

```

%_mprintto;
options notes nosource;
proc datasets lib=work nolist memtype=data kill; quit;
%put NOTE:
=====;
%put NOTE: Covance Study Number : 000000106324;
%put NOTE: Client Protocol ID   : ZRHR-REXC-03-EU;
%put NOTE: Program Name        : t_sdisp.sas;
%put NOTE: Purpose              : table of subject disposition;
%put NOTE: ;
%put NOTE: Input Data           : ADAM.ADSL ADAM.ADFA;
%put NOTE: Output               : t_15_2_1_1(sd);
%put NOTE: Macros Called        : _MPRINTTO;
%put NOTE: ;
%put NOTE: Programmed by        : cvn_jhardman;
%put NOTE: Creation Date        : 2014-07-29;
%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: 31Jul2014   JMH       1) Amended footnote;
%put NOTE: 16Sep2014   JR        2) Amended zero presentation;
%put NOTE: 18Sep2014   JMH       3) Made update before dual programming
table is created;
%put NOTE: 03Oct2014   JMH       4) Added a - before n (%);
%put NOTE:
=====;
options notes source source2 nofullstimer validvarname=upcase missing='
';
ods _all_ close;
ods listing;

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

%let tflno=T_15_02_01_01(sd);

%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",""));
run;

```

```

*****;
* read in data ;
*****;
data adsl;
    set adam.adsl;
run;

data adfa;
    set adam.adfa;
run;

data blanks;
    set adsl(firstobs=1 obs=14);
    statusn=_n_;
    keep statusn;
run;

data testdata;
    set adam.adsl;
    if not missing(dtestdtm) then testfl='Y';
run;

proc freq data=testdata;
    table siteid /noprint out=scrn (rename=(count=ov_scrn));
    table scrfl / noprint
out=sfl(where=(scrfl='Y') rename=(count=ov_scrn));
    table scrfl*testfl /noprint out=noprod(where=(scrfl='Y' and
(missing(testfl))) rename=(count=ov_scrn));
    table scrfl*testfl / noprint out=prod(where=(scrfl='Y' and (not
missing(testfl))) rename=(count=ov_scrn));
run;

data adfal;
    merge adfa testdata(keep=usubjid testfl);
    by usubjid;
run;

proc freq data=adfal(where=(scrfl='Y' and paramcd='WILLABL' and
aval=2));
    table parcat1*testfl /noprint out=willths(where=(parcat1='THS 2.2'
and testfl='Y') rename=(count=ov_scrn));
run;

proc freq data=adsl;
    tables trt01an*enrlfl / noprint
out=enr(where=(enrlfl='Y') rename=(count=enr));
    tables trt01an*enrlfl*enfl / noprint out=enr_nr(where=(enrlfl='Y'
and enfl='Y') rename=(count=enr_nr));
    tables trt01an*enrlfl*enfl / noprint out=enr_r(where=(enrlfl='Y'
and enfl='N') rename=(count=enr_r));
    tables trt01an*randfl*complfl / noprint
out=rand_c(where=(randfl='Y' and complfl='Y') rename=(count=rand_c));

```

```

        tables trt01an*randfl*complfl / noprint
out=rand_d(where=(randfl='Y' and complfl='N')rename=(count=rand_d));
        tables trt01an*randfl*complfl / noprint
out=rand_dw(where=(randfl='Y' and complfl='N' and trt01an
in(1,2,3))rename=(count=rand_dw));
        tables trt01an*randfl*complfl / noprint
out=rand_dwo(where=(randfl='Y' and complfl='N' and trt01an not
in(1,2,3))rename=(count=rand_dwo));
run;

data adsl2;
    set adsl(where=(complfl='N'));
    if trt01a in('THS 2.2','SA');
run;

data adsl3;
    set adsl (where=(complfl='N' and trt01a not in('THS 2.2','SA')));
run;

proc freq data=adsl3;
    tables siteid / noprint out=d_nru (rename=(count=ov_scrn));
run;

data treat;
    merge enr enr_nr enr_r rand_c rand_d rand_dw rand_dwo;
    by trt01an;
    rand=enr_r;
    keep trt01an enr enr_nr enr_r rand rand_c rand_d rand_dw rand_dwo;
run;

proc transpose data=treat out=treat2(drop= _label_);
    id trt01an;
    var enr enr_nr enr_r rand rand_c rand_d rand_dw rand_dwo ;
run;

data treat3;
    set treat2;
    if missing(_1) then do;
        _1=0; missingone='Y';
    end;
    if missing(_2) then do;
        _2=0; missingtwo='Y';
    end;
    if missing(_3) then do;
        _3=0; missingthree='Y';
    end;
    if missing(_97) then do;
        _97=0;missingfour='Y';
    end;
    ov_scrn=_1 + _2 + _3 + _97;
    if missingone='Y' then _1=.;
    if missingtwo='Y' then _2=.;
    if missingthree='Y' then _3=.;
    if missingfour='Y' then _97=.;

