

HTML PROGRAMMERS GUIDE LESSON 1

File:	HtmlGuideL1.pdf
Date Started:	Dec 14, 1999
Last Update:	March 15, 2003
ISBN:	0-9730824-0-2
Version:	1.0

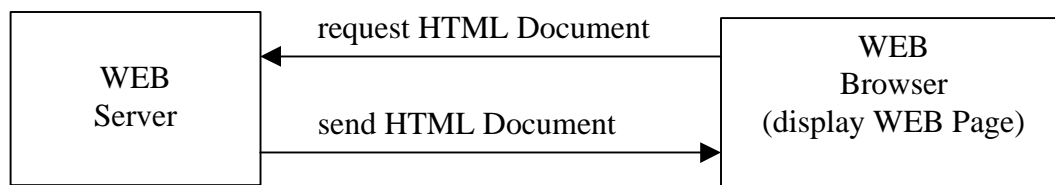
LESSON 1 HTML PROGRAMMING FUNDAMENTALS

Pre-requisites

You need to know how to use Microsoft windows and a WEB browser like Netscape Navigator or Microsoft Internet Explorer before you start these lessons. You need to know how to use a text editor like notepad or MSDOS Edit. You should already know how to surf the Net! We use **blue** for html code, **purple italics** for code definitions **bold** for new concept words underline for repeat concepts.

WEB Page

When you connect to a WEB site a WEB page is displayed on your WEB browser. There may be text, pictures and even sounds. How does the WEB browser know how to display this WEB page? The answer is this, the WEB pages are written in the **HTML** programming language and the WEB browser interprets the HTML. That's right the HTML tells the WEB browser where to display the text and pictures. When you connect to the Internet a Web Server sends the HTML Web Page document to your WEB Browser when your Web Browser requests the document.



HTML

HTML stands for **H**yper**T**ext **M**arkup **L**anguage. HTML relies on **tags** and **text**. **Tags** are HTML keywords to tell the WEB browser how to display the WEB page like:

a

Text tells the WEB browser what words you want displayed, on your WEB page. The text is also called **content**. Here is an example of text:

click here to learn more

Tags are enclosed in left and right **angle brackets** like:

<a>

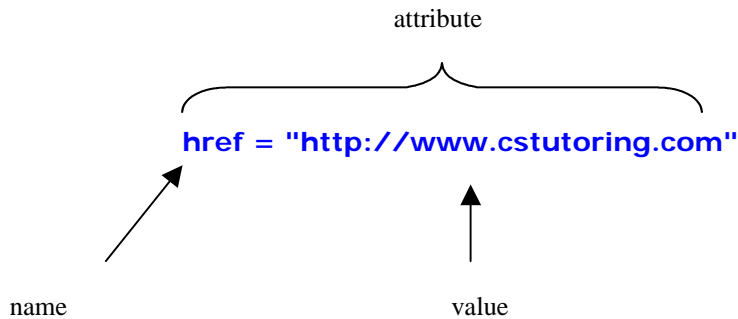
Tags come in **pairs**. There is a **start tag** `<a>` and a `` **end tag**. The **end Tag** is denoted by a `/`. A Text message is embedded between the start and end tag pairs as shown as follows:



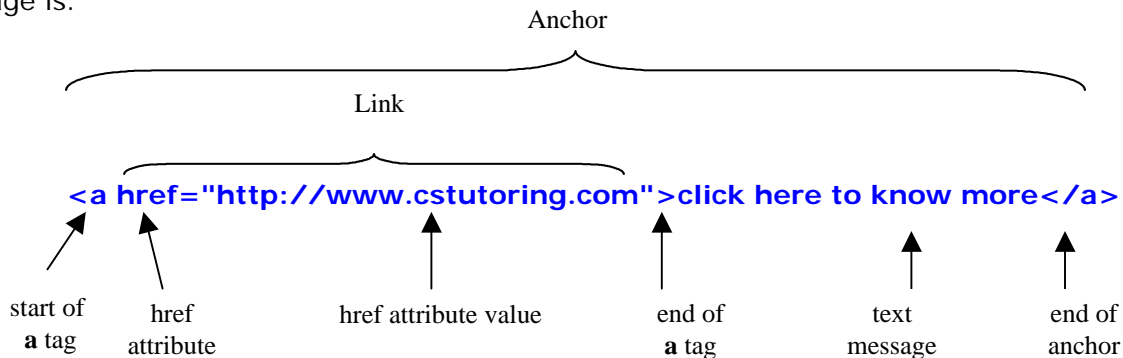
The tags also have **attributes**. Attributes supply extra information for the tag. An attribute has a name and a value. An example of an attribute name for the `<a>` tag is **href**, which stands for **hypertext reference**. A **href** is used to specify the location of a web **resource** like another web page or a file to download. Each attribute may have a **value**. An example of a value for the **href** is the actual web site address itself like:

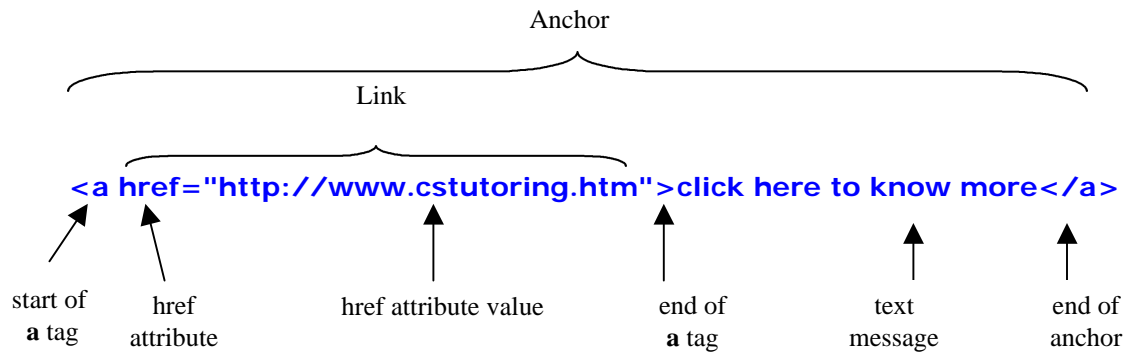
`"http://www.cstutoring.com"`

Values are usually enclosed in double quotes. A WEB page address is known as an **URL**, which stands for **U**niform **R**esource **L**ocator. The attribute name **href** has a URL value representing a web page address.

