

*****,

* Project : ZRHM-REXA-07-JP

*

* Program name : t1502020501_ZRHM-REXA-07_V1.sas

*

* Author : W. Yang

*

* Date created : 06/06/2015

*

* Purpose : Create Table Table 15.2.2.5.1 Summary of Average Daily Product Use by Product Use Category in Ambulatory Period - FAS

*

* Revision History :

*

* Date Author Ref Revision (Date in YYYYMMDD format)

*

*****,

%let prgname=T1502020501_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=(fasfl='Y'/* and index(disccat,'Period 1')=0*/));
```

```
    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;
```

```
run ;
```

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

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

```
run;
```

```
Data _null_;
```

```
    Set treatabt;
```

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

```
Run;
```

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

```
proc sort data=adam.adex out=adex;
```

```
    by usubjid;
```

```
    where fasfl='Y' and dtype='AVERAGE' and aval not in (.);
```

```
run;
```

```

data data0;

    set adex;

    param=tranwrd(param,'Ave. Daily ','');

    param=tranwrd(param,'Average Daily ','');

    param=tranwrd(param,' by Period','');

    param=tranwrd(param,' in Ambulatory','');

    param=strip(param);

    if param='THS 2.2' then param='THSm2.2';

    if param='mCC' then param='CC/mCC';

run;


proc sort; by param paramn apuper apuperc; run;

```

```

data data1(rename=(ord=paramn));

    length apuperc $50;

    set data0;

    by param paramn apuper apuperc;

    if parcat3n=3 then do; apuper=5; apuperc='Overall Ambulatory'; end;

    retain ord;

    if first.param then ord=paramn;

    drop paramn;

run;

```

```
proc sort; by usubjid paramn param apuper apuperc; run;
```

```
proc freq data=data1 noprint;
```

```
tables paramn*param*apuper*apuperc/out=param(drop=count percent);
```

```
run;
```

```
proc sql;
```

```
create table data1a as
```

```
select a.usubjid,b.*
```

```
from adsl as a cross join param as b
```

```
;
```

```
quit;
```

```
data data1a;
```

```
merge adsl(keep=usubjid trt01an gpucat: pucat:) data1a;
```

```
by usubjid;
```

```
run;
```

```
data data1b;
```

```
merge data1a(in=a) data1(in=b drop= gpucat: pucat:);
```

```
by usubjid paramn param apuper apuperc;
```

```
if a;
```

```
run;
```

```
data data2;
```

```

set data1b;

if trt01an=4 and (gpucat2n=2 and apuper=2 or gpucat3n=2 and apuper=3 or gpucat4n=2 and
apuper=4

    or gpucat5n=2 and apuper=5) then do; order1=1; trt=1; output; end;

if trt01an=4 and (gpucat2n=3 and apuper=2 or gpucat3n=3 and apuper=3 or gpucat4n=3 and
apuper=4

    or gpucat5n=3 and apuper=5) then do; order1=1; trt=2; output; end;

if trt01an=4 and (gpucat2n=1 and apuper=2 or gpucat3n=1 and apuper=3 or gpucat4n=1 and
apuper=4

    or gpucat5n=1 and apuper=5) then do; order1=1; trt=3; output; end;

if trt01an=4 and (gpucat2n=4 and apuper=2 or gpucat3n=4 and apuper=3 or gpucat4n=4 and
apuper=4

    or gpucat5n=4 and apuper=5) then do; order1=1; trt=4; output; end;

if trt01an=4 and (trt>0) then do; order1=1; trt=5; output; end;

/* if trt01an=4 and (gpucat2n>. and apuper=2 or gpucat3n>. and apuper=3 or gpucat4n>. and
apuper=4 */

/* or gpucat5n>. and apuper=5) then do; order1=1; trt=5; output; end; */

if trt01an=4 and (pucat2n=3 and apuper=2 or pucat3n=3 and apuper=3 or pucat4n=3 and apuper=4

    or pucat5n=3 and apuper=5) then do; order1=2; trt=1; output; end;

if trt01an=4 and (pucat2ex^="" and apuper=2 or pucat3ex^="" and apuper=3 or pucat4ex^="" and
apuper=4

    or pucat5ex^="" and apuper=5) then do; order1=2; trt=2; output; end;

if trt01an=4 and (pucat2n=4 and apuper=2 or pucat3n=4 and apuper=3 or pucat4n=4 and apuper=4

