



CHAPTER

7

SAS System Options

<i>Definition</i>	1048
<i>Syntax</i>	1048
<i>Using SAS System Options</i>	1048
<i>Default Settings</i>	1048
<i>Determining Which Settings Are in Effect</i>	1049
<i>Changing SAS System Option Settings</i>	1049
<i>How Long System Option Settings Are in Effect</i>	1050
<i>Order of Precedence</i>	1050
<i>Interaction with Data Set Options</i>	1051
<i>Comparisons</i>	1051
<i>SAS System Options by Category</i>	1052
<i>Dictionary</i>	1059
<i>APPLETLOC=</i>	1059
<i>ASYNCHIO</i>	1060
<i>BATCH</i>	1061
<i>BINDING=</i>	1061
<i>BOTTOMMARGIN=</i>	1062
<i>BUFNO=</i>	1063
<i>BUFSIZE=</i>	1064
<i>BYERR</i>	1066
<i>BYLINE</i>	1066
<i>CAPS</i>	1067
<i>CARDIMAGE</i>	1068
<i>CATCACHE=</i>	1069
<i>CBUFNO=</i>	1070
<i>CENTER</i>	1071
<i>CHARCODE</i>	1072
<i>CLEANUP</i>	1073
<i>CMDMAC</i>	1074
<i>CMPOPT</i>	1075
<i>COLLATE</i>	1076
<i>COLORPRINTING</i>	1077
<i>COMPRESS=</i>	1077
<i>CONNECTREMOTE=</i>	1079
<i>CONNECTSTATUS</i>	1080
<i>CONNECTWAIT</i>	1080
<i>CONSOLELOG=</i>	1081
<i>COPIES=</i>	1082
<i>CPUID</i>	1082
<i>DATASTMTCHK=</i>	1083
<i>DATE</i>	1084

DETAILS 1084
DEVICE= 1085
DFLANG= 1085
DKRCOND= 1087
DKROCOND= 1087
DLDMGACTION= 1088
DMR 1089
DMS 1090
DMSEXP 1090
DOCLOC= 1091
DSNFERR 1092
DUPLEX 1093
ECHOAUTO 1094
ENGINE= 1094
ERRORABEND 1095
ERRORCHECK= 1096
ERRORS= 1096
EXPLORER 1097
FIRSTOBS= 1098
FMTERR 1099
FMTSEARCH= 1100
FORMCHAR= 1101
FORMDLIM= 1102
FORMS= 1103
GISMAPS= 1103
GWINDOW 1104
HELPLOC= 1105
IMPLMAC 1105
INITCMD 1106
INITSTMT= 1108
INVALIDDATA= 1109
LABEL 1109
LAST= 1110
LEFTMARGIN= 1111
LINESIZE= 1112
MACRO 1113
MAPS= 1113
MAUTOSOURCE 1114
MERGENOBY 1115
MERROR 1115
MFILE 1116
MISSING= 1117
MLOGIC 1117
MPRINT 1118
MRECALL 1119
MSGLEVEL= 1120
MSTORED 1121
MSYMTABMAX= 1121
MULTENVAPPL 1122
MVARSIZE= 1123
NETENCRYPT 1123
NETENCRYPTALGORITHM= 1124
NETENCRYPTKEYLEN= 1125
NETMAC 1126

NEWS= 1127
NOTES 1127
NUMBER 1128
OBJECTSERVER 1128
OBS= 1129
ORIENTATION= 1131
OVP 1132
PAGENO= 1132
PAGESIZE= 1133
PAPERDEST= 1134
PAPERSIZE= 1134
PAPERSOURCE= 1135
PAPERTYPE= 1136
PARAM= 1137
PARMCARDS= 1137
PRINTERPATH= 1138
PRINTINIT 1139
PRINTMSGLIST 1140
PROBSIG= 1140
PROC 1141
REPLACE 1142
REP_MGRLOC= 1143
REUSE= 1143
RIGHTMARGIN= 1145
RSASUSER 1146
S= 1146
S2= 1148
SASAUTOS= 1149
SASCMD 1149
SASFRSCR 1150
SASHELP= 1150
SASMSTORE= 1151
SASSCRIPT= 1152
SASUSER= 1152
SEQ= 1153
ERROR 1154
SETINIT 1155
SKIP= 1155
SOLUTIONS 1156
SORTDUP= 1157
SORTSEQ= 1157
SORTSIZE= 1159
SOURCE 1160
SOURCE2 1160
SPOOL 1161
STARTLIB 1162
SUMSIZE= 1163
SYMBOLGEN 1163
SYNCHIO 1164
SYS Parm= 1165
TBUFSIZE= 1166
TCPPORTFIRST= 1166
TCPPORTLAST= 1167
TERMINAL 1168

TOPMARGIN= 1169
TRAINLOC= 1170
TRANTAB= 1171
USER= 1172
VALIDVARNAME= 1173
VNFERR 1174
WORK= 1175
WORKINIT 1176
WORKTERM 1176
YEARCUTOFF= 1177

Definition

System options are instructions that affect your SAS session. They control the way that SAS performs operations such as SAS System initialization, hardware and software interfacing, and the input, processing, and output of jobs and SAS files.

Syntax

The syntax for specifying system options in an `OPTIONS` statement is

```
OPTIONS option(s);
```

where

option

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

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

```
options nodate linesize=72;
```

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. Δ

Using SAS System Options

Default Settings

Operating Environment Information: 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. For details, see the SAS documentation for your operating environment. Δ

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 settings in system and user configuration files that were set when SAS was invoked. (See the SAS documentation for your operating environment for more information.)

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 error handling, use the GROUP= option:

```
proc options GROUP=errorhandling;
run;
```

(See the *SAS Procedures Guide* for more information.)

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. You can view this table with SAS Explorer, or you can extract information from the VOPTION table using PROC SQL.

dictionary.options SQL table

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

Changing SAS System Option Settings

At invocation, SAS provides default settings for SAS system options. There are several ways to override the default settings. Depending on the function of the system option, you can specify a setting

- *on the command line:* You can specify any SAS system option setting either on the command line or in a configuration file. 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 a configuration file:* If you use the same option settings frequently, it is usually more convenient to specify the options in a configuration file, rather than on the command line.
- *in an OPTIONS statement:* You can specify an OPTIONS statement at any time during a session except within data lines or parmcards. Settings remain in effect throughout the current program or process unless you reset them with another OPTIONS statement or change them in the SAS System Options window. You can also place an OPTIONS statement in an autoexec file.
- *in a SAS System Options window:* If you are in a windowing environment, type **options** in the toolbox to open the SAS System Options window. The SAS System Options window lists the names of the SAS system options and allows you to change their current settings. 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.

How Long System Option Settings Are in Effect

When you specify a SAS system option setting within a `DATA` or `PROC` step, the setting applies to that step and to *all subsequent steps* for the duration of the SAS session or until you reset, as shown:

```
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=`” on page 1129.

Order of Precedence

If the same system option appears in more than one place, the order of precedence from highest to lowest is

- 1 `OPTIONS` statement and SAS System Options window
- 2 autoexec file (that contains an `OPTIONS` statement)
- 3 command-line specification
- 4 configuration file specification
- 5 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. Δ

Table 7.1 on page 1051 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 7.1 Order of Precedence for SAS Execution Mode Options

Execution Mode Option	Precedence
OBJECTSERVER	Highest
DMR	2nd
INITCMD	3rd
DMS	3rd
DMSEXP	3rd
EXPLORER	3rd

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 unless they are respecified. The data set option, however, overrides the system option only for 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 5 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:

```
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.

SAS System Options by Category

Table 7.2 Categories and Descriptions of SAS System Options

Category	SAS System Option	Description
Communications: Networking and encryption	“CONNECTREMOTE=” on page 1079	Specifies the remote session ID that is used for SAS/CONNECT software
	“CONNECTSTATUS” on page 1080	Specifies whether or not to display the SAS/CONNECT transfer status window
	“CONNECTWAIT” on page 1080	Specifies whether or not to wait for a SAS/CONNECT remote submit statement (rsubmit) to complete before control returns to the local session
	“NETENCRYPT” on page 1123	Encrypts all network communications
	“NETENCRYPTALGORITHM=” on page 1124	Specifies the algorithm(s) available for the encryption of data that are passed over the network
	“NETENCRYPTKEYLEN=” on page 1125	Specifies the key size to use for the encryption of data that are passed over the network
	“NETMAC” on page 1126	Controls whether SAS uses Message Authentication Codes (MACs) to detect message corruption across a network
	“SASCMD” on page 1149	Used by the SIGNON portion of SAS/CONNECT to invoke a remote or server SAS session
	“SASFRSCR” on page 1150	Contains the fileref that is generated by the SASSCRIPT system option
	“SASSCRIPT=” on page 1152	Specifies one or more storage locations of SAS/CONNECT script files
	“TBUFSIZE=” on page 1166	Specifies the buffer size to use when you transmit data with SAS/CONNECT or SAS/SHARE software
	“TCPPORTFIRST=” on page 1166	Specifies the first TCP/IP port for SAS/CONNECT software
	“TCPPORTLAST=” on page 1167	Specifies the last TCP/IP port for SAS/CONNECT software

Environment control: Display	“CHARCODE” on page 1072	Determines whether character combinations are substituted for special characters that are not on the keyboard
	“FORMS=” on page 1103	Specifies the default form that is used for windowing output
	“SOLUTIONS” on page 1156	Specifies whether the SOLUTIONS menu choice appears in all SAS windows and whether the SOLUTIONS folder appears in the SAS Explorer window
Environment control: Error handling	“BYERR” on page 1066	Controls whether SAS generates an error message and sets the error flag when a <code>_NULL_</code> data set is used in the SORT procedure
	“CLEANUP” on page 1073	Specifies how to handle an out-of-resource condition
	“DSNFERR” on page 1092	Controls how SAS responds when a SAS data set is not found
	“ERRORABEND” on page 1095	Specifies how SAS responds to errors
	“ERRORCHECK=” on page 1096	Controls error handling in batch processing
	“ERRORS=” on page 1096	Controls the maximum number of observations for which complete error messages are printed
	“FMTERR” on page 1099	Determines whether or not SAS generates an error message when a format of a variable cannot be found
	“VNFERR” on page 1174	Controls how SAS responds when a <code>_NULL_</code> data set is used
Environment control: Files	“APPLETLOC=” on page 1059	Specifies the location of Java applets
	“DOCLOC=” on page 1091	Specifies the base location of SAS online documentation
	“FMTSEARCH=” on page 1100	Controls the order in which format catalogs are searched
	“HELPLoc=” on page 1105	Specifies the location of the text and index files for the facility that is used to view SAS help
	“NEWS=” on page 1127	Specifies a file that contains messages to be written to the SAS log
	“PARM=” on page 1137	Specifies a parameter string that is passed to an external program
	“PARMCARDS=” on page 1137	Specifies the file reference to use as the PARMCARDS file
	“REP_MGRLOC=” on page 1143	Specifies the location of the repository manager for common metadata
	“RSASUSER” on page 1146	Controls access to the SASUSER library
	“SASAUTOS=” on page 1149	Specifies the autocall macro library
“SASHELP=” on page 1150	Specifies the location of the SASHELP library	
“SASUSER=” on page 1152	Specifies the name of the SASUSER library	

	“SYSPARM=” on page 1165	Specifies a character string that can be passed to SAS programs
	“TRAINLOC=” on page 1170	Specifies the base location of SAS online training courses
	“USER=” on page 1172	Specifies the default permanent SAS data library
	“WORK=” on page 1175	Specifies the WORK data library
	“WORKINIT” on page 1176	Initializes the WORK data library
	“WORKTERM” on page 1176	Controls whether SAS erases WORK files at the termination of a SAS session
Environment control: Initialization and operation	“BATCH” on page 1061	Specifies whether batch settings for LINESIZE, OVP, PAGESIZE, and SOURCE are in effect when SAS executes
	“DMR” on page 1089	Controls the ability to invoke a remote SAS session so that you can run SAS/CONNECT software
	“DMS” on page 1090	Invokes the SAS windowing environment
	“DMSEXP” on page 1090	Invokes SAS with the Explorer, program editor, log, output, and results windows
	“EXPLORER” on page 1097	Controls whether or not you invoke SAS with only the Explorer window
	“INITCMD” on page 1106	Suppresses the Log, Output, and Program Editor windows when you enter a SAS/AF application
	“INITSTMT=” on page 1108	Specifies a SAS statement to be executed after any statements in the autoexec file and before any statements from the SYSIN= file
	“MULTENVAPPL” on page 1122	Controls whether SAS/AF, SAS/FSP, and base windowing applications use a default on an operating environment specific font selector window
Environment control: Initialization and operation	“OBJECTSERVER” on page 1128	Specifies whether or not to put the SAS session into DCOM/CORBA server mode
	“TERMINAL” on page 1168	Determines whether SAS evaluates the execution mode and, if needed, resets the option
Environment control: Language control	“DFLANG=” on page 1085	Specifies language for international date informats and formats
	“TRANTAB=” on page 1171	Specifies the translation tables that are used by various parts of SAS
Files: External files	“STARTLIB” on page 1162	Allows previous library references (librefs) to persist in a new SAS session
	“SYNCHIO” on page 1164	Specifies whether synchronous I/O is enabled
Files: SAS files	“ASYNCHIO” on page 1060	Specifies whether asynchronous I/O is enabled
	“BUFNO=” on page 1063	Specifies the number of buffers to use for SAS data sets

