# Contents

*About This Book*  ................................................................. v
What's New in SAS 9.3 System Options  ....................................... ix
Recommended Reading  .......................................................... xv

## PART 1  About SAS System Options  1

*Chapter 1 • What You Need to Know about SAS System Options*  ........................................ 3
  Definition of System Options  .............................................. 3
  Syntax  .............................................................................. 4
  Using SAS System Options  ................................................... 4
  Comparisons  ...................................................................... 19

## PART 2  SAS Functions for SAS System Options  21

*Chapter 2 • Dictionary of Functions for System Options*  ..................................................... 23
  Dictionary  .......................................................................... 23

## PART 3  SAS System Options  29

*Chapter 3 • Dictionary of System Options*  ......................................................... 31
  SAS System Options Documented in Other SAS Publications  ..................... 35
  SAS System Options by Category  .............................................. 35
  Dictionary  .......................................................................... 68

## PART 4  SAS Procedures for SAS System Options  299

*Chapter 4 • OPTIONS Procedure*  .................................................. 301
  Overview: OPTIONS Procedure  .............................................. 301
  Syntax: OPTIONS Procedure  .................................................. 302
  Displaying a List of System Options  ...................................... 306
  Displaying Information about One or More Options  .......................... 308
  Displaying Information about System Option Groups  .......................... 310
  Displaying Restricted Options  ................................................. 314
  Results: OPTIONS Procedure  ................................................ 315
  Examples: OPTIONS Procedure  .............................................. 316

*Chapter 5 • OPTLOAD Procedure*  .................................................. 321
  Overview: OPTLOAD Procedure  .............................................. 321
  Syntax: OPTLOAD Procedure  ................................................ 321
  Example: Load a Data Set of Saved System Options  .............................. 322
About This Book

Syntax Conventions for the SAS Language

Overview of Syntax Conventions for the SAS Language
SAS uses standard conventions in the documentation of syntax for SAS language elements. These conventions enable you to easily identify the components of SAS syntax. The conventions can be divided into these parts:

- syntax components
- style conventions
- special characters
- references to SAS libraries and external files

Syntax Components
The components of the syntax for most language elements include a keyword and arguments. For some language elements, only a keyword is necessary. For other language elements, the keyword is followed by an equal sign (=).

Keyword
specifies the name of the SAS language element that you use when you write your program. Keyword is a literal that is usually the first word in the syntax. In a CALL routine, the first two words are keywords.

In the following examples of SAS syntax, the keywords are the first words in the syntax:

- `CHAR (string, position)`
- `CALL RANBIN (seed, n, p, x);`
- `ALTER (alter-password)`
- `BEST w.`
- `REMOVE <data-set-name>`

In the following example, the first two words of the CALL routine are the keywords:

- `CALL RANBIN(seed, n, p, x)`

The syntax of some SAS statements consists of a single keyword without arguments:

- `DO;
  ... SAS code ...`
Some system options require that one of two keyword values be specified:

**DUPLEX | NODUPLEX**

The argument specifies a numeric or character constant, variable, or expression. Arguments follow the keyword or an equal sign after the keyword. The arguments are used by SAS to process the language element. Arguments can be required or optional. In the syntax, optional arguments are enclosed between angle brackets.

In the following example, `string` and `position` follow the keyword CHAR. These arguments are required arguments for the CHAR function:

**CHAR (string, position)**

Each argument has a value. In the following example of SAS code, the argument `string` has a value of 'summer', and the argument `position` has a value of 4:

```sas
4:x=char('summer', 4);
```

In the following example, `string` and `substring` are required arguments, while `modifiers` and `startpos` are optional.

**FIND(string, substring <,modifiers> <,startpos>**

**Note:** In most cases, example code in SAS documentation is written in lowercase with a monospace font. You can use uppercase, lowercase, or mixed case in the code that you write.

---

**Style Conventions**

The style conventions that are used in documenting SAS syntax include uppercase bold, uppercase, and italic:

**UPPERCASE BOLD**

Identifies SAS keywords such as the names of functions or statements. In the following example, the keyword ERROR is written in uppercase bold:

```sas
ERROR<message>;
```

**UPPERCASE**

Identifies arguments that are literals.

In the following example of the CMPMODEL= system option, the literals include BOTH, CATALOG, and XML:

```sas
CMPPMODEL = BOTH | CATALOG | XML
```

**italics**

Identifies arguments or values that you supply. Items in italics represent user-supplied values that are either one of the following:

- nonliteral arguments In the following example of the LINK statement, the argument `label` is a user-supplied value and is therefore written in italics:

  ```sas
  LINK label;
  ```

- nonliteral values that are assigned to an argument

  In the following example of the FORMAT statement, the argument DEFAULT is assigned the variable `default-format`:

  ```sas
  FORMAT = variable-1 < ..., variable-nformat><DEFAULT = default-format>;
  ```
Items in italics can also be the generic name for a list of arguments from which you can choose (for example, attribute-list). If more than one of an item in italics can be used, the items are expressed as item-1, ..., item-n.

Special Characters

The syntax of SAS language elements can contain the following special characters:

=  
an equal sign identifies a value for a literal in some language elements such as system options.

In the following example of the MAPS system option, the equal sign sets the value of MAPS:

MAPS = location-of-maps

< >  
angle brackets identify optional arguments. Any argument that is not enclosed in angle brackets is required.

In the following example of the CAT function, at least one item is required:

CAT (item-1 <, ..., item-n>)

|  
a vertical bar indicates that you can choose one value from a group of values. Values that are separated by the vertical bar are mutually exclusive.

In the following example of the CMPMODEL= system option, you can choose only one of the arguments:

CMPMODEL = BOTH | CATALOG | XML

...  
an ellipsis indicates that the argument or group of arguments following the ellipsis can be repeated. If the ellipsis and the following argument are enclosed in angle brackets, then the argument is optional.

In the following example of the CAT function, the ellipsis indicates that you can have multiple optional items:

CAT (item-1 <, ..., item-n>)

'value' or “value”  
indicates that an argument enclosed in single or double quotation marks must have a value that is also enclosed in single or double quotation marks.

In the following example of the FOOTNOTE statement, the argument text is enclosed in quotation marks:

FOOTNOTE <n> <ods-format-options 'text' | “text”>;  

;  
a semicolon indicates the end of a statement or CALL routine.

In the following example each statement ends with a semicolon: data namegame;
length color name $8; color = 'black'; name = 'jack'; game = trim(color) || name; run;
References to SAS Libraries and External Files

Many SAS statements and other language elements refer to SAS libraries and external files. You can choose whether to make the reference through a logical name (a libref or fileref) or use the physical filename enclosed in quotation marks. If you use a logical name, you usually have a choice of using a SAS statement (LIBNAME or FILENAME) or the operating environment's control language to make the association. Several methods of referring to SAS libraries and external files are available, and some of these methods depend on your operating environment.

In the examples that use external files, SAS documentation uses the italicized phrase `file-specification`. In the examples that use SAS libraries, SAS documentation uses the italicized phrase `SAS-library`. Note that `SAS-library` is enclosed in quotation marks:

```
infile file-specification obs = 100;
libname libref 'SAS-library';
```
What's New in SAS 9.3 System Options

Overview

The SAS system options documentation is no longer part of SAS Language Reference: Dictionary. See “Changes to SAS Language Reference: Dictionary” on page xiv. The SAS system options that were previously documented in SAS Language Reference: Dictionary are now documented here, in SAS System Options: Reference.

For easy access to all system options documentation, SAS System Options: Reference also contains the GETOPTION function, the system option procedures OPTIONS, OPTLOAD, and OPTSAVE, and links to system options that are documented in other publications.

New and enhanced features enable you to do the following:

• use checkpoint mode and restart mode for labeled code sections
• reset system options to their start-up or default values
• create a directory that is named in a LIBNAME statement
• use expanded rules for naming SAS data sets, SAS data views, and item store names
• change the page orientation (portrait or landscape) within a file using the ODS PRINTER destination
• control the autocorrection of SAS names
• specify the UTC offset in e-mail
• specify the encoding for the URLENCODE and URLDECODE functions
• use enhancements to the GETOPTION function, system options, and the OPTIONS procedure
• specify the size of the View buffer
• in the second maintenance release for SAS 9.3, the EVENTDS= option is new. The EVENTDS= option enables you to specify event data sets.
Use Checkpoint Mode and Restart Mode for Labeled Code Sections

Batch programs that terminate before completing can be resubmitted starting at labeled code sections if checkpoint mode and restart mode for labeled code sections is enabled.

If the CHKPTCLEAN system option is set and your batch program completes successfully, the contents of the Work library are erased.

For more information about labeled code sections, see “Checkpoint Mode and Restart Mode” in Chapter 8 of SAS Language Reference: Concepts and these system options:

- “CHKPTCLEAN System Option” on page 89
- “LABELCHKPT System Option” on page 164
- “LABELCHKPTLIB= System Option” on page 165
- “LABELRESTART System Option” on page 167

Reset System Options to Their Start-up or Default Values

You can reset a system option to the default shipped value or to the start-up value by using the GETOPTION function.

You can use the DEFAULTVALUE option to obtain the default shipped value for a system option if you want to reset a system option to its default value.

You can use the STARTUPVALUE option to obtain the value of a system option that was used to start SAS either on the command line or in a configuration file.

See “Resetting System Options to the Default or Starting Value” on page 16 and “GETOPTION Function” on page 23.

Create a Directory That Is Named in a LIBNAME Statement

When you specify the DLCREATEDIR system option, SAS creates a directory for the SAS library that is named in a LIBNAME statement if the directory does not exist. See “DLCREATEDIR System Option” on page 110.
Use Expanded Rules for Naming SAS Data Sets, SAS Data Views, and Item Store Names

When you run SAS in any execution mode other than the windowing environment, the rules for naming SAS data sets, data views, and item store names have been expanded to accommodate special and national characters. See “Names in the SAS Language” in Chapter 3 of SAS Language Reference: Concepts and “VALIDMEMNAME= System Option” on page 280.

Change the Orientation of Pages within a File Using ODS PRINTER

The ODS PRINTER destination now supports using the ORIENTATION= system option to change the page orientation within a file. The page orientation can be changed to portrait or landscape. See “ORIENTATION= System Option” on page 191.

Control Autocorrection of SAS Names

Prior to SAS 9.3, SAS automatically attempted to correct misspelled procedure names, procedure keywords, and global statement names. You can use the NOAUTOCORRECT system option to specify that SAS is not to automatically correct these names. See “AUTOCORRECT System Option” on page 72.

Specify the UTC Offset in E-mail

For e-mail that is sent using the FILENAME statement EMAIL (SMTP) access method, specifies a UTC offset that is used in the Date header field of the e-mail message. See “EMAILUTCOFFSET= System Option” on page 128.

Specify the Encoding for the URLENCODE and URLDECODE Functions

Use the URLENCODING= system option to specify whether the argument to the URLENCODE function and to the URLDECODE function is interpreted using the SAS session encoding or UTF-8 encoding. See “URLENCODING= System Option” on page 272.
Specify the Size of the View Buffer

Use the VBUFSIZE= system option to set a global value for the size of the view buffer based on the number of bytes. See “VBUFSIZE= System Option” on page 287.

Specify an Event Data Set

In the second maintenance release for SAS 9.3, the EVENTDS= option is new. The EVENTDS= option enables you to specify event data sets. See “EVENTDS= System Option” on page 133.

Enhancements to the GETOPTION Function

Use the HEXVALUE option to return a system option value as a hexadecimal value.

Use the LOGNUMBERFORMAT option to return a system option numeric value with the punctuation appropriate for a locale, such as a comma or a period.

See “GETOPTION Function” on page 23.

Enhancements to SAS System Options

The following system options have been enhanced:

- APPEND= (p. 68)
  The APPEND= system option cannot be restricted. Also, you can now specify the AUTOEXEC= system option as a value to the APPEND= system option.

- DKRICOND= (p. 108)
- DKROCOND= (p. 109)
  These options are now part of the Error handling system options group as well as the SAS Files group.

- FMTSEARCH= (p. 139)
  If you specify the LOCALE option for a catalog specification, SAS searches the catalog that is associated with the current SAS locale.

- INSERT= (p. 159)
  The INSERT= system option cannot be restricted. Also, you can now specify the AUTOEXEC= system option as a value to the INSERT= system option.

- ORIENTATION= (p. 191)
  You can modify the page orientation for different documents in an output file whose destination is an ODS destination or a universal printer.
VALIDVARNAME= (p. 282)
When VALIDVARNAME=V7 and the variable name is an n literal in a procedure step, the variable name is left-justified and trailing blanks are ignored.

VARLENCHK= (p. 284)
The VARLENCHK= system option indicates that BY variables are not affected by this system option.

Enhancements to the OPTIONS Procedure

For more information about new and enhanced options for the PROC OPTIONS statement, see “PROC OPTIONS Statement” on page 302.

These PROC OPTIONS statement options are new:

LISTINSERTAPPEND
This option lists the system options whose value can be modified by the INSERT and APPEND system options.

LISTRESTRICT
This option lists the system options that can be restricted by your site administrator.

These PROC OPTIONS statement options have been enhanced:

DEFINE
Valid values for an option now display in the SAS log when you specify the DEFINE option.

OPTION=
The OPTION= option now accepts one or more options.

VALUE
If the option was set by a configuration file, the name of the configuration file that set the option now displays in the SAS log when you specify the VALUE option.

System Options That Have Moved to Other Documentation

You can find the system options for the following SQL options in the *SAS SQL Query Window User's Guide*:

- SQLCONSTDATETIME
- SQLREDUCTPUT=
- SQLREDUCTPUTOBS=
- SQLREDUCTPUTVALUES=
- SQLREMERGE
- SQLUNDOPOLICY=

The UNIVERSALPRINT system option is now documented in the *SAS Companion for Windows*. 
Changes to SAS Language Reference: Dictionary

Prior to SAS 9.3, this document was part of SAS Language Reference: Dictionary. Starting with SAS 9.3, SAS Language Reference: Dictionary has been divided into seven documents:

- SAS Data Set Options: Reference
- SAS Formats and Informats: Reference
- SAS Functions and CALL Routines: Reference
- SAS Statements: Reference
- SAS System Options: Reference
- SAS Component Objects: Reference (contains the documentation for the Hash Object and the Java Object)
- Base SAS Utilities: Reference (contains the documentation for the SAS DATA step debugger and the SAS Utility macro %DS2CSV)
Recommended Reading

Here is the recommended reading list for this title:

- *Base SAS Glossary*
- *Base SAS Procedures Guide*
- *Base SAS Utilities: Reference*
- *SAS Companion for UNIX Environments*
- *SAS Companion for Windows*
- *SAS Companion for z/OS*
- *SAS Data Set Options: Reference*
- *SAS Formats and Informats: Reference*
- *SAS Functions and CALL Routines: Reference*
- *SAS Language Interfaces to Metadata*
- *SAS Language Reference: Concepts*
- *SAS Statements: Reference*
- *SAS Scalable Performance Data Engine: Reference*
- *Step-by-Step Programming with Base SAS Software*

The recommended reading list from SAS Press includes the following titles:

- *Carpenter’s Complete Guide to PROC REPORT*
- *Cody's Data Cleaning Techniques Using SAS, Second Edition*
- *Combining and Modifying SAS Data Sets: Examples, Second Edition*
- *Learning SAS by Example*
- *The Little SAS Book: A Primer, Fourth Edition*
- *Output Delivery System: The Basics and Beyond*
- *SAS Functions by Example, Second Edition*
- *SAS Guide to Report Writing: Examples*
For a complete list of SAS publications, go to support.sas.com/bookstore. If you have questions about which titles you need, please contact a SAS Publishing Sales Representative:

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-3228
Fax: 1-919-677-8166
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore
Part 1

About SAS System Options

Chapter 1
What You Need to Know about SAS System Options ............... 3
Chapter 1
What You Need to Know about SAS System Options

Definition of System Options

System options are instructions that affect the processing of an entire SAS program or interactive SAS session from the time the option is specified until it is changed. Examples of items that are controlled by SAS system options include the appearance of SAS output, the handling of some files that are used by SAS, the use of system variables, the processing of observations in SAS data sets, features of SAS initialization, and the way SAS interacts with your host operating environment.
Syntax

Specifying System Options in an OPTIONS Statement

The syntax for specifying system options in an OPTIONS statement is

\[ \text{OPTIONS option(s);} \]

where

option specifies one or more SAS system options that you want to change.

The following example shows how to use the system options NODATE and LINESIZE= in an OPTIONS statement:

\[ \text{options nodate linesize=72;} \]

Specifying System Options on the Command Line or in a Configuration File

Operating Environment Information

On the command line or in a configuration file, the syntax is specific to your operating environment. For details, see the SAS documentation for your operating environment.

Specifying Hexadecimal Values

Hexadecimal values for system options must begin with a number (0-9), followed by an X. For example, the following OPTIONS statement sets the line size to 160 using a hexadecimal number:

\[ \text{options linesize=0a0x;} \]

Using SAS System Options

Default Settings

SAS system options are initialized with default settings when SAS is invoked. However, the default settings for some SAS system options vary both by operating environment and by site. Your on-site SAS support personnel might have customized configuration files in order to provide a global set of default values that are specific for your site.

Information about creating customized configuration files is provided in the configuration guide for SAS software for your operating environment.

For more information, see “Resetting System Options to the Default or Starting Value” on page 16.
Saving and Loading SAS System Options

SAS system options can be saved to either the SAS registry or a SAS data set by using the OPTSAVE procedure or by using the DMOPTSAVE command in the SAS windowing environment. Some system options cannot be saved. You can specify DEFINE in the OPTIONS procedure to determine whether an option can be saved. In the log output, the line that begins with Optsave: indicates whether the option can be saved.

```sas
proc options option=pageno define;
run;
```

SAS (r) Proprietary Software Release 9.3  TS1B0
PAGENO=1
Option Definition Information for SAS Option PAGENO
Group= LISTCONTROL
Group Description: Procedure output and display settings
Description: Beginning page number for the next page of output produced by the SAS System
Type: The option value is of type LONG
  Range of Values: The minimum is 1 and the maximum is 2147483647
  Valid Syntax(any casing): MIN|MAX|n|nK|nM|nG|nT|hexadecimal
Numeric Format: Usage of LOGNUMBERFORMAT does not impact the value format
When Can Set: Startup or anytime during the SAS Session
Restricted: Your Site Administrator can restrict modification of this option
Optsave: PROC Optsave or command Dmoptsave will save this option

For a list of all options that can be saved, submit the OPTIONS procedure and the PRINT procedure:

```sas
proc optsave;
run;
proc print;
  var optname;
run;
```

For information about saving options, see Chapter 6, “OPTSAVE Procedure,” on page 327.

To load a set of saved system options you use either the OPTLOAD procedure or the DMOPTLOAD command. For information about loading system options, see Chapter 5, “OPTLOAD Procedure,” on page 321.

For information about the DMOPTSAVE command and the DMOPTLOAD command, see the SAS Help and Documentation.

Determining Which Settings Are in Effect

To determine which settings are in effect for SAS system options, use one of the following:

OPLIST system option
  Writes to the SAS log the system options that were specified on the SAS invocation command line. (See the SAS documentation for your operating environment for more information.)
VERBOSE system option
  Writes to the SAS log the system options that were specified in the configuration file
  and on the SAS invocation command line.

SAS System Options window
  Lists all system option settings.

OPTIONS procedure
  Writes system option settings to the SAS log. To display the settings of system
  options with a specific functionality, such as error handling, use the GROUP= option:
  
  proc options GROUP=errorhandling; run;

  For more information, see Chapter 4, “OPTIONS Procedure,” on page 301.

GETOPTION function
  Returns the value of a specified system option.

VOPTION Dictionary table
  Located in the Sashelp library, VOPTION contains a list of all current system option
  settings, a description of each option, the option type, whether the option is a
  portable or a host option, when the option can be set, and the group to which the
  option belongs. You can view this table with SAS Explorer, print the table by using
  the PRINT procedure, or you can extract information from the VOPTION table by
  using the SQL procedure.

dictionary.options SQL table
  Accessed with the SQL procedure, this table lists the system options that are in
  effect.

Restricted Options

Restricted options are system options whose values are determined by the site
administrator and cannot be overridden. The site administrator can create a restricted
options table that specifies the option values that are restricted when SAS starts. Any
attempt to modify a system option that is listed in the restricted options table results in a
message to the SAS log indicating that the system option has been restricted by the site
administrator and cannot be updated.

To determine which system options are restricted by your site administrator, use the
RESTRICT option of the OPTIONS procedure. The RESTRICT option displays the
option's value, scope, and setting. In the following example, the SAS log shows that only
one option, CMPOPT, is restricted:

  proc options restrict;
  run;

Log 1.1  Restricted Option Information

1           proc options restrict;
2          run;
SAS (r) Proprietary Software Release xxx  TS1B0
Option Value Information For SAS Option CMPOPT
Value: (NOPRECISE NOEXTRAMATH NOMISSCHECK NOGUARDCHECK NOGENSYMNAMES
NOFUNCDIFFERENCING)
Scope: SAS Session
How option value set: Site Administrator Restricted
The OPTIONS procedure displays this information for all options that are restricted. If your site administrator has not restricted any options, then the following message appears in the SAS log:

Your site administrator has not restricted any options.

You can use the OPTIONS procedure option LISTRESTRICT to view the options that your site administrator can restrict. These are not options that are restricted, but can be restricted.

proc options listrestrictable;
run;

Log 1.2  Partial Listing of Options That Can Be Restricted

<table>
<thead>
<tr>
<th>Portable Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLETLOC</td>
</tr>
<tr>
<td>ARMAGENT</td>
</tr>
<tr>
<td>ARMLOC</td>
</tr>
<tr>
<td>ARMSUBSYS</td>
</tr>
<tr>
<td>AUTOCORRECT</td>
</tr>
<tr>
<td>AUTOSAVELOC</td>
</tr>
<tr>
<td>AUTOSIGNON</td>
</tr>
<tr>
<td>BINDING</td>
</tr>
<tr>
<td>BUFNO</td>
</tr>
<tr>
<td>BUFSIZE</td>
</tr>
<tr>
<td>BYERR</td>
</tr>
<tr>
<td>BYLINE</td>
</tr>
<tr>
<td>BYSORTED</td>
</tr>
<tr>
<td>CAPS</td>
</tr>
<tr>
<td>CARDIMAGE</td>
</tr>
<tr>
<td>CATCACHE</td>
</tr>
<tr>
<td>CBUFFNO</td>
</tr>
<tr>
<td>CENTER</td>
</tr>
<tr>
<td>CGOPTIMIZE</td>
</tr>
<tr>
<td>CHARCODE</td>
</tr>
<tr>
<td>CLEANUP</td>
</tr>
<tr>
<td>CMDMAC</td>
</tr>
<tr>
<td>CMPLIB</td>
</tr>
</tbody>
</table>

For information, see Chapter 4, “OPTIONS Procedure,” on page 301.

The following table lists the system options that cannot be restricted:
### Table 1.1  System Options That Cannot Be Restricted

<table>
<thead>
<tr>
<th>Option</th>
<th>All Operating Environments</th>
<th>OpenVMS</th>
<th>UNIX</th>
<th>Windows</th>
<th>z/OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTLOG</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTPRINT</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>APPEND</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASYNCHIO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>AUTOEXEC</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOMFILE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOTTOMMARGIN</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMDEF</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>CONFIG</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPUCOUNT</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATESTYLE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBCS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>DFLANG</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLDMGACTION</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DMR</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DMS</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DMSEXP</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DMSPGMLINESIZE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENGINE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXPLORER</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILELOCKWAITMAX</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>INITCMD</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INITSTMT</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INSERT</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JREOPTIONS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Option</td>
<td>All Operating Environments</td>
<td>OpenVMS</td>
<td>UNIX</td>
<td>Windows</td>
<td>z/OS</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------</td>
<td>--------</td>
<td>------</td>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>LEFTMARGIN</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINESIZE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>LAST</em></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOG</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOGAPPLNAME</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOGPARM</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEMCACHE</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEMLIB</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>METAPASS</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>METAPROTOCOL</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>METAREPOSITORY</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>METASERVER</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>METAUSER</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSYMTABMAX</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVARSIZE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OBJECTSERVER</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORIENTATION</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OVP</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGESIZE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAPERSIZE</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>PDFPASSWORD</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRINT</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRINTERPATH</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESOURCESLOC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>RIGHTMARGIN</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Determining How a SAS System Option Value Was Set

To determine how a system option value was set, use either the OPTIONS procedure or the GETOPTION function:

- Use the OPTIONS procedure with the VALUE option specified in the OPTIONS statement. The VALUE option displays the specified option's value and scope.

- Use the GETOPTION function as an argument to the %SYSFUNC macro function:

  %put %sysfunc(getoption(option-name, howset));

This example shows how the option value for the system option CENTER was set using the OPTIONS procedure:

```plaintext
proc options option=center value;
run;
```
The following partial SAS log shows that the option value for CENTER was the shipped default.

**Log 1.3  Option Value Information for the System Option CENTER**

```
2   proc options option=center value;
3   run;
```

Option Value Information for SAS Option CENTER
Option Value: CENTER
Option Scope: Default
How option value set: Shipped Default

If a SAS option is set from a configuration file, SAS displays the name of the configuration file that set the option.

**Log 1.4  Option Value Information Showing an Option Set by a Configuration File**

```
7   proc options option=work value;
8   run;
```

SAS (r) Proprietary Software Release xxx  TS1B0

Option Value Information For SAS Option WORK
Value: C:\DOCUME~1\sasuser1\LOCALS~1\Temp\SAS Temporary Files\_TD5428_t20111_
Scope: SAS Session
How option value set: Config File
Config file name: C:\SASv9\SASv9.cfg

If a SAS option is modified using the INSERT or APPEND system options, you can use the VALUE option in a PROC OPTIONS statement to show that the value was inserted or appended:

**Log 1.5  Option Value Information for an Option Modified by the INSERT and APPEND Options**

```
24  options insert=(fmtsearch="c:/myformats");
25  options append=(fmtsearch="c:/mysas");
26  proc options option=fmtsearch value;
27  run;
```

SAS (r) Proprietary Software Release xxx  TS1B0

Option Value Information For SAS Option FMTSEARCH
Value: ('C:/MYFORMATS' WORK LIBRARY 'C:/MYSAS')
Scope: DMS Process
How option value set: Options Statement
Value Inserted: 'C:/MYFORMATS'

How option value set: Options Window
Value: WORK LIBRARY

How option value set: Options Statement
Value Appended: 'C:/MYSAS'
If no value is assigned to a character system option, then SAS assigns the option a value of " " (a space between two single quotation marks) and Option Value displays a blank space.

**Obtaining Descriptive Information about a System Option**

You can quickly obtain basic descriptive information about a system option by specifying the DEFINE option in the PROC OPTIONS statement.

The DEFINE option writes the following descriptive information about a system option to the SAS log:

- the value of the option
- a description of the option
- the name and description of each system option group that the option is a part of
- type information, such as whether it is numeric or character, whether to expand a value that is an environment variable, and valid values for the option
- when in the SAS session it can be set
- if it can be restricted by the system administrator
- if the OPTSAVE procedure or the DMOPTSAVE command will save the option

For example, the following statements write a message to the SAS log that contains descriptive information about the system option ERRORCHECK:

```sas
proc options option=errorcheck define;
run;
```

**Log 1.6  Descriptive Information about the System Option ERRORCHECK**

```
5    proc options option=errorcheck define;
6    run;

SAS (r) Proprietary Software Release xxx  TS1B0

ERRORCHECK=NORMAL
Option Definition Information for SAS Option ERRORCHECK
  Group= ERRORHANDLING
  Group Description: Error messages and error conditions settings
  Description: Level of special error processing to be performed
  Type: The option value is of type CHARACTER
    Maximum Number of Characters: 10
    Casing: The option value is retained uppercased
    Quotes: If present during "set", start and end quotes are removed
    Parentheses: The option value does not require enclosure within parentheses. If present, the parentheses are
    retained.
    Expansion: Environment variables, within the option value, are not expanded
    Number of valid values: 2
    Valid value: NORMAL
    Valid value: STRICT
  When Can Set: Startup or anytime during the SAS Session
  Restricted: Your Site Administrator can restrict modification of this option
  Optsave: PROC Optsave or command Dmoptsave will save this option
```
Changing SAS System Option Settings

SAS provides default settings for SAS system options. You can override the default settings of any unrestricted system option in several ways, depending on the function of the system option:

- On the command line or in a configuration file:
  
  specify any unrestricted SAS system option setting either on the SAS command line or in a configuration file. If you use the same option settings frequently, then it is usually more convenient to specify the options in a configuration file, rather than on the command line. Either method sets your SAS system options during SAS invocation. Many SAS system option settings can be specified only during SAS invocation. Descriptions of the individual options provide details.

- In an OPTIONS statement:
  
  You can specify an OPTIONS statement at any time during a session except within data lines or parmcard lines. Settings remain in effect throughout the current program or process unless you reset them with another OPTIONS statement, change them in the SAS System Options window, or use the OPTLOAD procedure to load previously saved options from a data set. You can also place an OPTIONS statement in an autoexec file.

  By specifying either the INSERT or APPEND option in the OPTIONS statement, you can add a value to certain system options that name libraries or files, such as the AUTOEXEC option and the FMTSEARCH option. For more information, see “Changing an Option Value by Using the INSERT and APPEND System Options” on page 13.

- In the OPTLOAD procedure or the DMOPTLOAD command:
  
  You can use the OPTLOAD procedure or the DMOPTLOAD command to read option settings that were specified with the OPTSAVE procedure and saved to a SAS data set.

- In a SAS System Options window:
  
  If you are in a windowing environment, type options in the toolbar or on the command line to open the SAS System Options window. The SAS System Options window lists the names of the SAS system option groups. You can then expand the groups to see the option names and to change their current settings to a new value or to the default value. Alternatively, you can use the Find Option command in the Options pop-up menu to go directly to an option. Changes take effect immediately and remain in effect throughout the session unless you reset them with an OPTIONS statement or change them in the SAS System Options window.

SAS system options can be restricted by a site administrator so that after they are set by the administrator, they cannot be changed by a user. Depending on your operating environment, system options can be restricted globally, by group, or by user. You can use the OPTIONS procedure to determine which options are restricted. For more information, see Chapter 4, “OPTIONS Procedure,” on page 301 and the SAS documentation for your operating environment. For more information about how to restrict options, see your site administrator.

Changing an Option Value by Using the INSERT and APPEND System Options

You can use the INSERT and APPEND options to modify the values of these options:
<table>
<thead>
<tr>
<th>Option</th>
<th>Where the Option Can Be Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOEXEC</td>
<td>configuration file, SAS invocation</td>
</tr>
<tr>
<td>CMPLIB</td>
<td>configuration file, SAS invocation, OPTIONS statement, SAS System Options window</td>
</tr>
<tr>
<td>FMTSEARCH</td>
<td>OPTIONS statement, SAS System Options window</td>
</tr>
<tr>
<td>HELPLOC</td>
<td>configuration file, SAS invocation</td>
</tr>
<tr>
<td>MAPS</td>
<td>configuration file, SAS invocation, OPTIONS statement, SAS System Options window</td>
</tr>
<tr>
<td>MSG</td>
<td>configuration file, SAS invocation</td>
</tr>
<tr>
<td>SASAUTOS</td>
<td>configuration file, SAS invocation, OPTIONS statement, SAS System Options window</td>
</tr>
<tr>
<td>SASHELP</td>
<td>configuration file, SAS invocation</td>
</tr>
<tr>
<td>SASSCRIPT</td>
<td>configuration file, SAS invocation, OPTIONS statement, SAS System Options window</td>
</tr>
<tr>
<td>SET</td>
<td>configuration file, SAS invocation</td>
</tr>
</tbody>
</table>

The value of these options is one or more libraries, files, or environment variables. You use the INSERT option to insert a value before the current value. You use the APPEND option to append a value to the end of the current value. Use the LISTINSERTAPPEND option in the PROC OPTIONS statement to see a list of options in the SAS log that you can use with the INSERT option and the APPEND option:

```
1   proc options listinsertappend;
2   run;
```

Core options that can utilize INSERT and APPEND

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOEXEC</td>
<td>Identifies AUTOEXEC files used during initialization</td>
</tr>
<tr>
<td>CMPLIB</td>
<td>Identify previously compiled libraries of CMP subroutines to use when linking</td>
</tr>
<tr>
<td>FMTSEARCH</td>
<td>List of catalogs to search for formats and informatns</td>
</tr>
<tr>
<td>MAPS</td>
<td>Location of maps for use with SAS/GRAPH</td>
</tr>
<tr>
<td>SASAUTOS</td>
<td>Search list for autocall macros</td>
</tr>
<tr>
<td>SASHELP</td>
<td>Location of the SASHELP library</td>
</tr>
<tr>
<td>SASSCRIPT</td>
<td>Location of SAS/CONNECT script files</td>
</tr>
</tbody>
</table>

Host options that can utilize INSERT and APPEND

SAS (r) Proprietary Software Release xxx TS1B0
The syntax for the INSERT option and the APPEND option is different when you start SAS as compared to using the OPTIONS statement after SAS starts. For the correct syntax to use when SAS starts, see the documentation for your operating environment:

• *SAS Companion for UNIX Environments*
• *SAS Companion for Windows*
• *SAS Companion for z/OS*

If you specify the INSERT option or the APPEND option after SAS starts, the syntax requires parentheses and is the same for all operating environments:

```
insert=(system-option-1=argument-1 system-option-n=argument-n)
append=(system-option-1=argument-1 system-option-n=argument-n)
```

The syntax for `system-option=argument` is the syntax that is required for the specified system option.

Here are two examples:

```sas
options insert=(fmtsearch="c:/myformats");
oxoptions append=(fmtsearch=("c:/mysasfmt" "u:/mysasfmt2"));
```

For more information, see “INSERT= System Option” on page 159 and “APPEND= System Option” on page 68.

You can use the VALUE option in the PROC OPTIONS statement to display values that have been inserted or appended to an option value:

```
26   proc options option=fmtsearch value;
27   run;
```

SAS inserts or appends values as they are specified in the INSERT and APPEND system options. SAS does not check for duplicate values.

The INSERT and APPEND system options only add values to a system option’s value. To delete a value from a system option, set the option to the value that you want.
Resetting System Options to the Default or Starting Value

Reset Options to the Default Value by Using the SAS System Options Window
To reset a system option to the default option by using the SAS System Options window:
1. From the SAS menu bar, select Tools ⇒ Options ⇒ System.
2. Right-click Options and select Find Option.
3. Type the option name and click OK.
4. Right-click the option name and select Set to Default.

Reset Options to the Default or Starting Value by Using the %Put Macro and the GETOPTIONS Function
You can use the SAS System Options window or macro processing and the GETOPTION function together to set a system option to the default value or to the value that was specified when SAS started.

You use the GETOPTION function DEFAULTVALUE option to set a system option its default value. You use the GETOPTION function STARTUPVALUE option to set a system option to the starting value.

The following code is an example of setting the PAPERSIZE= system option to its default value and to its starting value:

/* Check the value of papersize before we change it. */
/* The initial value is A4 as this value was used when SAS started. */
/* SAS started. */

%put %sysfunc(getoption(papersize,keyword));

/* Change the PAPERSIZE value and check the change. */
options papersize="600x800 Pixels";
%put %sysfunc(getoption(papersize,keyword));

/* Change PAPERSIZE back to the default value and check it. */
/* RESULT: LETTER */
%let defsize = %sysfunc(getoption(papersize,keyword,defaultvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));

/* Change the value to the startup value and check it. */
/* RESULT: A4 */
%let defsize = %sysfunc(getoption(papersize,keyword,startupvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));

The SAS log displays the following lines:
For more information, see GETOPTION function on page 23.

**How Long System Option Settings Are in Effect**

When you specify a SAS system option setting, the setting applies to the next step and to all subsequent steps for the duration of the SAS session, or until you reset the system option setting, as shown:

```sas
data one;
  set items;
run;

  /* option applies to all subsequent steps */
  options obs=5;

  /* printing ends with the fifth observation */
  proc print data=one;
  run;

  /* the SET statement stops reading
     after the fifth observation */
  data two;
    set items;
run;
```

To read more than five observations, you must reset the OBS= system option. For more information, see “OBS= System Option” on page 183.
Order of Precedence

If the same system option appears in more than one place, the order of precedence from highest to lowest is the following:
1. restricted options table, if it exists
2. OPTIONS statement and SAS System Options window
3. autoexec file (that contains an OPTIONS statement)
4. command-line specification
5. configuration file specification
6. SAS system default settings.

Operating Environment Information
In some operating environments, you can specify system options in other places. See the SAS documentation for your operating environment.

The following table shows the order of precedence that SAS uses for execution mode options. These options are a subset of the SAS invocation options and are specified on the command line during SAS invocation.

Table 1.2 Order of Precedence for SAS Execution Mode Options

<table>
<thead>
<tr>
<th>Execution Mode Option</th>
<th>Precedence</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECTSERVER</td>
<td>Highest</td>
</tr>
<tr>
<td>DMR</td>
<td>2nd</td>
</tr>
<tr>
<td>SYSIN</td>
<td>3rd</td>
</tr>
<tr>
<td>INITCMD</td>
<td>4th</td>
</tr>
<tr>
<td>DMS</td>
<td>4th</td>
</tr>
<tr>
<td>DMSEXP</td>
<td>4th</td>
</tr>
<tr>
<td>EXPLORER</td>
<td>4th</td>
</tr>
<tr>
<td>none (default is interactive line mode under UNIX and interactive full screen mode under z/OS)</td>
<td>5th</td>
</tr>
</tbody>
</table>

The order of precedence of SAS execution mode options consists of the following rules:
• SAS uses the execution mode option with the highest precedence.
• If you specify more than one execution mode option of equal precedence, SAS uses only the last option listed.

See the descriptions of the individual options for more details.
**Interaction with Data Set Options**

Many system options and data set options share the same name and have the same function. System options remain in effect for all DATA and PROC steps in a SAS job or session until their settings are changed. A data set option, however, overrides a system option only for the particular data set in the step in which it appears.

In this example, the OBS= system option in the OPTIONS statement specifies that only the first 100 observations will be read from any data set within the SAS job. The OBS= data set option in the SET statement, however, overrides the system option and specifies that only the first five observations will be read from data set TWO. The PROC PRINT step uses the system option setting and reads and prints the first 100 observations from data set THREE:

```sas
options obs=100;

data one;
  set two(obs=5);
run;

proc print data=three;
run;
```

---

**Comparisons**

Note the differences between system options, data set options, and statement options.

- **System options** remain in effect for all DATA and PROC steps in a SAS job or current process unless they are respecified.

- **Data set options** apply to the processing of the SAS data set with which they appear. Some data set options have corresponding system options or LIBNAME statement options. For an individual data set, you can use the data set option to override the setting of these other options.

- **Statement options** control the action of the statement in which they appear. Options in global statements, such as in the LIBNAME statement, can have a broader impact.
Part 2

SAS Functions for SAS System Options

Chapter 2
Dictionary of Functions for System Options .................. 23
Dictionary of Functions for System Options

GETOPTION Function

Returns the value of a SAS system or graphics option.

**Category:** Special

**Syntax**

```
GETOPTION(option-name, return-value-option, return-value-formatting-options)
```

**Required Argument**

- **option-name**
  - is a character constant, variable, or expression that specifies the name of the system option.
  
  **Tips:**
  - Do not put an equal sign after the name. For example, write PAGESIZE= as PAGESIZE.
  - SAS options that are passwords, such as EMAILPW and METAPASS, return the value `xxxxxxxx`, and not the actual password.

**Return Value Options**

- **DEFAULTVALUE**
  - returns the default option value.

  **Restriction:** DEFAULTVALUE is valid only for SAS system options. SAS issues a warning message when the DEFAULTVALUE option is specified and option-name is a graphics option.

- **HOWSCOPE**
  - returns a character string that specifies the scope of an option.
Restriction: HOWSCOPE is valid only for SAS system options. SAS issues a warning message when the HOWSCOPE option is specified and option-name is a graphics option.

HOWSET returns a character string that specifies how an option value was set.

Restriction: HOWSET is valid only for SAS system options. SAS issues a warning message when the HOWSET option is specified and option-name is a graphics option.

STARTUPVALUE returns the system option value that was used to start SAS either on the command line or in a configuration file.

Restriction: STARTUPVALUE is valid only for SAS system options. SAS issues a warning message when the STARTUPVALUE option is specified and option-name is a graphics option.

Return Value Formatting Options

CM
reports graphic units of measure in centimeters.

Restriction: CM is valid only for graphics options and the following SAS system options: BOTTOMMARGIN, TOPMARGIN, RIGHTMARGIN, and LEFTMARGIN. SAS writes a note to the log when the CM option is specified and option-name is not a graphics option or an option that specifies a margin value.

EXPAND
for options that contain environment variables, returns the option value with the value of the environment variable.

Restrictions:
Variable expansion is valid only in the Windows and UNIX operating environments.

EXPAND is valid only for character system option values. EXPAND is ignored if option-name has an option type of Boolean, such as CENTER or NOCENTER, or if the value of the option is numeric.

Note: SAS issues a note when EXPAND is specified for Boolean options and for options that have numeric values. SAS issues a warning when EXPAND is specified and the option is a graphics option.

Tip: By default, some option values are displayed with expanded variable values. Other options require the EXPAND option in the PROC OPTIONS statement. Use the DEFINE option in the PROC OPTIONS statement to determine whether an option value expands variables by default or if the EXPAND option is required. If the output from PROC OPTIONS DEFINE shows the following information, you must use the EXPAND option to expand variable values:

Expansion: Environment variables, within the option value, are not expanded

KEYEXPAND
for options that contain environment variables, returns the value in the format option-name=value.

Restriction: KEYEXPAND is valid only for character system option values. SAS issues an error message when the KEYEXPAND option is specified and option-name is a graphics option. KEYEXPAND is ignored if option-name has an option type of Boolean, such as CENTER or NOCENTER, or if the value of the option is numeric.
KEYWORD
returns option values in a option-name=value format that would be suitable for direct use in the SAS OPTIONS or GOPTIONS global statements.

Restrictions:
- KEYWORD is not valid when it is used with the HEXVALUE, EXPAND, KEYEXPAND, or LOGNUMBERFORMAT options. SAS writes a note to the log when the GETOPTION function contains conflicting options.
- KEYWORD is valid only for character or numeric system option values.
- KEYWORD is ignored for system options whose option type is Boolean, such as CENTER or NOCENTER. SAS issues an error message when the KEYWORD option is specified and option-name is a graphics option.

Note: For a system option with a null value, the GETOPTION function returns a value of ' ' (single quotation marks with a blank space between them). An example is EMAILID=' '.

HEXVALUE
returns the option value as a hexadecimal value.
Restriction: HEXVALUE is valid only for character or numeric system option values. If HEXVALUE is specified for system options whose option type is Boolean, such as CENTER or NOCENTER, or if option-name is a graphics option, SAS issues an error message.

IN
reports graphic units of measure in inches.
Restriction: IN is valid only for graphics options and the following SAS system options: BOTTOMMARGIN, TOPMARGIN, RIGHTMARGIN, and LEFTMARGIN. SAS writes a note to the log when the IN option is specified and option-name is not a graphics option or an option that specifies a margin value.

LOGNUMBERFORMAT
formats SAS system option values using locale-specific punctuation.
Restriction: Do not use LOGNUMBERFORMAT if the returned value is used to set an option value by using the OPTIONS statement. The OPTIONS statement does not accept commas in numeric values.

Examples

Example 1: Using GETOPTION to Save and Restore the YEARCUTOFF Option
This example saves the value of the YEARCUTOFF option, processes SAS statements based on the value of the YEARCUTOFF option, and then resets the value to 1920 if it is not already 1920.

/* Save the value of the YEARCUTOFF system option */
%let cutoff=%sysfunc(getoption(yearcutoff,keyword));

data ages;
  if getoption('yearcutoff') = '1920' then
do;
    ...more SAS statements...
  end;
else do;
  ...more SAS statements...
/* Reset YEARCUTOFF */
Example 2: Using GETOPTION to Obtain Different Reporting Options

This example defines a macro to illustrate the use of the GETOPTION function to obtain the value of system and graphics options by using different reporting options.

```sas
%macro showopts;
  %put MAPS= %sysfunc(
    getoption(MAPS));
  %put MAPSEXPA NDED= %sysfunc(
    getoption(MAPS, EXPAND));
  %put PAGESIZE= %sysfunc(
    getoption(PAGESIZE));
  %put PAGESIZESETBY= %sysfunc(
    getoption(PAGESIZE, HOWSET));
  %put PAGESIZESCOPE= %sysfunc(
    getoption(PAGESIZE, HOWSCOPE));
  %put PS= %sysfunc(
    getoption(PS));
  %put LS= %sysfunc(
    getoption(LS));
  %put PS(keyword form)= %sysfunc(
    getoption(PS,keyword));
  %put LS(keyword form)= %sysfunc(
    getoption(LS,keyword));
  %put FORMCHAR= %sysfunc(
    getoption(FORMCHAR));
  %put HSIZE= %sysfunc(
    getoption(HSIZE));
  %put VSIZE= %sysfunc(
    getoption(VSIZE));
  %put HSIZE(in/keyword form)= %sysfunc(
    getoption(HSIZE,in,keyword));
  %put HSIZE(cm/keyword form)= %sysfunc(
    getoption(HSIZE,cm,keyword));
  %put VSIZE(in/keyword form)= %sysfunc(
    getoption(VSIZE,in,keyword));
  %put VSIZE(cm/keyword form)= %sysfunc(
    getoption(VSIZE,cm,keyword));
%mend;

goptions VSIZE=8.5 in HSIZE=11 in;
options PAGESIZE=67;
%showopts
```

The following is the SAS log:
NOTE: PROCEDURE PRINTTO used (Total process time):
real time           0.00 seconds
cpu time            0.00 seconds

6 %macro showopts;
7 %put MAPS= %sysfunc(getoption(MAPS));
8 %put MAPSEXPANDED= %sysfunc(getoption(MAPS, EXPAND));
9 %put PAGESIZE= %sysfunc(getoption(PAGESIZE));
10 %put PAGESIZESETBY= %sysfunc(getoption(PAGESIZE, HOWSET));
11 %put PAGESIZESCOPE= %sysfunc(getoption(PAGESIZE, HOWSCOPE));
12 %put PS= %sysfunc(getoption(PS));
13 %put LS= %sysfunc(getoption(LS));
14 %put PS(keyword form)= %sysfunc(getoption(PS,keyword));
15 %put LS(keyword form)= %sysfunc(getoption(LS,keyword));
16 %put FORMCHAR= %sysfunc(getoption(FORMCHAR));
17 %put HSIZE= %sysfunc(getoption(HSIZE));
18 %put VSIZE= %sysfunc(getoption(VSIZE));
19 %put HSIZE(in/keyword form)= %sysfunc(getoption(HSIZE,in,keyword));
20 %put HSIZE(cm/keyword form)= %sysfunc(getoption(HSIZE,cm,keyword));
21 %put VSIZE(in/keyword form)= %sysfunc(getoption(VSIZE,in,keyword));
22 %put VSIZE(cm/keyword form)= %sysfunc(getoption(VSIZE,cm,keyword));
%end;

23 goptions VSIZE=8.5 in HSIZE=11 in;
24 options PAGESIZE=67;
25 %showopts
MAPS= ("C:sasroot\maps-path\en\maps")
MAPSEXPANDED= ("C:\maps-path\en\maps")
PAGESIZE= 67
PAGESIZESETBY= Options Statement
PAGESIZESCOPE= Line Mode Process
PS= 67
LS= 78
PS(keyword form)= PS=67
LS(keyword form)= LS=78
FORMCHAR= f_\f^t"\&\#x004d\&\#x007c\&\#x00a0\=|\<!\>*
HSIZE= 11.0000 in
VSIZE= 8.5000 in
HSIZE(in/keyword form)= HSIZE=11.0000 in
HSIZE(cm/keyword form)= HSIZE=27.9400 cm
VSIZE(in/keyword form)= VSIZE=8.5000 in
VSIZE(cm/keyword form)= VSIZE=21.5900 cm

Example 3: Returning Default and Start-up Values
This example changes the value of the PAPERSIZE system option to a specific value, the PAPERSIZE option default value, and to the value that was assigned to the PAPERSIZE option when SAS started.
/* Check the value of papersize before we change it. */
/* The initial value is A4 as this value was used when */
/* SAS started. */

%put %sysfunc(getoption(papersize,keyword));

/* Change the PAPERSIZE value and check the change. */

options papersize="600x800 Pixels";

%put %sysfunc(getoption(papersize,keyword));

/* Change PAPERSIZE back to the default value and check it. */
/* RESULT: LETTER */

%let defsize = %sysfunc(getoption(papersize,keyword,defaultvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));

/* Change the value to the startup value and check it. */
/* RESULT: A4 */

%let defsize = %sysfunc(getoption(papersize,keyword,startupvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));

The SAS log displays the following lines:

```
%put %sysfunc(getoption(papersize,keyword));
PAPERSIZE=A4

options papersize="600x800 Pixels";
%put %sysfunc(getoption(papersize,keyword));
PAPERSIZE=600X800 Pixels

%let defsize = %sysfunc(getoption(papersize,keyword,defaultvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));
PAPERSIZE=LETTER

%let defsize = %sysfunc(getoption(papersize,keyword,startupvalue)) ;
options &defsize; run;
%put %sysfunc(getoption(papersize,keyword));
PAPERSIZE=A4
```

Note: The default settings for the PAGESIZE= and the LINESIZE= options depend on the mode that you use to run SAS.
Part 3

SAS System Options

Chapter 3
Dictionary of System Options ........................................ 31
Chapter 3
Dictionary of System Options

SAS System Options Documented in Other SAS Publications .......................... 35
SAS System Options by Category .............................................................. 35

Dictionary ................................................................. 68
  APPEND= System Option .................................................. 68
  APPLETLOC= System Option ........................................... 70
  AUTHPROVIDERDOMAIN System Option ............................... 70
  AUTOCORRECT System Option ....................................... 72
  AUTOSAVELOC= System Option ..................................... 74
  BINDING= System Option ............................................. 75
  BOTTOMMARGIN= System Option .................................. 76
  BUFNO= System Option ............................................... 77
  BUFSIZE= System Option ............................................. 78
  BYERR System Option ................................................ 80
  BYLINE System Option ............................................... 81
  BYSORTED System Option .......................................... 82
  CAPS System Option .................................................. 83
  CARDIMAGE System Option ......................................... 84
  CATCACHE= System Option .......................................... 85
  CBUFNO= System Option ............................................. 86
  CENTER System Option ............................................... 87
  CGOPTIMIZE= System Option ..................................... 87
  CHARCODE System Option ........................................... 88
  CHKPTCLEAN System Option ....................................... 89
  CLEANUP System Option ............................................. 91
  CMPLIB= System Option ............................................ 93
  CMPMODEL= System Option ......................................... 94
  CMPOPT= System Option ............................................. 95
  COLLATE System Option ............................................. 97
  COLORPRINTING System Option .................................. 98
  COMPRESS= System Option ........................................ 99
  COPIES= System Option ............................................... 101
  CPUCOUNT= System Option ......................................... 101
  CPUID System Option ................................................ 103
  DATASTMTCHK= System Option .................................. 103
  DATE System Option ................................................ 104
  DATESTYLE= System Option ....................................... 105
  DEFLATION= System Option ....................................... 106
  DETAILS System Option ............................................ 107
  DEVICE= System Option ............................................. 108
  DKRICOND= System Option ......................................... 108
<table>
<thead>
<tr>
<th>System Option</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>DKOCOND=</td>
<td>109</td>
</tr>
<tr>
<td>DLCREATEDIR</td>
<td>110</td>
</tr>
<tr>
<td>DLDMGACTION=</td>
<td>111</td>
</tr>
<tr>
<td>DMR System Option</td>
<td>112</td>
</tr>
<tr>
<td>DMS System Option</td>
<td>113</td>
</tr>
<tr>
<td>DMSEXp System Option</td>
<td>114</td>
</tr>
<tr>
<td>DMSLOGSIZE=</td>
<td>115</td>
</tr>
<tr>
<td>DMSOUTSIZE=</td>
<td>116</td>
</tr>
<tr>
<td>DMPGMlinesize=</td>
<td>117</td>
</tr>
<tr>
<td>DMSynchk System Option</td>
<td>117</td>
</tr>
<tr>
<td>DNSFERR System Option</td>
<td>118</td>
</tr>
<tr>
<td>DTRESET System Option</td>
<td>119</td>
</tr>
<tr>
<td>DUILP System Option</td>
<td>120</td>
</tr>
<tr>
<td>ECHOAUTO System Option</td>
<td>121</td>
</tr>
<tr>
<td>EMAILAUTHPROTOCOL= System Option</td>
<td>122</td>
</tr>
<tr>
<td>EMAILFROM System Option</td>
<td>123</td>
</tr>
<tr>
<td>EMAILHOST=</td>
<td>123</td>
</tr>
<tr>
<td>EMAILID=</td>
<td>124</td>
</tr>
<tr>
<td>EMAILPORT System Option</td>
<td>126</td>
</tr>
<tr>
<td>EMAILPW=</td>
<td>127</td>
</tr>
<tr>
<td>EMAILOFFSET=</td>
<td>128</td>
</tr>
<tr>
<td>EMAILHOST=</td>
<td>123</td>
</tr>
<tr>
<td>ENGINE=</td>
<td>129</td>
</tr>
<tr>
<td>ERRORABEND System Option</td>
<td>130</td>
</tr>
<tr>
<td>ERRORBYABEND System Option</td>
<td>131</td>
</tr>
<tr>
<td>ERRORCHECK=</td>
<td>131</td>
</tr>
<tr>
<td>ERRORS=</td>
<td>132</td>
</tr>
<tr>
<td>EVENTDS=</td>
<td>133</td>
</tr>
<tr>
<td>EXPLORER System Option</td>
<td>135</td>
</tr>
<tr>
<td>FILESYNC=</td>
<td>136</td>
</tr>
<tr>
<td>FIRSTOBS=</td>
<td>137</td>
</tr>
<tr>
<td>FMTERR System Option</td>
<td>138</td>
</tr>
<tr>
<td>FMTSEARCH=</td>
<td>139</td>
</tr>
<tr>
<td>FONTEMBEDDING System Option</td>
<td>142</td>
</tr>
<tr>
<td>FONTRENDERING=</td>
<td>143</td>
</tr>
<tr>
<td>FONTSLOC=</td>
<td>145</td>
</tr>
<tr>
<td>FORMCHAR=</td>
<td>145</td>
</tr>
<tr>
<td>FORMDLIM=</td>
<td>146</td>
</tr>
<tr>
<td>FORMS=</td>
<td>147</td>
</tr>
<tr>
<td>GSTYLE System Option</td>
<td>148</td>
</tr>
<tr>
<td>GWINDOW System Option</td>
<td>148</td>
</tr>
<tr>
<td>HELPBROWSER=</td>
<td>149</td>
</tr>
<tr>
<td>HELPENCMD System Option</td>
<td>150</td>
</tr>
<tr>
<td>HELPHOST System Option</td>
<td>151</td>
</tr>
<tr>
<td>HELPPORT=</td>
<td>152</td>
</tr>
<tr>
<td>HTTPSERVERPORTMAX= System Option</td>
<td>152</td>
</tr>
<tr>
<td>HTTPSERVERPORTMIN= System Option</td>
<td>153</td>
</tr>
<tr>
<td>IBUFNO=</td>
<td>154</td>
</tr>
<tr>
<td>IBUFSIZE=</td>
<td>155</td>
</tr>
<tr>
<td>INITCMD System Option</td>
<td>156</td>
</tr>
<tr>
<td>INITSTMT=</td>
<td>158</td>
</tr>
<tr>
<td>INSERT=</td>
<td>159</td>
</tr>
<tr>
<td>INTERVALS=</td>
<td>160</td>
</tr>
<tr>
<td>INVALIDDATA=</td>
<td>161</td>
</tr>
<tr>
<td>JPEQUALITY=</td>
<td>162</td>
</tr>
<tr>
<td>LABEL System Option</td>
<td>163</td>
</tr>
<tr>
<td>LABELCHKPT System Option</td>
<td>164</td>
</tr>
</tbody>
</table>
LABELCHKPTLIB= System Option .................................................. 165
LABELRESTART System Option .................................................. 167
_LAST_ = System Option ......................................................... 168
LEFTMARGIN= System Option ..................................................... 169
LINESIZE= System Option ....................................................... 170
LOGPARN= System Option ....................................................... 171
LRECL= System Option ........................................................... 176
MAPS= System Option ........................................................... 177
MERGENOBY System Option .................................................... 177
MISSING= System Option ........................................................ 178
MSGLEVEL= System Option ...................................................... 179
MULTENVAPPL System Option .................................................. 180
NEWS= System Option .......................................................... 180
NOTES System Option ........................................................... 181
OBS= System Option ............................................................ 183
ORIENTATION= System Option ................................................ 191
OVP System Option ............................................................... 194
PAGEBREAKINITIAL System Option ........................................... 194
PAGENO= System Option ........................................................ 195
PAGESIZE= System Option ...................................................... 196
PAPERDEST= System Option ..................................................... 197
PAPERSIZE= System Option ..................................................... 198
PAPERSOURCE= System Option ................................................ 199
PAPERTYPE= System Option .................................................... 200
PARAM= System Option .......................................................... 201
PARMCARDS= System Option ................................................... 201
PDFACCESS System Option .................................................... 202
PDFASSEMBLY System Option ................................................. 203
PDFCOMMENT System Option ................................................ 204
PDFCONTENT System Option .................................................. 205
PDFCOPY System Option ........................................................ 206
PDFFILLIN System Option ...................................................... 207
PDFPAGELAYOOUT= System Option ........................................... 209
PDFPAGEVIEW= System Option ................................................ 210
PDFPASSWORD= System Option ................................................ 211
PDFPRINT= System Option ..................................................... 212
PDFSECURITY= System Option ................................................ 213
PRIMARYPROVIDERDOMAIN= System Option ................................. 215
PRINTERPATH= System Option ................................................. 216
PRINTINIT System Option ...................................................... 218
PRINTMSGLIST System Option ................................................ 219
QUOTELENMAX System Option ................................................ 220
REPLACE System Option ........................................................ 220
REUSE= System Option .......................................................... 221
RIGHTMARGIN= System Option ................................................ 222
RLANG System Option ........................................................... 223
RSASUSER System Option ....................................................... 224
S= System Option ............................................................... 225
S2= System Option ............................................................. 227
S2V= System Option ............................................................. 230
SASHELP= System Option ...................................................... 231
SASUSER= System Option ........................................................ 232
SEQ= System Option ............................................................. 233
SETINIT System Option .......................................................... 234
SKIP= System Option ............................................................ 234
SAS System Options Documented in Other SAS Publications

