

Initial Software Setup

January 18, 2022

Pratt

Goals

- The important software to install is **Git** and **Conda**
- This requires getting acquainted with the **command line**
- Launch an interactive computing environment with **Jupyter Lab**

Free and Open Source Software (FOSS)

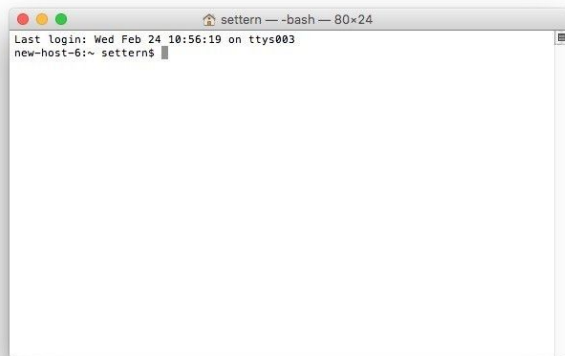
- All the required software for this course is free and open-source
- “Open source” is software that is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software and its source code

Command Line Interface (CLI)

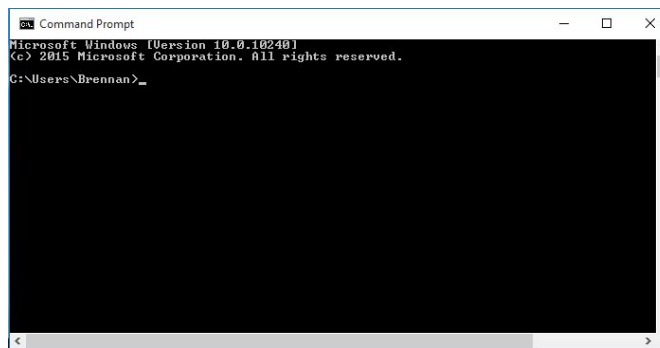
- The command line is a text interface for your computer
- Enables you to give your computer instructions via text commands, rather than point-and-click graphical user interfaces (GUIs)
- In order to install, write and run code, it will be helpful to first familiarize ourselves with the command line

Familiarize yourself with the command line

- Open the command line application on your computer
- On a Mac computer, this is called **Terminal**
- On a Windows computer, this is called **Command Prompt**



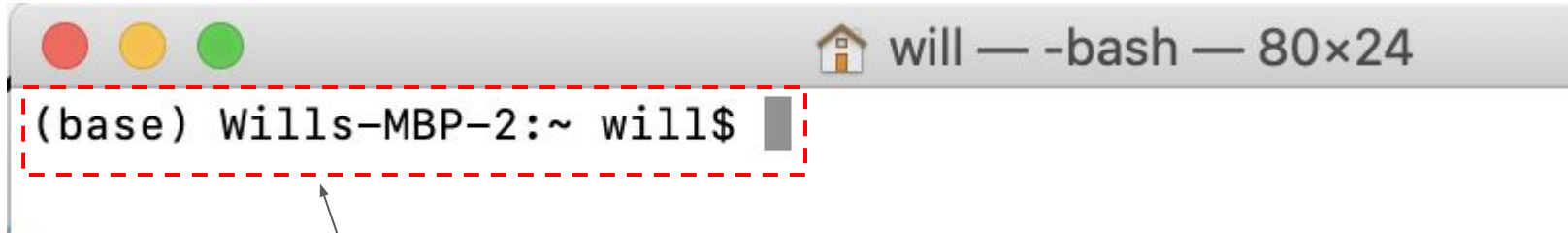
Mac Terminal



Windows Command Prompt

Familiarize yourself with the command line

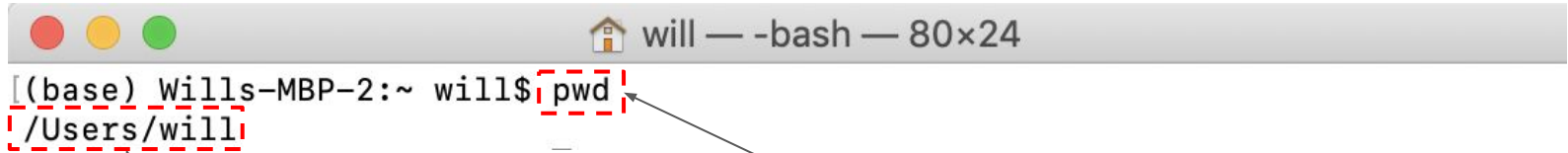
- The first bit of text that shows up is called the ***prompt***
- The prompt is supplied automatically, you do not need to type it
- The exact details of the prompt will differ, not important right now



The prompt. Waiting for me to do something.

