**Class Notes**

**Questions**

* Are there pros/cons to writing styles in the head section of HTML vs. linking to CSS?
* section {
* padding: 20px;
* margin-bottom: 20px;
* background: #FFA500;
* }

Why does this affect only the middle section?

**Class 3: Saturday November 17th**

How to style HTML?

* Elements
* Classes
* IDs

Relative vs. absolute file paths

Box Model

* box-sizing: makes it easier to make a box model the size you actually want it to be
  + e.g. make my box 400px x 400px, and factor in the margins/padding/borders from there to end with a 400px x 400px
  + benefits of this: if I need to create a page with specific-sized boxes, I can use box-sizing instead of having to calculate heights/widths including padding/margins/borders
  + <https://www.paulirish.com/2012/box-sizing-border-box-ftw/>

Floats

* Block elements vs. inline elements

CSS positioning

Position types

* Fixed
* Absolute
* Relative

Expand on overflow

Z-index: use large gaps to give yourself room to add elements in between

e.g. use 100 and 200 instead of 1 and 2

Flex Box

<http://jonibologna.com/content/images/flexboxsheet.pdf>

<https://www.youtube.com/watch?v=jV8B24rSN5o&feature=youtu.be>

<https://www.youtube.com/watch?v=jV8B24rSN5o&feature=youtu.be>

**Class 2: Wednesday November 14th**

Working with GitHub

What is GitHub?

* A web-based platform that stores code online
* Serves as a backup to local machines
* Useful tool for collaborative coding
* Users pull (download) or push (upload) code to/from a GitHub repository
* Major benefit of working through GitHub: version control

Version Control

What is it?

* Developing code in installments (think of it like saving a document in multiple versions as you work on it) 🡪 Version 1, Version 2, etc.

Why is it helpful?

* Version control makes it a lot easier to work collaboratively
  + Easy to track who has worked on what
  + Easy to ensure you’re working with the latest code
* Segmenting your code makes it clearer which portions of code are creating problems
  + E.g. if Version 3 worked fine but Version 4 has problems, the coding errors reside in what was added in Version 4

Basic Git commands

* git clone = download a repository from GitHub to your local directory
* git add = first step, prepares a file to be committed to the GitHub repository
* git commit = commits a file to GitHub repository (use “-m” to write a message annotating the change)
* git push = sends changes to the online GitHub version
* git pull = downloads the latest version of a GitHub repository

HTML

* Headings factor into SEO
* HTML tags resource: <http://www.w3schools.com/tags/>

CSS

* when modifying elements using CSS, use a period for classes and a hashtag for IDs
  + e.g. if a class is called ‘container’, we’d refer to it as .container in CSS
  + e.g. if an ID is called ‘main\_bio’, we’d refer to it as #main\_bio in CSS

**Class 1: Monday November 12th**

# Bash / Terminal Commands

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

*Moving in directories*

* Change Directory = cd
* 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]
* Open file or folder (PC only) = explorer [name of file]
* Open all files and folder in current directory (PC Only) = explorer .