

# HTML Basics

Welcome to HTML Basics. This workshop leads you through the basics of Hyper Text Markup Language (HTML). HTML is the building block for web pages. You will learn to use HTML to author an HTML page to display in a web browser.

## Objectives:

By the end of this workshop, you will be able to:

- Use a text editor to author an HTML document.
- Be able to use basic tags to denote paragraphs, emphasis or special type.
- Create hyperlinks to other documents.
- Create an email link.
- Add images to your document.
- Use a table for layout.
- Apply colors to your HTML document.

## Prerequisites:

You will need a text editor, such as Notepad and an Internet browser, such as Internet Explorer or Netscape.

**Q:** What is Notepad and where do I get it?

**A:** Notepad is the default Windows text editor. On most Windows systems, click your Start button and choose Programs then Accessories. It should be a little blue notebook.

**Mac Users:** SimpleText is the default text editor on the Mac. In OSX use TextEdit and change the following preferences: Select (in the preferences window) Plain text instead of Rich text and then select Ignore rich text commands in HTML files. This is very important because if you don't do this HTML codes probably won't work.

One thing you should avoid using is a word processor (like Microsoft Word) for authoring your HTML documents.

## What is an html File?

HTML is a format that tells a computer how to display a web page. The documents themselves are plain text files with special "tags" or codes that a web browser uses to interpret and display information on your computer screen.

- HTML stands for Hyper Text Markup Language
- An HTML file is a text file containing small markup tags
- The markup tags tell the Web browser how to display the page
- An HTML file must have an htm or html file extension

## Try It?

Open your text editor and type the following text:

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body>
This is my first homepage. <b>This text is bold</b>
</body>
</html>
```

Save the file as **mypage.html**. Start your Internet browser. Select **Open** (or Open Page) in the **File** menu of your browser. A dialog box will appear. Select **Browse** (or Choose File) and locate the html file you just created - **mypage.html** - select it and click **Open**. Now you should see an address in the

dialog box, for example **C:\MyDocuments\mypage.html**. Click **OK**, and the browser will display the page. To view how the page should look, visit this web page:  
**<http://profdevtrain.austincc.edu/html/mypage.html>**

### **Example Explained**

What you just made is a skeleton html document. This is the minimum required information for a web document and all web documents should contain these basic components. The first tag in your html document is `<html>`. This tag tells your browser that this is the start of an html document. The last tag in your document is `</html>`. This tag tells your browser that this is the end of the html document.

The text between the `<head>` tag and the `</head>` tag is header information. Header information is not displayed in the browser window.

The text between the `<title>` tags is the title of your document. The `<title>` tag is used to uniquely identify each document and is also displayed in the title bar of the browser window.

The text between the `<body>` tags is the text that will be displayed in your browser.

The text between the `<b>` and `</b>` tags will be displayed in a bold font.

### **HTM or HTML Extension?**

When you save an HTML file, you can use either the .htm or the .html extension. The .htm extension comes from the past when some of the commonly used software only allowed three letter extensions. It is perfectly safe to use either .html or .htm, but be consistent. **mypage.htm** and mypage.html are treated as different files by the browser.

### **How to View HTML Source**

A good way to learn HTML is to look at how other people have coded their html pages. To find out, simply click on the View option in your browsers toolbar and select Source or Page Source. This will open a window that shows you the actual HTML of the page. Go ahead and view the source html for this page.

## **HTML Tags**

### **What are HTML tags?**

- HTML tags are used to mark-up HTML elements
- HTML tags are surrounded by the two characters `<` and `>`
- The surrounding characters are called angle brackets
- HTML tags normally come in pairs like `<b>` and `</b>`
- The first tag in a pair is the start tag, the second tag is the end tag
- The text between the start and end tags is the element content
- HTML tags are not case sensitive, `<b>` means the same as `<B>`

### **Logical vs. Physical Tags**

In HTML there are both logical tags and physical tags. Logical tags are designed to describe (to the browser) the enclosed text's meaning. An example of a logical tag is the `<strong>` `</strong>` tag. By placing text in between these tags you are telling the browser that the text has some greater importance. By default all browsers make the text appear bold when in between the `<strong>` and `</strong>` tags.

Physical tags on the other hand provide specific instructions on how to display the text they enclose. Examples of physical tags include:

- `<b>`: Makes the text bold.
- `<big>`: Makes the text usually one size bigger than what's around it.
- `<i>`: Makes text italic.

Physical tags were invented to add style to HTML pages because style sheets were not around, though the original intention of HTML was to not have physical tags. Rather than use physical tags to style your HTML pages, you should use style sheets.

## HTML Elements

Remember the HTML example from the previous page:

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body>
This is my first homepage. <b>This text is bold</b>
</body>
</html>
```

