

/*-----

Covance Study ID : COV-000000106343

Program Name : t_demog.sas

Purpose : table of demographics Per-Protocol pop;

Author : cvn_pshe

Date of Creation : 5MAY015

Input Data : ADAM.ADSL, ADAM.ADVS, ADAM.ADQSD,

Output Data :

Macros Called :

Modification History

Modified by :

Modification Date :

Modification Description:

-----*/

proc datasets lib=work kill memtype=data nolist;

run;

%m_printto;

options notes nosource;

```
options mprint ;
```

```
options replace;
```

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

```
ods _all_ close;
```

```
ods listing;
```

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

```
* START OF PROGRAM CODE ;
```

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

```
%let tflno=T_15_02_01_04_03_01;
```

```
%let TFL_Part=%scan(&_amp;_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", ""));
```

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

```
run;
```

```
*****,
```

```

* read in data ;

*****.

/*Use ADSL to get N values for column headers*/

data adsl_n;

    set adam.adsl;;

                                output;

                                trt01an=9;

                                output;

run;

proc sql;

select count(distinct usubjid) into: N1THS from adsl_n(where=(trt01an = 4 and pprot1fl = "Y"));
select count(distinct usubjid) into: N1MCC from adsl_n(where=(trt01an = 5 and pprot1fl = "Y"));
select count(distinct usubjid) into: N1SAA from adsl_n(where=(trt01an = 3 and pprot1fl = "Y"));
select count(distinct usubjid) into: N1OPP from adsl_n(where=(trt01an = 9 and pprot1fl = "Y"));

select count(distinct usubjid) into: N2THS from adsl_n(where=(trt01an = 4 and pprot2fl = "Y"));
select count(distinct usubjid) into: N2MCC from adsl_n(where=(trt01an = 5 and pprot2fl = "Y"));
select count(distinct usubjid) into: N2SAA from adsl_n(where=(trt01an = 3 and pprot2fl = "Y"));
select count(distinct usubjid) into: N2OPP from adsl_n(where=(trt01an = 9 and pprot2fl = "Y"));

select count(distinct usubjid) into: N3THS from adsl_n(where=(trt01an = 4 and pprot3fl = "Y"));

```

```
select count(distinct usubjid) into: N3MCC from adsl_n(where=(trt01an = 5 and pprot3fl = "Y"));
select count(distinct usubjid) into: N3SAA from adsl_n(where=(trt01an = 3 and pprot3fl = "Y"));
select count(distinct usubjid) into: N3OPP from adsl_n(where=(trt01an = 9 and pprot3fl = "Y"));
```

```
select count(distinct usubjid) into: N4THS from adsl_n(where=(trt01an = 4 and pprot4fl = "Y"));
select count(distinct usubjid) into: N4MCC from adsl_n(where=(trt01an = 5 and pprot4fl = "Y"));
select count(distinct usubjid) into: N4SAA from adsl_n(where=(trt01an = 3 and pprot4fl = "Y"));
select count(distinct usubjid) into: N4OPP from adsl_n(where=(trt01an = 9 and pprot4fl = "Y"));
```

```
quit;
```

```
%macro demog(period=);
```

```
data adsl_all;
```

```
set adam.adsl;
```

```
if prodpref='THS 2.2 menthol' then prodprefgrn=1;
```

```
else if prodpref='mCC' then prodprefgrn=2;
```

```
else if prodpref='SA' then prodprefgrn=3;
```

```
else if prodpref='No preference' then prodprefgrn=4;
```

```
run;
```

```
data allpp;
```

```
set adsl_all(where=(pprot&period.fl='Y' and trt01an ne 98));
```

```
format trta $40.;
```

```

trta='Oversaf';

trtan=9;

output;

run;

/*Find the number of people in each treatment sequence for table headers*/

data adsl;

    set adsl_all(where=(pprot&period.fl='Y' and trt01an ne 98));

    format trta $40.;

    trta=trt01a;

    trtan=trt01an;

run;

data adsl1;

    set adsl allpp;

run;

proc sort data=adsl1;

    by usubjid trta trtan;

run;

proc freq data=adsl1 noprint;

    tables trta*trtan / out=adsl1tots(where=(not missing(trtan)) drop=percent);

run;

