INFORMATION SYSTEMS MILIEUX (67-250)

Lab 1 :: Due Thursday, March 2, 2017

What is HTML?

Hypertext Markup Language (HTML) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the browser how to display a web page's words and images for the user. The reason it is called Hyper is because it is not a linear process, and you can go anywhere on the internet. Markup is what HTML tags do to the text inside them. They mark it as a certain type of text. The language aspect is because it has code-words and a syntax just like language. HTML is the skeleton to any website, and it is important to understand what makes it up and how to use it.

What are tags?

As discussed before, HTML marks text so that special attributes are assigned to specific text when the browser reads the file. The way that the browser and HTML does this is by using tags.

Tags are the words that are between the <angle-brackets>. They are an essential part of HTML. They allow a browser to describe the content of the tag, and they are very simple to use. For example bolding a letter is as easy as adding the bolding tag.

Hello World!

If we viewed the the HTML above in a browser it would show it as **Hello** World! Notice the tag opens the element. The tag closes the element. Something that is important to notice is that most tags have a closing tag marked by the "/". But not all tags have to have an end or close tag. We will explore this later. As you can see they are very easy to use, and the most difficult part of HTML is knowing what all the tags do. So it should be pretty easy for you.

Lets Begin!

In this lab, we are going to build a personal website - including your resume - so you can show it off to your friends, family and recruiters. It is important to know the steps we are going to take before we begin. These steps will be separated by labs.

Lab 1: HTML
Lab 2: Tables & Forms
Lab 3: CSS

Each lab will focus on a specific topic of web development. And as they progress, more and more content will be combined together. So don't expect your website to look pretty right away.

Directions

0. You will need a web browser (e.g., Google Chrome, Firefox, Safari, etc.). You will also need any text editor (textpad, textedit, textmate, etc.). You do not need to have a web editor such as Dreamweaver or FrontPage.

   An extensive list of editors can be found at:
   http://en.wikipedia.org/wiki/List_of_text_editors

1. To begin, create a new folder called "<andrewId>-lab1". Of course replace 'andrewID' with your Andrew ID. Open your text editor. Once you open your editor, create a new file named "**index.html**". The .html is necessary so that the browser knows that it is a file it can read. Save that file to your directory.

2. Next we will add the necessary tags that all HTML files should include. The first you need to type in is the <!**DOCTYPE** html> tag. This is a special tag since it tells the browser that this code is in HTML5. This is similar to how there are different versions of python. Also, this tag lacks an end tag.

3. Next we are going to add the <**html** lang="eng"> </**html**> tag. This tag is the root of the html document and states that it is in english. An important rule to follow is that you write inside each tag. So for the next steps you should be able to write inside of it and look something like:

```
1
2
3  <html lang="en">
4
5  <!-- More will go inside here -->
6
7  </html>
8
```

4. All websites are split into two parts once inside the HTML tag. There is the **head** tag, and the **body** tag. These tags do not belong inside each other but are different from each other and the content that goes inside each of them differs as well. Make sure to include them and confirm your "index.html" looks like the screenshot below afterwards.

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <!--The HEAD section has (mostly) items that are not directly seen by the user-->
6  </head>
7
8  <body>
9      <!--The BODY section has (mostly) items that are seen by the user-->
10 </body>
11
12 </html>
13
```

(HTML comments are made using the <!-- -->, so you don't have to copy the grayed out text. This is very similar to using the # in python).

5. The **head** tag contains information about the resources the website needs, and also contains the **meta** data of the website (e.g., author of the website, or the description of the website). Head tag information does not show up on the web page - it is only informational. Lastly, there is information of the **<title>** of the website, which shows up in the tab. The head tag can contain a lot of information and if you are more interested in finding what else it can do, please refer to the Larsen book or http://www.w3schools.com/.

6. Before explaining the body tag, let's add a few elements to our **head**. We want to be the **authors** of this website, and to mark this we are going to use the meta tag.

   <meta name="**author**" content="Your Name">

   Be sure to replace "Your Name" with your name. There are other tags that we can use to fill the meta and they all need their own bracket. Make sure to include "**keywords,**" which is just the words that are important to your website. You also need "**description**", which is a short description of what the website is. They can be added the same way the name tag was added. Each meta attribute should be on its own line. If you are running into trouble make sure to google the problem or the tag. Googling is a great skill to have! Make sure after you're done, it looks like the screenshot below.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <!--The HEAD section has (mostly) items that are not directly seen by the user-->
    <meta charset = "utf-8"/>
    <meta name="author" content="Jeria Quesenberry">
    <meta name  = "keywords" content = "Carnegie Mellon University, CV">
    <meta name = "descriptions" content = "Jeria Quesenberry Professional Website">

    <!-- The title appears in the menu bar of the browser -->
    <title>Insert Here</title>

</head>

<body>
    <!--The BODY section has (mostly) items that are seen by the user-->
</body>

</html>
```

7. Right now you are probably wondering what does the website look like. Well first, save your html file, and navigate to it in your file directory. Once there, double-click on it, and your current browser should open and display it. This is because your browser is a compiler in lamen terms but that's another can of worms. But once open you will not see any content in your website. But celebrate, because you finished the first few steps of making a website. Granted it is missing a lot so let's keep working.

