# SAS Certification Prep Guide: Base Programming for SAS 9, Third Edition

by SAS Institute



---

# Chapter 2: Referencing Files and Setting Options

## Overview

### Introduction

When you begin a SAS session, it's often convenient to set up your environment first. For example, you may want to

- define libraries that contain the SAS data sets that you intend to use
- specify whether your procedure output is created as HTML (Hyper Text Markup Language) output, LISTING output, or another type of output.
- set features of your LISTING output, if you are creating any, such as whether the date and time appear
- specify how two-digit year values should be interpreted.

This chapter shows you how to define libraries, reference SAS files, and specify options for your SAS session. You also learn how to specify the form(s) of output to produce.

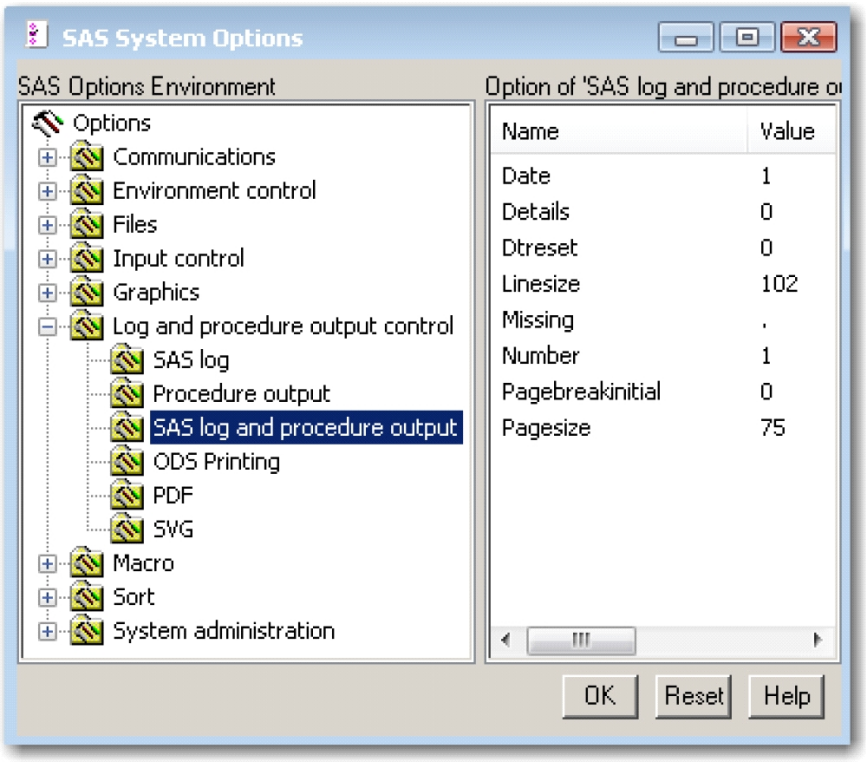



**Figure 2.1:** SAS System Options Window

### Objectives

In this chapter, you learn to

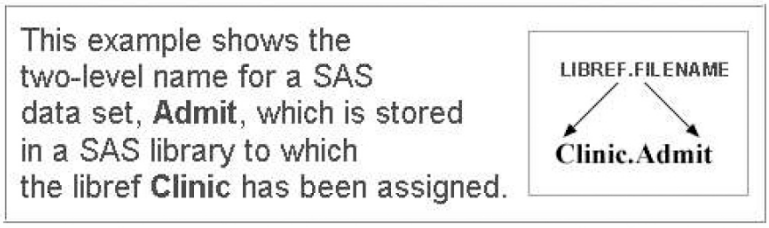
- define new libraries by using programming statements
- reference SAS files to be used during your SAS session
- set results options to determine the type or types of output produced (HTML, LISTING output, or other) in desktop operating environments
- set system options to determine how date values are read and to control the appearance of any LISTING output that is created during your SAS session.

## Defining Libraries

Often the first step in setting up your SAS session is to define the libraries. You can also use programming statements to assign library names.

Remember that to reference a permanent SAS file, you

1. assign a name *(libref)* to the SAS library in which the file is stored
2. use the libref as the first part of the file's two-level name *(libref.filename)* to reference the file within the library.



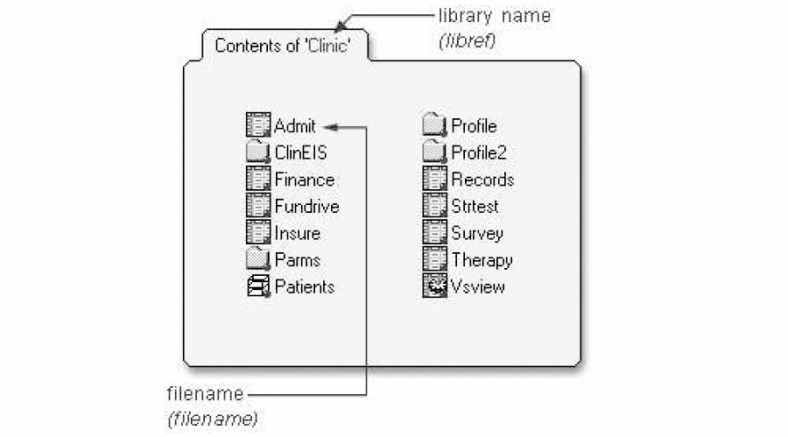




**Figure 2.2:** Defining Libraries

## Assigning Librefs

To define libraries, you can use a LIBNAME statement. You can store the LIBNAME statement with any SAS program so that the SAS data library is assigned each time the program is submitted.

---

General form, basic LIBNAME statement:

**LIBNAME** *libref 'SAS-data-library';*

where

- *libref* is 1 to 8 characters long, begins with a letter or underscore, and contains only letters, numbers, or underscores.
- *SAS-data-library* is the name of a SAS data library in which SAS data files are stored. The specification of the physical name of the library differs by operating environment.

---

The LIBNAME statement below assigns the libref Clinic to the SAS data library **`D:\Users\Qtr\Reports`** in the Windows environment.

```
libname clinic 'd:\users\qtr\reports';
```

Many of the examples in this book use the libref **`sasuser`**. The following LIBNAME statement assigns the libref sasuser to

the c:\Users\name\sasuser folder in a Windows operating environment:

```
libname sasuser 'c:\Users\name\sasuser';
```

The table below gives some examples of physical names for SAS data libraries in various operating environments.

**Table 2.1: Physical Names for SAS Data Libraries**

| Environment | Sample Physical Name |
|---|---|
| Windows | c:\fitness\data |
| UNIX | /users/april/fitness/sasdata |
| z/OS (OS/390) | april.fitness.sasdata |

**Additional Note** The code examples in this chapter are shown in the Windows operating environment. If you are running SAS within another operating environment, then the platform-specific names and locations will look different. Otherwise, SAS programming code will be the same across operating environments.

You can use multiple LIBNAME statements to assign as many librefs as needed.

### Verifying Librefs

After assigning a libref, it is a good idea to check the Log window to verify that the libref has been assigned successfully.


**Figure 2.3:** Log Output for Clinic libref

### How Long Librefs Remain in Effect

The LIBNAME statement is global, which means that the librefs remain in effect until you modify them, cancel them, or end your SAS session.

Therefore, the LIBNAME statement assigns the libref for the current SAS session only. Each time you begin a SAS session, you must assign a libref to each permanent SAS data library that contains files that you want to access in that session. (Remember that Work is the default libref for a temporary SAS data library.)

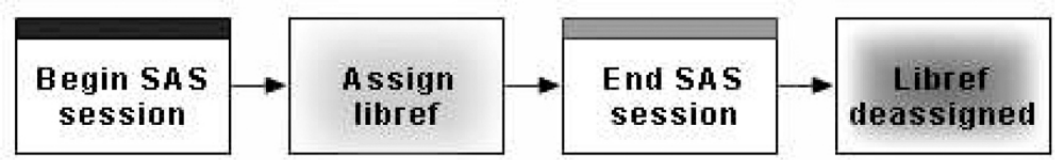



**Figure 2.4:** How Long Librefs Remain in Effect

When you end your SAS session or delete a libref, SAS no longer has access to the files in the library. However, the contents of the library still exist on your operating system.

**Additional Note** Remember that you can also assign a library from the SAS Explorer using the New Library window. Libraries that are created with the New Library window can be automatically assigned at startup by selecting **Enable at Startup**.

## Specifying Two-Level Names

After you assign a libref, you specify it as the first element in the two-level name for a SAS file.

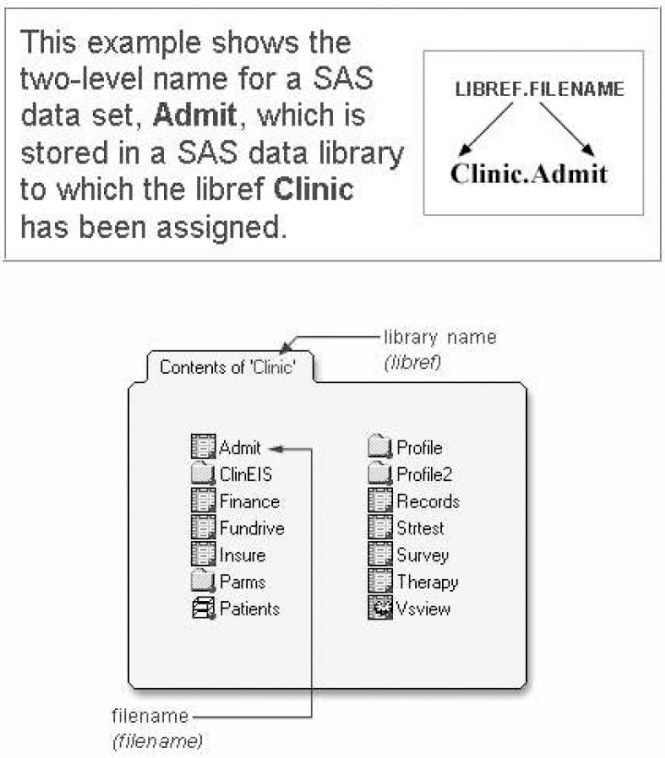



**Figure 2.5:** Specifying Two-Level Names

For example, in order for the PRINT procedure to read Clinic.Admit, you specify the two-level name of the file as follows:

```
proc print data=clinic.admit;
run;
```

## Referencing Files in Other Formats

You can use the LIBNAME statement to reference not only SAS files but also files that were created with other software products, such as database management systems.

SAS can read or write these files by using the appropriate engine for that file type. For some file types, you need to tell SAS which engine to use. For others, SAS automatically chooses the appropriate engine.

A SAS engine is a set of internal instructions that SAS uses for writing to and reading from files in a SAS library.

