Assignment Two: The Basic Web Code

INTRODUCTION

HTML (Hypertext Markup Language)

In the previous assignment you learned that FTP was just another language that computers use to communicate. The same holds true for HTML. HTML is used between computer and programs to communicate in a language designed to create web pages.

The Assignment

At the end of this assignment you should be able to...

- INTRODUCTION
- (1) Recognize HTML code
- (2) Understand the minimum requirements inside a HTML page
- (3) Know what the viewer sees and the system uses

(1) Recognize HTML Code

When looking at the text inside of a web page, you can easily separate the HTML code from the other information if you know what to look for. There are a few standards that have been imposed in the Internet Browser (i.e. Netscape, Internet Explorer, etc..) businesses that allow the developer (you) to create code that will work on any Internet Browser (from now on I'll just refer to them as **browsers**).

One of the most fundamental concepts/rules to HTML coding is to begin all commands with the ' **<**' symbol followed by the first letter of the command. The second part of that last sentence is important to remember, as you will soon see. The command must also end in the ' **>**' symbol.

One of the most used commands in HTML is the command:

<font>, this command will allow the developer (you again) to change the way text appears to the viewer. Here is why the second part of the above sentence was important! If you type a command and place a space after the **<** and before you

type the **font** command, the browsers will think you just want the 'less than' symbol displayed followed by the word 'font'. It will not understand your desire to affect the text. So remember this rule!!

For Example:

If I did the command right, I could do the following to some text:

Look at me Ma!

On the other hand, if I typed my HTML code in the wrong manner, I could end up with the following:

< font size=+2 color="#960000" >Look at me Ma!< /font >

By remembering this basic standard/rule with HTML commands, you can now easily pick out the commands from the text on (and others) web pages.

Tip: Now that you know how to dissect a web pages code, you can go to other websites that have tricks or formats you really like and use code just like theirs. Here is how you do it:

Go to the website that you like and go to the text toolbar at the top of the browser and ..
- (If you are using Netscape) click VIEW, then click on PAGE SOURCE.
- (If you are using Internet Exlorer) click VIEW, then SOURCE.

Violà! You can now see the code used to create any web page (including this one).

(2) Understand The Minimum Requirements Inside A HTML Page

All web pages have a minimum requirement as to what code is inside for the page to be displayed properly inside a browser. Without beating around the bush, here is the minimum code (there are many others that you can use to make your page sparkle, but these are the minimum):

```
<html>
<head>
<title>
</title>
</head>
<body>
</body>
</html>
```

Notice from point 1 above that the rules apply. Each command has started with the '<' symbol and immediately followed by the command (no spaces between). They all also end with the '>'symbol.

commands? Look at the code above and see if you can see a common pattern used with the commands....(it's okay I'll wait)

Did you notice that there is an identical copy of each command but with an extra '/' inserted between the '<' symbol and the command? This is a way of telling the browser that you have finished using a particular command.

Sometimes it also matters in what order you place commands, as with the minimum commands above. Now we can go into detail of what the above commands are and why they are in that order:

• **<html>**, this command tells a browser that you are starting a webpage. Notice that the **</html>** is at the very end of the code. The /html signifies to the browser that you have finished coding the webpage. Everything contained between the **<html>** and the **</html>** will be used in your webpage. Almost always you will never have anything above the **<html>** or below the **</html>**

• **<head>**, this command is the header (hence the **head** command). This area of code is always located right after the **<html>** command and contains information you provide to the browser. This area of code is not for the viewer to see, just for you to give information to the browser. Notice the **</head>** command follows just a couple lines further. You usually do not have much code inside the head. The most important and required code inside the **head** code is the **title** code.

• **<title>**, this command is always placed inside the **head** command. This command tells the browser what title to place at the top of the browsers window above the toolbar. It is also used by search engines to describe your webpage to customers looking for particular information.

For example, if you look at the top of this browser above the toolbar, you should be able to see the title **CIS 104 - Assignment 2**.

And as with **most** commands, you place the **</title>** at the end of what you want the title to be. To create the title for this page, I did the following HTML code:

<title>CIS 104 - Assignment 2</title>

• **<body>**, this command is used to tell the browser to start displaying information (text and graphics) to the viewer. Everything you place between the **<body>** and **</body>** command will affect what the viewer actually sees.

(3) Know What The Viewer Sees And The System Uses

Your finally at the stage where you are ready to begin your first web page assignment. To create a web page, you simply type in the command and information into any word processor. The best program for this purpose is Notepad. The reason most webmasters use Notepad instead of Word or other word processors is that Notepad only saves what you typed. On the other hand, Word for example will create a slew of other information/commands that you don't need. You can turn these off, but it is easier just to use Notepad.

To use Notepad, click on your START button, then PROGRAMS, then ACCESSORIES and finally Notepad.

After you open Notepad, just type in the basic web code above. When your ready to save, use the following trick to force your word processor (hopefully Notepad) to save it as a web page.

After you click on FILE, don't click on the SAVE button, but rather the SAVE AS button. This will allow you to choose how you want to save the file. Type your desired filename into the box (try not to use spaces) followed by '.htm'. For example, if I wanted a web page called dogcat, I could type **dogcat.htm**.

The last part of the trick is to force your word processor to keep the '.htm' on the filename. To do this, place quotes (") around the filename, like: **"dogcat.htm"**

Now save the file. You have just created a web page...pretty easy!

Another note on Notepad: When you want to edit your page later, Notepad will only try and open files that end in '.txt'. To force Notepad to show you web page files, type '***.htm**' into the filename box and click OPEN. This will make Notepad list any files in your current location that end in '.htm' (web pages). You could also change the FILE TYPE scroll list to ALL FILES and choose your desired web page, but this option will also show ALL files, not just web pages.

After you have the code looking exactly like the code above, place a title in the code that represents what the webpage is about. For this exercise I think we will create a page dealing with food (always a good subject to me). You will place the title you desire between the **<title>** and **</title>** commands. For example, I would have the following code my page:

```
<html>
<head>
```

```
<title>
All About Food
</title>
</head>
<body>
</body>
</html>
```

Of course I expect your title to be different to reflect your own style.

Now it is time to view your web page. Open up your Internet Browser and choose FILE from the menu, then OPEN. Find your web page on your computer and then double click it. Your page should open in your browser.

If you want to see how my page would look: Click Here

You should notice the page is blank (as yours will be), **except**: if you look at the top of the browser you should see your title. On my example page, you should see **All About Food**.