Session 4: Programming in Do-Files (II)

Manuel V. Montesinos

Computation Brush-Up Course
Competition and EPP Master Programs

Fall 2020

Macros

What are they?: a name associated with a text (contents). Everywhere a punctuated macro appears in a command, the macro contents are substituted for the macro name.

Why do we use them?: to avoid repetitive commands and typing.

Which macros are there?: local (valid for the current text) and global (valid for the current Stata session):

- `local localmacroname associatedtext` (evaluate using `‘localmacroname’`).
- `global globalmacroname associatedtext` (evaluate using `$globalmacroname`).

Conditional Execution

Use the command `if` **followed by a condition**, with the following structure:

```
if expression {
    commands to be executed if expression is true
}
else {
    optional block to be executed if expression
        is not true
}
```

Note that the opening bracket { must be the last thing on a line, and the closing bracket } must be on a new line by itself.

The expression is only one value (i.e. can't have variables here).

Loops

While loop

```
while expression {
    commands to be executed repeatedly until
        expression is false
}
```

The condition is an expression. The loop executes as long as the condition is true (nonzero).

Example:

```
local i=5
while ‘i’>0 {
    display ‘i’+1
    local i=‘i’-1
}
```

Loops

foreach and forvalues

`foreach` loops over elements of a local or global macro or variables in a list:

```
foreach i in 1 2 3 4 5 {
    display "`i' + 1 = "
    display `i' + 1
    display ""
}
```

`forvalues` loops over series or lists of numbers:

```
forvalues i=1/5 {
    display "`i' + 1 = "
    display `i' + 1
    display ""
}
```

Loops

Nested loops

A loop inside another loop:

```
forvalues i=43/45 {
    forvalues j=0/1 {
        display "-----"
        display "AGE==‘i’ & MARRIED==‘j’"
        display "-----"
        if ‘i’==45 {
            summarize EARNINGS if AGE==‘i’
                & MARRIED==‘j’
        }
        else {
            summarize HOURS EARNINGS if AGE==‘i’ \&
                MARRIED==‘j’
        }
    }
}
```

Programs

Stata allows you to write **your own** commands.

Make sure a program with the **same name** does not exist: `capture drop program programname`.

Define the new program:

```
program programname
    ...
    ...
end
```

Use **positional arguments**: `‘1’`, `‘2’`, `‘3’`...

Type the program name and the arguments to **execute** it: `programname A B C`.

Exercises

1. Write a loop to create 10 new variables: `var1`, ..., `var10` with `var1=10`, `var2=20`, ..., `var10=100`.
2. Use the *uslifeexp* dataset. For each life expectancy variable:
   - Create a two-way graph (year on the x-axis).
   - Save the graph in your working directory.
3. Download this dataset. It contains artificial data of people and their incomes for twelve months. Create a set of dummy variables for whether the person had any income in that month.
4. **If bored**: Write a program of your choice with at least three arguments.