The SAS® Programmer’s PROC REPORT Handbook
Basic to Advanced Reporting Techniques

Jane Eslinger
Contents

About this Book ........................................................................................................ ix
About the Author ................................................................................................... xiii
Acknowledgments ............................................................................................... xv
Preface .................................................................................................................. xvii
Chapter 1: Syntax – How to Use Statements and Their Options .................. 1
  1.1 Introduction ................................................................................................. 1
  1.2 PROC REPORT Statement ....................................................................... 3
    1.2.1 General Options .................................................................................. 3
    1.2.2 Report Contents Options .................................................................. 4
    1.2.3 Report Appearance Options ............................................................ 5
  1.3 COLUMN Statement ................................................................................... 6
  1.4 DEFINE Statement .................................................................................... 11
    1.4.1 Usage Options ................................................................................... 11
    1.4.2 Interaction Options ........................................................................... 12
    1.4.3 Appearance Options ........................................................................ 13
    1.4.4 Utility Options ................................................................................... 14
  1.5 BREAK Statement ..................................................................................... 15
  1.6 RBREAK Statement .................................................................................. 16
  1.7 COMPUTE Statement ............................................................................... 18
  1.8 ENDCOMP Statement ............................................................................. 18
  1.9 CALL DEFINE Statement ....................................................................... 18
    1.9.1 Argument 1: Column-ID ................................................................. 18
    1.9.2 Argument 2: Attribute Name ........................................................... 19
    1.9.3 Argument 3: Attribute Value ........................................................... 19
  1.10 LINE Statement ...................................................................................... 20
  1.11 Global Statements ................................................................................... 21
# Chapter 2: Concepts – How PROC REPORT Works Behind the Scenes

## 2.1 Introduction

23

## 2.2 General Execution

24

## 2.3 Compute Blocks

25

### 2.3.1 COMPUTE Statement

25

### 2.3.2 Execution of Compute Blocks

26

## 2.4 Referencing Report-items

29

## 2.5 Left to Right Availability

31

## 2.6 Repeating GROUP or ORDER Variable Values

38

### 2.6.1 Character Variables

38

### 2.6.2 Numeric Variables

39

### 2.6.3 Undesired Repeating Values

41

## 2.7 Column Widths

44

## 2.8 Date Variables

46

## 2.9 Temporary Variables

47

## 2.10 Sorting and the ORDER= Option

47

## 2.11 Paging

50

# Chapter 3: Examples – How to Get the Desired Report

## 3.1 Introduction

55

## 3.2 Standard Reports

56

### 3.2.1 Create a Basic Report

56

### 3.2.2 Define a Variable as ORDER

58

### 3.2.3 Define a Variable as GROUP

59

### 3.2.4 Create a New Report-Item

64

### 3.2.5 Produce Summary Rows

70

## 3.3 Nonstandard Reports

80

### 3.3.1 Calculate Percentages within Groups

80

### 3.3.2 Customized Sort Order

82

### 3.3.3 Multiple Summary Rows at One Location

84

### 3.3.4 Rows Created with a LINE Statement Versus a BREAK Statement

90

### 3.3.5 Conditionally Display a LINE Statement

91

## 3.4 Special Data Consideration Reports

93

### 3.4.1 Wide Tables

93

### 3.4.2 Using ORDER=DATA

95

### 3.4.3 Using the COMPLETEROWS Option

97
3.4.4 Output a Table with No Data

3.4.5 Dynamically Assign Spanning Header Text

Chapter 4: Examples – How to Use ACROSS Variables

4.1 Introduction

4.2 Standard Reports

4.2.1 Stack a Statistic

4.2.2 Multiple Variables under an ACROSS Variable

4.2.3 A DISPLAY Variable under an ACROSS Variable

4.2.4 A GROUP Variable under an ACROSS Variable

4.2.5 Create New Report-Items

4.2.6 Percentages

4.3 Header Section Rows

4.3.1 Default Header Section Created with ACROSS

4.3.2 Place a Spanning Header beside ACROSS Values

4.3.3 Place Spanning Headers beside ACROSS Label

4.3.4 Place Multiple Spanning Headers beside ACROSS Header Rows

4.3.5 Remove ACROSS Label Row

4.3.6 Counts As Part of ACROSS Values

4.4 Nonstandard Reports

4.4.1 Customized Sort Order

4.4.2 Creating Subtotal Columns

4.4.3 Nesting ACROSS Variables

4.4.4 Hide a Column under an ACROSS

4.4.5 Vertical Page Breaks

4.4.6 Use Macro to Create Column References

Chapter 5: Examples – How to Determine When to Pre-Process the Data

5.1 Introduction

5.2 Sort by Statistic

5.2.1 Grouped Report, No ACROSS Variable

5.2.2 Grouped Reports with an ACROSS Variable

5.3 Use COMPLETEROWS with ACROSS

5.3.1 COMPLETEROWS and a GROUP Variable under an ACROSS Variable

5.3.2 COMPLETEROWS and a DISPLAY Variable under an ACROSS

5.4 Use COMPLETEROWS with Multiple GROUP Variables

5.5 Use Information from a Variable in Header and Data Sections
Contents

5.6 Wrap Text at a Specific Place ......................................................................................... 169
5.7 Incorporate Various Data Pieces into One Report .......................................................... 171
  5.7.1 Combine Detail and Summary Information ............................................................. 171
  5.7.2 The Importance of ORDER Variables ....................................................................... 174
5.8 Create the Look of Merged Cells .................................................................................... 179
  5.8.1 Merge Vertically .......................................................................................................... 179
  5.8.2 Merge Horizontally ..................................................................................................... 180

Chapter 6: Styles – How to Change a Report’s Appearance ........................................... 185

  6.1 Introduction ...................................................................................................................... 186
  6.2 STYLE= Option ................................................................................................................ 186
  6.3 Borders ............................................................................................................................. 188
    6.3.1 FRAME= and RULES= Attributes .......................................................................... 189
    6.3.2 Column (Data) Borders ........................................................................................... 191
    6.3.3 Header Borders ........................................................................................................ 195
  6.4 Trafficlighting ................................................................................................................... 199
    6.4.1 Color Based on Cell’s Value ................................................................................... 199
    6.4.2 Color Based on Another Report-item .................................................................... 200
    6.4.3 Color on the Diagonal .............................................................................................. 204
  6.5 Trafficlighting under an ACROSS Variable .................................................................... 206
    6.5.1 Color Based on Cell’s Value ................................................................................... 206
    6.5.2 Color Based on Another Report-item .................................................................... 207
    6.5.3 Color on the Diagonal .............................................................................................. 211
  6.6 Color in Headers .............................................................................................................. 213
    6.6.1 STYLE(HEADER)= on PROC REPORT Statement ............................................... 213
    6.6.2 STYLE(HEADER)= on DEFINE Statement ............................................................. 213
    6.6.3 Spanning Header ....................................................................................................... 214
    6.6.4 ACROSS Variable Label and Value Headers ........................................................ 215
    6.6.5 ACROSS Value Headers ......................................................................................... 215
    6.6.6 ACROSS Label versus ACROSS Values ............................................................... 216
  6.7 LINE Statements ............................................................................................................ 217
  6.8 Advanced Color and Border Assignments ..................................................................... 220
    6.8.1 Apply Multiple Styles on One Cell ......................................................................... 220
    6.8.2 Color Every Other Row ......................................................................................... 225
    6.8.3 Change Borders in HTML Output .......................................................................... 226
    6.8.4 Special Instructions for the ODS Destination ....................................................... 228
6.9 Images ................................................................................................................................... 229
  6.9.1 Place an Image above or below a Report Table ..................................................... 230
  6.9.2 Place an Image inside of the Table ........................................................................... 231
6.10 URLs .................................................................................................................................... 235
  6.10.1 Hyperlink to a Static Location ................................................................................. 236
  6.10.2 Link to Numerous Files ............................................................................................ 237

Chapter 7: Table of Contents – How to Manipulate with CONTENTS= and
PROC DOCUMENT ................................................................................................. 241
7.1 Introduction ......................................................................................................................... 241
7.2 Default Nodes .................................................................................................................... 241
7.3 BY-Variable Nodes ............................................................................................................. 243
7.4 DEFINE Statement CONTENTS= .................................................................................... 246
7.5 BREAK Statement CONTENTS= ...................................................................................... 249
7.6 RBREAK Statement CONTENTS= .................................................................................. 251
7.7 ODS DOCUMENT and PROC DOCUMENT ..................................................................... 252
  7.7.1 ODS DOCUMENT Convention ................................................................................. 252
  7.7.2 PROC DOCUMENT Statements and Options .......................................................... 253
  7.7.3 Default Items ............................................................................................................... 253
  7.7.4 BY-Variable Item List ................................................................................................. 255
  7.7.5 Parent Node with Multiple Child Nodes ................................................................... 256
  7.7.6 Combine Multiple PROC REPORT Steps under One Node .................................... 262