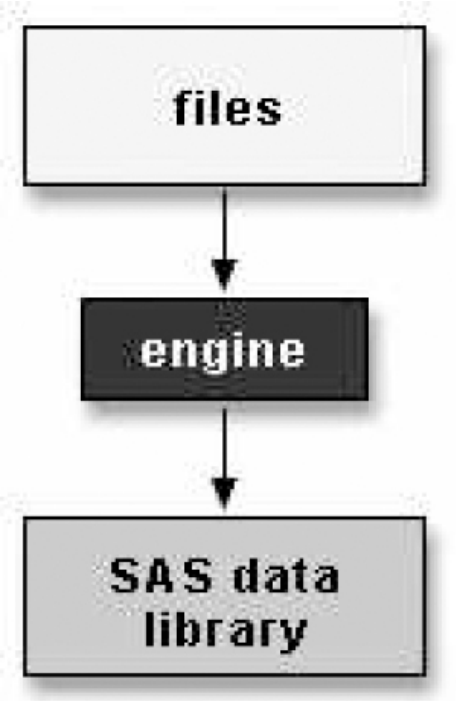



**Figure 2.6:** Referencing Files in Other Formats

### Specifying Engines

To indicate which engine to use, you specify the engine name in the LIBNAME statement, as shown below.

---

General form, LIBNAME statement for files in other formats:

**LIBNAME** *libref engine 'SAS-data-library';*

where

- *libref* is 1 to 8 characters long, begins with a letter or underscore, and contains only letters, numbers, or underscores.
- *engine* is the name of a library engine that is supported in your operating environment.
- *SAS-data-library* is the name of a SAS library in which SAS data files are stored. The specification of the physical name of the library differs by operating environment.

---

### Interface Library Engines

Interface library engines support read-only access to BMDP, OSIRIS, and SPSS files. With these engines, the physical filename that is associated with a libref is an actual filename, not a SAS library. This is an exception to the rules for librefs.

**Table 2.2: Engines and Their Descriptions**

| Engine | Description |
|---|---|
| BMDP | allows read-only access to BMDP files |
| OSIRIS | allows read-only access to OSIRIS files |
| SPSS | allows read-only access to SPSS files |

For example, the LIBNAME statement below specifies the libref Rptdata and the engine SPSS for the file **`G:\Myspss.dat`** in the Windows operating environment.

```
libname rptdata spss 'g:\myspss.spss';
```

For more information about interface library engines, see the SAS documentation for your operating environment.

### SAS/ACCESS Engines

If your site licenses SAS/ACCESS software, you can use the LIBNAME statement to access data that is stored in a DBMS file. The types of data you can access depend on your operating environment and on which SAS/ACCESS products you have licensed.

**Table 2.3: Relational Databases and Their Associated Files**

| Relational Databases | Nonrelational Files | PC Files |
|---|---|---|
| ORACLE | ADABAS | Excel (.xls) |
| SYBASE | IMS/DL-I | Lotus (.wkn) |
| Informix | CA-IDMS | dBase |
| DB2 for z/OS DB2 for UNIX and PC | SYSTEM 2000 | DIF |
| Oracle Rdb | Teradata | Access |
| ODBC | MySQL | SPSS |
| CA-OpenIngres | Netezza | Stata |
| | Ole DB | Paradox |

## Viewing SAS Libraries

### Viewing the Contents of a SAS Library

You've seen that you can assign librefs in order to access different types of data. "Using the Programming Workspace" on page 20 explained that after you have assigned a libref, you can view

- details about the library that the libref references
- the library's contents
- contents and properties of files in the library.

The libraries that are currently defined for your SAS session are listed under Libraries in the Explorer window. To view details about a library, double-click **Libraries** (or select **Libraries** ⇨ **Open** from the pop-up menu). Then select **View** ⇨ **Details**.

Information for each library (name, engine, type, host pathname, and date modified) is listed under Active Libraries.

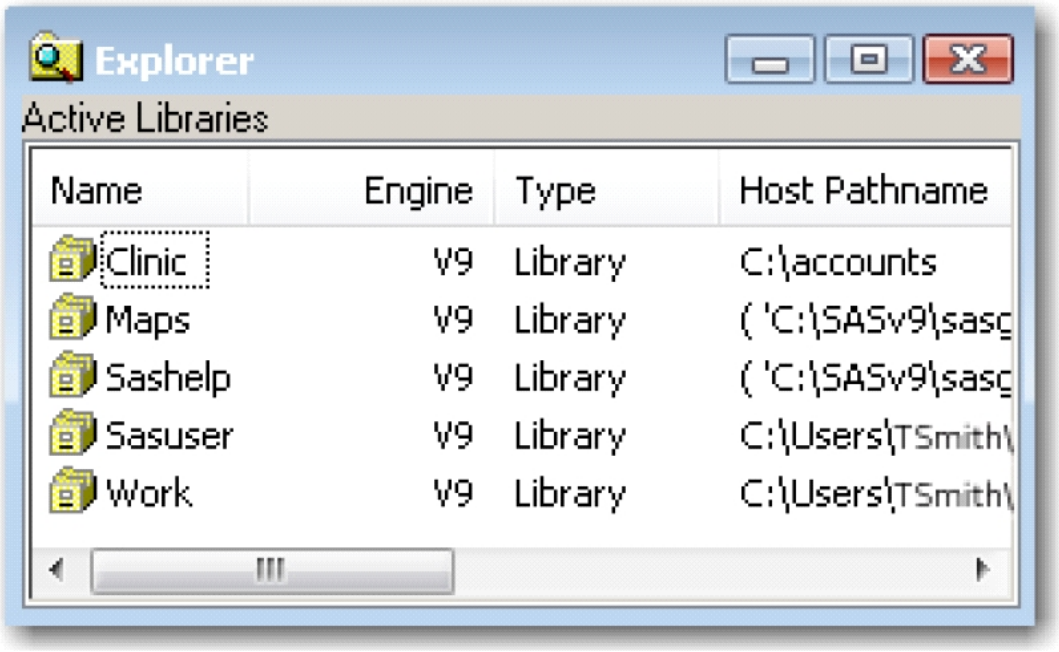



**Figure 2.7:** Viewing Active Libraries

To view the contents of a library, double-click the library name (or select the library name, and then select **Open** from the pop-up menu). A list of the files contained in the library is displayed. If you have the details feature turned on, then information about each file (name, size, type, description, and date modified) is also listed.

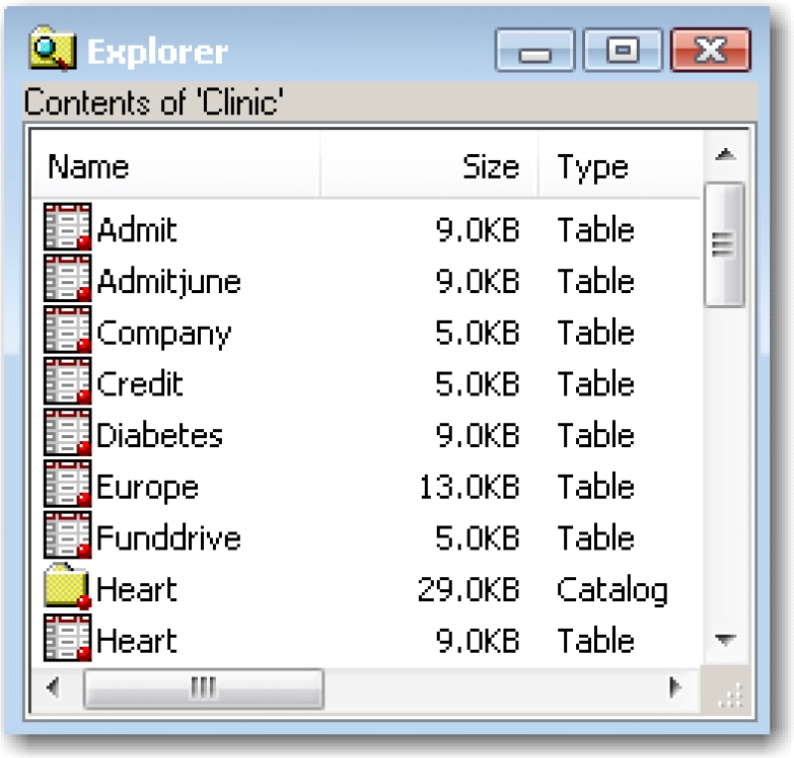
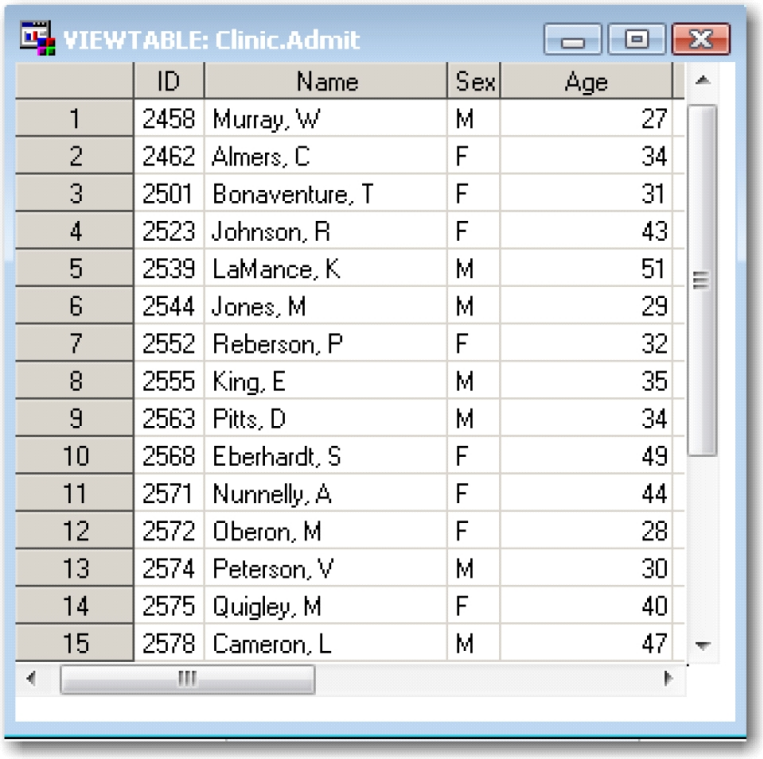



**Figure 2.8:** Viewing the Contents of a SAS Library

### Viewing a File's Contents

To display a file's contents in a windowing environment, you can double-click the filename (or select the filename, and then select **Open** from the pop-up menu). If you select a SAS data set (type Table or View), its contents are displayed in the VIEWTABLE window.




| | ID | Name | Sex | Age |
|---|---|---|---|---|
| 1 | 2458 | Murray, W | M | 27 |
| 2 | 2462 | Almers, C | F | 34 |
| 3 | 2501 | Bonaventure, T | F | 31 |
| 4 | 2523 | Johnson, R | F | 43 |
| 5 | 2539 | LaMance, K | M | 51 |
| 6 | 2544 | Jones, M | M | 29 |
| 7 | 2552 | Reberson, P | F | 32 |
| 8 | 2555 | King, E | M | 35 |
| 9 | 2563 | Pitts, D | M | 34 |
| 10 | 2568 | Eberhardt, S | F | 49 |
| 11 | 2571 | Nunnelly, A | F | 44 |
| 12 | 2572 | Oberon, M | F | 28 |
| 13 | 2574 | Peterson, V | M | 30 |
| 14 | 2575 | Quigley, M | F | 40 |
| 15 | 2578 | Cameron, L | M | 47 |

**Figure 2.9:** Viewing a File's Contents

To display a file's properties, you can select the filename, and then select **Properties** from the pop-up menu.

