

Chapter 1

Introduction to Web and HTML

What is the World Wide Web?

World Wide Web- the web, for short- is a network of computer able to exchange text, graphics, and multimedia information via the internet. By sitting at a computer that is attached to the web, using either a dialup phone line or a much faster broadband (Ethernet, cable, or DSL connection, you the can visit web- connectors next door, at a nearby university, or halfway around the world. And you can take full advantage of the resources these computers make available, including text, graphics, videos, sounds, and animation. Think

Of the web as the multimedia version of the internet, and you'll be right on the mark. The World Wide Web has come a long way from its humble beginnings, most internet historians recognize Gopher as the precursor to the web. Gopher was a revolutionary search tool that allowed the user to search hierarchical archives of textual documents. It enabled internet users to easily search, retrieve. And share information. Today's World Wide Web is capable of delivering information via any number of medium-text, audio, video. The content can be dynamic and even interactive. However, the web is not a panacea. The standards that make up the HTTP protocol are implemented in different ways by different browsers. What works on one platform may not work the same, if at all, on the next. Newly web- enabled devices- PDAs, cell phones, appliances, and so on- are still searching fop a suitable form of HTML to standardize on.

How Does the web work?

The computers that make all these web pages available are called web servers. On any computer that's connected to the web, you can rub nab application called a web browser. Technically, a web browser is called a web client-that is, a program that's able to contact a web server and request Information

When the web server receives the requested information, it looks for this information within its file system, and sends out the requested information via the internet. They all speak a common "language," called Hyper Text Transfer protocol (HTTP). (HTTP isn't really a language like the ones people speak. It's a set of rules or procedures, called protocols that enable computers to exchanges information over the web.) Regardless of where these computer reside-china., Nor- way, or Austin, Texas-they can communication with each other through HTTP.

The following illustrates how HTTP works (see figure)

- Most web pages contain hyperlinks, which are specially formatted words or phrases that enable you to access another page on the web. Although the hyperlink usually doesn't make the address of this page visible, it contains all the information needed for your computer to request a web page from another computer.

- When you click the hyperlink your computer sends a message called an HTTP request. The message says, in effect, "Please send me the web page that I want."

The web server receives the request, and looks within its stored files for the web page you request. When it finds the web page, it sends it to your computer, and your web browser displays it. If the page isn't found, you see an error message, which probably the HTTP code for this error' 404 "Not found."

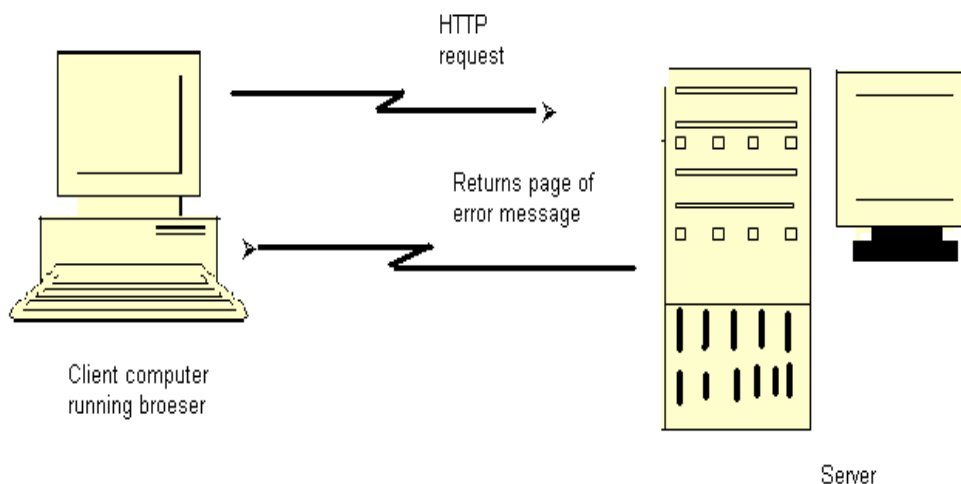


Figure: The client requests the page. Then the server evaluates the request and server the page or an error message.

What Is Hypertext?

You probably noticed the word "hypertext" in the spelled-out version of HTTP, Hypertext Transfer Protocol. Originated by computing pioneer Theodore Nelsons, the term "hypertext" doesn't mean "text that can't sit still," although some web authors do use a much-despised HTTP code that makes the text blink on-screen. Insert, the tern is an analogy to a time-honored (but physically impossible) science fiction concept, the hyperspace jump, which enables a starship to go immediately from one star system to another. Hypertext is a types of text

that contains hyperlinks (or just links for short), which enable the reader to jump from one hypertext page to another.

You may also hear the word hypermedia. A hypermedia system works just like hypertext, except that it includes graphics, sounds, videos, and animation as well as text. In contrast to ordinary text, hypertext gives readers the ability to choose their own path through the material that interests them. A book is designed to be read in sequence; page 2 follows page 1, and so on. Sure, you can skip around, but books don't provide much help, beyond including an index. Computer – based hypertexts let readers jump around all they want. The computer part is important because it's hard to build a hypertext system out of physical media, such as index cards or pieces of paper

The web is a giant computer- based hypermedia system, and you've probably already done a lot of jumping around from one page to another on the web-it's called surfing. If one web page doesn't seem all that interesting once you visit, you can click another link that seems more related to your needs (and so on). The web makes surfing so easy that you'll need to give some thought to keeping people on your sites-keeping them engaged and interested-so they won't surf away!

Where Does HTML Fit In?

Hypertext Markup Language (HTML) enables you to mark up text so that it can function as hypertext on the web. The term markup comes from printing; editors mark up manuscript pages with funny- looking symbols that tell printer how to print the page. HTML consists of its own set of funny-looking symbols that tell web browsers how to display the page. These symbols, called elements, include the ones needed to create hyperlinks.

The Hyper Text Markup Language (HTML) IS A Simple markup language that describes the structure and behavior of Web document. HTML is the standard language that all Web browsers are designed to; understand and interpret. HTML is implemented using another larger markup language—the Standard Generalized Markup Language, usually known by its acronym, SGML

The invention of HTML

HTML and HTTP were both invented by Tim Berners-Lee, who was then working as a computer and networking specialist at a Swiss research institute. He wanted to give the Institute's researchers a simple markup language, which would enable them to; share their research papers via the internet. Berners-Lee based HTML on Standard Generalized Markup Language (SGML), an international standard for marking up text for presentation on a variety of physical devices. The basic idea of SGML is that the document's structure should be separated from its presentation:

- Structure refers to the various components or parts of a document that authors create, such as titles, paragraphs, headings, and lists. For example, you're reading an item in an unordered list, as it is termed in SGML (most people use the more familiar bulleted list). In SGML, you markup this item as a bulleted list, but you don't say anything about how it's supposed to look. That's left up to whatever device displays or prints the marked-up file

- Presentation refers to the way these various components are actually displayed by a given media device, such as a computer or a printer. For example, this book displays this bulleted list item with an indentation and other special formatting.

- What's so great about separating structure from presentation? There are several very important advantages:

- Authors usually aren't very good designers. It's wise, especially in large organizations, to let writers compose their documents, and let designers worry about how the documents are supposed to look. That's particularly true when an organization has a corporate look or style, such as Apple computer's standard typeface, which you'll see in all of its documents. The designers make sure that every document produced with the organization conforms to that style. So SGML doesn't contain any features that control presentation.

- If markup consists of structure alone, the document's appearance can be changed quickly. All that's necessary is to change the presentation settings on whatever device is displaying the document.

- Documents containing only structural markup are much easier and cheaper to maintain. When presentation markup is included along with structural markup, the document becomes an unmanageable mess, and maintenance costs skyrocket.

- If a document contains only structural markup, it is more accessible to people with limited vision or other physical limitations. For example, a document marked up structurally might be presented by a Braille printer for those with limited vision, or by a text reader for those with limited hearing.

Sounds great, right? Still, from the beginning, HTML didn't make the structure versus presentation distinction as clearly as SGML purists would have liked. And as HTML developed and the internet became a commercial network, Web authors demanded more tools to make their documents look attractive on-screen. The companies that make web browsers responded by introducing new, nonstandardized HTML elements that contained presentation information. By 1996, many web experts were worried that HTML standards were spiraling out of control. The newly founded world wide consortium, hoping to develop at least some kind of standard in place, tried to standardize existing practices, including the use of presentation and structure. The result was the W3C's HTML 3.2 standard, which is still widely used. But

organizations found that HTML3.2 exposed them to excessive maintenance costs. The SGML purists were right: Structure and presentation should have been kept separate

A Short History of HTML

To date, HTML has gone through four major standards, including the latest 4.01. In addition to the HTML standards, cascading style sheets and XML have also provided valuable contributions to Web standards.

The following sections provide a brief overview of the various versions and technologies.

HTML 1.0

HTML 1.0 was never formally specified by the W3C because the W3C came along too late. It was the original specification Mosaic 1.0 used, and it supported few elements. What you couldn't do on a page is more interesting than what you could do. You couldn't set the background color or background image of the page. There were no tables or frames. You couldn't dictate the font. All inline images had to be GIFs; JPEGs were used for out-of-line images. And there were no forms.

Every page looked pretty much the same: gray background and Times Roman font. Links were indicated in blue until you'd visited them and then they were red. Because scanners and image-manipulation software weren't as available then, image limitation wasn't a huge problem. HTML 1.0 was only implemented in Mosaic and Lynx (a text-only browser that runs under UNIX).

HTML 2.0

Huge strides forward were made between HTML 1.0 AND html2.0. An HTML 1.1 actually did exist, created by Netscape to support what its first browser could do. Because only Netscape and Mosaic were available at the time (both written under the leadership of Marc Andreessen), browser makers were in the habit of adding their own new features and creating names for HTML elements to use those features.

Between HTML1.0 and HTML 2.0, the W3C also came into being, under the leadership of Tim Berners-Lee, founder of the web. HTML 2.0 was a huge improvement over HTML 1.0. Background colors and images could be set. Forms became available. With a limited set of fields, but nevertheless, for first time, visitors to a web page could submit information. Tables also became possible.

HTML 3.2

Why no 3.0? The W3C couldn't get a specification out in time for agreement by the members. HTML 3.2 was vastly richer than HTML 2.0. It included support for style sheets (CSS level 1). Even though CSS was supported in the 3.2 specification, the browser manufacturers didn't support CSS well enough for a designer to make use of it. HTML 3.2 expanded the number of attributes that enabled designers to customize the look of a page (exactly the opposite of HTML 4). HTML 3.2 didn't include support for frames, but the browser makers implemented them anyway.

NOTE a page with two frames is actually processed like three separate pages within your browser. The outer page is the frameset. The frameset indicates to the browser, which pages go where in the browser window. Implement a web site. A common use for frames is navigation in the left pane and content in the right.

HTML 4.0

What does HTML 4.0 add? Not so much new elements- although those do exist-as a rethinking of the direction HTML is taking. Up until now, HTML has encouraged interjecting presentation information into the page. HTML 4.0 now clearly deprecates any uses of HTML that relate to forcing a browser to format an element a certain way. All formatting has been moved into the style sheets. With formatting information strewn throughout the page, HTML 3.2 had reached a point where maintenance was expensive and difficult. This movement of presentation out of the document, once and for all, should facilitate the continued rapid growth of the web.

Tip Use the W3C Markup validation service, available at <http://validator.w3.org/>, to check your HTML against most of the versions mentioned in this chapter.

XML 1.0

Extensible Markup Language (XML) was originally designed to meet the needs of large-scale electronic publishing. As such, it was designed to help separate structure from presentation and provide enough power and flexibility to be applicable in a variety of publishing applications. In fact, many modern word processing contain XML components or even export their documents in XML-compliant formats.

CSS 1.0 AND 2.0

Cascading Style Sheets (CSS) were designed to help move formatting out of the HTML specification. Much like styles in a word processing program, CSS provides a mechanism to easily specify and change formatting without changing the underlying code. The

“cascade” in the name comes from the fact that the specification allows for multiple style sheets to interact, allowing individual Web document to be formatted slightly different from their kin (following department document guidelines but still adhering to the company standards, for example). The second version of CSS (2.0) builds on the capabilities of the first version, adding more attributes and properties for a Web designer to draw upon.

HTML 4.01

HTML 4.01 is a minor revision of the HTML 4.0 standard. In addition to fixing errors identified since the inception of 4.0, HTML 4.01 also provides the basic for meaning of XHTML elements and attributes, reducing the size of the XHTML 1.0 specification

XHTML 1.0

Extensible Hyper Text Markup Language (XHTML) is the first specification for the HTML and XML cross-breed. XHTML was created to be the next generation of markup language, infusing the standard of HTML with the extensibility of XML. It was designed to be used in XML compliant environments, yet compatible with standard HTML 4.01 user agents.

What is an HTML File?

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

Creating an HTML Document

If you are running Window, start Notepad.

If you are on a Mac, start simple Text.

In OSX start Text Edit and change the following preferences: Open the “Format” menu and select “Plain text” instead of “Rich text “. Then open the “preferences “window under the “Text Edit” menu and select “Ignore rich text commands in HTML file “. Your HTML code will probably not work if you do not change the preferences above!

Type in the following text:

```
<html>
<head>
<title>Title of page </title>
</head>
<body>
This is my first homepage. <b> This text is bold
</b>
</body>
</html>
```

Save the file as “mypage.html”.

Start your internet browser “Open “ (or “Open page”) in the file menu of your browser. A dialog box will appear. Select “Browser” (or “Choose File “) and locate the HTML file you just created – “mypage.htm” –select it and click “Open “. Now you should see an address in the dialog box, for example “C:\My Document\mypage.htm”. Click OK, and the browser will display the page.

The first tag in your HTML document is <html>. This tag tells your browser that this is the state of an HTML document. The last tag in your document is </html>. This tag tells your browser that this is the end of the HTML document. The text between the <head> tag and the </head> tag is header information. Header information is not displayed in the browser window.

The text between the <title> tags is the title of your document. The title is displayed in your browser’s caption. The text between the <body> tags is the text that will be displayed in your browser. The text between the and tags will be displayed in a bold font.

HTML or HTML Extension?

When you save an HTML file, you can use either the .htm or the html extension. We have used .htm in our examples. It might be a bad habit inherited from the past when some of the commonly used software only allowed three letter extensions. With newer software we think it will be perfectly safe to use.html.

Chapter 2

Basic HTML Tags And Elements

HTML Elements

HTML documents are text files made up of HTML elements.
HTML elements are defined using HTML tags

HTML Tags

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

HTML Elements

Remember the HTML example from the previous page:

```
<html>
<head>
<title> Title of page </title>
</head>
<body>
This is my first homepage.
</body>
</html>
```

This is an HTML element:

<code> This text is bold </code>

The HTML element starts with a start tag:

``

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

The HTML element ends with an end tag: ``

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

This is also an HTML element:

`<body>`

This is my first homepage. ` This text is bold`

`</body>`

This HTML element starts with the start tag `<body>`, and ends with the end tag `</body>`

The purpose of the `<body>` tag is to define the HTML element that contains the body of the HTML document.

Why do We Use Lowercase Tags?

We have just said that HTML tags are not case sensitive: `` means the same as ``. If you surf the web, you will notice that plenty of web sites use uppercase HTML tags in their source code. We always use lowercase tags. Why?

If you want to follow the latest web standards, you should always use lowercase tags. The World Wide Web Consortium (W3C) recommends lowercase tags in their HTML 4 recommendation, and XHTML (the next generation HTML) demands lowercase tags.

Basic HTML Tags

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

The best way to learn HTML is to work with examples. We have created a very nice HTML editor for you. With this editor; you can edit the HTML source code if you like, and click on a test button to view the result.

The Overall Structure: HTML, Head, and Body

All HTML documents have three, document-level tags in common. These tags, `<html>`, `<head>`, and `<body>`, delimit certain sections of the HTML document.

The `<html>` tag

The <html> tag surrounds the entire HTML document. This tag tells the client browser where the document begins and ends.

```
<html>
.....document contents....
</html>
```

Additional language options were declared within the <html> tag in previous versions of HTML.

However, those options (notably lang and dir) have been deprecated in HTML version 4.0. The language and directional information is routinely contained in the <head> section. A typical <head> section could resemble the following:

```
<head>
<link rel="stylesheet" type="text/css" href="/styles.css">
<title>TITLE OF THE Document</title>
<meta name= "description" content= "Sample Page">
<meta name= "keywords" content= "sample, heading, page">
<script language= "JavaScript ">
Function New Window (url) {
Fin= window. open (url,"",
"width=800, height=600, scrollbars=yes, resizable=yes");
}
</script>
</head>
```

Most of the content within the <head> section will not be visible on the rendered page in the client's browser. The <title> element determines what the browser displays as the page title-on Windows machines, the document title appears in the browser's title bar, as shown in Figure2-1.

Document title

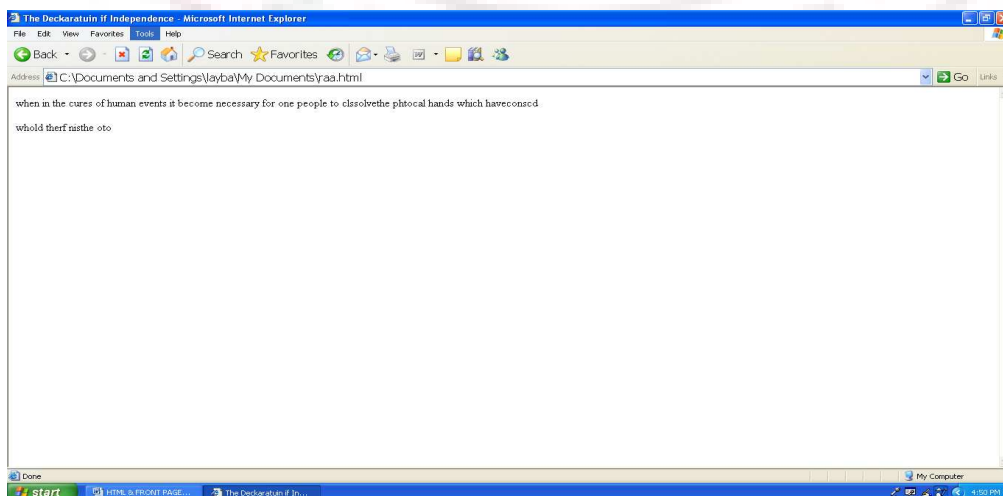


Figure: In windows, the document's <title> appears in the browser's title bar.

The main, visual content of the HTML document is contained within <body> tags. Note that with HTML version 4.0 most attributes of the <body> tag have been deprecated in favor of specifying the attributes as styles. In previous version Of HTML, you could specify a bevy of options, including the document background, text and link colors. The onload and onunload attributes of the <body> tag, as well as global attributes such as style, are still valid. However, you should specify the other attributes in styles instead of within the <body> tag, such as in the following example:

```
<html>
<head>
<title> Document Title</title>
<style type="text/css">
Body {background:black; color:white}
a: link {color: red}
a: visited {color: blue}
a: active {color: yellow}
</style>
</head>
<body>
... document body ...
</body>
</html>
```

The Head Element

The head element contains general information, also called meta-information, about a document. Meta means "information about".

You can say that meta-data means information; also called meta-information means information about information.

Information inside the Head Element

The elements inside the head element should not be displayed by a browser

According to the HTML standard, only a few tags are legal inside the head section. These are: <base>, <link>, <meta>, <title>, <style>, and <script>.

Look at the following illegal construct:

```
<head>
<p>This is some text</p>
</head>
```

In this case the browser has two options:

- Display the text because it is inside a paragraph element
- Hide the text because it is inside a head element

If you put an HTML element like `<h1>` or `<p>` inside a head element like this, most browser will display it, even if it is illegal.

Should browser forgive you for errors like this? We don't think so. Others do.

Head Tags

Tag	Description
<code><head></code>	Defines information about the Document
<code><title></code>	Defines the document title
<code><base></code>	Defines a base URL for all The links on a page
<code><link></code>	Defines a resource reference
<code><meta></code>	Defines meta information

Tag	Description
<code><!DOCTYPE></code>	Defines the document type. This tag goes before the <code><html></code> start tag,

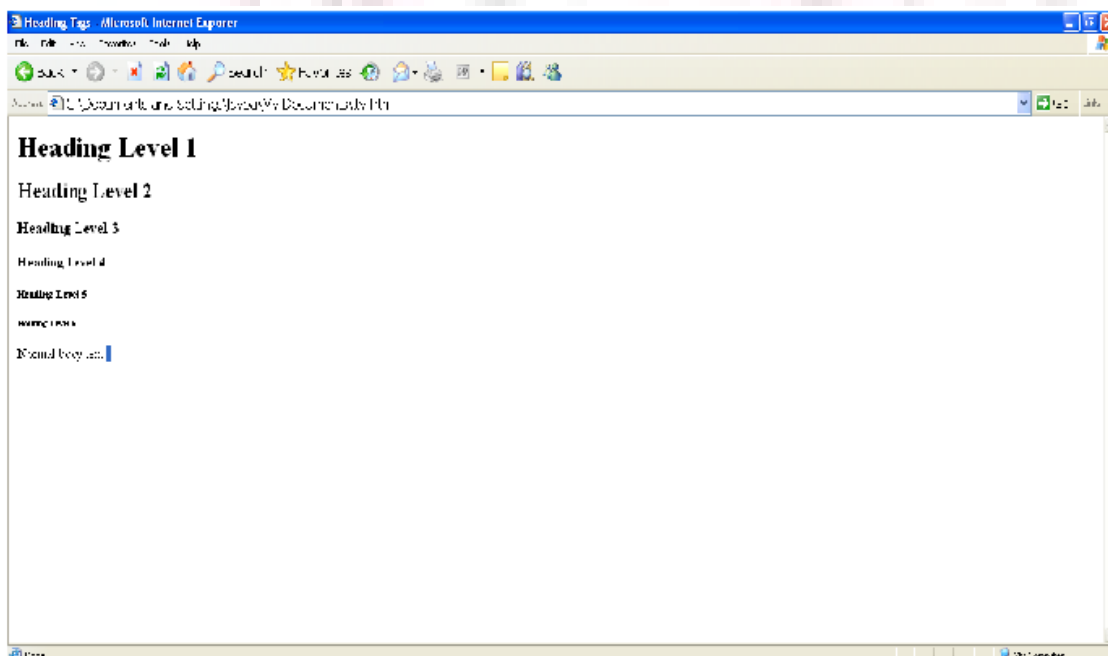
Headings

HTML has six predefined heading tags. Heading use `<h>` tags containing the number of the heading. The `<h1>` tag specifies the highest (most important) level of headings, while `<h6>` specifies the lowest (least important) level of headings.

As with most textual documents, HTML documents use larger fonts to specify higher- level headings. For example consider the following example and its output,

```
<html>
```

```
<head>
<title> Heading Tags </title>
</head>
<body>
<h1> Heading Level 1</h1>
<h2>Heading Level 2</h2>
<h3>Heading Level 3</h3>
<h4>Heading Level 4</h4>
<h5>Heading Level 5</h5>
<h6>Heading Level 6</h6>
<p>Normal body text.</p>
</body>
</html>
```



Headings are defined with the <h1> to <h6> tags. <h1> defines the largest heading. <h6> defines the smallest heading. <h1>This is a heading</h1>

```
<h2> This is a heading</h2>
<h3> This is a heading</h3>
<h4> This is a heading</h4>
<h5> This is a heading</h5>
<h6> This is a heading</h6>
```

HTML automatically adds an extra blank line before and after a heading,

Paragraphs

Paragraphs are defined with the `<p>` tag.

`<p>`This is a paragraph`</p>`

`<p>`This is another paragraph`</p>`

HTML automatically adds an extra blank line before and after a paragraph

Don't Forget the Closing Tag

You might have noticed that paragraphs can be written without end tags

`</p>` : `<p>`this is a paragraph

`<p>`This is another paragraph

The example above will work in most browsers, but don't rely on it. Future version of HTML will not allow you to skip ANY end tags.

Closing all HTML elements with an end tag is a future proof way of writing HTML. It also makes the code easier to understand (read and browse) when you to mark both where an element starts and where it ends.

Line Breaks

The `
` tag is used which you want to break a line, but don't want to start a new paragraph. The `
` tag forces a line break wherever you place it. `<p>` This `
` is a para`
`graph with line breaks`</p>`

Try it yourself

The `
`

Tag is an empty tag. It has no end tag like `</br>`, since a closing tag doesn't make any sense.

`
` or `
`

More and more often you will see the `
` tag written like this: `
`

Because the `
` tag has no end tag (or closing tag), it breaks one of the rules for future HTML (the XML based XHTML), namely that all elements must be closed.

Writing it like `
` is a future proof way of closing (or ending) the tag inside the opening tag, accepted by both HTML and XML.

Comments in HTML

The comment tag is used to insert a comment in the HTML source code. A comment will be ignored by the browser. You can use comments to explain your code, which can help you when you edit the source code at a later date.

```
<!-- This is a comment-- >
```

Note that you need an exclamation point after the opening bracket, but not before the closing bracket.

Recap on HTML Elements

- Each HTML element has an element name (body, h1, p, br)
- The start tag is the name surrounded by angle brackets: <h1>
- The end tag is a slash and the name surrounded by angle brackets </h1>
- The element content occurs between the start tag and the end tag
- Some HTML elements have no content
- Some HTML elements have no end tag

Basic Notes – Useful Tips

When you write HTML text, you can never be sure how the text is displayed in another browser. Some people have large computer displays, some have small. The text will be reformatted every time the user resizes his window.

HTML will truncate the spaces in your text. Any number of spaces counts as one. Some extra information: In HTML a new line counts as one space.

Using empty paragraphs <p> to insert blank lines is a bad habit. Use the
 tag instead. (But don't use the
 tag to create lists. Wait until you have learned about HTML lists.)

HTML automatically adds an extra blank line before and after some elements, like before and after a paragraph, and before and after a heading.

We use a horizontal rule (the <hr> tag), to separate the sections in our tutorials.

Basic HTML Tags

If you lookup the basic HTML tags in the reference below, you will see that the reference contains additional information about tag attributes.

You will learn more about HTML tag attributes in the next chapter of this tutorial.

Tag Description

<html>	Defines an HTML document
<body>	Defines the document's body
<h1> to </h6>	Defines header 1 to header 6
<p>	Defines a paragraph
 	Inserts a single line break
<hr>	Defines a horizontal rule
<!-->	Defines a comment

Horizontal Rules

Horizontal rules are used to visually break up sections of a document. The <hr> tag creates a line from the current position in the document to the right margin and breaks the line accordingly.

For example, if you were reproducing text from a book, you might want to use rules to show a break between chapters, as shown in the following excerpt from Anna Sewell's Black Beauty:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
http://www.w3.org/TR/html4/strict.dtd

```

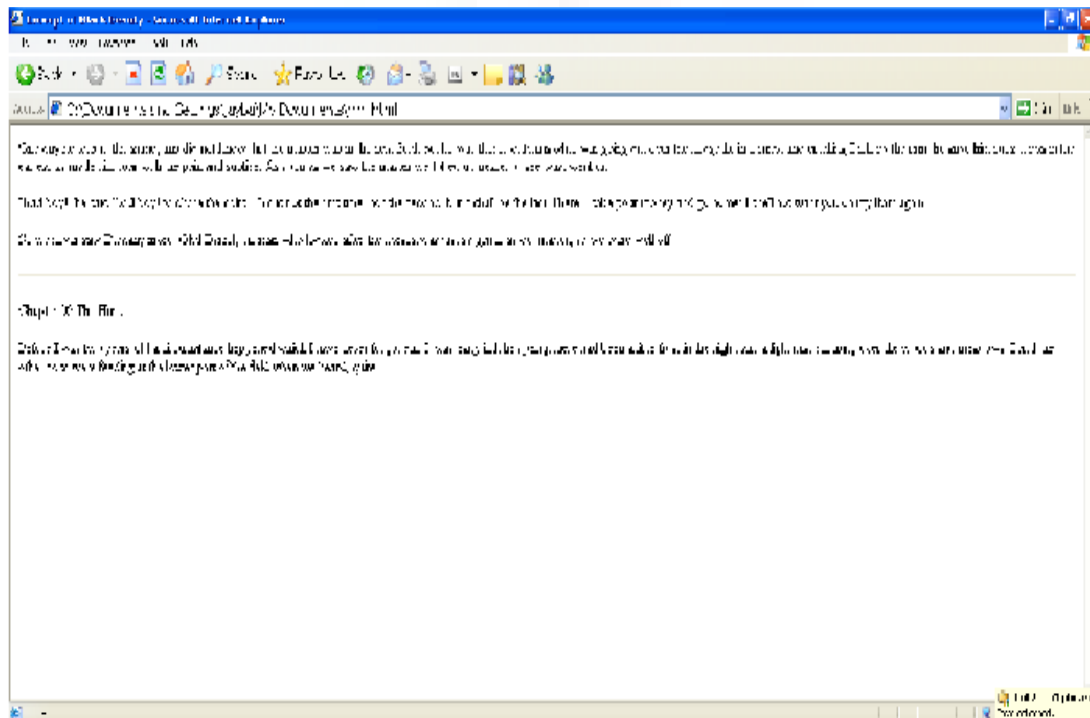
<p>One day he was at this game, and did not know that the master was in the next field; but he was there., watching what was going on; over the hedge he jumped in a snap, and catching Dick by the arm, he gave him such a box in the ear as made him roar with the pain and surprise. As soon as we saw the master we trotted up nearer to see what went on. </p> <p> “Bad boy!” he said, “bad boy! To chase the colts. This is not the first time, nor he second, nut it shall be the last.

There—take your money and go home; I shall not want you on my farm again. ”</p> <p> so we never saw Dick any more. Old Daniel, the men who looked after the horses, was just as gentle as our master, so we were well off. </p> <hr> <p>Chapter 02 The Hunt</p> <p>Before I was two years old a circumstance happened which I have never forgotten. It was

early in the spring; there had been a little frost in the night, and a light mist still hung over the woods and meadows. I and the other colts were feeding at the lower part of the field when we heard, quite ... </p>

</body>

</html>



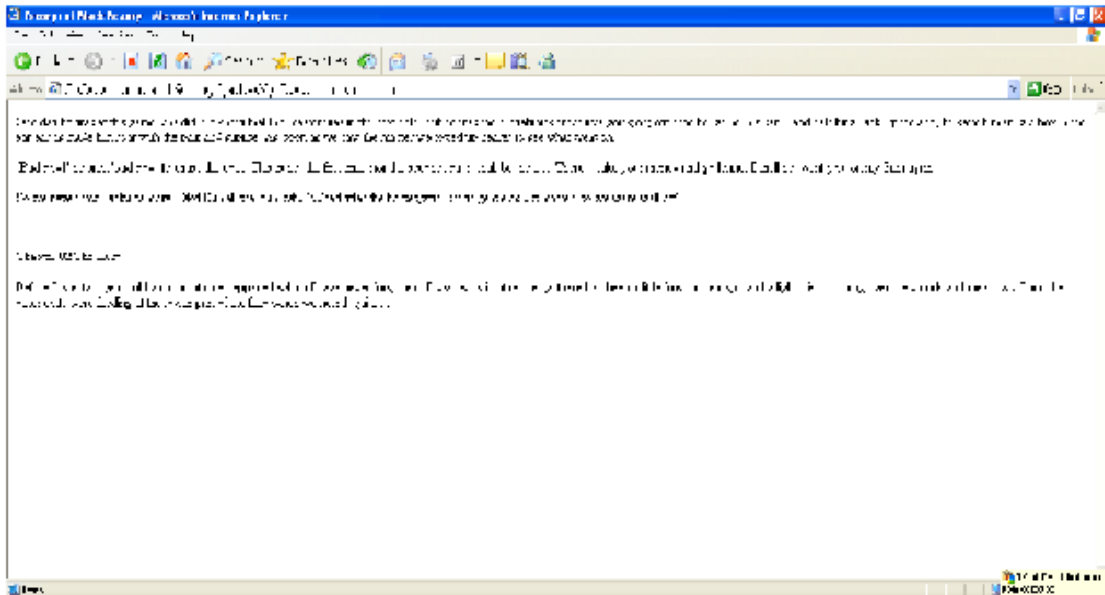
As with most tags, you can customize the look of the <hr> tag by using Styles. For example, consider the following style:

<style type="text/css">

Hr {color: red; height: 5px; width: 50%;}

</style>

If this style were added to our last example. The results would be similar to the output shown in Figure



HTML Attributes

Attributes provide additional information to an HTML element.

HTML Tag Attributes

HTML tags can have attributes. Attributes provide additional information to an HTML element.

Attributes always come in name/ value pairs like this: name = "value".

Attributes are always specified in the start tag of an HTML element.

Attributes Example 1:

`<h1>` defines the start of a heading.

`<h1 align= center">` has additional information about the alignment.

Attributes Example 2:

`<body>` defines the body of an HTML document.

`<body bgcolor="yellow">` has additional information about the background color.

Attributes Example 3:

<table> defines an HTML table. (You will learn more about HTML tables later)

<table border="1"> has additional information about the border around the table

Use Lowercase Attributes

Attributes and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation, and XHTML demands lowercase attributes/attribute values.

Always Quote Attribute Values

Attribute values should always be enclosed in quotes. Double style quotes are the most common, but single style quotes are also allowed.

In some rare situations, like when the attribute values itself contains quotes, it is necessary to use single quotes:

name= 'John "ShotGun" Nelson'

HTML Text Formatting

HTML defines a lot of elements for formatting output' like bold or italic text.

Text Formatting Tags

Tag	Description
	Defines bold text
<big>	Defines big text
	Defines emphasized text
<i>	Defines italic text
<small>	Defines small text
	Defines strong text
<sub>	Defines subscripted text
<sup>	Defines superscripted text
<ins>	Defines inserted text
	Defines deleted text
<s>	Deprecated. Use instead
<strike>	Deprecated. Use instead
<u>	Deprecated. Use styles instead

“Computer Output” Tags

Tag	Description
<code>	Defines computer code text
<kbd>	Defines keyboard text
<samp>	Defines sample computer cods
<tt>	Defines teletype text
<var>	Defines a variable
<pre>	Defines preformatted text
<listing>	Deprecated. Use <pre> instead
<plaintext>	Deprecated. Use <pre> instead
<xmp>	Deprecated. Use <pre> instead

Citations, Quotations, and Definition Tags

Tag	Description
<abbr>	Defines an abbreviation
<acronym>	Defines an acronym
<address>	Defines an address element
<bdo>	Defines the text direction
<blockquote>	Defines a long quotation
<q>	Defines a short quotation
<cite>	Defines a citation
<dfn>	Defines a definition term

Inserting texts, Images

HTML can also specify certain formatting to be applied to the text content of the web page.

- **Font formatting**

Font Face: Default, Arial, Courier, Time New Roman.

Size: -3,-2,-1, default, +1, +2, +3

Colour: Default, Black, Red, Green, Blue

Style formatting:

Bold

Italic

Strong

Emphasis

Code

The text formatting effects used above can be combined, as shown in the following example:

Text formatting effects can be combined by using nested HTML tags

Text can also be formatted to become a hypertext link to another web page, document or resource. This is described on the following page.

For images, HTML can be used to specify the image file to be included, the size (in pixels) at which the image should be displayed by the browser and how the image should be 'wrapped' around the surrounding text. In addition, images can be made into hyperlinks to other internet resources.



A text alternative that will be displayed if the image fails to load should also be provided.

How to View HTML Source

Have you ever seen a Web page and wondered “Hey! How did they do that?”

To find out, click the VIEW option in your browser’s toolbar and select SOURCE to PAGE SOURCE. This will open a window that shows you the HTML code of the page.

Chapter

3

HTML Entities and Links

HTML Character Entities

Some characters like the < character, have a special meaning in HTML, and therefore cannot be used in the text.

To display a less than sign (<) in HTML, we have to use a character entity.

Character Entities

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

A character entity has three parts: an ampersand (&), an entity name or # and an entity number, and finally a semicolon (;).

To display a less than sign in an HTML document we must write: < or <

The advantage of using a name instead of a number is that a name is easier to remember. The disadvantage is that not all browsers support the newest entity names, while the support for entity numbers is very good in almost all browsers.

Note that the entities are case sensitive.

Non-breaking space

The most common character entity in HTML is the non-breaking space.

Normally HTML will truncate spaces in your text. If you write 10 spaces in your text HTML will remove 9 of them. To add spaces to your text, use the character entity.

The Most Common Character Entities:

Result	Description	Entity Name	Entity Number
	Non-breaking space	 	
<	less than	<	<
>	Greater than	>	>
&	Ampersand	&	&
“	Quotation mark	"	"
‘	Apostrophe	'(does Not work in IE)	'

Some Other Commonly Used Character Entities:

Result	Description	Entity Name	Entity Number
¢	Cent	&Cent	
£	Pound	&Pound	£
¥	Yen	&Yen	¥
€	Euro	&Euro	€
§	Section	&Section	§
©	Copyright	&Copyright	©
®	Registered Trademark	&Registered Trademark	®
×	Multiplication	&Multiplication	×
÷	division	&Division	÷

Some Other Commonly Used Character Entities:

Result	Description	Entity Name	Entity Number
¢	Cent	&Cent	
£	Pound	&Pound	£
¥	Yen	&Yen	¥
€	Euro	&Euro	€
§	Section	&Section	§
©	Copyright	&Copyright	©
®	Registered Trademark	&Registered Trademark	®
×	Multiplication	&Multiplication	×
÷	division	&Division	÷

HTML Links

HTML uses a hyperlink to link to another document on the web.

Hyperlinks are what make using the web so easy. By using HTML tags to embed the instructions on where other resources are located on the internet and how to access them, resources can be retrieved from anywhere on the internet by simply clicking on a link.

Although HTML was initially developed as a hypertext system, in which the instructions were embedded within the text of a web page, hyperlinks can now be embedded within forms of media, including images, videos and animation.

The Anchor Tag and the Href Attribute

HTML uses the <a> (anchor) tag to create a link to another document.

An anchor can point to any resource on the web: an HTML page, an image, a sound file, a movie, etc.

The syntax of creating an anchor:

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

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

This anchor defines a link to Rgcsn:

```
<a href=http://www.rgcsn.org/> Visit RGCSM!</a>
```

The line above will look like this in a browser.

Visit RGCSM

The Target Attribute

With the target attribute, you can define where the linked document will be opened.

The line below will open the document in a new browser window : <a href=<http://www.rgcsn.org/> “
target =”_blank”>Visit Rgcsn!

The Anchor Tag and the Name Attribute

The name attribute is used to create a named anchor. When using named anchors we can create links that can jump directly into a specific section on a page, instead of letting the user scroll around to find what he/she is looking for,

Below is the syntax of a named anchor: ` Text to be displayed `

The name attribute is used to create a named anchor. The name of the anchor can be any text you care to use.

You should notice that a named anchor is not displayed in a special way.

To link directly to the “tips” section, add a # sign and the name of the anchor to the end of a URL. Like this: ``

Jump to the Useful Tips Section

A hyperlink to the Useful Tips Section from WITHIN the file “html_links.asp” will look like this: ` Jump to the Useful Tips Section `

Basic Notes – Useful Tips

Always add a trailing slash to subfolder references. If you link like this: `href="http://www.rgcsn.org/html"`, you will generate two HTTP requests to the server, because the server will add a slash to the address and create a new request like this: `href="http://www.rgcsn.org/html/"`,

Named anchors are often used to create “table of contents” at the beginning of a large document. Each chapter within the document is given a named anchor, and links to each of these anchors are put at the top of the document.

If a browser cannot find a named anchor that has been specified, it goes to the top of the document. No error occurs.

Chapter

4

HTML Frames, Tables and List

HTML Frames

With frames, you can display more than one Web page in the same browser window. The intelligent use of frames can give your pages a cleaner look and make them easier to navigate. Dividing a page into frames is actually quite simple. The basic concept goes like this: Each frame is a regular, complete html document. If you wanted to divide your page into 2 side by side frames, then you would put one complete html document in the left frame and another complete html document in the right frame, in addition you need to write a third html document. This MASTER PAGE (I'm coining a new phrase here folks) contains the <FRAMS> tags that specify what goes where. As a matter of fact, that's its only function.

Frames

With frames, you can display more than one HTML document in the same browser window. Each HTML document is called a frame, and each frame is independent of the others.

The disadvantages of using frames are:

- The web developer must keep track of more HTML documents.
- It is difficult to print the entire page.

The Frameset Tag

- The <frameset> tag defines how to divide the window into frames
- Each frameset defines a set of rows or columns
- The values of the rows/columns indicate the amount of screen area each row/column will occupy.

The Frametag

The <frame> tag defines what HTML document to put into each frame

In the example below we have a frameset with two columns. The first column is set to 25% of the width of the browser window. The second column is set to 75% of the width of the browser window. The HTML document “frame_a.htm” is put into the first column, and the HTML document “frame_b.htm” is put into the second column:

```
<frameset cols="25%,75%">
<frame src="frame_a.htm">
<frame src="frame_b.htm">
</frameset>
```

Note: The frameset column size value can also be set in pixels (cols="200,500"), and one of the columns can be set to use the remaining space (cols="25%,*")

Basic Notes – Useful Tips

If a frame has visible borders, the user can resize it by dragging the border. To prevent a user from doing this, you can add noresize="noresize" to the <frame> tag.

Add the <noframes> tag for browsers that do not support frames.

Tag	Description
<frameset>	Defines a set frames
<frame>	Defines a sub window (a frame)
<noframes>	Defines a noframe section for browsers that do not handle frames
<iframe>	Defines an inline sub window (frame)

Important: You cannot use the <body></body> tags together with the <frameset></frameset> tags! However, if you add a <noframes> tag containing some text for browsers that do not support frames, you will have to enclose the text in <body></body> tags! See how it is done in the first example below.

Frame Tags

First things first. For this tutorial we will need a few html documents. Fire up Notepad and copy the following to get you started.

```
<HTML>
<HEAD>
<TITLE>My Frame Page</TITLE>
</HEAD>
```

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

We will give each document a name. In Table Tutor and Form Tutor we used guy's names. I think its time to get some young ladies in here.

```
<HTML>
<HEAD>
<TITLE>My Frame Page </TITLE>
</HEAD>
<BODY>
Lisa
</BODY>
</HTML>
```

Create a new folder somewhere and save this as lisa.html.

Now make another html document.

```
<HTML>
<HEAD>
<TITLE> My Frame Page</TITLE>
</HEAD>
<BODY>
Terri
</BODY>
</HTML>
```

Save this in the same folder as terri.html.

Now do the same for Kim, Tina, Shannon, and Beth. Save them just like the others. You should now have a folder that contains 6 complete standalone html documents.

Ok, now the fun stuff...making your master page. Start with this.

```
<HTML>
<HEAD>
<TITLE> My Frame Page-The Master Page</TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Remove the </BODY> tags. The master page doesn't use them.....

```
<HTML>
<HEAD>
<TITLE>My Frame Page-The Master Page</TITLE>
</HEAD>
</HTML>
```

.....it uses <FRAMESET> tags instead.

```
<HTML>
<HEAD>

<TITLE>My Frame Page-The Master Page</TITLE>
</HEAD>

<FRAMESET>
</FRAMESET>
</HTML>
```

To keep things a little cleaner, I'm going to stop writing the <HTML>, <HEAD> and <TITLE> tags. Needless to say, keep them in your document.

```
<FRAMESET>
</FRAMESET>
```

Now's a good time to save it. Save it in your folder (with all the girls) as index.html. If you try to open it with your browser now it will be blank. All you have said so far is "This is my Master Page".

Now let's start defining just how things are going to look. Tell the browser to split the main window into 2 columns, each occupying 50% of the window.

```
<FRAMESET COLS="50%, 50%">
</FRAMESET>
```

This will still be blank, we have one more thing to do before our 'system is operational'.

We must tell the browser what to put in each frame.

```
<FRAMESETS COLS="50%,50%">
<FRAME SRC="lisa.html">
<FRAME SRC="terri.html">
</FRAMESET>
```

You are now the proud parent of a fully functional Framed page!

One last thing before we go on. Note that <FRAMESET> is a container tag, and <FRAME> is not. For those that don't know what that means, a container tag has an opening <TAG> and a closing </TAG>.

With the form tag you can add to your web pages a guestbook, order forms, surveys, get feedback or whatever.

The basic construction of a html form is this.....

<FORM> begin a form

<INPUT> ask for information in one of several different ways...

<INPUT> ...there can be as many input areas as you wish

</FORM> end a form

That's html forms in a nutshell. You are now ready to make some forms! Now's a good time to stress that if you want to learn how to make quality html documents, then you would be well served to take the time to teach yourself the tags. If you rely on the so-called "from wizards" in the "easy as pie html editors" out there, you will have greatly limited flexibility, and the end result may not be what you are trying to achieve. In my opinion the best html editors to use are text based editors. A few good ones that come to mind are listed here. These editors will make your html coding easier. They don't attempt to do it for you.

Just like "follow the bouncing ball", I want you to open up Notepad (Yes, Notepad!) and follow me. Copy and paste off this page the following to get you started:

```
<HTML>
<HEAD>
<TITLE> Joe's the handsomest guy I know </TITLE>
</HEAD>
</BODY>
<FORM>
</FROM>
</BODY>
</HTML>
```

Next we must tell the browser where to send the data we gather and how to send it. There are two basic things you can do. 1) You can send the data to a CGI script for processing. The parsed data is then emailed to you. This is the preferred method, or 2) You can have the data emailed directly to you using the mailto action. This is simpler but less reliable.

For the sake of simplicity, we will start with the second method.

```
<HTML>
```

```

<HEAD>
<TITLE> Joe's the handsomest guy I know </TITLE>
</HEAD>
<BODY>
<FROM      METHOD=      POST      ACTION="mailto:      xxx@xxx.xxx"
ENCTYPE="text/plain">
</FROM>
</BODY>
</HTML>

```

The only thing you have to do is plug in your email address after mailto:

To send the data to more than one email address, separate them with commas...
larry@3stooges.org,curly@3stooges.org,mo@3stooges.org

Now, I said that this method is simple, but not entirely reliable. For the time being, it is perfectly OK to use this method as you work through the tutorial. As you complete the lessons you can practice sending yourself information. For most people, under most circumstances, everything will be just fine. If you'd like, you can skip the rest of this lesson for now and dive right into making forms. After you learn a little about forms (or if you're having trouble submitting them), definitely come back here and learn more about form processing.

Every so often, depending on the sender's browser/email configuration, using the mailto action might just bring up an empty New Mail window instead of sending the form data. Sometimes (rarely) you'll click on the button and nothing will happen. In this case you will not be able to even practice with the simpler mailto action. You will have to straight into using a CGI form handler to practice with.

Before we into that, there is one more thing about the mailto action that I want to mention. Sometimes, depending on the person sending the data, you will receive the data in a very strange format. Instead of something nice & neat like this.....

```

FORMNAME=New Entrant
NAME=R.U. Havinfun
ADDRESS=13`3 Mockingbird Lane
CITY=Beverly Hills
STATE=CA
The info comes back looking like this.....
FROMNAME=New+Entrant&NAME=R.U. +Havinfun&ADDRESS=1313
+Mockingbird+Lane&CTTY+Beverly+Hills&STATE=CA

```

HTML Tables

With HTML you can create tables.

Tables

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

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
```

How it looks in a browser:

row 1, cell 1	row 1, cell 2
row 1, cell 2	row 2, cell 2

Tables and the Border Attribute

If you do not specify a border attribute the table will be displayed without any border sometimes this can be useful, but most of the time, you want the borders to show.

To display a table with borders, you will have to use the border attribute :< table border="1">

```
<TR>
<td>Row 1, cell 1</td>
<td>Row 2, cell 2</td>
</tr>
</table>
```

Headings in a table

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

```
<table border="1">
```

```

<tr>
<th>Heading</th>
<th>Another Heading</th>
</tr>
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>

```

How it looks in a browser

Heading	Another Heading
row 1, cell 1	row 1, cell 1
row 1, cell 2	row 1, cell 2

Empty Cells in a Table

Table cells with no content are not displayed very well in most browsers. `<table border="1">`

```

<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td></td>
</tr>
</table>

```

How it looks in a browser:

row 1, cell 1	row 1, cell 2
row 2, cell 1	

Note that the borders around the empty table cell are missing (NB! Mozilla Firefox displays the border).

To avoid this, add a non-breaking space () to empty data cells, to make the borders visible:

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td> row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td> &nbsp; </td>
</tr>
</table>
```

How it looks in a browser:

row 1, cell 1	row 1, cell 2
row 2, cell 1	

Basic Notes – Useful Tips

The <thead>, <tbody> and <tfoot> elements are seldom used, because of bad browser support. Expect this to change in future versions of XHTML.

Table Tags

Tag	description
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell
<caption>	Defines a table caption
<colgroup>	Defines groups of table column
<col>	Defines the attribute values for one or more columns in a table
<thead>	Defines a table head
<tbody>	Defines a table body

<code><tfoot></code>	Defines a table footer
----------------------------	------------------------

HTML Lists

HTML supports ordered, unordered and definition lists.

Unordered Lists

An unordered list is a list of items. The list items are marked with bullets (typically small black circles).

An unordered list starts with the `` tag. Each list item starts with the `` tag.

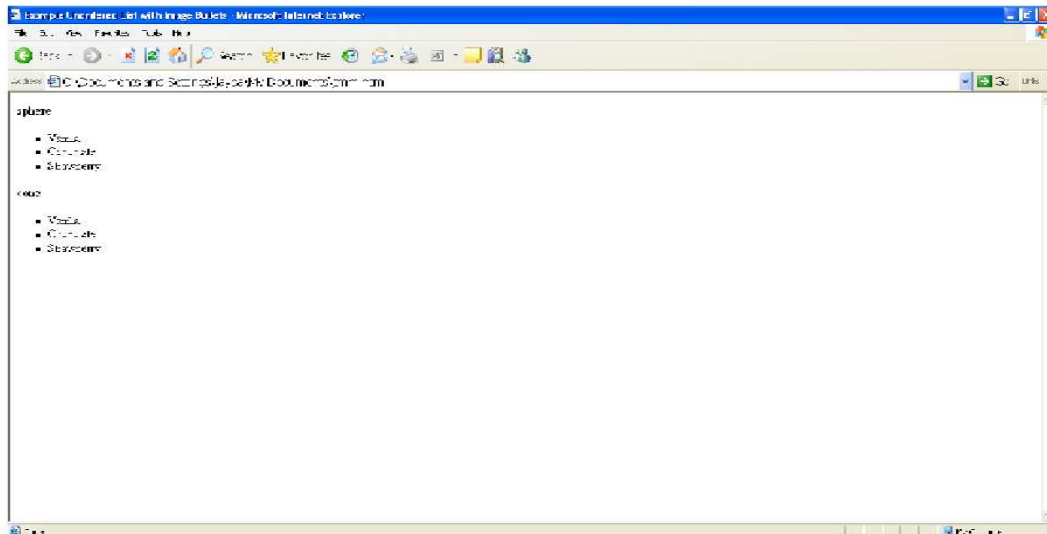
```
<ul>
<li>Coffee</li>
<li>Milk</li>
</ul>
```

Here is how it looks in a browser:

- Coffee
- Milk

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

```
<html>
<head>
<title>Example Unordered List with Image Bullets</title>
</head>
<body>
<p> <b>sphere</b></p>
<ul style="list-style-image: url(sphere.jpg);">
<li>Vanilla</li>
<li>Chocolate</li>
<li>Strawberry</li>
</ul>
<p> <b>cone</b></p>
<ul style="list-style-image: url(cone.jpg);">
<li>Vanilla</li>
<li>Chocolate</li>
<li>Strawberry</li>
</ul>
</body>
</html>
```



Ordered Lists

An ordered list is also a list of items. The list items are marked with numbers.

An ordered list starts with the `` tag. Each list item starts with the `` tag.

```
<ol>
<li>Coffee</li>
<li>Milk</li>
</ol>
```

Here is how it looks in a browser:

1. Coffee
2. Milk

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

Definition Lists

A definition list is not a list of items. This is a list of terms and explanation of the terms.

A definition list starts with the `<dl>` tag. Each definition- list term starts with the `<dt>` tag. Each definition-list definition starts with the `<dd>` tag.

```
<dl>
```

```

<dt>Coffee</dt>
<dd>Black hot drink</dd>
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>

```

Here is how it looks in a browser:

```

Coffee
    Black hot drink
Milk
    White cold drink

```

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

List Tags

Tag	Description
	Defines an ordered list
	Defines an unordered list
	Defines a list item
<dl>	Defines a definition list
<dt>	Defines a definition term
<dd>	Defines a definition description
<dir>	Deprecated. Use instead
<menu>	Deprecated. Use instead

Examples of Lists:

```

<HTML>
  <HEAD><TITLE>Lists</TITLE><HEAD>
  <BODY>
    // <B>Example on Unordered List</B> //
    <UL TYPE="FILLROUND">
      <LI>Sportstar
      <LI>Business Week
      <LI>Time
    </UL>
    // <B>Example on Ordered List</B> //
    <OL TYPE="I" START=4>

```

```

    <LI>Sportstar
    <LI>Business Week
    <LI>Time
</OL>

```

```

// <B>Example on Definition List</B> //
<DL>
<DT>Sports Magazine<DD>Sportstar
<DT>Business Magazine<DD>Business Week
<DT>General Magazine<DD>Time
</DL>
</BODY>
</HTML>

```

Nested Lists

You can nest lists of the same or different types. For example, suppose you have a bulleted list and need a numbered list beneath one of the items, as shown:

? Send us a letter detailing the problem. But sure to include the following:

1. Your name
2. Your order number
3. Your contact information
4. A detailed description of the problem

In such a case, you would nest an ordered list inside an unordered one, as shown in the following code:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"

```

```

http://www.w3.org/TR/html4/strict.dtd>

```

```

<html>

```

```

<head>

```

```

<title>Example Definition List </title>

```

```

</head>

```

```

<body>

```

```

<ul style="list-style: disc ;">

```

```

<li>Send us a letter detailing the problem. Be sure to

```

```

Include the following :< /li>

```

```

<ol style="list-style: decimal ;"> <li>Your name.</li>

```

```

<li>Your order number. </li>

```

```

<li>Your contact information. </li>

```

```

<li>A detailed description of the problem.</li>
</ol>
</ul>
</body>
</html>

```

The output of the code is shown in Figure

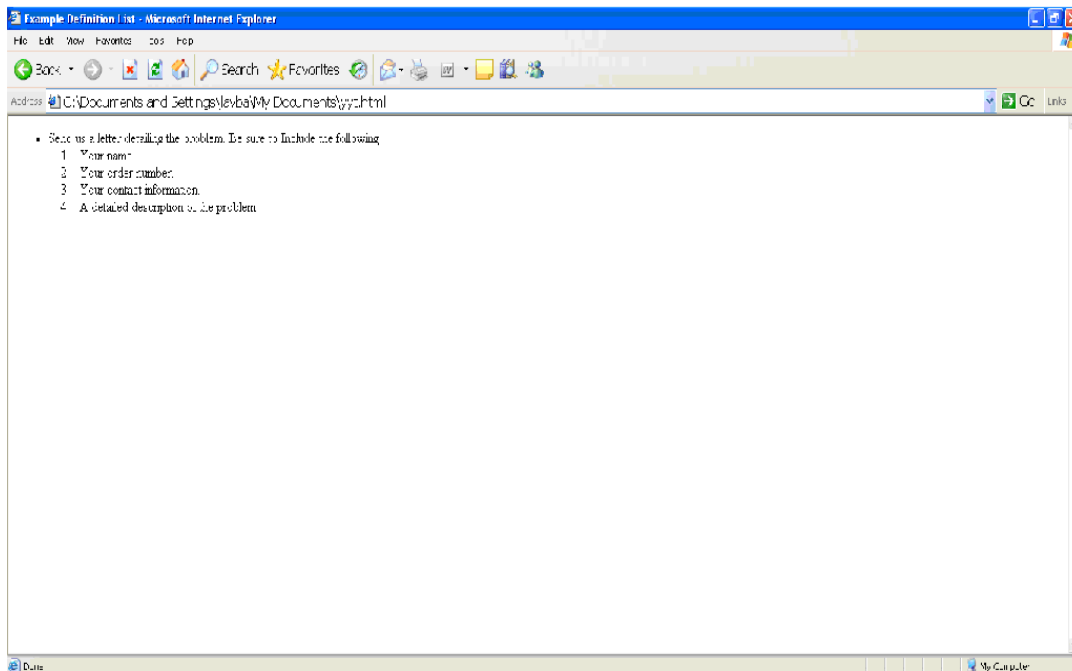


Figure: you can nest different types of lists within one another.

Note that the nested list does not span any open or close tags-it starts after the close tag of the parent's item and before any other tags in the parent list. It is also formatted (indented) to make it easier to identify in the code. Using this method, you can nest any list within any other list.

Changing the Start Value of Ordered Lists

Previous versions of HTML allowed the use of the start attribute in the `` tag to control what number or letter the list began with. For example, the following code starts a list with the decimal number 12:

```
<ol start="12" style="list-style: decimal ;">
```

However, the start attribute of the `` tag was deprecated, and a replacement CSS style has yet to be defined. Although you can use the start attribute, your document will no longer validate against strict HTML.

If you find yourself needing consistent, yet flexible numbering, consider using the new CSS2 automatic counters and numbering feature. This feature uses the content property along with the new counter-increment and counter-reset properties to provide a flexible yet powerful automatic counter function.

The following style code will define a counter and cause any list to begin with an item number of 12:

```
<style type="text/css">
ol {counter-reset: list 11 ;}
li {list-style-type: none ;}
li: before {
Content: counter (list, decimal)". ";
counter-increment: list ;}
</style>
```

This code introduces quite a few CSS2 concepts-pseudo-elements, counters, and relates properties and methods. However, it isn't as complex as it might first appear:?

- The ol definition causes the counter (list) to be reset to 11 every time the tag is used-that is, at the beginning of every ordered list?
- The li definition sets the list style type to none-the counter will display our number, if we left the list style type set to decimal, there would be an additional number with each item.?
- The li: before definition does two things: 1) causes the counter to be displayed before the item (using the begin pseudo-element and the content property) along with a period and a space; 2) increments the counter. Note that the counter increment happens first, before the display. That is the reason you need to reset the counter to one lower than your desired start.

Using the preceding styles along with the following list code in a document results in a list with items numbered 12-15:

```
<ol>
<li>item 12</li>
<li>item 12</li>
<li>item 12</li>
<li>item 12</li>
</ol>
```

Counters are a new, powerful feature of CSS2. Unfortunately, at the time of this writing, only the Opera browser fully supports counters. However, the other browsers are sure to follow suit. You'll find more information on counters and the content property in chapter 16. Tip

The various list properties can all be defined without one property, list-style. The list-style property has the following syntax:

```
list-style: <list-style-type> <list-style-image>  
<list-style-position>
```

You can use this one property to specify one, two, or all three list-style properties in one declaration. For example, to define an ordered list with lowercase letters and inside positioning, you could use the following tag:

```
<ol style="list-style: lower-alpha inside ;">  
See Chapters 16 and 17 for more information on styles.
```



Chapter 5

HTML Forms and Input

HTML Forms are used to select different kinds of user input.

Forms

A form is an area that can contain form elements.

Form elements are elements that allow the user to enter information (like text fields, text area fields, drop-down menus, radio buttons, checkboxes, etc.) in a form.

A form is defined with the `<form>` tag.

```
<form>  
<input>  
<input>  
</form>
```

Input

The most used form tag is the `<input>` tag. The type of input is specified with the type attribute. The most commonly used input typed is explained below.

Text Fields

Text fields are used when you want the user to type letters, numbers, etc. in a form.

```
<form>  
First name:  
<input type="text" name="firstname">  
<br>  
Last name:]  
<input type="text" name="lastname">  
</form>
```

How it looks in a browser:

First name:

Last name:

Note that the form itself is not visible.

Also note that in most browsers, the width of the text field is 20 characters by default.

Radio Buttons

Radio Buttons are used when you want the user to select one of a limited number of choices.

```
<form>
<input type="radio" name="sex" value="male"> Male
<br>
<input type="radio" name="sex" value="female"> Female
</form>
```

How it looks in a browser:

☐ **Male**

☐ **Female**

Note that only one option can be chosen.

Checkboxes

Checkboxes are used when you want the user to select one or more options of a limited number of choices.

```
<form>
I have a bike:
<input type="checkbox" name="vehicle" value="Bike">
<br>
I have a car:
<input type="checkbox" name="vehicle" value="Car">
<br>
I have an airplane:
<input type="checkbox" name="vehicle" value="Airplane">
</form>
```

How it looks in a browser:

I have a bike:

☐

I have a car:
I have an airplane:

The form's Action Attribute and the Submit Button

When the user clicks on the “Submit” button, the content of the form is sent to another file. The form's action attribute defines the name of the file to send the content to. The file defined in the action attribute usually does something with the received input.

```
<form name="input" action="html_form_action.asp"
method="get">
username:
<input type="text" name="user">
<input type="submit" value="submit">
</form>
```

How it looks in a browser

Username:

If you type some characters in the text field above, and click the “submit” button, you will send your input to a page called “html_form_action.asp”. The page will show you the received input .

Form Tags

Tag	Description
<form>	Defines a form for user input
<input>	Defines an input field
<textarea>	Defines a text-area (a multi Line text input control)
<label>	Defines a label to a control
<fieldset>	Defines a fieldset
<legend>	Defines a caption for a field Set
<select>	Defines a selectable list (a

	Drop-down box)
<optgroup>	Defines an option group
<option>	Defines an option in the drop-box
<button>	Defines a push button
<isindex>	Deprecated. Use <input> instead



Chapter 6

HTML Images and Colors

HTML Images

With HTML you can display images in a document.

The Image Tag and the Src Attribute

In HTML, images are defined with the `` tag.

The `` tag is empty, which means that it contains attributes only and it has no closing tag.

To display an image on a page, you need to use `src` attribute. `src` stands for “source”. The value of the `src` attribute is the URL of the image you want to display on your page.

The syntax of defining an image:

``

The URL points to the location where the image is stored. An image named “boat.gif” located in the directory “images” on www.rgcsn.org has the URL :`http://www.rgcsn.org/images/boat.gif`.

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

Example:

`<HTML>`

`<HEAD><TITLE> Working with Images </TITLE></HEAD><BODY`

`Background="images/textural.gif ">`

`Controlling Image Borders! <CENTER>`

**`<I>Image without a BORDER </I>

`**

**`

`**

**`<I> Image With BORDER =3</I>

`**

```
<IMG Border=3 src="images/corp./gif"><BR>
</CENTER>
</BODY>
</HTML>
```

The Alt Attribute

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

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

Example:

```
<HTML>
<HEAD><TITLE> Working with Images </TITLE></HEAD>
<BODY Background="images/textural.gif">
  <B> Use of ALT attribute</B><BR><CENTER>
  <I>Available Image: javacup.gif</I><BR><BR>
  <IMG Src="images/javacup.gif"><BR><BR>
  <I>Unavailable Image: javac.gif - Without the ALT Attribute
</I><BR><BR>
  <IMG>Src="images/javac.gif"><BR><BR>
  <I>Unavailable Image: javac.gif – With the ALT Attribute set to
“Java”</I><BR><BR>
  <IMG Alt="The Java Cup" Src="images/javac.gif"><BR>
</CENTER>
</BODY>
</HTML>
```

Basic Notes – Useful Tips

If an HTML file contains ten images – eleven files are required to display the page right. Loading images take time, so my best advice is: Use images carefully.

Image Tags

Tag	Description.
	Defines an image
<map>	Defines an image map
<area>	Defines a clickable area inside an image map

Example: Controlling image size:

```
<HTML>
<HEAD><TITLE> Working with Images </TITLE> </HEAD>
<BODY Background = "images/textural.gif">
<B>Controlling Image Sizes! </B><CENTER>
<I>Normal Image Size</I><BR><BR>
<IMG Src="images/computer.gif"><BR>
<I>Image With Size (Height And Width) Set To 200</I><BR><BR>
<IMG Height = 200 Src="images/computer.gif Width=200"><BR>
</CENTER>
</BODY>
</HTML>
```

Example:

Setting image alignment

```
<HTML>
<HEAD><TITLE>Working with Image </TITLE> </HEAD>
<BODY Background="images/textural.gif">
<B><I> Image Aligned Left </I></B>
<IMG Align=Left Src="images/sctonly2.gif"><BR><BR> Silicon Chip
Technologies, <BR>
```

```
<I> We Specialize In Corporate Training, Software Development and Placements. Our
Training Program and Software Development include Web Based Commercial Applications and
Commercial Application Development using Database. </I> <BR> <BR> <BR> <BR>
```

```
<B><I> Image Aligned Right </I></B>
<IMG Align=Right Src="images/sctonly2.gif"><BR><BR> Silicon
Chip Technologies, <BR>
```

<I> We Specialize In Corporate Training, Software Development And Placements. Our Training Program and Software Development include Web Based Commercial Applications and Commercial Application Development using Database. </I>

</BODY>
</HTML>

HTML Background

A good background can make a Web site look really great.

Backgrounds

The <body>tag has two attributes where you can specify background. The background can be a color or an image.

Bgcolor

<HTML>

<HEAD><TITLE>Working with Image </TITLE> </HEAD>

<BODY Background="images/textural.gif">

<I> Image Aligned Left </I>

 Silicon Chip Technologies,

<I> We Specialize In Corporate Training, Software Development And Placements. Our Training Program and Software Development include Web Based Commercial Applications and Commercial Application Development using Database. </I>

<I> Image Aligned Right </I>

 Silicon Chip Technologies,

<I> We Specialize In Corporate Training, Software Development And Placements. Our Training Program and Software Development include Web Based Commercial Applications and Commercial Application Development using Database. </I>

</BODY>

</HTML>The value of this attribute can be a hexadecimal number, an RGB value, or a color name:

```
<body bgcolor="#000000">  
<body bgcolor="rgb(0.0.0)">  
<body bgcolor="black">
```

The lines above all set the background-color to black.

Background

The background attribute specifies a background-image for an HTML page. The value of this attribute is the URL of the image you want to use. If the image is smaller than the browser window, the image will repeat until it fills the entire browser window.

