



THE
POWER
TO KNOW.

Technical Paper

Customizing FREQ Procedure Output in SAS® 9.2

*Introducing the new crosstabulation table template for
multi-way tables and SAS ODS Graphics options*

Table of Contents

Introduction	1
The Crosstabulation Template.....	1
Example 1: Changing the Color of Cell Values in Non-Listing Destinations	2
Example 2: Replacing Row, Column, and Table Headings.....	3
Example 3: Replacing Variable Names with Meaningful Labels	4
Example 4: Formatting Cell Contents for Requested Statistics and Changing Statistics Headings in the Legend	6
Example 5: Suppressing a Table's Frequency-Missing Row in All ODS Destinations	7
Example 6: Replacing the Row, Column, and Table Variables with Variable Labels and the ControllingFor String with Customized Text	9
ODS Statistical Graphics.....	10
Conclusion.....	12
Resources	13
Appendix: Complete Crosstabulation Template Definition	14

Introduction

Prior to SAS 9.2, only one alternative was available for modifying the output of a multi-way table from the FREQ procedure (PROC FREQ). You had to use the FORMAT= option in the TABLES statement, which enabled the software to display the frequency count with a different format. For more customization, particularly in the non-listing destinations (such as HTML, PDF, and so on), the TABULATE procedure was the recommended alternative because it has style options available within the procedure syntax.

This paper introduces the new crosstabulation table template, which enables you to customize PROC FREQ output and highlights how to use the template to accomplish some common tasks, such as the following:

- formatting numbers in a table
- changing or suppressing headers and footers
- changing or suppressing the legend
- adding distinct styles to each region of the table

This paper also introduces the new SAS Output Delivery System (ODS) Statistical Graphics capability that is available with PROC FREQ.

The information in this paper is intended for people who understand multi-way frequency tables and who have a basic knowledge of the FREQ procedure and the SAS Output Delivery System.

The Crosstabulation Template

The *crosstabulation template* for PROC FREQ is a TEMPLATE procedure that enables you to customize multi-way frequency tables. The template, written in the standard table template language, is completely defined in “Appendix: Complete Crosstabulation Template Definition.” The examples that follow illustrate how you can use the template to modify specific aspects of a multi-way frequency table.

Note: The default CrossTabFreqs table definition that is shipped with SAS is stored in the SASHELP.TMPLMST itemstore. You can return to that default definition by submitting the following statements:

```
proc template;
  delete base.freq.crosstabfreqs;
run;
```

Example 1: Changing the Color of Cell Values in Non-Listing Destinations

Adding colors to your output is useful when you want to highlight values. The following program changes the color of the frequency cell values in the table cell based on the color assigned in the FORMAT procedure. You can only add color to non-listing destinations such as HTML, PDF, and RTF.

```
proc format;
  value color low-2='green'
             3-5='blue'
             6-high='orange';
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    cell_style={foreground=color.};
    row_total_style={foreground=color.};
    col_total_style={foreground=color.};
    grand_total_style={foreground=color.};
  end;

ods listing close;
ods html file='body.html';

proc freq data=sashelp.class;
  tables age*sex / norow nocol nopercnt;
run;

ods html close;
ods listing;
```

The preceding program generates the following output:

	F	M	Total
11	1	1	2
12	2	3	5
13	2	1	3
14	2	2	4
15	2	2	4
16	0	1	1
Total	9	10	19

Output 1. Cell Colors Changed Based on Format

Example 2: Replacing Row, Column, and Table Headings

By default, PROC FREQ creates a table with specific row, column, and table headings, as shown in this example:

Frequency Percent Row Pct Col Pct	Table of Sex by Age				
	Sex	Age			
		11	12	13	14
F	1	2	2	2	7
	7.14	14.29	14.29	14.29	50.00
	14.29	28.57	28.57	28.57	
	50.00	40.00	66.67	50.00	
M	1	3	1	2	7
	7.14	21.43	7.14	14.29	50.00
	14.29	42.86	14.29	28.57	
	50.00	60.00	33.33	50.00	
Total	2	5	3	4	14
	14.29	35.71	21.43	28.57	100.00