**Additional Note** If you are working in the z/OS operating environment, you can type **?** in the selection field next to a filename in the Explorer window to display a pop-up menu with a list of options for working with that file.

If you have installed SAS/FSP software, you can type **B** or **L** in the selection field next to a data set name to browse the data set observation by observation or to list the contents of the data set, respectively. For more information, see the documentation for SAS/FSP.

If SAS/FSP is not installed, you can view the contents of a SAS data set by using the PRINT procedure (PROC PRINT). You can learn how to use PROC PRINT in "Creating List Reports" on page 112.

### Using PROC CONTENTS to View the Contents of a SAS Library

You've learned how to use SAS windows to view the contents of a SAS library or of a SAS file. Alternatively, you can use the CONTENTS procedure to create SAS output that describes either of the following:

- the contents of a library
- the descriptor information for an individual SAS data set.

---

General form, basic PROC CONTENTS step:

**PROC CONTENTS DATA**=*SAS-file-specification* **NODS;**

**RUN;**

where

- *SAS-file-specification* specifies an entire library or a specific SAS data set within a library. *SAS-file-specification* can take one of the following forms:
  - *<libref.>SAS-data-set* names one SAS data set to process.

- *<libref.>_ALL_* requests a listing of all files in the library. (Use a period (.) to append _ALL_ to the libref.)

- **NODS** suppresses the printing of detailed information about each file when you specify _ALL_. (You can specify NODS *only* when you specify _ALL_.)

---

### Examples

To view the contents of the entire clinic library, you can submit the following PROC CONTENTS step:

```
proc contents data=clinic._all_ nods;
run;
```

The output from this step lists only the names, types, sizes, and modification dates for the SAS files in the Clinic library.

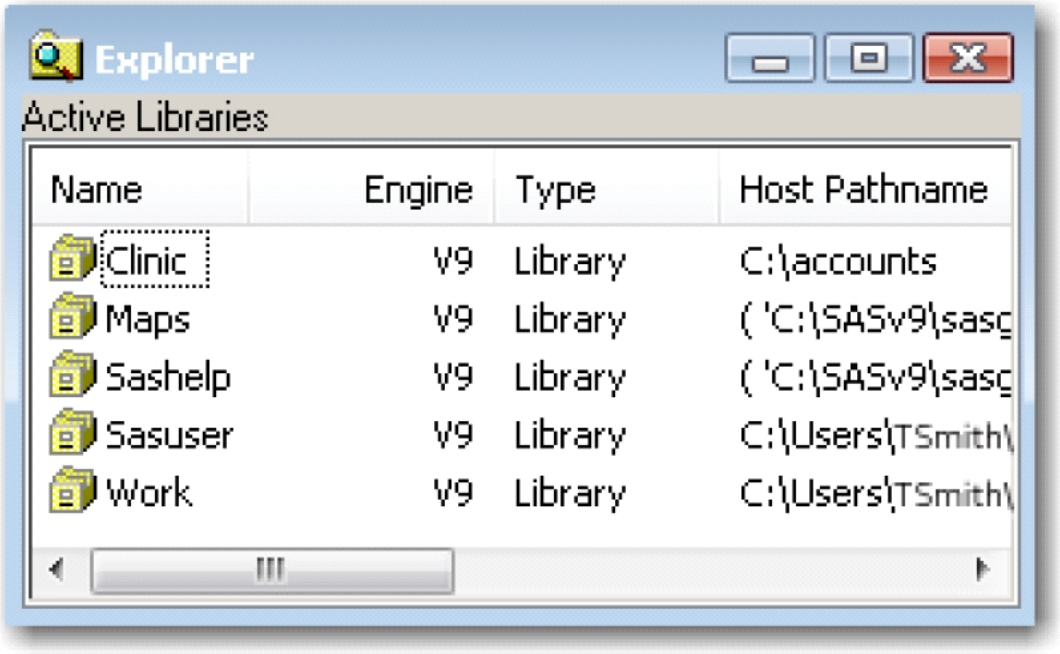



**Figure 2.10:** Using PROC CONTENTS to View the Contents of a Library

| # | Name | Member Type | File Size | Last Modified |
|---|---|---|---|---|
| 1 | ADMIT | DATA | 9216 | 05April 10:52:56 |
| 2 | ADMIT2 | DATA | 9216 | 05Apri1 12:35:45 |
| 3 | ADMITJUNE | DATA | 9216 | 05April 10:52:56 |
| 4 | COMPANY | DATA | 5120 | 05Apri1 10:52:56 |
| 5 | CREDIT | DATA | 5120 | 05Apri1 10:52:55 |
| 6 | DIABETES | DATA | 9216 | 05Apri1 10:52:57 |
| 7 | EUROPE | DATA | 13312 | 05April 10:52:57 |

**Figure 2.11:** Output from PROC CONTENTS on SAS Library Clinic

To view the descriptor information for only the clinic.Admit data set, you can submit the following PROC CONTENTS step:

```
proc contents data=clinic.admit;
run;
```

The output from this step lists information for clinic.admit, including an alphabetic list of the variables in the data set.

| | | | |
|---|---|---|---|
| Data Set Name | CLINIC.ADMIT | Observations | 21 |
| Member Type | DATA | Variables | 9 |

| | | | |
|---|---|---|---|
| Engine | V9 | Indexes | 0 |
| Created | Tuesday, January 25. 2011 11:13:25 AM | Observation Length | 64 |
| Last Modified | Tuesday, January 25, 2011 11:13:25 AM | Deleted Observations | 0 |
| Protection | | Compressed | NO |
| Data Set Type | | Sorted | NO |
| Label | | | |
| Data Representation | WINDOWS_32 | | |
| Encoding | wlatinl Western (Windows) | | |

**Figure 2.12:** PROC CONTENTS Output Showing the Descriptor Information for a Single Data Set in a Library

| Engine/Host Dependent Information | |
|---|---|
| Data Set Page Size | 8192 |
| Number of Data Set Pages | 1 |
| First Data Page | 1 |
| Max Obs per Page | 127 |
| Obs in First Data Page | 21 |
| Number of Data Set Repairs | 0 |
| Filename | c:\clinic\adnnit.sas7bdat |
| Release Created | 9.0301MO |
| Host Created | W32_7PRO |

**Figure 2.13:** PROC CONTENTS Output Showing the Engine/Host Dependent information for a Single Data Set in a Library

| Alphabetic List of Variables and Attributes | | | | |
|---|---|---|---|---|
| # | Variable | Type | Len | Format |
| 8 | ActLevel | Char | 4 | |
| 4 | Age | Num | 8 | |
| 5 | Date | Num | 8 | |
| 9 | Fee | Num | 8 | 7.2 |
| 6 | Height | Num | 8 | |
| 1 | ID | Char | 4 | |
| 2 | Name | Char | 14 | |
| 3 | Sex | Char | 1 | |
| 7 | Weight | Num | 8 | |

**Figure 2.14:** PROC CONTENTS Output Showing an Alphabetic List of the Variables in the Data Set

## Using PROC DATASETS

In addition to PROC CONTENTS, you can use PROC DATASETS to view the contents of a SAS library or a SAS data set. PROC DATASETS also enables you to perform a number of management tasks such as copying, deleting, or modifying SAS files.

PROC CONTENTS and PROC DATASETS overlap in terms of functionality. Generally, these two function the same:

- the CONTENTS procedure
- the CONTENTS statement in the DATASETS procedure.

```
PROC CONTENTS<options>;  PROC DATASETS<options>;
RUN;                          CONTENTS<options>;
                       QUIT;
```

The major difference between the CONTENTS procedure and the CONTENTS statement in PROC DATASETS is the default for *libref* in the DATA= option. For PROC CONTENTS, the default is either Work or User. For the CONTENTS statement, the default is the libref of the procedure input library. Notice also that PROC DATASETS supports RUN-group processing. It uses a QUIT statement to end the procedure. The QUIT statement and the RUN statement are not required.

However, the options for the PROC CONTENTS statement and the CONTENTS statement in the DATASETS procedure are the same. For example, the following PROC steps produce essentially the same output (with minor formatting differences):

```
proc datasets;
   contents data=clinic._all_ nods;

proc contents data=clinic._all_ nods;
run;
```

**Additional Note** In addition to the CONTENTS statement, PROC DATASETS also uses several other statements. These statements enable you to perform tasks that PROC CONTENTS does not perform. For more information about PROC DATASETS, see the SAS documentation.

### Viewing Descriptor Information for a SAS Data Set Using VARNUM

As with PROC CONTENTS, you can also use PROC DATASETS to display the descriptor information for a specific SAS data set.

By default, PROC CONTENTS and PROC DATASETS list variables *alphabetically*. To list variable names in the order of their *logical* position (or creation order) in the data set, you can specify the VARNUM option in PROC CONTENTS or in the CONTENTS statement in PROC DATASETS.

For example, either of these programs creates output that includes the list of variables shown below:

```
proc datasets;
   contents data=clinic.admit varnum;

proc contents data=clinic.admit varnum;
run;
```

*Note:* If you are using the sample data in the sasuser library, you may want to specify sasuser as the libref (instead of clinic):

```
contents data=sasuser.admit varnum;

proc contents data=sasuser.admit varnum;
run;
```

**Variables in Creation Order**

| # | Variable | Type | Len | Format |
|---|---|---|---|---|
| 1 | ID | Char | 4 | |
| 2 | Name | Char | 14 | |
| 3 | Sex | Char | 1 | |
| 4 | Age | Num | 8 | |
| 5 | Date | Num | 8 | |
| 6 | Height | Num | 8 | |
| 7 | Weight | Num | 8 | |
| 8 | ActLevel | Char | 4 | |
| 9 | Fee | Num | 8 | 7.2 |

**Figure 2.15:** Viewing Descriptor Information for a SAS Data Set Using VARNUM

## Specifying Results Formats

Next, let's consider the appearance and format of your SAS output.

