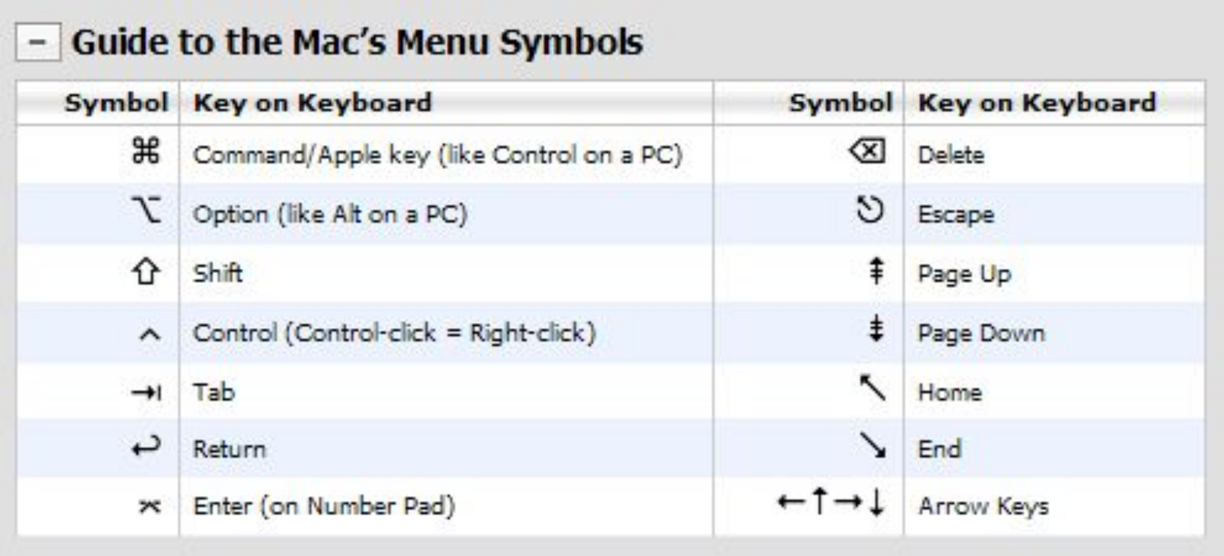
HTML: Hyper Text Markup Language
    Hyper Text: text with links
    Markup:
JavaScript: basic language for adding interactivity
CSS:

Cmd + Space: opens terminal

Mac Keyboard Symbols:



− Guide to the Mac’s Menu Symbols

Symbol	Key on Keyboard	Symbol	Key on Keyboard
⌘	Command/Apple key (like Control on a PC)	⌫	Delete
⌥	Option (like Alt on a PC)	⎋	Escape
⇧	Shift	⇞	Page Up
⌃	Control (Control-click = Right-click)	⇟	Page Down
⇥	Tab	↖	Home
↩	Return	↘	End
⌤	Enter (on Number Pad)	←↑→↓	Arrow Keys

Homebrew

- brew search ____ : search Homebrew’s available packages for an application
- brew info ____ : gives details on the application
- brew install ____ : installs the application
- brew remove ____ : uninstalls the application
- Cask: an extension to Homebrew which allows it to install graphical applications
    - brew cask install _____

Visual Studios Code

- Can now type “code” into terminal to open VS Code
- You can have multiple editors open side-by-side, vertically or horizontally
    - Cmd + \
- Cmd + P : quick open
- Command Palette:
    - Shift + Cmd + P

    - Gives access to all functionality of VS Code (keyboard shortcuts, etc)
    - Cmd + P: navigate to any file or symbol by typing its name
    - Ctrl + Shift + Tab: cycles you through the last set of files opened
    - Shift + Cmd + P: brings you directly to the editor commands
    - Shift + Cmd + O: lets you navigate to a specific symbol in a file
    - Ctrl + G: lets you navigate to a specific line in a file
    - Typing “?” (minus “”) into the input field returns a lost of available commands you can execute from here
- Shortcuts:
    - Option + Cmd + —> : go to the right editor
    - Option + Cmd + <— : go to the left editor
    - Ctrl + Tab: open the next editor in the editor group MRU list
    - Ctrl + Shift + Tab: open the previous editor in the editor group MRU list
    - Cmd + 1: go to the leftmost editor group
    - Cmd + 2: go to the center editor group
    - Cmd + 3: go to the rightmost editor group
    - Cmd + W: close the active editor
    - Cmd + K + W: close all editors in the editor group
    - Cmd + K + Cmd + W: close all editors
    - Shift + Cmd + X: extensions view
- Emmet Snippets: used as shortcuts to write longer code
    - Ex: “ul>li*3>span.hello$” —> 3 rows of list items that say “hello”

Terminal

- Pwd: command that tells you where you are in the computer’s file system
- Tar: archival command that can archive, compress, and extract files (needs to be told which files to take action on, what to do with them, and where to put them)
    - -c: create; -z: zip; -f: file
        - Can also combine into -czf
- Man: short for “manual”; use to find out which flags a command uses and what they mean
- ~ : home directory

Command	**Description**
cd	Change directory.
ls	List files and directories in current directory.
pwd	Display the path of the current directory.
touch	Create a file.
mkdir	Create a directory.

rm	Remove a file or directory. Warning: deleting a file or directory with this command is permanent!
cp	Copy a file or directory.
mv	Move or rename a file or directory.
echo	Print text to STDOUT.
cat	Display contents of a file.
more	Display contents of a file, starting at the top and letting the user scroll down.
less	Display contents of a file in an even more interactive way.
head	Display the first part of a file.
tail	Display the last part of a file.
man	Display documentation about a command.

- Ex: mkdir practice: makes a directory named “practice”
- Moving or renaming a file: mv [source] [destination]
- To remove a folder and all its contents, you need to specify the “-r” (recursive) option
    - Ex: rm -r practice
- Type “q” to exit “more”/“less”
- / - The root directory or a separator when listing directories
- . - The current directory (also ./) or the same level
- .. - The directory one level up (also ../)
- ../.. - Two levels up
- ~ - Your "home" directory, or the directory you are placed in when you log in.
- * - The "splat" or "glob" operator. This is the wildcard of the command line and represents "any characters."
    - Ex: “*ot” represents any file or directory that ends in “ot”
- the presence of a leading “/“ changes the meaning of the path
- Executables:
    - Different from other files because:
        - They have special characters at the beginning to tell the computer how to execute them
        - They have scripts or machine language as their content
        - They have executable permission
    - Ls -a : lists all files (including hidden ones) in a directory
        - Use “ ls -d .* “ to list only the hidden files (“dotfiles”)
    - Ls .. : lists a parent directory’s files
    - Difference between “ abc/ “ and “ /abc “

  - The path “abc/“ (also “./abc” and “./abc/“) are paths relative to your current working directory
  - The path “/abc” is the file or directory “abc” in the root directory
- “../abc” is a reference to the parent directory (..) and the file/directory called “abc” within that directory
- Suppose you are in a directory with 7 files. You need to move 6 of them into a "temp" directory, that you have yet to create. How do you do that?

  Solution
  Let's break this up into a few steps:
  First, create the temporary directory:

  ```
  $ mkdir ~/tmp
  ```

  The fastest way to move all but one file is to move all the files, then move the one file back. Assuming the one file that you don't want to move is 1.file, the following commands will do the trick:

  ```
  $ mv * ~/tmp       $ mv ~/tmp/1.file ./
  ```
- Use the “ -lah “ set of flags as arguments tp the ls command to get more detailed information about files and directories
  - “Ls -lah”
- You can use the “which” command to figure out a default executable’s location
  - Ex: which touch