    or pucat5n=4 and apuper=5) then do; order1=2; trt=3; output; end;

if trt01an=4 and (pucat2n in (3 4) and apuper=2 or pucat3n in (3 4) and apuper=3 or pucat4n in (3 4)
and apuper=4

    or pucat5n in (3 4) and apuper=5) then do; order1=2; trt=4; output; end;

```

if trt01an=4 and (pucat2n=5 and apuper=2 or pucat3n=5 and apuper=3 or pucat4n=5 and apuper=4
 or pucat5n=5 and apuper=5) then do; order1=3; trt=1; output; end;

if trt01an=4 and (pucat2n=6 and apuper=2 or pucat3n=6 and apuper=3 or pucat4n=6 and apuper=4
 or pucat5n=6 and apuper=5) then do; order1=3; trt=2; output; end;

if trt01an=4 and (pucat2n in (5 6) and apuper=2 or pucat3n in (5 6) and apuper=3 or pucat4n in (5 6)
 and apuper=4
 or pucat5n in (5 6) and apuper=5) then do; order1=3; trt=4; output; end;

if trt01an=4 and (pucat2n=8 and apuper=2 or pucat3n=8 and apuper=3 or pucat4n=8 and apuper=4
 or pucat5n=8 and apuper=5) then do; order1=4; trt=1; output; end;

if trt01an=4 and (pucat2n=9 and apuper=2 or pucat3n=9 and apuper=3 or pucat4n=9 and apuper=4
 or pucat5n=9 and apuper=5) then do; order1=4; trt=2; output; end;

if trt01an=4 and (pucat2n in (8 9) and apuper=2 or pucat3n in (8 9) and apuper=3 or pucat4n in (8 9)
 and apuper=4
 or pucat5n in (8 9) and apuper=5) then do; order1=4; trt=4; output; end;

if trt01an=5 and (pucat2n=1 and apuper=2 or pucat3n=1 and apuper=3 or pucat4n=1 and apuper=4
 or pucat5n=1 and apuper=5) then do; order1=5; trt=1; output; end;

if trt01an=5 and (pucat2n=1 and apuper=2 or pucat3n=1 and apuper=3 or pucat4n=1 and apuper=4
 or pucat5n=1 and apuper=5) then do; order1=5; trt=3; output; end;

if trt01an=3 and (pucat2n=12 and apuper=2 or pucat3n=12 and apuper=3 or pucat4n=12 and
 apuper=4
 or pucat5n=12 and apuper=5) then do; order1=6; trt=1; output; end;

if trt01an=3 and (pucat2n=11 and apuper=2 or pucat3n=11 and apuper=3 or pucat4n=11 and
 apuper=4

```

        or pucat5n=11 and apuper=5)    then do; order1=6; trt=2; output; end;

    if trt01an=3 and (pucat2n=10 and apuper=2 or pucat3n=10 and apuper=3 or pucat4n=10 and
apuper=4

        or pucat5n=10 and apuper=5)    then do; order1=6; trt=3; output; end;

    if trt01an=3 and (trt>0)    then do; order1=6; trt=4; output; end;

/*  if trt01an=3 and (pucat2n>9 and apuper=2 or pucat3n>9 and apuper=3 or pucat4n>9 and
apuper=4*/

/*    or pucat5n>9 and apuper=5 )    then do; order1=6; trt=4; output; end;*/

run;

*** Macro for Continuous variables statistics ***;

%macro mmeans(in_dsn=, class=, var=, order1=);

proc means data=&in_dsn noprint nway alpha=0.05;

    class &class trt;

    var &var;

    output out= stat1 n=_n nmiss=_nm mean=_mean std=_std median=_median min=_min max=_max
lclm=_lclm uclm=_uclm;

run;

proc sort data=stat1(keep=order1 paramn param) nodupkey out=shell; by order1 paramn param; run;

data shell;

    set shell;

    if    order1=1 then do; %do i=1 %to 5; trt=&i.; output; %end; end;

    else if order1=5 then do; %do i=1 %to 3; trt=&i.; output; %end; end;