Some system options are documented with related subject matter in other SAS publications:

- Encryption in SAS
- Grid Computing in SAS on support.sas.com
- SAS Interface to Application Response Measurement (ARM): Reference
- SAS Companion for Windows
- SAS Companion for UNIX Environments
- SAS Companion for z/OS
- SAS Data Quality Server: Reference
- SAS Intelligence Platform: Application Server Administration Guide on support.sas.com
- SAS Language Interfaces to Metadata
- SAS Logging: Configuration and Programming Reference
- SAS Macro Language: Reference
- SAS Output Delivery System User's Guide
- SAS Scalable Performance Data Engine: Reference
- SAS SQL Procedure User's Guide
- SAS VSAM Processing for z/OS
- SAS/ACCESS for Relational Databases: Reference
- SAS/CONNECT User's Guide
- SAS/GRAPH: Reference
- SAS/SHARE User's Guide

SAS System Options by Category

The categories for SAS system options correspond to the SAS system option groups and sub-groups:

- Communications: E-mail options associated with sending and receiving e-mail using SAS
- Communications: Networking and encryption options related to remote communication, shared settings, and encryption
- Communications: Metadata options to configure SAS to use metadata
Environment control: Display options to set SAS windows and display preferences
Environment control: Error handling options associated with error conditions and error messages
Environment control: Files options to set SAS library and file location preferences
Environment control: Help options used to configure the SAS Help
Environment control: Initialization and operation options that establish the SAS operating environment
Environment control: Language control options to set language and translation preferences
Files: External files options that define how to process files that are not created by SAS
Files: SAS files options that define how to process SAS files
Input control: Data processing options for data entry and data processing preferences
Input control: Data quality options to configure the SAS Data Quality server
Graphics: Driver settings options that define devices, graphics, and map preferences
Log and procedure output control: SAS log options that control the display of messages that are written to the SAS log
Log and procedure output control: Procedure output options that define procedure output and display preferences
Log and procedure output control: SAS log and procedure output options that control both SAS log and procedure output preferences
Log and procedure output control: ODS printing options to define preferences for printing to ODS destinations
Log and procedure output control: PDF options to define preferences for PDF files
Log and procedure output control: SVG options to define preferences for SVG files
Log and procedure output control: Animation options to define preferences for animated SVG files
Macro: SAS macro options that define SAS macro preferences
Sort: Procedure options options that define preferences for sorting SAS files
System administration: Installation options for defining site installation settings
System administration: Memory options that define computer memory preferences
System administration: Performance options that define performance preferences
### System administration: Code generation
- Options that define preferences for generating SAS language statements

### System administration: Security
- Options for defining security settings

### System administration: SQL
- Options that define settings for the SQL procedure

<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communications: E-mail</td>
<td>EMAILUTCOFFSET= System Option</td>
<td>For e-mail that is sent using the FILENAME statement EMAIL (SMTP) access method, specifies a UTC offset that is used in the Date header field of the e-mail message.</td>
</tr>
<tr>
<td></td>
<td>EMAILAUTHPROTOCOL= System Option</td>
<td>Specifies the authentication protocol for SMTP e-mail.</td>
</tr>
<tr>
<td></td>
<td>EMAILFROM System Option</td>
<td>When sending e-mail by using SMTP, specifies whether the e-mail option FROM is required in either the FILE or FILENAME statement.</td>
</tr>
<tr>
<td></td>
<td>EMAILHOST= System Option</td>
<td>Specifies one or more SMTP servers that support e-mail access.</td>
</tr>
<tr>
<td></td>
<td>EMAILID= System Option</td>
<td>Identifies an e-mail sender by specifying either a logon ID, an e-mail profile, or an e-mail address.</td>
</tr>
<tr>
<td></td>
<td>EMAILPORT System Option</td>
<td>Specifies the port that the SMTP server is attached to.</td>
</tr>
<tr>
<td></td>
<td>EMAILPW= System Option</td>
<td>Specifies an e-mail logon password.</td>
</tr>
<tr>
<td>Communications: Networking and encryption</td>
<td>HTTPSERVERPORTMAX= System Option</td>
<td>Specifies the highest port number that can be used by the SAS HTTP server for remote browsing.</td>
</tr>
<tr>
<td></td>
<td>HTTPSERVERPORTMIN= System Option</td>
<td>Specifies the lowest port number that can be used by the SAS HTTP server for remote browsing.</td>
</tr>
<tr>
<td>Environment control: Display</td>
<td>AUTOSAVELOC= System Option</td>
<td>Specifies the location of the Program Editor autosave file.</td>
</tr>
<tr>
<td></td>
<td>CHARCODE System Option</td>
<td>Specifies whether specific keyboard combinations are substituted for special characters that are not on the keyboard.</td>
</tr>
<tr>
<td></td>
<td>DMSLOGSIZE= System Option</td>
<td>Specifies the maximum number of rows that the SAS Log window can display.</td>
</tr>
<tr>
<td></td>
<td>DMSOUTSIZE= System Option</td>
<td>Specifies the maximum number of rows that the SAS Output window can display.</td>
</tr>
<tr>
<td></td>
<td>DMSPGMLINESIZE= System Option</td>
<td>Specifies the maximum number of characters in a Program Editor line.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Category</strong></td>
<td><strong>Language Elements</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td></td>
<td>FONTSL OC= System Option</td>
<td>Specifies the location of the fonts that are supplied by SAS; names the default font file location for registering fonts that use the FONTREG procedure.</td>
</tr>
<tr>
<td></td>
<td>FORMS= System Option</td>
<td>If forms are used for printing, specifies the default form to use.</td>
</tr>
<tr>
<td></td>
<td>SOLUTIONS System Option</td>
<td>Specifies whether the Solutions menu is included in SAS windows.</td>
</tr>
<tr>
<td></td>
<td>TOOLSMENU System Option</td>
<td>Specifies whether the Tools menu is included in SAS windows.</td>
</tr>
<tr>
<td></td>
<td>VIEWMENU System Option</td>
<td>Specifies whether the View menu is included in SAS windows.</td>
</tr>
<tr>
<td><strong>Environment control: Error handling</strong></td>
<td><strong>AUTOCORRECT System Option</strong></td>
<td>Specifies whether SAS attempts to automatically correct misspelled procedure names, misspelled procedure keywords, or misspelled global statement names.</td>
</tr>
<tr>
<td></td>
<td>BYERR System Option</td>
<td>Specifies whether SAS produces errors when the SORT procedure attempts to process a <em>NULL</em> data set.</td>
</tr>
<tr>
<td></td>
<td>CHKPTCLEAN System Option</td>
<td>When SAS is in checkpoint mode or restart mode, specifies whether to erase the contents of the Work library after a batch program executes successfully.</td>
</tr>
<tr>
<td></td>
<td>CLEANUP System Option</td>
<td>For an out-of-resource condition, specifies whether to perform an automatic cleanup or a user-specified cleanup.</td>
</tr>
<tr>
<td></td>
<td>DKRICOND= System Option</td>
<td>Specifies the level of error detection to report when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.</td>
</tr>
<tr>
<td></td>
<td>DKROCOND= System Option</td>
<td>Specifies the level of error detection to report when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.</td>
</tr>
<tr>
<td></td>
<td>DMSSYNCHK System Option</td>
<td>In the SAS windowing environment, specifies whether to enable syntax check mode for DATA step and PROC step processing.</td>
</tr>
<tr>
<td></td>
<td>DSNFERR System Option</td>
<td>When a SAS data set cannot be found, specifies whether SAS issues an error message.</td>
</tr>
<tr>
<td></td>
<td>ERRORABEND System Option</td>
<td>Specifies whether SAS responds to errors by terminating.</td>
</tr>
<tr>
<td></td>
<td>ERRORBYABEND System Option</td>
<td>Specifies whether SAS ends a program when an error occurs in BY-group processing.</td>
</tr>
<tr>
<td></td>
<td>ERRORCHECK= System Option</td>
<td>Specifies whether SAS enters syntax-check mode when errors are found in the LIBNAME, FILENAME, %INCLUDE, and LOCK statements.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Language Elements</td>
<td>ERRORS= System Option (p. 132)</td>
<td>Specifies the maximum number of observations for which SAS issues complete error messages.</td>
</tr>
<tr>
<td></td>
<td>FMTERR System Option (p. 138)</td>
<td>When a variable format cannot be found, specifies whether SAS generates an error or continues processing.</td>
</tr>
<tr>
<td></td>
<td>LABELCHKPT System Option (p. 164)</td>
<td>Specifies whether checkpoint-restart data for labeled code sections is to be recorded for batch programs.</td>
</tr>
<tr>
<td></td>
<td>LABELCHKPTLIB= System Option (p. 165)</td>
<td>Specifies the libref of the library where the checkpoint-restart data is saved for labeled code sections.</td>
</tr>
<tr>
<td></td>
<td>LABELRESTART System Option (p. 167)</td>
<td>Specifies whether to execute a batch program by using checkpoint-restart data for data collected at labeled code sections.</td>
</tr>
<tr>
<td></td>
<td>QUOTELENMAX System Option (p. 220)</td>
<td>If a quoted string exceeds the maximum length allowed, specifies whether SAS writes a warning message to the SAS log.</td>
</tr>
<tr>
<td></td>
<td>STEPCHECKPT System Option (p. 243)</td>
<td>Specifies whether checkpoint-restart data for DATA and PROC steps is to be recorded for a batch program.</td>
</tr>
<tr>
<td></td>
<td>STEPCHECKPTLIB= System Option (p. 244)</td>
<td>Specifies the libref of the library where checkpoint-restart data for DATA and PROC steps is saved.</td>
</tr>
<tr>
<td></td>
<td>STEPRESTART System Option (p. 246)</td>
<td>Specifies whether to execute a batch program by using checkpoint-restart data for DATA and PROC steps.</td>
</tr>
<tr>
<td></td>
<td>SYNTAXCHECK System Option (p. 261)</td>
<td>In non-interactive or batch SAS sessions, specifies whether to enable syntax check mode for multiple steps.</td>
</tr>
<tr>
<td></td>
<td>VNFERR System Option (p. 289)</td>
<td>Specifies whether SAS issues an error or warning when a BY variable exists in one data set but not another data set when the other data set is <em>NULL</em>. This option applies when processing the SET, MERGE, UPDATE, or MODIFY statements.</td>
</tr>
<tr>
<td>Environment control: Files</td>
<td>APPEND= System Option (p. 68)</td>
<td>Appends a value to the existing value of the specified system option.</td>
</tr>
<tr>
<td></td>
<td>APPELETLOC= System Option (p. 70)</td>
<td>Specifies the location of Java applets.</td>
</tr>
<tr>
<td></td>
<td>FMTSEARCH= System Option (p. 139)</td>
<td>Specifies the order in which format catalogs are searched.</td>
</tr>
<tr>
<td></td>
<td>INSERT= System Option (p. 159)</td>
<td>Inserts the specified value as the first value of the specified system option.</td>
</tr>
<tr>
<td></td>
<td>NEWS= System Option (p. 180)</td>
<td>Specifies an external file that contains messages to be written to the SAS log, immediately after the header.</td>
</tr>
<tr>
<td></td>
<td>PARM= System Option (p. 201)</td>
<td>Specifies a parameter string that is passed to an external program.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>PARMCARDS= System Option (p. 201)</td>
<td>Specifies the file reference to open when SAS encounters the PARMCARDS statement in a procedure.</td>
</tr>
<tr>
<td></td>
<td>RSASUSER System Option (p. 224)</td>
<td>Specifies whether to open the Sasuser library for Read access or Read-Write access.</td>
</tr>
<tr>
<td></td>
<td>SASHELP= System Option (p. 231)</td>
<td>Specifies the location of the Sashelp library.</td>
</tr>
<tr>
<td></td>
<td>SASUSER= System Option (p. 232)</td>
<td>Specifies the SAS library to use as the Sasuser library.</td>
</tr>
<tr>
<td></td>
<td>TRAINLOC= System Option (p. 270)</td>
<td>Specifies the URL for SAS online training courses.</td>
</tr>
<tr>
<td></td>
<td>USER= System Option (p. 273)</td>
<td>Specifies the default permanent SAS library.</td>
</tr>
<tr>
<td></td>
<td>UUIDCOUNT= System Option (p. 275)</td>
<td>Specifies the number of UUIDs to acquire from the UUID Generator Daemon.</td>
</tr>
<tr>
<td></td>
<td>UUIDGENDHOST= System Option (p. 276)</td>
<td>Identifies the host and port or the LDAP URL that the UUID Generator Daemon runs on.</td>
</tr>
<tr>
<td></td>
<td>WORK= System Option (p. 294)</td>
<td>Specifies the Work library.</td>
</tr>
<tr>
<td></td>
<td>WORKINIT System Option (p. 295)</td>
<td>Specifies whether to initialize the Work library at SAS invocation.</td>
</tr>
<tr>
<td></td>
<td>WORKTERM System Option (p. 296)</td>
<td>Specifies whether to erase the Work files when SAS terminates.</td>
</tr>
<tr>
<td>Environment control: Help</td>
<td>HELPBROWSER= System Option (p. 149)</td>
<td>Specifies the browser to use for SAS Help and ODS output.</td>
</tr>
<tr>
<td></td>
<td>HELPENCMD System Option (p. 150)</td>
<td>Specifies whether SAS uses the English version or the translated version of the keyword list for the command-line Help.</td>
</tr>
<tr>
<td></td>
<td>HELPHOST System Option (p. 151)</td>
<td>Specifies the name of the computer where the remote browser is to send Help and ODS output.</td>
</tr>
<tr>
<td></td>
<td>HELPPORT= System Option (p. 152)</td>
<td>Specifies the port number for the remote browser client.</td>
</tr>
<tr>
<td>Environment control: Initialization and operation</td>
<td>AUTHPROVIDERDOMAIN System Option (p. 70)</td>
<td>Associates a domain suffix with an authentication provider.</td>
</tr>
<tr>
<td></td>
<td>DMR System Option (p. 112)</td>
<td>Specifies whether to enable SAS to invoke a server session for use with a SAS/CONNECT client.</td>
</tr>
<tr>
<td></td>
<td>DMS System Option (p. 113)</td>
<td>Specifies whether to invoke the SAS windowing environment and display the Log, Editor, and Output windows.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>DMSEXP System Option (p. 114)</td>
<td>Specifies whether to invoke the SAS windowing environment and display the Explorer, Editor, Log, Output, and Results windows.</td>
<td></td>
</tr>
<tr>
<td>EXPLORER System Option (p. 135)</td>
<td>Specifies whether to invoke the SAS windowing environment and display only the Explorer and Program Editor windows.</td>
<td></td>
</tr>
<tr>
<td>INITCMD System Option (p. 156)</td>
<td>Specifies an application invocation command and optional SAS windowing environment or text editor commands that SAS executes before processing the AUTOEXEC= file during SAS invocation.</td>
<td></td>
</tr>
<tr>
<td>INITSTMT= System Option (p. 158)</td>
<td>Specifies a SAS statement to execute after any statements in the AUTOEXEC= file and before any statements from the SYSIN= file.</td>
<td></td>
</tr>
<tr>
<td>MULTENVAPPL System Option (p. 180)</td>
<td>Specifies whether the fonts that are available in a SAS application font selector window lists only the SAS fonts that are available in all operating environments.</td>
<td></td>
</tr>
<tr>
<td>PRIMARYPROVIDERDOMAIN= System Option (p. 215)</td>
<td>Specifies the domain name of the primary authentication provider.</td>
<td></td>
</tr>
<tr>
<td>TERMINAL System Option (p. 265)</td>
<td>Specifies whether to associate a terminal with a SAS session.</td>
<td></td>
</tr>
<tr>
<td>TERMSTMT= System Option (p. 266)</td>
<td>Specifies the SAS statements to execute when SAS terminates.</td>
<td></td>
</tr>
<tr>
<td>DATESTYLE= System Option (p. 105)</td>
<td>Specifies the sequence of month, day, and year when ANYDTDTE, ANYDTDTM, or ANYDTTME informat data is ambiguous.</td>
<td></td>
</tr>
<tr>
<td>PAPERSIZE= System Option (p. 198)</td>
<td>Specifies the paper size to use for printing.</td>
<td></td>
</tr>
<tr>
<td>URLENCODING= System Option (p. 272)</td>
<td>Specifies whether the argument to the URLENCODE function and to the URLDECODE function is interpreted using the SAS session encoding or UTF-8 encoding.</td>
<td></td>
</tr>
<tr>
<td>LRECL= System Option (p. 176)</td>
<td>Specifies the default logical record length to use for reading and writing external files.</td>
<td></td>
</tr>
<tr>
<td>STARTLIB System Option (p. 242)</td>
<td>Specifies whether SAS assigns user-defined permanent librefs when SAS starts.</td>
<td></td>
</tr>
<tr>
<td>BUFNO= System Option (p. 77)</td>
<td>Specifies the number of buffers to be allocated for processing SAS data sets.</td>
<td></td>
</tr>
<tr>
<td>BUFSIZE= System Option (p. 78)</td>
<td>Specifies the permanent buffer page size for output SAS data sets.</td>
<td></td>
</tr>
<tr>
<td>CATCACHE= System Option (p. 85)</td>
<td>Specifies the number of SAS catalogs to keep open in cache memory.</td>
<td></td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>CBUFNO= System Option (p. 86)</td>
<td>Specifies the number of extra page buffers to allocate for each open SAS catalog.</td>
</tr>
<tr>
<td></td>
<td>CMPLIB= System Option (p. 93)</td>
<td>Specifies one or more SAS data sets that contain compiler subroutines to include during program compilation.</td>
</tr>
<tr>
<td></td>
<td>COMPRESS= System Option (p. 99)</td>
<td>Specifies the type of compression of observations to use for output SAS data sets.</td>
</tr>
<tr>
<td></td>
<td>DATASTMTCHK= System Option (p. 103)</td>
<td>Specifies which SAS statement keywords are prohibited from being specified as a one-level DATA step name to protect against overwriting an input data set.</td>
</tr>
<tr>
<td></td>
<td>DKRICOND= System Option (p. 108)</td>
<td>Specifies the level of error detection to report when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.</td>
</tr>
<tr>
<td></td>
<td>DKROCOND= System Option (p. 109)</td>
<td>Specifies the level of error detection to report when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.</td>
</tr>
<tr>
<td></td>
<td>DLCREATEDIR System Option (p. 110)</td>
<td>Specifies to create a directory for the SAS library that is named in a LIBNAME statement if the directory does not exist.</td>
</tr>
<tr>
<td></td>
<td>DLDMGACTION= System Option (p. 111)</td>
<td>Specifies the type of action to take when a SAS data set or a SAS catalog is detected as damaged.</td>
</tr>
<tr>
<td></td>
<td>ENGINE= System Option (p. 129)</td>
<td>Specifies the default access method for SAS libraries.</td>
</tr>
<tr>
<td></td>
<td>FILESYNC= System Option (p. 136)</td>
<td>Specifies when operating system buffers that contain contents of permanent SAS files are written to disk.</td>
</tr>
<tr>
<td></td>
<td>FIRSTOBS= System Option (p. 137)</td>
<td>Specifies the observation number or external file record that SAS processes first.</td>
</tr>
<tr>
<td></td>
<td>IBUFNO= System Option (p. 154)</td>
<td>Specifies an optional number of extra buffers to be allocated for navigating an index file.</td>
</tr>
<tr>
<td></td>
<td>IBUFSIZE= System Option (p. 155)</td>
<td>Specifies the buffer page size for an index file.</td>
</tr>
<tr>
<td></td>
<td><em>LAST</em>= System Option (p. 168)</td>
<td>Specifies the most recently created data set.</td>
</tr>
<tr>
<td></td>
<td>MERGENOBY System Option (p. 177)</td>
<td>Specifies the type of message that is issued when MERGE processing occurs without an associated BY statement.</td>
</tr>
<tr>
<td></td>
<td>OBS= System Option (p. 183)</td>
<td>Specifies the observation that is used to determine the last observation to process, or specifies the last record to process.</td>
</tr>
<tr>
<td></td>
<td>REPLACE System Option (p. 220)</td>
<td>Specifies whether permanently stored SAS data sets can be replaced.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>REUSE= System Option (p. 221)</td>
<td>Specifies whether SAS reuses space when observations are added to a compressed SAS data set.</td>
<td></td>
</tr>
<tr>
<td>UTILLOC= System Option (p. 274)</td>
<td>Specifies one or more file system locations in which enabled threaded applications can store utility files.</td>
<td></td>
</tr>
<tr>
<td>V6CREATEUPDATE= System Option (p. 278)</td>
<td>Specifies the type of message to write to the SAS log when Version 6 data sets are created or updated.</td>
<td></td>
</tr>
<tr>
<td>VALIDFMTNAME= System Option (p. 278)</td>
<td>Specifies the maximum size (32 characters or 8 characters) that user-created format and informat names can be before an error or warning is issued.</td>
<td></td>
</tr>
<tr>
<td>VALIDMEMNAME= System Option (p. 280)</td>
<td>Specifies the rules for naming SAS data sets, SAS data views, and item stores.</td>
<td></td>
</tr>
<tr>
<td>VALIDVARNAME= System Option (p. 282)</td>
<td>Specifies the rules for valid SAS variable names that can be created and processed during a SAS session.</td>
<td></td>
</tr>
<tr>
<td>VARLENCHK= System Option (p. 284)</td>
<td>Specifies the type of message to write to the SAS log when the input data set is read using the SET, MERGE, UPDATE, or MODIFY statements.</td>
<td></td>
</tr>
<tr>
<td>DEVICE= System Option (p. 108)</td>
<td>Specifies the device driver to which SAS/GRAPH sends procedure output.</td>
<td></td>
</tr>
<tr>
<td>GSTYLE System Option (p. 148)</td>
<td>Specifies whether ODS styles can be used to generate graphs that are stored as GRSEG catalog entries.</td>
<td></td>
</tr>
<tr>
<td>GWINDOW System Option (p. 148)</td>
<td>Specifies whether SAS displays SAS/GRAPH output in the GRAPH window.</td>
<td></td>
</tr>
<tr>
<td>MAPS= System Option (p. 177)</td>
<td>Specifies the location of the SAS library that contains SAS/GRAPH map data sets.</td>
<td></td>
</tr>
<tr>
<td>BYSORTED System Option (p. 82)</td>
<td>Specifies whether observations in one or more data sets are sorted in alphabetic or numeric order or are grouped in another logical order.</td>
<td></td>
</tr>
<tr>
<td>CAPS System Option (p. 83)</td>
<td>Specifies whether to convert certain types of input to uppercase.</td>
<td></td>
</tr>
<tr>
<td>CARDIMAGE System Option (p. 84)</td>
<td>Specifies whether SAS processes source and data lines as 80-byte cards.</td>
<td></td>
</tr>
<tr>
<td>DATESTYLE= System Option (p. 105)</td>
<td>Specifies the sequence of month, day, and year when ANYYTDTE, ANYYTDTM, or ANYYDTTME informat data is ambiguous.</td>
<td></td>
</tr>
<tr>
<td>EVENTDS= System Option (p. 133)</td>
<td>Specifies whether SAS reuses space when observations are added to a compressed SAS data set.</td>
<td></td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

Input control: Data processing  

INTERVALDS= System Option (p. 160)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS System Options by Category</td>
<td>r v a l n a m e - v a l u e p a i r s, w h e r e t h e v a l u e i s a S A S d a t a s e t t h a t c o n t a i n s u</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

The intervals can be used as arguments to the supplied intervals.
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IN TXN and INT CK functions.</td>
<td>INVALIDDATA = System Option (p. 161)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPECIFIES THE VALUE THAT SAS ASSIGN TO A DATA VALUE</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

S= System Option (p. 225)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SAS System Options by Category
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
</table>

S2= System Option (p. 227)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
</table>

For statements on each line of a source statement from a %INCLUDE statement...
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
</table>

S2V= System Option (p. 230)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SAS System Options by Category

53
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SEQ=</td>
<td>System Option (p. 233)</td>
<td>Specifies the length of the numeric portion of the record format.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

SPOOL System Option (p. 241)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
</table>

SAS System Options by Category
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VBUFSIZE=</td>
<td>System Option (p. 287)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specifying the size of the view buffer.</td>
</tr>
<tr>
<td></td>
<td>YEARCUTOFF=</td>
<td>System Option (p. 297)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specifies the first year of a 100-year span that is used by date informats</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and functions to read a two-digit year.</td>
</tr>
<tr>
<td></td>
<td>BINDING=</td>
<td>System Option (p. 75)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specifies the binding of the binding.</td>
</tr>
</tbody>
</table>

Log and procedure output control:
ODS Printing
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>BOTTOMMARGIN= System Option (p. 76)</strong></td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

COLLATE System Option (p. 97)

Specifies whether to collate multiple copies.
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
</table>

COLORPRINTING System Option (p. 98)
<table>
<thead>
<tr>
<th>Category</th>
<th>Language Elements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>COPIES</strong> = System Option (p. 101)</td>
<td>Specifies the number of copies to print.</td>
</tr>
<tr>
<td></td>
<td><strong>DEFLATION</strong> = System Option (p. 106)</td>
<td>Specifies the level of compression for device drivers that support the Deflate compression algorithm.</td>
</tr>
<tr>
<td></td>
<td><strong>DUPLEX</strong> System Option (p. 120)</td>
<td>Specifies whether duplex (two-sided) printing is enabled.</td>
</tr>
<tr>
<td></td>
<td><strong>FONTEMBEDDING</strong> System Option (p. 142)</td>
<td>Specifies whether font embedding is enabled in Universal Printer and SAS/GRAPH printing.</td>
</tr>
<tr>
<td></td>
<td><strong>FONTRENDERING</strong> = System Option (p. 143)</td>
<td>Specifies whether SAS/GRAPH devices that are based on the SASGDGIF, SASGDTIF, and SASGDIMG modules render fonts by using the operating system or by using the FreeType engine.</td>
</tr>
<tr>
<td></td>
<td><strong>GSTYLE</strong> System Option (p. 148)</td>
<td>Specifies whether ODS styles can be used to generate graphs that are stored as GRSEG catalog entries.</td>
</tr>
<tr>
<td></td>
<td><strong>JPEGQUALITY</strong> = System Option (p. 162)</td>
<td>Specifies the JPEG quality factor that determines the ratio of image quality to the level of compression for JPEG files produced by the SAS/GRAPH JPEG device driver.</td>
</tr>
<tr>
<td></td>
<td><strong>LEFTMARGIN</strong> = System Option (p. 169)</td>
<td>Specifies the print margin for the left side of the page.</td>
</tr>
<tr>
<td></td>
<td><strong>ORIENTATION</strong> = System Option (p. 191)</td>
<td>Specifies the paper orientation to use when printing to a printer.</td>
</tr>
<tr>
<td></td>
<td><strong>PAPERDEST</strong> = System Option (p. 197)</td>
<td>Specifies the name of the output bin to receive printed output.</td>
</tr>
<tr>
<td></td>
<td><strong>PAPERSIZE</strong> = System Option (p. 198)</td>
<td>Specifies the paper size to use for printing.</td>
</tr>
<tr>
<td></td>
<td><strong>PAPERSOURCE</strong> = System Option (p. 199)</td>
<td>Specifies the name of the paper bin to use for printing.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PAPERTYPE= System Option (p. 200)</td>
<td><strong>PAPERTYPE= System Option</strong> (p. 200)</td>
<td>Specifies the type of paper to use for printing.</td>
</tr>
<tr>
<td>PRINTERPATH= System Option (p. 217)</td>
<td><strong>PRINTERPATH= System Option</strong> (p. 217)</td>
<td>Specifies the name of a registered printer to use for Universal Printing.</td>
</tr>
<tr>
<td>RIGHTMARGIN= System Option (p. 222)</td>
<td><strong>RIGHTMARGIN= System Option</strong> (p. 222)</td>
<td>Specifies the print margin for the right side of the page.</td>
</tr>
<tr>
<td>TEXTURELOC= System Option (p. 267)</td>
<td><strong>TEXTURELOC= System Option</strong> (p. 267)</td>
<td>Specifies the location of textures and images that are used by ODS styles.</td>
</tr>
<tr>
<td>TOPMARGIN= System Option (p. 269)</td>
<td><strong>TOPMARGIN= System Option</strong> (p. 269)</td>
<td>Specifies the print margin at the top of the page.</td>
</tr>
<tr>
<td>UPRINTCOMPRESSSION System Option (p. 271)</td>
<td><strong>UPRINTCOMPRESSSION System Option</strong> (p. 271)</td>
<td>Specifies whether to enable the compression of files created by some Universal Printer and SAS/GRAPH devices.</td>
</tr>
<tr>
<td>Log and procedure output control: PDF</td>
<td><strong>PDFACCESS System Option</strong> (p. 202)</td>
<td>Specifies whether text and graphics from PDF documents can be read by screen readers for the visually impaired.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFASSEMBLY System Option</strong> (p. 203)</td>
<td>Specifies whether PDF documents can be assembled.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFCOMMENT System Option</strong> (p. 204)</td>
<td>Specifies whether PDF document comments can be modified.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFCONTENT System Option</strong> (p. 205)</td>
<td>Specifies whether the contents of a PDF document can be changed.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFCOPY System Option</strong> (p. 206)</td>
<td>Specifies whether text and graphics from a PDF document can be copied.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFFILLIN System Option</strong> (p. 207)</td>
<td>Specifies whether PDF forms can be filled in.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFPAGELAYOUT= System Option</strong> (p. 209)</td>
<td>Specifies the page layout for PDF documents.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFPASSWORD= System Option</strong> (p. 211)</td>
<td>Specifies the password to use to open a PDF document and the password used by a PDF document owner.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFPRINT= System Option</strong> (p. 212)</td>
<td>Specifies the resolution to print PDF documents.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>PDFSECURITY= System Option</strong> (p. 213)</td>
<td>Specifies the level of encryption for PDF documents.</td>
</tr>
<tr>
<td>Log and procedure output control: Procedure output</td>
<td><strong>BYLINE System Option</strong> (p. 81)</td>
<td>Specifies whether to print BY lines above each BY group.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CENTER System Option</td>
<td>(p. 87)</td>
<td>Specifies whether to center or left-align SAS procedure output.</td>
</tr>
<tr>
<td>DATE System Option</td>
<td>(p. 104)</td>
<td>Specifies whether to print the date and time that a SAS program started.</td>
</tr>
<tr>
<td>DETAILS System Option</td>
<td>(p. 107)</td>
<td>Specifies whether to include additional information when files are listed in a SAS library.</td>
</tr>
<tr>
<td>DMSOUTSIZE= System Option</td>
<td>(p. 116)</td>
<td>Specifies the maximum number of rows that the SAS Output window can display.</td>
</tr>
<tr>
<td>DTRESET System Option</td>
<td>(p. 119)</td>
<td>Specifies whether to update the date and time in the SAS log and in the procedure output file.</td>
</tr>
<tr>
<td>FORMCHAR= System Option</td>
<td>(p. 145)</td>
<td>Specifies the default output formatting characters.</td>
</tr>
<tr>
<td>FORMDLIM= System Option</td>
<td>(p. 146)</td>
<td>Specifies a character to delimit page breaks in SAS output for the LISTING destination.</td>
</tr>
<tr>
<td>FORMS= System Option</td>
<td>(p. 147)</td>
<td>If forms are used for printing, specifies the default form to use.</td>
</tr>
<tr>
<td>LABEL System Option</td>
<td>(p. 163)</td>
<td>Specifies whether SAS procedures can use labels with variables.</td>
</tr>
<tr>
<td>LINESIZE= System Option</td>
<td>(p. 170)</td>
<td>Specifies the line size for the SAS log and for SAS procedure output.</td>
</tr>
<tr>
<td>MISSING= System Option</td>
<td>(p. 178)</td>
<td>Specifies the character to print for missing numeric values.</td>
</tr>
<tr>
<td>NUMBER System Option</td>
<td>(p. 182)</td>
<td>Specifies whether to print the page number in the title line of each page of SAS output.</td>
</tr>
<tr>
<td>PAGEBREAKINITIAL System Option</td>
<td>(p. 194)</td>
<td>Specifies whether to begin the SAS log and procedure output files for the LISTING destination on a new page.</td>
</tr>
<tr>
<td>PAGENO= System Option</td>
<td>(p. 195)</td>
<td>Resets the SAS output page number.</td>
</tr>
<tr>
<td>PAGESIZE= System Option</td>
<td>(p. 196)</td>
<td>Specifies the number of lines that compose a page of the SAS log and SAS output.</td>
</tr>
<tr>
<td>PRINTINIT System Option</td>
<td>(p. 218)</td>
<td>Specifies whether to initialize the SAS procedure output file for the LISTING destination.</td>
</tr>
<tr>
<td>SKIP= System Option</td>
<td>(p. 234)</td>
<td>Specifies the number of lines to skip at the top of each page of SAS output for the LISTING destination.</td>
</tr>
<tr>
<td>SYSPRINTFONT= System Option</td>
<td>(p. 263)</td>
<td>Specifies the default font to use for printing, which can be overridden by explicitly specifying a font and an ODS style.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Log and procedure output control: SAS log</td>
<td>CPUID System Option (p. 103)</td>
<td>Specifies whether the CPU identification number is written to the SAS log.</td>
</tr>
<tr>
<td></td>
<td>DATE System Option (p. 104)</td>
<td>Specifies whether to print the date and time that a SAS program started.</td>
</tr>
<tr>
<td></td>
<td>DETAILS System Option (p. 107)</td>
<td>Specifies whether to include additional information when files are listed in a SAS library.</td>
</tr>
<tr>
<td></td>
<td>DMSLOGSIZE= System Option (p. 115)</td>
<td>Specifies the maximum number of rows that the SAS Log window can display.</td>
</tr>
<tr>
<td></td>
<td>DTRESET System Option (p. 119)</td>
<td>Specifies whether to update the date and time in the SAS log and in the procedure output file.</td>
</tr>
<tr>
<td></td>
<td>ECHOAUTO System Option (p. 121)</td>
<td>Specifies whether the statements in the AUTOEXEC= file are written to the SAS log as they are executed.</td>
</tr>
<tr>
<td></td>
<td>ERRORS= System Option (p. 132)</td>
<td>Specifies the maximum number of observations for which SAS issues complete error messages.</td>
</tr>
<tr>
<td></td>
<td>LINESIZE= System Option (p. 170)</td>
<td>Specifies the line size for the SAS log and for SAS procedure output.</td>
</tr>
<tr>
<td></td>
<td>LOGPARM= System Option (p. 171)</td>
<td>Specifies when SAS log files are opened, closed, and, in conjunction with the LOG= system option, how they are named.</td>
</tr>
<tr>
<td></td>
<td>MISSING= System Option (p. 178)</td>
<td>Specifies the character to print for missing numeric values.</td>
</tr>
<tr>
<td></td>
<td>MSGLEVEL= System Option (p. 179)</td>
<td>Specifies the level of detail in messages that are written to the SAS log.</td>
</tr>
<tr>
<td></td>
<td>NEWS= System Option (p. 180)</td>
<td>Specifies an external file that contains messages to be written to the SAS log, immediately after the header.</td>
</tr>
<tr>
<td></td>
<td>NOTES System Option (p. 181)</td>
<td>Specifies whether notes are written to the SAS log.</td>
</tr>
<tr>
<td></td>
<td>NUMBER System Option (p. 182)</td>
<td>Specifies whether to print the page number in the title line of each page of SAS output.</td>
</tr>
<tr>
<td></td>
<td>OVP System Option (p. 194)</td>
<td>Specifies whether overprinting of error messages to make them bold, is enabled.</td>
</tr>
<tr>
<td></td>
<td>PAGEBREAKINITIAL System Option (p. 194)</td>
<td>Specifies whether to begin the SAS log and procedure output files for the LISTING destination on a new page.</td>
</tr>
<tr>
<td></td>
<td>PAGESIZE= System Option (p. 196)</td>
<td>Specifies the number of lines that compose a page of the SAS log and SAS output.</td>
</tr>
<tr>
<td></td>
<td>PRINTMSGLIST System Option (p. 219)</td>
<td>Specifies whether to print all messages to the SAS log or to print only top-level messages to the SAS log.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Log and procedure output control:</td>
<td><strong>SOURCE System Option (p. 240)</strong></td>
<td>Specifies whether SAS writes source statements to the SAS log.</td>
</tr>
<tr>
<td>SAS log and procedure output</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log and procedure output control: SVG</td>
<td><strong>SOURCE2 System Option (p. 240)</strong></td>
<td>Specifies whether SAS writes secondary source statements from included files to the SAS log.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>DATE System Option (p. 104)</strong></td>
<td></td>
<td>Specifies whether to print the date and time that a SAS program started.</td>
</tr>
<tr>
<td><strong>DETAILS System Option (p. 107)</strong></td>
<td></td>
<td>Specifies whether to include additional information when files are listed in a SAS library.</td>
</tr>
<tr>
<td><strong>DTRESET System Option (p. 119)</strong></td>
<td></td>
<td>Specifies whether to update the date and time in the SAS log and in the procedure output file.</td>
</tr>
<tr>
<td><strong>LINESIZE= System Option (p. 170)</strong></td>
<td></td>
<td>Specifies the line size for the SAS log and for SAS procedure output.</td>
</tr>
<tr>
<td><strong>MISSING= System Option (p. 178)</strong></td>
<td></td>
<td>Specifies the character to print for missing numeric values.</td>
</tr>
<tr>
<td><strong>NUMBER System Option (p. 182)</strong></td>
<td></td>
<td>Specifies whether to print the page number in the title line of each page of SAS output.</td>
</tr>
<tr>
<td><strong>PAGEBREAKINITIAL System Option (p. 194)</strong></td>
<td></td>
<td>Specifies whether to begin the SAS log and procedure output files for the LISTING destination on a new page.</td>
</tr>
<tr>
<td><strong>PAGESIZE= System Option (p. 196)</strong></td>
<td></td>
<td>Specifies the number of lines that compose a page of the SAS log and SAS output.</td>
</tr>
<tr>
<td>Log and procedure output control: SVG</td>
<td><strong>SVGCONTROLBUTTONS (p. 248)</strong></td>
<td>Specifies whether to display the paging control buttons and an index in a multipage SVG document.</td>
</tr>
<tr>
<td></td>
<td><strong>SVGHEIGHT= System Option (p. 249)</strong></td>
<td>Specifies the height of the viewport unless the SVG output is embedded in another SVG output; specifies the value of the height attribute of the outermost <code>&lt;svg&gt;</code> element in the SVG file.</td>
</tr>
<tr>
<td></td>
<td><strong>SVGPRESERVEASPECTRATIO= System Option (p. 251)</strong></td>
<td>Specifies whether to force uniform scaling of SVG output; specifies the preserveAspectRatio attribute on the outermost <code>&lt;svg&gt;</code> element.</td>
</tr>
<tr>
<td></td>
<td><strong>SVGTITLE= System Option (p. 254)</strong></td>
<td>Specifies the title in the title bar of the SVG output; specifies the value of the <code>&lt;title&gt;</code> element in the SVG file.</td>
</tr>
<tr>
<td></td>
<td><strong>SVGVIEWBOX= System Option (p. 255)</strong></td>
<td>Specifies the coordinates, width, and height that are used to set the viewBox attribute on the outermost <code>&lt;svg&gt;</code> element, which enables SVG output to scale to the viewport.</td>
</tr>
<tr>
<td></td>
<td><strong>SVGWIDETH= System Option (p. 257)</strong></td>
<td>Specifies the width of the viewport unless the SVG output is embedded in another SVG output; specifies the value of the width attribute in the outermost <code>&lt;svg&gt;</code> element in the SVG file.</td>
</tr>
<tr>
<td>Category</td>
<td>Language Elements</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SVGX= System Option (p. 259)</td>
<td>Specifies the x-axis coordinate of one corner of the rectangular region into which an embedded &lt;svg&gt; element is placed; specifies the x attribute in the outermost &lt;svg&gt; element in an SVG file.</td>
<td></td>
</tr>
<tr>
<td>SVGY= System Option (p. 260)</td>
<td>Specifies the y-axis coordinate of one corner of the rectangular region into which an embedded &lt;svg&gt; element is placed; specifies the y attribute in the outermost &lt;svg&gt; element in an SVG file.</td>
<td></td>
</tr>
<tr>
<td>Sort: Procedure options</td>
<td>SORTDUP= System Option (p. 235)</td>
<td>Specifies whether the SORT procedure removes duplicate variables based on all variables in a data set or the variables that remain after the DROP or KEEP data set options have been applied.</td>
</tr>
<tr>
<td></td>
<td>SORTEQUALS System Option (p. 236)</td>
<td>Specifies whether observations in the output data set with identical BY variable values are in a particular order.</td>
</tr>
<tr>
<td></td>
<td>SORTSIZE= System Option (p. 237)</td>
<td>Specifies the amount of memory that is available to the SORT procedure.</td>
</tr>
<tr>
<td></td>
<td>SORTVALIDATE System Option (p. 238)</td>
<td>Specifies whether the SORT procedure verifies if a data set is sorted according to the variables in the BY statement when a user-specified sort order is denoted in the sort indicator.</td>
</tr>
<tr>
<td>System administration: Code generation</td>
<td>CGOPTIMIZE= System Option (p. 87)</td>
<td>Specifies the level of optimization to perform during code compilation.</td>
</tr>
<tr>
<td>System administration: Installation</td>
<td>SETINIT System Option (p. 234)</td>
<td>Specifies whether site license information can be altered.</td>
</tr>
<tr>
<td>System administration: Memory</td>
<td>SORTSIZE= System Option (p. 237)</td>
<td>Specifies the amount of memory that is available to the SORT procedure.</td>
</tr>
<tr>
<td></td>
<td>SUMSIZE= System Option (p. 247)</td>
<td>Specifies a limit on the amount of memory that is available for data summarization procedures when class variables are active.</td>
</tr>
<tr>
<td>System administration: Performance</td>
<td>BUFNO= System Option (p. 77)</td>
<td>Specifies the number of buffers to be allocated for processing SAS data sets.</td>
</tr>
<tr>
<td></td>
<td>BUFSIZE= System Option (p. 78)</td>
<td>Specifies the permanent buffer page size for output SAS data sets.</td>
</tr>
<tr>
<td></td>
<td>CGOPTIMIZE= System Option (p. 87)</td>
<td>Specifies the level of optimization to perform during code compilation.</td>
</tr>
<tr>
<td></td>
<td>CMPMODEL= System Option (p. 94)</td>
<td>Specifies the output model type for the MODEL procedure.</td>
</tr>
<tr>
<td></td>
<td>CMPOPT= System Option (p. 95)</td>
<td>Specifies the type of code generation optimizations to use in the SAS language compiler.</td>
</tr>
<tr>
<td></td>
<td>COMPRESS= System Option (p. 99)</td>
<td>Specifies the type of compression of observations to use for output SAS data sets.</td>
</tr>
</tbody>
</table>
## Dictionary of System Options

### CPUCOUNT= System Option (p. 101)
Specifies the number of processors that the thread-enabled applications should assume will be available for concurrent processing.

### SORTSIZE= System Option (p. 237)
Specifies the amount of memory that is available to the SORT procedure.

### THREADS System Option (p. 268)
Specifies that SAS uses threaded processing if available.

### PDFPASSWORD= System Option (p. 211)
Specifies the password to use to open a PDF document and the password used by a PDF document owner.

### PDFSECURITY= System Option (p. 213)
Specifies the level of encryption for PDF documents.

### RLANG System Option (p. 223)
Specifies whether SAS executes R language statements.

### APPEND= System Option
Appends a value to the existing value of the specified system option.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation, OPTIONS statement, SAS System Options window</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Environment control: Files</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>ENVFILES</td>
</tr>
<tr>
<td>Note:</td>
<td>This option cannot be restricted by a site administrator. For more information, see &quot;Restricted Options&quot; on page 6.</td>
</tr>
<tr>
<td>See:</td>
<td>For the syntax to use in your operating environment when you start SAS, see the documentation for your operating environment: &quot;APPEND System Option: UNIX&quot; in SAS Companion for UNIX Environments &quot;APPEND System Option: Windows&quot; in SAS Companion for Windows &quot;APPEND= System Option: z/OS&quot; in SAS Companion for z/OS</td>
</tr>
</tbody>
</table>

### Syntax

\[
\text{APPEND}=(\text{system-option-1} = \text{argument-1} \ <\text{system-option-n} = \text{argument-n}>)
\]
Syntax Description

**system-option**

can be AUTOEXEC, CMPLIB, FMTSEARCH, HELPLOC, MAPS, MSG, SASAUTOS, SASHELP, SASSCRIPT, or SET.

**Note:** Some of these options are available only when SAS starts. These options can be specified in the APPEND= option only when the APPEND= option is specified in a configuration file or a SAS command.

**argument**

specifies a new value that you want to append to the current value of **system-option**.

**argument** can be any value that could be specified for **system-option** if **system-option** is set using the OPTIONS statement.

Details

If you specify a new value for the AUTOEXEC, CMPLIB, FMTSEARCH, SASHELP, MAPS, MSG, SASAUTOS, SASSCRIPT, or SET system options, the new value replaces the value of the option. Instead of replacing the value, you can use the APPEND= system option to append a new value to the current value of the option.

For a list of system options that the APPEND= system option and the INSERT= system option support, including the system options that can be used when SAS starts, submit the following OPTIONS procedure:

```sas
proc options listinsertappend;
run;
```

Comparisons

The APPEND= system option adds a new value to the end of the current value of the AUTOEXEC, CMPLIB, FMTSEARCH, HELPLOC, MAPS, MSG, SASAUTOS, SASSCRIPT, or SET system options. The INSERT= system option adds a new value as the first value of one of these system options.

Example

The following table shows the results of adding a value to the end of the FMTSEARCH= option value:

<table>
<thead>
<tr>
<th>Current FMTSEARCH= Value</th>
<th>Value of APPEND= System Option</th>
<th>New FMTSEARCH= Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(WORK LIBRARY)</td>
<td>(fmtsearch=(abc def))</td>
<td>(WORK LIBRARY ABC DEF)</td>
</tr>
</tbody>
</table>

See Also

- “Changing an Option Value by Using the INSERT and APPEND System Options” on page 13

System Options:

- “INSERT= System Option” on page 159
- “INSERT System Option: UNIX” in *SAS Companion for UNIX Environments*
APPLETLOC= System Option

Specifies the location of Java applets.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Files

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

APPLETLOC="base-URL"

Syntax Description

“base-URL” specifies the address where the SAS Java applets are located. The maximum address length is 256 characters.

Details

The APPLETLOC= system option specifies the base location (typically a URL) of Java applets. These applets are typically accessed from an intranet server or a local CD-ROM.

Example

Some examples of the base-URL are

• "file://e:\java"
• "http://server.abc.com/SAS/applets"

AUTHPROVIDERDOMAIN System Option

 Associates a domain suffix with an authentication provider.

Valid in: Configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP=

Alias: AUTHPD

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

In Windows and z/OS operating environments:

```plaintext
AUTHPROVIDERDOMAIN provider : domain

AUTHPROVIDERDOMAIN (provider–1 : domain–1<, ...provider-n : domain-n> )
```

In UNIX operation environments:

```plaintext
AUTHPROVIDERDOMAIN \( (provider–1 : domain–1<, ...provider-n : domain-n> ) \)
```

Syntax Description

**provider**

specifies the authentication provider that is associated with a domain. The following are valid values for `provider`:

- **ADIR**
  
specifies that the authentication provider be a Microsoft Active Directory server that accepts a bind containing user names and passwords for authentication.

- **HOSTUSER**
  
specifies that user names and passwords be authenticated by using the authentication processing that is provided by the host operating system.

  **Windows specifics:** Under the Windows operating environment, assigning the authentication provider using the HOSTUSER domain is the same as assigning the authentication provider using the AUTHSERVER system option. You might want to use the AUTHPROVIDERDOMAIN system option when you specify multiple authentication providers.

- **LDAP**
  
specifies that the authentication provider use a directory server to specify the bind distinguished name (BINDDN) and a password for authentication.

**domain**

specifies a site-specific domain name. Quotation marks are required if the domain name contains blanks.

Details

SAS is able to provide authentication of a user through the use of many authentication providers. The AUTHPROVIDERDOMAIN= system option associates a domain suffix with an authentication provider. This association enables the SAS server to choose the authentication provider by the domain name that is presented.

When a domain suffix is not specified or the domain suffix is unknown, authentication is performed on the user ID and password by the host operating system.

Parenthesis are required when you specify more than one set of `provider : domain` pairs.

The maximum length for the AUTHPROVIDERDOMAIN option value is 1,024 characters.

To use the Microsoft Active Directory or LDAP authentication providers, these environment variables must be set in the server or spawner startup script:

- **Microsoft Active Directory Server:**
  
  - `AD_PORT=Microsoft Active Directory port number`
  
  - `AD_HOST=Microsoft Active Directory host name`
LDAP Server:
- LDAP_PORT=LDAP port number
- LDAP_BASE=base distinguished name
- LDAP_HOST=LDAP host name

LDAP Server for users connecting with a user ID instead of a distinguished name (DN):
- LDAP_PRIV_DN=privileged DN that is allowed to search for users
- LDAP_PRIV_PW=LDAP_PRIV_DN password

Note: If the LDAP server allows anonymous binds, then LDAP_PRIV_DN and LDAP_PRIV_PW are not required.

In addition to setting these environment variables, you can set the LDAP_IDATTR environment variable to the name of the person-entry LDAP attribute that stores the user ID if the attribute does not contain the default value of uid.

Example

The following examples show you how to specify the AUTHPROVIDERDOMAIN option:

- `-authpd ldap:sas` causes the SAS server to send credentials for users who log on as anything@sas to LDAP for authentication.
- `-authpd adir:sas` causes the SAS server to send credentials for users who log on as anything@sas to Active Directory for authentication.
- `authproviderdomain (hostuser:'my domain', ldap:sas)` causes the SAS server to send credentials for users who log on as the following:
  - When a user logs on as anything@'my domain', authentication is provided by the operating system authentication system
  - When a user logs on as anything@sas, authentication is provided by LDAP

See Also

System Options:
- “PRIMARYPROVIDERDOMAIN= System Option” on page 215

AUTOCORRECT System Option

Specifies whether SAS attempts to automatically correct misspelled procedure names, misspelled procedure keywords, or misspelled global statement names.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

AUTOCORRECT | NOAUTOCORRECT

Syntax Description

**AUTOCORRECT**

specifies that SAS attempts to automatically correct misspelled procedure names, misspelled procedure keywords, or misspelled global statement names. This is the default.

**NOAUTOCORRECT**

specifies that SAS does not automatically attempt to correct misspelled procedure names, misspelled procedure keywords, or misspelled global statement names.

Details

In previous releases of SAS, SAS always attempted to correct misspellings. The AUTOCORRECT option enables you to turn off autocorrection.

When AUTOCORRECT is set and a procedure name, a procedure keyword, or a global statement name is misspelled in a SAS program, SAS attempts to interpret the misspelling when a program is compiled. If the attempt succeeds, SAS corrects the error, prints a warning message to the log, and continues processing. If the error cannot be corrected, SAS writes an error message to the log.

When NOAUTOCORRECT is set, SAS writes the misspelling notification to the SAS log and ends the program.

Example

The following example shows a misspelled global statement name, a misspelled procedure option name, and a misspelled procedure name.

```sas
/* AUTOCORRECT is the default value */
options autocorrect;
data numbers;
   input x y z;
datalines;
14.2   25.2   96.8
10.8   51.6   96.8
33.5   27.4   66.5
run;

options obs=1;

proc print ddata=numbers;
run;

options noautocorrect;

proc prints ddata=numbers;
run;
```
options autocorrect;
data numbers;
input x y z;
datalines;
NOTE: The data set WORK.NUMBERS has 3 observations and 3 variables.
NOTE: DATA statement used (Total process time):
    real time      2.75 seconds
    cpu time       0.64 seconds
run;
options obs=1;
run;
WARNING 14-169: Assuming the symbol OPTIONS was misspelled as optionss.
proc print ddata=numbers;
run;
WARNING 1-322: Assuming the symbol DATA was misspelled as ddata.
run;
NOTE: There were 1 observations read from the data set WORK.NUMBERS.
NOTE: PROCEDURE PRINT used (Total process time):
    real time      3.84 seconds
    cpu time       1.07 seconds
options noautocorrect;
proc prints ddata=numbers;
run;
NOTE: The SAS System stopped processing this step because of errors.

AUTOSAVELOC= System Option

Specifies the location of the Program Editor autosave file.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Display

PROC OPTIONS GROUP= ENVDISPLAY

Restriction: The location that is specified by the AUTOSAVELOC= system option is valid only for the Program Editor. This option does not apply to the Enhanced Editor.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “AUTOSAVELOC System Option: UNIX” in SAS Companion for UNIX Environments
Syntax

**AUTOSAVELOC=** "location"

**Syntax Description**

*location*

specifies the pathname of the autosave file. If *location* contains spaces or is specified in an OPTIONS statement, then enclose *location* in quotation marks.

**See Also**

- “Saving Program Editor Files Using Autosave” in Chapter 3 of *SAS Companion for Windows*
- “Program Editor Window” in the SAS Help and Documentation

---

**BINDING= System Option**

Specifies the binding edge for duplexed printed output.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
- Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP=**
- ODSPRINT

**Restriction:**
- This option is ignored if the printer does not support duplex (two-sided) printing.

**Note:**
- This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

**BINDING=** DEFAULTEDGE | LONGEDGE | SHORTEDGE

**Syntax Description**

**DEFAULT | DEFAULTEDGE**

specifies that duplexing is done using the default binding edge.

**LONG | LONGEDGE**

specifies the long edge as the binding edge for duplexed output.

**SHORT | SHORTEDGE**

specifies the short edge as the binding edge for duplexed output.

**Details**

The binding edge setting determines how the paper is oriented before output is printed on the second side.