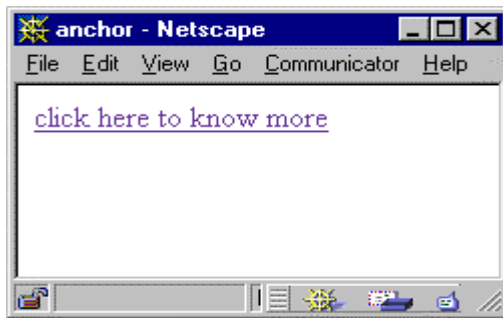


The `<a>` `` tag pairs are known as a **anchor**. An `<a>` tag may contain a **link** that lets the user click on a text message and be **re-directed** to another WEB site page. The link is specified using the **href** attribute and **URL** value. An example of an **anchor** tag that contains a link to a web page is:



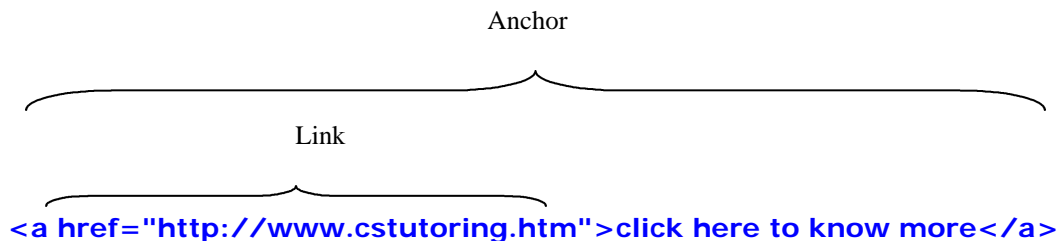


The **<a>** tag stands for **start anchor** and the **** stands for the **end anchor**. **href** is an attribute of the **<a>** tag and "**http://www.cstutoring.htm**" is a **value** of the **href** attribute. The **href** attribute specifies the location of another web page. **Text** requesting the user to "**click here**" is inserted between the **<a>** and **** tags. On the web browser the text usually appears a different color and underlined. The text on a WEB page that lets you go to another WEB page by clicking on it is known as a **hyperlink**.



a hyperlink example

I always get confused what is an anchor and what is a link. The only way to stay unconfused is to accept fact that the **anchor** contain the **link** !

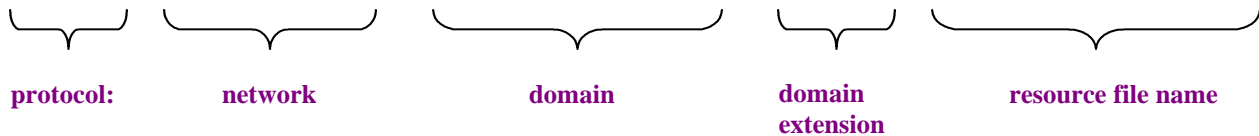


Hyperlinks may send you to another WEB page or display documents in another **area** of your current WEB page. **Anchor** tags with **links** help you navigate inside a WEB page. Instead of having to scroll up and down the web page where you want to go, you just click on a link and presto you automatically go there. WEB sites are made up of WEB pages each linking to each other.

URL'S

All WEB sites have an address known as a URL. A famous WEB address is The Computer Science Tutoring Center at WEB address: <http://www.cstutoring.com>. **URL** stands for **U**niform **R**esource **L**ocator which is simply a mechanism to locate web documents, in web servers. A URL has three major sections: **protocol**, **server location** and **resource file name**.

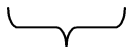
<http://www.cstutoring.com/index.html>



The slashes are the separators between the protocol and networks and domain and resource file name. They work like the slashes in your computer to separate directories from file names.

The **protocol** indicates what kind of service will handle the file.

[http:](http://)



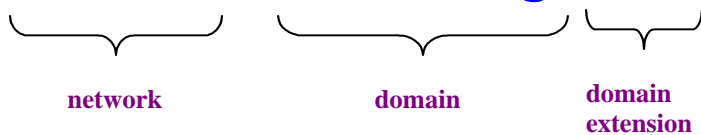
protocol:

There are many protocols available "http:" is the one used to read html documents. The following table lists all the other ones.

protocols	description
ftp:	used to transfer files
mailto:	used to send e-mails
file:	reading files
http:	use send pages to web browsers
https:	used to transfer data securely

The Web Server gets a **location** that consists of a **network** and **domain name**. The network is internal to the web server where it locates your domain name file space on its many servers. The most common network is "www". The domain name consists of a user name and an extension like **com**, **org**, **net** etc.

[//www.cstutoring.com](http://www.cstutoring.com)



Lastly the **resource file name**.

`/index.html`

resource file name

The file name may specify an html document, an image or sound file. If no resource file name is specified then a default resource file is sent. Some default resource file names may be index.html, index.htm etc. File names may also contain additional path names indicating sub directories, like: /images/cs.gif. Again the slash/ is used to separate the directory name from the file name.

MIME types

The type of file you are sending must be identified. The identification is known as a **MIME** type. MIME stands for **Multipurpose Internet Mail Extension**. The MIME type tells your WEB browser how to identify each document. More than 1 document type can be sent and received at a time, each identified by the MIME type. The file extension is used to identify the file type. File types are associated with the MIME type. The server will also send a message to the web browser to identify the type of file to be sent. The most common files sent are ones that contain text and html code. These files have the extension "txt", "htm" or "html". The following table lists some common MIME types with their file type extensions.