```

```
data dummy;  
    format trta $40.;  
    trta='THS 2.2';  
    trtan=1;  
    output;  
    trta='CC';  
    trtan=2;  
    output;  
    trta='SA';  
    trtan=3;  
    output;  
    trta='Oversaf';  
    trtan=9;  
    output;  
run;
```

```
proc sort data=adsltot;  
    by trtan trta;  
run;
```

```
proc sort data=dummy;  
    by trtan trta;  
run;
```

```
data adsl tots2;  
    merge adsl tots dummy;  
    by trtan trta;  
    if count=. then count=0;  
run;
```

```
data tot2a;  
    rename count=total;  
    set adsl tots2;  
run;
```

```
proc sort data=tot2a;  
    by trtan trta;  
run;
```

/* Obtaining data for body wt and vswc and FTND and QS1-5b */

```
data vs_wt;  
    set adam.advs(where=(paramcd='WEIGHT' and pprot&period.fl='Y' and trtan ne 98 and  
    avisit='Day -2'));  
    rename aval=bwtn avalc=bwtc;  
    keep usubjid aval avalc trta trtan;  
run;
```

```
proc sort data=vs_wt nodupkey ;  
    by usubjid trtan trta;
```

```
run;
```

```
data vs_wc;
```

```
    set adam.advs(where=(paramcd='WSTCIR' and pprot&period.fl='Y' and trtan ne 98 and  
avisit='Day -2'));
```

```
    rename aval=wcn avalc=wcc;
```

```
    keep usubjid aval avalc trta trtan;
```

```
run;
```

```
proc sort data=vs_wc nodupkey ;
```

```
    by usubjid trtan trta;
```

```
run;
```

```
data adqsndfsc (keep=usubjid ftcac ftcacn fscn fscn trta trtan );
```

```
    length ftcac $30;
```

```
    set adam.adqsnd (where=(paramcd='FTNDSC' and avisit='Screening' and pprot&period.fl='Y' and trtan  
ne 98));
```

```
    rename aval=fscn avalc=fscn;
```

```
    if avalcat1='Mild' then do;
```

```
        ftcacn=1;
```

```
            ftcac='Mild';
```

```
        end;
```

```
    else if avalcat1='Moderate' then do;
```

```
        ftcacn=2;
```

```
            ftcac='Moderate';
```

```
        end;
```



```

        else if avalcat1='Severe' then do;

                ftcatn=3;

        ftcat='Severe';

        end;

run;

proc sort data=adqsndfsc nodupkey;

        by usubjid trtan trta;

run;

data adqsndsesq1 (keep=usubjid sesq1c sesq1n trta trtan );

        length sesq1c $30;

        set adam.adqsnd(where=(paramcd='QSESEA' and pprot&period.fl='Y' and trtan ne 98));

                if aval=1 then do

                        sesq1n=1;

                                sesq1c='Low';

                end;

                else if aval=2 then do

                        sesq1n=2;

                                sesq1c='Moderate';

                end;

                else if aval in (3 4 5) then do

                        sesq1n=3;

                                sesq1c='High';

                end;

run;

