

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

```
| Covance Study Number   : 000000106343          |
```

```
| Program Name           : t_biomark_fas.sas      |
```

```
| Purpose                 :                      |
```

```
| Input Data              : ADAM.ADSL, ADAM.ADBX  |
|                          |
```

```
| Output Data             :                      |
```

```
| Macros Called           : %m_printto, %m_logchk |
```

```
| Originally Performed by : Deepthi Pippalla     |
```

```
| Date                    : 28APR2015            |
```

```
|                          |
```

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

```
| Modification History    |
```

```
|-----|
```

```
| Modified by            :                      |
```

```
| Modification Date      :
|                          |
```

```
| Modification Description :                      |
```

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

```
options replace;
```

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

```
run;
```

```
%let pgname=t_biomark.sas;
```

```

%let table=&tflout.;

%let TFL_Part=%scan(&_SASPROGRAMFILE,-3,%str(/));

%let TFLprg=t_biomar;


/* 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", ""));

run;


%put &tflpath;

%m_printto;


proc sql;


select count(distinct usubjid) into: NTHS from adam.adsl(where=(trt01an = 4 and FASFL = "Y"));

select count(distinct usubjid) into: NMCC from adam.adsl(where=(trt01an = 5 and FASFL = "Y"));

select count(distinct usubjid) into: NSAA from adam.adsl(where=(trt01an = 3 and FASFL = "Y"));


quit;

```

```

%macro table(where = , tfl = , title1 = );

%let pgrname=t_biomark.sas;

%let table=&tflout.;

%let tflno=&tfl.;

%let TFL_Part=%scan(&_SASPROGRAMFILE,-3,%str(/));

%let TFLprg=t_biomark;


/* 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", ""));

run;


%put &tflpath;


data adbx_1;

set adam.adbx;

WHERE FASFL = "Y";

if &where. and anl02fl = "Y" ;

run;

```

```

data adbx_2;

set adam.adbx;

atptn = .;

atpt = " ";

if &where. and anl02fl = "Y";

if ablfl = "Y" and FASFL = "Y" then do;

avisitn = 10;

avisit = "Baseline";

apuper = 1;

apuperc = "Period 1";

output;

end;

run;

```

```

data adbx;

set adbx_1 adbx_2;

if avisitn = 10 or avisitn > 100;

    if param="3-HPMA (µg)" then param="3-HPMA ("||'B5'X'||"g)";

    else if param="CEMA (µg)" then param="CEMA ("||'B5'X'||"g)";

    else if param="HMPMA (µg)" then param="HMPMA ("||'B5'X'||"g)";

run;

```

```

proc sort data=adbx;

by trtpn paramn param avalu apuper apuperc avisitn avisit atptn atpt;

```

```
run;
```

```
data n;
```

```
length txt $200.;
```

```
set adbx;
```

```
where aqlfl = "Y";
```

```
txtn = 3;
```

```
txt = "BLOQ, n(%)";
```

```
run;
```

```
proc sort data=n out=n1 nodupkey dupout=dup;
```

```
by usubjid paramn avalu avisitn txtn ;
```

```
run;
```

```
proc freq data=n1 noprint;
```

```
tables paramn*param*avalu*apuper*apuperc*avisitn*avisit*atptn*atpt*txtn*txt*trtpn/out=n_freq;
```

```
run;
```

```
data n_freq;
```

```
length countx $50.;
```

```
set n_freq;
```

```
countx = strip(put(count, best.));
```

```
run;
```

```
proc sort data=n_freq;
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;  
run;
```

```
proc transpose data=n_freq out=n_freq_t prefix=trt_;  
by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;  
var countx;  
id trtpn;  
run;
```

```
data n_freq_t;  
set n_freq_t;  
where paramn ne .;  
run;
```

```
proc sort data=adbx;  
by trtpn paramn param avalu avalu apuper apuperc avisitn avisit atptn atpt ;  
run;
```

```
proc means data=adbx(where=(aval ne .)) noprint;  
var aval;  
by trtpn paramn param avalu apuper apuperc avisitn avisit atptn atpt ;  
output out=aval n =n mean = mean std = std median = median min = min max = max q1 = q1 q3 = q3 lclm  
= lclm uclm = uclm;  
run;
```

```

data aval1;

set aval;

if lclm ne . then lclmx = 0.01*floor(100*lclm);

if uclm ne . then uclmx = 0.01*ceil(100*uclm);

length median1 Q2575 Minmax Meansd CIAM n1 $50.;

if median ne . then median1 = strip(put(round(median, 0.01), 15.2));

q2575 = strip(put(round(q1, 0.01), 15.2))||", "||strip(put(round(q3, 0.01), 15.2));

minmax = strip(put(round(min, 0.1), 15.1))||", "||strip(put(round(max, 0.1), 15.1));

if std ne . then meansd = strip(put(round(mean, 0.01), 15.2))||" ("||strip(put(ceil(std*1000)*0.001,
16.3))||")";

else meansd = strip(put(round(mean, 0.01), 15.2))||" (NA)";

if nmiss(lclm, uclm) = 0 then ciam = strip(put(lclmx, 15.2))||", "||strip(put(uclmx, 15.2));

else if lclm = . and uclm ne . then ciam = "NA, "||strip(put(uclmx, 15.2));

else if lclm ne . and uclm = . then ciam = strip(put(lclmx, 15.2))||", NA";

else if lclm = . and uclm = . then ciam = "NA, NA";

n1 = strip(put(n, best.));

if trtpn = 3 then Num = &NSAA;

if trtpn = 4 then Num = &NTHS;

if trtpn = 5 then Num = &NMCC;

missing = Num - n;

if missing ne 0 then miss = strip(put(missing, best.))||"
("||strip(put(round((missing/Num)*100,0.1),10.1))||")";

run;