Format	Extension	Name	use
text/html	.txt	text file	display text
	.htm	html document	display text and images
	.html	html document	display text and images

Format	Extension	Name	use
Sound	.ra	Real Audio	RealAudio WEB server and Real Audio Player
	.sbi	SoundBlaster Instrument	Sound Blaster Cards
	.snd, .au	8 khz mulaw	voice grade sound format
	.wav	Microsoft Waveform	play sounds, music

Format	Extension	Name	use
Graphics	.gif	Graphics Interchange Format	Compressed graphics
	.jpg, .jpeg	Joint Photographics Experts Group	Highly compressed graphics format
	.pdf	Portable Document Format	Adobes format for documents
	.ps	PostScript	Adobes description language

Format	Extension	Name	use
Video Motion	.avi	Audio Video Interleaved	Microsoft Video for Windows standard format
	.dvi	Digital Video Interactive	Motion –video format
	.fli	Flick	Autodesk Animator motion-video format
	.mov	QuickTime	Apples motion-video and audio format
	.dcr	Director	

plugins

The non-text data of a WEB page shows up as attachments identified by the **MIME** type in the HTML file. The browser may display the MIME types automatically like GIF or JPEG graphic images. For specialized MIME types, the browser will call a **plugin**. The plugin knows how to display the MIME type. For example when you view **pdf** files attached to an E-mail or part of a WEB page then the WEB browser will automatically call the Adobe Acrobat plugin to view the pdf file. If you do not have the plugin installed, then your WEB browser may ask you to download it and install it automatically for you.

E-Mail

E-mail is electronic mail that lets you send messages over the Internet. Each person sending or receiving gets an E-mail address. The address starts with the persons name known as an E-mail box then a @ symbol and a URL. The URL represents where the message is going to and is usually the same URL as a WEB site. One URL can represent many E-Mail boxes. An E-mail box is where E-Mail is received and sent to the person's name preceding the @ symbol. A famous E-Mail address is:

courses@cstutoring.com

The protocol for e-mail is **mailto:** All kinds of documents are sent by E-mail. The type of document you are sending or receiving must be identified using a MIME type. We will send some emails shortly.

WEB PAGE DESIGN

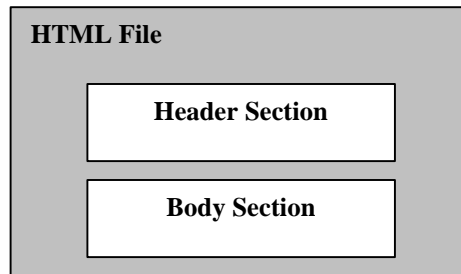
Now that we know a little bit about HTML and WEB pages it's now time to learn how to make a WEB page! As we told you earlier, a WEB page is controlled by **tags** that are also called **markups**. Tags are special HTML key words that tell the WEB browser how to display images and text. A tag starts with a **left angle** bracket < the **HTML keyword** or tag and ends with a right **angle bracket** >. The purple (top) is the definition where the blue (bottom) is the actual code

<keyword>

<HTML>

Case does not matter but tags are usually have a mix with upper and lower case. Important HTML keyword tags are usually upper case. Text is embedded between tags and is known as **content**. Content is just **text** displayed to the user. It is very important, how you combine the tags with the content text. This combination is important and is called **layout**. Layout refers how your WEB page is arranged. A good layout will give an appealing presentation to your viewer. Your layout should be impressive and informative. In WEB page design, the first impression is very important. You must be creative in combining images, text and colors in your WEB page. Before you start designing your WEB page you should sketch out what your web page would look like.

An html document will have two sections, A **header** section and a **body** section. The header section is just used to supply information about your web page to the web browser. the header section contents are not displayed in the web browser. the body section contains text and images to be displayed.



Using HTML Tags

We will now write our first HTML program. Open a directory on your computer call **HTMLCourse**. (put no space between HTML and Course). Make a sub directory called **Lesson1**. Inside this subdirectory use notepad or MSDOS edit and open a file called **L1p1.htm**. The first tag in a WEB page is the **<HTML>** tag. The **<HTML>** tag identifies that this file is a HTML document. Put this tag at the top of your L1p1.htm file. The next tag is the **<HEAD>** tag. This tag identifies the start of the **header** section of a WEB page. Put this tag next in your L1p1.htm file. The header section may contains a title, identification and search engine information. The header ends with the **</HEAD>** tag. The header section text is not displayed in the web browser, the header section is used to provide additional information but not to be displayed. Start tags and end tags are quite common in HTML and are known as **tag pairs**. Tag pairs are used to enclose a **section**. A html file will have many sections identified by tag pairs. Inside the header section you will find a **<TITLE>** start tag and a **</TITLE>** end tag. Every WEB page needs a title. The title text is listed between the title tag pairs.

<TITLE> My first WEB Page </TITLE>

Notice the title is just free text and is not enclosed with any quotes. The title is enclosed by the **<TITLE>** tag pairs. The title does not appear on the WEB page but rather on the title bar of the WEB browser. Next put the **<TITLE>** tag in your html document. The header section ends with the **</HEAD>** tag. Put this tag next in your L1p1.htm file. The next section of our HTML program is the **body** section identified with the start **<BODY>** tag and ends with the end **</BODY>** tag. Put this tag next in your L1p1.htm file. Inside the **<BODY>** tags are all the things you want the web browser to display. Usually the first sentence you display in your body will be a **heading**. Every WEB page needs a **heading**. The heading will be the first message displayed on your WEB page. Each heading can have levels **H1** to **H6**. Each level refers to a text size. Level H1 is the largest size and level H6 the smallest size. The headers are displayed in bold text. We will use heading level H3.

