

Report Customization Using PROC REPORT Procedure

Shruthi Amruthnath, EPITEC, INC., Southfield, MI

ABSTRACT

SAS® offers powerful report writing tools to generate customized reports. PROC REPORT procedure is one such report writing procedure in SAS® that conveys highly sophisticated information effectively. Since the introduction of PROC REPORT procedure in SAS® Version 6, this procedure is used to display a variety of information such as summarizing the data, statistical analysis and other high quality data listings. By integrating Output Delivery System (ODS) features along with this procedure, it simplifies the design, layout and appearance of the reports. This paper describes advanced features of PROC REPORT such as consolidating the data, specifying summary statistics, adding formatting features like specifying style elements using Output Delivery System, applying page breaks, handling missing values and using computed variables. Each of these topics is illustrated with an example. This paper also explains how to display the results of PROC REPORT procedure using ODS PDF and ODS TAGSETS EXCELXP options.

INTRODUCTION

PROC REPORT is a powerful and a flexible report writing tool that facilitates to generate custom reports. Using PROC REPORT procedure, it is possible to display both list and summarized data. By default PROC REPORT produces a list report. The standard form of PROC REPORT syntax is –

```
PROC REPORT <DATA = sas-dataset-name > <options>;  
RUN;
```

The examples that are explained throughout this paper uses SHOES dataset which is available in SAS® installed library SASHELP. This paper describes the process of building a report using PROC REPORT procedure and in particular covering following topics–

- Defining variables and its usage
- Applying style elements, assigning formats, defining column headings
- Applying page breaks and using COMPUTE blocks
- Finally, deliver the reports using ODS (Output Delivery System) packages such as PDF and Excel

Below is a simple form of PROC REPORT –

```
PROC REPORT DATA= SASHELP.SHOES NOWD;  
RUN;
```

By default, all the observations and variables in the dataset SASHELP.SHOES are printed. The order of variable in report is same as the input dataset SASHELP.SHOES. Using above code, the following sections details on customizing the SASHELP.SHOES dataset to be presented in the form of the report.

PROC REPORT TECHNIQUES

Devising blueprint of a report is an important step while creating a report. This is a crucial step to plan which variables to display, its appearance, order and applying other details such a statistical calculations and using computed variables.

Selecting variables

The COLUMN statement specifies the one or more variables to be displayed in the report. Figure 1 depicts simplest way to specify COLUMN statement in PROC REPORT procedure-

```
PROC REPORT DATA= SASHELP.SHOES NOWD;
    COLUMN REGION PRODUCT SUBSIDIARY SALES;
RUN;
```

Region	Product	Subsidiary	Total Sales
Africa	Boot	Addis Ababa	\$29,761
Africa	Men's Casual	Addis Ababa	\$67,242
Africa	Men's Dress	Addis Ababa	\$76,793
Africa	Sandal	Addis Ababa	\$62,819
Africa	Slipper	Addis Ababa	\$68,641
Africa	Sport Shoe	Addis Ababa	\$1,690
Africa	Women's Casual	Addis Ababa	\$51,541
Africa	Women's Dress	Addis Ababa	\$108,942

Figure 1. Example of a simple COLUMN Statement

From Figure 1, the order of variables that are mentioned in COLUMN statement appears in the same order in the report. So the COLUMN statement structures the variables that appear in the report.

Also, the COLUMN statement allows the column headers to be nested that can be spanned to display in different rows. Figure 2 shows advanced techniques to span headers in three rows.

```
PROC REPORT DATA= SASHELP.SHOES NOWD;
    COLUMN ('Global shoe sales in USD' ('Regional Shoe
        sales' REGION PRODUCT SUBSIDIARY) ('In USD
        only' SALES));
RUN;
```

Global shoe sales in USD			
Regional Shoe sales			In USD only
Region	Product	Subsidiary	Total Sales
Africa	Boot	Addis Ababa	\$29,761
Africa	Men's Casual	Addis Ababa	\$67,242
Africa	Men's Dress	Addis Ababa	\$76,793
Africa	Sandal	Addis Ababa	\$62,819
Africa	Slipper	Addis Ababa	\$68,641

Figure 2. Example of spanning headers using COLUMN Statement

The DEFINE statement

The variables that are specified in COLUMN statement can be described in DEFINE statement. The COLUMN statement only specifies the variables and its order that appears in the report, but it does not determine how the variables are used or how they are formatted. The optional attributes such as usage, formats, justification, column heading and statistics can be stated in DEFINE statement. The DEFINE statement does not determine the order of variables that appear in the report. So, they can be listed in any order after COLUMN statement.