Output 2. Default Table and Headings Generated by PROC FREQ

The following program suppresses the row-variable and the column-variable headings, replaces **Table of Sex by Age** with the header text (**Two-Way Table Output**), and suppresses the table legend in all ODS destinations:

```
proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    define header myheader;
      text 'Two-Way Table Output';
    end;
  end;
run;

ods listing close;
ods html file='body.html';

proc freq data=sashelp.class;
  tables sex*age;
run;
ods html close;
ods listing;
```

The preceding program generates the following output:

Two-Way Table Output							
	11	12	13	14	15	16	Total
F	1	2	2	2	2	0	9
	5.26	10.53	10.53	10.53	10.53	0.00	47.37
	11.11	22.22	22.22	22.22	22.22	0.00	
	50.00	40.00	66.67	50.00	50.00	0.00	
M	1	3	1	2	2	1	10
	5.26	15.79	5.26	10.53	10.53	5.26	52.63
	10.00	30.00	10.00	20.00	20.00	10.00	
	50.00	60.00	33.33	50.00	50.00	100.00	
Total	2	5	3	4	4	1	19
	10.53	26.32	15.79	21.05	21.05	5.26	100.00

Output 3. Suppressed Row and Column Variables and Changed Table Header Text

Example 3: Replacing Variable Names with Meaningful Labels

Rather than using variable names (such as Sex or Age), you might want to customize your output with variable labels that are more meaningful, as shown in the following example:

```
data class;
  set sashelp.class;
  label sex='Gender'
        age='Age in Years';
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    define header tableof;
      text "Table of " _row_label_ " by " _col_label_;
    end;

    define header rowsheader;
      text _row_label_ / _row_label_ ^= ' ';
      text _row_name_;
    end;
  end;
```



```

define header colsheader;
  text _col_label_ / _col_label_ ^= ' ';
  text _col_name_;
end;

cols_header=colsheader;
rows_header=rowsheader;
header tableof;
end;

run;

ods listing close;
ods html file='body.html';

proc freq data=class;
  tables sex*age;
run;

ods html close;
ods listing;

```

This program generates the following output, replacing the variable names Sex and Age with the labels **Gender** and **Age in Years** and suppressing the legend in all ODS destinations. **Note:** In cases where a label is not provided, SAS uses the variable name.

Table of Gender by Age in Years							
Gender	Age in Years						
	11	12	13	14	15	16	Total
F	1	2	2	2	2	0	9
	5.26	10.53	10.53	10.53	10.53	0.00	47.37
	11.11	22.22	22.22	22.22	22.22	0.00	
	50.00	40.00	66.67	50.00	50.00	0.00	
M	1	3	1	2	2	1	10
	5.26	15.79	5.26	10.53	10.53	5.26	52.63
	10.00	30.00	10.00	20.00	20.00	10.00	
	50.00	60.00	33.33	50.00	50.00	100.00	
Total	2	5	3	4	4	1	19
	10.53	26.32	15.79	21.05	21.05	5.26	100.00

Output 4. Replacing Variable Names (Sex and Age) with the Labels **Gender** and **Age in Years**

Example 4: Formatting Cell Contents for Requested Statistics and Changing Statistics Headings in the Legend

The following example changes the names of the default statistics to more meaningful names in the legend for all ODS destinations. The program also formats the percentage statistics (Percent, RowPercent, and ColPercent) with a percent (%) sign.

```
proc format;
  picture pctfmt (round) other='009.9%';
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    cellvalue frequency percent rowpercent colpercent;

    define frequency;
      format=8.;
      header='Count';
    end;

    define percent;
      format=pctfmt.;
      header='Overall %';
    end;

    define rowpercent;
      format=pctfmt.;
      header='Row %';
    end;

    define colpercent;
      format=pctfmt.;
      header='Col %';
    end;
  end;
run;

proc freq data=sashelp.class;
  tables sex*age;
run;
```

The preceding program generates the following table, shown as listing output:

