

USING SAS/AF® TO GENERATE THE DATA NULL STATEMENTS
COMMONLY USED IN CUSTOM REPORT WRITING

W. M. Cannon, J. C. Bullard
A. H. Robins Co., Richmond Virginia

ABSTRACT

Using PUT statements within the DATA NULL data step, in conjunction with carriage control, provides a SAS® user ultimate control over output when generating custom reports. This level of regulation is not available when using packaged procedures such as PROC TABULATE. However, additional expense is incurred to attain the power and flexibility afforded. Typically, the DATA NULL requires a certain degree of expertise, an increase in original programming time and because errors are also more prevalent, additional computer time.

The system we offer permits novice SAS users to easily generate otherwise complex and tedious title, header, and footer statements for the DATA NULL data step. This process will save development time and provide assurance that more programs will execute correctly on initial runs, conserving valuable computer resources. And, unlike similar software, the SAS code is output to an external file so that the more practiced SAS user can easily enhance final report layouts even beyond the capability of this system. This is software developed for the shop that chooses to maximize productivity through the integration of assorted SAS products, driven by an affable SAS/AF manager, that can quickly satisfy most custom reporting needs.

INTRODUCTION

The Gennull System was developed to fulfill the need for an easy and powerful technique to display SAS data sets in final report format. The system satisfies the reporting requirements of our statisticians, scientific editors, and the FDA. Gennull is versatile in design and functions in a dynamic environment of ever-changing report criteria, guidelines, data set structure, and naming conventions. The system is not tailored to a specific protocol or study type so that its utility is preserved over time, and maintenance requirements are minimal. This system eliminates the tedious task of column counting associated with most output schemes that use print files and enables the user to edit the output mechanism to meet the requirements of the format precisely. Gennull combines the ease and convenience of using packaged output procedures with the power and flexibility of the data step.

Gennull is easy to use; an operator simply

enters a single command (gennull) that evokes the interactive system. SAS/AF takes over and prompts the user through a series of customized SAS/AF and SAS/FSP® screens where information is gathered regarding the attributes of the display format. The user is given an opportunity to select the data set for output, enter up to 10 title statements, select the variables to be displayed, and revise the SAS code before execution if necessary. The entire process is supported by extensive error-checking capabilities that provide helpful hints in the event of an incorrect or undesired entry. Also, following the help screens, a second chance to enter the information is provided so that successful report generation is typically achieved on the first execution of the system. Finally, the resultant DATA NULL data step can be run as a program in itself^a, eliminating the need to run long jobs repeatedly for editing purposes; or it can be appended to one or more existing programs.

SYSTEM PREREQUISITES

Gennull requires that several programming tasks will be completed before invoking the system. These tasks are as follows:

1. A permanent SAS data set must be created.

Unlike PROC PRINT, the Gennull System is not used within a SAS program, but is called independently. Furthermore, Gennull is not designed for data manipulation; it is solely an instrument of output. For these reasons it is essential to create, structure, and store permanent SAS data sets before using the system.

2. The variables in this data set should be labeled.

Gennull will use these descriptive labels in lieu of the variable names within the spanner of the table. The length of these labels is used by the system in the calculation of the total width required to display the variable. The minimum required width is the greater of the longest label segment versus the length of the variable format. The user can control the way the labels are partitioned into column heads by delimiting the desired breaks with dollar signs. For example, if the variable DBP is labeled

'Diastolic\$Blood\$Pressure', and the format length is 8; then the width requirement for the variable is 9 columns (the length of the character string 'Diastolic'), and the column head will be written to 3 rows.

3. A permanent library of formats should be created.

If Gennull is to utilize user-written formats in the report, a permanent library of these formats should be produced before invoking the system.^b

IMPLEMENTATION OF APPLICATION

The Gennull EXEC (Appendix A) provides the user with a transparent interface to the output system. This EXEC uses PROC DISPLAY to exhibit the system screens built by using proc build and initially directs control to the main menu by using the CATALOG = option. Also, this EXEC issues the needed file definitions and discards extraneous work files from the account upon exiting the system. The main menu (Fig. 1) enables selection of either of the 2 major components of the system, i.e., creation of titles (PF2), and creation of the DATA_NULL_data step with which these titles are associated (PF3).

