Designing Web Pages With HTML

PLEASE SILENCE CELL PHONES

Instructor: Sean Luster

What is HTML?

Hyper Text Markup Language

Also known as HTML, could be described as the programming language of the Internet. Sticklers will tell you that it's not technically a programming language and shouldn't be described as such but the fact remains that, whether directly or indirectly, HTML is used in designing virtually every page you encounter online.

How We Use HTML

Once you are familiar with writing HTML you'll be able to use any text editor (Notepad, WordPad, Word, etc.) to create .html files. There are programs designed to speed up and enhance the web page designing process (examples include Dream Weaver, FrontPage, and various visual studios), but what they are actually doing is typing in the HTML for you based on buttons you click and menu items you suggest. If you ever want to make a change to these pages (or add a feature not explicitly listed) it's very helpful, if not necessary, to know HTML.

What We Won't Be Doing Today

We're not going to create a web page accessible online from anywhere in the world. Here at the Bloomingdale Public Library we don't host web sites for patrons (or ourselves) and I decided against using a 3rd party company (similar to how I use Hotmail to teach Email classes) believing that this class should focus tightly on the HTML. If you're interested in doing that you should attend the class "Creating a Web Site", it's a recommended next step and can help you put the skills learned today to a more practical use. We will be using the program "Notepad" to create, edit, and save our web pages, and Internet Explorer to view and use them. At the end of this handout I have recommended a location for you to put these skills into action and create a web

page accessible anywhere. I've also included detailed information on how to use that site. I don't expect us to have time to go through that in class today but anyone who needs help is welcome to contact me in the future.

Basics

<u>Getting Started</u>

Creating an HTML file is very much like writing out any document. You need to plan the content, layout, and appearance; and then start working to put them into place. Of course computers have made making changes much easier so it's not difficult to simply experiment with all three factors and see what you like (both for WebPages and other documents). Still, things go much smoother when you have some clear ideas of what you want to do right from the start.

Once you're ready to get started there are two things you need to put into this document:

1. The text
2. The markup

The text is going to be the actual words and other information you want to appear on your web page. The markup will be the items, called tags, you will insert to control the display.

<u>Entering Information</u>

To write HTML files you first need somewhere to write them. You can purchase several elaborate HTML editors that will provide plenty of shortcuts and functions to save you time or help better explain HTML to you. These programs are great but if this isn't something you spend a lot of time working on or you just don't want to invest the money then you can simply use any text editor you want. Today we'll be using Notepad. You can find Notepad on your computer located at C:\Windows\Notepad.exe

To begin writing an HTML file simply open Notepad (you should see a blank space with a blinking cursor) and start typing in the information you want.

IMPORTANT: You can create and edit HTML files with any word processor. The important thing to remember is when saving the file you must save it with the extension ".html" or ".htm"

NOTE: The reason we're using Notepad in class today instead of a more advanced word processor such as MS Word or WordPerfect is I want to maintain as much control over what we type in as possible and more elaborate word processors often perform many tasks automatically that would make controlling the display difficult.

<u>General Page Design</u>

The order that we're going to create our page in is as follows:

1.   Header: This is where you give the title of the page and determine such "look/feel" features as background color and design and text color. Not to be confused with the top of the page.

2.   Body: Where you will put the content that appears in your web page. Pictures, text, links and everything else is tossed in here.

3.   Footer: Mostly closing off things that were opened up or started above. This isn't particularly important and will probably be clearer to you as we move through the material.

<u>Other Things You Should Know</u>

Remember these files won't be available to other people on the Internet until you have them saved and uploaded to a company that runs web sites. This doesn't mean the file is useless. Even if you never make it part of the Internet you can still view the file using Internet Explorer.

Also remember that not everything you type in will appear exactly as you entered it when viewed with a browser (Chrome, FireFox, etc.). We'll learn more about this as we go.

Tags

<u>What are they?</u>

Tags are the controls and commands used by HTML. When you write out a normal document you don't specifically tell the reader "this is my title", "this is a paragraph", "picture goes here", or anything like that. However when you write HTML that's exactly what you have to do.   We indicate these things by use of tags. Some tags perform functions all by themselves (for example, if you type "<BR>" that makes a new line; the equivalent of pressing the enter key in a word processor); others perform functions on what you put inside of or between the opening and closing parts of the tags (for example, typing "<B>Hi there</B>" makes the words "Hi there" appear in bold on the page). As you can see there are specific rules for entering in and using tags, to help you with this I've included a handout with all of the most common tags. We'll now examine how to use a number of these tags. Some that open up more complex topics are discussed on their own below.