```

```
proc sort data=aval1;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;

run;
```

```
proc transpose data=aval1 out=aval_t prefix= trt_;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;

var n1 median1 q2575 minmax meansd ciam MISS;

id trtpn;

run;
```

```
data aval_t;

length txt $200.;

set aval_t;

if upcase(_name_) = "N1" then do;

txtn = 1;

txt = "n";

end;

if upcase(_name_) = "MISS" then do;

txtn = 2;

txt = "Missing, n (%)";

end;

else if upcase(_name_) = "MEDIAN1" then do;

txtn = 7;

txt = "Median";
```



```
end;  
  
else if upcase(_name_) = "Q2575" then do;  
  
    txtn = 8;  
  
    txt = "Q25, Q75";  
  
end;  
  
else if upcase(_name_) = "MINMAX" then do;  
  
    txtn = 9;  
  
    txt = "Min, Max";  
  
end;  
  
else if upcase(_name_) = "MEANS" then do;  
  
    txtn = 10;  
  
    txt = "Mean (SD)";  
  
end;  
  
else if upcase(_name_) = "CIAM" then do;  
  
    txtn = 11;  
  
    txt = "95% CI of Mean";  
  
end;  
  
run;
```

```
data adbx_log;  
  
set adbx;  
  
if aval ne . then logaval = log(aval);  
  
run;
```

```
proc sort data=adbx_log;
```

```
by trtpn trtp paramn param paramcd avalu apuper apuperc avisitn avisit atptn atpt ;  
run;
```

```
proc means data=adbx_log noprint;  
by trtpn trtp paramn param paramcd avalu apuper apuperc avisitn avisit atptn atpt ;  
output out=aval_log mean = mean std = std lclm = lclm uclm = uclm;  
var logaval;  
run;
```

```
data aval_log1;  
set aval_log;  
length geocv CIGM $50.;  
if lclm ne . then lclm1 = exp(lclm);  
if uclm ne . then uclm1 = exp(uclm);  
if lclm ne . then lclmx = 0.01*floor(lclm1 * 100);  
if uclm ne . then uclmx = 0.01*ceil(uclm1 * 100);  
if std ne . then cv_pct=100*sqrt(exp(std*std)-1) ;  
if mean ne . then mean1 = exp(mean);  
MEAN2 = ROUND(MEAN1, 0.01);  
if cv_pct ne . then geocv = strip(put(round(mean1,0.01), 15.2))|| "  
("||strip(put(ceil(cv_pct*1000)*0.001, 16.3))|| ")";  
else geocv = strip(put(round(mean1,0.01), 15.2))|| " (NA)";  
if nmiss(lclm, uclm) = 0 then CIGM = strip(put(lclmx, 15.2))|| ", "||strip(put(uclmx, 15.2));  
else if lclm = . and uclm ne . then CIGM = "NA, "||strip(put(uclmx, 15.2));  
else if lclm ne . and uclm = . then CIGM = strip(put(lclmx, 15.2))|| ", NA";  
else if lclm = . and uclm = . then CIGM = "NA, NA";
```

```
run;
```

```
proc sort data=aval_log1 ;
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt;
```

```
run;
```

```
%let tfl=&tfl.;
```

```
data tflds.&tfl._f;*(keep=txt txtn avalu avisitn avisit paramn param trt_3 trt_4 trt_5 chg_3 chg_4 chg_5);
```

```
set aval_log1(keep=param paramn paramcd avalu trtpn trtp avisitn avisit atptn atpt apuper apuperc  
mean2 lclmx uclmx);
```

```
rename mean2 = mean
```

```
lclmx = lclm
```

```
uclmx = uclm;
```

```
run;
```

```
proc transpose data=aval_log1 out=aval_log1_t prefix= trt_;
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;
```

```
var geocv cigm;
```

```
id trtpn;
```

```
run;
```

```
data aval_log1_t;
```

```
length txt $200.;
```

```
set aval_log1_t;
```

```
if upcase(_name_) = "GEOCV" then do;
    txtn = 5;
    txt = "Geometric Mean (CV%)";
    end;
else if upcase(_name_) = "CIGM" then do;
    txtn = 6;
    txt = "95% CI of Geometric Mean";
    end;
run;
```

```
data aval_f;
    set aval_t n_freq_t aval_log1_t;
run;
```

```
proc sort data=aval_f;
    by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;
run;
```

```
proc sort data=adbx;
    by trtpn paramn param avalu apuper apuperc avisitn avisit atptn atpt ;
run;
```

```
data n_chg;
    length txt $200.;
```

```

set adbx;

where pchg = .;

txtn = 2;

txt = "Missing, n(%)";

output;

run;


proc sort data=n_chg out=n1_chg nodupkey dupout=dup;

by usubjid paramn avalu avisitn txtn ;

run;


proc freq data=n1 noprint;

tables
paramn*param*avalu*apuper*apuperc*avisitn*avisit*txtn*txt*trtpn*atptn*atpt/out=n_chg_freq;

run;


data n_chg_freq;

length countx $50.;

set n_chg_freq;

countx = strip(put(count, best.));

run;


proc sort data=n_chg_freq;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;

run;