Most commonly used options(s) are given in the below table.

	Options	Description
Usage	ACROSS	It displays the variables horizontally in ascending order. The order can be changed by specifying the option DESCENDING. This option works more like PROC TABULATE
	ANALYSIS	It applies only for numeric variable(s) to calculate a statistic. By default when this option is used, it calculates SUM statistic
	DISPLAY	It displays one or more variables in the report. All character variable(s) by default are treated as DISPLAY variable(s)
	GROUP	It consolidates one or more variable(s) into one row for all the observations and they are detailed into groups according to their formatted values
	ORDER	It orders the observations according to their formatted values. This option works just like PROC SORT. The observations can be arranged either in ascending or descending order by specifying the options ASCENDING or DESCENDING
	COMPUTED	The variables that are not a part of the input data set use this option. Also, the variable(s) must be specified in COLUMN statement
Attribute(s)	FORMAT=	SAS or user defined format can be assigned to the variable in DEFINE statement
	WIDTH=	The width of the column can be explicitly specified using WIDTH= option
	SPACING=	Specifies number of blank characters to leave in a column immediately to its left. The default SPACING is 2
Option(s)	ASCENDING	Sorts the variable in ascending order specified in DEFINE statement. It works just like PROC SORT with BY statement
	DESCENDING	Sorts the variable in descending order specified in DEFINE statement. . It works just like PROC SORT using BY statement with DESCENDING option
	NOPRINT	It conceals the variable and its values in the report
	NOZERO	It conceals the variable having the values with zero or missing
	PAGE	Before printing the first column page break is inserted
	COLOR =	Inserts foreground color to the column name and its values
	MISSING	By default PROC REPORT excludes the observations having missing values. The variables containing missing values are considered as valid values. Using MISSING options, the observations with missing values are included in the report
	CENTER	It allows the formatted values to be centered within the

Justification		column width
	LEFT	It allows the formatted values to be left justified within the column width
	RIGHT	It allows the formatted values to be right justified within the column width
Column Heading	<column heading>	It works just like LABEL statement. The optional column heading can be defined with in quotation marks (single or double) in the DEFINE statement. Instead of variable name, the column heading that is specified in quotes in the DEFINE statement appears in the report. By using split character such as :- = _ .* + the column heading can be split into two or more lines. Make sure that SPLIT="<split-character>" is specified in PROC REPORT statement
Statistics	<statistic>	Statistics can be applied along with analysis option. It can be applied only to numeric variable. Most commonly used statistics are mean, min, max, sum, nmiss

The PROC REPORT procedure given below shows the different options used in DEFINE statement. The options used are:

- GROUP- for grouping the variable(s) by Region and by Product
- CENTER – the character variables Region and Product are centered, whereas the numeric variable Sales is right justified
- The statistic SUM option calculates the total sum of Sales by Region and Product
- The width of the column is specified using WIDTH= option

Also, the heading's appearance can be enhanced using HEADLINE and HEADSKIP options in PROC REPORT statement. The HEADLINE underlines the column headings and adds spaces between them. The HEADSKIP option writes a blank line before the first observation.

```
PROC REPORT DATA= SASHELP.SHOES NOWD HEADLINE HEADSKIP SPLIT='*';
  WHERE REGION = 'Africa';
  COLUMN ('Global shoe sales in USD' ('Regional Shoe sales' REGION
    PRODUCT) ('In USD only' SALES));
  DEFINE REGION      / 'Region' GROUP WIDTH=6 CENTER;
  DEFINE PRODUCT    / 'Product' GROUP WIDTH=6 CENTER;
  DEFINE SALES      / 'Regional * Sales' SUM RIGHT
    FORMAT=DOLLAR15.2;
RUN;
```