8. In the <body> tag you have to add three HTML 5 tags (<header>, <nav> and <footer>) and one tag with the corresponding end tags.

```
<body>
<!-- The BODY is where the main content goes -->

    <header>
    </header>

    <nav>
    </nav>

    <div id="content">
    </div>

    <footer>
    </footer>

</body>

</html>
```

9. So, now let's add content to the header tag. This tag element represents a container for introductory content or a set of navigational links. An html page can have multiple of this type, but for the most part there is one. Inside the header, you generally include **<h1>** header tags **(the number after the h represents the hierarchical importance of the header, so 1 is most important and 6 is least important)**. Add your name and school inside an **<h1>** tag and **<h2>** tag respectively.

```
23 ▼    <header>
24          <h1>Jeria Quesenberry</h1>
25          <h2>Carngeie Mellon University, Information Systems Program</h2>
26 ▲    </header>
```

10. Now we are going to add the navigation to your HTML document within the <**nav**> tag. Each **nav** item should be a link using the <**a**> or anchor tags. Anchor tags are used for creating links or bookmarks inside a document. If you give the **<a>** tag the **"href"** attribute, it will act as a hyperlink, and the url of the link is the value of the "href" attribute. **The link can either be a website like "http://www.google.com" or another html file in the directory.** More information about links and the anchor element can be found in chapter 3 of the Larsen book.
    a. We need to add three **<a>** tags to the nav container. Each one will link to another page on your website (you will make the other pages in Lab 2). For now have them link to "**index.html**" (the current page), "**projects.html**" and "**contact.html**" pages.

```
28      <nav>
29          <a href="index.html">About</a>
30          <a href="projects.html">Projects</a>
31          <a href="contact.html">Contact</a>
32          <!-- This anchor takes us to another page within the site; hence it is called a relative link. -->
33      </nav>
```

11. Save your HTML document. Open your HTML document in the browser. Verify your navigation links appear.
    a. If you try to click on either project or contact, it will be an error. This is because you don't have these files, so don't worry. (Note: You can try adding your own links to websites to see if it works.)
    b. Again, do not worry about appearance or styling at this point.

12. Next, add the page content for your professional website. This content should be in the . The tag is nothing more than a container unit that encapsulates other page elements and divides the HTML document into sections. Web developers use elements to group together HTML elements and apply CSS styles to many elements at once. Be sure your includes the following elements:
    a. Page header (e.g., About) with a **<h3>** tag.

b. Image (e.g., your photo) with the **<img>** tag. Note you will need to make an "**images**" folder in the project folder add an image file to it. It is important to keep your files organized in some sort of hierarchy / structure so you can easily access items.
    i. For the **<img>** tag, there are two main attribute it uses. First is the **src** attribute that is the indicator of what image to use, so it can be either a web link or a place in your directory. Second is the **alt** attribute that is used when your browser cannot load the image, it will display that text.
c. Page content (e.g., a bio) with **<p>** tags.
    i. **<p>** tags are the paragraph tags, and it is where most of your text goes.
d. A link to your resume with an **<a>** tag. Create a pdf resume and put it in the root directory with index.html. By adding an **href** attribute linked to the pdf file (just like you did for the html file) it will link to the actual pdf file in your browser. The **target="_blank"** attribute lets the **<a>** link to open up in a new tab. Note, the resume should open as a PDF in a new tab using the target attribute in the <a> tag.

```
35      <div id="content"> <!-- Use div tag with an attribute of an id = content (i.e., this div's unique name)
        -->
36          <h3>About</h3>
37
38          <img src="images/photo.png" alt="Photo" />
39
40          <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi consequat lacus vel rutrum blandit.
            Suspendisse ullamcorper commodo ipsum, vel bibendum nisi condimentum non. Morbi fringilla risus
            elementum, condimentum velit a, vulputate dolor. Vivamus a ipsum magna. Suspendisse et interdum
            dolor, at commodo libero. Vestibulum pellentesque tortor convallis libero cursus faucibus. Proin id
            sapien sollicitudin purus convallis dictum egestas rhoncus arcu. Mauris venenatis sagittis urna,
            aliquam laoreet nibh sodales a.</p>
41
42          <a href="resume.pdf" target="_blank">Resume</a>
43      </div> <!-- Close content -->
```

(Note: You may wish to use a dummy text place holder – I recommend using http://www.lipsum.com/. My code looks like this – but your content should be much different.)