	“BUFSIZE=” on page 1064	Specifies the permanent buffer size for output SAS data sets
	“CATCACHE=” on page 1069	Specifies the number of SAS catalogs to keep open
	“CBUFNO=” on page 1070	Controls the number of extra page buffers to allocate for each open SAS catalog
	“COMPRESS=” on page 1077	Controls the compression of observations in output SAS data sets
	“DATASTMTCHK=” on page 1083	Prevents certain errors by controlling the SAS keywords that are allowed in the DATA statement
	“DKRCOND=” on page 1087	Controls the level of error detection for input data sets during processing of DROP=, KEEP=, and RENAME= data set options
	“DKROCOND=” on page 1087	Controls the level of error detection for output data sets during the processing of DROP=, KEEP=, and RENAME= data set options and the corresponding DATA step statements
	“DLDMGACTION=” on page 1088	Specifies what type of action to take when a SAS catalog or a SAS data set in a SAS data library is detected as damaged
	“ENGINE=” on page 1094	Specifies the default access method for SAS data libraries
	“FIRSTOBS=” on page 1098	Causes SAS to begin reading at a specified observation or record
	“_LAST_=” on page 1110	Specifies the most recently created data set
	“Mergenoby” on page 1115	Controls whether a message is issued when MERGE processing occurs without an associated BY statement
	“OBS=” on page 1129	Specifies which observation SAS processes last
	“REPLACE” on page 1142	Controls whether you can replace permanently stored SAS data sets
	“REUSE=” on page 1143	Specifies whether or not SAS reuses space when observations are added to a compressed SAS data set
	“VALIDVARNAME=” on page 1173	Controls the type of SAS variable names that can be created and processed during a SAS session
Graphics: Driver settings	“DEVICE=” on page 1085	Specifies a terminal device driver for SAS/GRAPH software
	“GISMAPS=” on page 1103	Specifies the location of the SAS data library that contains SAS/GIS-supplied US Census Tract maps
	“GWINDOW” on page 1104	Controls whether SAS displays SAS/GRAPH output in the GRAPH window of the windowing environment
	“MAPS=” on page 1113	Specifies the list of locations to search for maps

Input control: Data processing	“CARDIMAGE” on page 1068	Processes SAS source and data lines as 80-byte cards
	“INVALIDDATA=” on page 1109	Specifies the value SAS is to assign to a variable when invalid numeric data are encountered
	“PROC” on page 1141	Enables a PROC statement to invoke external programs
	“S=” on page 1146	Specifies the length of statements on each line of a source statement and the length of data on lines that follow a DATALINES statement
	“S2=” on page 1148	Specifies the length of secondary source statements
	“SEQ=” on page 1153	Specifies the length of the numeric portion of the sequence field in input source lines or datalines
	“SPOOL” on page 1161	Controls whether SAS writes SAS statements to a utility data set in the WORK data library
Input control: Data processing	“CAPS” on page 1067	Indicates whether to translate input to uppercase
	“YEARCUTOFF=” on page 1177	Specifies the first year of a 100-year span that will be used by date informats and functions to read a two-digit year
Log and procedure output control: Procedure output	“FORMDLIM=” on page 1102	Specifies a character to delimit page breaks in SAS output
	“PRINTINIT” on page 1139	Initializes the SAS print file
Log and procedure output control: SAS log	“OVP” on page 1132	Overprints output lines
	“SOURCE” on page 1160	Controls whether SAS writes source statements to the SAS log
	“SOURCE2” on page 1160	Writes secondary source statements from included files to the SAS log
Log and procedure output control: ODS printing	“BINDING=” on page 1061	Specifies the binding edge for the ODS printer destination
	“BOTTOMMARGIN=” on page 1062	Specifies the size of the margin at the bottom of the page for the ODS printer destination
	“COLLATE” on page 1076	Specifies the collation of multiple copies for output for the ODS printer destination
	“COLORPRINTING” on page 1077	Specifies color printing, if it is supported, for the ODS printer destination
	“COPIES=” on page 1082	Specifies the number of copies to make when printing to the ODS printer destination
	“DUPLEX” on page 1093	Specifies duplexing controls for the ODS printer destination
	“LEFTMARGIN=” on page 1111	Specifies the size of the margin on the left side of the page for the ODS printer destination
	“ORIENTATION=” on page 1131	Specifies the paper orientation to use when printing to the ODS printer destination

	"PRINTERPATH=" on page 1138	Specifies a printer for SAS print jobs directed to the ODS printer destination
Log and procedure output control: Procedure output	"BYLINE" on page 1066	Controls whether BY lines are printed above each BY group
	"CENTER" on page 1071	Controls alignment of SAS procedure output
	"FORMCHAR=" on page 1101	Specifies the default output formatting characters
	"LABEL" on page 1109	Determines whether SAS procedures can use labels with variables
	"PAGENO=" on page 1132	Resets the page number
	"PROBSIG=" on page 1140	Specifies the number of significant digits in <i>p</i> -values for some statistical procedures
	"SKIP=" on page 1155	Specifies the number of lines to skip at the top of each page of SAS output
Log and procedure output control: SAS log	"CONSOLELOG=" on page 1081	Specifies the location of the console log
	"CPUID" on page 1082	Specifies whether hardware information is written to the SAS log
Log and procedure output control: SAS log and procedure output	"NUMBER" on page 1128	Controls the printing of page numbers
Log and procedure output control: SAS log and procedure output	"DATE" on page 1084	Prints the date and time that the SAS session was initialized
	"DETAILS" on page 1084	Specifies whether to include additional information when files are listed in a SAS data library
	"LINESIZE=" on page 1112	Specifies the line size of SAS procedure output
	"MISSING=" on page 1117	Specifies the character to print for missing numeric values
	"PAGESIZE=" on page 1133	Specifies the number of lines that compose a page of SAS output
Log and procedure output control: SAS log	"ECHOAUTO" on page 1094	Controls whether autoexec code in an input file is echoed to the log
	"MSGLEVEL=" on page 1120	Controls the level of detail in messages that are written to the SAS log
	"NOTES" on page 1127	Writes notes to the SAS log
Macro: SAS macro	"CMDMAC" on page 1074	Determines whether the macro processor recognizes a command-style macro invocation
	"IMPLMAC" on page 1105	Controls whether SAS allows statement-style macro calls
	"MACRO" on page 1113	Specifies whether the SAS macro language is available
	"MAUTOSOURCE" on page 1114	Determines whether the macro autocall feature is available

	"MERROR" on page 1115	Controls whether SAS issues a warning message when a macro-like name does not match a macro keyword
	"MFILE" on page 1116	Specifies whether MPRINT output is directed to an external file
	"MLOGIC" on page 1117	Controls whether SAS traces execution of the macro language processor
	"MPRINT" on page 1118	Displays SAS statements that are generated by macro execution
	"MRECALL" on page 1119	Controls whether SAS searches the autocall libraries for a file that was not found during an earlier search
	"MSTORED" on page 1121	Determines whether the macro facility searches a specific catalog for a stored, compiled macro
	"MSYMTABMAX=" on page 1121	Specifies the maximum amount of memory that is available to macro variable symbol tables
	"MVARSIZE=" on page 1123	Specifies the maximum size for macro variables that are stored in memory
	"SASMSTORE=" on page 1151	Specifies the libref of a SAS data library that contains a catalog of stored, compiled SAS macros
	"SERROR" on page 1154	Controls whether SAS issues a warning message when a defined macro variable reference does not match a macro variable
	"SYMBOLGEN" on page 1163	Controls whether the results of resolving macro variable references are written to the SAS log
SAS log and procedure output control: ODS printing	"PAPERDEST=" on page 1134	Specifies the bin to receive printed output for the ODS printer destination
	"PAPERSIZE=" on page 1134	Specifies the paper size to use when printing to the ODS printer destination
	"PAPERSOURCE=" on page 1135	Specifies the paper bin to use for printing to the ODS printer destination
	"PAPERTYPE=" on page 1136	Specifies the type of paper to use for printing to the ODS printer destination
	"RIGHTMARGIN=" on page 1145	Specifies the size of the margin at the right side of the page for printed output directed to the ODS printer destination
	"TOPMARGIN=" on page 1169	Specifies the size of the margin at the top of the page for the ODS printer destination
SAS log and procedure output: SAS log	"PRINTMSGLIST" on page 1140	Controls the printing of extended lists of messages to the SAS log
Sort: Procedure options	"SORTDUP=" on page 1157	Controls the SORT procedure's application of the NODUP option to physical or logical records
	"SORTSEQ=" on page 1157	Specifies which collating sequence the SORT procedure is to use

	“SORTSIZE=” on page 1159	Specifies the amount of memory that is available to the SORT procedure
System administration: Installation	“SETINIT” on page 1155	Controls whether site license information can be altered
System administration: Memory	“SUMSIZE=” on page 1163	Specifies a limit on the amount of memory that is available for data summarization procedures when class variables are active
System administration: Performance	“CMPOPT” on page 1075	Controls whether SAS language compiler optimization is in effect

Dictionary

APPLETLOC=

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= ENVFILES

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.

Examples

Some examples of the *base-URL* are

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

ASYNCHIO

Specifies whether asynchronous I/O is enabled

Valid in: configuration file, SAS invocation

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

ASYNCHIO | NOASYNCHIO

Syntax Description

ASYNCHIO

allows other logical SAS tasks to execute (if any are ready) while the I/O is being done. This might improve system performance.

NOASYNCHIO

causes data set I/O to wait for completion.

Details

If the ASYNCHIO option is not specified on the SAS invocation line or in the configuration file, its default value depends on the runmode with which SAS is invoked:

- If SAS is invoked in DMR mode, LINE mode, or BATCH mode, the value is ASYNCHIO (NOSYNCHIO).
- If SAS is invoked in DMS mode (FULLSCREEN) or OBJECTSERVER mode, the value is NOASYNCHIO (SYNCHIO).

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings for some SAS system options may vary both by operating environment and by site. Option values may also vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. Δ

Comparisons

The ASYNCHIO system option is equivalent to the NOSYNCHIO system option. NOASYNCHIO is equivalent to SYNCHIO.

See Also

System Options:
 “SYNCHIO” on page 1164

BATCH

Specifies whether batch settings for **LINESIZE**, **OVP**, **PAGESIZE**, and **SOURCE** are in effect when SAS executes

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

BATCH | NOBATCH

Syntax Description

BATCH

specifies that SAS use the batch settings of **LINESIZE=**, **OVP**, **PAGESIZE=**, and **SOURCE**. At the start of an interactive SAS session, you can use the **BATCH** setting to simulate the behavior of the system in batch mode.

NOBATCH

specifies that SAS not use the batch settings for **LINESIZE=**, **OVP**, **PAGESIZE=**, and **SOURCE**. While in batch mode, you can specify **NOBATCH** to use the default (nonbatch) settings for the options **LINESIZE=**, **OVP**, **PAGESIZE=**, and **SOURCE**.

Details

The setting of the **BATCH** option does not specify the method of operation. **BATCH** only sets the appropriate batch settings for a collection of options that are in effect when SAS executes.

Operating Environment Information: The default setting for **BATCH** depends on your operating environment and the SAS method of operation. Δ

BINDING=

Specifies the binding edge for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

BINDING=DEFAULTEDGE | LONGEDGE | SHORTEGE

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 | SHORTEGE

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.

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

See Also

System Option:

“DUPLEX” on page 1093

BOTTOMMARGIN=

Specifies the size of the margin at the bottom of the page for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

BOTTOMMARGIN=*margin-size* | “*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.

Interactions: Changing the value of this option may 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.

Default: inches

Requirement: When you specify margin-unit, enclose the entire option value in double quotation marks.

Details

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

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. Δ

For additional information on declaring an ODS printer destination see Chapter 3, ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

See Also

System Options:

“LEFTMARGIN=” on page 1111

“RIGHTMARGIN=” on page 1145

“TOPMARGIN=” on page 1169

BUFNO=

Specifies the number of buffers to use for SAS data sets

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

BUFNO=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the number of buffers in multiples of 1, 1,000, 1,000,000, and 1,000,000,000, respectively.

MIN

sets the number of buffers to use for the SAS data set to 0, and requires SAS to use a default value.

MAX

sets the number of buffers to use for the SAS data set to the largest signed, 4-byte integer that is representable in your operating environment.

CAUTION:

The recommended maximum setting for this option is 10. △

hex

specifies the number of buffers to use for the SAS data set as a hexadecimal number. This number must be followed by an X.

Details

The number of buffers is not a permanent attribute of the data set, and it is valid only for the current SAS session or job. BUFNO= applies to SAS data sets opened for input, output, or update. If the number of buffers is greater than 0 when a SAS data set is opened, that number is used as the default value for the BUFNO= data set option. If the number of buffers is 0, SAS uses operating environment default values.

Note: Using BUFNO= can speed up execution time by limiting the number of input/output operations that are required for a particular SAS data set. The improvement in execution time, however, comes at the expense of increased memory consumption. △

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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.cm △

Comparisons

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

See Also

Data Set Option:

“BUFNO=” on page 9

System Option:

“BUFSIZE=” on page 1064

BUFSIZE=

Specifies the permanent buffer size for output SAS data sets

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

BUFSIZE=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n | *nK* | *nM* | *nG*

specifies the permanent buffer size in bytes, kilobytes, megabytes, or gigabytes respectively.

MIN

sets the permanent buffer size to the smallest, signed, 4-byte integer representable in your operating environment. The BUFSIZE= option is defined as a signed integer so that negative values can be supplied for internal testing purposes. BUFSIZE=MIN sets the permanent buffer size to the smallest signed integer representable. That is, this value sets the permanent buffer size to the largest negative integer that is representable in your operating environment. This may cause unexpected results.

CAUTION:

This setting should be avoided. Use BUFSIZE=0 to reset BUFSIZE= to the default value in your operating environment. Δ

MAX

sets the permanent buffer size to the largest, signed, 4-byte integer that is representable in your operating environment.

hex

specifies the permanent buffer size as a hexadecimal number. This number must be followed by an X.

Details

If the number of bytes is greater than 0 when a SAS data set is created, that number is used as the default value for the BUFSIZE= data set option. If the BUFSIZE data set option is omitted and the number of bytes for the BUFSIZE= system option is 0, SAS chooses an operating environment default value that is optimal for the SAS data set.

Note: Using BUFSIZE= can speed up execution time by limiting the number of input/output operations that are required for a particular SAS data set. The improvement in execution time, however, comes at the expense of increased memory consumption. Δ

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. Δ

Comparisons

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

See Also

Data Set Option:
 “BUFSIZE=” on page 10
 System Option:
 “BUFNO=” on page 1063

BYERR

Controls whether SAS generates an error message and sets the error flag when a `_NULL_` data set is used in the SORT procedure

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

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

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 or if no BY statement or BY variable is specified.