```
<body background="clouds.gif">  
<body background=http://www.rgcsn.org/clouds.gif>
```

The URL can be relative (as in the first line above) or absolute (as in the second line above).

Note: If you want to use a background image, you should keep in mind:

- Will the background image increase the loading time too much?
- Will the background image look good with other images on the page?
- Will the background image look good with the text colors on the page?
- Will the background image look good when it is repeated on the page?
- Will the background image take away the focus from the text?

Basic Notes – Useful Tips

The bgcolor, background, and the text attributes in the <body> tag are deprecated in the latest versions of HTML (HTML 4 and XHTML). The World Wide Web consortium (w3c) has removed these attributes from its recommendations.

Style sheets (CSS) should be used instead (to define the layout and display properties of HTML elements).

HTML Colors

Colors are displayed combining RED, GREEN, and BLUE light sources.

Color Values

HTML colors can be defined as a hexadecimal notation for the combination of Red, Green, and Blue color values (RGB)

The lowest value that can be given to one light source is 0 (hex#00) and the highest value is 255 (hex#FF).

The table below shows the result of combining Red, Green, and Blue light sources

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

W3C

Standard Color Names

W3C has listed 16 color names that will validate with an HTML valuator.

The color names are: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow.

Cross-browser Color Values

Some years ago, when most computers only supported 256 different colors, a list of 216 Web Safe Colors was suggested as a Web standard. The reason for this was that the Microsoft and Mac operating system used 40 different “reserved” fixed system colors (about 20 each).

We are not sure how important this is now, since most computers today have the ability to display millions of different colors, but the choice is left to you.

HTML Color Values

Colors are displayed combining RED, GREEN, and BLUE light sources.

Color Values

HTML colors are defined using a hexadecimal notation for the combination of RED, GREEN, and BLUE color values (RGB). The lowest value that can be given to one of the light sources is 0 (hex#00). The highest value is 255 (hex#FF).

Turn Off the Red

If you turn off the Red light completely, there are 65536 different combinations of Green and Blue (256×256) to experiment with.

[Click here to see some of these combinations of Green and Blue.](#)

Turn On the Red

By setting the Red light completely, there are 65536 different combinations of Green and Blue (256×256) to experiment with.

[Click here to see some of these combinations of Green and Blue.](#)

16 Million Different Colors

The combination of Red, Green, and Blue values from 0 to 255 gives a total of more than 16 million different colors to play with ($256 \times 256 \times 256$).

Most modern monitors are capable of displaying at least 16384 different colors.

Chapter 7

HTML Layout fonts and Style

HTML Layout

Everywhere on the Web you will find pages that are formatted like newspaper pages using HTML columns.

HTML Layout – Using Tables

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

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

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

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

Example: Creating Table

```
<HTML>
<HEAD><TITLE>Table Attributes </TITLE> </HEAD>
<BODY BgColor=LIGHTGREY>
  <B>Specifying the BORDER and WIDTH of the Table!
</B><BR><BR><BR><BR>
  <CENTER><TABLE Borer=5 Width=50%>
    <CAPTION Align=Bottom><B>Personal Information </B> </CAPTION>
    <TR><TH>NAME</TH><TH>AGE</TH></TR>
    <TR Align=CENTER><TD>Shilpa</TD><TD>21</TD> </TR>
    <TR Align=CENTER><TD>Vaishali</TD><TD>22</TD> </TR>
  </TABLE><.CENTER>

  </BODY>
</HTML>
```

HTML Fonts

The `` tag in HTML is deprecated. It is supposed to be removed in a future version of HTML.

Even if a lot of people are using it, you should try to avoid it, and use styles instead.

The HTML `` Tag

With HTML code like this, you can specify both the size and the type of the browser output

```
<p>
<font size="2"face="Verdana">
This is a paragraph.
</font>
</p>
<p>
<font size="3"face="Times">
This is another paragraph.
</font>
</p>
```

Font Attributes

Attribute	Example	Purpose
size="number"	size="2"	Defines the font size
size="+number"	size="+1"	Increases the font size
size="-number"	size="-1"	Decreases the font size
face="face-name"	face="Times"	Defines the font-name
color="color-value"	color="#eeff00"	Defines the font color
color="color-name"	color="red"	Defines the font color

The `` Tag Should NOT be Used

The `` tag is deprecated in the latest versions of HTML (HTML 4 and XHTML).

The World Wide Web Consortium (w3c) has removed the tag from its recommendations. In future versions of HTML, style sheets (CSS) will be used to define the layout and display properties of HTML elements.

HTML Styles

With HTML 4.0 all formatting can be moved out of the HTML document and into a separate style sheet.

How to Use Styles

When a browser reads a style sheet, it will format the document according to it. There are three ways of inserting a style sheet:

External Style Sheet:

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look to an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section.

```
<head>
<link rel="stylesheet" type="text/css"
Href="mystyle.css">
</head>
```

Internal Style Sheet

An internal style sheet should be used when a single document has a unique style. You define internal style sheet in the head section with the <style> tag.

Inline Style

An internal style sheet should be used when a unique style is to applied to a single occurrence of an element.

To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a paragraph:


```
<p style="color: red; margin-left: 20px">  
This is a paragraph  
</p>
```

Style Tags

Tag	Description
<style>	Defines a style definition
<link>	Defines a resource reference
<div>	Defines a section in a document
	Defines a section in a document
	Deprecated. Use styles instead
<basefont>	Deprecated. Use styles instead
<center>	Deprecated. Use styles instead

Chapter 8

HTML Advanced

HTML Uniform Resource Locators

HTML Links

When you click on a link in an HTML document like this: Last Page, an underlying `<a>` tag points to a place (an address) on the Web with an href attribute value like this: `Last page`.

The Last Page link in the example is a link that is relative to the Web site that you are browsing, and your browser will construct a full Web address like `http://www.rgcsn.org/html/lastpage.htm` to access the page.

Creating an Email Link

Building an attractive, useful Web page is only part of the job. You also need to give viewers a way to contact you.

In the digital age, Web users expect to have instant and easy email contact with you. To make a link to your email address, you'll need to use the anchor tag again. You can separate your email contact with the `<HR>` tag, which inserts a horizontal line. We've decided to have the email contact link centered on the page. Put it below the list, like this:

```
<CENTER>
<HR>
<a href="mailto:comments@builder.com">Drop us a line! </a>
</CENTER>
```

Uniform Resource Locators

Something called a Uniform Resource Locator (URL) is used to address a document (or other data) on the World Wide Web. A full Web address like this: <http://www.rgcsn.org/jtml/lastpage.htm> follows these syntax rules:

scheme://host.domain:port/path/filename

The scheme is defining the type of Internet service. The most common type is http.

The domain is defining the Internet domain name like rgcsn.org.

The host is defining the domain host. If omitted, the default host for http is www.

The: port is defining the port number at the host. The port number is normally omitted. The default port number for http is 80.

The path is defining a path (a sub directory) at the server. If the path is normally omitted, the resource (the document) must be located at the root directory of the Web site.

The filename is defining the name of a document. The default filename might be default.asp, or index.html or something else depending on the settings of the Web server.

URL Schemes

Some examples of the most common schemes can be found below:

Schemes – Access

file – a file on your local PC

ftp – a file on an FTP server

http – a file on a World Wide Web server

gopher – a file on a Gopher server

news – a Usenet newsgroup

telnet – a Telnet connection

WAIS – a file on a WAIS server

Accessing a Newsgroup

The following HTML code:

```
<a href=news:alt.html>HTML Newsgroup</a>
```

Creates a link to a newsgroup like this HTML Newsgroup.

Downloading with FTP

The following HTML code:

`Download WinZip`

Creates a link to download a file like this: Download WinZip.

(The link doesn't work. Don't try it. It is just an example. W3C Schools doesn't really have an ftp directory.)

Link to your Mail system

The following HTML code:

`someone@rgcsn. Com`

Creates a link to your own mail system like this:

Someone@rgcsn.com

HTML Scripts

Add scripts to HTML pages to make them more dynamic and interactive.

Insert a Script into HTML Page

A script in HTML is defined with the `<script>` tag. Note that you will have to use the type attribute to specify the scripting language. `<html>`

```
<head>
</head>
<body>
<script type="text/javascript">
document. write ("Hello World!")
</script>
</body>
</html>
```

The script above will produce this output:

Hello World!

How to Handle Older Browsers

A browser that does not recognize the `<script>` tag at all, will display the `<script>` tag's content as text on the page. To prevent the browser from doing this, you should hide the script in comment tags. An old browser (that does not recognize the `<script>` tag) will ignore the comment and it will not write the tag's content on the page, while a new browser will understand that the script should be executed, even if it is surrounded by comment tags.

Example

JavaScript:

```
<script type="text/javascript">
<!--
document.write ("Hello World!")
//-- >
</script>
```

VBScript:

```
<script type="text/vbscript">
<!--
document.write("Hello World!")
'-- >
</script>
```

The <noscript> Tag

In addition to hiding the script inside a comment, you can also add a <noscript> tag.

The <noscript> tag is used to define an alternate text if a script is NOT executed. This tag is used for browsers that recognize the <script> tag, but do not support the script inside, so these browsers will display the text inside the <noscript> tag instead. However, if a browser supports the script inside the <script> tag it will ignore the <noscript> tag.

Example

JavaScript:

```
<script type="text/javascript">
<!--
document.write("Hello World!")
//-- >
</script>
<noscript>Your browser does not support Java Script! </noscript>
```

VBScript:

```

<script type="text/vbscript">
<!--
document.write("Hello World!")
'-- >
</script>
<noscript>Your browser does not support VBScript! </noscript>

```

Script Tags

Tag	Description
<script>	Defines a script
<noscript>	Defines an alternate text if the script is not executed
<object>	Defines an embedded object
<param>	Defines run-time settings (parameters) for an object
<applet>	Deprecated. Use <object> instead

The Meta Element

As we explained in the previous chapter, the head element contains general information (meta-information) about a document.

HTML also includes a meta element that goes inside the head element. The purpose of the meta element is to provide meta-information about the document.

Most often the meta element is used to provide information that is relevant to browsers or search engines like describing the content of your document.

Note: W3C states that “Some users agents support the use of META to refresh the current page after a specified number of seconds, with the option of replacing it by a different URL. Authors should not use this technique to forward users to different pages, as this makes the page inaccessible to some users. Instead, automatic page forwarding should be done using server-side redirects”

Keywords for Search Engines

Some search engines on the WWW will use the name and content attributes of the meta tag to index your pages.

This meta element defines a description of your page:

<meta name="description" content="Free Web tutorials on HTML, CSS, XML, and XHTML">

This meta element defines keywords for your page:

<meta name="keywords" content="HTML, DHTML, CSS, XML, XHTML, JavaScript, VBScript">

The intention of the name and content attributes is to describe the content of a page.

However, since too many webmasters have used meta tags for spamming, like repeating keywords to give pages a higher ranking, some search engines have stopped using them entirely.

Unknown Meta Attributes

Sometimes you will see meta attributes that are unknown to you like this:

<meta name="security" content="low">

Then you just have to accept that this is something unique to site or to the author of the site, and that it has probably no relevance to you.

HTML 4.0 Standard Attributes

HTML tags can have attributes. The special attributes for each tag are listed under each tag description. The attributes listed here are the core and language attributes that are standard for all tags (with a few exceptions):

Core Attributes

Not valid in base, head, html, meta, param, script, style, and title elements.

Attribute	Value	Description
class	class_rule or style_rule	The class of the element
id	id_name	A unique id for the element

style	style_definition	An inline style definition
title	tooltip_text	A text to display in a tool tip

Language Attributes

Not valid in base, br, frame, frameset, hr, iframe, param, and script elements

Attribute	Value	Description
dir	ltr rtl	Sets the text direction
lang	language_code	Sets the language code

Keyboard Attributes

Attribute	Value	Description
accesskey	character	Sets a keyboard shortcut to access an element
tabindex	number	Sets the tab order of an element

HTML 4.0 Event Attributes

New to HTML 4.0 is the ability to let HTML event trigger actions in the browser, like starting a JavaScript when a user clicks on an HTML element. Below is a list of attributes that can be inserted into HTML tags to define event actions.

Window Events

Only valid in body and frameset elements.

Attribute	value	Description
onload	script	Script to be run when a document loads
onunload	script	Script to be run when a document unloads

Form Element Events

Only valid in form elements.

Attribute	Value	Description
onchange	script	Script to be run when the element changes
onsubmit	script	Script to be run when the form is submitted
onreset	script	Script to be run when the form is reset
onselect	script	Script to be run when the element is selected
onblur	script	Script to be run when the element loses focus
onfocus	script	Script to be run when the element gets focus

Keyboard Events

Not valid in base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style, and title elements.

Attribute Value Description

onkeydown	script	what to do when key is pressed
onkeypress	script	what to do when key is pressed and released
onkeyup	script	what to do when key is released

Mouse Events

Not valid in base, bdo, br, frame, frameset, head, html, iframe, meta, param, script, style, and title elements.

Attribute	Value	Description
onclick	script	What to do on a mouse click
ondblclick	script	What to do on a mouse double-click
onmousedown	script	What to do on a mouse button is pressed
onmousemove	script	What to do on a mouse pointer moves
onmouseout	script	What to do on a mouse pointer moves out of an

		element
onmouseover	script	What to do on a mouse pointer moves over an element
onmouseup	script	What to do on a mouse button is released

Web Publishing

Your First Step: A Personal Web Server

- If you want other people to view your pages, you must publish them.
- To publish your work, you have to copy your files to a web server.
- Your own PC can act as a web server if it is connected to a network.
- If you are running Windows 98, you can use the PWS (personal Web Server).
- PWS is hiding in the PWS folder in your Windows CD.

Personal Web Server (PWS)

PWS turns any Windows computer into a Web server. PWS is easy to install and ideal for developing and testing Web application. PWS has been optimized for working use, but has all the requirements of a full Web server. It also runs Active Server Pages (ASP) just like its larger brother IIS.

How to Install a Personal Web Server (PWS)

Browse your Window installation to see if you have installed PWS. If not, install PWS from the PWS directory on your Window CD. Follow the instructions and get your personal Web Server up and running.

Note: Microsoft Windows XP Home Edition does not come with the option to turn your computer into a PWS! Internet Information Server(IIS)

Windows 2000's built-in Web server IIS, makes it easy to build large applications for the Web. Both PWS and IIS include ASP, a server- side scripting standard that can be used to create dynamic, and interactive Web applications, IIS is also available for Windows NT.

Your Next Step: A Professional Web Server

- If you do not want to use PWS or IIS, you must upload your files to a public server.
- Most Internet Service Providers (ISP's) will offer to host your web pages.

- If your employer has an Internet Server, you can ask him to host your Web site.
- If you are really serious about this, you should install your own Internet Server.



Chapter 9

Introduction to Dynamic HTML?

Web authors today face significant challenges when making their Web pages interactive. The static nature of HTML pages limits their creative choices, and interactive components can be difficult to build and reuse. In addition, using proprietary extensions means authoring browser-specify Web pages.

Microsoft Dynamic HTML technology helps to remove these barriers for content providers and offers users more engaging and interactive Web pages. Dynamic HTML provides authors with enhanced creative control so they can manipulate any page element at any time. Dynamic HTML is also the easiest way to make Web pages interactive, using open, standards-based technologies. Microsoft is working with the World Wide Web Consortium (W3C) to help ensure interoperability and support for users on multiple systems with different browsers.

According to Microsoft DHTML (first implemented in IE 4.0) is a set of innovative features which allows web developers to change the rendering and content of a document on the fly without relying on server-side program (such as Active Server Pages or ColdFusion) or complicated sets of HTML pages to achieve special effects.

DHTML can be described as an interface which makes it possible to access the browser object model through JavaScript or VBScript scripting language.

This way you can control your page more efficiently. And yes, you can actually make a page without HTML tags at all!

DHTML is the art of making HTML pages dynamic!

DHTML is a combination of technologies used to create dynamic and interactive Web sites.

To most people DHTML means a combination of HTML, Style Sheets and JavaScript.

What is DHTML?

“DHTML is the combination of several built-in browser features in fourth generation browsers that enable a web page to be more dynamic”

DHTML is NOT a scripting language (like JavaScript), but merely a browser feature-or enhancement-that gives your browser the ability to be dynamic. What you really want to learn is not DHTML itself, but rather, the syntax needed to use DHTML.

DHTML is a collection features that together, enable your web page be dynamic. I think its important now to define just what is meant by Dynamic?

“Dynamic” is defined as the ability of the browser to alter a web page’s look and style after the document has loaded. I remember when I was learning JavaScript, I was taught that you could use document. Writer () method of JavaScript to create webpages on the fly. For example:

```
<script>
document.write (“This is text created on the fly!”)
</script>
```

But what if we wanted to create content not only on the fly, but on demand? We can do that, by nesting the above code in a function, and calling it via a form button:

```
<input type=”button” onClick=”writeconent()” value=”text”>
```

Pressing the button will erase the entire web page and we were left with only the text the function produced.

That was back then. Now, with the introduction of DHTML, we can alter content on a web page on demand, whenever we feel like it, without having the browser erase everything else. That’s what DHTML is all about. The ability of the browser to change look and style even after the document has loaded.

The technology of DHTML is currently at its development stage, with NE, IE 5 differing quite greatly in their implementation of this great technology. Its currently not possible to write one DHTML code and expect it to function in both browsers properly. Furthermore, the two browsers are at different stages in their development of DHTML; form my own knowledge and what I’ve heard, DHTML in IE is far more powerful and versatile then NS’s.

Dynamic HTML Features

Dynamic HTML gives authors enhanced control of the elements on their pages. When you have applied one of the FrontPage Dynamic HTML commands to text or graphics, Microsoft Internet Explorer 4.0 and other Web browsers that support Dynamic HTML will

animate the text or graphics or apply other effects that you select. Because it does not require fetching information from the web server, Dynamic HTML is very efficient, and presents the user with a lively, compelling page without requiring time-consuming network activity.

The FrontPage Editor includes the following Dynamic HTML features:

- Animation
- Page transitions
- Outlining
- Form enhancements

Key features of Dynamic HTML include these:

- Document Object Model (DOM). Dynamic HTML provides a comprehensive object model for HTML. This model exposes all page elements as objects. These objects can easily be manipulated, by changing their attributes or applying methods to them at any time. Dynamic HTML also provides full support for keyboard and mouse events on all page elements. Support for the Document Object Model enables the following:

- Dynamic content. Text or graphics can be added, deleted, or modified on the fly. For example, a Web page can display an updated headline, without refreshing the page. The text surrounding the headline will reflow automatically.

- Dynamic styles. Internet Explorer 4.0 fully supports cascading Style Sheets (CSS). As such, any CSS attribute, including color and font, can be updated without a server round-trip. For instance, text can change color or size when a mouse pointer passes over it. Multimedia filters and transition effects can be applied to HTML elements simply by adding the filter CSS attribute.

- Absolute positioning. CSS positioning coordinates for existing page content can be updated at any time to create animated effects, without reloading the page.

- Data Binding. Data-driven application front ends can be built that present, manipulate (e.g., sort, filter), and update data on the client without numerous round-trips to the server.

- Scriptlets. A Scriptlets is a Web page, authored with Dynamic HTML, which content providers can use as a component in their Web applications. With Scriptlets, content providers can author content once, then easily reuse the content in other Web pages or applications.

Understanding Interactivity

It is important to define interactivity for the purposes of this guide. The following criteria are suggested for defining a truly interactive Web page:

“The page author must be able to update any element of the page at any time—during and after the loading of the page.

- The page should capture all user actions and enable authors to respond to these events using simple scripts.
- The page should reflect the relationship among page elements. A change to one element in the page may require readjusting other elements. An interactive page should do this automatically.
- If authors so choose, users should be able to freely change any element in the page at any time.

What you should already know

Before you continue you should have a basic understanding of the following:

- HTML
- CSS
- JavaScript

Cascading Style Sheets are design templates that provide augmented control over presentation and layout of HTML elements. They allow you to separate the way you design information from the HTML content.

JavaScript is used to create interactive web application supported by the Netscape browser. JavaScript offers many of the same advantages as VBScript. JavaScript is simple to use, lightweight, and dynamic. Developers can easily code functionality for interactive applications inside a web page.

DHTML is NOT a W3C Standard

DHTML stands for Dynamic HTML.

DHTML is not a standard defined by the World Wide Web Consortium (W3C). DHTML is a “marketing term” – used by Netscape and Microsoft to describe the new technologies the 4.x generation browsers would support.

DHTML is a combination of technologies used to create dynamic Web sites.

To most people DHTML means a combination of HTML 4.0, Style Sheets and JavaScript.

W3C once said: “Dynamic HTML is a term used by some vendors to describe the combination of HTML, style sheets and scripts that allows documents to be animated.”

DHTML Technologies

With DHTML a Web developer can control how to display and position HTML elements in a browser window.

HTML 4.0

With HTML 4.0 all formatting can be moved out of the HTML document and into a separate style sheet. Because HTML 4.0 separates the presentation of the document from its structure, we have total control of presentation layout without messing up the document control.

Cascading Style Sheets (CSS)

CSS was a breakthrough in Web design because it allowed developers to control the style and layout of multiple Web pages all at once. As a Web developer you can define a style for each HTML element and apply it to as many Web pages as you want. To make a global change, simply change the style, and all elements in the web are update automatically. With CSS we have a style and layout model for HTML documents.

The Document Object Model (DOM)

DOM stands for the Document Object Model. The HTML DOM is the Document Object Model for HTML. The HTML DOM defines a standard set of objects for HTML, and a standard way to access and manipulate HTML objects.

“The W3C Document Object Model (DOM) is a platform and language neutral interface that allows programs and scripts to dynamically access and updates the control, structure, and style of a document”.

JavaScript

Allows you to write code to control all HTML elements.

DHTML Technologies in Netscape 4.x and Internet Explorer 4.x

Netscape 4.x

- JSS (JavaScript Style Sheets) (allows you to control how different HTML elements will be displayed)
- Layers (allows you to control element positioning and visibility)

Cross-Browser DHTML

- CSS 1
- CSS 2 (allows you to control how different HTML elements will be displayed)
- CSS Positioning (allows you to control element positioning and visibility)
- JavaScript

Internet Explorer 4.x

- Visual Filters (allow you to apply visual effects to text and graphics)
- Dynamic CSS (allows you to control element positioning and visibility)

Note: Problems with coding DHTML technologies WILL occur as long as each browser creates its own proprietary features and technology that is not supported by other browser. A Web page may look great in one browser and horrible in another

Chapter 10

Introduction to VBScript

What is VBScript?

VBScript (Visual Basic Script) is a scaled down version of Visual Basic. Visual Basic is a full blown programming language. VBScript, on the other hand, is a scripting language that can be run client- side (i.e. via your web browser), or server-side (i.e. on the web server). VBScript is the most common scripting language on websites that use ASP (Active Server Pages).

When VBScript is run on the client side, the user's browser needs to support VBScript. When run on the sever side, the server needs to support it. In this tutorial, our VBScript examples are on the client-side, therefore, in order to get the most out of this tutorial, you should use Internet Explorer, as most other browsers don't support VBScript.

What do I need to create VBScript?

You can create JavaScript using the same equipment you use when creating HTML. That is:

- Computer
- Text editor. For example, Notepad (for Windows), Pico (for Linux), or simpletext (Mac). You could use a VBScript editor if you like but it's not needed. The next lesson will discuss some popular VBScript editors.

- Internet Explorer. Note: VBScript is not supported in most (if not all) other browsers – therefore., it's recommended that you use Internet Explorer.

VBScript editors

You don't necessarily need a special VBScript editor in order to code in VBScript – you could just use a normal text editor. However, a special VBScript editor will probably make your life easier, and enable you to code your scripts much faster.

If you'd prefer not to use a text editor, you can check out these HTML / VBScript editors:

- Text Hawk
- EditPlus
- Adobe Dreamweaver
- HTML Kit (Free)

OK, to be precise, most VBScript editors aren't solely VBScript editors. Often, they will be HTML editors with VBScript support. This generally means that they will include syntax highlighting for VBScript (ability to recognize VBScript and color code/format it for you). It may also have a toolbar with options that are specific to VBScript. Given you will probably be coding HTML at the same time, it makes sense to have an editor with HTML support anyway.

Regardless of whether you have a HTML / VBScript editor or not, you still need to test your pages in a web browser. As mentioned previously, Internet Explorer is your best bet for this when running VBScript on the client side.

OK, so you've got yourself an editor? Let's move on and learn some VBScript!

VBScript Syntax

What VBScript syntax refers to, is a set of rules that determine how the language will be written (by the programmer) and interpreted (by the browser or server).

The VBScript syntax is based on the Visual Basic (VB) syntax. VB is a full blown programming environment and VBScript is a scaled down version of VB.

You may also find the VBScript syntax similar to JavaScript in some ways. If you know JavaScript, or any other language for that matter, you will be familiar with terms such as variables, functions, statements, operators, data types, objects etc. VBScript includes these too.

Most of these will be covered throughout the rest of this tutorial. For now, we'll start with the basics.

Basic VBScript Code

Here's some basic VBScript that outputs some text to the browser. Again, if you know JavaScript, this will look familiar.

```
<script type="text/vbscript">  
document.write("VBScript is similar to JavaScript, but different.")  
</script>
```

This results in:

VBScript is similar to JavaScript, but different.

Explanation of code

- The <script> tags are actually HTML, and tell the browser to expect a script in between them. You specify the language using the type attribute. In this case, we're using VBScript.
- The part that writes the actual text is only 1 line (document.write ("VBScript is similar to JavaScript, but different,")). This is how you write text to a web page in VBScript. This is an example of using a VBScript function (also known as method).

Concatenation

When writing large blocks of text to the screen, you may need to break your code up over many lines. To ensure this doesn't affect the output, you can use an underscore to indicate a concatenation (joining two lines together).

```
<script type="text/vbscript">  
document.write("The VBScript syntax allows you to "&  
"concatenate multiple lines together, so that they "&  
"appear joined together seamlessly")  
</script>  
This results in:
```

The VBScript syntax allows you to concatenate multiple lines together, so that they appear joined together seamlessly

Where to Put Your Scripts?

You can place your scripts in any of the following locations:

- Between the HTML document's head tags.
- Within the HTML document's body (i.e. between the body tags).
- Both of the above.

VBScript Variables

A variable is a fundamental part of any programming language. Variables are best visualized as a container. This is because a variable's purpose is to hold something – a value.

How does the variable get the value? You, as a programmer, assign the value to it. This value could be dynamically produced, depending on your program/script.

Once the variable has a value, you can use subsequent code to check its value, and do something (or not do something) depending on its value.

Some programming languages require that you declare your variable before assigning a value to it. Others (such as VBScript) make this optional. In any case, it is good practice to declare all your variables first.

Declare a VBScript Variable

VBScript variables are declared using the dim statement:

```
<script type="text/vbscript">  
Dim firstName  
Dim age  
</script>
```

These variables could also have been declared on the same line, separated by a comma:

```
<script type="text/vbscript">  
Dim firstName, age  
</script>
```

Option Explicit

As mentioned, it's good practice to declare all variables before using them. Occasionally, if you're in a hurry, you might forget to do this. Then, one day, you might misspell the variable name and all sorts of problems could start occurring with your script.

VBScript has a way of ensuring you declare all your variables. This is done with the Option Explicit statement:

```
<script type="text/vbscript">  
Option Explicit  
Dim firstName  
Dim age  
</script>
```

Assign Values to your Variables

You assign value using an equals operator (equals sign).the variable name goes on the left, the value on the right.

```
<script type="text/vbscript">
```

```
Option Explicit
Dim firstName
Dim age
firstName= "Borat"
age= 25
</script>
```

Display your Variables

You can output the value of a variable by including it within the document.write() method.

```
<script type="text/vbscript">
Option Explicit
Dim firstName
Dim age
firstName= "Borat"
age= 25
document.write ("Firstname: " & firstName & "<br/>")
document.write("Age: " & age)
</script>
Display in Browser:
Firstname: Borat
Age: 25
```

VBScript Arrays

A VBScript array is a special type of variable that allows you to multiple values against a single variable.

For example, say you have shopping list that you want to store and write out to the screen. You could store these in a simple variable like this:

```
Item1 ="Bananas"
Item2 ="Bread"
Item3 ="Pasta"
```

This will work fine. But one problem with this approach is that you have to write out each variable name whenever you need to work with it. Also, you can't do things like, loop through all your variables. If these values were stored in an array, you could save yourself a lot of time.

What Does an Array Look Like?

Arrays can be visualized as a stack of elements. Each element has an index number (left column) and a value (right column).

Note that VBScript arrays start their numbering at zero.

Array	
0	Bananas
1	Bread
2	pasta

Creating Arrays in VBScript

VBScript arrays are created by first assigning an array object to a variable name...

```
Dim arrayName(numberOfElements)
then by assigning values to the array...
ArrayName(0) = "Array element 1"
ArrayName(1) = "Array element 2"
ArrayName(2) = "Array element 3"
So, using our prior example, we could write:
Dim shoppingList(3)
ShoppingList(0) = "Bananas"
ShoppingList(1) = "Bread"
ShoppingList(2) = "Pasta"
```

Accessing Array Data

You can access data in an array by referring to the name of the array and the element index number.

Display a Single Array Element

This example displays the second element of the array named shoppingList (remember that VBScript array index numbers begin at zero). In this case, the value would be Bread

```
document.write(shoppingList(1))
```

So, building on our previous example, you could use the following code to declare an array, assign values, then output the contents to the screen.

```
Dim shoppingList (3)
ShoppingList (0) = "Bananas"
ShoppingList (1) = "Bread"
ShoppingList (2) = "Pasta"
document.write(shoppingList(1))
```

This results in:
Bread

Modify the Contents of an Array

You can modify the contents of an array by specifying a value for a given index number:

```
ShoppingList(1) = "Wholemeal Bread"
```

Now, the value of the second element will be:
Wholemeal Bread

VBScript Date

To display the date using VBScript, you use the VBScript date() function.
Date()

VBScript Date Code

To display this to the user, you need to output it like you would anything else – for example, using the document.write() method.

```
Document.write(date())
```

Or even better, you could put the date into a variable first, then output the variable:

```
LocalDate = date()
document.write(LocalDate)
```

Both these result in:
4/7/2010

Note: If this is blank, your browser probably doesn't support VBScript. Try using Internet Explorer.

VBScript Date Format

You can use the VBScript `FormatDate Time()` method to format the date to either a long date format or a short date format.

The `FormatDate Time()` method accepts two arguments: The date being formatted, and the required format (indicated by a constant). The formats are specified as follows:

- 0 – This represents the default date format (as in the previous example)
- 1 – This represents the long date format (based on the user's regional settings)
- 2 – This represents the short date format (based on the user's regional settings)

Long Format

To use long date format, you pass the `FormatDate Time()` an argument of 1.

```
localDate = FormatDate Time(date(),1)  
document.write(localDate)
```

This results in:

Wednesday, April 01, 2010

Short Format

To use short date format, you pass the `FormatDate Time()` an argument of 2.

```
LocalDate = FormatDate Time (date (), 1)  
document.write(localDate)
```

This results in:

4/7/2010

VBScript If Statements

VBScript If statements are a handy tool for any VBScript programmer. In fact, If statements are a universal to all programming languages (that I'm aware of).

An If Statements allows you to tell your program to do something only if a condition is true.

For example, you might have a website that asks the user what color their hair is. You could write code that only executes if their is say, blank. OK, that's a pretty lame example but you get the idea!

Example If Statement

In this example, we'll stick with the lame subject of hair color, but to keep it simple, we won't ask for user input, we'll just set a variable with a hair color values.

```
<script type="text/vbscript">
Dim hairColor
hairColor = "Black"
If hairColor = "Black" Then
    document.write("Same color as my cat!")
End If
</script>
```

This results in:
Same color as my cat!

If Else Statement

An If Else statement is an extension to the If statement. This allows you to tell the program to do something else in the event that the condition is not true.

```
<script type="text/vbscript">
Dim hairColor
hairColor = "Black"
If hairColor = "Black" Then
    document.write("Same color as my cat!")
Else
    document.write("Oh well, whatever...")
End If
</script>
```

This time the value of hairColor is blue. Because it is not blank, the Else part of the statement is executed.

This results in:
Oh well, whatever...

Elseif Statement

You can add an Elseif to test for another condition if you like. You could do as many of these as you like too.

```
<script type="text/vbscript">
Dim hairColor
hairColor = "Black"
```

```
If hairColor = "Black" Then
  document.write("Same color as my cat!")
Elseif hairColor ="Blue" Then
  document.write("sane color as my parrot!")
Else
  document.write("Oh well, whatever...")
End If
</script>
```

is blue.
This time the 'Elseif' code is executed because the value of the hairColor variable

This results in:
Same color as my parrot



Chapter 11

Introduction to Javascript

Javascript

If you're new to programming/scripting, JavaScript is a good place to start. Having said that, it's still quite different to HTML so I recommend you take your time and cover off a little bit each day. Don't worry if it takes you several days to complete – it's better to fully understand everything than to brush over it and not fully comprehend it.

I suggest you bookmark this tutorial now, then continue on.

What is JavaScript?

JavaScript is a scripting language that enables web developers/designers to build more functional and interactive websites.

Common uses of JavaScript include:

- Alert messages
- Popup windows
- Dynamic dropdown menus
- Form validation
- Displaying date/time

JavaScript usually runs on the client-side (the browser's side), as opposed to server-side (on the web server). One benefit of doing this is performance. On the client side, JavaScript is loaded into the browser and can run as soon as it is called. Without running on the client side, the page would need to refresh each time you needed a script to run.

What do I need to create JavaScript?

You can create JavaScript using the same equipment you use when creating HTML. That is:

- Computer
- Text editor. For example, Notepad (for windows), Pico (for Linux , or Simpletext (Mac). You could use a HTML editor if you like it's not needed.
- Web Browser. For example, Internet Explorer or Firefox. You will need to ensure JavaScript is enabled within your browser's setting (this is normally enabled by default).

The next lesson will show you how to enable/disable JavaScript in your browser.

How to enable JavaScript

To view webpages with javascript, you need to ensure your browser has JavaScript enabled. Generally speaking, you can still view the webpage without JavaScript, but you will not be able to take advantage of the JavaScript functionality.

How do I check if my browser has JavaScript enabled?

You normally do this by checking your browser's option. This will depend on the browser you're using. Instructions for some of the more common browsers are below:

Internet Explorer (6.0):

1. Go to Tools from the top menu
2. Select Internet Options
3. Click on the Security tab
4. Click Custom Level
5. Scroll down until you see the Scripting section
6. Ensure that the Active Scripting option is set at Enabled
7. Click OK

How do I disable JavaScript?

You simply to through the steps above the JavaScript options are not checked/selected.

If you're developing web pages with JavaScript, it's good practice to view your website with JavaScript disabled. This will show you what your website will look like to users who choose to disable JavaScript.

Other browsers?

Most (if not all browsers) give you the option to enable/disable JavaScript. If your browser is not listed above, the steps above will give you some idea of how to find it. Just look for something called tools, options, preferences or something similar.

Warning

Java and JavaScript are two different things – make sure you're enabling/disabling the right option!

JavaScript Syntax

What does JavaScript syntax mean? JavaScript syntax refers to a set of rules that determine how the language will be written (by the programmer) and interpreted (by the browser).

The JavaScript syntax is loosely based on the Java syntax. Java is a full blown programming environment and JavaScript could be seen as a sup-set of the Java syntax. Having said this, that is where the similarities end – Java and JavaScript are two totally different things.

In learning JavaScript you will become familiar with terms such as variables, functions, statements, operators, data types, objects etc.

It will take most of this tutorial to show you the complete JavaScript syntax. For now, I'll give you a quick intro by showing you an example and explanation.

Explanation code

```
<script type="text/javascript">
<!--
    Document.write("JavaScript is not Java");
-- >
</script>
```

This results in:

JavaScript is not Java

The above example is how you write text to a web page using JavaScript.

Explanation of code

- The `<script>` tags tell the browser to; expect a script in between them. You specify the language using the type attribute. The most popular scripting language on the web is JavaScript .
- The bits that look like HTML comments tag (`<!-- -- >`) are just that- HTML comment tags. These are optional but recommended. They tell browsers that don't support JavaScript (or with JavaScript disabled) to ignore the code in between. This prevents the code from being written out to your website users.
- The part that writes the actual text is only 1 line (`document.write("JavaScript is not Java");`). This is how you write text to a web page in JavaScript. This is an example of using a JavaScript function (also known as method).

Where to put your scripts?

You can place your scripts in any of the following locations:

- Between the HTML document's head tags.
- Within the HTML documents body (I.e. between the body tags).
- In an external file (and link to it from your HTML document).

JavaScript Popup Boxes

You've probably encountered JavaScript popup boxes many times while visiting websites. Now, I don't mean "popup windows" – we'll cover that later. What I mean is, a popup box that displays a message, along with an "OK" button. Depending on the popup box, it might also have a "Cancel" button, and you might also be prompted to enter some text. These are all built into JavaScript and are what I call "JavaScript popup boxes". They can also referred to as "dialog boxes", "JavaScript dialogs", "popup dialog" etc.

Type of Popups

JavaScript has three different types of popup box available for you to use. Here they are:

Alert

Displays a message to the user. Example:

Confirm

Asks the user to confirm something. Often, this is conjunction with another (potentially significant) action that the user is attempting to perform. Example:

Prompt

Prompt the user for information. Example:

Popup code

Here's the syntax for specifying popup boxes, along with some examples.

Type of popup	syntax	Example code	Example display
Alert	Alert("message");	Alert ("Hey, remember to tell your friends about Quackit.com!");	
Confirm	Confirm ("message");	Confirm ("Are you sure you want to delete the internet?")	
Prompt	Prompt ("message");	Prompt ('please enter your favorite website', 'Quackit.com');	

Note that the user's browser determines what the popup box actually looks like. This is because our popup code is simply telling the browser to "display this type of popup with this message". It is up to the browser to render the correct type of popup with the specified message.

Integating JavaScript with HTML

You will have noticed that the (above) example popups didn't appear as soon as you loaded this page. They only appeared after you clicked the relevant button. This is because I placed code into each HTML button, so that when it was clicked, it triggered off our JavaScript.

This is a very common way of using JavaScript on the web. By "attaching" JavaScript to our HTML elements, we can make our pages much more responsive to our users' actions.

JavaScript and HTML

In previous lessons. We've learned about the syntax of JavaScript, but we haven't yet learned how to "attach" the JavaScript to our HTML elements. That's what I'll show you in this lesson.

Standalone JavaScript

First, let's look at what a standalone piece of JavaScript looks like.