<u>HTML</u>

This is the tag that you first enter in, this indicates to the computer that what follows will be written as HTML. Actual use of the tag would go as follows:

As the first line of the page type:

<HTML>

as the last line of the page type:

</HTML>

Notice how you begin the tag with the name in the '<>' signs and indicate the closing of the tag with a '/' in front of it. This is going to be a recurring theme.

<u>Head</u>

This is used to indicate the section of your HTML document controlling certain specific aspects of the page (title and body design, for example). In practice it would look like this:

<HEAD>

various header information typed in here

</HEAD>

Title

This tag has a very straightforward purpose. It controls what appears in the title bar (that dark blue bar that you use for moving the window). For example:

<TITLE>My First Page</TITLE>

Will make "My First Page – Microsoft Internet Explorer" appear in the title bar. The title tag is typically used within the header (i.e. between the <HEAD> tag's opening and closing).

Body

This is the most complex tag we've seen yet, within the body tag you control several elements of the web page. Here you determine what the background will look like, what color the text will be, how links will look, and how visited links will look. We'll discuss in detail in class but here is an example use of the body tag:

```
<BODY        background="images/stdsymbol.gif"
             text=#0
             link=#0
             vlink=#0
>
```

This would make a background matching the picture file "stdsymbol.gif" and black text with black links and black visited links.

Paragraph

The paragraph tag is used to show that you're starting a new paragraph, it's not strictly necessary but good form dictates that we use it. Example:

<P>

Everything we put in paragraph here

</P>

<u>Page Breaks</u>

One thing that doesn't carry over from your typing into the word processor to its appearance on the web page is the creation of new lines (what happens when you press "Enter"). If we don't take direct control over when new lines appear the text in our web pages will fit itself into the page in ways you would probably have never expected. To control the page breaks we have two tags at our disposal, the first ends the line. The second ends the line and puts a straight line ("----------") below it. Here are some examples:

This is the end of the line.

New Line

Last Line

Yields:

This is the end of the line.

New Line

Last Line

Now for the 2nd tag:

First Line<HR>

Second Line

This looks like this:

First Line

Second Line

Actually it looks much better than that on the web page, Word doesn't perform such tasks too easily.

<u>Font</u>

The Font tag allows you to control several elements of how your text appears. This is where you change the size, color, and style of your text. Example:

<FONT size = 20 color = "Yellow">Hello!

"Hello!" would appear in large, yellow text.

Other Formatting Tags

There are a number of tags that you use for making formatting changes, the kind of changes that are easily made within programs like Word. Here are a few examples:

<B>This text will be bold</B>

<CENTER>This text will be centered</CENTER>

<I>This text will be in italics</I>

<STRONG>Just like bold</STRONG>

<EM>Just like Italics</EM>

The rest of the tags we're going to learn are complex enough to merit their own section.

Links

Links are a large part of what makes the Internet work the way it does. A link is a section of a web page that connects the user to another web page when used. Creating a link only requires two bits of information:

1. What you're linking to
2. Where the link goes

Here's an example of how you would use a link:

<A HREF = "http://www.bloomingdale.lib.il.us">Library page</A>

Links can be to other pages within your own web site (we'll discuss) or they can be to any page accessible throughout the World Wide Web.

Images

Inserting images into your web page is very much like creating a link. Only difference is instead of linking to another web page file, you're linking to an image file. Here's a basic example of an image tag:

<IMG src="nameofpic.jpg">

A couple things to notice here,

1. This is the kind of tag you don't close (there's no </IMG>)

2. Inside the tag you have information regarding the picture, in this case we just entered in the name of the file we want displayed. We'll examine other things you can do within the tag next.

More settings you can control:

<IMG src="nameofpic.jpg" width=50 height=75 align=right>

In this tag you are controlling the width and height of the picture by specifying, in numbers of pixels, how big the image will be. You are also controlling the alignment of the image, by setting it to "right" this picture will now line up as far to the right as the page layout allows.

There are more complicated things that you can do with images, such as Image Mapping, that fall outside of the scope of this class.

Sounds

If you want you can insert sounds into your web pages. This is not often used for two reasons. First of all, audio files are large and will take up a big chunk of the space you have for your webpage. Secondly, the size of the sound files makes it difficult for most users to open your page quickly. However, if you still want to include a sound file in your web page we can do so to different ways.

