

# A Fat Free Guide to The Internet and HTML



*The contents of this manual are 100% fat free!*

*An Introduction to the Internet and HTML*  
**A Written Word, Inc. Production**

# A Fat-Free Guide to The Internet and HTML



The Written Word, Inc.  
5435 35th Ave. SW  
Seattle, WA 98126  
206.938.0990  
wrtnwrld@nwlk.com  
www.writtenword.com

## **PROPRIETARY STATEMENT**

This document contains proprietary information and data which is the exclusive property of The Written Word, Inc., Seattle, Washington. No part of this document may be reproduced, transmitted, stored in a retrievable system, translated into any language, or otherwise used in any form or by any means, electronic or mechanical, for any purpose, without the prior written consent of The Written Word, Inc..

© Copyright 1996 The Written Word, Inc., All rights reserved.

All brand and product names are trademarks or registered trademarks of their respective companies.

Published by The Written Word, Inc.

5435 34th Ave. SW, Seattle, WA 98126

Phone: (206) 938-0990

Fax: (206) 938-1159

Email: wrtnwrđ@nwlínk.com

Web: [www.writtenword.com](http://www.writtenword.com)

Printed in USA

# Preface

**In this chapter**

About the Fat-Free Guide

Using the Fat-Free Guide

# About the Fat-Free Guide

## What's In It

The first part of this book is a quick and simple introduction to the Internet and the World Wide Web. The second section is a short guide to HTML 3.0, the mark-up language used to create webpages. Illustrations and examples have been kept to a minimum, in order to present the information as directly as possible. *The Fat Free Guide to the Internet and HTML* is designed to provide you with starting point as you explore the World Wide Web and HTML.

It is expected that users of this book will soon outgrow it; as you go on to master the Internet and web page design you will still find this guide a useful refresher, or you can pass it along to introduce others to the World Wide Web.

## How to Get Updates

This book, like the HTML standard, is a work in progress. You can retrieve updates from our web site:

<http://www.writtenword.com>

Check our site periodically for updates to *The Fat Free Guide*, and for other useful information.

If you filled out the registration form when you downloaded the Guide, then we will email you when a new version is available.

## Our Price

The book costs nothing. Really. It's free. Enjoy it, use it, tell your friends about it. We do ask, however, that if you reprint it for distribution in a classroom or any other environment, or wish to redistribute it electronically, that you ask us first.

What's in it for us? Educated consumers are the best consumers. Philanthropic impulses aside, we know that people who like and use the Web are good for business.

## More Information and Resources

The Written Word, Inc. does publish a more elaborate version of both the HTML and Internet sections of this book. However, those versions will cost you money... Contact us for more information...

We also do custom on-site training.

# Using the Fat-Free Guide

## Online or Hardcopy?

Hardcopy's great — that is why we made this available in Acrobat format. You can print this document on any printer, and get something that looks almost exactly like the original print run.

However, The Fat-Free Guide was written in FrameMaker 5.1.1, and then converted to Adobe Acrobat format. We have built in some handy hypertext features that will let you jump around the online version of this document.

- For instance, if you come to a term in the document that looks **like this**, you can click on it to jump directly to the glossary. When you are done reading the term, click on the 'go back' button in Acrobat, and you are back to where you started.
- All cross references that look **like this** let you jump around the document.
- In addition, the Table of Contents is hypertext. Click on the title of the section you want to read, and you will 'jump' to it.
- Also, if you see any text that is **blue underlined**, you can click on that text to jump to a web site or relevant example stored on the WWW.



We have also put many of the samples in a single area of our web site. [Click here to see.](#)

We have also indexed the document using Adobe Acrobat Catalog. If you are using the Acrobat Reader with Acrobat Search, you can perform a full-text search of the document.

## Tools of the Trade

To put this book to work you will need a few simple tools:

- A desktop computer
- A web browser
- A simple text editor, such as Write, Notepad, vi, or any other word processing program.

You may also want to consider the following tools:

- An image editing and conversion program
- A What-You-See-Is-What-You-Get, or WYSIWIG, HTML editor. While these tools are handy, we recommend that you first learn to work with the HTML coding itself, and then move on to tools that streamline your work.



# Table of Contents

## Chapter 1 The Internet

The Internet .....	2
What is the Internet? .....	2
How do I Connect to the Internet? .....	2
How do I Get Information from the Internet? .....	3
The World Wide Web .....	4
What is the WWW? .....	4
How do I connect to the WWW? .....	4
How do I find something on the WWW? .....	4
If You Do Not Have the URL .....	4
The URL Naming Convention .....	5
Absolute and Relative URLs .....	5
How does it work? .....	6
The Web Site .....	7
What is a Web Site? .....	7
Hypertext Markup Language .....	8
What is Hypertext Markup Language? .....	8

## Chapter 2 Basic HTML Coding

Introduction to HTML .....	10
HTML Tags and Elements .....	10
HTML Attributes .....	10
Basic HTML Structure .....	11
Your First HTML Document .....	12
Types of Tags .....	13
Structure Tags .....	13
Descriptive Tags .....	14
Formatting Tags .....	15
Lists .....	17
The <PRE> Tag .....	19

## Chapter 3 Hypertext

Using Hypertext .....	22
Links .....	22
Anchor - The Linking Tag .....	22
Types of Links .....	24
Summary .....	24

## **Chapter 4      Tables**

Introduction to Tables .....	26
Basic Table Syntax .....	26
Spanning .....	26
Additional Table Tags and Attributes .....	28
Summary .....	28

## **Chapter 5      Images**

Introduction to Images .....	30
The Image .....	30
The Image Tag .....	30
Summary and a Word of Caution .....	31

## **Chapter 6      Embedding**

Introduction to Embedding .....	34
Sound .....	34
Video .....	34
Other Files .....	34
Summary .....	35

## **Chapter 7      Forms**

Forms: Making Web Pages Interactive .....	38
A Sample Form .....	38
What Forms Do .....	39
How Forms Work .....	39
Creating a Form .....	41
Inserting a Form in Your Web Page .....	41
The <FORM> Tag .....	41
Field Tags .....	41
Form and Field Tags List .....	42
Input Types List .....	44

## **Chapter 8      FTP**

Putting it on the Web .....	48
-----------------------------	----

## **Appendix A      Glossary**

# The Internet

**In this chapter**

[The Internet](#)

[The World Wide Web](#)

[The Web Site](#)

[Hypertext Markup Language](#)

NOTE: This chapter is an overview of information found in The Fat Free Guide to The Internet. Contact The Written Word to obtain the full publication.

# The Internet

## What is the Internet?

The **Internet** is a communications network linking computers worldwide. It forms the nervous system of the **World Wide Web (WWW)**

At the center of the Internet is a high speed “backbone” which carries data around the world. Telephone company land-lines, ISDN lines, and other high-speed data connections link literally millions of homes and businesses to this backbone itself.

However, the average computer user cannot link directly to the backbone. Instead, we connect to the Internet using a computer dedicated solely to maintaining an internet connection, or **Gateway**.