```
<script type="text/javascript">
<!--
    alert('Hey, remember to tell your friends about Yahoo.com!');
-->
</script>
```

If we place the above JavaScript between the 'head' tags (or 'body' tags), it will run as soon as we load the page.

Now this is fine – as long as you want it to run as soon as the page loads!

But, what if you don't want it to run as soon as the page loads? What if you only want it to run when a user clicks on a button?

Easy! This is where you need to use an "event handle".

What's an Event Handler?

In JavaScript/HTML, an event handler allows you to attach your JavaScript to your HTML elements.

Event handlers allow your web page to detect when a given "event" has occurred, so that it can run some JavaScript code. An example of an "event" could be say, a click on an HTML element.

In your code, an event handler is simply a special attribute that you add to your HTML element. For example, to run some JavaScript when the user clicks on an element, you add the `onClick` attribute to the element.

The examples below demonstrate this for you.

Adding JavaScript to an HTML Element

Here's a basic example of adding JavaScript to an HTML element. In this case, when the user clicks on our button, a JavaScript alert box is displayed. This is done by adding an `onClick` attribute and placing the JavaScript alert box code into it.

When you use JavaScript like this, you don't need to use script tags (like we did above). Simply place your JavaScript within the event handler and it will work.

Code:

```
<input type="button" onClick="alert(Hey, remember to tell your friends about  
Yahoo.com!);"value="Click Me!"/>
```

Result:

JavaScript “On Double Click”

You could just have easily used another event to trigger the same JavaScript. For example, you could run JavaScript only when the double clicks the HTML element. We can do this using the `ondblclick` event handler.

Code:

```
<input type="button" ondblclick="alert(Hey, remember to tell your friends about  
Yahoo.com!);"value="Click Me!"/>
```

Result:

There are plenty of other event handlers you can use within your JavaScript/HTML code. In total, there are 18 event handlers (as listed by the W3C).

Complex Code

Once you become well versed in JavaScript, you'll be able to write some complex programs using lots of code. When this happens, you'll want to place your code into a “function”. Here we call total, there are 18 event handlers (as listed by the W3C).

External JavaScript File

You can place all your scripts into an external file (with a `.js` extension), then link to that file from within your HTML document. This is handy if you need to use the same scripts across multiple HTML pages, or a whole website.

To link too an external JavaScript file, you add a src attribute to your HTML script tag and specify the location of the external JavaScript file.

Linking to an external JavaScript file

```
<script type="text/javascript" src="external_javascript.js"> </script>
```

Contents of your external JavaScript file

The code in your .js file should be no different to the code you would normally have placed in between the script tags. But remember, you don't need to create to script tag again – it is already on the HTML page calling the external file!

JavaScript Operators

JavaScript operators are used to perform an operation. There are different types of operators for different uses.

Below is a listing of JavaScript operators and a brief description of them. Don't worry if you don't understand all of them at this stage – just bookmark this page for reference and return when ever you need to.

Arithmetic Operators

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus (remainder of a division)
++	Increment
--	Decrement

Assignment Operators

Operator	Description
=	Assign
+=	Add and assign. For example, x+=y is the same as x=x+y.
- =	Subtract and assign. For example, x-=y is the same as x=x*-y.
* =	Multiply and assign. For example, x*=y is the same as x=x*y.
/ =	Divide and assign. For example, x/=y is the same as x=x/y.
% =	Modulus and assign. For example, X%=y is the same as x=% y.

Comparison Operators

Operator	Description
= =	Is equal to
= = =	Is identical (is equal to and is of same type)
!=	Is not equal to
!= =	Is nor identical
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to

Logical /Boolean Operators

Operator	Description
&&	and
	or
!	not

String Operator

Operator	Description
=	Assignment
+	Concatenate (join two string er)
+=	Concatenate and assign

You will learn how to use some of the most common of these JavaScript operators in the following pages.

JavaScript Variables

Like other programming languages, JavaScript variables are a container that contains a value – a value that can be changed as required.

For example, you could prompt your website user for their first name. when they enter their first name you could store it in a variable called say, firstName. Now that you have the user's first name assigned to a variable, you could do all sorts of things with it like display a personalized welcome message back to the user for example. By using a variable to store the user's first name, you can write one piece of code for all your users.

Declaring JavaScript variables

First, you need to declare your variables. You do this using the var keyword. You can declare one variable at a time or more than one. You can also assign values to the variables at the time you declare them.

Different methods of declaring JavaScript variables

```
//declaring one javascript variable
Var firstName;
//declaring multiple javascript variables
Var firstName, lastname;
//declaring and assigning one javascript variable
Var firstName = 'Homer';
//declaring and assigning multiple javascript variables
Var firstName = 'Homer', lastname = 'Simpson';
```

Using JavaScript variables

Although there are many user for JavaScript variable, here is a quick and dirty example:

```
</script language="javascript" type="text/javascript">
<!--hide me
Var firstName = prompt("what's your first name?","");
// end hide-- >
<!-- hide me
Document.write(firstName);
//end hide-- >
</script>
```

The above example opens a JavaScript prompt, prompting the user for their first name. It will then write the name to the page (in practice, you would output the name somewhere between the <body></body>tags).

I suspect you can find a much better use for your javascript variables but this simply to demonstrate how easily you can store data inside a variable – data that could change at any moment.

Rules for JavaScript Variables

- Can contain any letter of the alphabet, digits 0-9, and the underscore character.
- No spaces
- No punctuation characters (eg comma, full stop, etc)
- The first character of a variable name cannot be a digit.
- JavaScript variables' names are case-sensitive. For example firstName and firstName are two different variables.

JavaScript functions

in JavaScript, you will use functions a lot. A function (also known as a method) is a self-contained piece of code that performs a particular “function”. You can recognise a function by its format- it's a piece of descriptive text, followed by open and close brackets.

Sometimes there will be text in between the brackets. This text is known as an argument. An argument is passed to the function to provide it with further info that it needs to process. This info could be different depending on the context in which the function is being called.

Arguments can be extremely handy, such as allowing your users to provide information (say via a form) that is passed to a function to process. For example, your users

could enter their name into a form, and the function would take that name, do some processing, then present them with a personalized message that includes their name.

A function doesn't actually do anything until it is called. Once it is called, it takes any arguments, then performs its function (whatever that may be).

Writing a function in JavaScript

It's not that hard to write a function in JavaScript. Here's an example of a JavaScript function.

Write the function:

```
<script type="text/javascript">
<!--
Function display Message(firstName) {
    Alert("Hello " + firstName + ", hope you like JavaScript function!")
}
//-->
</script>
```

Call the function:

```
<form>
First name:
<input type="text" name="yourName"/>
<input
Type="button"
OnClick="displayMessage(from.yourName.value)"
Value="Display Message"/>
</form>
```

This should work like this:

First name:

Explanation of code

Writing the function:

1. We started by using the function keyword. This tells the browser that a function is about to be defined.

2. Then we gave the function a name, so we made up our own name called "displayMessage". We specified the name of an argument ("firstName") that will be passed in to this function.

3. After the function name came a curly bracket {. This opens the function. There is also a closing bracket later, to close the function.

4. In between the curly brackets we write all our code for the function. In this case, we use JavaScript's built in alert() function to pop up a message for the user.

Calling the function:

1. We created an HTML form with an input field and submit button
2. We assigned a name ("yourName") to the input field
3. We added the onClick event handler to the button. This event handler is called when the user clicks on the button. This is where we call our JavaScript function from. We pass it the value from the form's input field. We can reference this value by using "from.yourName.value".

JavaScript Events

In the previous lesson, we used an event handler to trigger off a call to our function. There are 18 event handlers that you can use to link your HTML elements to a piece of JavaScript.

When you write a JavaScript function, you will need to determine when it will run. Often, this will be when a user does something like click or hover over something, submit a form, double clicks on something etc.

These are examples of events.

Using JavaScript, you can respond to an event using event handlers. You can attach an event handler to the HTML element for which you want to respond to when a specific event occurs.

For example, you could attach JavaScript's on Mouse over event handler to a button and specify some JavaScript to run whenever

This event occurs against that button.

The HTML 4 specification refers to these as intrinsic events and defines 18 as listed below:

Event Handler	Event that it handles
onBlur	User has left the focus of the on object. For example, they clicked away from a text field

	that was previously selected.
onchange	User has changed the object, then attempts to leave that field (i.e. clicked elsewhere).
onClick	User clicked on the object.
ondblclick	User clicked twice on the object
onfocus	User brought the focus to the object (i.e. clicked on it/tabbed to it)
onkeydown	A key was pressed over an element.
onkeyup	A key was released over an element.
onkeypress	A key was pressed over an element then released.
onload	The object has loaded.
onMouseDown	The cursor moved over the object and mouse /pointing device was

The events in the above table provide you with many opportunities to trigger some JavaScript from within your HTML code.

Chapter 12

Introduction to Web Hosting

Domain name – the things that look something like your-domain-name.com - are an essential part of having a website. They represent the “web address” of your website.

You will need your own domain name if you intend to create a website. You will also need to register a domain name if you intend to set up email accounts for your business. Before you create a website (or email accounts for your business), you will need to know how to register a domain name.

Without your own domain name, your website will always be attached to another website (with its own domain name). either that, or your users will only be able to access your website using its IP address (which looks something like 202.153.16.25).

What Is a Domain Name?

Domain names are a special name that you can apply to your website or MySpace page. Once you apply it to your website, users can reach your website by typing this domain name into the browser’s address bar.

Example of domain names are Yahoo.com, great-workout.com and google.com. You can reach the “great workout” website by typing its domain name (great-workout.com) into the address bar.

About “www”

Most websites have a www subdomain applied to their domain name so that you can reach the website by typing “www” followed by the domain name (eg, www.yahoo.com). This is done on the DNS server after you’ve registered the registered the domain name). You can also add other subdomains as you wish.

You can reach any page on a website by typing the domain name followed by the path to the page. For example, www.rgcsn.com.in

Only one person/company can own a domain name at any time. Therefore, if you want mycompany.com but someone else has it, you will need to either find another name, or offer to buy it from them. You could also wait for it to expire and hope they don't re-register it.... but you could be waiting a long time!

Types of Domain Names

There are various types of domain names. The most common ones are the generic domains (which include .com domains) and the country code domains (which use a two letter country code, such as .uk for United Kingdom).

You can tell which type a domain name is by its extension. To find out more about the various types of domain names see this list of domain extensions and definitions.

Top-Level Domains

The last segment of a domain name (eg.com) is referred to as the top-level domain.

There are many more top-level domains than just those with a .com suffix. There are many other suffixes that can be used, such as .net, .org, .biz, .info to name a few. Most of these have a general purpose, for example, .org was created for organizations, .info was created for information sites etc.

If the .com version of your chosen name is unavailable, another top-level domain could be available. These are seen as different domain names. For example, mycompany.com and mycompany.org are two different domain names- one company could register the .com and another could register the .org version. The same applies for company specific domain names. Some countries have further criteria that you need to satisfy before they will allow you to register a domain name with that country's suffix/domain extension.

Choosing a Domain Name

You should choose a domain name that truly reflects what your website is all about. You should try and keep your domain name concise. If it consists of multiple words consider separating each word with a hyphen. This will largely depend on how the domain name looks with and without a hyphen. It will also depend on the availability of your preferred domain name.

If you find that your preferred domain name has already been registered under all applicable suffixes, you might need to get creative and think of another domain name. At this

point, you may also need to consider the name of your website. If this isn't an option, you could try buying a domain name off the current owner.

Registering a Domain Name

You don't actually buy a domain name, you register it. To do this, you need to register it with a domain name register. You can choose how long you'd like to register it for. Options typically include anywhere from 1 year to 10 years.

In a sense, you can buy a domain name. You can do this if someone else already has already registered the domain name. What you're really doing though is buying the right to register it. You still need to keep the domain name registered with a registrar, otherwise someone else will be able to register it once it expires.

Many web hosting provides include domain registration in their hosting packages. In this case, you don't need to register it through a separate domain name registrar.

You can register your domain name through ZappyHost.

You can also see this step by step guide for registering your domain name.

Hosting a Domain Name

Once you've registered a domain name, your domain name registrar will probably point it to a webpage that they've configured. This page may have ads on it – that way they can make money from your domain name!

If you need it to point to your website, you will need to update the authoritative DNS servers to be those of your website hosting provider. Your website hosting provider can provide you with these details. Once you've received them, you should be able to log in to a control panel via your domain name registrar's website. This control panel should have an option for you to update the authoritative DNS servers of your domain name.

If you have a MySpace page, you may want your domain name to point to it. If so, see the "Domain Forwarding and MySpace Pages" section below.

Behind the Scenes

You may be wondering how on earth the domain name ends up pointing to your website.

When your hosting provider configures your website, they assign it an IP address. An IP address looks something like this: 202.45.22.13. Your IP address is unique – no other IP address on the Internet is the one given to your website. Now, what this means is that anyone could access your website by typing in the IP address. Technically, you don't even need a domain name. Only problem with this is that IP address are hard to remember. It's much easier to remember a nice catchy domain name.

Anyway, after your hosting provider assigns an IP address to your website, you have the option of having a domain name resolve to that IP address. When you enter in the details of an authoritative DNS server, you are specifying which server should be used to resolve that domain name. The authoritative DNS server links your domain name with an IP address. You could have as many domain names as you like pointing to the same IP address. Therefore, you could have mycompany.com, mycompany.org and mycompanys-product.com, all pointing to the same website.

Domain Forwarding and MySpace Pages

You can make your domain name “forward” to another web page if you so wish. This is where your domain name redirects to another web page instead of being mapped to a website.

Many people do this for their MySpace profiles. When a user types in their domain name, they are redirected to the MySpace page.

Domain Name Usage

Although most people register domain name to use with website, there are other reasons people register domain names.

Here are some of the most common uses for domain names.

Domain Name Extensions

A domain name extension is the last part of a domain name. it is the part that follows the “name” part of your domain.

For example, in the domain name yahoo.com the domain extension is .com

Web Hosting

Web hosting has become an enormous business world wide. There are many companies providing a great web hosting service and many others who are just out for a quick buck. In choosing a web hosting provider, you need to have a basic understanding of what web hosting is about.

This tutorial covers all the basics of web hosting – what web hosting actually is, how to choose a web host, domain name and more.

This web hosting tutorial does not teach you about hosting your own website. It teaches you what you need to know in order to choose a good hosting provider and configure your website with them.

Web Site Hosting

Web site hosting refers to a service that makes your web site available to your users. In general, most web sites are intended to be viewed by anyone on the Internet, at any time. If you want anyone on the Internet to be able to view your web site at any time, you need to make sure your web site is running on a computer which is connected to the Internet 24 hours per day, 7 days per week (24/7).

To do this you have two options.

Option 1: Hosting the Web Site Yourself

Hosting the web site yourself means that you will need to have the right equipment, including a web server and a permanent connection to the Internet. You would need to ensure that your Internet service provider (ISP) allows this type of usage too. You would also need to have web site administration skills such as managing/configuring a web server, patch management, firewall, virus protection etc.

Most people don't do their own web site hosting. Even if they have the skills or equipment, it often makes more sense to hand this task to a third party who can monitor the web site 24/7.

Option 2: Find a Hosting Provider to Host your Web Site

This is the most common choice for web site hosting. Using a third party hosting provider allows you to concentrate on developing your web site (or whatever else it is you may do), and leave the hosting tasks up to a company who (presumably) specializes in web site hosting.

A good web site hosting company will provide at least the following services:

- 24/7 support
- 24/7 FTP access (so you can update your web site)
- A number of email accounts (i.e. yourName@yourDomain.com)
- Online control panel for managing your web site
- Online traffic statistics (so you can see how much traffic your web site receivers)
- A robust database management system, such as MySQL or MS SQL. This is so you can add your own database if required.

If you require specialized server application software such as ColdFusion, SQL Server, or PHP, you will need to check that the hosting provider supports this.

Once you start looking for a web site hosting provider, you will notice there is a very large difference in how much they charge. You will also notice that most web hosts have multiple hosting plans, each at different prices. Any good web host will allow you to upgrade a plan later on if required, so don't feel as though you need to start off on a plan that costs more than you need.

Types of Web Hosting

When searching for a hosting provider to host your website, you'll probably notice that most hosting providers have more than one web hosting plan. The plan you choose will depend on your hosting needs.

If you're just starting out with your own website, a shared hosting plan is usually fine. Shared hosting is usually much cheaper than other hosting plans. Unless you have a specify need to hire a dedicated server, a shared hosting plan should do web hosting:

Here's more about the different types of web hosting:

- **Shared Hosting**

Shared hosting refers to when your web site is hosted on a server along with many other customers' web sites. Don't worry, your users won't know this – your web site is still configured as a separate web site on the server and can still have its own domain name etc. It is simply sharing the server with other web sites.

The benefit of shared hosting is price. Having your own server (otherwise known as a “dedicated server”) costs a lot more. Unless you get lots of traffic, it doesn't make sense to pay more for a dedicated server.

- **Dedicated Servers**

A dedicated server is a server that hosts only your web site (or web sites). This can give you more control over your web site. It can also help in ensuring that other customers' web sites don't impact on your web site. Using dedicated servers is much more expensive than shared hosting, but if your site receives lots of traffic or you have other requirements (such as extra security requirements), a dedicated server could be for you.

You are able to log in to your dedicated server just as you would log in to your own computer. Once logged in, you can install and configure software as you wish.

- **Virtual Dedicated Servers**

Virtual dedicated servers are a low-cost alternative to dedicated servers. The web host can put many virtual servers on each machine, therefore reducing costs. When you log in to the virtual server, it appears as though you have your own dedicated server (even though other virtual servers are probably running on the same machine).

- **Managed Hosting**

A managed hosting solution is where you have a dedicated server and the web site hosting provider manages your server for you. Typically, the hosting provider will be responsible for the hardware, operating system, virus protection, patch management etc. You still have total control over the machine (depending on SLA arrangements), and you are still in charge of your own web site content. Managed hosting is usually a lot expensive, and in some cases, costs will be negotiated between the two parties.

Web Server

A web server is a piece of software that enables a website to be viewed using HTTP. HTTP (Hyper Text Transfer Protocol) is the key protocol for the transfer of data on the web. You know when you're using HTTP because the website URL begins with <http://> (for example, <http://www.yahoo.com>).

You might be thinking "I always thought a web server was a special, high-powered computer". Well, you'd be right too. Some high-powered computers are referred to as web servers as they have been built with web hosting in mind. But in most cases, when someone refers to a web server, they are referring to a piece of software that you install on a computer.

What Does a Web Server Look Like?

That depends on which web server you choose to install. Here's an example of Microsoft Internet Information Services (IIS) 5.1 looks like.

The left pane represents the various websites, FTP sites, and SMTP virtual servers. When an item in the left pane is selected, the contents are displayed in pane on the right hand side.

In the above screenshot, there is one website (called “Default Web Site”), one FTP site (called “Default FTP Site”), and one SMTP virtual server (called “Default SMTP Virtual Server”).

You can right click on an item to display it’s properties. For example, you can right click on “Default Web Site” to display (and configure) the properties of that website.

Do I Need are Easy!

You might also be thinking that web servers are way too advanced for you – that they are only used by professional web developers and/or hosting companies. Please don’t think that!

Think of web server as simply another piece of software you can install on your machine. Once you install it, you can configure it to suit your needs.

And, depending on your computer set up, you may even find that you already have a web server on your machine.

Now, having declared that “web servers are easy!”, there are many advanced topics regarding web server. I won’t be going into any detail in this tutorial. You can get a web server up and running on your machine with a minimum of technical knowledge. Then once you’ve done that, you’ll start to become familiar with the various options available to you. Then if required, you can research the more advanced topics to suit your needs (such as security, load issues, logging etc)

How Websites Work:-

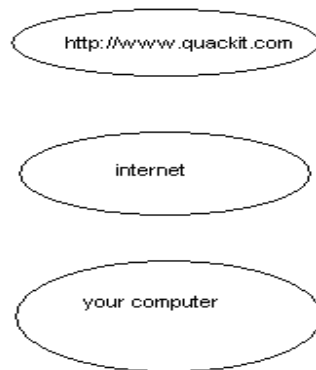
What is a website?

A website is a collection of web pages. These web pages are usually located on a web server that is connected to the Internet. The web server is a computer that has been built specifically to host websites, and contains web server software.

The web server is usually located with a web hosting provider (a company that provides web hosting to its customers).

What Happens When I view a Website?

The following diagram demonstrates what happens each time you view a website:



This diagram is only a simplistic version of what happens, but the key point is that, each time you view a web page in your browser, you initiate a request across the internet to a web server. This is true even if you view multiple pages from the same website – each new page is a new request.

Actually, in most cases, each web page results in multiple requests. This is because most web pages consist of more than one file (or “resource”). Therefore, if a page contains 3 images, there will be at least 4 requests; 1 for the web page, and 1 for each image.

So, for example, if you type www.natural-environment.com/index.cfm” into your browser, the following files might be requested:

- <http://www.natural-environment.com/index.cfm>
- http://www.natural-environment.com/images/image_1.gif
- http://www.natural-environment.com/images/image_2.gif
- http://www.natural-environment.com/images/image_3.gif

in this case there are 3 images (called “image_1.gif”, “image_2.gif”, and “image_3.gif”) located in a directory called “images”.

It’s important to note that each file – including the web page file (index.cfm) – needs to be downloaded to your computer before you can view it. That’s why you may sometimes see a web page appear without the images first, then one by one, the images appear. Larger images will take longer to appear because they take longer to download. This is why the speed of your Internet connection is important – a slower connection will make many websites appear slower (especially those with lots of images).

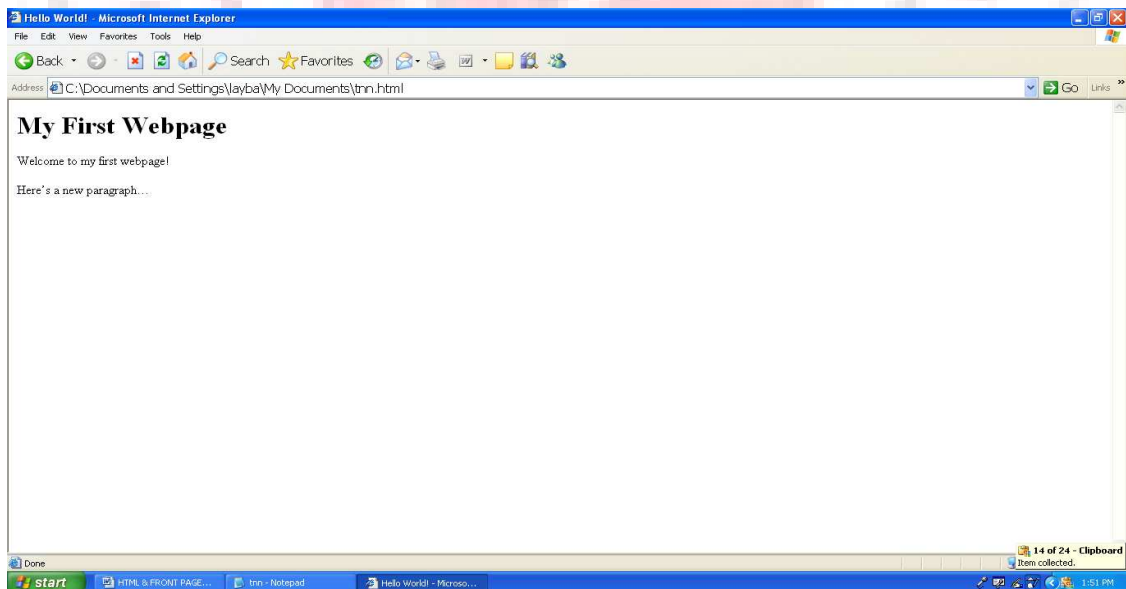
Web Page Content

A web page is simply a text file. The text file contains text and markup code that specifies what should appear when viewed with a web browser. When you view the page with your web browser, the web browser renders the markup code (it displays as intended), instead of just outputting the code (browsers know to do this). Behind the scenes, the web page looks different to what you see in your browser. When you view the file with a text editor or HTML editor, you'll see the HTML tags (as written by the programmer).

Below is an example of a what page looks like “behind the scenes”. This is what a web page looks like if you open it up using a text editor (such as Notepad, Simple Text, etc);