```

```

proc transpose data=n_chg_freq out=n_chg_freq_t prefix=chg_;
by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;
var countx;
id trtpn;
run;

```

```

data n_chg_freq_t;
set n_chg_freq_t;
where paramn ne .;
run;

```

```

proc means data=adbx(where=(pchg ne . and avisitn ne 10)) noprint;
var pchg;
by trtpn paramn param avalu apuper apuperc avisitn avisit atptn atpt;
output out=chg n =n mean = mean std = std median = median min = min max = max q1 = q1 q3 = q3 lclm
= lclm uclm = uclm;
run;

```

```

data chg1;
set chg;
if lclm ne . then lclmx = 0.01*floor(100*lclm);
if uclm ne . then uclmx = 0.01*ceil(100*uclm);
length median1 Q2575 Minmax Meansd CIAM n1 $50.;
if median ne . then median1 = strip(put(round(median, 0.01), 15.2));
q2575 = strip(put(round(q1, 0.01), 15.2)) || ", " || strip(put(round(q3, 0.01), 15.2));

```

```

minmax = strip(put(round(min, 0.1), 15.1)) || ", " || strip(put(round(max, 0.1), 15.1));

if std ne . then meansd = strip(put(round(mean, 0.01), 15.2)) || " (" || strip(put(ceil(std*1000)*0.001,
16.3)) || ")";

else meansd = strip(put(round(mean, 0.01), 15.2)) || " (NA)";

if nmiss(lclm, uclm) = 0 then ciam = strip(put(lclmx, 15.2)) || ", " || strip(put(uclmx, 15.2));

else if lclm = . and uclm ne . then ciam = "NA, " || strip(put(uclmx, 15.2));

else if lclm ne . and uclm = . then ciam = strip(put(lclmx, 15.2)) || ", NA";

else if lclm = . and uclm = . then ciam = "NA, NA";

n1 = strip(put(n, best.));

if trtpn = 3 then Num = &NSAA;

if trtpn = 4 then Num = &NTHS;

if trtpn = 5 then Num = &NMCC;


missing = Num - n;

if missing ne 0 then miss = strip(put(missing, best.)) || "
(" || strip(put(round((missing/Num)*100,0.1),10.1)) || ")";

run;


proc sort data=chg1;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;

run;


proc transpose data=chg1 out=chg_t prefix= chg_;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;

var n1 median1 q2575 minmax meansd ciam miss;