```

```
run;
```

```
proc sort data=adqsndsesq1 nodupkey;
```

```
  by usubjid trtan trta;
```

```
run;
```

```
data adqsndsesq2 (keep=usubjid sesq2c sesq2n trta trtan);
```

```
  length sesq2c $30;
```

```
  set adam.adqsnd(where=(paramcd='QSESAHI' and pprot&period.fl='Y' and trtan ne 98));
```

```
    if avalcat1='Low' then do; sesq2n=1; sesq2c='Low'; end;
```

```
    else if avalcat1='Moderate' then do; sesq2n=2; sesq2c='Moderate' ;
```

```
end;
```

```
    else if avalcat1='High' then do; sesq2n=3; sesq2c='High'; end;
```

```
run;
```

```
proc sort data=adqsndsesq2 nodupkey;
```

```
  by usubjid trtan trta;
```

```
run;
```

```
data adqsndsesq3 (keep=usubjid sesq3c sesq3n trta trtan);
```

```
  length sesq3c $30;
```

```
  set adam.adqsnd(where=(paramcd='QSESCPU' and pprot&period.fl='Y' and trtan ne 98));
```

```
    sesq3n=aval;
```

```
    sesq3c=propcase(avalcat1);
```

```
run;
```

```
proc sort data=adqsndsesq3 nodupkey;  
    by usubjid trtan trta;  
run;
```

```
data allparm;  
    merge adqsndfsc vs_wc vs_wt adqsndsesq1 adqsndsesq2 adqsndsesq3;  
    by usubjid trtan trta;  
run;
```

```
data allparm;  
    set allparm;  
        output;  
        trta='Oversaf';  
    trtan=9;  
        output;  
run;
```

```
proc sort data= allparm;  
    by usubjid trtan trta;  
run;
```

```
proc sort data=adsl1;  
    by usubjid trtan trta;  
run;
```

```
data adslall;
```

```
    merge adsl1 allparm;
```

```
    by usubjid trtan trta;
```

```
run;
```

```
proc sort data=adslall nodupkey;
```

```
    by usubjid trtan trta;
```

```
run;
```

```
/*Macro for all the class variables*/;
```

```
%macro _class(var1=, var2=, dout1=, dout2=, var=, num=);
```

```
proc freq data=adslall noprint;
```

```
    table trtan*trta*&var1*&var2*sexn*sexc / out =&dout1(drop=percent);
```

```
run;
```

```
data &dout2._1;
```

```
    merge &dout1 tot2a(where=(total ne 0));
```

```
    by trtan trta;
```

```
    percent=count/total*100; /*This works out the percentages*/
```

```
run;
```

```
data &dout2;
```

```
    set &dout2._1;
```

```
    format var $200. stat $20. svar $20.;
```

```

order=&num;

var=strip(&var2);

stat='n (%)';

if percent=100 then svar = strip(put(count,best.)) || ' ' || '(' || strip(put(percent,5.)) || ' ');

if 10<=percent<100 then svar = strip(put(count,best.)) || ' ' || '('
|| strip(put(round(percent,0.1),5.1)) || ' ');

if percent<10 then svar = strip(put(count,best.)) || ' ' || '(' || strip(put(round(percent,0.1),5.1))
|| ' ');

run;

proc sort data=&dout2 ; by order &var1 var stat; run;

proc transpose data=&dout2 (where=(nmiss(trtan, sexn)=0)) out=&dout2._2 prefix=t;

by order &var1 var stat;

var svar;

id trtan sexn ;

run;

/*Dummy will add a blank line, only the variable name for TFLs will be presented*/

data dummy;

format var $200. stat $20.;

var="&var";

stat=' ';

order=&num;

%do;

%if &num=3 %then