**See Also**


System Options:

- “DUPLEX System Option” on page 120

**BOTTOMMARGIN= System Option**

Specifies the size of the margin at the bottom of a printed page.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: ODS Printing
- **PROC OPTIONS GROUP= ODSPRINT**
- **Default:** 0.00 in
- **Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```plaintext
BOTTOMMARGIN=margin-size<margin-unit>
```

**Syntax Description**

- **margin-size** specifies the size of the margin.
  - **Restriction:** The bottom margin should be small enough so that the top margin plus the bottom margin is less than the height of the paper.
  - **Interaction:** Changing the value of this option might result in changes to the value of the PAGESIZE= system option.

- `<margin-unit>` specifies the units for margin-size. The margin-unit can be in for inches or cm for centimeters. `<margin-unit>` is saved as part of the value of the BOTTOMMARGIN system option.
  - **Default:** inches

**Details**

All margins have a minimum that is dependent on the printer and the paper size.

**See Also**

- “Understanding ODS Destinations” in Chapter 3 of SAS Output Delivery System: User’s Guide

System Options:

- “LEFTMARGIN= System Option” on page 169
BUFF= System Option

Specifies the number of buffers to be allocated for processing SAS data sets.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Files: SAS Files
System administration: Performance

PROC OPTIONS GROUP=

SASFILES
PERFORMANCE

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “BUFF System Option: UNIX” in SAS Companion for UNIX Environments
“BUFF System Option: Windows” in SAS Companion for Windows

Syntax

BUFF=n | nK | nM | nG | hexX | MIN | MAX

Syntax Description

n | nK | nM | nG
 specifies the number of buffers to be allocated in multiples of 1: 1,024 (kilo); 1,048,576 (mega); 1,073,741,824 (giga). For example, a value of 8 specifies 8 buffers, and a value of 3m specifies 3,145,728 buffers.

Tip: Use the notation that best fits the memory size of your system.

hexX
 specifies the number of buffers as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx specifies 45 buffers.

MIN
 sets the minimum number of buffers to 0, which causes SAS to use the minimum optimal value for the operating environment. This is the default.

MAX
 sets the number of buffers to the maximum possible number in your operating environment, up to the largest four-byte, signed integer, which is 2^31–1, or approximately 2 billion.

Details

The number of buffers is not a permanent attribute of the data set; it is valid only for the current SAS session or job.

BUFF applies to SAS data sets that are opened for input, output, or update.
Using BUFNO= can improve execution time by limiting the number of input/output operations that are required for a particular SAS data set. However, the improvement in execution time comes at the expense of increased memory consumption.

You can estimate the number of buffers that you need from the data set page size and the amount of memory in your system. The data set page size can be specified by the BUFSIZE= system option or by the BUFSIZE= data set option. If the default is used, SAS uses the minimal optimal page size for the operating environment. You can find the page size for a data set in the output of the CONTENTS procedure. Once you have the data set page size and the amount of memory available, you can estimate the number of buffers that you need. If the number of buffers is too large, SAS might not have enough memory to process the DATA or PROC step. You can change the page size for a data set by re-creating the data set by using the BUFSIZE= data set option.

**Operating Environment Information**

Under the Windows operating environment, if the SGIO system option is set, the maximum number of bytes that can be processed in an I/O operation is 64MB. Therefore, \( \text{number-of-buffers} \times \text{page-size} \leq 64\text{MB} \).

**Comparisons**

- You can override the BUFNO= system option by using the BUFNO= data set option.
- To request that SAS allocate the number of buffers based on the number of data set pages and index file pages, use the SASFILE statement.

**See Also**

**Data Set Options:**

- “BUFNO= Data Set Option” in *SAS Data Set Options: Reference*

**Procedures:**

- Chapter 13, “CONTENTS Procedure” in *Base SAS Procedures Guide*

**Statements:**

- “SASFILE Statement” in *SAS Statements: Reference*

**System Options:**

- “BUFSIZE= System Option” on page 78

---

**BUFSIZE= System Option**

Specifies the permanent buffer page size for output SAS data sets.

**Valid in:**

- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Categories:**

- Files: SAS Files
- System administration: Performance

**PROC OPTIONS GROUP=**

- SASFILES
- PERFORMANCE
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “BUFSIZE System Option: UNIX” in SAS Companion for UNIX Environments
“BUFSIZE System Option: Windows” in SAS Companion for Windows

Syntax

BUFSIZE=n | nK | nM | nG | nT | hexX | MAX

Syntax Description

n | nK | nM | nG | nT
specifies the page size in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 8 specifies 8 bytes, and a value of 3m specifies 3,145,728 bytes.

Note: If the system option and the data set option are not set, the default is 0. This causes SAS to use the minimum optimal page size for the operating environment. The BUFSIZE= system option will be used with either of the following scenarios:

• if the BUFSIZE= data set option is not set
• if the BUFSIZE= data set option is set to zero

Use BUFSIZE=0 in order to reset the buffer page size to the default value in your operating environment.

hexX
specifies the page size as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx sets the page size to 45 bytes.

MAX
sets the page size to the maximum possible number in your operating environment, up to the largest four-byte, signed integer, which is $2^{31}$-1, or approximately 2 billion bytes.

Details

The page size is the amount of data that can be transferred from a single input/output operation to one buffer. The page size is a permanent attribute of the data set and is used when the data set is processed.

A larger page size can improve execution time by reducing the number of times SAS has to read from or write to the storage medium. However, the improvement in execution time comes at the expense of increased memory consumption.

To change the page size, use a DATA step to copy the data set and either specify a new page or use the SAS default.

Note: If you use the COPY procedure to copy a data set to another library that is allocated with a different engine, the specified page size of the data set is not retained.

Operating Environment Information
The default value for BUFSIZE= is determined by your operating environment and is set to optimize sequential access. To improve performance for direct (random) access, you should change the value for BUFSIZE=. For the default setting and
possible settings for direct access, see the BUFSIZE= system option in the SAS documentation for your operating environment.

Comparisons
The BUFSIZE= system option can be overridden by the BUFSIZE= data set option.

See Also

Data Set Options:
• “BUFSIZE= Data Set Option” in SAS Data Set Options: Reference

System Options:
• “BUFNO= System Option” on page 77

BYERR System Option
Specifies whether SAS produces errors when the SORT procedure attempts to process a _NULL_ data set.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
BYERR | NOBYERR

Syntax Description

BYERR
specifies that SAS issue an error message and stop processing if the SORT procedure attempts to sort a _NULL_ data set.

NOBYERR
specifies that SAS ignore the error message and continue processing if the SORT procedure attempts to sort a _NULL_ data.

Details
The VNFERR system option sets the error flag for a missing variable when a _NULL_ data set is used. The DSNFERR system option specifies how SAS responds when a SAS data set is not found.
BYLINE System Option

Specifies whether to print BY lines above each BY group.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: Procedure output

PROC OPTIONS
GROUP= LISTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

BYLINE | NOBYLINE

Syntax Description

BYLINE
specifies that BY lines are printed above each BY group.

NOBYLINE
suppresses the automatic printing of BY lines.

Details

Use NOBYLINE to suppress the automatic printing of BY lines in procedure output. You can then use #BYVAL, #BYVAR, or #BYLINE to display BYLINE information in a TITLE statement.

These SAS procedures perform their own BY line processing by displaying output for multiple BY groups on the same page:

- MEANS
- PRINT
- STANDARD
- SUMMARY
- TTEST (in SAS/STAT software).

With these procedures, NOBYLINE causes a page eject between BY groups. For PROC PRINT, the page eject between BY groups has the same effect as specifying the right most BY variable in the PAGEBY statement.
BYSORTED System Option

Specifies whether observations in one or more data sets are sorted in alphabetic or numeric order or are grouped in another logical order.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Input control: Data processing

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

BYSORTED | NOBYSORTED

Syntax Description

BYSORTED

specifies that observations in a data set or data sets are sorted in alphabetic or numeric order.

Requirement: When you use the BYSORTED option, observations must be ordered or indexed according to the values of BY variables.

Interaction: If both the BYSORTED system option and the NOTSORTED statement option on a BY statement are specified, then the NOTSORTED option in the BY statement takes precedence over the BYSORTED system option.

Tip: If BYSORTED is specified, then SAS assumes that the data set is ordered by the BY variable. BYSORTED should be used if the data set is ordered by the BY variable for better performance.

NOBYSORTED

specifies that observations with the same BY value are grouped together but are not necessarily sorted in alphabetic or numeric order.

Note: If a procedure ignores the NOTSORTED option in a BY statement, then it ignores the NOBYSORTED system option also.

Tips:

When the NOBYSORTED option is specified, you do not have to specify NOTSORTED on every BY statement to access the data sets.

NOBYSORTED is useful if you have data that falls into other logical groupings such as chronological order or linguistic order. NOBYSORTED allows BY
processing to continue without failure when a data set is not actually sorted in alphabetic or numeric order.

**Details**

The requirement for ordering or indexing observations according to the values of BY variables is suspended for BY-group processing when you use the NOBYSORTED option. By default, BY-group processing requires that your data be sorted in alphabetic or numeric order. If your data is grouped in any other way but alphabetic or numeric, then you must use the NOBYSORTED option to allow BY-processing to continue without failure. For more information about BY-group processing, see Chapter 20, “BY-Group Processing in the DATA Step,” in *SAS Language Reference: Concepts*

**See Also**

**Statements:**

- NOTSORTED option, “NOTSORTED” in Chapter 2 of *SAS Statements: Reference*

---

### CAPS System Option

Specifies whether to convert certain types of input to uppercase.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Input control: Data processing

**PROC OPTIONS GROUP=**

**CAPS | NOCAPS**

**Syntax Description**

**CAPS**

- specifies that SAS translate lowercase characters to uppercase in these types of input:
  - data following CARDS, CARDS4, DATALINES, DATALINES4, and PARMCARDS statements
  - text enclosed in single or double quotation marks
  - values in VALUE and INVALUE statements in the FORMAT procedure
  - titles, footnotes, variable labels, and data set labels
  - constant text in macro definitions
  - values of macro variables
  - parameter values passed to macros.
**Note**: Data read from external files and SAS data sets are not translated to uppercase.

**NOCAPS**
specifies that lowercase characters that occur in the types of input that are listed above are not translated to uppercase.

**Comparisons**
The CAPS system option and the CAPS command both specify whether input is converted to uppercase. The CAPS command, which is available in windows that allow text editing, can act as a toggle. The CAPS command converts all text that is entered from the keyboard to uppercase. If either the CAPS system option or the CAPS command is in effect, all applicable input is translated to uppercase.

**See Also**

**Commands:**
- “CAPS Command” in SAS Help and Documentation

---

**CARDIMAGE System Option**
Specifies whether SAS processes source and data lines as 80-byte cards.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
Input control: Data processing

**PROC OPTIONS GROUP=**
INPUTCONTROL

**Operating environment:**
CARDIMAGE is generally used in the z/OS operating environment; NOCARDIMAGE is used in other operating environments.

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**See:**
CARDIMAGE System Option under z/OS

---

**Syntax**
CARDIMAGE | NOCARDIMAGE

**Syntax Description**
CARDIMAGE
specifies that SAS source and data lines be processed as if they were punched card images—all exactly 80 bytes long and padded with blanks. That is, column 1 of a line is treated as if it immediately followed column 80 of the previous line. Therefore, tokens can be split across lines. (A token is a character or series of characters that SAS treats as a discrete word.)

Strings in quotation marks (literal tokens) that begin on one line and end on another are treated as if they contained blanks out to column 80 of the first line. Data lines
longer than 80 bytes are split into two or more 80-byte lines. Data lines are not truncated regardless of their length.

**NOCARDIMAGE**

specifies that SAS source and data lines not be treated as if they were 80-byte card images. When NOCARDIMAGE is in effect, the end of a line is always treated as the end of the last token, except for strings in quotation marks. Strings in quotation marks can be split across lines. Other types of tokens cannot be split across lines under any circumstances. Strings in quotation marks that are split across lines are not padded with blanks.

**Example**

Consider the following DATA step:

```sas
data;
  x='A
  B';
run;
```

If CARDIMAGE is in effect, the variable X receives a value that consists of 78 characters: the A, 76 blanks, and the B. If NOCARDIMAGE is in effect, the variable X receives a value that consists of two characters: AB, with no intervening blanks.

**CATCACHE= System Option**

Specifies the number of SAS catalogs to keep open in cache memory.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Files: SAS Files
- **PROC OPTIONS GROUP=** SASFILES

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**See:** “CATCACHE System Option: UNIX” in SAS Companion for UNIX Environments
“CATCACHE System Option: Windows” in SAS Companion for Windows
“CATCACHE= System Option: z/OS” in SAS Companion for z/OS

**Syntax**

```
CATCACHE=n | hexX | MIN | MAX
```

**Syntax Description**

- **n** specifies any integer greater than or equal to 0 in terms of bytes. If $n > 0$, SAS places up to that number of open-file descriptors in cache memory instead of closing the catalogs.

- **hexX** specifies the number of open-file descriptors that are kept in cache memory as a hexadecimal number. You must specify the value beginning with a number (0-9),
followed by an X. For example, the value 2dx sets the number of catalogs to keep open to 45 catalogs.

**MIN**
sets the number of open-file descriptors that are kept in cache memory to 0.

**MAX**
sets the number of open-file descriptors that are kept in cache memory to the largest, signed, 4-byte integer representable in your operating environment.

**Tip:** The recommended maximum setting for this option is 10.

### Details
Use the CATCACHE= system option to tune an application by avoiding the overhead of repeatedly opening and closing the same SAS catalogs.

**CAUTION:** When using both the CBUFNO= and CATCACHE= options, if one of the option’s value is set higher than zero, the other option must be set to zero.

---

**CBUFNO= System Option**

Specifies the number of extra page buffers to allocate for each open SAS catalog.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Files: SAS Files

**PROC OPTIONS GROUP=** SASFILES

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

### Syntax

**Syntax**

CBUFNO= \( n \mid nK \mid nM \mid nG \mid nT \mid \text{hex}X \mid \text{MIN} \mid \text{MAX} \)

### Syntax Description

\( n \mid nK \mid nM \mid nG \mid nT \)
specifies the number of extra page buffers in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of \( 8 \) specifies 8 bytes, and a value of \( 3m \) specifies 3,145,728 bytes.

**MIN**
sets the number of extra page buffers to 0.

**MAX**
sets the number of extra page buffers to 20.

**hexX**
specifies the number of extra page buffers as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an X. For example, the value \( 0ax \) sets the number of extra page buffers to 10 buffers.
Details

The CBUFNO= option is similar to the BUFNO= option that is used for SAS data set processing.

Increasing the value for the CBUFNO= option might result in fewer I/O operations when your application reads very large objects from catalogs. Increasing this value also comes with the normal tradeoff between performance and memory usage. If memory is a serious constraint for your system, you should not increase the value of the CBUFNO= option. Do not increase the value of the CBUFNO= option if you have increased the value of the CATCACHE= option.

**CAUTION:**

When using both the CBUFNO= and CATCACHE= options, if one of the option's value is set higher than zero, the other option must be set to zero.

---

**CENTER System Option**

Specifies whether to center or left-align SAS procedure output.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: Procedure output

**PROC OPTIONS GROUP=** LISTCONTROL

**Alias:** CENTRE

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

CENTER | NOCENTER

**Syntax Description**

CENTER

centers SAS procedure output.

NOCENTER

left aligns SAS procedure output.

---

**CGOPTIMIZE= System Option**

Specifies the level of optimization to perform during code compilation.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Categories:** System administration: Performance

System administration: Code generation

**PROC OPTIONS GROUP=** PERFORMANCE
CODEGEN
Alias: CGOPT
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
CGOPTIMIZE=0 | 1 | 2 | 3

Syntax Description
0
specifies not to perform optimization.
1
specifies to perform stage 1 optimization. Stage 1 optimization removes redundant instructions, missing value checks, and repetitive computations for array subscriptions; detects patterns of instructions and replaces them with more efficient sequences.
2
specifies to perform stage 2 optimization. Stage 2 performs optimizations that pertain to the SAS register.
Interaction: Stage 2 optimization for a large DATA step program can result in a significant increase in compilation time and thus overall execution time.
3
specifies to perform full optimization, which is a combination of stages 1 and 2. This is the default value.

See Also
“Reducing CPU Time By Modifying Program Compilation Optimization” in Chapter 12 of SAS Language Reference: Concepts

CHARCODE System Option
Specifies whether specific keyboard combinations are substituted for special characters that are not on the keyboard.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Environment control: Display
PROC OPTIONS GROUP= ENVDISPLAY
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
CHARCODE | NOCHARCODE
Syntax Description

CHARCODE
allows certain character combinations to be substituted for special characters that might not be on your keyboard.

NOCHARCODE
does not allow substitutions for certain keyboard characters.

Details

If you do not have the following symbols on your keyboard, you can use these character combinations to create the symbols that you need when CHARCODE is active:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>back quotation mark ('`)</td>
<td>`:</td>
</tr>
<tr>
<td>backslash ('')</td>
<td>`,</td>
</tr>
<tr>
<td>left brace ('{')</td>
<td>`?</td>
</tr>
<tr>
<td>right brace ('}')</td>
<td>`)</td>
</tr>
<tr>
<td>logical not sign ('¬ or ^')</td>
<td><code>=</code></td>
</tr>
<tr>
<td>left square bracket ('[')</td>
<td>`&lt;?</td>
</tr>
<tr>
<td>right square bracket (']')</td>
<td>`?&gt;</td>
</tr>
<tr>
<td>underscore (_)</td>
<td>`?</td>
</tr>
<tr>
<td>vertical bar ('</td>
<td>')</td>
</tr>
</tbody>
</table>

Example

This statement produces the output [TEST TITLE]:

title '?<TEST TITLE?>';

CHKPTCLEAN System Option

When SAS is in checkpoint mode or restart mode, specifies whether to erase the contents of the Work library after a batch program executes successfully.

Valid in: Configuration file, SAS invocation
Category: Environment control: Error handling
PROC OPTIONS GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

CHKPTCLEAN | NOCHKPTCLEAN

Syntax Description

CHKPTCLEAN
specifies that the files in the Work library are to be erased after a batch program successfully executes in checkpoint mode or restart mode.

NOCHKPTCLEAN
specifies that the files in the Work library are not to be erased after a batch program successfully executes in checkpoint mode or restart mode. This is the default.

Details

Typically, checkpoint mode or restart mode is started with the NOWORKTERM and NOWORKINIT system options set. When these option are set, the Work library is preserved between SAS sessions. You can use the CHKPTCLEAN system option to erase all files from the Work library if you no longer need the files after your batch program successfully runs in checkpoint mode or restart mode.

This option is effective only when the following conditions are met:

• SAS is in either checkpoint mode or restart mode. SAS enters checkpoint mode when the STEPCHKPT option or the LABELCHKPT option is set. SAS enters restart mode when the STEPRESTART option or the LABELRESTART option is set.
• The checkpoint library is Work.
• The program runs successfully in batch mode.

If the program does not run successfully, the files in the Work library are not erased, regardless of whether the CHKPTCLEAN option is set.

Comparisons

The CHKPTCLEAN option erases the contents of the Work library after the successful completion of a batch program only in checkpoint mode or restart mode.

The WORKTERM option erases the contents of the Work library when a SAS session ends.

See Also

System Options:

• “LABELCHKPT System Option” on page 164
• “LABELRESTART System Option” on page 167
• “STEPCHKPT System Option” on page 243
• “STEPRESTART System Option” on page 246
• “WORKTERM System Option” on page 296
CLEANUP System Option

For an out-of-resource condition, specifies whether to perform an automatic cleanup or a user-specified cleanup.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Environment control: Error handling
PROC OPTIONS GROUP=ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
See: CLEANUP System Option under UNIX, Windows

Syntax

CLEANUP | NOCLEANUP

Syntax Description

CLEANUP
specifies that during the entire session, SAS attempts to perform automatic, continuous cleanup of resources that are not essential for execution. Nonessential resources include resources that are not visible to the user (for example, cache memory) and resources that are visible to the user (for example, the KEYS windows).

When CLEANUP is in effect and an out-of-resource condition occurs (except for a disk-full condition), a dialog box is not displayed, and no intervention is required by the user. When CLEANUP is in effect and a disk-full condition occurs, a dialog box displays that allows the user to decide how to proceed.

NOCLEANUP
specifies that SAS allow the user to choose how to handle an out-of-resource condition. When NOCLEANUP is in effect and SAS cannot execute because of a lack of resources, SAS automatically attempts to clean up resources that are not visible to the user (for example, cache memory). However, resources that are visible to the user (for example, windows) are not automatically cleaned up. Instead, a dialog box appears that allows the user to choose how to proceed.

Details

This table lists the dialog box choices:

<table>
<thead>
<tr>
<th>Dialog Box Choice</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Free windows</td>
<td>clears all windows not essential for execution.</td>
</tr>
<tr>
<td>Clear paste buffers</td>
<td>deletes paste buffer contents.</td>
</tr>
<tr>
<td>Deassign inactive librefs</td>
<td>prompts user for librefs to delete.</td>
</tr>
<tr>
<td>Dialog Box Choice</td>
<td>Action</td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>Delete definitions of all SAS macros and macro variables</td>
<td>deletes all macro definitions and variables.</td>
</tr>
<tr>
<td>Delete SAS files</td>
<td>allows user to select files to delete.</td>
</tr>
<tr>
<td>Clear Log window</td>
<td>erases Log window contents.</td>
</tr>
<tr>
<td>Clear Output window</td>
<td>erases Output window contents.</td>
</tr>
<tr>
<td>Clear Program Editor window</td>
<td>erases Program Editor window contents.</td>
</tr>
<tr>
<td>Clear source spooling/DMS recall buffers</td>
<td>erases recall buffers.</td>
</tr>
<tr>
<td>More items to clean up</td>
<td>displays a list of other resources that can be cleaned up.</td>
</tr>
<tr>
<td>Clean up everything</td>
<td>cleans up all other options that are shown on the dialog box.</td>
</tr>
<tr>
<td>Continuous clean up</td>
<td>performs automatic, continuous cleanup. When continuous clean up is selected, SAS cleans up as many resources as possible in order to continue execution, and it ceases to display the requester window. Selecting Continuous clean up has the same effect as specifying CLEANUP. This selection applies to the current cleanup request and to the remainder of the SAS session.</td>
</tr>
</tbody>
</table>

Some operating environments might also include these choices in the dialog box:

<table>
<thead>
<tr>
<th>Dialog Box Choice</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execute X command</td>
<td>enables the user to erase files and perform other cleanup operations.</td>
</tr>
<tr>
<td>Do nothing</td>
<td>halts the cleanup request and returns to the SAS session. This selection only applies to the current cleanup request, not to the entire SAS session.</td>
</tr>
</tbody>
</table>

If an out-of-resource condition cannot be resolved, the dialog box continues to display. In that case, see the SAS documentation for your operating environment for instructions on terminating the SAS session. When running in modes other than a windowing environment, the operation of CLEANUP depends on your operating environment. For details, see the SAS documentation for your operating environment.
**CMPLIB= System Option**

Specifies one or more SAS data sets that contain compiler subroutines to include during program compilation.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, System Options window

**Category:** Files: SAS Files

**PROC OPTIONS GROUP=** SASFILES

**Syntax**

\[
\text{CMPLIB=} \text{libref.data-set} \mid (\text{libref.data-set-1} \ldots \text{libref.data-set-n}) \mid (\text{libref.data-set-n} \ldots \text{libref.data-set-m})
\]

**Syntax Description**

- **libref.data-set** specifies the libref and the data set of the compiler subroutines that are to be included during program compilation. The *libref* and *data-set* must be valid SAS names.

- **libref.data-set-n – libref.data-set-m** specifies a range of compiler subroutines that are to be included during program compilation. The name of the libref and the data set must be valid SAS names that contain a numeric suffix.

**Details**

SAS procedures, DATA steps, and macro programs that perform non-linear statistical modeling or optimization use a SAS language compiler subsystem that compiles and executes your SAS programs. The compiler subsystem generates machine language code for the computer on which SAS is running. The SAS procedures that use the SAS language compiler are CALIS, COMPILE, GA, GENMOD, MODEL, NLIN, NLMIXED, NLP, PHREG, Risk Dimensions procedures, and SQL.

The subroutines that you want to include must already have been compiled. All the subroutines in *libref.data-set* are included.

You can specify a single *libref.data-set*, a list of *libref.data-set* names, or a range of *libref.data-set* names with numeric suffixes. When you specify more than one *libref.data-set* name, separate the names with a space and enclose the names in parentheses.
Example

<table>
<thead>
<tr>
<th>Number of Libraries</th>
<th>OPTIONS Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>One library</td>
<td>options cmplib=sasuser.cmpl;</td>
</tr>
<tr>
<td>Two or more libraries</td>
<td>options cmplib=(sasuser.cmpl sasuser.cmplA sasuser.cmpl3);</td>
</tr>
<tr>
<td>A range of libraries</td>
<td>options cmplib=(sasuser.cmpl1 - sasuser.cmpl6);</td>
</tr>
</tbody>
</table>

See Also

System Options:
- “APPEND= System Option” on page 68
- “INSERT= System Option” on page 159

**CMPMODEL= System Option**

Specifies the output model type for the MODEL procedure.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, System Options window

**Category:** System administration: Performance

**PROC OPTIONS GROUP=** Performance

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

CMPMODEL=**BOTH** | **CATALOG** | **XML**

**Syntax Description**

**BOTH**
- specifies that the MODEL procedure create two output types for a model, one as a SAS catalog entry and the other as an XML file. This is the default.

**CATALOG**
- specifies that the output model type is an entry in a SAS catalog.

**XML**
- specifies that the output model type is an XML file.

See Also

Procedures:
CMPOPT= System Option
Specifies the type of code generation optimizations to use in the SAS language compiler.

Valid in: Configuration file, SAS invocation, OPTIONS statement, System Options window
Category: System administration: Performance
PROC OPTIONS GROUP= PERFORMANCE
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
CMPOPT=optimization-value | (optimization-value-1…optimization-value-n) | "optimization-value-1…optimization-value-n" | ALL | NONE
NOCMPOPT

Syntax Description
optimization
specifies the type of optimization that the SAS compiler is to use. Valid values are
EXTRAMATH | NOEXTRAMATH
specifies to keep or remove mathematical operations that do not affect the outcome of a statement. When you specify EXTRAMATH, the compiler retains the extra mathematical operations. When you specify NOEXTRAMATH, the extra mathematical operations are removed.
FUNCDIFFERENCING | NOFUNCDIFFERENCING
specifies whether analytic derivatives are computed for user defined functions. When you specify NOFUNCDIFFERENCING, analytic derivatives are computed for user defined functions. When you specify FUNCDIFFERENCING, numeric differencing is used to calculate derivatives for user defined functions. The default is NOFUNCDIFFERENCING.
GUARDCHECK | NOGUARDCHECK
specifies whether to check for array boundary problems. When you specify GUARDCHECK, the compiler checks for array boundary problems. When you specify NOGUARDCHECK, the compiler does not check for array boundary problems.
Interaction: NOGUARDCHECK is set when CMPOPT is set to ALL and when CMPOPT is set to NONE.
MISSCHECK | NOMISSCHECK
specifies whether to check for missing values in the data. If the data contains a significant amount of missing data, then you can optimize the compilation by specifying MISSCHECK. If the data rarely contains missing values, then you can optimize the compilation by specifying NOMISSCHECK.
PRECISE | NOPRECISE
specifies to handle exceptions at an operation boundary or at a statement boundary. When you specify PRECISE, exceptions are handled at the operation
boundary. When you specify NOPRECISE, exceptions are handled at the statement boundary.

**Tip:** EXTRAMATH, MISSCHECK, PRECISE, GUARDCHECK, and FUNCDIFFERENCING can be specified in any combination when you specify one or more values.

**ALL**
specifies that the compiler is to optimize the machine language code by using the (NOEXTRAMATH NOMISSCHECK NOPRECISE NOGUARDCHECK NOFUNCDIFFERENCING) optimization values. This is the default.

**Restriction:** ALL cannot be specified in combination with any other values.

**NONE**
specifies that the compiler is not set to optimize the machine language code by using the (EXTRAMATH MISSCHECK PRECISE NOGUARDCHECK FUNCDIFFERENCING) optimization values.

**Restriction:** NONE cannot be specified in combination with any other values.

**NOCMPOPT**
specifies to set the value of CMPOPT to ALL. The compiler is to optimize the machine language code by using the (NOEXTRAMATH NOMISSCHECK NOPRECISE NOGUARDCHECK NOGENSYM NAMES NOFUNCDIFFERENCING) optimization values.

**Restriction:** NOCMPOPT cannot be specified in combination with values for the CMPOPT option.

**Note:** NOGENSYM NAMES is a value used only by SAS and cannot be set in this option.

**Details**
SAS procedures that perform non-linear statistical modeling or optimization use a SAS language compiler subsystem that compiles and executes your SAS programs. The compiler subsystem generates machine language code for the computer on which SAS is running. By specifying values with the CMPOPT option, the machine language code can be optimized for efficient execution. The SAS procedures that use the SAS language compiler are CALIS, COMPIL E, GENMOD, MODEL, PHREG, NLIN, NLMixed, NLP, and RISK.

To specify multiple optimization values, the values must be enclosed in either parentheses, single quotation marks, or double quotation marks.

If a value is entered more than once, then the last setting is used. For example, if you specify CMPOPT=(PRECISE NOEXTRAMATH NOPRECISE), then the values that are set are NOEXTRAMATH and NOPRECISE. All leading, trailing, and embedded blanks are removed.

When you specify EXTRAMATH or NOEXTRAMATH, some of the mathematical operations that are either included or excluded in the machine language code are the following:

- $x \times 1$
- $x \times -1$
- $x \div 1$
- $x \div -1$
- $x + 0$
- $x$
- $x \ast x$
- $x \div x$
- $-x$
- any operation on two literal constants
### Example

<table>
<thead>
<tr>
<th>OPTIONS Statement</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>options cmpopt=(extramath);</code></td>
<td><code>(NOPRECISE EXTRAMATH NOMISSCHECK NOGUARDCHECK NOGENSYMNNAMES NOFUNCDIFFERENCING)</code></td>
</tr>
<tr>
<td><code>options cmpopt=(extramath misscheck precise);</code></td>
<td><code>(PRECISE EXTRAMATH MISSCHECK NOGUARDCHECK NOGENSYMNNAMES NOFUNCDIFFERENCING)</code></td>
</tr>
<tr>
<td><code>options nocmpopt;</code></td>
<td><code>(NOEXTRAMATH NOMISSCHECK NOPRECISE NOGUARDCHECK NOGENSYMNNAMES NOFUNCDIFFERENCING)</code></td>
</tr>
</tbody>
</table>

### COLLATE System Option

Specifies whether to collate multiple copies of printed output.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP:** ODSPRINT

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

### Syntax

**COLLATE | NOCOLLATE**

### Syntax Description

**COLLATE**

specifies to collate multiple copies of printed output.

**NOCOLLATE**

specifies not to collate multiple copies of printed output. This is the default.

### Details

When you send a print job to the printer and you want multiple copies of multiple pages, the COLLATE option controls how the pages are ordered:

- COLLATE causes the pages to print consecutively: 123, 123, 123...
- NOCOLLATE causes the same-numbered pages to print together: 111, 222, 333...

**Note:** You can also control collation with the SAS windowing environment Page Setup window, invoked with the DMPAGESETUP command.
Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and option values for some SAS system options might vary both by operating environment and by site. For details, see the SAS documentation for your operating environment.

See Also

- Chapter 15, “Printing with SAS,” in *SAS Language Reference: Concepts*
- “Understanding ODS Destinations” in Chapter 3 of *SAS Output Delivery System: User's Guide*

System Options:

- “COPIES= System Option” on page 101

COLORPRINTING System Option

Specifies whether to print in color if color printing is supported.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP=** ODSPRINT

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

COLORPRINTING | NOCOLORPRINTING

**Syntax Description**

COLORPRINTING

  specifies to attempt to print in color.

NOCOLORPRINTING

  specifies not to print in color.

**Details**

Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and option values for some SAS system options might vary both by operating environment and by site. For details, see the SAS documentation for your operating environment.

See Also

- Printing with SAS

**Statements:**

COMPRESS= System Option

Specifies the type of compression of observations to use for output SAS data sets.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Files: SAS Files
            System administration: Performance

PROC OPTIONS
GROUP= SASFILES
PERFORMANCE

Restriction: The TAPE engine does not support the COMPRESS= system option.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

COMPRESS= NO | YES | CHAR | BINARY

Syntax Description

NO
specifies that the observations in a newly created SAS data set are uncompressed (fixed-length records).

YES | CHAR
specifies that the observations in a newly created SAS data set are compressed (variable-length records) by SAS using RLE (Run Length Encoding). RLE compresses observations by reducing repeated consecutive characters (including blanks) to two-byte or three-byte representations.

Alias: ON

Note: COMPRESS=CHAR is accepted by Version 7 and later versions.

Tip: Use this compression algorithm for character data.

BINARY
specifies that the observations in a newly created SAS data set are compressed (variable-length records) by SAS using RDC (Ross Data Compression). RDC combines run-length encoding and sliding-window compression to compress the file.

Tip: This method is highly effective for compressing medium to large (several hundred bytes or larger) blocks of binary data (numeric variables). Because the compression function operates on a single record at a time, the record length needs to be several hundred bytes or larger for effective compression.

Details

Compressing a file is a process that reduces the number of bytes required to represent each observation. Advantages of compressing a file include reduced storage requirements for the file and fewer I/O operations necessary to read or write to the data during processing. However, more CPU resources are required to read a compressed file (because of the overhead of uncompressing each observation), and there are situations when the resulting file size might increase rather than decrease.
Use the COMPRESS= system option to compress all output data sets that are created during a SAS session. Use the option only when you are creating SAS data files (member type DATA). You cannot compress SAS views, because they contain no data.

Once a file is compressed, the setting is a permanent attribute of the file, which means that to change the setting, you must re-create the file. That is, to uncompress a file, specify COMPRESS=NO for a DATA step that copies the compressed file.

Note: For the COPY procedure, the default value CLONE uses the compression attribute from the input data set for the output data set. If the engine for the input data set does not support the compression attribute, then PROC COPY uses the current value of the COMPRESS= system option. For more information about CLONE and NOCLONE, see the COPY Statement options, Chapter 16, “DATASETS Procedure” in Base SAS Procedures Guide. This interaction does not apply when using SAS/SHARE or SAS/CONNECT.

Comparisons

The COMPRESS= system option can be overridden by the COMPRESS= option in the LIBNAME statement and the COMPRESS= data set option.

The data set option POINTOBS=YES, which is the default, determines that a compressed data set can be processed with random access (by observation number) rather than sequential access. With random access, you can specify an observation number in the FSEDIT procedure and the POINT= option in the SET and MODIFY statements.

When you create a compressed file, you can also specify REUSE=YES (as a data set option or system option) in order to track and reuse space. With REUSE=YES, new observations are inserted in space freed when other observations are updated or deleted. When the default REUSE=NO is in effect, new observations are appended to the existing file.

POINTOBS=YES and REUSE=YES are mutually exclusive. That is, they cannot be used together. REUSE=YES takes precedence over POINTOBS=YES. That is, if you set REUSE=YES, SAS automatically sets POINTOBS=NO.

The TAPE engine does not support the COMPRESS= system option, but the engine does support the COMPRESS= data set option.

The XPORT engine does not support compression.

See Also

• “Definition of Compression” in Chapter 26 of SAS Language Reference: Concepts

Data Set Options:

• “COMPRESS= Data Set Option” in SAS Data Set Options: Reference
• “POINTOBS= Data Set Option” in SAS Data Set Options: Reference
• “REUSE= Data Set Option” in SAS Data Set Options: Reference

Statements:

• “LIBNAME Statement” in SAS Statements: Reference

System Options:

• “REUSE= System Option” on page 221
**COPIES= System Option**

Specifies the number of copies to print.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP=** ODSPRINT

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

`COPIES=n`

**Syntax Description**

`n`

specifies the number of copies.

**See Also**

- Chapter 15, “Printing with SAS,” in *SAS Language Reference: Concepts*
- “Understanding ODS Destinations” in Chapter 3 of *SAS Output Delivery System: User's Guide*

**System Options:**

- “COLLATE System Option” on page 97

---

**CPUCOUNT= System Option**

Specifies the number of processors that the thread-enabled applications should assume will be available for concurrent processing.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** System administration: Performance

**PROC OPTIONS GROUP=** PERFORMANCE

**Default:** Under Windows and z/OS, the default is ACTUAL. Under UNIX, the default is either ACTUAL or 4 for systems that have more than four processors.

**Interaction:** If the THREADS system option is set to NOTHREADS, the CPUCOUNT= option has no effect.

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

CPUCOUNT= 1 - 1024 | ACTUAL

Syntax Description

1-1024
is the number of CPUs that SAS will assume are available for use by thread-enabled applications.

Tips:
The value is typically set to the actual number of CPUs available to the current process by your configuration.
Setting CPUCOUNT= to a number greater than the actual number of available CPUs might result in reduced overall performance of SAS.

ACTUAL
returns the number of physical processors that are associated with the operating system where SAS is executing. If the operating system is executing in a partition, the value of the CPUCOUNT system is the number of physical processors that are associated with the operating system in that partition.

Tips:
This number can be less than the physical number of CPUs if the SAS process has been restricted by system administration tools.
Setting CPUCOUNT= to ACTUAL at any time causes the option to be reset to the number of physical processors that are associated with the operating system at that time. If the operating system is executing in a partition, the value of the CPUCOUNT system is the number of physical processors that are associated with the operating system in that partition.
If your system supports Simultaneous Multi-Threading (SMT), hyperthreading, or Chip Multi-Threading (CMT), the value of the CPUCOUNT= option represents the number of such threads on the system.

Details

Certain procedures have been modified to take advantage of multiple CPUs by threading the procedure processing. The Base SAS engine also uses threading to create an index. The CPUCOUNT= option provides the information that is needed to make decisions about the allocation of threads.

Changing the value of CPUCOUNT= affects the degree of parallelism each thread-enabled process attempts to achieve. Setting CPUCOUNT to a number greater than the actual number of available CPUs might result in reduced overall performance of SAS.

Comparisons

When the related system option THREADS is in effect, threading will be active where available. The value of the CPUCOUNT= option affects the performance of THREADS by suggesting how many system CPUs are available for use by thread-enabled SAS procedures.

See Also

• Support for Parallel Processing
System Options:
- “THREADS System Option” on page 268
- “UTILLOC= System Option” on page 274

CPUID System Option
Specifies whether the CPU identification number is written to the SAS log.

Valid in: Configuration file, SAS invocation
Category: Log and procedure output control: SAS log
PROC OPTIONS GROUP=
  LOGCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
CPUID | NOCPUID

Syntax Description
CPUID
specifies that the CPU identification number is printed at the top of the SAS log after the licensing information.

NOCPUID
specifies that the CPU identification number is not written to the SAS log.

See Also

DATASTMTCHK= System Option
Specifies which SAS statement keywords are prohibited from being specified as a one-level DATA step name to protect against overwriting an input data set.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Files: SAS Files
PROC OPTIONS GROUP=
  SASFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
DATASTMTCHK=COREKEYWORDS | ALLKEYWORDS | NONE
Syntax Description

COREKEYWORDS
prohibits certain words as one-level SAS data set names in the DATA statement. They can appear as two-level names. The following keywords cannot appear as one-level SAS data set names:

- MERGE
- RETAIN
- SET
- UPDATE.

For example, SET is not acceptable in the DATA statement, but SAVE.SET and WORK.SET are acceptable. COREKEYWORDS is the default.

ALLKEYWORDS
prohibits any keyword that can begin a statement in the DATA step (for example, ABORT, ARRAY, INFILE) as a one-level data set name in the DATA statement.

NONE
provides no protection against overwriting SAS data sets.

Details

If you omit a semicolon in the DATA statement, you can overwrite an input data set if the next statement is SET, MERGE, or UPDATE. Different, but significant, problems arise when the next statement is RETAIN. DATASTMTCHK= enables you to protect yourself against overwriting the input data set.

DATE System Option

Specifies whether to print the date and time that a SAS program started.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL LISTCONTROL LOGCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DATE | NODATE

Syntax Description

DATE
specifies that the date and the time that the SAS program started are printed at the top of each page of the SAS log and any output that is created by SAS.
Note: In an interactive SAS session, the date and time are noted only in the output window.

**NODATE**
specifies that the date and the time are not printed.

**See Also**
“The SAS Log” in Chapter 9 of *SAS Language Reference: Concepts*

---

**DATESTYLE= System Option**

Specifies the sequence of month, day, and year when ANYTDTE, ANYTDTDM, or ANYTTME informat data is ambiguous.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Categories:** Environment control: Language control
Input control: Data processing

**PROC OPTIONS**

**GROUP=**

**INPUTCONTROL**

**LANGUAGECONTROL**

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

**DATESTYLE=** MDY | YMD | DMY | LOCALE

**Syntax Description**

**MDY**
specifies that SAS set the order as month, day, year.

**YMD**
specifies that SAS set the order as year, month, day.

**DMY**
specifies that SAS set the order as day, month, year.

**LOCALE**
specifies that SAS set the order based on the value that corresponds to the LOCALE= system option value and is one of the following: MDY | YMD | DMY.

**Details**

System option DATESTYLE= identifies the order of month, day, and year. The default value is LOCALE. The default LOCALE system option value is English, therefore, the default DATESTYLE order is MDY.

To get the default settings for each locale option value, see Locale Values.

**See Also**

Informs:
DEFLATION= System Option

Specifies the level of compression for device drivers that support the Deflate compression algorithm.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Alias: DEFLATE

Requirement: The UPRINTCOMPRESS system option must be set in order to compress files.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DEFLATION=n | MIN | MAX

Syntax Description

\( n \)

specifies the level of compression. The larger the number, the greater the compression. For example, \( n=0 \) is the minimum compression level (completely uncompressed), and \( n=9 \) is the maximum compression level.

Default: 6

Range: 0–9

MIN

specifies the minimum compression level of 0.

MAX

specifies the maximum compression level of 9.

Details

The DEFLATION= system option controls the level of compression for device drivers that support Deflate, such as PDF and SVG.

The ODS PRINTER statement option, COMPRESS=, takes precedence over the DEFLATION system option.
DETAILS System Option

Specifies whether to include additional information when files are listed in a SAS library.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

PROC OPTIONS GROUP=
LOG_LISTCONTROL
LISTCONTROL
LOGCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DETAILS | NODETAILS

Syntax Description

DETAILS
includes additional information when some SAS procedures and windows display a listing of files in a SAS library.

NODETAILS
does not include additional information.

Details

The DETAILS specification sets the default display for these components of SAS:

- the CONTENTS procedure
- the DATASETS procedure.

The type and amount of additional information that displays depends on which procedure or window you use.

See Also

**DEVICE= System Option**

Specifies the device driver to which SAS/GRAPH sends procedure output.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Graphics: Driver settings

**PROC OPTIONS GROUP=**

**Alias:** DEV=

**Requirement:** This option is required for the LISTING destination when you run SAS in a non-interactive mode.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.


**Syntax**

```
DEVICE=device-driver-specification
```

**Syntax Description**

- **device-driver-specification** specifies the name of a device driver.

**Details**

If you omit the device-driver name, the Output Delivery System selects a device driver for the opened destinations. If you specify a device that is not compatible with the opened destinations, the Output Delivery System selects a device that is valid. If you enter an invalid device driver, SAS prompts you to enter a device driver when you execute a procedure that produces graphics.

A best practice is to let the Output Delivery System select the device driver, except when you use the LISTING destination. When the LISTING destination is open, the default device is the Graph window.

**See Also**

Chapter 6, “Using Graphics Devices,” in SAS/GRAPH: Reference

**DKRICOND= System Option**

Specifies the level of error detection to report when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.
Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Files: SAS Files
          Environment control: Error handling

PROC OPTIONS
  GROUP= ERRORHANDLING
  SASFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DKRICOND=ERROR | WARN | WARNING | NOWARN | NOWARNING

Syntax Description

ERROR
  sets the error flag and writes an error message to the SAS log when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

WARN | WARNING
  writes a warning message to the SAS log when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

NOWARN | NOWARNING
  does not write a warning message to the SAS log when a variable is missing from an input data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

Example

In the following statements, if the variable X is not in data set B and DKRICOND=ERROR, SAS sets the error flag to 1 and displays error messages:

data a;
  set b(drop=x);
run;

See Also

System Options:
  • “DKROCOND= System Option” on page 109

DKROCOND= System Option

Specifies the level of error detection to report when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Files: SAS Files
Environment control: Error handling

**PROC OPTIONS**

**GROUP=** ERRORHANDLING

**SASFILES**

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

### Syntax

**DKROCOND=** ERROR | WARN | WARNING | NOWARN | NOWARNING

### Syntax Description

**ERROR**

sets the error flag and writes an error message to the SAS log when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

**WARN | WARNING**

writes a warning message to the SAS log when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

**NOWARN | NOWARNING**

does not write a warning message to the SAS log when a variable is missing for an output data set during the processing of a DROP=, KEEP=, or RENAME= data set option.

### Example

In the following statements if the variable X is not in data set A and DKROCOND=ERROR, SAS sets the error flag to 1 and displays error messages:

