

```
/*=====
=====*
```

```
| Covance Study Number   : 000000106343      |
| Program Name           : t_conmed.sas       |
| Purpose                 : Summary of Concomitant Medication      |
| Input Data              : ADAM.ADSL, ADAM.ADCM      |
| Output Data             : tlfds.T_15_02_06_19_02 & RTF      |
| Macros Called           : m_printto, m_logchk      |
| Originally Performed by : kpothuri           |
| Date                   : 27Apr2015            |
```

```
|
|=====
=====|
```

```
| Modification History      |
|-----|
| Modified by              :      |
| Modification Date       :      |
| Modification Description :      |
```

```
+=====
=====*/
```

```
options notes source source2 nofullstimer validvarname=upcase missing=' ';
```

```
ods _all_ close;
```

```
ods listing;
```

```
%m_printto(route=YES);
```

```
*=====;
```

```

* START OF PROGRAM CODE                                ;

*=====;

proc datasets lib=work nolist memtype=data kill; quit;

%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", ""));

        call symput('TFLprg', reverse(scan(strip(reverse(compress("&_SASPROGRAMFILE", ""))),1, "/" )));

run;

data cm;

    set adam.adcm;

    output;

    trta="Total";

    trtan=1;

    output;

run;

*N - counts - pre-randomization;

data adsl;

```

```

        set adam.adsl;

        output;

        trt01a="Total";

        trt01an=1;

        output;

run;

proc sql noprint;

        create table big_n as

        select trt01an, trt01a, count(distinct(usubjid))

        from adsl

        where SAFBFL="Y"

        group by trt01an, trt01a;

quit;

data big_n;

        set big_n;

        if trt01a="THSm2.2" then trt01a="THS";

        if trt01a="Product Test" then trt01a="ProdT";

run;

data _null_;

        set big_n;

        call symput ("B_" ||compress(trt01a), put(_TEMG001,8.));

run;

%put &B_SA &B_THS &B_mCC &B_ProdT &B_Total;

```

*N - counts - after randomization;

```
proc sql noprint;
```

```
    create table big_n_aft as
```

```
    select trt01an, trt01a, count(distinct(usubjid))
```

```
    from adsl
```

```
    where SAFAFL="Y"
```

```
    group by trt01an, trt01a;
```

```
quit;
```

```
data big_n_aft;
```

```
    set big_n_aft;
```

```
    if trt01a="THSm2.2" then trt01a="THS";
```

```
    if trt01a="Product Test" then trt01a="ProdT";
```

```
run;
```

```
data _null_;
```

```
    set big_n_aft;
```

```
    call symput ("A_" || compress(trt01a), put(_TEMP001,8.));
```

```
run;
```

```
%put &A_SA &A_THS &A_mCC &A_Total;
```

```
%macro pd (tab=, name=, where=, outdsn=);
```

```
*any medication;
```

```
proc sql noprint;
```

```
    create table anymed as
```

```
    select trta, trtan, count(distinct(usubjid))
```

```

from cm

where CMFL="Y" and &where

group by trta, trtan;

create table anymed_ev as

select trta, trtan, count(usubjid)

from cm

where CMFL="Y" and &where

group by trta, trtan;

quit;

proc transpose data=anymed out=t_anymed prefix=N_;

    id trta;

    var _temg001;

run;

proc transpose data=anymed_ev out=t_anymed_ev prefix=E_;

    id trta;

    var _temg001;

run;

data med;

length cmdecod $200;

merge t_anymed t_anymed_ev;

by _name_;