```

	output;
var='WHITE';	
stat='n (%)';	
output;	
var='BLACK OR AFRICAN AMERICAN';	
	stat='n
(%)';	
output;	
	var=
'AMERICAN INDIAN OR ALASKA NATIVE';	
	stat='n
(%)';	
output;	
var='ASIAN';	
output;	
var='NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER';	
	stat='n
(%)';	
output;	
var='OTHER';	
	stat='n
(%)';	
output;	
var='';	
	stat='n
(%)';	
output;	

```
%end;
```

```
%if &num=602 %then %do;
```

```
output;
```

```
var='Underweight';
```

```
stat='n (%)';
```

```
output;
```

```
var='Normal weight';
```

```
stat='n (%)';
```

```
output;
```

```
var='Overweight';
```

```
stat='n (%)';
```

```
output;
```

```
var='Obese';
```

```
stat='n (%)';
```

```
output;
```

```
%end;
```

```
%if &num=902 %then %do;
```

```
output;
```

```
var='Mild';
```

```
stat='n (%)';
```

```
output;
```

```
var='Moderate';
```

```
stat='n (%)';
```

```
output;
```

```
var='Severe';
```

```

stat='n (%)';
output;

var=";
stat='n

(%)';
output;
%end;

13 %then %do;
%if &num = 11 | &num = 12 | &num =

output;

var='Low';
stat='n (%)';
output;
var='Moderate';
stat='n (%)';
output;
var='High';
stat='n (%)';
output;

var=";
stat='n

(%)';
output;

var=";
stat='n

(%)';
output;

```



```
%end;
```

```
run;
```

```
%if &num = 3 | &num = 902 | &num = 602 | &num = 11 | &num = 12 | &num = 13 %then %do;
```

```
proc sort data=&dout2._2;
```

```
by var;
```

```
run;
```

```
proc sort data=dummy;
```

```
by var;
```

```
run;
```

```
%end;
```

```
/*Combine the dummy and the datasets with actual data*/
```

```
data all&num.;
```

```
format var $200. stat $20.;
```

```
%if &num = 3 | &num = 902 | &num = 602 | &num = 11 | &num = 12 | &num = 13  
%then %do; merge dummy &dout2._2 (drop=order stat) ; %end; %else %do; set dummy &dout2._2;  
%end;
```

```
%if &num = 3 | &num = 902 | &num = 602 | &num = 11 | &num = 12 | &num = 13 %then %do;
```

```
by var;
```

```
%end;
```

```
%IF &NUM=1 %THEN %DO;
```

```
IF VAR='Male' THEN SORT=1;
```

```

ELSE IF VAR='Female' THEN SORT=2;

%END;

%if &num=3 %then %do;

    if var='WHITE' then do; var='White';

sort=1;end;

    else if var='BLACK OR AFRICAN AMERICAN' then do; var='Black or African American'; sort=2; end;

    else if var='AMERICAN INDIAN OR ALASKA NATIVE' then do; var='American Indian or Alaska
Native'; sort=3; end;

else if

var='ASIAN' then do; var='Asian'; sort=4; end;

else if

var='NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER' then do; var='Native Hawaiian or Other Pacific
Islander'; sort=5; end;

else if

var='OTHER' then do; var='Other'; sort=6; end;

    else if var=' ' then do; var='Missing'; sort=7; end;

%end;

%if &num=602 %then %do;

    if var='Underweight' then sort=1;

    else if var='Normal weight' then sort=2;

    else if var='Overweight' then sort=3;

    else if var='Obese' then sort=4;

%end;

%IF &NUM=8 %THEN %DO;

    IF VAR='10-19 cig/day' THEN SORT=1;

    ELSE IF VAR='>19 cig/day' THEN SORT=2;

    else if var=' ' then do; var='Missing'; sort=3; end;

```

%END;

%if &num=10 %then %do;

if var='THS 2.2 menthol' then sort=1;

else if var='mCC' then sort=2;

else if var='SA' then sort=3;

else if var='No preference' then sort=4;

else if

var=' ' then do; var='Missing'; sort=5; end;

%end;

%if &num=902 %then %do;

if var='Mild' then sort=1;

else if var='Moderate' then sort=2;

else if var='Severe' then sort=3;

else if

var=' ' then do;

var='Missing';

sort=4;

end;

%end;