<h3> Welcome to Ed's WEB page </h3>

Put this heading in your L1p1.htm file. Inside the rest of the **<body>** section below the heading we can write some plain text messages.

Thank you for landing on my WEB site.

Put this text in your L1p1.htm file. We close the body with the end body tag `</BODY>` and close the HTML file with the `</HTML>` end tag. Put this tag in your L1p1.htm file. Your html program file should now look something like this:

```

<HTML>
<HEAD>
<TITLE>My first WEB page</title>
</HEAD>
<BODY>
<h3>Ed's WEB page</h3>
Thank you for landing on my WEB page
</BODY>
</HTML>

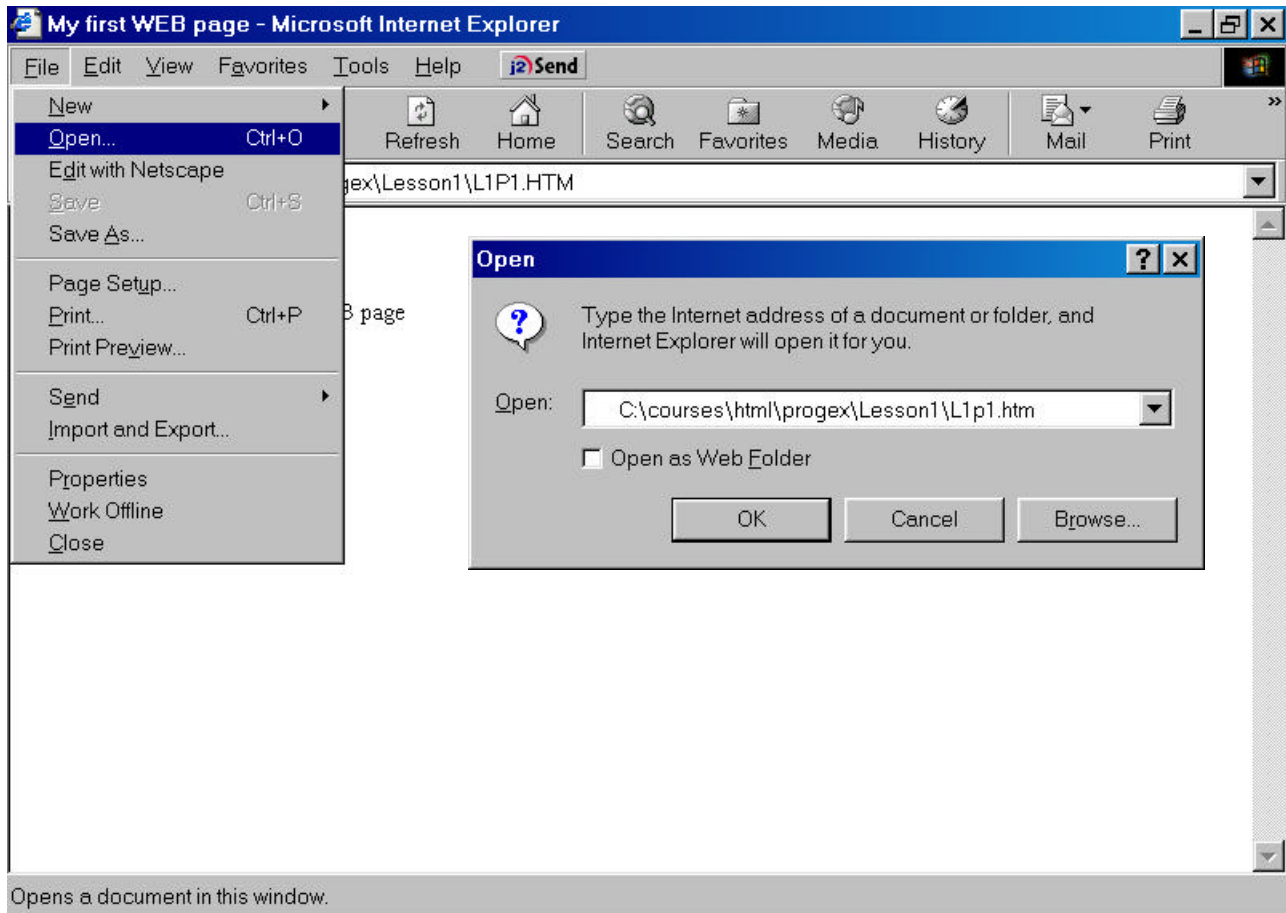
```

} Header Section

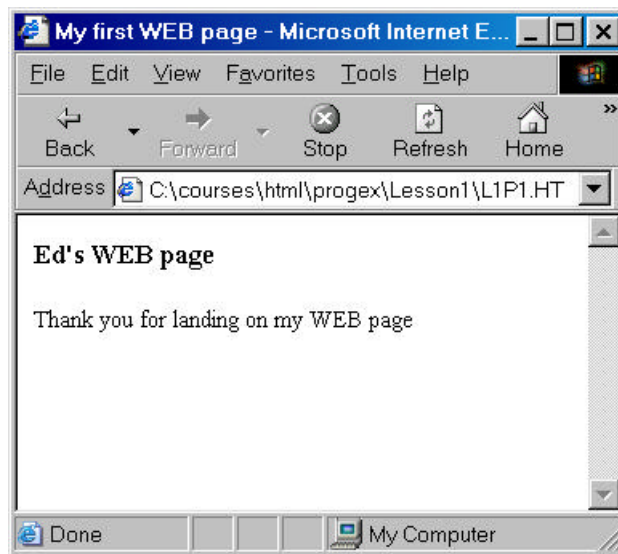
} Body Section

} html document

Our HTML file has two sections a **Header** section and a **Body** section. You should now run the html program just go to Internet explores or Navigator and select file open and the browse to your file name L1p1.htm.

