

Background

In Lab 1 and 2 you covered:

- Basics of HTML and HTML syntax
- HTML Tables and Forms

In Lab 3 you will cover:

- Basics of CSS and CSS syntax
- How to use CSS to style a web page
- How to apply CSS to tables and forms

WARNING: This lab is longer than normal. Go slowly and understand what you are doing.

An Introduction to CSS

What is CSS?

Formally, **CSS** stands for **Cascading Style Sheets**.

What does it do?

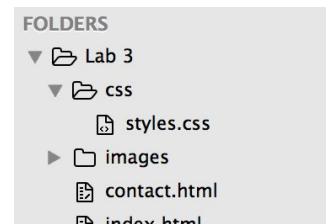
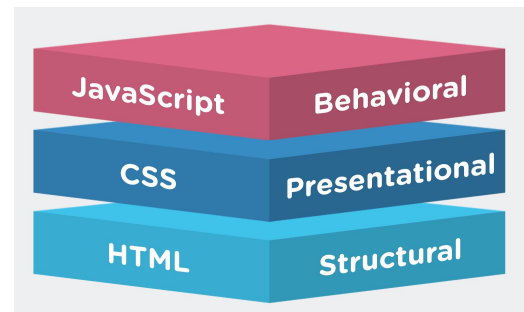
Describes how HTML elements are to be displayed.

Where do I write CSS?

CSS code can be written in three ways:

- Inline - using a **style attribute** in HTML elements
- Internal - using a **<style> element** in the HTML <head> section
- External - using one or more **external CSS files** and included using the **<link>** tag

We are going to be using the last option for now, **external CSS files**. In step 1 of this lab, you will create a new **css** folder in your lab 3 directory. You will also create and save a stylesheet: **styles.css** in that subfolder.



CSS Syntax



This is an example of a CSS rule.

Each rule is composed of the following:

- **Selector:** html element you want to style
- **Declaration:** has a property and value
 - **Property:** style attribute changing
 - **Value:** what to change to

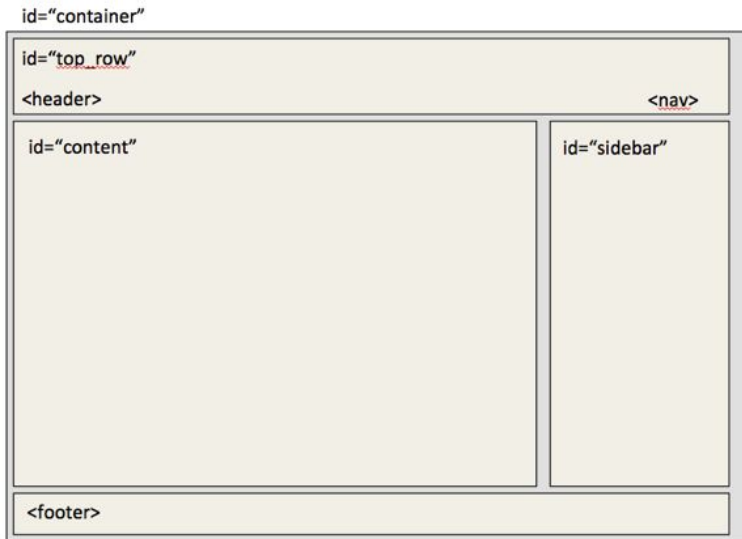
Our Grid Structure

What is a div?

The div tag is used to "divide" a web page into different sections.

These sections are boxes.
We use CSS to style these boxes.
We place content inside these boxes.

For our personal website, the left image shows the divs we will be using.



*If you don't understand the concept of a <div> then call over a TA to explain it.
This is a fundamental concept behind the lab.
It is important you understand before going forward.*

Part One: Setup and Home Page

1. Copy Lab 2 and save it as Lab 3. Create an external CSS stylesheet file by creating a new folder called **css** just like you did with the images folder. Then create a new file in the css folder, named **styles.css**.
2. Next, tell the HTML page which stylesheet to talk to! In your **index.html**, **projects.html** and **contact.html** files, add an external link to your CSS file in the **<head>**:

```
<link rel="stylesheet" media="all" type="text/css" href="css/styles.css" />
```

3. Let's write our first **CSS rule**! We will add the following **declarations** to the body **selector** in styles.css. Notice the key-words, syntax and structure of the rule.

```
body {
  font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;
  font-size: 100%;
  line-height: 1.7em;
  color: #000000;
}
```

What's going on here?

