IST 263

# Lab 06

## This lab covers:

* Wireframes
* CSS linked stylesheets
* CSS Selectors and Declarations
* CSS Units of Measure
* Inheritance and DOM
* CSS Formatting Text and Specific Selectors

## PEER Review of partner’s copy document

This week I will give instructions on peer reviewing another student’s website copy document. More details in lab class with questions below.

## WIREFRAMES

Wireframes help define layout obscuring the issues of design fonts, colors and graphics. A wireframe is an important planning document. We are going to practice wireframing by looking at an existing website and drawing the elements on the page.

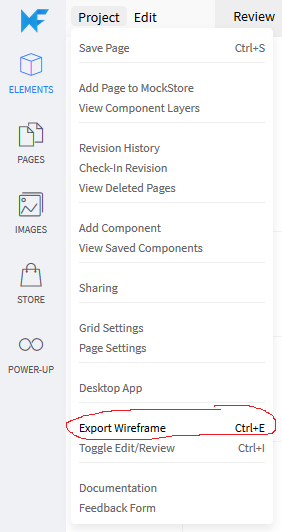
We are going to use MockFlow to get some practice making WireFrames. The first thing you need to do is create a free account.

1. Go to: <https://www.mockflow.com/signup/>
2. Fill out the form with your info.
3. Click the New Project button at the top left and choose the Web 5 icon.  
   
4. Type a project name and click the start button.
5. Scroll the list of elements on the left side and familiarize yourself with the options. There are elements for images, tables, form elements, text, etc.

We are going to practice wireframing a typical page. Put all the images, buttons, etc. **above the Customer Reviews box** into your wireframe. Here’s the page you will practice on:

<https://shop.delimacoffee.com/collections/coffee/products/classic-12-oz>.

Note the different elements to their page. They have a logo, product image, a form on the right and more.

1. Use MockFlow or equivalent layout software to draw a wireframe of this entire page. Mockflow only allows one wireframe to be built on the free account so you can delete the wireframe we made during lecture.
2. When you are done, you can screen shot your wireframe or, click Project –> Export WireFrame. Choose images and add the page you've been working on to the Added Pages Area. Follow the onscreen instructions to download the png image file. Save it in the lab06 folder you created above. You will be turning this in.  
   

## Lab Setup

1. Create a new folder in your Github repository called lab06. Place a copy of your skills page from lab04 and your bio and contact pages from lab05 into the lab06 folder. One way can do this is by copying your files in the repository to your new lab06 folder.

## CSS Linked Stylesheets

CSS (cascading style sheets) set rules for the looks of your website. There are three ways to put those rules into action.

|  |  |  |
| --- | --- | --- |
| **Type** | **Example** | **Note** |
| Linked/External |  | Style tags in a separate file specified by link element inside head. |
| Embedded |  | All style rules in style element inside head tag of your document. |
| Inline |  | Style rules in the elements in the body of your web page. |

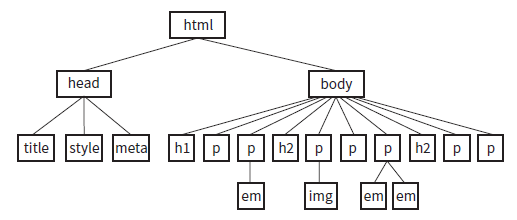
We will not use inline styles as this is not a web developer industry best practice. That means you should never have a CSS rule anywhere inside the body tag. When we can, we’ll use linked stylesheets. Why are linked stylesheets the best? You can make one stylesheet and use it on hundreds of website pages. Because you are using the same styles everywhere, it makes the looks of your website very consistent. It also means faster page loading as the stylesheet will be downloaded by the browser once, cached on your computer and used repeatedly.

1. Start by creating a CSS file in your lab06 folder called styles.css. Be careful if you do this in VS Code. Pick the CSS file type from the drop down or you could end up with the wrong file extension.
2. Use the example in the above table to add your external style sheet to all three of your html pages (bio, skills and contact). The value for the href attribute must be a relatively link to the styles.css file you just created so think about what that path will be.

## CSS Selectors, Declarations, Inheritance and text Formatting

Let's start by picking a font for your entire portfolio website. W3Schools has a listing of "Web Safe Font Families" – these are fonts you can expect most folks to have on their computer. [Take a look here](https://www.w3schools.com/cssref/css_websafe_fonts.asp) and pick a set you like.

Because of the DOM every tag inside the body will inherit the font you choose. The style will cascade to the other elements. For example, in the graphic shown below the body tag contains h1, p, h2, em and img tags. All of them will have their font face changed by the rule we write below if applicable.