Chapter 8: Debugging Techniques – How to Troubleshoot ........................................ 267
8.1 Introduction ......................................................................................................................... 267
8.2 Errors, Warnings, and Notes in the Log ........................................................................... 267
  8.2.1 DEFINE Statement ...................................................................................................... 267
  8.2.2 BREAK Statement ...................................................................................................... 271
  8.2.3 Compute Block Statements ...................................................................................... 272
8.3 Temporary Variable Values .............................................................................................. 275
  8.3.1 Output via a LINE Statement ..................................................................................... 275
  8.3.2 Output via a COMPUTED Variable ............................................................................ 276
8.4 General Tips ....................................................................................................................... 278
References .............................................................................................................................. 281
Books ...................................................................................................................................... 281
Conference Papers ................................................................................................................. 281
Technical Paper ....................................................................................................................... 281
Chapter 8: Debugging Techniques – How to Troubleshoot

8.1 Introduction .............................................................................................................267

8.2 Errors, Warnings, and Notes in the Log ...............................................................267
  8.2.1 DEFINE Statement .......................................................................................... 267
  8.2.2 BREAK Statement ..........................................................................................271
  8.2.3 Compute Block Statements............................................................................. 272

8.3 Temporary Variable Values ...................................................................................275
  8.3.1 Output via a LINE Statement ........................................................................ 275
  8.3.2 Output via a COMPUTED Variable ............................................................... 276

8.4 General Tips .........................................................................................................278

8.1 Introduction
No matter how well you know your data or how careful you are when writing your PROC REPORT code, something might still go wrong. This chapter discusses errors, warnings, and notes that PROC REPORT will generate if something is wrong. It also demonstrates how you can view the value of temporary variables to ensure they contain the value that you expect. Finally, the chapter provides general debugging tips.

8.2 Errors, Warnings, and Notes in the Log
This section includes a description of various errors, warnings, and notes that PROC REPORT might generate. The focus is on PROC REPORT specific messages. Messages for global statements or common syntax errors are not discussed. Also, this is not an exhaustive list of all possible messages PROC REPORT might generate, but it does cover frequently encountered messages. The messages are categorized by which statement generates the error.

8.2.1 DEFINE Statement
The following messages are written to the log based on issues with one or more DEFINE statements.

ERROR: XXXX conflicts with earlier use of XXXX.
PROC REPORT does not allow two different usages for the same report-item. For example, a variable cannot be used as both GROUP and ACROSS. This error is most often generated
when an alias is created on the COLUMN statement, but defined with another usage. An alias is most useful when you have an ANALYSIS variable that you want multiple statistics for, such as mean, minimum, maximum, or when you want to use a variable twice in the same manner but formatted two different ways.

Solution:

The workaround for this error is to create a duplicate variable on the input data set. The new variable contains the same information as the original variable but can defined with any usage in the PROC REPORT step.

**ERROR: The width of XXXX is not between 1 and NNN. Adjust the column width or line size.**
This error message is generated when the WIDTH= option or the length of a report-item is longer than the LINESIZE system option. This only affects the ODS Listing destination. If multiple destinations are open, the report will be successfully created in the other destinations.

Solution:

To avoid this error, close the ODS Listing destination if it is not needed. Otherwise, specify WIDTH= on a DEFINE statement for the report-item generating the error and set it to a value less than the value of LINESIZE. Also, increase the LINESIZE= system option if it is not at the highest possible value.

**ERROR: There is more than one ANALYSIS usage associated with the column defined by the following elements.**
A comma in the COLUMN statements means that you intend to stack columns. The error is generated when a comma is present but no variables have been defined with a usage of ACROSS.

Solution:

To eliminate the error remove the comma from the COLUMN statement or change the usage to ACROSS for one of the report-items next to the comma.

**ERROR: There is no statistic associated with XXXX**
When there is a DISPLAY under an ACROSS, there needs to be a statistic associated with it.

Solution:

The DISPLAY usage should be changed to GROUP or the N statistic inserted after the ACROSS grouping. See Chapter 4 for a more detailed description of how to use variables under ACROSS variables.
ERROR: A DISPLAY or GROUP variable above or below an ACROSS variable requires that there be an ORDER, GROUP, or DISPLAY variable in the report that is not above or below an ACROSS variable.

As indicated by the error message, a GROUP variable needs to be in the report but not under the ACROSS when a GROUP or DISPLAY variable is under the ACROSS.

Solution:

If you do not already have a suitable variable, you need to create a grouping variable in a DATA step to place before the ACROSS variable on the COLUMN statement in your PROC REPORT step. You can define it as NOPRINT so that it will not be displayed in the table.

ERROR: An ORDER variable appears above or below other report items.

An ACROSS variable cannot share a column with an ORDER variable.

Solution:

The ORDER usage should be changed to GROUP. Please note that another GROUP variable needs to exist that is not under the ACROSS.

ERROR: XXXX is not an ORDER, GROUP, or ACROSS variable and is marked DESCENDING.

The DESCENDING option is only valid for ORDER, GROUP, or ACROSS variables. This error will be generated if the option is placed on a DEFINE statement for a DISPLAY, ANALYSIS, or COMPUTED variable.

Solution:

Remove the DESCENDING option to eliminate the error.

ERROR: You cannot have a GROUP variable stacked with an ACROSS variable when there is a DISPLAY variable by itself in a separate column.

PROC REPORT has some restrictions when an ACROSS variable is used. One such restriction is that you cannot have a DISPLAY variable that is not under the ACROSS when a GROUP variable is under the ACROSS.

Solution:

Change the DISPLAY variable to GROUP to avoid this error.

ERROR 180-322: Statement is not valid or it is used out of proper order.

This is a generic error that can be generated by a number of statements. One common reason this error might be generated inside of PROC REPORT is because an invalid style attribute is placed within the STLYE(<LOCATION(s)>)= option. This error can also be generated by a CALL DEFINE statement within a compute block.

Solution:

Check the STYLE= statement or the style specification with the CALL DEFINE statement. Make sure the statement contains a valid style attribute.
ERROR 79-322: Expecting a (.
ERROR 200-322: The symbol is not recognized and will be ignored.
ERROR 76-322: Syntax error, statement will be ignored.
Again, this error might be caused for a number of reasons. This error can also be generated by a CALL DEFINE statement within a compute block.

Solution:

When this error is generated by a style override, it is mostly likely because the attribute value is not valid for the attribute name. For example, fontstyle=bold will generate the error because ‘bold’ is not a valid value for fontstyle.

WARNING: XXXX is not in the report definition.
This warning is generated by a DEFINE statement that references a report-item that is not on the COLUMN statement.

Solution:

Be sure the report-item on the DEFINE statement is spelled correctly. Otherwise, add the report-item to the COLUMN statement or remove the offending DEFINE statement.

WARNING: The PRELOADFMT option is valid only with GROUP and ACROSS variables. PRELOADFMT will have no effect for the variable XXXX.
As the warning indicates, the PRELOADFMT option is only valid for certain usage values. The message is generated if the DEFINE statement does not contain one of these usages.

Solution:

To eliminate the message, remove the PRELOADFMT option from the DEFINE statement or change the usage to GROUP or ACROSS.

WARNING: PRELOADFMT will have no effect on the output of variable XXXX without one of the following options: "COMPLETEROWS", "ORDER=DATA", or the define option "EXCLUSIVE".
PRELOADFMT must be used in conjunction with one of three other options. If at least one of those options is not also specified on the DEFINE statement, this warning message will be generated.

Solution:

To eliminate the message, remove the PRELOADFMT option from the DEFINE statement or add one of the other options listed in the message.
**WARNING:** The MLF option is valid only with GROUP and ACROSS variables. MLF will have no effect for the variable XXXX.

As the warning indicates, the MLF option is only valid for certain usage values. The message is generated if the DEFINE statement does not contain one of these usages.

Solution:

To eliminate the message, remove the MLF option from the DEFINE statement or change the usage to GROUP or ACROSS.

**WARNING:** A GROUP, ORDER, or ACROSS variable is missing on every observation.

PROC REPORT will issue this warning when, as it says, a GROUP/ORDER/ACROSS variable has a missing value on every observation of the input data set. PROC REPORT will issue the warning, but will not generate a table when this situation occurs.

Solution:

If a missing value is valid, then add the MISSING option to the PROC REPORT statement or the DEFINE statement for the offending grouping variable.

**NOTE:** Groups are not created because the usage of XXX is DISPLAY. To avoid this note, change all GROUP variables to ORDER variables.

