

Introduction

The Internet has fast become an extremely popular form of media for advertising and providing information to a wide range of people at a minimal cost.

As far as advertising goes, the old adage “*..a picture is worth a thousand words..*” says it all. Full colour advertisements for your Taekwon-Do club with details about who, what, when, where, how, and why are easily achieved. A real “feel” for your club can be projected on a well designed Web site, making viewers eager to find out more information.

You may also find that many current members at your club have access to the Internet, and use it regularly. In this case a club Web site is an ideal way of keeping your members up-to-date with upcoming events and newsletters.

The purpose of this document is to provide you with the necessary information to be able to create a Web page for your own Taekwon-Do club. It is not a complete resource, but it provides enough detail of the basics for you to be able to achieve effective and rewarding results on the Net. If you require further information about any of the topics covered in this document, a visit to a bookstore will probably spoil you for choice with the number of books and resources currently available on the topic of Web page creation.

I assume that you have a computer, or at least have access to one. You will also need an Internet browser loaded, such as Netscape Navigator or Microsoft Explorer. An actual Internet connection would be even better, and having an account with an Internet Service Provider (ISP) that grants you a certain amount of space on their Web server is ideal.

Chapter 1: What is HTML?

HTML stands for **Hypertext Markup Language**. It is a simple markup convention used to create “hypertext” documents.

HTML is based on SGML (Standard Generalised Markup Language), which is an international standard for document exchange. SGML defines a document structure, but does not determine its physical appearance.

HTML defines a documents structure and other attributes, but does not actually define the page format that is viewed in a browser. One of the most exciting features of HTML is the ability to “link” to other HTML documents, downloadable files, Newsgroups, graphics, remote computers or mail programs.

Another great thing about HTML is the fact that it is platform independent. This means you can create your Web pages on a Windows 95 PC and, regardless of operating system, have anyone else with Internet access view them

HTML markup originates from the hand-written annotations used in traditional publishing, where a manuscript is annotated with layout instructions for the typesetter. In HTML, the markup, or *tags*, work as annotations for browser software.

When an HTML document is accessed using a browser, the browser reads and interprets each tag in the document and uses that information to display the page. HTML files are distinguished by the .htm file extension, e.g. `index.htm`.

Chapter 2: Design Principles

Before forging ahead and attempting to produce a Web site from scratch, there are a few considerations that should be taken to ensure your viewers are getting the message you want them to.

1. Consider your audience. Obviously, in this case, your Web site will be aimed at people who are interested in Taekwon-Do, or who are practitioners currently. Demographic factors (age, education, geographic location) will range greatly, so it is a good idea to try and cater to as wide an audience as possible, but keep the information specific to Taekwon-Do.
2. Nobody likes having to wait for a Web site to download, so be careful to not put too many large pictures (in file size) on the main page – viewers may get sick of waiting and give up. Try to keep images on the main page small, maybe even thumbnailed if they are photos (smaller images that can be clicked on to show a larger version)
3. Try to not make each page too long. Aside from the fact that this also slows down the transmission of the page, it can also make the document difficult to read. If you have several different ideas on a page, it is perhaps a good idea to separate them onto individual pages. However, do not break up the contents of a document unnecessarily. It is generally not worth splitting one idea into smaller pieces in order to keep the length of a page down, as the reader can use the scroll bars on the browser to read through the document in an uninterrupted flow, if that is how the document is written.
4. Not all browsers are the same. Some browsers have greater capabilities than others, and some cannot process some types of HTML tags. Try to keep your pages as generic as possible to allow as great an audience as possible. A really good idea is to view each of your documents from several popular browsers (such as Netscape Navigator and Microsoft Explorer) before actually “going live” with them.
5. Be sure that your site is easy to navigate by providing a number of links to the other pages on your site. Icons to assist with navigation through your pages are sometimes a good idea, especially if you keep the icons the same throughout all of your pages (such as “forward”, “back” and “home” links, or links to specific pages). If you do use icons, try to make sure they are easily associated with the content of the location the link takes the viewer to. However, it is a good idea to not include too many links to locations outside of your own site on the main page, as this makes it easy for the viewer to leave your site without ever really experiencing it.

Planning Your Site

A good idea to get into the habit of practising is planning the content of your Web pages on paper.

Before you start developing pages for your Web site, do the following:

- collect the information you want to include in your Web site
- organise the information into topics.
- lay out the topics to create a storyboard - each topic has it's own page
- create a graphic logo or theme to be used on each page of your site

To be able to easily maintain your Web pages, you should create a separate Web document or page for each topic that you identify. The starting point for a Web site is referred to as the home page or index page. From the home page you provide links to other pages in your Web site. Try to match the look and feel of each page in your Web site to a common standard, perhaps by using the same background and a common

image or text naming the site. This will help the viewer understand the logical content and physical layout of your site.

When you save your Web pages, you should always give the home page a filename of **index.htm**, or **home.htm**. When a browser comes to a Web site it automatically searches for and loads files of either of these names first. All other pages in your Web site should be named appropriately, so that you can anticipate or identify what the file contains without having to load it.

When designing your site, there are three different layout structures you can use – linear, hierarchical, or a combination of both. Which layout you use will depend on the amount of information you want to provide and the complexity of your information.