This is an HTML element:

```
<b>This text is bold</b>
```

The HTML element begins with a start tag: `<b>`

The content of the HTML element is: This text is bold

The HTML element ends with an end tag: `</b>`

The purpose of the `<b>` tag is to define an HTML element that should be displayed as bold.

This is also an HTML element:

```
<body>
This is my first homepage. <b>This text is bold</b>
</body>
```

This HTML element starts with the start tag `<body>`, and ends with the end tag `</body>`. The purpose of the `<body>` tag is to define the HTML element that contains the body of the HTML document.

## Nested Tags

You may have noticed in the example above, the `<body>` tag also contains other tags, like the `<b>` tag. When you enclose an element in with multiple tags, the last tag opened should be the first tag closed. For example:

```
<p><b><em>This is NOT the proper way to close nested tags.</em></b></p>
```

```
<p><b><em>This is the proper way to close nested tags.</em></b></p>
```

<b>Note:</b> It doesn't matter which tag is first, but they must be closed in the proper order.
---

## Why Use Lowercase Tags?

You may notice we've used lowercase tags even though I said that HTML tags are not case sensitive. `<B>` means the same as `<b>`. The World Wide Web Consortium (W3C), the group responsible for developing web standards, recommends lowercase tags in their HTML 4 recommendation, and XHTML (the next generation HTML) requires lowercase tags.

## Tag Attributes

Tags can have attributes. Attributes can provide additional information about the HTML elements on your page. The `<tag>` tells the browser to do something, while the attribute tells the browser how to do it. For instance, if we add the `bgcolor` attribute, we can tell the browser that the background color of your page should be blue, like this: `<body bgcolor="blue">`.

This tag defines an HTML table: `<table>`. With an added border attribute, you can tell the browser that the table should have no borders: `<table border="0">`. Attributes always come in name/value pairs like this: `name="value"`. Attributes are always added to the start tag of an HTML element and the value is surrounded by quotes.

### Quote Styles, "red" or 'red'?

Attribute values should always be enclosed in quotes. Double style quotes are the most common, but single style quotes are also allowed. In some rare situations, like when the attribute value itself contains quotes, it is necessary to use single quotes:

`name='George "machine Gun" Kelly'`

**Note:** Some tags we will discuss are deprecated, meaning the World Wide Web Consortium (W3C) the governing body that sets HTML, XML, CSS, and other technical standards decided those tags and attributes are marked for deletion in future versions of HTML and XHTML. Browsers should continue to support deprecated tags and attributes, but eventually these tags are likely to become obsolete and so future support cannot be guaranteed.

For a complete list of tags, visit [W3C.org](http://W3C.org).

## Basic HTML Tags

The most important tags in HTML are tags that define headings, paragraphs and line breaks.

### Basic HTML Tags

Tag	Description
<code>&lt;html&gt;</code>	Defines an HTML document
<code>&lt;body&gt;</code>	Defines the document's body
<code>&lt;h1&gt;</code> to <code>&lt;h6&gt;</code>	Defines header 1 to header 6
<code>&lt;p&gt;</code>	Defines a paragraph
<code>&lt;br&gt;</code>	Inserts a single line break
<code>&lt;hr&gt;</code>	Defines a horizontal rule
<code>&lt;!--&gt;</code>	Defines a comment

### Headings

Headings are defined with the `<h1>` to `<h6>` tags. `<h1>` defines the largest heading while `<h6>` defines the smallest.

**`<h1>This is a heading</h1>`**

**`<h2>This is a heading</h2>`**

**`<h3>This is a heading</h3>`**

**`<h4>This is a heading</h4>`**

**`<h5>This is a heading</h5>`**

**`<h6> This is a heading</h6>`**

HTML automatically adds an extra blank line before and after a heading. A useful heading attribute is `align`.

```
<h5 align="left">I can align headings </h5>
```

```
<h5 align="center">This is a centered heading </h5>
```

```
<h5 align="right">This is a heading aligned to the right </h5>
```

## Paragraphs

Paragraphs are defined with the `<p>` tag. Think of a paragraph as a block of text. You can use the align attribute with a paragraph tag as well.

```
<p align="left">This is a paragraph</p>
```

```
<p align="center">this is another paragraph</p>
```

**Important:** You must indicate paragraphs with `<p>` elements. A browser ignores any indentations or blank lines in the source text. Without `<p>` elements, the document becomes one large paragraph. HTML automatically adds an extra blank line before and after a paragraph.