Count							
Overall %							
Row %							
Col %	11	12	13	14	15	16	Total
F	1 5.3%	2 10.5%	2 10.5%	2 10.5%	2 10.5%	0 0.0%	9 47.4%
	11.1%	22.2%	22.2%	22.2%	22.2%	0.0%	
	50.0%	40.0%	66.7%	50.0%	50.0%	0.0%	
M	1 5.3%	3 15.8%	1 5.3%	2 10.5%	2 10.5%	1 5.3%	10 52.6%
	10.0%	30.0%	10.0%	20.0%	20.0%	10.0%	
	50.0%	60.0%	33.3%	50.0%	50.0%	100.0%	
Total	2 10.5%	5 26.3%	3 15.8%	4 21.1%	4 21.1%	1 5.3%	19 100.0%

Output 5. Cell Contents Formatted for Requested Statistics with Changed Legend Headings

Example 5: Suppressing a Table's Frequency Missing Row in All ODS Destinations

The *Frequency Missing* row in a table specifies the number of observations that have missing values. The following table shows the default table output with the Frequency Missing row:

Frequency	Table of x by y		
	x	y	
		1	2
1	1	1	2
2	1	1	2
Total	2	2	4
Frequency Missing = 1			

Output 6. Default Output Containing the Frequency Missing Row

Because the Frequency Missing row specifies observations with missing values, it is not often useful to display it because the table output only shows non-missing values. The following program suppresses the Frequency Missing row from the output table in all ODS destinations by assigning a blank value to TEXT in the DEFINE FOOTER statement:

```

data test;
  input x y;
  cards;
1 1
1 2
2 1
2 2
. .
;
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    define footer foot;
      text ' ' ;
    end;
  end;
run;

ods listing close;
ods html file='body.html';

proc freq data=test;
  tables x*y / norow nocol nopercnt;
run;

ods html close;
ods listing;

```

This program generates the following table:

	1	2	Total
1	1	1	2
2	1	1	2
Total	2	2	4

Output 7. Table with Suppressed Frequency Missing Row

Example 6: Replacing the Row, Column, and Table Variables with Variable Labels and the ControllingFor String with Customized Text

In the following example, which is for a three-way table, the row-variable and the column-variable headings and the default text string (**Table of y by z**) are replaced with the variable labels. The table legend is also suppressed, and the ControllingFor string is replaced with the customized text **ID=a**. These changes affect all ODS destinations.

```

data test;
  label x='ID' y='Row Variable' z='Column Variable';
  input x $ y z;
  cards;
a 1 1
a 1 2
a 2 1
a 2 2
b 1 1
b 1 2
b 2 1
b 2 2
  ;
run;

proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    define header ControllingFor;
      dynamic StratNum StrataVariableNames StrataVariableLabels;
      text StrataVariableLabels / StratNum>0;
    end;

    define header tableof;
      text _row_label_ " by " _col_label_;
    end;
  end;

ods listing close;
ods html file='body.html';

proc freq data=test;
  tables x*y*z;
run;

ods html close;
ods listing;

```

The preceding program generates two tables. The tables are similar, so only one table is shown here.

ID=a			
Row Variable by Column Variable			
	1	2	Total
1	1 25.00 50.00 50.00	1 25.00 50.00 50.00	2 50.00
2	1 25.00 50.00 50.00	1 25.00 50.00 50.00	2 50.00
Total	2 50.00	2 50.00	4 100.00

Output 8. Headers Replaced with Labels and ControllingFor String Replaced with Custom Text

ODS Statistical Graphics

In addition to creating tables, it is now possible to create statistical graphs with PROC FREQ and integrate the graphs with tables using the SAS 9.2 ODS Statistical Graphics (ODS Graphics) functionality. ODS Graphics is an extension of ODS, so you can display your graphs to various ODS destinations.

To enable ODS Graphics, submit the following statement:

```
ods graphics on;
```

After you specify ODS GRAPHICS ON, you can then request specific plots with the PLOTS= option in a TABLES statement. The following table lists the names of graphs that you can generate with PROC FREQ.

