Paper 59-27

Quick Results in PROC REPORT

J. Meimei Ma, Quintiles, Research Triangle Park, North Carolina
Sandra Schlotzhauer, Schlotzhauer Consulting, Chapel Hill, North Carolina
Maria Ilieva, Quintiles, Research Triangle Park, North Carolina

ABSTRACT

What tools do you use to generate reports or listings? PROC PRINT, PROC TABULATE, or DATA _NULL_? Perhaps it is time to check out the flexibility and features of PROC REPORT. An easy way to begin is to follow the "80/20" rule by using the interactive REPORT windows to get a head start on the syntax and options. After quickly creating a program for the essential parts of a report, you will have time left over to consider enhancements that may require more complex features.

This beginning tutorial provides examples that you can follow to learn the steps involved in building reports interactively. The intent is to provide a starting point for making better use of PROC REPORT, not to present an overview of available features. The methods are for Version 8 but most also apply to Release 6.11/6.12. Although ODS and the STYLE options are shown, the focus is on basic features of REPORT that exist in Release 6.11 or later.

REPORT PROCEDURE BASICS

The REPORT procedure was designed as a flexible report writing tool that combines features of the PRINT, MEANS, and TABULATE procedures along with "PUT" capabilities commonly used in the DATA step. The features have remained stable since Release 6.11. You can use PROC REPORT in three ways:

1) windowing environment with prompting
2) windowing environment without prompts
3) batch program code

Using the windowing environment still creates program code, you just do not have to remember the syntax. We believe that only using the batch approach does not take full advantage of using PROC REPORT over "traditional" methods.

REQUIRED ELEMENTS

The fundamental elements of using PROC REPORT are straightforward. At NESUG'99, Moriak provided a concise description of three primary steps and four common options. Another way to show the elements and options is shown in Table 1.

Element	Option(s)
1) Initiate procedure (PROC REPORT)	a) Subset data (WHERE)
	b) BY processing
2) Assign report variables (COLUMN list)	c) special headers
	d) specify a statistic
3) Define report variables (DEFINE statements)	e) Calculate new variables
	f) Add summary lines

Table 1. REPORT Procedure Elements and Options
VARIABLE USAGE TYPES

Every variable put into a report must be assigned to a usage type. The assignment is made in the DEFINITION window and in the DEFINE statement. The six types are: GROUP, ORDER, DISPLAY, ANALYSIS, COMPUTED, and ACROSS.

DISPLAY lists values as they appear in the data set. DISPLAY is the default for character variables if no ORDER or GROUP variables exist.

GROUP applies to "class" variables. The report can be separated for each value of a GROUP variable with automatic summary statistics for some or all of the other variables. For instance, the sum can be calculated and displayed for all ANALYSIS variables for each different value of a GROUP variable. Sort order can be controlled in several ways.

ORDER sorts the data and creates groups for summary statistics.

ANALYSIS is the default type for numeric variables. Summary statistics such as SUM or MEAN can easily be calculated and displayed for the entire data set or each group or sub-group.

See the *SAS® Guide to the Report Procedure* for explanations of COMPUTED and ACROSS.

BASIC PROGRAM

Below is an example of a program with basic elements. NAME and SEX are character, while AGE and HEIGHT are numeric. The report will be sorted by SEX and NAME. AGE and HEIGHT will be usage type ANALYSIS by default. What do you think the HEADLINE option will do?

```
PROC REPORT DATA=class   HEADLINE  ;
  COLUMN sex  name  age  height  ;
  DEFINE sex / ORDER  WIDTH=3  ;
  DEFINE name / GROUP  WIDTH=10  ;
  DEFINE age / FORMAT=BEST4.  ;
  DEFINE height / FORMAT=4.1  WIDTH=6  ;
RUN ;
```

FUN FEATURES

The REPORT procedure includes many features. Here are just a few you can explore. Many are easier to learn initially by using REPORT windows so you can concentrate on the possibilities and results instead of the correct syntax.

- Summary statistics using BREAK and RBREAK on ORDER or GROUP variables, with or without detail lines and/or subtotals
- COMPUTE blocks to create new variables
- COMPUTE block LINE statements (like PUT)
- Fancy column headers (spanned headers)

- Easy separator lines for summary statistics
- BOX option (like PROC TABULATE)

We will show a few of these in the examples. The references provide more examples and ideas.

USING THE REPORT WINDOW

There is more than one way to get into the REPORT windowing environment. You can

1) Submit a PROC REPORT program with the WINDOWS option (default).

2) Use a menu selection from the main menu.
Rel. 6.12:   **Globals > Present > Create Report**
Ver. 8:   **Tools > Report Editor**
        or   **Solutions > Reporting > Design Report**

You can save the code once you have the report designed in the REPORT window. You can save the code as a SAS program in an ASCII text from (as demonstrated in Example 1) or save the source code as a catalog entry.

