

Before PROC REPORT, There Was PROC PRINT

Clarence Wm. Jackson, CSQA

Abstract

This paper will explore how to control PROC PRINT and using all of the options available to really make those reports stand out. Before the other PROCs that produced nice reports, there was the basic PROC PRINT, taking care of those SAS printing needs.

Introduction

Reporting procedures produce a display of info, which is the purpose of PROC PRINT. The PRINT procedure produces a display of variables from a SAS dataset and is easy to use.

Each observation in the dataset becomes a row on the report, and each variable becomes a column. You tell PROC PRINT what to do, not how to do it. PROC PRINT will use whatever formatting that is already included with the dataset, variables, and other special formatting. You can use the ATTRIB, FORMAT, LABEL, and WHERE statements in PROC PRINT, and it supports ODS. Global statements like TITLE and FOOTNOTE are also used in PROC PRINT. You can create a variety of reports ranging from a simple listing to a highly customized report that groups the data and calculates totals and subtotals for numeric variables.

PROC PRINT is limited in some ways. The procedure has improved to allow the use of the WHERE clause, but the way PROC PRINT works haven't changed since the early versions of SAS. It takes whatever data in whatever format you pass to it, arranges the most efficient page size listing that it can image from the data, and produces a listing. You don't tell PROC PRINT how to do it, just what to do. Do not tell PROC PRINT to do any computations other than adding numbers. Avoid using it to round values. Make sure the variables are passed to it the way you want them to look in most cases. Other limitations are in using PROC PRINT to create a formatted report. You can not tell PROC PRINT where and what spot on the report to print anything. That is where PROC REPORT and DATA _NULL_ comes in. PROC PRINT will create a different looking listing from one run to another if the data changes, and will not be the option if you must produce a consistent looking or highly structured report. But it will get the job done for most of your printing needs if you use the proper options.

PROC PRINT Usage

The PRINT procedure usage format is

```
PROC PRINT <option(s)>;  
BY <DESCENDING> variable-1 <...<DESCENDING> variable-n>  
<NOTSORTED>;  
PAGEBY BY-variable;  
SUMBY BY-variable;  
ID variable(s) <option>;  
SUM variable(s) <option>;  
VAR variable(s) <option>;
```

The keywords that include 'by' assumes that the data is sorted or arranged in the by variable order indicated. A brief explanation of the usage keywords

Before PROC REPORT, There Was PROC PRINT Clarence Wm. Jackson, CSQA

- **BY** - Produce a separate section of the report for each BY group. If the data is not sorted, include the keyword **NOTSORTED**
- **PAGEBY** - Control page ejects that occur before a page is full
- **SUMBY** - Limit the number of sums that appear in the report
- **ID** - Identify observations by the formatted values of the variables that you list instead of by observation numbers
- **SUM** - Total values of numeric variables
- **VAR** - Select variables that appear in the report and determine their order. Otherwise the variables will appear in the order in which they were created.

PROC PRINT without any options will produce a report that lists the observation number and the variables of a SAS dataset. The simplest invocation to print the variables of the most recently created dataset is

```
PROC PRINT;
```

This will produce a simple listing of all the observations in the dataset with the observations as rows and the variables will be the column with variable names as headings. The 'Obs' column is added as the 1st variable and is the number of the observation within the listing. The variables are listed in the order in which they were created in the DATA step. Any formatting applied will be inherited from a previous step.

		The SAS System			19:07 Tuesday, August 25, 2009	
1	Obs	NAME	SEX	AGE	HEIGHT	WEIGHT
	1	ALFRED	M	14	69	112
	2	ALICE	F	13	56	84
	3	BERNADETT	F	13	65	98
	4	BARBARA	F	14	63	102
	5	HENRY	M	14	63	102
	6	JAMES	M	12	57	83
	7	JANE	F	12	59	85
	8	JANET	F	15	62	112
	9	JEFFREY	M	13	62	84
	10	JOHN	M	12	59	99
	11	JOYCE	F	11	51	50
	12	JUDY	F	14	64	90
	13	LOUISE	F	12	56	77
	14	MARY	F	15	66	112
	15	PHILLIP	M	16	72	150
	16	ROBERT	M	12	64	128
	17	RONALD	M	15	67	133
	18	THOMAS	M	11	57	85
	19	WILLIAM	M	15	66	112

PROC PRINT Options

Options in PROC PRINT allow the user to apply some controls to the procedure in how it forms the listing. Some options work better with other options better than others, while some should be avoided. Just keep in mind before using an option the following considerations:

Before PROC REPORT, There Was PROC PRINT

Clarence Wm. Jackson, CSQA

- Number of observations. Since PROC PRINT will print all the observations, be careful when printing large datasets.
- Number of variables. PROC PRINT will seek to arrange all the variables per observation on one page. In cases where there are more variables than what will print across one row of print, the remaining variables will print in groups in the second part of the page.
- Size of variable values, formats, and types can affect how PROC PRINT arranges the info on the page. Long length variable values will cause some printing issues. To reduce the size of values trim them. Assign formats before using the procedure to ensure the values will be presented as you want to display.