By default, a character variable is defined as a DISPLAY. DISPLAY means that every row from the input data set will be printed. However, a GROUP variable is also defined in the PROC REPORT code. GROUP, by definition, means to consolidate the values to the lowest common level. When there is a DISPLAY and a GROUP in the code, PROC REPORT will treat GROUP as ORDER and issue this note.

Solution:

Changing the usage from GROUP to ORDER will eliminate the note.

---

**8.2.2 BREAK Statement**

These errors and warnings might be generated by the BREAK statement.

**ERROR:** You can only BREAK on GROUPing and ORDERing variables.

The variable listed on the BREAK statement is not defined as GROUP or ORDER.

Solution:

Remove the BREAK statement or change the usage on the DEFINE statement of that variable to GROUP or ORDER.
ERROR: The BREAK variable XXXX is not one of the GROUP or ORDER variables.
This error is generated when an alias is created on the COLUMN statement and the alias is listed on a BREAK statement. PROC REPORT cannot have multiple summary rows on the same variable or location. PROC REPORT considers the alias as the same variable that it copies.

Solution:

The workaround for this error is to create a duplicate variable on the input data set. The new variable contains the same information as the original variable, but can be used in any way in the PROC REPORT step.

WARNING: The CONTENTS option will have no effect for variable XXXX because the PAGE option is not specified.
As the warning indicates, the CONTENTS= option must be paired with the PAGE option on a BREAK statement.

Solution:

Either add the PAGE option or remove the CONTENTS= option to eliminate this warning.

8.2.3 Compute Block Statements
The messages below, as well as statements that refer to report-items within the compute block, are generated by the COMPUTE statement. This section does not include all errors that could be generated by DATA step code within the compute block.

ERROR: Missing an ENDCOMP statement.
A COMPUTE statement requires an ENDCOMP statement; this message is generated if that statement is missing.

Solution:

Add an ENDCOMP statement.

ERROR: There are multiple COMPUTE statements for XXXX.
Only one compute block is allowed for each report-item.

Solution:

Consolidate the statements from both blocks into one.

ERROR: There are multiple COMPUTE statements for BREAK AFTER XXXX.
Only one compute block is allowed for each location target pair.

Solution:

Consolidate the statements from both blocks into one.
ERROR 22-322: Syntax error, expecting one of the following: a name, AFTER, BEFORE.
A COMPUTE statement contains only the compute keyword and the semicolon. It does not contain a report-item or a location.

Solution:
Add a report-item or a location to the COMPUTE statement.

ERROR: The variable type of XXXX.SUM is invalid in this context.
ERROR: Illegal reference to the array XXXX.SUM.
These two error messages generated together can be caused by three different circumstances.

1. An ANALYSIS variable under an ACROSS is referred to by compound name rather than column number, in the form _cn_.
2. An alias is referred to by compound name.
3. An ANALYSIS variable’s name is spelled incorrectly on the right side of the equal sign in an assignment statement.

Solution:
Confirm the usage on the DEFINE statement for the XXXX variable and change the reference to the one that is appropriate for that usage. Also, make sure the variable name is spelled correctly.

ERROR: XXXX must use a character format
This occurs when a variable is used on a LINE statement and no format is specified after it. On a LINE statement, a format must be specified for each item (variable).

Solution:
Place a format behind the variable on the LINE statement.

ERROR 22-322: Syntax error, expecting one of the following: a name, a format name
This is a common error that can be generated for any number of reasons, especially when there is a problem with a CALL DEFINE statement.

Solution:
Check that all of the attributes are named correctly. Also, if a format has been specified within a STYLE argument in the CALL DEFINE statement, the error might be generated when a data value falls outside of the range of the format. Finally, be sure a space is placed between each attribute, especially if the statement wraps to another program line.
ERROR: PAGESIZE is too small for BREAK.
This error is generated when PROC REPORT does not have enough space to print all of the information for summary rows and LINE rows on one page. PROC REPORT must keep the LINE statements together and will not split across the page. This error is only generated when sending to the ODS Listing destination.

Solution:
To eliminate the error, close the destination or increase the PAGESIZE value.

ERROR: Invalid column specification in CALL DEFINE.
A variable is referenced in a CALL DEFINE statement that is not on the COLUMN statement. The error might also be generated if the column number used as the first argument to the CALL DEFINE statement does not exist in the table.

Solution:
The CALL DEFINE statement should be removed or the report-item should be added to the COLUMN statement.

ERROR: LINE statements must appear in a COMPUTE block that is associated with a location in the report.
This error message is generated if a LINE statement is inside of a compute report-item block. LINE statements can be used only in compute blocks associated with a location.

Solution:
The statement must be removed from the report-item block or a location needs to be added to the COMPUTE statement.

NOTE: Variable XXXX is uninitialized.
Variable XXXX is on the right side of the equal sign of an assignment statement, but the variable does not exist. It is not a GROUP/ORDER/DISPAY variable nor a previously defined temporary variable. This error might also be generated by ANALYSIS variables that are not properly referred to by their compound name.

Solution:
To eliminate the note, remove the offending variable, create it as a temporary variable prior to its use on an assignment statement, or change the reference to a compound name.
8.3 Temporary Variable Values

As mentioned in Chapter 2, temporary variables are created within compute blocks, but they do not exist on the input data set and are not part of the final report or the output data set. Their values are retained until overwritten with another assignment statement, but it requires extra work to see the values as PROC REPORT builds a report. You cannot use a PUT statement within a compute block. Therefore, you have to use another method to see the value of temporary variables.

There are two methods for seeing the value of a temporary variable. The method that you use depends on how often the value of the temporary variable changes. Temporary variables created in compute blocks executed at certain locations (that is, BEFORE or AFTER a grouping variable), usually do not change as often as temporary variables created in compute report-item blocks.

8.3.1 Output via a LINE Statement

The first method for seeing the temporary variable values is to use a LINE statement. This method is truly useful only for temporary variables that change as the value of a GROUP or ORDER variable changes. It works just like it would if you output a variable from the COLUMN statement. Generating the LINE statement does not have to be a permanent part of your PROC REPORT step. You can use it for troubleshooting and then remove the code. You can output the LINE statement in the block where it was created, or output it from another block referencing the other location.

Chapter 3 contains an example of calculating percentages for each value of CUSTOMER_COUNTRY. Recall calculating group percentages requires creating a temporary variable to hold the denominator value. Let’s revisit that code to demonstrate using a LINE statement to check the value of the temporary variable.

It is often very helpful to add text in the LINE statement prior to the variable name to remind yourself what you are looking at in the final report. Also, text is helpful if you have multiple LINE statements, because it might be confusing as to which one you wrote for debugging purposes. Example 8.1 outputs a LINE statement with the value of the temporary variable and text to draw attention to that row in the final report. The result is shown in Output 8.1.

Example 8.1: Use a LINE Statement to View Temporary Variable Values

```plaintext
proc report data=orders;
    column customer_country order_type total_retail_price pct;
    define customer_country / group format=$cntry.;
    define order_type / group format=typef.;
    define total_retail_price / 'Total Retail Price';
    define pct / 'computed' format=percent8.1 'Percent Retail Price';

    compute before customer_country;
        den = total_retail_price.sum;
    endcomp;

    compute after customer_country;
        line 'the denominator used was: ' den 8.2;
    endcomp;
```
compute pct;
    if den > 0 then pct = total_retail_price.sum / den;
endcomp;
run;

On the LINE statement, place helpful text along with the name of the temporary variable and an appropriate format.

Output 8.1: LINE Statement Contains Temporary Variable Values for Each Country

<table>
<thead>
<tr>
<th>Customer Country</th>
<th>Order Type</th>
<th>Total Retail Price</th>
<th>Percent Retail Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australia</td>
<td>Catalog Sale</td>
<td>$1,679.40</td>
<td>9.7%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$613.90</td>
<td>3.5%</td>
</tr>
<tr>
<td></td>
<td>Retail Sale</td>
<td>$15,028.19</td>
<td>86.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 17321.49</td>
</tr>
<tr>
<td>Canada</td>
<td>Catalog Sale</td>
<td>$5,422.38</td>
<td>45.4%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$6,528.70</td>
<td>54.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 11951.08</td>
</tr>
<tr>
<td>Germany</td>
<td>Catalog Sale</td>
<td>$10,034.40</td>
<td>65.2%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$5,360.20</td>
<td>34.8%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 15394.60</td>
</tr>
<tr>
<td>Israel</td>
<td>Catalog Sale</td>
<td>$1,316.10</td>
<td>84.4%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$243.40</td>
<td>15.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 1559.50</td>
</tr>
<tr>
<td>South Africa</td>
<td>Catalog Sale</td>
<td>$3,161.70</td>
<td>61.4%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$1,988.20</td>
<td>38.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 5149.90</td>
</tr>
<tr>
<td>Turkey</td>
<td>Catalog Sale</td>
<td>$4,690.20</td>
<td>90.6%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$485.60</td>
<td>9.4%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 5175.80</td>
</tr>
<tr>
<td>United States</td>
<td>Catalog Sale</td>
<td>$7,627.17</td>
<td>17.5%</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$6,271.55</td>
<td>14.4%</td>
</tr>
<tr>
<td></td>
<td>Retail Sale</td>
<td>$29,626.38</td>
<td>68.1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the denominator used was: 43525.10</td>
</tr>
</tbody>
</table>

8.3.2 Output via a COMPUTED Variable

The second method for seeing the values of a temporary variable is to assign them to a COMPUTED variable. This method works best if the value changes frequently, such as on every row. It is especially helpful when checking a temporary variable that is keeping a running total.
Chapter 3 contains an example of showing summary values for nested groups. The example requires temporary variables to hold a running tally of QUANTITY for each value of ORDER_TYPE. Let’s revisit the example, but modify it slightly so that the temporary variables change more frequently.