USING PROMPT

You may want use the PROMPT option when starting a REPORT window. To do so, start the procedure by submitting a PROC REPORT statement instead of using the menu. For example:

```
PROC REPORT DATA=libname.CLASS PROMPT; RUN;
```

You will start off selecting variables of interest in the order desired. This method is a workaround to the default strategy of only including the first few variables in the data set that fit in the linesize available. After the REPORT window appears, you will probably want to exit prompting instead of continuing to modify every variable using the prompts. The full DEFINITION window may be quicker when making multiple modifications.

Using prompting is especially helpful in Version 8 because the Report Layout window is no longer available. Although you can use menu choices to add variables (see Example 2) or rearrange the order of variables, selecting all variables in the correct order at the start can be quicker.

EXAMPLE 1: CLASS WITH MEANS

For the first example, we use the CLASS data set to create a report sorted by SEX with means calculated for HEIGHT, WEIGHT, AGE. See Appendix 1 for the data used. You should work through the example with your own SAS software at some point. Feel free to experiment.

The steps shown in Appendix 1 create the report using Version 8. Figures 1 through 5 show the work in progress. Be sure to look at both the front window and the interim report shown in the background in the REPORT window. Figure 7 includes the final code in upper case.

The steps to create the report using Version 6.12 are essentially the same as outlined in Appendix 1, but the menu selections are often different (see Figure 6). See Ma and Schlotzhauer for details.

EXAMPLE 2: CLASS WITH RANGE LINE

For the next example, suppose you want to add a customized summary line at the end of the report created in Example 1 that shows the range of Age. Once you have the core code for the main report, adding the extra code is simple.

The trick is to calculate the minimum and maximum for Age and then use a COMPUTE block with LINE statements to write out the information. The added code is in lowercase in Figure 7. The idea comes from Example 3 in the *SAS Guide to the Report Procedure*. Figure 8 shows the resulting report.

EXAMPLE 3: ADVERSE WITH VERSION 8

Have you had to deal with long text fields that were too long to display without wrapping onto multiple lines? What about having more than one record for the same identifier? The REPORT procedure can make it easy.

Consider a data set containing adverse events from a clinical trial. A patient is expected to provide data for multiple visits and may have more than one event per visit. The variables that contain descriptions of the events are text and usually fairly long.

By following the steps in Appendix 2, it becomes a relatively easy task to produce a readable report by selecting variables and using the FLOW option. Choosing the variables of interest is a point and click operation. Repeated values of PATID or VISIT are quickly eliminated once they are defined as GROUP variables. Using the DEFINITION window to adjust column widths makes it simpler to decide what will fit in the desired linesize.

Try these ideas on any data set with long character variables. You can accomplish the same things in Release 6.12, but will need to adjust to the old menus for most steps. No new Version 8 features are used.

ODS FEATURES IN VERSION 8

The advantage of Version 8 is mainly because of the Output Delivery System (ODS). With ODS, *styles* control colors, fonts, and other visual aspects of output. You can override the default template for PROC REPORT output by specifying style options for various parts of a report to enhance the results. Of course, what looks good or bad depends on whether you want HTML, Rich Text Format (RTF), and perhaps even Postscript or PDF. See Haworth paper for HTML ideas. You can also create customized templates.

Creating an HTML file is as easy as adding two ODS statements and turning off the WINDOWS option. When you submit the program, the Results Viewer window will automatically display the HTML results while the standard report will go to the Output window as always.

```
ODS HTML FILE="c:/home/neat.html";
 PROC REPORT DATA=class   HEADLINE   NOWD
    STYLE (HEADER)={BACKGROUND=RED}  ;
    COLUMN sex  name  age  height  ;
 RUN ;
ODS HTML CLOSE ;
```

The code below uses a template provided with the software.

```
ODS HTML FILE="c:/home/neat.html"
    STYLE=D3d;
 PROC REPORT DATA=class   HEADLINE  NOWD;
   COLUMN sex  name  age  height  ;
 RUN ;
ODS HTML CLOSE ;
```

You can experiment with the stored templates using menu options only. Check out Tools > Options > Preferences and select the Results tab.

CONCLUSION

In spite of the existence of a windowing environment, PROC REPORT has an underlying set of programming statements to learn just like other SAS procedures. Learning the statements, options, and syntax can be a challenge when you are also trying to produce reports for a deadline because of the breadth of functionality of the procedure. By learning the procedure interactively, you can quickly build 80% (or more) of the final report you need. Saving the generated program code lets you reproduce the report in the future, learn the language, and add your own programming statements for the final 20% of report features not available in a window.

REFERENCES

Deem, K. (1999), "Customizing Break Lines When PROC REPORT 'Does Not Computer'," *Proc. of the Twelfth Annual North East SAS Group Conference*, Cary, NC: SAS Institute Inc., 320–321. (Idea for adding summary lines for clinical trials reports.)

Haworth, L. (2001), "ODS for PRINT, REPORT and TABULATE," *Proc. of the Twenty-sixth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 3-26.