NOBYERR

specifies that SAS ignore the error message and continue processing if the SORT procedure attempts to sort a `_NULL_` data set or if no BY statement or BY variable is specified.

Comparisons

The VNFERR system option sets the error flag for a missing variable when a `_NULL_` data set is used. The DSNFERR system option controls how SAS responds when a SAS data set is not found.

See Also

System Options:
 “DSNFERR” on page 1092
 “VNFERR” on page 1174
 BY-Group Processing in *SAS Language Reference: Concepts*

BYLINE

Controls whether BY lines are printed 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

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 on the PAGEBY statement.

See Also

Statements:

#BYVAL, #BYVAR, and #BYLINE in the statement “TITLE” on page 1020

BY group processing in *SAS Language Reference: Concepts*

CAPS

Indicates whether to translate input to uppercase

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

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

Syntax

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. Δ

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

Command:

CAPS in online help

CARDIMAGE

Processes SAS 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

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.)

Quoted strings (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 quoted strings. Quoted strings can be split across lines. Other types of tokens cannot be split across lines under any circumstances. Quoted strings that are split across lines are not padded with blanks.

Operating Environment Information: CARDIMAGE is generally used in the OS/390 and CMS operating environments; NOCARDIMAGE is used in other operating environments. Δ

Examples

Consider the following DATA step:

```
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=

Specifies the number of SAS catalogs to keep open

Valid in: configuration file, SAS invocation

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

CATCACHE=*n* | MAX | MIN | *hex*

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.

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.

CAUTION:

The recommended maximum setting for this option is 10. △

MIN

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

hex

specifies the number of open-file descriptors that are kept in cache memory as a hexadecimal number. This number must be followed by an X.

Details

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

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

CBUFNO=

Controls 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

Syntax

CBUFNO=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the number of extra page buffers in terms of bytes, kilobytes, megabytes, or gigabytes, respectively.

MIN

sets the number of extra page buffers to 0, and requires SAS to use a default value.

MAX

sets the number of extra page buffers to 20.

hex

specifies the number of extra page buffers as a hexadecimal number. This number must be followed by an X.

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 may 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 probably should not increase the value of the CBUFNO= option. This is especially true if you have increased the value of the CATCACHE= option.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. Δ

CENTER

Controls alignment of 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

Syntax

CENTER | NOCENTER

Syntax Description

CENTER

centers SAS procedure output.

NOCENTER

left aligns SAS procedure output.

CHARCODE

Determines whether character 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

Syntax

CHARCODE | NOCHARCODE

Syntax Description**CHARCODE**

allows certain character combinations to be substituted for special characters that may 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:

<i>To create:</i>	<i>Use:</i>
backquote (`)	?:
backslash (\)	?,
left brace ({)	?{(
right brace (})	?)
logical not sign (\neg or \wedge)	?=
left square bracket ([)	?<
right square bracket (])	?>
underscore (_)	?-
vertical bar ()	?/

Examples

This statement produces the output [TEST TITLE]:

```
title '?<TEST TITLE?>';
```

CLEANUP

Specifies how to handle an out-of-resource condition

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

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Syntax

CLEANUP | NOCLEANUP

Syntax Description

CLEANUP

specifies that during the entire session, SAS attempts to perform automatic, continuous clean-up of resources that are not essential for execution. Nonessential resources include those that are not visible to the user (for example, cache memory) and those 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 requester window is not displayed, and no intervention is required by the user. When CLEANUP is in effect and a disk-full condition occurs, a requester window 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 requester window appears that allows the user to choose how to proceed.

Details

This table lists the requester window choices:

Requester Window Choice	Action
Free windows	clears all windows not essential for execution.
Clear paste buffers	deletes paste buffer contents.
Deassign inactive librefs	prompts user for librefs to delete.
Delete definitions of all SAS macros and macro variables	deletes all macro definitions and variables.
Delete SAS files	allows user to select files to delete.

Requester Window Choice	Action
Clear Log window	erases Log window contents.
Clear Output window	erases Output window contents.
Clear Program Editor window	erases Program Editor window contents.
Clear source spooling/DMS recall buffers	erases recall buffers.
More items to clean up	displays a list of other resources that can be cleaned up.
Clean up everything	cleans up all other options that are shown on the requestor window. This selection only applies to the current clean-up request, not to the entire SAS session.
Continuous clean up	performs automatic, continuous clean-up. 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 clean-up request and to the remainder of the SAS session.

Operating Environment Information: Some operating environments may also include these choices in the requester window:

Requester Window Choice	Action
Execute X command	enables the user to erase files and perform other clean-up operations.
Do nothing	halts the clean-up request and returns to the SAS session. This selection only applies to the current clean-up request, not to the entire SAS session.

If an out-of-resource condition cannot be resolved, the requester window 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. Δ

CMDMAC

Determines whether the macro processor recognizes a command-style macro invocation

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

CMDMAC | NOCMDMAC

Syntax Description

CMDMAC

specifies that the macro processor examine the first word of every window command to see whether it is a command-style macro invocation.

NOCMDMAC

specifies that no check will be made for command-style macro invocations.

Details

If the MAUTOSOURCE option is in effect, the macro processor searches the autocall libraries for the word. If you specify MAUTOSOURCE and CMDMAC, you can use computer resources more efficiently by specifying NOMRECALL, which causes the macro processor to search the autocall libraries only once for each keyword that begins a command.

CMPOPT

Controls whether SAS language compiler optimization is in effect

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

Category: System administration: Performance

PROC OPTIONS GROUP= PERFORMANCE

Syntax

CMPOPT | NOCMPOPT

Syntax Description

CMPOPT

optimizes your SAS code for efficient execution. This is the default.

Tip: Set CMPOPT when you do not need extensive error messages for debugging purposes.

NOCMPOPT

does not optimize your SAS code to allow for more specific error messages during program execution.

Tip: Set NOCMPOPT when debugging is more important than maximum system efficiency.

Details

This option can assist you in debugging errors in your SAS program statements, specifically, those that you use with certain SAS procedures that perform nonlinear statistical modeling or optimization. These procedures employ 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. When the CMPOPT option is in effect, the machine code is optimized for efficient execution. However, when an error occurs during execution of the optimized code, SAS might not be able to identify exactly which operation caused the error condition.

When NOCMPOPT is in effect, the SAS compiler subsystem generates machine code that processes operations more slowly, so that SAS can print diagnostic messages that more accurately indicate which operation in the SAS statement produced the error.

COLLATE

Specifies the collation of multiple copies for output for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

COLLATE | NOCOLLATE

Syntax Description

COLLATE

specifies collating multiple copies of output, if it is supported by the printer.

NOCOLLATE

specifies not collating multiple copies of output. This is the default.

Details

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

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

See Also

System Option:
 “COPIES=” on page 1082

COLORPRINTING

Specifies color printing, if it is supported, for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

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/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment.

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

COMPRESS=

Controls the compression of observations in output SAS data sets

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

COMPRESS=YES | NO | CHAR | BINARY

Syntax Description

YES | CHAR

specifies that observations in a newly created SAS output data set are compressed (variable-length records).

Tip: SAS uses RLE (Run Length Encoding) to compress observations. This compression algorithm (RLE) is better for character data.

Alias: Y

Note: COMPRESS=CHAR is accepted by Version 7 and later versions of SAS software. △

NO

specifies that observations in a newly created SAS output data set are uncompressed (fixed-length records).

Alias: N

BINARY

specifies that observations in a newly created SAS output data set are compressed.

Tip: SAS uses RDC (Ross Data Compression) for this setting. This method is highly effective for compressing medium to large (several hundred bytes or larger) blocks of binary data (that is, 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.

Note: COMPRESS=BINARY is accepted by Version 7 and later versions of SAS software. △

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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

When COMPRESS=YES or COMPRESS=CHAR, SAS compresses the size of the data set with run-length encoding. Run-length encoding compresses the data set by reducing repeated consecutive characters to two- or three-byte representations.

When COMPRESS=BINARY, SAS uses Ross Data Compression, which combines run-length encoding and sliding-window compression to compress the data set.

Use SAS/Toolkit software to specify your own compression method.

Note: Compression of observations is not supported by all engines. △

In Version 8, data sets created with engines that were available in earlier versions of SAS, such as the TAPE and XPORT engines, are still accessed by those engines. Therefore, if compression was unable for those engines, it is also not available when you access those data sets in Version 8.

The advantages that are gained by using the COMPRESS= system option include:

- reduced storage requirements for the data set
- fewer input and output operations necessary to read from or write to the data set during processing.

The disadvantages of using the COMPRESS= system option are that

- it may not compress the file at all (and may actually make the file larger), but a message that details the amount of compression is provided

- more CPU resources are required.

Comparisons

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

PERFORMANCE NOTE: Using this option increases the CPU time for reading a data set because of the overhead of uncompressing the record. In addition, some engines do not support compression of observations. When using COMPRESS=YES and REUSE=YES system option settings, observations cannot be addressed by observation number. For additional details, see the COMPRESS= data set option in “SAS Data Set Options.”

Note that REUSE=YES takes precedence over the POINTOBS=YES data set option setting.

See Also

Data Set Options:

“COMPRESS=” on page 12

“REUSE=” on page 38

System Option:

“REUSE=” on page 1143

CONNECTREMOTE=

Specifies the remote session ID that is used for SAS/CONNECT software

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

Alias: CREMOTE=, REMOTE=

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

CONNECTREMOTE=*session-id*

Syntax Description

session-id

specifies a remote session identifier to be used by SAS/CONNECT software.

Details

To establish a remote SAS session, you must either set the CONNECTREMOTE= option prior to sign-on or provide the remote *session-id* on the SIGNON statement. In the latter case, the CONNECTREMOTE= option will be set. For more details, see the *SAS/CONNECT User's Guide*.

Operating Environment Information: Not all access methods require you to specify CONNECTREMOTE= for the remote SAS session. A valid session ID and its syntax are specific to a given operating environment. For details, see the SAS documentation for your operating environment.

Although the syntax is generally consistent with the command-line syntax of your operating environment, it may include additional or alternate punctuation. △

CONNECTSTATUS

Specifies whether or not to display the SAS/CONNECT transfer status window

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

Alias: CSTATUS | NOCSTATUS

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

CONNECTSTATUS | NOCONNECTSTATUS

Syntax Description

CONNECTSTATUS

displays a window that shows the current status (number of bytes, percent complete) of a SAS/CONNECT transfer (upload or download).

NOCONNECTSTATUS

specifies that the status window not be displayed.

Details

For more information, see *SAS/CONNECT Software: Usage and Reference*.

CONNECTWAIT

Specifies whether or not to wait for a SAS/CONNECT remote submit statement (rsubmit) to complete before control returns to the local session

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

Alias: CWAIT | NOCWAIT

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

CONNECTWAIT | NOCONNECTWAIT

Syntax Description

CONNECTWAIT

(default) specifies that SAS wait for the SAS/CONNECT remote submit statement to complete before control returns to the local session; that is, the remote submit is executed serially.

NOCONNECTWAIT

specifies that SAS not wait for the SAS/CONNECT remote submit statement to complete before control returns to the local session; that is, the remote submit is executed asynchronously.

Details

CONNECTWAIT controls whether the rsubmit is synchronous or asynchronous. With asynchronous processing (NOCONNECTWAIT), you can start a remote submit and continue local processing or continue remote processing to other operating environments. For more information, see *SAS/CONNECT Software: Usage and Reference*.

CONSOLELOG=

Specifies the location of the console log

Valid in: configuration file, SAS invocation

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP= LOGCONTROL

Syntax

CONSOLELOG=*destination*

Syntax Description

destination

specifies either an external file or a device, depending on your operating environment, for writing messages when an error is generated in the SAS kernel.

Operating Environment Information: A valid destination 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 may include additional or

alternate punctuation. For details, see the SAS documentation for your operating environment. Δ

COPIES=

Specifies the number of copies to make when printing to the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

COPIES=*n*

Syntax Description

n
specifies the number of copies.

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. Δ

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

See Also

System Option:
"COLLATE" on page 1076

CPUID

Specifies whether hardware information is written to the SAS log

Valid in: configuration file, SAS invocation

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP= LOGCONTROL

Syntax

CPUID | NOCPUID

Syntax Description

CPUID

specifies that a note that contains the CPU identification number is printed at the top of the SAS log after the licensing information.

NOCPUID

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

DATASTMTCHK=

Prevents certain errors by controlling the SAS keywords that are allowed in the DATA statement

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

DATASTMTCHK=COREKEYWORDS | ALLKEYWORDS | NONE

Syntax Description

COREKEYWORDS

(default) prohibits certain words as one-level SAS data set names in the DATA statement. They can appear as two-level names, however. When you specify COREKEYWORDS, the keywords that cannot appear as one-level SAS data set names are

MERGE
RETAIN
SET
UPDATE.

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

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

It is possible to wipe out unintentionally an input data set when you omit a semicolon on the DATA statement. If the next statement is SET, MERGE, or UPDATE, the

original data set is overwritten. Different, but significant, problems arise when the next statement is RETAIN. DATASTMTCHK= enables you to protect yourself against overwriting the input data set.

DATE

Prints the date and time that the SAS session was initialized

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

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

Syntax

DATE | NODATE

Syntax Description

DATE

specifies that the date and the time that the SAS job began are printed at the top of each page of the SAS log and any print file that is created by SAS.

NODATE

specifies that the date and the time are not printed.

DETAILS

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

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

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

Syntax

DETAILS | NODETAILS

Syntax Description

DETAILS

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

DEVICE=

Specifies a terminal device driver for SAS/GRAPH software

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

Alias: DEV=

Category: Graphics: Driver settings

PROC OPTIONS GROUP= GRAPHICS

Syntax

DEVICE=*device-driver-specification*

Syntax Description***device-driver-specification***

specifies the name of a terminal device driver.

Details

If you omit the device-driver name, you are prompted to enter a driver name when you execute a procedure that produces graphics.

Operating Environment Information: Valid device-driver names depend on your operating environment. For a list of valid device-driver names, refer to the SAS documentation for your operating environment.

The syntax that is shown applies to the OPTIONS statement. However, when you specify DEVICE= either on the command-line or in a configuration file, the syntax is specific to your operating environment and may include additional or alternate punctuation. △

DFLANG=

Specifies language for international date informats and formats

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

Category: Environment control: Language control

PROC OPTIONS GROUP= LANGUAGECONTROL

Syntax

DFLANG=' *language*'

Syntax Description

'*language*'

specifies the language for international date informats and formats. The default value is English.

These are valid values for *language*:

- Afrikaans
- Catalan
- Croatian
- Czech
- Danish
- Dutch
- English
- Finnish
- French
- German
- Hungarian
- Italian
- Macedonian
- Norwegian
- Polish
- Portuguese
- Russian (Host-specific)
- Slovenian
- Spanish
- Swedish
- Swiss-French
- Swiss-German.

Details

You can change the value during a SAS session, but you can use only one language at a time. The values for *language* are not case-sensitive.

See Also

The section on International Formats and Informats in *SAS Language Reference: Concepts*

DKRICOND=

Controls the level of error detection for input data sets during processing of DROP=, KEEP=, and RENAME= data set options

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

DKRICOND=ERROR | WARN | WARNING | NOWARN | NOWARNING

Syntax Description

ERROR

sets the error flag and prints error messages if a variable on a KEEP=, DROP=, or RENAME= data set option or statement does not exist in the data set.

WARN | WARNING

prints warning messages only.

NOWARN | NOWARNING

does not print warning messages.

Examples

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 Option:

“DKROCOND=” on page 1087

DKROCOND=

Controls the level of error detection for output data sets during the processing of DROP=, KEEP=, and RENAME= data set options and the corresponding DATA step statements

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

DKROCOND=ERROR | WARN | WARNING | NOWARN | NOWARNING

Syntax Description

ERROR

sets the error flag and prints error messages if a variable on a KEEP=, DROP=, or RENAME= option or statement does not exist in the data set.

WARN | WARNING

prints warning messages only.

NOWARN | NOWARNING

does not print warning messages.

Examples

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

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

See Also

System Option:

“DKRCOND=” on page 1087

DLDMGACTION=

Specifies what type of action to take when a SAS catalog or a SAS data set in a SAS data library is detected as damaged

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

DLDMGACTION=FAIL | ABORT | REPAIR | 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 aborts the SAS session.

REPAIR

For catalogs, automatically deletes catalog entries for which an error occurs during the repair process. For data sets, automatically repairs and rebuilds indexes and integrity constraints. It issues a warning message to the log. This is the default for interactive mode.

PROMPT

displays a requester window that lets you select FAIL, ABORT, or REPAIR processing for the damaged catalog, data set, or library.

DMR

Controls the ability to invoke a remote SAS session so that you can run SAS/CONNECT software

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

DMR

Syntax Description

DMR

enables you to invoke a remote SAS session in order to run SAS/CONNECT software.

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 following SAS execution mode invocation option has precedence over this option:

- OBJECTSERVER

DMR overrides all other SAS execution mode invocation options. See “Order of Precedence” on page 1050 for more information on invocation option precedence.

See Also

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

DMS

Invokes the SAS windowing environment

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

DMS

Syntax Description

DMS

invokes SAS windowing environment and displays the Log, Program Editor, and Output windows. By default, the EXPLORER system option is in effect.

Details

The following SAS execution mode invocation options have precedence over this option:

- DMR
- OBJECTSERVER.

DMS has the same precedence as all other SAS execution mode invocation options. If you specify DMR with another execution mode invocation option of equal precedence, SAS uses only the last option listed. See “Order of Precedence” on page 1050 for more information on invocation option precedence.

See Also

System Options:

“DMSEXP” on page 1090

“EXPLORER” on page 1097

DMSEXP

Invokes SAS with the Explorer, program editor, log, output, and results windows

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

DMSEXP

Syntax Description

DMSEXP

invokes SAS with the Explorer, program editor, log, output, and results windows active.

Details

The following SAS execution mode invocation options have precedence over this option:

- DMR
- OBJECTSERVER.

DMSEXP has the same precedence as all other SAS execution mode invocation options. 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 1050 for more information on invocation option precedence.

See Also

System Options:

“DMS” on page 1090

“EXPLORER” on page 1097

DOCLOC=

Specifies the base location of SAS online documentation

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

DOCLOC=*base-URL*

Syntax Description

“*base-URL*”

specifies the address where the SAS online documentation is located. The maximum address length is 256 characters.

Details

The `DOCLOC=` system option specifies the base location (typically a URL) of SAS online documentation. This online documentation is typically accessed from an intranet server or a local CD-ROM.

Examples

Some examples of the *base-URL* are

- `"file:///e:\onlindoc"`
- `"http://server.abc.com/SAS/sasdoc"`

DSNFERR

Controls how SAS responds when a SAS data set is not found

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

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

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.

Comparisons

- `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” on page 1066

“VNFERR” on page 1174

DUPLEX

Specifies duplexing controls for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

DUPLEX | NODUPLEX

Syntax Description

DUPLEX

specifies that duplexing is performed.

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 duplexing is not performed. This is the default.

Details

Note that duplex printing can be used only on printers that support duplex output.

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings for some SAS system options may vary both by operating environment and by site. Option values may also vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*.

See Also

System Option:

“BINDING=” on page 1061

ECHOAUTO

Controls whether autoexec code in an input file is echoed to the log

Valid in: configuration file, SAS invocation

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP= LOGCONTROL

Syntax

ECHOAUTO | NOECHOAUTO

Syntax Description

ECHOAUTO

specifies that SAS source lines that are read from the autoexec file be printed in the SAS log.

NOECHOAUTO

specifies that SAS source lines that are read from the autoexec file not be printed 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.

ENGINE=

Specifies the default access method for SAS data libraries

Valid in: configuration file, SAS invocation

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

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 data library. The default engine is used when a SAS data 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.

△

See Also

“ENGINES” in *SAS Language Reference: Concepts*

ERRORABEND

Specifies how SAS responds to errors

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

Alias: ERRABEND | NOERRABEND

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Syntax

ERRORABEND | NOERRORABEND

Syntax Description

ERRORABEND

specifies that SAS abend for most errors (including syntax errors) that would normally cause it to issue an error message, set OBS=0, and go into syntax-check mode (in batch mode, only).

Tips: 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 abending. 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 (in batch mode, only).

ERRORCHECK=

Controls error handling in batch processing

Valid in: configuration file, SAS invocation

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Syntax

ERRORCHECK=NORMAL | STRICT

Syntax Description**NORMAL**

does not place the SAS job 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 job or session does not abend when a %INCLUDE statement fails due to a non-existent file.

STRICT

places the SAS job 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 job or session abends when a %INCLUDE statement fails due to a non-existent file.

ERRORS=

Controls the maximum number of observations for which complete error messages are printed

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

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Syntax

ERRORS=*n* | *n*K | *n*M | *n*G | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the maximum number of observations for which error messages are printed in terms of bytes, kilobytes, megabytes, or gigabytes, respectively.

MIN

sets the minimum number of observations for which error messages are printed to 0.

MAX

sets the maximum number of observations for which error messages are printed to the largest signed, 4-byte integer representable in your operating environment.

hex

specifies the maximum number of observations for which error messages are printed as a hexadecimal number. This number must be followed by an X.

Details

If data errors are detected in more than *n* observations, processing continues, but error messages do not print for the additional errors.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

EXPLORER

Controls whether or not you invoke SAS with only the Explorer window

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

EXPLORER

Syntax Description

EXPLORER

specifies that the SAS session be invoked with only the Explorer window.

Details

The following SAS execution mode invocation options have precedence over this option:

- DMR
- OBJECTSERVER.

EXPLORER has the same precedence as all other SAS execution mode invocation options. 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 1050 for more information on invocation option precedence.

See Also

System Options:

“DMS” on page 1090

“DMSEXP” on page 1090

FIRSTOBS=

Causes SAS to begin reading at a specified observation or record

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

FIRSTOBS=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the first observation or record that SAS reads in terms of bytes, kilobytes, megabytes, or gigabytes, respectively.

MIN

sets the first observation or record that SAS reads to 1.

MAX

sets the first observation or record that SAS reads as the largest signed, 4-byte integer representable in your operating environment.

hex

specifies the first observation or record that SAS reads as a hexadecimal number. This number must be followed by an X.

Details

FIRSTOBS must be set to 1 in order to use a WHERE statement or a WHERE= data set option. By default, FIRSTOBS=1.

CAUTION:

Using the FIRSTOBS= system option determines the first observation that is read for all steps for the duration of your current SAS session or until you change the setting. To affect any single file, you can use the FIRSTOBS= data set option. △

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.

Examples

If you specify FIRSTOBS=50, SAS reads 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 eleventh observation in the data sets OLD, A, and B:

```
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.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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

Data Set Option:

“FIRSTOBS=” on page 18

Statement:

“INFILE” on page 857

System Option:

“OBS=” on page 1129

FMTERR

Determines whether or not SAS generates an error message when a format of a variable cannot be found

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

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

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 Option:

“FMTSEARCH=” on page 1100

FMTSEARCH=

Controls 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= ENVFILES

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 *libref* can be either *libref* or *libref.catalog*. If only the *libref* is given, SAS assumes that FORMATS is the catalog name.

Details

The WORK.FORMATS catalog is always searched first, and the LIBRARY.FORMATS catalog is searched next, unless one of them appears in the FMTSEARCH= list.

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 item is ignored and searching continues.

Examples

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.

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.

See Also

System Option:

“FMterr” on page 1099

FORMCHAR=

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

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.

Details

Formatting characters are used to construct tabular output outlines and dividers for various procedures, such as the CALENDAR, FREQ, 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 hex constant with this option, SAS interprets the value of the hex constant as appropriate for your operating system.

See Also

For further information about how individual procedures use formatting characters, see the *SAS Procedures Guide*.

FORMDLIM=

Specifies a character to delimit page breaks in SAS 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

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:

```
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 dashes (- -) where normally a page break would occur:

```
options formdlim='-';
```

When a new page is to begin, SAS skips a single line, writes a line consisting of the dashes 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.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

FORMS=

Specifies the default form that is used for windowing output

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

Category: Environment control: Display

PROC OPTIONS GROUP= ENVDISPLAY

Syntax

FORMS=*form-name*

Syntax Description

form-name

specifies the name of the form.

Tip: To create a customized form, use the FSFORMS command in a windowing environment

Details

The FORMS= system option also customizes output from the PRINT command (when FORM= is omitted) or output from interactive windowing procedures. The default form contains settings that control various aspects of interactive windowing output, including printer selection, text body, and margins.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

GISMAPS=

Specifies the location of the SAS data library that contains SAS/GIS-supplied US Census Tract maps

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

Category: Graphics: Driver settings

PROC OPTIONS GROUP= Graphics

Syntax

GISMAPS=*library specification* | *path to library*

Syntax Description

library specification* | *path to library

specifies either a library or a physical path to a library that contains SAS/ GIS-supplied US Census Tract maps.

Operating Environment Information: The syntax shown here applies to the OPTIONS statement. 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. 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. △

See Also

System Option:

“MAPS=” on page 1113

GWINDOW

Controls whether SAS displays SAS/GRAPH output in the GRAPH window of the windowing environment

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

Category: Graphics: Driver settings

PROC OPTIONS GROUP= GRAPHICS

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 terminal has graphics capability.

NOGWINDOW

displays graphics outside of the windowing environment.

HELPLOC=

Specifies the location of the text and index files for the facility that is used to view SAS help

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

HELPLOC = (*path-1*<,*path-2*,...,*path-n*>)

Syntax Description***path***

specifies the directory path in which the file is located.

Details

The value for this option is not cumulative; the current value is the path or paths that are specified in the last occurrence of the option.

See Also

Operating Environment Information: Although the syntax for specifying a file is generally consistent with the command line syntax of your operating environment, it may include additional or alternate punctuation.

The syntax that is shown here applies to the OPTIONS statement. 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. △

IMPLMAC

Controls whether SAS allows statement-style macro calls

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

Category: Macro: SAS macro
 PROC OPTIONS GROUP= MACRO

Syntax

IMPLMAC | NOIMPLMAC

Syntax Description

IMPLMAC

causes the macro processor to examine the first word of every statement to see if that word is a statement-style macro call.

NOIMPLMAC

causes statement-style macro calls to be ignored. If the macro processor encounters a statement-style macro call when NOIMPLMAC is in effect, it treats the call as a SAS statement. The SAS compiler produces an error message if the statement is not valid or is not used correctly.

Details

The IMPLMAC system option controls whether macros that are defined as statement-style macros can be invoked with statement-style macro calls, or if the call must be a name-style macro call. These two statements illustrate statement-style and name-style macro calls, respectively:

- *name parameter-value-1 parameter-value-2;*
- *%name(parameter-value-1, parameter-value-2)*

Regardless of the setting of IMPLMAC, if the MACRO system option is in effect, you can call any macro with a name-style invocation, including those defined as statement-style macros.

PERFORMANCE NOTE: When you use IMPLMAC, processing time is increased because SAS checks every SAS statement to determine whether the first word is a macro call. When you use IMPLMAC with the MAUTOSOURCE system option, processing time increases because SAS also checks for a macro by that name in all macro autocall libraries that are defined in the SASAUTOS= option. When you use IMPLMAC and the MAUTOSOURCE system option with the MRECALL system option, processing time increases even further.

See Also

SAS Macro Language: Reference

INITCMD

Suppresses the Log, Output, and Program Editor windows when you enter a SAS/AF application

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

INITCMD "*command-1* <*DM-command-n*>"

Syntax Description

command-1

specifies any SAS command; the first command in this list must be a command that invokes a window. Some valid values are:

AFA
 AF
 ASSIST
 DESIGN
 EIS
 FORECAST
 HELP
 IMAGE
 LAB
 PHCLINICAL
 PHKINETICS
 PROJMAN
 QUERY
 RUNEIS
 SQC
 XADX.

Restriction: SAS/AF applications may not be available or licensed at all sites.

Interaction: If you specify FORECAST for *command-1*, you cannot use *DM-command-n*.

< ***DM-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 may delay or prohibit processing.

Restriction: Do not use the *DM-command-n* argument when you enter a SAS/AF 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 you enter a SAS/AF application, so that the SAS/AF

application is the first screen 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 have precedence over this option:

- DMR
- OBJECTSERVER.

INITCMD has the same precedence as all other SAS execution mode invocation options. 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 1050 for more information on invocation option precedence.

Examples