A new report-item has to be placed on the COLUMN statement. The best place to put the new report-item is at the end of the COLUMN statement so that it does not affect the creation of any of your other columns. A DEFINE statement and a compute block for this new report-item are needed. Again, having a COMPUTED column does not have to be a permanent part of your PROC REPORT step. You can use it for troubleshooting and then remove the column from the code or add the NOPRINT option to prevent the column from appearing in the final report. Example 8.2 and Output 8.2 demonstrate including the COMPUTED variable.

Example 8.2: Use a COMPUTED Variable to View Temporary Variable Values

```sas
data orders2;
  set orders;
  dummy1 = 1;
  dummy2 = 1;
  dummy3 = 1;
run;

proc report data=orders2;
  column dummy1 dummy2 dummy3 customer_group order_type
       total_retail_price quantity discount seetempvar; ❶
  define dummy1 / group noprint;
  define dummy2 / group noprint;
  define dummy3 / group noprint;
  define customer_group / group;
  define order_type / group format=typef. order=internal;
  define seetempvar / computed; ❷
  break after dummy1 /summarize;
  break after dummy2 /summarize;
  break after dummy3 /summarize;
  compute discount;
    if order_type in (1 2) then do;
      qnt1 + quantity.sum;
    end;
  endcomp;
  compute seetempvar; ❸
    seetempvar = qnt1; ❹
  endcomp;
run;
```

❶ Add SEETEMPVAR to the COLUMN statement. This report-item will hold the value of the temporary variable created in a compute block.

❷ Define SEETEMPVAR as COMPUTED.
A compute block is needed for the COMPUTED report-item. This compute block will execute on every row. It executes after the block where the temporary variable is assigned a value.

The SEETEMPVAR is assigned the current value of the temporary variable. Its value can now be seen in the final report.

### Output 8.2: A COMPUTED Variable Contains the Value of the Temporary Variable

<table>
<thead>
<tr>
<th>Customer Group Name</th>
<th>Order Type</th>
<th>Total Retail Price for This Product</th>
<th>Quantity Ordered</th>
<th>Discount in percent of Normal Total Retail Price</th>
<th>seetempvar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet/Catalog Customers</td>
<td>Catalog Sale</td>
<td>$11,216.30</td>
<td>99</td>
<td>.</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$2,964.15</td>
<td>39</td>
<td>.</td>
<td>99</td>
</tr>
<tr>
<td>Orion Club Gold members</td>
<td>Retail Sale</td>
<td>$13,710.45</td>
<td>169</td>
<td>40%</td>
<td>268</td>
</tr>
<tr>
<td></td>
<td>Catalog Sale</td>
<td>$6,836.47</td>
<td>58</td>
<td>.</td>
<td>326</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$11,234.10</td>
<td>81</td>
<td>.</td>
<td>326</td>
</tr>
<tr>
<td>Orion Club members</td>
<td>Retail Sale</td>
<td>$30,944.12</td>
<td>390</td>
<td>30%</td>
<td>716</td>
</tr>
<tr>
<td></td>
<td>Catalog Sale</td>
<td>$15,878.58</td>
<td>136</td>
<td>30%</td>
<td>852</td>
</tr>
<tr>
<td></td>
<td>Internet Sale</td>
<td>$7,293.30</td>
<td>106</td>
<td>.</td>
<td>852</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$100,077.47</td>
<td>1078</td>
<td>100%</td>
<td>852</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$100,077.47</td>
<td>1078</td>
<td>100%</td>
<td>852</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$100,077.47</td>
<td>1078</td>
<td>100%</td>
<td>852</td>
</tr>
</tbody>
</table>

### 8.4 General Tips

As with SAS in general, PROC REPORT does exactly what you tell it to do. It might not do what you want it to do, but it does what you tell it to do. The following tips give guidance on how to approach generating a report with PROC REPORT and places to check if something goes wrong.

**Tip #1: Know Your Data**

The first tip for determining why PROC REPORT did not give you what you want is universal to programming: **know your data**.

- Check that there are no missing values. Warning messages about missing values, whether they are grouping or ANALYSIS variables, are generated because the input data set does in fact have missing values.
- Check that all of the categories that you expect are present in your data. PROC REPORT is not going to output a category that does not exist. Chapter 3 contains an example of inserting categories with no data.
• Check the length of your character variables. A text value inserted via PROC REPORT might be truncated if the length of the variable that you are inserting it into is shorter than the length of the text.

Tip #2: Plan Your Report
The next tip is to plan your report before you start to program. The plan should help you decide whether you need to create formats or whether you need to add variables to your data set for grouping or ordering purposes. Based on what you have learned about PROC REPORT from this book, planning should also help you determine whether you are going to need to pre-process your data in some way.

Tip #3: Start Small
Once you begin to program, start small and work your way up to more complicated results. Perhaps use a subset of your data when working out all of the kinks. Start with two or three variables to get the basic report and then add to it. In this way you can see how the report changes as you add more code, and it is easier to determine where something went wrong. Also, the general recommendation is to finalize the structure of the report and the numbers before adding styling attributes.

Tip #4: Check the Variable Order on the Column Statement
One of the most important things to know about PROC REPORT is that it works in a left to right direction based on the report-items listed on the COLUMN statement. If a column has missing values that should not be missing or a format or color is not applied, check the COMPUTE statement and the assignment statements inside of the compute blocks. Any report-item to the right of the one on the COMPUTE statement is not available in the compute block and will be missing.

Tip #5: Include Only One Usage Value
On the DEFINE statement, be sure to have only one usage. The usage value closest to the semicolon is the one that is used. Commonly, ORDER is placed on a DEFINE statement when the programmer wants the report to be sorted by that variable. ORDER is a usage value. It has a specific meaning to PROC REPORT. It will overwrite any other usage listed prior to it on the DEFINE statement. The ORDER= option is different from the ORDER usage, and is the one you need to use when specifying the desired sort order.

Tip #6: Take Advantage of the SHOWALL Option
The SHOWALL option, explained in Chapter 1, is very useful if you have used the NOPRINT or NOZERO options. Hidden columns affect the column numbers needed inside of compute blocks. You can use this option to confirm you are using the correct column numbers. Hidden GROUP or ORDER variables can and do effect the GROUP or ORDER variables that are seen. Use the SHOWALL option to see hidden columns without having to alter any other statements in the PROC REPORT step.
Tip #7: Use the LIST Option
As discussed in Chapter 1, the LIST option is most useful when creating output for the Listing destination, but it can be handy when you need to confirm the label, format, and width PROC REPORT is using for a variable.

Tip #8: Remove Unnecessary Options
Use only the options that you need. PROC REPORT code can become long and complicated even without unnecessary code. Including options that do not impact the output are distracting and add clutter.