The control options and their alias are listed below.

Control general format

Write a blank line between observations (double space)	DOUBLE or D
Print the number of observations in the data set, in BY groups, or both, N and specify explanatory text to print with the number using the '='	N={text}
Suppress the column in the output that identifies each observation by number	NOOBS
Specify a column header for the column that identifies each observation by number	OBS={text}
Round unformatted numeric values to two decimal places	ROUND or R

Control page format

Format the rows on a page	ROWS=n
Use each variable's formatted width as its column width on all pages	WIDTH=UNIFORM or U

Control column format

Control the orientation of the column headings	HEADING=V or H
Use variables' labels as column headings	LABEL or SPLIT=
Specify the split character, which controls line breaks in column headings	SPLIT= or S=
Specify one or more style elements for the Output Delivery System to use for different parts of the report	STYLE
Determine the column width for each variable	WIDTH=

You are not limited to the number of options used in an instance of PROC PRINT, but keep in mind that some options will work better than others for the needs you might in mind. Remember, you tell PROC PRINT what options to use, and it will apply them to the best ability that it can in making the listing use less paper and space for all the observations in the dataset.

PROC PRINT without any options but using a BY statement will produce a report that groups the observations by the BY variable used in a PROC SORT- BY variable. Before you can use a BY variable in the PROC PRINT, it must be sorted.

```
PROC PRINT; BY SEX;
```

**Before PROC REPORT, There Was PROC PRINT
Clarence Wm. Jackson, CSQA**

```
1                                     The SAS System                18:37 Tuesday, August 25, 2009
```

```
----- SEX=F -----
```

Obs	NAME	AGE	HEIGHT	WEIGHT
1	ALICE	13	56	84
2	BERNADETT	13	65	98
3	BARBARA	14	63	102
4	JANE	12	59	85
5	JANET	15	62	112
6	JOYCE	11	51	50
7	JUDY	14	64	90
8	LOUISE	12	56	77
9	MARY	15	66	112

```
----- SEX=M -----
```

Obs	NAME	AGE	HEIGHT	WEIGHT
10	ALFRED	14	69	112
11	HENRY	14	63	102
12	JAMES	12	57	83
13	JEFFREY	13	62	84
14	JOHN	12	59	99
15	PHILLIP	16	72	150

Using the 'SPLIT=' option on variables that have labels is always a good way to display variable names. You can use the LABEL statement in a DATA step, or use it in the PROC PRINT statement. The SPLIT option allows the use of a character in the label to be used to split a long name over the variable instead of the variable's name. For example, a label for NAME could be 'Student*Name' and it will be split using the '*'.

PROC PRINT with 'N' option, ID statement, and using a BY statement will produce a report that groups the observations by the BY variable used in a PROC SORT BY variable. It will add the number of observations in each by group.

```
PROC PRINT N; BY SEX; ID NAME;
```

Before PROC REPORT, There Was PROC PRINT Clarence Wm. Jackson, CSQA

2

The SAS System 18:37 Tuesday, August 25, 2009

----- SEX=F

NAME	AGE	HEIGHT	WEIGHT
ALICE	13	56	84
BERNADETT	13	65	98
BARBARA	14	63	102
JANE	12	59	85
JANET	15	62	112
JOYCE	11	51	50
JUDY	14	64	90
LOUISE	12	56	77
MARY	15	66	112

N = 9

----- SEX=M

NAME	AGE	HEIGHT	WEIGHT
ALFRED	14	69	112
HENRY	14	63	102
JAMES	12	57	83
JEFFREY	13	62	84
JOHN	12	59	99
PHILLIP	16	72	150
ROBERT	12	64	128

PROC PRINT also produces output pages thru ODS. The following code will produce an html page named 'print2web.html' using the style info included in the PROC PRINT STYLE option.

```
ods html file='print2web.html';

PROC PRINT N S='*' style(HEADER) = {font_style=italic
                                   style(DATA) = {background=blue foreground = white};
BY SEX;
ID NAME;
LABEL NAME='KID*NAME' SEX='GENDER' AGE='CURRENT*AGE';

ods html close;
```

Please refer to the Output Delivery System's documentation for details on using ODS.

Conclusion

PROC PRINT is the default listing tool in the SAS Enterprise Guide, and it can be controlled to a certain point. You cannot control where the procedure will put info, only that it will put it on the list. If you want a procedure to do robust calculations, do not use PROC

**Before PROC REPORT, There Was PROC PRINT
Clarence Wm. Jackson, CSQA**

PRINT. But if you want to produce listings of observations and other quick lists, this is the right procedure to use.

Author info

Clarence Wm. Jackson, CSQA
CJac.csqa@sbcglobal.net