

What's Next?

Thomas J. Winn, Jr.

Texas State Auditor's Office, Austin, Texas

Abstract

Intended for attendees who have learned the material that was covered in the earlier presentations in the "Introduction to SAS" Section, this short paper concludes the Section by describing some recommendations for the next developmental steps which could be taken by beginning SAS programmers. The paper includes highlights of the following topics: writing customized reports (PROC PRINT, DATA _NULL_, PROC TABULATE, & PROC REPORT), the SAS macro language, and PROC SQL.

Recommendations for your next developmental steps

Let us assume that one has learned the material which was covered in the earlier presentations in the SSU 2001 "Introduction to SAS" Section. Now, I would like to conclude this Section by describing some recommendations regarding the next developmental steps that could be taken by beginning SAS programmers.

Specifically, I will include highlights of the following topics:

- ♦ Output Delivery System,
- ♦ customized report writing, using PROC PRINT, PROC REPORT, PROC TABULATE, and DATA _NULL_,
- ♦ the SAS macro language, and
- ♦ PROC SQL.

A working knowledge of these topics should be included in the basic toolkit of every SAS programmer.

Here are a few ways to take those next steps:

- ♦ Take a training class,
- ♦ Attend tutorial presentations at SAS users conferences,
- ♦ Consult with an expert,
- ♦ Read the documentation or other appropriate reference materials.

Most of the topics are covered in papers presented in the SSU 2001 Tutorials section!

Output Delivery System

Beginning with Version 7, the SAS System provided an ability to deliver procedure output in a flexible variety of file types and formats, through the SAS Output Delivery System (ODS). ODS combines raw data with table definitions to produce output objects, which can be sent to one or more ODS destinations. Currently, the available ODS destinations include: the Listing destination, the Output destination (SAS Data Set), the HTML destination, the Printer destination (PostScript, PDF, PCL), and the RTF destination.

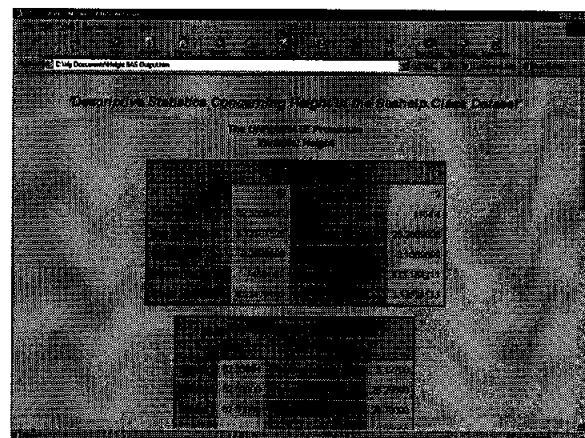
ODS destinations can be either open or closed. When a destination is open, ODS can send output objects to it. And whenever an ODS destination is closed, output

objects cannot be sent to it. ODS statements are used to control different features of the Output Delivery System. For example, they could be used to open, close, or manage an ODS destination. Moreover, they can be used anywhere in a SAS program.

Here is an example of the use of ODS statements to direct some SAS output to HTML:

```
ods html file='odshhtml-body.htm'  
      contents='odshhtml-contents.htm'  
      page='odshhtml-page.htm'  
      frame='odshhtml-frame.htm';  
proc univariate data=sashelp.class;  
  var height;  
  id name;  
  title 'Descriptive Statistics Concerning Height  
in the Sashelp.Class Dataset';  
run;  
ods html close;
```

And here is a portion of the HTML output that was generated:



In most cases, the default output style will be adequate. However, the programmer may customize ODS output by specifying style definitions, which affect the colors, font, size, etc.

Customized Reports with PROC PRINT

PROC PRINT has several features and options which can be used to customize the appearance of detail reports printed with this procedure.

- ♦ DATA=, DOUBLE, HEADING=, LABEL, N, NOOBS, OBS=, ROUND, SPLIT=, and UNIFORM,
- ♦ VAR, ID, BY, SUM, PAGEBY, SUMBY, FORMAT, TITLE, and FOOTNOTE.