# Index

## A

**ABOVE value** 189  
**ACROSS label**, versus **ACROSS value** 216–217  
**ACROSS option**, DEFINE statement 11, 130–132  
**ACROSS value**  
  versus **ACROSS label** 216–217  
  headers 215–216  
**ACROSS variable**  
  about 9, 24, 109–110  
  changing borders for headers 195–196  
  **COLUMN-ID** and 18  
  combining **ANALYSIS** variable, **GROUP** variable and 112–113  
  **COMPLETEROWS** and a **DISPLAY** variable under an 162–164  
  date variables and 46  
  debugging 268, 269, 271  
  **DISPLAY** variable under 113–116  
  **GROUP** variable under 116  
  grouped reports, no 146–148  
  grouped reports with an 148–155  
  header section rows 123–132  
  label headers 215  
  merging and 179  
  nesting 135–144  
  nonstandard reports 132–144  
  percentages 118–123  
  sorting and 47, 82–84  
  standard reports 110–123  
  trafficlighting under 206–212  
  using **COMPLETEROWS** option with 155–164  
  value headers 215  
**alias** 30, 121–122  
**ALL value** 189  
**ANALYSIS option**, DEFINE statement 11  
**ANALYSIS variable** 7, 18, 24, 59, 68–69, 80–82, 86, 88, 90, 110–113, 115, 146–147, 152–155, 157, 202–204, 224, 268, 273  

## appearance, report  
  about 186  
  advanced color and border assignments 220–229  
  borders 188–199  
  color in headers 213–217  
  images 229–235  
  **LINE** statements 217–220  
  **STYLE=** option 186–187  
  trafficlighting 199–212  
  trafficlighting under an **ACROSS** variable 206–212  
  **URLs** 235–239  

## appearance options  
  **DEFINE** statement 13–14  
  **REPORT** procedure statement 5–6  

**ARRAY** statement 25, 211  
**at-sign (@)** 20, 90  
**attribute name argument** 19  
**attribute value argument** 19–20  
**attributes**  
  *See also specific attributes*  
  applying changes using macro programs 208–210  
  applying to **ACROSS** variable headers 215  
  applying to **LINE** statements 218–219  
  assigning to all header cells 213  
  changing header attributes for gender 213–214  
  changing with **CALL DEFINE** statement 201  
**AUTONAME** variable 172

## B

**BACKGROUND= attribute** 20, 187, 200, 206–207  
**background color**  
  applying via **CALL DEFINE** statement 200  
  applying via **DEFINE** statement 199–200  
  applying with formats 216  
  changing with **CALL DEFINE** statement 35–36  
  setting for **TOTAL_RETAIL_PRICE** using  
  **LASTVAR** compute block 38  
**BACKGROUNDIMAGE= attribute** 234–235  
**backslash (\)** 196  
*Base SAS Procedures Guide* 2
BEGIN value  222
BELOW value  189
BOLD= attribute  225–226
BORDERCOLOR= attribute  187, 222, 228
borders
   about  188
   advanced assignments  220–229
   changing for ACROSS variable headers  195–196
   changing for spanning headers  197
   changing in HTML output  226–228
column (data)  191–195
FRAME= attribute  189–191
header  195–199
   inserting between columns  192
   inserting for one column  192–193
   inserting lines between groups  193–195
RULES= attribute  189–191
BORDERSPACING= attribute  222
BOX value  189
BREAK AFTER statement  249–251
BREAK BEFORE statement  245–246, 250
BREAK statement
   about  2, 3, 20, 24
   CONTENTS= option  15, 249–251, 256–261, 272
   creating rows with  70–75
   creating rows with LINE statement versus  90–91
debugging  271–272
PAGE option  15, 94, 254–255, 256–261, 272
paging  52–53
reports and  171, 175
STYLE= option  187
STYLE<(location(s))>= option  16
SUMMARIZE option  15, 70, 72–79, 87, 89
summary rows and  87–88
SUPPRESS option  16
BY statement  16, 78, 79, 150–151, 164, 244
BY variable  78–79, 106–107, 243–246, 255–256, 260
BYPAGENO= option  6
#BYVAL  244
#BYVAR  107, 244

C
CALC_QUANTITY variable  146–150, 152–155
CALL DEFINE statement
   about  2, 18–21, 25
   adding URL hyperlinks and flyovers with  237
   aliases and  30
   changing attributes with  201
   changing background color with  35–36
   color and  224–226, 228
   color in headers  213
color on the diagonal  205
column borders and  193, 194
column references and  144
debugging  269, 270, 273, 274
diagonal pattern  212
ingesting images into new columns via  232
inserting images via  231–232
linking files  237–239
ODS destination and  229
placing in different compute block  36–37
STYLE= option  187
summary rows and  72–75
trafficlighting and  199–204, 207–212
CALL EXECUTE statement  107
Carpenter, Art
   Carpenter's Complete Guide to the SAS Report Procedure  25
Carpenter's Complete Guide to the SAS Report Procedure  (Carpenter)  25
CATS function  33
C_COUNTRY variable  219
cell value, color based on  199–200
CELLHEIGHT= attribute  180, 235
CELLWIDTH= attribute  141, 180, 235
CELLWIDTH= option, DEFINE statement  13–14
CENTER|NOCENTER option, REPORT procedure statement  4
CHAR option, COMPUTE statement  65
color
   See also  background color
   advanced assignments  220–229
   applying in diagonal pattern  212
character report-items  65–68
character variables  38–39, 65–66
COUNTRY variable  219
child nodes  256–261
&CHKVAR  205
CLASS statement  147, 153, 158, 164, 165, 176
CLASS variable  147, 149, 159, 172
&CNTAGEG  209–210
CNTLIN= option  131–132
See also  background color
advanced assignments  220–229
applying in diagonal pattern  212
based on cell value 199–200, 206–207
based on report-item 200–204, 207–211
on diagonal 204–205, 211–212
for every other row 225–226
foreground 203–204
in headers 213–217
COLOR variable 203–204
COLS value 189
column (data) borders 191–195
column number 30–31
column percentages, within groups 122–123
column references, creating with macros 142–144
COLUMN statement
about 2, 3, 6–11
ACROSS variables and 110
calculating percentages within countries 81–82
checking order of variables on 279
color and 205, 225–226
column (data) borders 194
column references 143
combining ACROSS, GROUP and
ANALYSIS variables 112
COMPLETEEROWS and 160, 163, 165–166
creating basic reports 56–57
creating character COMPUTED columns
with numeric variables 66–68
debugging 267–268, 270, 272, 274
defining variables as GROUP 61
GROUP variable under ACROSS variable 116
header and data sections 168–169
including summary rows 86
left to right availability 31
LINE statements and 219
merging and 179, 183
multiple summary rows at one location 84–90
multiple variables and 115
multiple variables under ACROSS variable 111, 112
nesting ACROSS variables 136
numeric variables and 39–40
ORDER variables and 175, 177
output via a COMPUTED variable 277
output via a LINE statement 275–276
placing dummy report-item in 37–38
placing ORDER variable at end of 44
placing spanning headers beside ACROSS labels 126–127
removing header rows 136
repeating values and 41, 42–43
report-items 30, 34, 37, 64, 69
row percentages 121, 122
sorting by statistic 147, 151, 154
spanning headers 106, 125–126, 128
special data consideration reports 94
summary rows and 90
temporary variables and 47
trafficlighting and 200–204, 209–210
BY variables nodes and 246
column widths
about 44–45
effect of default COLWIDTH value of .9 45
using a format for desired width 46
column-ID argument 18–19
columns
hiding under ACROSS 138–139
inserting borders between 192
inserting borders for one 192–193
&COLVARS macro parameter 205
COLWIDTH= option 5, 45–46
COMPLETECOLS option 4
COMPLETEEROWS option
about 5
default behavior of 156–158
DISPLAY variable and 162–164
fixing behavior 158–161
pre-processing data and 146
using 97–102
using with ACROSS 155–164
using with multiple GROUP variables 164–167
COMPLETETYPES option 153, 158
compound name 30
COMPUTE AFTER statement 105
COMPUTE BEFORE statement 105
compute block statements, debugging 272–274
compute blocks
about 25
COMPUTE statement 25–26
execution of 26–29
COMPUTE statement
about 2, 18, 25–26, 279
calculating percentages within countries 82
CHAR option 65
color and 205
creating character report-items 65–68
creating new report-items 64
debugging 272, 273, 274
left to right availability 33, 36
LINE statements and 217
NOPRINT option 37, 65, 87, 89, 97
numeric report-items 68–69
rows and 90, 120
STYLE= option 20, 187
STYLE<(location(s))>= option 18
summary rows 74–75
tables and 232
trafficlighting and 200–204, 201, 203–204
COMPUTED option, DEFINE statement 11
COMPUTED variable 29, 59, 111, 116, 229, 232, 276–278
CONTENTS= option
about 5
BREAK statement 15, 249–251, 256–261, 272
DEFINE statement 15, 246–248
RBREAK statement 17, 251
REPORT procedure statement 4–5
COPY statement 253, 258–259, 265
COSTPRICE_PER_UNIT variable 67, 206–207, 208–211
countries, calculating percentages within 81–82
CUSTOMER_GROUP variable 202, 203–204, 224
CUSTOMER_ID variable 59
CUSTOMER_NAME variable 60–61, 63–64, 70, 72–78, 116
CUSTOMER_NAME-ORDER_TYPE variable 116
CUSTOMER_TYPE variable 172, 176, 236–237, 237–238