Up to 10 titles can be entered onto the AF title entry screen (Fig. 2). The system inserts these entries into SAS TITLE statements and automatically deletes all unused subsequent title statements. The resultant block of SAS code containing the titles will be appended to the DATA_NULL_data step. If requested, the system can display the titles to the user at this time in a temporary print file exactly as they will appear in the final report. An opportunity is also then given to change one or more of the title statements.

Creation of the DATA_NULL_data step begins with the naming of the SAS file for which the code is to be written. The user is prompted for an alphanumeric file name (Fig. 3), consisting of 3 letters and 3 digits. By convention, the letters represent the drug or compound, and the digits designate the protocol. For example, theophylline protocol 004 would be written to a file named TPH004. For the protection of current files, the system checks for the existence of a file by this name and will not write over the original without consent of the user. Because the system is designed to be forgiving, an-other opportunity is given to select a nonconflicting file name.

Next, the system prompts the user for the name of the data set to be written to the print file by the DATA_NULL_data step (Fig. 4). This can be any permanent SAS data set on or available to your account. However, the best results are obtained if this data set is one intended for use by Gennull, that is, containing delimited

labels and having associated permanent formats. The system tests for the availability of the chosen data set and permits 2 unsuccessful attempts before returning control to the main menu.

By using PROC CONTENTS, attributes of the variables are extracted from the selected data set. These include variable names, types, lengths, formats, and labels. Gennull incorporates the format length, the length of the variable name or the longest label segment, and the variable length into a single value of minimum width required for display to the print file. This block of minimum width becomes the cornerstone of all subsequent internal calculations. The results of this process are communicated to the user through an FSP screen (Fig. 5) where each variable and minimum block width are displayed. The user selects the desired variables for output by simply ordering the selections with Arabic numerals in the column headed 'select/order'. For example, a 1 in this column adjacent to the variable 'patient' indicates that this variable is selected for output and should appear as the first or left-most entry in the table. Similarly, a 2 next to the variable 'drug' would mark this variable for display as the second column entry in the table and so on. To exclude a variable from the final table, simply leave the 'select/order' column blank. The user should observe the minimum block width requirements displayed on this selection screen while choosing variables so that the total table width will not exceed space limitations of output devices. For example, if the total minimum block width amounts to 90 columns from the selection of 6 variables, and the user selects a 3 column separation buffer between display columns; then the output mechanism must be capable of displaying at least 105 [90+(5*3)] columns.

Once the last screen of information is entered, Gennull incorporates all previous user responses and internal system manipulations into a block of useful SAS code (Appendix B). In this code the title statements have been appended, the tedious task of column counting to position variables has been automatically performed and the results converted into the appropriate SAS PUT statements, the column head code will show that based on block width, variable labels have been correctly split and centered over the appropriate columns of data and the length of the horizontal header lines has been calculated and written to the appropriate PUT statements. This code, written to the user's account, is a SAS DATA_NULL_data step ready for further programming enhancements or immediate stand-alone execution.

CONCLUSION

Through Gennull and similar systems, the department of Statistics and Research Data

Services at A. H. Robins has increased productivity. Our experience with customized menu-driven applications has demonstrated that both novice and intermediate SAS users can be more efficient. By making greater demands of the machinery and software to perform repetitive tasks, such as column counting and the creation of the basic DATA _NULL_ data step, programming-related errors are reduced, and efficiency is improved. These systems increase the efficiency of each user and create a savings in man-hours that free programmers to enhance existing applications and develop new ones. Gennull is but one example of the positive impact that application building and implementation can provide - a savings in both human and computer resources.

^aThe special name _NULL_ in the data statement indicates that the data step will be performed as usual except that no observations are written to a data set. In this case, the data will be routed instead to a print file.

^bUsing the LIBRARY = option with the PROC FORMAT statement writes a permanent file to

minidisk A containing user-written formats for repetitive use.

