Web Application Development ECOM 5410
lab 1: HTML

Do you want to know how to make a website, but don't know what HTML code to use?

This lab covers the basics of HTML. While some of you may already be familiar with HTML.

What is HTML?

HTML (HyperText Markup Language) is a language for describing web pages.

- HTML is made up of tags, which may be nested inside one another.
- HTML represents the content and organization of pages.
- HTML files rendered by a web browser are called web pages.

Below is an example of a HTML document:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Page title. Shown in tabs.</title>
  </head>
  <body>
    <h1>Simple Page!</h1>
    <p>This is a totally bare-bones page.</p>
    <!-- this is a comment and is totally ignored -->
  </body>
</html>
```

As you can see, the document consists of tags and plain text. Tags are surrounded by angle brackets < and >. Tags come in pairs, such as <head> and </html>, and they are called the start tag and end tag, respectively. The end tag should be identical to the start tag, but with a forward slash / before the tag name. Tags indicate the semantics of their contents. For example, the <h1> tag indicates a level 1 header (the largest). It's good style to write HTML indented in the manner above to make it easier to find corresponding tag pairs, but it's not required.

Now, line by line:

1. The first line is a document type (doctype) declaration. These tell the browser which variant of HTML you're using. You may see other doctypes, but we'll be using this one to declare our document is HTML5.
2. The <html> tag defines an HTML document. It is not strictly necessary to include <html> tags, as most all web browsers will infer them if missing.
3. The <head> tag denotes an area where you provide information related to the page which is not page content. This can include any additional resources, the title of the document, encoding format, etc. As with <html> tags, <head> tags will be inferred if missing.

4. The example document is encoded with UTF-8 (note that no non-ASCII codepoints have been used up to this point). (If this is gibberish to you -- it's not so important, just include it. Or read: What is a character encoding?)
5. The <title> tag sets the title of the page, which is primarily used to set the text in the tab bar.
6. The <body> tag defines the content of the page. This is where you put information you want to be visible on the page, such as text, images, etc. As with <html> tags, <body> tags will be inferred if missing.
7. The <h1> tag is for 1st header. h2-h6 tags also exist for headers of decreasing size.
8. The <p> tag denotes a paragraph, and its contents will show up on the page with a line break above and below it.

This is how Example 1 looks when rendered (notice you can only see what is in the body of the document):

Simple Page!

This is a totally bare-bones page.

HTML Tags:

HTML is made up of tags,tags are special text that you use to mark up, or distinguish, parts of your web page.

These tags tell the browser to display whatever is inside the tag in a specific way.

Here's one example of a tag in action:

```
This is my very first website and I'm <b>extremely excited!!!!!</b>
```

You can see that the words "extremely excited" are in these <b> tags– "b" is for bold.

This is my very first website and I'm **extremely excited!!!!!!**

Basic Head Tags :

Meta Tags:

The first tag that should be in your head is this meta tag. This sets the character encoding.

```
<meta charset="utf-8">
```

Title Tag:

As you can probably guess, this sets the title of the web page. If a website has different pages, each page may have its own title.

```
<title>My First Website</title>
```

Attributes:

Before we learn some other common HTML tags, you should learn about attributes. Some tags have attributes, which provide additional information about that tag. In the <meta> tag in Example 1, you have already seen an example of attributes to specify which character set the document uses. Attributes are of the form name="value", where the value must always be enclosed in double-quotes.