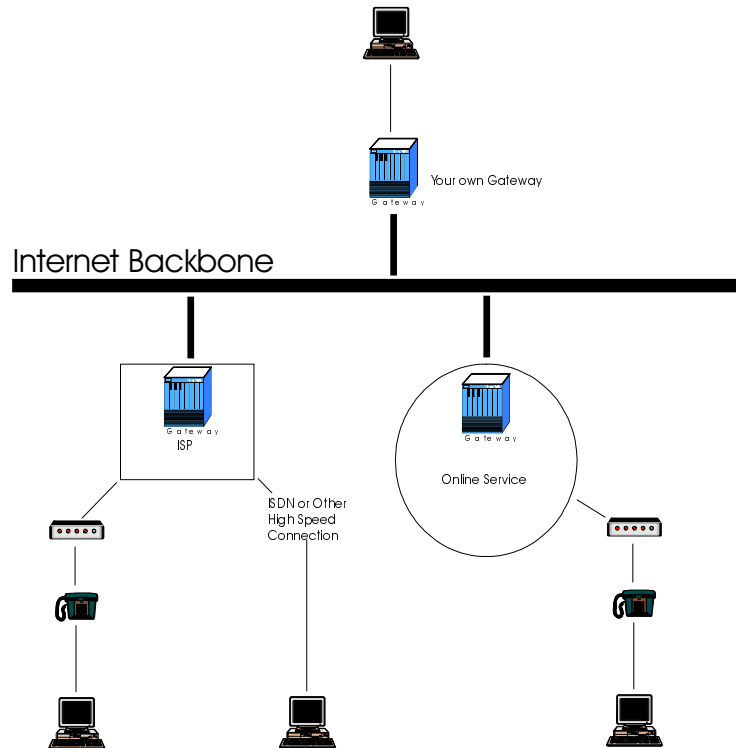
You may also hear gateway computers referred to as **Hosts**, or **servers**. Your computer is referred to as a **Client**.

## How do I Connect to the Internet?

There are a number of different ways to connect to a gateway:

- **Establish your own:** Set up your own computer and domain with the **Internic**. If you do not recognize any of the terms just used, this is not the recommend this option. However, there are many consulting firms which assist small businesses who wish to establish their own gateways — if you really have your heart set on being your own gateway, they can help you.
- **Connect through an on-line service:** Most of the large on-line services, such as CompuServe, America Online, and MSN have their own gateways, which you can use to connect to the Internet. This is a relatively simple, user-friendly way to access the Internet. However, the large volume of users on these services can slow access.
- **Connect through an Internet Service Provider:** An **Internet Service Provider (ISP)** provides you with direct dial-up access to their gateway. This allows you the closest thing to direct Internet access without owning your own gateway. This can offer you the most reliable Internet connection, but you must be careful that you select a quality ISP.

**Figure 1. The Internet**



There are many more details to the actual Internet connection, including routing, the TCP/IP protocol, and other networking issues. This training course does not cover these subjects. However, see the Information Sources section of this guide for books that do.

## How do I Get Information from the Internet?

There are several different types of services available to you on the Internet:

- **Communication Service:** These services include e-mail and Telnet. This is the simplest information access method: You connect to the Internet, and send and receive files directly from other computers.
- **Information Search Services:** These services allow you to search databases of information, plus learn of other internet services which may have the information you need. These services include Gopher, WAIS and Archie.
- **File Transfer Protocol (FTP):** This services allows you to transfer files from one location on the Internet to another.
- **The World Wide Web (WWW):** The World Wide Web is a graphical interface to information accessible on the Internet. You can access the WWW using any standard Web Browser.

This course will concentrate on the WWW, and on HTML, the language used to format documents for the WWW. However, we will touch on the other service types as they relate to the WWW.

# The World Wide Web

## What is the WWW?

The WWW is not actually a single entity. Rather, it is a *client-server* network that includes **web servers** which are designed to deliver files that are written in HyperText Markup Language, or HTML, to your computer, which is the *client*. These computers use the Internet to connect to each other.

## How do I connect to the WWW?

If you have a connection to the Internet, you can access the WWW using any web browser, such as Netscape or Internet Explorer. Web browsers interpret the HTML files stored on the web servers and display the result on your computer screen. All you need to know is the address of the HTML files you wish to view.

## How do I find something on the WWW?

To locate an HTML file on the WWW, you need to know its **Uniform Resource Locator (URL)**. The URL is the ‘human-friendly’ representation of a file’s address on the Internet. So, a website might have an IP address of 199.222.333 — this is not easy to remember. However, the URL for that address might be <http://www.mysite.com/index.html>.

To download an HTML file for which you know the URL, do the following:

1. Start your web browser.
2. Type the URL in the Location, Go To or Address field of the browser window, and press Enter.

## If You Do Not Have the URL

If you do not know the URL for the file you want, you can use any one of the many *search engines* on the WWW. These are the URLs for some of the more popular search engines:

- <http://www.yahoo.com> is the easiest search engine to use, and has a tremendous amount of content.
- <http://www.lycos.com> probably has the best keyword search capability on the WWW.
- <http://infoseek.com>



The name of the field may differ from browser to browser. However, it is always at the top of the web browser window.

## The URL Naming Convention

The URL concept is a naming system used to guarantee that all files stored on the Internet use a common set of tags to show their location. Think of URLs as ZIP codes for your files.

URLs contain location data for a particular file:

*service://host:port/path/file.ext*

- *service://* indicates what internet protocol you will use to access the document. Common services include the following:

Service	Description
file://	Download a file using FTP
ftp://	Download a file using FTP
<b>http://</b>	<b>Download an HTML file using HyperText Transfer Protocol</b>
gopher://	Download a file using Gopher
telnet://	Access a remote computer using telnet.

- *host* indicates the machine from which you are downloading the file.

Your host could be `www.writtenword.com`.

- *:port* is an optional item that tells the web browser how to access services that do not have default ports. Normally, you do not need to use this.
- *path* indicates where the file is located on the host.

So, if you are downloading a file located in the “pix” directory on `www.writtenword.com`, then you would use the URL `www.writtenword.com/pix`.

- *file.ext* is the name of the file you are downloading.

So, if you are downloading an HTML file called “page1.html”, located in the docs directory on `www.writtenword.com`, you would use the following URL:

`http://www.writtenword.com/docs/page1.html`

## Absolute and Relative URLs

There are two types of URLs:

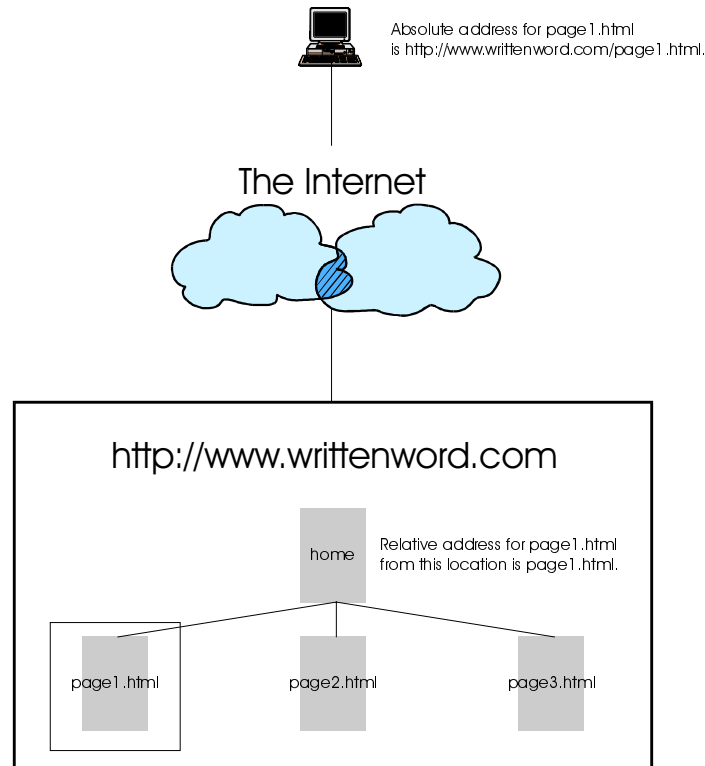
- **Absolute URLs** give you all of the information you need to find a file on the WWW. `http://www.writtenword.com/docs/page1.html` is an absolute URL.



