Paper 239-2011

# Utilizing PRELOADFMT Option with User-defined Formats to Create Summary Tables

Suwen Li, Everest Research Services Inc., Markham, ON.
Yi Hua, Cangene Corporation, Winnipeg, MB.
Daniel Li, Cangene Corporation, Winnipeg, MB.
Bob Lan, Everest Research Services Inc., Markham, ON.

## ABSTRACT

Occasionally people need to summarize data with all possible combinations of the class-level values to appear in the summary table, even if the data did not contain some of the class-level values, or display the output according to a specified order. Although we can accomplish this task with powerful SAS® DATA steps, the PRELOADFMT option with user-defined formats can achieve this easily without much programming work.

## INTRODUCTION

In pharmaceutical industry, it is common that we need to summarize data based on all possible combinations of values. However, if there is no data collected for all levels of these values in the dataset, by default, existing PROC procedures are not able to display them in a summary table. Another common request for summary tables is to present data in a user-defined order rather than in a simple increasing or descending order. In addition, for presentation purpose, extra space between lines is typically needed for summary tables. The traditional technique to solve these problems is to use a SAS data step to create additional variables or observations in a data set. Stacey D. Phillips described five ways to insert zero rows in summary tables [1].

In this paper, we propose an alternative approach which requires less data manipulation by utilizing PRELOADFMT option through user-defined formats. The option specifies that all formats are preloaded for the levels of all variables of interest. Combined with other options, we can easily show data at all the combinations even if not all levels are presented in the input dataset. Also, the proposed approach can easily deal with the user-defined ordering problem by using NOTSORTED option in the PROC FORMAT.

The PRELOADFMT option is currently available in:

- PROC REPORT,
- PROC TABLUATE,
- PROC MEANS,
- PROC SUMMARY,

which is used in conjunction with some other specific options in these procedures. In the following sections, we will provide some examples to demonstrate how to use the proposed approach in the above procedures. Examples 1 to 5 introduce the basic technique on how to use this approach to generate summary tables using the above procedures. Examples 6 and 7 will show you how to use this approach to generate complicated but typical summary tables in clinical trials.

## PRELOADFMT OPTION IN PROC REPORT

The PRELOADFMT option should be used in conjunction with COMPLETECOLS | COMPLETEROWS in PROC REPORT procedure to display all possible combinations of the values of the across or group variables, even if one or more levels of values do not exist in the input data set [2]. Here, we consider two typical summary tables to present post-marketing adverse event data in pharmacovigilance. In pharmacovigilance, the adverse events data usually come from three sources: clinical studies, literature, and spontaneous reports. The summary table is required to present the total number of AEs in the combination of AE source, expectedness, and seriousness. However, the reported adverse events in some reporting periods or some studies may not contain all of the required categories. For example, adverse events in the below data are only from spontaneous report, and all the events are serious. Example 1 shows you how to use PRELOADFMT and COMPLETECOLS in PROC REPORT for across variables to generate summary table, and example 2 show you how to use PRELOADFMT and COMPLETEROWS for group variables.

## EXAMPLE 1

```
   options missing=0 orientation=landscape nodate nonumber  ;
   data a;
      length pref $25 source $15;
      input soc_name $47. +1 pref $17. +1 unlist $3. +1 serious $3. +1 source $12.;
   datalines;
CARDIAC DISORDERS                               TACHYCARDIA       NO  YES Spontaneous
MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS ARTHRALGIA        NO  YES Spontaneous
MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS PAIN IN EXTREMITY YES YES Spontaneous
MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS MUSCULAR WEAKNESS YES YES Spontaneous
   ;
   run;

   proc format ;
      value $slc 'Spontaneous'='Spontaneous'
                 'Literature'='Literature'
                 'Clinical Trial'='Clinical Trial';
      value $unlist  'YES'='Unlisted'
                     'NO'='Listed';
      value  $serious  'YES'='Serious'
                       'NO'='Non|Serious';
   run;
   ods rtf file="c:\phsug.rtf" ;
   proc report data=a nowd headline headskip  completecols split='|'
      style(header)={background=white protectspecialchars=off}
      style(column)={protectspecialchars=off font_style=roman};
      column  soc_name  pref source,(unlist,serious);
      define soc_name/group  noprint ;
      define pref/ group 'System Organ Class|  Preferred Term';
      define source/' ' across  format=$slc. preloadfmt;
      define unlist/across ' ' format=$unlist. preloadfmt ;
      define serious/' ' across  format=$serious. preloadfmt;
      break after  soc_name / summarize ;
      compute pref/char ;
         if upcase(_break_)='SOC_NAME' then pref='\li300\b Subtotal number';
         else pref='\li300 '||trim(Left(pref));
      endcomp;

      compute before soc_name/style={ font_weight=bold};
         line @1 soc_name $;
      endcomp;
   run;
   ods rtf close;
   option missing=.;
```

