**BUA 235 Lab #1: Create a hypertext web site with HTML**

**(Thanks to Joe Clark, PhD who developed this assignment)**

HTML is a “markup” language, not a programming language. It uses “tags” to add information to ordinary text. For example, the <B> tag means “boldface”. The matching tag </B> ends the boldface font.

So this code:

<B>These three words</B> are in bold.

Would appear on screen as:

**These three words** are in bold.

**Getting Started**

An HTML file is just a “plain text” file that uses HTML markup. You can create and edit a plain text file with Notepad (on Windows) or TextEdit (on Mac), but these are really not ideal. There are several free choices that are much better for writing code:

* Notepad++ (for Windows): <https://notepad-plus-plus.org/>
* TextWrangler (for Mac): <http://www.barebones.com/>
* Sublime Text (both): <http://www.sublimetext.com/>

The advantages of these tools include syntax highlighting, which makes code much more readable and helps you find mistakes, as well as line numbers, find-and-replace, and other handy features.

Note: *Do not use a word processor like Microsoft Word or Apple Pages to create this code file. Word processors save lots of “non-text” information, like fonts, layouts, and settings, so they are not plain-text files.*

**Hello World!**

It is a tradition among computer programmers to test out new technology by making it say “Hello world!”

You can make this happen by coding your new web page like so:

<html>

Hello world!

</html>

Then double-click the file, or drag it onto a web browser window, to see it in the browser.

**A more proper beginning**

The above isn’t exactly kosher HTML, but it works because the browsers are smart enough to fill in what’s missing. The bare minimum for a “proper” web page looks like the following example. Many IDEs will give this to you automatically. Here it is for your reference:

<!DOCTYPE html>

<html>

<head>

<title>Hello world</title>

</head>

<body>

Hello world!

</body>

</html>

In this document, the <body> tag indicates the main content of the page. Your “Hello world” and any other web page content should go in between <body> and </body>. The <title> tag is also worth noticing. It sets the words that will appear in the bar at the top of the window.

**A personal web site**

Back in the days before Facebook, yes, even before MySpace and LiveJournal, college students would create their own personal web sites. This is a little “retro” now, but a good choice for this lab because none of you will lack things to talk about – yourselves!

**To Begin**

Step 1. Install an editor of your choosing

Step 2. Create a Folder in an area where you can find it again.

Step 3. Create an .html file for the Hello World page (see code in A More Proper Beginning above)

Step 4. Save your file in your folder from Step 2. Name the file: hworld.html

Step 5. Go to your folder and open hworld.html

Did your file open in a web browser? Yay!

**Next –**

Now that you have learned how to create a working webpage – start making your personal webpage.

Step 6. Design your Home Page

Begin by creating the text for your homepage. I suggest starting with your name, a subtitle of some kind, a paragraph of introduction, and placeholders for images and links. Here’s mine. (Remember, this goes between the <body> and </body> tags.)

Tanya Beaulieu, Ph.D.

Business Faculty at UMaine

[portrait]

Welcome to my web site! My background is in software development on large enterprise systems. I am currently an assistant professor at the University of Maine.

Go Blackbears!

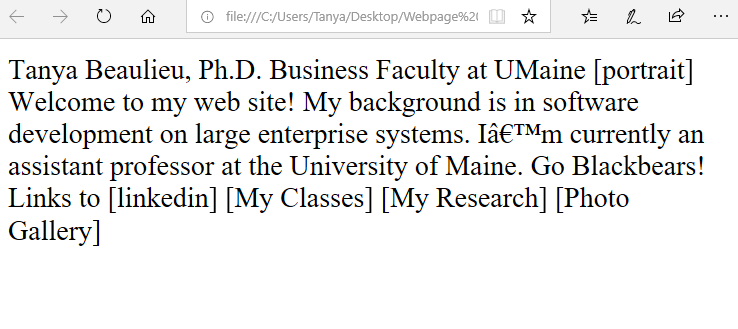
Links to [linkedin] [My Classes] [My Research] [Photo Gallery]