Navigating the command line

- Type `pwd` after the prompt and hit enter
- This prints the full path of the current location (i.e. “working directory”)



```
will — -bash — 80x24
[(base) Wills-MBP-2:~ will$ pwd
/Users/will
```

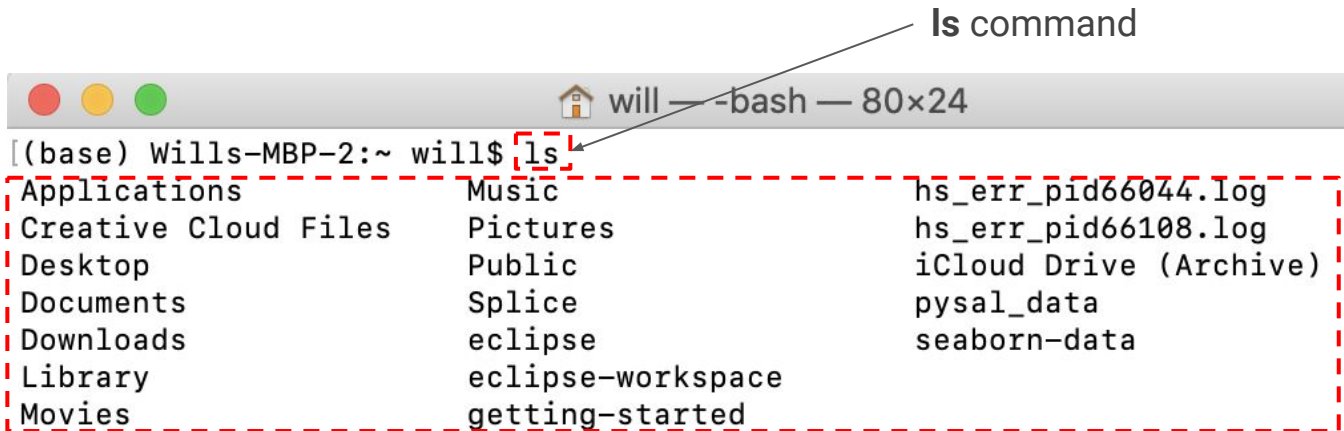
The image shows a macOS terminal window with a title bar containing three colored window control buttons (red, yellow, green) and a title text 'will — -bash — 80x24'. The terminal content shows a prompt '[(base) Wills-MBP-2:~ will\$' followed by the command 'pwd' which is enclosed in a red dashed box. Below the command, the output '/Users/will' is displayed and also enclosed in a red dashed box. An arrow points from the text 'pwd command' to the 'pwd' command in the terminal. Another arrow points from the text 'The result tells your current location...' to the output path '/Users/will'.

The result tells your current location (path) on your computer. I am currently located in a folder called **will**, which is inside a folder called **Users**

pwd command

Navigating the command line

- Type `ls` (that's "L" followed by "S", short for *list*) and hit enter
- This lists all the files and folders within your current directory (folder)



A screenshot of a macOS terminal window. The title bar shows three colored window control buttons (red, yellow, green) on the left, a home icon, the name 'will', and the window title '-bash — 80x24'. The terminal content shows the prompt '[(base) Wills-MBP-2:~ will\$' followed by the command 'ls' which has been executed. The output of the command is a list of files and folders arranged in three columns: Applications, Music, hs_err_pid66044.log; Creative Cloud Files, Pictures, hs_err_pid66108.log; Desktop, Public, iCloud Drive (Archive); Documents, Splice, pysal_data; Downloads, eclipse, seaborn-data; Library, eclipse-workspace; and Movies, getting-started. A red dashed rectangular box encloses the entire output of the 'ls' command. An arrow points from the text 'ls command' to the 'ls' command in the terminal. Another arrow points from the text 'A bunch of files and folders that happen to be in my current directory' to the red dashed box.

```
[(base) Wills-MBP-2:~ will$ ls
```

Applications	Music	hs_err_pid66044.log
Creative Cloud Files	Pictures	hs_err_pid66108.log
Desktop	Public	iCloud Drive (Archive)
Documents	Splice	pysal_data
Downloads	eclipse	seaborn-data
Library	eclipse-workspace	
Movies	getting-started	