Global shoe sales in USD		
Regional Shoe sales		In USD only
Region	Product	Regional Sales
Africa	Boot	\$119,835.00
	Men's Casual	\$562,794.00
	Men's Dress	\$318,500.00
	Sandal	\$190,409.00
	Slipper	\$337,076.00
	Sport Shoe	\$22,150.00
	Women's Casual	\$417,516.00
	Women's Dress	\$374,308.00

Figure 3. Example of different options used in DEFINE Statement

The COMPUTE block

The COMPUTE block begins with the COMPUTE statement and ends with the ENDCOMP statement. The COMPUTE statement allows adding your own values which is not a part of the input data set. The variable type in the COMPUTE statement can be either character or numeric. The computed variables do not alter the original input data set; it is exclusive to the PROC REPORT procedure only. Any new variables that are defined in a COMPUTE block must be included in a COLUMN statement, and specified as COMPUTED under variable usage in the DEFINE statement.

This paper focuses only on conditionally applying style elements to the variables using the COMPUTE block.

```
PROC REPORT DATA= SASHELP.SHOES NOWD HEADLINE HEADSKIP SPLIT='*';
  WHERE REGION = 'Africa';
  COLUMN ('Global shoe sales in USD' ('Regional Shoe sales' REGION
    PRODUCT) ('In USD only' SALES));
  DEFINE REGION      / 'Region' GROUP WIDTH=6 CENTER;
  DEFINE PRODUCT     / 'Product' ORDER GROUP WIDTH=6 CENTER;
  DEFINE SALES       / 'Regional * Sales' SUM RIGHT
    FORMAT=DOLLAR15.2;
  COMPUTE SALES;
    IF PRODUCT = 'SLIPPER' THEN
      CALL DEFINE (_COL_, "STYLE", "STYLE=[COLOR=MAGENTA]");
  ENDCOMP;

RUN;
```

Global shoe sales in USD		
Regional Shoe sales		In USD only
Region	Product	Regional Sales
Africa	Boot	\$119,835.00
	Men's Casual	\$562,794.00
	Men's Dress	\$318,500.00
	Sandal	\$190,409.00
	Slipper	\$337,076.00
	Sport Shoe	\$22,150.00
	Women's Casual	\$417,516.00
	Women's Dress	\$374,308.00

Figure 4. Conditionally applying style elements in COMPUTE block

The BREAK statement

PROC REPORT controls how the report items appear and where page breaks prevail. Using the BREAK statement the look and feel of the report can be governed. The BREAK statement allows applying breaks by page or by lines. Page breaks can be applied either on top of the page or on the bottom of the page. Breaks can be applied between the observations whenever GROUP or ORDER is specified in DEFINE statement.

One of the best features of BREAK statement is; several statistic options can be applied which combines features of PROC MEANS and PROC SUMMARY procedures. The program code shown in Figure 4 is slightly altered by inserting a BREAK statement. The SUMMARIZE option used in the BREAK statement creates a group summary by Region. The required argument AFTER in the BREAK statement applies a break line after the last observation which has the same values as the break variable. Similarly, BEFORE which is another required argument in the BREAK statement can also be used to apply a break line before the first observation which has the same values as the break lines.

```

PROC REPORT DATA= SASHELP.SHOES NOWD HEADLINE HEADSKIP SPLIT='*';
  WHERE REGION = 'Africa';
  COLUMN ('Global shoe sales in USD' ('Regional Shoe sales' REGION
    PRODUCT) ('In USD only' SALES));
  DEFINE REGION      / 'Region' GROUP WIDTH=6 CENTER;
  DEFINE PRODUCT    / 'Product' GROUP ORDER WIDTH=6 CENTER;
  DEFINE SALES      / 'Regional * Sales' SUM RIGHT
    FORMAT=DOLLAR15.2;
  BREAK AFTER REGION / SUMMARIZE;
  COMPUTE SALES;
    IF PRODUCT = 'SLIPPER' THEN
      CALL DEFINE (_COL_, "STYLE", "STYLE=[COLOR=MAGENTA]");
  ENDCOMP;

RUN;

