeqsub1;
    by subjid descending aval;
    if not first.subjid then delete;
run;

proc freq data=courfeqsub2;
    table trtan*aval*avalc / noprint out=courfeqsub3;
run;

proc sort data=courfeqsub3;
    by aval avalc;

```

```

run;
proc transpose data=courfeqsub3 out=courfeqsub4 (drop=_name_ _label_)
prefix=sub;
    by aval avalc;
    id trtan;
    var count;
run;
data courfeqsub5; /*THIS IS NUMBER OF SUBJECTS FOR COUGH FREQUENCY*/
    set courfeqsub4;
    length variable $80.;
    rown=aval+13;
    variable=put(avalc,$50.);
run;

/*SPUTUM*/
data sput2;
    set sput;
    if missing(avalc) then avalc='Missing';
run;

proc sort data=sput2;
    by subjid descending aval;
run;

proc freq data=sput2 noprint;
    tables subjid*trtan*aval*avalc / out=sput3(drop=percent);
run;

proc sort data=sput3;
    by subjid descending aval;
run;

data sput4;
    set sput3;
    by subjid descending aval;
    if first.subjid;
run;

proc summary data=sput4;
    class aval avalc trtan;
    var count;
    output out=sput5
    sum(count)=sum;
run;

data sput6(where=(avalc ne '' and trtan ne . and aval ne .));
    set sput5(keep=avalc aval trtan sum);
run;

proc sort data=sput6;
    by aval avalc;
run;

proc transpose data=sput6 out=sput7 (drop=_name_ _label_) prefix=ev;

```

```

        by aval avalc;
        id trtan;
        var sum;
run;

data sput8; /* THIS IS NUMBER OF EVENTS FOR SPUTUM */
    set sput7;
    rown=aval+21;
    variable=put(avalc,$80.);
run;

data sputsub;
    set sput(where=(not missing(aval)));
    by subjid;
    retain subcount maxint;
    maxint=max(maxint,aval);
    if first.subjid then do;
        maxint=aval;
        subcount=subjid;
    end;
    keep subjid aval avalc maxint subcount trtan;
run;
proc sort data=sputsub out=sputsub1;
    by subjid descending aval;
run;
data sputsub2;
    set sputsub1;
    by subjid descending aval;
    if not first.subjid then delete;
run;
proc freq data=sputsub2;
    table trtan*aval*avalc / noprint out=sputsub3;
run;
proc sort data=sputsub3;
    by aval avalc;
run;
proc transpose data=sputsub3 out=sputsub4 (drop=_name_ _label_)
prefix=sub;
    by aval avalc;
    id trtan;
    var count;
run;
data sputsub5; /*THIS IS NUMBER OF SUBJECTS FOR SPUTUM*/
    set sputsub4;
    length variable $80.;
    rown=aval+21;
    variable=put(avalc,$50.);
run;

data rows;
    length variable $80.;
    rown=1; variable='Has subject experienced cough in the study period
assessed?';
    output;

```



```

rown=2; variable='Yes';
output;
rown=3; variable='No';
output;
rown=4;      variable='';
output;
rown=5;      variable='';
output;
rown=6;      variable='Intensity of cough';
output;
rown=7;      variable='Very mild';
output;
rown=8; variable='Mild';
output;
rown=9; variable='Moderate';
output;
rown=10; variable='Severe';
output;
rown=11; variable='Very severe';
output;
rown=12; variable='';
output;
rown=13; variable='Frequency of cough';
output;
rown=14; variable='Rarely';
output;
rown=15; variable='Sometimes';
output;
rown=16; variable='Fairly often';
output;
rown=17; variable='Often';
output;
rown=18; variable='Almost always';
output;
rown=19; variable='';
output;
rown=20; variable='Amount of sputum';
output;
rown=21; variable='No sputum';
output;
rown=22; variable='A moderate amount of sputum';
output;
rown=23; variable='A large amount of sputum';
output;
rown=24; variable='A very large amount of sputum';
output;
run;
data table;
    merge ex_ev03 ex_sub03 int_ev03 int_sub05 sputsub5 sput8
courfeqsub5 courfeq8 rows;
    by rown;
/*    length variable1 variable2 $200.; */ /* 3) AOB 25Jul2014 */
    if rown=4 and variable='Missing' then go to zero;
    else if rown=4 and variable ne 'Missing' then delete;

```