D
data
  incorporating pieces into one report 171–179
  knowing your 278–279
  outputting tables with no 102–104
data (column) borders 191–195
DATA= option, REPORT procedure statement 3
data sections, using information from variables in 167–169
date variables 46
DDATE variable 134
debugging
  about 267
  errors 267–274
  general tips 278–280
  log notes 267–274
  temporary variable values 275–278
  warnings 267–274
default nodes 241–243
default percentages 118–119
DEFINE statement
  about 2, 9–11, 21, 24
  ACROSS option 11, 130–132
  ANALYSIS option 11
  appearance options 13–14
  CELLWIDTH= option 13–14
  color and 206–207, 213
  COMPUTED option 11
  CONTENTS= option 15, 246–248
  creating subtotal columns 135
debugging 267–271
  DESCENDING option 12, 269
  DISPLAY option 11, 268
  EXCLUSIVE option 13
  FLOW option 15
  FORMAT= option 14, 46, 83, 132, 134
  GROUP option 12, 89
  header and data sections 168
  ID option 14–15
  images and 233
  interaction options 12–13
  label rows 129, 130
  MISSING option 12
  MLF option 14, 271
  nesting ACROSS variables 135
  NOPRINT option 13, 39–40, 63
  NOZERO option 13, 138–139, 279
  ORDER= option 12, 47–50, 82–83, 95, 136, 279
  ORDER=DATA option 83, 95–97, 132, 134
PAGE option 14, 50–53, 93–95, 139, 247–248
  sorting and 131–132
SPACING= option 13
spanning headers 125–127
standard reports and 110, 111
STYLE= option 13–14, 187
STYLE(COLUMN)= option 231–234
STYLE(HEADER)= option 213–214, 215
STYLE<(location(s))>= option 14
trafficlighting and 199–200
usage options 11–12
usage values on 279
utility options 14–15
WEIGHT= option 12
WIDTH= option 13–14, 45–46, 268
defining
  variables as GROUP 59–64
  variables as ORDER 58–59
DEMOG variable
  assigning value to 34–35
  creating as a COMPUTED report-item 33–34
DESCENDING option, DEFINE statement 12, 269
detail information, combined with summary information 171–174
DETAILS option, LIST statement 253–254, 259–260
diagonal, color on 204–205
DISCOUNT variable 168
DISPLAY option, DEFINE statement 11, 268
DISPLAY variable 19, 29, 46, 65, 67, 111, 113–116, 147, 162–164, 269, 271, 274
DO loop 103, 143, 205, 211, 212
DOCUMENT procedure 242, 253, 256–261
DUMMY variable 205

Excel output, changing spanning header attributes for 215
EXCLUSIVE option, DEFINE statement 13
execution, general 24–29
expanding characters
  including in spanning headers for listing destinations 10
  spanning headers with for non-listing destinations 10–11

F
FILE PRINT statement 103
FINISH value 222
FLOW option 15, 169–170
FLYOVER= attribute 235–239
FMTSEARCH option, OPTIONS statement 252
FONT= attribute 187
FONTSIZE= attribute 187
FOOTNOTE statement 230–231
FOREGROUND= attribute 203–204, 225–226
foreground color, changing based on color variable 203–204
FORMAT= option 14, 46, 83, 132, 134, 233
FORMAT procedure 83, 131–135, 145, 252
FORMAT statement 21, 159, 165
forward slash (/) 15, 16
FRAME= attribute 187, 189–191
FREQ procedure 173

G
GLM procedure 146, 173
global statements 21
group combinations, showing all 97–98
GROUP option, DEFINE statement 12, 89
grouped reports
  with an ACROSS variable 148–155
  no ACROSS variable 146–148
groups
  calculating percentages within 80–82
  column percentages within 122–123
  inserting border lines between 193–195
GROUPS value 190
Header borders 195–199

Header section rows
  about 123
  counting as part of ACROSS values 130–132
  default created with ACROSS 123–124
  placing multiple spanning headers beside ACROSS 127–128
  placing spanning header beside ACROSS values 125–126
  placing spanning headers beside ACROSS label 126–127
  removing 136–137
  removing ACROSS label rows 128–130
  using information from variables in 167–169

Header text, putting category totals in 131–132

Headers
  ACROSS value 215–216
  color in 213–217

Hiding columns under ACROSS 138–139

HOLD variable 38–39

Horizontal borders, drawing with inline formatting 197–198

Horizontal merge 180–183

HTML output
  changing borders in 226–228
  changing spanning header attributes for 215
  HTMLSTYLE= attribute 228–229
  hyperlinking to static locations 236–237

ID option 14–15, 94

IF condition 29, 103, 202

IF/ELSE IF condition 201

IF-THEN logic 25, 91

Images
  about 229–230
  inserting for each value 233–234
  placing above/below report tables 230–231
  placing inside tables 231–232

IND variable 150–151, 154

Inline formatting
  changing spanning header attributes with 214–215
  drawing vertical and horizontal borders with 197–198

Interaction options, DEFINE statement 12–13

INTO clause 103

ITEM variable 67

JUST= option 20

LABEL statement 21

LAG function 42–43

LAST_GENDER variable 194

LASTPAGE function 53

LASTVAR compute block 38

Left to right availability 31–38

LENGTH statement 92

LEVELS= option, LIST statement 253–254, 259–260

LHS value 189

LINE statement
  about 2, 5, 17, 20, 217
  applying attributes to 218–219
  based on ORDER variables 176–179
  color and 224
  compute blocks and 25, 26
  creating copies of CUSTOMER_COUNTRY 219
  creating rows with 90–91
  debugging 273, 274
  displaying conditionally 91–93
  merging and 179
  ORDER variables and 175
  output via a 275–276
  tables and 105
  using inside compute 62–63
  writing second 219–220

LINESIZE option 50, 268

LINK variable 237–238

LINKCOLOR= attribute 236

Linking, to numerous files 237–239

LIST option, REPORT procedure statement 4, 280

LIST statement 253–254, 259–260

LOCATION argument 15, 25, 26

Log notes 267–274

LOGISTIC procedure 146
**M**

macros
- applying attribute changes using 208–210
- creating column references with 142–144

MAKE statement 253, 258, 261, 265
MAX statistic 85, 168
MEAN statistic 69, 85, 168
MEANS procedure 130–131, 146–147, 149–150, 158–161, 165, 173, 177
MERGE statement 151
MERGECROSS option 183
merged cells, creating look of 179–183
MIN statistic 85, 168
MISSING option 5, 12, 105
MLF option, DEFINE statement 14, 271
MOD function 225–226
MULTILABEL option 135

**N**

N statistics
- about 168
- under ACROSS variable 111
- placing under ACROSS variables 9–10
NAME variable 172
NAMED option 5
nested groups, showing summary values for 87–90
nesting ACROSS variables 135–144
NEWLINE function 169–170
NOCOMPLETECOLS option 4
NOCOMPLETEROWS option 5
nodes
- child 256–261
- combining multiple steps under one 262–265
- default 241–243
- parent 256–261
- BY variable 243–246
NOHEADER option, REPORT procedure statement 4, 180–181
NONE value 190
nonstandard reports
- about 80
- calculating percentages within groups 80–82
- conditionally displaying LINE statements 91–93
- customizing sort order 82–84
- multiple summary rows at one location 84–90
- rows created with LINE statement versus BREAK statement 90–91

NOPRINT option
- about 279
- color and 209–210, 211
- column (data) border and 194
- COMPLETE ROWS and 160
- COMPUTE statement 37, 65, 87, 89, 97
debugging 269
DEFINE statement 13, 39–40, 63
DISPLAY variable and 114, 116
ORDER variables and 177
output via a COMPUTED variable 277
- sorting by statistic 147, 151, 154
tables and 94
- By variable nodes and 246
NOTSORTED option 83, 132, 135
NOWD option, REPORT procedure statement 4
NOZERO option, DEFINE statement 13, 138–139, 279
numeric report-item 68–69
numeric variables
- about 39–40
- creating character COMPUTED columns with 66–68
- using with ORDER=INTERNAL 96–97
NUMOBS variable 103
NWAY option, MEANS procedure 147