ODS Graph Name	Plot Description	TABLES Statement Option
CumFreqPlot	Cumulative frequency plot	One-way table request
DeviationPlot	Deviation plot	CHISQ and one-way table request
FreqPlot	Frequency plot	Any table request
KappaPlot	Kappa plot	AGREE (h x r x r table)
ORPlot	Odds ratio plot	MEASURES or RELRISK (hx2x2 table)
WtKappaPlot	Weighted kappa plot	AGREE (h x r xr table, r > 2)

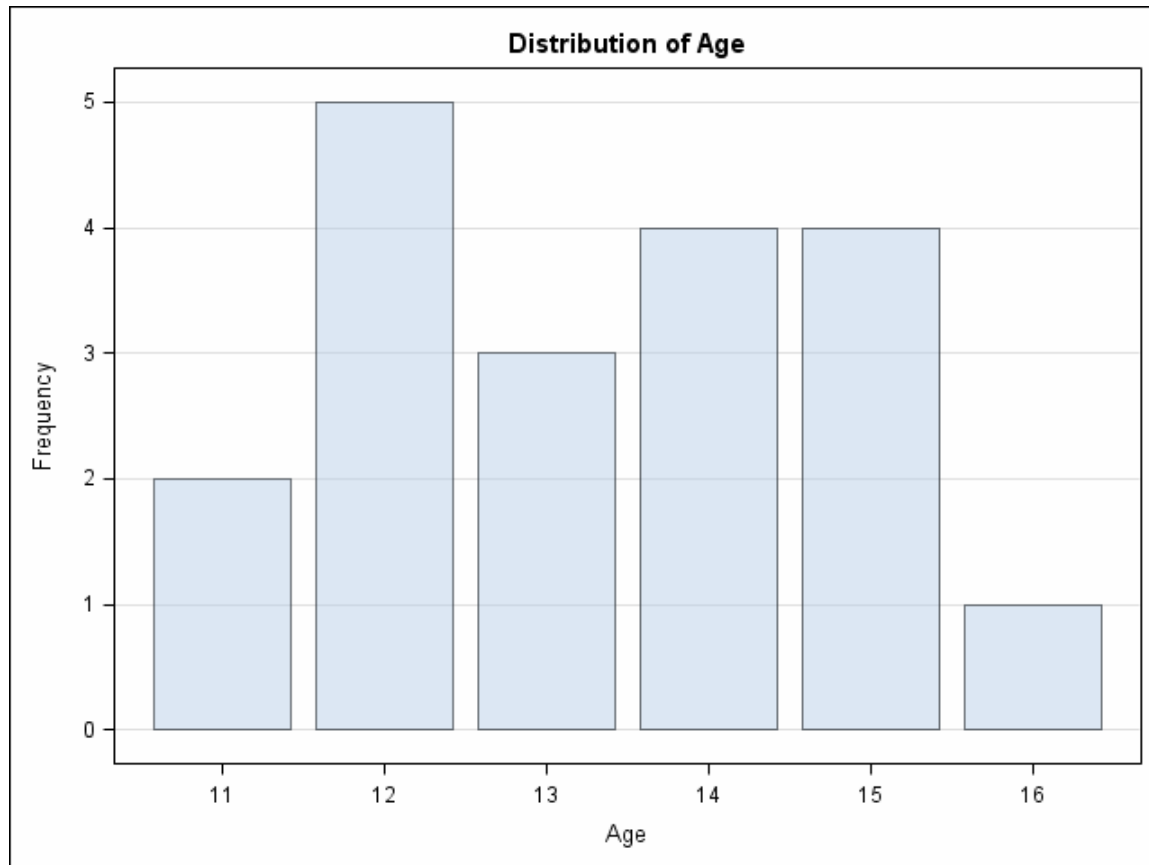
Table 1. Graphs Generated by PROC FREQ

The following program generates a frequency plot within a table:

```
ods graphics on;
proc freq data=sashelp.class;
  tables age / plots=cumfreqplot;
run;
ods graphics off;

ods graphics on;
proc freq data=sashelp.class;
  tables sex*age / plots=freqplot;
run;
ods graphics off;
```

The preceding program generates the following frequency plot:



Output 9. Frequency Plot Generated with the SAS 9.2 ODS Graphics Functionality

Unless you are planning to generate only computational results, you can specify the ODS GRAPHICS ON statement and then just leave it on. However, if you need to turn it off, submit the following statement:

```
ods graphics off;
```

Conclusion

In summary, the new crosstabulation table template for PROC FREQ in SAS 9.2 greatly improves the level of customization that you can apply to multi-way tables. With the new template, you can format, customize, enhance, and exclude various elements of the output table. In addition, the SAS 9.2 ODS Graphics functionality enables you to create statistical graphs with PROC FREQ, integrate the graphs into tables, and display them in various ODS destinations.