```

```
id trtpn;

run;

data chg_t;

length txt $200.;

set chg_t;

if upcase(_name_) = "N1" then do;

txtn = 1;

txt = "n";

end;

if upcase(_name_) = "MISS" then do;

txtn = 2;

txt = "Missing, n (%)";

end;

else if upcase(_name_) = "MEDIAN1" then do;

txtn = 7;

txt = "Median";

end;

else if upcase(_name_) = "Q2575" then do;

txtn = 8;

txt = "Q25, Q75";

end;

else if upcase(_name_) = "MINMAX" then do;

txtn = 9;

txt = "Min, Max";
```



```

end;

else if upcase(_name_) = "MEANS" then do;

txtn = 10;

txt = "Mean (SD)";

end;

else if upcase(_name_) = "CIAM" then do;

txtn = 11;

txt = "95% CI of Mean";

end;

run;


data chg_f;

set chg_t ;

run;


proc sort data=chg_f;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;

run;


proc sort data=aval_f;

by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;

run;


DATA AVAL_F_BLOQ;

SET AVAL_F;

```

```
WHERE TXTN = 1;
```

```
IF TRT_3 NE " " THEN BLOQ_3 = INPUT(TRT_3, BEST.);
```

```
IF TRT_4 NE " " THEN BLOQ_4 = INPUT(TRT_4, BEST.);
```

```
IF TRT_5 NE " " THEN BLOQ_5 = INPUT(TRT_5, BEST.);
```

```
RUN;
```

```
PROC SORT DATA=AVAL_F_BLOQ;
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;
```

```
RUN;
```

```
data final_F;
```

```
merge aval_f chg_f(drop=_name_);
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt txtn txt;
```

```
RUN;
```

```
DATA FINAL;
```

```
MERGE FINAL_F(IN=A) AVAL_F_BLOQ(IN=B KEEP=paramn param avalu apuper apuperc avisitn avisit  
atptn atpt BLOQ_3 BLOQ_4 BLOQ_5);
```

```
by paramn param avalu apuper apuperc avisitn avisit atptn atpt ;
```

```
THS = &NTHS;
```

```
mcc = &Nmcc;
```

```
sa = &Nsaa;
```

```
if txtn =2 and cmiss(trt_3, trt_4, trt_5, chg_3, chg_4, chg_5) = 6 then delete;
```

```

if txtn in (3,4) then do;

if trt_3 ne " " then num_3 = input(trt_3, best.);

if trt_4 ne " " then num_4 = input(trt_4, best.);

if trt_5 ne " " then num_5 = input(trt_5, best.);

if num_3 ne . then pp_3 = strip(put(round((num_3/BLOQ_3)* 100, 0.1), 15.1));

if num_4 ne . then pp_4 = strip(put(round((num_4/BLOQ_4)*100, 0.1), 15.1));

if num_5 ne . then pp_5 = strip(put(round((num_5/BLOQ_5)*100, 0.1), 15.1));

trt_3 = strip(trt_3)||" ("||strip(pp_3)||")";

trt_4 = strip(trt_4)||" ("||strip(pp_4)||")";

trt_5 = strip(trt_5)||" ("||strip(pp_5)||")";

```

```

end;

```

```

if strip(trt_3) = "(" then trt_3 = " ";

if strip(trt_4) = "(" then trt_4 = " ";

if strip(trt_5) = "(" then trt_5 = " ";

if strip(chg_3) = "(" then chg_3 = " ";

if strip(chg_4) = "(" then chg_4 = " ";

if strip(chg_5) = "(" then chg_5 = " ";

```

```

run;

```

```

proc sql;

