# SOC 5050: Week 11 Commands Quick Reference

*Christopher Prener, Ph.D.*

*October 31$^{st}$, 2016*

## Loops

*Looping over Variables*

```
foreach var of varlist variableList {
    commands
}
```

*Looping over Variables Stored in Local Macro*

```
foreach var of local localName {
    commands
}¹
```

¹ Note that reference to the local name is inconsistent with other Stata uses of local macros - you do not need the back-tick/apostrophe combination wrapped around the local name.

*localName* must be set ahead of time.

*Looping over New Variables*

```
foreach var of newlist variableList {
    generate 'var' = { .  | localName | varName }
    commands
}
```

*Looping over Values*

```
forvalues i = { lowVal/highVal | lowVal(step)highVal } {
    commands
}
```

## Macros

*Pulling Item from List Stored in Local Macro*

```
local localName :  word 'localVal' of 'sourceLocal'²
```

² Both *localVal* and *sourceLocal* must be set ahead of time. By convention, *localVal* is usually i, j, or k. You must increase the value of *localVal* by an increment with each iteration of the loop.

*Increasing Value of Local Macro by Increment*

```
local ++'localVal'
```

## Label Dataset

```
label dataset "dataset label text"
```
[3]

[3] The dataset label text is limited to 80 characters. Stata will truncate your text after that point.

## Notes

### Dataset-level Notes
```
notes _dta:  dataset note text
```
[4]

[4] You do not need to wrap the text of notes in double quotes (this is inconsistent with nearly every other instance of working with text in Stata). You can use the /// command to wrap your lines at or around 80 characters - Stata will fold all of your sentences together when the do-file executes.

### Variable-level Notes
```
notes varname:  variable note text
```
[5]

[5] See previous note - same applies for variable-level notes.

## Variable Index

```
describe
```
[6]

[6] This should be wrapped in a quietly log using and a quietly log close pair of commands.

## Codebook

```
codebook [varlist] [, header notes]
```
[7]

[7] This should be wrapped in a quietly log using and a quietly log close pair of commands.

## Create Dummy Variables

```
tabulate varname, generate(groupName)
```

## Logic Checks

```
assert expression [if] [additionalExpressions]
```
[8]

[8] No output means that the expression is true. If the expression is false, Stata will return output along with an error message that will halt the execution of your do-file.

*Document Details*

Document produced by Christopher Prener, Ph.D. for the Saint Louis
University course soc 5050 - quantitative analysis: applied
inferential statistics. See the course wiki and the repository
`README.md` file for additional details.