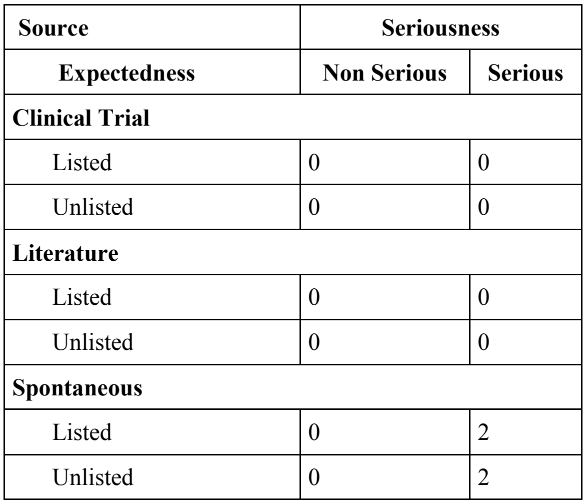
| | **Clinical Trial** | | | | **Literature** | | | | **Spontaneous** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Listed** | | **Unlisted** | | **Listed** | | **Unlisted** | | **Listed** | | **Unlisted** | |
| **System Organ Class**<br>**Preferred Term** | **Non Serious** | **Serious** | **Non Serious** | **Serious** | **Non Serious** | **Serious** | **Non Serious** | **Serious** | **Non Serious** | **Serious** | **Non Serious** | **Serious** |
| **CARDIAC DISORDERS** | | | | | | | | | | | | |
| TACHYCARDIA | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **Subtotal number** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **MUSCULOSKELETAL AND CONNECTIVE TISSUE DISORDERS** | | | | | | | | | | | | |
| ARTHRALGIA | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| MUSCULAR WEAKNESS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PAIN IN EXTREMITY | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| **Subtotal number** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 2 |

In this example, by using PRELOADFMT and COMPLETECOLS options, we created all possible combinations of the across variable values even if not all of them exist in the input data set. Although we can easily add columns with 0 values in PROC REPORT, for those periodical summary reports, we do not know which class level values that the data does not have. In some periods, the database may not have some class levels, but in the next report periods, other class levels may be missed. Moreover, it varies between studies. Therefore, simply adding columns with 0 values in PROC REPORT cannot be used for all periodically reports and all studies. However, with COMPLETECOLS and PRELOADFMT options, it is easy to generate the report in one step, and the program can be re-used without any modification.

COMPLETECOLS option can be omitted here, as it is a default option in PROC REPORT. PROC REPORT by default will create all possible combinations for the values of the across variables, even if one or more of the combinations are not in data set as long as the input data set contains the values of the across variables. The PRELOADFMT option ensures that PROC REPORT use all user defined format ranges for the combinations of the across variables.

## EXAMPLE 2

```
proc format ;
   value $slc 'Spontaneous'='Spontaneous'
              'Literature'='Literature'
              'Clinical Trial'='Clinical Trial';
   value $unlist  'YES'='\li300 Unlisted'
                  'NO'='\li300 Listed';
   value  $serious  'YES'='Serious'
                  'NO'='Non Serious';
run;
ods rtf file="c:\phsug2.rtf" ;
proc report data=a nowd  completerows
   style(header)={ background=white protectspecialchars=off}
   style(column)={protectspecialchars=off };
   column   Source ('\ql Source' unlist)  serious;
   define source/'' group format=$slc. preloadfmt noprint;
   define unlist/'\ql\li300 Expectedness' format=$unlist. group ;
   define serious/'Seriousness' across format=$serious. preloadfmt;
   compute before source/style={ font_weight=bold};
     line @1 source $;
   endcomp;
run;
ods rtf close;
```



| Source | Seriousness | |
|---|---|---|
| Expectedness | Non Serious | Serious |
| **Clinical Trial** | | |
| Listed | 0 | 0 |
| Unlisted | 0 | 0 |
| **Literature** | | |
| Listed | 0 | 0 |
| Unlisted | 0 | 0 |
| **Spontaneous** | | |
| Listed | 0 | 2 |
| Unlisted | 0 | 2 |