**O**

observations, checking number of 103–104
ODS destination 228–229
ODS DOCUMENT procedure 252, 257
ODS ESCAPECHAR (’) 196, 197
ODS GRIDDED LAYOUT statement 182
ODS LAYOUT statement 182
ODS RTF statement 182
ODS TEXT= statement 102, 230–231
ODSTEXT procedure 230–231
options
- See also specific options
- appearance 5–6, 13–14
- DOCUMENT procedure 253
- removing unnecessary 280
- REPORT procedure statement 3–4
OPTIONS statement, FMTSEARCH option 252
ORD variable 176, 177
ORDER= option, DEFINE statement 12, 47–50, 82–83, 95, 136, 279
ORDER variable 19, 23–24, 29, 38–44, 46–47,
58–59, 65, 94, 174–179, 193, 269,
271, 274–276
ORDER=DATA option, DEFINE statement 83,
95–97, 132, 134
ORDER_DATE variable 137, 138–139
ORDER_ID, transposing data by 168
ORDER=INTERNAL 96–97, 154
ORDER_SUM variable 147, 150
ORDER_TYPE variable 100–102, 114, 116, 120,
132–133, 136–137, 141, 164–167,
233–235
&ORDERVAR 205
OUT= option, REPORT procedure statement 3,
167
OUTPUT statement 147, 153, 159, 165
OUTPUTWIDTH= attribute 181, 187

P

page breaks, vertical 139–142
PAGE option
   BREAK statement 15, 94, 254–255, 256–
   261, 272
   DEFINE statement 14, 50–53, 93–95, 139,
   247–248
   RBREAK statement 17, 251
PAGEIT variable 94–95
PAGENO option 53
PAGEOF function 53
PAGESIZE option 50
paging 50–53
parent node 256–261
PCNT variable 224
PTSUM keyword 80–81, 118–119
percentages
   about 118
   calculating within countries 81–82
   calculating within groups 80–82
   column within groups 122–123
   default 118–119
   row 119–122
plain text, outputting 102–104
plus sign (+) 20, 90
POSTIMAGE= attribute 229–234
POSTTEXT variable 187
PREIMAGE= attribute 229–234
PRELOADFMT option 13, 83, 99, 132–133, 134,
158, 270

pre-processing data
   about 145–146
   creating merged cell look 179–183
   incorporating data pieces into one report 171–179
   sorting by statistic 146–155
   using COMPLETEROWS with ACROSS 155–164
   using COMPLETEROWS with multiple GROUP
   variables 164–167
   using information from variables in header and data
   sections 167–169
   wrapping text 169–170
PRETEXT variable 187
PRINT procedure 25
PROC DOCUMENTS by Example Using SAS (Tuchman)
   252
procedure label 242
PROCLABEL variable 245–246
PROD_INFO variable 65–66
PRODUCT_GROUP variable 169–170
PRODUCT_LINE variable 61–62, 65, 71–75, 100–102,
212, 221, 224, 226–228
PRODUCT_NAME variable 65, 168
PUT statement 103, 275

Q

QUANTITY variable 67, 70, 110–113, 116–117, 124–
126, 129, 134, 146–147, 149, 151–158, 168,
193, 196, 202, 206–211, 277

R

RBREAK statement
   about 3, 16–17, 20, 24
   changing values on summary rows 77–78
   CONTENTS= option 17, 251
   creating rows with 75–76
   PAGE option 17, 251
   reports and 171
   rows and 79, 87
   STYLE= option 187
   STYLE<(<location(s))>= option 17
   SUMMARIZE option 16
RENAME= option 176
repeated values, removing using LAG function 42–43
repeating values 41–44
REPLAY statement 252, 258, 261
report label 242

REPORT procedure
  about 1–2, 25
  MISSING option 105
  SPANROWS option 179
  STYLE= option 186–187, 228–229

REPORT procedure statement
  about 2, 3
  appearance options 5–6
  CENTER|NOCENTER option 4
  CONTENTS= option 4–5
  DATA= option 3
  LIST option 4, 280
  NOHEADER option 4, 180–181
  NOWD option 4
  options 3–4
  OUT= option 3, 167
  SHOWALL option 4, 211, 279
  STYLE(HEADER)= option 213, 214–215, 216–217

report tables, placing images above/below 230–231

report-items
  about 25–27, 29, 31–32, 35–37
  changing format of
    TOTAL_RETAIL_PRICE in 74–75
    character 65–68
    color based on 200–204, 207–211
    creating new 64–69, 116–117
    numeric 68–69
    numeric variables and 39–40
    placing dummy in COLUMN statement 37–38
    referencing 29–31

reports
  See also appearance, reports
  about 55–56
  grouped 146–155
  incorporating data pieces into one 171–179
  nonstandard 80–93
  planning 279
  special data consideration 93–108
  standard 56–79
%REPORTS macro program 103
RETAIN statement 150
RHS value 189
row percentages 119–122
ROW_NUM variable 212, 225–226

rows
  changing values on RBREAK summary 77–78
  color for every other 225–226
  creating with BREAK statement 70–75
  creating with LINE statement versus BREAK statement 90–91
  creating with RBREAK statement 75–76
  highlighting 202
  including statistics in detail 85–87
  unexpected 100–102
ROWS value 190
RTF command 228–229
RULES= attribute 179, 187, 189–191, 222

S
SAS 9.4 Macro Language: Reference 102
SAS 9.4 Output Delivery System: Advanced Topics 180, 183
SAS 9.4 Output Delivery System: Procedures Guide 252
SECTIONDATA option, ODS RTF statement 182
SELECT clause 103
SETLABEL statement 253, 258–259, 262–265
SHOWALL option, REPORT procedure statement 4, 211, 279
SKIP option 17
SKIP_SPACE option 182
sort order
  controlling with ORDER variable 43
  customizing 82–84, 132–133
  ORDER= option and 47–50
  by statistic 146–155, 149–152, 152–155
SORT procedure 147, 151, 193
SPACING= option, DEFINE statement 13
spanning headers
  about 214–215
  adding 57–58
  assigning text dynamically 106–108
  changing borders for 197
  on COLUMN statement 8
  with expanding characters for non-listing destinations 10–11
  including expanding characters in for listing destinations 10
  placing beside ACROSS labels 126–127
  placing beside ACROSS values 125–126
placing multiple beside ACROSS header rows 127–128
writing ACROSS labels using 217
SPANROWS option 5, 61, 179
special data consideration reports about 93
dynamically assigning spanning header text 106–108
outputting tables with no data 102–106
using COMPLETEROWS option 97–102
using ORDER=DATA 95–97
wide tables 93–95
SPENDER variable 114
SPLIT= option 6, 169–170, 173
SQL procedure 103, 130–131, 146, 167
stacking
statistics 7–8, 110–111
statistics under ACROSS variables 9
stair-step pattern 161
standard reports
about 56
ACROSS variables and 110–123
creating basic reports 56–58
creating new report-items 64–69
defining variables as GROUP 59–64
defining variables as ORDER 58–59
producing summary rows 70–79
statements
See also specific statements
about 1–2
DOCUMENT procedure 253
global 21
static locations, hyperlinking to 236–237
statistical keyword 12, 146–155
statistics
including in detail rows 85–87
sorting by 149–152, 152–155
stacking 7–8, 110–111
stacking under ACROSS variables 9
STYLE argument 25, 237, 269
STYLE= option
BREAK statement 187
CALL DEFINE statement 187
COMPUTE statement 20, 187
DEFINE statement 13–14, 187
RBREAK statement 187
REPORT procedure 186–187, 228–229
STYLE(COLUMN)= option, DEFINE statement 231–234
STYLE(HEADER)= option
about 236–237
DEFINE statement 213–214, 215
REPORT procedure statement 213, 214–215, 216–217
STYLE<(location(s))>= option
about 6
BREAK statement 16
COMPUTE statement 18
DEFINE statement 14
RBREAK statement 17
STYLE(REPORT)= option 230–231
styles, applying multiple to one cell 220–224
STYLE(SUMMARY)= option 71
subtotal columns, creating 133–135
SUM= 85, 147
SUMMARIZE option
about 251
BREAK statement 15, 70, 72–79, 87, 89
RBREAK statement 16
summary information, combining with detail information 171–174
SUMMARY procedure 130–131
summary rows
adding text 72–73
multiple at one location 84–90
producing 70–79
summary values, showing for nested groups 87–90
SUPPRESS option, BREAK statement 16
T
table label 242
Table of Contents
about 241
BREAK statement, CONTENTS= option 249–251
default nodes 241–243
DEFINE statement, CONTENTS= option 246–248
DOCUMENT procedure 252, 253–265
ODS document 252
RBREAK statement, CONTENTS= option 251
BY variable nodes 243–246
tables
outputting text at bottom of structure of 105–106
outputting text inside of structure of 104–106
outputting with no data 102–104
placing images inside 231–232
wide 93–95
TABULATE procedure 25, 146
TAGATTR= attribute  106, 183, 228–229
target argument  15, 25, 26, 27, 28
TEMPLATE procedure 213, 223
temporary variables  47, 275–278

text
adding to summary row  72–73
outputting at bottom of table structure  105–106
outputting inside of table structure  104–106
wrapping  169–170
TEXTDECORATION= attribute  239
THISPAGE function  53
TITLE statement 230–231
TOTAL_COST variable
assigning value to  32–33
creating as a COMPUTED report-item 31–32
creating as a COMPUTED variable  104–105
TOTALQ variable  114, 115, 116
TOTALR variable  114

trafficlighting
about  199
under ACROSS variables  206–212
color based on cell value  199–200, 206–207
color based on report-item  200–204, 207–211
color on diagonal  204–205, 211–212
TRANSPOSE procedure 167, 168
troubleshooting
See debugging
Tuchman, Michael
PROC DOCUMENTS by Example Using SAS 252
type-specification argument  25