## Line Breaks

The `<br>` tag is used when you want to start a new line, but don't want to start a new paragraph. The `<br>` tag forces a line break wherever you place it. It is similar to single spacing in a document.

This Code	Would Display
<code>&lt;p&gt;This &lt;br&gt; is a para&lt;br&gt; graph with line breaks&lt;/p&gt;</code>	This is a para graph with line breaks

The `<br>` tag has no closing tag.

## Horizontal Rule

The `<hr>` element is used for horizontal rules that act as dividers between sections, like this:

The horizontal rule does not have a closing tag. It takes attributes such as align and width. For instance:

This Code	Would Display
<code>&lt;hr width="50%" align="center"&gt;</code>	<hr/>

## Comments in HTML

The comment tag is used to insert a comment in the HTML source code. A comment can be placed anywhere in the document and the browser will ignore everything inside the brackets. You can use comments to write notes to yourself, or write a helpful message to someone looking at your source code.

This Code	Would Display
<code>&lt;p&gt; This html comment would &lt;!-- This is a comment --&gt; be displayed like this.&lt;/p&gt;</code>	This HTML comment would be displayed like this.

Notice you don't see the text between the tags `<!--` and `-->`. If you look at the source code, you would see the comment. To view the source code for this page, in your browser window, select **View** and then select **Source**.

**Note:** You need an exclamation point after the opening bracket <!-- but not before the closing bracket -->.

HTML automatically adds an extra blank line before and after some elements, like before and after a paragraph, and before and after a heading. If you want to insert blank lines into your document, use the <br> tag.

### **Try It Out!**

Open your text editor and type the following text:

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body>
<h1 align="center">My First Webpage</h1>
<p>Welcome to my first web page. I am writing this page using a text editor and plain
old html.</p>
<p>By learning html, I'll be able to create web pages like a pro....<br>
which I am of course.</p>
</body>
</html>
```

Save the page as **mypage2.html**. Open the file in your Internet browser. To view how the page should look, visit this web page: **<http://profdevtrain.austincc.edu/html/mypage2.html>**

### **Other HTML Tags**

As mentioned before, there are logical styles that describe what the text should be and physical styles which actually provide physical formatting. It is recommended to use the logical tags and use style sheets to style the text in those tags.

#### **Logical Tags**

Tag	Description
<abbr>	Defines an abbreviation
<acronym>	Defines an acronym
<address>	Defines an address element
<cite>	Defines a <i>citation</i>
<code>	Defines computer code text
<blockquote>	Defines a long quotation
<del>	Defines text
<dfn>	Defines a <i>definition</i> term
<em>	Defines <i>emphasized</i> text
<ins>	Defines inserted text
<kbd>	Defines keyboard text
<pre>	Defines preformatted text
<q>	Defines a short quotation
<samp>	Defines sample computer code
<strong>	Defines <b>strong</b> text
<var>	Defines a <i>variable</i>

#### **Physical Tags**

Tag	Description
<b>	Defines <b>bold</b> text
<big>	Defines <b>big</b> text
<i>	Defines <i>italic</i> text
<small>	Defines small text
<sup>	Defines <sup>superscripted</sup> text
<sub>	Defines <sub>subscripted</sub> text
<tt>	Defines teletype text
<u>	Deprecated. Use styles instead

Character tags like <strong> and <em> produce the same physical display as <b> and <i> but are more uniformly supported across different browsers.

### Some Examples:

The following paragraph uses the `<blockquote>` tag. In the previous sentence, the blockquote tag is enclosed in the `<sample>` Sample tag.

We the people of the United States, in order to form a more perfect Union, establish Justice, insure domestic Tranquility, provide for the common defense, promote the general Welfare, and secure the Blessings of Liberty to ourselves and our Posterity, do ordain and establish this Constitution for the United States of America.

Although most browsers render blockquoted text by indenting it, that's not specifically what it's designed to do. It's conceivable that some future browser may render blockquoted text in some other way. However, for the time being, it is perfectly safe to indent blocks of text with the `<blockquote>`.

This Code	Would Display
<code>&lt;abbr title="World Wide Web"&gt;WWW&lt;/abbr&gt;</code>	<u>WWW</u>

When you hold your mouse pointer over the WWW, text in the title attribute will appear in.

## HTML Character Entities

Some characters have a special meaning in HTML, like the less than sign (`<`) that defines the start of an HTML tag. If we want the browser to actually display these characters we must insert character entities in place of the actual characters themselves.

### The Most Common Character Entities:

Result	Description	Entity Name	Entity Number
	non-breaking space	<code>&amp;nbsp;</code>	<code>&amp;#160;</code>
<code>&lt;</code>	less than	<code>&amp;lt;</code>	<code>&amp;#60;</code>
<code>&gt;</code>	greater than	<code>&amp;gt;</code>	<code>&amp;#62;</code>
<code>&amp;</code>	ampersand	<code>&amp;amp;</code>	<code>&amp;#38;</code>
<code>"</code>	quotation mark	<code>&amp;quot;</code>	<code>&amp;#34;</code>
<code>'</code>	apostrophe	<code>&amp;apos;</code> (does not work in IE)	<code>&amp;#39;</code>

A character entity has three parts: an ampersand (`&`), an entity name or an entity number, and finally a semicolon (`;`). The `&` means we are beginning a special character, the `;` means ending a special character and the letters in between are sort of an abbreviation for what it's for. To display a less than sign in an HTML document we must write: `&lt;` or `&#60;`; The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.

**Note:** Entities are case sensitive.

### Non-breaking Space

The most common character entity in HTML is the non-breaking space `&nbsp;`. Normally HTML will truncate spaces in your text. If you add 10 spaces in your text, HTML will remove 9 of them. To add spaces to your text, use the `&nbsp;` character entity.

This Code	Would Display
<code>&lt;p&gt; This code would appear as this.&lt;/p&gt;</code>	This code would appear as this.





