Introduction to STATA

Duah Dwomoh, MPhil
School of Public Health,
University of Ghana, Accra

July 2016

International Workshop on Impact Evaluation of Population, Health and Nutrition Programs

Learning Objectives

- Familiarity with STATA environment
- Opening and closing STATA
- About the working directory
- Creating and maintaining ‘do’ and ‘log’ files
- Use of help files
- Some basic STATA commands
- Data processing

Introduction: Why use Stata?

According to www.stata.com:

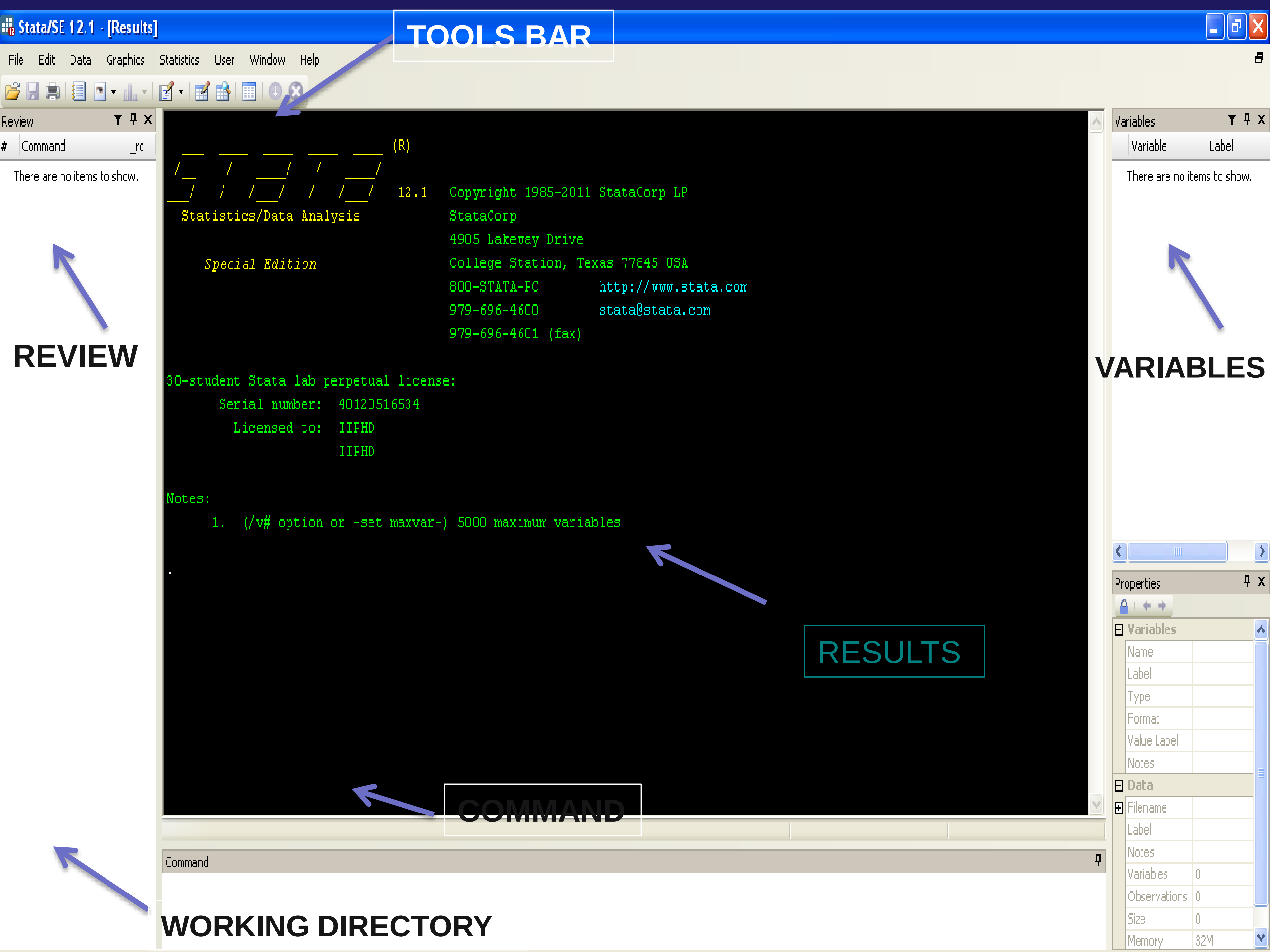
- Stata is a complete, integrated statistical package that provides everything you need for data analysis, data management, and graphics
- Fast, accurate, and easy to use
- Broad suite of statistical capabilities
- Complete data-management facilities
- Publication-quality graphics
- Technical support and learning resources

Menus vs. Commands

- Stata has a set of pull-down menus of commands.
    - Allows user to get results without needing to know syntax.
    - Alternatively, command syntax allows user to reproduce results easily.
        - Convenient if your datasets are updated repeatedly.