- You are declaring that all font in your **<body>** use the Helvetica font family

- At 100% of the screen
- With a line height of 1.7 em
- And the color black (#000000)

Generally, you should set the font-size property to a value of 100% and then alter the size as needed using em in the other sections of your code. Em values are probably the most difficult values to wrap the ol' noodle around, probably because the very concept of them is abstract and arbitrary. Here's the scoop: 1em is equal to the current font-size of the element in question. If you haven't set font size anywhere on the page, then it would be the browser default, which is probably 16px. So by default 1em = 16px. If you were to go and set a font-size of 20px on your body, then 1em = 20px. The line-height property controls the space between lines of text. It is often set in a unitless value (e.g., use line-height: 1.4).

4. **Save your CSS file. Refresh your index.html document in the browser.** Verify the font was updated. **(Check with the TA if you have problems.)**

Part Two: Working with <div> Tags

1. Next you will begin working with the **<div id="container">** to control your overall site layout. Verify your entire index.html document (everything between the <body> tag is included in the <div id="container">. Do not forget to close the <div> tag just before the <body> tag is closed. Then, add the following declarations to the **#container** selector in the styles.css file:

```
width: 70%;
margin: 10px auto; /* Sets 10px to the top and the the browser will center with auto */
border: 1px solid gray;
```

Note that your index.html document has a <div id="container">. Fundamental to working with CSS is the concept of an **element selector** - basically finding an HTML element and doing something to it with CSS. These selectors come in many forms: tag name, id, class, attributes, etc.

The tag name selector uses the name of the tag or element so for example, to select all the <p> tags, the selector would just be the letter "p". The id selector uses the id attribute of an HTML element to select a specific element. The id of an element should be unique within a page, so the id selector is used to select one unique element! To select an element with a specific id, write a hash (#) character, followed by the id of the element. The class selector selects elements with a specific class attribute. To select elements with a specific class, write a period (.) character, followed by the name of the class. More at:

http://www.w3schools.com/css/css_syntax.asp

Also, note that CSS uses /* comments here */ for comments – you may want to

comment-out the border in your CSS code (e.g., `/* border: 1px solid gray; */`). This code is only included to show how the CSS is building the page structure.

2. Now begin using the `<div id="top_row">` to control your header and nav content. Refer to the lab 2 solution or add a new `<div id="top_row">` to wrap the header and nav in your `index.html` file. Then add the following declarations to the `#top_row` sector in the `styles.css` file:

```
border-bottom: 1px dotted #cccccc;
padding: .5em .5em 3em .5em;
```

Again take note that the `#` hash character is used to select a specific id. Also remove extra spacing on `#top_row` header:

```
margin-bottom: -2.5em;
```

3. Use the header, `h1` and `h2` selectors to format the font and its placement in the `top_row` rule set. Your css code should look similar to the below - but alter some of the values to get a feel for the code:

```
25 #top_row h1 {
26     color: #c00000; /* Red color for header */
27     font-size: 3.75em; /* http://pxtoem.com/ */
28     letter-spacing: -2px; /* The letter-spacing property increases or decreases the
29     between characters in a text. */
30 }
31
32 #top_row h3 {
33     font-size: 1.125em;
34     font-weight: 200; /* The font-weight property sets how thick or thin characters in
35     text should be displayed. */
36 }
37
38 #top_row nav {
39     float: right; /* http://www.w3schools.com/css/css_float.asp */
40 }
41
```

In the code above, please notice that the id selector is being used to select the `#top_row` followed by a second parameter (e.g., `header`, `h1`, `h2`) to select a more specific element within the `#top_row` div.

