Week 1: Weekly Videos and Curriculum

2. Command Line Interface (CLI)

Command Line Interface (CLI)

Comparison: Command Line Interface (CLI) and Graphical User Interface (GUI):

If we have had experience working with a computer, odds are that we have used a **Graphical User Interface**, or **GUI**. A **GUI** is the visual representation of data that users interact with via mouse and keyboard. If we open a window or an application and there are buttons and text boxes, that is a **GUI**. **Graphical User Interfaces** or **GUI**s are great for displaying data in an extremely user-friendly manner, however, they are not always the quickest, most effective way to work with computers.

Command Line Interfaces or **CLI**s are text-based tools that allow us to interact with a computer and data. Rather than visual elements, everything on a **CLI** is represented via plain text. Instead of clicking on different buttons to perform some sort of action, users type in text commands. This removes the need for a mouse and reduces the time spent to perform certain operations.

Why do we care? Many programs used in software development only come as a **Command Line Interface** or **CLI** tool and have no **Graphical User Interface** or **GUI** version. There are many different **CLI**s, but the main two are **Command Prompt** (Windows) and **Terminal** (Mac and Linux).

Note: We will discuss some basic commands to navigate the file system via **CLI**, but before we do, it is important to realize that what we see in the **CLI** is the **same thing** we see in the **GUI**. If we are in a **folder** (also known as a **directory**) in our **GUI File Explorer** and in our **Command Prompt** (Windows), or our **GUI Finder** and in our **Terminal** (Mac), we will see the same files in each, the **GUI** will display the files graphically, and the **CLI** will display the files textually.

How to open a Command Prompt (Windows) **or a Terminal** (Mac):

Windows: To open our **CLI** on Windows, press the `windows` key and type `cmd`. Select the **Command Prompt**.

Mac: To open our **CLI** on Mac, press `command` + `space` and then type `terminal` into the search bar. Select the **Terminal**.

Common Commands To Navigate the CLI:

Note: there may be different commands depending on whether you are using **Command Prompt** (Windows) or **Terminal** (Mac).

- `dir` (Windows) or `ls` (Mac) -- show what is in the current directory. To use this command, type `dir` or `ls` and hit enter. We will see a list of everything in the current directory.
- `cd` - change directory (Windows & Mac). There are a few ways we can use this command.
    - (Mac) If we type `cd` and hit enter, we will navigate to our home directory.
    - (Windows) if we type `cd` and hit enter, your current drive and directory path will be displayed
    - If we would like to go back one directory, type `cd ..` and hit enter. For example, if we are in Documents/Projects and want to go back to just Documents, `cd ..` would do that.
    - If we want to go back multiple directories by adding another set of two dots separated by a slash. For example, if we are in Documents/Projects/MyProject and we want to go back to Documents, we can type `cd ../..` to go back 2 directories.
    - If we are in Documents and there is a directory inside Documents named Projects that we want to navigate to, we can type `cd /Projects`.
    - We can also navigate directly to any directory by typing in `cd` followed by the complete directory and hit enter. This enables us to jump to any directory at any time given that we know the complete path.
- `mkdir` - create a new directory (Windows & Mac). To use this command, we type in `mkdir` followed by the name of the directory we want to create. For example, `mkdir projects` would create a directory named `projects` in whichever directory we run the command.
- `copy` (Windows) or `cp` (Mac) - creates a copy of a file or directory, or copies a file or directory from one directory to another.
    - If we are in a directory that contains a file named test.txt and we want to make a copy of this file and call it testcopy.txt, we can run `copy test.txt testcopy.txt` or `cp test.txt testcopy.txt`.
    - If we want to copy the file to another directory named Documents/Projects, we could run `copy test.txt Documents/Projects` or `cp test.txt Documents/Projects`.
- `move` (Windows) or `mv` (Mac) - moves a file or directory to another location. This works similarly to `copy` or `cp` except that it will physically move a file instead of just copying a file. If we want to move a file named test.txt to a directory called Documents/Projects, we would run `move test.txt Documents/Projects` or `mv test.txt Documents/Projects`
- `cd` (Windows) or `pwd` (Mac) -- displays the path of your current directory
- `echo` (Windows) or `touch` (Mac) -- can be used to create a new file. The following two examples will create a file named ***filename.***   **Note**: that the `touch` command creates an empty file.
    - (Windows & Mac) `echo "text to put into the file" > filename`
    - (Mac specific) `touch filename`