### HTML and Listing Formats

In SAS 9.3 and later versions, when running SAS in windowing mode in the Windows and UNIX operating environments, HTML output is created by default. In other platforms, you can create HTML output using programming statements. When running SAS in batch mode, the default format is LISTING.

Let's look at these two result formats. The following PROC PRINT output is a listing of part of the SAS data set Clinic.Therapy:

| The SAS System | | | | 11:02 Tuesday, April 12, 2011 |
|---|---|---|---|---|
| Obs | Date | Aer Class | Walk JogRun | Swim |
| 7 | JUL2009 | 67 | 102 | 72 |
| 9 | SEP2009 | 78 | 77 | 54 |
| 10 | OCT2009 | 81 | 62 | 47 |
| 1 1 | NOV2009 | 84 | 31 | 52 |
| 22 | OCT2010 | 78 | 70 | 41 |
| 23 | NOV2010 | 82 | 44 | 58 |
| 24 | DEC2010 | 93 | 57 | 47 |

**Figure 2.16:** LISTING Output

This is HTML output from the same program:

| The SAS System | | | | |
|---|---|---|---|---|
| Obs | Date | AerClass | WalkJogRun | Swim |
| 7 | JUL2009 | 67 | 102 | 72 |
| 9 | SEP2009 | 78 | 77 | 54 |
| 10 | OCT2009 | 81 | 62 | 47 |
| 11 | NOV2009 | 84 | 31 | 52 |
| 22 | OCT2010 | 78 | 70 | 41 |
| 23 | NOV2010 | 82 | 44 | 58 |
| 24 | DEC2010 | 93 | 57 | 47 |

**Figure 2.17:** HTML Output

**Additional Note** If you aren't running SAS in a desktop operating environment, you might want to skip this topic and go on to "Setting System Options" on page 58. For details on creating HTML output using programming statements on any SAS platform, see "Producing HTML Output" on page 280.

### The Results Tab of the Preferences Window

You use the Preferences window to set the result format(s) that you prefer. Your preferences are saved until you modify them, and they apply to all output that you create.

To open this window in desktop operating environments, select **Tools** ⇨ **Options** ⇨ **Preferences**. Then click the **Results** tab. You can remember this sequence using the mnemonic TOPR (pronounced "topper"). You can choose **Listing, HTML**, or both. You can also specify options for displaying and storing your results.

**Additional Note** Results tab options may differ somewhat, depending on your operating environment. The example below is from the Windows operating environment.

The following display shows the SAS **Results** tab with the new default settings specified.

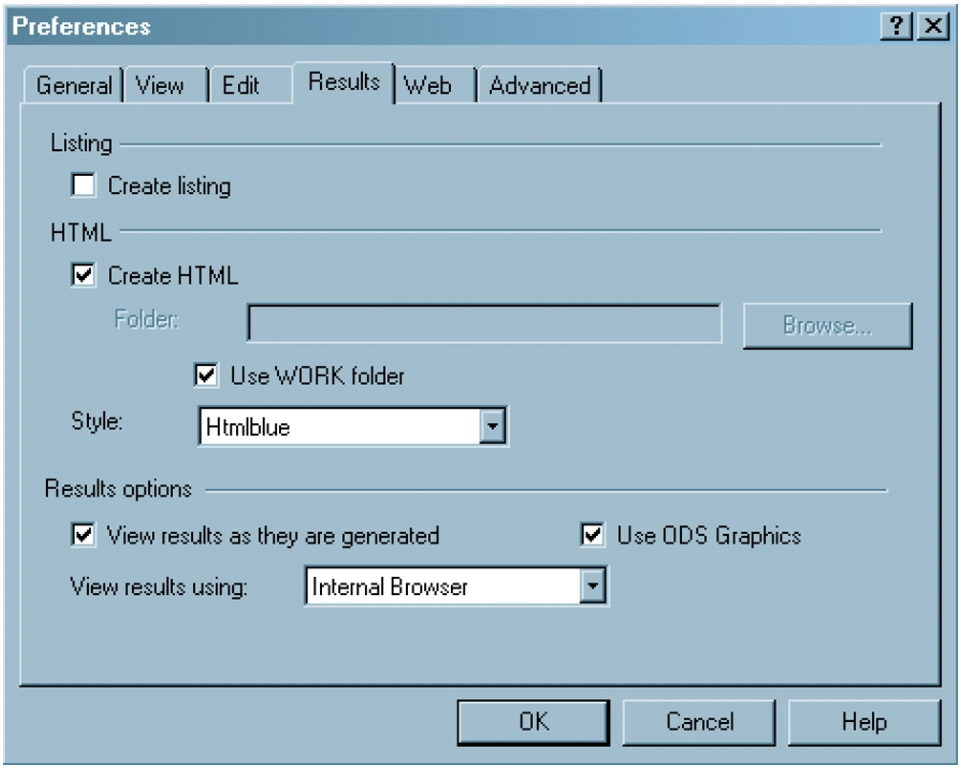



**Figure 2.18:** SAS Results Tab with the New Default Settings

The default settings in the **Results** tab are as follows:

- The **Create listing** check box is not selected, so LISTING output is not created.
- The **Create HTML** check box is selected, so HTML output is created.
- The **Use WORK folder** check box is selected, so both HTML and graph image files are saved in the WORK folder (and not your current directory).
- The default style, HTMLBlue, is selected from the **Style** drop-down list.
- The **Use ODS Graphics** check box is selected, so ODS Graphics is enabled.
- **Internal browser** is selected from the **View results using:** drop-down list, so results are viewed in an internal SAS browser.

If you create HTML files, they are stored in the folder that you specify and are by default incrementally named sashtml.htm, sashtml1.htm, sashtml2.htm, and so on, throughout your SAS session.

Now look at the two choices for viewing HTML results: Internal browser and Preferred web browser. (These options appear in the Results tab only in the Windows operating environment.)

### Internal Browser vs. Preferred Web Browser

To view HTML output, you can choose between these two options in the Results tab of the Preferences window in the Windows operating environment:

- the Internal browser, called the Results Viewer window. SAS provides this browser as part of your SAS installation.
- the Preferred web browser. If you select this option, SAS uses the browser that is specified in the Web tab of the

Preferences window. By default, this is the default browser for your PC.

### Internal Browser

The **Results Viewer** is displayed as a SAS window, as shown below.

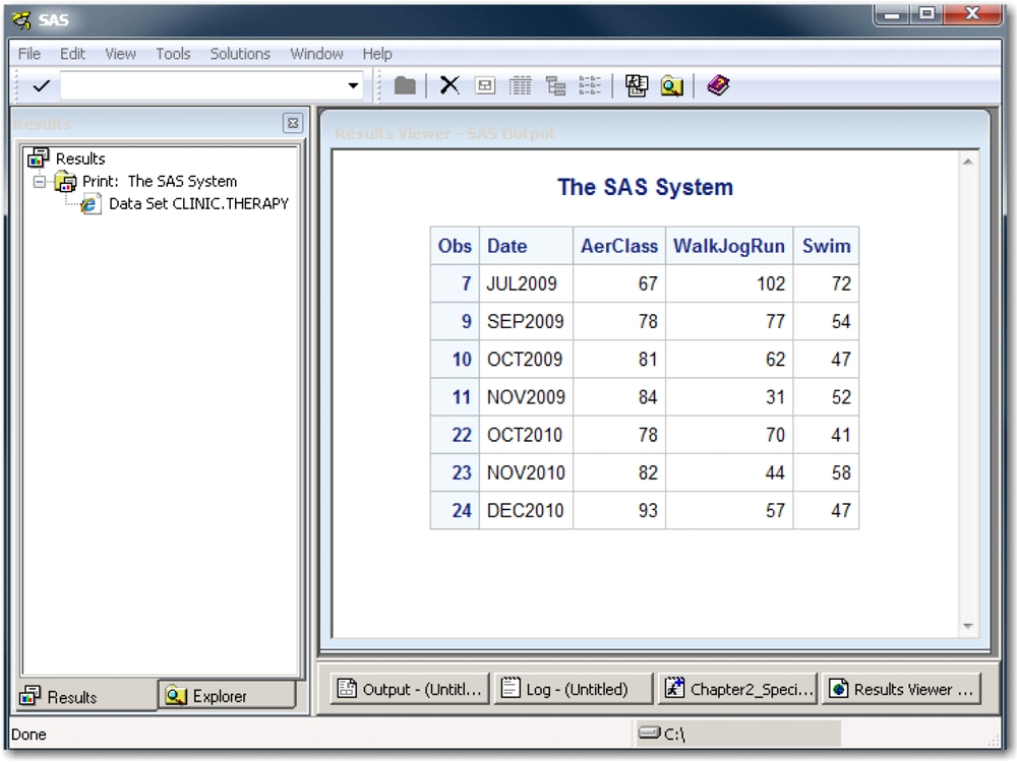
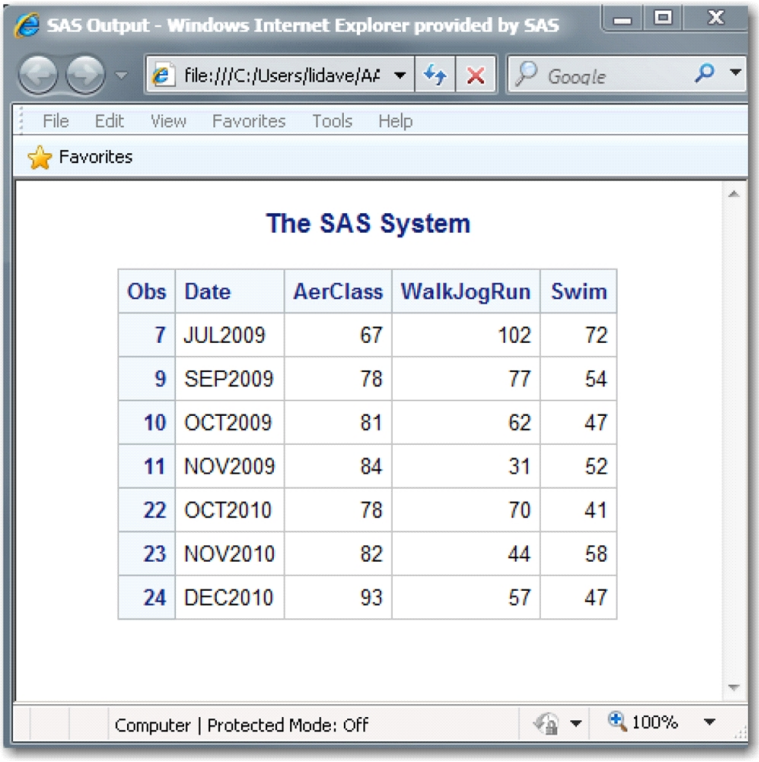



**Figure 2.19:** Internal Browser Showing HTML Output

### Preferred Web Browser

If you select the preferred web browser, your HTML output is displayed in a separate browser that is independent of SAS. For example, the HTML output below is displayed in Internet Explorer.




| Obs | Date | AerClass | WalkJogRun | Swim |
|---|---|---|---|---|
| 7 | JUL2009 | 67 | 102 | 72 |
| 9 | SEP2009 | 78 | 77 | 54 |
| 10 | OCT2009 | 81 | 62 | 47 |
| 11 | NOV2009 | 84 | 31 | 52 |
| 22 | OCT2010 | 78 | 70 | 41 |
| 23 | NOV2010 | 82 | 44 | 58 |
| 24 | DEC2010 | 93 | 57 | 47 |

**Figure 2.20:** Preferred Web Browser Showing HTML Output

## Setting System Options

### Overview

If you create your procedure output as LISTING output, you can also control the appearance of your output by setting system options such as

- line size (the maximum width of the log and output)
- page size (the number of lines per printed page of output)
- the display of page numbers
- the display of date and time.