```

<html>
<head>
<title>My First Webpage</title>
</head>
<body background="http://profdevtrain.austincc.edu/html/graphics/clouds.gif"
bgcolor="#EDDD9E">
<h1 align="center">My First Webpage</h1>
<p>Welcome to my <strong>first</strong> webpage. I am writing this page using a text
editor and plain old html.</p>
<p>By learning html, I'll be able to create webpages like a <del>beginner</del>
pro....<br>
which I am of course.</p>
</body>
</html>

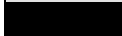








```

Save your page as **mypage3.html** and view it in your browser. To view how the page should look, visit this web page: **<http://profdevtrain.austincc.edu/html/mypage3.html>**  
Notice we gave our page a background color as well as a background image. If for some reason the web page is unable to find the picture, it will display our background color.

## HTML Colors

### Color Values

Colors are defined using a hexadecimal notation for the combination of red, green, and blue color values (RGB). The lowest value that can be given to one light source is 0 (hex #00). The highest value is 255 (hex #FF). This table shows the result of combining red, green, and blue:

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

### Color Names

A collection of color names is supported by most browsers. To view a table of color names that are supported by most browsers visit this web page:

**[http://profdevtrain.austincc.edu/html/color\\_names.htm](http://profdevtrain.austincc.edu/html/color_names.htm)**

**Note:** Only 16 **color names** are supported by the W3C HTML 4.0 standard (aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow). For all other colors you should use the **Color HEX** value.

Color	Color HEX	Color Name
	#F0F8FF	AliceBlue
	#FAEBD7	AntiqueWhite
	#7FFFD4	Aquamarine
	#000000	Black
	#0000FF	Blue
	#8A2BE2	BlueViolet
	#A52A2A	Brown

## Web Safe Colors

A few years ago, when most computers supported only 256 different colors, a list of 216 Web Safe Colors was suggested as a Web standard. The reason for this was that the Microsoft and Mac operating system used 40 different "reserved" fixed system colors (about 20 each). This 216 cross platform web safe color palette was originally created to ensure that all computers would display all colors correctly when running a 256 color palette. To view the 216 Cross Platform Colors visit this web page: <http://profdevtrain.austincc.edu/html/216.html>

## 16 Million Different Colors

The combination of Red, Green and Blue values from 0 to 255 gives a total of more than 16 million different colors to play with (256 x 256 x 256). Most modern monitors are capable of displaying at least 16,384 different colors. To assist you in using color schemes, check out <http://wellstyled.com/tools/colorscheme2/index-en.html>. This site lets you test different color schemes for page backgrounds, text and links.

## HTML Lists

HTML provides a simple way to show unordered lists (bullet lists) or ordered lists (numbered lists).

### Unordered Lists

An unordered list is a list of items marked with bullets (typically small black circles). An unordered list starts with the `<ul>` tag. Each list item starts with the `<li>` tag.

This Code	Would Display
<pre>&lt;ul&gt; &lt;li&gt;Coffee&lt;/li&gt; &lt;li&gt;Milk&lt;/li&gt; &lt;/ul&gt;</pre>	<ul style="list-style-type: none"><li>▪ Coffee</li><li>▪ Milk</li></ul>