```

```

        statusn=_n_+5;
        drop missingone missingtwo missingthree missingfour;
run;

data overall1;
    set scrn sfl noprod prod willths;
    statusn=_n_;
run;

data overall2;
    merge overall1 blanks ;
    by statusn;
run;

data overall2a;
    merge overall2 blanks;
    by statusn;
run;

data overall3;
    merge overall2a treat3;
    by statusn;
    length status $132.;
    sort=1;
    statusn=_n_;
    if ov_scrn=. then ov_scrn=0;
    if statusn=1 then status='Total screened - n';
/*    if statusn=2 then status='    Screen failures n (%)'; */
    IF STATUSN=2 THEN STATUS='    Screen failures - n (%)'; /* 4) JMH
03Oct2014 */
    if statusn=3 then status='        Screening failures, without product
test - n (%)';
    if statusn=4 then status='        Screening failures, with product test
- n (%)';
    if statusn=5 then status='        Unwilling to use THS 2.2';
    if statusn=6 then status='    Enrolled - n (%)';
    if statusn=7 then status='        Enrolled, not randomized - n (%)';
    if statusn=8 then status='        Randomized - n (%)';
    if statusn=9 then status='Total randomized - n';
    if statusn=10 then status='    Completed - n (%)';
    if statusn=11 then status='    Discontinued - n (%)';
    if statusn=12 then status='        Discontinued, with randomized product
use';
    if statusn=13 then status='        Discontinued, without randomized
product use';
    if statusn in(6,7) then
        do;
            _1=.;
            _2=.;
            _3=.;
            _97=.;
        end;
    bynum=1;
    keep statusn status _1 _2 _3 _97 ov_scrn sort bynum;

```

```

run;

data adsl;
  set adam.adsl(where=(not missing(trt01an)));
  output;
  trt01an=100;
  trt01a='Overall safety';
  output;
run;

proc freq data=adsl noprint;
  table trt01an*trt01a/ out =tot(drop=percent);
run;

data dumtrts;
  attrib trt01a length=$200. trt01an length=8.;
  dumres=0;
  rown=0.5;
  trt01an=1;
  trt01a='THS 2.2';
  output;
  trt01an=2;
  trt01a='CC';
  output;
  trt01an=3;
  trt01a='SA';
  output;
  trt01an=97;
  trt01a='Enrolled not randomized';
  output;
  trt01an=100;
  trt01a='Overall safety';
  output;
run;

data tot2;
  merge dumtrts(in=a drop=dumres rown) tot(in=b);
  if a or b;
  if a and not b then count=0;
  by trt01an;;
  rename count=total;
  call symput('trt' || compress(put(trt01an,best.)),
compress(count));
run;

proc transpose data=tot2 out=sum prefix=_;
  id trt01an;
  var total;
run;

data sum_a;
  set sum;
  bynum=1;
run;

```

```

data sum01;
  merge sum_a overall3 (where=(statusn=1) keep=ov_scrn statusn bynum);
  by bynum;
  total_overall=ov_scrn;

  ov_rand=_1+_2+_3+_97;

  rename _1=total_1 _2=total_2 _3=total_3 _97=total_97;
  drop _name_ _label_;
run;

proc sort data=overall3;
  by statusn;
run;

data overall3_extra;
  set overall3;
  where status='Total randomized - n';
  tottran=ov_scrn;
  keep bynum tottran;
run;

data all;
  merge sum01 overall3 overall3_extra;
  by bynum;
  retain total_1 total_2 total_3 total_97;
  if statusn in(1,9) then do;
    p_ov=put(ov_scrn,best.);
    ov_scrn=.;
  end;
  else if statusn not in(1,9) then do;
    if _1>0 then p1a=( _1/total_1)*100;
    if _2>0 then p2a=( _2/total_2)*100;
    if _3>0 then p3a=( _3/total_3)*100;
    if _97>0 then p97a=( _97/total_97)*100;
    if statusn not in (11, 12, 13,14) then do;
      if ov_scrn>0 then p_ova=(ov_scrn/total_overall)*100;
    end;
  else do;
    p_ova=(ov_scrn/ov_rand)*100;
  end;

  if statusn in(10 11 12) then
p_ova=(ov_scrn/tottran)*100;

  if p1a=100 then p1=trim('('||compress(put(p1a,8.))||' %)');
  else if not missing(p1a) and p1a>10 and p1a<100 then p1=trim('('
' ||compress(put(p1a,8.1))||'%)');
  else if not missing(p1a) and p1a<10 then p1=trim('(
' ||compress(put(p1a,8.1))||'%)');