Hull, B. (2001), "Now There Is an Easy Way to Get to Word, Just Use PROC TEMPLATE, PROC REPORT, and ODS RTF," *Proc. of the Twenty-sixth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 163-26.

Ma, J.M. and Schlotzhauer, S. (2000), "Fast Track to PROC REPORT Results" *Proc. of the Twenty-fifth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 67-25. (Release 6.12 examples.)

Moriak, C. (1999), "So, You Want To Learn PROC REPORT?," *Proc. of the Twelfth Annual North East SAS Group Conference*, Cary, NC: SAS Institute Inc., 547–554. (Basic introduction to usage and syntax with medical utilization example.)

Moriak, C. (1999), "Using the ID Options in PROC REPORT," *Proc. of the Twelfth Annual North East SAS Group Conference*, Cary, NC: SAS Institute Inc., 328–329. (Simple examples using ID.)

Pass, R. (2000), "PROC REPORT – Land of the Missing OBS Column," *Proc. of the Twenty-fifth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 105-25. (How to simulate OBS numbering using COMPUTE blocks.)

Pass, R. and Ewing, D. (2000), "So You're Still Not Using PROC REPORT. Why Not?" *Proc. of the Twenty-fifth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 147-25. (Focus on using REPORT in batch mode.)

SAS Institute Inc. (1993), SAS Technical Report P-258, *Using the REPORT Procedure in a Nonwindowing Environment, Release 6.07*, Cary, NC: SAS Institute Inc. 276 pp.

SAS Institute Inc. (1995), *SAS Guide to the Report Procedure: Reference, Release 6.11*, Cary, NC: SAS Institute Inc. 123 pp. (Good examples.)

SAS Institute Inc. (1999), *The Complete Guide to the SAS Output Delivery System*, Cary, NC: SAS Institute Inc.

Schneider, L.M. (2001), "Format Challenges with PROC REPORT," *Proc. of the Twenty-sixth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 89-26.

Trenery, D. (1999), "Jazzing Up Your Reports – Some Tricks with PROC REPORT," *Proc. of the Twenty-fourth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., 540–542. (Three easy enhancements for clinical trials reports.)

Yen, P. and Gudmundson, J. (2000) "One Macro Does It All – Advanced Enhancement of PROC REPORT," *Proc. of the Twenty-fifth Annual SAS Group Intl. Conference*, Cary, NC: SAS Institute Inc., paper 32-25. (Clinical trials methods for statistical tables.)

CONTACT INFORMATION

Maria Ilieva
Quintiles
P. O. Box 13979
Research Triangle Park, NC 27709
     Voice: 919-998-7424
     Internet: maria.ilieva@quintiles.com

Sandra Schlotzhauer
Schlotzhauer Consulting LLC
400 Gibbon Drive
Chapel Hill, NC 27516
     Voice: 919-933-6341
     Internet: sdschlotz@earthlink.net

Please use e-mail if you are interested in receiving a copy of the CLASS or ADVERSE data set.

APPENDIX 1: CLASS EXAMPLE

Follow these steps to create a report of CLASS using Version 8. The data adds BIRTHDAY to the CLASS data set shipped with Base SAS software. A DATA step to create the data set is shown below.

1) Start up SAS software, Version 8
2) Establish a libname to reference the CLASS data set (assumes CLASS exists)
3) Add a title
   a) Type **title** in the Command box to bring up the Title window
   b) Type in TITLE1, something like "Class Example"
4) Get into interactive REPORT window
   a) Select **Tools > Report Editor**
   b) Select library that contains CLASS
   c) Select CLASS data set
   d) Click on **OK** to get into REPORT window
5) Change order of variables
   a) Select SEX
   b) Select **Edit > Move > Left of the Next Selected Item** (see Figure 1)
   c) Move AGE to before BIRTHDAY
   d) Click on **Apply** to preview results
   e) Click on **OK** to close window and return to REPORT window
6) Set options related to column headers
   a) Select **Tools > Options > Report** to get ROPTIONS window
   b) Click on HEADLINE and HEADSKIP buttons
   c) Close window and return to REPORT window
7) Sort report by SEX
   a) Double-click on SEX (column header) to open DEFINITION window, see Figure 2
   b) Click on ORDER button
   c) Click on **Apply** to preview
   d) Close window and return to REPORT window (click on **OK**)
8) Set up summary by SEX
   a) Check that SEX is still highlighted
   b) Select **Edit > Summarize Information > After Item** to open BREAK window
   c) Click on buttons to select:
      Overline summary
      Skip line after break
      Summarize analysis columns
   d) Close window and return to REPORT window