```sas
data a;
  drop x;
run;
```

### See Also

**System Options:**

- “DKRICOND= System Option” on page 108

---

**DLCREATEDIR System Option**

Specifies to create a directory for the SAS library that is named in a LIBNAME statement if the directory does not exist.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Files: SAS Files

**PROC OPTIONS GROUP=** SASFILES
DLCREATEDIR System Option

Syntax

DLCREATEDIR | NODLCREATEDIR

Syntax Description

DLCREATEDIR

specifies to create a directory for a SAS library that is named in a LIBNAME
statement if the directory does not exist.

NODLCREATEDIR

specifies not to create a directory for a SAS library that is named in a LIBNAME
statement. This is the default.

Details

SAS issues a note to the log when a directory for a SAS library is created.

See Also

Statements:

• “LIBNAME Statement” in SAS Statements: Reference

DLDMGACTION= System Option

Specifies the type of action to take when a SAS data set or a SAS catalog is detected as damaged.

Valid in:

Configuration file, SAS invocation, OPTIONS statement, SAS System Options
window

Category:

Files: SAS Files

PROC OPTIONS

GROUP=

SASFILES

Note:

This option cannot be restricted by a site administrator. For more information, see
“Restricted Options” on page 6.

Syntax

DLDMGACTION=FAIL | ABORT | REPAIR | NOINDEX | PROMPT

Syntax Description

FAIL

stops the step and issues an error message to the log immediately. This is the default
for batch mode.

ABORT

terminates the step and issues an error message to the log, and ends the SAS session.
REPAIR
For data files, automatically repairs and rebuilds indexes and integrity constraints, unless the data file is truncated. You use the REPAIR statement to restore the truncated data file. It issues a warning message to the log. This is the default for interactive mode. For catalogs, automatically deletes catalog entries for which an error occurs during the repair process.

NOINDEX
For data files, automatically repairs the data file without the indexes and integrity constraints, deletes the index file, updates the data file to reflect the disabled indexes and integrity constraints, and limits the data file to be opened only in INPUT mode. A warning is written to the SAS log instructing you to execute the PROC DATASETS REBUILD statement to correct or delete the disabled indexes and integrity constraints.

Restriction: NOINDEX does not apply to damaged catalogs or libraries, only data files.

See:
REBUILD Statement, Chapter 16, “DATASETS Procedure” in Base SAS Procedures Guide
“Recovering Disabled Indexes and Integrity Constraints” in Chapter 36 of SAS Language Reference: Concepts

PROMPT
For data sets, displays a dialog box where you can specify either FAIL, ABORT, REPAIR, or NOINDEX. For a damaged catalog or library, PROMPT displays a dialog box where you can specify either FAIL, ABORT, or REPAIR.

DMR System Option
Specifies whether to enable SAS to invoke a server session for use with a SAS/CONNECT client.

Valid in: Configuration file, SAS invocation
Category: Environment control: Initialization and operation
PROC OPTIONS GROUP=

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
DMR | NODMR

Syntax Description
DMR
enables you to invoke a remote SAS session in order to connect with a SAS/CONNECT client.

NODMR
disables you from invoking a remote SAS session.
Details

You normally invoke the remote SAS session from a local session by including DMR with the SAS command in a script that contains a TYPE statement. (A *script* is a text file that contains statements to establish or terminate the SAS/CONNECT link between the local and the remote SAS sessions.)

The OBJECTSERVER SAS execution mode invocation option has precedence over the DMR option. DMR overrides all other SAS execution mode invocation options. For information about invocation option precedence, see “Order of Precedence” on page 18.

See Also

DMR information in *SAS/CONNECT User's Guide*

---

**DMS System Option**

Specifies whether to invoke the SAS windowing environment and display the Log, Editor, and Output windows.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Environment control: Initialization and operation</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>EXECMODES</td>
</tr>
<tr>
<td>Note:</td>
<td>This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.</td>
</tr>
</tbody>
</table>

**Syntax**

DMS | NODMS

**Syntax Description**

DMS

invokes the SAS windowing environment and displays the Log, an Editor window, and Output windows.

NODMS

invokes an interactive line mode SAS session.

**Details**

When you invoke SAS and you are using a configuration file or the command line to control your system option settings, it is possible to create a situation where some system option settings conflict with other system option settings. The following invocation system options, in order, have precedence over the DMS invocation system option:

1. OBJECTSERVER.
2. DMR
3. SYSIN
If you specify DMR while using another invocation option of equal precedence to invoke SAS, SAS uses the last option that is specified. See “Order of Precedence” on page 18 for information about invocation option precedence.

See Also

System Options:

- “DMR System Option” on page 112
- “DMSEXP System Option” on page 114
- “EXPLORER System Option” on page 135

DMSEXP System Option

Specifies whether to invoke the SAS windowing environment and display the Explorer, Editor, Log, Output, and Results windows.

Valid in: Configuration file, SAS invocation
Category: Environment control: Initialization and operation
PROC OPTIONS GROUP= EXECMODES

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DMSEXP | NODMSEXP

Syntax Description

DMSEXP
 invokes SAS with the Explorer, Editor, Log, Output, and Results windows active.

NODMSEXP
 invokes SAS with the Editor, Log, and Output windows active.

Details

In order to set DMSEXP or NODMSEXP, the DMS option must be set. The following SAS execution mode invocation options, in order, have precedence over this option:

1. OBJECTSERVER.
2. DMR
3. SYSIN

If you specify DMSEXP with another execution mode invocation option of equal precedence, SAS uses only the last option listed. See “Order of Precedence” on page 18 for information about invocation option precedence.
DMSLOGSIZE= System Option

Specifies the maximum number of rows that the SAS Log window can display.

Valid in: Configuration file, SAS invocation

Categories:
- Environment control: Display
- Log and procedure output control: SAS log

PROC OPTIONS
GROUP= ENVDISPLAY
LOGCONTROL

Restriction: This option is valid only in the SAS windowing environment.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DMSLOGSIZE= n | nK | hexX | MIN | MAX

Syntax Description

n | nK
specifies the maximum number of rows that can be displayed in the SAS windowing environment Log window in multiples of 1 (n) or 1,024 (nK). For example, a value of 800 specifies 800 rows, and a value of 3K specifies 3,072 rows. Valid values range from 500 to 999999. The default is 99999.

hexX
specifies the maximum number of rows that can be displayed in the SAS windowing environment Log window as a hexadecimal value. You must specify the value beginning with a number (0-9), followed by an X. For example, 2ffx specifies 767 rows and 0A00x specifies 2,560 rows.

MIN
specifies to set the maximum number of rows that can be displayed in the SAS windowing environment Log window to 500.

MAX
specifies to set the maximum number of rows that can be displayed in the SAS windowing environment Log window to 999999.

Details

When the maximum number of rows have been displayed in the Log window, SAS prompts you to either file, print, save, or clear the Log window.

See Also

System Options:
- “DMS System Option” on page 113
- “DMR System Option” on page 112
- “EXPLORER System Option” on page 135
See Also

- “The SAS Log” in Chapter 9 of *SAS Language Reference: Concepts*

System Options:

- “DMSOUTSIZE= System Option” on page 116

---

**DMSOUTSIZE= System Option**

Specifies the maximum number of rows that the SAS Output window can display.

- **Valid in:** Configuration file, SAS invocation
- **Categories:** Environment control: Display, Log and procedure output control: Procedure output
- **PROC OPTIONS GROUP=** ENVDISPLAY, LISTCONTROL
- **Restriction:** This option is valid only in the SAS windowing environment.
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

### Syntax

\[ \text{DMSOUTSIZE=} n \mid nK \mid \text{hexX} \mid \text{MIN} \mid \text{MAX} \]

#### Syntax Description

- **n | nK**
  
specifies the maximum number of rows that can be displayed in the SAS windowing environment Output window in multiples of 1 (n) or 1,024 (nK). For example, a value of 800 specifies 800 rows, and a value of 3K specifies 3,072 rows. Valid values range from 500 to 999999. The default is 99999.

- **hexX**
  
specifies the maximum number of rows that can be displayed in the SAS windowing environment Output window as a hexadecimal value. You must specify the value beginning with a number (0-9), followed by an X. For example, \text{2ffX} specifies 767 rows and \text{0A00X} specifies 2,560 rows.

- **MIN**
  
specifies to set the maximum number of rows that can be displayed in the SAS windowing environment Output window to 500.

- **MAX**
  
specifies to set the maximum number of rows that can be displayed in the SAS windowing environment Output window to 999999.

#### Details

When the maximum number of rows have been displayed in the Output window, SAS prompts you to either file, print, save, or clear the Output window.
See Also

System Options:
- “DMSLOGSIZE= System Option” on page 115

DMSPGMLINESIZE= System Option

Specifies the maximum number of characters in a Program Editor line.

Valid in: Configuration file, SAS invocation
Category: Environment control: Display

PROC OPTIONS
GROUP= ENVDISPLAY

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DMSPGMLINESIZE= n

Syntax Description

n
specifies the maximum number of characters in a Program Editor line.
Default: 136
Range: 136–960

DMSSYNCHK System Option

In the SAS windowing environment, specifies whether to enable syntax check mode for DATA step and PROC step processing.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Environment control: Error handling

PROC OPTIONS
GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

DMSSYNCHK | NODMSSYNCHK
**Syntax Description**

**DMSSYNCHK**

enables syntax check mode for statements that are submitted within the SAS windowing environment.

**NODMSSYNCHK**

does not enable syntax check mode for statements that are submitted within the SAS windowing environment.

**Details**

If a syntax or semantic error occurs in a DATA step after the DMSSYNCHK option is set, then SAS enters syntax check mode, which remains in effect from the point where SAS encountered the error to the end of the code that was submitted. After SAS enters syntax mode, all subsequent DATA step statements and PROC step statements are validated.

While in syntax check mode, only limited processing is performed. For a detailed explanation of syntax check mode, see “Syntax Check Mode” in Chapter 8 of *SAS Language Reference: Concepts*.

**CAUTION:**

Place the OPTIONS statement that enables DMSSYNCHK before the step for which you want it to take effect. If you place the OPTIONS statement inside a step, then DMSSYNCHK will not take effect until the beginning of the next step.

If NODMSSYNCHK is in effect, SAS processes the remaining steps even if an error occurs in the previous step.

**Comparisons**

You use the DMSSYNCHK system option to validate syntax in an interactive session by using the SAS windowing environment. You use the SYNTAXCHECK system option to validate syntax in a non-interactive or batch SAS session. You can use the ERRORCHECK= option to specify the syntax check mode for the LIBNAME statement, the FILENAME statement, the %INCLUDE statement, and the LOCK statement in SAS/SHARE.

**See Also**

- Chapter 8, “Error Processing and Debugging,” in *SAS Language Reference: Concepts*

**System Options:**

- “ERRORCHECK= System Option” on page 131
- “SYNTAXCHECK System Option” on page 261

---

**DSNFERR System Option**

When a SAS data set cannot be found, specifies whether SAS issues an error message.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Environment control: Error handling
PROC OPTIONS
GROUP=ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
DSNFERR | NODSNFERR

Syntax Description

DSNFERR
specifies that SAS issue an error message and stop processing if a reference is made to a SAS data set that does not exist.

NODSNFERR
specifies that SAS ignore the error message and continue processing if a reference is made to a SAS data set that does not exist. The data set reference is treated as if _NULL_ had been specified.

Details

• DSNFERR is similar to the BYERR system option, which issues an error message and stops processing if the SORT procedure attempts to sort a _NULL_ data set.

• DSNFERR is similar to the VNFERR system option, which sets the error flag for a missing variable when a _NULL_ data set is used.

See Also

System Options:

• “BYERR System Option” on page 80
• “VNFERR System Option” on page 289

DTRESET System Option

Specifies whether to update the date and time in the SAS log and in the procedure output file.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

PROC OPTIONS
GROUP=LOG_LISTCONTROL
LOGCONTROL
LISTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

**DTRESET | NODTRESET**

**Syntax Description**

**DTRESET**

specifies that SAS update the date and time in the titles of the SAS log and the procedure output file.

**NODTRESET**

specifies that SAS not update the date and time in the titles of the SAS log and the procedure output file.

**Details**

The DTRESET system option updates the date and time in the titles of the SAS log and the procedure output file. This update occurs when the page is being written. The smallest time increment that is reflected is minutes.

The DTRESET option is especially helpful in obtaining a more accurate date and time stamp when you run long SAS jobs.

When you use NODTRESET, SAS displays the date and time that the job originally started.

**See Also**

“The SAS Log” in Chapter 9 of *SAS Language Reference: Concepts*

---

**DUPLEX System Option**

Specifies whether duplex (two-sided) printing is enabled.

**Valid in:**

Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP=** ODSPRINT

**Restriction:** This option is ignored if the printer does not support duplex (two-sided) printing.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

**DUPLEX | NODUPLEX**

**Syntax Description**

**DUPLEX**

specifies that duplex (two-sided) printing is enabled.

**Interaction:** When DUPLEX is selected, the setting of the BINDING= option determines how the paper is oriented before output is printed on the second side.
NODUPLEX
   specifies that duplex (two-sided) printing is not enabled. This is the default.

Details
Note that duplex (two-sided) printing can be used only on printers that support duplex output.

See Also
• “ODS PRINTER Statement” in SAS Output Delivery System: User’s Guide
• “Universal Printing” in Chapter 15 of SAS Language Reference: Concepts

System Options:
• “BINDING= System Option” on page 75

---

**ECHOAUTO System Option**

Specifies whether the statements in the AUTOEXEC= file are written to the SAS log as they are executed.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Log and procedure output control: SAS log</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>LOGCONTROL</td>
</tr>
</tbody>
</table>

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

ECHOAUTO | NOECHOAUTO

**Syntax Description**

**ECHOAUTO**
   specifies that the SAS statements in the AUTOEXEC= file are written to the SAS log as they are executed.
   **Requirement:** To print autoexec file statements in the SAS log, the SOURCE system option must be set.

**NOECHOAUTO**
   specifies that SAS statements in the AUTOEXEC= file are not written in the SAS log, even though they are executed.

**Details**
Regardless of the setting of this option, messages that result from errors in the AUTOEXEC= files are printed in the SAS log.

**See Also**
• “The SAS Log” in Chapter 9 of SAS Language Reference: Concepts
System Options:
• “SOURCE System Option” on page 240

EMAILAUTHPROTOCOL= System Option

Specifies the authentication protocol for SMTP e-mail.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
**Category:** Communications: Email
**PROC OPTIONS GROUP=** EMAIL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**
EMAILAUTHPROTOCOL= NONE | LOGIN

**Syntax Description**
LOGIN
specifies that the LOGIN authentication protocol is used.

**Note:** When you specify LOGIN, you might also need to specify EMAILID and EMAILPW. If you omit EMAILID, SAS will look up your user ID and use it. If you omit EMAILPW, no password is used.

**See:** For more information about the order of authentication, see “Sending E-Mail through SMTP” in Chapter 38 of SAS Language Reference: Concepts.

NONE
specifies that no authentication protocol is used.

**Details**
For the SMTP access method, use this option in conjunction with the EMAILID=, EMAILPW=, EMAILPORT, and EMAILHOST system options. EMAILID= provides the user name. EMAILPW= provides the password. EMAILPORT specifies the port to which the SMTP server is attached. EMAILHOST specifies the SMTP server that supports e-mail access for your site. EMAILAUTHPROTOCOL= provides the protocol.

**See Also**

System Options:
• “EMAILHOST= System Option” on page 123
• “EMAILID= System Option” on page 124
• “EMAILPORT System Option” on page 126
• “EMAILPW= System Option” on page 127
EMAILFROM System Option
When sending e-mail by using SMTP, specifies whether the e-mail option FROM is required in either the FILE or FILENAME statement.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Communications: Email
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

EMAILFROM | NOEMAILFROM

Syntax Description

EMAILFROM
specifies that the FROM e-mail option is required when sending e-mail by using either the FILE or FILENAME statements.

NOEMAILFROM
specifies that the FROM e-mail option is not required when sending e-mail by using either the FILE or FILENAME statements.

See Also

Statements:
• “FILE Statement” in SAS Statements: Reference
• “FILENAME Statement, EMAIL (SMTP) Access Method” in SAS Statements: Reference

EMAILHOST= System Option
Specifies one or more SMTP servers that support e-mail access.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Communications: Email
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

EMAILHOST= server

EMAILHOST=( 'server-1' 'server-2' <... 'server-n' > )

Syntax Description

server

specifies one or more Simple Mail Transfer Protocol (SMTP) server domain names for your site.

Range: The maximum number of characters that can be specified for SMTP servers is 1,024

Requirement: When more than one server name is specified, the list must be enclosed in parentheses and each server name must be enclosed in single or double quotation marks..

Note: The system administrator for your site will provide this information.

Details

When more than one SMTP server is specified, SAS attempts to connect to e-mail servers in the order in which they are specified. E-mail is delivered to the first server that SAS connects to. If SAS is not able to connect to any of the specified servers, the attempt to deliver e-mail fails and SAS returns an error.

Operating Environment Information

To enable the SMTP interface that SAS provides, you must also specify the EMAILSYS=SMTP system option. For information about EMAILSYS, see the documentation for your operating environment.

Comparisons

For the SMTP access method, use this option in conjunction with the EMAILID=, EMAILAUTHPROTOCOL=, EMAILPORT, and EMAILPW system options. EMAILID= provides the user name. EMAILPW= provides the password. EMAILPORT specifies the port to which the SMTP server is attached. EMAILHOST specifies SMTP servers that supports e-mail access for your site. EMAILAUTHPROTOCOL= provides the protocol.

See Also

System Options:

- “EMAILAUTHPROTOCOL= System Option” on page 122
- “EMAILID= System Option” on page 124
- “EMAILPORT System Option” on page 126
- “EMAILPW= System Option” on page 127

EMAILID= System Option

Identifies an e-mail sender by specifying either a logon ID, an e-mail profile, or an e-mail address.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Syntax

EMAILID = logonid | profile | email-address

Syntax Description

logonid

specifies the logon ID for the user running SAS.

Note: The maximum number of characters is 32,000.

profile

see documentation for your e-mail system to determine the profile name.

e-mail-address

specifies the fully qualified e-mail address of the user running SAS.

Requirements:

The e-mail address is valid only when SMTP is enabled.

If the value of email-address contains a space, you must enclose it in double quotation marks.

Details

The EMAILID= system option specifies the logon ID, profile, or e-mail address to use with your e-mail system.

Comparisons

For the SMTP access method, use this option in conjunction with the EMAILAUTHPROTOCOL=, EMAILPW=, EMAILPORT, and EMAILHOST system options. EMAILID= provides the user name. EMAILPW= provides the password. EMAILPORT specifies the port to which the SMTP server is attached. EMAILHOST specifies the SMTP server that supports e-mail access for your site. EMAILAUTHPROTOCOL= provides the protocol.

See Also

System Options:

- “EMAILAUTHPROTOCOL= System Option” on page 122
- “EMAILHOST= System Option” on page 123
- “EMAILPORT System Option” on page 126
- “EMAILPW= System Option” on page 127
EMAILPORT System Option

Specifies the port that the SMTP server is attached to.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Communications: Email

PROC OPTIONS
GROUP= EMAIL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

EMAILPORT <port-number>

Syntax Description

port-number

specifies the port number that is used by the SMTP server that you specified on the EMAILHOST option.

Note: The system administrator for your site will provide this information.

Details

Operating Environment Information

If you use the SMTP protocol that SAS provides, you must also specify the EMAILSYS SMTP system option. For information about EMAILSYS, see the documentation for your operating environment.

Comparisons

For the SMTP access method, use this option in conjunction with the EMAILID=, EMAILAUTHPROTOCOL=, EMAILPW=, and EMAILHOST system options. EMAILID= provides the user name. EMAILPW= provides the password. EMAILPORT specifies the port to which the SMTP server is attached. EMAILHOST specifies the SMTP server that supports e-mail access for your site. EMAILAUTHPROTOCOL= provides the protocol.

See Also

System Options:

- “EMAILAUTHPROTOCOL= System Option” on page 122
- “EMAILHOST= System Option” on page 123
- “EMAILID= System Option” on page 124
- “EMAILPW= System Option” on page 127
EMAILPW= System Option

Specifies an e-mail logon password.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options
Category: Communications: Email
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see
"Restricted Options" on page 6.

Syntax

EMAILPW= "password"

Syntax Description

password specifies the logon password for your logon name.

Restriction: If “password” contains a space, you must enclose the value in double
quotation marks.

Details

You can use encoded e-mail passwords. When a password is encoded with PROC
PWENCODE, the output string includes a tag that identifies the string as having been
encoded. An example of a tag is {sas001}. The tag indicates the encoding method.
Encoding a password enables you to avoid e-mail access authentication with a password
in plaintext. Passwords that start with "{sas" trigger an attempt to be decoded. If the
decoding succeeds, then that decoded password is used. If the decoding fails, then the
password is used as is. For more information, see Chapter 3, “PWENCODE Procedure”
in Encryption in SAS.

Windows Specifics

In the Windows operating system, SAS prompts you for an e-mail ID and a password
if the EMAILSYS system option is set to MAPI or VIM if you do not specify the
EMAILID and EMAILPW system options at invocation, or if you are not otherwise
logged in to your e-mail system. If the EMAILSYS system option is set to SMTP,
SAS will not prompt you for an e-mail ID and a password.

Comparisons

For the SMTP access method, use this option in conjunction with the EMAILID=,
EMAILAUTHPROTOCOL=, EMAILPORT, and EMAILHOST system options.
EMAILID= provides the user name. EMAILPW= provides the password. EMAILPORT
specifies the port to which the SMTP server is attached. EMAILHOST specifies the
SMTP server that supports e-mail access for your site. EMAILAUTHPROTOCOL= provides the protocol.
EMAILUTCOFFSET= System Option

For e-mail that is sent using the FILENAME statement EMAIL (SMTP) access method, specifies a UTC offset that is used in the Date header field of the e-mail message.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options Window

**Category:** Communications: E-mail

**PROC OPTIONS GROUP=** EMAIL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

EMAILUTCOFFSET="*hhmm*" | "-hhmm"

**Syntax Description**

"*hhmm*" | "-hhmm"

specifies the number of hours and minutes that are used as the UTC offset in the Date header field of an e-mail. Use the UTC offset to establish the local time.

**Requirement:** The value of the EMAILUTCOFFSET= system option must be enclosed in double or single quotation marks.

**Details**

If your computer time setting is a local time, or if your computer time setting does not account for Daylight Saving Time, you can use the EMAILUTCOFFSET= system option to set a UTC offset in the Date header field of an SMTP e-mail. If the Date header field contains a UTC offset, the value specified by the EMAILUTCOFFSET= system option replaces that UTC offset.

**Example**

This example uses the date of January 1, 2011, one hour, one minute, and one second after midnight:
See Also


Statements:

- “FILENAME Statement, EMAIL (SMTP) Access Method” in SAS Statements: Reference

**ENGINE= System Option**

Specifies the default access method for SAS libraries.

Valid in: Configuration file, SAS invocation

Category: Files: SAS Files

PROC OPTIONS GROUP=

SASFILES

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See:

“ENGINE= System Option: UNIX” in SAS Companion for UNIX Environments,

“ENGINE System Option: Windows” in SAS Companion for Windows,

“ENGINE= System Option: z/OS” in SAS Companion for z/OS

**Syntax**

`ENGINE=engine-name`

**Syntax Description**

`engine-name`

specifies an engine name.

**Details**

The ENGINE= system option specifies which default engine name is associated with a SAS library. The default engine is used when a SAS library points to an empty directory or a new file. The default engine is also used on directory-based systems, which can store more than one SAS file type within a directory. For example, some operating environments can store SAS files from multiple versions in the same directory.

**Operating Environment Information**

Valid engine names depend on your operating environment. For details, see the SAS documentation for your operating environment.
ERRORABEND System Option

Specifies whether SAS responds to errors by terminating.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Environment control: Error handling

**PROC OPTIONS GROUP=** ERRORHANDLING

**Alias:** ERRABEND | NOERRABEND

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

ERRORABEND | NOERRORABEND

**Syntax Description**

**ERRORABEND**

specifies that SAS terminate for most errors (including syntax errors and file not found errors) that would normally cause it to issue an error message, set OBS=0, and go into syntax-check mode (if syntax checking is enabled). SAS also terminates if an error occurs in any global statement other than the LIBNAME and FILENAME statements.

**Tip:** Use the ERRORABEND system option with SAS production programs, which presumably should not encounter any errors. If errors are encountered and ERRORABEND is in effect, SAS brings the errors to your attention immediately by terminating. ERRORABEND does not affect how SAS handles notes such as invalid data messages.

**NOERRORABEND**

specifies that SAS handle errors normally, that is, issue an error message, set OBS=0, and go into syntax-check mode (if syntax checking is enabled).

**See Also**

- “Global Statements” in Chapter 1 of *SAS Statements: Reference*

**System Options:**

- “ERRORBYABEND System Option” on page 131
- “ERRORCHECK= System Option” on page 131
ERRORBYABEND System Option

Specifies whether SAS ends a program when an error occurs in BY-group processing.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

ERRORBYABEND | NOERRORBYABEND

Syntax Description

ERRORBYABEND
specifies that SAS ends a program for BY-group error conditions that would normally cause it to issue an error message.

NOERRORBYABEND
specifies that SAS handle BY-group errors normally, that is, by issuing an error message and continuing processing.

Details

If SAS encounters one or more BY-group errors while ERRORBYABEND is in effect, SAS brings the errors to your attention immediately by ending your program. ERRORBYABEND does not affect how SAS handles notes that are written to the SAS log.

Note: Use the ERRORBYABEND system option with SAS production programs that should be error free.

See Also

System Options:

• “ERRORABEND System Option” on page 130

ERRORCHECK= System Option

Specifies whether SAS enters syntax-check mode when errors are found in the LIBNAME, FILENAME, %INCLUDE, and LOCK statements.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Error handling
PROC OPTIONS
GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

ERRORCHECK= NORMAL | STRICT

Syntax Description

NORMAL
specifies not to place the SAS program into syntax-check mode when an error occurs in a LIBNAME or FILENAME statement, or in a LOCK statement in SAS/SHARE software. In addition, the program or session does not terminate when a %INCLUDE statement fails due to a non-existent file.

STRICT
specifies to place the SAS program into syntax-check mode when an error occurs in a LIBNAME or FILENAME statement, or in a LOCK statement in SAS/SHARE software. If the ERRORABEND system option is set and an error occurs in either a LIBNAME or FILENAME statement, SAS terminates. In addition, SAS terminates when a %INCLUDE statement fails due to a non-existent file.

See Also

System Options:

- “ERRORABEND System Option” on page 130

ERRORS= System Option

Specifies the maximum number of observations for which SAS issues complete error messages.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Environment control: Error handling
Log and procedure output control: SAS log

Syntax

ERRORS= n | nK | nM | nG | nT | MIN | MAX | hexX
Syntax Description

\[ n \mid nK \mid nM \mid nG \mid nT \]

specifies the number of observations for which SAS issues error messages in terms of 1 (n); 1,024 (nK); 1,048,576 (nM); 1,073,741,824 (nG); or 1,099,511,627,776 (nT). For example, a value of 8 specifies eight observations, and a value of 3M specifies 3,145,728 observations.

MIN

sets the number of observations for which SAS issues error messages to 0.

MAX

sets the maximum number of observations for which SAS issues error messages to the largest signed, 4-byte integer representable in your operating environment.

\[ \text{hexX} \]

specifies the maximum number of observations for which SAS issues error messages as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx sets the maximum number of observations for which SAS issues error messages to 45 observations.

Details

If data errors are detected in more than \( n \) observations, processing continues, but SAS does not issue error messages for the additional errors.

Note: If you set ERRORS=0 and an error occurs, or if the maximum number of errors has been reached, a warning message displays in the log which states that the limit set by the ERRORS option has been reached.

See Also


EVENTDS= System Option

Specifies one or more data sets that define events.

| Valid in: | Configuration file, SAS invocation, OPTIONS statement, SAS System Options window |
| Category: | Input control: Data Processing |
| PROC OPTIONS GROUP= | INPUTCONTROL |

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

\[ \text{EVENTDS=}(<\text{DEFAULTS | NODEFAULTS}> \text{ event-data-set-1 <…event-data-set-n>}) \]

Syntax Description

DEFAULTS

specifies to use the SAS predefined holiday events. This is the default.
NODEFAULTS
specifies not to use default event definitions. The only events that are used are specified by the event-data-set list.

event-data-set
specifies the name of a data set that contains event definitions. The data set can be specified as a one-level name, dataset, or a two-level name, libref.dataset.

Details
A SAS event is used to model any incident that disrupts the normal flow of the process that generated the time series. Examples of commonly used events include natural disasters, retail promotions, strikes, advertising campaigns, policy changes, and data recording errors. You can create your own set of events, or you can use events that are predefined by SAS.

The events that are defined in the event-data-set list appear in the Events Repository in SAS Forecast Studio. In SAS High-Performance Forecasting, you can use an event data set as a value for the INEVENT= option in the HPFDIAGNOSE and HPFENGINE procedures. In SAS/ETS, an event data set can be used by the INEVENT= option in the X12 procedure.

When you set the EVENTDS= option, all values that were specified by a previous EVENTDS= option are replaced. A new specification of the option is not appended to the value of the existing value. You can use this option to override the default events or to add or delete event data sets.

To create event data sets, see the HPFEVENTS procedure in the SAS High-Performance Forecasting User’s Guide.

Examples

Example 1
By default, the SAS predefined holiday events are valid events as long as NODEFAULTS is not specified in the EVENTDS= option. The valid events that are set using the following EVENTDS= option are the SAS predefined holiday events and the events that are specified in the events.WorldCup data set:

options eventds=(events.WorldCup);

Example 2
Because NODEFAULTS is specified, after the EVENTDS= option is set, the SAS predefined holiday events are not valid events. The only valid events are the events that are defined in the dubai_holidays data set:

options eventds=(nodefaults dubai_holidays);

Example 3
Reset the valid events to the predefined list of SAS holiday events:

options eventds=(defaults);

See Also
• SAS/ETS User’s Guide
• SAS Forecast Studio User’s Guide
EXPLORER System Option

Specifies whether to invoke the SAS windowing environment and display only the Explorer and Program Editor windows.

**Valid in:** Configuration file, SAS invocation  
**Category:** Environment control: Initialization and operation  
**PROC OPTIONS GROUP:** EXECMODES

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```plaintext
EXPLORER | NOEXPLORER
```

**Syntax Description**

- **EXPLORER** specifies that the SAS session be invoked with only the Explorer and Program Editor windows.
- **NOEXPLORER** specifies that the SAS session be invoked without the Explorer window.

**Details**

The following SAS execution mode invocation options, in order, have precedence over this option:

1. OBJECTSERVER.
2. DMR
3. SYSIN

If you specify EXPLORER with another execution mode invocation option of equal precedence, SAS uses only the last option listed. See “Order of Precedence” on page 18 for more information about invocation option precedence.

**See Also**

- “DMS System Option” on page 113
- “DMSEXP System Option” on page 114
FILESYNC= System Option

Specifies when operating system buffers that contain contents of permanent SAS files are written to disk.

Valid in: Configuration file, SAS invocation
Category: Files: SAS Files

PROC OPTIONS
GROUP=SASFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “FILESYNC= System Option: z/OS” in SAS Companion for z/OS

Syntax

FILESYNC= SAS | CLOSE | HOST | SAVE

Syntax Description

SAS
specifies that SAS requests the operating system to force buffered data to be written to disk when it is best for the integrity of the SAS file.

CLOSE
specifies that SAS requests the operating system to force buffered data to be written to disk when the SAS file is closed.

HOST
specifies that the operating system schedules when the buffered data for a SAS file is written to disk. This is the default.

SAVE
specifies that the buffers are written to disk when the SAS file is saved.

Details

By using the FILESYNC= system option, SAS can tell the operating system when to force data that is temporarily stored in operating system buffers to be written to disk. Only SAS files in a permanent SAS library are affected; files in a temporary library are not affected.

If you specify a value other than the default value of HOST, the following occurs:

- the length of time it takes to run a SAS job increases
- the small chance of losing data in the event of a system failure is further reduced

Consult with your system administrator before you change the value of the FILESYNC= system option to a value other than the default value.

z/OS Specifics

Under z/OS, the FILESYNC= system option affects SAS files only in UNIX file system (UFS) libraries. For more information, see “FILESYNC= System Option: z/OS” in SAS Companion for z/OS.
FIRSTOBS= System Option

Specifies the observation number or external file record that SAS processes first.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Files: SAS Files

PROC OPTIONS GROUP=

Interaction: When you specify the FIRSTOBS= option and EXTENDOBSCOUNTER=YES is set either as a data set option or as a LIBNAME option, data sets that have 2G–1 observations or more might perform better in a 32-bit environment. For more information, see “Extending the Observation Count in a SAS Data File” in Chapter 26 of SAS Language Reference: Concepts.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

FIRSTOBS= n | nK | nM | nG | nT | hexX | MIN | MAX

Syntax Description

n | nK | nM | nG | nT

specifies the number of the first observation or external file record to process, with n being an integer. Using one of the letter notations results in multiplying the integer by a specific value. That is, specifying K (kilo) multiplies the integer by 1,024; M (mega) multiplies by 1,048,576; G (giga) multiplies by 1,073,741,824; or T (tera) multiplies by 1,099,511,627,776. For example, a value of 8 specifies the eighth observations or records, and a value of 3m specifies observation or record 3,145,728.

hexX

specifies the number of the first observation or the external file record to process as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx specifies the 45th observation.

MIN

sets the number of the first observation or external file record to process to 1. This is the default.

MAX

sets the number of the first observation to process to the maximum number of observations in the data sets or records in the external file, up to the largest eight-byte, signed integer, which is 2^31-1, or approximately 9.2 quintillion observations.

Details

The FIRSTOBS= system option is valid for all steps for the duration of your current SAS session or until you change the setting. To affect any single SAS data set, use the FIRSTOBS= data set option.
You can apply FIRSTOBS= processing to WHERE processing. For details, see “Processing a Segment of Data That Is Conditionally Selected” in Chapter 11 of *SAS Language Reference: Concepts*.

**Comparisons**

- You can override the FIRSTOBS= system option by using the FIRSTOBS= data set option and by using the FIRSTOBS= option as a part of the INFILE statement.
- While the FIRSTOBS= system option specifies a starting point for processing, the OBS= system option specifies an ending point. The two options are often used together to define a range of observations or records to be processed.

**Example**

If you specify FIRSTOBS=50, SAS processes the 50th observation of the data set first. This option applies to every input data set that is used in a program or a SAS process. In this example, SAS begins reading at the 11th observation in the data sets OLD, A, and B:

```sas
options firstobs=11;
data a;
  set old; /* 100 observations */
run;
data b;
  set a;
run;
data c;
  set b;
run;
```

Data set OLD has 100 observations, data set A has 90, B has 80, and C has 70. To avoid decreasing the number of observations in successive data sets, use the FIRSTOBS= data set option in the SET statement. You can also reset FIRSTOBS=1 between a DATA step and a PROC step.

**See Also**

**Data Set Options:**
- “FIRSTOBS= Data Set Option” in *SAS Data Set Options: Reference*

**Statements:**
- “INFILE Statement” in *SAS Statements: Reference*

**System Options:**
- “OBS= System Option” on page 183

---

**FMTERR System Option**

When a variable format cannot be found, specifies whether SAS generates an error or continues processing.
```
PROC OPTIONS GROUP=ERRORHANDLING
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
FMTERR | NOFMTERR

Syntax Description
FMTERR
specifies that when SAS cannot find a specified variable format, it generates an error message and does not allow default substitution to occur.

NOFMTERR
replaces missing formats with the w. or $w. default format, issues a note, and continues processing.

See Also
System Options:
• “FMTSEARCH= System Option” on page 139

FMTSEARCH= System Option
Specifies the order in which format catalogs are searched.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Environment control: Files
PROC OPTIONS GROUP=
Requirement: Catalog specifications must be separated by a space.
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Tip: You can use the APPEND or INSERT system options to add additional catalog-specification.

Syntax
FMTSEARCH=(catalog-specification-1... catalog-specification-n)
```
**Syntax Description**

`catalog-specification`  
searches format catalogs in the order listed, until the desired member is found.

The value of `catalog-specification` can be one of the following:

`libref</LOCALE>`

specifies to search the FORMATS catalog in the storage location that is specified by `libref`. When a libref is specified without a catalog, SAS uses FORMATS as the default catalog name.

If you specify /LOCALE, SAS searches for a catalog that is associated with the current SAS locale before it searches the FORMATS catalog. The locale catalog name is based on the POSIX locale name for the current locale. Two catalogs might exist for each POSIX locale name: one catalog for the language and one catalog for the language_country. If your current SAS locale is English_India, the POSIX locale name is en_IN. The two possible locale catalogs are `librefIFORMATS_en` and `librefIFORMATS_en_IN`. SAS searches, in order, these catalogs in `libref`:

1. `librefIFORMATS_language_country`
2. `librefIFORMATS_language`
3. `librefIFORMATS`

**Tip:** You can obtain a POSIX locale value by using the GETPXLOCALE function. You can use the GETLOCENV function to obtain the current SAS locale. For more information, see *SAS National Language Support (NLS): Reference Guide*.

**See:** For a list of POSIX locale values and their corresponding SAS locale names, see “LOCALE= Values and Default Settings for ENCODING, PAPERSIZE, DFLANG, and DATESTYLE Options” in Chapter 18 of *SAS National Language Support (NLS): Reference Guide*.

`libref.catalog</LOCALE>`

specifies to search for a specific library and catalog.

If you specify /LOCALE, SAS searches `libref.catalog` for a catalog that is associated with the current SAS locale. The locale catalog name is based on the POSIX locale name for the current locale. Two catalogs might exist for each POSIX locale name: one catalog for the language and one catalog for the language_country. If your current SAS locale is English_India, the POSIX locale name is en_IN. The two possible locale catalogs are `libref.catalog_en` and `libref.catalog_en_IN`. SAS searches, in order, these catalogs in `libref`, if you specify /LOCALE:

1. `libref.catalog_language_country`
2. `libref.catalog_language`
3. `libref.catalog`

**Tip:** You can obtain a POSIX locale value by using the GETPXLOCALE function. You can use the GETLOCENV function to obtain the current SAS locale. For more information, see *SAS National Language Support (NLS): Reference Guide*.

**See:** For a list of POSIX locale values and their corresponding SAS locale names, see “LOCALE= Values and Default Settings for ENCODING, PAPERSIZE, DFLANG, and DATESTYLE Options” in Chapter 18 of *SAS National Language Support (NLS): Reference Guide*. 
Details

The default value for FMTSEARCH is (WORK LIBRARY). The catalogs WORK.FORMATS and LIBRARY.FORMATS are always searched, whether or not they appear in the search list. The WORK.FORMATS catalog is always searched first, and the LIBRARY.FORMATS catalog is searched next, unless one of the catalogs appears in the FMTSEARCH= list.

For example, FMTSEARCH=(MYLIB LIBRARY) results in searching these catalogs, in the order MYLIB.FORMATS, LIBRARY.FORMATS, and WORK.FORMATS.

If a catalog appears in the FMTSEARCH= list, the catalog is searched in the order in which it appears in the list. If a catalog in the list does not exist, that particular catalog is ignored and searching continues.

Examples

Example 1: Format Catalog Search Order with Default Libraries Searched First
If you specify FMTSEARCH=(ABC DEF.XYZ GHI), SAS searches for requested formats or informats in this order:

1. WORK.FORMATS
2. LIBRARY.FORMATS
3. ABC.FORMATS
4. DEF.XYZ
5. GHI.FORMATS.

Example 2: Format Catalog Search Order with Default Libraries Searched Last
If you specify FMTSEARCH=(ABC WORK LIBRARY) SAS searches in this order:

1. ABC.FORMATS
2. WORK.FORMATS
3. LIBRARY.FORMATS.

Because WORK appears in the FMTSEARCH list, WORK.FORMATS is not automatically searched first.

Example 3: Format Catalog Search Order with POSIX Locale Values
If you specify FMTSEARCH=(ABC/LOCALE) and the current locale is German_Germany, SAS searches in this order:

1. WORK.FORMATS
2. LIBRARY.FORMATS
3. ABC.FORMATS_de_DE
4. ABC.FORMATS_de
5. ABC.FORMATS
FONTEMBEDDING System Option

Specifies whether font embedding is enabled in Universal Printer and SAS/GRAPH printing.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP=ODSPRINT

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

FONTEMBEDDING | NOFONTEMBEDDING

Syntax Description

FONTEMBEDDING
specifies to enable font embedding. This is the default.

NOFONTEMBEDDING
specifies to disable font embedding.

Details

Font embedding is used mainly by Universal Printing. Not all printers support font embedding. To determine whether the printer that you are using supports font embedding, use the QDEVICE procedure. If Font Embedding is listed in the SAS log, the printer supports font embedding. Here is a partial log output from the QDEVICE procedure:

```sas
369  proc qdevice report=general;
370     printer pdf;
371  run;
```

Name: PDF
Description: Portable Document Format Version 1.4
Type: Universal Printer
Registry: SASHELP
When FONTEMBERBEDDING is set, fonts can be embedded, or included, in the output files that are created by the Universal Printer and SAS/GRAPH. Output files with embedded fonts do not rely on fonts being installed on the computer that is used to view or print the output file. File size is increased for vector output for printers such as PDF and PostScript.

When NOFONTEMBERBEDDING is set, the output files rely on the fonts being installed on the computer that is used to view or print the font. If a font is not found on the computer, the printer or the application that displays the output might perform font substitution. Image output is not affected when NOFONTEMBERBEDDING is set.

To determine which fonts will be substituted for a given printer, use the Print Setup window or the QDEVICE procedure to display the Printer Setup properties. Under Fonts, any individual fonts that are listed will be recognized by the printer. All other fonts, including those that are available via a link in the SAS Registry, will be substituted in the document when the document is created.

See Also

• SAS/GRAPH: Reference
• “Universal Printing” in Chapter 15 of SAS Language Reference: Concepts

FONTEMBERBEDDING= System Option

Specifies whether SAS/GRAPH devices that are based on the SASGDGIF, SASGDTIF, and SASGDIMG modules render fonts by using the operating system or by using the FreeType engine.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS GROUP= ODSPRINT

Restriction: This option is set to HOST_PIXELS for devices that begin with “Z”.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

FONTRENDERING=HOST_PIXELS | FREETYPE_POINTS

Syntax Description

HOST_PIXELS

specifies that fonts are rendered by the operating system and that font size is requested in pixels.

z/OS specifics: On z/OS, HOST_PIXELS is not supported. If HOST_PIXELS is specified, SAS uses FREETYPE_POINTS as the value for this option.

FREETYPE_POINTS

specifies that fonts are rendered by the FreeType engine and that font size is requested in points. This is the default.

Details

Use the FONTRENDERING= system option to specify how SAS/GRAPH devices that are based on the SASGDGIF, SASGDTIF, and SASGDIMG modules render fonts. When the operating system renders fonts, the font size is requested in pixels. When the FreeType engine renders fonts, the font size is requested in points.

Use the GDEVICE procedure to determine which module a SAS/GRAPH device uses:

proc gdevice c=sashelp.devices browse nofs;
  list devicename;
quit;

For example,

proc gdevice c=sashelp.devices browse nofs;
  list gif;
quit;

The following is partial output from the GDEVICE procedure output:

```
GDEVICE procedure
Listing from SASHELP.DEVICES - Entry
GIF
Orig Driver: GIF Module: SASGDGIF Model: 6031
Description: GIF File Format
*** Institute-supplied ***
Type: EXPORT
Lrows: 43 Xmax: 8.333 IN Hsize: 0.000 IN Xpixels: 800
Lcols: 88 Ymax: 6.250 IN Vsize: 0.000 IN Ypixels: 600
Prows: 0 Horigin: 0.000 IN
Pcols: 0 Vorigin: 0.000 IN
Aspect: 0.000 Rotate: 
Driver query: Y Queued messages: N
Paperfeed: 0.000 IN
```

The Module entry names the module used by the device.

See Also

Chapter 13, “Specifying Fonts in SAS/GRAPH Programs,” in SAS/GRAPH: Reference
**FONTSLOC= System Option**

Specifies the location of the fonts that are supplied by SAS; names the default font file location for registering fonts that use the FONTREG procedure.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Environment control: Display

**PROC OPTIONS GROUP= ENVDISPLAY**

- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
- **See:**
  - “FONTSLOC System Option: UNIX” in SAS Companion for UNIX Environments
  - “FONTSLOC System Option: Windows” in SAS Companion for Windows
  - “FONTSLOC= System Option: z/OS” in SAS Companion for z/OS

**Syntax**

**FONTSLOC= “location”**

**Syntax Description**

- **“location”** specifies a fileref or the location of the SAS fonts that are used during the SAS session.
  - **Note:** If “location” is a fileref, you do not need to enclose the value in quotation marks.

**FORMCHAR= System Option**

Specifies the default output formatting characters.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: Procedure output

**PROC OPTIONS GROUP= LISTCONTROL**

- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
- **See:** “FORMCHAR System Option: Windows” in SAS Companion for Windows

**Syntax**

**FORMCHAR= 'formatting-characters'**
Syntax Description

'formatting-characters'
specifies any string or list of strings of characters up to 64 bytes long. If fewer than 64 bytes are specified, the string is padded with blanks on the right.

Tip: For consistent results when you move your document to different computers, issue the following OPTIONS statement before using ODS destinations other than the Listing destination:

```options formchar"|----|+|---+=-/-\<>*";```

Details

Formatting characters are used to construct tabular output outlines and dividers for various procedures, such as the FREQ, REPORT, and TABULATE procedures. If you omit formatting characters as an option in the procedure, the default specifications given in the FORMCHAR= system option are used. Note that you can also specify a hexadecimal character constant as a formatting character. When you use a hexadecimal constant with this option, SAS interprets the value of the hexadecimal constant as appropriate for your operating system.

Note: To ensure that row and column separators and boxed tabular reports are printed legibly when using the standard forms characters, you must use these resources:

- either the SAS Monospace or the SAS Monospace Bold font
- a printer that supports TrueType fonts

See Also

- For information about how Base SAS procedures use formatting characters, see Base SAS Procedures Guide. For procedures in other products that use formatting characters, see the documentation for that product.

FORMDLIM= System Option

Specifies a character to delimit page breaks in SAS output for the LISTING destination.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LISTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

`FORMDLIM= 'delimiting-character'`
Syntax Description

'delimiting-character'

specifies in quotation marks a character written to delimit pages. Normally, the delimit character is null, as in this statement:

```plaintext
options formdlim='';
```

Details

When the delimit character is null, a new physical page starts whenever a new page occurs. However, you can conserve paper by allowing multiple pages of output to appear on the same page. For example, this statement writes a line of hyphens (- -) where normally a page break would occur:

```plaintext
options formdlim='--';
```

When a new page is to begin, SAS skips a single line, writes a line consisting of the hyphens that are repeated across the page, and skips another single line. There is no skip to the top of a new physical page. Resetting FORMDLIM= to null causes physical pages to be written normally again.

FORMS= System Option

If forms are used for printing, specifies the default form to use.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Categories:**
- Environment control: Display
- Log and procedure output control: Procedure output

**PROC OPTIONS GROUP=**
- ENVDISPLAY
- LISTCONTROL

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

```plaintext
FORMS=form-name
```

Syntax Description

**form-name**

specifies the name of the form.

**Tip:** To create a customized form, use the FSFORM command in a windowing environment.

Details

The default form contains settings that control various aspects of interactive windowing output, including printer selection, text body, and margins. The FORMS= system option also customizes output from the PRINT command (when FORM= is omitted) or output from interactive windowing procedures.
GSTYLE System Option

Specifies whether ODS styles can be used to generate graphs that are stored as GRSEG catalog entries.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Graphics: Driver settings
Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP=GRAPHICS
ODSPRINT

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

GSTYLE | NOGSTYLE

Syntax Description

GSTYLE specifies that ODS styles can be used in the generation of graphs that are stored as GRSEG catalog entries. If no style is specified, the default style for the given output destination is used. This is the default.

NOGSTYLE specifies to not use ODS styles in the generation of graphs that are stored as GRSEG catalog entries.

Tip: Use NOGSTYLE for compatibility of graphs generated before SAS 9.2.

Details

The GSTYLE system option affects only graphic output that is generated using GRSEGS. The GSTYLE option does not affect the use of ODS styles in graphs that are generated by the following means:

- Java device driver
- ActiveX device driver
- SAS/GRAPH statistical graphic procedures
- SAS/GRAPH template language
- ODS GRAPHICS ON statement

GWINDOW System Option

Specifies whether SAS displays SAS/GRAPH output in the GRAPH window.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Graphics: Driver settings
PROC OPTIONS  
GROUP=  

GRAPHICS

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

GWINDOW | NOGWINDOW

Syntax Description

GWINDOW  

displays SAS/GRAPH software output in the GRAPH window, if your site licenses SAS/GRAPH software and if your personal computer has graphics capability.

NOGWINDOW  

displays graphics outside of the windowing environment.

HELPBROWSER= System Option

Specifies the browser to use for SAS Help and ODS output.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Help

PROC OPTIONS GROUP=

HELP

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

HELPBROWSER=REMOTE | SAS

Syntax Description

REMOTE  

specifies to use the remote browser for the Help. The location of the remote browser is determined by the HELPHOST and the HELPPORT system options. This is the default value for the OpenVMS, UNIX, z/OS, and Windows 64-bit operating environments.

SAS  

specifies to use the SAS browser for the Help. This is the default for the Windows 32-bit operating environment.

See Also

• Chapter 6, “Viewing Output and Help in the SAS Remote Browser,” in SAS Companion for UNIX Environments
HELPENCMD System Option

Specifies whether SAS uses the English version or the translated version of the keyword list for the command-line Help.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Environment control: Help</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>HELP</td>
</tr>
</tbody>
</table>

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

HELPENCMD | NOHELPENCMD

Syntax Description

HELPENCMD

specifies that SAS use the English version of the keyword list for the command-line help, although the index will still be displayed with translated keywords. This is the default.

NOHELPENCMD

specifies that SAS use the translated version of the keyword list for the command-line help, if a translated version exists.

Details

Set NOHELPENCMD if you want the command-line help to locate keywords by using the localized terms. By default, all terms on the command line will be read as English.

See Also

System Options:

- “HELPINDEX System Option: Windows” in SAS Companion for Windows
- “HELPINDEX System Option: UNIX” in SAS Companion for UNIX Environments
- “HELPLOC System Option: Windows” in SAS Companion for Windows
- “HELPLOC= System Option: z/OS” in SAS Companion for z/OS
HELPHOST System Option

Specifies the name of the computer where the remote browser is to send Help and ODS output.

**Syntax**

```
HELPHOST="host"
```

**Syntax Description**

"host"

specifies the name of the computer where the remote help is to be displayed. Quotation marks or parentheses are required. The maximum number of characters is 2,048.

**Details**

**Operating Environment Information**

If you do not specify the HELPHOST option, the location where SAS displays the Help depends on your operating environment. See the HELPHOST system option in the documentation for your operating environment.

**See Also**

- Chapter 6, “Viewing Output and Help in the SAS Remote Browser,” in *SAS Companion for UNIX Environments*
- “Viewing Output and Help in the SAS Remote Browser” in Chapter 2 of *SAS Companion for Windows*
- “Using the SAS Remote Browser” in Chapter 1 of *SAS Companion for z/OS*

**System Options:**

- “HELPBROWSER= System Option” on page 149
- “HELPPORT= System Option” on page 152
**HELP PORT= System Option**

Specifies the port number for the remote browser client.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Help

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

`HELP PORT=port-number`

**Syntax Description**

*port-number*

specifies the port number for the SAS Remote Browser Server.

Default: 0

Range: 0–65535

**Details**

When HELP PORT is set to 0, SAS uses the default port number for the remote browser server.

**See Also**

- Chapter 6, “Viewing Output and Help in the SAS Remote Browser,” in *SAS Companion for UNIX Environments*
- “Viewing Output and Help in the SAS Remote Browser” in Chapter 2 of *SAS Companion for Windows*
- “Using the SAS Remote Browser” in Chapter 1 of *SAS Companion for z/OS*

**System Options:**

- “HELP BROWSER= System Option” on page 149
- “HELP HOST System Option” on page 151

**HTTP SERVER PORT MAX= System Option**

Specifies the highest port number that can be used by the SAS HTTP server for remote browsing.

Valid in: Configuration file, SAS invocation

Category: Communications: Networking and encryption
PROC OPTIONS
GROUP= Communications

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
HTTPSERVERPORTMAX=max-port-number

Syntax Description

max-port-number
  specifies the highest port number that can be used by the SAS HTTP server for remote browsing.
  Default: 0
  Range: 0–65535

Details

Use the HTTPSERVERPORTMAX= and HTTPSERVERPORTMIN= system options to specify a range of port values that the remote browser HTTP server can use to dynamically assign a port number when a firewall is configured between SAS and the HTTP server.

See Also

System Options:
  • “HTTPSERVERPORTMIN= System Option” on page 153

HTTPSERVERPORTMIN= System Option

Specifies the lowest port number that can be used by the SAS HTTP server for remote browsing.

Valid in: Configuration file, SAS invocation
Category: Communications: Networking and encryption
PROC OPTIONS
GROUP= Communications

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
HTTPSERVERPORTMIN=min-port-number

Syntax Description

min-port-number
  specifies the lowest port number that can be used by the SAS HTTP server for remote browsing.
Details

Use the HTTPSERVERPORTMIN and HTTPSERVERPORTMAX system options to specify a range of port values that the remote browser HTTP server can use to dynamically assign a port number when a firewall is configured between SAS and the HTTP server.

See Also

System Options:

- “HTTPSERVERPORTMAX= System Option” on page 152

### IBUFNO= System Option

Specifies an optional number of extra buffers to be allocated for navigating an index file.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Files: SAS Files

**PROC OPTIONS GROUP=** SASFILES

**Default:** 0

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
IBUFNO=n | nK | nM | nG | nT | hexX | MIN | MAX
```

**Syntax Description**

- **n | nK | nM | nG | nT**
  specifies the number of extra index buffers to be allocated in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 8 specifies eight buffers, and a value of 3k specifies 3,072 buffers.
  **Restriction:** Maximum value is 10,000.

- **hexX**
  specifies the number of extra index buffers as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx specifies 45 buffers.

- **MIN**
  sets the number of extra index buffers to 0. This is the default.

- **MAX**
  sets the maximum number of extra index buffers to 10,000.
Details

An index is an optional SAS file that you can create for a SAS data file in order to provide direct access to specific observations. The index file consists of entries that are organized into hierarchical levels, such as a tree structure, and connected by pointers. When an index is used to process a request, such as for WHERE processing, SAS does a binary search on the index file and positions the index to the first entry that contains a qualified value. SAS uses the value's identifier to directly access the observation that contains the value. SAS requires memory for buffers when an index is actually used. The buffers are not required unless SAS uses the index, but they must be allocated in preparation for the index that is being used.

SAS automatically allocates a minimal number of buffers in order to navigate the index file. Typically, you do not need to specify extra buffers. However, using IBUFNO= to specify extra buffers could improve execution time by limiting the number of input/output operations that are required for a particular index file. However, the improvement in execution time comes at the expense of increased memory consumption.

Note: Whereas too few buffers allocated to the index file decrease performance, over allocation of index buffers creates performance problems as well. Experimentation is the best way to determine the optimal number of index buffers. For example, experiment with ibufno=3, then ibufno=4, and so on, until you find the least number of buffers that produces satisfactory performance results.

See Also

- “Understanding SAS Indexes” in Chapter 26 of SAS Language Reference: Concepts

System Options:

- “IBUFSIZE= System Option” on page 155

IBUFSIZE= System Option

Specifies the buffer page size for an index file.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Files: SAS Files

PROC OPTIONS GROUP=SASFILES

Restriction: Specify a page size before the index file is created. After it is created, you cannot change the page size.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

IBUFSIZE=n | nK | nM | nG | nT | hexX | MAX
Syntax Description

$n\ |\ nK\ |\ nM\ |\ nG\ |\ nT$
 specifies the page size to process in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes).

For example, a value of $8$ specifies 8 bytes, and a value of $3k$ specifies 3,072 bytes.

**Default:** 0, which causes SAS to use the minimum optimal page size for the operating environment

$hexX$
 specifies the page size as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value $2dx$ sets the page size to 45 bytes.

**MAX**
 sets the page size for an index file to the maximum possible number. For IBUFSIZE=, the value is 32,767 bytes.

Details

An index is an optional SAS file that you can create for a SAS data file in order to provide direct access to specific observations. The index file consists of entries that are organized into hierarchical levels, such as a tree structure, and connected by pointers. When an index is used to process a request, such as for WHERE processing, SAS does a search on the index file in order to rapidly locate the requested records.

Typically, you do not need to specify an index page size. However, the following situations could require a different page size:

- The page size affects the number of levels in the index. The more pages there are, the more levels in the index. The more levels, the longer the index search takes. Increasing the page size allows more index values to be stored on each page, thus reducing the number of pages (and the number of levels). The number of pages required for the index varies with the page size, the length of the index value, and the values themselves. The main resource that is saved when reducing levels in the index is I/O. If your application is experiencing a lot of I/O in the index file, increasing the page size might help. However, you must re-create the index file after increasing the page size.

- The index file structure requires a minimum of three index values to be stored on a page. If the length of an index value is very large, you might get an error message that the index could not be created because the page size is too small to hold three index values. Increasing the page size should eliminate the error.

**Note:** Experimentation is the best way to determine the optimal index page size.

See Also

- “Understanding SAS Indexes” in Chapter 26 of *SAS Language Reference: Concepts*

System Options:

- “IBUFNO= System Option” on page 154

**INITCMD System Option**

Specifies an application invocation command and optional SAS windowing environment or text editor commands that SAS executes before processing the AUTOEXEC= file during SAS invocation.
Valid in: Configuration file, SAS invocation
Category: Environment control: Initialization and operation

PROC OPTIONS
GROUP=EXECMODES

Note: This option cannot be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

Syntax

INITCMD "command-1 <windowing-command-n> "

Syntax Description

command-1
specifies any SAS command that invokes an application window. Some valid values are:

AF       LAB
ANALYST  MINER
ASSIST   PHCLINICAL
DESIGN   PHKINETICS
EIS      PROJMAN
FORECAST QUERY
GRAPH    RUNEIS
HELP     SQC
IMAGE    XADX.

Restriction: If you specify FORECAST for command-1, you cannot use windowing-command-n.

windowing-command-n
specifies a valid windowing command or text editor command. Separate multiple commands with semicolons. These commands are processed in sequence. If you use a windowing command that impacts flow, such as the BYE command, it might delay or prohibit processing.

Restriction: Do not use the windowing-command-n argument when you enter a command for an application that submits SAS statements or commands during initialization of the application, that is, during autoexec file initialization.

Details

The INITCMD system option suppresses the Log, Output, Program Editor, and Explorer windows when SAS starts so that application window is the first screen that you see. The suppressed windows do not appear, but you can activate them. You can use the ALTLOG option to direct log output for viewing. If windows are initiated by an autoexec file or the INITSTMT option, the window that is displayed by the INITCMD option is displayed last. When you exit an application that is invoked with the INITCMD option, your SAS session ends.

You can use the INITCMD option in a windowing environment only. Otherwise, the option is ignored and a warning message is issued. If command-1 is not a valid command, the option is ignored and a warning message is issued.
The following SAS execution mode invocation options, in order, have precedence over this option:

1. OBJECTSERVER.
2. DMR
3. SYSIN

If you specify INITCMD with another execution mode invocation option of equal precedence, SAS uses only the last option listed. See “Order of Precedence” on page 18 for more information about invocation option precedence.

Example

```
INITCMD "AFA c=mylib.myapp.primary.frame dsname=a.b"
INITCMD "ASSIST; FSVIEW SASUSER.CLASS"
```

### INITSTMT= System Option

Specifies a SAS statement to execute after any statements in the AUTOEXEC= file and before any statements from the SYSIN= file.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Environment control: Initialization and operation
- **PROC OPTIONS GROUP=** EXECMODES
- **Alias:** IS=
- **Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
- **See:** INITSTMT= System Option under Windows

### Syntax

```
INITSTMT='statement'
```

### Syntax Description

- **'statement'** specifies any SAS statement or statements.
  - **Requirement:** `statement` must be able to run on a step boundary.

### Comparisons

INITSTMT= specifies the SAS statements to be executed at SAS initialization, and the TERMSTMT= system option specifies the SAS statements to be executed at SAS termination.

### Example

Here is an example of using this option on UNIX:
sas -initstmt '%put you have used the initstmt; data x; x=1; run;'  

See Also

System Options:

• “TERMSTMT= System Option” on page 266

---

**INSERT= System Option**

Inserts the specified value as the first value of the specified system option.

- **Valid in:** configuration file, SAS invocation, OPTIONS statement, SAS System Option Window
- **Category:** Environment control: Files

**PROC OPTIONS**

- **GROUP=** ENVFILES

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**See:** For the syntax to use when you start SAS, see the documentation for your operating environment:

- “INSERT System Option: UNIX” in *SAS Companion for UNIX Environments*
- “INSERT System Option: Windows” in *SAS Companion for Windows*
- “INSERT= System Option: z/OS” in *SAS Companion for z/OS*

**Syntax**

```
INSERT=(system-option-1 =argument-1<system-option-n =argument-n>)
```

**Syntax Description**

- **system-option**
  - can be AUTOEXEC, CMPLIB, FMTSEARCH, HELPLOC, MAPS, MSG, SASAUTOS, SASHELP, SASSCRIPT, or SET.
  - **Note:** Some of these options are available only when SAS starts. These options can be specified in the INSERT= option only when the INSERT= option is specified in a configuration file or a SAS command.

- **argument**
  - specifies a new value that you want as the first value of system-option.
  - **argument** can be any value that could be specified for system-option if system-option is set using the OPTIONS statement.

**Details**

If you specify a new value for the AUTOEXEC, CMPLIB, FMTSEARCH, HELPLOC, MAPS, MSG, SASAUTOS, SASHELP, SASSCRIPT, or SET system options, the new value replaces the value of the option. Instead of replacing the value, you can use the INSERT= system option to add an additional value to the option as the first value of the option.
For a list of system options that the \texttt{INSERT=} system option and the \texttt{APPEND=} system option support, including the system options that can be used when SAS starts, submit the following \texttt{OPTIONS} procedure:

\begin{verbatim}
proc options listinsertappend;
run;
\end{verbatim}

**Comparisons**

The \texttt{INSERT=} system option adds a new value to the beginning of the current value of the \texttt{AUTOEXEC}, \texttt{CMPLIB}, \texttt{FMTSEARCH}, \texttt{HELPLOC}, \texttt{MAPS}, \texttt{MSG}, \texttt{SASAUTOS}, \texttt{SASHHELP}, \texttt{SASSCRIPT}, or \texttt{SET} system options. The \texttt{APPEND=} system option adds a new value to the end of one of these system options.

**Example**

The following table shows the results of adding a value to the beginning of the \texttt{FMTSEARCH=} option value:

<table>
<thead>
<tr>
<th>Current FMTSEARCH= Value</th>
<th>Value of INSERT= System Option</th>
<th>New FMTSEARCH= Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(WORK LIBRARY)</td>
<td>(fmtsearch=(abc def))</td>
<td>(ABC DEF WORK LIBRARY)</td>
</tr>
</tbody>
</table>

**See Also**

- “Changing an Option Value by Using the \texttt{INSERT} and \texttt{APPEND} System Options” on page 13

**System Options:**

- “\texttt{APPEND=} System Option” on page 68
- “\texttt{APPEND} System Option: UNIX” in \textit{SAS Companion for UNIX Environments}
- “\texttt{APPEND} System Option: Windows” in \textit{SAS Companion for Windows}
- “\texttt{APPEND=} System Option: z/OS” in \textit{SAS Companion for z/OS}

**\texttt{INTERVALDS=} System Option**

Specifies one or more interval name-value pairs, where the value is a SAS data set that contains user-supplied intervals. The intervals can be used as arguments to the \texttt{INTNX} and \texttt{INTCK} functions.

- **Valid in:** Configuration file, SAS invocation, \texttt{OPTIONS} statement, SAS System Options window
- **Category:** Input control: Data processing
- **\texttt{PROC OPTIONS GROUP=}** \texttt{INPUTCONTROL}
- **Requirement:** The set of interval-value pairs must be enclosed in parentheses.
- **Note:** This option can be restricted by a site administrator. For more information, see “\textit{Restricted Options}” on page 6.
Syntax

INTERVALDS=(interval-1=libref.dataset-name-1 <...interval-n=libref.dataset-name-n>)

Syntax Description

interval
specifies the name of an interval. The value of interval is the data set that is named in libref.dataset-name.

Requirement: When you specify multiple intervals, the interval name must not be the same as another interval.

libref.dataset-name
specifies the libref and the data set name of the file that contains user-defined holidays.

Details

The INTCK and INTNX functions specify interval as the interval name in the function argument list to reference a data set that names user-supplied intervals.

The same libref.dataset-name can be assigned to different intervals. An error occurs when more than one interval of the same name is defined for the INTERVALDS system option.

Example

This example assigns a single data set to an interval on the SAS command line or in a configuration file.

-intervalds (mycompany=mycompany.storeHours)

The next example assigns multiple intervals using the OPTIONS statement. The intervals subsid1 and subsid2 are assigned the same libref and data set name.

options intervalds=(mycompany=mycompany.storeHours subsid1=subsid.storeHours subsid2=subsid.storeHours);

See Also

• “Custom Time Intervals” in the SAS/ETS User’s Guide
• “About Date and Time Intervals” in Chapter 7 of SAS Language Reference: Concepts

Functions:

• “INTCK Function” in SAS Functions and CALL Routines: Reference
• “INTNX Function” in SAS Functions and CALL Routines: Reference

INVALIDDATA= System Option

Specifies the value that SAS assigns to a variable when invalid numeric data is encountered.
** INVALIDDATA= System Option **

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Input control: Data processing

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

INVALIDDATA=’character’

**Syntax Description**

‘character’ specifies the value to be assigned, which can be a letter (A through Z, a through z), a period (.), or an underscore (_). The default value is a period.

**Details**

The INVALIDDATA= system option specifies the value that SAS is to assign to a variable when invalid numeric data is read with an INPUT statement or the INPUT function.

---

** JPEGQUALITY= System Option **

Specifies the JPEG quality factor that determines the ratio of image quality to the level of compression for JPEG files produced by the SAS/GRAPH JPEG device driver.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS GROUP=

Notes: This option is ignored when the DEVICE graphics option is not set to JPEG.

This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

JPEGQUALITY= n | MIN | MAX

**Syntax Description**

n specifies an integer that indicates the JPEG quality factor. The quality of the image increases with larger numbers and decreases with smaller numbers. JPEG files are compressed less for higher-quality images. Therefore, the JPEG file size is greater for higher-quality images. For example, n=100 is completely uncompressed and the
image quality is highest. When \( n=0 \), the image is produced at the maximum compression level with the lowest quality.

**Default:** 75  
**Range:** 0–100

**MIN**  
specifies to set the JPEG quality factor to 0, which has the lowest image quality and the greatest file compression.

**MAX**  
specifies to set the JPEG quality factor to 100, which has the highest image quality with no file compression.

**Details**

The optimal quality value varies for each image. The default value of 75 is a good starting value that you can use to optimize the quality of an image within a compressed file. You can increase or decrease the value until you are satisfied with the image quality. Values between 50 and 95 produce the best quality images.

When the value is 24 or less, some viewers might not be able to display the JPEG file. When you create such a file, SAS writes the following caution to the SAS log:

Caution: quantization tables are too coarse for baseline JPEG.

**See Also**

Chapter 6, “Using Graphics Devices,” in *SAS/GRAPH: Reference*
Details

A *label* is a string of up to 256 characters that can be written by certain procedures in place of the variable's name.

See Also

Data Set Options:
- “LABEL= Data Set Option” in *SAS Data Set Options: Reference*

Statements:

---

**LABELCHKPT System Option**

Specifies whether checkpoint-restart data for labeled code sections is to be recorded for batch programs.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Environment control: Error handling</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>ERRORHANDLING</td>
</tr>
</tbody>
</table>

**Restrictions:**

The LABELCHKPT system option can be specified only if the STEPCHKPT system option is not specified when SAS starts.

Checkpoint mode is not valid for batch programs that contain the DM statement, which submits commands to SAS. If checkpoint mode is enabled and SAS encounters a DM statement, checkpoint mode is disabled, the checkpoint catalog entry is deleted, and a warning is written to the SAS log.

**Requirement:**

This option can be used only in batch mode.

**Note:**

This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

`LABELCHKPT | NOLABELCHKPT`

**Syntax Description**

`LABELCHKPT`

enables checkpoint mode for labeled code sections, which specifies to record checkpoint-restart data.

`NOLABELCHKPT`

disables checkpoint mode for labeled code sections, which specifies not to record checkpoint-restart data. This is the default.

**Details**

Using the LABELCHKPT system option puts SAS in checkpoint mode for SAS programs that run in batch. Each time a label is encountered, SAS records data in a checkpoint-restart library. If a program terminates without completing, the program can
be resubmitted, beginning at the labeled code section that was executing when the program terminated.

To ensure that the checkpoint-restart data is accurate, specify the ERRORCHECK STRICT option and set the ERRORABEND option. By setting these options, SAS terminates for most errors.

SAS can run in checkpoint-restart mode either for labeled code sections or for DATA and PROC steps, but not both.

**Comparisons**

The LABELCHKPT system option enables checkpoint mode for labeled code sections in batch programs that terminate before completing. Execution resumes at the labeled code section that was executing when the failure occurred.

The STEPCHKPT system option enables checkpoint mode for DATA and PROC steps in batch programs that terminate before completing. Execution resumes with the DATA or PROC step that was executing when the failure occurred.

**See Also**

- “Checkpoint Mode and Restart Mode” in Chapter 8 of *SAS Language Reference: Concepts*

**Statements:**

- “CHECKPOINT EXECUTE_ALWAYS Statement” in *SAS Statements: Reference*

**System Options:**

- “CHKPTCLEAN System Option” on page 89
- “LABELCHKPTLIB= System Option” on page 165
- “LABELRESTART System Option” on page 167
- “STEPCHKPT System Option” on page 243

---

**LABELCHKPTLIB= System Option**

Specifies the libref of the library where the checkpoint-restart data is saved for labeled code sections.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Environment control: Error handling
- **PROC OPTIONS GROUP=** ERRORHANDLING
- **Restriction:** The LABELCHKPTLIB= system option can be specified only if the STEPCHKPT system option is not specified when SAS starts.
- **Requirement:** This option can be used only in batch mode.
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

`LABELCHKPTLIB=libref`

Syntax Description

`libref`

specifies the libref that identifies the library where the checkpoint-restart data is saved.

Default: Work

Requirement: The LIBNAME statement that identifies the checkpoint-restart library must use the BASE engine and be the first statement in the batch program.

Details

When the LABELCHKPT system option is specified, checkpoint-restart data for labeled code sections in batch programs is saved in the libref that is specified in the `LABELCHKPTLIB=` system option. If no libref is specified, SAS uses the Work library to save checkpoint data. The LIBNAME statement that defines the libref must be the first statement in the batch program.

