

## In-Class Exercise 5:

### Adding CSS Styles to your website

Next, we'll adjust the font on our website to make it look nicer. Follow the steps below to modify the appearance of text on your website. Add the CSS rules you write to *style.css*.

1. First, choose a pair of fonts that you think will look good together from [www.dafont.com](http://www.dafont.com). One of these fonts will be for <h1> through <h6> elements, and the other font will be for <p> and other text elements in your pages.
2. Create a new folder inside your website's root folder. Call it *fonts*. The *font* folder should be in the same folder as your *css* folder.
3. Download the fonts you selected and put them into your *fonts* folder. Your font file should end with the extension *.ttf*.
4. Using the information on page 9 of the course notes, load the fonts using the `@font-face` command. Remember that the relative path inside of *url()* is the location of the *.ttf* file relative to the file in which the CSS rule is being written. So if you are writing these rules inside an external stylesheet in your *css* file, you'll need to go up one directory (using the `'..'` syntax), then into your *font* folder, before finally specifying the name of the font file.
5. Repeat the last step to load in the second font.
6. Create a CSS selector to target the <h1> through <h6> elements on your pages. You'll need to use the comma (',') to select multiple types of HTML elements. Add style rules to do the following. You may choose values that you think make your page look good:
  - a. Use one of the fonts that you downloaded from the web (font-family).
  - b. Adjust the size (font-size)
  - c. Adjust the boldness (font-weight)
  - d. Adjust the font color (color)
  - e. Transform the text so that it is all upper case (text-transform). You can remove this later if you do not like how it looks.
7. Next, create a CSS selector to target all the other text on your page. You'll need to use the comma (',') to have your style rules apply to multiple types of elements. Apply these style changes to <p> and <li> elements. You may choose values that you think make your page look good:
  - a. Use the other font that you downloaded from the web (font-family).
  - b. Adjust the size (font-size)
  - c. Adjust the font color (color)
  - d. Change the leading (line-height)
  - e. Change the letter and word-spacing (letter-spacing, word-spacing).
8. Next, let's try and make our hyperlinks a little bit nicer. We will do this by removing the default underlining style and choosing a new color for them. Create a new CSS rule that will target <a> elements and create style rules to do the following:
  - a. Remove the underlining (text-decoration)
  - b. Change the color from blue to something more pleasing (color)

9. Next, let's do something fancy to the first letter in each of our <p> elements. We can target the first letter of an element specifically by using something called a *pseudo-selector*. To use a pseudo-selector, we take a regular selector (like *p*) and we add a colon (':') and a text-description of the pseudo-selector to the end of the regular selector. A CSS pseudo-selector targeting the first letter of a <p> looks like this:  

```
p:first-letter {...}
```
10. Add a pseudo selector to change the appearance of the first letter in each <p> element. Make it:
  - a. A little bit larger
  - b. A slightly different color
11. Now, let's use pseudo-selectors to make links look nicer. Two other pseudo-selectors relevant to <a> elements are :visited and :hover. The first selects hyperlinks that have already been visited (browsers usually show these in purple by default). The second applies rules to an element only when the mouse is hovering over it. Using these pseudo-selectors, create CSS rules so that your <a> elements appear:
  - a. A slightly different color after they have been visited (color)
  - b. Slightly more bold when the mouse is hovered over it (text-weight).

When you are done, save your files. Call the instructor over to take a look at your site. No need to turn this in yet.