With COMPLETETOWS and PRELOADFMT options, all three adverse events reporting sources appear in the summary table even if CLINICAL TRIAL and LITERATURE were not included in the input data. COMPLETEROWS displays all possible combinations of the values of group variables. COMPLETEROWS and PRELOADFMT options in the DEFINE statement ensure that PROC REPORT use all user-defined format ranges for the combinations of group variables, even when the values of group variables are not in the input data set. The option COMPLETEROWS must be specified to include all values of the user defined formats in the output even if some of them are not in the input data set when using PRELOADFMT option in PROC REPORT.

## PRELOADFMT OPTION IN PROC TABLUATE

The PRELOADFMT option is not only a PROC REPORT option, but also can be used in PROC TABLUEATE, PROC MEANS, and PROC SUMMARY. The following example shows you how to apply PRELOADFMT option in PROC TABLUATE.

### EXAMPLE 3

```
data demog;
   input subject race age sex $1. eth;
datalines;
 1 2  6 M 2
 2 4  0 M 4
 3 1 33 F 1
 4 2 38 F 2
 5 4  2 F 4
    ;
 run;

proc format;
   value age   0-17='<18'
               18-29='18-29'
               30-39='30-39'
               40-49='40-49';
   value $sex  'F'='Female'
               'M'='Male';
   Value race  1='Asian'
               2='Black or African American'
               3='White'
               4='American Indian or Alaska Native'
               5='Hispanic or Latino' ;
run;

options missing=0 orientation=landscape nodate nonumber;
ods rtf file="c:\phsug3.rtf" style=minimal;
PROC TABULATE DATA=demog ;
   class sex race age/preloadfmt;
   format sex $sex. race race. age age.;
   table age='' all='Total', sex='Sex'*(n pctn<sex>='Percent(%)')
race='Race'/RTS=13. printmiss BOX='Age' ;
RUN;
ods rtf close;
option missing=.;
```

| Age | Sex | | | | Race | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Female | | Male | | Asian | Black or African American | White | American Indian or Alaska Native | Hispanic or Latino |
| | N | Percent(%) | N | Percent(%) | N | N | N | N | N |
| <18 | 1 | 33.33 | 2 | 66.67 | 0 | 1 | 0 | 2 | 0 |
| 18-29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30-39 | 2 | 100.00 | 0 | 0.00 | 1 | 1 | 0 | 0 | 0 |
| 40-49 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total | 3 | 60.00 | 2 | 40.00 | 1 | 2 | 0 | 2 | 0 |

When PRELOADFMT option is used in the CLASS statement and PRINTMISS option is used in the TABLE statement, the table includes values of user-defined format WHITE and HISPANIC OR LATINO in the column and age groups 18-29 and 40-49 in the row even if these class levels are not included in the input data set.

By default, PROC TABULATE only displays heading for values appearing in the input data set. PRINTMISS option makes it possible to print all values of class variables even if there are no data for some of the combinations of class variables as long as the values of the class variables exist in the input data set. To include all ranges and values of user- defined formats in the summary table, PRELOADFMT must be used with PRINTMISS option in TABULATE procedure.

## PRELOADFMT OPTION IN PROC MEANS AND SUMMARY

Similar to PROC TABLUATE and PROC REPORT, you could analyze class variables in PROC MEANS and PROC SUMMARY with all ranges and values of user-defined formats. The difference is that you must specify COMPLETETYPES option in the PROC MEANS statement or PROC SUMMARY statement and PRELOADFMT in CLASS statement.  Since PROC SUMMARY and PROC MEANS are very similar, we only introduce PRELOADFMT option in PROC MEANS.