```
INITCMD "AFA c=mylib.myapp.primary.frame dsname=a.b"
INITCMD "ASSIST; FSVIEW SASUSER.CLASS"
```

INITSTMT=

Specifies a SAS statement to be executed after any statements in the autoexec file and before any statements from the SYSIN= file

Valid in: configuration file, SAS invocation

Alias: IS=

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

INITSTMT= *'statement'*

Syntax Description

'statement'

specifies any SAS statement or statements.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

Here is an example of using this option on UNIX:

```
sas -initstmt '%put you have used the initstmt; data x; x=1;
run;'
```

Here is an example of using this option on CMS:

```
sas (initstmt '%put you have used the initstmt; data x; x=1;
run;'
```

Note: The SYSIN= system option might not be supported your operating environment. See the SAS documentation for your operating environment for more information. △

INVALIDDATA=

Specifies the value SAS is to assign to a variable when invalid numeric data are encountered

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

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

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 are read with an INPUT statement or the INPUT function.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

LABEL

Determines whether SAS procedures can use labels with variables

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

Category: Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LISTCONTROL

Syntax

LABEL | NOLABEL

Syntax Description

LABEL

permits SAS procedures to use labels with variables. The LABEL system option must be in effect before the LABEL option of any procedure can be used.

NOLABEL

does not allow SAS procedures to use labels with variables. If NOLABEL is specified, the LABEL option of a procedure is ignored.

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 Option:

"LABEL=" on page 28

`_LAST_ =`

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

Syntax

`_LAST_ =SAS-data-set`

Syntax Description

SAS-data-set

specifies any SAS data set name.

Restriction: No data set options are allowed.

Restriction: You can specify the SAS data set only as a valid SAS one or two-level name, not as an actual filename enclosed in quotes.

Details

By default, SAS automatically keeps track of the most recently created SAS data set. Use the `_LAST_` system option to override the default.

`_LAST_` is not allowed with data set options.

Operating Environment Information: The syntax that is shown here applies to the `OPTIONS` statement. 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. △

LEFTMARGIN=

Specifies the size of the margin on the left side of the page for the ODS printer destination

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

Category: Log and procedure output control: ODS printing

`PROC OPTIONS GROUP=` ODSPRINT

Syntax

`LEFTMARGIN=margin-size` | "`margin-size [margin-unit]`"

Syntax Description

margin-size

specifies the size of the 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.

Interactions: Changing the value of this option may 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.

Default: inches

Requirement: When you specify `margin-unit`, enclose the entire option value in double quotation marks.

Details

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

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“BOTTOMMARGIN=” on page 1062

“RIGHTMARGIN=” on page 1145

“TOPMARGIN=” on page 1169

LINESIZE=

Specifies the line size of SAS procedure output

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

Alias: LS=

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

Syntax

LINESIZE=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the number of buffers in multiples of 1, 1,000, 1,000,000, and 1,000,000,000, respectively.

MIN

sets the line size of the SAS procedure output to 64.

MAX

sets the line size of the SAS procedure output to 256.

hex

specifies the line size of the SAS procedure output as a hexadecimal number. This number must be followed by an X.

Details

The LINESIZE= system option specifies the line size (printer line width) for the SAS log and the SAS procedure output file that are used by the DATA step and procedures.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

MACRO

Specifies whether the SAS macro language is available

Valid in: configuration file, SAS invocation

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MACRO | NOMACRO

Syntax Description

MACRO

enables SAS to recognize and process macro statements and macro variable references.

NOMACRO

prevents SAS from recognizing a macro statement or macro variable reference. The statement generally is not recognized, and an error message is issued.

See Also

SAS Macro Language: Reference

MAPS=

Specifies the list of locations to search for maps

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

Category: Graphics: Driver settings

PROC OPTIONS GROUP= GRAPHICS

Syntax

MAPS=*list-of-catalogs*

Syntax Description

list-of-catalogs

specifies either a libref or a physical path.

Operating Environment Information: The syntax shown here applies to the OPTIONS statement. 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 Option:
 “GISMAPS=” on page 1103

MAUTOSOURCE

Determines whether the macro autocall feature is available

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MAUTOSOURCE | NOMAUTOSOURCE

Syntax Description

MAUTOSOURCE

causes the macro processor to search the autocall library for a file with the requested name when a macro has been invoked but not compiled.

NOMAUTOSOURCE

prevents the macro processor from searching the autocall library when a macro is invoked.

Details

Any other required options, such as IMPLMAC for statement-style macros, must also be in effect.

See Also

System Option:
"SASAUTOS=" on page 1149

MERGENOBY

Controls whether a message 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

Syntax

MERGENOBY= NOWARN | WARN | ERROR

Syntax Description

NOWARN

Specifies that no warning/error message is issued. This is the default.

WARN

Specifies that a warning message is issued.

ERROR

Specifies that an error message is issued.

MERROR

Controls whether SAS issues a warning message when a macro-like name does not match a macro keyword

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MERROR | NOMERROR

Syntax Description

MERROR

allows SAS to issue a warning message if the macro processor cannot match a macro-like name (of the form *%name*) to an appropriate macro keyword.

NOMERROR

does not allow SAS to issue warning messages.

Details

Several conditions can cause a macro-like name to be unmatched with an appropriate macro keyword. These conditions appear when the following occur:

- a macro keyword, including a macro call, is misspelled
- a macro is called before being defined
- a string that contains a percent sign is encountered.

If your program contains a percent sign in a string that could be mistaken for a macro keyword, you can specify NOMERROR to suppress the warning messages.

See Also

SAS Macro Language: Reference

MFILE

Specifies whether MPRINT output is directed to an external file

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MFILE | NOMFILE

Syntax Description**MFILE**

routes the MPRINT output to an external file.

Restriction: The MPRINT option must be in effect to use MFILE.

NOMFILE

does not route the MPRINT output to an external file.

Details

MFILE is useful for debugging. Macro-generated code that is displayed by the MPRINT option in the SAS log during macro execution is written to the external file that is

referenced by the fileref MPRINT. If there is no MPRINT fileref, a warning message is issued and no file is written (MFILE is turned off).

See Also

System Option:

“MPRINT” on page 1118

SAS Macro Language: Reference

MISSING=

Specifies the character to print for missing numeric values

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

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

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.

Operating Environment Information: The syntax that is shown above applies to the OPTIONS statement. However, when you specify the MISSING= system option on the command line or in a configuration file, the syntax is specific to your operating environment and may include additional or alternate punctuation. For details, see the SAS documentation for your operating environment. △

MLOGIC

Controls whether SAS traces execution of the macro language processor

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MLOGIC | NOMLOGIC

Syntax Description

MLOGIC

specifies that the macro language processor trace its execution and write the trace information to the SAS log.

NOMLOGIC

specifies that the macro language processor not trace execution or write to the SAS log.

Details

Use MLOGIC to debug macros. If MLOGIC is in effect and the macro processor encounters a macro invocation, the macro processor displays messages that identify

- the beginning of macro execution
- the values of macro parameters at that point
- the execution of each macro program statement
- whether each %IF condition is true or false
- each iteration of the %DO loop
- the end of macro execution.

CAUTION:

MLOGIC can produce a great deal of output in the SAS log. Δ

See Also

SAS Macro Language: Reference

MPRINT

Displays SAS statements that are generated by macro execution

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MPRINT | NOMPRINT

Syntax Description

MPRINT

specifies that SAS statements that are generated by macro execution be displayed.

NOMPRINT

specifies that SAS statements that are generated by macro execution not be displayed.

Details

The statements are formatted with macro variable references and macro functions resolved, with each statement beginning on a new line and with one space between words.

Comparisons

The MFILE option provides a method to write MPRINT output to an external file without the MPRINT prefix printed at the beginning of each line.

See Also

SAS Macro Language: Reference

MRECALL

Controls whether SAS searches the autocall libraries for a file that was not found during an earlier search

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MRECALL | NOMRECALL

Syntax Description

MRECALL

searches the autocall libraries for an undefined macro name each time the macro is invoked.

NOMRECALL

searches the autocall libraries only once for a requested macro name.

Details

Use the MRECALL option primarily for

- debugging systems that require macros in autocall libraries.
- recovering from errors caused by an autocall to a macro that is in an unavailable library. Use MRECALL to call the macro again after making the library available.

See Also

SAS Macro Language: Reference

MSGLEVEL=

Controls 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

Syntax

MSGLEVEL= N | I

Syntax Description

N

prints notes, warnings, and error messages only. This is the default.

I

prints additional notes pertaining to index usage, merge processing, and sort utilities along with standard notes, warnings, and error messages.

Examples

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:
 - 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 to sort 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 whenever a MERGE statement would cause variables to be overwritten.
- If MSGLEVEL=I, SAS writes a message that indicates which sorting product was used.

MSTORED

Determines whether the macro facility searches a specific catalog for a stored, compiled macro

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MSTORED | NOMSTORED

Syntax Description

MSTORED

specifies that the macro facility search for stored compiled macros in the SASMACRO catalog of the SAS data library that is referenced by the SASMSTORE option.

NOMSTORED

specifies that the macro facility does not search for compiled macros.

Details

Regardless of the setting of MSTORED, the macro facility first searches for macros in the SASMACR catalog in the WORK library. If you specify MSTORED and the desired macro is not found in the WORK library, the macro facility searches the SASMACR catalog in the SAS data library that is referenced by SASMSTORE. If you specify NOMSTORED, the subsequent search is not performed.

MSYMTABMAX=

Specifies the maximum amount of memory that is available to macro variable symbol tables

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MSYMTABMAX= *n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the amount of memory in terms of bytes, kilobytes, megabytes, and gigabytes, respectively.

MIN

specifies 0, which causes all macro variables to be written to disk.

MAX

specifies the maximum amount of memory available for your operating environment.

hex

specifies the amount of memory as a hexadecimal number. This number must be followed by an X.

Details

Once the specified value is reached, additional macro variables are written out to disk.

Note: The value of MSYMTABMAX= can affect system performance. Before you specify the value for production jobs, run tests to determine the optimum value. Δ

MULTENVAPPL

Controls whether SAS/AF, SAS/FSP, and base windowing applications use a default on an operating environment specific font selector window

Valid in: configuration file, SAS invocation, OPTIONS statement

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

MULTENVAPPL | NOMULTENVAPPL

Syntax Description

MULTENVAPPL

specifies that the Select Font window is displayed in the Frame region manager when you set the font for a region title. Use the Select Font window to select different font families, styles, weights, and point sizes.

NOMULTENVAPPL

specifies that SAS/AF and SAS/FSP software display a Font Selector window that is specific to the operating environment in which the application is executing.

Details

The MULTENVAPPL system option enables SAS/AF and SAS/FSP applications, when created, to use the features of the SAS System that are commonly supported in all operating environments. Use this option to ensure portability of applications to multiple operating environments. If NOMULTENVAPPL is in effect when a SAS/AF or

SAS/FSP application is created, the application uses features of the SAS System that are more specific to your operating environment. If the application is ported to another environment, a message or a default action may occur, depending on the operating environment.

MVARSIZE=

Specifies the maximum size for macro variables that are stored in memory

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

MVARSIZE= *n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the maximum size for macro variables stored in memory in terms of bytes, kilobytes, megabytes, and gigabytes, respectively.

MIN

sets MVARSIZE to 0, which causes all macro variables to be written to disk.

MAX

specifies the maximum amount of memory available.

hex

sets the maximum macro variable size stored in memory as a hexadecimal number. This number must be followed by an X.

Details

If the size of the macro variable is larger than the maximum value that is specified, variables are written out to disk.

The value that you specify with the MVARSIZE= system option can range from 0 to the largest positive integer that is representable on your operating environment.

Operating Environment Information: The value of MVARSIZE= can affect system performance. Before you specify the value for production jobs, run tests to determine the optimum value. For further information and default settings, see the SAS documentation for your operating environment. Δ

NETENCRYPT

Encrypts all network communications

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

Alias: NETENCR | NONETENCR

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

See: SAS/CONNECT documentation, SAS/SHARE documentation

Syntax

NETENCRYPT | NONETENCRYPT

Syntax Description

NETENCRYPT

specifies that encryption is required for all network communications.

NONETENCRYPT

specifies that encryption is not required for all network communications.

See Also

System Options:

“NETENCRYPTALGORITHM=” on page 1124

“NETENCRYPTKEYLEN=” on page 1125

“NETMAC” on page 1126

NETENCRYPTALGORITHM=

Specifies the algorithm(s) available for the encryption of data that are passed over the network

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

Alias: NETENCRALG

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

See: SAS/CONNECT documentation, SAS/SHARE documentation

Syntax

NETENCRYPTALGORITHM=*algorithm* | (“*algorithm1*”, “*algorithm2*”, ...)

Syntax Description

algorithm | (“*algorithm1*”, “*algorithm2*”, ...)

identifies the algorithm(s) available for the encryption of data that are passed over the network by SAS/CONNECT or SAS/SHARE software. When you specify two or more encryption algorithms, you must enclose the specifications in parentheses. You can separate multiple specifications with a blank space. If there are embedded blanks in the algorithm name, enclose each algorithm with quotation marks.

The following are the available algorithms:

- RC2
- RC4
- DES
- Triple DES
- SAS Proprietary.

See Also

System Options:

“NETENCRYPT” on page 1123

“NETENCRYPTKEYLEN=” on page 1125

“NETMAC” on page 1126

NETENCRYPTKEYLEN=

Specifies the key size to use for the encryption of data that are passed over the network

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

Alias: NETENCRKEY

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

See: SAS/CONNECT documentation, SAS/SHARE documentation

Syntax

NETENCRYPTKEYLEN=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the key size to use for the encryption of data passed over the network by SAS/CONNECT or SAS/SHARE software in terms of bytes, kilobytes, megabytes, or gigabytes, respectively.

MIN

sets NETENCRYPTKEYLEN= to the minimum key size, 0.

MAX

sets NETENCRYPTKEYLEN= to the maximum key size, 1024.

hex

specifies the key size to use for the encryption of data that are passed over the network by SAS/CONNECT or SAS/SHARE software as a hexadecimal number. This number must be followed by an X.

See Also

System Options:

“NETENCRYPTALGORITHM=” on page 1124

“NETENCRYPT” on page 1123

“NETMAC” on page 1126

NETMAC

Controls whether SAS uses Message Authentication Codes (MACs) to detect message corruption across a network

Valid in: configuration file, SAS invocation

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

See: SAS/CONNECT documentation, SAS/SHARE documentation

Syntax

NETMAC | NONETMAC

Syntax Description**NETMAC**

specifies that SAS use MACs (a one-way hash that is computed from a message and some secret data) for network communications to detect if the message has been altered. If you specify NETMAC, SAS uses the MD5 one-way hash function to convert an arbitrarily long data stream into a hash.

NONETMAC

specifies that SAS not use MACs for network communications.

See Also

System Options:

“NETENCRYPT” on page 1123

“NETENCRYPTALGORITHM=” on page 1124

“NETENCRYPTKEYLEN=” on page 1125

NEWS=

Specifies a file that contains messages to be written to the SAS log

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

NEWS=*external-file*

Syntax Description

external-file

specifies an external file.

Operating Environment Information: 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 may include additional or alternate punctuation. For details, see the SAS documentation for your operating environment. △

NOTES

Writes notes 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

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 Institute for problem determination and resolution.

NUMBER**Controls the printing of page numbers**

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

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

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.

OBJECTSERVER**Specifies whether or not to put the SAS session into DCOM/CORBA server mode**

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

OBJECTSERVER

Syntax Description

OBJECTSERVER

specifies that the SAS session be put into DCOM/CORBA server mode.

Interaction: OBJECTSERVER overrides all other SAS execution mode invocation options. See “Order of Precedence” on page 1050 for more information on invocation option precedence.

Details

DCOM/CORBA server mode enables external clients to use all the features of the SAS System without running inside SAS.

OBS=

Specifies which observation SAS processes last

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

OBS=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n

specifies the number of the last observation to process.

If OBS=0 and the NOREPLACE option is in effect, 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 are deleted.

nK* | *nM* | *nG

specifies the number of the last observation to process in kilobytes, megabytes, and gigabytes, respectively.

MIN

sets the number of the last observation to process to 0.

MAX

specifies the maximum number of observations to process. The default value is MAX, which is the largest signed, four-byte integer that is representable in your operating environment.

hex

specifies the number of the last observation to process as a hexadecimal number. This number must be followed by an X.

Details

The OBS= system option specifies which observation from a data set or which record from a raw data file that SAS reads last. You can also use OBS= to control analysis of SAS data sets in PROC steps.

If SAS is processing a file of raw data, 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 are on a single line.

You must specify OBS=MAX to use a WHERE statement or WHERE= data set option.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

CAUTION:

Using the OBS= system option determines the last observation that is read for all steps during your current SAS session or until you change the setting. To affect any single file, you can use the OBS= data set option. △

Comparisons

An OBS= specification from either a data set option or an INFILE statement option takes precedence over the OBS= system option.

Examples

- This statement causes SAS to read only through the 50th observation of a SAS data set or the 50th record of an external file:

```
options obs=50;
```

- Check the syntax of your SAS program by specifying the following statement as the first statement in a program:

```
options obs=0 noreplace;
```

- Use the FIRSTOBS= and OBS= options together to process a set of observations from the middle of a data set. For example, this statement processes only observations 1000 through 1100:

```
options firstobs=1000 obs=1100;
```

See Also

Data Set Options:

“FIRSTOBS=” on page 18

“OBS=” on page 29

“REPLACE=” on page 37

System Option:

“FIRSTOBS=” on page 1098

ORIENTATION=

Specifies the paper orientation to use when printing to the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

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 to enable printing on paper with prepunched holes. The reverse side of the page is printed upside down.

REVERSELANDSCAPE

specifies the paper orientation as reverse landscape to enable printing on paper with prepunched holes. The reverse side of the page is printed upside down.

Details

Changing the value of this option may result in changes to the values of the portable `LINESIZE=` and `PAGESIZE=` system options.

Operating Environment Information: Most SAS system options are initialized with default settings when you invoke SAS. However, the default settings for some SAS system options vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

OVP

Overprints output lines

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

Category: Log and procedure output control: SAS log

PROC OPTIONS GROUP= LOG_LISTCONTROL

Syntax

OVP | NOOVP

Syntax Description

OVP

prints underscores beneath the word in error when SAS encounters an error in a SAS statement. This option may enable overprinting in some procedures.

NOOVP

prints dashes on the next line below the word in error when SAS encounters an error in a SAS statement.

Details

When output is displayed to a terminal, OVP is overridden and is changed to NOOVP.

PAGENO=

Resets the page number

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

Category: Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LISTCONTROL

Syntax

PAGENO=*n* | MIN | MAX | *hex*

Syntax Description

n
specifies the page number in terms of bytes.

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.

hex
specifies the page number as a hexadecimal number. This number must be followed by an X.

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.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

PAGESIZE=

Specifies the number of lines that compose a page of SAS output

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

Alias: PS=

Category: Log and procedure output control: SAS log and procedure output

PROC OPTIONS GROUP= LOG_LISTCONTROL

Syntax

PAGESIZE=*n* | *nK* | MIN | MAX | *hex*

Syntax Description

n* | *nK
specifies the number of lines that compose a page in terms of number of pages or units of 1,024 pages.

MIN
sets the number of lines that compose a page to the minimum setting, 15.

MAX
sets the number of lines that compose a page to the maximum setting, 32,767.

hex

specifies the number of lines that compose a page as a hexadecimal number. This number must be followed by an X.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. On the command line or in a configuration file, valid values and range vary with your operating environment. For details, see the SAS documentation for your operating environment. △

PAPERDEST=

Specifies the bin to receive printed output for the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

PAPERDEST=*printer-bin-name*

Syntax Description***printer-bin-name***

specifies the bin to receive printed output.

Restriction: Maximum length is 200 characters.

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination see Chapter 3, ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“PAPERSIZE=” on page 1134

“PAPERSOURCE=” on page 1135

“PAPERTYPE=” on page 1136

PAPERSIZE=

Specifies the paper size to use when printing to the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

PAPERSIZE=*paper_size_name* | ("*width_value*" <,> "*height_value*") | ('*width_value*'<,> '*height_value*') | (*width_value height_value*)

Syntax Description

paper_size_name

specifies a predefined paper size. The default is LETTER. Maximum length is 200 characters.

Default: Letter

Valid Values: Refer to the Registry Editor, or use PROC REGISTRY to obtain a listing of supported paper sizes. Additional values can be added.

Restriction: The maximum length is 200 characters.

("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, LETTER is used.

Fields specifying values for paper sizes may either be separated by blanks or commas.

Note: Changing the value of this option may result in changes to the values of the portable LINESIZE= and PAGESIZE= system options. △

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“PAPERDEST=” on page 1134

“PAPERSOURCE=” on page 1135

“PAPERTYPE=” on page 1136

PAPERSOURCE=

Specifies the paper bin to use for printing to the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODS_{PRINT}

Syntax

PAPER_{SOURCE}=*printer-bin-name*

Syntax Description

printer-bin-name

specifies the bin that sends paper to the printer.

Operating Environment Information: For instructions on how to specify a printer bin, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“PAPER_{DEST}=” on page 1134

“PAPER_{SIZE}=” on page 1134

“PAPER_{TYPE}=” on page 1136

PAPER_{TYPE}=

Specifies the type of paper to use for printing to the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODS_{PRINT}

Syntax

PAPER_{TYPE}=*paper-type-string*

Syntax Description

paper-type-string

specifies the type of paper. Maximum length is 200.

Range: Values vary by site and operating environment.

Default: Values vary by site and operating environment.

Operating Environment Information: For instructions on how to specify the type of paper, see the SAS documentation for your operating environment. There is a very large number of possible values for this option. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“PAPERDEST=” on page 1134

“PAPERSIZE=” on page 1134

“PAPERSOURCE=” on page 1135

PARM=

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

Syntax

PARM=<'>*string*<'>

Syntax Description

<'>*string*<'>

specifies a character string that contains a parameter.

Examples

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=

Specifies the file reference to use as the PARMCARDS file

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

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

PARMCARDS=*file-ref*

Syntax Description

file-ref

specifies the file reference to use as the PARMCARDS= file.

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.

Operating Environment Information: The syntax shown here applies to the OPTIONS statement. 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. △

PRINTERPATH=

Specifies a printer for SAS print jobs directed to the ODS printer destination

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

Category: Log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

PRINTERPATH=(*printer name* <*fileref*>)

Syntax Description

printer name

must be one of the printers defined in the Registry Editor under

Core ► Printing ► Printers

Quotes are required only when the *printer name* contains blanks.

fileref

is an optional fileref. If a fileref is specified, it must be defined with a FILENAME statement or external allocation. If a fileref is not specified, the default output destination defined in the Registry Editor under

Core ► Printing ► Printers ► Printer Setup ► Output

is used. Parentheses are required only when a *fileref* is specified.

Details

The following example specifies an output destination that is different from the default:

```
options PRINTERPATH=(corelab out);
filename out 'whatever';
```

Operating Environment Information: In some operating environments, such as the PC, the SYSPRINT option enables operating-environment printing and disables SAS printing by setting the PRINTERPATH option to a null string. Also, 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.

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System* △

PRINTINIT

Initializes the SAS print file

Valid in: configuration file, SAS invocation

Category: Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LISTCONTROL

Syntax

PRINTINIT | NOPRINTINIT

Syntax Description

PRINTINIT

empties the SAS output file and resets the file attributes upon initialization.

Tip: Specifying PRINTINIT causes the SAS output file to be emptied even when output is not generated.

NOPRINTINIT

preserves the existing output file if no new output is generated. This is the default.

Tip: Specifying NOPRINTINIT causes the SAS 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. Δ

PRINTMSGLIST

Controls the printing of extended lists of messages to the SAS log

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

Category: SAS log and procedure output: SAS log

PROC OPTIONS GROUP= LOGCONTROL

Syntax

PRINTMSGLIST | NOPRINTMSGLIST

Syntax Description

PRINTMSGLIST

prints the entire list of messages to the SAS log.

NOPRINTMSGLIST

prints 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. The default is set to "ON."

PROBSIG=

Specifies the number of significant digits in p -values for some statistical procedures

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

Category: Log and procedure output control: Procedure output

PROC OPTIONS GROUP= LISTCONTROL

Syntax

PROBSIG= n | MIN | MAX | hex

Syntax Description

n
specifies the number of significant digits.

MIN
sets the minimum number of significant digits to 0.

MAX
sets the maximum number of significant digits to 2.

hex
specifies the number of significant digits as hexadecimal. This number must be followed by an X.

Details

When PROBSIG=0, *p*-values are written with four decimal places and truncated at .0001.

PROBSIG=1 guarantees that *p*-values are written with at least one significant digit; that is, values greater than or equal to .000095 are written with four decimal places, but values less than .000095 are written in E-notation.

PROBSIG=2 guarantees that *p*-values are written with at least two significant digits. Values greater than .0000995 are written with five decimal places, and smaller values are written in E-notation.

Operating Environment Information: The syntax shown here applies to the OPTIONS statement. 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. Δ

PROC

Enables a PROC statement to invoke external programs

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

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

Syntax

PROC | NOPROC

Syntax Description

PROC

enables you to invoke programs that are external to SAS, such as utility programs written by other vendors, using a PROC statement.

NOPROC

does not enable you to invoke programs that are external to SAS.

Examples

To invoke the utility program UTLPGM1, which is not a part of the SAS System, issue these statements:

```
options proc;
proc utlpgm1;
run;
```

REPLACE

Controls whether you can replace permanently stored SAS data sets

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

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. This 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

System Option:
 “WORKTERM” on page 1176

Data Set Option:
 “REPLACE=” on page 37

REP_MGRLOC=

Specifies the location of the repository manager for common metadata

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

REP_MGRLOC= "*SCL-libname*" | '*SCL-libname*' |
 (*SCL-libname*) | *libref*

Syntax Description

SCL-libname

any valid SCL (SAS Component Language) libname, with the following syntax:
 (*libref* <,SAS-data-library<,engine<,options>>>). The SCL libname can be enclosed in single or double quotation marks, or in parentheses. When you specify two or more locations, you must enclose the specifications in parentheses. The maximum length for the SCL libname is 200 characters.

libref

a previously defined libref.

Details

REP_MGRLOC= specifies the location of the repository manager for common metadata. Normally, this option is a site-wide option that is set by the SAS Installation Representative.

REUSE=

Specifies whether or not 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

Syntax

REUSE=YES | NO

Syntax Description

YES

tracks free space and reuses it whenever observations are added to an existing compressed data set.

NO

does not 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. This means that 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.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. Δ

Comparisons

The REUSE= data set option overrides the REUSE= system option.

PERFORMANCE NOTE: When using COMPRESS=YES and REUSE=YES system options settings, observations cannot be addressed by observation number.

Note that REUSE=YES takes precedence over the POINTOBS=YES data set option setting.

See Also

System Option:
 "COMPRESS=" on page 1077

Data Set Options:
 "COMPRESS=" on page 12
 "REUSE=" on page 38

RIGHTMARGIN=

Specifies the size of the margin at the right side of the page for printed output directed to the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing
PROC OPTIONS GROUP= ODSPRINT

Syntax

RIGHTMARGIN=*margin-size* | "*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.

Interactions: Changing the value of this option may 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.

Default: inches

Requirement: When you specify margin-unit, enclose the entire option value in double quotation marks.

Details

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

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. Δ

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“BOTTOMMARGIN=” on page 1062

“LEFTMARGIN=” on page 1111

“TOPMARGIN=” on page 1169

RSASUSER

Controls access to the SASUSER library

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

RSASUSER | NORSASUSER

Syntax Description

RSASUSER

opens the SASUSER library in input (that is, read-only) mode.

NORSASUSER

opens the SASUSER library in update (that is, 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. For details, see the SAS documentation for your operating environment.

S=

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

Syntax

S=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the length of statements and data in terms of bytes, kilobytes, megabytes, and gigabytes, respectively.

MIN

sets the length of statements and data to 0, and requires SAS to use a default value.

MAX

sets the length of statements and data to the largest unsigned, four-byte integer that is representable in your operating environment.

hex

specifies the length of statements and data as a hexadecimal number. This number must be followed by an X.

Details

Input can be from either fixed- or variable-length records. Both fixed-length and variable-length records can be either unsequenced or sequenced. Unsequenced records do not contain sequence fields. Fixed-length sequenced records contain sequence fields at the end of each record. Variable-length sequenced records contain sequence fields at the beginning of each record.

SAS determines whether the input contains sequence numbers that are based on the value of S. If S=0 and you have fixed-length records, SAS inspects the last *n* columns (where *n* is the value of the SEQ= option) of the first sequence field, which is at the end of the first line of input. If those columns contain numeric characters, SAS assumes that the file contains sequence fields and ignores the last eight columns of each line.

If S \geq 0 or MAX and you have fixed-length records, SAS uses that value as the length of the source or data to be scanned, ignores everything beyond that length on each line, and does not look for sequence numbers.

If S=0 and you have variable-length records, SAS inspects the last *n* columns (where *n* is the value of SEQ=) of the first sequence field, which is at the beginning of the first line of input. If those columns contain numeric characters, SAS assumes the file contains sequence fields and ignores the first eight columns of each line.

If S \geq 0 or MAX and you have variable-length records, SAS uses that value as the starting column of the source or data to be scanned, ignores everything before that length on each line, and does not look for sequence numbers.

Comparisons

The S= system option operates exactly like the S2= system option except that S2= only controls input from a %INCLUDE statement, an autoexec file, or an autocall macro file.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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:

“S2=” on page 1148

“SEQ=” on page 1153

S2=

Specifies the length of secondary source statements

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

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

PROC OPTIONS GROUP= INPUTCONTROL

Syntax

S2=S | *n*

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

uses the value of *n* to compute the record length of text that comes from a %INCLUDE statement, an autoexec file, or an autocall macro file.

Comparisons

The S2= system option operates exactly like the S= system option except that the S2= option controls input from only a %INCLUDE statement, an autoexec file, or an autocall macro file.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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:

“S=” on page 1146

“SEQ=” on page 1153

SASAUTOS=

Specifies the autocall macro library

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

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

SASAUTOS= *library-specification* | (*library-specification-1*, ..., *library-specification-n*)

Syntax Description

library-specification

(*library-specification-1*, ..., *library-specification-n*)

identifies one or more aggregate storage locations, each of which can contain files or members that contain definitions of SAS macros. When you specify two or more autocall libraries, enclose the specifications in parentheses. Separate multiple specifications with either a comma or a blank space.

Operating Environment Information: 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. △

SASCMD

Used by the SIGNON portion of SAS/CONNECT to invoke a remote or server SAS session

Valid in: configuration file, SAS invocation, OPTIONS statement, OPTIONS window

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

SASCMD=" *invocation* "

Syntax Description

“invocation”

is a text string that invokes a remote SAS session.

Interaction: the text string is used by SAS/CONNECT software.

SASFRSCR

Contains the fileref that is generated by the SASSCRIPT system option

Valid as part of: SAS Component Language

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

SASFRSCR

Syntax Description

SASFRSCR

contains the fileref that is generated by the SASSCRIPT= option. This is a read-only option.

Details

The SASFRSCR system option accepts no input, but its value changes when the specification of the SASSCRIPT= system option changes. You can read the value of this option as part of an application in *SAS Component Language (SCL)* to prompt a user for the correct SAS/CONNECT sign-on script.

See Also

System Option:

“SASSCRIPT=” on page 1152

SASHELP=

Specifies the location of the SASHELP library

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

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. △

SASMSTORE=

Specifies the libref of a SAS data library that contains a catalog of stored, compiled SAS macros

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

SASMSTORE=*libref*

Syntax Description

libref

specifies the libref of a SAS data library that contains, or will contain, a catalog of stored, compiled SAS macros. This libref cannot be WORK.

See Also

System Option:

“MSTORED” on page 1121

SASSCRIPT=

Specifies one or more storage locations of SAS/CONNECT script files

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

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

SASSCRIPT=(*location-1* *location-2* ... *location-n*)

Syntax Description

'location'

specifies the name of an aggregate storage location, such as a PDS for MVS, a maclib for CMS, or a directory for directory-based systems.

Details

The SASSCRIPT= system option uses the value of *location* to generate an internal fileref that is stored as the value of the SASFRSCR= system option.

To specify one or more aggregate storage locations, enclose each location in quotation marks, separate multiple locations with blanks, and place the entire specification in parentheses.

After you specify the location of script files with SASSCRIPT=, you can do the following:

- In SAS/ASSIST software, when you use the REMOTE CONNECT item to edit a remote configuration, enter a question mark (?) in the SCRIPT field of the CREATE NEW CONFIGURATION submenu to display a list of script file names.
- In an SCL program, you can use the fileref stored as the value of SASFRSCR= when you want to build a list of available scripts for establishing a link with SAS/CONNECT software.

SASUSER=

Specifies the name of the SASUSER library

Valid in: configuration file, SAS invocation

Category: Environment control: Files
 PROC OPTIONS GROUP= ENVFILES

Syntax

SASUSER=*SAS-data-library*

Syntax Description

SAS-data-library

identifies a SAS data library.

Details

The SAS library contains a user's profile catalog. The library and catalog are created automatically by SAS; you do not have to create them explicitly.

Operating Environment Information: 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. △

SEQ=

Specifies the length of the numeric portion of the sequence field in input source lines or datalines

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

Category: Input control: Data processing
 PROC OPTIONS GROUP= INPUTCONTROL

Syntax

SEQ=*n* | MIN | MAX | *hex*

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.

hex

specifies the length as a hexadecimal. This number must be 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 file name) 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=” on page 1146

“S2=” on page 1148

SERROR

Controls whether SAS issues a warning message when a defined macro variable reference does not match a macro variable

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

SERROR | NOSERROR

Syntax Description**SERROR**

issues a warning message when a macro variable reference does not match a defined macro variable.

NOSERROR

does not issue the SAS system warning messages when a defined macro variable reference does not match a macro variable.

CAUTION:

When you develop and test programs, the SERROR setting is more useful to you than the NOSERROR setting. △

Details

Several conditions can occur that cause a macro variable reference to go unmatched with an appropriate macro variable. These conditions appear when

- the name in a macro variable reference is misspelled
- the variable is referenced before it has been defined
- the program contains an ampersand (&) that is followed by a string, without intervening blanks between the ampersand and the string.

For example, this statement uses an ampersand as the symbol for the logical operator AND with no intervening blanks:

```
if x&y then do;
```

- the program contains an ampersand (&) that is used as a literal and is enclosed in double quotation marks:

```
if buyer="Smith&Jones, Inc." then do;
```

See Also

SAS Macro Language: Reference

SETINIT

Controls whether site license information can be altered

Valid in: configuration file, SAS invocation

Category: System administration: Installation

PROC OPTIONS GROUP= INSTALL

Syntax

SETINIT | NOSETINIT

Syntax Description

SETINIT

In a windowing environment, allows the SETINIT window to be displayed so that you can make changes. In a non-windowing environment, SETINIT enables you to change license information by running the SETINIT procedure.

NOSETINIT

does not allow you to alter site license information after installation.

Details

SETINIT is set in the installation process and is not normally changed after installation.

SKIP=

Specifies the number of lines to skip at the top of each page of SAS 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

Syntax

SKIP=*n* | MIN | MAX | *hex*

Syntax Description

n

specifies the range of lines to skip from 0 to 20. (The SKIP= value does not affect the maximum number of lines printed on each page, which is controlled by the PAGESIZE= system option.)

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. This number must be followed by an X.

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 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.

SOLUTIONS

Specifies whether the SOLUTIONS menu choice appears in all SAS windows and whether the SOLUTIONS folder appears in the SAS Explorer window

Valid in: configuration file, SAS invocation

Category: Environment control: Display

PROC OPTIONS GROUP= ENVDISPLAY

Syntax

SOLUTIONS | NOSOLUTIONS

Syntax Description

SOLUTIONS

specifies that the SOLUTIONS menu choice appears in all SAS windows and that the SOLUTIONS folder appears in the SAS Explorer window tree view.

NOSOLUTIONS

specifies that the SOLUTIONS menu choice does not appear in all SAS windows and the SOLUTIONS folder does not appear in the SAS Explorer window tree view.

SORTDUP=

Controls the SORT procedure's application of the NODUP option to physical or logical records

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

Category: Sort: Procedure options

PROC OPTIONS GROUP= SORT

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.

Interactions:

The SORTDUP option is relevant only when DROP= or KEEP= data set options are specified on the input data set for the SORT procedure. The NODUP option is a SORT procedure option that removes duplicate records. For further information about how NODUP and SORTDUP interact, see the *SAS Procedures Guide*.

See Also

System Options:

“SORTSEQ=” on page 1157

“SORTSIZE=” on page 1159

The SORT Procedure in the *SAS Procedures Guide*

SORTSEQ=

Specifies which collating sequence the SORT procedure is to use

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

Category: Sort: Procedure options

PROC OPTIONS GROUP= SORT

Syntax

`SORTSEQ=collating-sequence`

Syntax Description

collating-sequence

specifies the collating-sequence that the SORT procedure is to use. Valid values can be user-supplied, or they can be one of the following:

- ASCII
- DANISH (alias NORWEGIAN)
- EBCDIC
- FINNISH
- ITALIAN
- NATIONAL
- REVERSE
- SPANISH
- SWEDISH.

Details

If you want to create or change a collating sequence, use the TRANTAB= procedure to create or modify translation tables. When you create your own translation tables, they are stored in your PROFILE catalog, and they override any translation tables with the same name that are stored in the HOST catalog.

Note: System managers can modify the HOST catalog by copying newly created tables from the PROFILE catalog to the HOST catalog. All users can access the new or modified translation tables. △

If you are in a windowing environment, use the Explorer window to display the SASHELP HOST catalog. In the HOST catalog, entries of type TRANTAB contain collating sequences that are identified by the entry name.

If you are not in a windowing environment, issue the following statements to generate a list of the contents of the HOST catalog. Collating sequences are entries of the type TRANTAB.

```
proc catalog catalog=sashelp.host;
  contents;
run;
```

To see the contents of a particular translation table, use these statements:

```
proc trantab table=translation-table-name;
  list;
run;
```

The contents of collating sequences are displayed in the SAS log.

See Also

System Option:
 "TRANTAB=" on page 1171

SORTSIZE=

Specifies the amount of memory that is available to the SORT procedure

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

Category: Sort: Procedure options

PROC OPTIONS GROUP= SORT

Syntax

`SORTSIZE=n | nK | nM | nG | MIN | MAX | hex`

Syntax Description

***n* | *n*K | *n*M | *n*G**

specifies the amount of memory in bytes, kilobytes, megabytes, and gigabytes, respectively. If *n*=0, the sort utility uses its default.

MIN

specifies the minimum amount of memory available.

MAX

specifies the maximum amount of memory available.

hex

specifies the amount of memory as a hexadecimal number. This number must be followed by an X.

Operating Environment Information: Values for MIN and MAX will vary, depending on your operating environment. For details, see the 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

System Option:

“SUMSIZE=” on page 1163

The SORT procedure in the SAS documentation for your operating environment

SOURCE

Controls 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= LOGCONTROL

Syntax

SOURCE | NOSOURCE

Syntax Description

SOURCE

writes SAS source statements to the SAS log.

NOSOURCE

does not write SAS source statements to the SAS log.

Details

The SOURCE system option does not affect whether statements from the autoexec file, 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 Institute for problem determination and resolution. Δ

SOURCE2

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= LOGCONTROL

Syntax

SOURCE2 | NOSOURCE2

Syntax Description

SOURCE2

writes to the SAS log secondary source statements from files that have been included by %INCLUDE statements.

NOSOURCE2

does not 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 Institute for problem determination and resolution. △

SPOOL

Controls whether SAS writes SAS statements to a utility data set in the WORK data library

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

Category: Input control: Data processing

PROC OPTIONS GROUP= INPUTCONTROL

Syntax

SPOOL | NOSPOOL

Syntax Description

SPOOL

writes SAS statements to a utility data set in the WORK data library for later use by a %INCLUDE or %LIST statement, or by the RECALL command, within a windowing environment.

NOSPOOL

does not write SAS 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.

Examples

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 also can 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

Allows previous library references (librefs) to persist in a new SAS session

Valid in: configuration file, SAS invocation

Category: Files: External files

PROC OPTIONS GROUP= EXTFILES

Syntax

STARTLIB | NOSTARTLIB

Syntax Description

STARTLIB

causes library references (librefs) specified in a previous SAS session to persist in a new SAS session. This option only affects librefs that were assigned in the New Library window and selected in that window to be enabled at startup.

NOSTARTLIB

does not autoassign librefs at startup. This option only affects librefs that were assigned in the New Library window and selected in that window to be enabled at startup.

Details

You must set this option by using the New Library window. Enter the name of your libref and select the checkbox labeled **Enable at startup** to save your librefs. The only way to create librefs that can be autoassigned is through this window. The STARTLIB system option controls whether or not the librefs assigned in this window are assigned at startup.

SUMSIZE=

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

Syntax

SUMSIZE=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the amount of memory in bytes, kilobytes, megabytes, and gigabytes, respectively. When *n*=0, the default value, 8M, is used.

MIN

specifies the minimum amount of memory available.

MAX

specifies the maximum amount of memory available.

hex

specifies the amount of memory as a hexadecimal number. This number must be followed by an X.

Details

The SUMSIZE= system option affects the MEANS, SUMMARY, TABULATE, and SURVEYMEANS procedures.

Generally, the value of the SUMSIZE= system option should be less than the physical memory available to your process. If the procedure you are using needs more memory than you specify, the system creates a temporary utility file.

PERFORMANCE NOTE: Proper specification of SUMSIZE= can improve procedure performance by restricting the swapping of memory controlled by the operating environment.

See Also

System Option:

“SORTSIZE=” on page 1159

SYMBOLGEN

Controls whether the results of resolving macro variable references are written to the SAS log

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

Category: Macro: SAS macro

PROC OPTIONS GROUP= MACRO

Syntax

SYMBOLGEN | NOSYMBOLGEN

Syntax Description

SYMBOLGEN

displays the results of resolving macro variable references.

Tip: Use SYMBOLGEN for debugging programs.

CAUTION:

Using SYMBOLGEN can cause the log to become large. Δ

NOSYMBOLGEN

does not display the results of resolving macro variable references.

Details

SYMBOLGEN specifies that SAS display the results in this form:

```
SYMBOLGEN: Macro variable name
           resolves to value
```

SYNCHIO

Specifies whether synchronous I/O is enabled

Valid in: configuration file, SAS invocation

Category: Files: External files

PROC OPTIONS GROUP= EXTFILES and SASFILES

Syntax

SYNCHIO | NOSYNCHIO

Syntax Description

SYNCHIO

causes data set I/O to wait for completion.

NOSYNCHIO

llows other logical SAS tasks to execute (if any are ready) while the I/O is being done.

Details

If the SYNCHIO option is not specified on the SAS invocation line or in the configuration file, its default value depends on the runmode with which SAS was invoked.

If SAS is invoked in DMR mode, BATCH mode, or LINE mode, the value is NOSYNCHIO (ASYNCHIO).

If SAS is invoked in full screen mode or OBJECTSERVER mode, the value will be SYNCHIO (NOASYNCHIO).

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings for some SAS system options may vary both by operating environment and by site. Option values may also vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

Comparisons

The SYNCHIO system option is a mirror alias of the system option NOASYNCHIO. NOSYNCHIO is equivalent to ASYNCHIO.

See Also

System Options:
 “ASYNCHIO” on page 1060

SYSPARM=

Specifies a character string that can be passed to SAS programs

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

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

SYSPARM= *'characters'*

Syntax Description

'characters'

specifies a character string with a maximum length of 32,767.

Details

The character string that is specified can be accessed in a SAS DATA step by the SYSPARM() function, or anywhere in a SAS program by using the automatic macro variable reference &SYSPARM.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. △

TBUFSIZE=

Specifies the buffer size to use when you transmit data with SAS/CONNECT or SAS/SHARE software

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

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

TBUFSIZE=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n | *nK* | *nM* | *nG*

specifies the buffer size in bytes, kilobytes, megabytes, or gigabytes, respectively.

Range: 0 to MAX

MIN

sets the buffer size to 0, and requires SAS to use a default value.

MAX

sets the buffer size to the largest signed, four-byte integer that is representable in your operating environment.

hex

specifies the buffer size as a hexadecimal number. This number must be followed by an X.

Details

For more information, see *SAS/CONNECT Software: Usage and Reference* or *SAS/SHARE Software: Usage and Reference*.

TCPPOORTFIRST=

Specifies the first TCP/IP port for SAS/CONNECT software

Valid in: configuration file, SAS invocation

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

TCPPORTFIRST=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the first TCP/IP port for SAS/CONNECT to use for a remote operating environment connection in terms of bytes, kilobytes, megabytes, and gigabytes, respectively.

MIN

sets the minimum for the first TCP/IP port for SAS/CONNECT to use for a remote operating environment connection to 0.

MAX

sets the minimum for the first TCP/IP port for SAS/CONNECT to use for a remote operating environment connection to 65535.

hex

specifies the first TCP/IP port for SAS/CONNECT to use for a remote operating environment connection as a hexadecimal number. This number must be followed by an X.

Details

To assign the range of ports, assign the first port with the TCPPORTFIRST= system option and the last port with the TCPPORTLAST= system option. To restrict the connection to one port, set these options to the same value. This option is only useful in a SAS/CONNECT remote session.

Operating Environment Information: Valid values may be specific to a given operating environment. For details, see the SAS documentation for your operating environment or contact your system administrator for information about valid values. △

See Also

System Option:

“TCPPORTLAST=” on page 1167

Communications Access Methods for SAS/CONNECT and SAS/SHARE Software

TCPPORTLAST=

Specifies the last TCP/IP port for SAS/CONNECT software

Valid in: configuration file, SAS invocation

Category: Communications: Networking and encryption

PROC OPTIONS GROUP= COMMUNICATIONS

Syntax

TCPPORTLAST=*n* | *nK* | *nM* | *nG* | MIN | MAX | *hex*

Syntax Description

n* | *nK* | *nM* | *nG

specifies the last TCP/IP port for SAS/CONNECT to use for a remote operating environment connection in terms of bytes, kilobytes, megabytes, and gigabytes respectively.

MIN

sets the minimum for the last TCP/IP port for SAS/CONNECT to use for a remote operating environment connection to 0.

MAX

sets the maximum for the last TCP/IP port for SAS/CONNECT to use for a remote operating environment connection to 65535.

hex

specifies the last TCP/IP port for SAS/CONNECT to use for a remote operating environment connection as a hexadecimal number. This number must be followed by an X.

Details

To assign the range of ports, assign the first port with the TCPPORTFIRST= system option and the last port with the TCPPORTLAST= system option. To restrict the connection to one port, set these options to the same value. This option is only useful for a SAS/CONNECT remote session.

Operating Environment Information: Valid values may be specific to a given operating environment. For details, see the SAS documentation for your operating environment or contact your system administrator for information about valid values. Δ

See Also

System Option:

“TCPPORTFIRST=” on page 1166

Communications Access Methods for SAS/CONNECT and SAS/SHARE Software

TERMINAL

Determines whether SAS evaluates the execution mode and, if needed, resets the option

Valid in: configuration file, SAS invocation

Category: Environment control: Initialization and operation

PROC OPTIONS GROUP= EXECMODES

Syntax

TERMINAL | NOTERMINAL

Syntax Description

TERMINAL

causes SAS to evaluate the execution environment and to set the option to NOTERMINAL if an interactive environment (physical display) is not available. Specify TERMINAL when you use a windowing environment.

NOTERMINAL

causes SAS not to evaluate the execution environment. When you specify NOTERMINAL, SAS uses option settings that are associated with the BATCH system option.

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, requester windows are not displayed.

The TERMINAL option is normally used with the execution modes of windowing mode, interactive line mode, and noninteractive mode.

See Also

System Option:
 "BATCH" on page 1061

TOPMARGIN=

Specifies the size of the margin at the top of the page for the ODS printer destination

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

Category: SAS log and procedure output control: ODS printing

PROC OPTIONS GROUP= ODSPRINT

Syntax

TOPMARGIN=*margin-size* | "*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.

Interactions: Changing the value of this option may 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.

Default: inches

Requirement: When you specify margin-unit, enclose the entire option value in double quotation marks.

Details

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

Operating Environment Information: Most SAS system options are initialized with default settings when SAS is invoked. However, the default settings and/or option values for some SAS system options may vary both by operating environment and by site. For details, see the SAS documentation for your operating environment. △

For additional information on declaring an ODS printer destination, see ODS Statements in *The Complete Guide to the SAS Output Delivery System*

See Also

System Options:

“BOTTOMMARGIN=” on page 1062

“LEFTMARGIN=” on page 1111

“RIGHTMARGIN=” on page 1145

TRAINLOC=

Specifies the base location of SAS online training courses

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

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.

Examples

Some examples of the *base-URL* are:

- "file:///e:\onlintut"
- "http://server.abc.com/SAS/sastrain"

TRANTAB=

Specifies the translation tables that are used by various parts of SAS

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

Category: Environment control: Language control

PROC OPTIONS GROUP= LANGUAGECONTROL

Syntax

TRANTAB=(*catalog-entries*)

Syntax Description

catalog-entries

SAS catalog entries that contain translation tables. If you specify *entry-name.type*, SAS searches SASUSER.PROFILE first and then SASUSER.HOST.

Details

Translation tables are specified in a parenthesized list that has ten positions. The position in which a table appears in the list determines the type of translation table that is specified. Individual entries in the list are separated by commas. See the list of positions and types that follows:

Position	Type of translation table
1st	local-to-transport-format
2nd	transport-to-local-format
3rd	lowercase-to-uppercase
4th	uppercase-to-lowercase
5th	character classification
6th	scanner translation
7th	delta characters
8th	scanner character classification
9th	DBCS system table
10th	DBCS system table

CAUTION:

Do not change translation tables unless you are familiar with their purpose. △

Translation tables are used internally by the SAS supervisor to implement National Language Support (NLS). If you are unfamiliar with the purpose of translation tables, do not change the specifications without proper technical advice.

To change one table, specify null entries for the other tables. For example, to change the lowercase-to-uppercase table, which is third in the list, specify

```
options trantab = ( , , new-uppercase-table);
```

The other tables remain unchanged. The output from the OPTIONS procedure reflects the last specification for the TRANTAB= option and not the composite specification. Here is an example:

```
options trantab = ( , , new-uppercase-table);
options trantab = ( , , , new-lowercase-table);
```

PROC OPTIONS shows that the value for TRANTAB= is

(, , , new-lowercase-table), but both the *new-uppercase* and *new-lowercase* tables are in effect.

See Also

SAS Procedures Guide

SAS/CONNECT Software: Usage and Reference

USER=

Specifies the default permanent SAS data library

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

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

USER= *library-specification*

Syntax Description

library-specification

specifies the libref or physical name of a SAS data 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.

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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. Δ

VALIDVARNAME=

Controls the type of SAS variable names that can be created and processed during a SAS session

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

Category: Files: SAS files

PROC OPTIONS GROUP= SASFILES

Syntax

VALIDVARNAME=V7 | V6 | UPCASE | ANY

Syntax Description

V7

(default) indicates that up to 32 mixed case alphanumeric characters are allowed. Names must begin with alphabetic characters or an underscore.

V6

is the same as when VALIDVARNAME=V7, except that variable names are uppercased and they may be only 8 bytes long, as in earlier versions of SAS.

CAUTION:

Transitional use only: VALIDVARNAME=V6 is intended for transitional use only.

Use the V6 setting for applications that you need to use under both Version 6 and Version 7 of the SAS System. Δ

CAUTION:

May render some applications unusable: Using VALIDVARNAME=V6 may render some Version 7 and Version 8 macros or SAS/AF applications unusable. △

UPCASE

is the same as when VALIDVARNAME=V7, except that variable names are uppercased, as in earlier versions of SAS.

ANY

allows any characters to appear as valid SAS variable names. Symbols, such as "=" and "*", must be contained in a *'varname'n* construct.

Restriction: You can use variable names in this form only with the DATA step and the SQL procedure. This type of variable name may not be supported in all environments.

Restriction: If you use any characters other than when VALIDVARNAME=V7, then you must express the variable name as a *name literal*.

See: "SAS Name Literals" on page 1187

See Also

For more information on variable names see "Names in the SAS Language" on page 1184.

VNFERR

Controls how SAS responds when a _NULL_ data set is used

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

Category: Environment control: Error handling

PROC OPTIONS GROUP= ERRORHANDLING

Syntax

VNFERR | NOVNFERR

Syntax Description**VNFERR**

specifies that SAS either issue a warning and error message, or set `_ERROR_=1` and stop processing, or both if the error is severe enough to interrupt processing.

NOVNFERR

specifies that SAS issue a warning for a variable that is not found, but not set `_ERROR_=1` or stop processing.

Details

The VNFERR system option specifies whether SAS sets the error flag (`_ERROR_=1`) for a missing variable when a `_NULL_` data set (or a data set that is bypassed by the

operating environment control language) is used in a MERGE statement within a DATA step.

Comparisons

- VNFERR 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.
- VNFERR is similar to the DSNFERR system option, which generates an error message when a SAS data set is not found.

See Also

System Options:

“BYERR” on page 1066

“DSNFERR” on page 1092

WORK=

Specifies the WORK data library

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

WORK=library-specification

Syntax Description

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.

Operating Environment Information: 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 Option:
 “WORKTERM” on page 1176

WORKINIT

Initializes the WORK data library

Valid in: configuration file, SAS invocation

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

Syntax

WORKINIT | NOWORKINIT

Syntax Description

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.

Comparisons

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 or not SAS erases WORK files at the end of a SAS session.

See Also

System Option:
 “WORKTERM” on page 1176

Operating Environment Information: WORKINIT has behavior and functions specific to the UNIX operating environment. For details, see the SAS documentation for the UNIX operating environment. Δ

WORKTERM

Controls whether SAS erases WORK files at the termination of a SAS session

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

Category: Environment control: Files

PROC OPTIONS GROUP= ENVFILES

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 WORKINIT system option to prevent SAS from erasing existing WORK files on invocation. Use the WORKTERM system option to prevent SAS from erasing existing WORK files on termination.

See Also

System Option:

“WORKINIT” on page 1176

YEARCUTOFF=

Specifies the first year of a 100-year span that will be 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

Syntax

YEARCUTOFF= *nnnn* / *nnnnn*

Syntax Description

nnnn / nnnnn

specifies the first year of the 100-year span.

Range: 1582–19900

Default: 1920

Details

The YEARCUTOFF= value is the default 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 assumes a prefix of 19. For example, the value 92 refers to the year 1992.

The value specified 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. Figure 7.1 on page 1178 illustrates the relationship between the 100-year span and the two centuries if YEARCUTOFF=1950.

Figure 7.1 A 100-year Span with Values in Two Centuries

YEARCUTOFF= applies to the informats and formats that specify one- and two-digit years.

Note: YEARCUTOFF= has no effect on existing SAS dates or dates read from input data that include a four-digit year. △

Operating Environment Information: The syntax that is shown here applies to the OPTIONS statement. 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

“Year 2000” in *SAS Language Reference: Concepts*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS® Language Reference, Version 8*, Cary, NC: SAS Institute Inc., 1999.

SAS® Language Reference, Version 8

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-369-5

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.