GitHub Notes

- Repo means repository which is where the entire project is stored
- Branches are like different sections of the repo where work can be done, but we are supposed to just use the main branch and make changes to that

How to use the terminal with GitHub:

- cd name-of-folder: to go to folder
- git clone link: to clone repo
- git pull: to get all updates
- git add . : to add all changed files
- git commit -m: to commit all changed files
- git push: to push all changed files to GitHub and then changes show up on other people's end

GitHub problems I've encountered:

- My computer's terminal asks for my GitHub username so I type that in and then it asks for the password but after entering that it gives an error saying that this functionality was removed a few years ago
  - To combat this issue, take all the files that have new changes and drag them into the correct GitHub folder
- Terminal gives me an error when I try to pull information from the repo every class, not sure why or what it means
  - To combat this issue, re-clone the repo into a new folder

W3 Schools HTML Tutorial Notes

HTML Introduction:

- HTML stands for Hypertext Markup Language. It is the standard markup language for creating web pages.
- Markup consists of tags that define the elements of a page.
- Example:
  - A simple HTML segment demonstrates the structure of a basic webpage; includes tags like <html> <head> <title> <body> <h1> <p> <img>.
- Tags:
  - Keywords enclosed in angle brackets that define elements.
  - Come in pairs: opening and closing tags, <> // <> with content nested between them.
  - Attributes provide more info about elements
- Web Browsers:

    - Web browsers interpret HTML code and display web pages to users (Chrome, Firefox, Safari, and Edge)
    - This is what happens when I click open with live server
- HTML Editors are like what I use VS Code
- Basic Syntax:
    - HTML documents start with a document type declaration <!DOCTYPE html>.
    - <html> element has entire HTML content
    - <head> has meta-info about the document
    - <title> sets title displayed in browser's title bar
    - <body> has the visible content of document

HTML Editors/Basics:

- <p>: paragraph
- <a>: links
- <img>: images
    - Ex: <img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">
    - Self-closing, doesn’t require closing tag

Elements:

- Element is everything from start tag to end tag
- Attributes provide more info about elements and are specified in the opening tag.
    - Written as name/value pairs, like attribute="value"
    - Like the img tag example above
- More tags to create elements:
    - <div>: Division
    - <span>: Inline container
    - <table>: Table
    - <form>: Makes HTML form
    - <input>: Input fields for forms
    - <button>: Clickable button
- Semantic elements have meaning regarding the content they contain
    - Ex: <header>, <nav>, <main>, <footer>, <section>, <article>, <aside>, etc.
- Comments: <!-- comment goes here -->.

Attributes:

- Common examples:
    - id: specifies a unique identifier for an element
    - class: specifies one or more classes for an element, which can be used for styling with CSS

    - style: allows inline CSS styling for an element
    - title: provides a title or tooltip for an element
    - href: specifies the URL of a hyperlink
    - src: specifies the URL of the resource, such as an image or script
    - alt: specifies an alternate text for an image, displayed if the image can't load
    - width and height: sets width and height of an element, like an image or table
    - disabled: disables an input element or button
- Global attributes can be used with any HTML element
- Event attributes define JavaScript to be executed when an event happens
    - Ex: onclick, onmouseover, onfocus, onblur, etc
- Attributes can be single or double-quoted
- Data attributes, prefixed with data-, allow storing custom data for an element
    - Can be helpful for scripting or styling

Format:

- Headers can automatically be smaller in size as they go if you number them properly so <h1>, <h2>, <h3>, and so on will get smaller as it goes
- <p> tags display while ignoring any newlines or spaces so
    - <br>: line break
    - <pre>: preserves line breaks and spaces in the given paragraph so displays as typed

HTML Styles/CSS:

- Style attribute is used to add styles to an element, such as color, font, size, etc
- Inline styles are applied directly to individual HTML elements using the style attribute.
    - <tagname style="property: value;">
- Internal styles are within the <style> element in the <head> section
- External styles are defined in separate CSS files and linked to HTML documents using the <link> element.
    - This is what our project is doing
- Examples:
    - Use the style attribute for styling HTML elements
    - Use background-color for background color
    - Use color for text colors
    - Use font-family for text fonts
    - Use font-size for text sizes
    - Use text-align for text alignment
    - Use the CSS border property for borders
    - Use the CSS padding property for space inside the border
    - Use the CSS margin property for space outside the border

- HTML documents can link to multiple external CSS files for different styling purposes or one CSS file can be used for multiple HTMLs

Div:

- <div> elements are commonly used to group related content together like sections of a webpage; provide a way to structure and organize content within a webpage
- can contain any other HTML elements
- CSS float property was not originally meant to align <div> elements side-by-side but has been used for this purpose a lot

Classes:

- Classes used to apply styles or behaviors to multiple elements
- Class attribute is added to HTML elements to assign one+ class names to those elements
    - <tagname class="classname1 classname2 ...">
- Classes are commonly used in conjunction with CSS to apply styles to multiple elements with the same class so use to class names begin with a dot (.) followed by the class name.
- To define multiple classes, separate class names with a space
    - <div class="city main"> so belongs to both the city class and also to the main class
        - Element will be styled according to all the classes specified.
- Different elements can have the same class
- class name can be used by JavaScript to perform tasks for specific elements

My contributions to the project:

- **Quick Intro:** I don’t have any coding experience except for taking CS 1 & Data Structures but really wanted to get more hands-on with my learning!
- **Progress:** Learned GitHub, learned HTML, Created sketches for the front-end pages, Worked on the front-end dining hall pages, Came up with the idea for implementing a dietary restrictions option and tried to implement it but didn’t really work so was removed for the final presentation
- **Methods:** Mostly Used HTML to add features to the dining hall pages, attended a workshop and learned a lot more about Figma (much later after making website sketches)
- **What I learned:** I learned the basics of HTML enough to feel comfortable applying some basic features like buttons, titles, etc which helped improve the layout of the site.