Normally, you do not have to type in the path or filename when contacting a web site. That is because most web servers are configured to automatically send you to a file named `index.html` when you type in the host name.

- **Relative URLs** only provide the address of a file relative to your current location. So, if you were in the `pix` directory of `www.writtenword.com`, the URL for `page1.jpg` would be “`page1.html`”.

Now you are probably wondering when to use a relative URL and when to use an absolute URL. The simple answer is, when you are accessing files using the Internet, always use the absolute URL. However, the absolute/relative distinction becomes more complex when you are linking HTML files or graphics. The next chapters will discuss this in more detail.



**Figure 2. Absolute vs. Relative URLs**

## How does it work?

Exactly how your computer finds, retrieves and sends information to the WWW is outside the scope of this manual. For the purposes of this manual, however, one can break the process into five steps:

1. You enter the URL for a particular file into your web browser, or click on a **link** that connects to the file.
2. The browser sends a request to the WWW for that file.
3. Your request is routed to the correct web server.
4. The web server delivers the file to the web browser on your PC.
5. Your web browser interprets the HTML coding in the file and displays the result on your screen.



How your web browser actually finds the specific file requested in the millions of files on the WWW could be an entire book of its own, and is not covered here.

# The Web Site

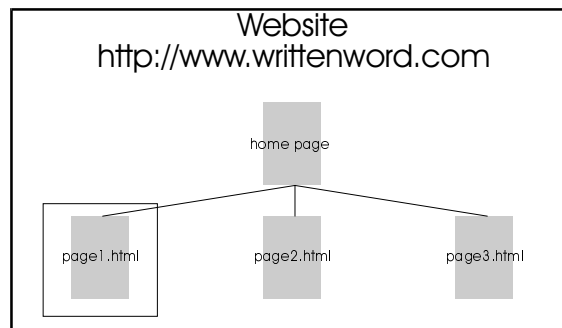
## What is a Web Site?

A web site is simply a collection of files, linked together and stored on a web server. These files are text files, coded in the HTML language.

Files are stored on a web server just as they are stored on a PC: They are saved to a directory or set of directories.

There are many different terms used to define different components of a web site. Sometimes the entire web site is referred to as a *web page*, or *home page*. To avoid adding confusion to this subject, we will use the following terms throughout this book:

- **Web Site:** The completed collection of files installed on a web server.
- **Page:** A single file or document in a web site.
- **Home Page:** The first page a web browser sees when it connects to a site.



**Figure 3. Typical web site, with a home page and three linked pages.**

These files are usually linked together with *hypertext links*, which allow the user to browse through the documents by clicking on words or graphics configured to act as links.

# Hypertext Markup Language

## What is Hypertext Markup Language?

The WWW connects a tremendous number and variety of computers: Macintoshes, PCs, UNIX terminals and other computers all share information on the Web. All of the users of these computers want to see graphically rich content, including photographs, formatted text, and tables. Somehow, all of these computers need to have a common language — a single piece of information should look nearly identical on all of the different platforms, even if it includes graphics and tables.



HTML looks **almost** the same from platform to platform, and from browser to browser. Not **exactly** the same.

**HyperText Markup Language (HTML)** is the language of the WWW. It is a common language that all web browsers are designed to interpret the same way. An HTML file displayed in Netscape on a Macintosh will almost match the look of an HTML file displayed in Internet Explorer on a PC.

HTML allows the user to display formatted text and graphics on a wide variety of platforms using a single file. It also allows the user to *link* different files together using **Hypertext** links. A hypertext link is an area of text or graphics that, when clicked by a mouse, takes you to another area of text or graphics.

The remainder of this book will discuss how you can write HTML, and how to set up a web site using HTML.

# Basic HTML Coding

**In this chapter:**

[Introduction to HTML](#)  
[Your First HTML Document](#)  
[Types of Tags](#)

# Introduction to HTML

## HTML Tags and Elements

At the core of HTML are **Tags** and **Elements**.

A tag is a set of text characters you use to format your document. For example, the `<h1>` tag will create a level one heading:

`<h1>Hi there</h1>` will generate **Hi there** when you view that phrase in a browser.

The text, and the beginning and ending tags, are known together as an “**Element**”. So, in the example above, `<h1>Hi there</h1>` is an element — in fact, it is an ‘h1 element’.

## HTML Attributes

Many HTML elements can also have **Attributes** assigned to them.

An attribute is an additional feature you can use to configure the element. For example, the `<h1>` tag can also have the attribute “align” assigned to it, so that `<h1 align="left">` will generate a centered level one heading.

Attributes are optional, depending on how you want to modify the tag — an `<h1>` tag will work just as well as an `<h1 align="center">` tag. It just will not be centered on the page.

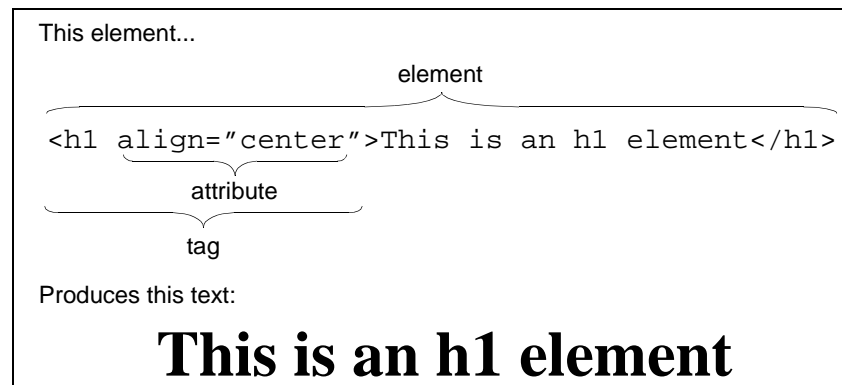


Figure 1. Anatomy of an HTML Element

### *Closing An Element*

You probably noticed that for every `<h1>` tag in the examples above, there was a corresponding `</h1>` tag.

Most HTML tags require the `</tag>` at the end of the element — that is how each web browser knows to look for a new element, and apply new formatting to that new element. So, if you use the `<h1>` tag to

create a level one heading, then you have to add a `</h1>` at the end of the heading. You do not have to include the attributes in the closing tag.

This is true of almost all HTML tags, with a few exceptions that we will discuss later.

## Basic HTML Structure

### *HTML is a hierarchical language*

HTML is a hierarchical language. That means you place elements in location relative to other elements. Their location relative to each other determines their meaning and how text is displayed.

### *Nesting HTML Elements*

You can place one HTML element inside the other. For example:

```
<h1 align="center"><i>This is an italicized  
level one heading</i></h1>
```

However, you cannot overlap HTML elements:

```
<h1 align="center"><i>This is a mistake</h1></  
i>
```

# Your First HTML Document

Now that you have seen the individual elements, you need to see it all together. Here is what you need to do to create and view your first HTML document:

1. Start your text editor or word processor.
2. Enter the text in figure 2. Or, if you do not like to type, copy the text and past it in. We won't tell.
3. Save the text in text-only format (also called ASCII or raw text on some UNIX and Windows-based computers) under the name *first.htm*.
4. Start your web browser.
5. Open the file **first.htm** using your browser's "Open file in browser" option.
6. *Voila!* Your first web page!



The .htm or .html file extension tells your computer that this file is readable by a WWW browser.

As you look at this sample HTML document, pay particular attention to the following:

- The structure of the elements
- Where text enclosed by the different elements is displayed in the web browser
- For now, do not worry about the actual tags and element names; in the next section, you will learn about basic HTML tags and how to use them.