```

```

        if p2a=100 then p2=trim('('||compress(put(p2a,8.))||' %)');
        else if not missing(p2a) and p2a>10 and p2a<100 then p2=trim('('
' ||compress(put(p2a,8.1))||'%)');
        else if not missing(p2a) and p2a<10 then p2=trim('('
' ||compress(put(p2a,8.1))||'%)');

        if p3a=100 then p3=trim('('||compress(put(p3a,8.))||' %)');
        else if not missing(p3a) and p3a>10 and p3a<100 then p3=trim('('
' ||compress(put(p3a,8.1))||'%)');
        else if not missing(p3a) and p3a<10 then p3=trim('('
' ||compress(put(p3a,8.1))||'%)');

        if p97a=100 then p97=trim('('||compress(put(p97a,8.))||' %)');
        else if not missing(p97a) and p97a>10 and p97a<100 then p97=trim('('
' ||compress(put(p97a,8.1))||'%)');
        else if not missing(p97a) and p97a<10 then p97=trim('('
' ||compress(put(p97a,8.1))||'%)');

        if p_ova=100 then p_ov=trim('('||compress(put(p_ova,8.))||' %)');
        else if not missing(p_ova) and p_ova>10 and p_ova<100 then
p_ov=trim('(' ||compress(put(p_ova,8.1))||'%)');
        else if not missing(p_ova) and p_ova<10 then p_ov=trim('('
' ||compress(put(p_ova,8.1))||'%)');
        end;
        if statusn in (13,14) and ov_scrn=0 then delete;

        if statusn in (12,13) then do;
            if missing(_1) then _1=0;
/*            if missing(_2) then _2=0;*/ /* 2) JR 16Sep2014 */
/*            if missing(_3) then _3=0;*/
            if missing(_97) then _97=0;
        end;

        if statusn in (1 9) then do;
            ov_scrn=input(p_ov,best.);
            p_ov='';
        end;

        if statusn=11 then do;
            if _1=. then _1=0;
            if _2=. then _2=0;
            if _3=. then _3=0;
        end;

        rename _1=_tot1 _2=_tot2 _3=_tot3 _97=_tot97 ov_scrn=totoverall;
run;

proc sort data=all;
    by statusn;
run;

```

```

data all2;
    set all;
    attrib wrap length = $200;
    wrap = status;
i=55; *this is the max length allowed on a single line - change as
needed;
if length(wrap)>i then do;
    nwraps = int(length(wrap)/i); *calculate how many lines the text will
wrap over;
    do while(nwraps > 0);
        fin=0;
        j = i*nwraps; *calculate starting point - loop will cycle backwards
from this point looking for a space;
        test=j;
        do while(fin=0 and j gt 1);
            if substr(wrap,j,1)=' ' then do;
                wrap=substr(wrap,1,j-1) || " |S={foreground=white} . |S={} " ||
substr(wrap,j+1);
                fin=1;
            end;
            else j=j-1; *no space found - move back one character;
        end;
        nwraps=nwraps-1; *once this wrap is handled, move up a line until all
are handled (when nwraps = 0);
    end;
end;

    if statusn not in(1,10) then status=wrap;

    flag=1;

    IF STATUSN=5 THEN DELETE; /*No subjects unwilling to use product*/
/* 3) JMH 18Sep2014 */
run;

proc sql noprint;
    create table table.T_15_02_01_01 as
    select status, _tot1, p1, _tot2, p2, _tot3, p3, /*_tot97, p97,*/
totoverall, p_ov /* 3) JMH 18Sep2014 */
    from all2
    order by statusn;
quit;

data paging;
    set all2;
    by statusn;

    /*if statusn=5 then delete;*/ /*No subjects unwilling to use
product*/ /* 3) JMH 18Sep2014 */

    if ln gt 13 then ln=1;
    else ln+1;
    if ln=1 then page+1;

