SO YOU THINK YOU KNOW THE SAS® SYSTEM,
OR THE INTERACTIVE SAS® QUIZ

Alan B. Howard, University of Vermont

ABSTRACT

The SAS® System for Personal Computers implements several features not yet found in mainframe or minicomputer versions: the WINDOW and DISPLAY statements and the Stored Program Facility. An application of the SAS System for Personal Computers was designed and written to take advantage of these unique features. This interactive program uses the WINDOW and DISPLAY statements to quiz a user on his/her knowledge of SAS syntax. The compiled program was saved using the Stored Program Facility. By storing and subsequently loading compiled code, about two minutes of waiting time is avoided to start the application.

INTRODUCTION

The inspiration for this application came from several sources. First, I wanted to gain experience with the WINDOW and DISPLAY statements. Since the University of Vermont does not license SAS/AF® or SAS/FSP® on any of its mainframes, minis, or PC's, these two statements seemed to offer the ability to create screens for data entry and/or computer based training. Second, after teaching an "Introduction to SAS" short course, I wanted something to use as a "final exam" not the typical paper and pencil type exam, but something a little more interesting and even fun to do. Third, I remembered seeing an article in the proceedings of a previous SUGI conference which evaluated a prospective employee's SAS expertise with a written test. Combining these three things resulted in an interactive quiz, using windows to display questions and provide feedback to a user, testing him/her on SAS syntax in any of three categories (i.e., levels of SAS expertise).

QUIZ PROGRAM STRUCTURE

Multiple choice questions in each category were devised and entered into a file using eight lines per question. The first three lines comprised the actual question, each of the next four lines contained a possible answer, and the eighth line had the correct answer. SAS data sets containing the questions, the choices, and the correct answer were then constructed, reading eight lines per observation from the input data file.

The quiz program itself consists of a series of windows which prompt for input, display the question, and give the user feedback after responding to each question. The welcoming window prompts for the user's name, the level of expertise at which he/she wishes to be tested, and the desired number of multiple-choice questions. There are three levels of expertise (beginner, intermediate, or advanced) and a maximum of ten questions at each level. It is a simple matter to add more questions and modify the code to allow many more questions per level. Should the user enter anything other than one of the appropriate choices of expertise level, or anything other than an integer between 1 and 10 for number of desired questions, a window appears to inform him/her that the input was not accepted. They will then be returned to the offending field for another try.

Once these choices have been made and accepted by the program, the first question is displayed and the user is prompted for a response. The questions are accessed from the appropriate SAS data set in a random fashion, using the SET statement with the POINT~ option. This method of sampling from a SAS data set without replacement is described in the SAS Applications Guide, 1987 Edition.

If the user's response is correct, a window is displayed to tell them so. If the response is not correct, a different window is shown which gives the correct answer. Each of these two windows displays the number of questions answered correctly so far out of the total number asked, and the number remaining. A continuation window is then shown before bringing up the next question. The user can quit prematurely at this continuation window as well as at other points in the program. (The prompts for level of expertise and response to the current question both accept Q to quit.)

After answering all questions, a tally window is displayed which shows the user's score (in percent correct) and asks if another try is desired. If the user replies yes to this prompt, the welcoming window is redisplayed and the user can enter a new level of expertise and number of questions. If the response is no, the program resets all variables for a new user.

679
This program could be modified easily to quiz a user on any topic of interest, not just SAS statement syntax. Students taking a quiz like this in a classroom/lab setting would get the questions in a different order than any of his/her neighbors, due to random sampling from the questions in the SAS data set. Another simple modification could output final results for each user to a file for later retrieval by the instructor.

REFERENCES