### EXAMPLE 4

```
option missing='-';
ods rtf file='c:\file transfer\programdevelop\phsug4.rtf' style=minimal;
proc means data=demog n mean std min max completetypes nonobs maxdec=1;
   class race/preloadfmt  ;
   format race race.;
   var age;
run;
ods rtf close;
option missing=.;
```

| Analysis Variable : age | | | | | |
|---|---|---|---|---|---|
| Race | N | Mean | Std Dev | Minimum | Maximum |
| Asian | 1 | 33.0 | - | 33.0 | 33.0 |
| Black or African American | 2 | 22.0 | 22.6 | 6.0 | 38.0 |
| White | 0 | - | - | - | - |
| American Indian or Alaska Native | 2 | 1.0 | 1.4 | 0.0 | 2.0 |
| Hispanic or Latino | 0 | - | - | - | - |

## USING PRELOADFMT TO PRESERVE USER-DEFINED ORDER

In the examples above, the headings in the rows and columns were displayed according to the alphanumeric order of the variable values. In the following example, we will show you how to use PRELOADFMT option to order the output in user-specified order rather than alphanumeric order.

### EXAMPLE 5

```
options missing=0 orientation=landscape nodate nonumber;
proc format;
    value age(multilabel notsorted)
                0-11='Children'
                12-17='Adolescents'
                18-high='Adult'
                0-17='<18'
                18-29='18-29'
                30-39='30-39'
                40-49='40-49';

    Value  race(notsorted)
                3='White'
```

```
                    2='Black or African American'
                    5='Hispanic or Latino'
                    1='Asian'
                    4='American Indian or Alaska Native';
run;

ods rtf file="c:\phsug5.rtf" style=minimal;
proc tabulate data=demog ;
   class  age/preloadfmt mlf order=data;
   class  race/preloadfmt order=data;
   format race race. age age.;
   table   age='' all='Total', race='Race'
       /RTS=13.  printmiss box='Age' ;
RUN;
ods rtf close;
options missing=.;
```

| Age | Race | | | | |
|---|---|---|---|---|---|
| | White | Black or African American | Hispanic or Latino | Asian | American Indian or Alaska Native |
| | N | N | N | N | N |
| Children | 0 | 1 | 0 | 0 | 2 |
| Adolescents | 0 | 0 | 0 | 0 | 0 |
| Adult | 0 | 1 | 0 | 1 | 0 |
| <18 | 0 | 1 | 0 | 0 | 2 |
| 18-29 | 0 | 0 | 0 | 0 | 0 |
| 30-39 | 0 | 1 | 0 | 1 | 0 |
| 40-49 | 0 | 0 | 0 | 0 | 0 |
| Total | 0 | 2 | 0 | 1 | 2 |

The orders of race and age in this example are displayed in the order of label in the formats, rather than the order of the values. To display the column and row headings in your own specified order, you need to create user-defined formats with the NOTSORTED option, and to specify the options PRELOADFMT and ORDER=DATA in the CLASS statement. The ORDER=DATA option orders values according to their order in the input data set. When NOTSORTED is specified in PROC FORMAT and PRELOADFMT and ORDER=DATA are used together in the CLASS statement, the order of the values of each class variable matches the order that PROC FORMAT uses to store the values of the associated user-defined format. Multilabel format is assigned to the class variable AGE. With the option MLF in the CLASS statement, the age is summarized in two levels of categories. With MULTILABEL and NOTSORTED options in PROC FORMAT and MLF, PRELOADFMT, ORDER=DATA options in the CLASS statement, the user specified order rather than alphanumeric order will be used in the row and column headings.

ORDER=DATA option and PRELOADFMT option can be specified in the CLASS statement in PROC MEANS, PROC SUMMARY, or PROC TABULATE, and in the DEFINE statement in PROC REPORT. Nevertheless, MLF option is only available in PROC MEANS, PROC SUMMARY, and PROC TABULATE.

## USING PRELOADFMT WITH EXCLUSIVE OPTION

When PRELOADFMT and EXCLUSIVE options are specified, all combinations of the class variables that are not found in the user-defined formats will be excluded from the analysis. When the values of class variables exist in data set, this feature has the same results as simply using WHERE statement in those procedures to choose specified class levels. Nevertheless, if the values specified in WHERE statement do not exist in the data set, the program will not generate any output as no observation is selected from the input data set. However, the combination of PRELOADFMT and EXCLUSIVE options will produce a table with zeros and / or missing values for all of the cells.