### Ordered Lists

An ordered list is also a list of items. The list items are marked with numbers. An ordered list starts with the `<ol>` tag. Each list item starts with the `<li>` tag.

This Code	Would Display
<pre>&lt;ol&gt; &lt;li&gt;Coffee&lt;/li&gt; &lt;li&gt;Milk&lt;/li&gt; &lt;/ol&gt;</pre>	<ol style="list-style-type: none"><li>1. Coffee</li><li>2. Milk</li></ol>

Inside a list item you can put paragraphs, line breaks, images, links, other lists, etc.

### Definition Lists

Definition lists consist of two parts: a **term** and a **description**. To mark up a definition list, you need three HTML elements; a container `<dl>`, a definition term `<dt>`, and a definition description `<dd>`.

This Code	Would Display
<pre>&lt;dl&gt; &lt;dt&gt;Cascading Style Sheets&lt;/dt&gt; &lt;dd&gt;Style sheets are used to provide presentational suggestions for documents marked up in HTML. &lt;/dd&gt; &lt;/dl&gt;</pre>	<div>Cascading Style Sheets</div> <div>Style sheets are used to provide presentational suggestions for documents marked up in HTML.</div>

Inside a definition-list definition (the `<dd>` tag) you can put paragraphs, line breaks, images, links, other lists, etc

## Try It Out

Open your text editor and type the following:

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body bgcolor="#EDDD9E">
<h1 align="center">My First Webpage</h1>
<p>Welcome to my <strong>first</strong> webpage. I am writing this page using a text
editor and plain old html.</p>
<p>By learning html, I'll be able to create web pages like a pro....<br>
which I am of course.</p>
Here's what I've learned:
<ul>
<li>How to use HTML tags</li>
<li>How to use HTML colors</li>
<li>How to create Lists</li>
</ul>
</body>
</html>
```

Save your page as **mypage4.html** and view it in your browser. To see how your page should look visit this web page: **<http://profdevtrain.austincc.edu/html/mypage4.html>**

## HTML Links

HTML uses the <a> anchor tag to create a link to another document or web page.

### The Anchor Tag and the Href Attribute

An anchor can point to any resource on the Web: an HTML page, an image, a sound file, a movie, etc. The syntax of creating an anchor:

```
<a href="url">Text to be displayed</a>
```

The <a> tag is used to create an anchor to link from, the href attribute is used to tell the address of the document or page we are linking to, and the words between the open and close of the anchor tag will be displayed as a hyperlink.

This Code	Would Display
<code>&lt;a href="http://www.austincc.edu/"&gt;Visit ACC!&lt;/a&gt;</code>	<a href="http://www.austincc.edu/">Visit ACC!</a>

### The Target Attribute

With the target attribute, you can define **where** the linked document will be opened. By default, the link will open in the current window. The code below will open the document in a new browser window:

```
<a href=http://www.austincc.edu/ target="_blank">Visit ACC!</a>
```

### Email Links

To create an email link, you will use mailto: plus your email address. Here is a link to ACC's Help Desk:

```
<a href="mailto:helpdesk@austincc.edu">Email Help Desk</a>
```

To add a subject for the email message, you would add ?subject= after the email address. For example:

```
<a href="mailto:helpdesk@austincc.edu?subject=Email Assistance">Email Help Desk</a>
```

