

```

%_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_cough2.sas;
%put NOTE: Purpose              : table of cough assessments by study
day;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADQSSYM ADAM.ADSL;
%put NOTE: Output               : t_15_2_6_18_1(cough);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_aobyrne;
%put NOTE: Creation Date        : 2014-07-04;
%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  AOB       1) Footnotes amended;
%put NOTE: 11Aug2014  AOB       2) Column header amended;
%put NOTE: 11Aug2014  AOB       3) Presentation of zero counts
amended;
%put NOTE: 23Sep2014  JR        4) Stats amended and nc footnoted;
%put NOTE: 24Sep2014  JR        5) Removed 0 for VAS;
%put NOTE: 29Sep2014  JR        6) Amended header as per client
comments;
%put NOTE: 07Oct2014  JMH       7) Updated as per cleint comments;
%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_18_01(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;

/*Use ADSL to get N values for column headers*/
data adsl;
    set adam.adsl(where=(saffl='Y'));
    output;
    trtseqan=99;
    trtseqa='Overall Safety';
    output;
run;

proc sort data=adsl nodupkey out=adsl1;
    by trtseqan trtseqa subjid;
run;

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

data tot2;
    set tot;
    call symput('trt' || compress(put(trtseqan,best.)),
compress(total));
    rename total=COUNT;
run;

proc transpose data=tot2 out=tot3 prefix=total;
    var count;
    id trtseqan;
    idlabel trtseqa;
run;

*****;
* read in data ;
*****;
data adqssym;
    set adam.adqssym(where=(saffl='Y' and anl01fl = "Y"));
    /* 7) start JMH 07Oct2014 */
    IF AVISITN=100 THEN AVISIT='Day -1';
    ELSE IF AVISITN=101 THEN AVISIT='Day 0';
    ELSE IF AVISITN=102 THEN AVISIT='Day 1';
    ELSE IF AVISITN=103 THEN AVISIT='Day 2';
    ELSE IF AVISITN=104 THEN AVISIT='Day 3';
    ELSE PUT "WA" "RNING: Unexpected value fo AVISITN, please check
code " AVISITN= ;

```

```

/* 7) end JMH 07Oct2014 */

output;
trtseqan=99;
trtsega='Overall Safety';
output;
run;

proc sort data=adqssym(where=(paramn=2)) out=vas01;
  by trtseqan trtsega avisitn avisit;
run;

proc means data=vas01 noprint;
  by trtseqan trtsega avisitn avisit paramn;
  var aval;
  output out=vas02 n=n1 mean=mean1 std=std1 median=median1 min=min1
max=max1;
run;

data vas03;
  set vas02;
  n = left(compress(put(n1,8.)));
/* START 4) JR 23Sep2014 */
  IF N1 GE 4 THEN DO;
    if not missing(median1) then median =
left(compress(put(median1,8.1)));
    if not missing(mean1) and not missing(std1) then meansd =
left(compress(put(round(mean1,0.1),8.1))) || ' (' ||
left(compress(put(0.01*ceil(std1/0.01),8.2))) || ')';
    if not missing(min1) and not missing(max1) then minmax =
left(compress(put(min1,8.))) || ', ' || left(compress(put(max1,8.)));
  END;
  ELSE IF N1 < 4 AND N1 > 0 THEN DO;
    MEDIAN = 'NC';
    MEANSD = 'NC';
    MINMAX = 'NC';
  END;

  drop mean1 std1 _: n1;
run;
/* END 4) JR 23Sep2014 */

proc sort data=vas03;
  by avisitn avisit;
run;

proc transpose data=vas03 out=vas04 prefix=_ ;
  by avisitn avisit paramn;
  var n meansd median minmax;
  id trtseqan;
  idlabel trtsega;
run;

data vas05;