To see this sample on the web, [click here](#).

```
<html>
<head>
<title>This is your sample HTML document</title>
</head>
<body>
<h1 align="center"><i>This is a centered, italicized heading.</i></h1>
Note how everything that is enclosed in the <b>body</b>
element of this page is displayed as a document in the web
browser. The text enclosed in the <b>title</b> element is
displayed in the title bar of the web browser, instead,
because the title element is enclosed within the <b>head</
b> element. . . .
</body>
</html>
```

**Figure 2. HTML for Sample Document**

Now that you have a basic understanding of how tags and elements interact, we are going to take a close look at some of the different types of tags available.

# Types of Tags



Formatting tags help to standardize the look of your documents across platforms. As a document's format affects what your viewers get out of it, you will find that the largest number of tags are devoted to formatting.

Within each element you will use different tags to create the effects you desire and insure your message is received. The different types of tags can be broken into four general categories:

- **Structure** - These tags define the document and its different sections.
- **Descriptive** - These tags give a title to your document or provide other information that can be used by a browser or server.
- **Formatting** - Bold face, carriage returns, indents and anything related to layout.
- **Special** - These would allow for sounds to be played or animations to be run by browsers that support such features. Some of these tags will be discussed in the advanced chapters.

## Structure Tags

Structure tags define and distinguish the different elements of an HTML document. There are four basic structure tags:

Tag	Name: Description
<code>&lt;HTML&gt;...&lt;/HTML&gt;</code>	HTML: The opening and closing tag of any HTML document. It allows the web browser to identify file contents as an HTML document.
<code>&lt;HEAD&gt;...&lt;/HEAD&gt;</code>	Header: The information you include within this tag will not appear in the document proper, but elsewhere — such as in the menubar, or as indexed text on a search engine. The <code>&lt;head&gt;</code> tag follows the opening <code>&lt;HTML&gt;</code> tag and closes before the opening <code>&lt;BODY&gt;</code> tag.
<code>&lt;BODY&gt;...&lt;/BODY&gt;</code>	Body: Encloses the main body of your document and, most likely, the majority of your work.
<b>&lt;BODY&gt; Attributes</b>	
ALINK	Specifies color of activated links.
BACKGROUND	Specifies an image to be used as a tiled background or wallpaper.
BGCOLOR	Specifies the background color of the document.

Tag	Name: Description
BGPROPERTIES=FIXED	“Fixes” the background image so it does not scroll.
LEFTMARGIN	Specifies the left margin for the document.
LINK	Specifies the color of links in the document.
TEXT	Specifies the color for text in the document.
TOPMARGIN	Specifies the top margin of the document.
VLINK	Specifies the color of links that have been followed.
<!--...-->	<p>Comment: This tag allows you to include messages, or “comments,” within your HTML document, without displaying those messages in the web browser. Your viewers will not see these messages unless they decide to view your HTML code.</p> <p>The comment tag is commonly used to:</p> <ul style="list-style-type: none"> <li>• Make notes about incomplete pages.</li> <li>• Earmark areas that need improvement, for yourself or for someone else.</li> <li>• Identify yourself or others, date the document, or insert other information about the document itself.</li> </ul>



Do **not** assume that viewers will not be interested in looking at the HTML behind your web page. Browsers make this an easy thing to do. . .

## Descriptive Tags

Descriptive tags allow you to include information about your web page, such as a title, or information for use by search engines when they index your page. Descriptive tags must be inside the <head> element:

Tag	Name: Description
<TITLE>...</TITLE>	Title: This tag displays your document title in the menubar of the web browser.
<META>...</META>	Meta: Search engines can use the meta tag to find and index web pages. You can also use the meta tag to describe the software used to generate the page, or to provide special information for custom applications.
<b>&lt;Meta&gt; Attributes</b>	
Name	Name is the attribute which tells any program reading the web page the purpose of that <meta> tag. Name may include “description”, “author,” or many other descriptive terms.
Content	Content follows the name attribute, and provides the text assigned to that <meta> tag.

Here are some examples of descriptive tags:

```
<head>
<META NAME="description" CONTENT="E.A. Poe"></META>
</head>
```

This example would allow search engines to index this page under “E.A. Poe”, so that any searches for “Poe” or “E.A.” would include this document in the search results.

```
<head>
<title>My First Web Page</title>
</head>
```

This example would display “My First Web Page” in the menubar of the web browser.

## Formatting Tags

Formatting tags are used inside the <BODY> of a document to format your text. This is the largest category of tags and you will find yourself continually using these tags to code your HTML documents.

The basic text formatting tag is <P>, meaning paragraph. The



To see some of these on the web, [click here](#).

<P> tag does not need a closing tag. The browser knows that a paragraph has ended when it encounters a break tag <BR> or another <P> tag.

---

Tag	Name: Description
<B>...</B>	<b>Bold:</b> This tag boldfaces the enclosed text.
 	<b>Break:</b> This tag creates a blank line or carriage return.
<BLOCKQUOTE>...</BLOCKQUOTE>	<b>Blockquote:</b> This tag was designed for use with long quotations. As such it indents the enclosed text.
<CENTER>...</CENTER>	<b>Center:</b> This tag centers the text on the screen.
<EM>...</EM>	<b>Emphasis:</b> This tag emphasizes text according to the default emphasis mode for the web browser; the text will typically be displayed in italics.
<H1>...</H1> to <H6>...</H6>	<b>Heading:</b> H1 makes for a large heading and H6 is the smallest.
<HR>	<b>Horizontal rule:</b> This tag draws a line across the page, and is useful to separate different sections of a single page.
<I>...</I>	<b>Italics:</b> Italicizes the displayed text.
<P>	<b>Paragraph:</b> This tag indicates a paragraph of text, with a space between each paragraph. The paragraph tag does not require a closing tag.
<STRONG>...</STRONG>	<b>Strong:</b> This tag is very similar to the emphasis tag — it emphasizes text according to the default <i>strong</i> mode for the web browser; the text will typically be displayed in bold.
<U>...</U>	<b>Underline:</b> This tag underlines the enclosed text. Be careful where you use this tag, because links are automatically underlined, and you may confuse your readers.

---

Tag	Name: Description
<code>&lt;FONT&gt;...&lt;/FONT&gt;</code>	Font: The <code>&lt;font&gt;</code> tag allows you to specify typeface and font size on your web page. However, not all computers will have the font you specify, and not all web browsers support the <code>&lt;font&gt;</code> tag. If the either the font or the tag is unsupported, the text is displayed in the browser's default font.
<b>&lt;FONT&gt; Attributes</b>	
FACE	The face attribute specifies the font style you want to display.
SIZE	The size attribute specifies the size of the font .Value can range from 1 to 7. The default is 3, SIZE can be set as a relative value using + or - with a range of -3 to +4.
COLOR	The color of a font can be specified using a hexadecimal number value six characters long. Typical colors are: Black = #000000, White = #FFFFFF, Red = #FF0000, Green = #008000



Be careful using FONT SIZE. Your audience may have a different size monitor than you or have it set to a different resolution. Always consider using heading tags to modify size, first.

The font tag can be tricky to use at first. More than one FACE should be specified, in case a particular browser does not have the font style you indicated. To give you an idea on how to use it, here is an example:

```
<FONT FACE="helvetica, arial" SIZE="7"
COLOR="#FF0000">The Written Word</FONT>
```

The displayed text would in helvetica or, if the computer reading the document did not have helvetica, arial. The font would be displayed in 7-point type.

## Lists

HTML has many ways to format lists information. This is one of the areas in which HTML really shines, and it warrants a separate discussion:

You can create numbered lists which display each list item with sequential numbers, or unnumbered lists which display each list item with a bullet before it. You can even create a glossary list, which will



To see some of these on the web, [click here](#).

display a term and its definition in a manner similar to the glossary of a book. Past versions of HTML supported two other types of lists, MENU and DIRECTORY. These are no longer supported by HTML 3.0, because these effects can be produced using other tags.

