UNIX Cheatsheet

> **This document is NOWHERE NEAR COMPREHENSIVE.** To learn more, visit Dr. Spielman's website and see "Resources"!

Directory

- A *folder* on your computer which contains files. UNIX filesystems are organized as hierarchical directories.
- Forward slashes divide levels in the nested hierarchy of directories, e.g. `/top_level_directory/second_level_directory`
- The directory at the top of this hierarchy is called the **root** directory and is denoted simply as `/`.

Path

- The *address* to a directory or file on your computer. There are, generally, two types of paths:
  - **Absolute/full path** represents the path of a given directory/file beginning at the root directory.
  - **Relative path** represents the path of a given directory/file relative to the working/current directory.
- Your **home directory** is the top-level of *your account* (`<>` signifies PLACEHOLDER text):
  - **Mac:** `/Users/<yourusername>/` (for me, `/Users/spielman/`)
  - **PC**: `C:\Users\<yourusername>\` <– NOT UNIX!!
  - **Rstudio Cloud Project**: `/cloud/project/`
  - **Linux**: `/home/<yourusername/`

Basic UNIX commands

Command	Description	Examples

`cd`	**C**hange **d**irectory	`cd Desktop`
		`cd .. # one directory back`
		`cd  # shortcut to go home`
		`cd ~ # shortcut to go home`

`ls`	**Li**s**t** files and directories in current directory.	`ls`
		`ls -l # display long`
		`ls -a # show hidden files`
		`ls *Rmd # list all files that end with "Rmd"`

`pwd`	Display the path of the current directory (aka **p**rint **w**orking **d**irectory)	`pwd # yup that's it!`
`rm`	**R**e**m**ove a file or directory (use `rm -r`). **Warning: This is PERMANENT!.**	`rm file_I_dont_need.txt` `rm -r directory_I_dont_need/`

`cp`	**C**o**p**y a file or directory(use `cp -r`)	`cp oldfile.txt newfile.txt`
		`cp oldfile.txt .. # make a copy of the file to live one directory up`
		`cp -r first_directory second_directory`

`mv`	**M**o**v**e or rename a file or directory (*original file is changed* - like "cut" aka ctl+X)	`cp oldfile.txt newfile.txt` `cp oldfile.txt .. # MOVE the file to live one directory up`
`mkdir`	Create a new directory (aka **m**a**k**e **dir**ectory)	`mkdir new_directory`
`touch`	Create an empty file.	`touch new_blank_file.txt`
`cat`	Display contents of a file (aka con**cat**enate. this command can do more than this FYI)	`cat file_whose_contents_i_want_to_see.txt`
`head` and `tail`	Display the first or last 10 lines of a file.	`head file_whose_contents_i_want_to_see.txt`
`man`	Display documentation about a command (aka **man**ual). These are not well-written. You have been warned. *To exit just press* `q`	`man ls # look up how to use ls`
`clear`	**Clear** screen (simply scroll up for past commands). Can also use Ctrl + L	`clear # yup that's it!`
`history`	View history of recent past commands	`history # yup that's it!`

Shortcuts and symbols

Shortcut/symbol	Description
`Ctrl + C`	Kills current process/command
`*`	wildcard character
`tab` (tab key)	autocomplete word
↑ (up arrow)	scroll back through previous commands
`.`	Current directory
`..`	One directory level up
`~`	Home directory