```

Global shoe sales in USD		
Regional Shoe sales		In USD only
Region	Product	Regional Sales
Africa	Boot	\$119,835.00
	Men's Casual	\$562,794.00
	Men's Dress	\$318,500.00
	Sandal	\$190,409.00
	Slipper	\$337,076.00
	Sport Shoe	\$22,150.00
	Women's Casual	\$417,516.00
	Women's Dress	\$374,308.00
Africa		\$2,342,588.00

Figure 5. Using the BREAK Statement

Applying STYLE elements

Style elements can be applied individually for data elements in the reports. The style elements can be employed using a STYLE = statement. It governs visual formatting features such as formats, foreground color, background color, borders, fonts, cell padding and several other display attributes.

The style element can be applied by row, column, headers, summary or the entire report. The style attributes can be implemented conditionally within the COMPUTE BLOCK using a CALL DEFINE statement as shown in Figure 4. The style attributes specified in STYLE = option are registered with the Output Delivery System. They can be either customized within the PROC REPORT procedure or by using PROC TEMPLATE procedure.

The table below shows some of the style attributes and its resulting effects.

BACKGROUND=YELLOW	Men's Casual
FONTWEIGHT=BOLD	Men's Casual
FONTFAMILY=ARIAL	Men's Casual
COLOR=PURPLE	Men's Casual
FONTSIZE=8pt	Men's Casual
BORDERBOTTOMCOLOR=MAGENTA BORDERBOTTOMWIDTH=1	Men's Casual

Here's an example of PROC REPORT using STYLE elements. The style effects have been defined in three different places in the code.

1. The column headers and the summary line have been formatted with style effects. They are specified in the beginning of the program in PROC REPORT statement of PROC REPORT procedure. The headers are highlighted in ORANGE and the SUMMARY line is highlighted in GREEN

- The style effects are applied in DEFINE statement. By using STYLE = options, the effects are applied to both column header and the values. It is possible to define separate effects for column and row by specifying as STYLE(ROW)= and STYLE(REPORT)= option
- Style effects have been applied conditionally to a single cell using CALL DEFINE statement in COMPUTE BLOCK.

```

PROC REPORT DATA= SASHELP.SHOES NOWD HEADLINE HEADSKIP SPLIT='*'
  STYLE(HEADER)=[ BACKGROUND=ORANGE FONT WEIGHT = bold]

  STYLE(SUMMARY)=[COLOR=WHITE BACKGROUND=GREEN
                  FONTFAMILY=ARIAL FONTSIZE=2 TEXTALIGH=CENTER
                  CELLWIDTH=1IN]
;
WHERE REGION = 'Africa';
COLUMN ('Global shoe sales in USD' ('Regional Shoe sales' REGION
  PRODUCT) ('In USD only' SALES));
DEFINE REGION      / 'Region' GROUP WIDTH=6 CENTER;
DEFINE PRODUCT     / 'Product' ORDER GROUP WIDTH=6 CENTER
  STYLE=[ borderbottomwidth=1];
DEFINE SALES       / 'Regional * Sales' SUM RIGHT
  FORMAT=DOLLAR15.2;
BREAK AFTER REGION / SUMMARIZE SKIP OL;
COMPUTE SALES;
  IF PRODUCT = 'Slipper' THEN
    CALL DEFINE ( COL , "STYLE", "STYLE=[COLOR=MAGENTA]");
ENDCOMP;

RUN;

```

Global shoe sales in USD		
Regional Shoe sales		In USD only
Region	Product	Regional Sales
Africa	Boot	\$119,835.00
	Men's Casual	\$562,794.00
	Men's Dress	\$318,500.00
	Sandal	\$190,409.00
	Slipper	\$337,076.00
	Sport Shoe	\$22,150.00
	Women's Casual	\$417,516.00
	Women's Dress	\$374,308.00
Africa		\$2,342,588.00

Figure 6. Applying STYLE elements in PROC REPORT

OUTPUT DELIVERY SYSTEM (ODS) BASICS

The data that is modified and formatted using a DATA step and PROC step can be customized into reports embedded with graphs and tables. Using an ODS statement, these reports can be written with

different output formats such as PDF (Portable Document Format), Microsoft Word, Microsoft Excel, Microsoft Power Point, allowing them to be accessed by a variety of software packages.