## The Anchor Tag and the Name Attribute

The name attribute is used to create a named anchor. When using named anchors we can create links that can jump directly to a specific section on a page, instead of letting the user scroll around to find what he/she is looking for. Unlike an anchor that uses href, a named anchor doesn't change the appearance of the text (unless you set styles for that anchor) or indicate in any way that there is anything special about the text. Below is the syntax of a named anchor:

```
<a name="top">Text to be displayed</a>
```

To link directly to the top section, add a # sign and the name of the anchor to the end of a URL, like this:

This Code	Would Display
<pre>&lt;a href="http://profdevtrain.austincc.edu/html/10links.html#top"&gt;Back to top of page &lt;/a&gt;</pre> <p>A hyperlink to the top of the page from within the file 10links.html will look like this:</p> <pre>&lt;a href="#top"&gt;Back to top of page &lt;/a&gt;</pre>	<p><a href="#">Back to top of page</a></p> <p><a href="#">Back to top of page</a></p>

**Note:** Always add a trailing slash to subfolder references. If you link like this: href="http://profdevtrain.austincc.edu/html", you will generate two HTTP requests to the server, because the server will add a slash to the address and create a new request like this: href="http://profdevtrain.austincc.edu/html/"

Named anchors are often used to create "table of contents" at the beginning of a large document. Each chapter within the document is given a named anchor, and links to each of these anchors are put at the top of the document. If a browser cannot find a named anchor that has been specified, it goes to the top of the document. No error occurs.

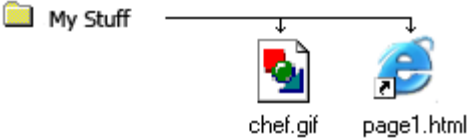
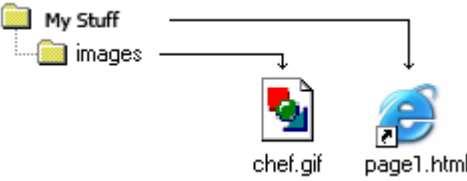


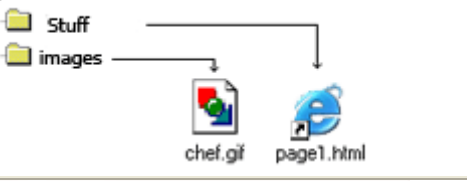

## HTML Images

### The Image Tag and the Src Attribute

The <img> tag is empty, which means that it contains attributes only and it has no closing tag. To display an image on a page, you need to use the src attribute. Src stands for "source". The value of the src attribute is the URL of the image you want to display on your page. The syntax of defining an image:

This Code	Would Display
<pre>&lt;img src="graphics/chef.gif"&gt;</pre>	

Not only does the source attribute specify what image to use, but where the image is located. The above image, graphics/chef.gif, means that the browser will look for the image name **chef.gif** in a **graphics** folder in the same folder as the html document itself.

	<p><code>src="chef.gif"</code> means that the image is in the same folder as the html document calling for it.</p>
	<p><code>src="images/chef.gif"</code> means that the image is one folder down from the html document that called for it. This can go on down as many layers as necessary.</p>
	<p><code>src="../../chef.gif"</code> means that the image is in one folder up from the html document that called for it.</p>
	<p><code>src="../../../chef.gif"</code> means that the image is two folders up from the html document that called for it.</p>
	<p><code>src="../../images/chef.gif"</code> means that the image is one folder up and then another folder down in the images directory.</p>
	<p><code>src="../../../../other/images/chef.gif"</code> means this goes multiple layers up.</p>

The browser puts the image where the image tag occurs in the document. If you put an image tag between two paragraphs, the browser shows the first paragraph, then the image, and then the second paragraph.

## **The Alt Attribute**

The alt attribute is used to define an alternate text for an image. The value of the alt attribute is author-defined text:

```

```

The alt attribute tells the reader what he or she is missing on a page if the browser can't load images. The browser will then display the alternate text instead of the image. It is a good practice to include the alt attribute for each image on a page, to improve the display and usefulness of your document for people who have text-only browsers or use screen readers.