PROC TABULATE

PROC TABULATE is used to build tabular reports containing descriptive statistical information, including

hierarchical relationships among variables. PROC TABULATE is particularly well-suited for preparing summary reports (where each row represents multiple observations). However, it is less well-suited for producing detail reports (single row for each observation).

Here is some SAS code for a simple tabular report:

```
libname company 'C:\Program Files\SAS
Institute\SAS\V8\core\sample';
proc tabulate data=company.empinfo
format=5.0;
class division gender;
keylabel n= '
all='Total';
label division = 'Division'
gender = 'Gender';
table division all, gender all / rts=30
misstext='0';
title 'Summary of Employee Information';
run;
```

Here is the associated report:

Division	Gender	Total
DEVELOPMENT		1
DEVELOPMENT	Male	1
DEVELOPMENT	Female	0
MARKETING		5
MARKETING	Male	5
MARKETING	Female	0
SALES		11
SALES	Male	11
SALES	Female	0
FINANCE		1
FINANCE	Male	1
FINANCE	Female	0
OPERATIONS		1
OPERATIONS	Male	1
OPERATIONS	Female	0
ADMINISTRATIVE		1
ADMINISTRATIVE	Male	1
ADMINISTRATIVE	Female	0
UNCLASSIFIED		1
UNCLASSIFIED	Male	1
UNCLASSIFIED	Female	0
TOTAL		20
TOTAL	Male	20
TOTAL	Female	0

PROC REPORT

PROC REPORT is a flexible procedure that can produce a variety of reports. PROC REPORT offers more control and customization than PROC PRINT. PROC REPORT can be used to produce both detail and summary reports; however, it is not as good as PROC TABULATE for producing hierarchical tables.

Here is some SAS code for a simple PROC REPORT step:

```
proc report data=sashelp.shoes headskip
headline missing;
column subsidiary product sales;
define subsidiary / order 'Subsidiary'
format=$15.;
define product / order 'Product'
format=$15.;
define sales / analysis sum 'Sales'
format=dollar15.2;
break after subsidiary / ol summarize skip
suppress;
title 'Product Sales According to Subsidiary';
run;
```

Here is the report which was produced:

Subsidiary	Product	Sales
American	Shoes	100,000.00
American	Socks	100,000.00
American	Shirts	100,000.00
American	Shoes	100,000.00
American	Socks	100,000.00
American	Shirts	100,000.00
British	Shoes	100,000.00
British	Socks	100,000.00
British	Shirts	100,000.00
British	Shoes	100,000.00
British	Socks	100,000.00
British	Shirts	100,000.00
French	Shoes	100,000.00
French	Socks	100,000.00
French	Shirts	100,000.00
French	Shoes	100,000.00
French	Socks	100,000.00
French	Shirts	100,000.00
German	Shoes	100,000.00
German	Socks	100,000.00
German	Shirts	100,000.00
German	Shoes	100,000.00
German	Socks	100,000.00
German	Shirts	100,000.00
Italian	Shoes	100,000.00
Italian	Socks	100,000.00
Italian	Shirts	100,000.00
Italian	Shoes	100,000.00
Italian	Socks	100,000.00
Italian	Shirts	100,000.00
Japanese	Shoes	100,000.00
Japanese	Socks	100,000.00
Japanese	Shirts	100,000.00
Japanese	Shoes	100,000.00
Japanese	Socks	100,000.00
Japanese	Shirts	100,000.00
Spanish	Shoes	100,000.00
Spanish	Socks	100,000.00
Spanish	Shirts	100,000.00
Spanish	Shoes	100,000.00
Spanish	Socks	100,000.00
Spanish	Shirts	100,000.00
Swedish	Shoes	100,000.00
Swedish	Socks	100,000.00
Swedish	Shirts	100,000.00
Swedish	Shoes	100,000.00
Swedish	Socks	100,000.00
Swedish	Shirts	100,000.00
Swiss	Shoes	100,000.00
Swiss	Socks	100,000.00
Swiss	Shirts	100,000.00
Swiss	Shoes	100,000.00
Swiss	Socks	100,000.00
Swiss	Shirts	100,000.00
TOTAL		1,000,000.00

Writing Customized Reports

FILE and PUT statements can be used in a DATA _NULL_ step to write special reports according to detailed specifications. The disadvantage is that pointer controls must be used to specify the placement of every item in the report.

