

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

```
| Covance Study Number   : 000000106343          |
| Program Name           : t_icam1_fas.sas        |
| Purpose                 : Program to table T_15_2_4_30_2 |
| Input Data              : ADAM.ADSL, ADAM.adlb    |
|                          |                        |
| Output Data             : T_15_02_04_30_02       |
| Macros Called           :                        |
| Originally Performed by :Sree Bikki             |
| Date                    : 12MAY2015              |
|                          |                        |
```

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

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

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

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

```
run;
```

```
%m_printto;
```

```

proc sql;

select count(distinct usubjid) into: N1THS from adam.adsl(where=(trt01pn = 4 and fasfl = "Y"));

select count(distinct usubjid) into: N1MCC from adam.adsl(where=(trt01pn = 5 and fasfl = "Y"));

select count(distinct usubjid) into: N1SAA from adam.adsl(where=(trt01pn = 3 and fasfl = "Y"));

quit;


%let tflno=T_15_02_04_30_02;


/* Standard - leave this */

%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", ""));

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

run;


data adlb_1;

set adam.adlb;

where fasfl = "Y";

if paramcd in ("ICAM1") and anl01fl = "Y";

run;

```

```
data adlb_2;  
set adam.adlb;  
if paramcd in ("ICAM1") and anl01fl = "Y";  
if ablfl = "Y" and fasfl = "Y";  
avisitn = 10;  
avisit = "Baseline";  
run;
```

```
data adlb;  
set adlb_1 adlb_2;  
if avisit ne 'Baseline' and avisitn lt 101 then delete;  
run;
```

```
proc sort data=adlb;  
by trtpn trtp paramn param paramcd avisitn avisit atptn atpt;  
run;
```

```
/*GEOMEAN AN CI*/
```

```
data adlb_log;  
set adlb;  
if aval ne . then logaval = log(aval);  
run;
```

```
proc sort data=adlb_log;

by trtpn trtp paramn param paramcd avisitn avisit ;

run;
```

```
proc means data=adlb_log noprint;

by trtpn trtp paramn param paramcd avisitn avisit ;

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.;

    gmean1=exp(mean);

    glci=exp(lclm);

    guci=exp(uclm);

    gmean=left(compress(put(round(gmean1,0.1), 8.1)));

    gcv=compress(put(0.01*ceil((sqrt(exp(std*std)-1)*100)/0.01),8.2));

        if not missing(gcv) then geocv=left(trim(gmean)) || ' (' || left(trim(gcv))||')';

    else geocv=left(trim(gmean));

        if not missing(glci) and not missing(guci) then cigm = strip(strip(put(0.1*floor(glci/0.1),8.1)) || ' ', '
|| strip(put(0.1*ceil(guci/0.1),8.1)));
```

```
else if glci ne . and guci = . then cigm = strip(put(0.1*floor(glci/0.1), 8.1))||", NA";  
else if glci = . and guci ne . then cigm = "NA, "||strip(put(0.1*ceil(guci/0.1),8.1));  
else if glci = . and guci = . then cigm = "NA, NA";  
  
run;
```

```
/*for figure dataset t_15_02_04_30_02_F*/
```

```
data figure;  
  
set aval_log1;  
  
drop std _type_ _freq_;  
  
logf=1;  
  
mean = gmean1;  
  
lclm = glci;  
  
uclm = guci;  
  
keep avisit avisitn paramcd param paramn trtp trtpn mean lclm uclm logf;  
  
run;
```

```
/*end for figure dataset t_15_02_04_30_02_F*/
```

```
proc sort data=aval_log1 ;  
  
by paramn param avisitn avisit ;
```

```
run;
```

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

```
by paramn param avisitn avisit ;
```

```
var geocv cigm;
```

```
id trtpn;
```

```
run;
```

```
/*Q25 Q75 MEdian*/
```

```
proc means data=adlb(where=(aval ne .)) noprint;
```

```
var aval;
```

```
by trtpn paramn param avisitn avisit ;
```

```
output out=aval n =n mean = mean std = std median = median min = min max = max q1 = q1 q3 = q3;
```

```
run;
```

```
data aval1;
```

```
set aval;
```

```
length median1 Q2575 Minmax n1 $50.;
```

```
median1 = strip(put(round(median, 0.01), 15.1));
```

```
q2575 = strip(put(round(q1, 0.01), 15.1))||", "||strip(put(round(q3, 0.01), 15.1));
```

```
minmax = strip(put(round(min, 0.1), 15.0))||", "||strip(put(round(max, 0.1), 15.0));
```

