

Does That Programmer Really Know SAS 6.07? – A SAS Skills Test.

When it comes time to convert your SAS applications to version 6, or optimize them, or even develop new version 6 applications, you're going to need someone who is familiar with the in and outs of 6. And to help you gauge someone's familiarity (or even your own), I've put together a list of questions. These questions are based on Standish Sibley's "Reasonably Comprehensive Test of SAS Programming Skills" that was presented at SUGI 16, and some of his questions are repeated here. Mr. Sibley has given permission to duplicate and use his questions, upon the condition that he is credited as author.

However, this test is not intended to be a comprehensive one, but is designed to test a programmer's Base SAS skills, and in particular, items specific to version 6. The test is difficult; I certainly wouldn't expect a programmer to get all the questions correct, but their answers will give you a good feel for their version 6 expertise. This version deals with SAS on MVS, but you should be able to modify the MVS-specific questions for your particular operating system.

SAS test.

Unless otherwise noted, all questions relate to Version 6.07 of SAS on MVS.

Some questions supplied by Standish W. Sibley.

1. Three SAS datasets, DATA.JAN, DATA.FEB, and DATA.MAR, each contains a variable named ACCT, and only one observation per ACCT. Write SAS code to merge these three datasets by ACCT, into a single SAS dataset (named DATA.MERGE) that contains only those observations for account numbers (ACCT) found in all three datasets.
2. Show three different ways to submit a command to TSO from within a SAS program. How may you then check the return code from TSO?
3. Name different ways to change (or set) the value of a SAS system option.
4. Can any SAS system option be set or changed using any of the methods that you listed above? If not, why not?

5. What is the significance of the dataset associated with the DDname SASEXEC below:

```
//STEP1 EXEC SAS
//SASEXEC DD DISP=SHR,DSN=USER.SASEXEC.FILE
```
6. Approximately what date does the SAS date value of 730 represent?
7. What SAS system options might you set on or off when you are debugging macro code?
8. What method of commenting code works in base SAS as well as in macro code? What comment method is unique to the macro facility?
9. What is the maximum amount of disk space allocated by the card below? About how many cylinders of 3380 disk is this? What will SAS use this space for?

```
//WORK DD SPACE=(6144,(1000,400))
```
10. The %INCLUDE command allows the use of options SOURCE2 and JCLEXCL. Explain what each option does.
11. What is the difference between coding DISP=OLD and DISP=SHR on a DD statement that references a SAS data library? Do these options work differently in Version 5 SAS and Version 6 SAS, and if so, how?
12. What are the different member types of Version 6 SAS files?
13. Code a LIBNAME statement that associates the dataset "USER.SASDB" with the Version 5 compatibility engine.
14. Code a SAS step that converts all the files in the Version 5 SAS data library "USER.V5SAS" to a Version 6 library called "USER.V6SAS".
15. What are some ways to reroute the SAS log in a batch SAS job?
16. Code a PROC SQL step equivalent to the following:

```
PROC SORT DATA=SAMPLES OUT=SORTSAMP;
  BY KEY;
DATA SUBSET;
  SET SORTSAMP;
  KEEP KEY;
  BY KEY;
  IF FIRST.KEY;
  IF KEY = 'A' OR KEY = 'C' OR KEY = 'R' OR KEY = 'Q';
RUN;
```

17. What effect does the new SAS system option FILEPROMPT have?
18. What SAS system options control whether performance statistics are written to the log?
19. What does the SAS statement LEAVE do?
20. Name one interesting thing new in Version 6.07 SAS.

SAS test answers.

1.

```
PROC SORT DATA=DATA.JAN;
    BY ACCT;
PROC SORT DATA=DATA.FEB;
    BY ACCT;
PROC SORT DATA=DATA.MAR;
    BY ACCT;
DATA DATA.MERGE;
    MERGE    DATA.JAN (IN=IN_JAN)
            DATA.FEB (IN=IN_FEB)
            DATA.MAR (IN=IN_MAR);
    BY ACCT;
    IF IN_JAN AND IN_FEB AND IN_MAR;
```

or

```
PROC SQL;
    CREATE TABLE MERGE AS
    SELECT JAN.ACCT FROM JAN, FEB, MAR
    WHERE (JAN.ACCT = FEB.ACCT) AND
          (JAN.ACCT = MAR.ACCT) AND
          (FEB.ACCT = MAR.ACCT);
```
2.

```
TSO command;
%TSO command;
var = TSO('command');
```

For "TSO command" and "%TSO command", inspect the value of &SYSRC. For "var = TSO('command')", inspect the value of "var".

3. Options can be specified
 - at invocation time, e.g. `// EXEC SAS,OPTIONS = 'LINESIZE = 80'`
 - on an options statement, e.g. `OPTIONS NONOTES;`
 - in the `OPTIONS` window
 - in a configuration file
4. No. Some options must be known to SAS at invocation time; for example, DMS (Display Management System) cannot be turned on and off once the SAS session has begun.
5. It identifies the dataset containing AUTOEXEC code. This dataset contains statements that are executed automatically at invocation.
6. 31 DEC 1961
7. MPRINT MLOGIC SYMBOLGEN. See "SAS Guide to Macro Processing".
8. `"/*` and `*/` will work in both. `"%*" and ";"` is for macro code only.
9. 7000 blocks of 6144 bytes, approx. 67 cylinders. SAS will use this space for temporary ("WORK.") datasets, utility files, user profile, etc.
10. SOURCE2 prints included source on SASLOG. JCLEXEC causes SAS to ignore any embedded JCL statements in the included code.
11. DISP=OLD allows only one job at a time to use the library; that job can read, modify, or delete a SAS file in that library. DISP=SHR allows multiple jobs to read the library; however, no job can modify or delete a SAS file in that library. There is no difference between Version 5 and Version 6.
12. DATA, PROGRAM, INDEX, VIEW, CATALOG, and ACCESS (in SAS/ACCESS)
13. LIBNAME V5 'USER.SASDB';
14. LIBNAME SOURCE 'version5-data-library';
LIBNAME DEST 'version6-data-library';
PROC V5TOV6 IN=SOURCE OUT=DEST;

15. You can reroute your SAS log in a batch job by
- using the LOG = system option to route the SAS log to an external data set
 - using the FILENAME statement to dynamically allocate an external file and then executing a PROC PRINTTO statement with the LOG = option
 - supplying a new SASLOG DD statement with the destination you want
 - specifying the ALTLOG = system option at invocation to send a copy of your SAS log to an external file
 - using the LOG operand on the SAS CLIST for an interactive line mode or noninteractive session
16. PROC SQL;
CREATE TABLE SUBSET AS
SELECT DISTINCT KEY FROM SAMPLES
WHERE KEY IN ('A','C','R','Q');
17. FILEPROMPT controls whether you are prompted if the operations system data set that is referenced in a FILENAME or LIBNAME statement does not exist.
18. FULLSTATS, MEMRPT, STATS, STIMER
19. LEAVE quits the current iteration of the DO loop