Lists can be nested one inside the other. For example, you can insert a numbered list inside an unnumbered list or vice versa. If you are nesting lists, take care to use the tags correctly: Indicate the beginning and ending of each list.

Tag	Name: Description
<code>&lt;OL&gt;...&lt;/OL&gt;</code>	Numbered list: This tag marks the beginning and end of a numbered list.
<code>&lt;UL&gt;...&lt;/UL&gt;</code>	Unnumbered list: This tag marks the beginning and end of a bulleted list.
<code>&lt;LI&gt;</code>	List item: This tag marks one line of text (separated by a carriage return in your text editor) as an item in a numbered or bulleted list. No ending tag is required.
<code>&lt;DL&gt;...&lt;/DL&gt;</code>	Glossary list: This tag marks the beginning and end of a glossary list. The glossary list will create a dictionary-style listing of terms and definitions.
<code>&lt;DT&gt;</code>	Glossary list term: This tag marks a term in a glossary list.
<code>&lt;DD&gt;</code>	Glossary list definition: This tag marks one line of text (separated by a carriage return in your text editor) a term in a glossary list.

### ***How to make a list***

Lists actually require two sets of tags: The list tag itself, and the tag or tags used to define individual list items. If you want to create an numbered or bulleted list, for example, you would need to use the `<UL>` or `<OL>` tags to designate the beginning and end of the list, plus the `<LI>` tag to designate each list item:

```
<UL>
<LI>This is the first bulleted list item
<LI>This is the second bulleted list item
<LI>This is the last bulleted list item (for now)
</UL>
```

```
<OL>
<LI>This is the first numbered list item
```

```
<LI>This is the second numbered list item
<LI>This is the final numbered list item
</OL>
```

For a glossary list, you must use the `<DL>` tag to define the beginning and end of the list, plus the `<DT>` and `<DD>` tags to indicate terms and definitions:

```
<DL>
<DT>Spam<DD>A trademark for a kind of canned lun-
cheon meat made from pieces of seasoned pork and ham
pressed into a loaf.
<DT>Sp Am<DD>Spanish America or Spanish-American
</DL>
```

## The `<PRE>` Tag

When you code text into an HTML format, browsers will ignore excess blank spaces and indents. You have to use tags to indicate large blank spaces, such as indents, with one exception: The `<PRE>` tag.

The `<PRE>` tag indicates the enclosed text should be displayed in the fashion that it has been entered. Any blank spaces you added among the text characters are displayed by the web browser. This allows several effects:

- Tables (though you can now use the `<TABLE>` tag to do this)
- ASCII artwork
- Create formatted web pages. `<PRE>` tag can enclose text cut from another document, because the white space in the document is maintained the formatting of the text inside the `<PRE>` tags will be the same as it was in the source document.

## Summary

You have just learned the basic structure and elements of a web page, as well as the most common types of tags. With this information you can design and create basic, stand-alone web pages.



# Hypertext

**In this chapter:**      [Using Hypertext](#)

# Using Hypertext

**Hypertext** refers to online documents which are linked together with clickable words, phrases, or images. This allows the user to jump from one part of a document to another part of the same document, or from one document to another. In addition, hypertext links on the WWW can connect the user to software, pictures, or any other files.

Hypertext involves two components: A link, and an anchor.

## Links

A link points to a location on the Internet. This location may be a bookmark inside the same document, or another document or file. To create a link, you need the URL of the jump destination.

## Anchor - The Linking Tag

With an anchor, you can create hypertext: That is, you can link two documents together.

An anchor must always have an attribute with it. “**HREF**” and “**NAME**” are the two most commonly used attributes with anchor:

<b>Attribute</b>	<b>Name: Description</b>
HREF	Hypertext Reference: This attribute points the link to a bookmark, another file, either within the same web site, or elsewhere on the Internet.
NAME	Name: The name of a bookmark. This attribute lets you “bookmark” a location on a web page. An HREF anchor can then point a link to that area of the page.

Think of HREF and NAME as the wide and narrow focus settings on a camera: You can use HREF alone to point to a URL, and allow the reader to view a file from the beginning. Or, you can use HREF to point to a specific *area* of that file, indicated by a NAME bookmark, so that the reader goes straight to that section of the document. This is useful in long files.

Here is an example of an HREF anchor used to point to [www.writtenword.com](http://www.writtenword.com):



When designing your own web site always use relative URLs to describe different pages in your site. If you have a link going to another web site use the absolute URL. See page 6.



Want to see some links on the web? [Click here.](#)

This:

```
<A HREF="HTTP://WWW.WRITTENWORD.COM"> The Written Word</A>
```

Will look like this when viewed in a web browser:

The Written Word

The user then follows the link by clicking on the link with a mouse. After the user clicks on the link, the default home page for [www.writtenword.com](http://www.writtenword.com) will appear on the browser, from there the user should have links directing them to various pages that make up the site.

But you may want the user to immediately see a particular section of a particular page. To do that, you need to use a more detailed URL, and a bookmark. Let's say you want the user to have a direct link to the luncheon meats section of the page [food.htm](http://www.writtenword.com/food.htm) on [www.writtenword.com](http://www.writtenword.com).

1. First, you would enter a bookmark into the luncheon meats section, in HTML:

```
<A NAME=LUNCHMEAT>
```

That will not show up in a web browser. It is strictly a way for you to focus your hypertext links more tightly.

2. Now, enter the HREF anchor on the page where you want the link:

```
<A HREF="HTTP://WWW.WRITTENWORD.COM/FOOD.HTM#LUNCHMEAT">Learn about what the staff have for lunch</A>
```

All the user will see is:

[Learn about what the staff have for lunch.](#)

When a user clicks on that line, the web browser will load [food.htm](http://www.writtenword.com/food.htm), scrolled to the "lunchmeat" bookmark.

## Types of Links

A link can go from one type of protocol to another: Links in your web site can connect your users to any files on the Internet:

```
<A HREF="FTP://MYSTUFF">  
<A HREF="GOPHER://COOLBOOKS"  
<A HREF="A_PROGRAM.EXE"
```

## Summary

Links and anchors are one of the distinct features of HTML documents: The connection to other documents and to different locations within those documents allows you a great deal of flexibility in presentation.

# Tables

**In this Chapter**

[Introduction to Tables](#)

# Introduction to Tables

One of the most versatile and useful tools for arranging information on an HTML page is the table tag. The table tag allows you to arrange text or images in a table, like on a written page. Conveniently, you can put any HTML code you desire inside table tags, which can make tables very useful for arranging page layout.

## Basic Table Syntax

You need to use several different tags to build a table:



To see some sample tables, [click here](#).

---

Tag/Attribute	Name: Description
<code>&lt;TABLE&gt;...&lt;/TABLE&gt;</code>	Table: This creates the table and designates the beginning and end of that table.
<code>&lt;TR&gt;...&lt;/TR&gt;</code>	Table row: This tag creates a row within a table, and must be part of a <code>&lt;TABLE&gt;</code> element.
<code>&lt;TH&gt;...&lt;/TH&gt;</code>	Table heading: This tag creates a heading for each row or column. Some browsers will automatically emphasize text enclosed by these tags. This tag must be part of a <code>&lt;TABLE&gt;</code> element.
<code>&lt;TD&gt;...&lt;/TD&gt;</code>	Table cell: This tag creates a single cell inside a table, or inside a row, and must be part of a <code>&lt;TABLE&gt;</code> element. If the table has more than one row, then the cell tag must be part of a row element.

---

## Spanning

You can make one cell straddle two or more rows and/or columns using the COLSPAN or ROWSPAN attributes:



We use tables a lot to format our pages. Take a look: [Click here](#), and then view the source for the page you visit.