```

        else if rown in(2,3,7,8,9,10,11, 14, 15, 16, 17, 18, 21, 22, 23,
24) then do;
            zero: if missing(ev1) then ev1=0;
            if missing(ev2) then ev2=0;
            if missing(ev3) then ev3=0;
            if missing(ev97) then ev97=0;
            if missing(sub1) then sub1=0;
            if missing(sub2) then sub2=0;
            if missing(sub3) then sub3=0;
            if missing(sub97) then sub97=0;
        end;
        ov_sub=sub1+sub2+sub3+sub97;
        ov_ev=ev1+ev2+ev3+ev97;attrib wrap length = $200;
wrap =variable;
IF ROWN NOT IN(1 5 6 12 13 19 20) THEN DO;/* 3)  AOB 25Jul2014 */
i=/*11*/30;/* 3)  AOB 25Jul2014 */
if length(wrap)>i then
    do;
        fin=0;
        do while(fin=0);
            if substr(wrap,i,1)=' ' then
                do;
                    wrap=substr(wrap,1,i-1) || "$n
$S={foreground=white} . $S={}" || substr(wrap,i+1);
                    fin=1;
                end;
            else i=i-1;
        end;
    end;
end;
/* START 3)  AOB 25Jul2014 */
    VARIABLE="$S={foreground=white} . $S={}"||WRAP;
END;
/*    if rown in(2,3,4,7,8,9,10,11, 14, 15, 16, 17, 18, 21, 22, 23, 24)
then variable1="$S={foreground=white} . $S={}" || wrap ;*/
/*    if rown in (22, 23, 24) then variable1=tranwrd(variable1,'of
sputum','$n $S={foreground=white} . $S={}'||'of sputum'); */
/*    variable2=variable1; */
/*    if variable2='' and variable ne '' then variable2=variable; */
    bynum=1;
/*    drop aval variable1; */
/* END 3)  AOB 25Jul2014 */
run;
data adsl;
    set adam.adsl(where=(trt01an not in (98 99) and not
missing(trt01an)));
    output;
    trt01an=99;
    trt01a='Overall Randomized';
    output;

    rename trt01an=trtan trt01a=trta;
run;
proc freq data=adsl noprint;
    table trtan*trta/ out =tot(drop=percent);

```

```

run;
data dumtrts;
attrib trta length=$200. trtan length=8.;
    dumres=0;
    rown=0.5;
    trtan=1;
    trta='THS 2.2';
    output;
    trtan=2;
    trta='CC';
    output;
    trtan=3;
    trta='SA';
    output;
    trtan=97;
    trta='Enrolled not randomized';
    output;
    trtan=99;
    trta='Overall Safety';
    output;
run;
data tot2;
    merge dumtrts(in=a drop=dumres rown) tot(in=b);
    if a or b;
    if a and not b then count=0;
    by trtan;
    rename count=total;
    call symput('trt' || compress(put(trtan,best.)), compress(count));
run;
proc transpose data=tot2 out=sum prefix=_;
    id trtan;
    var total;
run;
data sum01;
    set sum;
    total_overall=_1+_2+_3+_97;
    bynum=1;
    rename _1=total_1 _2=total_2 _3=total_3 _97=total_97;
    drop _name_ _label_;
run;
data table01;
    merge sum01 table;
    attrib      sub1 sub2 sub3 sub97 ov_sub label="n"
                p1 p2 p3 p97 ov_p label='(%)' length=$8.
                ev1 ev2 ev3 ev97 ov_ev label="Events";
    by bynum;
    if sub1>0 then pla=(sub1/total_1)*100;
    if sub2>0 then p2a=(sub2/total_2)*100;
    if sub3>0 then p3a=(sub3/total_3)*100;
    if sub97>0 then p97a=(sub97/total_97)*100;
    if ov_sub>0 then ov_pa=(ov_sub/total_overall)*100;

    if pla=100 then p1=trim('('||compress(put(pla,8.))||' %');

```

```

        else if not missing(pla) and pla>10 and pla<100 then p1=trim('(
||compress(put(pla,8.1))||'%)');
        else if not missing(pla) and pla<10 then p1=trim('(
||compress(put(pla,8.1))||'%)');

        if p2a=100 then p2=trim('( ||compress(put(p2a,8.))||' %)');
        else if not missing(p2a) and p2a>10 and p2a<100 then p2=trim('(
||compress(put(p2a,8.1))||'%)');
        else if not missing(p2a) and p2a<10 then p2=trim('(
||compress(put(p2a,8.1))||'%)');

        if p3a=100 then p3=trim('( ||compress(put(p3a,8.))||' %)');
        else if not missing(p3a) and p3a>10 and p3a<100 then p3=trim('(
||compress(put(p3a,8.1))||'%)');
        else if not missing(p3a) and p3a<10 then p3=trim('(
||compress(put(p3a,8.1))||'%)');

        if p97a=100 then p97=trim('( ||compress(put(p97a,8.))||' %)');
        else if not missing(p97a) and p97a>10 and p97a<100 then p97=trim('(
||compress(put(p97a,8.1))||'%)');
        else if not missing(p97a) and p97a<10 then p97=trim('(
||compress(put(p97a,8.1))||'%)');

        if ov_pa=100 then ov_p=trim('( ||compress(put(ov_pa,8.))||' %)');
        else if not missing(ov_pa) and ov_pa>10 and ov_pa<100 then
ov_p=trim('( ||compress(put(ov_pa,8.1))||'%)');
        else if not missing(ov_pa) and ov_pa<10 then ov_p=trim('(
||compress(put(ov_pa,8.1))||'%)');