PRELOADFMT and EXCLUSIVE options are available in PROC MEANS, PROC SUMMARY, PROC REPORT, and PROC TABULATE. However, only PROC MEANS, SUMMARY, and REPORT will generate output when EXCLUSIVE option is specified and the input data do not contain any classes in the user-defined format. PROC TABULATE will give warning message in SAS log and will not generate any output in that case in SAS 9.1.3.

## CLINICAL TRIAL APPLICATIONS

### EXAMPLE 6: USING PRELOADFMT TO GENERATE LABORATORY TEST VALUE SHIFT TABLE

Shift tables show the number of patients who are low, normal, or high at baseline and then at selected time intervals [3]. The following example will show you how to generate a shift table with PRELOADFMT option in PROC REPORT.

```
data temp;
   input subject @4 chemtest $10. @15 baseline $6. @22 final $6. @29 trt;
datalines;
1  Chloride   NORMAL NORMAL 0
2  Creatinine HIGH   NORMAL 1
3  Glucose    NORMAL        1
4  Chloride   NORMAL NORMAL 0
5  Creatinine NORMAL NORMAL 1
6  Glucose    HIGH   NORMAL 1
7  Creatinine NORMAL        0
8  Glucose    HIGH          1
9  Chloride          NORMAL 1
10 Creatinine NORMAL NORMAL 1
11 Glucose    HIGH   NORMAL 0
12 Creatinine HIGH          1
;
run;

option orientation=landscape  ;

proc format;
   value $hl(notsorted)
      'LOW'='Low'
      'NORMAL'='Normal'
      'HIGH'='High'
      ' '='Missing';
   value trt(notsorted)
      1='Active|---'
      0='Placebo|---';

run;
proc report data=temp nowd headline out=a  completerows missing split='|';
   column chemtest  ("Baseline" baseline) trt, final,n;
   define chemtest/'Laboratory Parameter' group flow width=10;
   define baseline/'' width=8 group format=$hl. order=data preloadfmt;
   define trt/'Final|---' format=trt. across order=data preloadfmt;
   define final/across '' format=$hl.   order=data preloadfmt ;
   define n/'' width=7 spacing=1 ;
   break before chemtest/skip;
run;
```

```
                                                  Final
                     ---------------------------------------------------------------
                                 Active                          Placebo
                     ------------------------------- -------------------------------

Laboratory  Baseline     Low  Normal    High Missing     Low  Normal    High Missing
Parameter
------------------------------------------------------------------------------------

Chloride    Low            0       0       0       0       0       0       0       0
            Normal         0       0       0       0       0       2       0       0
            High           0       0       0       0       0       0       0       0
            Missing        0       1       0       0       0       0       0       0

Creatinine  Low            0       0       0       0       0       0       0       0
            Normal         0       2       0       0       0       0       0       1
            High           0       1       0       1       0       0       0       0
            Missing        0       0       0       0       0       0       0       0

Glucose     Low            0       0       0       0       0       0       0       0
            Normal         0       0       0       1       0       0       0       0
            High           0       1       0       1       0       1       0       0
            Missing        0       0       0       0       0       0       0       0
```

The summary number was generated with the ACROSS and GROUP option in PROC REPORT. PRELOADFMT option, and PROC FORMAT option NOTSORTED were used to present the data in the order of Low, Normal, High, and Missing in the columns and rows. In order to include missing value in the output, the MISSING option must be specified in PROC REPORT statement.

## EXAMPLE 7: USING PRELOADFMT TO INSERT BLANK LINE IN DISPOSITION TABLE

A summary table with blank lines which are inserted to separate different categories is easy to read, but it is particularly challenge to create. However, using PRELOADFMT option and user-defined formats can easily insert blank lines in the summary report without much programming involved. This example will use disposition table to show how to use PRELOADFMT option and user-define format to insert blank lines in a summary table.