Window Layout

- Stata has different windows.
    - Command: where commands are entered.
        - All commands and variables are case sensitive.
    - Results: where results appear.
    - Review: where past commands are listed.
        - Clicking a past command in Review window brings it to the command window where it can be modified and re-executed.
    - Graph: where graphs are displayed (appears only when graphs are requested).
    - Variable: where variables in current dataset are listed.



TOOLS BAR

REVIEW

VARIABLES

RESULTS

COMMAND

WORKING DIRECTORY

```
  ___  ____  ____  ____  ____ (R)
 /__    /   ____/   /   ____/
___/   /   /___/   /   /___/   12.1   Copyright 1985-2011 StataCorp LP
  Statistics/Data Analysis            StataCorp
                                      4905 Lakeway Drive
     Special Edition                  College Station, Texas 77845 USA
                                      800-STATA-PC        http://www.stata.com
                                      979-696-4600        stata@stata.com
                                      979-696-4601 (fax)

30-student Stata lab perpetual license:
       Serial number:  40120516534
         Licensed to:  IIPHD
                       IIPHD

Notes:
      1.  (/v# option or -set maxvar-) 5000 maximum variables

.
```

Opening STATA

- Double click the STATA shortcut icon from the desktop
- Go from the start button
- Double click the STATA file directly

Closing STATA

- Click on the Close icon (red cross) at the top right hand corner
- Type ‘exit’ in the Command Window
- Select ***Exit*** from the ***File*** menu

Working directory

- By default STATA will save the files in the folder where STATA was installed initially
- But the directory can be changed to some other where you want to save your files
- The command for changing the working directory is:

```
cd “path of the folder”
```

- For example in my laptop, the command will be:

```
cd “D:\Stata\Impact Evaluation”
```

Command Syntax

- Commands should always be in lower case
- STATA is very sensitive to spelling mistakes

Convention for the STATA command

```
[prefix:]command varlist [if] [in] [weights] [,options]
```

For example

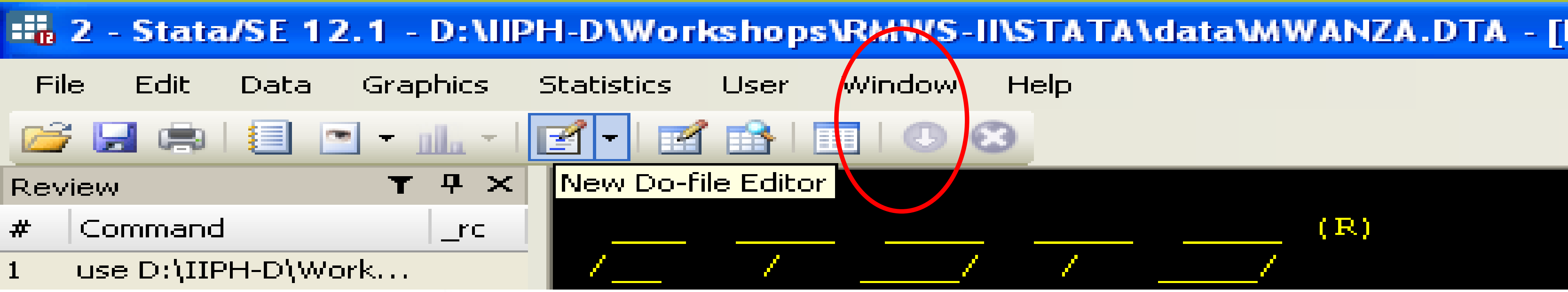
```
summarize var_name
sum var_name if var_name==1
sum var_name in 1/100
sum var_name if var_name ==1 in 1/100
sum var_name if var_name ==1, detail
bysort var_name: sum var_name, detail
```

‘do’ files

- A do-file is a text (also called batch) file with a series of commands to be executed in order by Stata.
- Also great for composing, revising, and saving Stata commands.
- To use a do-file:
  - Click on Do-File Editor.
  - Enter commands.
  - Save file with .do extension.
- To execute a do-file:
  - Via command: **do** pathoffile/filename.do.
  - Via drop- menu: File → Do …

‘do’ files

- What it does?
  - It saves all your commands
  - You can use these do files later and execute commands directly from the do file
- How to create a do file?
  - Click on the short cut toolbar
  - Window > Do-file editor> New Do-file editor