```
<tag attribute="value" foo="bar">
```

More Common HTML Tags:

Images:

To insert images, you can use the <img> tag.
Example:

```
<img src="http://i.imgur.com/xTOON7z.png" alt="Take CS132">
```

<img> tags have two common attributes:
An absolute link: An image named "banner.jpg", located in the "images" directory on "www.example.com" has the URL "http://www.example.com/images/banner.jpg".
A relative link: If the document is being served on the same domain as your image, you can also use the relative locations of the image as the URL. If you had, for example, a document at http://example.com/page.html, you could link to http://example.com/pic.jpg by just specifying "pic.jpg" in the URL.
The alt attribute indicates a textual name that shows up in the place of the image if the image cannot be displayed. (This is especially useful for blind users as screen readers will read out the alt attribute.)

Hyperlinks:

A hyperlink (or link) is some text or an image that you can click on to jump to another URL. These can be created with the <a> tag.
Example:

```
<a href="http://iug.ps.edu/courses/ECOM 5410/">Some text displayed to represent the link</a>
```

The <a> tag must use an href attribute with a URL or file path (to another html file within your file system) to indicate where the hyperlink will direct the user. This attribute accepts the same type of links as image hrefs: absolute or relative links.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a has even more information about links if you're curious.

Line Break:

Now, if you want to separate your content onto multiple lines, but you don't want that space that comes with a paragraph, you can use a line break, or a **
** tag.

Horizontal Rule:

The horizontal rule tag will create a horizontal line on your web page that goes all the way across.
You write it this way: **<hr>**

Ordered Lists and Unordered Lists:

To insert a list, use either the <ol> tag for ordered lists, or the <ul> tag for unordered lists. Within the <ol> and <ul> tags you will place your list items in <li> tags.
An ordered list:

```
<ol>
   <li>The first list item</li>
   <li>The second list item</li>
   <li>The third list item</li>
</ol>
```

will look like:

1. The first list item
2. The second list item
3. The third list item

An unordered list:

```
<ul>
   <li>List item</li>
   <li>List item</li>
   <li>List item</li>
</ul>
```

will look like:

- List item
- List item
- List item

Tables :

This is how you create a table:

```
<table>
  <thead>
    <tr>
      <th>AND</th>
      <th>False</th>
      <th>True</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>False</td>
      <td>False</td>
      <td>False</td>
    </tr>
    <tr>
      <td>True</td>
      <td>False</td>
      <td>True</td>
    </tr>
  </tbody>
 </table>
```

While this may seem complex at first, tables follow a simple structure. The <table> tag surrounds the entire table. <thead> determines which row of the table is the header while <tbody> usually contains the table data. Any cell that you want stylized as a header should use <th> while other cells should use <td>.
Here's what the above basic table would look like:It might not look so pleasing now, but you'll soon learn how to stylize your html in the CSS section.

Forms

To create a form, use the <form>, <label>, and <input> tags. The <form> tags will surround the other tags.