**Additional Note** The above options do not affect the appearance of HTML output.

All SAS system options have default settings that are used unless you specify otherwise. For example, page numbers are automatically displayed in LISTING output (unless your site modifies this default).

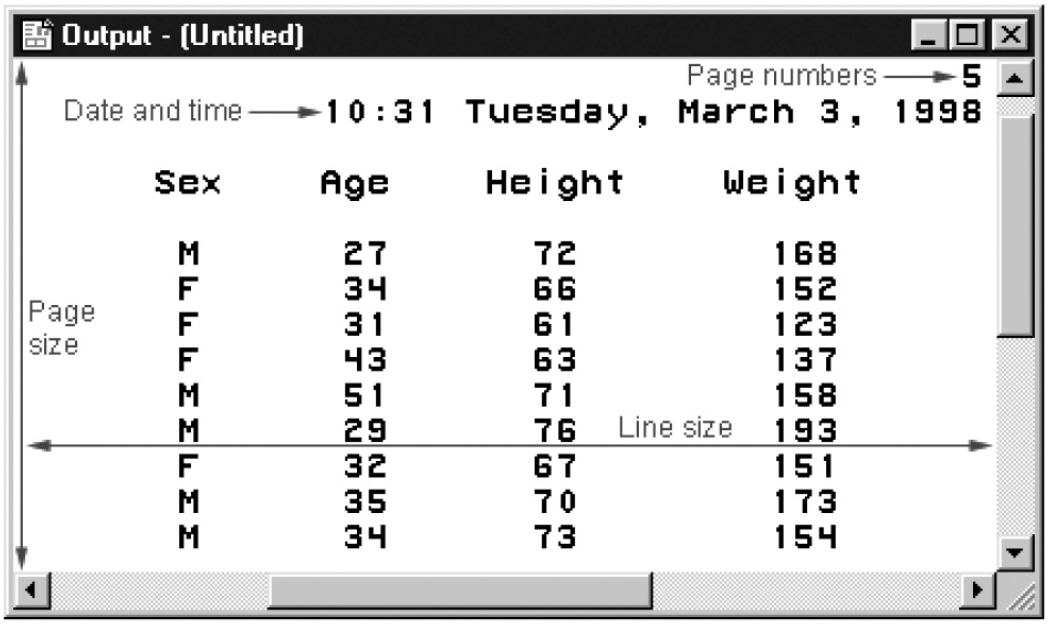



**Figure 2.21:** LISTING Output Showing Default Settings

## Changing System Options

To modify system options in your LISTING output, you submit an OPTIONS statement. You can place an OPTIONS statement anywhere in a SAS program to change the settings from that point onward. However, it is good programming practice to place OPTIONS statements outside of DATA or PROC steps so that your programs are easier to read and debug.

Because the OPTIONS statement is global, the settings remain in effect until you modify them, or until you end your SAS session.

---

General form, OPTIONS statement:

**OPTIONS** *options*;

where *options* specifies one or more system options to be changed. The available system options depend on your host operating system.

---

## Example: DATE | NODATE and NUMBER | NONUMBER Options

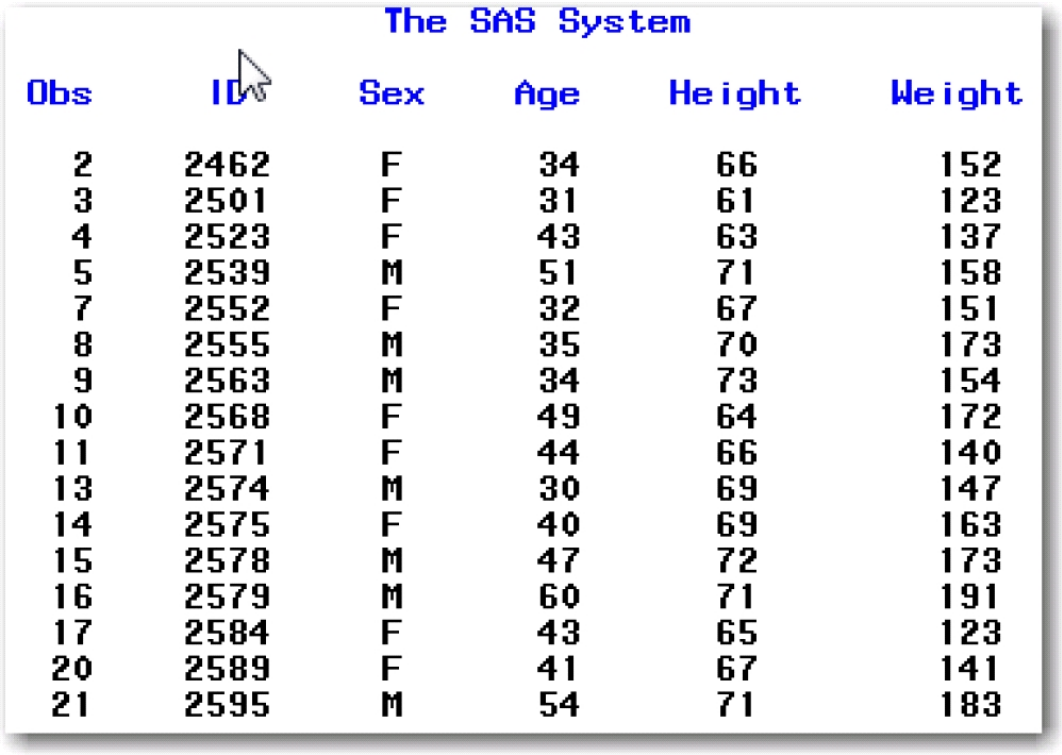
By default, page numbers and dates appear with LISTING output. The following OPTIONS statement suppresses the printing of both page numbers and the date and time in LISTING output.

```
options nonumber nodate;
```

In the following example, the NONUMBER and NODATE system options suppress the display of page numbers and the date in the PROC PRINT output. Page numbers are not displayed in the PROC FREQ output, either, but the date does appear at the top of the PROC FREQ output since the DATE option was specified.

```
ods listing;
options nonumber nodate;
proc print data=sasuser.admit;
   var id sex age height weight;
   where age>=30;
run;
options date;
proc freq data=sasuser.diabetes;
   where fastgluc>=300;
   tables sex;
run;
```

```
proc print data=sasuser.diabetes;
run;
ods listing close;
```



The SAS System

| Obs | ID | Sex | Age | Height | Weight |
|---|---|---|---|---|---|
| 2 | 2462 | F | 34 | 66 | 152 |
| 3 | 2501 | F | 31 | 61 | 123 |
| 4 | 2523 | F | 43 | 63 | 137 |
| 5 | 2539 | M | 51 | 71 | 158 |
| 7 | 2552 | F | 32 | 67 | 151 |
| 8 | 2555 | M | 35 | 70 | 173 |
| 9 | 2563 | M | 34 | 73 | 154 |
| 10 | 2568 | F | 49 | 64 | 172 |
| 11 | 2571 | F | 44 | 66 | 140 |
| 13 | 2574 | M | 30 | 69 | 147 |
| 14 | 2575 | F | 40 | 69 | 163 |
| 15 | 2578 | M | 47 | 72 | 173 |
| 16 | 2579 | M | 60 | 71 | 191 |
| 17 | 2584 | F | 43 | 65 | 123 |
| 20 | 2589 | F | 41 | 67 | 141 |
| 21 | 2595 | M | 54 | 71 | 183 |

**Figure 2.22:** PROC PRINT LISTING Output for sasuser.diabetes

| The SAS System | | | | 11:11 Saturday, May 14, 2011 |
|---|---|---|---|---|
| The FREQ Procedure | | | | |
| Sex | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
| F | 2 | 25.00 | 2 | 25.00 |
| M | 6 | 75.00 | 8 | 100.00 |

**Figure 2.23:** PROC FREQ LISTING Output for sasuser.diabetes

| The SAS System | | | | | | | 11 :03 Monday, May 16, 2011 | |
|---|---|---|---|---|---|---|---|---|
| Obs | ID | Sex | Age | Height | Weight | Pulse | Fast Glue | Post Glue |
| 1 | 2304 | F | 16 | 61 | 102 | 100 | 568 | 625 |
| 2 | 1128 | M | 43 | 71 | 218 | 76 | 156 | 208 |
| 3 | 4425 | F | 48 | 66 | 162 | 80 | 244 | 322 |
| 4 | 1387 | F | 57 | 64 | 142 | 70 | 177 | 206 |
| 5 | 9012 | F | 39 | 63 | 157 | 68 | 257 | 318 |
| 6 | 6312 | M | 52 | 72 | 240 | 77 | 362 | 413 |
| 7 | 5438 | F | 42 | 62 | 168 | 83 | 247 | 304 |
| 8 | 3788 | M | 38 | 73 | 234 | 71 | 486 | 544 |
| 9 | 9125 | F | 56 | 64 | 159 | 70 | 166 | 215 |
| 10 | 3438 | M | 15 | 66 | 140 | 67 | 492 | 547 |
| 11 | 1274 | F | 50 | 65 | 153 | 70 | 193 | 271 |
| 12 | 3347 | M | 53 | 70 | 193 | 78 | 271 | 313 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 13 | 2486 | F | 63 | 65 | 157 | 70 | 152 | 224 |
| 14 | 1129 | F | 48 | 61 | 137 | 69 | 267 | 319 |
| 15 | 9723 | M | 52 | 75 | 219 | 65 | 348 | 403 |
| 16 | 8653 | M | 49 | 68 | 185 | 79 | 259 | 311 |
| 17 | 4451 | M | 54 | 71 | 196 | 81 | 373 | 431 |
| 18 | 3279 | M | 40 | 70 | 213 | 82 | 447 | 504 |
| 19 | 4759 | F | 60 | 68 | 164 | 71 | 155 | 215 |
| 20 | 6488 | F | 59 | 64 | 154 | 75 | 362 | 409 |

**Figure 2.24:** PROC PRINT LISTING Output for sasuser.diabetes

| The SAS System | | | | 13:31 Monday, April 18, 2011 |
|---|---|---|---|---|
| The FREQ Procedure | | | | |
| Sex | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
| F | 2 | 25.00 | 2 | 25.00 |
| M | 6 | 75.00 | 8 | 100.00 |

**Figure 2.25:** PROC FREQ LISTING Output with Date and Still No Page Number

### Example: PAGENO= Option

If you print page numbers, you can specify the beginning page number for your LISTING report by using the PAGENO= option. If you don't specify PAGENO=, output is numbered sequentially throughout your SAS session, starting with page 1.