Once you have designed your homepage – save it in your folder that you created earlier. **Name your file: index.html**

|  |
| --- |
| **IMPORTANT!! Name your file EXACTLY: index.html**  The file name is important once you set up your webpage. When people go to your URL, the system will look within your URL for a folder called: index.html as the default page.  HTML is case sensitive!! Make sure your file name is all in lowercase.  DON’T NAME YOUR FILE:  Index.html  Index.HTML  HomePage.html  Index.txt  etc. … |

Once you have saved your file, go to the folder and open the file. How does it look?

Well – mine did not look so great:



The reason that everything is so jammed together is because there is no HTML code to tell the browser how to display the information.

**The “ML” in “HTML” is Markup Language**

The next step is to mark up the code with HTML tags. Here are some tags that will be useful:

<H1> Heading </H1>

Makes the text large, bold, and on a line of its own.   
There are also H2, H3, H4, H5, and H6 tags for sub-headings.

<P> Paragraph </P>

Wrap a block of text with this tag and it will be separated from preceding and following text by line breaks.

<DIV> Text block </DIV>

Like <P>, but with no extra whitespace between the blocks above and below.

<B>Bold</B>

Markup to style text as bold.

<I>Italic</I>

Markup to style text as italic.

<UL>

<LI>list item 1</LI>

<LI>list item 2</LI>

<LI>list item 3</LI>

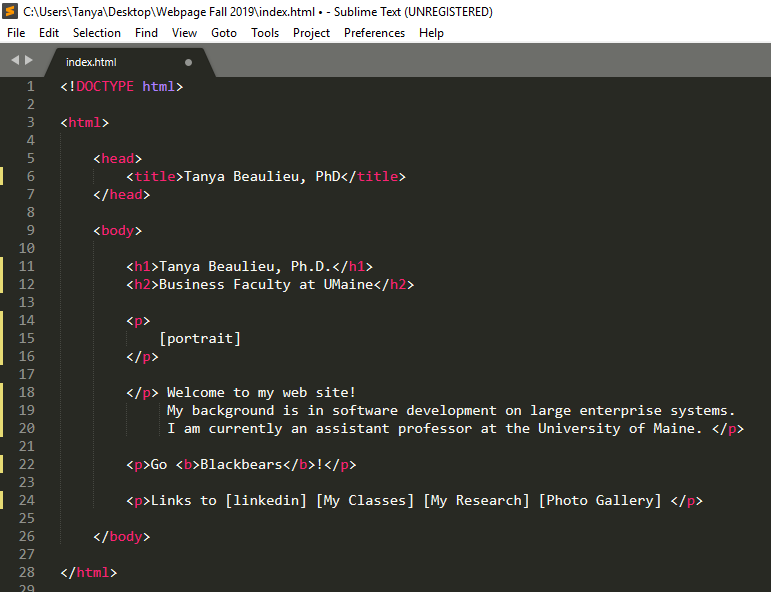
</UL>

An “unordered list” (bullet list) with three items. Use <OL> for a numbered list.