A bunch of files and folders that happen to be in my current directory

Navigate to your Desktop

- The `cd` command allows us to move to different folders on our computer
- `cd` stands for “change directory”
- From the root directory, enter `cd Desktop` to change to your Desktop folder
- Use `pwd` command to confirm that you successfully moved to your Desktop
- Use `ls` command to list all of the items on your Desktop

Move up one directory level

- Change up one directory (folder) with:

```
cd ..
```

- That's `cd` followed by two periods

Create a new directory

- Create a new directory (folder) with:

```
mkdir myFolder
```

- Insert your desired folder name into **myFolder**

Common navigation commands

Windows CMD	Task	Mac OS Terminal
<code>dir</code>	List files and folders	<code>ls</code>
<code>cd</code>	Full path of current folder/directory	<code>pwd</code>
<code>cd <path to directory></code>	Change folder/directory	<code>cd <path to directory></code>
<code>cd..</code>	One directory up in directory tree	<code>cd ..</code>
<code>cd</code>	Move to root directory	<code>cd /</code>
<code>mkdir newFolder</code>	Create new directory in current directory	<code>mkdir myFolder</code>
<code>echo some-text > fileName(.txt)</code>	Create new file	<code>cat > fileName(.txt)</code>
<code>rmdir myFolder</code>	Remove a directory*	<code>rmdir myFolder</code>
<code>ren oldFolderName newFolderName</code>	Rename a directory	<code>mv oldFolderName newFolderName</code>
<code>robocopy myFolder <path to destination directory></code>	Copy a directory	<code>cp -r myFolder <path to destination directory></code>
<code>move myFolder <path to destination directory></code>	Move a directory	<code>mv myFolder <path to destination directory></code>
<code>del myFile</code>	Remove a file*	<code>rm myFile</code>
<code>ren oldFileName newFileName</code>	Rename a file	<code>mv oldFileName newFileName</code>
<code>copy myFile <path to destination directory></code>	Copy a file	<code>cp myFile <path to destination directory></code>
<code>move myFile <path to destination directory></code>	Move a file	<code>mv myFile <path to destination directory></code>
<code>cls</code>	Clear the terminal screen	<code>clear</code>

Familiarize yourself with the command line

- Print some text in the CLI with:

```
echo "Hello world"
```

```
.$ echo "Hello world"
```

```
Hello world
```

Download Git

- Download [Git](#) (64-bit, use default options) if you don't already have it
- Git is a distributed version control system
- Used to track changes in a set of files, usually used for coordinating work among programmers collaboratively developing source code during software development

Download Conda

- Download and install [miniconda](#) (64-bit, use default options)
- Conda is a package manager and environment management system
- It contains Python itself in addition to many other useful things

Create an account on Github

- Create an account on [Github](#) if you don't already have one
- Github is a popular site for hosting software projects
- Use your personal email address rather than university email
- Select the free account option

Use Git to clone the course directory

- Open your command line
- Navigate to your Desktop directory using `cd Desktop`
- Enter the following:

```
git clone https://github.com/willgeary/info615.git
```

- Now you have a copy of the course repository on your desktop

Navigate to the course directory

- Change into the main directory with:

```
cd info615
```

View the contents of the directory

- List the contents with:

```
ls
```

```
LICENSE  
README.md
```

```
assignments  
environment.yml
```

```
modules  
software
```

```
syllabus
```

View the contents of the directory


- List the contents with

```
ls
```

```
LICENSE  
README.md
```

```
  assignments  modules  
[environment.yml] software
```

```
syllabus
```



The **environment.yml** file provides the computer with some instructions to set up a Python environment for this course

Run these commands (only need once)

- Run these commands, one line at a time:

```
conda config --prepend channels conda-forge
```

```
conda config --set channel_priority strict
```

```
conda clean --all --yes
```

Create an environment

- Create the environment with:

```
conda env create --file environment.yml --force
```

Activate the environment

- Activate the conda environment with:

```
conda activate info615
```

Deactivating the environment

- If you want to deactivate this or any environment, you can do so with:

```
conda deactivate
```


Re-activate the environment

- Re-activate the conda environment with:

```
conda activate info615
```

Add Interactive Python support for Jupyter Lab

- Install a necessary dependence for interactive python:

```
python -m ipykernel install --sys-prefix --name info615
```

Launch Jupyter Lab

- Launch a Jupyter Lab interactive computing session with:

```
jupyter lab
```