%if &num=11 | &num=12 | &num=13

%then %do;

if var='Low' then sort=1;

else if var='Moderate' then sort=2;

else if var='High' then sort=3;

else if var=' ' then do; var='Missing'; sort=4;

end;

```

%end;

drop &var1;

run;

proc sort data=all&num.;

by sort;

run;

%mend _class;

%_class(var1=racen, var2=race, dout1=race, dout2=race2, var=Race, num=3);

%_class(var1=bmigr1n, var2=bmigr1, dout1=bmiclass, dout2=bmiclass2, var=%str(BMI (kg/m{\super
2})),num=602);

%_class(var1=ucpdgr1n, var2=ucpdgr1, dout1=ccconsum, dout2=ccconsum2, var=Cigarette
consumption,num=8);

%_class(var1=ftcatn, var2=ftcat, dout1=ftsc1, dout2=ftsc2, var=FTND score,num=902);

%_class(var1=prodprefgrn, var2=prodpref, dout1=prodpref, dout2=prodpref2, var=Individual product
preference, num=10);

%_class(var1=sesq1n, var2=sesq1c, dout1=sesq1, dout2=sesq12, var=%str(Day 4 SES Questionnaire
SES educational attainment), num=11);

%_class(var1=sesq2n, var2=sesq2c, dout1=sesq2, dout2=sesq22, var=%str(SES annual household
income), num=12);

%_class(var1=sesq3n, var2=sesq3c, dout1=sesq3, dout2=sesq32, var=%str(SES Composite), num=13);

/*End of macro for all the class variables*/

```

```
/*get data for continuous avriables*/
```

```
data adslall1;
```

```
    set adslall;
```

```
    if nmiss(height)=0 then height2=height/100;
```

```
run;
```

```
proc sort data=adslall1;
```

```
    by trtan trta sexn sexc;
```

```
run;
```

```
proc means data=adslall1 noprint;
```

```
    by trtan trta sexn sexc;
```

```
    var age height2 bwtn bmi fscn wcn; /*Make sure all variables in the split macro are here too*/
```

```
    output out=stats;
```

```
    output out=median median=;
```

```
run;
```

```
data stats1;
```

```
    set stats median(in=in2);
```

```
    if in2 then _stat_ = 'MEDIAN';
```

```
    rename _stat_=_name_;
```

```
run;
```

```
data stats2;
```

```

set stats1;

    format stat $20.;

    if _name_='N'    then order2=1001;

    if _name_='MEAN' then order2=1002;

    if _name_='STD'  then order2=1003;

    if _name_='MEDIAN' then order2=1004;

    if _name_='MIN'  then order2=1005;

    if _name_='MAX'  then order2=1006;


if _name_ eq 'STD' then stat='SD';

else if _name_ eq 'N' then stat='n';

else stat=propcase(_name_);

                        if sexn=. then delete;

run;


%macro split(set=,num=,var=,stat=,units=,dp=);

    data &set;

        format stat $20. svar $20.;

    set stats2(keep=trtan trta sexn sexc order2 stat &var);


    order=&num;


    %if &num=2 or &num=7 or &num=9 %then %do;

        if order2=1003 then svar=compress(put(0.01*ceil(&var/0.01),8.2));

        else if order2=1001 then svar=compress(put(&var,8.));

```

```

        else if order2=1002 or order2=1004 then svar=compress(put(&var,8.%eval(&dp+1)));

        else svar=compress(put(&var,8.&dp));

    %end;

    %if &num=6 or &num=4 %then %do;

        if order2=1003 then svar=compress(put(0.001*ceil(&var/0.001),8.3));

        else if order2=1001 then svar=compress(put(&var,8.));

        else if order2=1002 or order2=1004 then svar=compress(put(&var,8.%eval(&dp+1)));

        else svar=compress(put(&var,8.&dp));

    %end;

    %if &num=5 %then %do;

        if order2=1003 then svar=compress(put(0.0001*ceil(&var/0.0001),8.4));

        else if order2=1001 then svar=compress(put(&var,8.));

        else if order2=1002 or order2=1004 then svar=compress(put(&var,8.%eval(&dp+1)));

        else svar=compress(put(&var,8.&dp));

                                %end;

run;

