Title

log — Echo copy of session to file

Syntax

Report status of log file

```
log

log query [logname | _all]
```

Open log file

```
log using filename [, append replace [text | smcl] name(logname)]
```

Close log

```
log close [logname | _all]
```

Temporarily suspend logging or resume logging

```
log {off | on} [logname]
```

Report status of command log file

```
cmdlog
```

Open command log file

```
cmdlog using filename [, append replace]
```

Close command log, temporarily suspend logging, or resume logging

```
cmdlog {close | on | off}
```

Set default format for logs

```
set logtype {text | smcl} [, permanently]
```

Specify screen width

```
set linesize #
```

In addition to using the `log` command, you may access the capabilities of `log` by selecting **File > Log** from the menu and choosing one of the options in the list.

Menu

File > Log

Description

log allows you to make a full record of your Stata session. A log is a file containing what you type and Stata's output. You may start multiple log files at the same time, and you may refer to them with a *logname*. If you do not specify a *logname*, Stata will use the name <unnamed>.

cmdlog allows you to make a record of what you type during your Stata session. A command log contains only what you type, so it is a subset of a full log.

You can make full logs, command logs, or both simultaneously. Neither is produced until you tell Stata to start logging.

Command logs are always text files, making them easy to convert into do-files. (In this respect, it would make more sense if the default extension of a command log file was .do because command logs are do-files. The default is .txt, not .do, however, to keep you from accidentally overwriting your important do-files.)

Full logs are recorded in one of two formats: Stata Markup and Control Language (SMCL) or plain text. The default is SMCL, but you can use set logtype to change that, or you can specify an option to state the format you wish. We recommend SMCL because it preserves fonts and colors. SMCL logs can be converted to text or to other formats by using the translate command; see [R] **translate**. You can also use translate to produce printable versions of SMCL logs. SMCL logs can be viewed and printed from the Viewer, as can any text file; see [R] **view**.

When using multiple log files, you may have up to five SMCL logs and five text logs open at the same time.

log or cmdlog, typed without arguments, reports the status of logging. log query, when passed an optional *logname*, reports the status of that log.

log using and cmdlog using open a log file. log close and cmdlog close close the file. Between times, log off and cmdlog off, and log on and cmdlog on, can temporarily suspend and resume logging.

If *filename* is specified without an extension, one of the suffixes .smcl, .log, or .txt is added. The extension .smcl or .log is added by log, depending on whether the file format is SMCL or text. The extension .txt is added by cmdlog. If *filename* contains embedded spaces, remember to enclose it in double quotes.

set logtype specifies the default format in which full logs are to be recorded. Initially, full logs are recorded in SMCL format.

set linesize specifies the maximum width, in characters, of Stata output. Most commands in Stata do not respect linesize, because it is not important for most commands. Most users never need to set linesize, because it will automatically be reset if you resize your Results window. This is also why there is no permanently option allowed with set linesize. set linesize is for use with commands such as list and display and is typically used by programmers who wish the output of those commands to be wider or narrower than the current width of the Results window.

Options for use with both log and cmdlog

`append` specifies that results be appended to an existing file. If the file does not already exist, a new file is created.

`replace` specifies that *filename*, if it already exists, be overwritten. When you do not specify either `replace` or `append`, the file is assumed to be new. If the specified file already exists, an error message is issued and logging is not started.

Options for use with log

`text` and `smcl` specify the format in which the log is to be recorded. The default is complicated to describe but is what you would expect:

If you specify the file as *filename*`.smcl`, the default is to write the log in SMCL format (regardless of the value of `set logtype`).

If you specify the file as *filename*`.log`, the default is to write the log in text format (regardless of the value of `set logtype`).

If you type *filename* without an extension and specify neither the `smcl` option nor the `text` option, the default is to write the file according to the value of `set logtype`. If you have not `set logtype`, then the default is SMCL. Also, the *filename* you specified will be fixed to read *filename*`.smcl` if a SMCL log is being created or *filename*`.log` if a text log is being created.

If you specify either the `text` or `smcl` option, then what you specify determines how the log is written. If *filename* was specified without an extension, the appropriate extension is added for you.

If you open multiple log files, you may choose a different format for each file.

`name(`*logname*`)` specifies an optional name you may use to refer to the log while it is open. You can start multiple log files, give each a different *logname*, and then close, temporarily suspend, or resume them each individually.

Option for use with set logtype

`permanently` specifies that, in addition to making the change right now, the `logtype` setting be remembered and become the default setting when you invoke Stata.

Remarks and examples

stata.com

For a detailed explanation of logs, see [U] **15 Saving and printing output—log files**.

When you open a full log, the default is to show the name of the file and a time and date stamp:

```
. log using myfile
-------------------------------------------------------------------------------
      name:  <unnamed>
       log:  C:\data\proj1\myfile.smcl
  log type:  smcl
 opened on:  12 Jan 2013, 12:28:23
.
```

The above information will appear in the log. If you do not want this information to appear, precede the command by `quietly`:

```
. quietly log using myfile
```

`quietly` will not suppress any error messages or anything else you need to know.

Similarly, when you close a full log, the default is to show the full information,

```
. log close
      name:  <unnamed>
       log:  C:\data\proj1\myfile.smcl
  log type:  smcl
 closed on:  12 Jan 2013, 12:32:41
────────────────────────────────────────────────────────────────────────────────
```

and that information will also appear in the log. If you want to suppress that, type `quietly log close`.

Stored results

`log` and `cmdlog` store the following in `r()`:

Macros

`r(name)`	*logname*
`r(filename)`	name of file
`r(status)`	`on` or `off`
`r(type)`	`smcl` or `text`

`log query _all` stores the following in `r()`:

Scalars

`r(numlogs)`	number of open log files

For each open log file, `log query _all` also stores

`r(name#)`	*logname*
`r(filename#)`	name of file
`r(status#)`	`on` or `off`
`r(type#)`	`smcl` or `text`

where # varies between 1 and the value of `r(numlogs)`. Be aware that # will not necessarily represent the order in which the log files were first opened, nor will it necessarily remain constant for a given log file upon multiple calls to `log query`.

Also see

[R] **query** — Display system parameters

[R] **translate** — Print and translate logs

[GSM] **16 Saving and printing results by using logs**

[GSW] **16 Saving and printing results by using logs**

[GSU] **16 Saving and printing results by using logs**

[U] **15 Saving and printing output—log files**