9) Set up overall summary
   a) Select **Edit > Summarize Information > At The Bottom** to open BREAK window, see Figure 3
   b) Click on buttons to select:
      Double overline summary
      Skip line after break
      Summarize analysis columns
   c) Select a color
   d) Close window and return to REPORT window
10) Clean up definitions for analysis variables
   a) Modify Height
      • Double-click on HEIGHT (column header) to open DEFINITION window
      • Change Format by typing *6.1* after highlighting existing value
      • Change Statistic to *MEAN* after highlighting existing value
      • Optionally, click on APPLY to preview
      • Close window
   b) Modify Weight for Format: *6.1* and Statistic: *MEAN*
   c) Modify Age for Format: *BEST4.* and Statistic: *MEAN*
   d) Modify Birthday
      • Change Type to DISPLAY by clicking on button (Statistic will blank out)
      • Change Format (optional)
11) Add a header across Height and Weight columns
   a) Highlight HEIGHT (column header)
   b) Select **Edit > Span to next selected item**
   c) Click on WEIGHT, cursor should move to just above HEIGHT
   d) Type *-VITAL STATISTICS-* (hyphen before and after text) starting at beginning of span area, see Figure 4
   e) Select **View > Refresh**
12) Look at REPORT source code and save program to a text file
   a) Select **Tools > Report Statements** (you cannot edit these statements), see Figure 5
   b) Select **File > Save as > Write to file**
   c) Provide an appropriate destination and file name
   d) Close window
13) Exit REPORT window
   a) Select **File > Close** (not **File > Exit**)
   b) Click **OK** to confirm leaving REPORT window

```
title "Create CLASS";
data libname.class;
input Name $ Sex $ Age Height Weight
      Birthday:mmddyy10.;
  format birthday mmddyy10.;
cards;
Alfred  M  14    69     112.5   01/22/1984
Alice   F  13    56.5   84      06/15/1985
Barbara F  13    65.3   98      07/23/1985
Carol   F  14    62.8   102.5   02/14/1984
Henry   M  14    63.5   102.5   02/24/1984
James   M  12    57.3   83      10/15/1986
Jane    F  12    59.8   84.5    10/11/1986
Janet   F  15    62.5   112.5   04/03/1983
```

```
Jeffrey M  13    62.5   84      08/14/1985
John    M  12    59     99.5    05/12/1986
Joyce   F  11    51.3   50.5    01/04/1987
Judy    F  14    64.3   90      03/28/1984
Louise  F  12    56.3   77      03/15/1986
Mary    F  15    66.5   112     06/15/1983
Philip  M  16    72     150     06/14/1982
Robert  M  12    64.8   128     04/20/1986
Ronald  M  15    67     133     09/11/1983
Thomas  M  11    57.5   85      02/02/1987
William M  15    66.5   112     09/22/1983
run;
proc contents data=libname.class position;
run;
```

APPENDIX 2: ADVERSE EXAMPLE

These are the steps to follow to create the report of ADVERSE in Figure 10. The principles apply to any data set of adverse events from a clinical trial database or any data set with repeating identifiers and long text fields.

Pos#	Variable	Type/Len	Notes
1	PATID	num	Subject ID number
4	VISIT	num	LABEL=Visit Number
6	AEDESC	$30	AE verbatim description
16	ADVRS_C	$70	AE code text
8	ONDAT	$8	Onset Date
9	RESDAT	$8	Resolved Date

1) Start up SAS software, Version 8.
2) Establish a LIBNAME to the data set.
3) Get into the interactive REPORT window.
   a) Select **Tools > Report Editor**. (Alternatively, **Solutions > Reporting > Design Report.**)
   b) Select the library and the ADVERSE data set.
   c) Select **OK** to get into the REPORT window.
4) Select **OK** to exit the message window that tells you REPORT shows only the first few variables (because they cannot all fit on one line).
5) Delete unwanted variables.
   a) Select SITE by clicking on the variable.
   b) Select **Edit > Delete** (or press DELETE key).
   c) Repeat for the INVEST and AEDATE variables.
6) Add additional variables.
   a) Select AEDESC, which is the column after which you want to add variables to the report.
   b) Select **Edit > Add Item > Data Column > Right**
   c) In the Data Columns window, select ADVRS_C, ONDAT, and RESDAT. The selected variables move to the top of the list in the window, and asterisks appear left of the variable names.
   d) Select **File > Accept Selection**. See Figure 9.
   e) Depending on linesize settings and variable lengths, the new variables may not display on the current page of the report. You can select **View > Next Page** to confirm that the variables were added. Select **View > Previous Page** to return to the first page. In situations where the variables should be able to display on a single page, you may need to select **View > Refresh** to see the effect of adding variables.
7) Set Report Options.
   a) Select **Tools > Options > Report**.
   b) In the ROPTIONS window, select **HEADLINE**, **HEADSKIP** and **MISSING**.
   c) Select the **backslash** (/) used in SPLIT= and change to insert mode using the **Insert** key.
   d) Enter an asterisk (*****).
   e) Select **OK**.
8) Clean up identifiers: PATID and VISIT.
Eliminate repeated values for identification variables by setting the usage type to GROUP. Here PATID and VISIT are likely to have repeated values. But, usage type is set to ANALYSIS initially for both variables because they are numeric. Note that all character variables to the right will change from DISPLAY to GROUP.
   a) Modify PATID.
      i) Select PATID.
      ii) In the Definition window, change the header