V
values
changing on RBREAK summary rows  77–78
inserting images for each  233–234
multiple under one heading  8
repeating 41–44

VAR statement 168
variables
See also specific variables
color 38–39
date 46
multiple under ACROSS variable 111–113
numerics 39–40
temporary  47
using information from in header and data sections 167–169

&W&VARSLEFT 209–210
&W&VARSUNDER 209–210
vertical borders, drawing with inline formatting  197–198
vertical merge 179–180
vertical page breaks 139–142
VOID value 189
VOLUME variable 114, 116
VSIDES value 189

Warnings 267–274
WAYS statement 172
WEIGHT= option, DEFINE statement 12
WHERE clause 103
wide tables 93–95
WIDTH= option, DEFINE statement 13–14, 45–46, 268
WRAP option 169–170
wrapping text 169–170

U
unexpected rows  100–102
URL= attribute 237
URLs
about 235–236
hyperlinking to static locations 236–237
linking to numerous files 237–239
usage options, for DEFINE statement 11–12
utility options, for DEFINE statement 14–15

About This Book

Purpose
The purpose of this book is to describe every aspect of PROC REPORT. The book reviews options and syntax and how the data set is processed behind the scenes. Most importantly, it provides many examples of the kinds of reports programmers need to create every day. The book explains why specific options and statements are required for certain kinds of reports and provides the most efficient code for generating the desired reports.

Is This Book for You?
This book is meant for SAS programmers of all skill levels in all industries who need to create reports. PROC REPORT can create easy, bland reports, but programmers, no matter the skill level, are rarely tasked with creating such reports. This book will help you increase your skill level and proficiency in generating the reports you need.

Prerequisites
Basic knowledge of SAS programming is necessary to understand the concepts and examples in this book. You should understand data structure and formats. You need a general understanding of what the Macro Facility is and you should know the basic concepts of sending output to Output Delivery System (ODS) destinations.

Scope of This Book
This book is entirely about PROC REPORT. It covers everything there is to know about PROC REPORT.

Although it includes the FORMAT, TRANSPOSE, MEANS, SQL, and DOCUMENT procedures, they are not covered in depth in this book. The syntax of those procedures is described only enough to convey the reason the procedure is needed in conjunction with PROC REPORT. The book also includes use of the Macro Facility.

PROC REPORT is designed specifically for generating reports; therefore, ODS plays a major role in the final appearance of the report. A full description of the syntax and use of ODS is beyond the
scope of this book. However, where necessary, the behavior of a certain destination is explained as it pertains to PROC REPORT.

About the Examples

Software Used to Develop the Book's Content

The examples in this book were developed using Base SAS 9.4TS1M3. The examples can be run in 9.4TS1M0-9.4TS1M3, with all available hot fixes installed.

The examples should work in SAS 9.3 as well, but please note: if the example programs are run interactively in 9.3, the NOWD option should be added to the PROC REPORT statement. This book is also compatible with SAS University Edition.

Example Code and Data

The Orion Star data used throughout this book is with the permission of Sean O’Brien and Eric Rossland. The majority of the examples in this book use one data set, ORDERS. The other data sets are either a subset or restructuring of the ORDERS data set. Using just one data set should limit the confusion of having to understand the data structure of multiple data sets.

ORDERS contains purchase order data. The data set is unique at CUSTOMER_ID, ORDER_ID, and PRODUCT_ID level. A customer can have multiple orders and within each unique order, the customer can purchase multiple products.

The QUANTITY variable contains the quantity ordered of a specific product.

The CUSTOMER variable contains one record for each CUSTOMER_NAME value.

The SUMINFO variable was created by running a PROC MEANS step on the ORDERS data set. It contains one record for each CUSTOMER_COUNTRY-CUSTOMER_GROUP combination.

The TRAN_ORDERS variable was created by running a PROC TRANSPOSE step on the ORDERS data set. It contains one record for each CUSTOMER_ID-CUSTOMER_NAME combination and has one variable for each order the customer placed.

The ORDERS_3OBS variable contains three observations and four variables. The variables match those from the ORDERS data set that are used throughout the book.

To use the example programs from the author page, you need to store the five data sets described above in the WORK location. You also need to submit the createfmts.sas program. The program creates two formats, called $cntry and typef, used throughout the book. The formats should be stored in a catalog in the WORK location as well.
You do not have to read this book cover to cover to gain valuable knowledge about PROC REPORT. Each example demonstrates one technique and a specific outcome. However, the examples and chapters do build on each other, so you might find it helpful to read entire sections or chapters in order to fully understand the purpose of each example.

You can access the example code and data for this book by linking to its author page at http://support.sas.com/publishing/authors. Select the name of the author. Then, look for the cover thumbnail of this book, and select Example Code and Data to display the SAS programs that are included in this book.

If you are unable to access the code through the website, send email to saspress@sas.com.

**SAS University Edition**

If you are using SAS University Edition to access data and run your programs, then please check the SAS University Edition page to ensure that the software contains the product or products that you need to run the code: http://support.sas.com/software/products/university-edition/index.html.

**Output and Graphics Used in This Book**

The output in this book is sent to many of the ODS destinations, including PDF, HTML, RTF, Tagsets.ExcelXP, and the Excel destination. Most of the output is sent to the PDF destination. When another destination is used, it is noted with the example.

**Terminology Used in This Book**

The terms report-item, variable, and column are used interchangeably throughout the book and the term that is used is determined by the context. The term report-item distinguishes the type of compute block or required syntax for a specific statement. Otherwise, the term variable or column is used.

The terms location and target are italicized to distinguish how variables are used on BREAK, RBREAK, and COMPUTE statements. Location controls the placement of the break rows or where a compute block executes. Target controls when the execution takes place. This book capitalizes the value of a specific location or target when referred to within text.

The words group, order, and across are capitalized when the context refers to the usage value on the DEFINE statement. All statement names, options, and variable names are capitalized in text.

**Additional Help**

Although this book illustrates many analyses regularly performed in businesses across industries, questions specific to your aims and issues might arise. To fully support you, SAS Institute and SAS Press offer you the following help resources:
• For questions about topics covered in this book, contact the author through SAS Press:
  ◦ Send questions by email to saspress@sas.com; include the book title in your correspondence.
  ◦ Submit feedback on the author’s page at http://support.sas.com/author_feedback.
• For questions about topics in or beyond the scope of this book, post queries to the relevant SAS Support Communities at https://communities.sas.com/welcome.
• SAS Institute maintains a comprehensive website with up-to-date information. One page that is particularly useful to both the novice and the seasoned SAS user is its Knowledge Base. Search for relevant notes in the “Samples and SAS Notes” section of the Knowledge Base at http://support.sas.com/resources.
• Registered SAS users or their organizations can access SAS Customer Support at http://support.sas.com. Here you can pose specific questions to SAS Customer Support; under Support, click Submit a Problem. You will need to provide an email address to which replies can be sent, identify your organization, and provide a customer site number or license information. This information can be found in your SAS logs.

Recommended Reading
If you enjoy this book, consider reading these SAS Press books next.


Keep in Touch
We look forward to hearing from you. We invite questions, comments, and concerns. If you want to contact us about a specific book, please include the book title in your correspondence.

Contact the Author through SAS Press
• By email: saspress@sas.com
• Via the web: http://support.sas.com/author_feedback
About The Author

Jane Eslinger is a Senior Technical Support Analyst at SAS World Headquarters in Cary, North Carolina. Jane is a SAS Certified Advanced Programmer for SAS(R)9 and a SAS Certified Advanced Visual Business Analyst. She has presented at numerous conferences and users groups, including the 2015 SAS Global Forum conference where she presented a paper on compute blocks in PROC REPORT.

In her day-to-day work, Jane enjoys supporting SAS customers using ODS and Base SAS procedures, with an emphasis on PROC REPORT. Prior to joining SAS, Jane served as a statistician and statistical programmer in the social science and clinical research fields. She has a Bachelor of Science in Statistics from North Carolina State University.

Learn more about this author by visiting her author page at http://support.sas.com/eslinger. There you can download free book excerpts, access example code and data, read the latest reviews, get updates, and more.
Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore for additional books and resources.