---

<b>Attribute</b>	<b>Name: Description</b>
COLSPAN	Column span: This attribute forces a cell to span two or more columns. This attribute must be part of a table cell element.
ROWSPAN	Row span: This attribute forces a cell to span two or more rows. This attribute must be part of table cell element.

---

These attributes, used with the <TD> tag, are useful in controlling the layout of your page.

For a table with two rows, the first row containing a single cell and the second row containing two cells the code might look like this:

```
<TABLE>
<TR>
<TD COLSPAN=2>The ants went marching one by one</TD>
</TR>
<TR>
<TD>Hoorah!</TD><TD>Hoorah!</TD>
</TR>
</TABLE>
```

# Additional Table Tags and Attributes



All of these tags and attributes must be part of a table element.

There are a variety of additional attributes and tags that can be used to customize a table to create the appearance you desire:

Tag/Attribute	Name: Description
<code>&lt;CAPTION&gt;...&lt;/CAPTION&gt;</code>	<b>Caption:</b> This tag creates a caption for the table; the default position for the caption is at the top of the table. <code>ALIGN=BOTTOM</code> can be used to put a caption below a table.
<code>ALIGN=LEFT, RIGHT, CENTER</code>	<b>Alignment:</b> This attribute can horizontally align the contents of a table cell to the <code>LEFT</code> , <code>CENTER</code> , or <code>RIGHT</code> . Must be part of a table cell element. Use with the <code>&lt;TD&gt;</code> and <code>&lt;TR&gt;</code> tags.
<code>VALIGN=TOP, MIDDLE, BOTTOM</code>	<b>Vertical alignment:</b> This attribute can vertically align the contents of a table cell or caption. Use with the <code>&lt;TD&gt;</code> and <code>&lt;TR&gt;</code> tags.
<code>BORDER=#</code>	<b>Border:</b> Indicates the width of the table border, in pixels. <code>BORDER=0</code> can be used to suppress the border. Used with the <code>&lt;TABLE&gt;</code> tag.
<code>CELLSPACING=#</code>	<b>Cell spacing:</b> Indicates the amount space, in pixels, between each cell. Must be part of a table cell element. Used with the <code>&lt;TABLE&gt;</code> tag.
<code>CELLPADDING=#</code>	<b>Cell padding:</b> Indicates the amount of space, in pixels, between cell borders and its contents. Must be part of a table cell element. Used with the <code>&lt;TABLE&gt;</code> tag.
<code>NOWRAP</code>	<b>No wrap:</b> This attribute deactivates automatic text wrap in a cell. Must be part of a table cell element. Used with the <code>&lt;TABLE&gt;</code> tag.

## Summary

Tables are one of the more versatile and convenient tools in HTML. Work with tables as often as you can. They are complex, but the options for page layout are endless, and the more you use them, the more intricate page layout you can create.

# Images

**In this chapter**

[Introduction to Images](#)

# Introduction to Images

Hypertext is one important feature of HTML. The other is the ability to mix graphics and text in a cross-platform medium. In this chapter we will discuss how to incorporate in-line images to your HTML documents.

## The Image

The WWW uses two standard image formats: GIF and JPG. Both of these formats were designed for use on the web, and on different computer platforms. The differences between the formats is beyond the scope of this book — if you want to learn more, there are many documents on the web explaining the formats, as well as enough books to fill a small library.

If you do not have GIF or JPG format images, you can find literally hundreds of pieces of clip art, and image conversion utilities, on the Web — try searching for them on Yahoo. Before putting these images on your site, be sure they are royalty free.

## The Image Tag

To insert an image into an HTML document, you must use the image tag, and the source element:

Tag/Attribute	Name: Description
<IMG>	Image: This tag places an image in an HTML document.
<b>&lt;IMG&gt; Attributes</b>	
<i>SRC="imagefilename"</i>	Source: This attribute specifies the location of the image to be displayed.
<i>ALIGN=TOP, MIDDLE, BOTTOM</i>	Alignment: This attribute positions the image in relation to its surrounding text.
<i>ALIGN=RIGHT, LEFT, TEXTTOP, ABSMIDDLE, BOTTOM</i>	Alignment (Netscape options): Netscape defined additional alignment options. Most of these options will also work in Internet Explorer.
<i>ALT="text"</i>	Alternate: This attribute specifies a line of text to display during a slow download, or if a browser does not support images.

Tag/Attribute	Name: Description
VSPACE=#	Vertical Spacing: This attribute indicates vertical space around the image, in pixels.
HSPACE=#	Horizontal Spacing: This attribute indicates horizontal space around the image, in pixels.
BORDER=#	Border: This attribute defines a border around an image. If you set border to 0, then images used as links will not show a blue border.
LOWSRC=#	Low Resolution Source: This attribute sets the URL for a lower resolution/smaller file to load before the main image is loaded. This is handy if you are using a lot of large, slow-downloading files, because it lets your users get a preview while they wait.



If you would like to see some sample images, [click here](#).

If you wanted to display the file `image.gif` in your HTML page, you would insert the following:

```
<IMG SRC="IMAGE.GIF">
```

To combine images and text, simply insert the image tag inside the text element:

```
<H1>This is an image:<IMG SRC="IMAGE.GIF"></H1>
```

Images can also be used inside an anchor element (`<A>...</A>`) to make clickable images, or “buttons,” that take the user to different locations. For example:

```
<A HREF="NEXTPAGE.HTML"><IMG SRC="BUTTON.GIF"></A>
```

images and text are displayed together.

## Summary and a Word of Caution

Images add a great deal to a web site — nothing will attract visitors more than a well-designed, graphically-rich site.

**However**, images also add a great deal to the size of the web site. And the larger the web site, the slower the download for each page. The most beautiful site in the world will not attract return visitors if it takes 20 minutes to download to the browser.

In short: Have mercy on your users. Use only those images that add value to the site itself, and keep the image sizes as small as possible.

# Embedding

**In this chapter**

[Introduction to Embedding](#)

# Introduction to Embedding

In the last section, you learned how to add images to your web page. But HTML lets you add a variety of other special objects to create a true multimedia experience, including movies and sound. In addition, you can use your web page to deliver files to your users.



Whenever you embed a file, be sure to warn your audience of its size. Some movies and sound files tend to be quite large.

While most web browsers cannot read movies or sound files on their own, most are designed to use programs resident — called *helper applications* — on your computer to read the files after download. MPEG's, ZIPed files, and WAV's are just a few of the different types of objects that may be embedded into an HTML document.

## Sound

You can make sound files available for download in the same manner as images. If the user clicks on a link to a sound file, the browser downloads the file, and attempts to play it using the designated helper application. Most browsers will automatically select a player and then play the sound sample once it is downloaded.



For some examples of embedded objects, [click here](#). But LOOKOUT!!! The embedded files are quite large, and may take some time to download.

You should always tell your readers that they are about to download a sound file, and warn them of the size of the file. Your code would then look something like this:

```
<A HREF="CATERWAULING.WAV">Hideous noise, WAV  
format, 320K</A>
```

## Video

Motion pictures can be digitally encoded and saved. If you visit web sites for motion picture studios you will frequently find trailers to current movies are available. Other sites have different movies available as well; game demos, animated logos and pirated television clips are frequently found. To obtain movies for your own use, look for royalty free movies on CD-ROM.

The most common extensions for a movie are .mpg for MPEG files and .mov for QuickTime movies. You anchor these just like any other embedded file.

```
<A HREF="WHOA.MPG">My horse drags me to Phoe-  
nix, 1420K</A>
```

## Other Files

The least exciting, but most common, use for embedding is to deliver document or software files to your users. You can create a link to a

word processing document, for example, so that users can click on the link and download the file:

```
<A HREF="WARANDPEACE.DOC">This is a really big  
file.</A>
```

## Summary

Embedding can make a wide variety of media available to your audience. It is the simplest method for delivering information to users via the WWW.