4. **Save your `styles.css` file and view the `index.html` file in your browser.** Your results should look similar to:

Jane Doe

Carnegie Mellon University, Information Systems Program

About

5. Now use the `<div id="top_row">` to control nav content. You will be placing the navigation to the right side of the top_row and decorating the navigation with background and font changes. In this case we used `#top_row nav` rather than just `nav` because we wanted to be more specific just in case, since it selects only the `<nav>` tags within the `<div id="top_row">` tag. For this lab, using just the `nav` selector works just the same since there is only 1 `nav` tag in this page anyways.

Also note, you have `class="selected"` to style the About link slightly different to show the user what page he / she is currently on. Add the following declarations to the styles.css file:

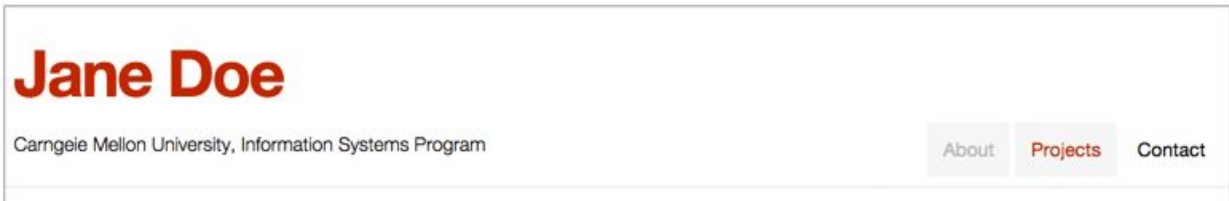
```
38 #top_row nav {
39     float: right; /* http://www.w3schools.com/css/css_float.asp */
40 }
41
42 #top_row nav a {
43     text-decoration: none;
44     font-size: 1.125em;
45     font-weight: 400;
46     padding: 15px;
47     color: #000000;
48 }
49
50 #top_row nav a.selected {
51     background: #f6f6f6;
52     color: #bbbbbb;
53 }
```

6. Next you should continue to work with `<div id="top_row">` to control nav content. You will be working with a pseudo-element to add a special effect to the `<a>` selector when a user hovers over the link. Add the following declarations to the styles.css file:

```
55 #top_row nav a.selected:hover {  
56     background: #f6f6f6;  
57     color: #bbbbbb;  
58 }  
59  
60 #top_row nav a:hover {  
61     background: #f6f6f6;  
62     color: #c0392b;  
63 }
```

More information about these interaction elements can be found at:
http://www.w3schools.com/css/css_pseudo_elements.asp

7. Save your styles.css file and view the index.html file in your browser. Your results should look similar to the image below. Your links should also change background color and text color when you mouse over them (Check with the TAs if you have problems):



Part Three: Sidebar and Footer

1. Now begin using the <div> for content and sidebar and the <footer> to control your page content, structure and footer. Add the following declarations to styles.css:

```
65 #content {
66     float: left;
67     width: 70%;
68     margin: 0;
69     padding: 1em;
70     border-right: 1px dotted #ccc;
71 }
72
73 /* With the above code you are placing the <div id="content">
74 to the left side of the page and it will expand to 70% of the available space.
75 You have no margins but a very small padding. There is also a dotted
76 border along the bottom */
77
78 #content h3 {
79     font-family: Georgia, serif;
80     font-size: 1.5em;
81     margin: 30px 0 10px 0;
82     font-weight: normal;
83 }
84
85 /* Font styling on the h3 within <div id="content"> */
86
87 #sidebar {
88     float: left;
89     padding: 1em;
90     font-size: .875em;
91     /* border-left: 1px solid gray; */
92 }
93
94 /* With the above code you are placing the <div id="sidebar"> to the
95 left side of the <div id="content"> and it will expand to the available space.
96 You also have very small padding and a change in font size. You can uncomment
97 the border to see the exact placement. */
```