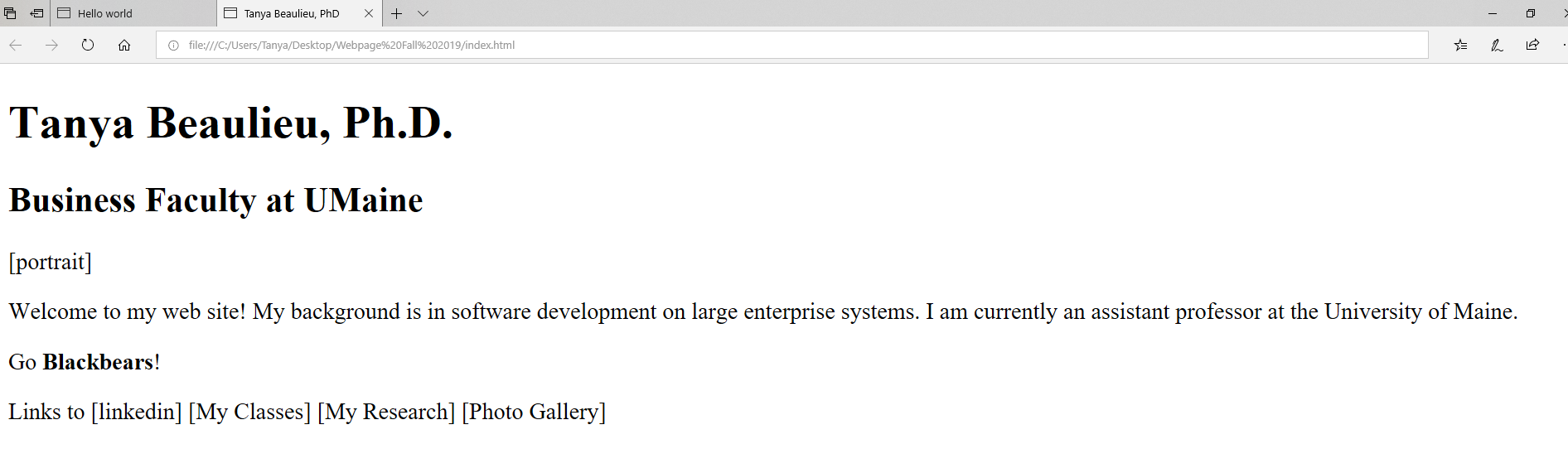
Go back to your text editor and add some html tags to your homepage.

Save the document and then redisplay in the browser to test it out. Better?

Here is a screen shot of my html file in sublime with some tags:



And here is what it looks like when displayed in a browser:



Much Better!

**Adding Links**

The “ht” in html stands for hyperlink, or link for short. Hyperlinks are a reference point to some other location. When the user clicks on the link – the browser will then take some action based on the type of reference included in the html tag.

Here is some code for hyperlinks that you will be using:

* Include an image on your webpage

<IMG SRC="portrait.jpg">

Includes the specified image into the page. “portrait.jpg” is the name of the file that contains the image. Remember html is case sensitive! Include your images in the same folder as your webpage, then you don’t need to specify the file path.

* Link to an external webpage

<A HREF="https://umaine.edu/"> UMaine </A>

A hyperlink to another web address. “UMaine” is the text that is displayed on the webpage – that is the text that is clicked. Clicking the link takes the user to the URL address in HREF.

* Link to another page within the same website

<A HREF="aboutme.html"> About Me</A>

A hyperlink to another page in the same website. “aboutme.html” is the name of the file that contains the html code for that webpage. Clinking the link takes the user to that page.

Remember – file names are case sensitive. Aboutme.html is NOT the same as aboutme.html

This should be enough to get you started. For more HTML tags, take a look at <http://www.w3schools.com/>, by far the most popular online reference site for web developers.

One of the next more advanced things you might want to try is a <TABLE> for laying out text in a grid of rows and columns.

**For this first week**

Create 4-5 HTML files with your text. Add html tags to make paragraphs, add headers, etc. Add html to create the links between pages, making this a “hypertext” document. The homepage should have links to all the other pages, and every page should have links back to the homepage (at least). You can add links to your blog, your LinkedIn profile, and any other sites that interest you.

Before next Thursday you should:

* have 4-5 pages that is formatted with html tags
* add at least one picture, just to make sure you know how
* have links between them
* fix any bugs or broken HTML tags so the pages work

Don’t worry about the fact that the visual style of your site is pretty plain. In the next lab, I’ll introduce you to CSS, the “style sheet” language for adding fonts, colors, and styles.

Remember – put all of your files (webpages and images) into one folder!