```

/*Combine meand and SD*/

```

    data meansd&num;

        set &set(keep=trtan trta stat sexn sexc order2 svar &var);

        where stat in('Mean' 'SD'); /*Only keep mean and SD as this is all we want here*/

run;

```

```

proc sort data=meansd&num; by trtan trta sexn sexc; run;

```

```
proc transpose data=meansd&num out=meansd&num.a prefix=m;
```

```
    by trtan trta sexn sexc;
```

```
    id order2;
```

```
    var svar;
```

```
run;
```

```
data meansd&num.b;
```

```
    format stat $20. svar $20.;
```

```
    set meansd&num.a;
```

```
    svar=left(compress(m1002)) || ' (' || left(compress(m1003)) || ')';
```

```
    stat='Mean (SD)';
```

```
    order2=1002;
```

```
    order=&num;
```

```
    drop m1002 m1003;
```

```
run;
```

```
/*End of combining mean and SD*/
```

```
/*Combine Min and Max*/
```

```
data minmax&num;
```

```
    set &set(keep=trtan trta stat sexn sexc order2 svar &var);
```

```
    where stat in('Min' 'Max'); /*Only keep min and max as this is all we want here*/
```

```
run;
```

```
proc sort data=minmax&num; by trtan trta; run;
```



```

proc transpose data=minmax&num out=minmax&num.a prefix=m;

    by trtan trta sexn sexc;

    id order2;

    var svar;

run;

```

```

data minmax&num.b;

    format stat $20. svar $20.;

    set minmax&num.a;

    svar=left(compress(m1005)) || ', ' || left(compress(m1006));

    stat='Min, Max';

    order2=1005;

    order=&num;

    drop m1005 m1006;

run;

```

/*End of combining min and max*/

/*Combine the new datasets which have Min, Max and Mean (SD)*/

```

data minmaxmeansd;

    set minmax&num.b meansd&num.b;

run;

```

/*Add these new datasets to the original dataset, deleting individual min, max, mean and SD*/

```

data comb&set;

    set &set(where=(order2 not in(1002 1003 1005 1006))) minmaxmeansd;

```

```
run;
```

```
proc sort data= comb&set; by order order2 stat; run;
```

```
/*Transpose the data by treatment sequence*/
```

```
proc transpose data=comb&set (where=(nmiss(trtan, sexn)=0)) out=trans&set prefix=t;
```

```
by order order2 stat;
```

```
var svar;
```

```
id trtan sexn;
```

```
run;
```

```
/*This dummy will add in line with only the variable name to present in the TFLs*/
```

```
data dummy;
```

```
format var $200. stat $20.;
```

```
var="&stat (&units)";
```

```
stat="n";
```

```
order=&num;
```

```
ORDER2=1001;
```

```
run;
```

```
PROC SORT DATA=DUMMY; BY ORDER ORDER2 STAT; RUN;
```

```
PROC SORT DATA=trans&set; BY ORDER ORDER2 STAT; RUN;
```

```
/*Add the dummy to the transposed dataset*/
```

```
data all&num.;
```

```

format var $200. stat $20.;

MERGE dummy trans&set;

BY ORDER ORDER2 STAT;

run;

proc sort data=all&num.; by order2; run;

%mend split;

%split(set=age, num=2, var=age, stat=Age, units=years, dp=0);
%split(set=weight, num=4, var=bwtn, stat=Weight, units=kg, dp=1);
%split(set=height, num=5, var=height2, stat=Height, units=m, dp=2);
%split(set=bmi, num=6, var=bmi, stat=BMI, units=kg/m{\super 2}, dp=1);
%split(set=wc, num=7, var=wc, stat=Waist circumference, units=cm, dp=0);
%split(set=fscn, num=9, var=fscn, stat=FTND score, dp=0);

/*End of macro for continuous variables*/

/*Combine all the dataets which have been made from the macros and make any formatting
changes*/

data combined&period.;

set all2 all3 all4 all5 all6 all602 all7 all8 all9 all902 all10 all11 all12 all13;

if (order=9) and index(var,'()') then var=tranwrd(var,'(',')');

else if order=902 then order=9.2;

else if order=602 then order=6.2;

run;

