

```

%_mprintto;
%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",""));
run;

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_banlqsu.sas;
%put NOTE: Purpose              : table and listing of bootstrapped qsu
data;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADQSSU DSDTM.QSUBOOT;
%put NOTE: Output               : t_15_2_4_13(QSU) t_15_2_4_13(QSU);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_ahall;
%put NOTE: Creation Date        : 2014-05-27;
%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: 08Aug2014  AMH        1) Ammend Output to format of PK02;
%put NOTE: ;
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing=' '
NOQUOTELNMAX/*turn off warnings about quoted strings to long*/;
ods _all_ close;
ods listing;

%include
"/cvn/projects/prj/development/000000106326/dev/adhoc/TMPLTMIX.sas";

*=====;
* START OF PROGRAM CODE                                     ;
*=====;

```

```

/* Calculate totals for products */
data adsl;
    set adam.adsl(where=(pprotfl='Y'));
    if analgrln=1 then do;
        if index(trt01a,'THS 2.2') or index(trt02a,'THS 2.2') then
colord=1;
        output;
        if index(trt01a,'CC') or index(trt02a,'CC') then colord=2;
        output;
    end;
    else if analgrln=2 then do;
        if index(trt01a,'THS 2.2') or index(trt02a,'THS 2.2') then
colord=1;
        output;
        if index(trt01a,'NRT') or index(trt02a,'NRT') then colord=2;
        output;
    end;
    else if missing(analgrln) then delete;
run;

proc sort data=adsl nodupkey out=adsl1;
    by analgrln analgr1 colord subjid;
run;

proc freq data=adsl1(where=(not missing(colord))) noprint;
    table analgrln*analgr1*colord/ out =totals2(drop=percent
rename=(count=total));
run;

data _null_;
    set totals2;
    call
symput('tot'||strip(put(colord,best.))||strip(put(analgrln,best.)),strip(
put(total,best.)));
run;

*****;
* read in data ;
*****;
proc sort data=adam.adqssu(where = (anl01fl = 'Y' and atptn>0 and
PPROTFL='Y')) out=adqssul;
    by analgr1 paramcd;
run;
/*make adjustments for different timepoints*/
data adqssu;
    set adqssul;
    if analgrln=2 and atptn=6 then atptn=7;
    if analgrln=2 and atptn=7 then atpt='T0 + 15/20 min';
run;

```

```

data diff lsmean seed;
  set dsdtm.qsubboot1;
  ATPTN=.;
  if not missing(seed) then output seed;
  else if not missing(_trtan) then output diff;
  else output lsmean;
run;

%MACRO IMPUTEPARAM(NUM=,PARAM=);

%macro impute(analgr=, dset=);

data &dset.&analgr;
set &dset(where=(analgrln=&analgr AND PARAMCD="&PARAM"));
n=_n_;
run;

data _null_;
  if 0 then set &dset.&analgr nobs=obs;
  nmiss=2000-obs;
  nmiss2=(4000-obs)/2;
  call symput('missobs',compress(put(nmiss,best.)));
  call symput('missobs2',compress(put(nmiss2,best.)));
  stop;
run;

data ex&dset.&analgr;
%if "&dset"="diff" %then %do;
  do replicate=2001 to 2000+&missobs;
    analgrln=&analgr;
    paramcd="&PARAM";
    trtan=4;
    estimate=-10000;
    output;
    estimate=10000;
    output;
  end;
%end;
%else %if "&dset"="lsmean" and "&analgr"="1" %then %do;
  do replicate=2001 to 2000+&missobs2;
    do trtan=4 to 5;
      analgrln=&analgr;
      paramcd="&PARAM";
      estimate=-10000;
      output;
      estimate=10000;
      output;
    end;
  end;
%end;
%else %if "&dset"="lsmean" and "&analgr"="2" %then %do;
  do replicate=2001 to 2000+&missobs2;
    do trtan=4 to 7 by 3;