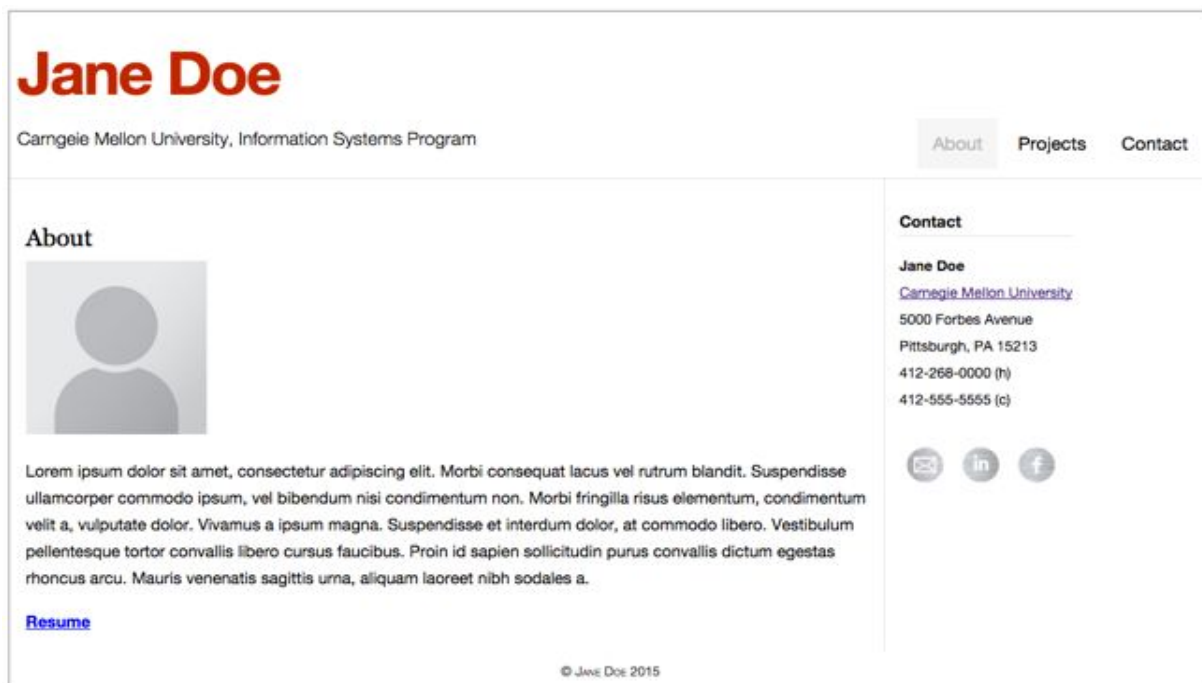


```

102 #sidebar h3 {
103     font-size: 1.125em;
104     border-bottom: 1px dotted #ccc;
105 }
106
107 /* Font styling on the h3 within <div id="sidebar"> */
108
109 footer {
110     clear: both; /* The clear property specifies which side(s) of an
111                  element other floating elements are not allowed. */
112     margin: 0;
113     padding: .5em;
114     color: #333333;
115     font-variant: small-caps;
116     font-size: .813em;
117     text-align: center;
118     /* background-color: #ddd; */
119 }
120
121 /* With the above code you are placing the footer fully below the <div id="sidebar">
122    and the <div id="content">. The clear: both means do not float this element on
123    either side of another. There is no margin but some padding. The font is altered
124    to small caps and centered. You may uncomment the background color to see the exact
125    fill / placement */
126
127
128

```

2. Save your styles.css file and view the index.html file in your browser. Your results should look similar to the image below.