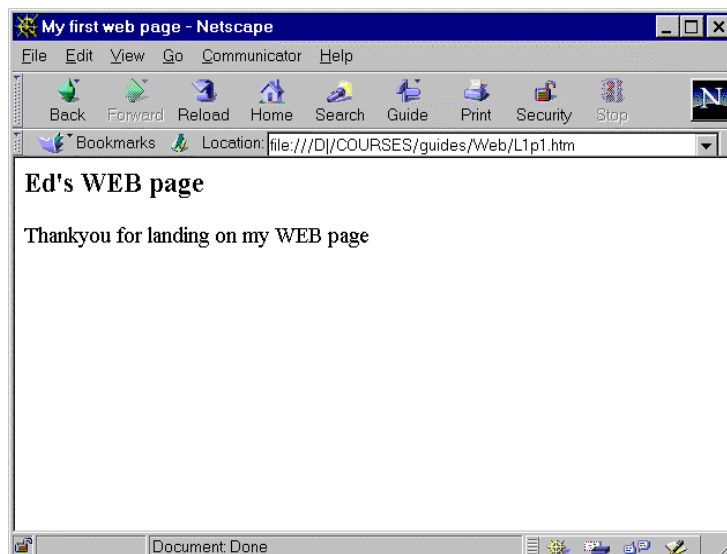


You should get something like this:



LESSON 1 EXERCISE 1

If you have not done so already, type the above HTML program in an editor like notepad or msdos edit. name your file L1p1.htm and run. You should not use a word processor, because the word processor may insert extra control characters into the file and mess things up. Substitute your own name for Ed and maybe your own message. Save your file as L1ex1.htm or L1ex1.html (we are using the .htm extension). You should make a separate directory for your entire WEB programming courses lessons and place your html files there. After typing in your file, open up your WEB browser and run the file. To run the file go to file menu of your WEB browser, select open page and go to the directory where you keep your WEB programming course files. You should get something like this:




HTML LANGUAGE DEFINITIONS AND EXAMPLES

A Language definition is like a formula that defines all the components needed to write the language. We use *purple italics* color to **define** the HTML language. The following is a language definition of a tag. The tag gets an attribute to supply additional information for the tag.

<tag attribute>

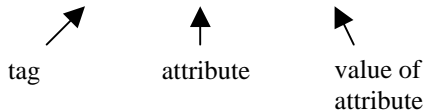
We use **blue** for an actual HTML code **example**. The following is a **table** tag with a border attribute. A table displays information in a grid where the border specifies the table will have a border.

<table border>


You will notice some tag attributes get a value:

<tag attribute="value">

The value is enclosed in double quotes. The value may be a URL, image file, an identification name, a number, text or a color. The align attribute and value "center" says to place the table in the center of the web page.

<table align="center">


Putting an image in a WEB page

It's now time to put an image in your WEB page to add some excitement. You need an image before you can display one. You can download an image from an existing WEB page by using "Save Image As" in your WEB browser, or you may have a few images of your own kicking around. The **image tag ** is used to put an image on your WEB page. Images are usually in the JPEG or GIF format. The JPEG format is used for large color pictures whereas as the GIF format is used for smaller pictures. The JPEG format has much better color resolution. To display an image on your WEB page you use the **** tag the **src** attribute and the image file name as it value.

**

The **** tag has the **src** attribute that has the value **"airplane.jpg"**. The value indicates the GIF or JPEG image file that will be used to display the image on the web page. In this case the image will be a picture of an airplane. Fortunately not a picture of Ed! To display the image it must be in the same file directory where your HTML file is:

or you must specify the file directory path with the image file name.

```
<IMG SRC="./images/airplane.jpg">
```

The **dot** means current directory, the / means sub directory. If the path is another WEB page address then you must include the complete URL in your path.

```

```

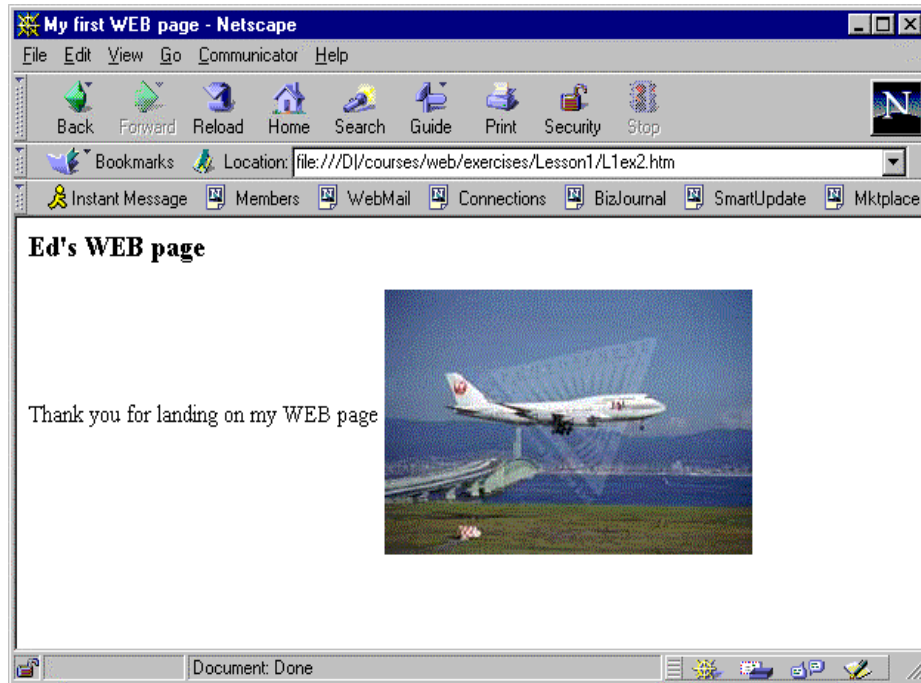
Image tag attributes

The `` tag has attributes to positioning the location of an image where it will appear on the WEB page or to specify the size of an image etc. When changing the size of an image just use the height or width attributes but not both, as to keep the same **aspect ratio** of the image (prevent stretching of an image). The aspect ratio determines the horizontal and vertical appearance of the image. Here are the attributes for the `` tags with description and examples.