```

```

        analgr1n=&analgr;
        paramcd="&PARAM";
        estimate=-10000;
        output;
        estimate=10000;
        output;
    end;
end;
%end;
run;

data &dset.&analgr.ex;
set &dset.&analgr ex&dset.&analgr;
run;

%mend;

%impute(analgr=1, dset=diff);
%impute(analgr=2, dset=diff);
%impute(analgr=1, dset=lsmean);
%impute(analgr=2, dset=lsmean);

data diffex&NUM;
set difflex diff2ex;
run;

data lsmeanex&NUM(drop=atptn);
set lsmeanlex lsmean2ex;
run;

%MEND;

%IMPUTEPARAM(NUM=1,PARAM=QSUTOTAL);
%IMPUTEPARAM(NUM=2,PARAM=QSUFAC1);
%IMPUTEPARAM(NUM=3,PARAM=QSUFAC2);

DATA DIFFEX;
SET DIFFEX1 DIFFEX2 DIFFEX3;
RUN;

DATA LSMEANEX;
SET LSMEANEX1 LSMEANEX2 LSMEANEX3;
RUN;

proc sort data=diffEX;
    by analgr1n paramcd trtan replicate;
run;

proc univariate data=diffEX noprint;
    by analgr1n paramcd trtan ;
    var estimate;

```

```

        output out=diffsbd MEDIAN=MED pctlpts=2.5 97.5 pctlpre=P;
run;

proc univariate data=diffEX(wher=(replicate 1e 2000)) noprint;
    by analgr1n paramcd trtan ;
    var estimate;
    output out=diffsbe mean=estimate ;
run;

proc univariate data=diffEX noprint;
    by analgr1n paramcd trtan ;
    var stderr;
    output out=diffsbf mean=stderr;
run;

data diffs;
    merge diffsbd diffsbe diffsbf;
    by analgr1n paramcd trtan ;
run;

proc sort data=lsmeanEX;
    by analgr1n paramcd trtan replicate;
run;

proc univariate data=lsmeanEX noprint;
    by analgr1n paramcd trtan ;
    var estimate;
    output out=lsmeansba mean=estimate MEDIAN=MED pctlpts=2.5 97.5
pctlpre=P n=nobs ;
run;

proc univariate data=lsmeanEX(wher=(replicate 1e 2000)) noprint;
    by analgr1n paramcd trtan ;
    var estimate;
    output out=lsmeansbb mean=estimate ;
run;

data lsmeans;
    merge lsmeansba lsmeansbb;
    by analgr1n paramcd trtan ;
run;

proc sort data=adqssu;
    by analgr1n paramcd trtan usubjid;
run;

PROC SQL;
CREATE TABLE NUM1 AS
SELECT ANALGR1N, PARAMCD, TRTAN, COUNT(DISTINCT USUBJID) AS N1
FROM ADQSSU
WHERE NOT MISSING(AVAL)
GROUP BY ANALGR1N, PARAMCD, TRTAN;

```

```
QUIT;
```

```
/*Read in bootstrap data*/
```

```
data bootin;  
    set diffsb lsmeansb num1 ;  
    rename p2_5=lower P97_5=upper;  
run;
```

```
/*Manipulate datasets for output all relevent stats on each row*/  
/* _____ */
```

```
data tabout;  
    length out $100 stat $100;  
    set bootin;  
    /*ordering columns of treatmntents*/  
    if not missing(stderr) then colord=3;  
    else if trtan=4 then colord=1;  
    else colord=2;  
    /* N row*/  
    if not missing(n1) then do;  
        ord=1;  
        stat='n';  
        out=compress(put(n1,best.));  
        output;  
    end;  
    /*mean (sd) row*/  
    if missing(n1) then do;  
        ord=2;  
        stat='Mean (SE)';  
        out=compress(put(round(estimate,0.01),8.2));  
        if colord=3 then out=compress(out)||'  
(||compress(put(CEIL(STDERR*1000)/1000,8.3))||)';  
        output;  
    /*95% CI row*/  
        ord=3;  
        stat='95% CI';  
        if lower gt -10000 and upper lt 10000 then do;  
  
out=compress(put(/*round(lower,0.01)*/FLOOR(100*LOWER)/100,8.2))||',  
'||compress(put(/*round(upper,0.01)*/CEIL(100*UPPER)/100,8.2)); /* 4) AMH  
17Jun2014 */  
  
        end;  
        else do;  
            out='NC';  
        end;  
        output;  
    /*MEDIAN ROW*/  
    ORD=4;  
    STAT='Median';  
    OUT=COMPRESS(PUT(ROUND(MED,0.01),8.2));  
    OUTPUT;  
end;
```

```

run;

/*Add labels for all number variables*/
/*_____*/
data tabout1;
    set tabout ;
/* timepoint label*/
    atptn=0;
    atpt='Overall';
/*Variable label*/
    if paramcd='QSUFAC1' then var='Factor 1 - Reward';
    else if paramcd='QSUFAC2' then var='Factor 2 - Relief';
    else if paramcd='QSUTOTAL' then DO;
        PARAMCD='QSUTOTAL';
        var='Total Score';
    END;
run;

/*transpose for output*/
proc sort data=tabout1;
    by analgr1n paramcd atptn ord colord;
run;

proc transpose data=tabout1 out=ttabout(drop=_NAME_) prefix=col;
    by analgr1n paramcd var atptn atpt ord stat;
    id colord;
    var out;
run;

/* Standard - macro for paging */
%macro outrtf(blankn=68, halfblnk=N, ref=);

/* treatment column headers and footnotes */

/* treatment column headers and footnotes */
/*group 1*/
%let col11=THS 2.2 Menthol#(N=&tot11);
%let col21=mCC#(N=&tot21);
%let col31=THS 2.2 Menthol - mCC;
%let foot1=%str(mCC = menthol conventional cigarettes);
/*group 2*/
%let col12=THS 2.2 Menthol#(N=&tot12);
%let col22=NRT gum#(N=&tot22);
%let col32=THS 2.2 Menthol - NRT gum;
%let foot2=%str(NRT gum = Nicotine Replacement Therapy gum);

%if &halfblnk=N %then %let halfblnk=;
%else %if &halfblnk=Y %then %let halfblnk=\~;

/* Standard - just change the number to match the listing you're working
on. Also change the letters in the*/

