TEACHING SAS TO THE ENTRY-LEVEL PROGRAMMER

Debbie D. Wilson, Kentucky Department for Human Resources

ABSTRACT

This paper will discuss the SAS training techniques used in the training program at Kentucky's Department for Human Resources for all entry-level programmers. These techniques include: A TSO CLIST as a tutorial, an in-house written primer, and 3 sample SAS jobs as examples. Since SAS is such a powerful and useful tool for many applications, the Department for Human Resources teaches SAS to programmers, regardless of the responsibilities they are to assume. These programmers may or may not use SAS after training, but they are made aware of its applicability, simplicity, and quick results.

The CLIST tutorial forms the core of our SAS training with the primer and sample jobs serving as resource aids. Trainees are able, with a minimum of personal assistance, upon completion of the tutorial to write SAS programs to fill most simple requests we receive. Our primary use of SAS is for one-time reports on a wide range of file layouts.

INTRODUCTION

At this installation, SAS was formerly taught by an instructor in a class that lasted approximately 3 to 4 hours. This method of teaching presented several problems:

1. The instructor didn't always have the time to devote to teaching the class since he was also involved in other activities that had a higher priority.

2. The class was only feasible when there were several students, which meant some trainees had to wait until other trainees were also ready for the class.

3. It was hard to coordinate a time and a meeting place that was agreeable to all those involved.

In view of all the problems that having a formal class presented, we found the need to incorporate an at-your-own-pace type of instruction that would fulfill the general needs of this installation.

DESIGN APPROACH

The suggestion was made that it would be helpful if the trainees (entry-level programmers) could have access to some kind of independent study program via TSO, instead of a formal class. I had already written a small 16 page primer (this was before the SAS INTRODUCTORY GUIDE) that explained how to use SAS for the general needs at this installation. From this primer I wrote a TSO CLIST to serve as a tutorial to take the place of the formal class. The CLIST follows along the same lines as the in-house primer. The primer covers the following topics and statements in general terms:

1. Two sample jobs, including JCL statements necessary to execute as a batch job and where to get a copy to execute.

2. SAS statements:
   a) syntax
   b) type of statements
      i) the DATA statement
      ii) the INFILE statement
      iii) the INPUT statement
      iv) the CARDS statement
      v) the SET statement
   c) program statements (those that follow the DATA statement)
      i) ASSIGNMENT statements
      ii) SUM statements
      iii) IF statement including IF-THEN and IF-THEN-ELSE
      iv) GOTO statement
      v) STATEMENT LABELS
      vi) DROP statement
      vii) KEEP statement
      viii) DELETE statement
      ix) OUTPUT statement
      x) RETAIN statement
xi) FORMAT statement
xii) TITLE statement
xiii) COMMENT statement
d) procedures (only the ones that we use most frequently)
i) PROC PRINT
ii) PROC SORT
iii) PROC FREQ
iv) PROC MEANS
e) procedure information statements
i) BY statement

3. An example of how SAS works using one of the sample jobs

4. A list of all SAS command formats (much like the reference card included in the SAS79 USER'S GUIDE)

5. A list of SAS options, special variables and symbols
a) missing values
b) special variables
c) logical operators
d) arithmetic operators
e) pointer directions
f) formats
g) miscellaneous items

6. A list of all available SAS PROCEDURES and a one sentence description of each

(At several places in the primer the reader is instructed to refer to the SAS User's Guide for more detailed descriptions. They are also advised where to get a User's Guide if they don't have one or have access to one.)

We have found that use of this primer is much better than handing a new programmer the SAS User's Guide and having her understand what she has read, or know where to look up something.

The TSO CLIST was written to go along with our in-house primer and the SAS INTRODUCTORY GUIDE. The CLIST consists of approximately 900 statements, most of which are WRITE statements to write the text to the screen. It is divided into sections so that the trainee can, if necessary, break out and restart the lesson at the point where it was interrupted instead of having to repeat parts already covered. The sections are:

1. INTRO (General information)
2. SYNTAX (Rules for variable naming, etc.)
3. DATASETS (How to read in data)
4. PROGRAM (Statements for building a dataset after reading in the data)
5. PROCS (Procedures to produce output)
6. JCL (Job statements to run SAS in the batch environment)
7. CLIST (How to run a SAS job on-line)
8. NOTES (Miscellaneous notes)

The CLIST uses the material presented and builds onto a sample job using the topics covered; by the end of the CLIST tutorial the trainee has been shown the statements needed to build a simple job.

At several places in the CLIST the trainee is given a simple multiple choice test with such questions as:

1. Which of the following is a valid SAS dataset name?
2. Which of the following special characters terminates a SAS statement?

If the wrong response is entered, the correct answer is displayed along with an explanation of why the answer was wrong. Just for fun, a score is kept for the quizzes and displayed at the end of each section.

Since the CLIST tutorial assumes that the trainee has very little (if any) prior TSO experience, if she breaks out of the CLIST, she is reminded of how to get back into the section she was in and the proper name of the section.

It takes the average student about 1 hour to complete the CLIST tutorial.

To go along with both the CLIST and the primer, the trainees are given the SAS
statements and output from 3 sample jobs (2 jobs written in-house that use data the trainees use in other phases of their training so that they are familiar with it, and the DATASHOW example from SAS.SAMPLE). These jobs help to show what kind of statements create what kinds of output and how you can manipulate data.

**Utilization**

After the Kentucky Department for Human Resources had implemented this approach of teaching SAS to their beginning programmers with very good success, the computer center for state government learned of it and wanted to make it accessible to all D.P. departments of Kentucky state government. (Although all departments work for the state, they are actually separate in physical location and do not have access to other department's files, resources, personnel, etc)

In view of the various hard copy documentation that a person should have in her possession (CLIST documentation, the primer, the 3 sample jobs), I found it necessary (in order to make the study package as independent of human intervention as possible) to write another CLIST explaining how to access the tutorial and also to do all the hard-copy documentation and send it to the person no matter where she was in the state.

**Benefits**

The Commonwealth of Kentucky's Department for Human Resources has been using this method of teaching SAS to their new programmers for about 1 year, and has received excellent results from the 12 new programmers who have gone through it. In view of the volume of new programmers the Department for Human Resources hires in a year, it has been very beneficial to use this kind of approach for the teaching of SAS to new programmers as a regular part of their training.

I estimate that by using this teaching package, the savings in the past year have been approximately $200 in instructor time, not including what has been saved by the instructor's not being available to perform his regular duties.

Although, as with most training, there must be someone around to help with the problems encountered, the trainees are able, with minimal assistance, to write simple SAS programs upon comple-tion of this package. This will enable them to fill most simple requests they might receive when they are put into a real-world programming environment.

Although the new programmers do not have an extensive knowledge of SAS, they have been made aware of how SAS can be used, where to get help, and what facilities are available to them.

**Conclusion**

To sum up the various items that I used to make up our SAS teaching package, here is a list of all components used:

1. SASDOC (a CLIST that produces all needed hardcopy documentation listed here a-d)
   a. PRIMER (our in-house written primer)
   b. TUTOR (documentation of the tutorial clist)
   c. the 3 sample SAS jobs
   d. SASCLIST (a description of our clist to run SAS on-line)

2. TRACHSAS (the tutorial CLIST)

With a package such as I have described here, most any installation can design a package that will meet their own individual requirements. It does not require a great deal of time and effort, and yet can be modified as teaching needs change.

******************************************************************************

The author would like to thank Mr. Earl Hohman of this department for his assistance in writing both the primer and this paper, and also Ms. Carolyn Brown for her assistance in typing and proofreading.

******************************************************************************