```
<html>
<head>
<title> Hello World!</title>
</head>
<h1> My First Webpage</h1>
<p> Welcome to my first webpage!</p>
<p> Here's a new paragraph...</p>
</body>
</html>
```

And here's what it looks like when you view it using a web browser (such as FireFox, Internet Explorer, etc):



The Website Creation Process

Now if you chose not to use a website builder, you are probably curious to know what is involved in making a website. Here I example the three main steps of creating a website.

Creating a website basically involves the following 3 steps:

Step 1>>	Step 2>>	Step 3>>
Register a Domain Name	Get a Web Host	Build your Website
<p>A domain name is the thing that looks like mysite.com. You register these on an annual basis.</p> <p>Domain names can cost anywhere between \$10 and \$50 per year. I'll show you how to get one for \$1.99 (below).</p>	<p>A web host is a company that enables your website to be viewed by the rest of the world 24/7</p> <p>A good hosting plan can be anywhere between \$10 per month and many thousands per month. Below, I 'll show you Excellent hosting plans Starting from \$5.35 per Month.</p>	<p>You can either build it yourself or use a website builder.</p> <p>Below, I'll show you an online website builder that includes templates and hosting. And the great thing is it's cheaper than most hosting plans!</p>

A more detailed explanation of these 3 steps follows.

STEP 1: REGISTER A DOMAIN NAME

Before you even think about building your website, you should register a domain name. a domain name is the thing that looks this : "yourdomain.com ".

Your domain name represents the URL (or permanent web address) of your website. Therefore, when anyone types in "yourdomain.com" or www.yourdomain.com", they will see your website.

http://www.your-domain.com

Your register your domain name through a domain name register or hosting provider. You simply check that your preferred domain name is available, then register it (online).

Domain names are registered on an annual basis, and you can usually register it many years in advance (or set 'auto-renew').

Prices can range from between \$10 to \$15 up to as much as \$35 or even \$45 (this is for the sane product!).

STEP 2: GET A WEB HOST

A web host (or hosting provider), is a company that makes your website available for the world to see. They have the equipment and technical skills to make sure your website is available to the world, 24 hours a day, 7 days a week.

Choose your hosting provider carefully – a good host will have excellent support. A bad host can have almost no support at all! If you're new to creating websites, good support can reduce the stress that comes from doing something for the first time.

Also, some web hosts have an online website builder. A good website builder can enable anyone to make a website – even complete novices.

If you like the idea of using an online website builder, check out the website builder below. This enables you to build a website virtually within minutes, and hosting is include as well as plenty of other extras).

If you don't need a website builder (i.e. you prefer to build your website by yourself), you can find some great hosting plans here. Once you've built your website, simply upload it via FTP.

If you'd like to learn more about web hosts, check out my web hosting tutorial.

STEP 3: BUILD YOUR WEBSITE

This part can be as easy or as hard as you like it to be!

The easy way? The easiest way to build a website is to use your web host's website builder (assuming they have one). As mentioned, ZappyHost provides Website Tonight which makes it extremely easy for you to make your website.

The hare way? OK, if you think a website builder sounds too much like cheating, you probably went to learn how to build a website the way the pros do it – by coding/programming. If this sounds like you, then see this introduction to making your own website.



Chapter

13

Creating and Editing Web Pages Using Front Page

The World Wide Web is a great way for people to communicate with each other. While you have conversations with other people over Internet newsgroups, you can also publish your own personal Web site – a collection of one or more pages on which you can share all sorts of information. This book will introduce you to web page creation and Web site management, two aspects that will show you how easy and fun it is to build and maintain a Web site Microsoft FrontPage 2000.

In this first lesson, you'll create a "Millennium Celebration Web" that provides information about the Year 2000. Its pages will contain travel destinations for millennium celebrations, an online photo album, and a list of links to other sites.

We've prepared a folder of various files for you to practice with while you create this web. When you have completed the lessons, you will have a good understanding of FrontPage and its features.

Understanding How FrontPage Works

Microsoft Office FrontPage 2003 is a comprehensive application that you can use to develop Web sites. This sophisticated program includes everything you need to create Web sites ranging from a simple Web-based resume to a complex Web-based retail store.

In spite of its sophistication, FrontPage is easy to use. As a member of The Microsoft Office System 2003 suite of applications, it works pretty much the same way the other Office applications do. If you've avoided trying to create Web sites because you didn't want to learn how to program in Hypertext Markup Language (HTML), FrontPage might well be the answer you've been waiting for. With FrontPage, you can easily create good-looking, interesting Web sites that incorporate complex elements, without typing a single line of programming code. But if you have some HTML programming experience or want to feel more in control,

FrontPage gives you easy access to the code that it creates behind the scenes. You can view and edit the underlying HTML code at any time; but the great thing is that you don't have to. No programming experience is necessary to become a successful FrontPage developer.

This chapter introduces FrontPage and explains the concept of a FrontPage-based Web site. You will learn how to open an existing Web site, how to navigate between Web pages, and how to view ht pages in different ways. You will then look at various ways of working in FrontPage and learn how to locate and control the FrontPage features you are likely to want to use in your own Web sites. In addition, you will learn how to view the underlying HTML code that makes all Web sites work. You will also get an overview of the different types of Web sites you can create with FrontPage **and of the decision-making tools and resources that are necessary to create, manage, and maintain a personal or commercial Web site.**

Starting FrontPage

If you haven't already installed FrontPage, you'll need to do so before you begin the book.

➤ To start Microsoft FrontPage



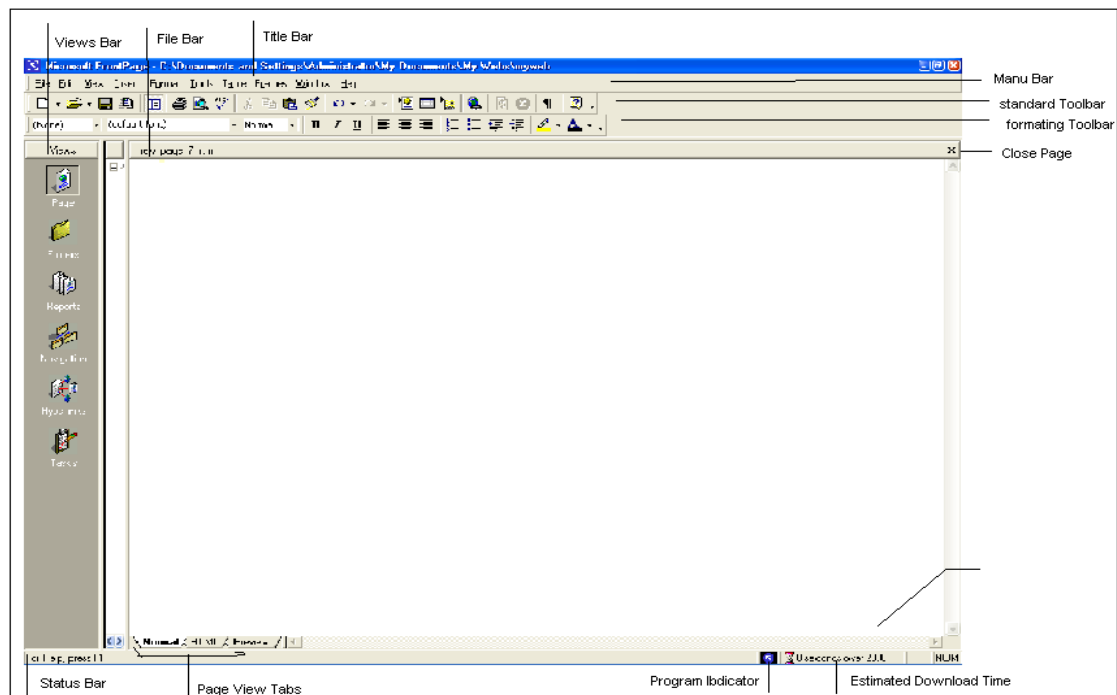
Start button

- On the Windows taskbar, click the **Start** button, point to **programs**, and then click **Microsoft FrontPage**.

FrontPage opens and displays a blank page ready for editing


Workspace Overview

FrontPage 2000 has a new, integrated interface that lets you create and edit Web pages and manage entire Web sites – all within one application. All toolbars and menus are consistent with Microsoft Office applications and can be fully customized. You can also use convenient keyboard shortcuts to accelerate common tasks such as opening webs and pages, printing, and many other commands.



The Standard and Formatting toolbars are displayed by default. They provide easy access to the commands you will use most often when working in FrontPage.

The Standard and Formatting toolbars are displayed by default. They provide easy access to the commands you will use most often when working in FrontPage.

 **Showing additional toolbars** You can customize your workspace by displaying additional toolbars or changing the buttons they contain. On the **View** menu, point to **Toolbars**, and then select the toolbar you want to display. To add or remove buttons from toolbars, click **Customize**.

What you see in the main application window depends on the currently selected view.

The icons on the Views bar provide different ways of looking at the information on your page or in your web. As you work with FrontPage, you'll frequently switch between views to match the current task at hand – from the early steps of creating a page, to the moment a whole Web site is ready to be published on the World Wide Web.

When you start FrontPage, Page view is displayed by default. Page view is a powerful editing tool for creating and designing Web pages. As you enter text, place pictures,

insert documents, create tables, make lists, and design the appearance of your Web pages, Page view displays them as they will appear in a Web browser. All HTML code is automatically created in the background and you don't need to manually edit any code unless you want to.

You'll begin working in Page view and learn about the other views later in this lesson.

Getting Stared

For this book, you'll create a Web site with five pages, on which you will tell visitors about the new millennium. Until it is published for the first time, a Web site is a work in progress. If the task of putting together a whole site seems daunting to you, don't worry. You can gradually add information and other pages to your web until it is finished. Unlike printed letters, memos, and word-processing documents, Web sites can be dynamically changed or updated even after they've been published. You can add, delete, and modify text, pictures, and entire pages at any time.

With FrontPage, you can get started easily by simply typing text on the blank document that Page view provides. For this lesson, we'll begin with the home page – the default document that greets your visitors when they surf to your Web site.

To create a home page

The home page is the front door to your Web site. Greeting your visitors as you might do in person and providing some information about the content or subject matter of your site will spark interest in the people looking at your site. The home page also contains links to the other pages in a web.

1. On the blank page in page view, type Welcome to my Web site! And then press Enter.

Just like in a word processor, pressing ENTER puts the cursor on a new line.

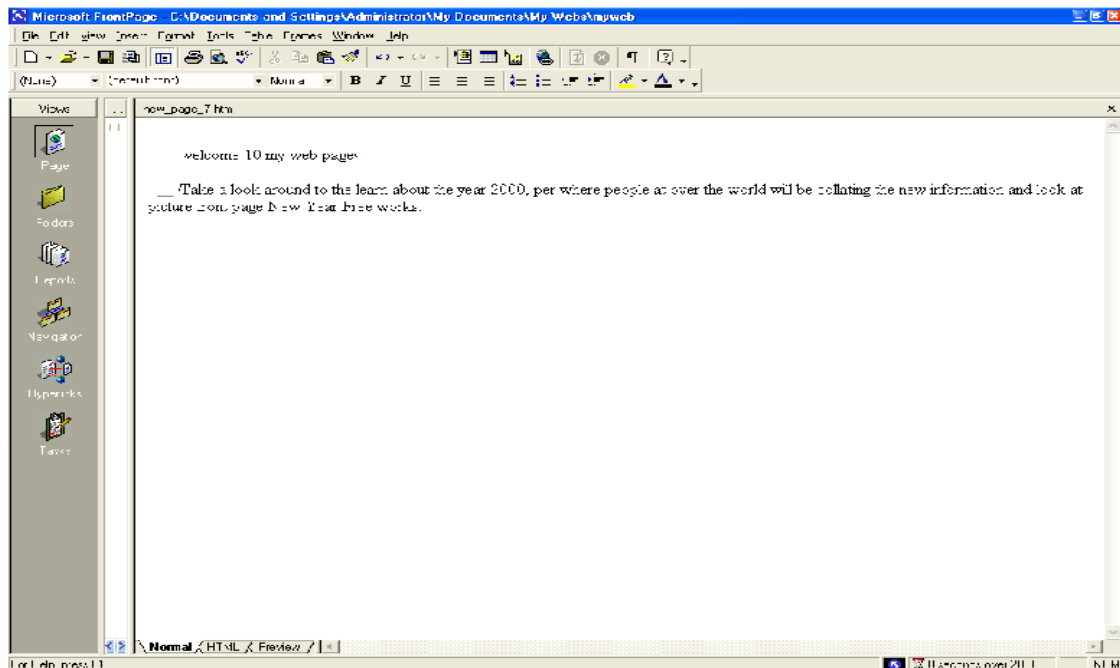
2. Next, type the sentence Take a look around to learn more about the Year 2000, see where people all over the world will be celebrating the new millennium, and look at pictures from past New Year's fireworks.

3. Press ENTERS.

After typing such a long sentence. You may wonder how much typing you'll need to do before getting to the fun stuff. Don't worry, for most of the Millennium Celebration Web content, we've already done the typing for you. And when you're ready to make your own Web

site, FrontPage lets you import any of your existing documents directly onto your Web pages without having to retype anything.

Your page should now look like this:



Next, you will add a picture to the bottom of the current page. Pictures can be scanned photographs, drawings, or computer graphics created in a drawing or image-editing program.

For this example, the picture you'll insert is a graphical button of the FrontPage 2000 web logo.

To insert a picture on the home page

1. On the insert menu, point to Picture, and then click From File.

FrontPage displays the Picture dialog box. Because you are editing a page that isn't part of a web yet, FrontPage also opens the select File dialog box, which lets you choose a picture to insert from your local file system. The picture file you'll insert is located in the Book folder that was installed with the FrontPage program files.

2. In the select File dialog box, navigate to the folder named Program Files\Microsoft Office\Office\Book by double-clicking each folder in this path until the Look in box displays the Book folder.

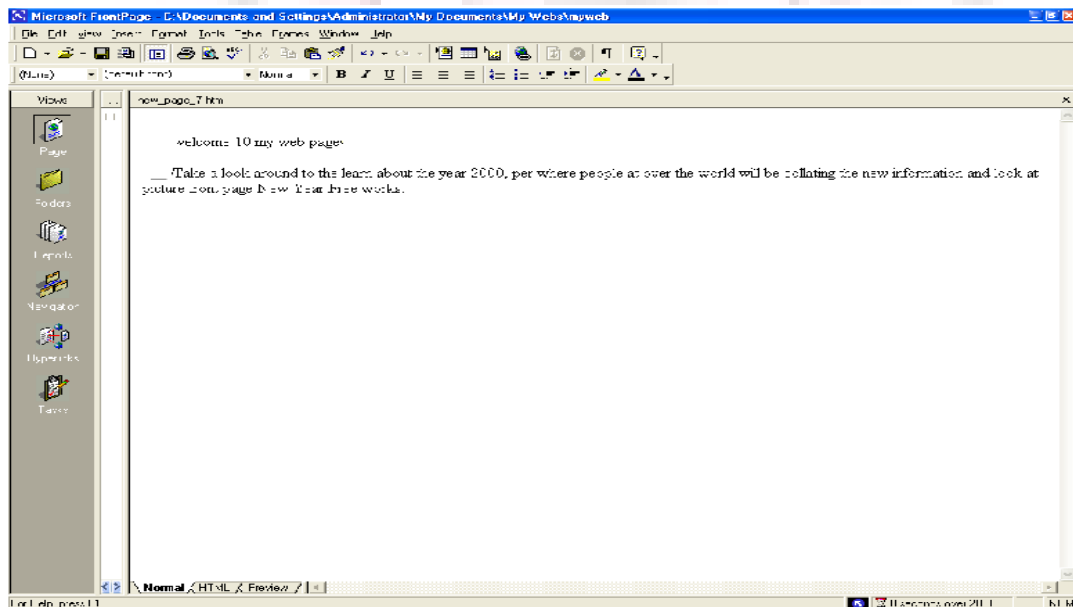
If you downloaded the book files from the www.microsoft.com, navigate to the folder named Fptutor\Samples, or to the folder where you placed the files. In the Book folder, several files will be displayed. By default, FrontPage searches for picture files.

3. Click the file named fp2000, and then click OK.

FrontPage inserts the selected picture file on the current page. It is a picture of a button that your site visitors will be able to click to learn more about FrontPage 2000.

4. Press ENTER to create a new line.

Your page should now look like this:



Merely inserting a picture of a button doesn't mean that anything will happen when someone clicks it in a Web browser. To make a picture or a word "clickable," it must have a hyperlink associated with it,

A hyperlink is a pointer from text or from a picture to another page or file on the World Wide Web. On the World Wide Web, hyperlinks are the primary way to navigate between pages and Web sites.

In the next steps, you'll create a hyperlink from the button you just placed on the home page.

To create a hyperlink from a picture

1. On the home page, click the picture of the Front page 2000 button you previously inserted.

When a picture is selected, it is shown with file handles – eight small squares around the outline of the picture. These can be used to resize a picture or change its appearance. When a picture is selected, FrontPage also displays the Picture toolbar below page view. The Pictures toolbar provides picture editing and formatting tools, which you'll learn about later.

2. On the insert menu, click Hyperlink.

FrontPage displays the Create Hyperlink dialog box. Here, you specify the target of the hyperlink you are creating. This can be a page or a file in your web, on your local file system, on a Web server, or on another site on the World Wide Web.

Because you're creating a hyperlink from a button that is labeled "Microsoft FrontPage," you'll link to the Microsoft FrontPage home page on the World Wide Web. When site visitors click the button in a Web browser, they will be taken to the right place.

3. In the URL box, type www.microsoft.com/frontpage immediately after the http:// prefix that FrontPage has provided for you.

URL is an acronym for Uniform Resource Locator. It is the technical term for what's commonly known as an "Internet address" or "Web site address." A URL specifies the unique location of a file or a collection of files on the World Wide Web.

4. Click OK to finish creating the hyperlink.

You may notice that the appearance of the button itself hasn't changed. Unlike text hyperlinks, which change the color of the clickable text and underline it, picture hyperlinks do not automatically indicate the presence of the hyperlink in an obvious way. This is intentional, because changing the appearance of the picture could obscure the intended page design in some cases.

You can quickly check the existence of a hyperlink from a picture by moving the mouse pointer over the picture. If a hyperlink is present, FrontPage displays the URL the hyperlink points to on the status bar. Next, you'll insert an animation of the number 2000 at the top of the page. Animated picture are inserted in the same way as normal pictures.

To insert an animated picture on the home page

1. Press CTRL+HOME to quickly jump to the beginning of the current page.

The key combination CTRL+HOME places the cursor in the home position – the top left margin on the current page.

2. On the Insert menu, point to Picture, and then click From File.

This time, FrontPage immediately displays the contents of the Book folder. For the duration of each work session, FrontPage remembers the names and locations of the folders you've already navigated to.

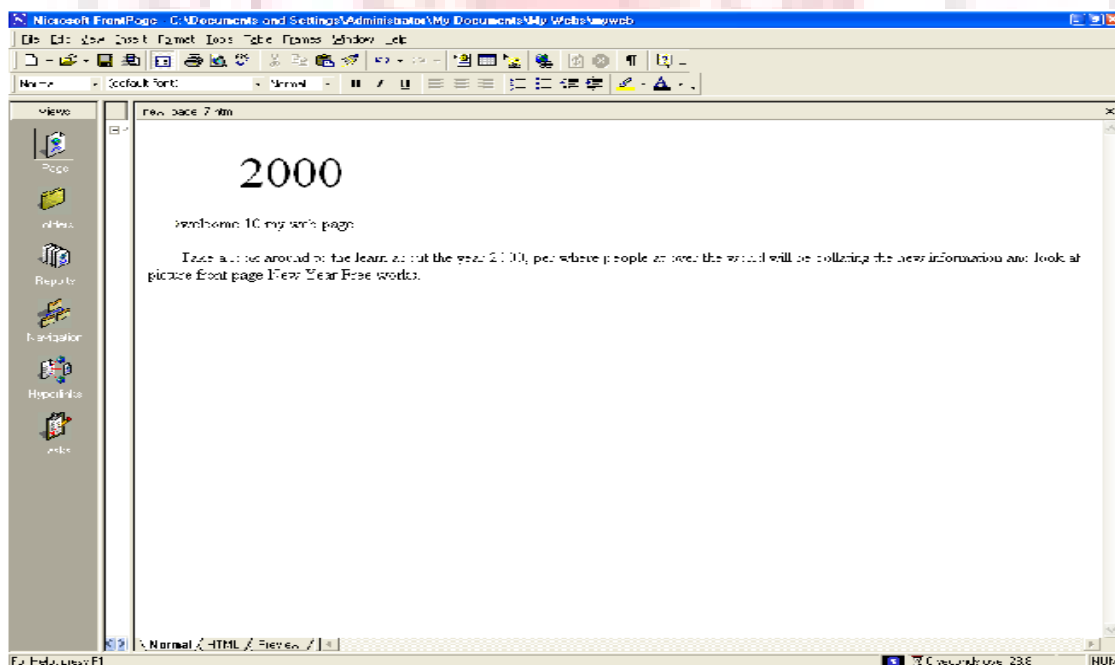
3. In the Select File dialog box, double-click the file named 2000.

Double-clicking file names are faster than selecting each file and clicking OK.

FrontPage inserts the animated picture of the number 2000 on the current page. This picture will look great against a dark background, which you'll apply later. While you edit pages in Page view, FrontPage purposely does not show text or picture animations, so they don't distract you from your work. You will be able to see what the animation looks like when you preview the home page later in this lesson.

4. Press ENTER to move the welcome text to the line below.

Your page should now look like this:



To finish the home page, you'll center the text and pictures on it.

To center elements on a page

1. On the Edit menu, click Select All.

FrontPage selects everything on the current page.

2. On the Format menu, click Paragraph.

FrontPage displays the Paragraph dialog box. Here, you can change the alignment of selected elements, and apply indentation and custom spacing for text and graphics.

3. In the Alignment list, click Center, and then click OK.

FrontPage centers the text and the pictures on the home page.

4. Click anywhere on the page to deselect all page elements.

Now that you've invested some time and completed a number of steps, it's a good idea to save your page.

To save the current page

1. On the File menu, click Save As.

FrontPage displays the Save As dialog box. Here, you can specify the location for the current page, and review or change the page title, the file name, and the file type.

2. In the Save As dialog box, click the My Document icon on the vertical Places bar.

The contents of your My Documents folder is displayed. If no files are displayed in the file list, then you currently do not have any other Web pages stored here.

3. Next to the page title field, click the Change button.

FrontPage displays the Set Page Title dialog box. Here, the default page title is based on the first line of text on the current page. A title identifies the contents of a page when it is displayed in a Web browser. For this book, you'll change the page title to something more descriptive.

4. In the Set Page title box, type Millennium Celebration – Home Page and then click OK.

In the Save As dialog box, the default file name is based on the first line of text on the current page. For this book, change the file name to something more descriptive.

5. In the File name box, change the suggested text to homepage, and then click Save.

FrontPage saves the current page.

Cascading Style Sheets

Cascading style sheets (CSS) are documents that define formats and styles for page elements including headings, paragraphs, tables, lists, and so forth. The style sheet can be either an embedded cascading style within a Web page, or as an external cascading style sheet. External style sheets can be referenced by multiple documents to provide a consistent look across pages and sites.

Web authors can also use cascading style sheets to stipulate how page elements are to be displayed by different browsers. Many Web sites utilize a browser sniffer that detects the Web browser and version used by each Web visitor and attaches the appropriate cascading style sheet to the site at that time.

To create an embedded cascading style sheet in FrontPage, click Style on the Format menu, and then define your own styles. The definitions are saved in the HTML code of the page.

To create an external cascading style sheet in FrontPage, select the CSS type from the options available on the Style Sheets tab of the page Templates dialog box, click OK to create a CSS file, and then define your styles within the file.

To attach a style sheet in FrontPage, click Style Sheet Links on the Format menu, click Add, and browser to the CSS file on your computer or (if you have an Internet connection) anywhere on the Web.

The World Wide Web Consortium (W3C) originally developed cascading style sheets. For more information about current and future CSS specifications and how various browsers support CSS, visit their Web site at www.w3c.org.

Page View Options



Page icon


While creating the home page, you've worked exclusively in normal page view, but there are three different ways you can choose to look at the current page.

To display HTML tags on the current page

- In Page view, click Reveal Tags on the view menu.

FrontPage displays graphical representations of standard HTML tags for the current page. This display is useful for people who want to know where HTML tags are placed while they design their pages.

- To hide the tags, click Reveal Tags on the View menu a second time.

-  **Getting more information about tags** When the **Reveal Tags** command is selected, you can view more information about many HTML tags by moving your mouse over each tag until a Screen Tip appears.


To display the HTML of current page

- In Page view, click the HTML tab at the bottom of the page.

This is the HTML code that FrontPage has created so far while you were designing the home page. Web browser decode these instructions to display the page. The HTML tab in page view is intended for experienced web developers and page designers who want to customize the HTML that FrontPage.

If you want to set your preferences for the way FrontPage will generate HTML code, click Page Options on the Tools menu, and then click the HTML Source tab. If you're not experienced in HTML, you don't need to make any changes here. Click Cancel to close the Page Options dialog box.

- Click the Normal tab at the bottom of the page to return to normal Page view.

-  **Using menu commands and toolbars on the HTML tab** while working in the HTML tab, you can use many menu commands and toolbar buttons just like in normal Page view.

To preview the current page

- In Page view, click the Preview tab at the bottom of the page.

Note

If you do not have Microsoft Internet Explorer installed on your computer, the Preview tab will not be displayed, and you will not be able to preview your pages this way. For more information, see Before You Begin.

Looking at your page on the Preview tab is a quick and convenient way to see how certain elements – including animations, movie clips, tables, and lists – will appear in a Web browser.

When you preview the home page you've created, you can see what the animation at the top of the page looks like. Each of the four digits rotates into place one by one, until the number 2000 is displayed. The color and edges of the four digits will look great against a dark background, which you will add in the next lesson. The animation effect will be repeated for as long as the current page is previewed or displayed in a Web browser.

- Click the Normal tab at the bottom of the page to return to normal Page view once again.

To Create a New Web Site by Using a Wizard

In addition to using templates to create Web sites, you can create sites that are a little more complex by using one of FrontPage's wizards. Wizards are similar to templates, but even better. A wizard not only creates the layout of a page or site for you, it also leads you through the process of personalizing the content and the appearance of the final product.

For example, supposed the Karen Berg, the owner of a small, fictitious plans and garden store called The Garden Company, wants to communicate with her existing customers and expand her customer base by having a corporate presence on the Internet. If maintaining a web site meets these modest goals, she might later choose to expand the site's capabilities to permit online retailing beyond the store's Seattle, Washington, location.

However, to begin with, she wants to use FrontPage to create a good – looking Web site. This is a job for one of FrontPage's wizards. In this exercise, you will use the Corporate Presence Web Wizard to create the basic corporate Web site.

BE SURE TO start FrontPage before beginning this exercise.

1. Click the down arrow to the right of the Create a new normal page button, and then click Web Site. The Web Site Templates dialog box appears.

2. In the Web Site Templates dialog box, click (don't double-click) the Corporate Presence Wizard icon.

3. In the Options area, click the Browser button, and browse to the My Web Site folder.

4. On the dialog box toolbar, click the Create New Folder button. Name the new folder CorpSite, and click OK.

5. Back in the New Web Site Location dialog box, click Open and then click OK.

The first of a series of Corporate Presence Web Wizard dialog boxes, called pages, **appears. The wizard uses these pages to prompt you to make choices and enter basic corporate information.**

6. Read the information on the first page, and then click Next to move to the second page.

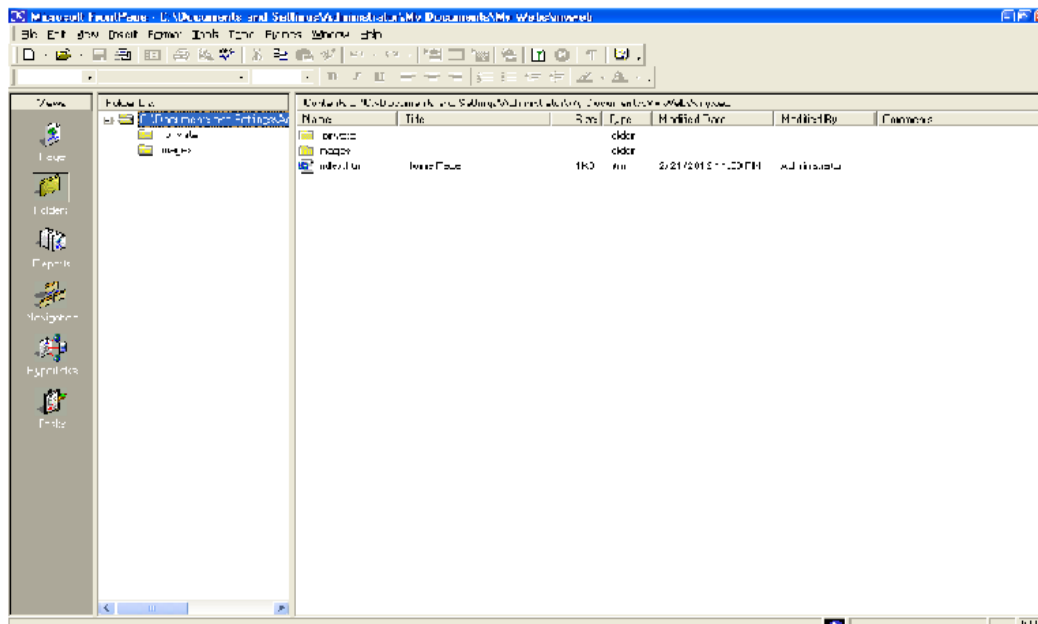
7. Continue reading the information and clicking Next to accept all the default selections in each of the Corporate Presence Web Wizard pages, until you come to the one that requests the name and address of the company.

8. Enter the information shown here (or your own personalized information), and then click Next:

Enter the corporate contact information shown here (or your personalized information):

9. Click next, and then click Finish.

FrontPage creates your site using the information you provided and then displays a **list of the tasks that need to be completed to finish the site.**



10. In the Folder List, double-click `index.htm` to open the home page on Page view.

11. Scroll to the bottom of the page, and notice the contact information you provided in the wizard.

E – Mail Aliases

When creating a Web site for a company or organization, it's wise to use generic e-mail addresses (called aliases) in your Web site contact information rather than specific people's e-mail addresses, so the address can stay the same no matter who is actually assigned to respond to the inquiry. For example, if The Garden Company listed its information contact address as that of Karen Berg and then Karen was away for an extended period of time, messages might build up in her mailbox with no one to answer them. Similarly, if all Web site inquiries were directed to David Ortiz and then David left the company, valuable customer contacts could go unnoticed. Using an e-mail alias that automatically forwards received e-mail messages to one or more individuals ensures that the appropriate person always receives customers' questions.

Crating a Web with FrontPage

A web is the collection of a home page and its associated pages, graphics, documents, multimedia, and other files. Webs are stored on a Web server or on a computer's hard drive. FrontPage-based webs also contain files that support FrontPage- specific

functionality and allow webs to be opened, copied, edited, published, and administered with FrontPage.

In the previous procedures, you learned how easy it is to create a Web page with FrontPage. As soon as you start the application, you can start typing and editing, then save the document to your hard drive – much like a word processor. While you can certainly choose to put together an entire Web site like this, it can take a lot of manual work and attention to detail to maintain hyperlinks and source files, and keep your content up to date.

When you save your pages to a web, FrontPage can automatically manage and repair hyperlinks, organize files and folders, maintain dynamic navigation bars, check spelling across all pages in the web, and generate reports that point out problems with pages and files.

To create a new web

1. On the File menu, click Close to close the current page.
2. On the File menu, point to New, and then click Web.

FrontPage displays the New dialog box. Here, you can choose from several web templates and wizards, specify where you want to save your Web, and what you want to call it.

3. Make sure the One Page Web template is selected, and then press TAB.

Pressing the TAB key moves the selection to the field where you specify the name and location of the new web.

4. In the Specify the location of the new Web box, change the suggested name to C:\My Documents\My Webs\Millennium and then click OK.

FrontPage creates a new web named “Millennium” and displays its name in the title bar at the top of the FrontPage application window. Because you’ll be working with several files in your web, FrontPage also displays the Folder List, where you can see the files and folders in your current web, similar to files and folders in Windows Explorer. You’ll learn how to use the Folder List later, in Lesson 2.



Navigation icon

5. Click the Navigation icon on the Views bar.

When you have a web open, the icons on the Views bar let you look at the information in your web in different ways.

Navigation view shows a graphical representation of the structure of your Web site. Because you created a one-page web, FrontPage has automatically designated it as the web's home page – indicated with a small icon of a house. While in Navigation view, FrontPage also displays the Navigation toolbar, which you can drag anywhere on your screen.

Next to the Views bar, FrontPage displays the optional Folder List, just like it did in page view.

In a moment, you'll replace the new, empty home page with the one you created earlier in this lesson. First, however, you'll create the structure for the other four pages that the Millennium Celebration Web will have.

Creating a web structure in Navigation view enables features such as page banners and navigation bars that are automatically updated whenever you change, add, or remove pages in your web. This makes it easy to change things around. You'll learn more about these features later.

To create a navigation structure



New Page button

1. In Navigation view, click the new Page button on the toolbar.