```

```

ord=1;

cmdecod="Any medication";

*pre-randomization;

%if &tab=1 %then %do;

  np_PRODT=strip(put(N_PRODUCT_TEST,8.) || " (" ||
compress(put(round(((N_PRODUCT_TEST/&B_ProdT)*100),.01),5.1)) || " "));

  np_TOTAL=strip(put(N_TOTAL,8.) || " (" ||
compress(put(round(((N_TOTAL/&B_Total)*100),.01),5.1)) || " "));

  if not missing (N_SA) then np_SA=strip(put(N_SA,8.) || " (" ||
compress(put(round(((N_SA/&B_SA)*100),.01),5.1)) || " "));

  if not missing (N_THSM2_2) then np_THS=strip(put(N_THSM2_2,8.) || " (" ||
compress(put(round(((N_THSM2_2/&B_THS)*100),.01),5.1)) || " "));

  if not missing (N_MCC) then np_MCC=strip(put(N_MCC,8.) || " (" ||
compress(put(round(((N_MCC/&B_MCC)*100),.01),5.1)) || " "));

%end;

*after randomization;

%if &tab=2 %then %do;

  if not missing (N_SA) then np_SA=strip(put(N_SA,8.) || " (" ||
compress(put(round(((N_SA/&A_SA)*100),.01),5.1)) || " "));

  if not missing (N_THSM2_2) then np_THS=strip(put(N_THSM2_2,8.) || " (" ||
compress(put(round(((N_THSM2_2/&A_THS)*100),.01),5.1)) || " "));

  if not missing (N_MCC) then np_MCC=strip(put(N_MCC,8.) || " (" ||
compress(put(round(((N_MCC/&A_MCC)*100),.01),5.1)) || " "));

  if not missing (N_TOTAL) then np_TOTAL=strip(put(N_TOTAL,8.) || " (" ||
compress(put(round(((N_TOTAL/&A_Total)*100),.01),5.1)) || " "));

%end;

```

```
run;
```

```
data dum;
```

```
    ord=1.5;
```

```
    cmdecod="";
```

```
    output;
```

```
run;
```

```
data med_1;
```

```
    merge med dum;
```

```
    by ord cmdecod;
```

```
run;
```

```
*preferred terms;
```

```
proc sql noprint;
```

```
    create table pt as
```

```
    select trta, trtan, cmdecod, count(distinct(usubjid))
```

```
    from cm
```

```
    where CMFL="Y" and &where
```

```
    group by trta, trtan, cmdecod;
```

```
    create table pt_ev as
```

```
    select trta, trtan, cmdecod, count(usubjid)
```

```
    from cm
```

```
    where CMFL="Y" and &where
```

```

        group by trta, trtan, cmdecod;

quit;

proc sort data=pt; by cmdecod; run;

proc transpose data=pt out=t_pt (drop=_name_) prefix=N_;

    id trta;

    var _temg001;

    by cmdecod;

run;

proc sort data=pt_ev; by cmdecod; run;

proc transpose data=pt_ev out=t_pt_ev (drop=_name_) prefix=E_;

    id trta;

    var _temg001;

    by cmdecod;

run;

data pref;

    merge t_pt t_pt_ev;

    by cmdecod;

    *pre-randomization;

    %if &tab=1 %then %do;

        if not missing (N_PRODUCT_TEST) then np_PRODT=strip(put(N_PRODUCT_TEST,8.) || " (" ||
compress(put(round(((N_PRODUCT_TEST/&B_ProdT)*100),.01),5.1)) || ")");

```



```

        np_TOTAL=strip(put(N_TOTAL,8.) || " (" ||
compress(put(round(((N_TOTAL/&B_Total)*100),.01),5.1)) || ")");

        if not missing (N_SA) then np_SA=strip(put(N_SA,8.) || " (" ||
compress(put(round(((N_SA/&B_SA)*100),.01),5.1)) || ")");

        if not missing (N_THSM2_2) then np_THS=strip(put(N_THSM2_2,8.) || " (" ||
compress(put(round(((N_THSM2_2/&B_THS)*100),.01),5.1)) || ")");

        if not missing (N_MCC) then np_MCC=strip(put(N_MCC,8.) || " (" ||
compress(put(round(((N_MCC/&B_MCC)*100),.01),5.1)) || ")");

    %end;

    *after randomization;

    %if &tab=2 %then %do;

        if not missing (N_SA) then np_SA=strip(put(N_SA,8.) || " (" ||
compress(put(round(((N_SA/&A_SA)*100),.01),5.1)) || ")");

        if not missing (N_THSM2_2) then np_THS=strip(put(N_THSM2_2,8.) || " (" ||
compress(put(round(((N_THSM2_2/&A_THS)*100),.01),5.1)) || ")");

        if not missing (N_MCC) then np_MCC=strip(put(N_MCC,8.) || " (" ||
compress(put(round(((N_MCC/&A_MCC)*100),.01),5.1)) || ")");

        np_TOTAL=strip(put(N_TOTAL,8.) || " (" ||
compress(put(round(((N_TOTAL/&A_Total)*100),.01),5.1)) || ")");

    %end;

run;

proc sort data=pref out=pref_1; by descending n_total descending e_total; run;

data pref_2;

    set pref_1;

    ord=2;

run;