```

```

/* bracket, eg ccb = current cigarette brands. Make sure to do this at
the top of the code too. */

%let tflno=T_15_02_04_13(QSU);

/*page numbers*/
data paging;
  set ttabout;
  by analgrln paramcd atptn;
  flag=1;
  retain ln 0 page 0;
  if first.atptn then ln+1;
  if first.analgrln or ln>4 then do;
    page+1;
    ln=1;
  end;
  if last.analgrln then call symput("tpage",compress(put(page,best.)));
run;

/* Standard - leave this */
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;
%let linebot2 = \brdrb\brdrs\brdrw15;

ods path stdlib.tl06326 (read) ;
ods results off;
ods rtf toc_data/* contents*/
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 &tpage;

ODS PROCLABEL = ' ';
title ;
footnote;
%let wd=0;

data comp;
  set paging end=eof;
  by paramcd atptn ord;
  where page=&i;
  call symput('grp',compress(put(analgrln,best.)));
  /* Amend title as needed */
  _firtitl="Table 15.2.4.13 Analysis of QSU-brief Questionnaire
Factors and Total Score by Bootstrapping Techniques - PK Population";

```



```

        _upcas=(length(_firtitl)-
length(compress(_firtitl,'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
        len=&blankn.-length("(Page &i of &tpage)");
        if eof then do;
            call symput('_FSRTITL', trim(left(_firtitl)));
            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 missing nowd split = '#'
spanrows
%IF &I=1 %THEN %DO; CONTENTS=' ' %END; %ELSE %DO; CONTENTS='' %END;;
    column flag page paramcd var atptn atpt ord stat ("Group-&grp PK
&linebot." col1 col2 col3) ;
    define flag / order noprint;
    define page          / order order = internal noprint;
    define paramcd        / order order=internal noprint;
    define var            / group style={just=left cellwidth=2.5cm} "Variable";
    define atptn          / order order=internal noprint;
    define atpt           / group style={just=left cellwidth=2.1cm}
"Timepoint";
    define ord            / order order=internal noprint;
    define stat           / display style={just=left cellwidth=3cm}
"Statistic";
    define col1           / display style={just=c cellwidth=3cm}
style(header)={just=center} "&&col1&grp";
    define col2           / display style={just=c cellwidth=3cm}
style(header)={just=center} "&&col2&grp";
    define col3           / display style={just=c cellwidth=3cm}
style(header)={just=center} "&&col3&grp";

    break after page / page;

break before flag / page %IF &I=1 %THEN %DO;
    CONTENTS="&_FSRTITL" %END; %ELSE %DO; CONTENTS='' %END;;

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 "&linebot";
    endcomp;

    compute after atptn;
        line " ";
    endcomp;

    compute after _page_ / style={just=left protectspecialchars=off
pretext="&linetop."} ;
    LINE "Note: &&foot&grp; THS = Tobacco Heating System.";
    LINE "Note: Higher scores indicate a higher urge to smoke.";
    LINE "Note: Point and 95% interval estimates by means of the
percentile bootstrap technique of the adjusted least squares means from an
ANOVA model.";
    LINE "Note: NC = Not calculable, due to greater than 50
bootstrap samples failing to converge.";
    line "";
    line "Appendix &ref.";
    line "Path: &TFLpath." &_blankn.*"\~\~" "(Page &i of
&tpage)";
    line "Program Run: &sysdate &sysuserid Program Status:
&status";
    endcomp;

run;
%end;
ods rtf close;
ods results on;
ods path reset;

%mend ;

%outrtf(blankn=70, halfblnk=N, ref=15.4.4.13 and 15.3.6.11);

*****;
*Listing Output;
*****;
/*ordering columns of treatmnts*/

data listout;
    set diff(in=a) lsmean;

    if a then do;
        colord=3;
        out=estimate;
        output;
        colord=4;
        out=stderr;
        output;
    end;