Resources

Basic information on the FREQ procedure:

SAS Institute Inc. 2008. *SAS 9.2 Documentation*. Available at support.sas.com/cdlsearch?ct=80000.

Information on the new crosstabulation table template:

SAS Institute Inc. 2008. "Crosstabs Template in PROC FREQ" in *Base SAS: Enhancements to Procedures for SAS 9.2*. Available at support.sas.com/rnd/base/new92/92procs.html#freq.

Questions about the content of this paper:

Send e-mail to support@sas.com.

Appendix: Complete Crosstabulation Template Definition

The complete crosstabulation template definition for the FREQ procedure is as follows:

```
proc template;
  define crosstabs Base.Freq.CrossTabFreqs;
    notes "Crosstabulation table";
    cellvalue Frequency Expected Deviation CellChiSquare TotalPercent
              Percent RowPercent ColPercent CumColPercent;
    header TableOf ControllingFor;
    footer noobs Missing;

  define TableOf;
    dynamic StratNum NoTitle;
    text "Table " StratNum 10. " of " _ROW_NAME_ " by " _COL_NAME_ /
        (NoTitle=0) and (StratNum>0);
    text "Table of " _ROW_NAME_ " by " _COL_NAME_ / NoTitle=0;
  end;

  define ControllingFor;
    dynamic StratNum StrataVariableNames StrataVariableLabels;
    text "Controlling for" StrataVariableNames / StratNum>0;
  end;

  define header RowsHeader / nolist;
    text _ROW_NAME_ "(" _ROW_LABEL_ ")" / _ROW_LABEL_ not = '';
    text _ROW_NAME_;
    cindent = ";";
    space=0;
  end;

  define header ColsHeader / nolist;
    text _COL_NAME_ "(" _COL_LABEL_ ")" / _COL_LABEL_ not = '';
    text _COL_NAME_;
    cindent = ";";
    space=1;
  end;

  define Missing;
    dynamic FMissing;
    text "Frequency Missing = " FMissing -12.99 / FMissing not = 0;
    space=1;
  end;

  define noobs;
    dynamic SampleSize;
    text "Effective Sample Size = 0" / SampleSize=0;
    space=1;
  end;
end;
```

```
define Frequency;
  header="Frequency";
  format=BEST7.;
  label="Frequency Count";
  print;
  data_format_override;
end;

define Expected;
  header="Expected";
  format=BEST6.;
  label="Expected Frequency";
  print;
  data_format_override;
end;

define Deviation;
  header="Deviation";
  format=BEST6.;
  label="Deviation from Expected Frequency";
  print;
  data_format_override;
end;

define CellChiSquare;
  header="Cell Chi-Square";
  format=BEST6.;
  label="Cell Chi-Square";
  print;
end;

define TotalPercent;
  header="Tot Pct";
  format=6.2;
  label="Percent of Total Frequency";
  print;
end;

define Percent;
  header="Percent";
  format=6.2;
  label="Percent of Two-Way Table Frequency";
  print;
end;

define RowPercent;
  header="Row Pct";
  format=6.2;
  label="Percent of Row Frequency";
  print;
end;

define ColPercent;
  header="Col Pct";
  format=6.2;
  label="Percent of Column Frequency";
  print;
end;
```

```
define CumColPercent;  
  header=%nrstr("Cumulative Col%%");  
  format=6.2;  
  label="Cumulative Percent of Column Frequency";  
  print;  
end;  
  
cols_header=ColsHeader;  
rows_header=RowsHeader;  
end;  
run;
```