The general form of an ODS statement is –

```
ODS <destination-1> <option(s)>;
ODS <destination-2> <option(s)>;
...
ODS <destination-n><option(s)>;
...
<DATA step (s) and PROC step (s)>
...
ODS <destination-1> CLOSE;
ODS <destination-2> CLOSE;
..
ODS <destination-n> CLOSE;
```

This paper focuses on ODS PDF and ODS TAGSETS.EXCELXP destinations only.

The ODS PDF block

The ODS PDF belongs to printer family of statements which is designed to produce a printable document. The .pdf files that are generated by ODS PDF can be accessed commonly by Adobe Acrobat. By default, when the ODS PDF statement is turned on, other printer family statements are also turned on. They are PDS PCL, ODS PRINTER and ODS PS statements.

The general form of ODS PDF is –

```
ODS PDF <option(s)>;
<DATA and PROC step(s)>
ODS PDF CLOSE;
```

The output generated in Figure 6 can be exported to a .pdf file using ODS PDF statement. Using optional arguments in ODS PDF statement, the PROC REPORT procedure produces high quality procedural output in printable format.

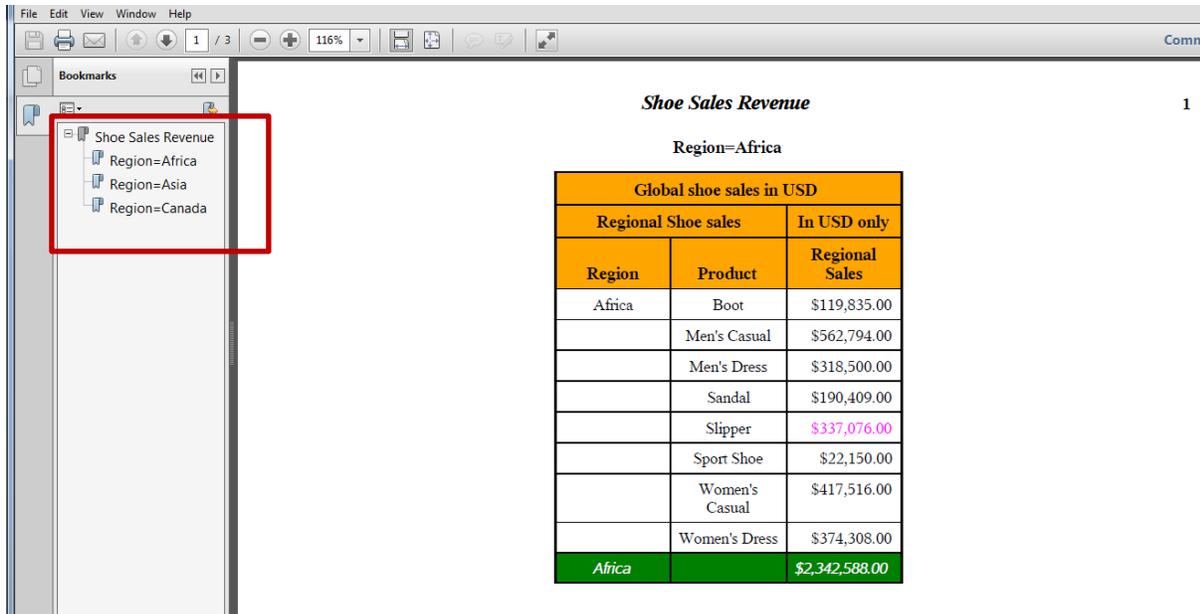


Figure 7. Generating .pdf file

Let's look at the options used in ODS PDF statement to create the report shown in Figure 7. The options are given below in table.