```

```

        else if trtan=4 then do;
            out=estimate;
            colord=1;
            output;
        end;
        else do;
            out=estimate;
            colord=2;
            output;
        end;
run;

```

```

data listout1;
    set listout ;
    atptn=0;
    atpt='Overall';
/*Variable label*/
    if paramcd='QSUFAC1' then var='Factor 1 - Reward';
    else if paramcd='QSUFAC2' then var='Factor 2 - Relief';
    else if paramcd='QSUTOTAL' then DO;
        PARAMCD='QSUTOTAL';
var='Total Score';
END;
run;

```

```

/*transpose for output*/
proc sort data=listout1;
    by analgr1n paramcd var replicate atptn atpt;
run;

```

```

proc transpose data=listout1 out=tlistout1(drop=_NAME_) prefix=col;
    by analgr1n paramcd var replicate atptn atpt;
    id colord;
    var out;
run;

```

```

DATA TLISTOUT1A;
SET TLISTOUT1 (WHERE=(ANALGR1N=1) DROP=COL4);
RUN;

```

```

DATA TLISTOUT1B;
SET TLISTOUT1 (WHERE=(ANALGR1N=2) DROP=COL4);
RENAME COL1=COL4 COL2=COL5 COL3=COL6;
RUN;

```

```

PROC SORT DATA=TLISTOUT1A; BY PARAMCD REPLICATE; RUN;
PROC SORT DATA=TLISTOUT1B; BY PARAMCD REPLICATE; RUN;

```

```

DATA TLISTOUT;
MERGE TLISTOUT1A TLISTOUT1B;
BY PARAMCD REPLICATE;

```

```

RUN;

/* Standard - macro for paging */
%macro outrtf(blankn=68, halfblnk=N);

/* treatment column headers and footnotes */
/* treatment column headers and footnotes */
/*group 1*/
%let lab11=THS 2.2 Menthol#(N=&tot11);
%let lab21=mCC#(N=&tot21);
%let lab31=THS 2.2 Menthol - mCC;
%let foot1=%str(mCC = menthol conventional cigarettes);
/*group 2*/
%let lab12=THS 2.2 Menthol#(N=&tot12);
%let lab22=NRT gum#(N=&tot22);
%let lab32=THS 2.2 Menthol - NRT gum;
%let foot2=%str(NRT gum = Nicotine Replacement Therapy gum);

%if &halfblnk=N %then %let halfblnk=;
%else %if &halfblnk=Y %then %let halfblnk=\~;

/* Standard - just change the number to match the listing you're working
on. Also change the letters in the*/
/* bracket, eg ccb = current cigarette brands. Make sure to do this at
the top of the code too. */

%let tflno=L_15_04_04_13(QSU);

/* Standard - leave this */
%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",""));
run;

/*page numbers*/
data paging;
    set tlistout;

    flag=1;
    retain ln 0 page 1;
    ln+1;

    IF LN>20 THEN DO;
        page+1;
        ln=1;
    end;
    call symput("tpage",compress(put(page,best.)));
run;

```

```

DATA PAGING;
SET PAGING;
IF PAGE=&TPAGE AND LN>16 THEN PAGE=PAGE+1;
CALL SYMPUT ("TPAGE", COMPRESS (PUT (PAGE,BEST.)));
RUN;

```

```

data _null_;
set seed;
call symput ('seed', put (seed, 8.0));
run;