      to *Subject*ID*.
      iii) Change Usage to **GROUP**.
      iv) Select **Apply**, view the results, and select **OK** to close the window.
   b) For VISIT, change Usage to **GROUP**. Note that this variable had an assigned label in the data set, so REPORT used the label as the column header. If desired, you could change the label used in the report by changing the column header in the Definition window.
9) Shorten long text columns and flow the text.
   a) Select AEDESC by double-clicking on the variable name.
   b) In the Definition window, change the header to *Verbatim*Term*.
   c) Select **FLOW**.
   d) Change the value of WIDTH to *20*.
   e) Select **Apply** and view the results. The objective is to get all variables to appear on a single page assuming linesize is set for landscape printing.
   f) Select **OK** to exit the Definition window.
   g) Change ADVRS_C to also have a WIDTH of *20*, **FLOW**, and a header of *Coded*Adverse Event*Term*. Select **View > Next Page** if needed.
10) Modify date variables: RESDAT, ONDAT.
   a) Select RESDAT by double-clicking on the variable name to get the DEFINITION window.
   b) Select the header and change it to *Resolution*Date* by typing in the new text.
   c) Select **Apply** to view the results.
   d) Select the Width value (default of 8) and change it to *10*.
   e) Select **Apply** and view the results.
   f) Select **OK** to exit the Definition window.
   g) Change ONDAT using a similar process and a header of *Onset*Date*.
11) Add a spanned header across several columns.
   a) Select AEDESC (Verbatim Term) as the first variable under the spanned header.
   b) Select **Edit > Span to next selected item**.
   c) Select RESDAT (Resolution Date), which is now on the same page as AEDESC.
   d) Locate the cursor at the beginning of the entry field above the existing column headers. Type in *-Adverse Event Details-* being sure to use a single hyphen at each end.
   e) Select **View > Refresh** to view spanned header.
12) Add a title.
   a) If desired, use the Title window to enter titles. Bring up the window using the Command box.
   b) Select **View > Refresh**.
13) View and save the report code.
   a) To view the code, select **Tools > Report Statements**. See Figure 10.
   b) With the Source window open, select **File > Save As** to save the file as a SAS program.
   c) Or, to save the code as a catalog entry, select **File > Save Report** and specify a libname, catalog name, and description for the report.
14) Finish with REPORT.
   a) Select **File > Close** (not **File > Exit**).

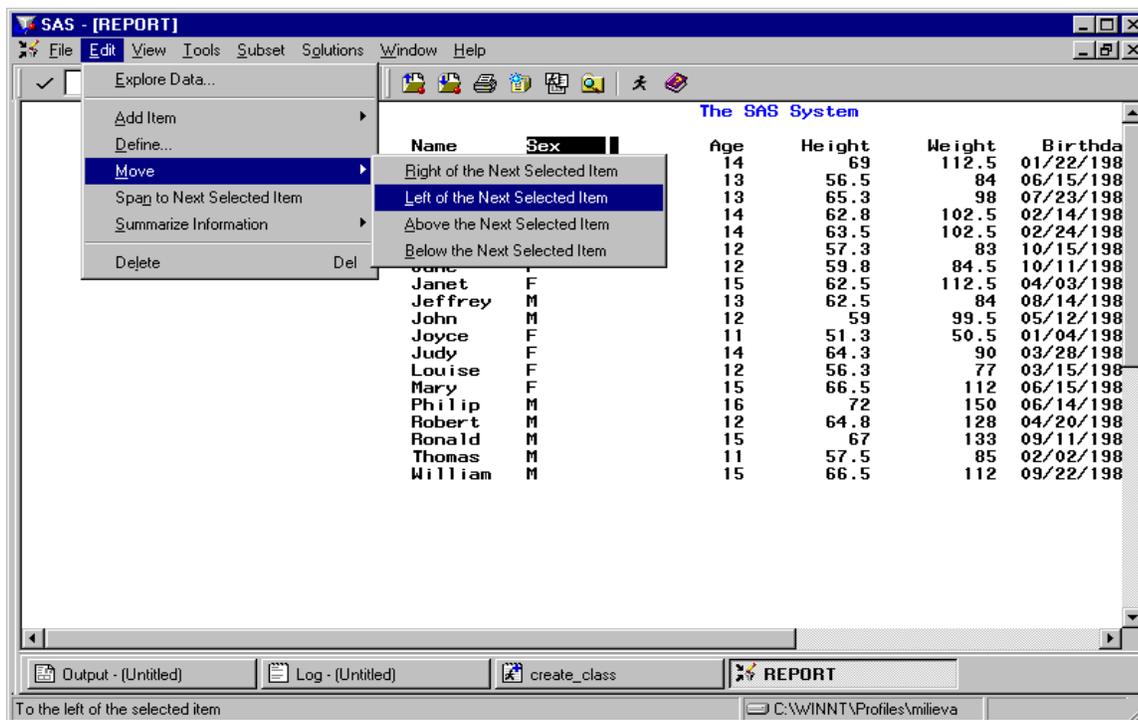## Example 1: CLASS Report Using Version 8