## **Image Dimensions**

When you have an image, the browser usually figures out how big the image is all by itself. If you put in the image dimensions in pixels however, the browser simply reserves a space for the image, then

loads the rest of the page. Once the entire page is loads it can go back and fill in the images. Without dimensions, when it runs into an image, the browser has to pause loading the page, load the image, then continue loading the page. The chef image would then be:

```

```

Open the file **mypage2.html** in your text editor and add code highlighted in bold:

```
<html>
<head>
<title>My First Webpage</title>
</head>
<body>
<h1 align="center">My First Web page</h1>
<p>Welcome to my first webpage. I am writing this page using a text editor and plain old html.</p>
<p>By learning html, I'll be able to create web pages like a pro....<br>
which I am of course.</p>
<!-- Who would have guessed how easy this would be :) -->
<p></p>
<p align="center">This is my Chef</p>
</body>
</html>
```

Save your page as **mypage5.html** and view it in your browser. To see how your page should look visit this web page: <http://profdevtrain.austincc.edu/html/mypage5.html>

Tables

Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag). The letters td stands for table data, which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

This Code	Would Display
<table> <tr> <td>row 1, cell 1</td> <td>row 1, cell 2</td> </tr> <tr> <td>row 2, cell 1</td> <td>row 2, cell 2</td> </tr> </table>	row 1, cell 1 row 1, cell 2 row 2, cell 1 row 2, cell 2

Tables and the Border Attribute

To display a table with borders, you will use the border attribute.

This Code	Would Display		
<table border="1"> <tr> <td>Row 1, cell 1</td> <td>Row 1, cell 2</td> </tr> </table>	<table border="1"><tr><td>row 1, cell 1</td><td>row 1, cell 2</td></tr></table>	row 1, cell 1	row 1, cell 2
row 1, cell 1	row 1, cell 2		

and....

This Code	Would Display		
<pre>&lt;table border="5"&gt; &lt;tr&gt; &lt;td&gt;Row 1, cell 1&lt;/td&gt; &lt;td&gt;Row 1, cell 2&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;</pre>	<table border="5"><tr><td>row 1, cell 1</td><td>row 1, cell 2</td></tr></table>	row 1, cell 1	row 1, cell 2
row 1, cell 1	row 1, cell 2		

Open up your text editor. Type in your `<html>`, `<head>` and `<body>` tags. From here on I will only be writing what goes between the `<body>` tags. Type in the following:

```
<table border="1">
<tr>
<td>Tables can be used to layout information</td>
<td>&nbsp;  &nbsp; </td>
</tr>
</table>
```

Save your page as **mytable1.html** and view it in your browser. To see how your page should look visit this web page: <http://profdevtrain.austincc.edu/html/mytable1.html>

### Headings in a Table

Headings in a table are defined with the `<th>` tag.

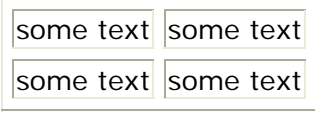
This code	Would Display						
<pre>&lt;table border="1"&gt; &lt;tr&gt; &lt;th&gt;Heading&lt;/th&gt; &lt;th&gt;Another Heading&lt;/th&gt; &lt;/tr&gt; &lt;tr&gt; &lt;td&gt;row 1, cell 1&lt;/td&gt; &lt;td&gt;row 1, cell 2&lt;/td&gt; &lt;/tr&gt; &lt;tr&gt; &lt;td&gt;row 2, cell 1&lt;/td&gt; &lt;td&gt;row 2, cell 2&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;</pre>	<table border="1"><tr><th>Heading</th><th>Another Heading</th></tr><tr><td>row 1, cell 1</td><td>row 1, cell 2</td></tr><tr><td>row 2, cell 1</td><td>row 2, cell 2</td></tr></table>	Heading	Another Heading	row 1, cell 1	row 1, cell 2	row 2, cell 1	row 2, cell 2
Heading	Another Heading						
row 1, cell 1	row 1, cell 2						
row 2, cell 1	row 2, cell 2						

### Cell Padding and Spacing

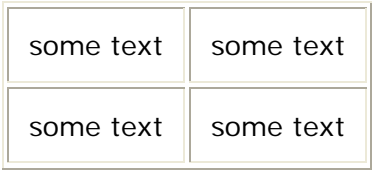
The `<table>` tag has two attributes known as cellspacing and cellpadding. Here is a table example without these properties. These properties may be used separately or together.

This Code	Would Display				
<pre>&lt;table border="1"&gt; &lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt; &lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;</pre>	<table border="1"><tr><td>some text</td><td>some text</td></tr><tr><td>some text</td><td>some text</td></tr></table>	some text	some text	some text	some text
some text	some text				
some text	some text				

Cellspacing is the pixel width between the individual data cells in the table (The thickness of the lines making the table grid). The default is zero. If the border is set at 0, the cellspacing lines will be invisible.