Play Sound On Start Up

Just enter this line of code into the header and the sound will play as your page opens:

<bgsound src="name_of_sound_file.jpg">

If you want the sound to continue playing indefinitely make this change:

<bgsound src="name_of_sound_file.jpg" loop="infinite">

Play Sound On Click

This is getting into complex and interactive programming, involving the creation of buttons on your web page and scripts (we'll discuss later) to execute when those buttons are clicked.

Tables

Tables are used for organizing information and strictly controlling layouts. You can put whatever you want into a table, simple text, pictures, links, etc. Unlike everything we've done so far, creating a table is not done all in one step. It's an ongoing process that requires attention to detail. Here's a sample table (with line by line explanations underneath):

<TABLE border = 1 align = "center" valign = "top">

The "TABLE" tag opens a new table; we're setting the border width to '1', which will create a thin line around each item in the table. Alignment is set to "center" and vertical alignment is set to "top".

<TR>

This creates a new Row in the table.

<TD width = 75>

    Hi There, here's my picture→

</TD>

This section created the first box (or cell) of the first row and put the bit of dialogue explaining the neighboring image, then closed the first cell.

<TD width = 300>

    <IMG src="mypicture.jpg" width = 300>

</TD>

This section created the second cell and inserted an image into it. Notice that this cell is larger than the previous one.

</TR>

<TR>

This text closed the first row and started the second one.

<TD width = 375>

      <A HREF = http://www.bloomingdale.lib.il.us>This is where I work</A>

</TD>

This created a cell in the second row as wide as the one above it. In that cell we put a link to the library's web page. Notice that you can put different numbers of cells in different rows.

</TR>

</TABLE>

Here we close the tags for the second row and the tag for the whole table. It's very important to close all your cell, row, and table tags. Otherwise your page will have all sorts of hard to understand problems.

Lists

Back to a more straightforward topic, lists are for the many situations where you want to organize some form of information into a list. You can use this technique with everything we've discussed so far (text, links, images). To create a list first you need to declare that you're starting the list:

<UL>

Then for each item in the list you need to follow this form:

<LI>Item 1</LI>

<LI>Item 2</LI>

And so on until you are finished with the list, at which point you use the closing tag:

</UL>

Note: this is an unordered list, that means there will be a little dot next to each item but it won't be using any numbers or other counting devices. You do have other options available with lists, which I'd encourage you to explore (using numbered lists, creating menus, etc.).

Scripts

If you want to go on to making more advanced web pages one of the most effective ways to do this is to incorporate scripts. A script is a little bit of another programming language that you have running within your HTML. For example, if you had learned how to write in JavaScript you could then create a program using that language and embed it into the middle of your HTML file. Here's an example:

<SCRIPT Language = "Javascript">

type in everything involved in the program here

</SCRIPT>

<HTML>

<TITLE>JavaScript Practice</TITLE>

</HTML>

…..

The Learning Process

At this point I'd like to discuss what, I feel, is the best way to come to understand that big bulk of information above. Assembling a web page from scratch can be a bit intimidating, and time consuming. A better way to illustrate the basic lessons I want to get across to you is to examine and edit existing pages. Essentially this is a way to get across the causality of HTML. I'm hoping that over time you'll get the hang of trying things, seeing the results, and understanding the relationship between what you did and what happened. Naturally there are pitfalls and confusing spots that have to be overcome but we'll try to examine and bypass those today. First

I'd like to go over how to view what other people have done with their pages, then we'll edit some examples I've put together and look at the changes we make.

Learning From Existing Web Pages

One of the best ways to start learning HTML is to look at what others have done with it and try to replicate this with adjustments to fit what you'd like to do. To view the HTML that's making most websites run follow these steps:

1. Visit desired site

2. Open the "View" menu

3. Choose "Source"

This will open the Notepad program showing you the complete HTML for that page. The best way to use this information is to try to find an anchor. What I mean by that is some point of reference where you can begin figuring out what is going on and expand out from there. Whether it is text that's typed in or an HTML command you already know, just find something that you can identify the purpose for. We'll perform examples of this in class.

Some basic sites you can visit to view source:

Learning by Making Changes

The other way to get the cause and effect of HTML composing clearer in your mind is to make slight changes and watch the effect it has. We'll go through several exercises for this in class.

Recommended Sites

http://werbach.com/barebones/barebones.html

http://www.htmlgoodies.com/

http://www.pagetutor.com/

There's about 2 million more of these sites available, just go to www.google.com and search and you'll find help on any HTML topic.

Tag List

http://www.w3schools.com/tags/

http://www.quackit.com/html_5/tags/