    else          do; %do i=1 %to 4; trt=&i.; output; %end; end;

```

```
run;
```

```
data shell;
```

```
    set shell;
```

```
    length apuperc $50;
```

```
    apuper=2; apuperc='Period 2'; output;
```

```
    apuper=3; apuperc='Period 3'; output;
```

```
    apuper=4; apuperc='Period 4'; output;
```

```
    apuper=5; apuperc='Overall Ambulatory'; output;
```

```
run;
```

```
proc sort data=shell; by order1 paramn param apuper apuperc trt; run;
```

```
proc sort data=stat1; by order1 paramn param apuper apuperc trt; run;
```

```
data stat2;
```

```
    merge shell stat1 ;
```

```
    by order1 paramn param apuper apuperc trt;
```

```
    if _n=. then _n=0;
```

```
run;
```

```
proc sort data=stat2; by order1 paramn param apuper apuperc _n trt; run;
```

```
data stat2;
```

```
    set stat2;
```

```
    by order1 paramn param apuper apuperc _n trt;
```



```

retain pmin;

if first.apuperc then pmin=_n;

run;

proc sort data=stat2; by order1 paramn param descending pmin apuper apuperc trt; run;

data stat3;

set stat2;

by order1 paramn param descending pmin apuper apuperc trt;

retain mpmin;

if first.param then mpmin=pmin;

if order1 in (2 3 4) and mpmin<=4 then delete;

run;

proc sort; by &class trt; run;

data stat4 ;

length n meansd median minmax ci $50 fm1-fm3 $10;

set stat3 ;

deci=0;

fm1=cats(12,".",strip(put(deci+1,3.)));

fm2=cats(12,".",strip(put(deci+2,3.)));

fm3=cats(12,".",strip(put(deci+0,3.)));

```

```

if _std>. then _std=ceil(_std*100)/100;

if _uclm>. then _uclm=ceil(_uclm*10)/10;

if _lclm>. then _lclm=floor(_lclm*10)/10;

if _lclm=. and _std=0 and _mean>. then do; _lclm=_mean; _uclm=_mean; end;


if _mean ne . and _std ne . then meansd=strip(putn(_mean, fm1))||"
("||strip(putn(_std,fm2))||");

else if _mean ne . and _std eq . then meansd=strip(putn(_mean, fm1))||" (NA)";

if _lclm ne . then ci=strip(putn(_lclm,fm1))||", "||strip(putn(_uclm,fm1));

if _min ne . then minmax=strip(putn(_min,fm3))||", "||strip(putn(_max,fm3));

if _n>. then n=strip(put(_n,best12.));

if _nm>. then nm=strip(put(_nm,best12.));

if _median ne . then median=strip(putn(_median, fm1));


if _n<4 then do; meansd='NC'; ci='NC'; median='NC'; minmax='NC'; end;

run;


proc transpose data=stat4 out=stat5 prefix=col;

by &class;

var n nm meansd ci median minmax;

id trt;

run;


data final_&order1. (drop=_name_);

length stat $100;

set stat5;

```

```

if _name_='N' then do; stat="n"; order2 =1; end;

if _name_='NM' then do; stat="Missing"; order2 =2; end;

if _name_='MEANS' then do; stat="Mean (SD)"; order2 =3; end;

if _name_='CI' then do; stat="95% CI"; order2 =4; end;

if _name_='MEDIAN' then do; stat="Median"; order2 =5; end;

if _name_='MINMAX' then do; stat="Min, Max"; order2 =6; end;

%do i=1 %to 5; if order2=2 and col&i=" then col&i='0'; %end;

run;

%mend mmeans;

%mmeans(in_dsn=data2, class=order1 paramn param apuper apuperc, var=aval, order1=1);

data final;

set final_;;

if order2 in (1 2) then do;

if col1=" then col1='0'; if col2=" then col2='0';

if col3=" then col3='0'; if col4=" then col4='0'; if col5=" then col5='0';

end;

if order1 in (5 6) and paramn=2 then delete;

run;

proc sort data=final; by order1 paramn param apuper apuperc; run;

data final;

```

```
set final;

by order1 paramn param apuper apuperc;

retain count pageno 0;

if first.order1 then do;

    pageno+1;

    count=0;

end;

count+1;

if count=25 then do;

    pageno+1;

    count=1;

end;

run;


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

    set final;

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;
```

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

```
options nomprint nosymbolgen;
```

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

```
%macro reppart;
```

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

```
data out;
```

```
set final(where=(pageno=&i.));
```

```
call symput('order1', strip(put(order1,8.)));
```

```
run;
```

```
proc report data=out headskip headline spacing=4 nowd split='|' style=[outputwidth=100%]  
style(header column)=[protectspecialchars=off];
```