```

```

data &outdsn (drop=cmdecod);

    set med_1 pref_2;

    cmdecod_=propcase(cmdecod);

    tabnum=&tab;

    set _name="&name";

run;

%mend;

%pd (tab=1, name=prior rand, where=%str(SAFBFL="Y" and ASPER=1), outdsn=b_rand);

%pd (tab=2, name=after rand, where=%str(SAFAFL="Y" and ASPER in (2,3,4)), outdsn=a_rand);


data comb (drop=_: n_:);

    set b_rand a_rand;


    if ord ^= 1.5 then do;

        if np_SA = "" then np_SA = "0";

        if np_THS = "" then np_THS = "0";

        if np_MCC = "" then np_MCC = "0";

        if np_PRODT = "" then np_PRODT = "0";

        if np_TOTAL = "" then np_TOTAL = "0";


        if e_SA = . then e_SA = 0;

        if e_THSm2_2 = . then e_THSm2_2 = 0;

        if e_MCC = . then e_MCC = 0;

        if e_PRODUCT_TEST = . then e_PRODUCT_TEST = 0;

        if e_TOTAL = . then e_TOTAL = 0;

```

```

end;

else do;

    call missing(np_SA, np_THS, np_MCC, np_PRODT, np_TOTAL, e_SA, e_THSm2_2,
e_MCC, e_PRODUCT_TEST, e_TOTAL);

end;

*line number may need update;

if tabnum=1 then line=_n_;

    else if tabnum=2 then line=_n_-64;

run;

/*Check for page overflows, this may need changing*/

data final;

set comb;

    if tabnum=1 then do;

        if ln gt 9 then ln=1;

            else ln+1;

        if ln=1 then page+1;

    end;

else if tabnum=2 then do;

        if ln_ gt 9 then ln_=1;

            else ln_+1;

        if ln_=1 then page_+1;

    end;

run;

```

```
data final_1 (drop=page rename=(page_=page));  
    set final;  
  
    if page_=0 then page_=page;  
    if tabnum=2 then do;  
        page_=page_+7; *may need to change;  
    end;  
  
    call symput("page",compress(put(page_,best.)));  
run;  
%put &page;
```

```
%let tflno = %str(T_15_02_06_19_02);
```

```
data tflds.&tflno;  
set final_1;  
run;
```

```
/* Standard - leave this */
```

```
options number nodate orientation=landscape missing=' ';
```

```
%let linetop = \brdrt\brdrs\brdrw30; * needs to be 1.5pt so calculated in twips (1/20 pt) ;
```

```
%let linebot = \brdrb\brdrs\brdrw30;
```

```
/* Standard - macro for paging */
```

```
%macro outrtf(blankn=130, halfblk=N);
```

```
%if &halfblk=N %then %let halfblk=;
```

```
%else %if &halfblk=Y %then %let halfblk=~;
```

```
ods path stdlib.t106343 (read) ;
```

```
ods results off;
```

```
ods rtf toc_data file="/cvn/projects/prj/data/000000106343/TFL/&TFL_Part./Tables/&tflno..rtf"  
style=t106343 startpage=yes headery=1440 footery=1440 ;
```

```
ods noproctitle;
```

```
%do i=1 %to &page;
```

```
title ;
```

```
footnote;
```

```
%let wd=0;
```

```
ods proclabel = ' ';
```

```
data comp;
```

```
set final_1 end=eof;
```

```
where page=&i;
```

```
/* Amend title as needed */
```

```
_firtitl="Table 15.2.6.19.2 Summary of Concomitant Medication by Preferred Drug  
Name - 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('period1', compress(put(tabnum,best.)));

        call symput('_blankn', compress(put(len,best.)));

    end;

drop _firtitl _upcas len;

run;

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 page line cmdecod_

%if &period1.=1 %then %do; ("THSm2.2$(N=%left(&B_THS))&linebot" np_ths e_thsm2_2)
("mCC$(N=%left(&B_mCC))&linebot" np_mcc e_mcc)

("SA$(N=%left(&B_SA))&linebot" np_sa e_sa) ("Product Test$(N=%left(&B_Prodt))&linebot" np_prodt
e_product_test)

("Overall Safety$(N=%left(&B_Total))&linebot" np_total e_total); %end;

