1. What is Stata useful for?
2. The Famous Black Screen
3. Data Input
4. Commands’ Syntax
5. Where and How to get HELP
6. Now we go to Stata!
What is Stata Useful for?

- Stata is made for statistical analysis
  - It allows you to develop a wide range of statistical estimations
  - Very powerful (once you get it) and relatively user friendly (compared to other statistics software)
  - Kind of open source: People (myself included) are all the time creating new commands to make available to the public.

- Do not expect it to work as a worksheet like excel

- Aside note: **you will not learn Stata only by listening to me or reading instructions. You need to sit try handling stata by yourself. There is a start-up cost, but is highly rewarded.**
The Famous Black Screen
The Soul of Stata is in the Background
How to fill this?

- Open an existing data file (.dta)
- Open an existing non-Stata data file (.xls, .csv, .txt)
  - There are formats Stata cannot open. You need to convert them to .dta using a program like Stat/Transfer
- Copy and paste in data browser
- Type in browser (avoid!!!)
- Create variables using commands
- Memory size
  - If your Stata version is earlier than 12, you need to tell Stata if you are using a relatively big data file. He requires this to allocate RAM. Therefore, in most computers you cannot use data files larger than your RAM available.
  - Modify memory allocated by typing the command: set mem #g or set mem #m
Open an Existing .dta Data File

There are several ways

1. Use “open” icon, as in any software
2. Type the command `use`

Example

```
use "Users/MS/NLSY/NLSYforFactor.dta", clear
```
Open an existing non-Stata data file (.xls, .csv, .txt)

There are several ways

1. Using the “import” menu

   ![Stata Import Menu](image)

2. Type import commands: insheet, infile or infix

   Example

   ```
   infile using "~/Documents/UMD/PS2/default.dct"
   ```

3. I encourage you to use the “import” menu and then copy the instruction it produces in the result window into your do-file for future uses
The Soul of Stata is in the Background
Consider the instructions you give Stata as modified sentences with the form:

**verb** predicate, adverbs

**Examples**

use "Users/MS/NLSY/NLSYforFactor.dta", clear
regress y x1 x2, noconstant

- The verb is the command, the predicate comprises variables or files you are going to use the command on and the adverbs are the options of the command that tells the command specific instructions on how to function.
The log Files

- Stata does not save the output it produces automatically. However, you can save it using Stata’s log files.

- Log files have the characteristic that they save everything from the moment you open them to the moment you close them. Their format is somewhat special, although you can save them in .txt as well.

- You can close a log file and re-start it again afterwards. It is called “appending.”

- You can open a log file in several ways:
  - Menu “File -> Log -> Begin”
  - Use the log icon on the top of the page
  - Type: `log using name, replace [or append]`

- You need to close it when you finish, if not it will continue recording every output Stata produces.
## Things We Need to know Before Going to Stata

### Table: If Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>==</code></td>
<td>Equal to</td>
</tr>
<tr>
<td><code>!=</code></td>
<td>Different to</td>
</tr>
<tr>
<td><code>&gt;</code></td>
<td>Greater than</td>
</tr>
<tr>
<td><code>&gt;=</code></td>
<td>Greater or equal than</td>
</tr>
<tr>
<td><code>&lt;</code></td>
<td>Less than</td>
</tr>
<tr>
<td><code>&lt;=</code></td>
<td>Less or equal than</td>
</tr>
<tr>
<td><code>&amp;</code></td>
<td>And</td>
</tr>
<tr>
<td>`</td>
<td>`</td>
</tr>
<tr>
<td><code>inrange(var1,#1,#2)</code></td>
<td>Values between #1 and #2 inclusive</td>
</tr>
<tr>
<td><code>inlist(var1,#1,#2,...)</code></td>
<td>Values equal to #1,#2, ...</td>
</tr>
</tbody>
</table>
Things We Need to know Before Going to Stata 2

Stata macros:

- **Locals**: Spaces of memory that store values. They are temporary, once the routine finished, they are deleted. They are recalled by preceding the name with a ‘‘ character and a ’ character at the end of the name.

  **Example**

  ```stata
  local i=2
  generate age_last_world_cup = age_2012 - ‘i’
  ```

- **Globals**: Other type of spaces of memory that store values. The difference with the locals is that they are not temporary, they remain after the routine has finished. They are recalled by preceding the name with a $ character.

- **Tempfiles and tempnames**: Are also spaces of memory that store temporarily data files and names. They are used in advanced routines.
Where and How to get HELP

An advantage of Stata is that it is relatively easy to get help.

1. The menu “Help -> Contents” opens a window where you can look for a command to check what it does and its syntax
   - This is an amazing help system. Every command needs to have a help in order to be in stata
   - However, you need to know the name of the command you are looking for

2. Google what you need. 99% of the times someone has faced the same challenge you are facing and in consequence the internet has an answer.
   - People post questions (not basic questions) on a service called Statalist and the answers people give are kept on the web for users to see
   - Helps if you do not know the name of the command you are looking for

3. Go to the Stata manuals. The are very good once you get to know them

4. There is a growing number of books about stata
Now we go to Stata!