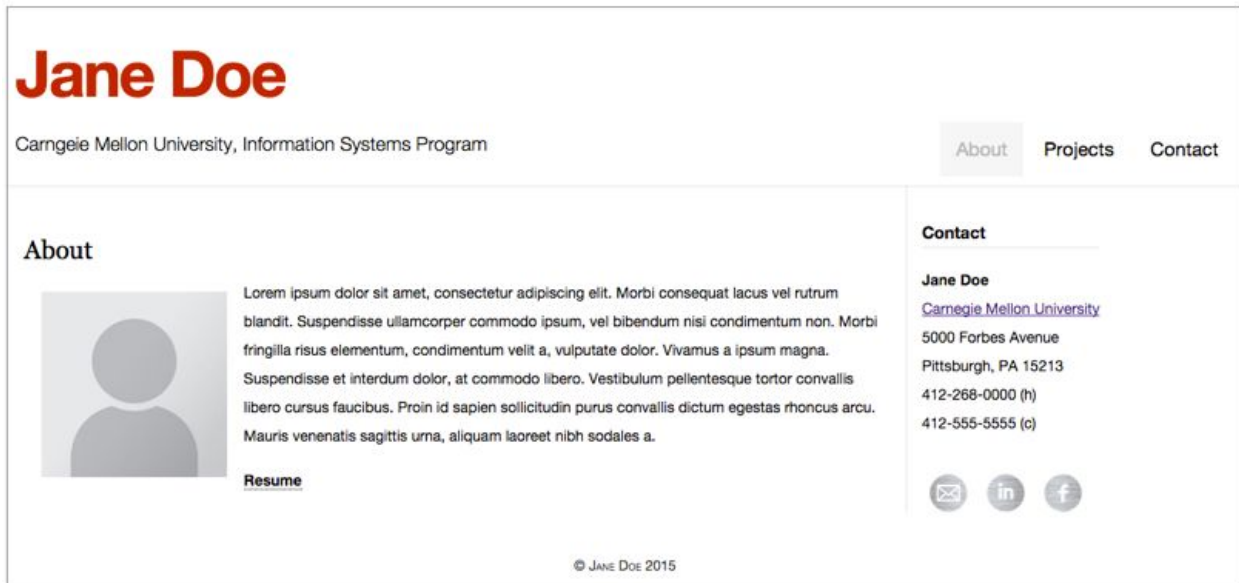


Part Four: Paragraph Styling

1. Next you should style several of the elements in the `<div id="bio">`. Note the `index.html` file includes an id named `bio`, which is used to help place the `<p>` and `<a>` for the resume next to the image. Add the following declarations to the `styles.css` file:

```
127 #content img {
128     float: left;
129     margin: 1em;
130 }
131
132 #bio {
133     margin-left: 3em;
134     font-size: .875em;
135 }
136
137 #bio a {
138     text-decoration: none;
139     border-bottom: 1px dotted #000;
140     color: #000;
141 }
142
143 #bio a:hover {
144     color: #ca1e00;
145 }
146
147
```

2. Save your `styles.css` file and view the `index.html` file in your browser. Your results should look similar to the image below. (Check with the TA if you have problems.)

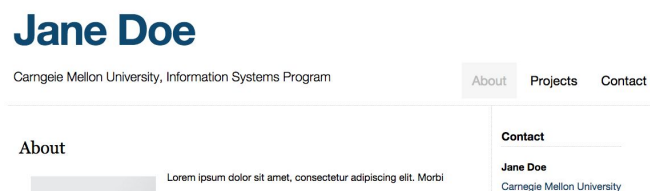


Part Five: Styling a Table

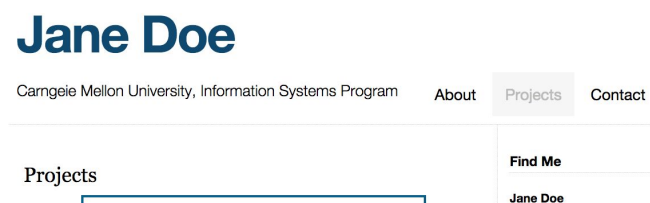
Note: Part Five of the lab will be a lot less guided compared to the previous steps. You will have to use the information provided before to help you figure out the css selectors and styles needed to complete these following steps. Also don't forget that you also have <http://www.w3schools.com/css/> and Google at your disposal.

1. Verify in the browser that the projects.html and contact.html pages also inherit the CSS declarations you created in the above steps. If not, revisit Step 2 in Part One.
2. Use the **class="selected"** to change the navigation highlight for the projects.html and contact.html pages. The links in the navigation should have a gray background only if that page is currently being viewed:

Index.html page - the "About" link has a grey background.



Projects.html page - the "Projects" link has a grey background.