If the Work library is used to save checkpoint data, the NOWORKTERM and NOWORKINIT system options must be specified. When you set these options, the checkpoint-restart data is available when the batch program is resubmitted. These two options ensure that the Work library is saved when SAS ends and is restored when SAS starts. If the NOWORKTERM option is not specified, the Work library is deleted at the end of the SAS session and the checkpoint-restart data is lost. If the NOWORKINIT option is not specified, a new Work library is created when SAS starts, and again the checkpoint-restart data is lost.

The `LABELCHKPTLIB=` option must be specified for any SAS session that accesses checkpoint-restart data that is collected at label points and that is not saved to the Work library.

Operating Environment Information

When the Work library resides in a UNIX directory in UNIX or z/OS operating environments and you want to run the CLEANWORK utility, the Work library directory and its contents are deleted when the utility is run after the SAS session ends. When you run SAS in batch mode in the z/OS operating environment, the Work library is usually assigned to a temporary data set that is deleted at the end of the SAS job. To preserve the checkpoint-restart data in these cases, specify a permanent library as the value for the STEPCHECKPTLIB option.

Comparisons

When the LABELCKPT system option is set, the library that is specified by the `LABELCHKPTLIB` system option names the library where the checkpoint-restart data is saved for labeled code sections. When the LABELRESTART system option is set, the library that is specified by the `LABELCHKPTLIB` system option names the library where the checkpoint-restart data is used to resume execution of labeled code sections.

When the STEPCHECKPT system option is set, the library that is specified by the `STEPCHKPTLIB` system option names the library where the checkpoint-restart data is saved for DATA and PROC steps. When the STEPRESTART system option is set, the library that is specified by the `STEPCHKPTLIB` system option names the library where the checkpoint-restart data is used to resume execution of DATA and PROC steps.
LABELRESTART System Option

Specifies whether to execute a batch program by using checkpoint-restart data for data collected at labeled code sections.

Valid in: Configuration file, SAS invocation
Category: Environment control: Error handling
PROC OPTIONS GROUP= ERRORHANDLING

Restriction: The LABELRESTART system option can be specified only if the STEPCHKPT system option is not specified when SAS starts.
Requirement: This option can be used only in batch mode.
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
LABELRESTART | NOLABELRESTART

Syntax Description

LABELRESTART
enables restart mode, which specifies to execute the batch program by using the checkpoint-restart data.

NOLABELRESTART
disables restart mode, which specifies not to execute the batch program by using checkpoint-restart data.

Details
You specify the LABELRESTART option when you want to resubmit a batch program that ran in checkpoint mode for labeled code sections and terminated before the batch
program completed. When you resubmit the batch program, SAS determines from the checkpoint data the label that was executing when the program terminated. The program resumes executing the batch program at that label.

**Comparisons**

When you specify the LABELRESTART option, SAS uses the checkpoint-restart data for labeled code sections to resume execution of batch programs.

When you specify the STEPRESTART option, SAS uses the checkpoint-restart data for DATA and PROC steps to resume execution of batch programs.

**See Also**

- “Checkpoint Mode and Restart Mode” in Chapter 8 of *SAS Language Reference: Concepts*

**Statements:**

- “CHECKPOINT EXECUTE_ALWAYS Statement” in *SAS Statements: Reference*

**System Options:**

- “CHKPTCLEAN System Option” on page 89
- “LABELCHKPT System Option” on page 164
- “LABELCHKPTLIB= System Option” on page 165
- “STEPCHKPT System Option” on page 243
- “STEPRESTART System Option” on page 246

---

**_LAST_= System Option**

Specifies the most recently created data set.

**Valid in:**

- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**

Files: SAS Files

**PROC OPTIONS GROUP=**

SASFILES

**Restriction:**

-LAST_= is not allowed with data set options.

**Note:**

This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

-LAST_=SAS-data-set

**Syntax Description**

*SAS-data-set*

specifies a SAS data set name.

**Restriction:**

No data set options are allowed.
Tip: Use libref.membername or membername syntax, not a string that is enclosed in quotation marks, to specify a SAS data set name. You can use quotation marks in the libref.membername or membername syntax if the libref or member name is associated with a SAS/ACCESS engine that supports member names with syntax that requires quoting or name literal (n-literal) specification. For more information, see SAS/ACCESS for Relational Databases: Reference.

Details

By default, SAS automatically keeps track of the most recently created SAS data set. Use the _LAST_= system option to override the default.

**LEFTMARGIN= System Option**

Specifies the print margin for the left side of the page.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: ODS Printing
- **PROC OPTIONS GROUP=** ODSPRINT

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
LEFTMARGIN=margin-size<margin-unit>
```

**Syntax Description**

- `margin-size` specifies the size of the left print margin.
  - **Restriction:** The left margin should be small enough so that the left margin plus the right margin is less than the width of the paper.
  - **Interaction:** Changing the value of this option might result in changes to the value of the LINESIZE= system option.

- `<margin-unit>` specifies the units for margin-size. The margin-unit can be `in` for inches or `cm` for centimeters. `<margin-unit>` is saved as part of the value of the LEFTMARGIN system option whether it is specified.
  - **Default:** inches

**Details**

All margins have a minimum that is dependent on the printer and the paper size. The default value of the LEFTMARGIN system option is `0.00 in`.

**See Also**

- Chapter 15, “Printing with SAS,” in *SAS Language Reference: Concepts*
System Options:

- “BOTTOMMARGIN= System Option” on page 76
- “LINESIZE= System Option” on page 170
- “RIGHTMARGIN= System Option” on page 222
- “TOPMARGIN= System Option” on page 269

**LINESIZE= System Option**

Specifies the line size for the SAS log and for SAS procedure output.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Categories:** Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

**PROC OPTIONS**

**GROUP=** LOG_LISTCONTROL
LISTCONTROL
LOGCONTROL

**Alias:** LS=

**Note:** This option cannot be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

**See:**
- “LINESIZE System Option: UNIX” in *SAS Companion for UNIX Environments*
- “LINESIZE System Option: Windows” in *SAS Companion for Windows*
- “LINESIZE= System Option: z/OS” in *SAS Companion for z/OS*

**Syntax**

LINESIZE=n | MIN | MAX | hexX

**Syntax Description**

- **n** specifies the number of characters in a line.
- **MIN**
  - sets the number of characters in a line to 64.
- **MAX**
  - sets the number of characters in a line to 256.
- **hexX**
  - specifies the number of characters in a line as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an X. For example, the value **0FAx** sets the line size of the SAS procedure output to 250.
Details

The LINESIZE= system option specifies the line size (printer line width) in characters for the SAS log and the SAS output that are used by the DATA step and procedures. The LINESIZE= system option affects the following output:

- the Output window for the ODS LISTING destination
- output produced for an ODS markup destination by a DATA step where the FILE statement destination is PRINT (the FILE PRINT ODS statement is not affected by the LINESIZE= system option)
- procedures that produce only characters that cannot be scaled, such as the PLOT procedure, the CALENDAR procedure, the TIMEPLOT procedure, the FORMS procedure, and the CHART procedure

See Also


LOGPARM= System Option

Specifies when SAS log files are opened, closed, and, in conjunction with the LOG= system option, how they are named.

Valid in: Configuration file, SAS invocation
Category: Log and procedure output control: SAS log
PROC OPTIONS
GROUP= LOGCONTROL
Restriction: LOGPARM= is valid only in line mode and in batch mode
Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
See: “LOGPARM= System Option: z/OS” in SAS Companion for z/OS

Syntax

LOGPARM=

"<OPEN= APPEND | REPLACE | REPLACEOLD> 
< ROLLOVER= AUTO | NONE | SESSION | n | nK | nM | nG> 
< WRITE= BUFFERED | IMMEDIATE> ”

Syntax Description

OPEN=APPEND | REPLACE | REPLACEOLD
when a log file already exists, specifies how the contents of the existing file are treated.

APPEND
appends the log when opening an existing file. If the file does not already exist, a new file is created.

REPLACE
overwrites the current contents when opening an existing file. If the file does not already exist, a new file is created.
REPLACEOLD replaces files that are more than one day old. If the file does not already exist, a new file is created.

**Default:** REPLACE

**z/OS specifics:** See the SAS documentation for your operating environment for limitations on the use of OPEN=REPLACEOLD.

**ROLLOVER=AUTO|NONE|SESSION | n | nK | nM | nG**

specifies when or if the SAS log “rolls over”. That is, when the current log is closed and a new one is opened.

**AUTO**
causes an automatic “rollover” of the log when the directives in the value of the LOG= option change, that is, the current log is closed and a new log file is opened.

**Restriction:** Rollover will not occur more often than once a minute.

**Interactions:**
Rollover is triggered by a change in the value of the LOG= option. The name of the new log file is determined by the value of the LOG= system option. If LOG= does not contain a directive, however, the name would never change, so the log would never roll over, even when ROLLOVER=AUTO.

**NONE**
specifies that rollover does not occur, even when a change occurs in the name that is specified with the LOG= option.

**Interaction:** If the LOG= value contains any directives, they do not resolve. For example, if Log=”#b.log” is specified, the directive “#” does not resolve, and the name of the log file remains “#b.log”.

**SESSION**
at the beginning of each SAS session, opens the log file, resolves directives that are specified in the LOG= system option, and uses its resolved value to name the new log file. During the course of the session, no rollover is performed.

**n | nK | nM | nG**
causes the log to rollover when the log reaches a specific size, stated in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); or 1,073,741,824 (gigabytes). When the log reaches the specified size, it is closed and renamed by appending “old” to the log filename, and if it exists, the lock file for a server log. For example, a filename of mylog.log would be renamed mylogold.log. A new log file is opened using the name specified in the LOG= option.

**Restriction:** The minimum log file size is 10K.

**Interaction:** When a rollover occurs because of size and the LOG= value contains any directives, the directives do not resolve. For example, if Log=”#b.log” is specified, the directive “#” does not resolve, and the name of the log file remains “#b.log”.

**Note:** If you use ROLLOVER=n to roll over your files, the OPEN= parameter is ignored, and the initial log file is opened with OPEN=APPEND.

**See:** “Rolling Over the SAS Log” in Chapter 9 of *SAS Language Reference: Concepts*

**CAUTION:** Old log files can be overwritten. SAS maintains only one old log file with the same name as the open log file. If rollover occurs more than once, the old log file is overwritten.

**Default:** NONE

**See:**
WRITE=BUFFERED | IMMEDIATE
specifies when content is written to the SAS log.

BUFFERED
writes content to the SAS log only when a buffer is full in order to increase efficiency.

IMMEDIATE
writes to the SAS log each time that statements are submitted that produce content for the SAS log. SAS does no buffering of log messages.

Default: BUFFERED

Windows specifics: The buffered log contents are written periodically, using an interval that is specified by SAS.

Details

The LOGPARM= system option controls the opening and closing of SAS log files when SAS is operating in batch mode or in line mode. This option also controls the naming of new log files, in conjunction with the LOG= system option and the use of directives in the value of LOG=.

Using directives in the value of the LOG= system option enables you to control when logs are open and closed and how they are named, based on actual time events, such as time, month, and day of week.

Operating Environment Information
Under the Windows and UNIX operating environments, you can begin directives with either the % symbol or the # symbol, and use both symbols in the same directive. For example, -log=mylog%b#C.log. Under z/OS, begin directives only with the # symbol. For example, -log=mylog#b#c.log. Under OpenVMS, begin directives only with the % symbol. For example, -log=mylog%b%c.log.

The following table contains a list of directives that are valid in LOG= values:

<table>
<thead>
<tr>
<th>Directive</th>
<th>Description</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>%a or #a</td>
<td>Locale's abbreviated day of week</td>
<td>Sun–Sat</td>
</tr>
<tr>
<td>%A or #A</td>
<td>Locale's full day of week</td>
<td>Sunday–Saturday</td>
</tr>
<tr>
<td>%b or #b</td>
<td>Local's abbreviated month</td>
<td>Jan–Dec</td>
</tr>
<tr>
<td>%B or #B</td>
<td>Locale's full month</td>
<td>January–December</td>
</tr>
<tr>
<td>%C or #C</td>
<td>Century number</td>
<td>00–99</td>
</tr>
<tr>
<td>%d or #d</td>
<td>Day of the month</td>
<td>01–31</td>
</tr>
<tr>
<td>Directive</td>
<td>Description</td>
<td>Range</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>%H or #H</td>
<td>Hour</td>
<td>00–23</td>
</tr>
<tr>
<td>%j or #j</td>
<td>Julian day</td>
<td>001–366</td>
</tr>
<tr>
<td>%l or #l *</td>
<td>User name</td>
<td>alphanumeric string that is the name of the user that started SAS</td>
</tr>
<tr>
<td>%M or #M</td>
<td>Minutes</td>
<td>00–59</td>
</tr>
<tr>
<td>%m or #m</td>
<td>Month number</td>
<td>01–12</td>
</tr>
<tr>
<td>%n or #n</td>
<td>Current system node name (without domain name)</td>
<td>none</td>
</tr>
<tr>
<td>%p or #p *</td>
<td>Process ID</td>
<td>alphanumeric string that is the SAS session process ID</td>
</tr>
<tr>
<td>%s or #s</td>
<td>Seconds</td>
<td>00–59</td>
</tr>
<tr>
<td>%u or #u</td>
<td>Day of week</td>
<td>1= Monday–7=Sunday</td>
</tr>
<tr>
<td>%v or #v *</td>
<td>Unique identifier</td>
<td>alphanumeric string that creates a log filename that does not currently exist</td>
</tr>
<tr>
<td>%w or #w</td>
<td>Day of week</td>
<td>0=Sunday–6=Saturday</td>
</tr>
<tr>
<td>%W or #W</td>
<td>Week number (Monday as first day; all days in new year preceding first Monday are in week 00)</td>
<td>00–53</td>
</tr>
<tr>
<td>%y or #y</td>
<td>Year without century</td>
<td>00–99</td>
</tr>
<tr>
<td>%Y or #Y</td>
<td>Full year</td>
<td>1970–9999</td>
</tr>
<tr>
<td>%%</td>
<td>Percent escape writes a single percent sign in the log filename.</td>
<td>%</td>
</tr>
<tr>
<td>##</td>
<td>Pound escape writes a single pound sign in the log filename.</td>
<td>#</td>
</tr>
</tbody>
</table>

* Because %v, %l, and %p are not a time-based format, the log filename will never change after it has been generated. Therefore, the log will never roll over. In these situations, specifying ROLLOVER=AUTO is equivalent to specifying ROLLOVER=SESSION.

**Operating Environment Information**

See SAS Companion for z/OS for limitations on the length of the log filename under z/OS.

**Note:** Directives that you specify in the LOG= system option are not the same as the conversion characters that you specify to format logging facility logs. Directives specify a format for a log name. Conversion characters specify a format for log
messages. Directives and conversion characters that use the same characters might function differently.

Note: If you start SAS in batch mode or server mode and the LOGCONFIGLOC= option is specified, logging is done by the SAS logging facility. The traditional SAS log option LOGPARM= is ignored. The traditional SAS log option LOG= is honored only when the %S{App.Log} conversion character is specified in the logging configuration file. For more information, see Chapter 1, “The SAS Logging Facility,” in SAS Logging: Configuration and Programming Reference.

Example

Operating Environment Information

The LOGPARM= system option is executed when SAS is invoked. When you invoke SAS at your site, the form of the syntax is specific to your operating environment. See the SAS documentation for your operating environment for details.

Rolling over the log at a certain time and using directives to name the log according to the time:

If this command is submitted at 9:43 AM, this example creates a log file called test0943.log, and the log rolls over each time the log filename changes. In this example, at 9:44 AM, the test0943.log file will be closed, and the test0944.log file will be opened.

sas -log "test%H%M.log" -logparm "rollover=auto"

Preventing log rollover but using directives to name the log:

For a SAS session that begins at 9:34 AM, this example creates a log file named test0934.log, and prevents the log file from rolling over:

sas -log "test%H%M.log" -logparm "rollover=session"

Preventing log rollover and preventing the resolution of directives:

This example creates a log file named test%H%M.log, ignores the directives, and prevents the log file from rolling over during the session:

sas -log "test%H%M.log" -logparm "rollover=none"

Creating log files with unique identifiers:

This example uses a unique identifier to create a log file with a unique name:

sas -log "test%v.log" -logparm "rollover=session"

SAS replaces the directive %v with process_IDv, where process_ID is a numeric process identifier that is determined by the operating system and n is an integer number, starting with 1. The letter v that is between process_ID and n is always a lowercase letter.

For this example, process_ID is 3755. If the file does not already exist, SAS creates a log file with the name test3755v1.log. If test3755v1.log does exist, SAS attempts to create a log file by incrementing n by 1, and this process continues until SAS can generate a log file. For example, if the file test3755v1.log exists, SAS attempts to create the file test3755v2.log.

Naming a log file by the user that started SAS:

This example creates a log filename that contains the user name that started the SAS session:

sas -log "%l.log" -logparm "rollover=session";
LRECL= System Option

Specifies the default logical record length to use for reading and writing external files.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Files: External files

PROC OPTIONS GROUP= EXTFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

LRECL=n | nK | hexX | MIN | MAX

Syntax Description

\( n \)

specifies the logical record length in multiples of 1 (bytes) or 1,024 (kilobytes). For example, a value of 32 specifies 32 bytes, and a value of 32k specifies 32,767 bytes.

Default: 256

Range: 1–32767

\( hexX \)

specifies the logical record length as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 2dx sets the logical record length to 45 characters.

MIN

specifies a logical record length of 1.

MAX

specifies a logical record length of 32,767.

Details

The logical record length for reading or writing external files is first determined by the LRECL= option in the access method statement, function, or command that is used to read or write an individual file, or the DDName value in the z/OS operating environment. If the logical record length is not specified by any of these means, SAS uses the value that is specified by the LRECL= system option.

Use a value for the LRECL= system option that is not an arbitrary large value. Large values for this option can result in excessive use of memory, which can degrade performance.

z/OS Specifics

Under z/OS, the LRECL= system option is recognized only for reading and writing HFS files.

See Also

MAPS= System Option
Specifies the location of the SAS library that contains SAS/GRAPH map data sets.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Graphics: Driver settings

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Tip: You can use the APPEND or INSERT system options to add additional location-of-maps.
See: “MAPS System Option: UNIX” in SAS Companion for UNIX Environments
“MAPS System Option: Windows” in SAS Companion for Windows

Syntax
MAPS=location-of-maps

Syntax Description
location-of-maps
specifies either a physical path, an environment variable, or a libref to locate the SAS/GRAPH map data sets.
Default: MAPS

See Also
• “Using SAS/GRAPH Map Data Sets” in Chapter 52 of SAS/GRAPH: Reference

System Options:
• “APPEND= System Option” on page 68
• “INSERT= System Option” on page 159

MERGENOBY System Option
Specifies the type of message that is issued when MERGE processing occurs without an associated BY statement.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Files: SAS Files
PROC OPTIONS GROUP=
SASFILES
**MISSING= System Option**

Specifies the character to print for missing numeric values.

**Syntax**

`MISSING= '<character>'`

**Syntax Description**

`character`

specifies the value to be printed. The value can be any character. Single or double quotation marks are optional. The period is the default.

**Details**

The MISSING= system option does not apply to special missing values such as .A and .Z.

**See Also**

MSGLEVEL= System Option

Specifies the level of detail in messages that are written to the SAS log.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: SAS log

**PROC OPTIONS GROUP=** LOGCONTROL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

\texttt{MSGLEVEL=} \texttt{N | I}

**Syntax Description**

\texttt{N}

specifies to print notes, warnings, CEDA message, and error messages only. N is the default.

\texttt{I}

specifies to print additional notes pertaining to index usage, merge processing, and sort utilities, along with standard notes, warnings, CEDA message, and error messages.

**Details**

Some of the conditions under which the MSGLEVEL= system option applies are as follows:

- If MSGLEVEL=I, SAS writes informative messages to the SAS log about index processing. In general, when a WHERE expression is executed for a data set with indexes, the following information appears in the SAS log:
  - if an index is used, a message displays that specifies the name of the index
  - if an index is not used but one exists that could optimize at least one condition in the WHERE expression, messages provide suggestions that describe what you can do to influence SAS to use the index. For example, a message could suggest sorting the data set into index order or to specify more buffers.
  - a message displays the IDXWHERE= or IDXNAME= data set option value if the setting can affect index processing.

- If MSGLEVEL=I, SAS writes a warning to the SAS log when a MERGE statement would cause variables to be overwritten.

- If MSGLEVEL=I, SAS writes a message that indicates which sorting product was used.

- For informative messages about queries by an application to a SAS/SHARE server, MSGLEVEL=I must be set for the SAS session where the SAS/SHARE server is
running. The messages are written to the SAS log for the SAS session that runs the SAS/SHARE server.

See Also


MULTENVAPPL System Option

Specifies whether the fonts that are available in a SAS application font selector window lists only the SAS fonts that are available in all operating environments.

Valid in:  Configuration file, SAS invocation, OPTIONS statement
Category:  Environment control: Initialization and operation
PROC OPTIONS
GROUP= EXECMODES
Note:  This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

MULTENVAPPL | NOMULTENVAPPL

Syntax Description

MULTENVAPPL
  specifies that an application font selector window lists only the SAS fonts.

NOMULTENVAPPL
  specifies that an application font selector window lists only the operating environment fonts.

Details

The MULTENVAPPL system option enables applications that support a font selection window, such as SAS/AF, SAS/FSP, SAS/EIS, or SAS/GIS, to choose a SAS font that is supported in all operating environments. Choosing a SAS font ensures portability of applications across all operating environments.

When NOMULTENVAPPL is in effect, the application font selector window has available only the fonts that are specific to your operating environment. SAS might need to resize operating environment fonts, which could result in text that is difficult to read. If the application is ported to another environment and the font is not available, a font is selected by the operating environment.

NEWS= System Option

Specifies an external file that contains messages to be written to the SAS log, immediately after the header.

Valid in:  Configuration file, SAS invocation
Categories:  Environment control: Files
Log and procedure output control: SAS log

PROC OPTIONS
  GROUP= ENVFILES
  LOGCONTROL

Operating environment: Although the syntax is generally consistent with the command line syntax of your operating environment, it might include additional or alternate punctuation. For details, see the SAS documentation for your operating environment.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.


Syntax

NEWS=external-file

Syntax Description

external-file
  specifies an external file.

Operating environment: A valid file specification and its syntax are specific to your operating environment. Although the syntax is generally consistent with the command line syntax of your operating environment, it might include additional or alternate punctuation. For details, see the SAS documentation for your operating environment.

Details

The NEWS file can contain information for uses, including news items about SAS.

The contents of the NEWS file are written to the SAS log immediately after the SAS header.

See Also


NOTES System Option

Specifies whether notes are written to the SAS log.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: SAS log

PROC OPTIONS
  GROUP= LOGCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

NOTES | NONOTES

Syntax Description

NOTES
specifies that SAS write notes to the SAS log.

NONOTES
specifies that SAS does not write notes to the SAS log. NONOTES does not suppress error and warning messages.

Details

You must specify NOTES for SAS programs that you send to SAS for problem determination and resolution.

See Also


NUMBER System Option

Specifies whether to print the page number in the title line of each page of SAS output.

Valid in:
Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories:
Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

PROC OPTIONS
GROUP= LOG_LISTCONTROL
LISTCONTROL
LOGCONTROL

Note:
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

NUMBER | NONUMBER

Syntax Description

NUMBER
specifies that SAS print the page number on the first title line of each page of SAS output.

NONUMBER
specifies that SAS not print the page number on the first title line of each page of SAS output.
See Also


OBS= System Option

Specifies the observation that is used to determine the last observation to process, or specifies the last record to process.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Files: SAS Files

PROC OPTIONS GROUP=SASFILES

Interaction: When you specify the OBS= option and EXTENDOBSCOUNTER=YES is set either as a data set option or as a LIBNAME option, data sets that have 2G–1 observations or more might perform better in a 32-bit environment. For more information, see “Extending the Observation Count in a SAS Data File” in Chapter 26 of SAS Language Reference: Concepts.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “OBS System Option: UNIX” in SAS Companion for UNIX Environments
    “OBS System Option: Windows” in SAS Companion for Windows

Syntax

OBS= n | nK | nM | nG | nT | hexX | MIN | MAX

Syntax Description

n | nK | nM | nG | nT
specifies a number to indicate when to stop processing, with n being an integer. Using one of the letter notations results in multiplying the integer by a specific value. That is, specifying K (kilo) multiplies the integer by 1,024; M (mega) multiplies by 1,048,576; G (giga) multiplies by 1,073,741,824; or T (tera) multiplies by 1,099,511,627,776. For example, a value of 20 specifies 20 observations or records, while a value of 3m specifies 3,145,728 observations or records.

hexX
specifies a number to indicate when to stop processing as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the hexadecimal value F8 must be specified as 0F8X in order to specify the decimal equivalent of 248. The value 2DX specifies the decimal equivalent of 45.

MIN
sets the number to 0 to indicate when to stop processing.

Interaction: If OBS=0 and the NOREPLACE option is in effect, then SAS can still take certain actions because it actually executes each DATA and PROC step in the program, using no observations. For example, SAS executes procedures, such as CONTENTS and DATASETS, that process libraries or SAS data sets. External files are also opened and closed. Therefore, even if you specify OBS=0,
when your program writes to an external file with a PUT statement, an end-of-file mark is written, and any existing data in the file is deleted.

**MAX**
sets the number to indicate when to stop processing to the maximum number of observations or records, up to the largest eight-byte, signed integer, which is $2^{63}-1$, or approximately 9.2 quintillion. This is the default.

**Details**

**OBS=** tells SAS when to stop processing observations or records. To determine when to stop processing, SAS uses the value for **OBS=** in a formula that includes the value for **OBS=** and the value for **FIRSTOBS=**. The formula is

$$\text{results} = (\text{obs} - \text{firstobs}) + 1$$

For example, if **OBS=10** and **FIRSTOBS=1** (which is the default for **FIRSTOBS=**), the result is 10 observations or records, that is, \((10 - 1) + 1 = 10\). If **OBS=10** and **FIRSTOBS=2**, the result is nine observations or records, that is, \((10 - 2) + 1 = 9\).

**OBS=** is valid for all steps during your current SAS session or until you change the setting.

You can also use **OBS=** to control analysis of SAS data sets in PROC steps.

If SAS is processing a raw data file, **OBS=** specifies the last line of data to read. SAS counts a line of input data as one observation, even if the raw data for several SAS data set observations is on a single line.

**Comparisons**

- An **OBS=** specification from either a data set option or an INFILE statement option takes precedence over the **OBS=** system option.
- While the **OBS=** system option specifies an ending point for processing, the **FIRSTOBS=** system option specifies a starting point. The two options are often used together to define a range of observations to be processed.

**Examples**

**Example 1: Using OBS= to Specify When to Stop Processing Observations**

This example illustrates the result of using **OBS=** to tell SAS when to stop processing observations. This example creates a SAS data set, executes the OPTIONS statement by specifying FIRSTOBS=2 and OBS=12, and executes the PRINT procedure. The result is 11 observations, that is, \((12 - 2) + 1 = 11\). The result of **OBS=** in this situation appears to be the observation number that SAS processes last, because the output starts with observation 2, and ends with observation 12, but this result is only a coincidence.

```sas
data Ages;
  input Name $ Age;
datalines;
Miguel 53
Brad 27
Willie 69
Marc 50
Sylvia 40
Arun 25
```

Chapter 3 • Dictionary of System Options
Gary 40
Becky 51
Alma 39
Tom 62
Kris 66
Paul 60
Randy 43
Barbara 52
Virginia 72
run;

options firstobs=2 obs=12;
proc print data=Ages;
run;

Output 3.1  PROC PRINT Output Using OBS= and FIRSTOBS=

Example 2: Using OBS= with WHERE Processing
This example illustrates the result of using OBS= along with WHERE processing. The example uses the data set that was created in Example 1, which contains 15 observations, and the example assumes a new SAS session with the defaults FIRSTOBS=1 and OBS=MAX.

First, here is the PRINT procedure with a WHERE statement. The subset of the data results in 12 observations:

    proc print data=Ages;
       where Age LT 65;
    run;
Executing the OPTIONS statement with OBS=10 and the PRINT procedure with the WHERE statement results in 10 observations, that is, \((10 - 1) + 1 = 10\). Note that with WHERE processing, SAS first subsets the data and then SAS applies OBS= to the subset.

```sas
options obs=10;
proc print data=Ages;
    where Age LT 65;
run;
```
The result of OBS= appears to be how many observations to process, because the output consists of 10 observations, ending with the observation number 12. However, the result is only a coincidence. If you apply FIRSTOBS=2 and OBS=10 to the subset, the result is nine observations, that is, \( (10 - 2) + 1 = 9 \). OBS= in this situation is neither the observation number to end with nor how many observations to process; the value is used in the formula to determine when to stop processing.

```sas
options firstobs=2 obs=10;
proc print data=Ages;
   where Age LT 65;
run;
```
**Output 3.4**  PROC PRINT Output Using WHERE Statement, OBS=, and FIRSTOBS=

**Example 3: Using OBS= When Observations Are Deleted**
This example illustrates the result of using OBS= for a data set that has deleted observations. The example uses the data set that was created in Example 1, with observation 6 deleted. The example also assumes a new SAS session with the defaults FIRSTOBS=1 and OBS=MAX.

First, here is PROC PRINT output of the modified file:

```sas
options firstobs=1 obs=max nodate pageno=1;
proc print data=Ages;
run;
```
Executing the OPTIONS statement with OBS=12, then the PRINT procedure, results in 12 observations, that is, \((12 - 1) + 1 = 12\):

```latex
options obs=12;
proc print data=Ages;
run;
```
The result of OBS= appears to be how many observations to process, because the output consists of 12 observations, ending with the observation number 13. However, if you apply FIRSTOBS=2 and OBS=12, the result is 11 observations, that is \((12 - 2) + 1 = 11\). OBS= in this situation is neither the observation number to end with nor how many observations to process; the value is used in the formula to determine when to stop processing.

```sas
options firstobs=2 obs=12;
proc print data=Ages;
run;
```
**ORIENTATION= System Option**

Specifies the paper orientation to use when printing to a printer.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: ODS Printing
- **PROC OPTIONS GROUP=** ODSPRINT

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

ORIENTATION=PORTRAIT | LANDSCAPE | REVERSEPORTRAIT | REVERSELANDSCAPE

Syntax Description

PORTRAIT
specifies the paper orientation as portrait. This is the default.

LANDSCAPE
specifies the paper orientation as landscape.

REVERSEPORTRAIT
specifies the paper orientation as reverse portrait. Use this value to control the top of the page relative to how the paper is inserted into the input paper tray. REVERSEPORTRAIT can be used when you print preprinted or punched forms.

REVERSELANDSCAPE
specifies the paper orientation as reverse landscape. Use this value to control the top of the page relative to how the paper is inserted into the input paper tray. REVERSELANDSCAPE can be used when you print preprinted or punched forms.

Details

Changing the value of this option might result in changes to the values of the portable LINESIZE= and PAGESIZE= system options.

You can change the orientation between document pages for the following output types:
• the LISTING destination
• the RTF destination
• a Universal Printing printer

Note: Changing the orientation between document pages is supported only for Universal Printing. It is not supported for Windows printing.

Use the OPTIONS statement between the steps that create output to change the page orientation.

Example

This example creates a PDF file with both portrait and landscape orientations.

```plaintext
options orientation=landscape obs=5;
ods pdf file="File3.pdf";
proc print data=sashelp.class;
run;
options orientation=portrait;
proc print data=sashelp.retail; run;
ods pdf close;
```

Here is the output:
**Display 3.1** The First Page of the PDF Has a Landscape Orientation

**Display 3.2** The Second Page of the PDF Has a Portrait Orientation

**See Also**
- “Universal Printing” in Chapter 15 of *SAS Language Reference: Concepts*

**System Options:**
- “LINESIZE= System Option” on page 170
- “PAGESIZE= System Option” on page 196
OVP System Option

Specifies whether overprinting of error messages to make them bold, is enabled.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP=

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

OVP | NOOVP

Syntax Description

OVP

specifies that overprinting of error messages is enabled.

NOOVP

specifies that overprinting of error messages is disabled. This is the default.

Details

When OVP is specified, error messages are emphasized when SAS overprints the error message two additional times with overprint characters.

When output is displayed to a monitor, OVP is overridden and is changed to NOOVP.

See Also


PAGEBREAKINITIAL System Option

Specifies whether to begin the SAS log and procedure output files for the LISTING destination on a new page.

Valid in: Configuration file, SAS invocation

Categories: Log and procedure output control: SAS log and procedure output
Log and procedure output control: SAS log
Log and procedure output control: Procedure output

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
See: “PAGEBREAKINITIAL System Option: z/OS” in SAS Companion for z/OS

Syntax

\texttt{PAGEBREAKINITIAL | NOPAGEBREAKINITIAL}

\textbf{Syntax Description}

\texttt{PAGEBREAKINITIAL}  
specifies to begin the SAS log and procedure output files on a new page.

\texttt{NOPAGEBREAKINITIAL}  
specifies not to begin the SAS log and procedure output files on a new page.  
\texttt{NOPAGEBREAKINITIAL} is the default.

\textbf{Details}

The \texttt{PAGEBREAKINITIAL} option inserts a page break at the start of the SAS log and procedure output files for the \texttt{LISTING} destination.

\textbf{See Also}


---

\textbf{PAGENO= System Option}

Resets the SAS output page number.

\textbf{Valid in:}  
Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

\textbf{Category:}  
Log and procedure output control: Procedure output

\textbf{PROC OPTIONS GROUP=}  
LISTCONTROL

\textbf{Note:}  
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

\textbf{See:}  
PAGENO= System Option under Windows

\textbf{Syntax}

\texttt{PAGENO= n | nK | hexX | MIN | MAX}

\textbf{Syntax Description}

\texttt{n \textbackslash{} nK}  
\texttt{n | nK}  
specifies the page number in multiples of 1 (\texttt{n}); 1,024 (\texttt{nK}). For example, a value of 8 sets the page number to 8 and a value of 3k sets the page number to 3,072.

\texttt{hexX}  
specifies the page number as a hexadecimal number. You must specify the value beginning with a number (0-9), followed by an X. For example, the value \texttt{2dx} sets the page number to 45.
MIN
sets the page number to the minimum number, 1.

MAX
specifies the maximum page number as the largest signed, four-byte integer that is
representable in your operating environment.

**Details**
The PAGENO= system option specifies a beginning page number for the next page of
output that SAS produces. Use PAGENO= to reset page numbering during a SAS
session.

---

**PAGESIZE= System Option**

Specifies the number of lines that compose a page of the SAS log and SAS output.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options
  window
- **Categories:** Log and procedure output control: SAS log and procedure output
  Log and procedure output control: SAS log
  Log and procedure output control: Procedure output

**PROC OPTIONS**

- **GROUP=** LOG_LISTCONTROL
  - LISTCONTROL
  - LOGCONTROL
- **Alias:** PS=
- **Note:** This option cannot be restricted by a site administrator. For more information, see
  “Restricted Options” on page 6.
- **See:** “PAGESIZE System Option: UNIX” in SAS Companion for UNIX Environments
  “PAGESIZE System Option: Windows” in SAS Companion for Windows
  “PAGESIZE= System Option: z/OS” in SAS Companion for z/OS

**Syntax**

PAGESIZE= \( n \mid nK \mid \text{hex}X \mid \text{MIN} \mid \text{MAX} \)

**Syntax Description**

\( n \mid nK \)

specifies the number of lines that compose a page in terms of lines (\( n \))or units of
1,024 lines (\( nK \)).

\( \text{hex}X \)

specifies the number of lines that compose a page as a hexadecimal number. You
must specify the value beginning with a number (0–9), followed by an X. For
example, the value \( 2\text{dx} \) sets the number of lines that compose a page to 45 lines.

\( \text{MIN} \)

sets the number of lines that compose a page to the minimum setting, 15.

\( \text{MAX} \)

sets the number of lines that compose a page to the maximum setting, 32,767.
Details
The PAGESIZE= system option affects the following output:
• the Output window for the ODS LISTING destination
• the SAS log in batch and non-interactive modes
• the ODS markup destinations when the PRINT option is used in the FILE statement in a DATA step (the FILE PRINT ODS statement is not affected by the PAGESIZE= system option)
• procedures that produce characters that cannot be scaled, such as the PLOT procedure, the CALENDAR procedure, the TIMEPLOT procedure, the FORMS procedure, and the CHART procedure

See Also

PAPERDEST= System Option
Specifies the name of the output bin to receive printed output.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Log and procedure output control: ODS Printing
PROC OPTIONS GROUP= ODSPRINT
Restrictions: This option is ignored if the printer does not have multiple output bins. This option is not valid in the Windows operating environment.
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
PAPERDEST=printer-bin-name

Syntax Description
printer-bin-name specifies the bin to receive printed output.
Restriction: Maximum length is 200 characters.

See Also
• “Universal Printing” in Chapter 15 of SAS Language Reference: Concepts

System Options:
• “PAPERSIZE= System Option” on page 198
• “PAPERSOURCE= System Option” on page 199
PAPERSIZE= System Option

Specifies the paper size to use for printing.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Environment control: Language control
Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP= LANGUAGECONTROL
ODSPRINT

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PAPERSIZE= paper_size_name | ("width_value"<,">height_value")
| (’width_value’<,’>height_value’) | (width_valueheight_value)

Syntax Description

dpaper_size_name
specifies a predefined paper size.

Default: Letter or A4, depending on the locale

Restriction: The maximum length is 200 characters.

Requirements:

When the name of a predefined paper size contains spaces, enclose the name in single or double quotation marks.

A space is required between the width and the height values if you do not use either single or double quotation marks for values.

Tip: Refer to the Registry Editor, or use PROC REGISTRY to obtain a listing of supported paper sizes. Additional values can be added.

("width_value","height_value")

specifies paper width and height as positive floating-point values.

Default: inches

Range: in or cm for width_value, height_value

Details

If you specify a predefined paper size or a custom size that is not supported by your printer, the printer default paper size is used. The printer default paper size is locale dependent and can be changed using the Page Setup dialog box.

Fields that specify values for paper sizes can either be separated by blanks or commas.

Note: Changing the value of this option can result in changes to the values of the portable LINESIZE= and PAGESIZE= system options.
Comparisons

The first OPTIONS statement sets a paper size value that is a paper size name from the SAS Registry. The second OPTIONS statement sets a specific width and height for a paper size.

```sas
options papersize="480x640 Pixels";
options papersize=('4.5' '7');
```

In the first example, quotation marks are required because there is a space in the name.

In the second example, quotation marks are not required. When no measurement units are specified, SAS writes the following warning to the SAS log:

```
WARNING: Units were not specified on the PAPERSIZE option. Inches will be used.
WARNING: Units were not specified on the PAPERSIZE option. Inches will be used.
```

You can avoid the warning message by adding the unit type, `in` or `cm`, to the value with no space separating the value and the unit type:

```sas
options papersize=(4.5in 7in);
```

See Also

- For information about declaring an ODS printer destination using ODS statements, see SAS Output Delivery System: User's Guide.

System Options:

- “PAPERDEST= System Option” on page 197
- “PAPERSOURCE= System Option” on page 199
- “PAPERTYPE= System Option” on page 200

---

PAPERSOURCE= System Option

Specifies the name of the paper bin to use for printing.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: ODS Printing

**PROC OPTIONS GROUP= ODSPRINT**

**Restriction:** This option is ignored if the printer does not have multiple input bins.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
PAPERSOURCE=printer-bin-name
```
**Syntax Description**

`printer-bin-name`

specifies the bin that sends paper to the printer.

**See Also**

- “Universal Printing” in Chapter 15 of *SAS Language Reference: Concepts*

**System Options:**

- “PAPERDEST= System Option” on page 197
- “PAPERSIZE= System Option” on page 198
- “PAPERTYPE= System Option” on page 200

---

**PAPERTYPE= System Option**

Specifies the type of paper to use for printing.

| Valid in: | Configuration file, SAS invocation, OPTIONS statement SAS System Options window |
| Category: | Log and procedure output control: ODS Printing |
| PROC OPTIONS GROUP= | ODSPRINT |

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

`PAPERTYPE=paper-type-string`

**Syntax Description**

`paper-type-string`

specifies the type of paper. Maximum length is 200.

**Default:** Values vary by site and operating environment.

**Range:** Values vary by printer, site, and operating environment.

**Operating environment:** For instructions on how to specify the type of paper, see the SAS documentation for your operating environment. There are a number of possible values for this option.

**See Also**

- “Printing with SAS” in Chapter 15 of *SAS Language Reference: Concepts*

**System Options:**

- “PAPERDEST= System Option” on page 197
PARM= System Option

Specifies a parameter string that is passed to an external program.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Files

PROC OPTIONS
GROUP= ENVFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

```PARM=<"string">```

Syntax Description

`string`

specifies a character string that contains a parameter.

Example

This statement passes the parameter X=2 to an external program:

```options parm='x=2';```

Operating Environment Information

Other methods of passing parameters to external programs depend on your operating environment and on whether you are running in interactive line mode or batch mode. For details, see the SAS documentation for your operating environment.

PARMCARDS= System Option

Specifies the file reference to open when SAS encounters the PARMCARDS statement in a procedure.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Files

PROC OPTIONS
GROUP= ENVFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “PARMCARDS= System Option: z/OS” in SAS Companion for z/OS
**Syntax**

PARMCARDS=file-ref

**Syntax Description**

file-ref

specifies the file reference to open.

**Details**

The PARMCARDS= system option specifies the file reference of a file that SAS opens when it encounters a PARMCARDS (or PARMCARDS4) statement in a procedure.

SAS writes all data lines after the PARMCARDS (or PARMCARDS4) statement to the file until it encounters a delimiter line of either one or four semicolons. The file is then closed and made available to the procedure to read. There is no parsing or macro expansion of the data lines.

---

**PDFACCESS System Option**

Specifies whether text and graphics from PDF documents can be read by screen readers for the visually impaired.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: PDF
- **PROC OPTIONS GROUP=** PDF
- **Requirement:** Adobe Acrobat Reader or Professional 5.0 and later versions
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

PDFACCESS | NOPDFACCESS

**Syntax Description**

PDFACCESS

specifies that text and graphics from a PDF document can be read by screen readers for the visually impaired. This is the default.

NOPDFACCESS

specifies that text and graphics from a PDF document cannot be read by screen readers for the visually impaired.

**Details**

The PDFACCESS option can affect the Content Accessibility Enabled document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.
The following table shows how the Content Accessibility Enabled document property is set when you specify the PDFACCESS option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th>PDFACCESS</th>
<th>NOPDFACCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY=LOW</td>
<td>PDFSECURITY=HIGH</td>
</tr>
<tr>
<td>Content Accessibility Enabled</td>
<td>Allowed</td>
</tr>
<tr>
<td></td>
<td>Allowed</td>
</tr>
<tr>
<td></td>
<td>Not Allowed</td>
</tr>
<tr>
<td></td>
<td>Allowed</td>
</tr>
</tbody>
</table>

See Also

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*

System Options:

- “PDFSECURITY= System Option” on page 213

**PDFASSEMBLY System Option**

Specifies whether PDF documents can be assembled.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: PDF

**PROC OPTIONS GROUP=** PDF

**Requirement:** Adobe Acrobat Reader or Professional 5.0 and later versions

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
PDFASSEMBLY | NOPDFASSEMBLY
```

**Syntax Description**

**PDFASSEMBLY**

specifies that PDF documents can be assembled.

**NOPDFASSEMBLY**

specifies that PDF documents cannot be assembled. This is the default.

**Details**

When a PDF document is assembled, pages can be rotated, inserted, and deleted, and bookmarks and thumbnail images can be added.
The PDFASSEMBLY option can affect the Document Assembly document property. The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Document Assembly document property is set when you specify the PDFASSEMBLY option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th>PDFASSEMBLY</th>
<th>NOPDFASSEMBLY</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY=LOW</td>
<td>PDFSECURITY=HIGH</td>
</tr>
<tr>
<td>Document Assembly</td>
<td>Allowed</td>
</tr>
</tbody>
</table>

See Also

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*

System Options:

- “PDFSECURITY= System Option” on page 213

PDFCOMMENT System Option

Specifies whether PDF document comments can be modified.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: PDF

PROC OPTIONS
GROUP= PDF

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PDFCOMMENT | NOPDFCOMMENT

Syntax Description

PDFCOMMENT
specifies that PDF document comments can be modified.

NOPDFCOMMENT
specifies that PDF document comments cannot be modified. This is the default.
Details

The PDFCOMMENT option can affect the Commenting document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY=LOW, the setting of the Form Field Fill-in or Signing document property is dependent on the PDFCOMMENT option. A change in the PDFCOMMENT option changes the Form Field Fill-in or Signing document property to be the same setting as the Commenting document property. For example, if PDFSECURITY=LOW and PDFCOMMENT is specified, the Commenting and the Form Field Fill-in or Signing document properties are both set to Allowed. If NOPDFCOMMENT is specified, both document properties are set to Not Allowed. The Commenting document property is not affected by the PDFFILLIN option. The value of the PDFCOMMENT option does not affect the value of the PDFFILLIN option.

When PDFSECURITY=HIGH, PDFCOMMENT and PDFFILLIN can be set independently.

The following table shows how the Commenting and Form Field Fill-in or Signing document properties are set when you specify the PDFCOMMENT option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th>PDFCOMMENT</th>
<th>NOPDFCOMMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY =LOW</td>
<td>PDFSECURITY =HIGH</td>
</tr>
<tr>
<td>Commenting</td>
<td>Allowed</td>
</tr>
<tr>
<td>Form Field Fill-in or Signing</td>
<td>Allowed</td>
</tr>
</tbody>
</table>

See Also


System Options:

- “PDFFILLIN System Option” on page 207
- “PDFSECURITY= System Option” on page 213

PDFCONTENT System Option

Specifies whether the contents of a PDF document can be changed.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: PDF

PROC OPTIONS GROUP= PDF

Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PDFCONTENT | NOPDFCONTENT

Syntax Description

PDFCONTENT
specifies that the contents of a PDF document can be changed.

NOPDFCONTENT
specifies that the contents of a PDF document cannot be changed. This is the default.

Details

The PDFCONTENT option can affect the Changing the Document document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

The following table shows how the Changing the Document document property is set when you specify the PDFCONTENT option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th>PDFCONTENT</th>
<th>NOPDFCONTENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY = LOW</td>
<td>PDFSECURITY = HIGH</td>
</tr>
<tr>
<td>Changing the Document</td>
<td>Not applicable*</td>
</tr>
</tbody>
</table>

* The Changing the Document property is not set when PDFSECURITY=LOW. The option value changes as specified, but the property value remains as the value that was last set.

See Also


System Options:

- “PDFSECURITY= System Option” on page 213

PDFCOPY System Option

Specifies whether text and graphics from a PDF document can be copied.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: PDF

PROC OPTIONS GROUP= PDF
Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

PDFCOPY | NOPDFCOPY

**Syntax Description**

**PDFCOPY**  
specifies that text and graphics from a PDF document can be copied. This is the default.

**NOPDFCOPY**  
specifies that text and graphics from a PDF document cannot be copied.

**Details**

The PDFCOPY option can affect the Content Copying document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY=LOW and PDFCOPY or NOPDFCOPY is specified, the option value is set, but the value of the Content Copying property does not change.

The following table shows how the Content Copying document property is set when you specify the PDFCOPY option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th></th>
<th>PDFCOPY</th>
<th>NOPDFCOPY</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY=LOW</td>
<td>Content Copying Not applicable*</td>
<td>Allowed</td>
</tr>
<tr>
<td>PDFSECURITY=HIGH</td>
<td>Not applicable*</td>
<td>Not Allowed</td>
</tr>
</tbody>
</table>

* The Content Copying property is not set when PDFSECURITY=LOW. The option value changes as specified, but the property value remains as the value that was last set.

**See Also**

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*

**System Options:**

- “PDFSECURITY= System Option” on page 213

**PDFFILLIN System Option**

Specifies whether PDF forms can be filled in.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Log and procedure output control: PDF

PROC OPTIONS
GROUP= PDF

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PDFFILLIN | NOPDFFILLIN

Syntax Description

PDFFILLIN
specifies that PDF forms can be filled in. This is the default.

NOPDFFILLIN
specifies that PDF forms cannot be filled in.

Details

The PDFFILLIN option can affect the Form Field Fill-in or Signing document property.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY=LOW, the settings of the Form Field Fill-in or Signing document property is dependent on the PDFCOMMENT option. A change in the PDFCOMMENT option changes the Form Field Fill-in or Signing document property to be the same setting as the Commenting document property. For example, if PDFSECURITY=LOW and PDFCOMMENT is specified, the Commenting and the Form Field Fill-in or Signing document properties are both set to Allowed. If NOPDFCOMMENT is specified, both document properties are set to Not Allowed. The Commenting document property is not affected by the PDFFILLIN option. The value of the PDFCOMMENT option does not affect the value of the PDFFILLIN option.

The Form Field Fill-in or Signing document property is set by the PDFFILLIN option only when PDFSECURITY=HIGH. When PDFSECURITY=HIGH, PDFCOMMENT and PDFFILLIN can be set independently.

The following table shows how the Form Field Fill-in or Signing document property is set when you specify the PDFFILLIN option and set the PDFSECURITY= option to LOW or HIGH:

<table>
<thead>
<tr>
<th>PDFFILLIN</th>
<th>NOPDFFILLIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFSECURITY =LOW</td>
<td>PDFSECURITY =HIGH</td>
</tr>
<tr>
<td>Form Field Fill-in or Signing</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

* The Field From Fill-in or Signing property is not set when PDFSECURITY=LOW. The property value remains as the value that was last set.
See Also

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*

System Options:

- “PDFCOMMENT System Option” on page 204
- “PDFSECURITY= System Option” on page 213

**PDFPAGELAYOUT= System Option**

Specifies the page layout for PDF documents.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: PDF

**PROC OPTIONS**

**GROUP= PDF**

**Requirement:** Adobe Acrobat Reader or Professional 5.0 and later versions

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
PDFPAGELAYOUT= DEFAULT | SINGLEPAGE | CONTINUOUS
               | FACING | CONTINUOUSFACING
```

**Syntax Description**

**DEFAULT**

specifies to use the current page layout for Acrobat Reader. This is the default.

**SINGLEPAGE**

specifies to display one page at a time in the viewing area.

**CONTINUOUS**

specifies to display all document pages in the viewing area in a single column.

**FACING**

specifies to display only two pages in the viewing area, with the even pages on the left and the odd pages on the right.

**Requirement:** Acrobat Reader 5.0 or later version is required.

**CONTINUOUSFACING**

specifies to display all pages in the viewing area, two pages side by side. The even pages display on the left, and the odd pages display on the right.

See Also

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*
PDFPAGEVIEW= System Option

Specifies the page viewing mode for PDF documents.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: PDF

PROC OPTIONS GROUP= PDF

Requirement: Adobe Acrobat Reader or Professional 5.0 and later versions

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PDFPAGEVIEW = DEFAULT | ACTUAL | FITPAGE | FITWIDTH | FULLSCREEN

Syntax Description

DEFAULT
specifies to use the current page view setting for Acrobat Reader. This is the default.

ACTUAL
specifies to set the page view setting to 100%.

FITPAGE
specifies to view a page using the full extent of the viewing window, maintaining the height and width aspect ratio.

FITWIDTH
specifies to view a page using the full width of the viewing window. The height of the document is not scaled to fit the page.

FULLSCREEN
specifies to view a page using the full screen. This option disables the table of contents, bookmarks, and all other document access aids, such as accessing a specific page.

See Also

• “Securing ODS Generated PDF Files” in Chapter 3 of SAS Output Delivery System: User's Guide

System Options:

• “PDFPAGEVIEW= System Option” on page 210
PDFPASSWORD= System Option

Specifies the password to use to open a PDF document and the password used by a PDF document owner.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement

**Categories:** Log and procedure output control: PDF
System administration: Security

**PROC OPTIONS GROUP=** PDF

**Alias:** PDFPW

**Requirement:** Adobe Acrobat Reader or Professional 3.0 and later versions

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```plaintext
PDFPASSWORD=(OPEN=password | OPEN="password"
<OWNER=password | OWNER="password">)

PDFPASSWORD=(OWNER=password | OWNER="password"
<OPEN=password | OPEN="password">)

PDFPASSWORD=(OPEN=password | OPEN="password")

PDFPASSWORD=(OWNER=password | OWNER="password")
```

**Syntax Description**

**OPEN="password"**

specifies the password to open a PDF document. Enclosing the password in single or double quotation marks is optional.

password

specifies a set of characters, up to 32 characters, that are used to validate that a user has permission to open a PDF document.

**Restrictions:**

The OPEN password must be different from the OWNER password.
Password values cannot be set to a null value ("") or a blank character.

**OWNER="password"**

specifies the password for the PDF document owner. Enclosing the password in quotation marks is optional.

password

specifies a set of characters, up to 32 characters, that are used to validate the owner of a PDF document.

**Restrictions:**

The OWNER password must be different from the OPEN password.
Password values cannot be set to a null value ("") or a blank character.
Details
You can set the PDFPASSWORD option at any time, but it is ignored until the
PDFSECURITY system option is set to either LOW or HIGH. When the
PDFSECURITY option is set to NONE, passwords for a PDF document are not needed.

See Also
• “Securing ODS Generated PDF Files” in Chapter 3 of SAS Output Delivery System:
  User's Guide

System Options:
• “ PDFPAGEVIEW= System Option” on page 210
• “PDFSECURITY= System Option” on page 213

PDFPRINT= System Option
Specifies the resolution to print PDF documents.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options
Category: Log and procedure output control: PDF
PROC OPTIONS GROUP= PDF
Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions, depending on
PDFPRINT setting
Note: This option can be restricted by a site administrator. For more information, see
“Restricted Options” on page 6.

Syntax
PDFPRINT= HRES | LRES | NONE

Syntax Description
HRES
specifies to print PDF documents at the highest resolution available on the printer.
This is the default for Acrobat Reader or Professional 5.0 and later versions.
Restriction: PDFPRINT=HRES can be set only when the PDFSECURITY option is
set to HIGH.
Requirement: Acrobat Reader or Professional 5.0 and later versions.

LRES
specifies to print PDF documents at a lower resolution for draft-quality documents.
Restriction: PDFPRINT=LRES can be set only when the PDFSECURITY option is
set to HIGH.
Requirement: Acrobat Reader or Professional 3.0 and later versions.

NONE
specifies the PDF documents have no print resolution.
Restriction: PDFPRINT=NONE can be set only when the PDFSECURITY option is set to HIGH or LOW.

Requirement: Any version of Acrobat Reader or Professional.

Details

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

When PDFSECURITY= is set to LOW or HIGH, the value of the Printing document property is determined by the value of the PDFPRINT= option:

<table>
<thead>
<tr>
<th>PDFPRINT</th>
<th>PDFSECURITY=LOW</th>
<th>PDFSECURITY=HIGH</th>
</tr>
</thead>
<tbody>
<tr>
<td>HRES</td>
<td>High Resolution</td>
<td>High Resolution</td>
</tr>
<tr>
<td>LRES</td>
<td>High Resolution</td>
<td>Low Resolution (150 dpi)</td>
</tr>
<tr>
<td>NONE</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>

The Printing document property is set to High Resolution when PDFPRINT= is set to HRES or LRES, because the Printing document property can be set to High Resolution only when PDFSECURITY=LOW.

See Also


System Options:

- “PDFPAGEVIEW= System Option” on page 210

PDFSECURITY= System Option

Specifies the level of encryption for PDF documents.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Log and procedure output control: PDF
System administration: Security

**PROC OPTIONS**

GROUP= PDF SECURITY

Restriction: The PDFSECURITY option is valid for UNIX, Windows, and z/OS operating systems, but only in countries where importing encryption software is legal.

Requirement: Adobe Acrobat Reader or Professional 3.0 and later versions, unless otherwise noted.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

PDFSECURITY= HIGH | LOW | NONE

**Syntax Description**

**HIGH**

specifies that SAS encrypts PDF documents using a 128-bit encryption algorithm.

**Requirement:** When PDFSECURITY=HIGH, you must use Acrobat 5.0 or later version.

**Interaction:** At least one password must be set using the PDFPASSWORD= system option when PDFSECURITY=HIGH or LOW.

**LOW**

specifies that SAS encrypts PDF documents using a 40-bit encryption algorithm.

**Interaction:** At least one password must be set using the PDFPASSWORD= system option when PDFSECURITY=HIGH or LOW.

**NONE**

specifies that no encryption is performed on PDF documents. This is the default.

The document property values for PDF security are not changed when you set PDFSECURITY=NONE. The results are the same as not specifying the option at all.

**Details**

The following table shows the default document properties that are set when the PDFSECURITY= option is set to NONE, LOW, or HIGH. When PDFSECURITY=NONE, there are no restrictions on PDF documents.

<table>
<thead>
<tr>
<th></th>
<th>NONE</th>
<th>LOW</th>
<th>HIGH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printing</td>
<td>Allowed</td>
<td>High Resolution</td>
<td>High Resolution</td>
</tr>
<tr>
<td>Changing the Document</td>
<td>Allowed</td>
<td>Not Allowed</td>
<td>Not Allowed</td>
</tr>
<tr>
<td>Commenting</td>
<td>Allowed</td>
<td>Not Allowed</td>
<td>Not Allowed</td>
</tr>
<tr>
<td>Form Field Fill-in or Signing</td>
<td>Allowed</td>
<td>Not Allowed</td>
<td>Allowed</td>
</tr>
<tr>
<td>Document Assembly</td>
<td>Allowed</td>
<td>Not Allowed</td>
<td>Not Allowed</td>
</tr>
<tr>
<td>NONE</td>
<td>LOW</td>
<td>HIGH</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>Content Copying</td>
<td>Allowed</td>
<td>Allowed</td>
<td>Allowed</td>
</tr>
<tr>
<td>Content Accessibility Enabled</td>
<td>Allowed</td>
<td>Allowed</td>
<td>Allowed</td>
</tr>
<tr>
<td>Page Extraction</td>
<td>Allowed</td>
<td>Allowed</td>
<td>Allowed</td>
</tr>
<tr>
<td>Encryption Level</td>
<td>None</td>
<td>40-bit RC4</td>
<td>128-bit RC4</td>
</tr>
</tbody>
</table>

Encryption Level

- None
- 40-bit RC4
- 128-bit RC4

See Also

- “Securing ODS Generated PDF Files” in Chapter 3 of *SAS Output Delivery System: User's Guide*

System Options:

- “PDFACCESS System Option” on page 202
- “PDFASSEMBLY System Option” on page 203
- “PDFCOMMENT System Option” on page 204
- “PDFCONTENT System Option” on page 205
- “PDFCOPY System Option” on page 206
- “PDFFILLIN System Option” on page 207
- “PDFPASSWORD= System Option” on page 211
- “PDFPRINT= System Option” on page 212

---

**PRIMARYPROVIDERDOMAIN= System Option**

Specifies the domain name of the primary authentication provider.

**Valid in:** Configuration file, SAS invocation

**Category:** Environment control: Initialization and operation

**PROC OPTIONS GROUP=** EXECMODES

**Alias:** PRIMPD=

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
PRIMARYPROVIDERDOMAIN=domain-name
```
**Syntax Description**

domain-name  
specifies the name of the domain that authenticates user names.