```

```

%if &period1.=2 %then %do; ("THSm2.2$(N=%left(&A_THS))&linebot" np_ths e_thsm2_2)
("mCC$(N=%left(&A_mCC))&linebot" np_mcc e_mcc)

("SA$(N=%left(&A_SA))&linebot" np_sa e_sa)

("Overall Safety$(N=%left(&A_Total))&linebot" np_total e_total); %end;

define page / order order = internal noprint;

define line / order order = internal noprint;

define cmdecod_ /"Preferred Term" display style={just=left cellwidth=5.4cm}
style(header)={just=left} "";

define np_ths /"n (%)" display style={just=c cellwidth=1.4cm}
style(header)={just=center} ;

define e_thsm2_2 /"Events" display style={just=c cellwidth=1.0cm}
style(header)={just=center};

define np_mcc /"n (%)" display style={just=c cellwidth=1.2cm}
style(header)={just=center} ;

define e_mcc /"Events" display style={just=c cellwidth=1.0cm}
style(header)={just=center};

define np_sa /"n (%)" display style={just=c cellwidth=1.2cm}
style(header)={just=center} ;

define e_sa /"Events" display style={just=c cellwidth=1.0cm}
style(header)={just=center};

%if &period1.=1 %then %do;

define np_prodt /"n (%)" display style={just=c cellwidth=1.2cm}
style(header)={just=center} ;

define e_product_test /"Events" display style={just=c cellwidth=1.0cm}
style(header)={just=center};

%end;

define np_total /"n (%)" display style={just=c cellwidth=1.1cm}
style(header)={just=center} ;

define e_total /"Events" display style={just=c cellwidth=1.0cm}
style(header)={just=center};

```

break after page / page;

compute after page;

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 "";

%if &period1.=1 %then %do; line "Safety Time Period: Pre-Randomization"; %end;

%else %do; line "Safety Time Period: Randomization"; %end;

line "&linebot";

endcomp;

compute after _page_ / style={just=left protectspecialchars=off pretext="&linetop."};

%if &period1.=1 %then %do;

line 'Note: "Product Test" refers to all subjects who tested the THS product but were not randomized. The Overall Safety refers to all subjects exposed to THSm2.2.';

%end;


```

%else %do; line 'Note: The Overall Safety refers to all subjects exposed to THSm2.2.';
%end;

line 'Note: mCC = Menthol conventional cigarettes; SA = Smoking abstinence; THSm2.2 =
Tobacco Heating System 2.2 Menthol.';

line 'Note: Percentages are based on the number of subjects indicated in the column
header (N).';

line ";

line 'Appendix 15.3.6.4';

line "Study ID: ZRHM-REXA-08-US Program: &TFLprg Status: &status"
&_blankn.*"\~\~" "&sysdate" &_blankn.*"\~\~" "(Page &i of &page)";

endcomp;

run;

%end;

ods rtf close;

ods results on;

ods path sashelp.tmplmst (read);

%mend ;

%outrtf(blankn=36, halfblnk=N);

ods listing;

*=====;

* END OF PROGRAM CODE ;

*=====;

```

%m_logchk;