attribute	description	
src="url"	location of image file to display	
	<code></code>	
alt="text"	A text message to display if the image file cannot be found Which will probably be the graphics file name.	
	<code></code>	
align="text"	Allows you how to place the graphics image	
	LEFT	(default) image on left side of page
	CENTER	image in center of page
	RIGHT	image on right side of page
	JUSTIFY	image location set by surrounding text
	<code></code>	
vspace=number	the vertical space between the image and the text above or below it	
	<code></code>	
hspace=number	the horizontal space between the image and the text to the left or right	
	<code></code>	
width=number	the width in pixels or percentage of the image, expand or shrink image	
width=percent%	<code></code>	
height=number	the height in pixels or percentage of the image, expand or shrink image	
height=percent%	<code></code>	
border=number	draws a border of the specified pixels around the image, 0=no border	
	<code></code>	
lowsrc="url"	load a lower resolution image first before the higher resolution images is loaded (used for fast page loading)	
	<code></code>	

LESSON 1 EXERCISE 2

Add the image file to the your program L1ex1.htm and save as L1ex2.htm then run it. Use the **alt**, **align** and other attributes. Experiment with LEFT, CENTER, RIGHT and JUSTIFY alignment values. Try resizing your image using **width** or height **attributes**. Call your file L1ex2.htm. You should get something like *this:



ANCHORS

Anchors contain **links** that allow you to go to other WEB pages or different parts of WEB pages in your current WEB page. There are two types of links:

- (1) **relative links** refer to web pages inside your WEB site
- (2) **absolute links** refer to web pages in other WEB sites.

Each WEB site will have a collection of WEB pages all interconnected by links. The following is an anchor with a **relative link** to another web page in this web site.

[click here to know more](more.htm)

The **<a>** tag has the attribute **href** whose value is the **URL** of the web page you want to go to. A relative link is just a file name inside your web page. The URL is already specified and is the one you are presently at. In the above example we have a **relative address link** to another HTML file in the current WEB site

In the next example the anchor gets an **absolute address link** to a WEB page in another WEB site. The absolute address link is a complete URL.

```
<a href="http://www.cscourses.htm">click here to know more</a>
```

We can now add the absolute link to cstutoring to Ed's WEB page.

```
<HTML>
<HEAD>
<TITLE>My first WEB page</title>
</HEAD>
<BODY>
<h3>Ed's WEB page</h3>
Thank you for landing on my WEB page
<a href="http://www.cstutoring.com">click here to know more</a>

</BODY>
</HTML>
```

It is sometimes confusing what is an anchor and what is a link. The only solution to this confusion is to accept the fact that anchors contains links.

LESSON 1 EXERCISE 3

Add an absolute link using an anchor tag to your favorite WEB page or call another web page using a relative link. You may keep your image. Call your file L1ex3.htm. You should get something like this:

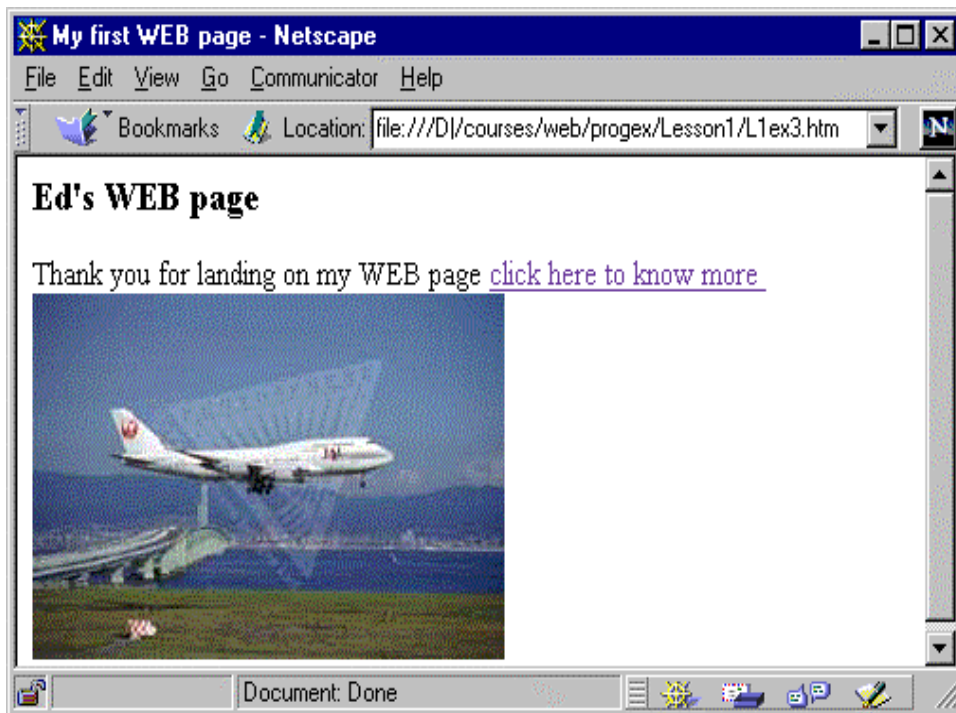


Image anchors

Instead of telling the user to [click here](#) on a text message you may want the user to click on the graphic image itself. It is easy to combine an anchor with a graphic image. The `<src>` tag is now **embedded** or **nested** in the `<a>` tag. You will see that HTML is a powerful language.

```
<a href="http://www.cstiutoring.htm"></a>
```

start of a tag href attribute href attribute value end of a tag start of img tag src attribute image file name end of a tag anchor

the `` tag is used to display an image, the `src` attribute tells which image to display. The `<a>` tag encloses the link. The `href` attribute tells where to go when someone clicks on the image. An image that is a link will automatically have a default border around it.