**Requirement:** If the domain name contains one or more spaces, the domain name must be enclosed in quotation marks.

**Details**

By default, users who log on to the SAS Metadata Server are authenticated by the operating system that hosts the SAS Metadata Server. You can specify an alternate authentication provider by using the AUTHPROVIDERDOMAIN= system option. User IDs that are verified by an alternate authentication provider must be in the format user-ID@domain-name (for example, user1@sas.com).

By specifying an authentication provider and a domain name that use the AUTHPROVIDERDOMAIN= and PRIMARYPROVIDERDOMAIN= system options, respectively, you enable users to log on to the SAS Metadata Server by using their usual user ID without using a domain-name suffix on the user ID. For example, by specifying the following system options, users who log on as user-ID or user-ID@mycompany.com can be verified by the authentication provider that is specified by the AUTHPROVIDERDOMAIN= system option:

- authproviderdomain ldap:mycompany
- primaryproviderdomain mycompany.com

If you specify the PRIMARYPROVIDERDOMAIN system option without specifying the AUTHPROVIDERDOMAIN system option, authentication is performed by the host provider.

**Comparisons**

You use the AUTHPROVIDERDOMAIN system option to register and name your Active Directory provider or other LDAP provider. You use the PRIMARYPROVIDERDOMAIN system option to designate the primary authentication provider.

**Example**

The following examples show the system options that you might use in a configuration file to define a primary authentication provider domain-name:

**Active Directory**

/* Environment variables that describe your Active Directory server */
-set AD_HOST myhost
/* Define authentication provider */
-authpd ADIR:mycompany.com
-primpd mycompany.com

**LDAP**

/* Environment variables that describe your LDAP server */
-set LDAP_HOST myhost
-set LDAP_BASE "ou=emp, o=us"
/* Define authentication provider */
-authpd LDAP:mycompany.com
-primpd mycompany.com
See Also

- “Direct LDAP Authentication” in SAS Intelligence Platform: Security Administration Guide

System Options:

- “AUTHPROVIDERDOMAIN System Option” on page 70
- “AUTHSERVER System Option: Windows” in SAS Companion for Windows

PRINTERPATH= System Option

Specifies the name of a registered printer to use for Universal Printing.

Valid in:
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category:
- Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PRINTERPATH=(’printer-name’<fileref> )

Syntax Description

’printer-name’
- must be one of the printers defined in the Registry Editor under CorePrintingPrinters
- Requirement: When the printer name contains blanks, you must enclose it in quotation marks.

fileref
- is an optional fileref. If a fileref is specified, it must be defined with a FILENAME statement or an external allocation. If a fileref is not specified, the default output destination can specify a printer in the Printer Setup dialog box, which you open by selecting FilePrinter Setup. Parentheses are required only when a fileref is specified.

Details

If the PRINTERPATH= option is not a null string, then Universal Printing will be used. If the PRINTERPATH= option does not specify a valid Universal Printing printer, then the default Universal Printer is used.

Comparisons

A related system option SYSPRINT specifies which operating system printer will be used for printing. PRINTERPATH= specifies which Universal Printing printer will be used for printing.
The operating system printer specified by the SYSPRINT option is used when PRINTERPATH="" (two double quotation marks with no space between them sets a null string).

Example

The following example specifies an output destination that is different from the default:

options PRINTERPATH=(corelab out);
filename out 'your_file';

Operating Environment Information

In some operating environments, setting the PRINTERPATH= option might not change the setting of the PMENU print button, which might continue to use operating environment printing. See the SAS documentation for your operating environment for more information.

The PRINTERPATH option is used only for ODS PRINTER and when the DEVICE= system option is set to SASPRTC, SASPRTRTG, SASPRTM, or SASPRT. If DEVICE=WINPRTC, WINPRTG, or WINPRTM, the devices behave respectively as SASPRTC, SASPRTG, or SASPRTM.

See Also


PRINTINIT System Option

Specifies whether to initialize the SAS procedure output file for the LISTING destination.

Valid in: Configuration file, SAS invocation
Category: Log and procedure output control: Procedure output
PROC OPTIONS GROUP= LISTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “PRINTINIT System Option: z/OS” in SAS Companion for z/OS

Syntax

PRINTINIT | NOPRINTINIT

Syntax Description

PRINTINIT specifies to initialize the SAS procedure output file for the LISTING destination and resets the file attributes.

Tip: Specifying PRINTINIT causes the SAS procedure output file to be cleared even when output is not generated.
NOPRINTINIT
 specifies to preserve the existing procedure output file for the LISTING destination if no new output is generated. This is the default.

Tip: Specifying NOPRINTINIT causes the SAS procedure output file to be overwritten only when new output is generated.

Details

Operating Environment Information
The behavior of the PRINTINIT system option depends on your operating environment. For additional information, see the SAS documentation for your operating environment.

PRINTMSGLIST System Option
Specifies whether to print all messages to the SAS log or to print only top-level messages to the SAS log.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Log and procedure output control: SAS log
PROC OPTIONS GROUP=
     LOGCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

PRINTMSGLIST | NOPRINTMSGLIST

Syntax Description

PRINTMSGLIST
 specifies to print the entire list of messages to the SAS log. PRINTMSGLIST is the default.

NOPRINTMSGLIST
 specifies to print only the top-level message to the SAS log.

Details

For Version 7 and later versions, the return code subsystem allows for lists of return codes. All of the messages in a list are related, in general, to a single error condition, but give different levels of information. This option enables you to see the entire list of messages or just the top-level message.

See Also

QUOTELENMAX System Option

If a quoted string exceeds the maximum length allowed, specifies whether SAS writes a warning message to the SAS log.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
Environment control: Error handling

**PROC OPTIONS GROUP=**
ERRORHANDLING

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

QUOTELENMAX | NOQUOTELENMAX

**Syntax Description**

**QUOTELENMAX**
- specifies that SAS write a warning message to the SAS log about the maximum length for strings in quotation marks.

**NOQUOTELENMAX**
- specifies that SAS does not write a warning message to the SAS log about the maximum length for strings in quotation marks.

**Details**

If a string in quotation marks is too long, SAS writes the following warning to the SAS log:

WARNING 32-169: The quoted string currently being processed has become more than 262 characters long. You may have unbalanced quotation marks.

If you are running a program that has long strings in quotation marks, and you do not want to see this warning, use the NOQUOTELENMAX system option to turn off the warning.

REPLACE System Option

Specifies whether permanently stored SAS data sets can be replaced.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
Files: SAS Files

**PROC OPTIONS GROUP=**
SASFILES

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

REPLACE | NOREPLACE

Syntax Description

REPLACE

specifies that a permanently stored SAS data set can be replaced with another SAS data set of the same name.

NOREPLACE

specifies that a permanently stored SAS data set cannot be replaced with another SAS data set of the same name, which prevents the accidental replacement of existing SAS data sets.

Details

This option has no effect on data sets in the WORK library, even if you use the WORKTERM= system option to store the WORK library files permanently.

Comparisons

The REPLACE= data set option overrides the REPLACE system option.

See Also

Data Set Options:

• “PARMCommands= System Option: z/OS” in SAS Companion for z/OS

System Options:

• “WORKTERM System Option” on page 296

---

REUSE= System Option

Specifies whether SAS reuses space when observations are added to a compressed SAS data set.

Valid in:

Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category:

Files: SAS Files

PROC OPTIONS GROUP=

SASFILES

Interaction:

The REUSE= data set option overrides the REUSE= system option.

Note:

This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

REUSE=YES | NO
Syntax Description

YES
specifies to track free space and reuses it whenever observations are added to an existing compressed data set.

Interactions:
REUSE=YES takes precedence over the POINTOBS=YES data set option setting.
When using COMPRESS=YES and REUSE=YES system options settings, observations cannot be addressed by observation number.

NO
specifies not to track free space. This is the default.

Details

If space is reused, observations that are added to the SAS data set are inserted wherever enough free space exists, instead of at the end of the SAS data set.

Specifying REUSE=NO results in less efficient usage of space if you delete or update many observations in a SAS data set. However, the APPEND procedure, the FSEDIT procedure, and other procedures that add observations to the SAS data set continue to add observations to the end of the data set, as they do for uncompressed SAS data sets.

You cannot change the REUSE= attribute of a compressed SAS data set after it is created. Space is tracked and reused in the compressed SAS data set according to the REUSE= value that was specified when the SAS data set was created, not when you add and delete observations. Even with REUSE=YES, the APPEND procedure will add observations at the end.

See Also

Data Set Options:
- “COMPRESS= Data Set Option” in SAS Data Set Options: Reference
- “REUSE= Data Set Option” in SAS Data Set Options: Reference

System Options:
- “COMPRESS= System Option” on page 99

RIGHTMARGIN= System Option

Specifies the print margin for the right side of the page.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Default: 0.00 in

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

RIGHTMARGIN=margin-size<margin-unit>

Syntax Description

margin-size
specifies the size of the margin.

Restriction: The right margin should be small enough so that the left margin plus
the right margin is less than the width of the paper.

Interaction: Changing the value of this option might result in changes to the value
of the LINESIZE= system option.

<margin-unit>
specifies the units for margin-size. The margin-unit can be in for inches or cm for
centimeters. <margin-unit> is saved as part of the value of the RIGHTMARGIN
system option.

Default: inches

Details

All margins have a minimum that is dependent on the printer and the paper size.

See Also

• For information about declaring an ODS printer destination using ODS statements,

System Options:

• “BOTTOMMARGIN= System Option” on page 76
• “LEFTMARGIN= System Option” on page 169
• “TOPMARGIN= System Option” on page 269

RLANG System Option

Specifies whether SAS executes R language statements.

Valid in: Configuration file, SAS invocation

Category: System administration: Security

PROC OPTIONS GROUP=SECURITY

Note: This option can be restricted by a site administrator. For more information, see
“Restricted Options” on page 6.

Syntax

RLANG | NORLANG
Syntax Description

RLANG
specifies that SAS can execute R language statements in operating environments that support the R language.

NORLANG
specifies that SAS is not to execute R language statements. This is the default value.

Details
If RLANG is specified and the R language is not supported in the operating environment, SAS writes a message to the SAS log. The message indicates that the R language is not supported and asks you to call SAS Technical Support. SAS Technical Support would like to track the operating environments where users would like SAS to execute R language statements, but the R language is not supported.

See Also
SAS/IML User's Guide

RSASUSER System Option

Specifies whether to open the Sasuser library for Read access or Read-Write access.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “RSASUSER System Option: UNIX” in SAS Companion for UNIX Environments
“RSASUSER System Option: Windows” in SAS Companion for Windows

Syntax

RSASUSER | NORSASUSER

Syntax Description

RSASUSER
opens the SASUSER library in read-only mode.

NORSASUSER
opens the SASUSER library in read-write mode.

Details
The RSASUSER system option is useful for sites that use a single SASUSER library for all users and want to prevent users from modifying it. However, it is not useful when users use SAS/ASSIST software, because SAS/ASSIST requires writing to the SASUSER library.

Operating Environment Information
For network considerations about using the RSASUSER system option, see the SAS documentation for your operating environment.

---

**S= System Option**

Specifies the length of statements on each line of a source statement and the length of data on lines that follow a DATALINES statement.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Input control: Data processing

**PROC OPTIONS GROUP=** INPUTCONTROL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

S=\(n | nK | nM | nG | nT | hexX | \text{MIN} | \text{MAX}\)

**Syntax Description**

\(n | nK | nM | nG | nT\)

specifies the length of statements and data in terms of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes).

For example, a value of \(8\) specifies 8 bytes, and a value of \(3m\) specifies 3,145,728 bytes.

\(hexX\)

specifies the length of statements and data as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an X. For example, the value \(2dx\) sets the length of statements and data to 45.

**MIN**

sets the length of statements and data to 0.

**MAX**

sets the length of statements and data to 2,147,483,647.

---

**Details**

Input can be from either fixed-length or variable-length records. Both fixed-length and variable-length records can be sequenced or unsequenced. The location of the sequence numbers is determined by whether the file record format is fixed-length or variable-length.

SAS uses the value of S to determine whether to look for sequence numbers in the input, and to determine how to read the input:
<table>
<thead>
<tr>
<th>Record Type</th>
<th>Value of S</th>
<th>SAS Looks for Sequence Numbers</th>
<th>How SAS Reads The Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed-length</td>
<td>S&gt;0 or S=MAX</td>
<td>No</td>
<td>The value of S is used as the length of the source or data to be scanned and ignores everything beyond that length on each line.</td>
</tr>
<tr>
<td>Fixed-length</td>
<td>S=0 or S=MIN</td>
<td>Yes, at the end of the line of input.</td>
<td>SAS inspects the last ( n ) columns (where ( n ) is the value of the SEQ= system option) of the first sequence field. If those columns contain numbers, they are assumed to be sequence numbers and SAS ignores the last eight columns of each line. If the ( n ) columns contain non-digit characters, SAS reads the last eight columns as data columns.</td>
</tr>
<tr>
<td>Variable-length</td>
<td>S&gt;0 or S=MAX</td>
<td>No</td>
<td>The value of S is used as the starting column of the source or data to be scanned and ignores everything before that length on each line.</td>
</tr>
<tr>
<td>Record Type</td>
<td>Value of S</td>
<td>SAS Looks for Sequence Numbers</td>
<td>How SAS Reads The Input</td>
</tr>
<tr>
<td>------------------</td>
<td>------------</td>
<td>--------------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Variable-length</td>
<td>S=0 or S=MIN</td>
<td>Yes, at the beginning of each line of input.</td>
<td>SAS inspects the last ( n ) columns (where ( n ) is the value of the SEQ= system option) of the first sequence field. If those columns contain numbers, they are assumed to be sequence numbers and SAS ignores the first eight columns of each line. If the ( n ) columns contain non-digit characters, SAS reads the first eight columns as data columns.</td>
</tr>
</tbody>
</table>

**Comparisons**

The S= system option operates exactly like the S2= system option except that S2= controls input only from a %INCLUDE statement, an autoexec file, or an autocall macro file.

**See Also**

**System Options:**

- “S2= System Option” on page 227
- “S2V= System Option” on page 230
- “SEQ= System Option” on page 233

**S2= System Option**

Specifies the length of statements on each line of a source statement from a %INCLUDE statement, an AUTOEXEC= file, or an autocall macro file.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Input control: Data processing

**PROC OPTIONS GROUP=** INPUTCONTROL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

S2= S | n | nK | nM | nG | nT | hexX | MIN | MAX

Syntax Description

S
uses the current value of the S= system option to compute the record length of text that comes from a %INCLUDE statement, an AUTOEXEC= file, or an autocall macro file.

n | nK | nM | nG | nT
specifies the length of the statements in a file that is specified in a %INCLUDE statement, an autoexec file, or an autocall macro file, in terms of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 8 specifies 8 bytes, and a value of 3m specifies 3,145,728 bytes.

hexX
specifies the length of statements as a hexadecimal number. You must specify the value beginning with a number (0-9), followed by an X. For example, the value 2dx sets the length of statements to 45.

MIN
sets the length of statements and data to 0.

MAX
sets the length of statements and data to 2,147,483,647.

Details

Input can be from either fixed-length or variable-length records. Both fixed-length and variable-length records can be sequenced or unsequenced. The location of the sequence numbers is determined by whether the file record format is fixed-length or variable-length.

SAS uses the value of S2 to determine whether to look for sequence numbers in the input, and to determine how to read the input:

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Value of S2</th>
<th>SAS Looks for Sequence Numbers</th>
<th>How SAS Reads The Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed-length</td>
<td>S2&gt;0 or S2=MAX</td>
<td>No</td>
<td>The value of S2 is used as the length of the source or data to be scanned and ignores everything beyond that length on each line.</td>
</tr>
<tr>
<td>Record Type</td>
<td>Value of S2</td>
<td>SAS Looks for Sequence Numbers</td>
<td>How SAS Reads The Input</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------</td>
<td>--------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Fixed-length</td>
<td>S2=0 or S2=MIN</td>
<td>Yes, at the end of the line of input.</td>
<td>SAS inspects the last $n$ columns (where $n$ is the value of the SEQ= system option) of the first sequence field. If those columns contain numbers, they are assumed to be sequence numbers and SAS ignores the last eight columns of each line. If the $n$ columns contain non-digit characters, SAS reads the last eight columns as data columns.</td>
</tr>
<tr>
<td>Variable-length</td>
<td>S2&gt;0 or S2=MAX</td>
<td>No</td>
<td>The value of S2 is used as the starting column of the source or data to be scanned and ignores everything before that length on each line.</td>
</tr>
<tr>
<td>Variable-length</td>
<td>S2=0 or S2=MIN</td>
<td>Yes, at the beginning of each line of input.</td>
<td>SAS inspects the last $n$ columns (where $n$ is the value of the SEQ= system option) of the first sequence field. If those columns contain numbers, they are assumed to be sequence numbers and SAS ignores the first eight columns of each line. If the $n$ columns contain non-digit characters, SAS reads the first eight columns as data columns.</td>
</tr>
</tbody>
</table>

**Comparisons**

The S2= system option operates exactly like the S= system option except that the S2= option controls input from a %INCLUDE statement, an autoexec file, or an autocall macro file.
The S2= system option reads both fixed-length and variable-length record formats from a file specified in a %INCLUDE statement, an autoexec file, or an autocall macro file. The S2V= system option reads only a variable-length record format from a file specified in a %INCLUDE statement, an autoexec file, or an autocall macro file.

**See Also**

**System Options:**
- “S= System Option” on page 225
- “S2V= System Option” on page 230
- “SEQ= System Option” on page 233

---

**S2V= System Option**

Specifies the starting position to begin reading a file that is specified in a %INCLUDE statement, an autoexec file, or an autocall macro file with a variable length record format.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Input control: Data processing

**PROC OPTIONS**

**GROUP=** INPUTCONTROL

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

S2V= S2 | S | n | nK | nM | nG | nT | MIN | MAX | hexX

**Syntax Description**

- **S2**
  
  Specifies to use the current value of the S2= system option to compute the starting position of the variable-sized record to read from a %INCLUDE statement, an autoexec file, or an autocall macro file. This is the default.

- **S**
  
  Specifies to use the current value of the S= system option to compute the starting position of the variable-sized record to read from a %INCLUDE statement, an autoexec file, or an autocall macro file.

- **n | nK | nM | nG | nT**
  
  Specifies the starting position of the variable-length record to read that comes from a %INCLUDE statement, an autoexec file, or an autocall macro file, in terms of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 8 specifies 8 bytes, and a value of 3m specifies 3,145,728 bytes.

- **MIN**
  
  Sets the starting position of the variable-length record to read that comes from a %INCLUDE statement, an autoexec file, or an autocall macro, to 0.
MAX
sets the starting position of the variable-length record to read that comes from a
%INCLUDE statement, an autoexec file, or an autocall macro, to 2,147,483,647.

hexX
specifies the starting position of the variable-length record to read that comes from a
%INCLUDE statement, an autoexec file, or an autocall macro, as a hexadecimal
number. You must specify the value beginning with a number (0–9), followed by an
X.

Details
Both the S2V= system option and the S2= system option specify the starting position for
reading variable-sized record input from a %INCLUDE statement, an autoexec file, or
an autocall macro file. When values for both options are specified, the value of the S2V=
��统 option takes precedence over the value specified for the S2= system option.

Comparisons
The S2= system option specifies the starting position for reading both fixed-length and
variable-length record formats for input from a %INCLUDE statement, an autoexec file,
or an autocall macro file. The S2V= system option specifies the starting position for
reading only variable-length record formats for input from a %INCLUDE statement, an
autoexec file, or an autocall macro file.

See Also
System Options:

• “S= System Option” on page 225
• “S2= System Option” on page 227
• “SEQ= System Option” on page 233

SASHELP= System Option
Specifies the location of the Sashelp library.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see
“Restricted Options” on page 6.

Tip: You can use the APPEND or INSERT system options to add additional library-
specifications.

See: “SASHELP System Option: UNIX” in SAS Companion for UNIX Environments
“SASHELP System Option: Windows” in SAS Companion for Windows
“SASHELP= System Option: z/OS” in SAS Companion for z/OS
Syntax
SASHELP=library-specification

Syntax Description
library-specification
identifies an external library.

Details
The SASHELP= system option is set during the installation process and normally is not changed after installation.

Operating Environment Information
A valid external library specification is specific to your operating environment. On the command line or in a configuration file, the syntax is specific to your operating environment. For details, see the SAS documentation for your operating environment.

See Also

System Options:
• “APPEND= System Option” on page 68
• “INSERT= System Option” on page 159

SASUSER= System Option
Specifies the SAS library to use as the Sasuser library.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files
PROC OPTIONS GROUP= ENVFILES

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

See: “SASUSER System Option: UNIX” in SAS Companion for UNIX Environments,
“SASUSER System Option: Windows” in SAS Companion for Windows,
“SASUSER= System Option: z/OS” in SAS Companion for z/OS

Syntax
SASUSER=library-specification

Syntax Description
library-specification
specifies the libref or the physical name that contains a user's Profile catalog.
Details
The library and catalog are created automatically by SAS; you do not have to create
them explicitly.

SEQ= System Option
Specifies the length of the numeric portion of the sequence field in input source lines or data lines.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options
window
Category: Input control: Data processing

PROC OPTIONS
GROUP=

Note: This option can be restricted by a site administrator. For more information, see
“Restricted Options” on page 6.

Syntax
SEQ= n | MIN | MAX | hexX

Syntax Description

n
specifies the length in terms of bytes.

MIN
sets the minimum length to 1.

MAX
sets the maximum length to 8.

Tip: When SEQ=8, all eight characters in the sequence field are assumed to be
numeric.

hexX
specifies the length as a hexadecimal. You must specify the value beginning with a
number (0–9), followed by an X.

Details
Unless the S= or S2= system option specifies otherwise, SAS assumes an eight-character
sequence field. However, some editors place some alphabetic information (for example,
the filename) in the first several characters. The SEQ= value specifies the number of
digits that are right-justified in the eight-character field. For example, if you specify
SEQ=5 for the sequence field AAA00010, SAS looks at only the last five characters of
the eight-character sequence field and, if the characters are numeric, treats the entire
eight-character field as a sequence field.

See Also

System Options:

• “S= System Option” on page 225
• “S2= System Option” on page 227
SETINIT System Option

Specifies whether site license information can be altered.

Valid in: Configuration file, SAS invocation
Category: System administration: Installation

PROC OPTIONS
GROUP= INSTALL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SETINIT | NOSETINIT

Syntax Description

SETINIT

in a non-windowing environment, specifies that you can change license information by running the SETINIT procedure.

NOSETINIT

specifies not to enable you to alter site license information after installation.

Details

SETINIT is set in the installation process and is not normally changed after installation. The SETINIT option is valid only in a non-windowing SAS session.

SKIP= System Option

Specifies the number of lines to skip at the top of each page of SAS output for the LISTING destination.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Log and procedure output control: Procedure output

PROC OPTIONS
GROUP= LISTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SKIP= n | MIN | MAX | hexX

Syntax Description

n

specifies the range of lines to skip from 0 to 20.
MIN
sets the number of lines to skip to 0, so no lines are skipped.

MAX
sets the number of lines to skip to 20.

hex
specifies the number of lines to skip as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an X. For example, the value 0ax specifies to skip 10 lines.

Details
The location of the first line is relative to the position established by carriage control or by the forms control buffer on the printer. Most sites define this position so that the first line of a new page begins three or four lines down the form. If this spacing is sufficient, specify SKIP=0 so that additional lines are not skipped.

The SKIP= value does not affect the maximum number of lines printed on each page, which is controlled by the PAGESIZE= system option.

SOLUTIONS System Option
Specifies whether the Solutions menu is included in SAS windows.

Valid in: Configuration file, SAS invocation
Category: Environment control: Display
PROC OPTIONS GROUP= ENVDISPLAY

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax
SOLUTIONS | NOSOLUTIONS

Syntax Description
SOLUTIONS
specifies that the SOLUTIONS menu is included in SAS windows.

NOSOLUTIONS
specifies that the SOLUTIONS menu is not included in SAS windows.

SORTDUP= System Option
Specifies whether the SORT procedure removes duplicate variables based on all variables in a data set or the variables that remain after the DROP or KEEP data set options have been applied.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
Category: Sort: Procedure options
PROC OPTIONS
GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SORTDUP=PHYSICAL | LOGICAL

Syntax Description

PHYSICAL
removes duplicates based on all the variables that are present in the data set. This is the default.

LOGICAL
removes duplicates based on only the variables remaining after the DROP= and KEEP= data set options are processed.

Details

The SORTDUP= option specifies what variables to sort to remove duplicate observations when the SORT procedure NODUPRECS option is specified.

When SORTDUP= is set to LOGICAL and NODUPRECS is specified in the SORT procedure, duplicate observations are removed based on the variables that remain after a DROP or KEEP operation on the input data set. Setting SORTDUP=LOGICAL increases the number of duplicate observations that are removed because it eliminates variables before observations are compared. Setting SORTDUP=LOGICAL might improve performance.

When SORTDUP= is set to PHYSICAL and NODUPRECS is specified in the SORT procedure, duplicate observations are removed based on all of the variables in the input data set.

See Also

Procedures:

• Chapter 50, “SORT Procedure,” in Base SAS Procedures Guide

SORTEQUALS System Option

Specifies whether observations in the output data set with identical BY variable values are in a particular order.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System OPTIONS window

Category: Sort: Procedure options

PROC OPTIONS
GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

SORTEQUALS | NOSORTEQUALS

Syntax Description

SORTEQUALS
specifies that observations with identical BY variable values are to retain the same relative positions in the output data set as in the input data set.

NOSORTEQUALS
specifies that no resources be used to control the order of observations with identical BY variable values in the output data set.

Interaction: To achieve the best sorting performance when using the THREADS= system option, specify THREADS=YES and NOSORTEQUALS.

Tip: To save resources, use NOSORTEQUALS when you do not need to maintain a specific order of observations with identical BY variable values.

Comparisons

The SORTEQUALS and NOSORTEQUALS system options set the sorting behavior of PROC SORT for your SAS session. The EQUAL or NOEQUAL option in the PROC SORT statement overrides the setting of the system option for an individual PROC step and specifies the sorting behavior for that PROC step only.

See Also

Procedure Statement Options:

• PROC SORT statement EQUALS option, Chapter 50, “SORT Procedure” in Base SAS Procedures Guide

System Options:

• “THREADS System Option” on page 268

SORTSIZE= System Option

Specifies the amount of memory that is available to the SORT procedure.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Categories: Sort: Procedure options
System administration: Memory
System administration: Performance

PROC OPTIONS GROUP= MEMORY
PERFORMANCE
SORT

See: “SORTSIZE System Option: UNIX” in SAS Companion for UNIX Environments
“SORTSIZE System Option: Windows” in SAS Companion for Windows
SORTSIZE= System Option: z/OS

Syntax

SORTSIZE= n | nK | nM | nG | nT | hexX | MIN | MAX

Syntax Description

n | nK | nM | nG | nT
specifies the amount of memory in terms of 1 (byte); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 4000 specifies 4,000 bytes and a value of 2m specifies 2,097,152 bytes. If n=0, the sort utility uses its default. Valid values for SORTSIZE range from 0 to 9,223,372,036,854,775,807.

hexX
specifies the amount of memory as a hexadecimal number. This number must begin with a number (0-9), followed by an X. For example, 0fffx specifies 4095 bytes of memory.

MIN
specifies the minimum amount of memory available.

See: Values for MIN will vary, depending on your operating environment. For details, see the SAS documentation for your operating environment

MAX
specifies the maximum amount of memory available.

See: Values for MAX will vary, depending on your operating environment. For details, see the SAS documentation for your operating environment

Details

Generally, the value of the SORTSIZE= system option should be less than the physical memory available to your process. If the SORT procedure needs more memory than you specify, the system creates a temporary utility file.

PERFORMANCE NOTE: Proper specification of SORTSIZE= can improve sort performance by restricting the swapping of memory that is controlled by the operating environment.

See Also

Procedures:

• Chapter 50, “SORT Procedure” in Base SAS Procedures Guide

System Options:

• “SUMSIZE= System Option” on page 247

SORTVALIDATE System Option

Specifies whether the SORT procedure verifies if a data set is sorted according to the variables in the BY statement when a user-specified sort order is denoted in the sort indicator.
Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Sort: Procedure options

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SORTVALIDATE | NOSORTVALIDATE

Syntax Description

SORTVALIDATE
specifies that the SORT procedure verifies if the observations in the data set are sorted by the variables specified in the BY statement.

NOSORTVALIDATE
specifies that the SORT procedure is not to verify if the observations in the data set are sorted. This is the default.

Details

You can use the SORTVALIDATE system option to specify whether the SORT procedure validates that a data set is sorted correctly when the data set sort indicator shows a user-specified sort order. The user can specify a sort order by using the SORTEDBY= data set option in a DATA statement or by using the SORTEDBY= option in the DATASETS procedure MODIFY statement. When the sort indicator is set by a user, SAS cannot be absolutely certain that a data set is sorted according to the variables in the BY statement.

If the SORTVALIDATE system option is set and the data set sort indicator was set by a user, the SORT procedure performs a sequence check on each observation to ensure that the data set is sorted according to the variables in the BY statement. If the data set is not sorted correctly, SAS sorts the data set.

At the end of a successful sequence check or at the end of a sort, the SORT procedure sets the Validated sort information to Yes. If a sort is performed, the SORT procedure updates the Sortedby sort information to the variables that are specified in the BY statement.

If an output data set is specified, the Validated sort information in the output data set is set to Yes. If no sort is necessary, the data set is copied to the output data set.

See Also

• “Sorted Data Sets” in Chapter 25 of SAS Language Reference: Concepts

Data Set Options:

• “SORTEDBY= Data Set Option” in SAS Data Set Options: Reference

Procedures:

• Chapter 16, “DATASETS Procedure” in Base SAS Procedures Guide
SOURCE System Option

Specifies whether SAS writes source statements to the SAS log.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP=

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SOURCE | NOSOURCE

Syntax Description

SOURCE specifies to write SAS source statements to the SAS log.

NOSOURCE specifies not to write SAS source statements to the SAS log.

Details

The SOURCE system option does not affect whether statements from a file read with %INCLUDE or from an autocall macro are printed in the SAS log.

Note: SOURCE must be in effect when you execute SAS programs that you want to send to SAS for problem determination and resolution.

See Also


SOURCE2 System Option

Specifies whether SAS writes secondary source statements from included files to the SAS log.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
**Syntax**

SOURCE2 | NOSOURCE2

**Syntax Description**

SOURCE2

specifies to write to the SAS log secondary source statements from files that have been included by %INCLUDE statements.

NOSOURCE2

specifies not to write secondary source statements to the SAS log.

**Details**

*Note:* SOURCE2 must be in effect when you execute SAS programs that you want to send to SAS for problem determination and resolution.

**See Also**

“The SAS Log” in Chapter 9 of *SAS Language Reference: Concepts*

---

**SPOOL System Option**

Specifies whether SAS statements are written to a utility data set in the Work library.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation, OPTIONS statement, SAS System Options window</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Input control: Data processing</td>
</tr>
</tbody>
</table>

**Syntax**

SPOOL | NOSPOOL

**Syntax Description**

SPOOL

specifies that SAS write statements to a utility data set in the Work library for later use by a %INCLUDE or %LIST statement, or by the RECALL command, within a windowing environment.

NOSPOOL

specifies that SAS does not write statements to a utility data set. Specifying NOSPOOL accelerates execution time, but you cannot use the %INCLUDE and %LIST statements to resubmit SAS statements that were executed earlier in the session.

---

*Note:* This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Example
Specifying SPOOL is especially helpful in interactive line mode because you can resubmit a line or lines of code by referring to the line numbers. Here is an example of code including line numbers:

```
00001  data test;
00002      input w x y z;
00003      datalines;
00004      411.365 101.945 323.782 512.398
00005  ;
```

If SPOOL is in effect, you can resubmit line number 1 by submitting this statement:
```
%inc 1;
```

You can also resubmit a range of lines by placing a colon (:) or dash (-) between the line numbers. For example, these statements resubmit lines 1 through 3 and 4 through 5 of the above example:
```
%inc 1:3;
%inc 4-5;
```

---

STARTLIB System Option
Specifies whether SAS assigns user-defined permanent librefs when SAS starts.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Files: External files</td>
</tr>
<tr>
<td>PROC OPTIONS</td>
<td>EXTFILES</td>
</tr>
<tr>
<td>GROUP=</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
STARTLIB | NOSTARTLIB
```

**Syntax Description**

`STARTLIB`

specifies that when SAS starts, SAS assigns user-defined permanent librefs. STARTLIB is the default for the windowing environment.

`NOSTARTLIB`

specifies that SAS is not to assign user-defined permanent librefs when SAS starts. NOSTARTLIB is the default for batch mode, interactive line mode, and noninteractive mode.

**Details**

You assign a permanent libref only in the windowing environment by using the New Library window and by selecting the **Enable at startup** check box. SAS stores the permanent libref in the SAS registry. To open the New Library window, right-mouse
click Libraries in the Explorer window and select New. Alternatively, type DMLIBASSIGN in the command box.

In the windowing environment, SAS automatically assigns permanent librefs when SAS starts because STARTLIB is the default.

In all other execution modes (batch, interactive line, and noninteractive), SAS assigns permanent librefs only when you start SAS with the STARTLIB option specified either on the command line or in the configuration file.

---

**STEPCHKPT System Option**

Specifies whether checkpoint-restart data for DATA and PROC steps is to be recorded for a batch program.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Environment control: Error handling</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>ERRORHANDLING</td>
</tr>
<tr>
<td>Restriction:</td>
<td>The STEPCHKPT system option can be specified only if the LABELCHKPT system option is not specified when SAS starts.</td>
</tr>
<tr>
<td>Requirement:</td>
<td>This option can be used only in batch mode.</td>
</tr>
<tr>
<td>Note:</td>
<td>This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.</td>
</tr>
</tbody>
</table>

**Syntax**

STEPCHKPT | NOSTEPCHKPT

**Syntax Description**

**STEPCHKPT**

enables checkpoint mode, which specifies to record checkpoint-restart data for DATA and PROC steps.

**NOSTEPCHKPT**

disables checkpoint mode, which specifies not to record checkpoint-restart data. This is the default.

**Details**

Using the STEPCHKPT system option puts SAS in checkpoint mode for SAS programs that run in batch. Each time a DATA step or PROC step executes, SAS records data in a checkpoint-restart library. If a program terminates without completing, the program can be resubmitted. Execution begins with the step that was executing when the program terminated.

To ensure that the checkpoint-restart data is accurate, when you specify the STEPCHKPT option, also specify the ERRORCHECK STRICT option and set the ERRORABEND option so that SAS terminates for most errors.

Checkpoint mode is not valid for batch programs that contain the DM statement, which submits commands to SAS. If checkpoint mode is enabled and SAS encounters a DM statement, checkpoint mode is disabled and the checkpoint catalog entry is deleted.
Comparisons

The STEPCHKPT system option enables checkpoint mode for DATA and PROC steps in batch programs that terminate before completing. Execution resumes with the DATA or PROC step that was executing when the failure occurred.

The LABELCHKPT system option enables checkpoint mode for labeled code sections in batch programs that terminate before completing. Execution resumes at the labeled code section that was executing when the failure occurred.

See Also

• “Checkpoint Mode and Restart Mode” in Chapter 8 of SAS Language Reference: Concepts

Statements:

• “CHECKPOINT EXECUTE_ALWAYS Statement” in SAS Statements: Reference

System Options:

• “CHKPTCLEAN System Option” on page 89
• “ERRORABEND System Option” on page 130
• “ERRORCHECK= System Option” on page 131
• “LABELCHKPT System Option” on page 164
• “STEPCHKPTLIB= System Option” on page 244
• “STEPRESTART System Option” on page 246

STEPCHKPTLIB= System Option

Specifies the libref of the library where checkpoint-restart data for DATA and PROC steps is saved.

Valid in: Configuration file, SAS invocation
Category: Environment control: Error handling
PROC OPTIONS GROUP= ERRORHANDLING
Restriction: The STEPCHKPTLIB system option can be specified only if the LABELCHKPT system option is not specified when SAS starts.
Requirement: This option can be used only in batch mode.
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

STEPCHKPTLIB=libref
Syntax Description

libref
 specifies the libref that identifies the library where the checkpoint-restart data for DATA and PROC steps is saved.

Default: Work

Requirement: The LIBNAME statement that identifies the checkpoint-restart library must use the BASE engine and be the first statement in the batch program.

Details

When the STEPCHKPT system option is specified, checkpoint-restart data for batch programs is saved in the libref that is specified in the STEPCHKPTLIB= system option. If no libref is specified, SAS uses the Work library to save checkpoint data. The LIBNAME statement that defines the libref must be the first statement in the batch program.

If the Work library is used to save checkpoint data, the NOWORKTERM and NOWORKINIT system options must be specified so that the checkpoint-restart data is available when the batch program is resubmitted. These two options ensure that the Work library is saved when SAS ends and is restored when SAS starts. If the NOWORKTERM option is not specified, the Work library is deleted at the end of the SAS session and the checkpoint-restart data is lost. If the NOWORKINIT option is not specified, a new Work library is created when SAS starts, and again the checkpoint-restart data is lost.

The STEPCHKPTLIB= option must be specified for any SAS session that accesses checkpoint-restart data that is not saved to the Work library.

Operating Environment Information

When the Work library resides in a UNIX directory in UNIX or z/OS operating environments and you want to run the CLEANWORK utility, the Work library directory and its contents are deleted when the utility is run after the SAS session ends. When you run SAS in batch mode in the z/OS operating environment, the Work library is usually assigned to a temporary data set that is deleted at the end of the SAS job. To preserve the checkpoint-restart data in these cases, specify a permanent library as the value for the STEPCHKPTLIB option.

Comparisons

When the STEPCHKPT system option is set, the library specified by the STEPCHKPTLIB system option names the library where checkpoint-restart data is saved for DATA and PROC steps. When the STEPRESTART system option is set, the library specified by the STEPCHKPTLIB system option names the library where checkpoint-restart data is used to resume execution of DATA and PROC steps.

When the LABELCKPT system option is set, the library specified by the LABELCHKPTLIB system option names the library where checkpoint-restart data is saved for labeled code sections. When the LABELRESTART system option is set, the library specified by the LABELCHKPTLIB system option names the library where checkpoint-restart data is used to resume execution of labeled code sections.

See Also

• “Checkpoint Mode and Restart Mode” in Chapter 8 of SAS Language Reference: Concepts
STEPRESTART System Option

Specifies whether to execute a batch program by using checkpoint-restart data for DATA and PROC steps.

Valid in: Configuration file, SAS invocation
Category: Environment control: Error handling
PROC OPTIONS
GROUP= ERRORHANDLING

Restriction: The STEPRESTART system option can be specified only if the LABELCHKPT system option is not specified when SAS starts.
Requirement: This option can be used only in batch mode.
Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

STEPRESTART | NOSTEPRESTART

Syntax Description

STEPRESTART
enables restart mode, which specifies to execute the batch program by using the checkpoint-restart data for DATA and PROC steps.

NOSTEPRESTART
disables restart mode, which specifies not to execute the batch program using checkpoint-restart data.

Details

You specify the STEPRESTART option when you want to resubmit a batch program that ran in checkpoint mode and terminated before it completed. When you resubmit the batch program, SAS determines from the checkpoint data which DATA step or PROC step was executing when the program terminated, and resumes executing the batch program by using that DATA or PROC step.
Comparisons
When you specify the STEPRESTART option, SAS uses the checkpoint-restart data for
DATA and PROC steps to resume execution of batch programs.

When you specify the LABELRESTART option, SAS uses the checkpoint-restart data
for labeled code sections to resume execution of batch programs.

See Also
• “Checkpoint Mode and Restart Mode” in Chapter 8 of SAS Language Reference:
  Concepts

Statements:
• “CHECKPOINT EXECUTE_ALWAYS Statement” in SAS Statements: Reference

System Options:
• “CHKPTCLEAN System Option” on page 89
• “LABELCHKPT System Option” on page 164
• “LABELRESTART System Option” on page 167
• “STEPCHKPT System Option” on page 243
• “STEPCHKPTLIB= System Option” on page 244

SUMSIZE= System Option
Specifies a limit on the amount of memory that is available for data summarization procedures when class
variables are active.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options
  window

Category: System administration: Memory

PROC OPTIONS
GROUP= MEMORY

Note: This option can be restricted by a site administrator. For more information, see
  “Restricted Options” on page 6.

Syntax
SUMSIZE= n | nK | nM | nG | nT | hexX | MIN | MAX

Syntax Description
n | nK | nM | nG | nT
specifies the amount of memory in terms of 1 (bytes); 1,024 (kilobytes); 1,048,576
(megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). When
n=0, the default value, the amount of memory is determined by values of the
MEMSIZE option and the REALMEMSIZE option. Valid values for SUMSIZE
range from 0 to $2^{(n-1)}$ where $n$ is the data width in bits (32 or 64) of the operating
system.
hex\text{x}

specifies the amount of memory as a hexadecimal number. You must specify the value beginning with a number (0–9), followed by an \text{x}. For example, a value of \text{0fffx} specifies 4,095 bytes of memory.

\text{MIN}

specifies the minimum amount of memory available.

\text{MAX}

specifies the maximum amount of memory available.

\textbf{Details}

The \texttt{SUMSIZE=} system option affects the MEANS, OLAP, REPORT, SUMMARY, SURVEYFREQ, SURVEYLOGISTIC, SURVEYMEANS, and TABULATE procedures.

Proper specification of \texttt{SUMSIZE=} can improve procedure performance by restricting the swapping of memory that is controlled by the operating environment.

Generally, the value of the \texttt{SUMSIZE=} system option should be less than the physical memory available to your process. If the procedure that you are using needs more memory than you specify, the system creates a temporary utility file.

If the value of \texttt{SUMSIZE} is greater than the values of the \texttt{MEMSIZE} option and the \texttt{REALMEMSIZE} option, SAS uses the values of the \texttt{MEMSIZE} option and \texttt{REALMEMSIZE} option.

\textbf{See Also}

\textbf{System Options}:

\begin{itemize}
  \item “\texttt{MEMSIZE} System Option: UNIX” in \textit{SAS Companion for UNIX Environments}
  \item “\texttt{MEMSIZE} System Option: Windows” in \textit{SAS Companion for Windows}
  \item “\texttt{REALMEMSIZE} System Option: UNIX” in \textit{SAS Companion for UNIX Environments}
  \item “\texttt{REALMEMSIZE} System Option: Windows” in \textit{SAS Companion for Windows}
  \item “\texttt{REALMEMSIZE=} System Option: z/OS” in \textit{SAS Companion for z/OS}
  \item “\texttt{SORTSIZE=} System Option” on page 237
\end{itemize}

\textbf{SVGCONTROLBUTTONS}

Specifies whether to display the paging control buttons and an index in a multipage SVG document.

\begin{itemize}
  \item \textbf{Valid in:} Configuration file, SAS invocation, \texttt{OPTIONS} statement, SAS System Options window
  \item \textbf{Category:} Log and procedure output control: SVG
  \item \texttt{PROC OPTIONS GROUP=} SVG
  \item \textbf{Note:} This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
\end{itemize}
Syntax

**SVGCONTROLBUTTONS | NOSVGCONTROLBUTTONS**

**Syntax Description**

**SVGCONTROLBUTTONS**
- Specifies to display the paging control buttons in the SVG document.

**NOSVGCONTROLBUTTONS**
- Specifies not to display the paging control buttons in the SVG document. This is the default.

**Details**

When **SVGCONTROLBUTTONS** is specified, the size of the SVG is increased to accommodate the script that controls paging in the SVG document.

The **SVGView** printer sets the option to **SVGCONTROLBUTTONS**.

---

**SVGHEIGHT= System Option**

Specifies the height of the viewport unless the SVG output is embedded in another SVG output; specifies the value of the height attribute of the outermost `<svg>` element in the SVG file.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Log and procedure output control: SVG
- **PROC OPTIONS GROUP=** SVG
- **Restriction:** The **SVGHEIGHT=** option sets the height attribute only on the outermost `<svg>` element.
- **Note:** This option can be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

**Syntax**

**SVGHEIGHT= number-of-units<unit-of-measure> | "" | "**

**Syntax Description**

- **number-of-units**
  - Specifies the height as a number of **unit-of-measure**.
  - **Requirement:** **number-of-units** must be a positive integer value.
  - **Interaction:** If **number-of-units** is a negative number, the SVG document is not rendered by the browser.

- **unit-of-measure**
  - Specifies the unit of measurement, which can be one of the following:
    - % percentage
    - cm centimeters
em  the height of the element's font
ex  the height of the letter x
in  inches
mm  millimeters
pc  picas
pt  points
px  pixels

Default: px

"" | ''
specifies to reset the height to the default value of 600 pixels.

Requirement: Use two double quotation marks or two single quotation marks with no space between them.

Details

For embedded <svg> elements, the SVGHEIGHT= option specifies the height of the rectangular region into which the <svg> element is placed. The SVG output is scaled to fit the viewBox if SVGHEIGHT="100%".

If the SVGHEIGHT= option is not specified, the height attribute on the <svg> element is not set, which effectively provides full scalability by using a height of 100%.

The value for the SVGHEIGHT= option can be specified using no delimiters, enclosed in single or double quotation marks, or enclosed in parentheses.

Example

The following OPTIONS statement specifies to size the SVG output to portrait letter-sized and to scale the output to 100% of the viewport:

options printerpath=svg orientation=portrait svgheight="100%" svgwidth="100%"
papersize=letter;

By using these option values, SAS creates the following <svg> element:

<svg xmlns="http://www.w3.org/2000/svg"
     xmlns:xlink="http://www.w3.org/1999/xlink"
     xml:space="preserve"
     onload='Init(evt)' version="1.1"
     width="100%" height="100%"
     viewBox="-1 -1 817 1057"
</svg>

The value of "100%" in the SVGHEIGHT= option specifies to scale the SVG output height to 100% of the viewport, which is based on the value of the PAPERSIZE= option. The paper size is letter in the portrait orientation, which has a height of 11" at 100 dpi.

See Also

• “Using SAS System Options” on page 4
• Chapter 14, “The SAS Registry,” in SAS Language Reference: Concepts
• “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of SAS Language Reference: Concepts
SVGPRESERVEASPECTRATIO= System Option

Specifies whether to force uniform scaling of SVG output; specifies the preserveAspectRatio attribute on the outermost <svg> element.

**Valid in:**
Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
Log and procedure output control: SVG

**PROC OPTIONS GROUP=**
SVG

**Restriction:**
The SVGPRESERVEASPECTRATIO= option sets the preserveAspectRatio attribute only on the outermost <svg> element.

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
SVGPRESERVEASPECTRATIO=align | meetOrSlice | NONE | ""
SVGPRESERVEASPECTRATIO="align meetOrSlice"
```

**Syntax Description**

- **align**
  specifies to force uniform scaling by specifying the alignment method to use. The value for *align* can be one of the following:

  - **xMinYMin**
    specifies to force uniform scaling by using the following alignment:
    - Align the `<min–x>` of the element's viewBox with the smallest X value of the viewport.
    - Align the `<min–y>` of the element's viewBox with the smallest Y value of the viewport.

  - **xMidYMin**
    specifies to force uniform scaling by using the following alignment:
    - Align the midpoint X value of the element's viewBox with the midpoint X value of the viewport.
• Align the $<$min–y$>$ of the element’s viewBox with the smallest Y value of the viewport.

xMaxYMin
specifies to force uniform scaling by using the following alignment:
• Align the $<$min–x$>$+<width$>$ of the element's viewBox with the maximum X value of the viewport.
• Align the $<$min–y$>$ of the element's viewBox with the smallest Y value of the viewport.

xMinYMid
specifies to force uniform scaling by using the following alignment:
• Align the $<$min–x$>$ of the element's viewBox with the smallest X value of the viewport.
• Align the midpoint Y value of the element's viewBox with the midpoint Y value of the viewport.

xMidYMid
specifies to force uniform scaling by using the following alignment:
• Align the midpoint X value of the element's viewBox with the midpoint X value of the viewport.
• Align the midpoint Y value of the element's viewBox with the midpoint Y value of the viewport. This is the default.

xMaxYMid
specifies to force uniform scaling by using the following alignment:
• Align the $<$min–x$>$+<width$>$ of the element's viewBox with the maximum X value of the viewport.
• Align the midpoint Y value of the element's viewBox with the midpoint Y value of the viewport.

xMinYMax
specifies to force uniform scaling by using the following alignment:
• Align the $<$min–x$>$ of the element's viewBox with the smallest X value of the viewport.
• Align the $<$min–y$>$+<height$>$ of the element's viewBox with the maximum Y value of the viewport.

xMidYMax
specifies to force uniform scaling by using the following alignment:
• Align the midpoint X value of the element's viewBox with the midpoint X value of the viewport.
• Align the $<$min–y$>$+<height$>$ of the element's viewBox with the maximum Y value of the viewport.

xMaxYMax
specifies to force uniform scaling by using the following alignment:
• Align the $<$min–x$>$+<width$>$ of the element's viewBox with the maximum X value of the viewport.
• Align the $<$min–y$>$+<height$>$ of the element's viewBox with the maximum Y value of the viewport.
meetOrSlice

specifies to preserve the aspect ratio and how the viewBox displays. The following values are valid for meetOrSlice:

meet
specifies to scale the SVG graphic as follows:

• preserve the aspect ratio
• make the entire viewBox visible within the viewport
• scale up the viewBox as much as possible while meeting other criteria

If the aspect ratio of the graphic does not match the viewport, some of the viewport will extend beyond the bounds of the viewBox.

slice
specifies to scale the SVG graphic as follows:

• preserve the aspect ratio
• cover the entire viewBox with the viewport
• scale down the viewBox as much as possible while meeting other criteria

If the aspect ratio of the viewBox does not match the viewport, some of the viewBox will extend the bounds of the viewport.

NONE
specifies not to force uniform scaling and to scale the SVG output nonuniformly so that the element's bounding box exactly matches the viewport rectangle.

"

specifies to reset the preserveAspectRatio attribute of the <svg> element to the default value of xMidYMid meet.

Requirement: Use two double quotation marks with no space between them.

Details

When the value of the SVGPRESEVERASPECTRATIO= option includes both align and meetOrSlice, you can delimit the value by using single or double quotation marks or parentheses.

The preserveAspectRatio attribute applies only when a value is provided for the viewBox on the same <svg> element. If the viewBox attribute is not provided, the preserveAspectRatio attribute is ignored.

Example

The following OPTIONS statements are examples of using the SVGPRESEVERASPECTRATIO= system option:

options preserveaspectratio=xMinYMax;
options preserveaspectratio="xMinYMin meet";
options preserveaspectratio=(xMinYMin meet);
options preserveaspectratio="";

See Also

• “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of SAS Language Reference: Concepts
SVGTITLE= System Option

Specifies the title in the title bar of the SVG output; specifies the value of the <title> element in the SVG file.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: SVG

**PROC OPTIONS GROUP=** SVG

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

SVGTITLE="title" | "" | ""

**Syntax Description**

"title"

specifies the title of the SVG.

"" | ""

specifies to reset the title to empty.

**Requirement:** Use two double quotation marks or two single quotation marks with no space between them.

**Details**

If the SVGTITLE option is not specified, the title bar of the SVG output displays the filename of the SVG output.

The value for the SVGTITLE= option must be enclosed in single or double quotation marks, or enclosed in parentheses.

**See Also**

- “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of *SAS Language Reference: Concepts*
**SVGVIEWBOX= System Option**

Specifies the coordinates, width, and height that are used to set the viewBox attribute on the outermost `<svg>` element, which enables SVG output to scale to the viewport.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: SVG

**PROC OPTIONS GROUP=** SVG

**Restriction:** The SVGVIEWBOX= option sets the `viewBox` attribute only on the outermost `<svg>` element.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

`SVGVIEWBOX="min-x min-y width height"` | `none` | "" | ""

**Syntax Description**

`min-x`

specifies the beginning x coordinate of the viewBox, in user units.

**Requirement:** `min-x` can be 0, or a positive or a negative integer value.

`min-y`

specifies the beginning y coordinate of the viewBox, in user units.

**Requirement:** `min-y` can be 0, or a positive or negative integer value.

`width`

specifies the width of the viewBox, in user units.

**Requirement:** `width` must be a positive integer value.

`height`

specifies the height of the viewBox, in user units.

**Requirement:** `height` must be a positive integer value.

`none`

specifies that no `viewBox` attribute is to be specified on the outermost `<svg>` element, which will effectively create a static SVG document.
"" | "" specifies to reset the width and height of the viewBox to the width and height of the paper size for the SVG printer.

Requirement: Use two double quotation marks or two single quotation marks with no space between them.

Details

When the viewBox attribute is specified, the SVG output is scaled to be rendered in the viewport and the current coordinate system is updated to be the dimensions that are specified by the viewBox attribute. If it is not specified, the viewBox attribute on the outermost <svg> element sets the height and width arguments of the viewBox attribute to the paper height and paper width as defined by the PAPERSIZE= system option.

The coordinates, width, and height of the viewBox attribute should be mapped to the coordinates, width, and height of the viewport, taking into account the values of the preserveAspectRatio attribute.

The value for the SVGVIEWBOX= option must be enclosed in single or double quotation marks, or enclosed in parentheses.

You can use a negative value for min-x and min-y to place the SVG document in the output. A negative value of min-x shifts the output to the right. A negative value of min-y shifts the placement of the output downward.

Example

The following OPTIONS statement specifies to scale the output to a width of 100 user units and a height of 200 user units:

options printerpath=svg svgviewbox="0 0 100 200" dev=sasprtc;

By using these option values, SAS creates the following <svg> element:

<svg> xmlns="http://www.w3.org/2000/svg"
   xmlns:xlink="http://www.w3.org/1999/xlink"
   xml:space="preserve"
   onload='Init(evt)' version="1.1"
   viewBox="0 0 100 200"
</svg>

See Also

• “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of SAS Language Reference: Concepts

System Options:

• “SVGCONTROLBUTTONS” on page 248
• “SVGHEIGHT= System Option” on page 249
• “SVGPRESAVEASPECTRATIO= System Option” on page 251
• “SVGTITLE= System Option” on page 254
• “SVGWIDTHT= System Option” on page 257
• “SVGX= System Option” on page 259
• “SVGY= System Option” on page 260
SVGWIDTH= System Option

Specifies the width of the viewport unless the SVG output is embedded in another SVG output; specifies the value of the width attribute in the outermost <svg> element in the SVG file.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: SVG

PROC OPTIONS GROUP=

Restriction: The SVGWIDTH= option sets the width attribute only on the outermost <svg> element.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SVGWIDTH= number-of-units<unit-of-measure> | "" | ""

Syntax Description

number-of-units

specifies the width as a number of unit-of-measure.

Requirement: number-of-units must be a positive integer value.

Interaction: If number-of-units is a negative number, the SVG document is not rendered by the browser.

unit-of-measure

specifies the unit of measurement, which can be one of the following:

% percentage
cm centimeters
dm the height of the element's font
ex the height of the letter x
in inches
mm millimeters
pc picas
pt points
px pixels

Default: px

"" "" specifies to reset the width to the default value of 800 pixels.

