

*****,

* Project : ZRHM-REXA-07-JP

*

* Program name : t1502010301_ZRHM-REXA-07_V1.sas

*

* Author : W. Yang

*

* Date created : 06/01/2015

*

* Purpose : Create Table 15.2.1.3.1 Summary of Protocol Deviations ◆ Safety Population

*

* Revision History :

*

* Date Author Ref Revision (Date in YYYYMMDD format)

*

*****,

%let prgname=T1502010301_ZRHM_REXA_07_JP_V1;

options nomprint nosymbolgen nomlogic validvarname=upcase;

options sasautos=("W:\pmp07\macros" sasautos) notes;

%init(delivery=9);

%titlecsv(prgname=&prgname., version=5);

%put &title1;

%put &title2;

%put &APPENDIX;

```
%put &endpoint;
```

```
%put &outname.;
```

```
data adsl;
```

```
    set adam.adsl(where=(safafi='Y'));
```

```
    if   trt01an=4 then do; trt=1; output; end;
```

```
    else if trt01an=5 then do; trt=2; output; end;
```

```
    else if trt01an=3 then do; trt=3; output; end;
```

```
    trt=4; output;
```

```
run ;
```

```
proc freq data =adsl noprint;
```

```
    table trt/out=treatabt (rename =(count=total)drop=percent);
```

```
run;
```

```
data _null_;
```

```
    Set treatabt;
```

```
    Call symput('n1' || strip(put(trt, best.)),strip(put(total, best.)));
```

```
Run;
```

```
%put &n1 &n2 &n3 &n4;
```

```
*** Prepare data for analysis ***;
```

```
proc sql;
```

```
    create table data1 as
```

```
    select a.trt, a.disccat, a.cmpcp1fl, a.cmpcp2fl, a.cmpcp3fl, a.cmpcp4fl, b.*
```

```
from adsl as a cross join adam.addv as b

where a.usubjid=b.usubjid

;

quit;
```

```
data datax0;
```

```
set data1;
```

```
length cat scat $50;
```

```
disc=substr(disccat,21,1);
```

```
order1=1; order2=1; cat='Total';scat=cat; output;
```

```
if dvsig='Major' then do; order1=2; order2=1; cat='Major'; scat=cat; output; end;
```

```
if dvsig^='Minor' and upcase(evalcat)^='EVALUABLE' then do; order1=3; order2=1; cat='With
evaluability impact'; scat=cat; output; end;
```

```
if dvsig^='Minor' and upcase(evalcat)^='EVALUABLE' then do; order1=3; order2=2; cat='With
evaluability impact'; scat=dvcat; output; end;
```

```
if dvsig^='Minor' and upcase(evalcat)^='EVALUABLE' then do; order1=4; order2=1; cat='Without
evaluability impact'; scat=cat; output; end;
```

```
if dvsig^='Minor' and upcase(evalcat)^='EVALUABLE' then do; order1=4; order2=2; cat='Without
evaluability impact'; scat=dvcat; output; end;
```

```
if dvsig='Minor' then do; order1=5; order2=1; cat='Minor'; scat=cat; output; end;
```

```
if dvsig='Minor' then do; order1=5; order2=2; cat='Minor'; scat=param; output; end;
```

```
/* if dvsig='Major' and upcase(evalcat)^='EVALUABLE' and cmpcp1fl='N' then do; order1=3; order2=2;
cat='With evaluability impact'; scat='Mis-use of product in Period 1'; output; end;*/
```

```
/* if dvsig='Major' and upcase(evalcat)^='EVALUABLE' and disc not in ('1') and cmpcp2fl='N' then
do; order1=3; order2=2; cat='With evaluability impact'; scat='Mis-use of product in Period 2'; output;
end;*/
```

```
/* if dvsig='Major' and upcase(evalcat)='EVALUABLE' and disc not in ('1' '2') and cmpcp3fl='N' then  
do; order1=3; order2=2; cat='With evaluability impact'; scat='Mis-use of product in Period 3'; output;  
end;*/
```

```
/* if dvsig='Major' and upcase(evalcat)='EVALUABLE' and disc not in ('1' '2' '3') and cmpcp4fl='N' then  
do; order1=3; order2=2; cat='With evaluability impact'; scat='Mis-use of product in Period 4'; output;  
end;*/
```

```
run;
```

```
proc sort data=datax0 out=datax nodupkey; by usubjid trt cat scat; run;
```

```
%macro mfreq(in_dsn=, n_max=, order1=, order2=, class=);
```

```
proc means data=&in_dsn noprint nway;
```

```
class &class trt;
```

```
var trt;
```

```
output out=stat1_a n=count;
```

```
run;
```

```
proc means data=&in_dsn.0 noprint nway;
```

```
class &class trt;
```

```
var trt;
```

```
output out=stat1_b n=count0;
```

```
run;
```

```
data stat2 (drop=_type_ _freq_ percentx);
```

```
merge stat1_a stat1_b (drop=_type_ _freq_);
```

```
by &class trt;
```

```

length percentage $25;

%do i=1 %to &n_max;

    if trt=&i and count>. then percentx=count/%eval(&&n&i)*100;

    if    percentx=100 then percentage=put(count,4.)||' (100) '||strip(put(count0,best.));

    else if percentx>=0.1 then percentage=put(count,4.)||' ('||strip(put(percentx,5.1))||' )
' ||strip(put(count0,best.));

    else if percentx>. then percentage=put(count,4.)||' (<0.1) '||strip(put(count0,best.));

%end;

run;

