program — Define and manipulate programs

Description

program define defines and manipulates programs. define is required if program_name is any of the words: define, dir, drop, list, or plugin.

program dir lists the names of all the programs stored in memory.

program list lists the contents of the named program or programs. program list _all lists the contents of all programs stored in memory.

program drop eliminates the named program or programs from memory. program drop _all eliminates all programs stored in memory. program drop _allado eliminates all programs stored in memory that were loaded from ado-files. See [U] 17 Ado-files for an explanation of ado-files.

See [U] 18 Programming Stata for a description of programs. The remarks below address only the use of the program dir, program drop, and program list commands.

See [P] trace for information on debugging programs.

See the Combined subject table of contents, which immediately follows the Contents, for a subject summary of the programming commands.

Syntax

Define program

```
program [define] program_name [, [nclass|rclass|eclass|sclass]
   byable(recall[ , noheader] | onecall) properties(namelist) sort preserve
   plugin]
```

List names of programs stored in memory

```
program dir
```

Eliminate program from memory

```
program drop { program_name [ program_name [ ... ]] | _all | _allado }
```

List contents of program

```
program list [ program_name [ program_name [ ... ] ] | _all ]
```

Options

nclass states that the program being defined does not return results in r(), e(), or s(), and is the default.
rclass states that the program being defined returns results in \( r() \). This is done using the \texttt{return} command; see \texttt{[P] return}. If the program is not explicitly declared to be \texttt{rclass}, then it may not change or replace results in \( r() \).

eclass states that the program being defined returns results in \( e() \) or modifies already existing results in \( e() \). This is done using the \texttt{ereturn} command; see \texttt{[P] return} and \texttt{[P] ereturn}. If the program is not explicitly declared to be \texttt{eclass}, it may not directly replace or change results in \( e() \).

sclass states that the program being defined returns results in \( s() \). This is done using the \texttt{sreturn} command; see \texttt{[P] return}. If the program is not explicitly declared to be \texttt{sclass}, then it may not directly change or replace results in \( s() \), but it still may clear \( s() \) by using \texttt{sreturn clear}.

\texttt{byable(recall[, noheader]|onecall)} specifies that the program allow Stata’s \texttt{by varlist:} prefix. There are two styles for writing \texttt{byable} programs: \texttt{byable(recall)} and \texttt{byable(onecall)}. The writing of \texttt{byable} programs is discussed in \texttt{[P] byable}.

\texttt{properties(namelist)} states that \texttt{program_name} has the specified properties. \texttt{namelist} may contain up to 80 characters, including separating spaces. See \texttt{[P] program properties}.

\texttt{sortpreserve} states that the program changes the sort order of the data and that Stata is to restore the original order when the program concludes; see \texttt{[P] sortpreserve}.

\texttt{plugin} specifies that a plugin (a specially compiled C program) be dynamically loaded and that the plugin define the new command; see \texttt{[P] plugin}.

### Remarks and examples

The \texttt{program dir} command lists the names of all the programs stored in memory. \texttt{program list} lists contents of the program or programs.

> **Example 1**

When you start Stata, there are no programs stored in memory. If you type \texttt{program dir}, Stata displays an empty list:

```
. program dir
```

---

\[2\text{ program — Define and manipulate programs}\]
Later during the session, you might see

```
. program dir
(output omitted)
ado  756  _pred_se
ado  644  logit_p.GenScores
ado  306  logit_p.GetRhs
ado  5296 logit_p
ado  339  predict
(output omitted)
ado  559  logit.Replay
ado  4272 logit.Estimate
ado  827  logit
ado  287  webuse.Query
ado  588  webuse.Set
ado  269  webuse.GetDefault
ado  686  webuse
```

The ado in front indicates that the program was automatically loaded and thus can be automatically dropped should memory become scarce; see [U] 17 Ado-files. The number is the size, in bytes, of the program. The total amount of memory occupied by programs is 114,306 bytes. Notice the `logit_p.GetRhs` and `logit_p.GenScores` entries. These programs are defined in the `logit_p.ado` file and were loaded when `logit_p` was loaded.

Let’s now create two of our own programs with `program`:

```
. program rng
  1. args n a b
  2. if "'b'"==""
  3.   display "You must type three arguments: n a b"
  4.   exit
  5. }
  6. drop _all
  7. set obs 'n'
  8. generate x = (_n-1)/(_N-1)*('b'-'a')+'a'
  9. end

. program smooth
  1. args v1 v2
  2. confirm variable 'v1'
  3. confirm new variable 'v2'
  4. generate 'v2' = cond(_n==1|_n==N,'v1',('v1'[_n-1]+v1+v1[_n+1])/3)
  5. end
```
After you type `program`, lines are collected until you type a line with the word `end`. For our purposes, it does not matter what these programs do. If we were now to type `program dir`, we would see:

```
  . program dir
  286   smooth
  319   rng
(output omitted)
  ado  756   _pred_se
  ado  644   logit_p.GenScores
  ado  306   logit_p.GetRhs
  ado  5296  logit_p
  ado  339   predict
(output omitted)
  ado  559   logit.Replay
  ado  4272  logit.Estimate
  ado  827   logit
  ado  287   webuse.Query
  ado  588   webuse.Set
  ado  269   webuse.GetDefault
  ado  686   webuse
```

We can list a program by using the `program list` command:

```
  . program list smooth
  smooth:
    1. args v1 v2
    2. confirm variable ‘v1’
    3. confirm new variable ‘v2’
    4. generate ‘v2’ = cond(_n==1|_n==_N,’v1’,(_‘v1’[_n-1]+‘v1’+‘v1’[_n+1])/3)
```

If we do not specify the program that we want listed, `program list` lists all the programs stored in memory.
The `program drop` command eliminates programs from memory. Typing `program drop program_name` eliminates `program_name` from memory. Typing `program drop _all` eliminates all programs from memory.

```
. program drop smooth
. program dir
  319   rng
(output omitted)
ado  756  _pred_se
ado  644  logit_p.GenScores
ado  306  logit_p.GetRhs
ado  5296 logit_p
ado   339 predict
(output omitted)
ado   559  logit.Replay
ado  4272  logit.Estimate
ado   827  logit
ado   287  webuse.Query
ado   588  webuse.Set
ado   269  webuse.GetDefault
ado   686  webuse

118506
. program drop _all
. program dir
```

Also see

[P] byable — Make programs byable
[P] discard — Drop automatically loaded programs
[P] sortpreserve — Sort within programs
[P] trace — Debug Stata programs
[D] clear — Clear memory
[R] query — Display system parameters
[U] 18 Programming Stata