Useful notes CSS

* Units of measurement
  + Width, margin, font-size, etc.
  + px; em; or %
    - no space between number and unit (15em)
    - 1em = 24px on standard page
    - 1em = 100% of standard font size
    - 1.5em = 150%
  + absolute measurements – are same size no matter how large the page is
    - px (pixel)
  + relative measurements – scale based on certain things
    - em or %
    - better for programming because it will work with phones or computers
* Block vs Inline Elements
  + Every element is inside its own invisible box – the box can be a block or inline space
    - Boxes – default take up 100% of the space they have – don’t allow other elements to share the line they are on
      * Ex: div, header, main, footer, nav, p, ul, li, section, article, form, h1-h6
    - Inline – allow other elements to share the same line until its full
      * Ex: a, img, span, button, label
    - Turn a block into an inline by: target the code in css with {display: inline;}
    - Turn an inline into a block by: target the code in css with {display: block;}
    - You can also make something an inline-block: {display: inline-block}
  + Centering
    - Text-align: center
      * Content is centered inside of invisible box
    - Margin 0 auto:
      * The box itself is centered
      * If it takes up the whole page, you won’t see it unless you give the box itself a width
      * The first says to set the top and bottom margins to 0 and the second says to set the left and right margins to auto: meaning to split any margin available between the left and right.
    - For images, you need to make it a block or put it inside a block level element, then center it with margin 0 auto;
  + CSS Selectors and Specificity
    - Can target in CSS with many things; p, h1, div, ID names (#[name]), class names (.[name])
    - You can also be specific on the order the code was written
      * Useful for fonts; you can set a font for the whole page at the top of the body, but then later on you can give a h1 a different font that will override the base font, because it comes later in the code
    - Specificity
      * Most specific > ID > class > elements (p, div, h1) – more specific will override less specific
    - Can select multiple things with commas
      * Ex: h1, h4 {font-family: ‘Lemon’}
    - Can select all things within an element
      * Ex: article p {font-family: ‘Lemon’}
        + This will only select elements of “p” under the parent element article
    - Pseudo-selectors
      * You can just select the nth element
        + Ex: li:nth-of-type(2) {background-color: #ccc;}

2 selects the 2nd element within a list

* + - Pseudo-classes
      * Allow to change appearance of an element when a user is interacting with it
        + Hover over a button and it changes color
        + Ex: nav a:hover {color: black; background-color: blue} button:hover {background-color: white}