```

```

data combined&period.;

set combined&period.;

        if order in (6.2) and var ne ' ' and stat ='' then delete; *delete the txtra bank line;

        if order = . then delete;

length mcc sa ths opp $8;


array prod [8] t31 t32 t41 t42 t51 t52 t91 t92;

do i=1 to 8;

    if not missing(stat) and missing(prod[i]) then prod[i]='0';

end;


        period=&period;


if period =1 then do; ths="&N1THS"; mcc="&N1MCC"; sa="&N1SAA"; opp="&N1OPP"; end;
if period =2 then do; ths="&N2THS"; mcc="&N2MCC"; sa="&N2SAA"; opp="&N2OPP"; end;
if period =3 then do; ths="&N3THS"; mcc="&N3MCC"; sa="&N3SAA"; opp="&N3OPP"; end;
if period =4 then do; ths="&N4THS"; mcc="&N4MCC"; sa="&N4SAA"; opp="&N4OPP"; end;

run;


proc sort data=combined&period.;

    by order order2 sort;

run;


%mend demog;

```

```
%demog(period=1);
```

```
%demog(period=2);
```

```
%demog(period=3);
```

```
%demog(period=4);
```

```
data combined_all;
```

```
set combined1 combined2 combined3 combined4;
```

```
if var="Missing" and t31="0" and t32="0" and t41="0" and t42='0' and t51="0"  
and t52="0" and t91="0" and t92='0' then delete;
```

```
run;
```

```
proc sql noprint;
```

```
create table tflds.&tflno as
```

```
select period, var, stat, t31 as SA_M, t32 as SA_F, t41 as THSm2_2_M, t42 as THSm2_2_F, t51 as  
mCC_M, t52 as mCC_F,
```

```
t91 as Overall_pp_set_M, t92 as Overall_pp_set_F
```

```
from combined_all
```

```
order by period, order, order2, sort;
```

```
quit;
```

```
data combined_all;
```

```
set combined_all;
```

```
if stat = 'n (%)' then var = "$R'\li360 '" || trim(left(var));
```

```
run;
```

```
proc sort data=combined_all nodupkey;
```

```
by period order order2 sort;  
run;
```

```
data paging;
```

```
set combined_all;
```

```
by period order order2 sort;
```

```
if period =1 then do;
```

```
if order in (2) then page=1;
```

```
else if order in (3) then page =2;
```

```
else if order in (4 5) then page =3;
```

```
else if order in (6 6.2) then page =4;
```

```
else if order in (7 8) then page =5;
```

```
else if order in (9 9.2) then page =6;
```

```
else if order in (10) then page =7;
```

```
else if order in (11 ) then page =8;
```

```
else if order in (12) then page =9;
```

```
else if order in (13) then page =10;
```

```
end;
```

```
if period =2 then do;
```

```
if order in (2) then page=11;
```

```
else if order in (3) then page =12;
```

```
else if order in (4 5) then page =13;
```

```
else if order in (6 6.2) then page =14;
```

```
else if order in (7 8) then page =15;
        else if order in (9 9.2) then page =16;
else if order in (10) then page =17;
        else if order in (11 ) then page =18;
else if order in (12) then page =19;
else if order in (13) then page =20;
end;
```

```
        if period =3 then do;
            if order in (2) then page=21;
                else if order in (3) then page =22;
else if order in (4 5) then page =23;
                else if order in (6 6.2) then page =24;
else if order in (7 8) then page =25;
                else if order in (9 9.2) then page =26;
else if order in (10) then page =27;
                else if order in (11 ) then page =28;
else if order in (12) then page =29;
else if order in (13) then page =30;
end;
```

```
        if period =4 then do;
            if order in (2) then page=31;
                else if order in (3) then page =32;
else if order in (4 5) then page =33;
```

```

        else if order in (6 6.2) then page =34;

else if order in (7 8) then page =35;

        else if order in (9 9.2) then page =36;

else if order in (10) then page =37;

        else if order in (11 ) then page =38;

else if order in (12) then page =39;

else if order in (13) then page =40;

end;

call symput("page",compress(put(page,best.)));

    flag=1;

run;

* Output table*;

options number nodate orientation=landscape papersize=Letter /*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.t106343 (read) ;