In the following example, the output pages are numbered sequentially throughout the SAS session, beginning with number 3.

```
ods listing;
options nodate number pageno=3;
proc print data=hrd.funddrive;
run;
ods listing close;
```

Since SAS 9.3 does not creating LISTING output by default, the ODS LISTING statement was added to generate LISTING output.

| The SAS System | | | | | |
|---|---|---|---|---|---|
| Obs | Last Name | Qtr 1 | Qtr2 | Qtr3 | Qtr4 |
| 1 | ADAMS | 18 | 18 | 20 | 20 |
| 2 | ALEXANDER | 15 | 18 | 15 | 10 |
| 3 | APPLE | 25 | 25 | 25 | 25 |
| 4 | ARTHUR | 10 | 25 | 20 | 30 |
| 5 | AVERY | 15 | 15 | 15 | 15 |
| 6 | BAREFOOT | 20 | 20 | 20 | 20 |
| 7 | BAUCOM | 25 | 20 | 20 | 30 |
| 8 | BLAIR | 10 | 10 | 5 | 10 |
| 9 | BLALOCK | 5 | 10 | 10 | 15 |
| 10 | BOSTIC | 20 | 25 | 30 | 25 |
| 11 | BRADLEY | 12 | 16 | 14 | 18 |
| 12 | BRADY | 20 | 20 | 20 | 20 |
| 13 | BROWN | 18 | 18 | 18 | 18 |
| 14 | BRYANT | 16 | 18 | 20 | 18 |

| | | | | | |
|---|---|---|---|---|---|
| 15 | BURNETTE | 10 | 10 | 10 | 10 |
| IE | CHEUNG | 30 | 30 | 30 | 30 |
| 17 | LEHMAN | 20 | 20 | 20 | 20 |
| 18 | VALADEZ | 14 | 18 | 40 | 25 |

**Figure 2.26:** LISTING Output with the PAGENO= Option Set

### Example: PAGESIZE= Option

The PAGESIZE= option (alias PS=) specifies how many lines each page of output contains. In the following example, each page of the output that the PRINT procedure produces contains 15 lines (including those used by the title, date, and so on).

```
options number date pagesize=15;
proc print data=sasuser.admit;
run;
```

| | | | | | |
|---|---|---|---|---|---|
| | **The SAS System** | | | **13:23 Tuesday, April 12, 2011 17** | |
| Obs | LastName | Qtr 1 | Qtr2 | Qtr3 | Qtr4 |
| 12 | BRADY | 20 | 20 | 20 | 20 |
| 13 | BROWN | 18 | 18 | 18 | 18 |
| 14 | BRYANT | 16 | 18 | 20 | 18 |
| 15 | BURNETTE | 10 | 10 | 10 | 10 |
| 17 | CHEUNG | 30 | 30 | 30 | 30 |
| 17 | LEHMAN | 20 | 20 | 20 | 20 |
| 18 | VALADEZ | 14 | 18 | 40 | 25 |

**Figure 2.27:** LISTING Output with PAGESIZE= 15

### Example: LINESIZE= Option

The LINESIZE= option (alias LS=) specifies the width of the print line for your procedure output and log. Observations that do not fit within the line size continue on a different line.

In the following example, the observations are longer than 64 characters, so the observations continue on a subsequent page.

```
ODS listing;
options number linesize=64;
proc print data=flights.europe;
run;
ODS listing close;
```

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | **The SAS System** | | | **55** |
| | | | | | | **13:23 Tuesday, Apri1 12, 2011** |
| Obs | Transferred | Non Revenue | Deplaned | Capacity | Day of Month | Revenue |
| 1 | 17 | 7 | 222 | 250 | 1 | 150634 |
| 2 | 8 | 3 | 163 | 250 | 1 | 156804 |
| 3 | 15 | 5 | 227 | 250 | 1 | 190098 |
| 4 | 13 | 4 | 222 | 250 | 1 | 150634 |
| 5 | 14 | 6 | 158 | 250 | 1 | 193930 |
| 6 | 18 | 7 | 172 | 250 | 2 | 166470 |
| 7 | 8 | 1 | 114 | 250 | 2 | 167772 |
| 8 | 7 | 4 | 187 | 250 | 2 | 163248 |

**Figure 2.28:** LISTING Output with the LINESIZE= Option Set, Page 1

| The SAS System | | | | | | | | | 56 |
|---|---|---|---|---|---|---|---|---|---|
| 13:23 Tuesday, April 12, 2011 | | | | | | | | | |
| Obs | Flight | Date | Depart | Orig | Dest | Miles | Mai1 | Freight | Boarded |
| 9 | 821 | 09MAR99 | 14.56 | LGA | LON | 3442 | 219 | 368 | 203 |
| 10 | 271 | 09MAR99 | 13.17 | LGA | PAR | 3635 | 357 | 282 | 159 |
| 11 | 821 | 10MAR99 | 9.31 | LGA | LON | 3442 | 389 | 479 | 188 |
| 12 | 271 | 10MAR99 | 11.40 | LGA | PAR | 3856 | 415 | 463 | 182 |
| 13 | 622 | 03MAR99 | 12.19 | LGA | FAR | 3857 | 296 | 414 | 180 |
| 14 | 821 | 03MAR99 | 14.56 | LGA | LON | 3442 | 448 | 282 | 151 |
| 15 | 271 | 03MAR99 | 13.17 | LGA | PAR | 3635 | 352 | 351 | 147 |
| 16 | 219 | 04MAR99 | 9.31 | LGA | LON | 3442 | 331 | 376 | 232 |

**Figure 2.29:** LISTING Output with the LINESIZE= Option Set, Page 2

## Handling Two-Digit Years: Year 2000 Compliance

If you use two-digit year values in your data lines, external files, or programming statements, you should consider another important system option, the YEARCUTOFF= option. This option specifies which 100-year span is used to interpret two-digit year values.

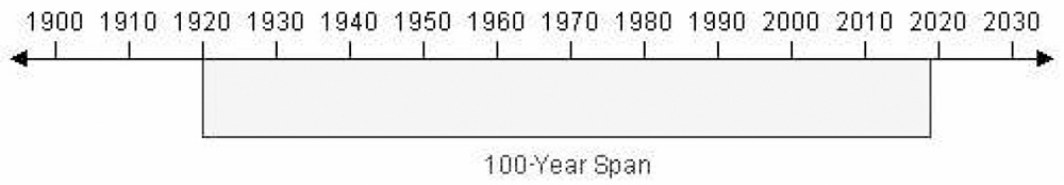



**Figure 2.30:** The Default 100 Year Span in SAS

All versions of SAS represent dates correctly from 1582 A.D. to 20,000 A.D. (Leap years, century, and fourth-century adjustments are made automatically. Leap seconds are ignored, and SAS does not adjust for Daylight Savings Time.) However, you should be aware of the YEARCUTOFF= value to ensure that you are properly interpreting two-digit years in data lines.

As with other system options, you specify the YEARCUTOFF= option in the OPTIONS statement:

```
options yearcutoff=1925;
```

## How the YEARCUTOFF= Option Works

When a two-digit year value is read, SAS interprets it based on a 100-year span which starts with the YEARCUTOFF= value. The default value of YEARCUTOFF= is **1920**.


**Figure 2.31:** Default YEARCUTOFF= Date (1920)

### Table 2.4: Date Expressions and How They Are Interpreted

| Date Expression | Interpreted As |
|---|---|

| | |
|---|---|
| 12/07/41 | 12/07/1941 |
| 18Dec15 | 18Dec2015 |
| 04/15/30 | 04/15/1930 |
| 15Apr95 | 15Apr1995 |

However, you can override the default and change the value of YEARCUTOFF= to the first year of another 100-year span. For example, if you specify YEARCUTOFF=1950, then the 100-year span will be from 1950 to 2049.

```
options yearcutoff=1950;
```

Using YEARCUTOFF=1950, dates are interpreted as shown below:


**Figure 2.32:** Interpreting Dates When YEARCUTOFF=1950

**Table 2.5: Date Expressions and How They Are Interpreted**

| Date Expression | Interpreted As |
|---|---|
| 12/07/41 | 12/07/2041 |
| 18Dec15 | 18Dec2015 |
| 04/15/30 | 04/15/2030 |
| 15Apr95 | 15Apr1995 |

### How Four-Digit Years Are Handled

Remember, the value of the YEARCUTOFF= system option affects only two-digit year values. A date value that contains a four-digit year value will be interpreted correctly even if it does not fall within the 100-year span set by the YEARCUTOFF= system option.

You can learn more about reading date values in "Reading Date and Time Values" on page 571.

### Using System Options to Specify Observations

You've seen how to use SAS system options to change the appearance of output and interpret two-digit year values. You can also use the FIRSTOBS= and OBS= system options to specify the observations to process from SAS data sets.

You can specify either or both of these options as needed. That is, you can use

- FIRSTOBS= to start processing at a specific observation
- OBS= to stop processing after a specific observation
- FIRSTOBS= and OBS= together to process a specific group of observations.

---

General form, FIRSTOBS= and OBS= options in an OPTIONS statement:

**FIRSTOBS**=*n*

**OBS**=*n*

where *n* is a positive integer. For FIRSTOBS=, *n* specifies the number of the *first* observation to process. For OBS=, *n* specifies the number of the *last* observation to process. By default, FIRSTOBS=1. The default value for OBS= is MAX,

which is the largest signed, eight-byte integer that is representable in your operating environment. The number can vary depending on your operating system.