```



```

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

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;

%macro outrtf(blankn=130, halfblnk=N);

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

ods path stdlib.tl06324 (read) ;
ods results off;
ods rtf toc_data/* contents*/
file="/cvn/projects/prj/data/000000106324/TFL/&TFL_Part./&tflno..rtf"
style=tl06324 startpage=yes headery=1440 footery=1440 ;
ods noproctitle;
%do i=1 %to &page;

title ;
footnote;

data comp;
    set paging end=eof;
    where page=&i;

    /* Amend title as needed */
    _firtitl="Table 15.2.1.1 Summary of Subject Disposition -
All Screened Subjects";
    _upcas=(length(_firtitl)-
length(compress(_firtitl,'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.)));
    end;
    drop _firtitl _upcas len;
run;

ods proclabel = ' ';
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 = '$' %if &i=1 %then
%do; contents=' ' %end; %else %do; contents='' %end;;
    column flag page statusn ("Status$" status) ("THS
2.2$(N=&trt1)" _tot1 p1)
    ("CC$(N=&trt2)" _tot2 p2) ("SA$(N=&trt3)" _tot3 p3)
("Overall$Screened$" totoverall p_ov);
    define flag          / order order = internal noprint;
    define page          / order order = internal noprint;
    define statusn       / order order = internal noprint;
    define status        / display style={just=left
cellwidth=7cm}'';
    define _tot1          / display style={just=d
cellwidth=0.6cm} style(header)={just=center} "";
    define _tot2          / display style={just=d
cellwidth=0.6cm} style(header)={just=center} "";
    define _tot3          / display style={just=d
cellwidth=0.6cm} style(header)={just=center} "";
    define totoverall / display style={just=d cellwidth=0.6cm}
style(header)={just=center} "";
    define p1             / display style={just=r cellwidth=1.6cm}
style(header)={just=c}"";
    define p2             / display style={just=r cellwidth=1.6cm}
style(header)={just=c}"";
    define p3             / display style={just=r cellwidth=1.6cm}
style(header)={just=c}"";
    define p_ov           / display style={just=r cellwidth=1.4cm}
style(header)={just=c}"";

    break before flag / page %if &i=1 %then %do;
    contents="&_fsrtitl" %end; %else %do; contents='' %end;;

    break after page / page;

    compute status;
        if statusn=1 or statusn=9 then
            call define(_row_,'style','style=[font_weight=bold]');
        endcomp;

    compute before page / style={just=left protectspecialchars=off};
        line "&linetop";
    endcomp;

    compute after page / style={just=left protectspecialchars=off};
        line "&linebot";
    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};
/*      line 'Note: CC = Conventional cigarettes; SA = Smoking
abstinence; THS = Tobacco Heating System.'; */
        line 'Note: CC = Conventional cigarettes; SA = Smoking
abstinence; THS = Tobacco Heating System.'; /* 1) JMH 31Jul2014 */
        line 'Note: Enrolled subjects are those who are eligible on
Day -2; Discontinued refers to randomized subjects who discontinued the
study before the planned ';
        line 'discharge at Day 6; Completed refers to randomized
subjects who did not discontinue the study before the planned discharge
at Day 6.';
        line 'Note: Percentages are based on the number of subjects
indicated in the column header (N), apart from the Overall column where
percentages of completed and ';
        line 'discontinued refer to the total of subjects
randomized.';
        line ' ';
        LINE "Appendix 15.3.1.7";
        line "Path: &TFLpath." &_blankn.*"\~\~" "(Page &i of &page)";
;
        line "Program Run: &sysdate   &sysuserid   Program Status:
&status";
        endcomp;
run;
%end;
ods rtf close;
ods results on;
ods path sashelp.tmplmst (read);

%mend ;

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

ods listing;
proc printto print = "&table./T_15_02_01_01.lst" new;
run;

proc contents data = table.T_15_02_01_01 varnum;
run;
ods listing close;

proc printto ; run;
*=====;
* END OF PROGRAM CODE ;
*=====;

```