LESSON 1 EXERCISE 4

Add to your Web page from last exercise, an image link so that the viewer can click on an image and go to your favorite WEB page.

E-MAIL ANCHORS

You may want to be able to send e-mails from your web page. A `mailto:` link containing the `mailto:` protocol allows users to send an E-mail to the WEB page owner. Before you can use a `mailto:` link, the WEB browser usually has to be set up to send and receive E-mail. In your web browser's preference you need to set up the outgoing mail server name.

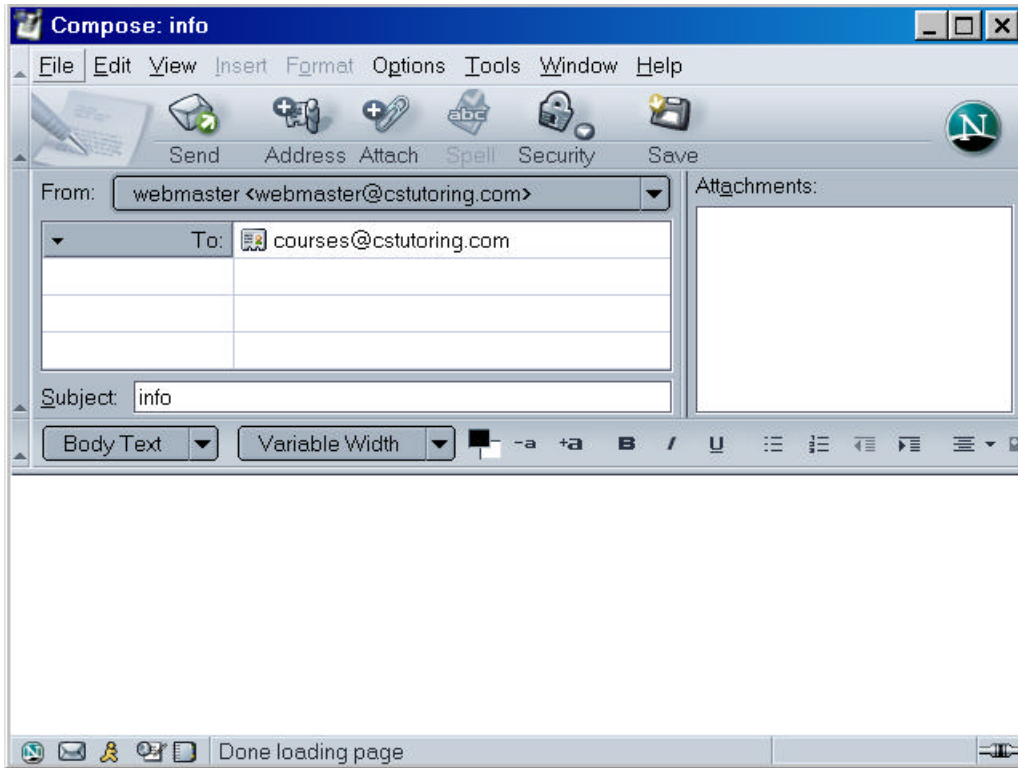
```
<a href="mailto:courses@cstutoring.com">Send me more information</a>
```

mailto: protocol email address

You may want to specify the subject of this e0mail by specifying a query string. A query string starts with a ? and the has a name/value pair. the name is subject and the value is name.

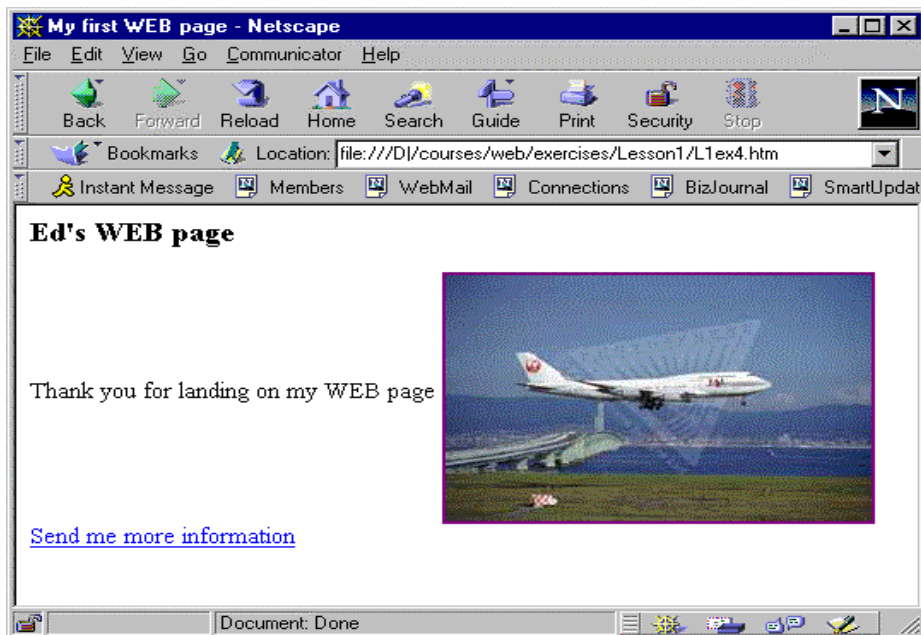
```
<a href="mailto:courses@cstutoring.com?subject=info">Send me more information</a>
```

mailto: protocol email address



LESSON 1 EXERCISE 5

Allow a user to send you an E-mail by adding an E-mail anchor on your web page. Call your file L1Ex5.htm. You should get something like this:



SOME USEFUL TAGS

Line Breaks

The `
` tag let you start a new line. The `
` tag applies to individual lines.

```
Hello how are you today <br>
```

Paragraphs

The `<p>` tag starts a new paragraph and the `</p>` tag ends the paragraph. An empty line is placed at the beginning of a paragraph and at the end. The `<p>` tag applies to one or many lines.

```
<p>
It is very nice you are learning HTML
I hope you are enjoying these Lessons.
</p>
```