# Forms

**In this chapter**

Forms: Making Web Pages Interactive

Creating a Form

# Forms: Making Web Pages Interactive

So far, you have learned how to create and display graphical information on the World Wide Web. That, in itself, is pretty great. However, your visitors still cannot communicate with you, except by email — you have no way to solicit specific information from them. That is where forms come in.

## A Sample Form



To see this form in use in real life, [click here](#).

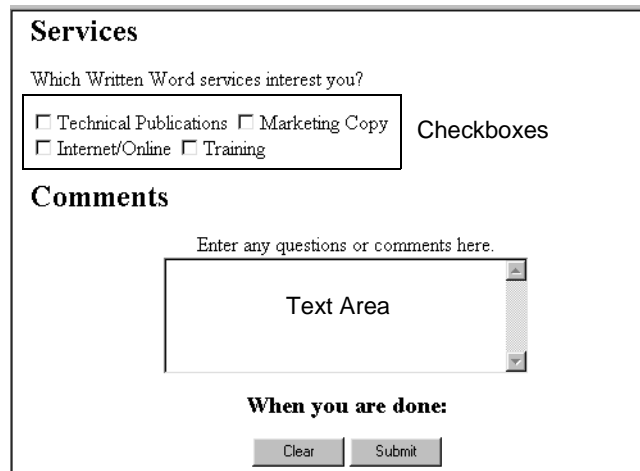
The illustration below shows a typical form used to obtain contact information from web site visitors. Don't sweat the details yet — just get an idea of how the form looks, and what it is doing:

The screenshot shows a web form titled "Contact Us". Below the title is a horizontal line. The text "To contact us with your comments/questions, fill out the form below, or email us at [wrttnwrld@nwlink.com](mailto:wrttnwrld@nwlink.com)." is displayed. The form is divided into sections: "Name", "Address", and "Honorific". The "Name" section has "First name:" and "Last name:" labels, each followed by a text box. The "Honorific" section has a label "Honorific:" followed by radio buttons for "Mr.", "Ms.", "Mrs.", "Miss", and "Dr.". The "Address" section has "Title:", "Organization:", and "Street address:" labels, each followed by a text box. Annotations include a box around the "Last name" text box labeled "Text Box" and a line pointing to the radio buttons labeled "Radio Buttons".

Web site visitors can enter their name, address and other information in the *text boxes* and by clicking the appropriate *radio button*.

In addition, visitors to The Written Word web site can indicate the services about which they would like to learn more by selecting from the *checkboxes*. Finally, visitors can use the *scrolling text box* to enter their thoughts on the web site. When they are done, visitors can click

the Submit button to send the information, or click Clear to wipe the form clean and start over again.



The screenshot shows a web form with the following elements:

- Services**: A section header.
- Which Written Word services interest you?**: A question text.
- Checkboxes**: A label for a group of four checkboxes:  Technical Publications,  Marketing Copy,  Internet/Online, and  Training.
- Comments**: A section header.
- Enter any questions or comments here.**: A prompt text above a text area.
- Text Area**: A large rectangular input field with a vertical scrollbar on the right side.
- When you are done:**: A label for the final buttons.
- Clear** and **Submit**: Two buttons at the bottom of the form.

## What Forms Do

Forms connect your visitors to you or to an automated system you set up. An HTML form acts just like a form on a piece of paper, only better: The form contains a list of questions, checkboxes and other requests for information. When a visitor fills out the form and clicks Submit, the web browser transmits the information to the web server, which can use that information for any number of useful tasks:

- **Transmission:** The server can send the information directly to you, via email.
- **Database:** The server can assemble the information in a database, which you can search or use for research.
- **Orders:** The server can take the information from the form, process it, and deliver software, or order delivery of another product type, according to items selected by the visitor.

The possibilities are truly endless.

## How Forms Work

A form is a set of *fields* which can record information. The form itself really cannot do anything else.



CGI scripts are not the only way to handle form information. There are also off-the-shelf programs, such as [Allaire's Cold Fusion](#).

Big deal, you say? Well, forms become powerful tools when connected to a server application. For example, when a visitor clicks Submit on the sample form you saw on the last page, their web browser transmits all of the information in the fields to a *Common Gateway Interface*, or CGI, script. The script is a simple program that, in this case, automatically transmits the information to web master at The Written Word, Inc.

CGI and other web-based data-handling systems are outside the scope of this guide. However, if you are using a web hosting company to host your web page, call them. Typically, these services have a pre-designed set of CGI scripts, plus documentation on how to connect them to your forms. This will save you the sometimes painful process of learning to use the CGI. Don't tell anyone, but all of the web sites we set up use our web hosting service's pre-designed scripts!

- **Checkboxes** allow the user to select one or more items from a group of items.
- **Scrolling text boxes** behave just like a regular text box, except that the user can enter more information.
- **Buttons** allow the user to perform a command, usually involving the form itself.
- **Text boxes** allow the user to type words or phrases.
- **Radio buttons** allow the user to select one item from a group of options.

# Creating a Form

## Inserting a Form in Your Web Page

Forms are generated using tags and attributes, just like any other HTML element. To create a form, you must use the `<FORM>` and field tags.

### The `<FORM>` Tag

The `<FORM>` tag tells the web browser how to handle the form data.

#### *`<FORM>` is Like Most Tags*

`<FORM>` is similar to most tags:

- It must have a `<FORM>` and `</FORM>` to complete the element.
- You can nest other elements inside a form element.
- It has attributes.

However, there are some unique features of the form element which make it at once more complex, and more versatile.

#### *`<FORM>` is Different, Too*

Unlike most other tags, a form has certain mandatory attributes — without the attributes, the form does not work.

Form attributes tell the web browser where to send the data when the visitor clicks the Submit button:

### Field Tags

Field tags specify an area on a form in which a visitor may enter data or otherwise indicate an answer to a prompt.

# Form and Field Tags List

Tag	Name: Description
<FORM>...</FORM>	Form: Defines the beginning and end of a form.
<b>&lt;FORM&gt; Attributes</b>	
ACTION= <i>"script"</i>	Action: This attribute specifies the URL of the script used to handle data sent by visiting web browsers from a specific form.
METHOD= <i>post, get</i>	Method: This attribute specifies how the input from the form will be sent to the server side gateway. The most common method is <i>"post."</i>
<INPUT>...</INPUT>	Input: The input field tag
<b>&lt;INPUT&gt; Attributes</b>	
TYPE	The type attribute specifies the type of field displayed. See the Input Types List below for an explanation of each input type.
<ul style="list-style-type: none"><li>• Text</li><li>• Radio</li><li>• Checkbox</li><li>• Select</li><li>• Submit</li><li>• Reset</li></ul>	
NAME	name= <i>"name"</i> The name attribute tells you what the field is called. If action of the form is to mail you the text entered by the user the <i>"name"</i> will appear next to the contents entered by the user.
SIZE	size= <i>"number"</i> The number defines the length of the text field.
<TEXTAREA>...</TEXTAREA>	Text Area: This displays a scrolling text box on the page, in which a visitor may enter one or more lines of information.
<b>&lt;TEXTAREA&gt; Attributes</b>	
ROWS	Specifies the height of the text box, in rows.

---

<b>Tag</b>	<b>Name: Description</b>
COLUMNS	Specifies the width of the text box, in columns.