1. Your styles.css file should be completely blank right now. In your styles.css file, add a selector and declaration for the font family you chose. It's important that you type the font family exactly as shown on W3Schools because some specific fonts need quotes around them and the more generic do not. Recall that the format for all CSS rules is the following:



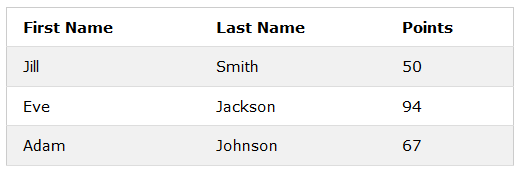
Replace selector with body, replace property with font-family and replace value with the fonts you chose.  
  
Warning: The styles.css file can only contain CSS rules. You cannot put HTML tags or text on the CSS page. All your HTML tags must be in the .html pages.

1. Pick a color for your headings other than black. A list of color names is located on the [W3Schools website here](https://www.w3schools.com/colors/colors_names.asp). This will be an example of selecting multiple elements because we are going to style all the h1 and h2 elements on all the pages. Recall that if you have more than one selector you can group them in the selector area with commas in between like this:



1. Go back to the styles.css file and add a new grouped selector to it under the font-family rule. In our case the selectors are h1, h2. The property is color and the value is whatever color name you chose in the last direction.

Next we will work on your skills page. Currently it contains one table. We are going to turn it into a zebra table. What is that? It's a table that stripes every other row with a background color so that it's easier for users to see the differentiation between the rows. Here's an example of a zebra table.



If we tried to style the tr tag, we'd change the color of every row in the table. We don't want that. Instead, we’ll use a class selector. Class is an attribute that you can add to any element. It names the element for the purpose of styling it. Here's an element with the corresponding class selector. Note the period before the class name in the stylesheet. That period is how the web browser knows to look for a class attribute to apply the style to instead of an element.

Class Selector Example

|  |  |
| --- | --- |
| In your html page | <h1 class="special">Lorem Ipsum</h1> |
| In your CSS linked stylesheet | .special {color: purple} |

1. To style every other table row in the skills table, we need to add a class attribute to all the even table rows (tr tags). Add the attribute class="evenrow" to these rows.
2. Now we are going to change the background color of those rows with a rule in styles.css. Add a rule to styles.css with the selector .evenrow, the property background-color and pick a color value from the color names that we used above.

## Units of Measure Inheritance and DOM

CSS has different units of measure for specifying the size of fonts. We are going to use the em measurement. The browser has a default size for each element. That default size is specified as 1 em. 2 em would specify the browser to render the text twice the size of default. .5 em would tell the browser to render the text half the size of the browser default.

1. Let's change our heading font sizes for all pages (h1 and h2). Different fonts display at different sizes so I'm going to let you play around with this one and decide what the best size is for your font-family. We are going to write separate rules for h1 and h2.
   1. For the first rule your selector should be h1, your property should be font-size and your value should be x em <- where you x with the size number you choose.
   2. Repeat the same process for h2. It's customary to make the h2 heading slightly smaller to show the user that is less important.

## What will You hand IN?

Create a word document, pdf or use the "write submission" option in blackboard to provide the following:

1. To turn in your wireframe, submit the GitHub URL for the wireframe image you exported above. How do you get a GitHub URL for the wireframe? Put it in the images folder in your repository and commit it. Once it’s there you should be able to create a path to it.
   1. <https://coconnol-su.github.io/ist263/images/wireframe.png>

1. Submit the GitHub URL for all three pages from this lab plus the URL for the CSS file.
2. In your book on pages 301-302 are a list of font and text properties. Pick a property off this list that we did not use in either the lecture or lab. Explain how you could see this property being applied to something on either the bio, skills or contact pages to make the design better.
3. Hand it the peer copy document review with answers to these questions. Note that I have read all the copy documents, and everyone has something to improve. Don’t say everything is great.
   1. Who’s copy document did you review?
      1. William Grabau
   2. What typos are in your partner’s copy document?
      1. Some formatting errors with spacing (easy fix)
      2. Short description #1 – run on sentence
   3. What didn’t they communicate clearly? Give me details here?
      1. Communicates the message of the website- being a guide for someone who is applying for a job at ESPN.
   4. Did the copy seem like it was too much/not enough? Explain.
      1. I think that the copy needs more text and a couple of different talking points on each page. Maybe you could mention key internships that ESPN looks for on page 2. This would inform the reader that this would increase their experiences and chances of landing these jobs.
   5. Where could your partner use headings, bullet points, etc. to make the copy stand out more to users?
      1. I think that you could use small headings for the things that I mentioned above. On each page have a couple of small headings, then list information that would further the guide below them.
   6. Any other comments?
      1. I really like your idea with this guide, but I think that you could include a lot more about the requirements and experience it would take to get on of these jobs.