/*    variable3=left(strip(variable2)); *//* 3)  AOB 25Jul2014 */

/* START 2)  AOB 25Jul2014 */
    ARRAY A [5] SUB1 SUB2 SUB3 SUB97 OV_SUB;
    ARRAY B [5] EV1 EV2 EV3 EV97 OV_EV;
    DO I=1 TO 5;
        IF A[I]=0 AND B[I]=0 THEN B[I] =.;
    END;
/* END 2)  AOB 25Jul2014 */
    IF ROWN=12 THEN DELETE; /* 7)  JR 04Aug2014 */
run;

proc sql noprint;
    create table table.T_15_02_06_20 as
    select variable, /*variable3,*/ sub1, p1, ev1, sub2, p2, ev2, sub3,
p3, ev3, sub97, p97, ev97, ov_sub, ov_p, ov_ev/* 3)  AOB 25Jul2014 */
    from table01
    order by rown;
quit;

data paging;
    set table01;
    by rown;

```

```

        if /*rown>3 and variable=''*/ MISSING(OV_SUB) AND LN>8 then ln=1; /*
1) AOB 25Jul2014 */
        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;
%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 page_x=0;
ods proclabel = ' ';

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

        _firtitl="Table 15.2.6.20 Summary of Cough Assessments Over
Study- Safety Population";
        _upcas=(length("Path: &TFLpath.")-
length(compress("Path:&TFLpath.",'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
        len=&blankn.-ceil(_upcas)-length("Path: &TFLpath.")+6-
length("(Page &i of &page)");
        if eof then do;
            call symput('_FSRTITL', trim(left(_firtitl)));
            call symput('_blankn', compress(put(len,best.)));
            if page in (2 3 4) then call symput('PAGE_X','1');
        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;
/* Update with your variables as needed */
ODS LISTING CLOSE; /* 4) KB 18Jun2014 */
proc report data = comp headline headsip nowd split = '$' /*ps = 60 ls =
120*/%if &i=1 %then %do; contents=' ' %end; %else %do; contents='' %end;;
    column flag page
/* START 4) AOB 25Jul2014 */
/*(variable*//3*//*) ("Sequence &linebot" ("THS 2.2$(N=&trt1)" sub1 p1
ev1) ("CC$(N=&trt2)" sub2 p2 ev2)*//3) AOB 25Jul2014 */
/*("SA$(N=&trt3)" sub3 p3 ev3) */
/*("Enrolled Not$Randomized$(N=&trt97)" sub97 p97
ev97)) ("Overall$Safety$(N=&trt99)" ov_sub ov_p ov_ev); ;*/
/*          define flag                / order order = internal noprint;*/
/*          define page                / order order = internal noprint;*/
/*          define variable/*3*//*/ / display style={just=1
cellwidth=2.5cm} style(header)={just=center}"Variable";*//3) AOB
25Jul2014 */
/*          define sub1                / display style={just=d cellwidth=0.4cm}
style(header)={just=center};*/
/*          define ev1                / display style={just=c
cellwidth=1.23cm} style(header)={just=center};*/
/*          define sub2                / display style={just=d
cellwidth=0.4cm} style(header)={just=center};*/
/*          define ev2                / display style={just=c
cellwidth=1.23cm} style(header)={just=center};*/
/*          define sub3                / display style={just=d
cellwidth=0.4cm} style(header)={just=center};*/
/*          define ev3                / display style={just=c
cellwidth=1.23cm} style(header)={just=center};*/
/*          define sub97               / display style={just=d
cellwidth=0.4cm} style(header)={just=center};*/
/*          define ev97               / display style={just=c
cellwidth=1.23cm} style(header)={just=center};*/
/*          define ov_sub              / display style={just=d
cellwidth=0.5cm} style(header)={just=center};*/
/*          define ov_ev              / display style={just=c
cellwidth=1.23cm} style(header)={just=center};*/
/*          define p1                 / display style={just=r cellwidth=1.42cm}
style(header)={just=c};*/
/*          define p2                 / display style={just=r cellwidth=1.42cm}
style(header)={just=c};*/
/*          define p3                 / display style={just=r cellwidth=1.42cm}
style(header)={just=c};*/
/*          define p97                / display style={just=r cellwidth=1.42cm}
style(header)={just=c};*/
/*          define ov_p                / display style={just=r cellwidth=1.42cm}
style(header)={just=c};*/

```