```

```
create table page as  
select distinct apuper, apuperc, paramn, avisitn,avalu  
from final  
order by paramn, avalu,apuper, avisitn;  
quit;
```

```
data page1_;  
set page;  
by paramn avalu apuper avisitn;  
if index(upcase(avalu), "CREAT") > 0 then pagord = 1;  
else pagord = 2;  
run;
```

```
proc sort data=page1_;  
by pagord paramn avalu apuper avisitn;  
run;
```

```
data page1;  
set page1_;  
by pagord paramn avalu apuper avisitn;  
if _n_ = 0 then page = 0;  
page+ 1;  
run;
```

```
data pagex;  
set page1;  
by pagord paramn avalu apuper avisitn;  
if last.paramn;  
run;
```

```
data pagex1;  
set pagex;  
if _n_ = 1 then call symputx("pagex", page);  
if _n_ = 2 then call symputx("paramnx", paramn);  
run;
```

```
%put &pagex &paramnx;
```

```
proc sql;  
create table final_page as  
select distinct a.*, b.page, pagord  
from final as a  
left join page1 as b  
on a.paramn = b.paramn and a.avalu = b.avalu and a.avisitn = b.avisitn and a.apuper = b.apuper  
order by pagord, paramn,avalu,apuper,avisitn, txtn;  
quit;
```

```
data final_page;  
set final_page end=last;
```

```

by pagord paramn avalu apuper avisitn txtn;

if last then call symputx("page", page);

if paramn = &paramnx then do;

chg_3 = " ";

chg_4 = " ";

chg_5 = " ";

end;

if upcase(_name_) = "MISS" then do;

if trt_3 = " " and trt_4 = " " and trt_5 = " " and chg_3 = " " and chg_4 = " " and chg_5 = " " then delete;

end;

IF TXTN IN (2,3) THEN DO;

IF TRT_3 = " " THEN TRT_3 = "0";

IF TRT_4 = " " THEN TRT_4 = "0";

IF TRT_5 = " " THEN TRT_5 = "0";

END;

run;

data tflds.&tflno(keep=txt txtn avalu avisitn avisit paramn param trt_3 trt_4 trt_5 chg_3 chg_4 chg_5
apuper );

set final_page;

run;

%put &page;

```

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

```
options number nodate orientation=landscape /* papersize=&P_PGSIZE*/ 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;
```

```
/* 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/dev/Tables/&tflno..rtf" style=t106343  
startpage=yes headery=1440 footery=1440 ;
```

```
ods noproctitle;
```

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

```
title ;
```

```

footnote;

%let wd=0;

ods proclabel = ' ';


data comp;

    set final_page end=eof;

        where page=&i;


        /* Amend title as needed */

            _firtitl="&title1.";

        _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.)));

                call symput('param', strip(param));

                call symput('N3', strip(put(sa, best.)));

                call symput('N4', strip(put(th, best.)));

                call symput('N5', strip(put(mcc, 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 paramn avisitn avisit txtn txt ("THSm2.2$(N=&N4)$&linebot" trt_4 chg_4 )
("mCC$(N=&N5)$&linebot" trt_5 chg_5)
```

```
("SA$(N=&N3)$&linebot" trt_3 chg_3);
```

```
define paramn / order order = internal noprint;
```

```
define page / order order = internal noprint;
```

```
define avisitn / order order = internal noprint;
```

```
define txtn / order order = internal noprint;
```

```
define avisit /"Timepoint" order order=internal style={just=left cellwidth=0.9cm}
style(header)={just=left} ;
```

```
define txt /"Statistic" display style={just=left cellwidth=1.9cm}
style(header)={just=left} ;
```

```
define trt_3 /"Value" display style={JUST=c cellwidth=1.2cm}
style(header)={just=center} ;
```

```
define trt_4 /"Value" display style={just=c cellwidth=1.2cm}
style(header)={just=center} ;
```

```

        define trt_5      /"Value" display style={just=c cellwidth=1.2cm}
style(header)={just=center};

        define chg_3      /"% Change(*)" display style={JUST=c cellwidth=1cm}
style(header)={just=center};

        define chg_4      /"% Change(*)" display style={just=c cellwidth=1cm}
style(header)={just=center};

        define chg_5      /"% Change(*)" display style={just=c cellwidth=1cm}
style(header)={just=center};

```

```

compute after avisitn;

```

```

    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 " ";

```

```

    line "Parameter (units): &param";

```

```

    line "&linebot";

```

```

endcomp;

```

```

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

```

```

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

```

```

    LINE 'Note: * % Change from baseline, where baseline is defined as the last assessment
prior to first randomized product use in mCC / THS 2.2 Menthol arms or the last assessment prior to
10AM on Day 1 in the SA arm.';

```

LINE "Note: 'Missing' percentages are based on the number of subjects indicated in the column header (N), while 'BLOQ' percentages are based on the number of subjects being summarized (n).";

LINE ' ';

line 'Appendix 15.3.3.1';

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

endcomp;

run;

%end;

%let j = %eval(&pagex + 1);

%do i=&j %to &page;

%let wd=0;

ods proclabel = ' ';

data comp;

set final\_page end=eof;

where page=&i;

/\* Amend title as needed \*/

\_firtitl="&title1.";