```
<form action="myURL" method="POST">
  <label for="noodle">What is your favorite type of noodle?</label>
  <input id="noodle" type="text" required/>
  <p>Would you eat noodles every day?</p>
  <input type="radio" name="daily" value="yes" id="yes"/>
  <label for="yes">Yes, gladly!!</label>
  <input type="radio" name="daily" value="no" id="no"/>
  <label for="no">No, that's impossible</label>
  <p></p>
  <input type="submit" value="Submit Response" />
 </form>
```

Here is what it looks like:

This may seem like a lot, so let's break it down:

- The action attribute of the form tag will determine where the data from the form will be submitted after hitting the submit button. If unused, the data is sent to the webpage that it is currently on. After pressing submit, the user will also be redirected to this URL.
- The method attribute of the form tag will determine how the data from the form will be submitted. The two most common methods are GET and POST. You will learn more about these methods later.
- The <input> tag is an extremely versatile tag that changes based on what value the type attribute has. You can observe what the text and radio types look like above. Other useful types include password, and checkbox. Notice how input elements do not require an ending tag! https://www.w3schools.com/html/html_form_input_types.asp has a list of input types, but be aware that most input types not mentioned above are not available on all browsers or are not backwards compatible.
- The <input> tag must always be accompanied by a <label> tag. If the <label> tags were missing, the text written would still show in the rendered result. However, this makes forms difficult to understand using screen readers. Additionally, by using the <label> tag, clicking on the text label will correspond to clicking on the input element itself. This comes in handy with small input elements, like radio buttons or checkboxes. Simply match the for attribute of the label with the id attribute of the input element.
- The is there simply for spacing. Another tag you could use instead is
, which requires no end tag. This tag will insert a single line break.
- The "Submit Response" button is actually another <input> element! Once clicked, all of the information the user has filled in will be sent to the address specified by the form action attribute. The user is also redirected to this address by default. (Note that if you click submit in the above rendered form, the page will not redirect to "myUrl". This was done using JavaScript and will be something you will learn how to do in a later assignment.)

What is New in HTML5?

Let's look at the factors we will be using for the comparison on HTML vs HTML5:

Video & Audio :

Here was no such concept of media in case of HTML. But it is one of the integral parts of the 5th Version.

Vector Graphics :

Vector graphics were used in HTML with the help of various technologies such as VML, Flash, Silverlight etc. But it is an integral part of HTML 5 such as canvas and svg. This was a new addition to the revised version.

Storage

In case of HTML, we can use the browser cache as the temporary storage whereas in case of HTML5, application cache, web SQL database and web storage is used.

Web Browser Support:
Now for the first version of HTML, all the old web browsers run smoothly for creating web pages. For HTML5, the new browsers have started supporting its specifications. Some of the browsers include mozilla firefox, chrome, opera .

Ease of Use:
HTML5 does have risks like constant updates, it is generally easy to keep up with the changes and updates because of the simpler syntax as compared to other versions of HTML.

New Elements in HTML 5 :

Audio & Video
The audio tag and the video tag are the two major additions to the HTML5. The audio tag and the video tag enables developers to embed music and audio on their website.The audio & video tags also have a number of attributes for additional controls.

```
<audio autoplay="autoplay" controls="controls">
        <source src="music.ogg" />
        <source src="music.mp3" />
</audio>
```

Canvas
One of the key added elements in HTML5 is <canvas> which has hugely impacted the use of Adobe Flash in websites. The <canvas> element can be used to draw graphics with various shapes and colors via scripting, usually JavaScript. The element is simply a container for the graphics.

```
<canvas id="myCanvas" width="200" height="100" style="border:1px
solid #000000;">
</canvas>
```

Header & Footer
Another big addition that HTML5 comes with are the new <header> and <footer> elements, indicating a new web anatomy. With these new tags, there is no longer a need to identify the two elements with a tag.

```
<header>
        <h1>Title </h1>
        <div class="post-meta">
                <p>By Author</a> <span class="category">Filed in
                Web 2.0</span></p>
        </div>
</header>
<article>
        <p>Body of text</p>
</article>
<footer>
        <p>End of page</p>
</footer>
```

menu

Given the fact that the web has changed into more just linked paged and documents, the <menu> tag is a welcome addition for greater web interactivity. It represents menu commands for simplicity in desktop and mobile applications.

```
<body contextmenu="new-context-menu">
        <menu id="new-context-menu" type="context">
                <menuitem>Hello World</menuitem>
        </menu>
</body>
```

HTML Validation:

You can (and should!) validate your HTML syntax using services like https://validator.w3.org/. Make sure to do this now with your examples!

Homework 1 :HTML

Ethics Readings :
For this lab, please take a look at the following articles.
1- HTML Accessibility
2- An Alphabet of Accessibility Issues

Requirements :
You are responsible for creating HTML page:
1 - Create a simple webpage, write a resume about yourself,add all tags that you learned from this lab together, add some images ,list, table ,form.
2- In addition you should use some New Elements from HTML5.
3- Don`t use CSS.
4- Create a git repository for this HW.
5- Push your repository to the github.
6- For adding some elements or features don`t push to the branch master directly, you should create a new branch and push to this branch then create a pull request and merge your branch with master.

For more resources :
https://www.freecodecamp.org/
https://www.codecademy.com/learn
https://www.w3schools.com/html/default.asp
https://htmldog.com/guides/html/beginner/