FILE=	It is the name of the external file to access the document in .pdf format. The external path and the file name must be specified within the quotation marks
SUBJECT=	It is a text string as a metadata file which is activated when ODS PRINTER and ODS PS with PDFMARK option is used. The subject should be specified with in the quotation marks
PDFTOC=<n level>	It specifies the node level expansion for table of contents. In Figure 7, the PDFTOC is set to 2
STYLE =	It specifies the template for printer output. The default styles are PRINTER, PDF, PS, and PCL
STARTPAGE=	Apply page break 1. BYGROUP: apply page break for each BY group 2. NEVER : specifies not to apply page breaks 3. YES : apply page break for each PROC step specified with in ODS PDF block 4. NO : specifies not to apply page break for each PROC STEP with in the ODS PDF block

Also, the labels for the procedure can be defined in the ODS PROCLABEL statement. The ODS PROCLABEL statement specifies the contents of the table of contents section. It gives more meaningful names, thus making it easier to navigate through the report. By using the option CONTENTS="" in PROC REPORT procedure, it hides the extraneous bookmarks created under the ODS PROCLABEL statement.

The ODS TAGSETS.EXCELXP block

The ODS TAGSETS.EXCELXP is used to generate XML files in Microsoft Excel version 2002 and later. The TAGSETS.EXCELXP is packed with magnificent supply of Excel options to create XML output in Microsoft Excel. The general form of ODS TAGSETS.EXCELXP is-

```
ODS TAGSETS.EXCELXP FILE="<filename>"
      OPTIONS=(<name1=value1>
              <name2=value2>
              ...
              <namen=valuen>);
<DATA and PROC step(s)>
ODS TAGSETS.EXCELXP CLOSE;
```

	A	B	C	D	E	F
2	Global shoe sales in USD					
3	Regional Shoe sales		In USD only			
4	Region	Product	Regional Sales			
5	Africa	Boot	\$119,835.00			
6		Men's Casual	\$562,794.00			
7		Men's Dress	\$318,500.00			
8		Sandal	\$190,409.00			
9		Slipper	\$337,076.00			
10		Sport Shoe	\$22,150.00			
11		Women's Casual	\$417,516.00			
12		Women's Dress	\$374,308.00			
13	Africa		\$2,342,588.00			
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						

Figure 8. Generating .xls file

Let's look at the options used in ODS TAGSETS.EXCELXP statement to create the report shown in Figure 8. The options are given below in table.

ORIENTATION=	It specifies the page orientation. The option is either PORTRAIT or LANDSCAPE
FROZEN_HEADERS=	It freezes the rows from scrolling. Default is set to no. In the code, it freezes upto 3 rows
ABSOLUTE_COLUMN_WIDTH=	It specifies the width of individual column. Default it is set to none
EMBEDDED_TITLES=	It specifies the titles in worksheet
SHEET_INTERVAL=	It divides the output into multiple worksheets. In this example, they are divided into BYGROUP. Notice that three worksheets are created-

	Africa, Asia and Canada using BYGROUP option
SHEET_NAME=	It specifies the name of the worksheet. Default it is set to none
SHEET_LABEL=	The prefixed worksheet name is replaces with its value. Default it is set to none
SUPRESS_BYLINES=	The bylines in the worksheet is suppresses. Default it is set to none
GRIDLINES=	The lines are turned on for printing
ROW_HEIGHT_FUDGE=	It specifies the height of each row with a fudge value
AUTOFIT_HEIGHT=	It specifies the auto height for each row. Default it is set to no
SKIP_SPACE=	It controls the space with different kinds of output. The default value is '1,0,1,1,1'

```
DATA SHOES;
    SET SASHELP.SHOES;
    COUNT=1;
RUN;
```

```
OPTIONS NODATE CENTER ORIENTATION=PORTRAIT;
```

```
ODS PDF FILE="MYPDF.pdf"
        SUBJECT='Shoe Sales'
        PDFTOC=2
        STYLE=PRINTER
        STARTPAGE =BYGROUP;
ODS PROCLABEL="Shoe Sales Revenue";
```

