

THE TABLE MACROS: A PATHWAY FOR CONVERTING
PROC QPRINT IN SAS VERSION 6

GEORGE DEMUTH, BURROUGHS WELLCOME CO.

Introduction

The Version 5 SAS supplemental procedure QPRINT provided an efficient way to create sophisticated data lists. The QPRINT procedure does not appear in Version 6.06 of SAS, and PROC REPORT is intended to offer advanced formatting capabilities for tables and data lists. As users move to implement SAS Version 6.06, they may need to convert the QPRINT routines to something compatible with the new system. A set of code, which will be referred to as the TABLE macros, was developed to simulate output and structure of the QPRINT procedure. The TABLE macros have been used with Version 6.06, and should be of interest to those that enjoyed the utility of PROC QPRINT in the past.

The TABLE Macros

The TABLE macros use a DATA _NULL_ with the FILE PRINT option to create customized output. A variety of centering options are allowed for the variable labels, titles, and footnotes. Like the QPRINT procedure, headers that span more than one column and multiple-level variable labels are permitted.

Using the TABLE macros does not require any knowledge of the QPRINT procedure, but descriptions of the macros may refer to the procedure for those interested in converting Version 5 code.

Two options for suppressing duplicate sorting values are available as options. The first repeats sorting values when a new page is reached, the second method creates blocks from the first sorting variable and attempts to keep the whole block on one page.

The TABLE macros were intended to simulate the actions that we used QPRINT for on a regular basis at Burroughs Wellcome Co. Hence, the macros offer only a subset of the QPRINT procedure's options. For example, the FOOT statement not duplicated, although TABLE macro footnotes may be left, right, or center justified. More importantly, multiple panels are not written on a page, so, the macros handle only single page output.

Using the TABLE Macros

There are a series of macros included under the umbrella of the TABLE macros. They will be described briefly, along with their required parameters. The following rules apply when using the macros:

1. The table must be initiated with a %TABLE call and terminated with a %CREATE call.
2. Variables appear in the order %VAR calls appear.
3. Spanning headers are centered over the variables contained within %SPAN and %SEND calls.
4. %TITLE and %FOOT statements may appear anywhere, but are printed in the order that they appear.
5. %PAGEBY may appear anywhere, and only one the last PAGEBY variable is used.

In the macro descriptions below, default parameter values are shown as the " = " values described below.

TABLE Macro Calls

%TABLE starts a table by initializing various counters. The line size, skip, and duplicate variable suppression option are all selected at this point. %TABLE parameters are:

- LSX = 132 Specifies the line size
- SKIPX = 0 Specifies the number of lines to skip at the top of the page (like SKIP = option).
- STYLEX = NOSORT Designates the sorting scenario, other options are REPEAT and BLOCK.
- SVAR = Required when a sorting option is selected. It lists the variables in the order to sort and suppress duplicate sorting values.

Ex. % table (lsx=80, skipx=4, stylex=BLOCK, svars= PATNO DATE);

%VAR enters a single variable into the output. %VAR parameters are:

- NAME Specifies the variable name.
- FORMAT Specifies the variable format.
- LABEL Specifies the variable label. The label may have unlimited levels and unlimited length (within reason). The ! character is used as a delimiter and one ! is required to begin a label. Additional !s indicate new levels.
- J = C Label justification, options are (L, R, or C).

Ex. %var(PATNO,3,!,Patient!ID);

%SPAN and %SEND specify a title that spans multiple variable columns. They are used together, i.e. a title is specified in %SPAN and will span the columns until it finds a %SEND call. They may be nested like HEAD statements in the QPRINT procedure, although delimiters are not permitted. The %SPAN parameters are:

- TEXT Text span the columns
- ULINE = Specifies a character to appear in a spanning string immediately below the TEXT and before the labels.

%SEND has no parameters.

Ex. %span (Demographic Info,uline = _);
%var(age,3,!,Age);
%var(race,\$racef,!,Race);
%var(sex,\$sexf,!,Sex);
%send;

%PAGEBY specifies a page-by variable. This variable should not appear in the sorting option variable list specified in %TABLE. %PAGEBY parameters are:

NAME specifies the name of the variable
 FORMAT specifies the format for the variable values
 LABEL specifies the variable label.

Ex. %pageby(drug,\$drugf.,Treatment);

%TITLE and %FOOT specify titles and footnotes for the output. The macros use a NOTITLES option in the FILE PRINT, so all titles must be entered within the macro. Both have a similar syntax, their parameters are:

TEXT specifies the text of the title or footnote.
 J = C/J = L specifies the justification for the string (L, R, or C).
 ADD1 = ,ADD2 = ,ADD3 = are used too add extra text to a title or footnote. This was intended for programmers who want keep the maximum column width of their program to 72 characters.

Ex. %title(This title is default center justified);

Ex. %foot(This line is default left justified);

%CREATE terminates the call to the TABLE macros and creates the output. The parameters are:

DS name of the data set.
 ULINE = character to appear in the underline.
 OLINE = character to appear in the overline.
 UBLKS = 1 number of blanks between header and beginning of data list.
 OBLKS = 0 number of blanks between data list and overline.

Ex. %create(demo,uline = __,oline = __);

Example

Sample Table Code

```
%table(lsx = 80,skipx = 4);
  %pageby(drug,$dfmt.,Dose);
  %var(pt,4.,!Patient);
  %span(Stratum Factors,uline = %str(-));
    %var(sex,$sexf6.,!Sex);
    %var(weight,4.1,%str(!Weight!%(lbs%)));
  %send;
  %span(Maternal Factors,uline = %str(-));
    %var(race,$racef5.,!Race);
    %var(smoking,$ynf3.,!Maternal!Smoking);
  %send;
  %span(Other Factors,uline = %str(-));
    %var(mbirth,$ynf3.,!Child!of!Multiple!Birth);
    %var(apgar,2.,!Apgar!Score);
  %send;
  %title(Pediatric Dosing Study);
  %title(Baseline Characteristics);
  %foot(trial6 sas);
%create(demo,uline = %str(-),oline = %str(-));
```

SAMPLE OUTPUT:

Pediatric Dosing Study Baseline Characteristics

Dose = 100 mg.

Patient	Sex	Stratum Factors		Maternal Factors		Other Factors	
		Weight (lbs)	Race	Maternal Smoking	Child of Multiple Birth	Apgar Score	
1001	Female	7.8	White	No	Yes	6	
1002	Male	8.1	White	Yes	No	7	
....	
1039	Male	6.7	Black	No	No	7	
1040	Male	7.7	White	Yes	No	7	

trial6 sas

Macro Code for Tables

The code for the macros was too long to appear in this paper. An ASCII file containing the code can be obtained from the author. The author may be contacted at:

Burroughs Wellcome Co.
 3030 Cornwallis Rd
 Research Triangle Park, NC 27709

Notes

SAS is a registered trademark of SAS Institute, Cary, NC, USA.