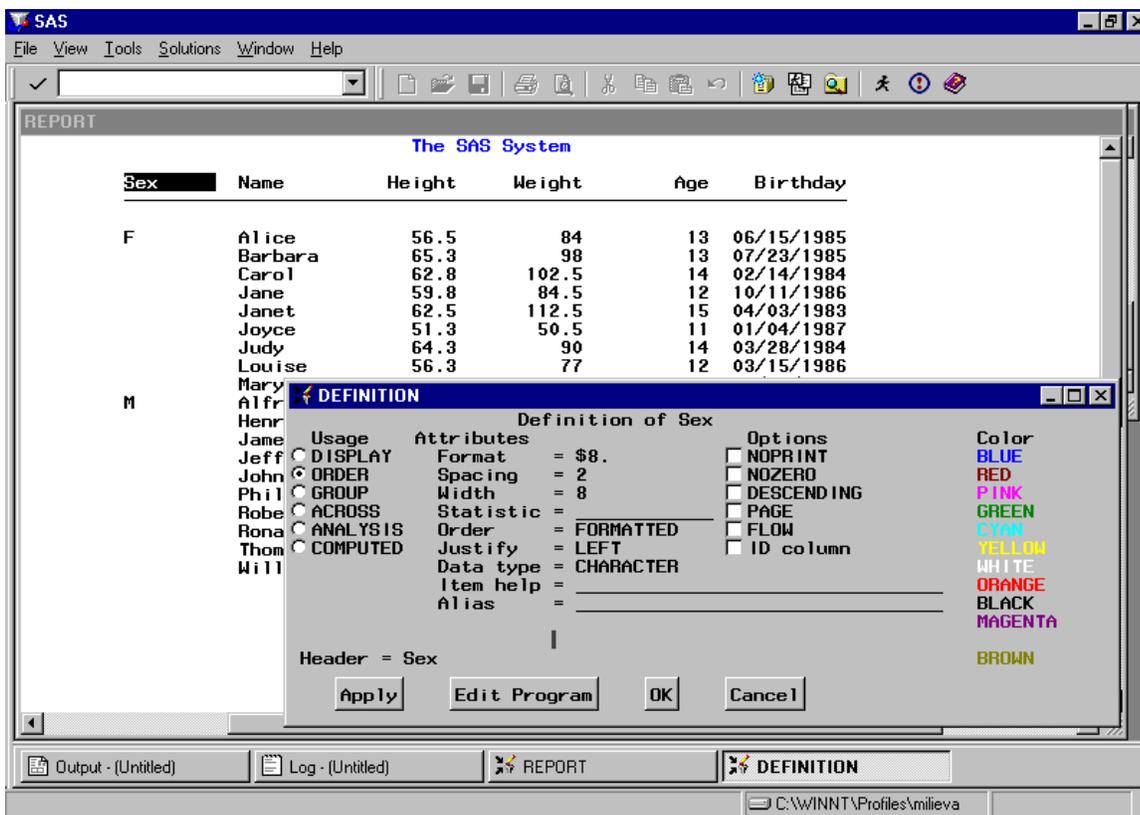Figure 1. Menu Choices to Move Variables - before any changes made to default report

Figure 2. DEFINITION Window - after SEX defined as an ORDER variable

Example 1: CLASS Report Using Version 8 (continued)

Figure 3. BREAK Window - summary on SEX values applied but before overall summary

Figure 4. Adding Spanned Header - report almost complete
Variables re-ordered and formatted, mean values added as summary statistics, title added

Example 1: CLASS Report Using Release Version 8 (continued)

Figure 5. Microsoft Internet Explorer – ODS Results

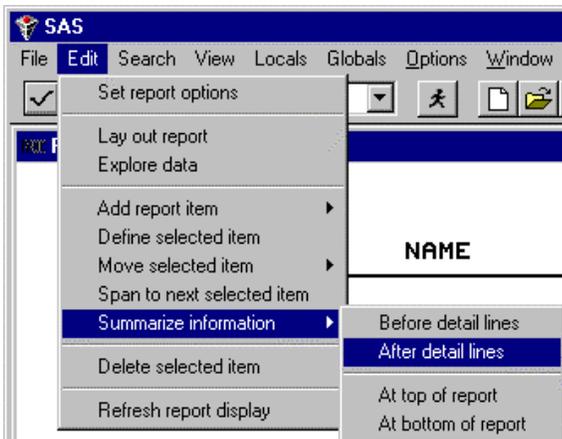## Example 1: Edit Menu Differences Between Release 6.12 and Version 8