FrontPage creates a new page labeled “New Page 1” below the home page. Pages in Navigation view aren't the actual pages in the current web; they are placeholders that point to them. This way, you can easily experiment with the structure and organization of a web before you create its content.

2. To quickly create the remaining three pages, hold down CTRL on your keyboard and press N three times.

CTRL+N is a keyboard shortcut for the New Page command. FrontPage supports common Window and Microsoft Office accelerator keys that help speed up repetitive tasks. The pages you just created appear below the home page, because the home page was selected when you issued the command.

In Navigation view, the selected page is blue, while others are yellow.

3. With the home page still selected, press TAB.

Pressing the TAB key moves the selection to the next page in the structure and activates the page title for editing.

4. When New Page 1 is selected, type Background and then press TAB.

“Background” is the page title of one of the pages you’ll create for the Millennium Celebration Web. Next, you’ll specify the page titles for the other pages.

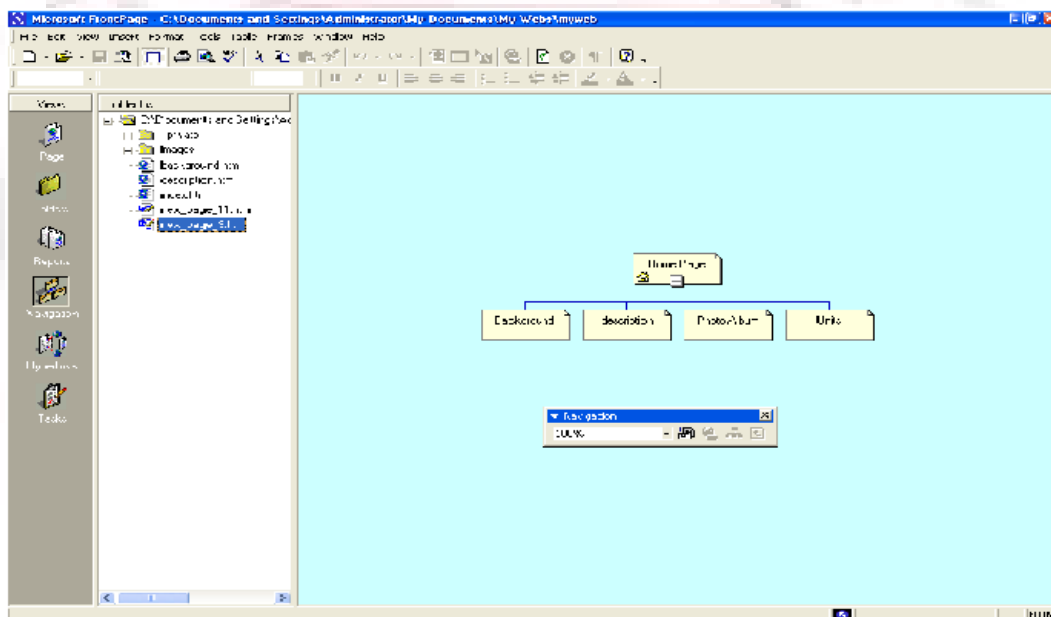
5. When New Page 2 is selected, type Destinations and then press TAB.

6. When New Page 3 is selected, type Photo Album and then TAB.

7. Finally, when New Page 4 is selected, type Links and then press ENTER.

Pressing ENTER after editing a page title saves the new title without selecting another page. To deselect all pages, click anywhere outside the pages in Navigation view.

Your screen should now look like this:



As alternate way to add pages You can add new pages in Navigation view by right-clicking a page and choosing the New Page command from the shortcut menu. This command will create a new page below the selected page.

You can quickly open pages in page view for editing by double-clicking the pages in Navigation view or in the Folder List.

Next, you’ll replace the blank home page FrontPage created from the web template by importing the home page you created and saved to your My Documents folder earlier in this lesson.

To import a page into a web

1. In Navigation view, double-click the Home Page to open it in Page view.

FrontPage opens the blank home page that was created from it in web template.



Folder List button

2. On the toolbar, click the Folder List button to hide the Folder List in Page view.

3. On the Insert menu, click File.

FrontPage displays the Select File dialog box. Here, you can insert Web pages, word-processing documents, text files, and other documents on the current page.

4. In the Select File dialog box, navigate to the My Documents folder.

5. When My Documents is displayed in the Look in list of the Select File dialog box, click the file named homepage, and then click Open.

FrontPage imports your previously saved home page to the current page.



Save button

6. To save the current page to your web, click Save on the File menu, or click the Save button on the toolbar.

FrontPage displays the Save Embedded Files dialog box. Here, you can preview, rename save, and update embedded files that the current web will use.

When you previously saved this page to the My Documents folder on your file system, FrontPage left the two pictures you inserted in their original location – the FrontPage Book folder. The home page merely pointed to the picture files without copying them to the same folder the page was saved to. To keep Web sites portable, however, you should always keep associated pages and files as part of web uses them.

7. In the Save Embedded Files dialog box, click OK.

FrontPage saves the home page as Index.htm and saves copies of the embedded picture files, 2000.gif and Fp2000.gif, to the current web.

Home pages have reserved names FrontPage will automatically name home pages one of two-reserved name that you should not change unless you need to. If you are running local Web server software such as Microsoft Personal Web Server or Internet Information services (IIS) on your computer, the home page will be named Default.htm. If you do not have a local Web server installed, or when you save a page to your local hard drive instead of a web on a Web server, the home page will be named Index.htm.

These names are reserved for home pages because Web browser will automatically look for them when a site visitor types the URL to your Web site without a specific page reference.

Creating Web Content

Now that the home page is part of the current web, you will create the content for the other pages in the Millennium Celebration Web.

To edit the Background page

1. Click the Navigation icon on the Views bar to switch back to Navigation view.

Note that the Folder List now shows the two picture files you saved to the current web. The file Index.htm is the new home page. You can later discard the remaining copy of the home page from your My Documents folder. In the Folder List, the file names of the other pages were automatically derived from the page titles you typed into the pages in Navigation view. For this book, we won't change the names.

2. Double-click the Background page to open it in page view.

This page will provide some background about the new millennium and the Year 2000 for site visitors. For this book, we have provided this text for you, so you can simply insert it on the page without having to type it.

3. When the blank page is displayed in Page view, click File on the Insert menu.

4. In the Select File dialog box, navigate to the folder named program File\Microsoft Office\Office\Book by double-clicking each folder in this path until the Look in box displays the Book folder. If you downloaded the book files from the www.microsoft.com, navigate to the folder named Fptutor\Samples, or to the folder where you placed the files.

5. Next, click the Files of type list and click Text Files (*.txt) to display the text files in the Book folder. FrontPage displays the single file matching the criteria.

6. Click year 2000 in the list, and then click Open.

The text you are inserting isn't saved in HTML format, so FrontPage displays the Convert Text dialog box to let you control how the text will be imported.

7. In the Convert Text dialog box, click Normal paragraphs with line breaks, and then click OK.

FrontPage imports the text file and places it at the insertion point on the Background page.

8. On the toolbar, click the Save button to save changes to the Background page.

Next, you will prepare the page titled Destinations. It will tell site visitors about popular travel destinations that many people will visit to celebrate the new millennium. On this page, you will also provide a feedback form that collects travel ideas from people browsing the Millennium Celebration Web.

To edit the Destinations page

1. On the toolbar, click the Folder List button to show the Folder List in Page view.

2. Double - click destinations.htm in the Folder List to open the page in Page view.

3. Click the Folder List button to hide the Folder List.

4. When the blank page is displayed in Page view, click File on the Insert menu.

5. In the Select File dialog box, click the Files of type list, and then click Rich Text Format (*.rtf) to display the formatted text file in the Book folder.

FrontPage displays the file matching the criteria.

6. Double - click the file events.

Because this type of file contains formatting, FrontPage automatically converts the formatted text to HTML format.

7. On the toolbar, click the Save button to save changes to the Destinations page.

Automatic Spelling Checks

Take a moment to scroll to the top of the Destinations page. In the first paragraph, note the red, wavy underline under the name “Balleny,” the name of an island in Antarctica.

In Page view, FrontPage automatically checks the spelling of text you type on the current page, just like Microsoft Word, PowerPoint, and other Microsoft Office applications do. An underlined word doesn’t necessarily mean the word is spelled incorrectly. FrontPage may simply prompt you to verify unknown or suspected words, which happens most commonly with name of people and places.

If you know that the spelling of a suspected word is correct, you can either choose to ignore such words and keep them unchanged, or add them to a custom dictionary that FrontPage will keep for subsequent spelling checks. If the word is indeed misspelled, you can quickly insert the corrected spelling by right-clicking the suspected word and selecting a suggested correction.

There are three ways FrontPage can check spelling for you:

- Automatic spelling check as you type text
- Manual spelling check of the current page
- Cross-web spelling checks of all pages in a web

Page-based spelling checks are available in Page view by right-clicking suspect words or by clicking the Spelling command on the Tools menu. Cross- web spelling checks are available in every web view.

Spelling checks are important if you web content to give visitors a professional impression. If words are misspelled features in FrontPage on a single page, people might question the accuracy of your entire Web site content. The flexible spelling features in FrontPage give you the option of checking spelling page by page as you create and edit content, or doing it all at once, just before you publish your web to the World Wide Web.

Designing a Page

The Background page will inherit its formatting from a graphical theme that you will apply to the Millennium Celebration Web later, in Lesson 2. The Destinations page, however, requires some more design work. To help the reader differentiate the paragraph

headings, list of travel destinations, and event details that the text on this page talks about, you will add some pictures, format paragraph styles, and create a bulleted list.

To create a bulleted list

1. With the Destinations page still displayed in Page view, find the words “Times Square in New York.”
2. Use the scroll bar to bring the entire list of destinations into view, beginning with “Times Square in New York” and ending with “The Acropolis in Greece.”
3. Click just to the left of the letter T in “Times Square,” hold down SHIFT, click just after the word “Greece,” and then release SHIFT.

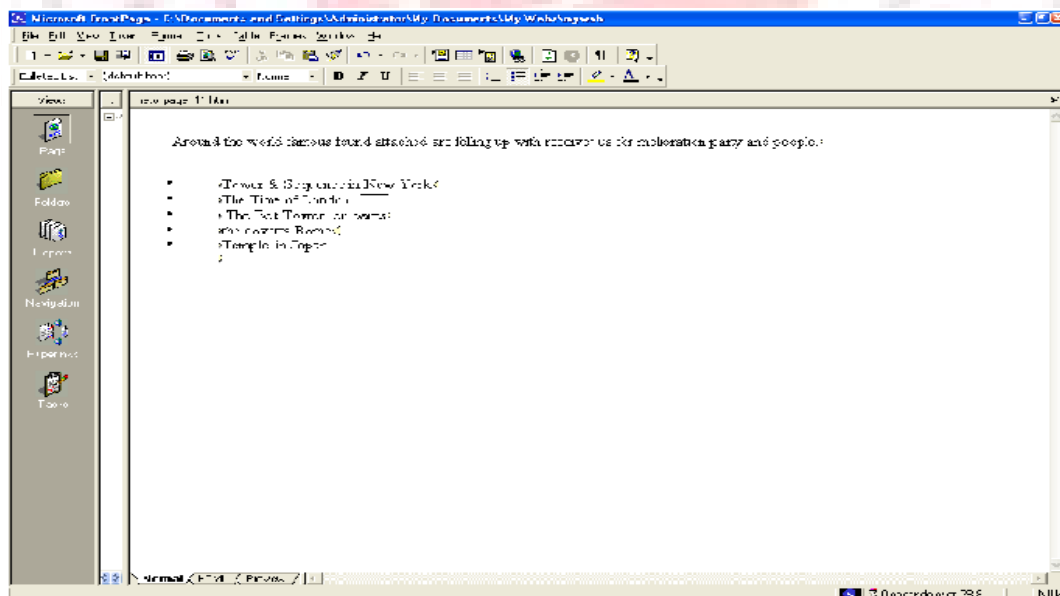


Bullets button

4. On the toolbar, click the Bullets button.

FrontPage converts the selected text to a bulleted list.

5. Click anywhere on the page to deselect the list.



You can also create numbered lists with FrontPage. When you add new items to a numbered list, FrontPage automatically numbers them sequentially. You can add to bulleted and

numbered lists by pressing ENTER after an item in the list. To end a list, press ENTER twice after typing the last list item. Next, you will place four pictures on the current page and use positioning features in FrontPage to align the pictures with the paragraphs they are associated with. This will create a more interesting page layout.

To position pictures with text

1. With the Destinations page still displayed in Page view, scroll down to the words “France and England” in the paragraph just following the list.



Insert Picture From File button

2. Click just to the left of the letter F in “France,” press HOME, and then click the Insert Picture From File button on the toolbar.

When you last inserted a picture, you did not have a web open, and FrontPage automatically displayed the Selected File dialog box. Now that a web is open, FrontPage assumes you want to work with pictures that are already part of your web, and therefore displays the Picture dialog box.



Select File button

3. Because you haven't yet added the picture you want to the current web, click the Select File button in the Picture dialog box.

FrontPage displays the Select File dialog box.

4. Click the file named paris, and then click OK.

FrontPage inserts a picture of the Eiffel Tower in Paris just before the current paragraph.

5. Next, click the picture of the Eiffel Tower in Paris just before the current paragraph.

6. On the Format menu, click Position.

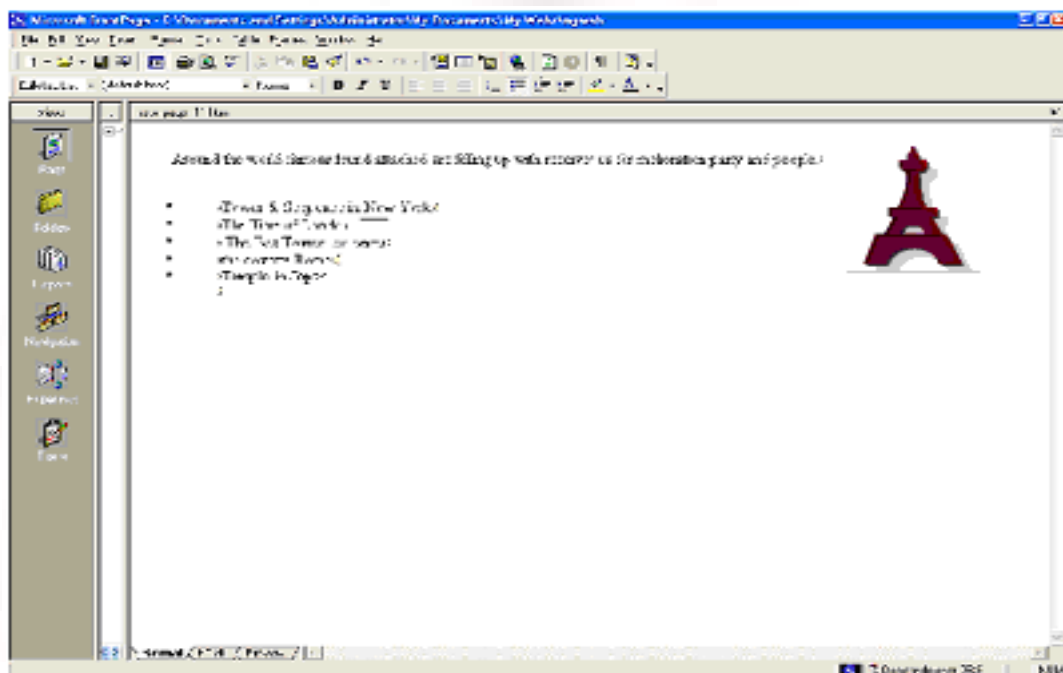
FrontPage displays the Position dialog box.

7. Under wrapping style, click right, and then click OK.

The picture is aligned with the right margin of the current page and the text now flows around it.

8. On the toolbar, click the Save button to save changes to the Destinations page.
9. In the Save Embedded Files dialog box, click OK.

Your page should now look like this:



You can either place pictures one by one in this way, or you can import all the pictures you will use on your pages all at once. While importing single files is done in Page view, inserting a group of files or entire folders is done in Folders view.

To add a group of files to the current web



Folder icon

1. Click the **Folders** icon on the Views bar to switch to Folder view.
2. **On the File menu**, click Import.
3. In the **Import** dialog box, click **Add File**.

4. in the **Add File to Import List** dialog box, navigate to the folder named Program Files\Microsoft Office\Office\Book by double-clicking each folder in this path until the Look in box displays the Book folder. If you downloaded the book files from the www.microsoft.com, navigate to the folder named Fptutor\Samples, or to the folder where you placed the files.

5. Next, click the Files of type list, and then click GIF and JPEG (*.gif, *.jpg) to display all picture files in the Book folder.

6. Click the file named firewks 1 in the list to select it.

7. Next, hold down CTRL, and while doing so, click to select the files named firewks2, firewks3, firewks4, Japan, London, and sanfran.

FrontPage supports standard Window file selection methods.

8. When the files are selected in the Add File to Import List dialog box, release CTRL, and then click Open.

FrontPage adds the pictures you selected to the list in the Import dialog box.

9. Click OK to import the listed files to the current web.

Now that the remaining pictures are added to your web, it's time to finish the layout of the Destinations page.

To finish the page layout

1. Click the Page icon on the Views bar to return to Page view.

2. Locate the sentence beginning with "England is planning to reconstruct", click just to the left of the letter E in "England," and then click the Insert Picture From File button on the toolbar.

3. In the Picture dialog box, note that your previously imported pictures are now available, and then select the file london.gif from the list.

When you click a single picture file in the Picture dialog box, FrontPage displays a preview of the picture, so you can make sure it's the one you want to insert. This is one of the benefits of adding all your pictures to the web before inserting them on your pages.

The Picture dialog box also links to the clip art gallery that is included with FrontPage. And if you have a scanner or a digital camera, you can click the Scan button in this dialog box to acquire original pictures from those sources.

4. Click OK to insert the picture



Align Left button

5. Click the picture of the Tower of London you just inserted, and then click the Align Left button on the toolbar to position the picture in the left margin and make the text wrap around it.

6. Next, scroll down to the sentence beginning with “World-famous Times Square....,” click just to the left of the letter W in “World-famous,” and then click the Insert Picture From File button on the toolbar.

7. Select the file sanfran.gif from the list, and then click OK.



Align Right button

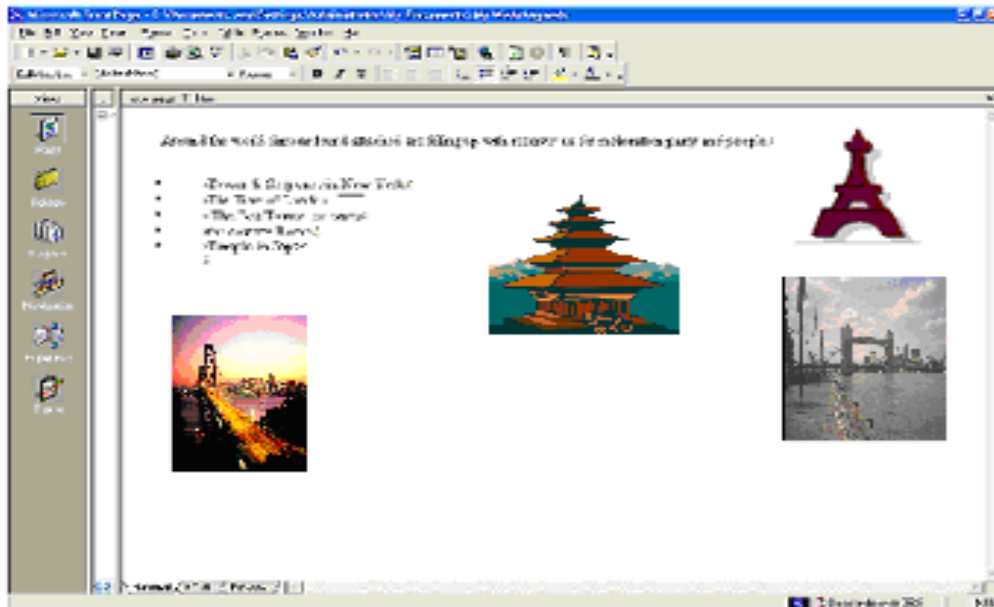
8. Click the picture of the Golden Gate Bridge, and then click the Align Right button on the toolbar to position the picture in the right margin and make the text wrap around it.

9. Finally, scroll down to the sentence beginning with “ A travel agency in Japan ...,” click just to the left of the letter A at the beginning of the sentence, and then click the Insert Picture From File button on the toolbar.

10. Double-click the file japan.gif on the list.

11. Click the picture of the Japanese temple, and then click the Align Left button on the toolbar to position the image in the margin and make the text wrap around it.

Your page should now look like this:



Positioning pictures and other page elements around text on your page makes for a more interesting design, much like pages in a magazine or newspaper. By positioning pictures in the margin, your page layout will be preserved even when the page is viewed at a different screen size and resolution in a Web browser.

To finish the Destinations page, you will create a feedback form so that you can interact with site visitors who want to participate. A feedback form can be used to collect comments and information from people visiting your Web site.

To create a feedback form

1. In Page view, press CTRL+END to quickly jump to the bottom of the current page, or scroll all the way down using the scroll bar.
2. On the new, blank line, type Tell Us Where You'll Be! And then press ENTER.
3. On the next line, type Do you have great travel plans for a millennium party? Then tell us about them below!
4. Press ENTER to create a new line.
5. On the Insert menu, point to Form, and then click Form.

FrontPage inserts a new form on the current page. The dashed lines indicate the form's boundary. By default, a new form contains Submit and Reset push buttons. Next, you

will customize the default form by adding form-fields and form-field labels, to let site visitors know what kind of information you want them to enter into the form

To customize the form



Center button

1. With the cursor still positioned to the left of the submit button, click the Center button on the toolbar, and then press ENTER to add some space to the form.

2. Press the UP ARROW key on your keyboard to return the cursor to the beginning of the form.

FrontPage moves the cursor to the middle of the first line of the form.

3. On the first line, type Your Name: and then press SHIFT+ENTER.

Holding SHIFT while pressing ENTER creates a line break. Line breaks are useful for spacing lines of text more closely together than standard paragraph spacing.

4. On the Insert menu, point to Form, click One-Line Text Box, and then press ENTER.

FrontPage inserts a one-line text input field into the form.

5. On the next line, type Your E-mail Address: and then press SHIFT-ENTER.

6. On the Insert menu, point to Form, click One-Line Text Box once more, and then press ENTER.

7. On the next line. Type Your travel plans are: and then press SHIFT-ENTER.

8. On the Insert menu, point to Form, and then click Scrolling Text Box.

FrontPage inserts a scrolling text input field into the form.

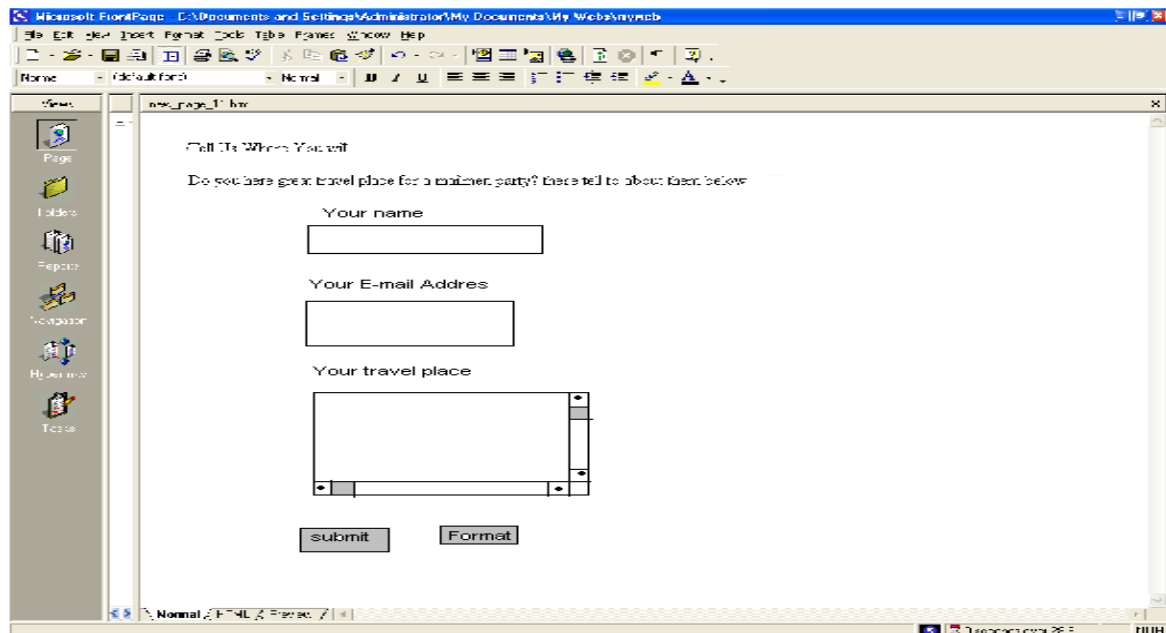
9. Double-click the scrolling text box you just inserted.

FrontPage displays the Scrolling Text Box Properties dialog box. Here, you can change the appearance of the text box.

10. In the Scrolling Text Box Properties dialog box, change the Width in characters to 30 and the Number of lines to 5, and then click OK.

11. On the toolbar, click the Save button to save changes to the Destinations page.

Your page should now look like this:



Good work! The feedback form is finished and so is the Destinations page. In the next part of the lesson, we'll add the last two pages – an online photo album and a list of links to your favorite sites on the World Wide Web.

Creating a Photo Album

The World Wide Web has a graphical interface, so it's no surprise that the most popular Web sites have pictures to look at. Scanners and digital cameras have become much more affordable, and many photo-developing places now offer to put your pictures on a CD-ROM so you can share them online.

While working with the graphics and clip art in the previous procedures, you saw how easy it is to place pictures on Web pages using FrontPage. In this part of the lesson, you'll create an online photo album of actual photographs. For the Millennium Celebration Web, you'll share some great photos of fireworks.

To edit the Photo Album page

1. On the toolbar, click the Folder List button to show the Folder List in Page view.

2. Double-click photo_album.htm in the Folder List to open the page.

3. Click the Folder List button to hide the Folder List.

4. When the blank page is displayed in Page view, type New Year's Fireworks on the first line, then press ENTER.

5. On the next line, type Here are some great pictures from past New Year's celebrations. Click each thumbnail to see the full-size picture, then use your Web browser's Back button to return to this page.

6. Press ENTER twice to create some space.

Next, you'll place four pictures on the current page. They are already part of the web because you imported them in the previous procedure.

7. On the toolbar, click the Center button.

8. Click the Insert Picture From File button.

9. In the Picture dialog box, select firewks.jpg, and then click OK.

FrontPage inserts the first of the four fireworks pictures you will place on this page. Next, insert the other three pictures, one after the other. Don't worry about their large size and proper placement just yet.

10. Repeat steps 8 and 9 with the picture files named firewks2.jpg, firewks3.jpg and firewks4.jpg. The remaining pictures are placed on the page, one after the other.

11. On the toolbar, click the Save button to save changes to the Photo Album page.


Having lots of large pictures on your Web page for people to look at is great, but not everyone has a fast connection to the Internet. Over a dial-up connection with a modem, pages with large pictures can take a long time to download.

Imagine changing the channel on your TV and having to wait several minutes before you see what's playing on that channel. It's no different on the World Wide Web. There are many other "channels" or Web sites to look at. No matter how interesting your site may be, people may quickly lose interest if it takes too long to download.

On the status bar, FrontPage automatically displays the estimated time it will take for the current page to download over the Internet when the page is opened in a Web browser. The default measurement assumes that your site visitors will have a 28.8K modem connection. You can adjust this measurement for other common connection speeds by right-clicking the red hourglass icon and choosing another connection speed from the shortcut menu.

For this book, we'll leave the download speed at 28.8K. On the current page, you can see that the estimated download time for the Photo Album page is 97 seconds. This means that people who will visit the Millennium Celebration Web will have to wait a minute and a half before they can see the four pictures you've just inserted. That's quite a long time to wait for just four pictures.

By creating thumbnails – small preview images of pictures – you can give your visitors the choice of whether they want to spend time downloading the full-size pictures on your page. FrontPage makes creating thumbnails easy with the Auto Thumbnail tool.

 **Setting thumbnail preference** you can change default options for the thumbnails FrontPage creates of your pictures. On the **Tools** menu, click **Page Options**, and then click the **AutoThumbnail** tab. Here, you can specify the size of automatic thumbnails FrontPage creates, whether each thumbnail should have a border to indicate the presence of its associated hyperlink, and whether thumbnails should be displayed with a beveled edge to simulate a button.

Before proceeding with the next procedure, change the default setting from **Width** to **Height** in the **AutoThumbnail** tab of the **Page Options**.

To create thumbnails of pictures

1. With the Photo Album page still displayed in Page view, press CTRL+HOME to quickly jump to the beginning of the current page, or scroll all the way up using the scroll bar.
2. Click the first fireworks picture to select it.

FrontPage displays the Pictures toolbar below view.



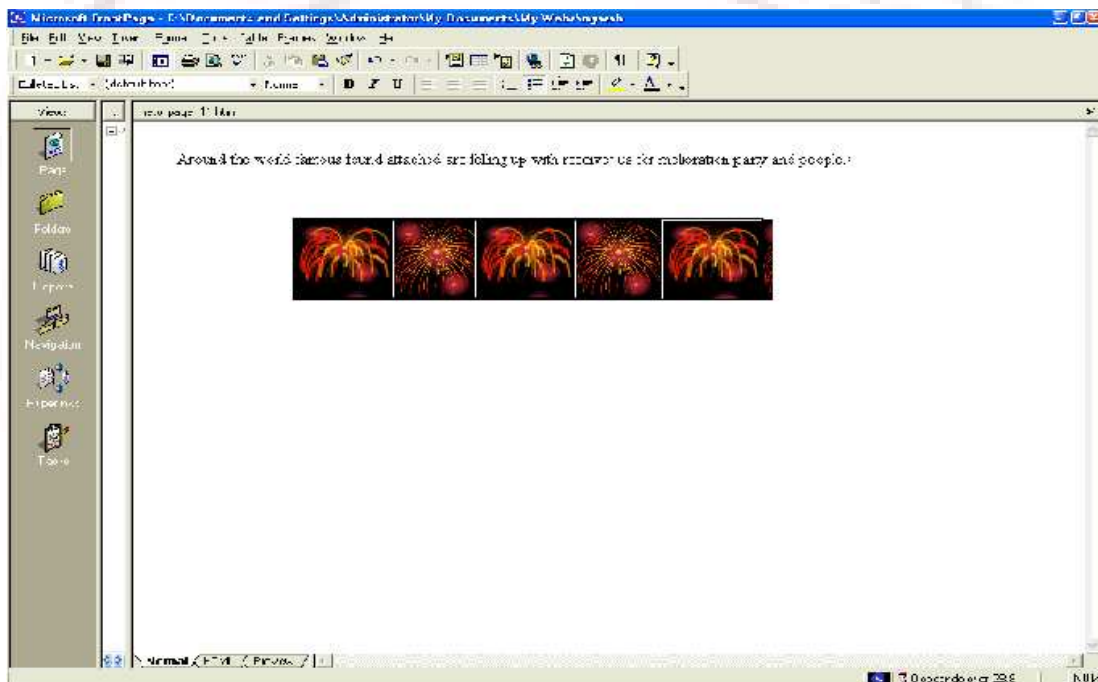
Auto Thumbnail button

3. On the Picture toolbar, click the Auto Thumbnail button.

FrontPage creates a thumbnail of the selected picture and adds a blue border to indicate that it contains a hyperlink to the original picture in your web. When site visitors visit this page, they can click each thumbnail to download the full-size pictures.

4. Repeat steps 2 and 3 for the other three pictures on the Photo Album page.
5. Click the second fireworks thumbnail to select it. Drag one of the top corner handles to make this thumbnail the same height as the first thumbnail. Repeat this step for the third fireworks thumbnail.
6. Click anywhere on the page to deselect the last thumbnail.

your page should now look like this:



When FrontPage creates thumbnail, it doesn't actually modify the original picture files in any way. Instead, it quickly makes a copy of each picture, resizes it, downsamples the display resolution of the picture, inserts a hyperlink pointing to the original picture file, and adds a border around the thumbnail to indicate the presence of a hyperlink.

You can see why the Auto Thumbnail button is a real timesaver. Doing all of that manually for each picture could take a while.

7. On the toolbar, click the Save button to save changes to the Photo Album page.

Because FrontPage made small copies of the pictures that are represented by a thumbnail, it needs to save the thumbnails to the current web. The names of the thumbnail picture files are the same as the original pictures, but FrontPage adds a “_small” suffix to each file name for easy identification.

8. In the Save Embedded Files dialog box, click OK.

FrontPage saves the four thumbnails to the Millennium Celebration Web. Because you have an even number of pictures, you can arrange them a little better than all on one line. You can treat thumbnails like other pictures on your pages and move them where you want them.

To finish the Photo Album page

1. Click the first thumbnail and then press HOME to move the cursor to the left of it.

2. Press the RIGHT ARROW key on your keyboard to move the cursor between the first and second thumbnail.

3. Press the TAB key.

FrontPage creates space between the first and second thumbnail.

4. Next, press the RIGHT ARROW key again to move the cursor between the second and third thumbnail.

5. Press ENTER.

FrontPage moves the third and fourth thumbnail to the line.

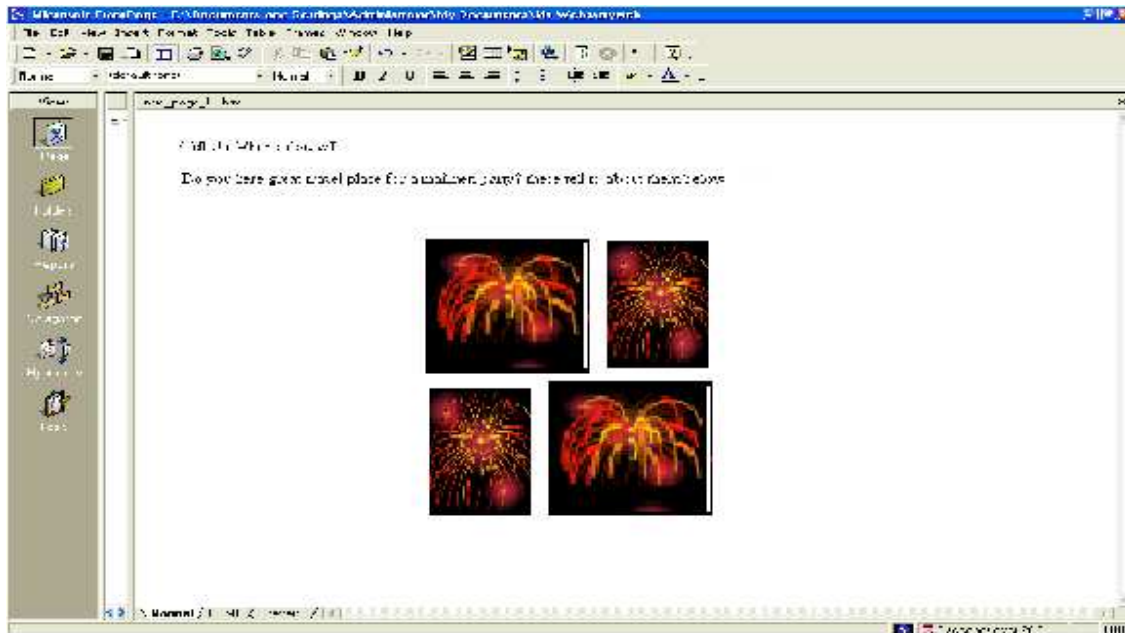
6.Press the RIGHT ARROW key once more to move the cursor between the third and fourth thumbnail.

7. Press the TAB key.

FrontPage creates space between the third and fourth thumbnail.

8. On the toolbar, click the Save button to save changes to the Photo Album page.

Your page should now look like this:



Now take another look at the FrontPage status bar. The Photo Album page previously would have taken 97 seconds to download.

After replacing the large pictures with thumbnails, the Photo Album page now only takes 4 seconds to download. That's much better!

Linking to Other Web Sites

Now only the Links page remains to be edited. For this book, this page will contain a list of text hyperlinks to some other sites on the World Wide Web. When you create your own webs, you can create hyperlinks pointing to other Web sites that relate to the subject matter of your own pages. This lets visitor's browser to similar sites without having to search for them.

To begin the Links page

1. On the toolbar, click the Folder List button to show the Folder List in Page view.
2. Double-click links.htm in the Folder List to open the page in Page view.
3. Click the Folder List button to hide the Folder List again.
4. when the blank page is displayed in Page view, type Links to My Favorite Sites and then press ENTER.

Next, you will create a simple text animation of the paragraph heading. FrontPage includes a collection of fun text effects that you can easily apply to text headings or entire paragraphs.

To create a dynamic text effect

1. On the Links page, click anywhere in the text Links to My Favorite Sites.
2. On the Format menu, click Dynamic HTML Effects.

FrontPage displays the DHTML Effects toolbar. Here, you make sequential selections that will create a simple dynamic HTML (DHTML) script to animate the text when it is displayed in a Web browser

Dynamic HTML is an extension of the HTML lets you create presentation effects for text and objects, much like in a Microsoft PowerPoint slide. Using the DHTML Effects toolbar, you can add simple effects to your pages without the need to know programming.

3. In the On list, click Page load.

This will instruct the Web browser to begin the effect when the page loads.

4. In the Apply list, click Hop.

FrontPage applies the Hop effect. In a Web browser, this effect will cause each work to bounce onto the page.

5. Click the Close box in the upper right corner of the DHTML Effects toolbar.

The DHTML Effects toolbar closes and the dynamic text effect is indicated in Page view with light blue shading.



Previewing text effects if you have Microsoft Internet Explorer installed, you can click the **Preview** tab in Page view to see the effect as it will appear in a Web browser. To change the effect, click **Normal** to return to page view, select the text containing the effect, click **Dynamic HTML Effects** on the **Format** menu, and then apply the new effect you want.

Next, you will add text hyperlinks pointing to other sites on the World Wide Web. With FrontPage, you can create text a hyperlink in a number of ways' which you will learn next. When you create your own webs, you can create hyperlinks using your preferred method.

To create hyperlink from text

1. On the Links page, press the DOWN ARROW key, and then type MSN – The Microsoft Network and then press ENTER.
2. Click and drag the mouse over the words you just typed to select them.
3. On the Insert menu, click Hyperlink.