Requirement: Use two double quotation marks or two single quotation marks with no space between them.
Details

For embedded `<svg>` elements, the SVGWIDTH= option specifies the width of the rectangular region into which the `<svg>` element is placed. The SVG output is scaled to fit the viewBox if SVGWIDTH="100%".

If the SVGWIDTH= option is not specified, the width attribute on the `<svg>` element is not set, which effectively provides full scalability by using a width of 100%.

The value for the SVGWIDTH= option can be specified without delimiters, enclosed in single or double quotation marks, or enclosed in parentheses.

Example

The following OPTIONS statement specifies to size the SVG output to portrait letter-sized and to scale the output to 100% of the viewport:

```plaintext
options printerpath=svg orientation=portrait svgheight="100%" svgwidth="100%" papersize=letter;
```

By using these option values, SAS creates the following `<svg>` element:

```plaintext
<svg xmlns="http://www.w3.org/2000/svg"
     xmlns:xlink="http://www.w3.org/1999/xlink"
     xml:space="preserve"
     onload='Init(evt)'
     version="1.1"
     width="100%" height="100%"
     viewBox="-1 -1 817 1057"
     
</svg>
```

The value of "100%" in the SVGWIDTH= option specifies to scale the SVG output width to 100% of the viewport, which is based on the value of the PAPERSIZE= option. The paper size is letter in the portrait orientation, which has a width of 8.5" at 96 dpi.

See Also

- “Using SAS System Options” on page 4

System Options:

- “SVGCONTROLBUTTONS” on page 248
- “SVGHEIGHT= System Option” on page 249
- “SVGPRESERVEASPECTRATIO= System Option” on page 251
- “SVGTITLE= System Option” on page 254
- “SVGVIEWBOX= System Option” on page 255
- “SVGX= System Option” on page 259
- “SVGY= System Option” on page 260
**SVGX= System Option**

Specifies the x-axis coordinate of one corner of the rectangular region into which an embedded `<svg>` element is placed; specifies the x attribute in the outermost `<svg>` element in an SVG file.

### Valid in:
- Configuration file
- SAS invocation
- OPTIONS statement
- SAS System Options window

### Category:
Log and procedure output control: SVG

### PROC OPTIONS
- **GROUP=** SVG

### Restriction:
The SVGX= option sets the x attribute only on the outermost `<svg>` element.

### Note:
This option can be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

---

**Syntax**

```
SVGX= number-of-units<unit-of-measure> | "" | ''
```

**Syntax Description**

- **number-of-units**
  - specifies the x-axis coordinate as a number of *unit-of-measure*.

- **unit-of-measure**
  - specifies the unit of measurement, which can be one of the following:
    - `%` percentage
    - `cm` centimeters
    - `em` the height of the element's font
    - `ex` the height of the letter x
    - `in` inches
    - `mm` millimeters
    - `pc` picas
    - `pt` points
    - `px` pixels

  **Default:** px

- `"" | ''`
  - specifies to reset the x attribute to 0 on the `<svg>` element and the x-axis coordinate for embedded SVG to 0.

  **Requirement:** Use two double quotation marks or two single quotation marks with no space between them.

---

**Details**

If the SVGX= option is not set, the x attribute on the `<svg>` element effectively has a value of 0 and no x-axis coordinate is set for embedded SVG output.
The value for the SVGX= option can be specified without delimiters, enclosed in single or double quotation marks, or enclosed in parentheses.

The \texttt{x} attribute on the outermost \texttt{<svg>} element has no effect on SVG documents that are produced by SAS. You can use the SVGX= system option to specify the x-axis coordinate if the SVG document is processed outside of SAS.

**See Also**

- “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of *SAS Language Reference: Concepts*

**System Options:**

- “SVGCONTROLBUTTONS” on page 248
- “SVGHHEIGHT= System Option” on page 249
- “SVGPRESERVEASPECTRATIO= System Option” on page 251
- “SVGTITLE= System Option” on page 254
- “SVGWWIDTH= System Option” on page 257
- “SVGVVIEWBOX= System Option” on page 255
- “SVGY= System Option” on page 260

---

**SVGY= System Option**

Specifies the y-axis coordinate of one corner of the rectangular region into which an embedded \texttt{<svg>} element is placed; specifies the y attribute in the outermost \texttt{<svg>} element in an SVG file.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: SVG

**PROC OPTIONS GROUP=** SVG

**Restriction:** The SVGY= option sets the y attribute only on the outermost \texttt{<svg>} element.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

\[
\texttt{SVGY= number-of-units<unit-of-measure> | "" | "} \]

**Syntax Description**

- \texttt{number-of-units}
  - specifies the y-axis coordinate as a number of \texttt{unit-of-measure}.

- \texttt{unit-of-measure}
  - specifies the unit of measurement, which can be one of the following:
    - \texttt{%} percentage
cm    centimeters
em    the height of the element's font
ex    the height of the letter x
in    inches
mm    millimeters
pc    picas
pt    points
px    pixels

**Default:** px

"" | "

specifies to reset the y attribute on the `<svg>` element and the y-axis coordinate for embedded SVG output to 0.

**Requirement:** Use two double quotation marks or two single quotation marks with no space between them.

**Details**

If the SVGY= option is not set, the y attribute on the `<svg>` element effectively has a value of 0 and no y-axis coordinate is set for embedded SVG output.

The value for the SVGY= option can be specified without delimiters, enclosed in single or double quotation marks, or enclosed in parentheses.

The y attribute on the outermost `<svg>` element has no effect on SVG documents that are produced by SAS. You can use the SVGY= system option to specify the y-axis coordinate if the SVG document is processed outside of SAS.

**See Also**

- “Creating SVG (Scalable Vector Graphics) Files Using Universal Printing” in Chapter 15 of *SAS Language Reference: Concepts*

**System Options:**

- “SVGCONTROLBUTTONS” on page 248
- “SVGHEIGHT= System Option” on page 249
- “SVGPRESERVEASPECTRATIO= System Option” on page 251
- “SVGTITLE= System Option” on page 254
- “SVGWIDTH= System Option” on page 257
- “SVGVIEWBOX= System Option” on page 255
- “SVGX= System Option” on page 259

**SYNTAXCHECK System Option**

In non-interactive or batch SAS sessions, specifies whether to enable syntax check mode for multiple steps.
Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

SYNTAXCHECK | NOSYNTAXCHECK

Syntax Description

SYNTAXCHECK

enables syntax check mode for statements that are submitted within a non-interactive or batch SAS session.

NOSYNTAXCHECK

does not enable syntax check mode for statements that are submitted within a non-interactive or batch SAS session.

CAUTION: Setting NOSYNTAXCHECK might cause a loss of data. Manipulating and deleting data by using untested code might result in a loss of data if your code contains invalid syntax. Be sure to test code completely before placing it in a production environment.

Details

If a syntax or semantic error occurs in a DATA step after the SYNTAXCHECK option is set, then SAS enters syntax check mode, which remains in effect from the point where SAS encountered the error to the end of the code that was submitted. After SAS enters syntax mode, all subsequent DATA step statements and PROC step statements are validated.

While in syntax check mode, only limited processing is performed. For a detailed explanation of syntax check mode, see “Syntax Check Mode” in Chapter 8 of SAS Language Reference: Concepts.

Place the OPTIONS statement that enables SYNTAXCHECK before the step for which you want it to take effect. If you place the OPTIONS statement inside a step, then SYNTAXCHECK will not take effect until the beginning of the next step.

NOSYNTAXCHECK enables continuous processing of statements regardless of syntax error conditions.

SYNTAXCHECK is ignored in the SAS windowing environment and in SAS line-mode sessions.

Comparisons

You use the SYNTAXCHECK system option to validate syntax in a non-interactive or a batch SAS session. You use the DMSSYNCHK system option to validate syntax in an interactive session by using the SAS windowing environment.

The ERRORCHECK= option can be set to enable or disable syntax check mode for the LIBNAME statement, the FILENAME statement, the %INCLUDE statement, and the
LOCK statement in SAS/SHARE. If you specify the NOSYNTAXCHECK option and the ERRORCHECK=STRICT option, then SAS does not enter syntax check mode when an error occurs.

See Also

- “Error Processing in SAS” in Chapter 8 of *SAS Language Reference: Concepts*

System Options:

- “DMSSYNCHK System Option” on page 117
- “ERRORCHECK= System Option” on page 131

---

**SYSPRINTFONT= System Option**

Specifies the default font to use for printing, which can be overridden by explicitly specifying a font and an ODS style.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Log and procedure output control: Procedure output

**PROC OPTIONS GROUP=** LISTCONTROL

**Note:** This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**See:** “SYSPRINTFONT System Option: Windows” in *SAS Companion for Windows*

**Syntax**

```
SYSPRINTFONT=("face-name" <weight> <style> <character-set> <point-size>
<NAMED "printer-name " | UPRINT="printer-name " | DEFAULT | ALL>)
```

**Syntax Description**

*face-name*

specifies the name of the font face to use for printing.

**Requirements:**

If *face-name* consists of more than one word, you must be enclose the value in single or double quotation marks. The quotation marks are stored with the face-name.

When you use the SYSPRINTFONT= option with multiple arguments, you must enclose the arguments in parentheses.

**Interaction:** When you specify UPRINT=printer-name, face-name must be a valid font for printer-name.

*weight*

specifies the weight of the font, such as BOLD. A list of valid values for your specified printer appears in the SAS: Printer Properties window.

**Default:** NORMAL
style
specifies the style of the font, such as ITALIC. A list of valid values for your specified printer appears in the SAS: Printer Properties window.

Default: REGULAR

character-set
specifies the character set to use for printing.

Default: If the font does not support the specified character set, the default character set is used. If the default character set is not supported by the font, the font's default character set is used.

Range: Valid values are listed in the SAS: Printer Properties window, under the Font tab.

point-size
specifies the point size to use for printing. If you omit this argument, SAS uses the default.

Requirement: Point-size must be an integer. It must also be placed after the face-name, weight, style, and character-set arguments.

NAMED “printer-name”
specifies a printer in the Windows operating environment to which these settings apply.

Restriction: This argument is valid only for printers in the Windows operating environment. To specify a Universal Printer, use the UPRINT=argument.

Requirements:
The printer-name must exactly match the name shown in the Print Setup dialog box (except that the printer name is not case sensitive).

If the printer is more than one word, the printer-name must be enclosed in double quotation marks. The quotation marks are stored with the printer-name.

UPRINT=“printer-name”
specifies a Universal Printer to which these settings apply.

Restriction: This argument is valid only for printers that are listed in the SAS Registry.

Requirements:
The printer-name must match exactly the name shown in the Print Setup dialog box (except that the printer name is not case sensitive).

If the printer-name is more than one word, it must be enclosed in single or double quotation marks. The quotation marks are stored with the printer-name.

DEFAULT | ALL
specifies whether the font settings apply to the default printer or to all printers:

DEFAULT
specifies that the font settings apply to the current default printer that is specified by the SYSPRINT= system option.

ALL
specifies that the font settings apply to all installed printers.

Details
The SYSPRINTFONT= system option sets the font to use when printing to the current default printer, to a specified printer or to all printers.

In some cases, you might need to specify the font from a SAS program. In this case, you might want to view the SAS: Printer Properties window for allowable names, styles
weights, and sizes for your fonts. For examples of how to apply the SYSPRINTFONT= option in a SAS program, see “Comparisons” on page 265.

If you specified SYSPRINTFONT= with DEFAULT or without a keyword and later use the Print Setup dialog box to change the current default printer, then the font used with the current default printer will be the font that was specified with SYSPRINTFONT, if the specified font exists on the printer. If the current printer does not support the specified font, the printer’s default font is used.

The following fonts are widely supported:

- Helvetica
- Times
- Courier
- Symbol

By specifying one of these fonts in a SAS program, you can usually avoid returning an error. If that particular font is not supported, a similar-looking font prints in its place.

All Universal printers and many SAS/GRAPH devices use the FreeType engine to render TrueType fonts. For more information, see “Using Fonts with Universal Printers and SAS/GRAPH Devices” in Chapter 15 of SAS Language Reference: Concepts.

Note: As an alternative to using the SYSPRINTFONT= system option, you can set fonts with the SAS: Printer Properties window, under the Font tab. From the drop-down menu select File ⇒ Print Setup ⇒ Properties ⇒ Font. Using a dialog box is fast and easy because you choose your font, style, weight, size, and character set from a list of options that your selected printer supports.

**Comparisons**

**Specifying a Font to the Default Printer**

This example specifies the 12–point Times font on the default printer:

```sas
options sysprintfont=("times" 12);
```

**Specifying a Font to a Named Windows Printer**

This example specifies to use Courier on the printer named HP LaserJet IIIsi Postscript. Specify the printer name in the same way that it is specified in the SAS Print Setup dialog box:

```sas
options sysprintfont= ("courier" named "hp laserjet 111s, postscript");
```

**Specifying a Font to a Universal Printer, on the SAS command line**

This example specifies the Albany AMT font for the PDF Universal Printer:

```sas
sysprintfont=('courier' 11 uprint='PDF')
```

---

**TERMINAL System Option**

Specifies whether to associate a terminal with a SAS session.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Environment control: Initialization and operation
- **PROC OPTIONS GROUP=** EXECMODES
Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

TERMINAL | NOTERMINAL

**Syntax Description**

**TERMINAL**

specifies that SAS evaluate the execution environment and if a physical display is not available for an interactive environment, sets the option to NOTERMINAL. Specify TERMINAL when you use the SAS windowing environment.

**NOTERMINAL**

specifies that SAS not evaluate the execution environment.

**Details**

SAS defaults to the appropriate setting for the TERMINAL system option based on whether the session is invoked in the foreground or the background. If NOTERMINAL is specified, dialog boxes are not displayed.

The TERMINAL option is normally used with the following execution modes:

- SAS windowing environment mode
- interactive line mode
- noninteractive mode.

---

**TERMSTMT= System Option**

Specifies the SAS statements to execute when SAS terminates.

**Valid in:** Configuration file, SAS invocation

**Category:** Environment control: Initialization and operation

**PROC OPTIONS GROUP=** EXECMODES

**Operating environment:** In some operating system environments there is a limit to the size of the value for TERMSTMT=. To circumvent this limitation, you can use the %INCLUDE statement.

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

TERMSTMT='statement(s)'

**Syntax Description**

`statement(s)`

is one or more SAS statements.

**Length:** maximum length is 2,048 characters
Details

TERMSTMT= is fully supported in batch mode. In interactive modes, TERMSTMT= is executed only when you submit the ENDSAS statement from an editor window to terminate the SAS session. Terminating SAS by any other means in interactive mode results in TERMSTMT= not being executed.

An alternate method for specifying TERMSTMT= is to put a %INCLUDE statement at the end of a batch file or to submit a %INCLUDE statement before terminating the SAS session in interactive mode.

Comparisons

TERMSTMT= specifies the SAS statements to be executed at SAS termination, and INITSTMT= specifies the SAS statements to be executed at SAS initialization.

See Also

Statements:

- “%INCLUDE Statement” in SAS Statements: Reference

System Options:

- “INITSTMT= System Option” on page 158

TEXTURELOC= System Option

Specifies the location of textures and images that are used by ODS styles.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP= ODSPRINT

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

TEXTURELOC=location

Syntax Description

location

specifies the location of textures and images used by ODS styles. Location can refer either to the physical name of the directory or to a URL reference to the directory.

Restriction: Only one location is allowed per statement.

Requirements:

If location is not a fileref, then you must enclose the value in quotation marks. The files in the directory must be in the form of gif, jpeg, or bitmap. Location must refer to a directory.
THREADS System Option

Specifies that SAS uses threaded processing if available.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- PROC OPTIONS GROUP= PERFORMANCE

**Category:**
System administration: Performance

**Note:**
This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

THREADS | NOTHREADS

**Syntax Description**

**THREADS**

specifies to use threaded processing for SAS applications that support it.

**Interaction:**
If THREADS is specified either as a SAS system option or in PROC SORT and another program has the input SAS data set open for reading, writing, or updating using the SPD engine, then the procedure might fail and write a subsequent message to the SAS log.

**NOTHREADS**

specifies not to use threaded processing for running SAS applications that support it.

**Interaction:**
When you specify NOTHREADS, CPUCOUNT= is ignored unless you specify a procedure option that overrides the NOTHREADS system option.

**Details**

The THREADS system option enables some legacy SAS processes that are thread-enabled to take advantage of multiple CPUs by threading the processing and I/O operations. Threading the processing and I/O operations achieves a degree of parallelism that generally reduces the real time to completion for a given operation at the possible cost of additional CPU resources. In SAS 9 and SAS 9.1, the thread-enabled processes include

- Base SAS engine indexing
- Base SAS procedures: SORT, SUMMARY, MEANS, REPORT, TABULATE, and SQL
- SAS/STAT procedures: GLM, LOESS, REG, ROBUSTREG.

For example, in some cases, processing small data sets, SAS might determine to use a single-threaded operation.
Set this option to NOTHREADS to achieve SAS behavior most compatible with releases before to SAS 9, if you find that threading does not improve performance or if threading might be related to an unexplainable problem. See the specific documentation for each product to determine whether it has functionality that is enabled by the THREADS option.

**Comparisons**

The system option THREADS determines when threaded processing is in effect. The SAS system option CPUCOUNT= suggests how many system CPUs are available for use by thread-enabled SAS procedures.

**See Also**

- Chapter 13, “Support for Parallel Processing,” in *SAS Language Reference: Concepts*

**System Options:**

- “CPUCOUNT= System Option” on page 101
- “UTILLOC= System Option” on page 274

---

**TOOLSMENU System Option**

Specifies whether the Tools menu is included in SAS windows.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Environment control: Display
- **PROC OPTIONS GROUP=** ENVDISPLAY
- **Default:** TOOLSMENU
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

```
TOOLSMENU | NOTOOLSMENU
```

**Syntax Description**

- **TOOLSMENU** specifies that the Tools menu is included in SAS windows.
- **NOTOOLSMENU** specifies that the Tools menu is not included in SAS windows.

---

**TOPMARGIN= System Option**

Specifies the print margin at the top of the page.
Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP=ODSPRINT

Note: This option cannot be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

TOPMARGIN= margin-size<margin-unit>

Syntax Description

margin-size

specifies the size of the margin.

Restriction: The bottom margin should be small enough so that the top margin plus the bottom margin is less than the height of the paper.

Interaction: Changing the value of this option might result in changes to the value of the PAGESIZE= system option.

<margin-unit>

specifies the units for margin-size. The margin-unit can be in for inches or cm for centimeters. <margin-unit> is saved as part of the value of the TOPMARGIN system option.

Default: inches

Details

All margins have a minimum that is dependent on the printer and the paper size. The default value of the TOPMARGIN system option is 0.00 in.

See Also

• “Universal Printing” in Chapter 15 of SAS Language Reference: Concepts

Statements:


System Options:

• “BOTTOMMARGIN= System Option” on page 76
• “LEFTMARGIN= System Option” on page 169
• “RIGHTMARGIN= System Option” on page 222

TRAINLOC= System Option

Specifies the URL for SAS online training courses.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files

PROC OPTIONS
GROUP=ENVFILES

Note: This option can be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

Syntax

TRAINLOC="base-URL"

Syntax Description

base-URL specifies the address where the SAS online training courses are located.

Details

The TRAINLOC= system option specifies the base location (typically a URL) of SAS online training courses. These online training courses are typically accessed from an intranet server or a local CD-ROM.

Example

Some examples of the base-URL are:

- "file://e:/onlintut"
- "http://server.abc.com/SAS/sastrain"

UPRINTCOMPRESSION System Option

Specifies whether to enable the compression of files created by some Universal Printer and SAS/GRAPH devices.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Log and procedure output control: ODS Printing

PROC OPTIONS
GROUP=ODSPRINT

Alias: UPC | NOUPC

Note: This option can be restricted by a site administrator. For more information, see "Restricted Options" on page 6.

Syntax

UPRINTCOMPRESSION | NOUPRINTCOMPRESSION
Syntax Description

**UPRINTCOMPRESSION**

specifies to enable compression of files created by some Universal Printers and some SAS/GRAPH devices. This is the default.

**NOUPRINTCOMPRESSION**

specifies to disable compression of files created by some Universal Printers and some SAS/GRAPH devices.

Details

The following table lists the Universal Printers and the SAS/GRAPH devices that are affected by the UPRINTCOMPRESSION system option:

<table>
<thead>
<tr>
<th>Universal Printers</th>
<th>SAS/GRAPH Device Drivers</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCL5, PCL5C, PCL5E</td>
<td>PCL5, PCL5C, PCL5E</td>
</tr>
<tr>
<td>PDF</td>
<td>PDF, PDFA, PDFC</td>
</tr>
<tr>
<td>SVG</td>
<td>SVG</td>
</tr>
<tr>
<td>PS</td>
<td>SASPTC, SASPTG, SASPTM</td>
</tr>
<tr>
<td></td>
<td>UEPS, UPSC, UPCL5, UPCL5C, UPCL5E, UPDF, UPSL, UPSLC</td>
</tr>
</tbody>
</table>

When NOUPRINTCOMPRESSION is set, the DEFLATION= option is ignored.

The ODS PRINTER statement option, COMPRESS=, takes precedence over the UPRINTCOMPRESSION system option.

See Also

Statements:


System Options:

- “DEFLATION= System Option” on page 106

**URLENCODING= System Option**

Specifies whether the argument to the URLENCODE function and to the URLDECODE function is interpreted using the SAS session encoding or UTF-8 encoding.

**Valid in:**
- Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:**
- Environment control: Language control

**PROC OPTIONS**

**GROUP=***

**LANGUAGECONTROL**
Syntax

**URLENCODING=** SESSION | UTF8

**Syntax Description**

**SESSION**

specifies that the argument to the URLENCODE function and to the URLDECODE function is interpreted using the SAS session encoding. This is the default.

*Note:* SAS session encoding uses the URL encoding standard RFC 1738.

*Tip:* SESSION is compatible with previous releases of SAS.

**UTF8**

specifies that the argument to the URLENCODE function and to the URLDECODE function is interpreted using UTF-8 encoding.

*Note:* UTF-8 encoding uses the URL encoding standard RFC 3986.

**See Also**

**Functions:**

- “URLDECODE Function” in *SAS Functions and CALL Routines: Reference*
- “URLENCODE Function” in *SAS Functions and CALL Routines: Reference*
**Syntax Description**

library-specification

specifies the libref or physical name of a SAS library.

**Details**

If this option is specified, you can use one-level names to reference permanent SAS files in SAS statements. However, if USER=WORK is specified, SAS assumes that files referenced with one-level names refer to temporary work files.

---

**UTILLOC= System Option**

Specifies one or more file system locations in which enabled threaded applications can store utility files.

| Valid in: | configuration file and SAS invocation |
| Category: | Files: SAS Files |
| PROC OPTIONS GROUP= | SASFILES |

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**See:** "UTILLOC= System Option: z/OS" in SAS Companion for z/OS

---

**Syntax**

UTILLOC= WORK | location | (location-1...location-n)

**Syntax Description**

**WORK**

specifies that SAS creates utility files in the same directory as the Work library.

This is the default.

**location**

specifies the location of an existing directory for utility files that are created by applications. Enclose location in single or double quotation marks when the location contains spaces.

**z/OS specifics:** On z/OS each location is a list of DCB and SMS options to be used when creating utility files.

**(location-1 ... location-n)**

specifies a list of existing directories that can be accessed in parallel for utility files that are created by applications. A single utility file cannot span locations. Enclose a location in single or double quotation marks when the location contains spaces. Any location that does not exist is deleted from the value of the UTILLOC= system option.

**Requirement:** If you have more than one location, then you must enclose the list of locations in parentheses.

**z/OS specifics:** On z/OS, each location is a list of DCB and SMS options to be used when creating utility files.
Details

The UTILLOC option specifies one or more locations for a type of utility file that is introduced as part of the SAS 9 architecture. These utility files are comparable to SAS files with a type of UTILITY, but they are not members of the Work library or any other SAS library. UTILLOC utility files are primarily used by applications that are enabled for multiple threads of execution.

Each location that is specified for the UTILLOC option identifies a single place at which utility files can be created. If multiple locations are specified, then the locations are used on a rotating basis by SAS applications as utility files are required.

For applications that use multiple utility files at the same time, specifying multiple locations that correspond to separate physical I/O devices might improve performance by reducing competition for device resources.

For the SORT procedure, the UTILLOC= system option affects the placement of the utility files only if the multi-threaded SAS sort is used. The multi-threaded SAS sort can be invoked when the THREAD system option is specified and the value of the CPUCOUNT= system option is greater than 1. The multi-threaded SAS sort can also be invoked when you specify the THREADS option in the PROC SORT statement. The multi-threaded sort stores all temporary data in a single utility file within one of the locations that are specified by the UTILLOC= system option. The size of this utility file is proportional to the amount of data that is read from the input data set. A second utility file of the same size can be created in another of these locations when the amount of data that is read from the input data set is large or the amount of memory that is available to the SORT procedure is small.

See Also


Procedures:

- Chapter 50, “SORT Procedure,” in Base SAS Procedures Guide

System Options:

- “CPUCOUNT= System Option” on page 101
- “THREADS System Option” on page 268

UUIDCOUNT= System Option

Specifies the number of UUIDs to acquire from the UUID Generator Daemon.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

UUIDCOUNT= n | MIN | MAX

Syntax Description

n
specifies the number of UUIDs to acquire. Zero indicates that the UUID Generator Daemon is not required.

Default: 100
Range: 0–1000

MIN | MAX

MIN
specifies that the number of UUIDs to acquire is zero, indicating that the UUID Generator Daemon is not required.

MAX
specifies that 1000 UUIDs at a time should be acquired from the UUID Generator Daemon.

Details

If a SAS application will generate a large number of UUIDs, this value can be adjusted at any time during a SAS session to reduce the number of times that the SAS session would have to contact the SAS UUID Generator Daemon.

See Also

• “Universal Unique Identifiers and the Object Spawner” in Chapter 39 of SAS Language Reference: Concepts

Functions:

• “UUIDGEN Function” in SAS Functions and CALL Routines: Reference

System Option;

• “UUIDGENDHOST= System Option” on page 276

UUIDGENDHOST= System Option

Identifies the host and port or the LDAP URL that the UUID Generator Daemon runs on.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files
PROC OPTIONS GROUP=

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

**UUIDGENDHOST= 'host-string'**

**Syntax Description**

'host-string'

is either of the form host-name:port or an LDAP URL. The value must be in one string. Enclose an LDAP URL string with quotation marks.

**Details**

SAS does not guarantee that all UUIDs are unique. Use the SAS UUID Generator Daemon (UUIDGEN) to ensure unique UUIDs.

**Example**

- Specifying host-name:port as the 'host-string':
  ```
sas -UUIDGENDHOST 'myhost.com:5306'
  
or
  sas UUIDGENDHOST= 'myhost.com:5306'
  ```

- Specifying an LDAP URL as the 'host-string':
  ```
  "ldap://ldap-host-name/sasspawner-distinguished-name"
  ```

- A more detailed example of an LDAP URL as the 'host-string':
  ```
  "ldap://ldaphost/sasSpawnercn=UUIDGEND,sascomponent=sasServer, cn=ABC,o=ABC Inc,c=US"
  ```

- Specifying your binddn and password, if your LDAP server is secure:
  ```
  "ldap://ldap-host-name/sasspawner-distinguished-name bindname=bindname,password=bind-password"
  ```

- An example with a bindname value and a password value:
  ```
  "ldap://ldaphost/
  sasSpawnercn=UUIDGEND,sascomponent=sasServer,cn=ABC,o=ABC Inc,c=US
  bindname=cn=me%2co=ABC Inc %2cc=US, password=itsme"
  ```

  **Note:** When specifying your bindname and password, commas that are a part of your bindname and your password must be replaced with the string "%2c". In the previous example, the bindname is as follows:

  ```
  cn=me,o=ABC Inc, c=US
  ```

**See Also**

**Functions:**

- “UUIDGEN Function” in *SAS Functions and CALL Routines: Reference*

**System Options:**

- “UUIDCOUNT= System Option” on page 275
**V6CREATEUPDATE= System Option**

Specifies the type of message to write to the SAS log when Version 6 data sets are created or updated.

- **Valid in:** Configuration file, SAS invocation
- **Category:** Files: SAS Files
- **PROC OPTIONS GROUP=** SASFILES
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

**Syntax**

\[
\text{V6CREATEUPDATE = ERROR | NOTE | WARNING | IGNORE}
\]

**Syntax Description**

- **ERROR**
  - Specifies that an ERROR is written to the SAS log when the V6 engine is used to open a SAS data set for creation or update. The attempt to create or update a SAS data set in Version 6 format will fail. Reading Version 6 data sets will not generate an error.

- **NOTE**
  - Specifies that a NOTE is written to the SAS log when the V6 engine is used; all other processing occurs normally.

- **WARNING**
  - Specifies that a WARNING is written to the SAS log when the V6 engine is used; all other processing occurs normally.

- **IGNORE**
  - Disables the V6CREATEUPDATE= system option. Nothing is written to the SAS log when the V6 engine is used.

**VALIDFMTNAME= System Option**

Specifies the maximum size (32 characters or 8 characters) that user-created format and informat names can be before an error or warning is issued.

- **Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window
- **Category:** Files: SAS Files
- **PROC OPTIONS GROUP=** SASFILES
- **Default:** LONG
- **Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.
Syntax

**VALIDFMTNAME=** LONG | FAIL | WARN

**Syntax Description**

**LONG**

specifies that format and informat names can be up to 32 alphanumeric characters. This is the default.

**FAIL**

specifies that creating a format or informat name that is longer than eight characters results in an error message.

**Interaction:** If you explicitly specify the V7 or V8 Base SAS engine, such as in a LIBNAME statement, then SAS automatically uses the VALIDFMTNAME=FAIL behavior for data sets that are associated with those engines.

**Tip:** Specify this setting for using formats and informats that are valid in both SAS 9 and previous releases of SAS.

**WARN**

specifies that creating a format or informat name that is longer than eight characters results in a warning message to remind you that the format or informat cannot be used with releases before to SAS 9.

**Details**

SAS 9 enables you to define format and informat names up to 32 characters. Previous releases were limited to eight characters. The VALIDFMTNAME= system option applies to format and informat names in both data sets and format catalogs. VALIDFMTNAME= does not control the length of format and informat names. It only controls the length of format and informat names that you associate with variables when you create a SAS data set.

If a SAS data set has a variable with a long format or informat name, which means that a release before SAS 9 cannot read it, then you can remove the long name so that the data set can be accessed by an earlier release. However, in order to retain the format attribute of the variable, an identical format with a short name would have to be applied to the variable.

**Note:** After you create a format or informat using a name that is longer than eight characters, if you rename it using eight or fewer characters, a release before SAS 9 cannot use the format or informat. You must recreate the format or informat using the shorter name.

**See Also**

- “Names in the SAS Language” in Chapter 3 of *SAS Language Reference: Concepts*
- Chapter 33, “SAS 9.3 Compatibility with SAS Files from Earlier Releases,” in *SAS Language Reference: Concepts*

**Procedures:**

- Chapter 23, “FORMAT Procedure” in *Base SAS Procedures Guide*
VALIDMEMNAME= System Option

Specifies the rules for naming SAS data sets, SAS data views, and item stores.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Files: SAS Files

**PROC OPTIONS GROUP= SASFILES**

**Applies to:** Base SAS engine and SPD Engine

**Restriction:** The VALIDMEMNAME= option is not supported by the tape engines V9TAPE, V8TAPE, V7TAPE, and V6TAPE

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

```
VALIDMEMNAME=COMPATIBLE | EXTEND
```

**Syntax Description**

**COMPATIBLE**

specifies that a SAS data set name, a SAS data view name, or an item store name must follow these rules:

- The length of the names can be up to 32 characters.
- Names must begin with a letter of the Latin alphabet (A–Z, a–z) or an underscore. Subsequent characters can be letters of the Latin alphabet, numerals, or underscores.
- Names cannot contain blanks or special characters except for the underscore.
- Names can contain mixed-case letters. SAS internally converts the member name to uppercase. Therefore, you cannot use the same member name with a different combination of uppercase and lowercase letters to represent different variables. For example, `customer`, `Customer`, and `CUSTOMER` all represent the same member name. How the name is saved on disk is determined by the operating environment.

This is the default.

**Alias:** COMPAT

**EXTEND**

specifies that a SAS data set name, a SAS data view name, or an item store name must follow these rules:

- Names can include national characters.
- The name can include special characters, except for the `\ * ? " < > | : -` characters.

**Note:** The SPD Engine does not allow ‘.’ (the period) anywhere in the member name.
• The name must contain at least one character.
• The length of the name can be up to 32 bytes.
• Null bytes are not allowed.
• Names cannot begin with a blank or a ‘.’ (the period).
  Note: The SPD Engine does not allow ‘$’ as the first character of the member name.
• Leading and trailing blanks are deleted when the member is created.
• Names can contain mixed-case letters. SAS internally converts the member name to uppercase. Therefore, you cannot use the same member name with a different combination of uppercase and lowercase letters to represent different variables. For example, customer, Customer, and CUSTOMER all represent the same member name. How the name appears is determined by the operating environment.

Restriction: The windowing environment supports the extended rules in the Editor, Log, and Output windows when VALIDMEMNAME=EXTEND is set. In most SAS windows, these extended rules are not supported. For example, these rules are not supported in SAS Explorer, the VIEWTABLE window, and windows that you open using the Solutions menu.

Requirement: When VALIDMEMNAME=EXTEND, SAS data set names, SAS data view names, and item store names must be written as a SAS name literal. If you use either the percent sign (%) or the ampersand (&), then you must use single quotation marks in the name literal in order to avoid interaction with the SAS Macro Facility. For more information, see “SAS Name Literals” in Chapter 3 of SAS Language Reference: Concepts.

Tip: The name displays in uppercase letters.


Examples:
  data “August Purchases”n;
  data ‘Años de empleo’n.;

CAUTION: Throughout SAS, using the name literal syntax with SAS member names that exceed the 32-byte limit or that have excessive embedded quotation marks might cause unexpected results. The intent of the VALIDMEMNAME=EXTEND system option is to enable compatibility with other DBMS member naming conventions, such as allowing embedded blanks and national characters.

CAUTION: Using the special character # when VALIDMEMNAME=EXTEND could cause a SAS data set to be overwritten by a generation data set. When VALIDMEMNAME= is set to EXTEND, you can name a SAS data set that uses the naming conventions for generation data sets, which append the special character # and a three-digit number to its member name. To avoid conflict, do not name SAS data sets similar to archived SAS data sets. For example, for a data set named A, generation data sets would automatically be named A#001, A#002, and so on. If you name a SAS data set A#003, the SAS data set could be deleted by SAS in the process of adding to a generation group.

Details

When VALIDMEMNAME= EXTEND, valid characters that are allowed in a SAS data set name, SAS data view name, and an item store name are extended to these characters:
Only the DATA, VIEW, and ITEMSTORE SAS member types support the extension of characters. The other member types, such as CATALOG and PROGRAM, do not support the extended characters. INDEX and AUDIT types that exist only with the associated DATA member support extended characters.

See Also


System Options:

- “VALIDVARNAME= System Option” on page 282

### VALIDVARNAME= System Option

Specifies the rules for valid SAS variable names that can be created and processed during a SAS session.

<table>
<thead>
<tr>
<th>Valid in:</th>
<th>Configuration file, SAS invocation, OPTIONS statement, SAS System Options window</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>Files: SAS Files</td>
</tr>
<tr>
<td>PROC OPTIONS GROUP=</td>
<td>SASFILES</td>
</tr>
<tr>
<td>Default:</td>
<td>V7</td>
</tr>
<tr>
<td>Note:</td>
<td>This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.</td>
</tr>
</tbody>
</table>

#### Syntax

```
VALIDVARNAME= V7 | UPCASE | ANY
```

#### Syntax Description

- **V7**
  
  specifies that variable names must follow these rules:
  
  - The length of a SAS variable names can be up to 32 characters.
  - The first character must begin with a letter of the Latin alphabet (A - Z, a - z) or the underscore. Subsequent characters can be letters of the Latin alphabet, numerals, or underscores.
  - Trailing blanks are ignored. The variable name alignment is left justified.
  - A variable name cannot contain blanks or special characters except for the underscore.
  - A variable name can contain mixed-case letters. SAS stores and writes the variable name in the same case that is used in the first reference to the variable. However, when SAS processes a variable name, SAS internally converts it to
uppercase. Therefore, you cannot use the same variable name with a different combination of uppercase and lowercase letters to represent different variables. For example, `cat`, `Cat`, and `CAT` all represent the same variable.

- Do not assign variables the names of special SAS automatic variables (such as `_N_` and `_ERROR_`) or variable list names (such as `_NUMERIC_`, `_CHARACTER_`, and `_ALL_`) to variables.

Examples:

```sas
season='summer';
percent_of_profit=percent;
```

UPCASE

specifies that the variable name follows the same rules as V7, except that the variable name is uppercase, as in earlier versions of SAS.

ANY

specifies that SAS variable names must follow these rules:

- The name can begin with or contain any characters, including blanks, national characters, special characters, and multi-byte characters.
- The name can be up to 32 bytes in length
- The name cannot contain any null bytes
- Leading blanks are preserved, but trailing blanks are ignored
- The name must contain at least one character. A name with all blanks is not permitted.
- The name contain mixed-case letters. SAS stores and writes the variable name in the same case that is used in the first reference to the variable. However, when SAS processes a variable name, SAS internally converts it to uppercase. Therefore, you cannot use the same variable name with a different combination of uppercase and lowercase letters to represent different variables. For example, `cat`, `Cat`, and `CAT` all represent the same variable.

Requirement: If you use any characters other than the ones that are valid when the VALIDVARNAME system option is set to V7 (letters of the Latin alphabet, numerals, or underscores), then you must express the variable name as a name literal and you must set VALIDVARNAME=ANY. If the name includes either the percent sign (%) or the ampersand (&), then you must use single quotation marks in the name literal in order to avoid interaction with the SAS Macro Facility. See “SAS Name Literals” in Chapter 3 of SAS Language Reference: Concepts and “Avoiding Errors When Using Name Literals” in Chapter 3 of SAS Language Reference: Concepts.


Examples:

```sas
'\% of profit'n=percent;
'items@warehouse'n=itemnum;
```

CAUTION: Throughout SAS, using the name literal syntax with SAS member names that exceed the 32-byte limit or have excessive embedded quotation marks might cause unexpected results. The intent of the VALIDVARNAME=ANY system option is to enable compatibility with other DBMS variable (column) naming conventions, such as allowing embedded blanks and national characters.
See Also


System Options:
- “VALIDMEMNAME= System Option” on page 280

VARLENCHK= System Option
Specifies the type of message to write to the SAS log when the input data set is read using the SET, MERGE, UPDATE, or MODIFY statements.

| Valid in: | Configuration file, SAS invocation, OPTIONS statement, SAS System Options window |
| Category: | Files: SAS Files |
| PROC OPTIONS GROUP= | SASFILES |

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

VARLENCHK=NOWARN | WARN | ERROR

Syntax Description

NOWARN
specifies that no warning message is issued when the length of a variable that is being read is larger than the length that is defined for the variable.

WARN
specifies that a warning is issued when the length of a variable that is being read is larger than the length that is defined for the variable. This is the default.

ERROR
specifies that an error message is issued when the length of a variable that is being read is larger than the length that is defined for the variable.

Details

After a variable is defined, the length of a variable can be changed only by a LENGTH statement. If a variable is read by the SET, MERGE, UPDATE, or MODIFY statements and the length of the variable is longer than a variable of the same name, SAS issues a warning message and uses the shorter, original length of the variable. By using the shorter length, data will not be truncated.

When you intentionally truncate data, perhaps to remove unnecessary blanks from character variables, SAS issues a warning message that might not be useful to you. To make it so that SAS does not issue the warning message or set a nonzero return code, you can set the VARLENCHK= system option to NOWARN. When VARLENCHK=NOWARN, SAS does not issue a warning message and sets the return code to SYSRC=0.
Alternatively, if you set VARLENCHK=ERROR and the length of a variable that is being read is larger than the length that is defined for the variable, SAS issues an error and sets the return code SYSRC=8.

The VARLENCHECK= system option does not have any effect on BY variables named in a BY statement that follows a SET, MERGE, or UPDATE statement. The VARLENCHK= option applies only to variables with the same name that have different lengths in two or more data sets. BY variables are excluded by design.

Note: When a BY variable has different lengths in two or more data sets, a separate warning message is produced, which is the correct behavior.

WARNING: Multiple lengths were specified for the BY variable x by input data sets. This may cause unexpected results.

To avoid this warning message, you can specify the LENGTH statement prior to the SET, MERGE, or UPDATE statement to set the BY variable to the same length.

Examples

**Example 1: SAS Issues a Warning Message Merging Two Data Sets with Different Variable Lengths**

This example merges two data sets, the sashelp.class data set and the exam_schedule data set. The length of the variable Name is set to 8 by the first SET statement, set sashelp.class; The exam_schedule data set sets the length of Name to 10. When exam_schedule is read in the second SET statement, set exam_schedule key=Name;, SAS issues a warning message because the length of Name in the exam_schedule data set is longer than the length of Name in the sashelp.class data set, and data might have been truncated.

```sas
/* Create the exam_schedule data set. */
data exam_schedule(index=(Name));
  input Name : $10.
  Exam_Date : mmddyy10.;
  format Exam_Date mmddyy10.;
datalines;
Carol 06/09/2011
Hui 06/09/2011
Janet 06/09/2011
Geoffrey 06/09/2011
John 06/09/2011
Joyce 06/09/2011
Helga 06/09/2011
Mary 06/09/2011
Roberto 06/09/2011
Ronald 06/09/2011
Barbara 06/10/2011
Louise 06/10/2011
Alfred 06/11/2011
Alice 06/11/2011
Henri 06/11/2011
James 06/11/2011
Philip 06/11/2011
Tomas 06/11/2011
William 06/11/2011
;
run
```
/* Merge the data sets sashelp.class and exam_schedule */
data exams;
  set sashelp.class;
  set exam_schedule key=Name;
run;

The following SAS log shows the warning message:

Output 3.8  The Warning Message in the SAS Log

NOTE: The data set WORK.EXAM_SCHEDULE has 19 observations and 2 variables.
NOTE: DATA statement used (Total process time):
  real time           0.09 seconds
  cpu time            0.00 seconds

WARNING: Multiple lengths were specified for the variable Name by input data set(s). This may cause truncation of data.
Name=Henry Sex=M Age=14 Height=63.5 Weight=102.5 Exam_Date=06/09/2011 _ERROR_=1 _IORC_=1230015 _N_=5
Name=Jane Sex=F Age=12 Height=59.8 Weight=84.5 Exam_Date=06/11/2011 _ERROR_=1 _IORC_=1230015 _N_=7
Name=Jeffrey Sex=M Age=13 Height=62.5 Weight=84 Exam_Date=06/09/2011 _ERROR_=1 _IORC_=1230015 _N_=9
Name=Judy Sex=F Age=14 Height=64.3 Weight=90 Exam_Date=06/09/2011 _ERROR_=1 _IORC_=1230015 _N_=12
Name=Robert Sex=M Age=12 Height=64.8 Weight=128 Exam_Date=06/11/2011 _ERROR_=1 _IORC_=1230015 _N_=16
Name=Thomas Sex=M Age=11 Height=57.5 Weight=85 Exam_Date=06/09/2011 _ERROR_=1 _IORC_=1230015 _N_=18
NOTE: There were 19 observations read from the data set SASHELP.CLASS.
NOTE: The data set WORK.EXAMS has 19 observations and 6 variables.

Example 2: Turn Off the Warning Message and Use the LENGTH Statement to Match Variable Lengths

In order to merge the two data sets, sashelp.class and exam_schedule, you can examine the values of Name in exam_schedule. You can see that there are no values that are greater than 8 and that you can change the length of Name without losing data.

To change the length of the variable Name, you use a LENGTH= statement in a DATA step before the set exam_schedule; statement. If the value of VARLENCHK is WARN (the default), SAS issues the warning message that the value of Name is truncated when it is read from work.exam_schedule. Because you know that data is not lost, you might want to turn the warning message off:
options varlenchk=nowarn;
data exam_schedule(index=(Name));
    length Name $ 8;
    set exam_schedule;
run;

The following is the SAS log output:

76   options varlenchk=nowarn;
77   data exam_schedule(index=(Name));
78     length Name $ 8;
79     set exam_schedule;
80   run;

NOTE: There were 19 observations read from the data set WORK.EXAM_SCHEDULE.
NOTE: The data set WORK.EXAM_SCHEDULE has 19 observations and 2 variables.

See Also

“Looking at Sources of Common Problems” in Chapter 21 of SAS Language Reference: Concepts

VBUFSIZE= System Option

Specifies the size of the view buffer.

Valid in: Configuration file, SAS invocation, OPTIONS statement, System Options window

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

Restriction: The VBUFSIZE= system option does not apply to SQL views.

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

VBUFSIZE=n | nK | nM | nG | nT | hexX | MIN | MAX

Required Arguments

n | nK | nM | nG | nT
specifies the size of the view buffer in multiples of 1 (bytes); 1,024 (kilobytes); 1,048,576 (megabytes); 1,073,741,824 (gigabytes); or 1,099,511,627,776 (terabytes). For example, a value of 8 specifies 8 bytes, and a value of 3m specifies 3,145,728 bytes.

Default: 32767

hexX
specifies the size of the view buffer as a hexadecimal value. You must specify the value beginning with a number (0–9), followed by an X. For example, the value fffeX specifies a buffer size of 65,534 bytes.

MIN
sets the minimum number of buffers to 0.
MAX

sets the view buffer size to $2^{63}-1$, or approximately 9.2 quintillion bytes.

**Note:** If you set VBUFSIZE=MAX and your system does not have enough memory, SAS stops processing the view.

**Details**

The view buffer is a segment of memory that is allocated to hold output observations that are generated for a view. The size of the buffer determines how much data can be held in memory at one time.

The view buffer is shared between the request that opens the view (for example, a SAS procedure) and the view itself. Two computer tasks coordinate between requesting data and generating and returning the data as follows:

- When a request task (such as a PRINT procedure) requests data, task switching occurs from the request task to the view task in order to execute the view and generate the observations. The view fills the view buffer with as many observations as possible.
- When the view buffer is full, task switching occurs from the view task back to the request task in order to return the requested data. The observations are cleared from the view buffer.

The size of the view buffer and the size of an observation determine how many observations can be held in the buffer. To determine the observation length, use PROC CONTENTS for the view. The number of observations then determines how many times the computer must switch between the request task and the view task. The larger the view buffer is, the less task switching is needed to process a view, which can speed up execution time.

To improve efficiency, first determine how many observations fit into the default buffer size, then set the view buffer so that it can hold more generated observations.

If OBSBUF= is set for a view, SAS uses the value of OBSBUF= and not the value of VBUFSIZE= to determine the size of the view buffer.

The view buffer is released when the view completes execution.

**Comparisons**

The VBUFSIZE= system option enables you to specify the size of the view buffer based on a number of bytes. The number of observations that can be read into the view buffer at one time is the value of VBUFSIZE= divided by the length of the observation. VBUFSIZE= is a system option and is set for the length of a SAS session.

The OBSBUF= data set option sets the view buffer size based on a specified number of observations that can be read into the view buffer at one time. The size of the view buffer is determined by the value of OBSBUF= multiplied by the length of the observation. OBSBUF= is a data set option and is set for the length of processing a view.

**See Also**

**Data Set Options:**

- “OBSBUF= Data Set Option” in *SAS Data Set Options: Reference*
VIEWMENUSystem Option
Specifies whether the View menu is included in SAS windows.

**Valid in:** Configuration file, SAS invocation

**Category:** Environment control: Display

**PROC OPTIONS**

**GROUP= ENVDISPLAY**

**Default:** VIEWMEN

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

VIEWMEN | NOVIEWMEN

**Syntax Description**

**VIEWMEN** specifies that the View menu is included in SAS windows.

**NOVIEWMEN** specifies that the View menu is not included in SAS windows.

---

VNFERR System Option
Specifies whether SAS issues an error or warning when a BY variable exists in one data set but not another data set when the other data set is _NULL_. This option applies when processing the SET, MERGE, UPDATE, or MODIFY statements.

**Valid in:** Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

**Category:** Environment control: Error handling

**PROC OPTIONS**

**GROUP= ERRORHANDLING**

**Note:** This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

---

**Syntax**

VNFERR | NOVNFERR

**Syntax Description**

**VNFERR** specifies that SAS issue an error when a BY variable exists in one data set but not in another data set when the other data set is _NULL_. This option applies when
processing the SET, MERGE, UPDATE, or MODIFY statements. When the error occurs, SAS enters into syntax-check mode.

**NOVNFERR**

specifies that SAS issue a warning when a BY variable exists in one data set but not in another data set when the other data set is _NULL_. This option applies when processing the SET, MERGE, UPDATE, or MODIFY statements. When the warning occurs, SAS does not enter into syntax-check mode.

**Details**

VNF stands for variable not found.

This option is useful when macro variables store data set names and these macro variables are used by the SET, MERGE, UPDATE, or MODIFY statements. If you set NOVNFERR and one of these statements contains a macro variable with a value _NULL_, SAS issues a warning instead of an error and processing continues.

**z/OS Specifics**

Under z/OS, SAS issues an error or a warning when the data set that is specified by DDNAME points to a DUMMY library.

**Comparisons**

• VNFERR is similar to the BYERR system option, which issues an error and enters into syntax-check mode if the SORT procedure attempts to sort a _NULL_ data set.

• VNFERR is similar to the DSNFERR system option, which issues an error when a SAS data set is not found.

**Examples**

**Example 1**

This example shows the results of setting the VNFERR option and the NOVNFERR option:

```sas
/* treat variable not found on _NULL_ SAS data set as an error */

/* turn option off - should not get an error */
options novnferr; run;

data a;
  x = 1;
  y = 2;
run;

data b;
  x = 2;
  y = 3;
run;

data _null;
  y = 2;
run;

/* option is off - should not get an error */
data result;
```

290  Chapter 3 • Dictionary of System Options
merge a b _null_;
   by x;
run;

/* turn option on - should get an error */
options vnferr; run;

data result2;
   merge a b _null_;
   by x;
run;

Here is the SAS log:
/* treat variable not found on _NULL_ SAS data set as an error */

/* turn option off - should not get an error */
options novnferr; run;

data a;
  x = 1;
  y = 2;
run;

NOTE: The data set WORK.A has 1 observations and 2 variables.
NOTE: DATA statement used (Total process time):
  real time           0.01 seconds
  cpu time            0.00 seconds

data b;
  x = 2;
  y = 3;
run;

NOTE: The data set WORK.B has 1 observations and 2 variables.
NOTE: DATA statement used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds

data _null;
  y = 2;
run;

NOTE: The data set WORK._NULL has 1 observations and 1 variables.
NOTE: DATA statement used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds

/* option is off - should not get an error */
data result;
  merge a b _null_
  by x;
run;

WARNING: BY variable x is not on input data set WORK._null_.
NOTE: There were 1 observations read from the data set WORK.A.
NOTE: There were 1 observations read from the data set WORK.B.
NOTE: The data set WORK.RESULT has 2 observations and 2 variables.
NOTE: DATA statement used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds

/* turn option on - should get an error */
options vnferr; run;

data result2;
  merge a b _null_
  by x;
run;

ERROR: BY variable x is not on input data set WORK._null_.
NOTE: The SAS System stopped processing this step because of errors.
WARNING: The data set WORK.RESULT2 may be incomplete.  When this step was stopped there were 0 observations and 2 variables.
Example 2
In this example, the data set, Result, reads from three data sets by using the SET statement. The SET statement values are all macro variables. One of these macro variables, &dataset3, has a value of _NULL_. SAS issues a warning message when it reads &dataset3; and completes the DATA step without an error.

```
options novnferr;

data a;
  x = 1;
  y = 2;
run;
data b;
  x = 2;
  y = 3;
run;

%let dataset1=a;
%let dataset2=b;
%let dataset3=_null_;

data result;
  set &dataset1 &dataset2 &dataset3;
  by x;
run;
```

Here is the SAS log:
options novnferr;
data a;
x = 1;
y = 2;
run;

NOTE: The data set WORK.A has 1 observations and 2 variables.
NOTE: DATA statement used (Total process time):
     real time       0.01 seconds
     cpu time       0.01 seconds

data b;
x = 2;
y = 3;
run;

NOTE: The data set WORK.B has 1 observations and 2 variables.
NOTE: DATA statement used (Total process time):
     real time       0.00 seconds
     cpu time       0.00 seconds

%let dataset1=a;
%let dataset2=b;
%let dataset3=_null_

data result;
set &dataset1 &dataset2 &dataset3;
by x;
run;

WARNING: BY variable x is not on input data set WORK._null_.
NOTE: There were 1 observations read from the data set WORK.A.
NOTE: There were 1 observations read from the data set WORK.B.

See Also
• “Syntax Check Mode” in Chapter 8 of SAS Language Reference: Concepts

System Options:
• “BYERR System Option” on page 80
• “DSNFERR System Option” on page 118

WORK= System Option

Specifies the Work library.

Valid in: Configuration file, SAS invocation
Category: Environment control: Files
PROC OPTIONS
GROUP= ENVFILES

Note: This option can be restricted by a site administrator on UNIX. It cannot be restricted by a site administrator on Windows or z/OS. For more information, see “Restricted Options” on page 6.
See:  “WORK System Option: UNIX” in SAS Companion for UNIX Environments  
“WORK System Option: Windows” in SAS Companion for Windows  
“WORK= System Option: z/OS” in SAS Companion for z/OS

Syntax

\texttt{WORK=library-specification}

Syntax Description

\textit{library-specification}

specifies the libref or physical name of the storage space where all data sets with 
one-level names are stored. This library must exist.

\textbf{Operating environment:} A valid library specification and its syntax are specific to 
your operating environment. On the command line or in a configuration file, the 
syntax is specific to your operating environment. For details, see the SAS 
documentation for your operating environment.

Details

This library is deleted at the end of your SAS session by default. To prevent the files 
from being deleted, specify the NOWORKTERM system option.

See Also

System Options:

- “WORKTERM System Option” on page 296

\textbf{WORKINIT System Option}

Specifies whether to initialize the Work library at SAS invocation.

\textbf{Valid in:} Configuration file, SAS invocation

\textbf{Category:} Environment control: Files

\textbf{PROC OPTIONS GROUP=} ENVFILES

\textbf{Note:} This option can be restricted by a site administrator. For more information, see 
“Restricted Options” on page 6.

\textbf{See:} “WORKINIT System Option: UNIX” in SAS Companion for UNIX Environments

Syntax

\texttt{WORKINIT | NOWORKINIT}

Syntax Description

\texttt{WORKINIT}

erases files that exist from a previous SAS session in an existing Work library at 
SAS invocation.
NOWORKINIT

does not erase files from the Work library at SAS invocation.

Details

The WORKINIT system option initializes the Work data library and erases all files from a previous SAS session at SAS invocation. The WORKTERM system option controls whether SAS erases Work files at the end of a SAS session.

UNIX Specifics

WORKINIT has behavior and functions specific to the UNIX operating environment. For details, see the SAS documentation for the UNIX operating environment.

See Also

System Options:

• “WORKTERM System Option” on page 296

WORKTERM System Option

Specifies whether to erase the Work files when SAS terminates.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

WORKTERM | NOWORKTERM

Syntax Description

WORKTERM

erases the Work files at the termination of a SAS session.

NOWORKTERM

does not erase the Work files.

Details

Although NOWORKTERM prevents the Work data sets from being deleted, it has no effect on initialization of the Work library by SAS. SAS normally initializes the Work library at the start of each session, which effectively destroys any pre-existing information.
Comparisons

Use the NOWORKINIT system option to prevent SAS from erasing existing Work files on invocation. Use the NOWORKTERM system option to prevent SAS from erasing existing Work files on termination.

See Also

System Options:

- “WORKINIT System Option” on page 295

YEARCUTOFF= System Option

Specifies the first year of a 100-year span that is used by date informats and functions to read a two-digit year.

Valid in: Configuration file, SAS invocation, OPTIONS statement, SAS System Options window

Category: Input control: Data processing

PROC OPTIONS

GROUP= INPUTCONTROL

Note: This option can be restricted by a site administrator. For more information, see “Restricted Options” on page 6.

Syntax

YEARCUTOFF= nnnn | nnnnn

Syntax Description

nnnn | nnnnn

- specifies the first year of the 100-year span.

Default: 1920

Range: 1582–19900

Details

The YEARCUTOFF= value is the default that is used by various date and datetime informats and functions.

If the default value of nnnn (1920) is in effect, the 100-year span begins with 1920 and ends with 2019. Therefore, any informat or function that uses a two-digit year value that ranges from 20 to 99 assumes a prefix of 19. For example, the value 92 refers to the year 1992.

The value that you specify in YEARCUTOFF= can result in a range of years that span two centuries. For example, if you specify YEARCUTOFF=1950, any two-digit value between 50 and 99 inclusive refers to the first half of the 100-year span, which is in the 1900s. Any two-digit value between 00 and 49, inclusive, refers to the second half of the 100-year span, which is in the 2000s. The following figure illustrates the relationship between the 100-year span and the two centuries if YEARCUTOFF=1950.
Figure 3.1  A 100-Year Span with Values in Two Centuries

Note: YEARCUTOFF= has no effect on existing SAS dates or dates that are read from input data that include a four-digit year, except years with leading zeros. For example, 0076 with yearcutoff=1990 indicates 2076.