A paragraph tag can enclose any other tag. For example to add space around an anchor:

```
<p>
<a href="http://www.cstutoring.com">click here to know more</a>
</p>
```

Paragraphs and Breaks

You find it is necessary to combine `
` tags with `<p>` tag. You will find that you will need a `
` tag every time to start a new line.

```
<p>
It is very nice you are learning HTML <br>
I hope you are enjoying these Lessons. <br>
</p>
```

Address tags

The address tag is used to display information about the WEB page, the person who wrote it, their copyright notice etc. An address tag is like the **bottom** or **footer** of the WEB page and usually displayed in italics.

```
<address>
This WEB page Copyright 1999 by "cstutoring"
</address>
```

Horizontal lines

Horizontal lines are used to separate sections of your WEB page using the `<hr>` tag

Comments

Comments are used to explain parts of a html program that you type in. Comments are not displayed on the web browser. Comments start with a `<! --` left angle bracket, exclamation mark followed by 2 hyphens and ends with a `-->` two hyphens ending with a right angle bracket.

`<! -- That's right this is a comment -->`

You should add comments your HTML files not only for you when you read your HTML file several months later and want to figure out what you have written but also for courtesy to your readers.

WEB page layout format.

You have now been introduced to all the components of a simple WEB page. The following outline shows the format for an html program. An HTML program starts with the `<HTML>` tag and then the `<HEAD>` tag section. The `<TITLE>` tags goes inside the `<HEAD>` tags. The `<HEAD>` tag sections used to give information to your web browser and not to display things. Following the `<head>` tag section is the `<BODY>` tag section. The `<BODY>` section is used to display things on the we browser screen. The tags enclosed inside the `<BODY>` tags specify what is to be displayed and where it is to be located. Tags inside tags are known as **nested tags**.



Displaying and playing sound files

You can use the anchor tag to display graphic files, play sound etc. The following html code will play animal sounds if the user has a sound card. The animals.wav file is on the server..

```
<a href="animals.wav"> click here for animal sounds </a>
```

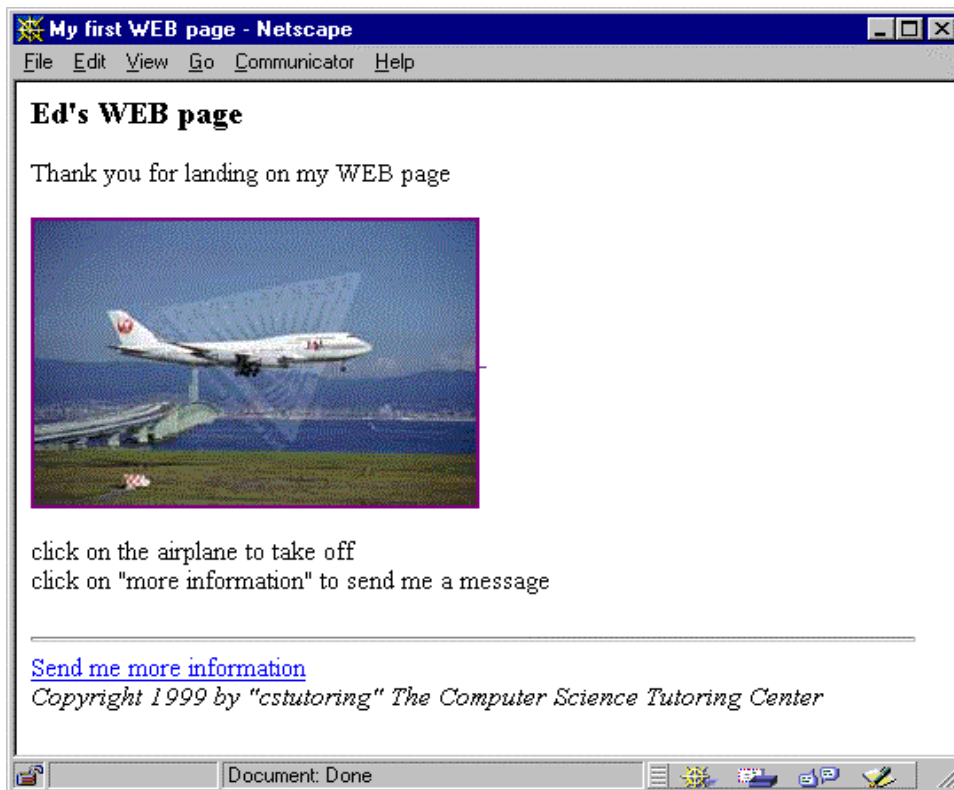
You can use the embed tag to start the sound file when the web page loads.

```
<embed src="animals.wav" width = 2 height = 2>
```

The `<embed>` tag should be the last tag in your web page html file.

LESSON 1 EXERCISE 6

Add to your WEB page extra text using line breaks `
` and paragraph `<p>` tags. Also add a horizontal line using the `<hr>` tag and a footer section using a `<address>` tag. And maybe play some music when they click on a link or when the web page is first shown (loads). You may want to put your mailto: anchor in the address section. You should get something like this:



IMPORTANT

You should use all the material in all the lessons to do the questions and exercises. If you do not know how to do something or have to use additional books or references to do the questions or exercises, please let us know immediately. We want to have all the required information in our lessons. By letting us know we can add the required information to the lessons. The lessons are updated on daily bases. We call our lessons the "living lessons". Please let us keep our lessons alive.

E-Mail all typos, unclear test, and additional information required to:

courses@cstutoring.com

E-Mail all attached files of your completed exercises to:

students@cstutoring.com

Order your next lesson from:

courses@cstutoring.com/html.htm

This lesson is copyright (C) 1998-2003 by The Computer Science Tutoring Center "cstutoring"

This document is not to be copied or reproduced in any form. For use of student only.