FrontPage displays the Create Hyperlink dialog box. Here, you specify the target of the hyperlink you are creating. This can be a page of a file in your web, on your local file system, on a Web server, or on another site on the World Wide Web.

4. In the URL box, type www.msn.com immediately after the http:// prefix that FrontPage has provided for you, and then click OK.

“HTTP” stands for Hypertext Transfer Protocol. That is the Internet protocol that allows World Wide Web browsers to retrieve information from Web servers. The text www.msn.com “ is the URL of MSN, the Microsoft Network.

5. On your keyboard, press the DOWN ARROW key to deselect the text. The words “MSN- The Microsoft Network” have changed from black default text to blue text, and the words are now underlined to indicate the presence of a hyperlink. When this page is displayed in a Web browser, clicking this hyperlink will retrieve and display the MSN home page. Before creating the next hyperlink, you'll insert a special character symbol to indicate a trademark on the current page.

To insert special characters or symbols

1. Click the mouse just after the letters MSN in the hyperlink you just created,
2. On the Insert menu, click Symbol
FrontPage displays the Symbol dialog box. Here, you can select and insert special characters at the insertion point. You can insert multiple symbols while this dialog box is displayed.
3. In the Symbol dialog box, select the trademark (TM) symbol, click Insert, and then click Close.

FrontPage inserts the trademark symbol after the letters MSN. You can use the Symbol command to insert characters that you may not be able to type directly with your keyboard. Next, you will create an automatic hyperlink. This method of creating hyperlinks is quick and easy, because it lets you bypass the Create Hyperlink dialog box.

To create an automatic hyperlink

1. On the Links page, press the DOWN ARROW key, type <http://www.yahoo.com>, and then press ENTER.

Yahoo! is a popular internet service that lets you look for information on the World Wide Web using search keywords and subject categories. As soon as you press ENTER, the URL you typed changes from black to blue text and is underlined to indicate the presence of a hyperlink. Like other Microsoft Office applications, FrontPage supports automatic hyperlink creation.

Since a URL by itself is not always very descriptive, however, you'll want to change it to the name of the site that the hyperlink points to. You can overwrite the text without erasing the hyperlink.

2. Using the mouse, click and drag over the URL you just typed to select it.
3. When the URL <http://www.yahoo.com> is selected, type Yahoo! to replace the selected text. The hyperlink still points to the same URL, but it is now labeled with the site's name. Next, you'll create a hyperlink using your Web browser. This method of creating hyperlinks is the most accurate. Because you actually visit the page the hyperlink will point to before creating the hyperlink. In addition, FrontPage copies the URL from the Web browser address field, so once the address is verified, you don't have to type it again.

Note

If you do not have access to the World Wide Web while taking the FrontPage Book, skip the following procedure and practice these steps the next time you're connected to the Internet.

To create a verified hyperlink

1. Press the DOWN ARROW key to move the insertion point to the blank line below the previous hyperlink.
2. Type Microsoft Front Page 2000 and then press ENTER.

3. Click and drag the mouse over the words you just typed to select them.

Hyperlink button

4. On the toolbar, click the Hyperlink button.

FrontPage displays the Create Hyperlink dialog box.

Web Browser button

5. In the Create Hyperlink dialog box, click the Web Browser button.

FrontPage starts your Web browser. When you visit the page that the hyperlink should point to and then switch back to FrontPage, the URL box will contain the address of the target page.

6. In your Web browser's Address (or Location) box, type <http://www.microsoft.com/frontpage> and then press ENTER.

7.

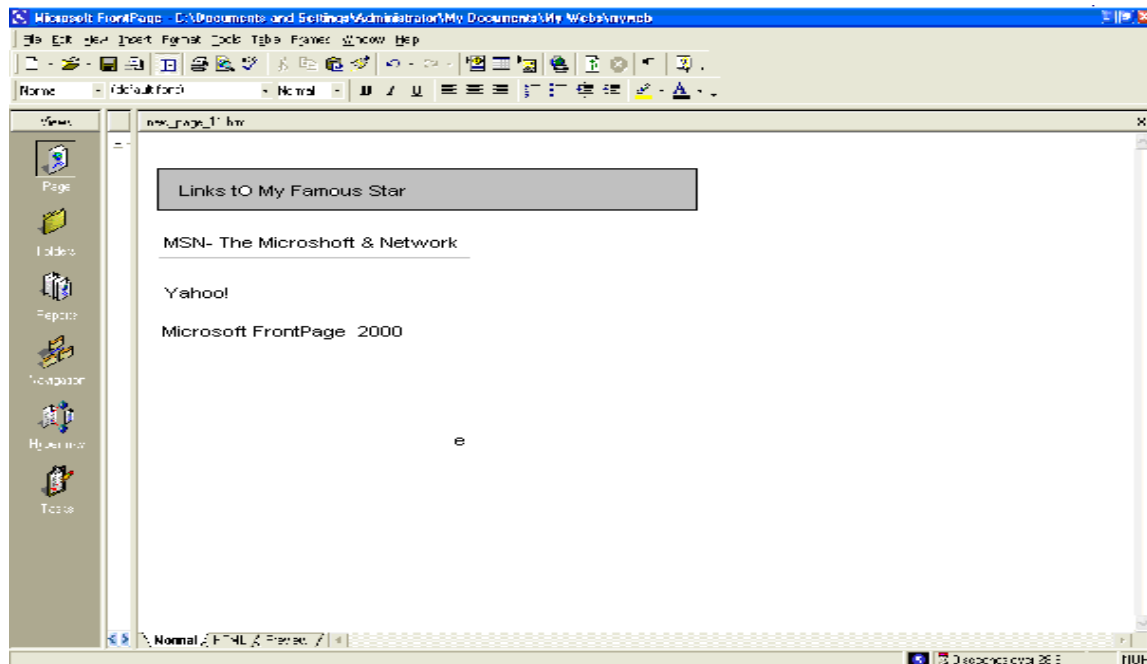
8. On your keyboard, press ALT+TAB (or click the Microsoft FrontPage taskbar button on the Windows taskbar) to switch back to the Create Hyperlink dialog box. The URL of the Microsoft FrontPage home page is now entered into the URL box in the Create Hyperlink dialog box.

9. Click OK.

10. On your keyboard, press the DOWN ARROW key to deselect the text. The words "Microsoft FrontPage 2000" are now underlined to indicate the presence of a hyperlink.

11. On the toolbar, click the Save button to save changes to the Links page.

Your page should now look like this:



Formatting Paragraph Headings

Now that you have successfully created the content on all the pages in your web, you'll apply paragraph and font formatting to make the paragraph headings on all pages look consistent.

To apply paragraph style to headings

1. On the Window menu, click index.htm.

FrontPage brings the home page back into view. When you have more than one page open in Page view, you can use the Window menu to switch between them. The current page will always be listed at the top of this menu.

2. Click anywhere in the heading Welcome to my Web site! at the top of the page.

3. Click the Style list on the Formatting toolbar, and change Normal to Heading 3

FrontPage applies the Heading 3 style to the current line of text. The size of the text isn't affected, but text is now bold. Heading styles in the Style list are based on universal HTML standards. A level 1 heading is the largest possible text style for Web pages, and a level 6 heading is the smallest.

4. On the toolbar, click the Save button to save the home page.

5. On the Window menu, click background.htm

FrontPage brings the Background page back into view.

6. Click anywhere in the heading The New Millennium at the top to the page.

7. Click the Style list on the toolbar, and then click Heading 4.

8. Repeat steps 6 and 7 with the heading What's in a Number? on the lower half of the Background page.

9. On the toolbar, click the Save button to save changes to the Background page.

To repeat paragraph formatting with the Format Painter

FrontPage provides a convenient way to copy the formatting styles from one selection of text to another with the push of a button. You'll practice how to do this on the Destinations page.

1. On the Window menu, click destinations.htm. FrontPage brings the Destinations page back into view.

2. Click anywhere in th

3. Click the Style list on the toolbar, and then click Heading4.



Format Painter button

4. When the style has been applied, click anywhere in the heading, and then click the Format Painter button on the toolbar.


5. Using the scroll bar, scroll down to the heading Tell Us Where You'll Be.



Format Painter pointer

6. Click the mouse to drag the Format Painter pointer over the words Tell Us Where You'll Be, and then release the mouse button. FrontPage copies the formatting from the previous heading and applies it to the current heading. The Format Painter tool is particularly useful when you want to easily duplicate several concurrent formatting choices from one selection of text to another.

7. On the toolbar, click the Save button to save the Destinations page.

 **Copying formatting to multiple locations** you can repeat text formatting over more than one selection of text by double-clicking the **Format Painter** button after selecting the format you want to copy select the various text locations you want to copy the formatting to, then click the **Format Painter** button again when you're finished.

Next, you'll finish formatting the page heading on the remaining two pages.

To finish applying paragraph styles to headings

1. On the Window menu, click photo_album.htm.
FrontPage brings the Photo Album page back into view.
2. Click anywhere in the sentence New Year's Fireworks at the top of the page.
3. Click the style list on the toolbar, and then click Heading 4.
4. On the toolbar, click the Save button to save the photo Album page.
5. On the Window menu, click links.htm.
6. Click anywhere in the sentence Links to My Favorite Sites.
7. Click the Style list on the toolbar, and then click Heading4.
8. On the toolbar, click the Save button to save the Links page.

To close Microsoft FrontPage

- On the File menu, click Exit.

The FrontPage application closes.

Chapter 14

Designing and Publishing a Web Communicating with Your Visitors

Very few people create a Web site with the expectation that no one will ever see it. Web sites are designed to be published and viewed by other people. The number of visitors, or hits, a Web site receives is a common benchmark of its success. In fact, in the late 90s a lot of dot-com companies made money not by selling products or services, but by selling advertising space based on the number of visitors that passed through their sites. Assuming that you are interested in developing a loyal group of visitors who will associate your site positively with your company or organization, you need to spend some time thinking about how you will communicate with those visitors.

Communication is a two-way process: you give your visitor information, and if your Web site meets their needs, they give information back to you. You can attract repeat traffic by ensuring that your information is timely and easy to find. You can solicit information from your visitors by providing.

In this chapter, you will first insert information that is to be displayed on a Web page for only one month. Then you will create a feedback form so that visitors can give you specific information you want, as well as suggestions and requests. Finally, you will look at a simple way to enable your visitors to search for information on your site.

Automatically Updating Information

Microsoft Office FrontPage 2003 includes a set of Web components, called Include content components, that you can use to create links to the text or graphics you want to display on a Web page, rather than inserting them directly. Why would you want to do that? Suppose The Garden Company frequently updates the document in which it maintains its calendar of classes and other events. If the company also displays the calendar information on its Web site, it has to update not only the document but also the Web page. By displaying the calendar document as included content, the company can maintain the calendar in just one place and know that the Web page always displays the most up-to-date information. Because included

content is automatically updated when ever an included page or graphic is updated, a writer or graphic artist can make changes to Web site control without having to open or edit any Web pages. Included content simplifies the process of reusing content across multiple pages or sites.

FrontPage offers five types to included content:

- **The Substitution component** associates names, called variables, with text. In the Web Settings dialog box, you can assign a variable to a block of text and then insert the variable on a Web page instead of inserting the text itself. For example, you might assign a variable named Disclaimer to a block of text that consists of a 200- word legal disclaimer, and then insert the variable in a Substitution component on every page of your Web site. If you need to change the wording of the disclaimer, you change it once in the Web Setting dialog box, and it is instantly updated on every Web page.

- **The page component** displays the contents of a file wherever it is inserted.

- **The Page Based On Schedule components** display the contents of a file for a limited period of time. You can stipulate the beginning and end dates or times of the period during which the file should be displayed. You can also specify alternate content that should be displayed outside of the scheduled time period.

- **The Picture Based On Schedule component** has the same function as the Page Based On Schedule component, but it works with graphics files.

The Page Banner component is used to create a page title consisting of either text or graphics that appears on every page where the component is inserted. This is the equivalent of inserting a page banner from the Insert menu. In this exercise, you will include a page and a scheduled picture in an existing Web site.

1. In the Folder List, double-click specials.htm to open the file in the Page view editing window. The page is currently empty.

2. On the Insert menu, click Web Component to open the Insert Web Component dialog box.

3. In the Component type list, click Included Conte

4. In the Choose a type of content list, click Page, and then click Finish.

The Include Page Properties dialog box appears.

5. Click the Browser button.

Tip Unlike most Browser dialog boxes, this one limits you to browsing the current Web site.

6. Click `montly_specials.htm`, and then click OK to return to the Include Page Properties dialog box

FrontPage enters the file name in the “Page to include” box.

7. Click OK to close the Include Page Properties dialog box and insert the included component.

The content of the Monthly Specials page is inserted in the Specials page.

8. Click anywhere on the page.

The included content is selected as a single block and cannot be edited from the Specials page.

9. At the bottom of the Page view editing window, click the Show Code View button. Show to view the HTML code that makes up this page.

Instead of displaying the page content within the `<body>` tags, you will see the following code, which specifies where the contents of the `monthly_specials.htm` file should be displayed.

10. At the bottom of the Page view editing window, click the Show Design View button to return to Design view.

11. On the Standard toolbar, click the Save button to save your change to the specials page.

12. Open the `monthly_specials.htm` file in Design view.

The file itself contains no special formatting. The formatting of the host page is applied to the file content when it is displayed there.

13. Select the word August, and replace it with September.

14. Save the file and then click the `specials.htm` page tab to switch back to the Specials page.

The Specials page reflects the changes you made to the included content.

15. Open products.htm in Design view.
16. Press H+B to move the insertion point to the end of the page
17. On the standard toolbar, click the Web component button to display the Insert Web Component dialog box.
18. In the Component type list, click Included Content.
19. In the Choose a type of content list, click Picture Based On Schedule, and then click Finish.
20. Click OK to insert the Web component.

If your system date is currently set to any date in September, you will see a graphic titled September Savings. Otherwise, you will see a graphic titled We Love Great Prices!

21. Click the Scheduled Picture component to select it, and then on the Standard toolbar, Align Right click the Align Right button.

The graphic is aligned with the right edge of the page.

22. Right-click the component, and click Scheduled Picture Properties on the shortcut menu.

23. Set the Starting and Ending dates for today's date, leave the times as they are, and click OK.

The September Savings advertisement is now visible and will be displayed for the entire day.

Allowing for Visitor Input

Providing visitors with information is probably the most common reason for building a Web site. But after you've attracted visitors' attention, you would be missing a prime opportunity if you did not also provide a mechanism for visitors to send information back to you.

The value of the information you receive from your visitors depends to a large extent on the way you present your request. Visitors are unlikely to fill out surveys and provide personal information that might be sold to mailing- list vendors without some significant reward. To get them to take the time and the risk of giving you information, you have to appeal

to their best interests. They are more likely to be willing to help you gather information if they have an interest in your specialty area and if your Web site makes an effort to provide them with useful information that goes beyond your money- making endeavors.

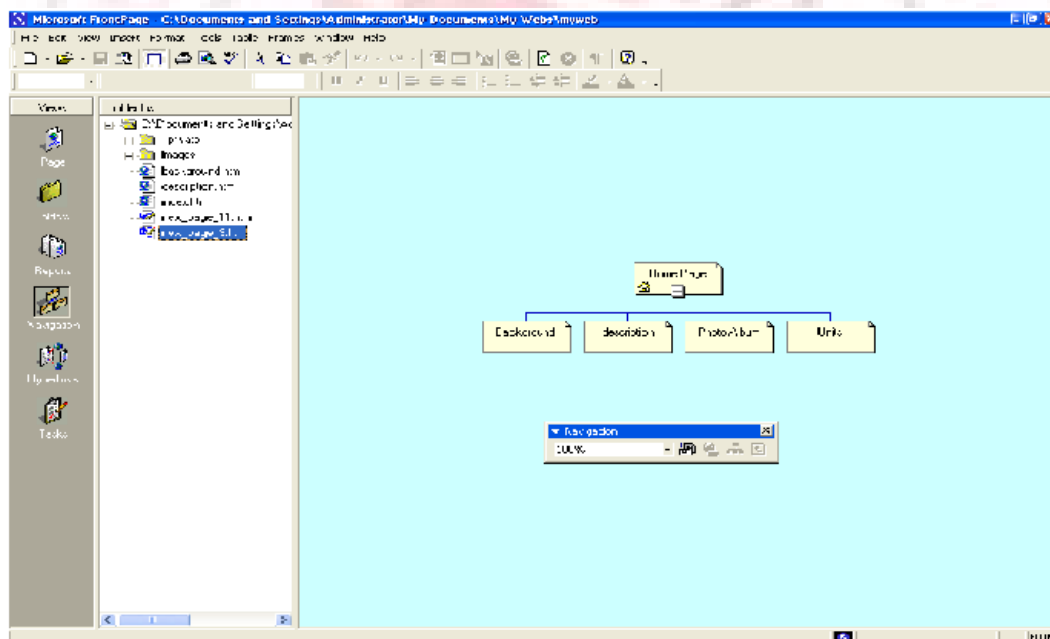
Only you can decide how you want to present your request for information to your visitors, and only you can ensure that your Web site offers plenty of value in return. But FrontPage can help by providing ready- made templates for a feedback form and confirmation page that you can use as is or customize to meet your needs.

In this exercise, you will create and personalize a feedback form and its accompanying confirmation form for the Garden Co Web site, and you will position them appropriately within the site's navigational structure.

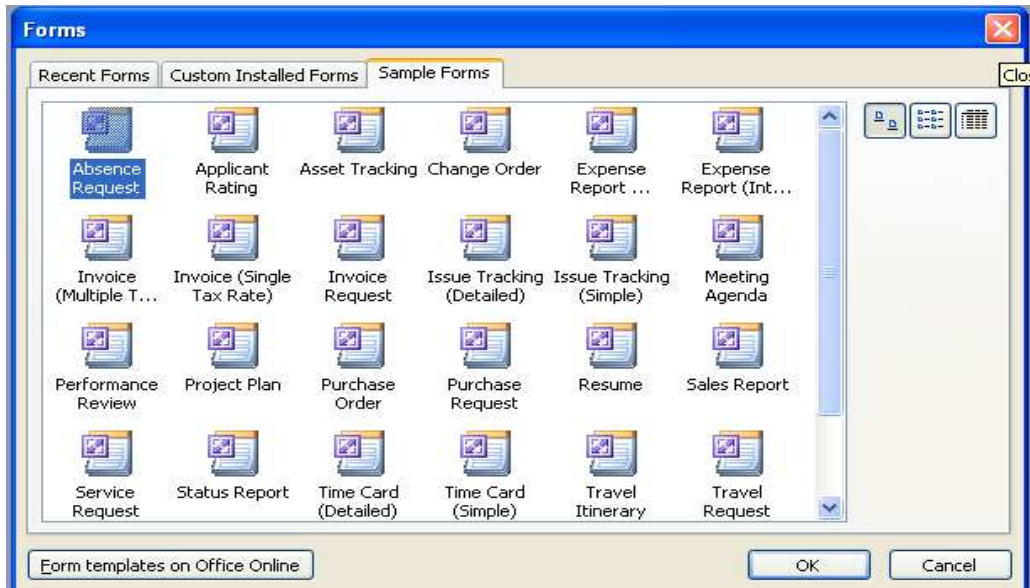
1. Familiarize yourself with the site in Navigation view. If the site doesn't fit on your screen, change the zoom level or switch to Portrait orientation.

2. On the File menu, click New.

The New task pane appears.



3. In the New page area, click More page templates to open the page Templates dialog box.



4. On the General tab, click Feedback From, and then click OK. FrontPage creates a new page containing a general feedback form. Because the page hasn't yet been added to the navigation structure of the site, the page banner and link bar are not operational.

5. On the Standard toolbar, click the Save button to display the Save As dialog box.

6. In the File name box, type feedback.

7. Click the Change title button. In the Set Page Title dialog box, change the pages title to Feedback.

8. Click OK to close the Set Page Title dialog box, and then click Save to close the Save As dialog box and save your file. The file appears in the Folder List.

9. Click the Web Site tab to return to Navigation view.

10. Drag the feedback.htm file from the Folder List to the navigation structure at the same level as the home page.

In this position, the feedback form is not a child page of any other page, and will not show up on the current navigation link bars.

11. Click the feedback.htm page tab to return to Design view. The page title and navigation link bars are now shown correctly on the Feedback page.

12. Review the content of the Feedback page, and then near the middle of the page double-click the box that is currently set to Web Site. **The Drop-Down Box Properties dialog box appears.**

13. Click the Add button.

14. In the Add Choice dialog box, type chooses one.... In the Choice box.

15. In the Initial state area, select the selected option, and then click OK. Your new entry is added to Choice list in the Drop-Down Box Properties dialog box.

16. Clicks choose one....in the list, and then click the Move Up button until it is the first choice.

17. Use the Move Up and Move Down buttons to arrange the other items on the list in alphabetical order, with the exception of the (Other) choice, which should remain last.

18. Click the Validate button.

19. In the Drop-Down Box Validation dialog box, select the Disallow first choice check box to indicate that the form cannot be submitted with choose one....selected.

Then click OK.

20. Click OK again to close the Drop-Down Box Properties dialog box, and then click Show the Show Preview View button to preview the page. Preview view

21. Scroll down to the drop-down box, and then click the down arrow to display the list choices and view the results of your work.

22. Return to Design view, and save the page.

23. On the View menu, click Task Pane to open the New task pane.

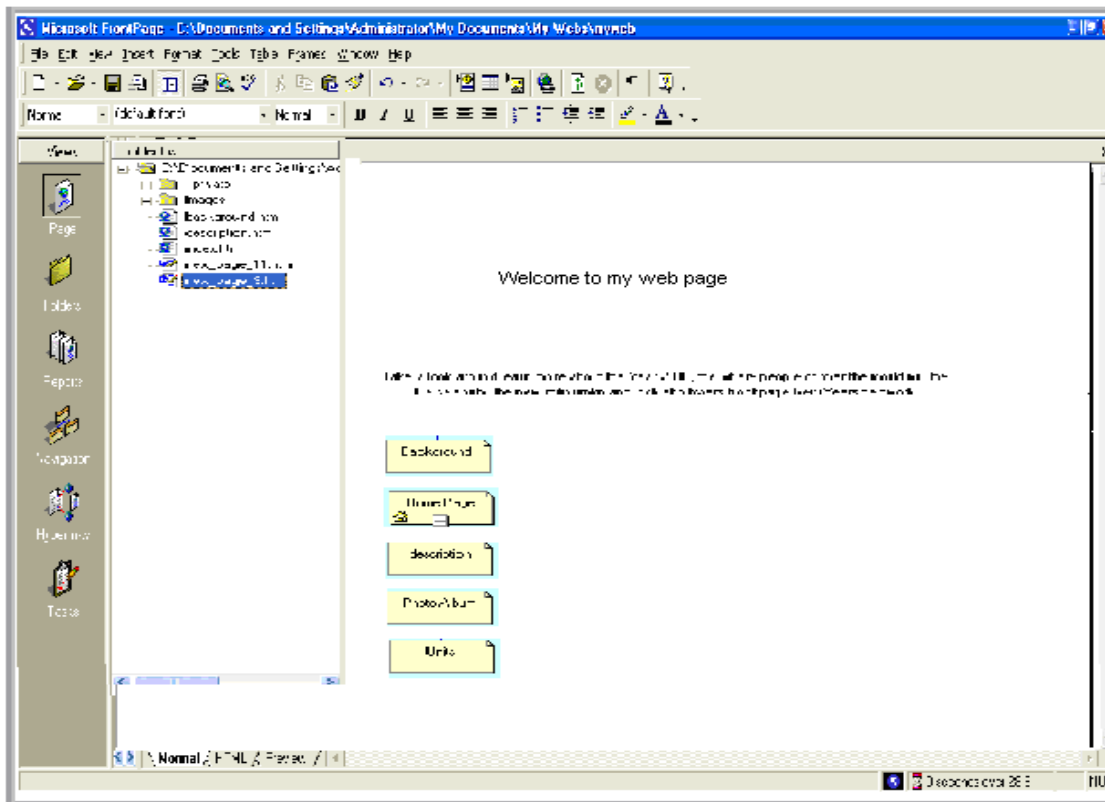
24. In the New page area, click More page templates to open the Page Templates dialog box.

25. On the General tab, click Confirmation From, and then click OK. FrontPage creates a standard confirmation form. Visitors will see this acknowledgment after submitting the feedback is pulled from the feedback form to this page.

26. Save the page as confirmation, with a page title of Thanks for Your Feedback!

27. Click the Web Site tab to switch to Navigation view, and drag the confirmation page to the navigation structure at the same level as the Feedback page.

28. Click the confirmation.htm page tab to return to Design view to see the results.



Enhancing the Appearance of a Web

If you're continuing this lesson directly from Lesson 1, the Millennium Celebration Web should still be open in FrontPage. If this is the case, skip down to the procedure named "To create hyperlinks to other pages."

If you're continuing this book from a previous session, then you must first open the web before you can work with its pages.

To open an existing web

1. On the Windows taskbar, click the Start button, point to Programs, and then click Microsoft FrontPage

2. On the File menu, point to Recent Webs and then click C:\My Documents\My Webs\Millennium to open the Millennium Celebration Web you created in Lesson 1. FrontPage opens the web. The application title bar now reads “Microsoft FrontPage - C:\My Documents\My Webs\Millennium.”

Because you’ll be working with the pages you’ve already created, you can close the blank page that opened by default in page view.

Microsoft FrontPage - C:\My Documents\My webs\Millennium

3. ...On the File menu, click Close, or click the Close button in the upper right corner of the page. FrontPage closes the current page. Page view is now blank, but the Millennium Celebration Web remains open. While creating hyperlinks from pictures and text in Lesson 1, you may have noticed that you don’t have any connections yet between the pages in your web. Even if someone surfed to your current home page, they would have no way of getting to the other pages. In the next section, you’ll learn how easy it is to make navigation hyperlinks to other pages.

To create hyperlinks to other pages

1. On the toolbar, click the Folder List button to show the Folder List in Page view.

2. Double-click index.htm in the Folder List to open the home page in Page view. You’ll keep the Folder List visible while you create hyperlinks to the other pages in your web.

3. When the home page is displayed in Page view, press CTRL+END to place the cursor at the end of the home page.

4. Next, locate the page background.htm in the Folder List. The folder and files in the Folder List are shown in alphabetical order. The icon of each file gives you a clue about what kind of file it is. You will now drag and drop the Background page onto the bottom of the home page. When you do this, FrontPage will create a hyperlink to the Background page on the home page.

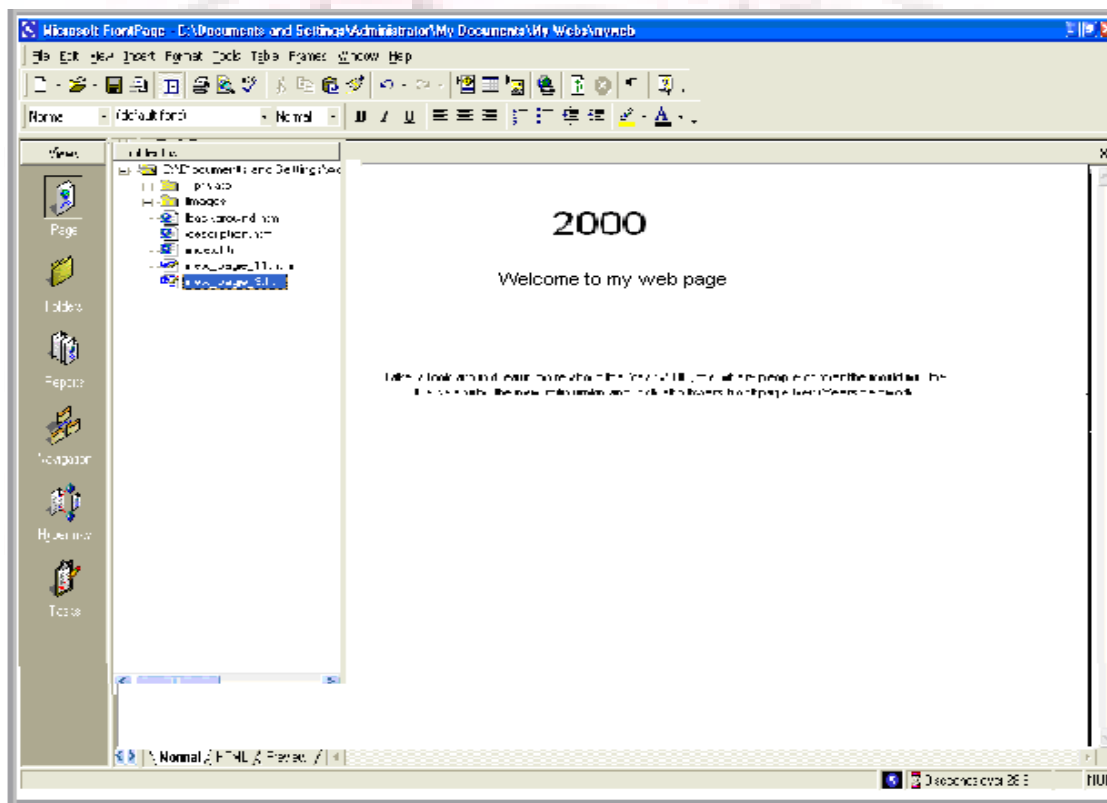
5. Click and hold the mouse button on background.htm in the Folder List, move the mouse pointer on the line below the FrontPage displays the shortcut mouse pointer while you drag the mouse to indicate that it will not actually insert the Background page, but will create a hyperlink pointing to it. FrontPage inserts the page title of the Background.htm file

("Background") as the hyperlink text. The blue underlined text shows the presence of the hyperlink.

6. Repeat steps 4 and 5 with the other pages in the Millennium Celebration Web, including Destinaton.htm, Links.htm, and Photo_album.htm. Place each link just after the previous one.

7. On your keyboard, press the DOWN ARROW key to deselect the last hyperlink.

Your page should now look like this:



While you can manually create hyperlinks to the other pages in your web this way, doing so for all pages in a web can become a time-consuming and tedious task, especially for larger webs. Worse, if you decide to add to remove pages in the current web after creating hyperlinks, you'll have to manually add or remove the hyperlinks to them. FrontPage has a better way to create, manage, and automatically update the navigation hyperlinks that connect your pages tighter. Before you learn how to do this, let's get rid of the four hyperlinks you just made.

To use the multiple undo command



Undo button with arrow

1. On the toolbar, click the small arrow just to the right of the Undo button.
2. FrontPage displays the Undo history, which shows the last several actions you can reverse. The first of these actions is selected by default. If you were to click it, then only that action would be reversed. You can also move the mouse over other entries in this list to include them in the Undo command.
3. Since we want to get rid of all four hyperlink you just dragged and dropped onto the home page, move window should read Undo 4 Actions.
4. Click the mouse on the last occurrence of Drop in the list. FrontPage reverses the last four actions you took , and the four hyperlinks you created are removed from the home page.
5. to save the current page, click Save on the File menu, or click the Save button on the toolbar.

Adding Shared Borders and Navigation Bars

For the Millennium Celebration Web, you will let FrontPage manage the hyperlinks that site visitors will click to move around the pages in your web. FrontPage achieves this with a combination of two powerful features: shared borders and automatic navigation bars. Shared borders are page regions reserved for contain that you want to appear consistently throughout the pages in your web. These borders can contain page banners and navigation bars. Page banners display the page title you gave each page when you created or saved it. Navigation bars are a row or column of hyperlinks to the other pages in the current web. FrontPage can automatically update shared borders and navigation bars, so the navigation structure of your web will always work correctly, even when you add, move, or delete pages from the web's structure.

In Lesson 1, you already completed the first step required for automatic navigation bars: creating the basic web structure in Navigation view. Because you have already done this, you'll now enable shared borders throughout your web.

To create shared borders across a web

1. Click the Navigation icon on the Views bar to switch to Navigation view.

2. Click the Folder List button to hide the Folder List in this view.
3. On the Format menu, click Shared Border.
4. FrontPage displays the Shared Borders dialog box. Here, you can specify where on your pages FrontPage should insert shared borders. Because your web structure has two levels of pages the home page and the pages below it – you will use two kinds of shared borders and two kinds of navigation bars.
5. In the Shared Borders dialog box, make sure the All pages option is selected.
6. For a horizontal shared border, select the Left check box and select the Include navigation buttons check box just below it.
7. For a vertical shared border, select the Left check box and select the Include navigation buttons check box below it.
8. Leave the Right and Bottom check boxes unchecked, and then click OK. FrontPage create shared border and default navigation bars for all the pages in the current web. You'll see what these look like when you return to Page view. Next, you'll customize the appearance of the default navigation bars. Because they are shared across all pages in the current web, you can change their properties on any page and the change will be reflected across the entire web.

To test navigation bar hyperlinks

1. In Navigation view, double-click the Home Page.
2. Click the Folder List button to hide the Folder List in Page view.

Note the changes FrontPage has made to the home page. It now contains a top and left shared border. The top border contains a page banner with that look exactly like the ones you manually created at the beginning of this lesson. In Page view, you can easily test hyperlinks that point to pages and files in your web.

3. Hold down CTRL and then click the first navigational hyperlink named Background on the left side of the page. FrontPage opens the page the hyperlink points to. On the Background page that is now open, shared borders and navigation bars have also been inserted. On this page, however, the links to the other pages are displayed in the top border, just under the page banner. This is because FrontPage uses the web structure you created in Navigation view to determine the level the current page is on. By default, the top shared border

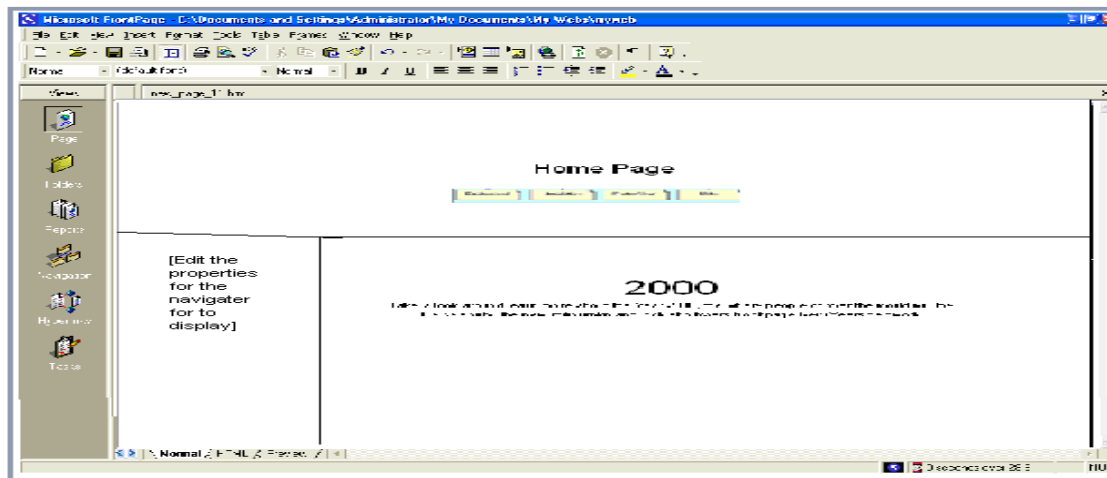
points to pages on the same level as the current one, whereas the left border points to pages below the current one. In the next section, we'll change this default to another design.

To customize navigation bars

1. On the Window menu, click index.htm. FrontPage brings the home page back into view.
2. In the top of the home page, double-click the text that Edit the properties for this Navigation Bar to display hyperlinks here.
3. Double-clicking a navigation bar opens the Navigation Bar Properties dialog box. Here, you can customize the appearance of a navigation bar and the hyperlinks it creates.
4. Currently, the horizontal navigation bar is set to link to pages on the same level. Since the home page is on its own level in your navigation structure of your web, there are no other pages on the same level. FrontPage therefore doesn't show any navigation bars in this shared border.
5. For the Millennium Celebration Web, we want to have a horizontal navigation bar on the home page and vertical navigation bars on the other pages. To do this, we'll change the default setup of both navigation bars. You can make these changes on the current page and they'll be reflected throughout your web.
6. In the Navigation Bar Properties dialog box, click Child level, clear the check boxes for Home page and Parent page, and then click OK. FrontPage creates a navigation bar with hyperlinks to all the pages below the home page level.