```

```

        set vas04;
        attrib stat format=$20.;
        by avisitn avisit;
        if first.avisit then varnum=1;
        else varnum+1;
        if _name_='MEDIAN' then stat='Median';
        else if _name_='MINMAX' then stat='Min, Max';
        else if _name_='MEANSD' then stat='Mean (SD)';
        else stat=lowcase(_name_);
        vargroup=2;
        drop _name_;
run;

/*Pull out n numbers for categorical parameters*/

proc freq data=adqssym(where=(paramn ne 2)) noprint;
    tables trtseqan*trtseqa*avisit*avisitn*paramn*param*avalc /
out=cat01;
run;

proc sort data=cat01;
    by avisit avisitn paramn param avalc;
run;

proc transpose data=cat01 out=cat02 prefix=a_
    by avisit avisitn paramn param avalc;
    var count;
    id trtseqan;
    idlabel trtseqa;
run;

data cat03;
    merge cat02 tot3;
    by _name_;
    length var $132.;
    stat='n (%)';
    var=avalc;
run;

proc sort data=cat03;
    by paramn avisitn ;
run;

data blanks;
    set cat03;
    by paramn avisitn;
    if not first.avisitn then delete;
    varnum=0.5;
    vargroup=paramn;
    if paramn=1 then do; var='Has subject experienced cough in last 24
hours'; end;
    if paramn=3 then do; var='Intensity of cough'; end;
    if paramn=4 then do; var='Frequency of cough'; end;
    if paramn=5 then do; var='Amount of sputum produced';end;

```

```

        keep var paramn avisitn avisit varnum;
run;

proc sort data=cat03;
    by avisitn ;
run;

data blanks2;
    set cat03;
    by avisitn;
    if not first.avisitn then delete;
    var=tranwrd(avisit,'/','/ ');
    varnum=0;
    vargroup=paramn;
    keep paramn avisitn avisit var;;
run;

data vcat01;
    set vas05(in=a) cat03 blanks blanks2;
    if var='Yes' then varnum=1;
    if var='No' then varnum=2;
    if var='Very mild' then varnum=1;
    if var='Mild' then varnum=2;
    if var='Moderate' then varnum=3;
    if var='Severe' then varnum=4;
    if var='Very severe' then varnum=5;
    if var='Rarely' then varnum=1;
    if var='Sometimes' then varnum=2;
    if var='Fairly often' then varnum=3;
    if var='Often' then varnum=4;
    if var='Almost always' then varnum=5;
    if var='No sputum' then varnum=1;
    if var='A moderate amount of sputum' then varnum=2;
    if var='A larger amount of sputum' then varnum=3;
    if var='A very large amount of sputum' then varnum=4;
    if varnum=0 and not index(var,'Day ') then varnum=0.5;
run;

proc sort data=vcat01;
    by avisitn avisit paramn varnum var;
run;

data dumrows;
    set cat02(keep=avisit:);
    by avisitn;
    if not first.avisitn then delete;
    length var $132.;
    stat='n (%)';
    paramn=1; var='Yes'; varnum=1; vargroup=1; output;
    paramn=1; var='No'; varnum=2; vargroup=1; output;
    paramn=3; var='Very mild'; varnum=1; vargroup=3; output;
    paramn=3; var='Mild'; varnum=2; vargroup=3; output;
    paramn=3; var='Moderate'; varnum=3; vargroup=3; output;
    paramn=3; var='Severe'; varnum=4; vargroup=3; output;