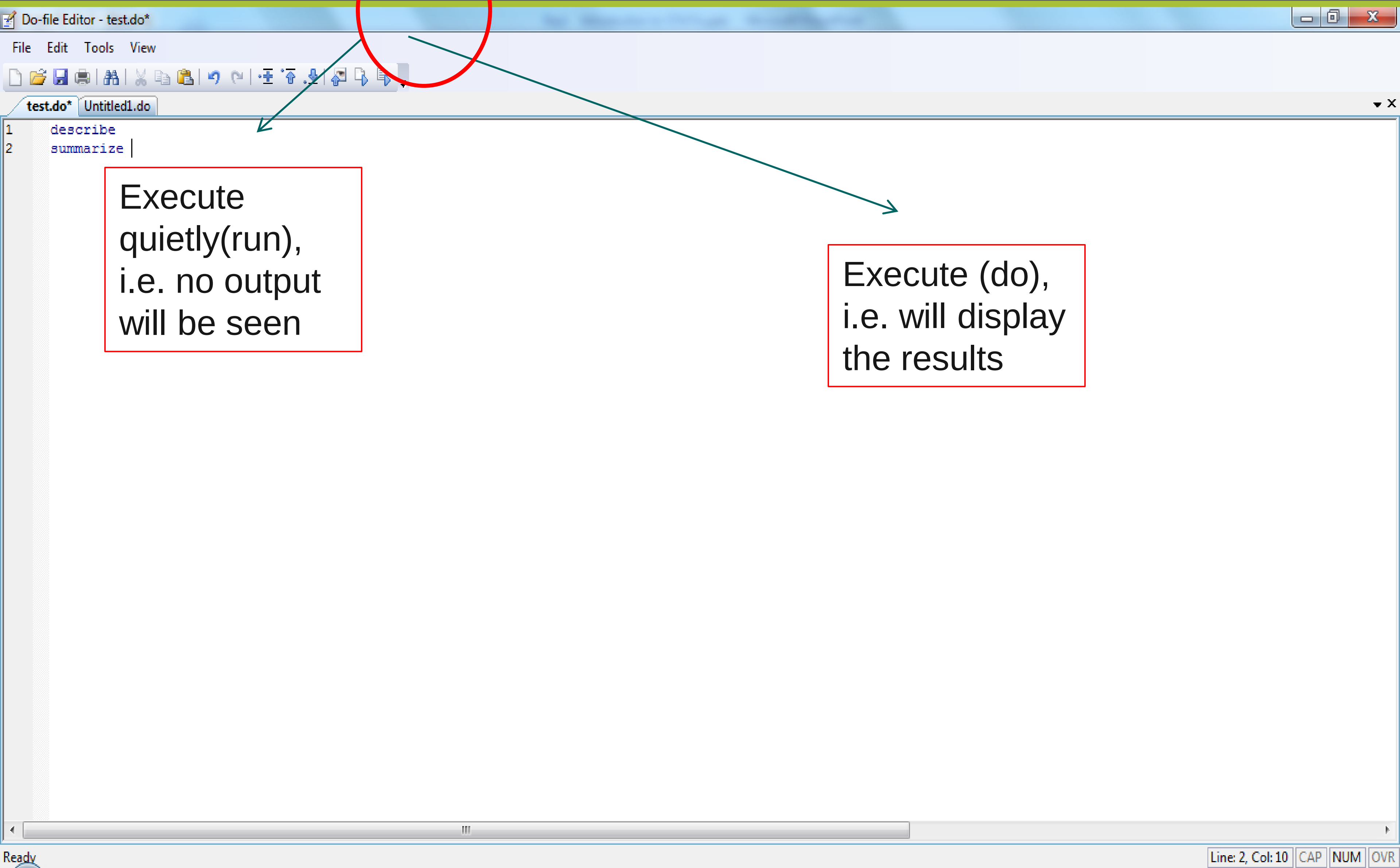


1 use D:\IIPH-D\Work...