The SAS Macro Language

The SAS Macro Language is a powerful programming tool ...

- ♦ for simplifying repetitive coding,
- ♦ for communicating information between program steps,
- ♦ for generating data-dependent SAS statements,
- ♦ for permitting conditional execution of SAS code, and
- ♦ for dynamically importing certain information from the SAS Supervisor.

Macros are stored text that contain entire blocks of SAS code, and which are identified by a name. The stored text can include SAS statements, literals, numbers, macro variables, macro functions, macro expressions, or calls to other macros.

Macro variables are used to facilitate symbolic substitution of strings of text, whereas macros can be used to manipulate SAS source statements. Macro information can be inserted at any point in a SAS program simply by referring to the macro entity by name, preceded by a special character, which distinguishes macro statements from ordinary SAS code.

PROC SQL

Structured Query Language (SQL) is a language that talks to a relational database management system. PROC SQL processes SQL statements that read and update tables. Besides being useful for queries, it also is a powerful tool for data manipulation.

PROC SQL uses Structured Query Language to ...

- ♦ retrieve and manipulate SAS data sets,
- ♦ create and delete data sets,

INTRODUCTION TO SAS

- ♦ add or modify data values in a data set,
- ♦ add, modify, or drop columns in a data set,
- ♦ create and delete indexes on columns in a data set.

Components of the SAS System

Do you remember this display from the very first "Introduction to SAS" presentation? It is a listing of many of the components of the SAS System.

- | | |
|---------------|-------------------------------|
| - Base SAS | - SAS/LAB |
| - SAS/ACCESS | - SAS/MDDb Server |
| - SAS/AF | - SAS/OR |
| - SAS/ASSIST | - SAS/QC |
| - SAS/CONNECT | - SAS/SHARE |
| - SAS/EIS | - SAS/SPECTRAVIEW |
| - SAS/ETS | - SAS/STAT |
| - SAS/FSP | - SAS/TOOLKIT |
| - SAS/GIS | - SAS/AppDev Studio |
| - SAS/GRAPH | - SAS/Enterprise Guide |
| - SAS/IML | - SAS/Enterprise Miner |
| - SAS/INSIGHT | - SAS Universal ODBC Driver |
| - SAS/IntrNet | - SAS/Warehouse Administrator |

After becoming familiar with Base SAS, novice SAS programmers might begin learning other pertinent components of the SAS System.

"What's Next?" Summary

Build on the foundation of knowledge you already have acquired. Your next developmental steps might include the following suggested topics:

- ♦ Output Delivery System,
- ♦ customized report writing, using PROC PRINT, PROC REPORT, PROC TABULATE, and DATA _NULL_,
- ♦ the SAS macro language,
- ♦ PROC SQL,
- ♦ other components of the SAS System.

Here are some ways by which you might take those next developmental steps:

- ♦ Take a training class.
- ♦ Attend tutorial presentations at SAS users conferences (Be sure to review the many excellent papers in the SSU 2001 Tutorials Section!).
- ♦ Consult with an expert.
- ♦ Read the documentation or other appropriate reference materials.

The next step is up to you.

Suggested References:

SAS Institute Inc., SAS OnlineDoc, Version 8
SAS Institute Inc., Getting Started With the SAS System, Version 8
SAS Institute Inc., SAS Language Reference: Concepts, Version 8
SAS Institute Inc., SAS Language Reference: Dictionary, Version 8, Volumes 1 and 2
SAS Institute Inc., SAS Procedures Guide, Version 8, Volumes 1 and 2

Author Information

Tom Winn
Texas State Auditor's Office
P.O. Box 12067
Austin, TX 78711-2067

Telephone: 512 / 936-9735
E-mail: twinn@sao.state.tx.us