```



```

ods results off;

ods rtf toc_data/* contents*/
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;

ods proclabel = ' ';

data comp;

    set paging end=eof;

    where page=&i;

    /* Amend title as needed */

    _firtitl="Table 15.2.1.4.3.1 Summary of Demographics and Other Baseline Characteristics by
Sex - PP Set";

    _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,best8.)));

        call symput('perid', strip(put(period, best8.)));

        call symput('N3', strip(sa));

        call symput('N4', strip(thss));

```

```

        call symput('N5', strip(mcc));

        call symput('N6', strip(opp));

    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;

ods listing close;

proc report data = comp headline headskip nowd split = '#' /*ps = 60 ls = 120*/%if &i=1 %then %do;
contents=' ' %end; %else %do; contents="" %end;;

        column flag period page order var stat

("THSm2.2#(N=&N4)#&linebot" t41 t42) ("mCC#(N=&N5)#&linebot" t51 t52)

("SA#(N=&N3)#&linebot" t31 t32) ("Overall#PP Set#(N=&N6)#&linebot" t91 t92);

        define flag      / order order = internal noprint;

        define page      / order order = internal noprint;

define period    / order order = internal noprint;

        define order     / order order = internal noprint;


define var      /"$R"\li360\fi-360 Variable" display style={just=l cellwidth=3.0cm}
style(header)={just=left};

define stat     /"Statistic" display style={just=l cellwidth=1.1cm} style(header)={just=l} ;

```

```

define t31      /"Male" display style={just=c cellwidth=1.15cm} style(header)={just=center} ;
define t32      /"Female" display style={just=c cellwidth=1.15cm} style(header)={just=center};
define t41      /"Male" display style={JUST=c cellwidth=1.15cm}
style(header)={just=center};
define t42      /"Female" display style={just=c cellwidth=1.15cm} style(header)={just=center};
define t51      /"Male" display style={JUST=c cellwidth=1.15cm} style(header)={just=center};
define t52      /"Female" display style={just=c cellwidth=1.15cm} style(header)={just=center};
define t91      /"Male" display style={JUST=c cellwidth=1.15cm} style(header)={just=center};
define t92      /"Female" display style={just=c cellwidth=1.15cm} 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 after order;

      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 " ";

line "Product Use Time Period: Period &perid.";

line "&linebot";

endcomp;

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

line "Note: Periods defined as Period 1 ([Day 1 - Day 6 confinement]), Period 2 ([Day 6 ambulatory - Day
30 Visit]), Period 3 ([Day 30 Visit - Day 60 Visit]) and Period 4 ([Day 60 Visit - Day 90 Visit].";

line "Note: Height at Screening and body weight and waist circumference at Admission (Day -2) are
used.";

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

line 'Note: BMI = Body Mass Index; FTND = Fagerstrom Test for Nicotine Dependence,
SES = Socio-Economic Status.';

line ' ';

line "Appendix 15.3.1.5, 15.3.1.7, 15.3.1.11, 15.3.6.12, and 15.3.6.13";

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, halfblank=N);
```

```
options noreplace;
```

```
ods listing close;
```

```
proc printto ; run;
```

```
%m_logchk;
```

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

```
* END OF PROGRAM CODE ;
```

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