7. Press Home to deselect the navigation bar.

Your page should now look like this:

Note that the left navigation bar still contains the same set of hyperlinks as the top navigation bar. In the next steps, you'll remove the obvious redundancy, and format the left navigation bar so it is displayed only on the other pages that the home page points to.



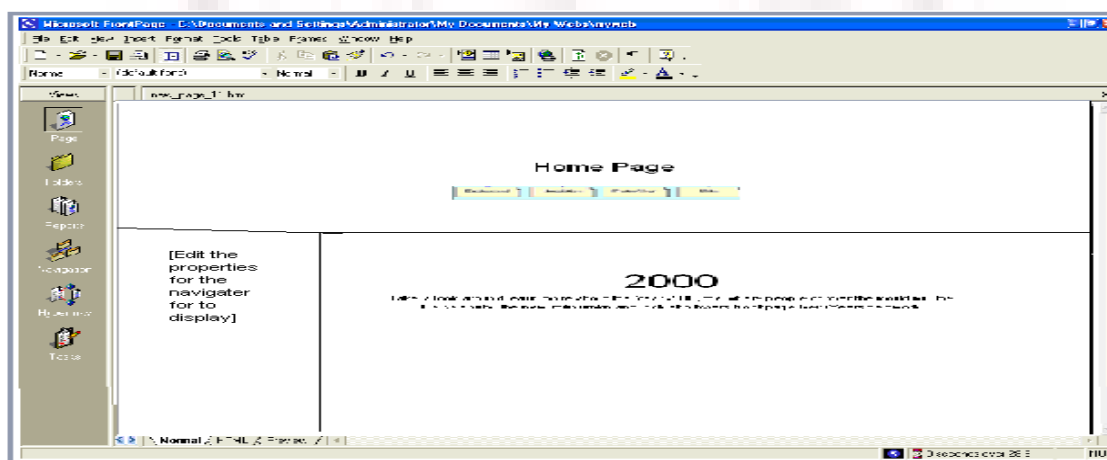
8. In the left border of the home page, double-click the vertical navigation bar.

9. In the Navigation Bar Properties dialog box, click Same level, select the Home page check box, and then click OK. FrontPage changes the navigation bar to the placeholder text that reads “Edit the properties for this Navigation Bar to display hyperlinks here.” This text is only shown in Page view while you work’ it will not appear in a Web browser.

By pointing the hyperlinks in this navigation bar to the same level as the home page, you are effectively removing the hyperlinks from the left border, because there are no pages on the same level as the home page. This also removes the redundancy between the horizontal and vertical navigation bars.

10. To save the home page, click Save on the File menu, or click the Save button on the toolbar.

Your page should now look like this:



11. On the Window menu, click background.htm.

FrontPage brings the Background page back into view. Note that the changes you've made on the home page to both the horizontal and vertical navigation bars are automatically reflected here, as well as on all the other pages in your web.



Excluding pages from navigation bars By default, all pages in your web's navigation structure are included in navigation bars. You can select pages to be excluded from your navigation bars by right-clicking the page in **Navigation** view and deselecting **Included in Navigation Bars** on the shortcut menu.

Applying a Theme

Although the addition of pictures, lists, forms, shared borders, and navigation bars has given the pages in the Millennium Celebration Web a more streamlined and organized look, you may wonder what to do about the rather bland appearance of black and blue text on a white background. After all, this web is about celebrating an event. You want the pages to look more lively and fun.

Imagine how time-consuming it would be if you had to design a color scheme for text and graphics, and create graphical page banners, navigation buttons list bullets, and background textures for all the pages in your web. Now imagine how many more custom graphics you would need to create if you maintained more than one Web site and you didn't want any of your webs to look the same.

FrontPage includes more than 50 professionally designed themes with matching color schemes that you can apply to any or all pages in your web. A theme consists of design elements for bullets, fonts, pictures, navigation buttons, and other graphics. When applied, a theme gives pages, page banners, navigation bars, and other elements of a web an attractive and consistent appearance

To apply a theme to the Millennium Celebration Web

1. On the Window menu, click index.htm. FrontPage brings the home page back into view.

2. On the Format menu, click Theme. FrontPage displays the Themes dialog box. Here, you can select from a list of themes that FrontPage installed by default, or choose to install the complete set of themes from your FrontPage 2000 CD-ROM. You can make choices about the appearance of the theme, preview theme elements, and modify the selected theme.

3. Click on some of the different theme names in the scrolling list box. When you click the name of a theme, the Sample of Theme window shows a sample of the graphical elements that are contained in the selected themes. This way, you can first preview a theme before applying it to selected or all pages in your web.

Before applying a theme, you can select theme options that affect the appearance of the theme's components. For example., selection Vivid colors applies brighter colors to text and graphics, selection Active graphics animates certain theme components, and selecting Background picture applies a graphics background to the pages in your web. You can also choose to apply a theme as a cascading style sheet (apply using CSS). For the Millennium Celebration Web, you'll clear these defaults.

4. Under Apply theme to, make sure All pages is selected

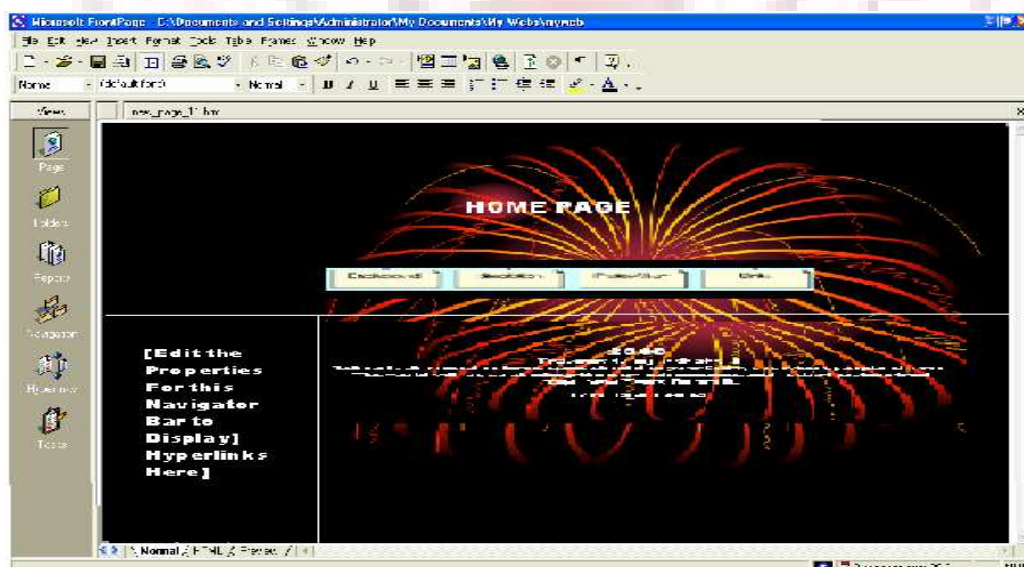
5. In the list of installed themes, click Artsy.

6. Clear the check boxes for Active graphics and Background picture, then click OK to apply the theme. Since this is the first time you're applying a theme to a web, FrontPage displays a message to let you know that applying a theme will overwrite some of the manual design work in this book, so you can acknowledge this message and proceed to apply the theme.

7. Click Yes to apply the theme. The theme named "Artsy" is applied to all pages in your current web.

8. To save the home page, click Save on the File menu, or click the Save button on the toolbar.

Your page should now look like this:



As you can see, applying the theme has dramatically changed the appearance of the home page. The page banner and navigation buttons are no longer plain text; now they're colorful graphics. The page background has changed from white to black, which simulates the night sky that the millennium fireworks will appear in, and the font has changed color and is a little larger.

Displaying graphical navigation buttons on all pages

1. On the Window menu, click background.htm. FrontPage brings the Background page back into view., note that the page has inherited its theme and theme elements from the home page, but the vertical navigation bar in the left border still shows plain text hyperlinks. By default, vertical navigation bars are displayed as plain text, so they look this way even after you apply a theme. You can easily change navigation bar settings even after a theme is applied

2. In the left border of the Background page, double-click the vertical navigation bar.

3. Under Orientation and appearance in the Navigation Bar Properties dialog box, click Buttons, and then click OK.

4. Click anywhere on the page to deselect the navigation bar. FrontPage changes the navigation formatting and uses the graphical buttons included with the theme. The web now has an attractive and professional look.

5. To save the page, click Save on the File menu, or click the save button on the toolbar.

Some themes contain animations When you apply a theme, you can select Active graphics to enable page banner animations and navigation bar rollover effects, if the theme contains such elements. The theme sample will not show you what these effects look like. To see a theme's active graphics effects, apply the theme and then display the page in the Preview tab, or click the Preview in Browser command on the File menu.

Modifying a Theme

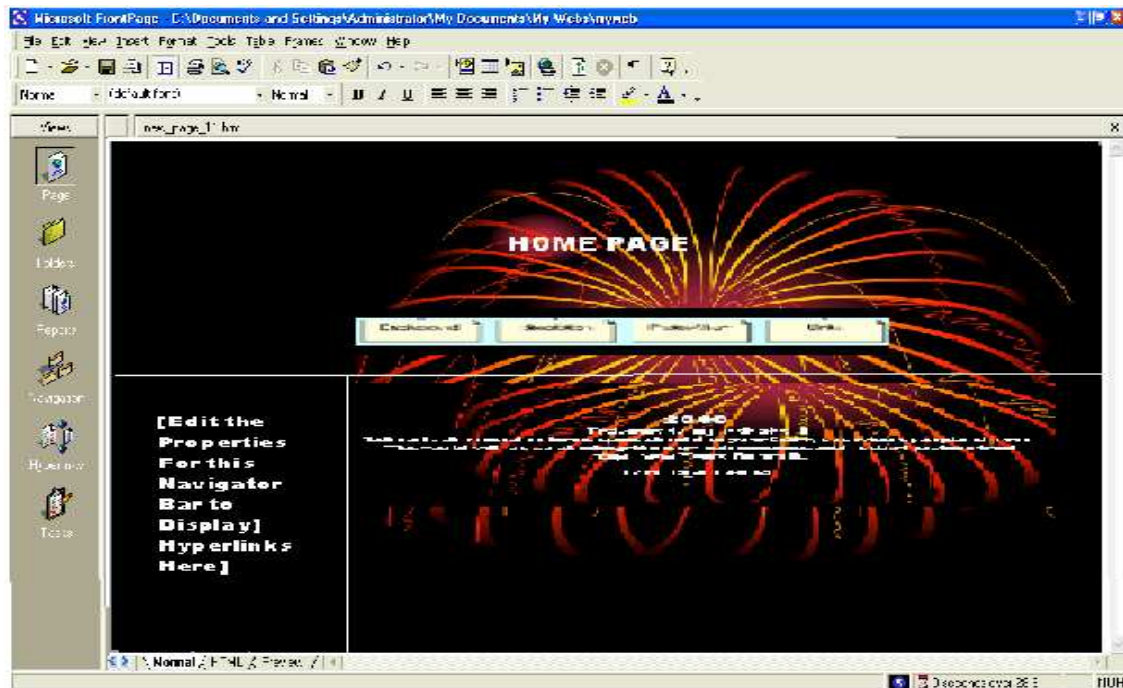
Although the page banner of this theme looks nice, something directly related to the subject matter of the Millennium Celebration Web might fit better. We've prepared a custom page banner that you will use to modify the current theme with. This custom banner provides a colorful fireworks backdrop for the page banner text.

To modify a theme

1. On the Window menu, click index.htm. FrontPage brings the home page back into view.
2. On the Format menu, click Theme. FrontPage displays the Theme dialog box. In the list of themes, the Artsy theme is now the default theme because it has been applied to the current web.
3. In the Theme dialog box, make sure All Page is selected.
4. Next, click Modify
5. Under the question What would you like to modify> click Graphics. FrontPage displays the Modify Theme dialog box. Here, you can supply custom graphics for various theme elements such as page banners, navigation buttons, background pictures, and other elements. FrontPage superimposes text over these graphics, so there is no need to change graphics when you change the names of your pages, or add or remove pages. For this example, we will change the graphical page banner on which FrontPage places the titles of the pages in the Millennium Celebration Web.
6. In the Item list, click Banner.
7. On the Picture tab, click the Browser button below the file name of the current banner graphic. FrontPage displays the Select Picture dialog box and shows the current pictures in your current web. Since the graphical banner we want to use isn't part of the web yet, you will search your file system for it.
8. In the Select Picture dialog box, click the Select File button. FrontPage displays the Select File dialog box.
9. Navigate to the folder named Program Files\Microsoft Office\Office\Book by double-clicking each folder in this path until the Look in box displays the Book folder. If you downloaded the book files from www.microsoft.com, navigate to the folder named Fptutor\Samples, or to the folder where you placed the files.]
10. Double-click the file 2000ban. FrontPage replaces the current page banner graphic with the custom graphic.
11. Click OK in the Modify Theme dialog box, and then click OK in the Themes dialog box. FrontPage displays a message asking you whether you want to save changes to the current theme.
12. Click Yes. FrontPage Millennium as the title of the modified theme, and then click OK. FrontPage saves the modified theme and applies the new banner to all pages.

13. To save the page, click Save on the File menu, or click the Save button on the toolbar.

Your page should now look like this:



Making your own themes for this book, we've provided only a custom page banner to help show you how easy it is to customize existing themes. To create your own designs, repeat the steps in the previous procedure to customize other theme elements with your own custom graphics.

Congratulations, the Millennium Celebration Web is almost finished! To make sure everything will look great on the World Wide Web, you'll now preview the web in your Web browser....

Previewing the Millennium Celebration Web

Although Page view shows you the appearance of your Web pages as closely as possible to how they will appear in a Web browser, it displays some page elements and placeholders differently to help you while you design the page. By previewing a page or your entire site in a Web browser before you publish the Web, you can make sure that everything looks the way you want it to.

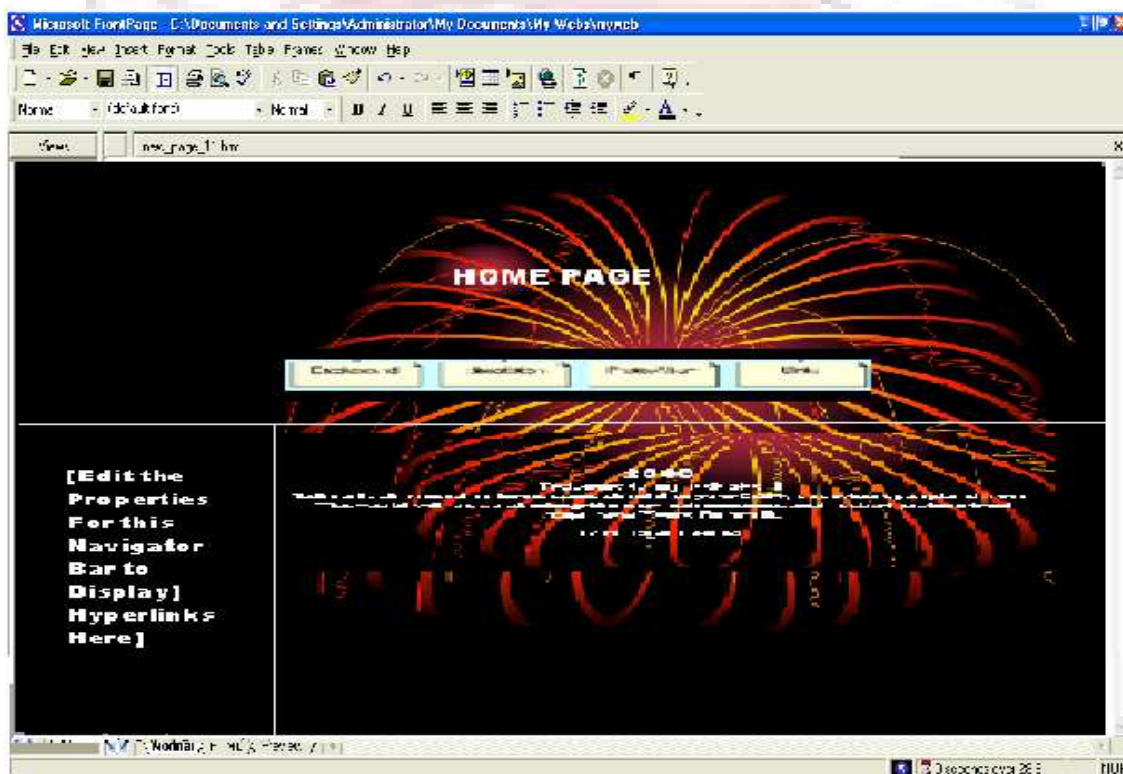
To preview the current web in a Web browser

1. On the File menu, click Preview in Browser. FrontPage displays the Preview in Browser dialog box. Here, you can select from the installed Web browsers on your computer window size in which you want to preview your web.

The Preview in Browser feature loads the current page in your Web browser, so you can see exactly how the page will appear in your favorite Web browser before you publish your web. You must have at least one Web browser installed on your system for this feature to work.

2. In the Preview In Browser dialog box, click Microsoft Internet Explorer, and then click Preview FrontPage launches the Web browser and opens the home page.

You screen should now look like this:



Note that all placeholder text and formatting marks are hidden from view (for example, the empty vertical navigation bar on the left side of the home page does not appear).

3. Click the buttons on the navigation bar to preview some of the other pages. Note the vertical navigation bar on the pages below the home page. On the Destinations page, scroll down to the feedback form you added to the page. You can enter text in the fields, but

the form won't actually work until you publish the Millennium Celebration Web to a Web server.

On the Photo Album page, click the picture thumbnails to test the hyperlinks to the full-size pictures. Use your Web browser's Back button to return to the Photo Album page. Finally, on the Links page, note the dynamic of the paragraph heading that reads "Links to My Favorite Sites."

4. Close your Web browser when you when you have finished previewing the Millennium Celebration Web.

5. Optimizing Your Screen Display Properties

6. The width and height of your computer monitor display in pixels is called the screen

7. Resolution. When personal computer first became popular, most computer monitors

8. Were capable of displaying a screen resolution of only 740 pixels wide by 480 pixels

9. High (more commonly known as 640*489). Now most computer monitors can display

10. At 800*600 pixels and 1024*768 pixels. Some monitors can even display a screen

11. Resolution of 1280*1024 pixels, or larger. Newer monitors no longer offer a 640*480

12. Screen resolution.

13. Most computer users have the choice of at least two different screen area sizes. Some

14. people prefer to work at an 800*600 screen resolution because everything on the screen

15. Appears larger. Others prefer being able to fit more information on their screen with a

16. 1024*768 display.

important

17. when designing a Web page that consists of more than free- flowing text, it is

lowest

18. to consider the likely screen resolution of your Web visitors and design for the

look their

19. Common denominator. It is currently common practice to design Web sites to

visitors

20. Best when the visitor's screen area is set to 800*600 pixels. (This means that

entire page.)

21. Who view your site with a 640*480 area will have to scroll to display the

22. To check and change your screen resolution on a Window XP computer:

Panel.

23.1 At the left end of the Windows taskbar, click Start, and then click Control

Display

24.2 In the Control panel window, click Appearance and Themes, and then click

25. To open the Display Properties dialog box.

resolution

26.3 On the Settings tab, look at the Screen resolution slider. The current screen

27. Appears beneath the slider.

changes.

28.4 Drag the slider to change the screen resolution, and click Apply to apply your

29.5 If a dialog box appears prompting you to confirm the change, click Yes.

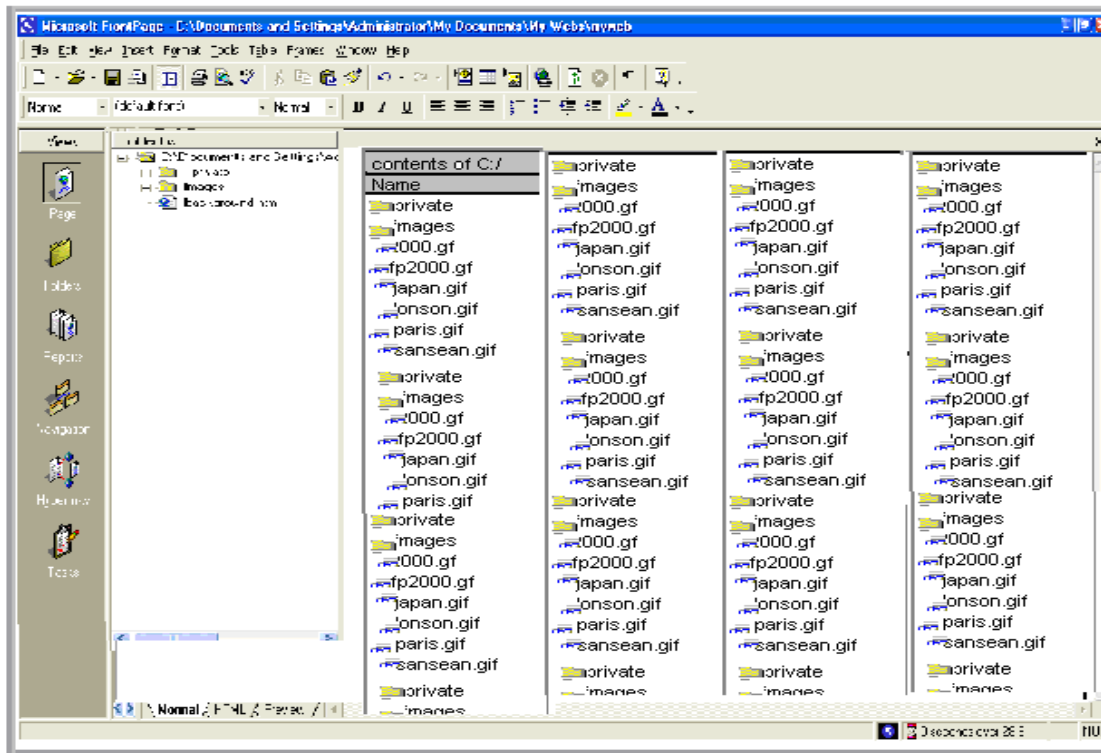
30.6 When the screen resolution is the way you want, click OK.

Organizing the Files in your Web

Now that your web contains several pages and files, you will use Folders view to organize them. Similar to Windows Explorer, Folders view lets you manage the files and folders

in your web. You can safely rearrange the pages and files in your web without breaking hyperlinks, page banner titles, or navigation button labels.

In Folders view, FrontPage displays a hierarchical list of the folders in your web on the left side of the screen. Clicking on a folder in the Folder List displays its contents on the right side- the contents pane.



In the following steps, you will move all the picture files in the Millennium Celebration Web to the Image folder FrontPage created as part of the web.

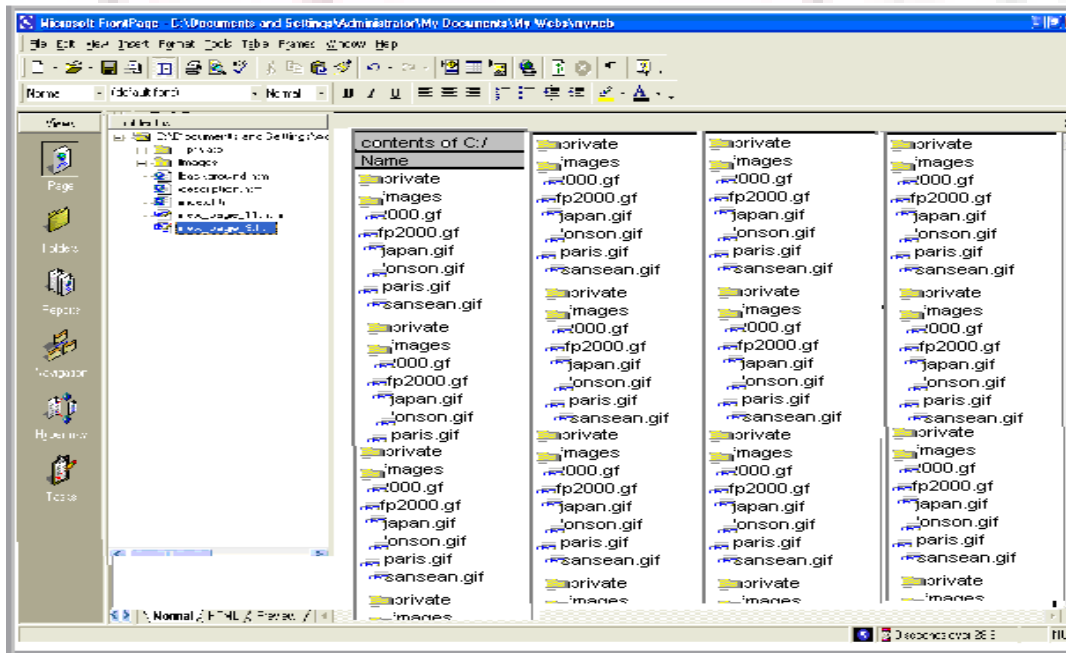
If you were to use Windows Explorer or another file manager to move pages and files from one folder to another, you would break the hyperlinks between your pages and page elements. However, when you maintain your web in Folders view, FrontPage keeps every page and hyperlink in your web updated to keep of the new locations of files and folders that have been moved.

To move picture files to the Images folder



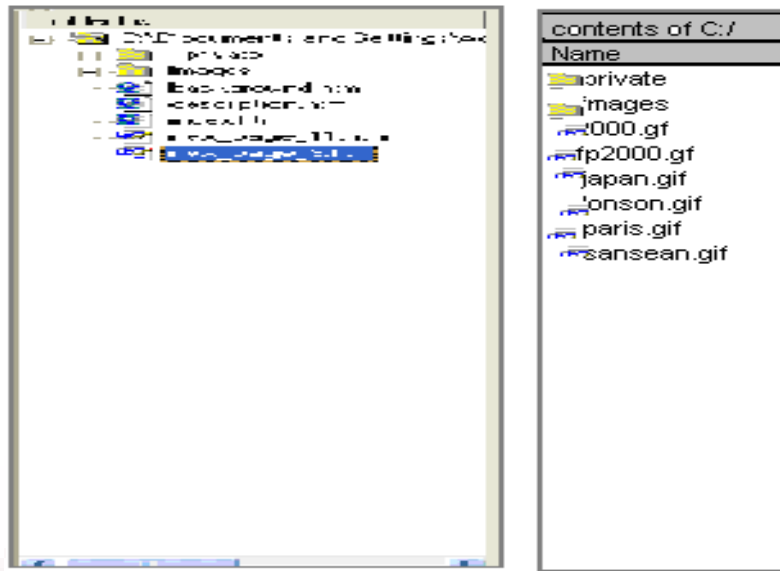
Folder icon

1. On the Views bar, click the Folders icon. FrontPage switches to Folders view.
2. In the Folder list pane, click the top-level folder labeled C:\My Documents\My Webs\Millennium.
3. In the Contents pane, click the Type column label. Clicking on a column label, the list is sorted in ascending order; when you click it a second time, it is sorted in descending order. The list of files is now grouped by file type, with all GIF picture files at the top of the list, followed by HTM files (pages) in the middle, and all JPG pictures at the bottom of the list.
4. In the Contents page, click the first picture file (2000.gif) at the list to select it.
5. Next, while holding down SHIFT, click the last GIF picture file in the list (sanfran.gif).



In Folders view, FrontPage supports all standard Windows selection shortcuts, such as SHIFT-CLICK for selecting ranges of files, and CTRL+CLICK for selecting noncontiguous files.

6. Click and hold the right mouse button while the pointer is over any of the selected GIF file icons.
7. Next, drag the mouse pointer over to the Images folder in the Folder List pane.



8. When the Images folder is selected, release the mouse button and click Move Here on the shortcut menu. FrontPage displays the Rename dialog box while it is moving the selected GIF image files to Images folder because it is automatically updating all hyperlinks to these files in the current web.

9. Repeat steps 4 through 8 with all JPG picture files, starting with Firewks1.jpg and ending with Firewks4_small.jpg.

10. In the Contents pane, click the Name column label to arrange the remaining list of folders and files by their name again. You've successfully grouped all picture files in the Images folder.

When you work with your own webs, you can group sound files, movie clips, and other types of files in their own folders. You can create new folders in Folders view as needed and delete the ones you no longer need.

To create a new folder

1. In the Folder List, click the folder in which you want to create a new subfolder. Folders can be expanded and collapsed in the Folder List to bring their subfolders into view. Click the plus (+) and minus (-) signs next to a folder's name to display or hide its subfolders.

2. On the File menu, point to New and then click Folder. FrontPage creates a new folder with a temporary name.

3. When the folder's temporary name (New_Folder) is selected, type a new name for the folder, then press ENTER. The new folder is renamed, and you can now drag and drop files into it. For this book, we don't need the extra folder you just created, so you will delete it before we get ready to publish the web.

4. In the folder List, right-click the folder you just created.

5. On the shortcut menu, click Delete.

6. In the Confirm Delete dialog box, click Yes, FrontPage removes the folder from the web. Generating a Site Summary Reports view is an important tool that shows you the overall health and condition of your web before you publish it to the World Wide Web. You can generate custom reports about your web in up to 14 categories. To generate a Site Summary report.

Reports icons

- On the Views bar, click the Reports icon.

FrontPage switches to Reports view. The default report is the Site Summary. This report shows you the overall statistics of the pages and files in the Millennium Celebration Web. Here are some important ones to look at before you publish your web:

- All files: you currently have 21 files in your web, totaling approximately 275K in size. This is the amount of space you'll need to have available on the web server that will host your web.
- Slow pages: this category shows pages that are slow to download at the targeted download speed. Because you created small thumbnails of the large fireworks picture in your online Photo Album, your web currently doesn't have any slow pages for you to worry about.
- Broken hyperlinks: If any broken hyperlinks are reported here, double-click the Broken hyperlinks to view details about this category, FrontPage lists unverified hyperlinks, such as the external hyperlinks on your Links page, and links that are broken and do not work.
- You can verify that a hyperlink still points to an active Web site by right-clicking the link in Reports view and choosing Verify from the shortcut menu. To fix a broken hyperlink, you must open the page it is on and repair the URL the hyperlink points to.

Spell checking

While you can use automatic background spell-checking and per-page spell – checking in Page view, using the Spelling command in any web view lets you check the spelling of all (or selected) pages across the current web. You can check the spelling of page elements that can be edited directly on the page. Other text, such as page banners or text contained in FrontPage based components, are not included in the spelling check.

To check spelling in the current web

1. On the Tools menu, click Spelling. FrontPage displays the Spelling dialog box. Here, you can specify whether FrontPage should check the spelling of selected pages only, or of the entire web.
2. In the Spelling dialog box, click Entire web, and then select the Add a task for each page with misspellings check box. FrontPage will add a task to the Tasks list for each page on which misspelled text is found. You will learn about Tasks view in the next section.
3. In the Spelling dialog box, click Start to begin the spelling check. FrontPage expands the Spelling dialog box to display the progress of the spelling check. When the operation has been completed, FrontPage displays the misspelled words and the number of tasks that were added to the Tasks list in Tasks view.
4. Click Cancel to dismiss the spelling dialog box. The spelling check is complete, but the corrections will not be made until you complete the tasks in the Tasks list

Replacing Text on Pages

The Replacing command makes it easy to find and replace content on selected pages or all pages in the current web. While you can use the command to replace text on the current page in Page view, using it in any web view lets you replace text in all (or selected) pages across the current web. You can replace any text that can be edited directly on the page. Other text, such as page titles in page banners or text contained in FrontPage-based components cannot be automatically replaced.

To replace text on all pages in the current web

1. On the Edit menu, click Replace.

2. FrontPage displays the Replace dialog box. Here, you enter the string of text to be found and what you to replace it with. You can choose to replace text on all pages in the current web, or an selected pages only.

3. In the Replace dialog box, type Welcome to my Web site in the Find what box.

4. In the Replace with box, type Thanks for visiting my Web site.

5. Click the Match case check box, and then click Find in Web.

FrontPage expands the Replace dialog box to display the progress of the search. The search text you want to replace is found on the home page, index.htm. When the operation has been completed, FrontPage displays the number of occurrences it has found.

6. Click Add Task in the Replace dialog box.

7. Click Cancel to dismiss the Replace dialog box. The replacement search is complete, but the actual replacement will not occur until you complete the task in Tasks view.

Completing Web Tasks

Tasks view displays the list of all outstanding tasks associated with the current web. Tasks are items that need your attention before you publish the web. In the previous exercises, you added tasks to a list when you deferred certain actions. For example, when you checked the spelling of the pages in your web, you choose to add a new task for each page containing misspellings. By adding tasks to the list, you can complete such corrections all at once.

If you are working in a web development environment or on an intranet, Tasks view makes it easy to track web tasks and assign them to authors who work on the same web.

To complete tasks in Tasks view

Tasks icon

1. On the Views bar, click the Tasks icon. FrontPage displays the Tasks list.

2. Double-click the first task on the list, labeled “Fix misspelled words.”

3. FrontPage displays the Task Details dialog box. Here, you can see details about the task you’re selected. You can set the priority of the task, assign it to another author on your network, or complete the task and remove it from the list.

4. In the Task Details dialog box, click Start Task. FrontPage switches to Page view and opens the page containing the misspelled words.

5. In the Spelling dialog box, click Ignore when FrontPage questions the name “Balleny”.

6. Click Add to add “Cheops,” the name of the Egyptian king, to your dictionary. FrontPage shares custom dictionaries with other Microsoft Office applications, so you don’t need to add custom words in each application separately.

When you add verified words to your dictionary, they will not be questioned again.

7. Click OK. FrontPage completes the spelling check. If you want to, you can now return to Tasks view and mark this completed. Although it is not required that you complete every task before publishing your web, it is good idea to review this list when you are finished making changes to the web. Tasks view helps you manage webs by flagging important reminders for you.

Publishing the Millennium Celebration Web

When you publish your web on the World Wide Web – or your company intranet- FrontPage automatically verifies your hyperlinks, the addresses of your pages, and the paths to your files.

Note

If you do not want to publish the Millennium Celebration Web to your Web server, read this procedure for reference only, without actually completing the steps.

To publish the Millennium Celebration Web

1. Close all open pages in Page view.

Publish Web button

2. On the File menu, click publish Web, or click the Publish Web button on the toolbar. FrontPage displays the Publish Web dialog box. Here, you specify the location on the World Wide Web or your corporate intranet to which you want to publish your web. Your Internet service provider can tell you this information.

You need Internet access through an Internet service provider before you can publish your web to the World Wide Web. If you want Presence Provider that can host FrontPage –enabled webs, click the WPPs button in the Publish Web dialog box.

3. In the Publish Web dialog box, enter the URL of your target Web server, (such as `http://example.microsoft.com/~myweb`), and then click Publish. FrontPage publishes the current web from your computer to the World Wide Web or intranet Web server you specified.



One-button publishing After you publish a web for the first time, you can bypass the **Publish Web** dialog box by using the **Publish Web** button on the toolbar. This quickly publishes any updates you've made to your pages without having to provide any information about the web's location. To again display the **Publish Web** dialog box, use the **Publish Web** command on the **File** menu instead of the toolbar button.

If FrontPage detects that you are publishing to a Web server that does not support the FrontPage Server Extensions, it will publish the current web via the FTP file transfer protocol. If the Web server to which you are publishing your webs has the FrontPage Server Extensions installed, your webs will have full functionality of FrontPage- based components and Web scripts that you may have inserted on your pages.

Publishing webs to a Web server that does not have the FrontPage Server Extensions installed may disable some functionality contained on your pages, such as the feedback from you added. FrontPage will display informational messages during the publication process to alert you of such conditions.

During the publishing process, FrontPage displays a progress bar to indicate how much time is required to transfer your web to the target Web server.

The speed at which FrontPage publishes your web depends on your connection speed, as well as the number and complexity of the pages and files in your web.



Excluding pages from being published when you publish a web, all of its pages and files are published by default. To exclude pages or files from being published right-click the page or file in a web view, click **Properties** on the shortcut menu, and select the **Exclude this file when publishing the rest of the web** check box on the **Workgroup** tab.

When FrontPage has successfully published your web, it provides a hyperlink to your new Web site in the confirmation dialog box. Click the link to open the published Web site in your Web browser.

Deleting a Web Site

When you first start creating web sites with FrontPage, you will probably want to experiment. As a result, you will more than likely end up with Web sites on your hard disk drive that you no longer need. What's more, if you make a mess when creating a real Web site and decide to start over, because you already have a Web site with your chosen name stored on your hard disk driver FrontPage will not allow you to overwrite the existing site with a new one. You must create a whole new set of files by appending a number to the name you want to use. To solve these problems, you might be tempted to simply delete existing sites in Windows Explorer, but if you do, you risk leaving behind extraneous hidden files. Instead you must delete the sites from FrontPage. In this exercise, you will delete the two Web sites you created with templates at the beginning of the chapter. Important if you did not create the OnePage and Personal Web sites in the first exercise of this chapter skip this exercise. USE the OnePage and personal Web sites you created in the first exercise of this chapter. OPEN the OnePage Web site.

1. In the Folder List, right-click the top-level folder of the site, and click Delete on the shortcut menu to open the Confirm Delete dialog box.
2. Select the Delete this Web site entirely option, and then click OK to delete the Web site. The Web site is deleted and the Folder List closes because the displayed content no longer exists.
3. click the down arrow to the right of the Open button, and click Open Site in the Open drop down list,
4. Browser to and open the Personal web site created at the beginning of this chapter.
5. Repeat steps 1 and 2 to delete the Personal Web site.