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

```
run;
```

```
data results03;
```

```
length missc $30;
```

```
set aval1;
```

```
if trtpn=3 then do;
```

```
if &n1saa.=n then
```

```
missc="";
```

```
else
```

```
missc=strip(put((&n1saa.- n), 8.)) || ' (' || strip(put(((&n1saa.-n)*100)/&n1saa., 8.1)) || ")";
```

```
end;
```

```
else if trtpn=4 then do;
```

```
if &n1ths.=n then
```

```
missc="";
```

```
else
```

```
missc=strip(put((&n1ths.- n), 8.)) || ' (' || strip(put(((&n1ths.-n)*100)/&n1ths., 8.1)) || ")";
```

```
end;
```

```
else if trtpn=5 then do;
```

```
if &n1mcc.=n
```

```
then missc="";
```

```
else
```

```
missc=strip(put((&n1mcc.-n), 8.)) || ' (' || strip(put(((&n1mcc.-n)*100)/&n1mcc., 8.1)) || ")";
```

```
end;
```

```
run;
```

```
proc sort data=results03 out=aval1;
```

```
by paramn param avisitn avisit ;
```

```
run;
```

```
proc transpose data=aval1 out=aval_t prefix= trt_;  
by paramn param avisitn avisit ;  
var n1 median1 q2575 minmax missc;  
id trtpn;  
run;
```

```
data aval_t;  
length txt $200.;  
set aval_t aval_log1_t;  
if upcase(_name_) = "N1" then do;  
txtn = 1;  
txt = "n";  
end;  
if upcase(_name_) = "MISSC" then do;  
txtn = 2;  
txt = "Missing, n (%)";  
end;  
else if upcase(_name_) = "MEDIAN1" then do;  
txtn = 6;  
txt = "Median";  
end;  
else if upcase(_name_) = "Q2575" then do;  
txtn = 7;  
txt = "Q25, Q75";
```



```

end;

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

txtn = 8;

txt = "Min, Max";

end;

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

txtn = 4;

txt = "Geometric Mean (CV%)";

end;

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

txtn = 5;

txt = "95% CI of Geometric Mean";

end;

run;

```

```

*      ALOQ AND BLOOQ */;

```

```

data Stat;

length txt $100.;

set adlb;

if aqlfl="Y" then do;

txtn = 3;

txt = "BLOQ, n (%)";

output;

end;

```

```
run;
```

```
proc sort data=stat out=stat1 nodupkey dupout=dup;
```

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

```
run;
```

```
proc freq data=stat1 noprint;
```

```
tables paramn*param*avisitn*avisit*txtn*txt*trtpn/out=stat_freq;
```

```
run;
```

```
data stat_freq;
```

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

```
set stat_freq;
```

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

```
run;
```

```
proc sort data=stat_freq;
```

```
by paramn param avisitn avisit txtn txt;
```

```
run;
```

```
proc transpose data=stat_freq out=stat_freq_t prefix=trt_;
```

```
by paramn param avisitn avisit txtn txt;
```

```
var countx;
```

```
id trtpn;
```

```
run;
```

```

data dummy;

length trt_3 trt_4 trt_5 $50;

set stat_freq_t(keep= paramn param avisitn avisit txtn txt);

trt_4 = "";

trt_3 = "";

trt_5 = "";

run;

```

```

data stat_freq_t1;

update dummy stat_freq_t;

by paramn param avisitn avisit txtn txt;

run;

```

```

data stat_freq_t2;

set stat_freq_t1(drop=_name_);

where paramn ne .;

run;

```

```

/**code to change denominators for BLOQ records*/

data test;

set aval1;

keep paramn param avisitn avisit trtpn n ;

```

```
run;
```

```
proc sort data=test;
```

```
by paramn param avisitn avisit trtpn ;
```

```
run;
```

```
data testa(rename= (n=r3_n)) testb(rename= (n=r4_n)) testc(rename= (n=r5_n));
```

```
set test;
```

```
if trtpn =3 then output testa;
```

```
if trtpn = 4 then output testb;
```

```
if trtpn=5 then output testc;
```

```
run;
```

```
data test1;
```

```
merge stat_freq_t2(in=a) testa(in=b drop=trtpn) testb(in=c drop=trtpn) testc(in=d drop=trtpn);
```

```
by paramn param avisitn avisit ;
```

```
if a ;
```

```
run;
```

```
/** end of code to change denominators for BLOQ records*/
```

```

data aval_f;

set aval_t test1;

THS = &N1THS;

mcc = &N1mcc;

sa = &N1saa;

if txtn in (3) 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/r3_n)*100, 0.1), 15.1));

if num_4 ne . then pp_4 = strip(put(round((num_4/r4_n)*100, 0.1), 15.1));

if num_5 ne . then pp_5 = strip(put(round((num_5/r5_n)*100, 0.1), 15.1));

if not missing(trt_3) and not missing(pp_3) then trt_3 = strip(trt_3)||" ("||strip(pp_3)||")";

if not missing(trt_4) and not missing(pp_4) then trt_4 = strip(trt_4)||" ("||strip(pp_4)||")";

if not missing(trt_5) and not missing(pp_5) then trt_5 = strip(trt_5)||" ("||strip(pp_5)||")";

end;

drop num_3 num_4 num_5 pp_3 pp_4 pp_5 r3_n r4_n r5_n;

run;