**Caution** Each of these options applies to every input data set that is used in a program or a SAS process.

### Examples: FIRSTOBS= and OBS= Options

The data set clinic.heart contains 20 observations. If you specify FIRSTOBS=10, SAS reads the 10th observation of the data set first and reads through the last observation (for a total of 11 observations).

```
options firstobs=10;
proc print data=sasuser.heart;
run;
```

The PROC PRINT step produces the following output:

The SAS System

| Obs | Patient | Sex | Survive | Shock | Arterial | Heart | Cardiac | Urinary |
|---|---|---|---|---|---|---|---|---|
| 10 | 509 | 2 | SURV | OTHER | 79 | 84 | 256 | 90 |
| 11 | 742 | 1 | DIED | HYPOVOL | 100 | 54 | 135 | 0 |
| 12 | 609 | 2 | DIED | NONSHOCK | 93 | 101 | 260 | 90 |
| 13 | 318 | 2 | DIED | OTHER | 72 | 81 | 410 | 405 |
| 14 | 412 | 1 | SURV | BACTER | 61 | 87 | 296 | 44 |
| 15 | 601 | 1 | DIED | BACTER | 84 | 101 | 260 | 377 |
| 16 | 402 | 1 | SURV | CARDIO | 88 | 137 | 312 | 75 |
| 17 | 98 | 2 | SURV | CARDIO | 84 | 87 | 260 | 377 |
| 18 | 4 | 1 | SURV | HYPOVOL | 81 | 149 | 406 | 200 |
| 19 | 50 | 2 | SURV | HYPOVOL | 72 | 111 | 332 | 12 |
| 20 | 2 | 2 | DIED | OTHER | 101 | 114 | 424 | 97 |

**Figure 2.33:** PROC PRINT Output with FIRSTOBS=10

If you specify OBS=10 instead, SAS reads through the 10th observation, in this case for a total of 10 observations. (Notice that FIRSTOBS= has been reset to the default value.)

```
options firstobs=1 obs=10;
proc print data=sasuser.heart;
run;
```

Now the PROC PRINT step produces this output:

The SAS System

| Obs | Patient | Sex | Survive | Shock | Arterial | Heart | Cardiac | Urinary |
|---|---|---|---|---|---|---|---|---|
| 1 | 203 | 1 | SURV | NONSHOCK | 88 | 95 | 66 | 110 |
| 2 | 54 | 1 | DIED | HYPOVOL | 83 | 183 | 95 | 0 |
| 3 | 654 | 2 | SURV | CARDIO | 72 | 111 | 332 | 12 |
| 4 | 210 | 2 | DIED | BACTER | 74 | 97 | 369 | 0 |
| 5 | 101 | 2 | DIED | NEURO | 80 | 130 | 291 | 0 |
| 6 | 102 | 2 | SURV | OTHER | 87 | 107 | 471 | 65 |
| 7 | 529 | 1 | DIED | CARDIO | 103 | 106 | 217 | 15 |
| 8 | 524 | 2 | DIED | CARDIO | 145 | 99 | 156 | 10 |
| 9 | 426 | 1 | SURV | OTHER | 68 | 77 | 410 | 75 |
| 10 | 509 | 2 | SURV | OTHER | 79 | 84 | 256 | 90 |

**Figure 2.34:** PROC PRINT Output with FIRSTOBS=1 and Obs=10

Combining FIRSTOBS= and OBS= processes observations in the middle of the data set. For example, the following program processes only observations 10 through 15, for a total of 6 observations:

```
options firstobs=10 obs=15;
proc print data=sasuser.heart;
run;
```

Here is the output:

**The SAS System**

| Obs | Patient | Sex | Survive | Shock | Arterial | Heart | Cardiac | Urinary |
|---|---|---|---|---|---|---|---|---|
| 10 | 509 | 2 | SURV | OTHER | 79 | 84 | 256 | 90 |
| 11 | 742 | 1 | DIED | HYPOVOL | 100 | 54 | 135 | 0 |
| 12 | 609 | 2 | DIED | NONSHOCK | 93 | 101 | 260 | 90 |
| 13 | 318 | 2 | DIED | OTHER | 72 | 81 | 410 | 405 |
| 14 | 412 | 1 | SURV | BACTER | 61 | 87 | 296 | 44 |
| 15 | 601 | 1 | DIED | BACTER | 84 | 101 | 260 | 377 |

**Figure 2.35:** PROC PRINT Output with FIRSTOBS=10 and Obs=15

To reset the number of the last observation to process, you can specify OBS=MAX in the OPTIONS statement.

```
options obs=max;
```

This instructs any subsequent SAS programs in the SAS session to process through the last observation in the data set being read.

### Using FIRSTOBS= and OBS= for Specific Data Sets

As you saw above, using the FIRSTOBS= or OBS= system options determines the first or last observation, respectively, that is read for all steps for the duration of your current SAS session or until you change the setting. However, you may want to

- override these options for a given data set
- apply these options to a specific data set only.

To affect any single file, you can use FIRSTOBS= or OBS= as data set options instead of as system options. You specify the data set option in parentheses immediately following the input data set name.

**Additional Note** A FIRSTOBS= or OBS= specification from a data set option overrides the corresponding FIRSTOBS= or OBS= system option.

### Example: FIRSTOBS= and OBS= as Data Set Options

As shown in the last example below, this program processes only observations 10 through 15, for a total of 6 observations:

```
options firstobs=10 obs=15;
proc print data=sasuser.heart;
run;
```

You can create the same output by specifying FIRSTOBS= and OBS= as data set options, as follows. The data set options override the system options for this instance only.

```
options firstobs=10 obs=15;
proc print data=sasuser.heart(firstobs=10 obs=15);
run;
```

To specify FIRSTOBS= or OBS= for this program only, you could omit the OPTIONS statement altogether and simply use the data set options.

### The SAS System Options Window

You can also set system options by using the SAS System Options window. The changed options are reset to the defaults at the end of your SAS session.

To view the SAS System Options window, select **Tools** ⇨ **Options** ⇨ **System**.

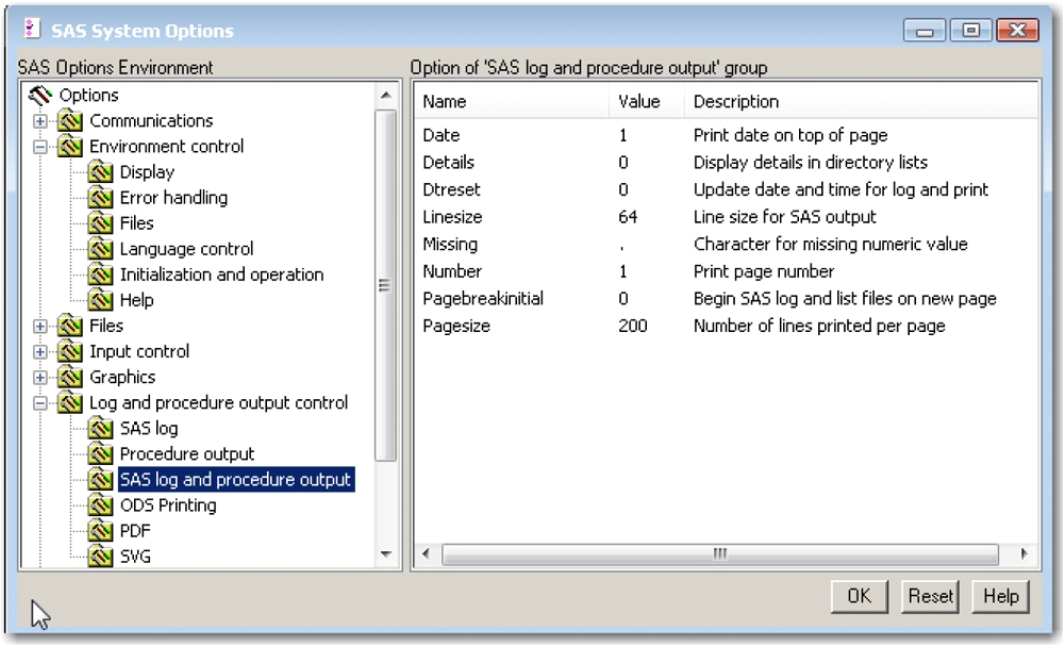



**Figure 2.36:** The SAS System Options Window

### Changing Options

To change an option:

1. Expand the groups and subgroups under SAS Options Environment until you find the option that you want to change. (Options in subgroups are listed in the right pane of the window.)
2. Click the name of the option that you want to change, and display its pop-up menu. Then select one of the choices:
   - **Modify Value** opens a window in which you type or select a new value for the option.
   - **Set to Default** immediately resets the option to its default value.

For example, the SAS System Options window above shows options for the **SAS log and procedure output** subgroup under the group **Log and procedure output control**.

### Finding Options Quickly

To locate an option in the SAS System Options window:

1. Place your cursor over the name of any option group or subgroup, and display its pop-up menu.
2. Click **Find Option**. The Find Option dialog box appears.
3. Type the name of the option that you want to locate, and click **OK**.

The SAS System Options window expands to the appropriate option subgroup. All subgroup options also appear, and the option that you located is highlighted.

### Additional Features

When you set up your SAS session, you can set SAS system options that affect LISTING output, information written to the SAS log, and much more. Here are some additional system options that you are likely to use with SAS:

### Table 2.6: Selected System Options and Their Descriptions

| | |
|---|---|
| FORMCHAR= '*formatting-characters*' | specifies the formatting characters for your output device. Formatting characters are used to construct the outlines of tables as well as dividers for various procedures, such as the FREQ and TABULATE procedures. If you do not specify formatting characters as an option in the procedure, then the default specifications given in the FORMCHAR= system option are used. |
| FORMDLIM= '*delimit-character*' | specifies a character that is used to delimit page breaks in SAS System output. Normally, the delimit character is null. When the delimit character is null, a new physical page starts whenever a page break occurs. |
| LABEL\|NOLABEL | permits SAS procedures to temporarily replace variable names with descriptive labels. The LABEL system option must be in effect before the LABEL option of any procedure can be used. If NOLABEL is specified, then the LABEL option of a procedure is ignored. The default setting is LABEL. |
| REPLACE\|NOREPLACE | specifies whether permanently stored SAS data sets are replaced. If you specify NOREPLACE, a permanently stored SAS data set cannot be replaced with one that has the same name. This prevents you from inadvertently replacing existing SAS data sets. The default setting is REPLACE. |
| SOURCE\|NOSOURCE | controls whether SAS source statements are written to the SAS log. NOSOURCE specifies not to write SAS source statements to the SAS log. The default setting is SOURCE. |

You can also use programming statements to control the result format of each item of procedure output individually. For more information, see "Producing HTML Output" on page 280.

## Chapter Summary

## Text Summary

#### *Referencing SAS Files in SAS Libraries*

To reference a SAS file, you assign a libref (library reference) to the SAS library in which the file is stored. Then you use the libref as the first part of the two-level name *(libref.filename)* for the file. To assign a libref, you can submit a LIBNAME statement. You can store the LIBNAME statement with any SAS program to assign the libref automatically when you submit the program. The LIBNAME statement assigns the libref for the current SAS session only. You must assign a libref each time you begin a SAS session in order to access SAS files that are stored in a permanent SAS library other than clinic. (Work is the default libref for a temporary SAS library.)