```

```

/* Standard - leave this */
options number nodate orientation=landscape papersize=&p_pgsz missing='
' NOQUOTELNMAX/*turn off warnings about quoted strings to long*/;
ods escapechar='`';
%let linetop = \brdrt\brdrs\brdrw30; * needs to be 1.5pt so calculated
in twips (1/20 pt) ;
%let linebot = \brdrb\brdrs\brdrw30;
%let linebot2 = \brdrb\brdrs\brdrw15;

```

```

ods path stdlib.tl06326 (read) ;
ods results off;
ods rtf toc_data/* contents*/
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 &tpage;

```

```

ODS PROCLABEL = ' ';
title ;
footnote;
%let wd=0;

```

```

data comp;
    set paging end=eof;
    by paramcd;
    where page=&i;
    _firtitl="Listing 15.4.4.13 Analysis of QSU-brief Questionnaire
Factors and Total Score by Bootstrapping Techniques - `n PK Population";
    _upcas=(length(_firtitl)-
length(compress(_firtitl,'ABCDEFGHIJKLMNOPQRSTUVWXYZ')))/2;
    len=&blankn.-length("(Page &i of &tpage)");
    if eof then do;
        call symput('_FSRTITL', trim(left(_firtitl)));
        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 missing nowd spanrows split =
'#'
%IF &I=1 %THEN %DO; CONTENTS=' ' %END; %ELSE %DO; CONTENTS='' %END;;

COLUMN FLAG PAGE PARAMCD VAR REPLICATE ATPT ("Group-1 PK &LINEBOT."
("&lab11" COL1) ("&LAB21" COL2) ("&LAB31 &LINEBOT." COL3))
      ("Group-2 PK &LINEBOT." ("&LAB12" COL4) ("&LAB22" COL5) ("&LAB32
&LINEBOT." COL6));
  define flag / order noprint;
  define page      / order order = internal noprint;
  define paramcd    / order order=internal noprint;
  define var        / group order=data style={just=left cellwidth=1.5cm}
"Variable";
  define replicate      / group order=internal style={just=left
cellwidth=1.5cm} "Replicate";
  define atpt          / group order=data style={just=left cellwidth=1.1cm}
"Time point";
  define col1          / display style={just=c cellwidth=1.5cm}
style(header)={just=center} format=8.2 "Lsmean estimate";
  define col2          / display style={just=c cellwidth=1.5cm}
style(header)={just=center} format=8.2 "Lsmean estimate";
  define col3          / display style={just=c cellwidth=1.5cm}
style(header)={just=center} format=8.2 "estimate";

  DEFINE COL4          / DISPLAY STYLE={JUST=C CELLWIDTH=1.5CM}
STYLE(HEADER)={JUST=CENTER} FORMAT=8.2 "Lsmean estimate";
  DEFINE COL5          / DISPLAY STYLE={JUST=C CELLWIDTH=1.5CM}
STYLE(HEADER)={JUST=CENTER} FORMAT=8.2 "Lsmean estimate";
  DEFINE COL6          / DISPLAY STYLE={JUST=C CELLWIDTH=1.5CM}
STYLE(HEADER)={JUST=CENTER} FORMAT=8.2 "estimate";

  /*headers*/

  break after page / page;

  break before flag / page %IF &I=1 %THEN %DO;
    CONTENTS="&_FSRTITL" %END; %ELSE %DO; CONTENTS='' %END;;

  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 "&linebot";
    endcomp;

    compute after _page_ / style={just=left protectspecialchars=off
pretext="&linetop."};
    %IF &I=&TPAGE %THEN %DO;
LINE "Note: &FOOT1; &FOOT2; THS = Tobacco Heating System.";

        line "Note: &seed. is the seed used to generate samples from
the SAS procedure proc survey select";
        line "Note: Estimates are the adjusted least squares means
from an ANOVA model.";
        line "Note: Where estimates are missing, the model did not
converge for that sample";
        line "";
line "Path: &TFLpath.";
%END;
line "Program Run: &sysdate   &sysuserid   Program Status: &status"
&_blankn.*"\~\~" "(Page &I of &TPAGE)";
    endcomp;

run;
%end;
ods rtf close;
ods results on;
ods path sashelp.tmplmst (read);

%mend ;

%outrtf(blankn=70, halfblnk=N );

*=====;
* END OF PROGRAM CODE ;
*=====;

```