3. Change the color of your name in the header of your webpage. Select any new color you prefer. Please visit <https://color.adobe.com/explore/newest/?time=all> for possible color theme choices. The lab solution will be working with a #3F7C9D for a blue hue.
4. Change the color of all hover effects on your navigation links (including the link to a resume). The lab solution will continue to use a #3F7C9D for a blue hue.
5. Change the color and remove the underline for the link to Carnegie Mellon University in the sidebar. Add an underline when the user hovers on the link.
6. Navigate to the projects.html page and style the table:
 - a. Give it a width of 75%.
 - b. Give it a solid style border of 3px and color should be a complementary color to your header color.The lab solution will work with #075F8A.
 - c. Space between the borders should be of 3px.
 - d. Style the margin in such a way that it is displayed in the middle of the page.
 - e. Table layout should be adjustable by the browser to the best fit of the content.
 - f. Only one border of any two cells should be displayed.
7. Style the table header cells with the following:
 - a. Height of the header cells should be 40px.
 - b. Font should be bold.
8. Style both header and data cells of the table with the following rules:
 - a. Give a padding of 2px.
 - b. Text should be aligned to the center.
 - c. Should have a solid border of width 1px and color #333333.
9. **Save your html and css files. View the projects.html file in your browser – it should look something like below (Check with the TA if you have problems):**

Jane Doe

Carnegie Mellon University, Information Systems Program

[About](#) [Projects](#) [Contact](#)

Projects

Course Projects		
Title	Course	Semester
Case Analysis Project	67-250	Spring 2015
Web Development Project	67-250	Spring 2015
Database Project	67-250	Spring 2015

Find Me

Jane Doe
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
412-268-0000 (h)
412-555-5555 (c)



© Jane Doe 2015

Part Six: Styling a Form

1. Navigate to the contact.html page and style the form:
 - a. Give it a width of 75%.
 - b. Style the margin in such a way that it is displayed in the middle of the page.
 - c. Give it a solid style border of 1px and color should be a complementary color to your header color. The lab solution will work with #075F8A.
 - d. Space between the border and the form's content should be of 10px.
2. Style the legends with the following:
 - a. Font should be bold.
 - b. Text should be aligned to center.
 - c. Color of the text should be #666666.
 - d. Give it a margin of 5px at the bottom.
3. Style the labels with the following:
 - a. Font should be bold.
 - b. Text should be aligned to center.
4. **Save your html and css files. View the projects.html file in your browser – it should look something like below:**

Jane Doe
Carnegie Mellon University, Information Systems Program

About Projects Contact

Contact
Please use the below form to get in touch with me.

Tell me more about yourself

Name:

Your date of birth:

Tell me about your preferences

Your favorite flower color:

Message:

Let's stay in touch

How did you find out about me?




Please indicate if you want to receive updates via email:
☐ Yes, please update me!

Please provide your email address:

Choose a password:

Find Me

Jane Doe
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
412-268-0000 (h)
412-555-5555 (c)

Part Seven: Finishing Touches

1. If you pay close attention to the layout of your three pages (quickly quick back and forth between the pages), you will notice the contact.html page has longer content than the other pages and thus is making your layout appear to shift a bit. You need to work with the overflow-y property to fix this problem.

The overflow-y property specifies what to do with the top/bottom edges of the content - if it overflows the element's content area. The value should be set to scroll to allow the content to be clipped and a scrolling mechanism to be provided. More details: http://www.w3schools.com/cssref/css3_pr_overflow-y.asp.

Add the following code to your stylesheet:

```
194
195  html {
196      overflow-y: scroll;
197  }
198
```

2. Congratulations – you have just styled more of your website with CSS! Now the time remaining polish up the overall look of the site by tweaking the CSS as needed. For example, you might want to change the fonts (be bold, italics, different color, different font, etc.) and can do that pretty easily since they are all wrapped in classes and ids for easy styling.
3. Take a screenshot of each of your webpages displayed in the browser. Save these images to your directory. The TAs will use this (plus your code) to grade your lab.
4. **If you complete the lab during class – show your work to the TAs or Prof Q before you leave.**

Whether you finish in class or at home – please submit your completed lab (source code files and images) to the dropbox in Canvas before class begins on Thursday, March 9, 2017.

Your submission must be titled andrewID-lab3.zip. Thus, my submission would be titled jquesenberry-lab3.zip. Late submissions will not be accepted.

Congratulations – you have just built a website with better styling!