```

```

proc transpose data=stat2 out=stat3 prefix=col;

    by &class;

    id trt;

    var percentage;

run;

```

```

data final_1 (drop=_name_);

    set stat3;

    %do i=1 %to &n_max;

        if col&i=" then col&i='0';

    %end;

run;

%mend mfreq;

```

```

%mfreq(in_dsn=datax, n_max=4, class=order1 cat order2 scat);

```

```

*** Prepare the output data set per mock-up/shell ***;

data final;

    set final_1;

    by order1 order2;

    if order1=1 then order0=1; else order0=2;

    scat=strip(uppercase(substr(scat,1,1)))||lowercase(substr(scat,2));

    stat=scat;

    if order1 in (3 4) and order2=1 then stat=' '|strip(stat);

    if order1 in (3 4) and order2=2 then stat=' '|strip(stat);

    if order1 in (5) and order2=2 then stat=' '|strip(stat);

    pageno=1;

    if order1=5 then pageno=2;

run;


data odata.%sysfunc(scan(&prgname,1,'_'));

    set final(in=a);

    if a then group="Part1";

run;


%global totalpage;


data _null_;

    set final end=eof;

    if eof then do;

        call symput('totalpage', trim(left(put(pageno,8)))));

```

```

end;

run;

%put totalpage=&totalpage;

%let sline=\brdrb\brdrs\brdrw10\brsp20;

%trtrtf(pgmname=&outname., pgmid=1, new=0, style=, bookmark=%lowercase(&outname.));

%macro reppart;

%do i = 1 %to &totalpage;

proc report data=final(where=(pageno=&i.)) headskip headline spacing=4 nowd split='|'
style=[outputwidth=100%] style(header column)=[protectspecialchars=off];

    column pageno order0 order1 order2 stat ("THSm2.2|(N=&n1) &sline." col1)

        ("mCC|(N=&n2) &sline." col2) ("SA|(N=&n3) &sline." col3) ("Overall|Safety|(N=&n4) &sline."
col4) ;

    define pageno /order order=internal noprint;

    define order0 /order order=internal noprint;

    define order1 /order order=internal noprint;

    define order2 /order order=internal noprint;

    define stat /display "Classification of Deviation" style(column)=[cellwidth=30% asis=on]
style(header)=[just=l];

    define col1 /display "n (%) Events" flow style(column)=[cellwidth=11% just=c];

    define col2 /display "n (%) Events" flow style(column)=[cellwidth=11% just=c];

    define col3 /display "n (%) Events" flow style(column)=[cellwidth=11% just=c];

```

```

define col4 /display "n (%)" Events" flow style(column)=[cellwidth=11% just=c];

compute before order0 ;

    line "";

endcomp;

compute before _page_/style=[fontweight=bold fontsize=3.75];

    line @1 "&title1 &title2";

    line @1 "^R/RTF'\brdrb\brdrs\brdrw30\brsp20\b ' ";

endcomp;

compute after _page_/style=[fontsize=1.75];

    line @1 "Note: mCC = Menthol conventional cigarettes; SA = Smoking abstinence; THSm2.2 =
Tobacco Heating System 2.2 Menthol.";

    line @1 "Note: Percentages are based on the number of subjects indicated in the column header
(N).";

    line @1 "Note: Subjects are counted only once per deviation category but can be counted in more
than one deviation category.";

    line @1 "";

    line @1 "&APPENDIX.";

    line @1 "Study ID:ZRHM-REXA-07-JP      Program: &fprgname..sas      Status:
&repversion./&fdate.      Page: &i of &totalpage.";

endcomp;

compute after pageno ;

    line "";

endcomp;

run;

```


%end;

%mend;

%reppart;

ods listing;

ods rtf close;