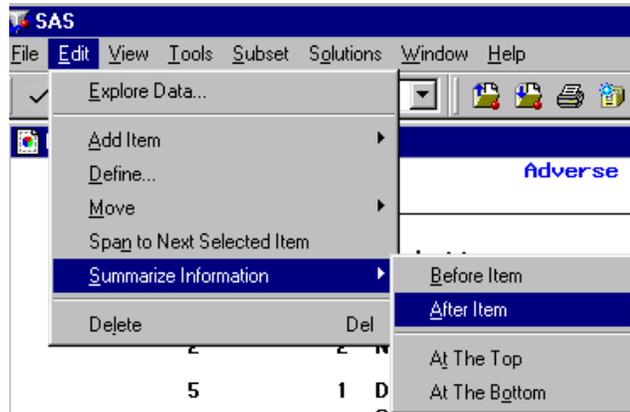Figure 6a. REPORT Edit Menu in Release 6.12

Figure 6b. REPORT Edit Menu in Version 8

Report Options under Tools > Options > Report
LAYOUT window no longer available
Refresh under View > Refresh

Example 2: CLASS Report With LINE Statements Using Version 8

```
title'CLASS EXAMPLE 2 - RANGE STATEMENT' ;

PROC REPORT DATA=indata.CLASS LS=132 PS=55 NOWD SPLIT="/" HEADLINE HEADSKIP CENTER ;
COLUMN  SEX NAME HEIGHT WEIGHT AGE BIRTHDAY
         age=agemin
         age=agemax ; /* For range Statement */
DEFINE  Sex / ORDER FORMAT= $8. WIDTH=8     SPACING=2   LEFT "SEX" ;
DEFINE  Name / ORDER FORMAT= $8. WIDTH=8     SPACING=2   LEFT "NAME" ;
DEFINE  Height / MEAN FORMAT= 6.1 WIDTH=9     SPACING=2   RIGHT "HEIGHT" ;
DEFINE  Weight / MEAN FORMAT= 6.1 WIDTH=9     SPACING=2   RIGHT "WEIGHT" ;
DEFINE  Age / MEAN FORMAT= best4. WIDTH=9     SPACING=2   RIGHT "AGE" ;
DEFINE  Birthday / DISPLAY FORMAT= MMDDYY10. WIDTH=10    SPACING=2   RIGHT "BIRTHDAY" ;
define agemin/analysis min noprint;    /* For range Statement */
define agemax/analysis max noprint;
BREAK AFTER SEX / OL SKIP SUMMARIZE ;
RBREAK AFTER  / DOL SKIP SUMMARIZE COLOR=BLUE     ;

compute after ;   /* Puts range statement at bottom of report */
  line ' ' ;
  line @35 29*'-' ;
  line @35 '| Age ranged from 'agemin 2. +1 'to ' agemax 2. '. |' ;
  line @35 29*'-' ;
endcomp ;
RUN;
```

Figure 7. Enhanced Program Code - adding customized summary of Age range using LINE

```
                    CLASS EXAMPLE 2 - RANGE STATEMENT
          SEX        NAME       HEIGHT      WEIGHT         AGE    BIRTHDAY

          F          Alice        56.5        84.0          13   06/15/1985
                     Barbara      65.3        98.0          13   07/23/1985
                     Carol        62.8       102.5          14   02/14/1984
                     Jane         59.8        84.5          12   10/11/1986
                     Janet        62.5       112.5          15   04/03/1983
                     Joyce        51.3        50.5          11   01/04/1987
                     Judy         64.3        90.0          14   03/28/1984
                     Louise       56.3        77.0          12   03/15/1986
                     Mary         66.5       112.0          15   06/15/1983
                                 _____      _____        ____
          F                        60.6        90.1        13.2

          M          Alfred       69.0       112.5          14   01/22/1984
                     Henry        63.5       102.5          14   02/24/1984
                     James        57.3        83.0          12   10/15/1986
                     Jeffrey      62.5        84.0          13   08/14/1985
                     John         59.0        99.5          12   05/12/1986
                     Philip       72.0       150.0          16   06/14/1982
                     Robert       64.8       128.0          12   04/20/1986
                     Ronald       67.0       133.0          15   09/11/1983
                     Thomas       57.5        85.0          11   02/02/1987
                     William      66.5       112.0          15   09/22/1983
                                 _____      _____        ____
          M                        63.9       109.0        13.4

                                =========   =========   =========
                                   62.3       100.0        13.3

                    -----------------------------
                    | Age ranged from 11 to 16. |
                    -----------------------------
```