We would like to acknowledge the editing services of Howard Smith.

REFERENCES

1. SAS User's Guide: Basics, 1985 Edition. SAS Institute, Inc., Cary, N.C.
2. SAS/AF User's Guide, 1985 Edition. SAS Institute, Inc., Cary, N.C.
3. SAS/FSP User's Guide, 1985 Edition. SAS Institute, Inc., Cary, N.C.

SAS, SAS/AF and SAS/FSP are trademarks of SAS Institute Inc., Cary, N.C.

For further information contact:

Wayne Cannon or Chris Bullard
A. H. Robins Co.
1211 Sherwood Ave.
Richmond, VA 23220

Figure 1

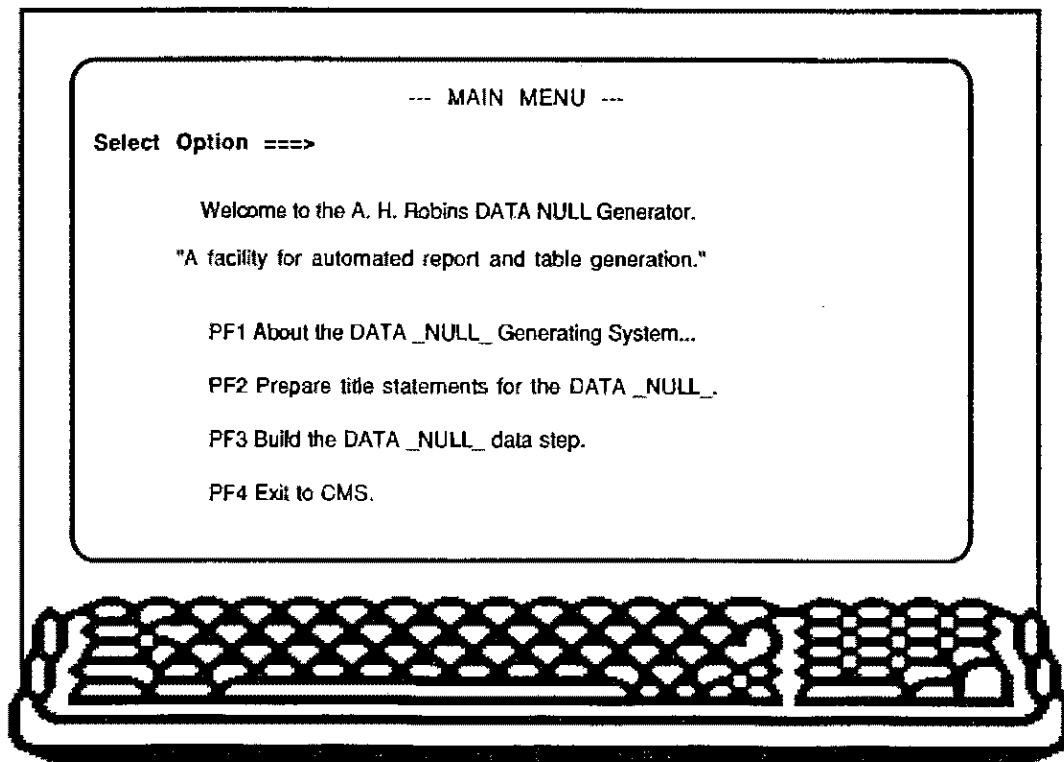


Figure 2

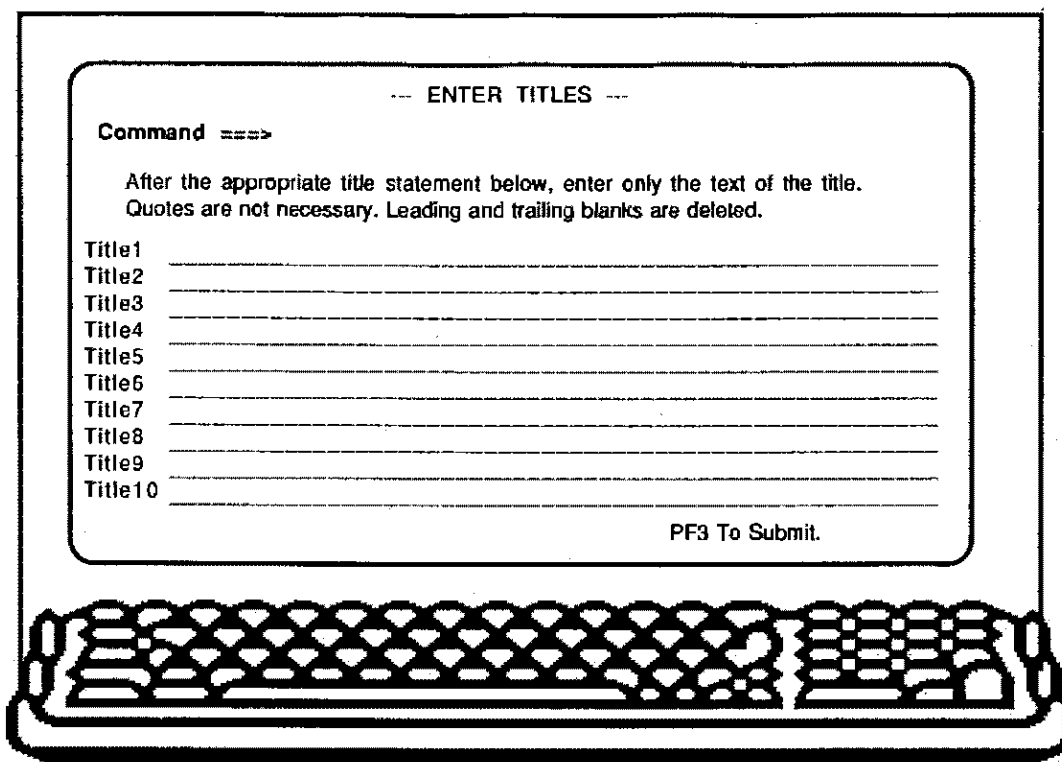


Figure 3

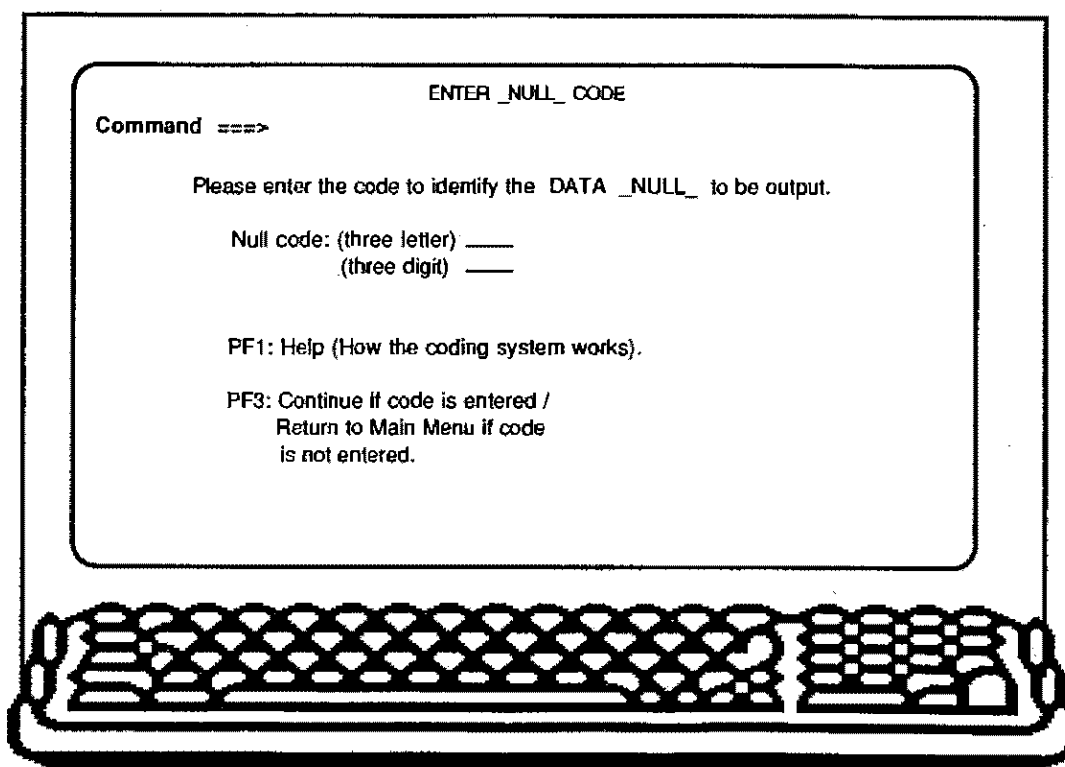


Figure 4

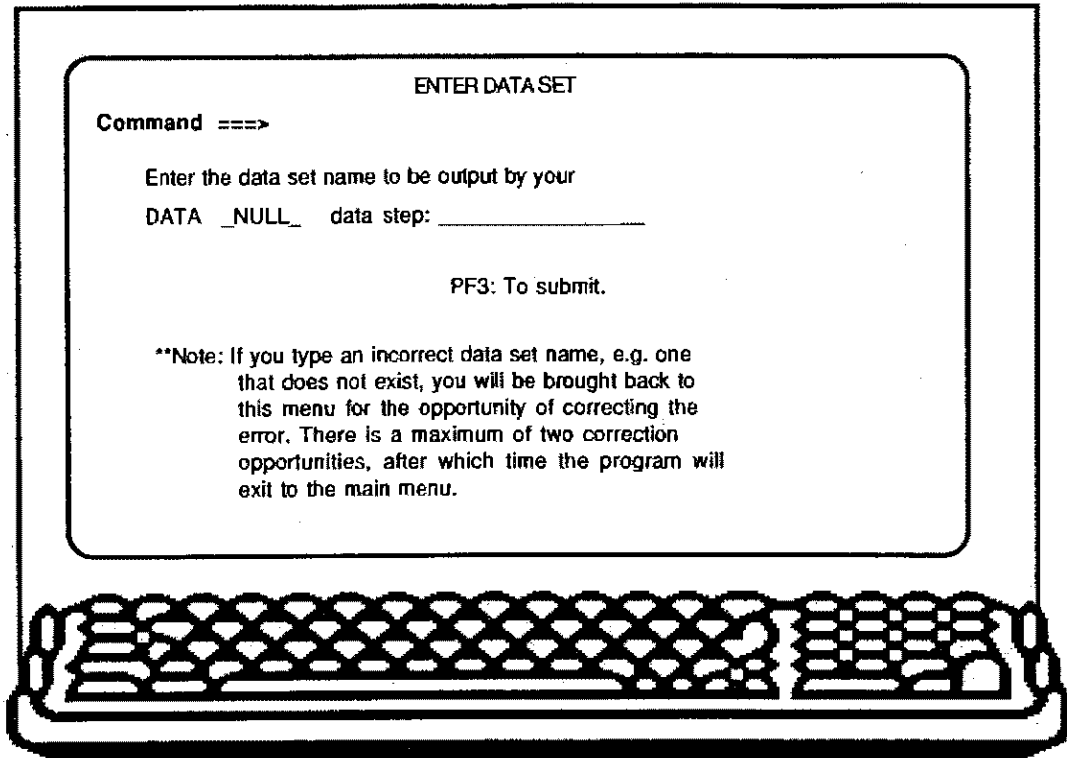
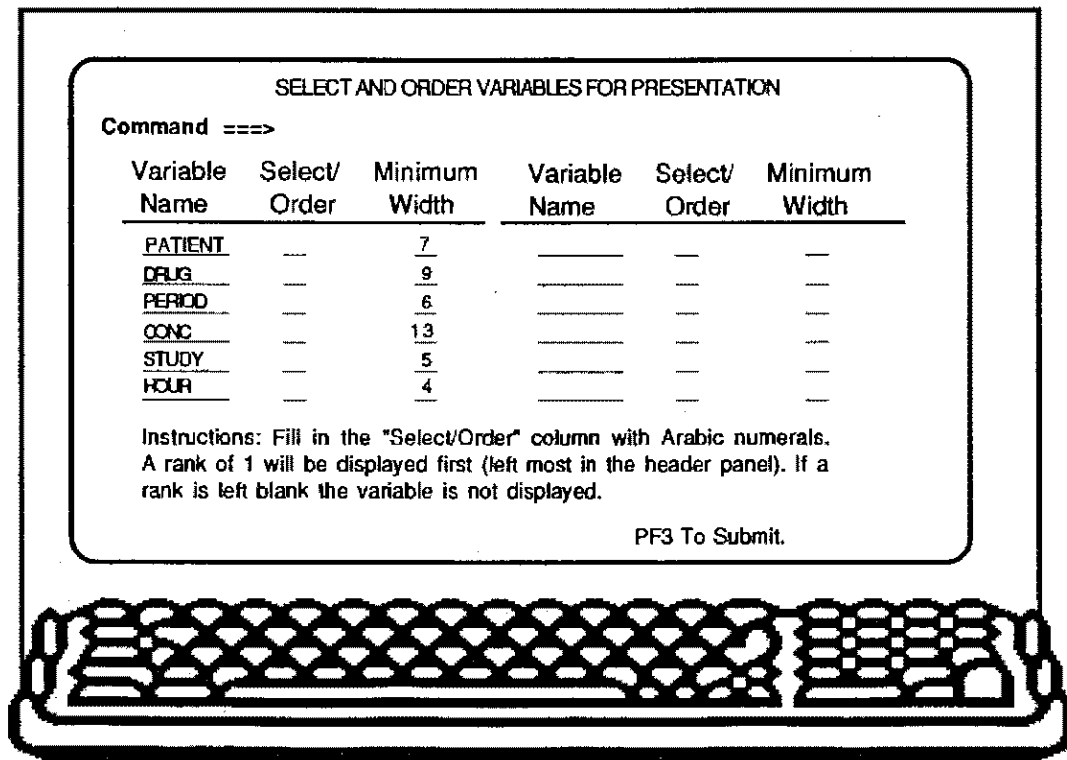