```

```

proc sort data=aval_f;

by paramn param avisitn avisit txtn txt;

run;

```

```
proc sort data=adlb;
```

```
by trtpn paramn param avisitn avisit ;
```

```
run;
```

```
proc means data=adlb(where=(pchg ne . and ablfl ne "Y")) noprint;
```

```
var pchg;
```

```
by trtpn paramn param avisitn avisit ;
```

```
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;
```

```
length geocv CIGM $50.;
```

```
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 n1 $50.;
```

```
median1 = strip(put(round(median, 0.01), 15.1));
```

```
q2575 = strip(put(round(q1, 0.01), 15.1))||", "||strip(put(round(q3, 0.01), 15.1));
```

```
minmax = strip(put(round(min, 0.1), 15.0))||", "||strip(put(round(max, 0.1), 15.0));
```

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

```
geocv = " ";
```

```
cigm = " ";
```

```
run;
```

```
data results04;
```

```
length missc $30;
```

```
set chg1;
```

```
if trtpn=3 then do;
```

```
if &n1saa.=n then
```

```
missc="";
```

```
else
```

```
missc=strip(put((&n1saa.- n), 8.)) || ' (' || strip(put(((&n1saa.-n)*100)/&n1saa., 8.1)) || ")";
```

```
end;
```

```
else if trtpn=4 then do;
```

```
if &n1ths.=n then
```

```
missc="";
```

```
else
```

```
missc=strip(put((&n1ths.- n), 8.)) || ' (' || strip(put(((&n1ths.-n)*100)/&n1ths., 8.1)) || ")";
```

```
end;
```

```
else if trtpn=5 then do;
```

```
if &n1mcc.=n
```

```
then missc="";
```

```
else
```

```
missc=strip(put((&n1mcc.-n), 8.)) || ' (' || strip(put(((&n1mcc.-n)*100)/&n1mcc., 8.1)) || ")";
```

```
end;
```

```
run;
```

```
proc sort data=results04 out=chg1;
```

```
by paramn param avisitn avisit ;
```

```
run;
```

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

```
by paramn param avisitn avisit ;
```

```
var n1 median1 q2575 minmax geocv cigm missc;
```

```
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_) = "MISSC" then do;
```

```
txtn = 2;
```

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

```
end;
```

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

```
txtn = 6;
```

```
txt = "Median";
```

```
end;
```

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



```
txtn = 7;

txt = "Q25, Q75";

end;

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

txtn = 8;

txt = "Min, Max";

end;

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

txtn = 4;

txt = "Geometric Mean (CV%)";

end;

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

txtn = 5;

txt = "95% CI of Geometric Mean";

end;

run;
```

```
data chg_f;

set chg_t ;

run;
```

```
proc sort data=chg_f;

by paramn param avisitn avisit txtn txt;

run;
```

```
proc sort data=aval_f;  
by paramn param avisitn avisit txtn txt;  
run;
```

```
data final;  
merge aval_f chg_f(drop=_name_);  
by paramn param avisitn avisit txtn txt;  
THS = &N1THS;  
mcc = &N1mcc;  
sa = &N1saa;  
if trt_3 = " " and trt_4 = " " and trt_5 = " " and chg_3 = " " and chg_4 = " " and chg_5 = " " then delete;  
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;  
if txtn in (2,3) and avisitn ne 10 then do;  
if chg_3 = " " then chg_3 = "0";  
if chg_4 = " " then chg_4 = "0";  
if chg_5 = " " then chg_5 = "0";  
end;  
run;
```

```
data page1;  
  
set final;  
  
by paramn avisitn;  
  
obs = _n_;  
  
page = ceil(obs/8);  
  
run;
```

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

```
data final_page;  
  
set final_page end=last;  
  
by paramn avisitn txtn;  
  
if last then call symputx("page", page);  
  
run;
```

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

```
set final_page;
```

```
run;
```

```
data tflds.t_15_02_04_30_02_F;
```

```
set figure;
```

```
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, halfblnk=N);
```

```
%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;

%let wd=0;

ods proclabel = ' ';

data comp;

    set final_page end=eof;

        where page=&i;

/* Amend title as needed */

        _firtitl="Table 15.2.4.30.2 Descriptive Statistics of sICAM (ng/mL)- FAS";

_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=2.9cm}
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=1.2cm}
style(header)={just=center};

        define chg_4          /"% Change(*)" display style={just=c cellwidth=1.2cm}
style(header)={just=center};

        define chg_5          /"% Change(*)" 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: '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 '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 ' ';
```

```
line 'Appendix 15.3.3.2';
```

```
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);
```

```
ods listing;
```



%m\_logchk2;