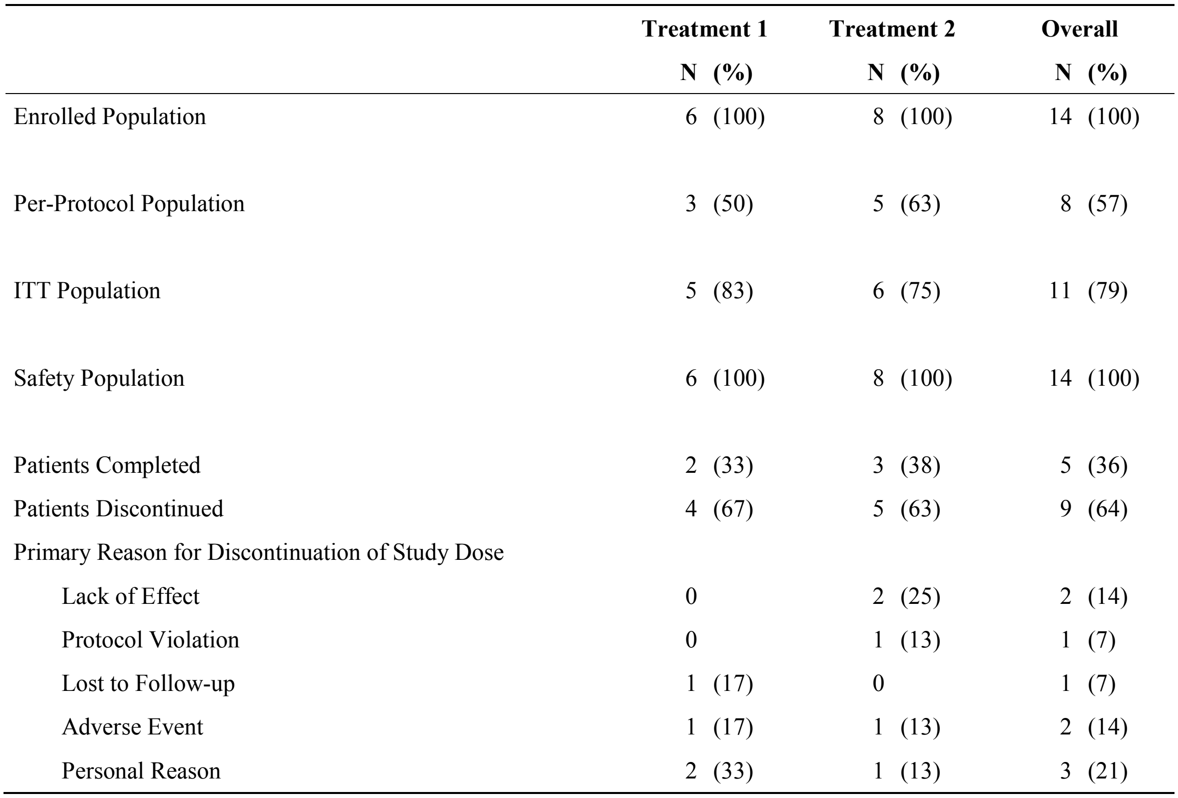
```
data a;
infile datalines missover;
input subjid trt itt safety pp complete enroll disreason;
uncomplete=(complete^=1);
datalines;
1 1 1 1 1 1 1
2 2 0 1 1 1 1
3 1 1 1 1 0 1  4
4 2 1 1 1 1 1
5 1 1 1 0 0 1  5
6 2 1 1 1 1 1
7 2 1 1 1 0 1  1
8 1 0 1 0 1 1
9 1 1 1 1 0 1  5
10 2 0 1 0 0 1 5
11 2 1 1 1 0 1 1
12 2 1 1 0 0 1 2
13 1 1 1 0 0 1 3
14 2 1 1 0 0 1 4
;
run;

data b;
   set a(in=a) a(in=b);
   if b then trt=3;
run;

data c;
   length cat $20;
   set b;
   array a(6) itt safety pp complete uncomplete enroll;
   array b(5) r1 r2 r3 r4 r5;
   do i=1 to 6;
      cat=upcase(vname(a(i)));
      value=a(i);
      output;
   end;
   do j=1 to 5;
      if disreason=j then value=1;
      else value=0;
      cat=upcase(vname(b(j)));
      output;
   end;
   keep trt cat  value;
run;

proc format ;
   value $newcat(notsorted)
         'ENROLL'='Enrolled Population'
         '1'='1'
         'PP'='Per-Protocol Population'
         '2'='2'
         'ITT'='ITT Population'
         '3'='3'
         'SAFETY'='Safety Population'
         '4'='4'
         'COMPLETE'='Patients Completed'
         'UNCOMPLETE'='Patients Discontinued'
         '5'='Primary Reason for Discontinuation of Study Dose'
         'R1'='\li360 Lack of Effect'
         'R2'='\li360 Protocol Violation'
         'R3'='\li360 Lost to Follow-up'
         'R4'='\li360 Adverse Event'
         'R5'='\li360 Personal Reason';
value trt 1='Treatment 1'
          2='Treatment 2'
          3='Overall';
picture pct(round)
          0<-100='0009)'(prefix='(' mult=100)
          0=0;
run;

option missing='' nodate nonumber orientation=landscape;
ods rtf file='c:\dispoistion.rtf';
proc report data=c completerows nowd
   style(report)={frame=hsides rules=groups}
   style(header)={background=white};
   column cat cat2 trt, value, (sum mean);
   define cat/group format=$newcat. preloadfmt order=data noprint;
   define cat2/computed ' ' style(column)={cellwidth=30%  PROTECTSPECIALCHARS=OFF
};
   define trt/across format=trt. '' order=internal;
   define value/analysis '';
   define sum/ 'N' style(column)={cellwidth=35pt} style(header)={just=right};
   define mean/ '(%)' format=pct. style(column)={cellwidth=35pt just=left}
style(header)={just=left};
   compute cat2/char length=50;
      cat2=put(cat, $newcat.);
      if cat2 in ('1', '2', '3','4') then cat2='';
   endcomp;
run;
ods rtf close;
```