Figure 5



APPENDIX A

- * This EXEC initiates the SAS/AF System called Generate.null.,
- * a facility for automated report generation.
- *
- * To use: type 'gennull' on the command line and press the enter.
- * key.

&TRACE

FILEDEF GENERATE DISK GENERATE NULL A

&STACK PROC DISPLAY CATALOG=GENERATE.NULL.SETUP.PROGRAM;

&STACK RUN

EXEC SAS (MACRO NODMS LD NAME APPS NOMPRINT SYMBOLGEN MACROGEN

*DISCARD WORK FILES

ERASE DISPTIT LISTING A

ERASE TITLES SAS A

&EXIT

APPENDIX B

* The following data step outputs the data set TPH001.ASSAY to a print file.

*

* Created By: " NAME " Using the GENNULL system on " DATE ";

OPTIONS NODATE;

DATA _NULL_;

FILE PRINT HEADER = HDR LINESLEFT = LL;

SET THH001.ASSAY END = EOF;

IF LL < 10 THEN DO;

PUT @30 73 * ' ';

PUT _PAGE_;

END;

PUT @32 PATIENT @45 DRUG DRUG. @60 PERIOD @72 HOUR @90 CONC;

IF EOF THEN DO;

PUT @30 73 * ' ' / @30 73 * ' ';

END;

RETURN;

HDR:

PUT / @30 73 * ' ';

PUT @30 'SUBJECT' @42 'DRUG CODE' @56 'DOSE PERIOD' @74 'HOUR'

@ 85 'DRUG CONCENTRATION';

PUT @30 73 * ' ';

RETURN;

TITLE1 'Sample Table';

TITLE2 'Concentrations of Theophylline in Plasma';

TITLE3 'AHR-#### Study #####';