13. Now you realize that you want to include a list of skills/interests or even courses you have taken in your About section. You should include a bulleted list of information using the **<ul>** and **<li>** tags. My code looks like this – but your content should be much different. There are two types of list in html **<ul>**, and **<ol>** tags. **<ul>** represents an "unordered list" or in other words a bulleted list, and **<ol>** represents and "ordered list" or in other others numbered list. Each one will have multiple **<li>** tags which represents "list item" tags. For this example we are using the **<ul>** tag. Inside the tags we are going to add the items which are used using the **<li>** tag. This content should be added between your paragraph content and the resume link.

```
41
42          <h4>Courses Taken</h4>
43          <ul>
44              <li>67-101: Concepts in Information Systems</li>
45              <li>67-250: Information Systems Milieux</li>
46              <li>67-325: Global System Delivery Models</li>
47              <li>67-329: Contemporary Topics in Global Systems</li>
48              <li>67-373: Software Development Project</li>
49              <li>67-475: Information Systems Applications</li>
50          </ul>
```

14. To make your HTML file more readable and more module, you should separate out your content div. In your content div, you should currently have the heading, your image, the paragraph bio, courses taken (or any other list), and your resume. For the purpose of this lab, we are combining the paragraph bio, course taken, and resume link into one section within the content. To do so, just wrap those three sections in a ** ** tag with id **"bio"**. This will greatly help with styling your web page as you will learn more in Lab 3. If you have trouble follow the screenshot below, note that your web page will look different.

```
35      <div id="content">
36          <h3>About</h3>
37          <img src="images/photo.png" alt="Photo"  />
38          <div id="bio">
39              <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi consequat lacus vel rutrum blandit. Suspendi
                nisi condimentum non. Morbi fringilla risus              elementum, condimentum velit a, vulputate dolor. Viv
                dolor, at commodo libero. Vestibulum pellentesque tortor convallis libero cursus faucibus. Proin id sapien
                dictum egestas rhoncus arcu. Mauris venenatis sagittis urna, aliquam laoreet nibh sodales a.</p>
40
41              <h4>Courses Taken</h4>
42              <ul>
43                  <li>67-101: Concepts in Information Systems</li>
44                  <li>67-250: Information Systems Milieux</li>
45                  <li>67-325: Global System Delivery Models</li>
46                  <li>67-329: Contemporary Topics in Global Systems</li>
47                  <li>67-373: Software Development Project</li>
48                  <li>67-475: Information Systems Applications</li>
49              </ul>
50
51              <a href="resume.pdf" target="_blank"><strong>Resume</strong></a>
52          </div>
53
54      </div> <!-- Close content -->
```

15. Save your HTML document. Open your HTML document in the browser. Verify your photo and content appear. Test your link to your resume also opens. Again, do not worry about appearance or styling at this point.

16. Add the final content to the **<footer>** of your HTML document. Information you may want to include could your name, contact information, version history, etc. Just like the header tag is for the top of website, the footer is for the lower part. (Note: the © that is used before your name is actually the symbol representation for the copyright symbol ©)
    a. In the example below, please note a few items. First, the **<a>** tag uses a **mailto** in the URL to send an email message. Add your email to the mailto attribute off the <a> tag.
    b. Also, the **<a>** to CMU's website opens in a new window with the target attribute.

c. Finally, a horizontal line with the **<hr/>** and the line break <**br**/> tag are used. These HTML tags do not need a corresponding end tags but are closed when opened, hence the **"/"**.

```
45      <footer>
46          <hr/>
47          &copy; Jeria Quesenberry 2017<br/>
48          <a href="mailto:jquesenberry@cmu.edu">Email Me</a>
49          <!-- Anchor tags can take you to pages outside the site, within the site or within the page. -->
50          <a href="http://cmu.edu" target="_blank">Carnegie Mellon University</a>
51          <!-- This anchor takes us to another site (href starts with "http://...") so this is an absolute link. -->
52      </footer>
```

17. Save your HTML document. Open your HTML document in the browser. Verify your footer information appears.

18. Now you are finished working with "index.html."

19. Now that you finished this page create a new file as "projects.html", and add it to your directory next to your index file. You will copy all the contents of your index file into this new projects file. Now delete all of the content in the ** ** element in the projects.html file since you will have different content for the page. However, this will leave the same header and footer for the "projects.html" page and "index.html" page which you want.

20. Add new content for this page in the ** .** For now you will only need to add an **<h3>** heading for your Projects page just like you did with your About page. (hint: name the heading "Projects")

21. Save your HTML document. Open your HTML document in the browser. Verify the content in the new page appears. The links between your two pages in the navigation should also link back and forth.

22. Take a screen shot 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.

23. **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 2, 2017.

    Your submission must be titled andrewID-lab1.zip. Thus, my submission would be titled jquesenberry-lab1.zip. Late submissions will not be accepted.

Congratulations – you have just completed your very first website!