```

```

paramn=3; var='Very severe'; varnum=5; vargroup=3;output;
paramn=4; var='Rarely';varnum=1; vargroup=4; output;
paramn=4; var='Sometimes'; varnum=2; vargroup=4;output;
paramn=4; var='Fairly often'; varnum=3; vargroup=4;output;
paramn=4; var='Often'; varnum=4; vargroup=4;output;
paramn=4; var='Almost always'; varnum=5; vargroup=4;output;
paramn=5; var='No sputum'; varnum=1; vargroup=5;output;
paramn=5; var='A moderate amount of sputum';
varnum=2;vargroup=5;output;
paramn=5; var='A larger amount of sputum';
varnum=3;vargroup=5;output;
paramn=5; var='A very large amount of sputum';
varnum=4;vargroup=5;output;
run;

proc sort data=dumrows;
by avisitn avisit paramn varnum var;
run;

data vcat02;
merge vcat01 dumrows;
by avisitn avisit paramn varnum var;

array a[6] a_1 a_2 a_3 a_4 a_5 a_99;
array b[6] total1 total2 total3 total4 total5 total99;
array c[6] _1 _2 _3 _4 _5 _99;
array d[6] p_1 p_2 p_3 p_4 p_5 p_99;
if not missing(stat) then do i=1 to 6;
if not missing(a[i]) then do;
d[i]=round((a[i]/b[i])*100,0.1);
if d[i]=100 then c[i] = strip(put(a[i],best.)) || ' ' ||
'(' ||strip(put(d[i],5.)) || ' %)';
if 10<=d[i]<100 then c[i] = strip(put(a[i],best.)) || '
' || '( ' ||strip(put(round(d[i],0.1),5.1)) || ' %)';
if d[i]<10 then c[i] = strip(put(a[i],best.)) || ' ' ||
'(' ' ||strip(put(round(d[i],0.1),5.1)) || ' %)';
end;
if missing(c[i]) then c[i]='0';
end;
if var='Abandoned' then varnum=99;
if stat='n' then var='VAS';
if missing(vargroup) then vargroup=paramn;
drop a_ : p_ : total;;
run;

/* START 3) AOB 11Aug2014 */
DATA VCAT02A;
SET VCAT02;
BY AVISITN;
IF VAR NE 'Yes' THEN DELETE;
ARRAY E[6] _1 _2 _3 _4 _5 _99;
ARRAY F[6] FL_1 FL_2 FL_3 FL_4 FL_5 FL_99;
DO I=1 TO 6;
IF NOT FIRST.AVISITN AND E[I]='0' THEN F[I]=1;

```

```

        END;
        KEEP AVISITN FL_;;
RUN;
/* END 3)   AOB 11Aug2014 */

data vcat03;
/*      set vcat02;*/
MERGE VCAT02 VCAT02A; /* 3)   AOB 11Aug2014 */
BY AVISITN; /* 3)   AOB 11Aug2014 */
attrib wrap length=$132.;
wrap=var;
i=30;
if length(wrap)>i then do;
nwraps=int(length(wrap)/i);
do while(nwraps>0);
        fin=0;
        j=i*nwraps;
        do while(fin=0 and j gt 1);

                if substr(wrap,j,1)=' ' then do;
                        wrap=substr(wrap,1,j-1) || " |n |S={FOREGROUND=WHITE} .
|S={} " || substr(wrap,j+1);
                        fin=1;
                        end;
                else j=j-1;
                end;
        nwraps=nwraps-1;
        end;
        end;
        if stat='n (%)' then var = "|S={FOREGROUND=WHITE} . |S={} " || WRAP
|| wrap ;
        if varnum=0 then var = "|{style
[fontweight=bold]"||strip(var)||" ";
        if var='|S={FOREGROUND=WHITE} . |S={} ' then do;
                _1=''; _2=''; _3=''; _4=''; _5=''; _99=''; stat='';
                end;
/* START 3)   AOB 11Aug2014 */
        ARRAY E[6] _1 _2 _3 _4 _5 _99;
        ARRAY F[6] FL_1 FL_2 FL_3 FL_4 FL_5 FL_99;
        DO I=1 TO 6;
                IF PARAMN>1 AND F[I]=1 THEN E[I]='';
        END;
/* END 3)   AOB 11Aug2014 */
/* START 4) JR 23Sep2014 */
/*      ARRAY G[6] _1 _2 _3 _4 _5 _99; */
/*      DO I=1 TO 6;*/
/*      IF G[I]=' ' AND VAR='VAS' THEN G[I]=0;*/
/*      END;*/
/* END 4) JR 23Sep2014 */
        drop _name_ _label_ i param;
run;