This Code	Would Display
<pre>&lt;table border="1" cellspacing="5"&gt; &lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt;&lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;</pre>	

Cellpadding is the pixel space between the cell contents and the cell border. The default for this property is also zero. This feature is not used often, but sometimes comes in handy when you have your borders turned on and you want the contents to be away from the border a bit for easy viewing. Cellpadding is invisible, even with the border property turned on. Cellpadding can be handled in a style sheet.

This Code	Would Display
<pre>&lt;table border="1" cellpadding="10"&gt; &lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt;&lt;tr&gt; &lt;td&gt;some text&lt;/td&gt; &lt;td&gt;some text&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;</pre>	

## Table Tags

Tag	Description
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell
<caption>	Defines a table caption
<colgroup>	Defines groups of table columns
<col>	Defines the attribute values for one or more columns in a table

## Table Size

### Table Width

The width attribute can be used to define the width of your table. It can be defined as a fixed width or a relative width. A fixed table width is one where the width of the table is specified in pixels. For example, this code, `<table width="550">`, will produce a table that is 550 pixels wide. A relative table width is specified as a percentage of the width of the visitor's viewing window. Hence this code, `<table width="80%">`, will produce a table that occupies 80 percent of the screen.

This table width is 250 pixels
--------------------------------

This table width is 50%
-------------------------



There are arguments in favor of giving your tables a relative width because such table widths yield pages that work regardless of the visitor's screen resolution. For example, a table width of 100% will always span the entire width of the browser window whether the visitor has a 800x600 display or a 1024x768 display (etc). Your visitor never needs to scroll horizontally to read your page, something that is regarded by most people as being very annoying.

## HTML Layout - Using Tables

One very common practice with HTML, is to use HTML tables to format the layout of an HTML page.

A part of this page is formatted with two columns. As you can see on this page, there is a left column and a right column.

This text is displayed in the left column.

An HTML <table> is used to divide a part of this Web page into two columns.

The trick is to use a table without borders, and maybe a little extra cell-padding.

No matter how much text you add to this page, it will stay inside its column borders.

### Try It Out!

Let's put everything you've learned together to create a simple page. Open your text editor and type the following text:

```
<html>
<head>
<title>My First Web Page </title>
</head>
<body>
<table width="90%" cellpadding="5" cellspacing="0" >
  <tr bgcolor="#EDDD9E">
    <td width="200" valign="top"></td>
    <td valign="top"><h1 align="right">Janet Doeson</h1>
    <h3 align="right">Technical Specialist</h3></td>
  </tr>
  <tr>
    <td width="200">
      <h3>Menu</h3>
      <ul>
        <li><a href="home.html">Home</a></li>
        <li> <a href="faq.html">FAQ</a></li>
        <li> <a href="contact.html">Contact</a></li>
        <li> <a href="http://www.austincc.edu">Links</a> </li>
      </ul></td>
    <td valign="top"><h2 align="center">Welcome!</h2>
    <p>Welcome to my first webpage. I created this webpage without the assistance of a
webpage editor. Just my little text editor and a keen understanding of html.</p>
    <p>Look around. Notice I'm able to use paragraphs, lists and headings. You may not
be able to tell, but the layout is done with a table. I'm very clever. </p>
    <blockquote>
      <p>I always wanted to be somebody, but now I realize I should have been more
specific.</p>
      <cite>Lily Tomlin </cite> </blockquote>
    </td>
  </tr>
</table>
<hr width="90%" align="left">
<address>
  Janet Doeson<br>
  Technical Specialist<br>
  512.555.5555
</address>
<p>Contact me at <a href="mailto:jdoeson@acme.com">jdoeson@acme.com</a> </p>
```

```
</body>  
</html>
```

Save your page as **mytable2.html** and view it in your browser. To see how your page should look visit this web page: **<http://profdevtrain.austincc.edu/html/mytable2.html>**

I have indented some of the HTML code in the above example. Indenting the code can make your HTML document easier to read.

## Create Your Own Page

It's time to create your own page. Use your text editor to create a page which contains the following:

- the required HTML page codes
- link to another web page
- an email link
- a picture/graphic
- a list of information

Save the file as **xyhtml\_basics.html** where xy is your initials. Email the file to [jmorales@austincc.edu](mailto:jmorales@austincc.edu).