```
column pageno order1 paramn param apuper apuperc order2 stat
```

```
%if &order1=1 %then %do; ("THSm2.2 Product Use Category &sline." col1-col4) ("Overall"  
col5); %end;
```

```
%else %if &order1=2 %then %do; ("THSm2.2 Product Use Category &sline." col1-col3) col4; %end;
```

```
%else %if &order1=3 %then %do; ("THSm2.2 Product Use Category &sline." col1-col3) col4; %end;
```

```
%else %if &order1=4 %then %do; ("THSm2.2 Product Use Category &sline." col1-col3) col4; %end;
```

```
%else %if &order1=5 %then %do; ("mCC Product Use Category &sline." col1-col2) col3; %end;
```

```

%else %if &order1=6 %then %do; ("SA Product Use Category &sline." col1-col3) col4; %end;

define pageno /order order=internal noprint;

define order1 /order order=internal noprint;

define paramn /order order=internal noprint;

define apuper /order order=internal noprint;

define order2 /order order=internal noprint;

define param /order "Product" style(column)=[cellwidth=17% asis=on] style(header)=[just=l];

define apuperc /order "Period" style(column)=[cellwidth=15% asis=on] style(header)=[just=l];

define stat /display "Statistic" style(column)=[cellwidth=10% asis=on] style(header)=[just=l];

%if &order1=1 %then %do;

define col1 /display "THS 2.2|[70-100%]" flow style(column)=[cellwidth=12% just=c];

define col2 /display "Dual Use|]30-70%[" flow style(column)=[cellwidth=9% just=c];

define col3 /display "CC|[0-30%]" flow style(column)=[cellwidth=9% just=c];

define col4 /display "Not|Abstinent" flow style(column)=[cellwidth=9% just=c];

define col5 /display "THSm2.2|(N=&n1.)" flow style(column)=[cellwidth=12% just=c];

%end;

%else %if &order1=2 %then %do;

define col1 /display "Primarily" flow style(column)=[cellwidth=12% just=c];

define col2 /display "Exclusively" flow style(column)=[cellwidth=12% just=c];

define col3 /display "Predominantly" flow style(column)=[cellwidth=12% just=c];

define col4 /display "THS 2.2 [70-100]" flow style(column)=[cellwidth=12% just=c];

%end;

%else %if &order1=3 %then %do;

define col1 /display "Dual mostly THS" flow style(column)=[cellwidth=12% just=c];

```

```

define col2 /display "Dual Balance"    flow style(column)=[cellwidth=12% just=c];

define col3 /display "Dual mostly CC"  flow style(column)=[cellwidth=12% just=c];

define col4 /display "Dual ]30-70["   flow style(column)=[cellwidth=12% just=c];

%end;

%else %if &order1=4 %then %do;

    define col1 /display "Predominantly CC" flow style(column)=[cellwidth=12% just=c];

    define col2 /display "Primarily CC"    flow style(column)=[cellwidth=12% just=c];

    define col3 /display "Exclusively CC"  flow style(column)=[cellwidth=12% just=c];

    define col4 /display "CC [0-30]"       flow style(column)=[cellwidth=12% just=c];

%end;

%else %if &order1=5 %then %do;

    define col1 /display "Exclusively CC"      flow style(column)=[cellwidth=12% just=c];

    define col2 /display "Use of other | products" flow style(column)=[cellwidth=12% just=c];

    define col3 /display "Total mCC arm |(N=&n2.)" flow style(column)=[cellwidth=12% just=c];

%end;

%else %if &order1=6 %then %do;

    define col1 /display "Abstinent"          flow style(column)=[cellwidth=12% just=c];

    define col2 /display "Predominantly | Abstinent" flow style(column)=[cellwidth=12% just=c];

    define col3 /display "Not Abstinent"       flow style(column)=[cellwidth=12% just=c];

    define col4 /display "Total SA arm |(N=&n3.)" flow style(column)=[cellwidth=12% just=c];

%end;

compute before apuper ;

    *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; NC = Not calculated.";

    line @1 "Note: Periods defined as Period 2 ([Day 6 ambulatory ♦ Day 30 Visit]), Period 3 ([Day 30
Visit ♦ Day 60 Visit]) and Period 4 ([Day 60 Visit ♦ Day 90 Visit]).";

/*    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;
```