```

```

proc sql noprint;

```

```

        create table table.t_15_02_06_18_01 as
        select var, stat, _1, _2, _3, _4, _5, _99
        from vcat03
        order by avisitn, paramn, varnum;
quit;

proc sort data=vcat03;
    by avisitn vargroup varnum;
run;

data paging;
    set vcat03;
    by avisitn vargroup varnum;
    if first.avisitn or (varnum<1 and ln>7) or ln>12 then ln=1;
    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.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;
%LET NC=0; /* 4) JR 23Sep2014 */
ods proclabel = ' ';

data comp;
    set page=end=eof;
    where page=&i;
    IF INDEX(_1, 'NC') OR INDEX(_1, 'NC') OR INDEX(_2, 'NC') OR
INDEX(_3, 'NC') OR INDEX(_4, 'NC') OR INDEX(_5, 'NC') OR INDEX(_99, 'NC')
THEN CALL SYMPUT('NC', 1); /* 4) JR 23Sep2014 */

```



```

        _firtitl="Table 15.2.6.18.1    Summary of Cough Assessments
by Study Day - Safety 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.)));
        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 */
proc report data = comp style(header)={just=center} headline headskip
missing nowd split = '#' /*ps = 60 ls = 120*/%if &i=1 %then %do;
contents=' ' %end; %else %do; contents='' %end;;
    column flag page avisitn vargroup ln
    (/*"Variable"*/"Study Day/#Variable" var)("Statistic" stat)("THS
2.2 Menthol#- mCC#(N=&trt1)" _1)("mCC -#THS 2.2 Menthol#(N=&trt2)" _2) /*
6) JR 29Sep2014 */
    ("THS 2.2 Menthol#- NRT gum#(N=&trt3)" _3)("NRT gum -#THS 2.2
Menthol#(N=&trt4)" _4)
    ("Enrolled Not#Randomized#(N=&trt5)"/*"Enrolled
not#randomized#(N=&trt5)"*/ _5)("Overall#Safety#(N=&trt99)" _99);/* 2)
AOB 11Aug2014 */

        define flag                                / order order = internal noprint;
        define page                                / order order = internal noprint;
        define avisitn                            / order order = internal noprint;
        define vargroup                            / order order = internal noprint;
        define ln                                  / order order = internal noprint;
        define var                                  / group style={just=l
cellwidth=5cm}"";
        define stat                                / display style={just=l
cellwidth=1.8cm}"";
        define _1-_5                                / display style={just=c
cellwidth=2cm}"";
        define _99                                / display style={just=c
cellwidth=2cm}"";

        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 vargroup / style={just=left protectspecialchars=off};
    line "";
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: mCC = menthol conventional cigarettes; NRT gum =
Nicotine Replacement Therapy gum; THS = Tobacco Heating System.'; /* 1)
AOB 11Aug2014 */
    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 for number of subjects with a cough
are calculated using N in the column header.'; */
    LINE 'Note: Percentages are based on the number of subjects
indicated in the column header (N).'; /* 1) AOB 11Aug2014 */
/*      line 'Note: The assessments performed at Day 0 to Day 4 will
be used to evaluate cough at Day -1 to Day 3.'; */
    LINE 'Note: The assessments performed at Day 0 to Day 4 are
used to evaluate cough at Day -1 to Day 3, respectively.'; /* 7) JMH
07Oct2014 */
%IF &NC = 1 %THEN %DO; /* 4) JR 23Sep2014 */
    LINE 'Note: NC = Not calculated.'; /* 4) JR 23Sep2014 */
%END; /* 4) JR 23Sep2014 */
    line "";
/*      line 'Appendix 15.3.6.14'; */
    LINE 'Appendix 15.3.6.13'; /* 1) AOB 11Aug2014 */
    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_18_01.lst" new;
run;

proc contents data = table.T_15_02_06_18_01 varnum;
run;
ods listing close;

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