‘do’ files

- Maintaining a ‘do’ file
  - Write commands directly in the do file
  - Copy and paste commands from the review window of STATA
  - Save the do file with the ‘.do’ extension
- Opening a do file
  - Open a new do file editor, then open the required do file from the ‘file’ menu

Executing the commands using do-files



```
describe
summarize
```

Execute quietly(run), i.e. no output will be seen

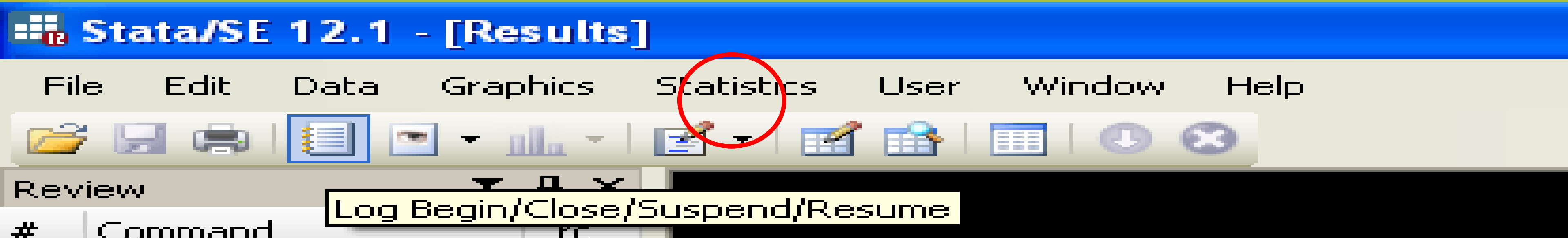
Execute (do), i.e. will display the results

‘log’ files

- Can be used to record (and print):
  1. Executed commands.
  2. Resulting output (except for graphs).
- Recommend that the first thing you do in Stata is open a log file.
- Two types of Log files:
  - Unformatted Log files:
    - Lacks formatting, but is simpler to use if you plan to insert and edit in text editor.
    - Common file extension: .log.
  - Formatted Log files:
    - “**S**tata **M**arkup and **C**ontrol **L**anguage” file. Great for viewing and printing within Stata.
    - Common file extension: .smcl.

‘log’ files

- What it does
  - It saves all your output
  - By default it gets save as ‘.smcl’ extension, but prefer to save as ‘.log’ format
- How to create?
  - Click on the shortcut tool bar
  - It will ask to save the same
  - After opening, whatever you will execute will get saved



Creating/Suspending/Closing log files

- You can suspend the log file at any time and resume again by typing the following:

```
log off
log on
```

- For closing a log file, type:

```
log close
```

- Want to append after closing the log file, type:

```
log using file_name, append
```

- Want to replace the old file with a new one, type:

```
log using file_name, replace
```

How to use STATA help

- By using STATA help menu
- By using STATA commands as follows:

```
help (command_name)
findit (keyword)
```

- For example
  - `help summarize`
  - `findit table`

Inputting Data

- Many Options:
  - Manually enter data into the Stata Data Editor.
  - Copy data into the Data Editor from another source (ex.: Excel).
  - Importing an ASCII (text) file.
  - Reading in an Excel spreadsheet (tab- or comma-delimited text file).

Inputting Data

- Many Options:
  - Open existing Stata Data file.
    - Common file extension: .dta.
  - Use a conversion package (eg, StatTransfer or DBMSCopy) to read in data from another package (eg, SAS data file).

Importing data into STATA

- Directly copy paste from excel to STATA data editor
- Load an existing dataset saved in STATA's own binary format using the `use` command
- Load an existing dataset saved in excel

```
 import excel datafile_name.xlsx, firstrow
```

- Enter data in key board using `input` or `edit` commands

Loading data using `input`

Type the following in the command window:

```
input age sex income
27 1  12000
45 2  13000
34 1  15000
end
```

If the variable sex is a string , then

```
input age str6 sex income
```

Loading data using `.edit`

- `edit` or click on the data editor short cut in the menu bar
- Enter the data as you enter in spread sheet
- What is typed in the first cell will automatically determined the storage type
- Name the variable by clicking on the variable cell (default name *var1 var2*…..)

Types of data

- Numeric – Black
- Numeric – Blue (Stored as numeric but visible as text)
- String - Red
- Dates (Before formatting) – Red
- Dates (After formatting) - Black

Note: STATA either considers a variable as string or numeric, cannot accept mixed formats

Things to remember before import

- Make sure the files are in the working directory
- If the data is missing leave it blank and not “0” or “NA”
- Even if one cell contains any non numeric entry, the variable will be read as string by STATA

Converting variables type

- You can convert string variable to numeric variable

```
encode var_name [if] [in] , generate(new_var)
```

- Numeric variable to string variable

```
decode var_name [if] [in] , generate(new_var)
[maxlength(#)]
```

- Convert string variables to numeric variables

```
destring [varlist], {generate(newvarlist)
|replace} [destring_options]
```

Operators in STATA

Arithmetic		**Logical**		**Relational**	
+	addition	! (or ~)	not	>	greater than
-	subtraction	|	or	<	less than
*	multiplication	&	and	>=	greater than or equal
/	division			<=	less than or equal
^	power			==	equal
				!= (or ~=)	not equal