ODS PDF block

```
ODS TAGSETS.EXCELXP STYLE=Styles.sansPrinter FILE="MyExcel.xls";
ODS TAGSETS.EXCELXP OPTIONS (ORIENTATION='landscape'
        FROZEN_HEADERS='3'
        ABSOLUTE_COLUMN_WIDTH='10,8,8,8,8,8'
        EMBEDDED_TITLES='yes'
        SHEET_INTERVAL = 'bygroup'
        SHEET_NAME = 'none'
        SHEET_LABEL = ' '
        SUPPRESS_BYLINES = 'yes'
        GRIDLINES='yes'
        ROW_HEIGHT_FUDGE='0'
        AUTOFIT_HEIGHT='yes'
        SKIP_SPACE = '0,0,0,0,0'
        )
;
```

ODS TAGSETS.EXCELXP block

```
TITLE1 JUSTIFY=C 'Shoe Sales Revenue';
```

```
PROC REPORT DATA= SHOES NOWD HEADLINE HEADSKIP SPLIT='*' CONTENTS=""
    STYLE (HEADER)=[BACKGROUND=ORANGE FONT_WEIGHT=BOLD]
```

```
    STYLE (SUMMARY)=[COLOR=WHITE BACKGROUNDCOLOR=GREEN
        FONTFAMILY=ARIAL FONTSIZE=2 TEXTALIGN=CENTER
        CELLWIDTH=1IN]
```

```
    ;
    WHERE REGION IN ('Africa','Asia','Canada');
```

```

BY REGION;
COLUMN COUNT ('Global shoe sales in USD' ('Regional Shoe sales'
      REGION PRODUCT) ('In USD only' SALES));
DEFINE COUNT      / NOPRINT GROUP;
  DEFINE REGION    / 'Region' GROUP WIDTH=6 CENTER;
  DEFINE PRODUCT   / 'Product' ORDER GROUP WIDTH=6 CENTER;
  DEFINE SALES     / 'Regional * Sales' SUM RIGHT
      FORMAT=DOLLAR15.2;
BREAK AFTER REGION / SUMMARIZE SKIP OL;
COMPUTE SALES;
  IF PRODUCT = 'Slipper' THEN
    CALL DEFINE (_COL_, "STYLE", "STYLE=[COLOR=MAGENTA]");
ENDCOMP;
BREAK BEFORE COUNT / CONTENTS="" PAGE;
RUN;

ODS TAGSETS.EXCELXP CLOSE;
ODS PDF CLOSE;
ODS LISTING;

```

CONCLUSION

The techniques that are available in the PROC REPORT procedure produce high quality procedural output. PROC REPORT combines features of several procedures such as PROC SORT, PROC MEANS, PROC SUMMARY, PROC PRINT, and PROC UNIVARIATE to create a report. It is highly customizable and an easily modifiable procedure. It offers the flexibility to produce output in a variety of formats, accessible with different business applications using the Output Delivery System (ODS) functionality. The Output Delivery System enhances the report contents and presentation; and creates easy to access SAS output.

REFERENCES

- <http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/viewer.htm#a000146851.htm#a002473613>
- <http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/viewer.htm#a002473627.htm>
- https://support.sas.com/rnd/base/ods/odsmarkup/TipSheet_ods_xl_xp.pdf
- Arthur L. Carpenter, PROC REPORT Basics: Getting Started with the Primary Statements, SAS Global Forum, 2012, Paper HW07
- Shruthi Amruthnath, PROC SGPLOT over PROC GPLOT, Midwest SAS® User Group, 2013, Paper DV01-2013

RECOMMENDED READING

PROC REPORT by Example: Techniques for Building Professional Reports Using SAS by Lisa Fine

CONTACT INFORMATION

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

Name: Shruthi Amruthnath
 Enterprise: EPITEC, INC
 Address: 24800 Denso Dr. Suite 150
 City, State ZIP: Southfield, MI 48033

E-mail: shruthiamruthnath@gmail.com
Web: www.epitecinc.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.