| | Treatment 1 | Treatment 2 | Overall |
|---|---|---|---|
| | N (%) | N (%) | N (%) |
| Enrolled Population | 6 (100) | 8 (100) | 14 (100) |
| Per-Protocol Population | 3 (50) | 5 (63) | 8 (57) |
| ITT Population | 5 (83) | 6 (75) | 11 (79) |
| Safety Population | 6 (100) | 8 (100) | 14 (100) |
| Patients Completed | 2 (33) | 3 (38) | 5 (36) |
| Patients Discontinued | 4 (67) | 5 (63) | 9 (64) |
| Primary Reason for Discontinuation of Study Dose | | | |
| Lack of Effect | 0 | 2 (25) | 2 (14) |
| Protocol Violation | 0 | 1 (13) | 1 (7) |
| Lost to Follow-up | 1 (17) | 0 | 1 (7) |
| Adverse Event | 1 (17) | 1 (13) | 2 (14) |
| Personal Reason | 2 (33) | 1 (13) | 3 (21) |

In this example, we inserted additional class levels 1, 2, 3, 4, and 5 in the user-defined format $CAT in which the blank lines and additional category levels should be. The class levels 1, 2, 3, 4, and 5 were summarized in the REPORT procedure but with all missing values for the summary statistics as those class levels were not in the input data when the option PRELOADFMT and user-defined format $CAT were specified in the DEFINE statement. By setting the MISSING system option to a blank and changing 1, 2, 3, and 4 to blank and labeling 5 to 'Primary Reason for Discontinuation of Study Dose', we got the customized disposition report.

## CONCLUSION

In conclusion, PRELOADFMT option is a very powerful tool to generate reports with all possible user-defined levels combination. With this option and user-defined format, it is easy to generate summary tables using PROC MEANS, SUMMARY, REPORT, and TABULATE with all possible combinations of the class variables even if the values of the class variables do not exist in the input data set. Also, if we combine PRELOADFMT option with other appropriate options, the column and row headings can be displayed in the user-defined format order. The two features simplify the programming for producing summary tables.

## REFERENCES

[1] Stacey D. Phillips, Gary Klein. Oh No, a Zero Row: 5 Ways to Summarize Absolutely Nothing, PharmaSUG, 2005.

[2] Base SAS® 9.1.3 Procedures Guide, Second Edition, SAS Institute Inc. 2006.

[3] ICH E3: Structure and Content of Clinical Study Reports

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Suwen Li
Everest Clinical Research Services Inc.
675 Cochrane Drive
Suite 408, East Tower

Markham, Ontario, Canada L3R 0B8
(905) 752-5253
suwen.li@ecrscorp.com