See Also

Part 4

SAS Procedures for SAS System Options

Chapter 4
OPTIONS Procedure ....................................................... 301

Chapter 5
OPTLOAD Procedure ......................................................... 321

Chapter 6
OPTSAVE Procedure ......................................................... 327
Overview: OPTIONS Procedure

The OPTIONS procedure lists the current settings of SAS system options in the SAS log.

SAS system options control how SAS formats output, handles files, processes data sets, interacts with the operating environment, and does other tasks that are not specific to a single SAS program or data set. You use the OPTIONS procedure to obtain information about an option or a group of options. Here is some of the information that the OPTIONS procedure provides:

• the current value of an option and how it was set
• a description of an option
• valid syntax for the option, valid option values, and the range of values
• where you can set the system option
• if the option can be restricted by your site administrator
• if the option has been restricted
• system options that belong to a system option group
• system options that are specific for an operating environment
• if an option value has been modified by the INSERT or APPEND system options

For additional information about SAS system options, see *SAS System Options: Reference*.

**Syntax: OPTIONS Procedure**

**PROC OPTIONS** <option(s)>;

<table>
<thead>
<tr>
<th>Statement</th>
<th>Task</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>“PROC OPTIONS Statement”</td>
<td>List the current system option settings to the SAS Log</td>
<td>Ex. 1, Ex. 2, Ex. 3, Ex. 4</td>
</tr>
</tbody>
</table>

**PROC OPTIONS Statement**

Lists the current settings of SAS system options in the SAS log.

**Examples:**

- “Example 1: Producing the Short Form of the Options Listing” on page 316
- “Example 2: Displaying the Setting of a Single Option” on page 317
- “Example 3: Displaying Expanded Path Environment Variables” on page 318
- “Example 4: List the Options That Can Be Specified by the INSERT and APPEND Options” on page 319

**Syntax**

**PROC OPTIONS** <option(s)>;

**Summary of Optional Arguments**

**LISTGROUPS**

displays system option groups as well as a description of each group.

**Choose the format of the listing**

**DEFINE**

displays the short description of the option, the option group, and the option type.

**EXPAND**

when displaying a character option, replaces an environment variable in the option value with the value of the environment variable. EXPAND is ignored
if the option is a Boolean option, such as CENTER or NOCENTER, or if the value of the option is numeric.

HEXVALUE
displays system option character values as hexadecimal values.

LOGNUMBERFORMAT
displays numeric system option values using locale-specific punctuation.

LONG
lists each system option on a separate line with a description.

NOEXPAND
when displaying a path, displays the path using environment variable(s) and not the value of the environment variable(s). This is the default.

NOLOGNUMBERFORMAT
displays numeric system option values without using punctuation, such as a comma or a period. This is the default.

SHORT
specifies to display a compressed listing of options without descriptions.

VALUE
displays the option's value and scope, as well as how the value was set.

Restrict the number of options displayed

GROUP=group-name
gROUP=(group-name–1 ... group-name-n)
displays the options in one or more groups specified by group-name.

HOST
displays only host options.

LISTINSERTAPPEND
lists the system options whose value can be modified by the INSERT and APPEND system options.

LISTRESTRICT
lists the system options that can be restricted by your site administrator.

NOHOST
displays only portable options.

OPTION=option-name
OPTION=(option-name-1 ... option-name-n)
displays information about one or more system options.

RESTRICT
displays system options that the site administrator has restricted from being updated.

Optional Arguments

DEFINE
displays the short description of the option, the option group, and the option type. SAS displays information about when the option can be set, whether an option can be restricted, the valid values for the option, and whether the OPTSAVE procedure will save the option.

Interaction: This option is ignored when SHORT is specified.

Example: “Example 2: Displaying the Setting of a Single Option” on page 317

EXPAND
when displaying a character option, replaces an environment variable in the option value with the value of the environment variable. EXPAND is ignored if the option
is a Boolean option, such as CENTER or NOCENTER, or if the value of the option is numeric.

**Restriction:** Variable expansion is valid only in the Windows and UNIX operating environments.

**Tip:** By default, some option values are displayed with expanded variables. Other options require the EXPAND option in the PROC OPTIONS statement. Use the DEFINE option in the PROC OPTIONS statement to determine whether an option value expands variables by default or if the EXPAND option is required. If the output from PROC OPTIONS DEFINE shows the following information, you must use the EXPAND option to expand variable values:

Expansion: Environment variables, within the option value, are not expanded

**See:** “NOEXPAND” on page 305 option to view paths that display the environment variable

**Example:** “Example 3: Displaying Expanded Path Environment Variables” on page 318

**GROUP=** group-name
**GROUP=(**group-name–1 ... group-name–n**)**

- **GROUP=** displays the options in one or more groups specified by group-name.
- **Requirement:** When you specify more than one group, enclose the group names in parenthesis and separate the group names by a space.
- **See:** “Displaying Information about System Option Groups” on page 310

**HEXVALUE**

- **HEXVALUE** displays system option character values as hexadecimal values.

**HOST**

- **HOST** displays only host options.
- **See:** “NOHOST” on page 305 option to display only portable options.

**LISTINSERTAPPEND**

- **LISTINSERTAPPEND** lists the system options whose value can be modified by the INSERT and APPEND system options. The INSERT option specifies a value that is inserted as the first value of a system option value list. The APPEND option specifies a value that is appended as the last value of a system option value list. Use the LISTINSERTAPPEND option to display which system options can have values inserted at the beginning or appended at the end of their value lists.

- **See:** “INSERT= System Option” on page 159 and “APPEND= System Option” on page 68
- **Example:** “Example 4: List the Options That Can Be Specified by the INSERT and APPEND Options” on page 319

**LISTGROUPS**

- **LISTGROUPS** displays system option groups as well as a description of each group.
- **See:** “Displaying Information about System Option Groups” on page 310

**LISTRESTRICT**

- **LISTRESTRICT** lists the system options that can be restricted by your site administrator.
- **See:** “RESTRICT” on page 305 option to list options that have been restricted by the site administrator

**LONG**

- **LONG** lists each system option on a separate line with a description. This is the default. Alternatively, you can create a compressed listing without descriptions.
LOGNUMBERFORMAT

displays numeric system option values using locale-specific punctuation.

See: “NOLOGNUMBERFORMAT” on page 305 option to display numeric option
values without using commas

Example: “Example 2: Displaying the Setting of a Single Option” on page 317

NOEXPAND

when displaying a path, displays the path using environment variable(s) and not the
value of the environment variable(s). This is the default.

See: “EXPAND” on page 303 option to display a path by expanding the value of
environment variables

NOHOST

displays only portable options.

Alias: PORTABLE or PORT

See: “HOST” on page 304 option to display only host options

NOLOGNUMBERFORMAT

displays numeric system option values without using punctuation, such as a comma
or a period. This is the default.

See: “LOGNUMBERFORMAT” on page 305 option to display numeric system
options using commas

OPTION=option-name

OPTION=(option-name-1 ... option-name-n)

displays a short description and the value (if any) of the option specified by option-
name. DEFINE and VALUE options provide additional information about the
option.

option-name

specifies the option to use as input to the procedure.

Requirement: If a SAS system option uses an equal sign, such as PAGESIZE=, do
not include the equal sign when specifying the option to OPTION=.

Example: “Example 2: Displaying the Setting of a Single Option” on page 317

RESTRICT

displays the system options that have been set by your site administrator in a
restricted options configuration file. These options cannot be changed by the user.
For each option that is restricted, the RESTRICT option displays the option's value,
scope, and how it was set.

If your site administrator has not restricted any options, then the following message
appears in the SAS log:

Your Site Administrator has not restricted any SAS options.

See: “LISTRESTRICT” on page 304 option to list options that can be restricted by
the site administrator

SHORT

specifies to display a compressed listing of options without descriptions.

See: “LONG” on page 304 option to create a listing with descriptions of the
options.
VALUE

displays the option's value and scope, as well as how the value was set. If the value
was set using a configuration file, the SAS log displays the name of the configuration
file. If the option was set using the INSERT or APPEND system options, the SAS
log displays the value that was inserted or appended.

Interaction: This option has no effect when SHORT is specified.

Note: SAS options that are passwords, such as EMAILPW and METAPASS, return
the value xxxxxxxx and not the actual password.

Example: “Example 2: Displaying the Setting of a Single Option” on page 317

Displaying a List of System Options

The log that results from running PROC OPTIONS can show the system options for the
options that are available for all operating environment and those that are specific to a
single operating environment. Options that are available for all operating environments
are referred to as portable options. Options that are specific to a single operating
environment are referred to as host options.

The following example shows a partial log that displays the settings of portable options.

    proc options;
    run;
### Log 4.1  The SAS Log Showing a Partial Listing of SAS System Options

**Portable Options:**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPEND=</td>
<td>Append at the end of the option value</td>
</tr>
<tr>
<td>APPLETLOC=site-specific-path</td>
<td>Location of Java applets</td>
</tr>
<tr>
<td>ARMAGENT=</td>
<td>ARM Agent to use to collect ARM records</td>
</tr>
<tr>
<td>ARMLOC=ARMLOG.LOG</td>
<td>Identify location where ARM records are to be written</td>
</tr>
<tr>
<td>ARMSUBSYS=(ARM_NONE)</td>
<td>Enable/Disable ARMing of SAS subsystems</td>
</tr>
<tr>
<td>AUTOCORRECT</td>
<td>Perform auto-correction for misspelled procedure names, keywords or global statement names</td>
</tr>
<tr>
<td>AUTOEXEC=</td>
<td>Identifies AUTOEXEC files used during initialization</td>
</tr>
<tr>
<td>AUTOSAVELOC=</td>
<td>Identifies the location where program editor contents are auto saved</td>
</tr>
<tr>
<td>NOAUTOSIGNON</td>
<td>SAS/CONNECT remote submit will not automatically attempt to SIGNON</td>
</tr>
<tr>
<td>BINDING=DEFAULT</td>
<td>Controls the binding edge for duplexed output</td>
</tr>
<tr>
<td>BOMFILE</td>
<td>Add Byte Order Mark when creating Unicode files</td>
</tr>
<tr>
<td>BOTTOMMARGIN=0.000 IN</td>
<td>Bottom margin for printed output</td>
</tr>
<tr>
<td>BUFNO=1</td>
<td>Number of buffers for each SAS data set</td>
</tr>
<tr>
<td>BUFSIZE=0</td>
<td>Size of buffer for page of SAS data set</td>
</tr>
<tr>
<td>BYERR</td>
<td>Set the error flag if a null data set is input to the SORT procedure</td>
</tr>
<tr>
<td>BYLINE</td>
<td>Print the BY line at the beginning of each BY group</td>
</tr>
<tr>
<td>BYSORTED</td>
<td>Require SAS data set observations to be sorted for BY processing</td>
</tr>
<tr>
<td>NOCAPS</td>
<td>Do not translate source input to uppercase</td>
</tr>
<tr>
<td>NOCARDIMAGE</td>
<td>Do not process SAS source and data lines as 80-byte records</td>
</tr>
<tr>
<td>CATCACHE=0</td>
<td>Number of SAS catalogs to keep in cache memory</td>
</tr>
<tr>
<td>CBUFNO=0</td>
<td>Number of buffers to use for each SAS catalog</td>
</tr>
<tr>
<td>CENTER</td>
<td>Center SAS procedure output</td>
</tr>
<tr>
<td>CGOPTIMIZE=3</td>
<td>Control code generation optimization</td>
</tr>
<tr>
<td>NOCHARCODE</td>
<td>Do not use character combinations as substitute for special characters not on the keyboard</td>
</tr>
<tr>
<td>CLEANUP</td>
<td>Attempt recovery from out-of-resources condition</td>
</tr>
<tr>
<td>NOCMDMAC</td>
<td>Do not support command-style macros</td>
</tr>
<tr>
<td>CMPLIB=</td>
<td>Identify previously compiled libraries of CMP subroutines to use when linking</td>
</tr>
<tr>
<td>CMPMODEL=BOTH</td>
<td>Identify CMP model storage type</td>
</tr>
<tr>
<td>CMPOPT=(NOEXTRAMATH NOMISSCHECK NOPRECISE NOGUARDCHECK NOGENSYMNAMES NOFUNCDIFFERENCING)</td>
<td>Enable SAS compiler performance optimizations</td>
</tr>
</tbody>
</table>

The log displays both portable and host options when you submit `proc options;`. To view only host options, use this version of the OPTIONS procedure:

```sas
proc options host;
run;
```
Log 4.2  The SAS Log Showing a Partial List of Host Options

1    proc options host;
2    run;

SAS (r) Proprietary Software Release xxx  TS1B0

Host Options:

ACCESSIBILITY=STANDARD
   Enable Extended Accessibility
ALTLOG= Specifies the destination for a copy of the SAS log
ALTPRINT= Specifies the destination for a copy of the SAS procedure
output file
AUTHPROVIDERDOMAIN=
   Authentication providers associated with domain suffixes
AUTHSERVER= Specify the authentication server or domain.
AWSCONTROL=(SYSTEMMENU MINMAX TITLE)
   Used to customize the appearance for the SAS AWS. Valid
parameters are: TITLE/ NOTITLE
   SYSTEMMENU/NOSYSTEMMENU MINMAX/NOMINMAX
AWSDEF=(0 0 80 80)
   Specify the initial size and position of the SAS AWS. This
should be specified as follows: 0 0 100 100
AWSMENU Show the main window's (AWS) menu.

Displaying Information about One or More Options

To view the setting of one or more particular options, you can use the OPTION= and
DEFINE options in the PROC OPTIONS statement. The following example shows a log
that PROC OPTIONS produces for a single SAS system option.

    proc options option=errorcheck define;
    run;
Log 4.3  The Setting of a Single SAS System Option

11 proc options option=errorcheck define;
  2 run;

SAS (r) Proprietary Software Release xxx  TS1B0

ERRORCHECK=NORMAL

Option Definition Information for SAS Option ERRORCHECK
Group= ERRORHANDLING
Group Description: Error messages and error conditions settings
Description: Level of special error processing to be performed
Type: The option value is of type CHARACTER
  Maximum Number of Characters: 10
  Casing: The option value is retained uppercased
  Quotes: If present during "set", start and end quotes are removed
  Parentheses: The option value does not require enclosure within parentheses. If present, the parentheses are retained.
  Expansion: Environment variables, within the option value, are not expanded
Number of valid values: 2
  Valid value: NORMAL
  Valid value: STRICT

When Can Set: Startup or anytime during the SAS Session
Restricted: Your Site Administrator can restrict modification of this option
Optsave: PROC Optsave or command Dmoptsave will save this option

To view the settings for more than one option, enclose the options in parentheses and separate the options with a space:

proc options option=(pdfsecurity pdfpassword) define;
  run;
Displaying Information about System Option Groups

Each SAS system option belongs to one or more groups, which are based on functionality, such as error handling or sorting. You can display a list of system-option groups and the system options that belong to one or more of the groups.

Use the LISTGROUPS option to display a list of system-option groups.
proc options listgroups;
run;
### Log 4.5  List of SAS System Option Groups

26   proc options listgroups;
27   run;

SAS (r) Proprietary Software Release xxx  TS1B0

<table>
<thead>
<tr>
<th>Option Groups</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP=ADABAS</td>
<td>ADABAS</td>
</tr>
<tr>
<td>GROUP=CODEGEN</td>
<td>Code generation</td>
</tr>
<tr>
<td>GROUP=COMMUNICATIONS</td>
<td>Networking and encryption</td>
</tr>
<tr>
<td>GROUP=DATACOM</td>
<td>Datacom</td>
</tr>
<tr>
<td>GROUP=DATAQUALITY</td>
<td>Data Quality</td>
</tr>
<tr>
<td>GROUP=DB2</td>
<td>DB2</td>
</tr>
<tr>
<td>GROUP=EMAIL</td>
<td>E-mail</td>
</tr>
<tr>
<td>GROUP=ENVDISPLAY</td>
<td>Display</td>
</tr>
<tr>
<td>GROUP=ENVFILES</td>
<td>Files</td>
</tr>
<tr>
<td>GROUP=ERRORHANDLING</td>
<td>Error handling</td>
</tr>
<tr>
<td>GROUP=EXECMODES</td>
<td>Initialization and operation</td>
</tr>
<tr>
<td>GROUP=EXTFILES</td>
<td>External files</td>
</tr>
<tr>
<td>GROUP=GRAPHICS</td>
<td>Driver settings</td>
</tr>
<tr>
<td>GROUP=HELP</td>
<td>Help</td>
</tr>
<tr>
<td>GROUP=IDMS</td>
<td>IDMS</td>
</tr>
<tr>
<td>GROUP=IMS</td>
<td>IMS</td>
</tr>
<tr>
<td>GROUP=INPUTCONTROL</td>
<td>Data Processing</td>
</tr>
<tr>
<td>GROUP=INSTALL</td>
<td>Installation</td>
</tr>
<tr>
<td>GROUP=ISPF</td>
<td>ISPF</td>
</tr>
<tr>
<td>GROUP=LANGUAGECONTROL</td>
<td>Language control</td>
</tr>
<tr>
<td>GROUP=LISTCONTROL</td>
<td>Procedure output</td>
</tr>
<tr>
<td>GROUP=LOGCONTROL</td>
<td>SAS log</td>
</tr>
<tr>
<td>GROUP=LOG_LISTCONTROL</td>
<td>SAS log and procedure output</td>
</tr>
<tr>
<td>GROUP=MACRO</td>
<td>SAS macro</td>
</tr>
<tr>
<td>GROUP=MEMORY</td>
<td>Memory</td>
</tr>
<tr>
<td>GROUP=META</td>
<td>Metadata</td>
</tr>
<tr>
<td>GROUP=ODSPRINT</td>
<td>ODS Printing</td>
</tr>
<tr>
<td>GROUP=PDF</td>
<td>PDF</td>
</tr>
<tr>
<td>GROUP=PERFORMANCE</td>
<td>Performance</td>
</tr>
</tbody>
</table>
Use the GROUP= option to display system options that belong to a particular group. You can specify one or more groups.

```
proc options group=(svg graphics);
run;
```

**Log 4.6 Sample Output Using the GROUP= Option**

```
5    proc options group=(svg graphics);
6    run;

SAS (r) Proprietary Software Release xxx  TS1B0

Group=SVG
NOSVGCONTROLBUTTONS
   Do not display paging control buttons in multi-page SVG documents
SVGHEIGHT=  Identifies the height of the SVG outermost element and height of the initial viewport
SVGPRESERVEASPECTRATIO=  Identifies the preserveAspectRatio attribute for the outermost SVG element
SVGTITLE=  Identifies SVG title element
SVGVIEWBOX=  Identifies the viewBox attribute for the outermost SVG element and width of the initial viewport
SVGX=  Identifies the x-axis attribute for the outermost SVG element
SVGY=  Identifies the y-axis attribute for the outermost SVG element

Group=GRAPHICS
DEVICE=  Graphics device driver
GSTYLE  Use ODS styles in the generation of graphs. GSTYLE does not affect ODS styles for graphs generated with Java, ActiveX drivers or Statistical Graphics
GWINDOW  Display SAS/GRAPH output in the GRAPH window of Display Manager
MAPS= ("!sasroot\your-site-path\en\maps")  Location of maps for use with SAS/GRAPH
FONTALIAS=  Assigns a new host font facename to a portable SAS font family. The first parameter is the SAS font family and the second is the host-specific font facename.
```

The following table lists the values that are available in all operating environments when you use the GROUP= option with PROC OPTIONS.
The following table lists operating environment–specific values that might be available when you use the GROUP= option with PROC OPTIONS.

### Possible Operating Environment-Specific Values for Use with GROUP=

<table>
<thead>
<tr>
<th>Values</th>
<th>Help</th>
<th>Meta</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODEGEN</td>
<td>INPUTCONTROL</td>
<td>ODSPRINT</td>
</tr>
<tr>
<td>COMMUNICATIONS</td>
<td>INSTALL</td>
<td>PDF</td>
</tr>
<tr>
<td>EMAIL</td>
<td>LANGUAGECONTROL</td>
<td>PERFORMANCE</td>
</tr>
<tr>
<td>ENVDISPLAY</td>
<td>LISTCONTROL</td>
<td>SASFILES</td>
</tr>
<tr>
<td>ENVFILES</td>
<td>LOG_LISTCONTROL</td>
<td>SECURITY</td>
</tr>
<tr>
<td>ERRORHANDLING</td>
<td>LOGCONTROL</td>
<td>SORT</td>
</tr>
<tr>
<td>EXECMODES</td>
<td>MACRO</td>
<td>SQL</td>
</tr>
<tr>
<td>EXTFILES</td>
<td>MEMORY</td>
<td>SVG</td>
</tr>
<tr>
<td>GRAPHICS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following table lists operating environment–specific values that might be available when you use the GROUP= option with PROC OPTIONS.

### Possible Operating Environment-Specific Values for Use with GROUP=

<table>
<thead>
<tr>
<th>Values</th>
<th>Help</th>
<th>Meta</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODEGEN</td>
<td>IDMS</td>
<td>ORACLE</td>
</tr>
<tr>
<td>DATACOM</td>
<td>IMS</td>
<td>REXX</td>
</tr>
</tbody>
</table>

**Operating Environment Information**

Refer to the SAS documentation for your operating environment for more information about these host-specific options.

---

### Displaying Restricted Options

Your site administrator can restrict some system options so that your SAS session adheres to options that are set for your site. Restricted options can be modified only by your site administrator. The OPTIONS procedure provides two options that display information about restricted options. The RESTRICT option lists the system options that your site administrator has restricted. The LISTRESTRICT option lists the options that can be restricted by your site administrator. For a listing of options that cannot be restricted, see Table 1.1 on page 8.

The following SAS logs shows the output when the RESTRICT option is specified and partial output when the LISTRESTRICT option is specified.
Log 4.7  A List of Options That Have Been Restricted by the Site Administrator

```
1   proc options restrict;
2   run;
SAS (r) Proprietary Software Release xxx  TS1B0
Option Value Information For SAS Option CMPOPT
   Option Value: (NOEXTRAMATH NOMISSCHECK NOPRECISE NOGUARDCHECK
   NOGENSYMNAME NOPUNCDIFFERENCING)
   Option Scope: SAS Session
   How option value set: Site Administrator Restricted
```

Log 4.8  A Partial Log That Lists Options That Can Be Restricted

```
33  proc options listrestrict;
34  run;
SAS (r) Proprietary Software Release xxx  TS1B0
Your Site Administrator can restrict the ability to modify the following Portable Options:
   APPEND       Append at the end of the option value
   APPLETLOC    Location of Java applets
   ARMAGENT     ARM Agent to use to collect ARM records
   ARMLOC       Identify location where ARM records are to be written
   ARMSUBSYS    Enable/Disable ARMing of SAS subsystems
   AUTOCORRECT  Perform auto-correction for misspelled procedure names,
                 keywords or global statement names
   AUTOSAVELOC  Identifies the location where program editor contents
                 are auto saved
```

Results: OPTIONS Procedure

SAS writes the options list to the SAS log. SAS system options of the form `option | NOoption` are listed as either `option` or `NOoption`, depending on the current setting. They are always sorted by the positive form. For example, NOCAPS would be listed under the Cs.

**Operating Environment Information**

PROC OPTIONS produces additional information that is specific to the environment under which you are running the SAS System. Refer to the SAS documentation for your operating environment for more information about this and for descriptions of host-specific options.

**See Also**

- *SAS Companion for UNIX Environments*
- *SAS Companion for Windows*
- *SAS Companion for z/OS*
Examples: OPTIONS Procedure

Example 1: Producing the Short Form of the Options Listing

Features: PROC OPTIONS statement option:
SHORT

Details
This example shows how to generate the short form of the listing of SAS system option settings. Compare this short form with the long form that is shown in “Displaying a List of System Options” on page 306.

Program

    proc options short;
    run;

Program Description

List all options and their settings. SHORT lists the SAS system options and their settings without any descriptions.

    proc options short;
    run;
Example 2: Displaying the Setting of a Single Option

**Features:**

PROC OPTIONS statement option:

<table>
<thead>
<tr>
<th>OPTION</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFINE</td>
<td></td>
</tr>
<tr>
<td>LOGNUMBERFORMAT</td>
<td></td>
</tr>
<tr>
<td>VALUE</td>
<td></td>
</tr>
</tbody>
</table>

**Details**

This example shows how to display the setting of a single SAS system option. The log shows the current setting of the SAS system option MEMBLKSIZE. The DEFINE and VALUE options display additional information. The LOGNUMBERFORMAT displays the value using commas.

**Program**

```sas
proc options option=memblksz define value lognumberformat;
run;
```
Program Description

Specify the MEMBLKSZ SAS system option. OPTION=MEMBLKSZ displays option value information. DEFINE and VALUE display additional information. LOGNUMBERFORMAT specifies to format the value using commas.

```sas
proc options option=memblksz define value lognumberformat;
run;
```

Log

Log 4.10  Log Output from Specifying the MEMBLKSZ Option

```
13 proc options option=memblksz define value lognumberformat;
14 run;
```

SAS (r) Proprietary Software Release xxx

Option Value Information For SAS Option MEMBLKSZ
Value: 16,777,216  Scope: Default  How option value set: Shipped Default

Option Definition Information for SAS Option MEMBLKSZ
Group= MEMORY  Group Description: Memory settings  Description: Size of memory blocks allocated to support MEMLIB and MEMCACHE options.  Type: The option value is of type INTMAX  Range of Values: The minimum is 0 and the maximum is 9223372036854775807  Valid Syntax(any casing): MIN|MAX|n|nK|nM|nG|nT|hexadecimal  Numeric Format: Usage of LOGNUMBERFORMAT does not impact the value format  When Can Set: Session startup (command line or config) only  Restricted: Your Site Administrator can restrict modification of this option  Optsave: PROC Optsave or command Dmoptsave will not save this option

Example 3: Displaying Expanded Path Environment Variables

Features: PROC OPTIONS statement options:
OPTION=
EXPAND
NOEXPAND
HOST

Details

This example shows the value of an environment variable when the path is displayed.

Program

```sas
proc options option=msg expand;
run;
```
Example 4: List the Options That Can Be Specified by the INSERT and APPEND Options

Features:
PROC OPTIONS statement option:
LISTINSERTAPPEND

Details
This example shows how to display the options that can be specified by the INSERT and APPEND system options.

Program

```sas
proc options listinsertappend;
run;
```
**Program Description**

List all options that can be specified by the INSERT and APPEND options. The LISTINSERTAPPEND option provides a list and a description of these options.

```
proc options listinsertappend;
run;
```

**Log**

**Log 4.12**  Displaying the Options That Can Be Specified by the INSERT and APPEND Options

<table>
<thead>
<tr>
<th>Core options that can utilize INSERT and APPEND</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTOEXEC</td>
</tr>
<tr>
<td>CMPLIB</td>
</tr>
<tr>
<td>FMTSEARCH</td>
</tr>
<tr>
<td>MAPS</td>
</tr>
<tr>
<td>SASAUTOS</td>
</tr>
<tr>
<td>SASHELP</td>
</tr>
<tr>
<td>SASSCRIPT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Host options that can utilize INSERT and APPEND</th>
</tr>
</thead>
<tbody>
<tr>
<td>HELPLOC</td>
</tr>
<tr>
<td>MSG</td>
</tr>
<tr>
<td>SET</td>
</tr>
</tbody>
</table>
Overview: OPTLOAD Procedure

The OPTLOAD procedure reads SAS system option settings that are stored in the SAS registry or a SAS data set and puts them into effect.

You can load SAS system option settings from a SAS data set or registry key by using one of these methods:

- the DMOPTLOAD command from a command line in the SAS windowing environment. For example, DMOPTLOAD key= “core/options”.
- the PROC OPTLOAD statement.

When an option is restricted by the site administrator, and the option value that is being set by PROC OPTLOAD differs from the option value that was established by the site administrator, SAS issues a warning message to the log.

Syntax: OPTLOAD Procedure

PROC OPTLOAD <options>;

<table>
<thead>
<tr>
<th>Statement</th>
<th>Task</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>“PROC OPTLOAD Statement”</td>
<td>Use SAS system option settings that are stored in the SAS registry or in a SAS data set</td>
<td>Ex. 1</td>
</tr>
</tbody>
</table>
PROC OPTLOAD Statement
Loads saved setting of SAS system options that are stored in the SAS registry or in a SAS data set.

Syntax
PROC OPTLOAD <options>;

Summary of Optional Arguments

DATA=libref.dataset
Load SAS system option settings from an existing data set.

KEY="SAS registry key"
Load SAS system option settings from an existing registry key.

Optional Arguments

DATA=libref.dataset
specifies the library and data set name from where SAS system option settings are loaded. The SAS variable OPTNAME contains the character value of the SAS system option name, and the SAS variable OPTVALUE contains the character value of the SAS system option setting.

Default: If you omit the DATA= option and the KEY= option, the procedure will use the default SAS library and data set. The default library is where the current user profile resides. Unless you specify a library, the default library is SASUSER. If SASUSER is being used by another active SAS session, then the temporary WORK library is the default location from which the data set is loaded. The default data set name is MYOPTS.

Requirement: The SAS library and data set must exist.

KEY="SAS registry key"
specifies the location in the SAS registry of stored SAS system option settings. The registry is retained in SASUSER. If SASUSER is not available, then the temporary WORK library is used. For example, KEY="OPTIONS".

Requirements:
"SAS registry key" must be an existing SAS registry key.
You must use quotation marks around the "SAS registry key" name. Separate the names in a sequence of key names with a backslash (\). For example, KEY="CORE:OPTIONS".

Example: Load a Data Set of Saved System Options

Features: PROC OPTLOAD option
DATA=
Details
This example saves the current system option settings using the OPTSAVE procedure, modifies the YEARCUTOFF system option, and then loads the original set of system options.

Program
libname mysas "c:\mysas";
proc options option=yearcutoff;
run;
proc optsave out=mysas.options;
run;
options yearcutoff=2000;
proc options option=yearcutoff;
run;
proc optload data=mysas.options;
run;
proc options option=yearcutoff;
run;

Program Description
These statements and procedures were submitted one at a time and not run as a SAS program to allow the display of the YEARCUTOFF option.

Assign the libref.
libname mysas "c:\mysas";

Display the value of the YEARCUTOFF= system option.
proc options option=yearcutoff;
run;

Save the current system option settings in mysas.options.
proc optsave out=mysas.options;
run;

Use the OPTIONS statement to set the YEARCUTOFF= system option to the value 2000.
options yearcutoff=2000;

Display the value of the YEARCUTOFF= system option.
proc options option=yearcutoff;
run;

Load the saved system option settings.
proc optload data=mysas.options;
run;
Display the value of the `YEARCUTOFF=` system option. After loading the saved system option settings, the value of the `YEARCUTOFF=` option has been restored to the original value.

```plaintext
proc options option=yearcutoff;
run;
```
The following is the SAS log output after submitting the previous statements and procedures:

```
libname mysas "c:\mysas";
NOTE: Libref MYSAS was successfully assigned as follows:
  Engine:        V9
  Physical Name: c:\mysas
proc options option=yearcutoff;
run;
SAS (r) Proprietary Software Release xxx  TS1B0
YEARCUTOFF=1920   Cutoff year for DATE and DATETIME informats and functions
NOTE: PROCEDURE OPTIONS used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds
proc optsave out=mysas.options;
run;
NOTE: The data set MYSAS.OPTIONS has 259 observations and 2 variables.
NOTE: PROCEDURE OPTSAVE used (Total process time):
  real time           0.03 seconds
  cpu time            0.03 seconds
options yearcutoff=2000;
proc options option=yearcutoff;
run;
SAS (r) Proprietary Software Release xxx  TS1B0
YEARCUTOFF=2000   Cutoff year for DATE and DATETIME informats and functions
NOTE: PROCEDURE OPTIONS used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds
proc optload data=mysas.options;
run;
NOTE: PROCEDURE OPTLOAD used (Total process time):
  real time           0.06 seconds
  cpu time            0.01 seconds
proc options option=yearcutoff;
run;
SAS (r) Proprietary Software Release xxx  TS1B0
YEARCUTOFF=1920   Cutoff year for DATE and DATETIME informats and functions
NOTE: PROCEDURE OPTIONS used (Total process time):
  real time           0.00 seconds
  cpu time            0.00 seconds
```
Overview: OPTSAVE Procedure

PROC OPTSAVE saves the current SAS system option settings in the SAS registry or in a SAS data set.

SAS system options can be saved across SAS sessions. You can save the settings of the SAS system options in a SAS data set or registry key by using one of these methods:

- the DMOPTSSAVE command from a command line in the SAS windowing environment. Use the command like this: DMOPTSSAVE <save-location>.
- the PROC OPTSAVE statement.

Syntax: OPTSAVE Procedure

PROC OPTSAVE <options>;

<table>
<thead>
<tr>
<th>Statement</th>
<th>Task</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>“PROC OPTSAVE Statement”</td>
<td>Save the current SAS system option settings to the SAS registry or to a SAS data set</td>
<td>Ex. 1</td>
</tr>
</tbody>
</table>
PROC OPTSAVE Statement
Saves the current SAS system option settings in the SAS registry or in a SAS data set.

Syntax
PROC OPTSAVE <options>;

Summary of Optional Arguments

KEY="SAS registry key"
   Save SAS system option settings to a registry key.
OUT=libref.dataset
   Save SAS system option settings to a SAS data set.

Optional Arguments

KEY="SAS registry key"
specifies the location in the SAS registry of stored SAS system option settings. The registry is retained in SASUSER. If SASUSER is not available, then the temporary WORK library is used. For example, KEY="OPTIONS".
Restriction: “SAS registry key” names cannot span multiple lines.
Requirements:
   Separate the names in a sequence of key names with a backslash (\). Individual key names can contain any character except a backslash.
   The length of a key name cannot exceed 255 characters (including the backslashes).
   You must use quotation marks around the “SAS registry key” name.
Tip: To specify a subkey, enter multiple key names starting with the root key.
CAUTION: If the key already exists, it will be overwritten. If the specified key does not already exist in the current SAS registry, then the key is automatically created when option settings are saved in the SAS registry.

OUT=libref.dataset
specifies the names of the library and data set where SAS system option settings are saved. The SAS variable OPTNAME contains the character value of the SAS system option name. The SAS variable OPTVALUE contains the character value of the SAS system option setting.
Default: If you omit the OUT= and the KEY= options, the procedure will use the default SAS library and data set. The default SAS library is where the current user profile resides. Unless you specify a SAS library, the default library is SASUSER. If SASUSER is in use by another active SAS session, then the temporary WORK library is the default location where the data set is saved. The default data set name is MYOPTS.
CAUTION: If the data set already exists, it will be overwritten.
Determining If a Single Option Can Be Saved

You can specify DEFINE in the OPTIONS procedure to determine whether an option can be saved. In the log output, the line beginning with **Optsave:** indicates whether the option can be saved.

```sas
proc options option=pageno define;
run;
```

Creating a List of Options That Can Be Saved

Some system options cannot be saved. To create a list of options that can be saved, submit this SAS code:

```sas
proc optsave;
run;
```

```sas
proc print;
  var optname;
run;
```

The PRINT procedure uses the value of the _LAST_ system option to determine the input data set. The default data set name for the OPTSAVE procedure is SASUSER.MYOPTS.

Example: Saving System Options in a Data Set

**Features:**

- **PROC OPTSAVE option:** `OUT=`
Details
This example saves the current system option settings using the OPTSAVE procedure.

Program

libname mysas "c:\mysas";

proc optsave out=mysas.options;
run;

Program Description

Create a libref.

libname mysas "c:\mysas";

Save the current system option settings.

proc optsave out=mysas.options;
run;

Log

The following is the SAS log output:

```plaintext
1  libname mysas "c:\mysas";
   NOTE: Libref MYSAS was successfully assigned as follows:
        Engine: V9
        Physical Name: c:\mysas
2  proc optsave out=mysas.options;
3  run;
   NOTE: The data set MYSAS.OPTIONS has 259 observations and 2 variables.
   NOTE: PROCEDURE OPTSAVE used (Total process time):
            real time           0.03 seconds
            cpu time            0.03 seconds
```
Index

Special Characters
_LAST_ = system option 168
%INCLUDE statement
  starting position for reading variable-sized record input 230

A
alignment of output 87
APPEND= system option 68
  modify system option values 13
applet location 70
APPLETLOC= system option 70
authentication provider 70
AUTHPD 70
AUTHPROVIDERDOMAIN system option 70
autocall macro files
  starting position for reading variable-sized record input 230
AUTOEXEC file
  echoing to the log 121
autoexec files
  starting position for reading variable-sized record input 230
autosave file
  location of 74
AUTOSAVELOC= system option 74

B
batch processing
  DATA and PROC step checkpoint-restart data 246
  error handling 131
labeled code sections checkpoint-restart data 167
recording checkpoint-restart data for 164, 243
bin specification
  for printed output 197
  name of paper bin 199
  binding edge 75
  BINGING= system option 75
  BOTTOMMARGIN= system option 76
  browsers
    for ODS output 149
    for SAS Help 149
  buffers
    extra buffers for navigating index files 154
    number for data sets 77
    page buffers for catalogs 86
    size of 78
    view size 287
    writing to disk 136
  BUFNO system option 77
  BUFSIZE= system option 78
BY lines
  printing 81
BY variables
  existing in one data set but not another 289
BYERR system option 80
BYLINE system option 81
BYSORTED system option 82

C
CAPS system option 83
CARDIMAGE system option 84
catalogs
  error handling 111
  number to keep open 85
  page buffers 86
  search order for 139
  CATCACHE= system option 85
  CBUFNO= system option 86
  CENTER system option 87
  CGOPTIMIZE= system option 87
  character combinations 88
CHARCODE system option 88
checkpoint-restart data
erase Work library 89
libref for DATA and PROC step library 244
libref for labeled code section library 165
recording for DATA and PROC steps 243
recording for labeled code sections 164
specifying batch programs with 167, 246
CHKPTCLEAN system option 89
CLEANUP system option 91
CMPLIB 93
CMPLIB= system option 93
CMPPMODEL= system option 94
CMPOPT= system option 95
code compilation
optimization level during 87
code generation optimization 95
COLLATE system option 97
collating output 97
color printing 98
COLORPRINTING system option 98
comments
in PDF documents 204
compilation
optimization level during 87
compiler optimization 95
compiler subroutines 93
COMPRESS= system option 99
compressed data sets
reusing space when adding observations 221
compressing observations 99
compression
for device drivers supporting Deflate algorithm 106
Universal Printers and SAS/GRAH files 271
COPIES 101
copies, specifying number of 101
COPIES= system option 101
copying
PDF documents 206
CPUCOUNT= system option 101
CPUID system option 103

D

damaged data sets or catalogs 111
data lines
as card images 84
length of sequence field 233
data set options

system options and 19
data sets
buffer size 78
compressing on output 99
damaged 111
loading system options from 321
damaged most recently created 168
not found 118
number of buffers 77
replacing permanently stored 220
reusing space in compressed data sets 221
saving system option settings in 327
DATA statement
keywords allowed in 103
DATA= option
PROC OPTLOAD statement 322
DATALINES statement
length of data 225
datasets
compiler subroutines in 93
DASTMTCHK= system option 103
date informats and functions
year cutoff 297
date stamp 104
DATE system option 104
DATESTYLE= system option 105
datetime informats and functions
year cutoff 297
DEFINE option
PROC OPTIONS statement 303
Deflate compression algorithm 106
DEFATION 106
DEFLATION= system option 106
DETAILS system option 107
DEVICE 108
device drivers
supporting Deflate Compression algorithm 106
DEVICE= system option 108
directory
creating 110
displaying system option information 308
DKRCOND= system option 108
DKROCOND= system option 109
DLCREATEDIR system option 110
DLDMGACTION= system option 111
DMOPTLOAD command 321
DMOPTSAVE command 327
DMR system option 112
DM system option 113
DMSEXPS system option 114
DMSLOGSIZE= system option 115
DMSOUTSIZE= system option 116
DMSPGMLINESIZE= system option 117
DMSSYNCHK system option 117
  domain suffix
  associating with authentication provider 70
DROP statement
  error detection for output data sets 109
  DROP= data set option
  error detection for input data sets 108
  DROP= DATA step option
  error detection for output data sets 109
DSNFERR system option 118
DTRESET system option 119
DUPLEX 120
duplex printing 120
DUPLEX system option 120

E
e-mail
  authentication protocol 122
  password 127
  UTC offset 128
ECHOTOAUTO system option 121
EMAILAUTHPROTOCOL= system option 122
EMAILCUTOFFSET= system option 128
EMAILFROM system option 123
EMAILHOST system option 123
EMAILID= system option 124
EMAILPORT system option 126
EMAILPW= system option 127
  encoding
  setting for the URLENCODE and URLDECODE functions 272
ENGINE 129
ENGINE= system option 129
  error detection levels
  input data sets 108
  output data sets 109
  error handling
  BY-group processing 131
catalogs 111
  format not found 138
  in batch processing 131
  numeric data 161
  error messages
  BY variable exists in one data set but not another 289
  maximum number printed 132
  overprinting 194
  SORT procedure 80
error response 130
ERRORABEND system option 130
ERRORBYABEND system option 131
ERRORCHECK= system option 131
ERRORS= system option 132
EVENTDS= system option 133
EXPLORER system option 135
Explorer window
  invoking 114, 135
eXternal files
  logical record length for reading and writing 176
  external programs
  passing parameter strings to 201
F
FILENAME statement EMAIL (SMTP)
  UTC offset 128
FILESYNC= system option 136
FIRSTOBS= system option 137
FMTERR system option 138
FMTSEARCH= system option 139
  font embedding 142
  font selector window
  listing only SAS fonts 180
FONTEMBEDDING system option 142
FONTRENDERING= system option 143
  fonts
  rendering with operating system or FreeType engine 143
FORMCHAR= system option 145
FORMAT catalog
  associated locale 139
format catalogs
  search order for 139
formats
  name length 278
  not found 138
  formatting characters 145
FORMCHAR= system option 145
FORMDLIM= system option 146
  forms
  default form for printing 147
FORMS= system option 147
FROM e-mail option 123
G
GETOPTION function 23
  changing YEARCUTOFF system option with 25
  obtaining reporting options 26
GRAPH window
  displaying SAS/GRAPH output in 148
  graphics options
  returning value of 23
GROUP= option
PROC OPTIONS statement 304
GRSEG catalog entries
ODS styles in graphs stored as 148
GSTYLE system option 148
GWINDOW system option 148

H
hardware information, writing to SAS log 103
Help
  browser for 149
  online training courses 270
  remote help browser 151
  remote help client 152
HELPBROWSER= system option 149
HELPENCMD system option 150
HELPPORT= system option 151
HELPPORT= system option 152
hexadecimal values
  for system options 4
HEXVALUE option
  PROC OPTIONS statement 304
holidays
  user-supplied 160
HOST option
  PROC OPTIONS statement 304
HTTP server
  highest port number for 152
  lowest port number for 153
HTTPSERVERPORTMAX= system option 152
HTTPSERVERPORTMIN= system option 153

I
IBUFNO= system option 154
IBUFSIZE 155
IBUFSIZE= system option 155
index files
  extra buffers for navigating 154
informs
  ambiguous data 105
  name length 278
INITCMD system option 156
initializing procedure output files 218
INITSTMT= system option 158
input
  as card images 84
  uppercasing 83
input data sets
  error detection levels 108
input source lines
  length of numeric portion of sequence field 233
INSERT system option 159
INSERT= system option
  modify system option values 13
  interval values
  user-supplied holidays 160
  INTERVALDS= system option
    user-supplied holidays 160
invalid data
  numeric 161
INVALIDDATA= system option 161
IS= system option 158
item stores
  naming 280

J
Java applet location 70
JPEG files
  quality factor 162
JPEGQUALITY= system option 162

K
KEEP statement
  error detection for output data sets 109
KEEP= data set option
  error detection for input data sets 108
KEEP= DATA step option
  error detection for output data sets 109
KEY= option
  PROC OPTLOAD statement 322
  PROC OPTSAVE statement 328
keyboard 88
keywords, allowed in DATA statement 103

L
LABEL system option 163
LABELCHKPT system option 164
LABELCHKPTLIB system option 165
LABELRESTART system option 167
labels
  using with variables in SAS procedures 163
landscape orientation 191
layout
  for PDF documents 209
left margin 169
LEFTMARGIN= system option 169
libraries
  creating a nonexistent library 110
  damaged data sets or catalogs 111
  default access method 129
  default permanent SAS library 273
  listing details 107
  SAS library to use as SASUSER library 232
librefs
  assigning user-defined, at startup 242
license information
  altering 234
line size
  Program Editor 117
LINESIZE= system option 170
LISTINSERTAPPEND
  PROC OPTIONS statement 304
LISTRESTRICT
  PROC OPTIONS statement 304
locale
  associated catalog 139
log files 171
Log window
  invoking 114
  maximum number of rows 115
  suppressing 156
logical record length
  for reading and writing external files 176
LOGNUMBERFORMAT
  PROC OPTIONS statement 305
LOGPARM= system option 171
LONG option
  PROC OPTIONS statement 304
LRECL= system option 176
maps
  location to search for 177
MAPS= system option 177
margins
  bottom margin size 76
  left margin 169
  right margin 222
  top margin 269
memory
  for data summarization procedures 247
  SORT procedure 237
menus
  SOLUTIONS menu in SAS windows 235
MERGE processing
  without BY statement 177
MERGENOBY system option 177
messages
  detail level of 179
  MERGE processing without BY statement 177
  news file for writing to SAS log 180
  printing to SAS log, all vs. top-level 219
missing values
  character to print for numeric values 178
  MISSING= system option 178
misspelled names
  autocorrection of 72
MODEL procedure
  output model type 94
MSGLEVEL= system option 179
MULTENVAPPL system option 180

N
NEWS= system option 180
notes
  writing to SAS log 181
NOTES system option 181
NUMBER system option 182
numeric data
  character to print for missing values 178
  invalid 161

O
OBS= system option 183
  comparisons 184
  data set with deleted observations 188
  details 184
  examples 184
  WHERE processing with 185
  observations
    compressing on output 99
    error messages, number printed 132
    starting at a specific 137
    stopping processing 183
ODS output
  browser for 149
ODS styles
  in graphs stored as GRSEG catalog entries 148
online training courses 270
optimization
  during code compilation 87
OPTION= option
  PROC OPTIONS statement 305
OPTIONS procedure 301
  display settings for a group of options 310
  overview 301
  results 315
  syntax 302
OPTIONS statement
  specifying system options 4
OPTLOAD procedure 321
  overview 321
  syntax 321
task table 322
OPTSAVE procedure 327
overview 327
syntax 327
task table 328
orientation, for printing 191
ORIENTATION= system option 191
out-of-resource conditions 91
OUT= option
PROC OPTSAVE statement 328
output
aligning 87
bin specification 197
collating 97
compressing 99
default form for printing 147
delimiting page breaks 146
displaying SAS/GRAPH output in
GRAPH window 148
formatting characters 145
left margin 169
overprinting error messages 194
page size 196
right margin 222
skipping lines 234
spooling 241
top margin 269
output data sets
error detection 109
output model type 94
Output window
invoking 114
maximum number of rows 116
suppressing 156
overprinting error messages 194
OVP system option 194

P
page breaks
delimiting 146
page buffers
for catalogs 86
page layout
for PDF documents 209
page numbers
printing in title line of each page 182
resetting 195
page size 196
page viewing mode 210
PAGEBREAKINITIAL system option 194
PAGENO= system option 195
PAGESIZE= system option 196
paging control buttons
SVG documents 248
task table 322
OPTSAVE procedure 327
overview 327
syntax 327
task table 328
orientation, for printing 191
ORIENTATION= system option 191
out-of-resource conditions 91
OUT= option
PROC OPTSAVE statement 328
output
aligning 87
bin specification 197
collating 97
compressing 99
default form for printing 147
delimiting page breaks 146
displaying SAS/GRAPH output in
GRAPH window 148
formatting characters 145
left margin 169
overprinting error messages 194
page size 196
right margin 222
skipping lines 234
spooling 241
top margin 269
output data sets
error detection 109
output model type 94
Output window
invoking 114
maximum number of rows 116
suppressing 156
overprinting error messages 194
OVP system option 194

P
page breaks
delimiting 146
page buffers
for catalogs 86
page layout
for PDF documents 209
page numbers
printing in title line of each page 182
resetting 195
page size 196
page viewing mode 210
PAGEBREAKINITIAL system option 194
PAGENO= system option 195
PAGESIZE= system option 196
paging control buttons
SVG documents 248

paper orientation 191
paper size 198
paper type 200
PAPERDEST= system option 197
PAPERSIZE= system option 198
PAPERSOURCE= system option 199
PAPERTYPE= system option 200
parameter strings
passing to external programs 201
PARM= system option 201
PARMCARDS statement
file reference to open for 201
PARMCARDS= system option 201
passwords
PDF documents 211
PDF documents
assembly of 203
changing the content of 205
copying 206
modifying comments 204
page layout for 209
page viewing mode 210
passwords for 211
printing permissions for 213
resolution for printing 212
screen readers for visually impaired 202
PDF forms
filling in 207
PDFACCESS system option 202
PDFASSEMBLY system option 203
PDFCOMMENT system option 204
PDFCONTENT system option 205
PDFCOPY system option 206
PDFFILLIN system option 207
PDFPAGELAYOUT= system option 209
PDFPAGEVIEW= system option 210
PDFPASSWORD= system option 211
PDFPRINT= system option 212
PDFSECURITY= system option 213
permissions
printing PDF documents 213
port number
for HTTP server for remote browsing 152
for remote browser client 152
HTTP server for remote browsing 153
portrait orientation 191
PRIMARYPROVIDERDOMAIN= system option 215
PRINTERPATH 217
PRINTERPATH= system option 217
printers
binding edge 75
font for default printer 263
for Universal Printing 217
paper size  198
printing
  bin specification  197
  color printing  98
  default form for  147
  duplexing controls  120
  initializing SAS procedure output files  218
  name of paper bin  199
  number of copies  101
  overprinting error messages  194
  page numbers in title line of each page  182
  page orientation  191
  paper size  198
  paper type  200
  PDF documents  212, 213
  PRINTINIT system option  218
  PRINTMSGLIST system option  219
  PROC OPTIONS statement  302
  PROC OPTLOAD statement  322
  PROC OPTSAVE statement  328
  procedure output
    aligning  87
    linesize  170
  procedure output files
    initializing  218
  procedures
    using labels with variables  163
  Program Editor
    autosave file  74
    maximum characters in a line  117
  Program Editor window
    invoking  114
    suppressing  156
  PS= system option  196
  punctuating numbers  305

Q
  QUOTELENMAX system option  220

R
  R language
    interface to SAS  223
  reading variable-sized record input  230
  records
    stopping processing  183
  registry
    loading system options from  321
  saving system option settings in  327
  remote browsing
    highest port number for HTTP server  152
    lowest port number for HTTP server  153
  remote help browser  151
  remote help client
    port number  152
  remote SAS sessions  112
  RENAME statement
    error detection for output data sets  109
  RENAME= data set option
    error detection for input data sets  108
  RENAME= DATA step option
    error detection for output data sets  109
  REPLACE system option  220
  resetting system option values  16, 23
  resolution
    for printing PDF documents  212
  restricted options  6
  Results window
    invoking  114
  return codes  219
  REUSE= system option  221
  RIGHTMARGIN= system option  222
  RLANG system option  223
  RSASUSER system option  224

S
  S= system option  225
  S2= system option  227
  S2V= system option  230
  SAS data sets
    naming  280
  SAS data views
    naming  280
  SAS invocation
    initializing Work library at  295
  SAS log
    AUTOEXEC input  121
    date and time, printing  104
    detail level of messages  179
    news file for writing messages to  180
    printing messages to, all vs. top-level  219
    writing hardware information to  103
    writing notes to  181
    writing secondary source statements to  240
    writing source statements to  240
  SAS procedure output files
    initializing  218
  SAS procedures
    using labels with variables  163
  SAS session encoding
    setting for the URLENCODE and URLDECODE functions  272
associating terminal with 265
executing statements at termination 266
SAS statements
executing at startup 158
writing to utility data set in Work data library 241
SAS windowing environment
invoking 113
syntax checking 117
SAS/AF software
suppressing windows 156
SAS/CONNECT software
remote session ability 112
SAS/GRAPH
displaying output in GRAPH window 148
terminal device driver, specifying 108
SAS/GRAPH files
compression of 271
SASHHELP library
location of 231
SASHHELP= system option 231
SASUSER library
opening for read or read-write access 224
SAS library to use as 232
SASUSER= system option 232
screen readers
PDF documents for visually impaired 202
search order
format catalogs 139
secondary source statements
writing to SAS log 240
SEQ= system option 233
sequence field
length of numeric portion 233
SETINIT system option 234
SHORT option
PROC OPTIONS statement 304
site license information
altering 234
SKIP= system option 234
skipping lines 234
SOLUTIONS menu
including in SAS windows 235
SOLUTIONS system option 235
sort order
verifying user-specified 238
SORT procedure
error messages 80
memory for 237
removing duplicate variables 235
verifying user-specified sort order 238
SORTDUP= system option 235
SORTEQUALS 236
SORTEQUALS system option 236
SORTSIZE= system option 237
SORTVALIDATE system option 238
source lines
as card images 84
source statements
length of 225, 227
writing secondary statements to SAS log 240
writing to SAS log 240
SOURCE system option 240
SOURCE2 system option 240
special characters not on keyboard 88
SPOOL system option 241
starting position
for reading variable-sized record input 230
STARTLIB system option 242
startup
assigning user-defined permanent librefs at 242
executing SAS statements at 158
statements
executing at termination of SAS sessions 266
length of 225, 227
writing to utility data set in Work data library 241
STEPCHKPT system option 243
STEPCHKPTLIB= system option 244
STEPRESTART system option 246
SUMSIZE= system option 247
SVG documents
paging control buttons 248
SVG output
forcing uniform scaling 251
height of viewport 249
preserving aspect ratio 251
setting the viewBox 255
title in title bar 254
value of title element in XML file 254
width of viewport 257
SVGCONTROLBUTTONS system option 248
SVGHEIGHT= system option 249
SVGPRESERVEASPECTRATIO 251
SVGPRESERVEASPECTRATIO=
 system option 251
SVGTITLE= system option 254
SVGVIEWBOX= system option 255
SVGWIDTH= system option 257
SVGX= system option 259
SVGY= system option 260
syntax checking 261
SAS windowing environment 117
SYNTAXCHECK system option 261
SYSPRINTFONT= system option 263
change using INSERT and APPEND 13
changing settings 13
comparisons 19
data set options and 19
default settings 4
determining current settings 5
determining how set 10
determining how value was set 10
determining restricted options 6
display setting for single option 317
display settings for a group 310
displaying a list 306
displaying information about 308
displaying restricted options 314
duration of settings 17
hexadecimal values for 4
information about 12
list of current settings 301
loading from registry or data sets 321
OPTIONS procedure 301
order of precedence 18
resetting default and starting values 16, 23
restricted 6
returning value of 23
saving and loading 5
saving current settings 327
short form listing 316
specifying in configuration file 4
specifying in OPTIONS statement 4
syntax 4

T
terminal device driver 108
TERMINAL system option 265
terminals
associating with SAS session 265
TERMSTMT= system option 266
TEXTURELOC= system option 267
threaded processing 268
threads
concurrent processing 101
THREADS system option 268
time stamp 104
title lines
printing page numbers in 182
titles
SVG output 254
TOOLSMENU system option 269
top margin 269
TOPMARGIN 269

TOMPARGIN= system option 269
training courses, online 270
TRAINLOC= system option 270

U
Universal Printers
compression of files 271
Universal Printing
font embedding 142
printer designation 217
uppercase
translating input to 83
UPRINTCOMPRESSION system option 271
URLENCODING= system option 272
user-defined librefs
assigning at startup 242
user-supplied holidays 160
USER= system option 273
UTF-8 encoding 272
UTILLOC= system option 274
UUID Generator Daemon
host and port 276
number of UUIDs to acquire 275
UUIDCOUNT= system option 275
UUIDGENDHOST= system option 276

V
V6CREATEUPDATE= system option 278
VALIDFMTNAME 278
VALIDFMTNAME= system option 278
VALIDMEMNAME system option 280
VALIDVARNAME= system option 282
VALUE option
PROC OPTIONS statement 306
variable names
rules for valid names 282
variable-sized record input
starting position for reading 230
variables
format not found 138
using labels with, in SAS procedures 163
VARLENCHK= system option 284
VBUFSIZE= system option 287
VIEWMENU system option 289
visual impairment
screen readers for PDF documents 202
VNFERR system option 289

W
WHERE processing
OBS= system option with 185
Work data library
  erasing checkpoint-restart data 89
  initializing at SAS invocation 295
  specifying 294
  writing SAS statements to utility data set in 241
Work files
  erasing at end of session 296
WORK= system option 294
WORKINIT system option 295
WORKTERM system option 296

X
XML files
  value of title element 254

Y
YEARCUTOFF= system option 297
  changing with GETOPTION function 25