Note: the double equal (==) is not a mistake and must be used for equality testing

First look at the data

- Some basic STATA commands to understand the data are as follows:

```
describe
browse
summarize
```

Summarizing Variables

- Continuous variables

```
sum var_name, detail

table var_name, contents (freq mean age sd age)
```

- Categorical variables

`tab var_name`	one way table for one variable
`tab1 varlist`	one way table for all variables listed
`tab var1 var2`	two-way table
`tab2 var_name varlist`	All possible combination of two way tables

Graphical presentation

- STATA commands for some basic graphs

```
histogram var_name, normal
scatter var_name
graph pie var_name,over(var_name)
graph box var_name
```

Basic data processing commands

- Generating a new variable

```
generate new_var=expression [if] [in]
```

- Modifying existing variable

```
replace old_var=exp [if] [in]

recode var_name (rule)(rule)…, generate(new_var)
```

- Reducing data

`drop varlist` (drops variables)

`keep varlist` (keeps variables)

Missing data

- Missing data in STATA appears as “.”
- Missing value in STATA is considered as largest number
- In datasets missing data may be entered as 9, 999
- So if missing values are coded as 999, you can change it to “.” by using following:

```
mvdecode var_name, mv(999)
```

Exploring data

- **Describe:** Describe a dataset
- **List:** List the contents of a dataset
- **Codebook:** Detailed contents of a dataset
- **Log:** Create a log file
- **Summarize:** Descriptive statistics
- **Tabstat:** Table of descriptive statistics
- **Table:** Create a table of statistics

Exploring data

- **Stem:** Stem-and-leaf plot
- **Graph:** High resolution graphs
- **Sort:** Sort observations in a dataset
- **Histogram:** Histogram for continuous and categorical variables
- **Tabulate:** One- and two-way frequency tables
- **Type:** Display an ASCII file

Modifying Data

- **label data:** Apply a label to a data set
- **Order:** Order the variables in a data set
- **label variable:** Apply a label to a variable
- **label define:** Define a set of a labels for the levels of a categorical variable
- **label values:** Apply value labels to a variable
- **List:** Lists the observations

Modifying Data

- **Rename:** Rename a variable
- **Recode:** Recode the values of a variable
- **Generate:** Creates a new variable
- **Replace:** Replaces one value with another value

Managing Data

- **Pwd:** Show current directory (pwd=print working directory)
- **dir** or **ls:** Show files in current directory
- **cd** Change directory
- **keep if:** Keep observations if condition is met
- **Keep:** Keep variables (dropping others)
- **Drop:** Drop variables (keeping others)
- **append using:** Append a data file to current file
- **Merge:** Merge a data file with current file

Analyzing Data

- **ttest:** t-test
- **regress:** Regression
- **predict:** Predicts after model estimation
- **kdensity:** Kernel density estimates and graphs
- **pnorm:** Graphs a standardized normal plot
- **qnorm:** Graphs a quantile plot
- **rvfplot:** Graphs a residual versus fitted plot
- **rvpplot:** Graphs a residual versus individual predictor plot
- **xi:** Creates dummy variables during model estimation

Analyzing Data

- **test:** Test linear hypotheses after model estimation
- **oneway:** One-way analysis of variance
- **anova:** Analysis of variance
- **logistic:** Logistic regression
- **logit:** Logistic regression
- **probit**: Probit regression
- **regress**: Linear regression
- **glm**: generalized linear model
- **xtgee**: panel data analysis (generalized estimation equation)

MEASURE Evaluation is a MEASURE project funded by the U.S. Agency for International Development and implemented by the Carolina Population Center at the University of North Carolina at Chapel Hill in partnership with Futures Group International, ICF Macro, John Snow, Inc., Management Sciences for Health, and Tulane University. Views expressed in this presentation do not necessarily reflect the views of USAID or the U.S. Government. MEASURE Evaluation is the USAID Global Health Bureau's primary vehicle for supporting improvements in monitoring and evaluation in population, health and nutrition worldwide.

GEMNet Health
Global Evaluation & Monitoring Network for Health

USAID
FROM THE AMERICAN PEOPLE

MEASURE Evaluation

References

- *International Workshop on Impact Evaluation of Population, Health and Nutrition Programs conducted by MEASURE Evaluation and PHFI, India (Dr. Ranjana Singh, IIPH Delhi)*
- *A brief Introduction to STATA with 50+ Basic Commands by Tobias Pfaff*
- *www.stata.com*