\_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.)));

    call symput('param', strip(param));

    call symput('N3', strip(put(sa, best.)));

    call symput('N4', strip(put(ths, best.)));

    call symput('N5', strip(put(mcc, best.)));

end;

drop _firtitl _upcas len;

run;

```

ods listing close;

title ;

footnote;

\* 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 paramn avisitn avisit txtn txt ("THSm2.2$(N=&N4)$&linebot" trt_4 )
("mCC$(N=&N5)$&linebot" trt_5 )
```

```
("SA$(N=&N3)$&linebot" trt_3 );
```

```
define paramn / order order = internal noprint;
```

```
define page / order order = internal noprint;
```

```
define avisitn / order order = internal noprint;
```

```
define txtn / order order = internal noprint;
```

```
define avisit /"Timepoint" order order=internal style={just=left cellwidth=0.9cm}
style(header)={just=left} ;
```

```
define txt /"Statistic" display style={just=left cellwidth=1.9cm}
style(header)={just=left} ;
```

```
define trt_3 /"Value" display style={JUST=c cellwidth=1.2cm}
style(header)={just=center} ;
```

```
define trt_4 /"Value" display style={just=c cellwidth=1.2cm}
style(header)={just=center} ;
```

```
define trt_5 /"Value" display style={just=c cellwidth=1.2cm}
style(header)={just=center};
```

```
compute after avisitn;
```

```
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 " ";
```

```
line "Parameter (units): &param";
```

```
line "&linebot";
```

```
endcomp;
```

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

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

```
LINE 'Note: * % Change from baseline, where baseline is defined as the last assessment  
prior to first randomized product use in mCC / THS 2.2 Menthol arms or the last assessment prior to  
10AM on Day 1 in the SA arm.';
```

```
LINE "Note: 'Missing' percentages are based on the number of subjects indicated in the  
column header (N), while 'BLOQ' percentages are based on the number of subjects being summarized  
(n).";
```

```
LINE ' ';
```

```
line 'Appendix 15.3.3.1';
```

```
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 ;
```

```
%outtrtf(blankn=36, halfblank=N);
```

```
ods listing;
```

```
%mend;
```

```

%table(where = %str(paramcd in ("UNEQ24U" "UNEQCRE")), tfl = %str(T_15_02_04_18_02), title1 =
%str(Table 15.2.4.18.2 Descriptive Statistics of NEQ in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("USBMA24U" "USBMACRE")), tfl = %str(T_15_02_04_17_02), title1 =
%str(Table 15.2.4.17.2 Descriptive Statistics of S-BMA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UHMPMCRE" "UHMPM24U")), tfl = %str(T_15_02_04_16_02), title1 =
%str(Table 15.2.4.16.2 Descriptive Statistics of HMPMA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UBAPCRE" "UBAP24U")), tfl = %str(T_15_02_04_15_02), title1 =
%str(Table 15.2.4.15.2 Descriptive Statistics of B[a]P in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UHEMACRE" "UHEMA24U")), tfl = %str(T_15_02_04_14_02), title1 =
%str(Table 15.2.4.14.2 Descriptive Statistics of HEMA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UCEMACRE" "UCEMA24U")), tfl = %str(T_15_02_04_13_02), title1 =
%str(Table 15.2.4.13.2 Descriptive Statistics of CEMA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UOTOLCRE" "UOTOL24U")), tfl = %str(T_15_02_04_12_02), title1 =
%str(Table 15.2.4.12.2 Descriptive Statistics of o-tol in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("U2NACRE" "U2NA24U")), tfl = %str(T_15_02_04_11_02), title1 =
%str(Table 15.2.4.11.2 Descriptive Statistics of 2-NA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("U1NACRE" "U1NA24U")), tfl = %str(T_15_02_04_10_02), title1 =
%str(Table 15.2.4.10.2 Descriptive Statistics of 1-NA in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("U4ABPCRE" "U4ABP24U")), tfl = %str(T_15_02_04_09_02), title1 =
%str(Table 15.2.4.9.2 Descriptive Statistics of 4-ABP in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("UNNNCRE" "UNNN24U")), tfl = %str(T_15_02_04_08_02), title1 =
%str(Table 15.2.4.8.2 Descriptive Statistics of Total NNN in 24-hour Urine Collection - FAS));

%table(where = %str(paramcd in ("U1OHPCRE" "U1OHP24U")), tfl = %str(T_15_02_04_07_02), title1 =
%str(Table 15.2.4.7.2 Descriptive Statistics of 1-OHP in 24-hour Urine Collection - FAS));

%m_logchk;

```