

Introducing HTML

In its infancy HTML only supported the use of text but since those early days it has matured to facilitate the embedding of a wide range of video or interactive material into HTML documents. Whilst this material cannot actually be viewed using the browser itself many HELPER APPLICATIONS have been developed to be launched alongside the browser to open these files. A special type of helper application called a PLUG-IN can display files right inside the browser window, such as those produced by Macromedia to view Flash and Shockwave material.

The ability of HTML to support graphics and other media has made the World Wide Web the first service on the Internet that lends itself to any kind of graphic design. That, perhaps more than any other attribute, is what made it the first service on the Internet to grab the attention of the public, publishers and businesses. Thanks to its use of hyperlinks for navigation, it presents an exciting design problem that challenges designers to create fluid and friendly interfaces, as well as attractive graphic design, to help visitors find, scan, and enjoy the material published there.

Before we consider any elements of Web page design we should look at HTML.

In the early days of the Internet if you wanted to design a Web page there were no applications available to do the hard work of generating the HTML you had to type up the information yourself, complete with the appropriate tags, usually using a word processor.

Today's designers have many WYSIWYG (What You See Is What You Get) HTML generators available to them, such as Macromedia DreamWeaver. As a consequence you can get by without understanding or writing a single line of code. It is, however, a good idea to learn the principles behind HTML to understand what is it possible to achieve and also to allow you to work with colleagues who may code directly with HTML – there are more of them around than you think!

We will start to look at some basic HTML tags so that we may begin to get a feel of how they are used. Look in the folder 'html examples *f*' and view the example files as Simple Text files and in a browser to see how the files appear. Alternately, you can view the pages directly in a browser and then select the 'View Source' or 'View Page Source' option. This is probably the best way to learn how to work with HTML. Whenever you come across Web page layouts that you like, save them to disk and then examine the HTML at leisure. Once you get comfortable with some of the tags you should be able to deconstruct how these pages were made.

A typical HTML document normally has the following structure:

```
<HTML>
<HEAD>
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
</HEAD>

<BODY>
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXX
</BODY>
</HTML>
```

Each document has two main sections: the HEAD which gives the browser general information about the file, and the BODY, which includes the information that will be displayed in the browser window. The head of the document is delineated by <HEAD> and </HEAD/> tags and the body is delineated by <BODY and </BODY> tags. Some browsers also require an <HTML> tag at the top of the file and a </HTML> tag at the

end, which tell the browser to interpret the file as HTML and not plain text. All other HTML tags fall within these delimiters.

Let's look at two typical tags:

`<BOLD> This text will appear bold </BOLD>`

`<BOLD>` is the start tag defining the beginning of an element
The text that the tag applies to goes between the start and end tags
`</BOLD>` is the end tag defining the end of an element. The end tag is a repetition of the start tag, but with a slash (/) character.

Another common tag is:

` applicable text `

`size=3` is an ATTRIBUTE and adds additional information about how the tag should be interpreted. Attributes are added inside the start tag delimiters. Many attribute names are followed by an equal sign and then by the value of the attribute. In this case the attribute sets the font size at which the tagged text will be displayed.

Sometimes you will find some code in HTML documents that doesn't appear to actually be HTML. To explain why that is it may be a good idea to explain another term you may come across. The term you may hear being used is 'dynamic HTML.'

Dynamic HTML is actually a collection of different technologies:

HTML

JAVASCRIPT – A scripting language that extends the capabilities of HTML

CASCADING STYLE SHEETS – A page layout system understood by newer web browsers, which allows for better control over the appearance and position of elements on a web page.

DOCUMENT OBJECT MODEL – A method used to connect to outside scripting programs (e.g. ActiveX) or Plug-Ins such as Shockwave or Flash.

We will look at these more closely at a later date.