8/1/17

CSS

* Day 1
* Learn to create CSS files and hook them up to your HTML pages
* CSS – Cascading Style Sheets
* Language to *style* the HTML content on web pages
* Helps modify colors, font types, font sizes, shadows, images, positioning, etc…
* Added “<link href="style.css" type="text/css" rel="stylesheet">” to the head and the website became pretty
* You can write CSS directly in HTML code using *inline styles*
* Add the “style” attribute to and HTML element
  + <p style="color: red;">I'm learning to code!</p>
  + <p style="color: red; font-size: 20px;">I'm learning to code!</p>
  + Inline styling is good for one-off styling, but if you want to do it
* There is also the <style> element that can go in the <head> of the HTML doc
  + <head>
    - <style>
    - p {
    - color: red;
    - font-size: 20px;
    - }
    - </style>
  + </head>
* Using the <style> element in the HTML doc is not a great move because it leads to overcrowded, complex code that’s hard to read.
* Create a style.css, or another .css file you name whatever
* So now you have two files: a .css and a .html… you need to link them to make the HTML file locate the CSS code and apply it
  + Use the <link> element in the head of the HTML file
    - Href – must be the address/path to the CSS file
    - Type – indicated the type of file linked. In this case, type “text/css”
    - Rel – describes how the linked file is related. In this case, write “stylesheet”
    - <link href="./style.css" type="text/css" rel="stylesheet">
* **CSS Selectors**
* I will learn how to select HTML elements using CSS, allowing me to modify the page’s appearance
* To style an HTML element, CSS first has to select it with a Selector.
* CSS follows a 2 part process
  + Select HTML elements
  + Apply styles to elements
* Selectors
  + Tag name – the word or character in between angle brackets in HTML. For example, <p> for paragraph. The CSS syntax for selecting paragraphs is
    - p{
    - }
  + Class name – a class is an attribute of HTML elements (similar to the id attribute on a div)
    - So when writing HTML… <p class= “title”> Words words words </p>
    - To style a class in CSS:
      * .title{
      * }
    - .[className]{ }
  + Multiple Classes
    - <h1 class =”green bold”> … </h1>
    - Both the rules for bold and the rules for green classes would be applied
    - Separate the classes with a space
  + ID Name – If an element needs to be styled uniquely (regardless of its class), add an ID.
    - <h1 id=”large-title”> … </h1>
    - To select an id with css, use “#[idName]{ }”
    - #large-title{
    - }
  + IDs override the styling of tags and classes
  + Chaining Selectors
    - You can combine multiple classes and tags into a single selector. Ex:
    - h1.special{
    - }
    - That selector would only style h1 elements that had the class special. So a <p> with the special class wouldn’t be styled
  + Selecting nested elements
    - CSS can specifically select list items
    - HTML:
      * <ul class=”main-list”>
        + <li>…</li>
        + <li>…</li>
        + <li>…</li>
      * </ul>
    - CSS:
      * .main-list li{
      * }
  + One selector that is even more specific than IDs: **!important**
    - Will override any other style, no matter how specific
    - P{
    - Color: blue !important
    - }
    - Will even override <p>s in a .main class if you have a specific selector choosing .main <p>s
    - Avoid using this!
  + Can define multiple selectors at the same time with a comma:
    - .menu, h1 {
    - }
    - Will be the style rules for both menu class and h1 tags

8/1/17

* Day 2: Quiz on yesterday’s concepts and then learning about visual CSS rules
* Got a 8/8 on second try for quiz1 (CSS Setup).
* Got a 6/6 on first try for quiz2 (CSS Selectors).
* Continued on 8/2…

8/2/17

* CSS Visual Rules
* CSS declarations have a property and a value
  + Property: size, color, etc
  + Value: 18px, blue, etc
* Semicolon always used at the end of a declaration
* Font-family changes the font or typeface of an element
  + The font specified in a CSS stylesheet must be installed to display
  + If no font-family is defined, Times New Roman is used by default
  + Limit to 2 or 3 typefaces on a page
  + If the name of a font family has more than one word, put the whole name in quotes like “Open Sans”
* Font-size changes font size
* Font weight: normal or bold
* Text-align
  + Left, right, or center
* Color
  + Color: sets the foreground color (the text)
  + Background-color changes background color (the space behind the text)
* Opacity
  + Value between 0 and 1
  + 1 is fully visible, 0 is invisible
* Background image
  + Background-image: url(“<https://www.example.com/image.jpg>”);
  + Even if linking to another file in the directory, still use “url”
* Day 3: Quiz on yesterday’s concepts
* Got a 7/7 on third try for quiz (CSS visuals).
* **Project** – Customize a recipe website that needs styling
* Read article on how to create HTML/CSS project on my computer
* Day 4: Olivia Woodruff – fictional developer, let’s make her portfolio look great
* Guided exercise, not on my own yet
* Article: teaching about how to use Chrome devtools with CSS

8/3/17

* Day 5
* Complete an HTML/CSS page on my own computer. “Beach Paradise” project
* Day 6
* Create a fully functional HTML / CSS site. No guidance
* Dasmoto’s Arts and Crafts project
  + Submit for review!

NOTES BELOW ON SUBMITTED PROJECT

Notes on submitted project

* To make the “&” (ampersand) symbol in an HTML page, type “&amp” instead of just “&”
* I should try to save images locally instead of using the URL to the image on another site
  + This will improve stability and doesn’t rely on the external site being up and running
* Use CSS comments because it’s not as easy to understand just by reading like HTML is