Figure 8. Complete Report

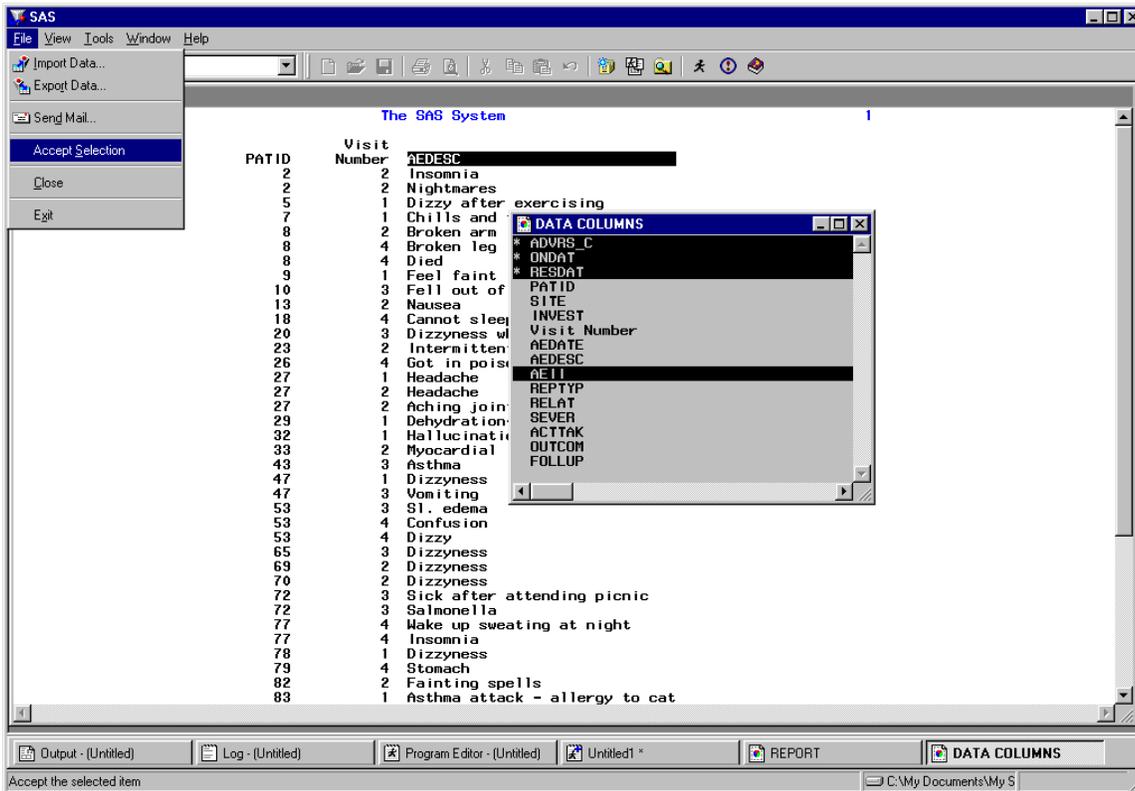## Example 3: ADVERSE Report Using Version 8

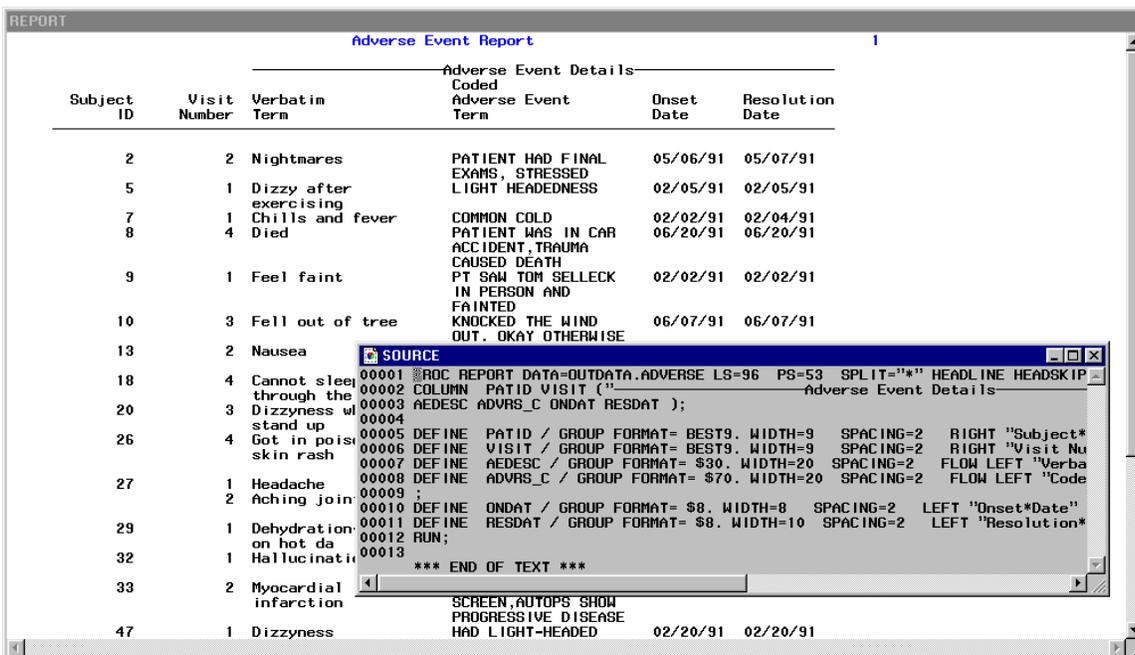Figure 9. DATA COLUMNS Window - use File menu when done selecting variables

Figure 10. Program Code and First Screen of Report