You can also use the LIBNAME statement to reference data in files that were created with other software products, such as database management systems. SAS can write to or read from the files by using the appropriate engine for that file type. For some file types, you need to tell SAS which engine to use. For others, SAS automatically chooses the appropriate engine.

#### *Viewing Librefs*

The librefs that are in effect for your SAS session are listed under Libraries in the Explorer window. To view details about a library, double-click **Libraries** (or select **Libraries** ⇨ **Open** from the pop-up menu). Then select **View** ⇨ **Details**. The library's name, engine, host pathname, and date are listed under Active Libraries.

#### *Viewing the Contents of a Library*

To view the contents of a library, double-click the library name in the Explorer window (or select the library name and then select **Open** from the pop-up menu). Files contained in the library are listed under Contents.

#### *Viewing the Contents of a File*

If you are working in a windowing environment, you can display the contents of a file by double-clicking the filename (or selecting the filename and then selecting **Open** from the pop-up menu) under Contents in the Explorer window. If you select a SAS data set, its contents are displayed in the VIEWTABLE window.

If you are working in the z/OS operating environment, you can type ? in the selection field next to a filename in the Explorer window to display a pop-up menu with a list of options for working with that file.

#### *Listing the Contents of a Library*

To list the contents of a library, use the CONTENTS procedure. Append a period and the _ALL_ option to the libref to get a listing of all files in the library. Add the NODS option to suppress detailed information about the files. As an alternative to

PROC CONTENTS, you can use PROC DATASETS.

***Specifying Result Formats***

In desktop operating environments, you can choose to create your SAS procedure output as an HTML document, a listing (traditional SAS output), or both. You choose the results format(s) that you prefer in the Preferences window. Your preferences are saved until you modify them, and they apply to all output that is created during your SAS session. To open this window, select **Tools** ⇨ **Options** ⇨ **Preferences**. Then click the **Results** tab. Choose **Create listing, Create HTML**, or both.

If you choose **Create HTML**, then each HTML file is displayed in the browser that you specify (in the Windows operating environment, the internal browser is the Results Viewer window). HTML files are stored in the location that you specify and are by default incrementally named sashtml.htm, sashtml1.htm, sashtml2.htm, and so on throughout your SAS session. To specify where HTML files are stored, type a path in the **Folder** box (or click **Browse** to locate a pathname). If you prefer to store your HTML files temporarily and to delete them at the end of your SAS session, click **Use WORK** folder instead of specifying a folder. To specify the presentation style for HTML output, you can select an item in the **Style** box.

***Setting System Options***

For your LISTING output, you can also control the appearance of your output by setting system options such as line size, page size, the display of page numbers, and the display of the date and time. (These options do not affect the appearance of HTML output.)

All SAS system options have default settings that are used unless you specify otherwise. For example, page numbers are automatically displayed (unless your site modifies this default). To modify system options, you submit an OPTIONS statement. You can place an OPTIONS statement anywhere in a SAS program to change the current settings. Because the OPTIONS statement is global, the settings remain in effect until you modify them or until you end your SAS session.

If you use two-digit year values in your SAS data lines, you must be aware of the YEARCUTOFF= option to ensure that you are properly interpreting two-digit years in your SAS program. This option specifies which 100-year span is used to interpret two-digit year values.

To specify the observations to process from SAS data sets, you can use the FIRSTOBS= and OBS= options.

You can also use the SAS System Options window to set system options.

***Additional Features***

You can set a number of additional SAS system options that are commonly used.

***Syntax***

```
LIBNAME libref 'SAS-data-library';
LIBNAME libref engine'SAS-data-library';
OPTIONS options;
PROC CONTENTS DATA= libref._ALL_ NODS;
PROC DATASETS;
     CONTENTS DATA= libref._ALL_NODS;
QUIT;
```

***Points to Remember***

- LIBNAME and OPTIONS statements remain in effect for the current SAS session only.
- When you work with date values, check the default value of the YEARCUTOFF= system option and change it if necessary.

## Chapter Quiz

Select the best answer for each question. After completing the quiz, check your answers using the answer key in the appendix.

**1.** If you submit the following program, how does the output look? ?

```
options pagesize=55 nonumber;
proc tabulate data=clinic.admit;
   class actlevel;
```

```
   var age height weight;
   table actlevel,(age height weight)*mean;
run;
options linesize=80;
proc means data=clinic.heart min max maxdec=1;
   var arterial heart cardiac urinary;
   class survive sex;
run;
```

a. The PROC MEANS output has a print line width of 80 characters, but the PROC TABULATE output has no print line width.

b. The PROC TABULATE output has no page numbers, but the PROC MEANS output has page numbers.

c. Each page of output from both PROC steps is 55 lines long and has no page numbers, and the PROC MEANS output has a print line width of 80 characters.

d. The date does not appear on output from either PROC step.

**2.** How can you create SAS output in HTML format on any SAS platform? ?

a. by specifying system options

b. by using programming statements

c. by using SAS windows to specify the result format

d. you can't create HTML output on all SAS platforms

**3.** In order for the date values `05May1955` and `04Mar2046` to be read correctly, what value must the YEARCUTOFF= option have? ?

a. a value between 1947 and 1954, inclusive

b. 1955 or higher

c. 1946 or higher

d. any value

**4.** When you specify an engine for a library, you are always specifying ?

a. the file format for files that are stored in the library.

b. the version of SAS that you are using.

c. access to other software vendors' files.

d. instructions for creating temporary SAS files.

**5.** Which statement prints a summary of all the files stored in the library named Area51? ?

a. `proc contents data=area51._all_ nods;`

b. `proc contents data=area51 _all_ nods;`

c. `proc contents data=area51 _all_ noobs;`

d. `proc contents data=area51 _all_.nods;`

**6.** The following PROC PRINT output was created immediately after PROC TABULATE output. Which system options were specified when the report was created? ?

---

**1**

**10 : 03 Friday, March 17, 2000**

| Obs | ID | Height | Weight | Act Level | Fee |
|---|---|---|---|---|---|
| 1 | 2458 | 72 | 168 | HIGH | 85.20 |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 2462 | 66 | 152 | HIGH | 124.80 |
| 3 | 2501 | 61 | 123 | LOW | 149.75 |
| 4 | 2523 | 63 | 137 | MOD | 149.75 |
| 5 | 2539 | 71 | 158 | LOW | 124.80 |
| 6 | 2544 | 76 | 193 | HIGH | 124.80 |
| 7 | 2552 | 67 | 151 | MOD | 149.75 |
| 8 | 2555 | 70 | 173 | MOD | 149.75 |
| 9 | 2563 | 73 | 154 | LOW | 124.80 |

a. OBS=, DATE, and NONUMBER

b. NUMBER, PAGENO=1, and DATE

c. NUMBER and DATE only

d. none of the above

**7.** Which of the following programs correctly references a SAS data set named *SalesAnalysis* that is stored in a permanent SAS library? ?

a.
```
data saleslibrary.salesanalysis;
     set mydata.quarter1sales;
     if sales>100000;
run;
```

b.
```
data mysales.totals;
     set sales_99.salesanalysis;
     if totalsales>50000;
run;
```

c.
```
proc print data=salesanalysis.quarter1;
      var sales salesrep month;
run;
```

d.
```
proc freq data=1999data.salesanalysis;
     tables quarter*sales;
run;
```

**8.** Which time span is used to interpret two-digit year values if the YEARCUTOFF=option is set to 1950? ?

a. 1950-2049

b. 1950-2050

c. 1949-2050

d. 1950-2000

**9.** Assuming you are using SAS code and not special SAS windows, which one of the following statements is *false*? ?

a. LIBNAME statements can be stored with a SAS program to reference the SAS library automatically when you submit the program.

b. When you delete a libref, SAS no longer has access to the files in the library. However, the contents of the library still exist on your operating system.

c. Librefs can last from one SAS session to another.

d. You can access files that were created with other vendors' software by submitting a LIBNAME statement.

**10.** What does the following statement do? ?

```
libname osiris spss 'c:\myfiles\sasdata\data.spss';
```

a. defines a library called Spss using the OSIRIS engine

b. defines a library called Osiris using the SPSS engine

c. defines two libraries called Osiris and Spss using the default engine

d. defines the default library using the OSIRIS and SPSS engines

### Answers

**1.** Correct answer: c

When you specify a system option, it remains in effect until you change the option or end your SAS session, so both PROC steps generate output that is printed 55 lines per page with no page numbers. If you don't specify a system option, SAS uses the default value for that system option.

**2.** Correct answer: b

You can create HTML output using programming statements on any SAS platform. In addition, on all except mainframe platforms, you can use SAS windows to specify HTML as a result format.

**3.** Correct answer: d

As long as you specify an informat with the correct field width for reading the entire date value, the YEARCUTOFF= option doesn't affect date values that have fourdigit years.

**4.** Correct answer: a

A SAS engine is a set of internal instructions that SAS uses for writing to and reading from files in a SAS library. Each engine specifies the file format for files that are stored in the library, which in turn enables SAS to access files with a particular format. Some engines access SAS files, and other engines support access to other vendors' files.

**5.** Correct answer: a

To print a summary of library contents with the CONTENTS procedure, use a period to append the _ALL_ option to the libref. Adding the NODS option suppresses detailed information about the files.

**6.** Correct answer: b

Clearly, the DATE and NUMBER (page number) options are specified. Because the page number on the output is 1, even though PROC TABULATE output was just produced, PAGENO=1 must also have been specified. If you don't specify PAGENO=, all output in the Output window is numbered sequentially throughout your SAS session.

**7.** Correct answer: b

Librefs must be 1 to 8 characters long, must begin with a letter or underscore, and can contain only letters, numerals, or underscores. After you assign a libref, you specify it as the first level in the two-level name for a SAS file.

**8.** Correct answer: a

The YEARCUTOFF= option specifies which 100-year span is used to interpret two-digit year values. The default value of YEARCUTOFF= is `1920`. However, you can override the default and change the value of YEARCUTOFF= to the first year of another 100-year span. If you specify YEARCUTOFF=`1950`, then the 100-year span will be from `1950` to `2049`.

**9.** Correct answer: c

The LIBNAME statement is global, which means that librefs remain in effect until you modify them, cancel them, or end your SAS session. Therefore, the LIBNAME statement assigns the libref for the current SAS session only. You must assign a libref before accessing SAS files that are stored in a permanent SAS data library.

**10.** Correct answer: b

In the LIBNAME statement, you specify the library name before the engine name. Both are followed by the path.