# Bash / Terminal Commands

These are the basic commands for navigating directories in a Terminal or Git Bash window

### Moving in directories

#### Change Directory

`cd [path/to/desired/directory]`

#### Change to Home Directory

`cd ~`

#### Move to One Directory Up

`cd ..`

#### View Folders and Files in the Directory

`ls`

#### Show the current Directory

`pwd`

#### Autocomplete a File Name in the Current Directory

Press `tab` key once to autocomplete once you have typed a unique portion of a file name.

### File Manipulations

#### Make New File

`touch [name of file to create]`

#### Make New Folder

`mkdir [name of directory to create]`

#### Delete file

`rm [name of file to remove]`

#### Delete folder

`rm -r [name of directory to remove]`

#### Copying File

`cp [filename1] [filename2]`

#### Move/Rename File

`mv [filename1] [filename2]`

### Mac Only

#### Open file or folder (Mac only)

`open [name of file]`

#### Open all files and folder in current directory (Mac only)

`open .`

### PC Only

#### Open file or folder (PC only)

`explorer [name of file]`

#### Open all files and folder in current directory (PC Only)

`explorer .`HTML

GitHub & Git Commands

* Will create different feature branches as you work on your project
  + Individuals will work on their own branches
* \*\*RULE\*\* Pull first, then push changes every time you push
  + This way you are always working with the latest version of the master branch
  + VSCode will show you where the conflict is (if any) between current version and master branch
* Master branch is the production branch \*\*NEVER work on the master branch\*\*
* git clone <SSH link from GitHub>
  + Clones the repository linked to the working directory
* git remote -v
  + Pulls up the origin of the project
* git add
  + Add files to the staging area
  + \* adds all
  + <file name> adds individual files
* git commit
  + Commit changes to git
  + -m allows us to put commit message
* git pull
  + Tells git to pull from the master branch on the remote repository
* git push <remote link> <branch>
  + Remote link:
    - Origin (remote GitHub link)
    - Heroku (remote Heroku link)
  + Branch - branch that you’re working on
  + \*\*Pull before pushing\*\*
  + \*\*Only push working code, not broken code\*\*

Terminal/GitBash Commands:

* ls
  + List all files/folders in current working directory
* cd <path>
  + Change directory (without any modifiers takes back to the root)
* <Tab>
  + Pressing tab autocompletes the name of a file/directory in working directory
* pwd
  + Print working directory
* cd ..
  + Goes back to previous directory
* cd ~
  + Go back to root directory
* mkdir <name>
  + Creates a new directory with given name
  + Stay away from including a space in directory names
* touch <name+extension>
  + Creates a new file with the given extension
* rm <filename+extension>
  + Removes a file from the current directory
* rm -r <directory>
  + Remove a directory
* mv <path of file 1> <path of file 2>
  + Move a file from one place to another
  + Can also be used to rename a file
* cp <file1> <file2>
  + Copy information from one file to another
* help <command>
  + Gives you the information for what a command does and any modifiers
* open <folder> (FOR MAC)
  + Opens the folder in the explorer
* explorer <folder> (FOR WINDOWS)
  + Opens the folder in the explorer
  + \*\* Works on command prompt but not on GitBash\*\*