continue — Break out of loops

Description Syntax Option Remarks and examples Also see

Description

The continue command within a foreach, forvalues, or while loop breaks execution of the current loop iteration and skips the remaining commands within the loop. Execution resumes at the top of the loop unless the break option is specified, in which case execution resumes with the command following the looping command. See [P] **foreach**, [P] **forvalues**, and [P] **while** for a discussion of the looping commands.

Syntax

continue [, break]

Option

break indicates that the loop is to be exited. The default is to skip the remaining steps of the current iteration and to resume loop execution again at the top of the loop.

Remarks and examples

We illustrate continue with the forvalues command, but it can be used in the same way with the foreach and while commands.

▷ Example 1

The following forvalues loop lists the odd and even numbers from one to four:

```
. forvalues x = 1(1)4 {
  2.         if mod(‘x’,2) {
  3.                 display "‘x’ is odd"
  4.         }
  5.         else {
  6.                 display "‘x’ is even"
  7.         }
  8. }
1 is odd
2 is even
3 is odd
4 is even
```

It could be coded using the `continue` command instead of `else`:

```
. forvalues x = 1(1)4 {
  2.          if mod(‘x’,2) {
  3.                  display "‘x’ is odd"
  4.                  continue
  5.          }
  6.          display "‘x’ is even"
  7. }
1 is odd
2 is even
3 is odd
4 is even
```

When `continue` is executed, any remaining statements that exist in the loop are ignored. Execution continues at the top of the loop where, here, `forvalues` sets the next value of `‘x’`, compares that with 4, and then perhaps begins the loop again.

Example 2

`continue, break` causes execution of the loop to stop; it prematurely exits the loop.

```
. forvalues x = 6/1000 {
  2.          if mod(‘x’,2)==0 & mod(‘x’,3)==0 & mod(‘x’,5)==0 {
  3.                  display "The least common multiple of 2, 3, and 5 is ‘x’"
  4.                  continue, break
  5.          }
  6. }
The least common multiple of 2, 3, and 5 is 30
```

Although the `forvalues` loop was scheduled to go over the values 6–1,000, the `continue, break` statement forced it to stop after 30.

Also see

[P] **foreach** — Loop over items

[P] **forvalues** — Loop over consecutive values

[P] **while** — Looping

[P] **exit** — Exit from a program or do-file

[P] **if** — if programming command

[U] 18 Programming Stata

For suggested citations, see the FAQ on citing Stata documentation.