```

("Variable" VARIABLE) ("Sequence &linebot" ("THS 2.2$(N=&trt1)"("n (%)
Events" SUB1 P1 EV1))
("CC$(N=&trt2)"("n (%) Events" SUB2 P2 EV2))
("SA$(N=&trt3)"("n (%) Events" SUB3 P3 EV3))
("Enrolled Not$Randomized$(N=&trt97)"("n (%) Events" SUB97 P97 EV97)))
("Overall$Safety$(N=&trt99)"("n (%) Events" OV_SUB OV_P OV_EV)); ;

        DEFINE FLAG                / ORDER ORDER = INTERNAL NOPRINT;
        DEFINE PAGE                / ORDER ORDER = INTERNAL NOPRINT;
        DEFINE VARIABLE            / DISPLAY STYLE={JUST=L
CELLWIDTH=3.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE SUB1                / DISPLAY STYLE={JUST=D CELLWIDTH=0.3CM}
STYLE(HEADER)={JUST=CENTER}"";
        DEFINE SUB2                / DISPLAY STYLE={JUST=D
CELLWIDTH=0.3CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE SUB3                / DISPLAY STYLE={JUST=D
CELLWIDTH=0.3CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE SUB97               / DISPLAY STYLE={JUST=D CELLWIDTH=0.3CM}
STYLE(HEADER)={JUST=CENTER}"";
        DEFINE OV_SUB              / DISPLAY STYLE={JUST=D
CELLWIDTH=/*0.3*/0.5CM} STYLE(HEADER)={JUST=CENTER}""; /* 6) JR 04Aug2014
*/
        DEFINE EV1                 / DISPLAY STYLE={JUST=C
CELLWIDTH=0.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE EV2                 / DISPLAY STYLE={JUST=C
CELLWIDTH=0.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE EV3                 / DISPLAY STYLE={JUST=C
CELLWIDTH=0.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE EV97                / DISPLAY STYLE={JUST=C
CELLWIDTH=0.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE OV_EV               / DISPLAY STYLE={JUST=C
CELLWIDTH=0.4CM} STYLE(HEADER)={JUST=CENTER}"";
        DEFINE P1                  / DISPLAY STYLE={JUST=C CELLWIDTH=1.1CM}
STYLE(HEADER)={JUST=C}"";
        DEFINE P2                  / DISPLAY STYLE={JUST=C CELLWIDTH=1.1CM}
STYLE(HEADER)={JUST=C}"";
        DEFINE P3                  / DISPLAY STYLE={JUST=C CELLWIDTH=1.1CM}
STYLE(HEADER)={JUST=C}"";
        DEFINE P97                 / DISPLAY STYLE={JUST=C CELLWIDTH=1.1CM}
STYLE(HEADER)={JUST=C}"";
        DEFINE OV_P                / DISPLAY STYLE={JUST=C CELLWIDTH=1.1CM}
STYLE(HEADER)={JUST=C}"";
/* END 4) AOB 25Jul2014 */

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

        break after page / page;

        compute before page / style={just=left protectspecialchars=off};
        line "&linetop";
        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: CC = Conventional cigarettes; SA = Smoking
abstinence; THS = Tobacco Heating System.'; /* 4) JR 04Aug2014 */
        line 'Note: Enrolled Not Randomized refers to all subjects
enrolled but not randomized. Overall Safety refers to enrolled subjects
exposed to THS 2.2.';
/*        line 'Note: CC = Conventional cigarettes; SA = Smoking
abstinence; THS = Tobacco Heating System.';*/
        line "Note: Percentages are based on the number of subjects
indicated in the column header (N).";
        line 'Note: Cough experienced in the previous 24h by the
subjects is assessed in the morning of Day 0 to Day 6 (to assess Day -1
to Day 5).';
        %if &page_x=1 %then %do;
        line "Note: If subject has answered question more than once
then the most severe intensity is presented.";
        %end;
        line ' ';
/*        line 'Appendix 15.3.6.13';*/
        line 'Appendix 15.3.6.14'; /* 5) JR 04Aug2014 */
        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=138, halfblnk=N);

ods listing;
proc printto print = "&table./T_15_02_06_18.lst" new;
run;

proc contents data = table.T_15_02_06_18 varnum;
run;
ods listing close;

```



```
proc printto ; run;  
*=====;  
* END OF PROGRAM CODE ;  
*=====;
```