---

## Input Types List

---

Input Type	Description and Options
TEXT	<p>Text displays a standard text field in the form.</p> <p><i>name</i>= “<i>name</i>” Describes the field. If action of the form is to mail you the text entered by the user the “<i>name</i>” will appear next to the contents entered by the user.</p> <p><i>size</i>= “<i>number</i>” The number defines the length of the text field.</p>
RADIO	<p>A radio button lets a visitor select one and only one of a set of options.</p> <p><i>name</i>= “<i>category</i>” Describes the set of radio buttons.</p> <p><i>value</i>= “<i>button option</i>” The data that sent to the server if a visitor selects that radio button .</p> <p>Example: If a visitor selects the radio button with the value “MR.” from a set of radio buttons with the name “HONORIFIC”, then the web browser will send “HONORIFIC=MR” to the web server.</p>
CHECKBOX	<p>A checkbox lets a visitor select one or more options from a set of options.</p> <p><i>name</i>= “<i>category</i>” Describes the set of checkboxes.</p> <p><i>value</i>= “<i>button option</i>” The data that sent to the server if a visitor selects that checkbox .</p> <p>Example: If a visitor selects the checkbox with the value “MR.” from a set of checkboxes with the name “HONORIFIC”, then the web browser will send “HONORIFIC=MR” to the web server.</p>
SELECT	<p>This creates a list box, similar in appearance to a Windows ‘95 drop-down menu.</p> <p><i>name</i>= “<i>name</i>” The name of the list box.</p> <p><i>option</i> = “<i>items in the category</i>” The options listed in the list box.</p>
SUBMIT	<p>Submit displays a single grey button that can be selected to submit the contents of the form.</p> <p><i>Value</i>= “<i>words on button</i>” Optional: If used, allows you to customize the button label.</p>

---

---

**RESET**

Reset displays a single grey button that can be selected to clear the contents of the form.

*Value= "words on button"* Optional: If used, allows you to customize the button label.

---



If you want to download the actual source for our contact form and customize it for your web site, [click here](#) to download the text. You can then open it using your computer's text editor, and copy it into a web page. You can also cut and paste it out of this document, if you are viewing it online.

Here is the source HTML for the sample form we showed you earlier in this chapter. Review this sample to see the actual syntax for forms, text areas, and buttons:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
<html>
<head>
<form action="cgi-bin/pubform.pl" method="post">
<input type="hidden" name="form_userid"
value="wrtnwrdr@nwlink.com">
<input type="hidden" name="form_webpage"
value="http://www.writtenword.com">
<input type="hidden" name="form_show" value="no">
<h2>Name</h2>
<p>First name: <input type="text" size="15"
name="name_first">
Last name:<input type="text" size="15"
name="name_last"><br>
Honorific: <input type="radio" checked name="honorific"
value="Mr">Mr.
<input type="radio" name="honorific" value="Ms">Ms.
<input type="radio" name="honorific" value="Mrs">Mrs.
<input type="radio" name="honorific" value="Miss">Miss
<input type="radio" name="honorific" value="Dr">Dr.<br>
</p>
<h2>Address</h2><pre>
Title:      <input type="text" size="30"
name="title">
Organization: <input type="text" size="30"
name="organiz">
Street address: <input type="text" size="25"
name="street_addr">
City:      <input type="text" size="25"
name="city">
State or Prov: <input type="text" size="2"
name="state">
Postal code: <input type="text" size="10"
name="title">
Country:    <input type="text" size="25"
name="country">
Phone:     <input type="text" size="25"
name="phone">
Fax:      <input type="text" size="25" name="fax">
E-mail address: <input type="text" size="34" name="e-
mail">
</pre>
<h2>Comments</h2>
<p align="center">Enter any questions or comments
here.<br>
<textarea name="comments" rows="5" cols="57"></textarea>
</p>
<h3 align="center">When you are done:</h3>
<p align="center"><input type="reset" value="Clear">
<input type="submit" value="Submit"> </p>
</form>
<hr width="75%">
b<</font><</a><</p>
</td>
</tr>
</table>
</div>
</body>
</html>
```



# FTP

**In this Chapter**

[Putting it on the Web](#)

# Putting it on the Web

Once you have made your web site, you want to put it on the WWW. To do this, you need to find a web hosting service who will host your web site on their server.

Then, you need to upload your web site to their server, using an **File Transfer Protocol (FTP)** program. Transferring a site from your desktop to a host machine is not as tricky as it may seem.

The particulars will vary from server to server, and from one FTP program to another. However, this should give you some idea what to expect when you have your own web site prepared for uploading.

Here are the basics:

1. Connect to your ISP using an FTP client.
2. Once logged on you will (normally) automatically be in your own directory.
3. Select files from your local system for transfer and upload them to the host machine.
4. The client will need to have certain preferences selected (assuming you are using a PC system):
  - ASCII mode for .htm and .html files
  - Binary for .gif and .jpg files
5. Inspect the files using a web browser to determine if the transfer has been performed correctly.

Here are a few general rules for transferring files:

- Transfer precisely: Make sure that all of the files get transferred to the same *relative* locations on the web server as they were in on your computer at home. If all of your pictures are in a directory called “pix” on your computer at home, then they should be in a directory called “pix” on the web server. Otherwise, the relative URLs will not work.
- Transfer carefully: Make sure that you transfer everything. Missing files can cripple your site.
- Verify your transfer: Use your FTP software to browse through the files you downloaded. Make sure everything is there. This could save you some debugging time later.



When we copy a file from the WWW to our PCs, we call it **downloading**. When we copy a file from our PC to the WWW, we call it **uploading**.



Most FTP clients also have an **automatic** setting, which will let the software decide whether it needs to use ASCII or Binary mode.

# Glossary

## **Absolute URL**

A file's location, as defined by its position in the system as a whole.

## **Attribute**

A modifier to an HTML *tag* that controls the appearance or behavior of text in the *element*.

## **Client**

Refers to your computer when it is connected to an *ISP*. Client can also be used to refer to browsers or other programs used to access the *WWW*.

## **Element**

A complete set of beginning and ending *tags*, and all of the text surrounded by those tags.

## **File Transfer Protocol (FTP)**

The set of standards that allows transfer of complete files between *hosts* and/or *clients*.

## **Gateway**

A server dedicated to providing access to the *Internet* backbone.

## **Host**

A dedicated computer that is the repository of documents or applications available on a network.

## **Hypertext**

Clickable text that allows navigation through an online document.

## **HyperText Markup Language (HTML)**

The language used to create *web pages*.

## **HyperText Transport Protocol (HTTP)**

The set of standards that allows web browsers to access, link to, and retrieve text and graphical information from *World Wide Web Servers*.

## **Intranet**

An internal network using a *WWW Server* to provide users with access to *WWW*-based services and applications, without *Internet* access. Intranets are often used to serve applications and information within companies that use different platforms, such as UNIX and Windows '95.

## **Internet**

The global system of networks interconnected by the TCP/IP protocol.

**Internet Service Provider (ISP)**

A service that offers dial-up or dedicated access to the *Internet*. Typically, this service is offered on a fee basis, and may include additional services, such as email, web site hosting and *FTP*.

**Internic**

The organization that controls registration of internet domain names.

**Relative URL**

A file's location, as defined by its position compared to the location of another file.

**Server**

see host.

**Tag**

The basic formatting and structural unit in *HTML*. Tags can modify and format text, insert images, or create forms.

**Uniform Resource Locator (URL)**

A naming convention by which Web browsers can locate and access information on the *Internet* or on an *Intranet*.

**Web Browser**

A program used to view HTML files

**Web Page**

An *HTML* document that is interpreted by a *web browser*.

**World Wide Web (WWW)**

A graphical, interactive, hypertext information system designed for cross-platform compatibility, that is run over the *Internet*.