Linear Layout

A linear approach to Web site organisation can be used when you want the user to view each page sequentially without skipping around. Each page contains only one link, which is to the next page in the sequence. A link on the last page could take the user back to the homepage. You probably won't find a linear approach necessary, but you never know.

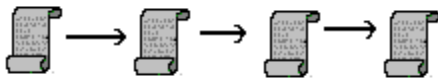


Figure 2.1
Linear Page Layout

Hierarchical Layout

The most common structure of Web sites is the hierarchical layout.

A hierarchical layout begins with a home page, from which the viewers can follow links to and from subordinate pages within the Web site. This allows the user access to specific information in the Web site without having to browse each page.

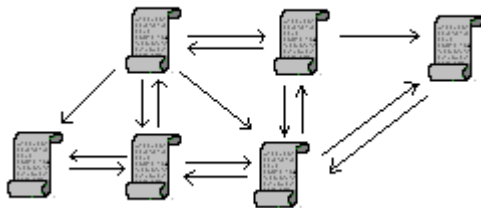


Figure 2.2
Hierarchical Page Layout

Linear and Hierarchical Layout

Some complicated Web sites contain a combination of linear and hierarchical layout.

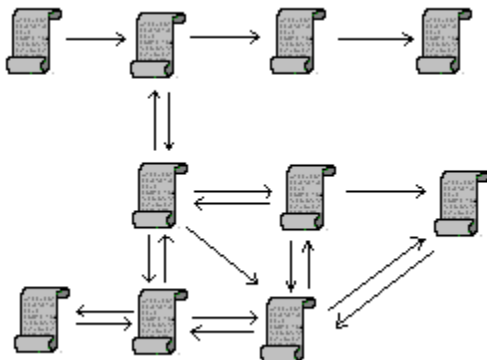


Figure 2.3
Linear and Hierarchical Layout

Chapter 3: HTML Tools

These days, there are an astonishing amount of software applications available that can be used to create Web pages simply.

Some of these applications are shareware (freely downloadable, relatively cheap to purchase the full version). Others are commercial software. For a comprehensive list of what is currently available have a look at this Web site, which lists hundreds of HTML editors by platform type in alphabetical order:

<http://sdg.ncsa.uiuc.edu/~mag/work/HTMLEditors/>

You may be familiar with a few of the popular packages that can be used to design Web pages, such as **Microsoft FrontPage**, and **Microsoft Publisher**. **Microsoft Word 97** even has a “*Save as HTML*” function and earlier versions of Word can also be given this functionality with an additional application called **Internet Assistant for Word**. **Netscape Navigator Gold 3** also has a built in HTML editor, as does the **Netscape Communicator** program, which is Netscape’s latest Web software.

Programs such as these can produce very effective results. However, some of the programs are more reliable than others when it comes to creating the necessary HTML to display your pages in a Web browser. You will also find that the better Web page design applications cost a little more than the not so good ones, and you have to buy them separately.

Fortunately, you do not have to splash out for any new software in order to create really effective Web pages. Any text editor is fine as an HTML editor, even a word processor can be used, so long as you save the file as a text file rather than a document specific to the word processor (such as *.doc in Microsoft Word or *.wpd in WordPerfect). I like to Windows Notepad. Literally every computer comes with Notepad or an equivalent application.

Marking up your own documents gives you maximum control over the final appearance of a Web page. Even if you do choose to use an HTML editor, you will probably need to know a bit about HTML markup in order to touch-up any aspects of your pages that did not appear as desired.

Chapter 4: HTML Tags

Ok, now you have some idea about what HTML is, and you are aware of some of the things you should consider when designing a Web page. You can now begin to explore the language itself.

An HTML *tag* is an instruction to the browser that it should do something other than simply display text on the screen in the default font. Tags are instructions that are offset from the displayable text by less than (<) and greater than (>) signs. For example, the following line of text:

```
<B>Bold</B> characters highlight and emphasise a word or phrase.
```

would look like this in a browser:

Bold characters highlight and emphasise a word or phrase.

The and are HTML Tags. The tag instructs the browser to turn on bold formatting, and the tag instructs the browser to turn it off.

There are two types of HTML tags: container tags and stand-alone tags.

Container Tags

Container tags are tags that contain formatting instructions that, once switched on and applied to a specified object or area of text, must be switched off again. The tag from above is an example of a container tag. and its companion tag cause the text they contain to be displayed in bold. In a container tag pair, the first tag (like) is often called an *opening tag* and the second tag (like) is called a *closing tag*.

Stand-Alone Tags

An *empty* or *stand-alone* tag does not have a companion tag and does not contain displayable text. An example of a stand-alone tag is the tag. is simply used to place a specified image on a Web page. It produces no effect that needs to be carried over any amount of text, so no tag is required.

Two Rules for Browsers Processing HTML

While there are differences between browsers with the support of some HTML functions, there are two rules all browsers follow:

➤ HTML is not case-sensitive

This means that when writing HTML tags, you can use any combination of uppercase and lowercase letters inside the tags. Each of the following examples will be interpreted in the same way:

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

Note: Most HTML authors choose to write their HTML tags in either all uppercase or all lowercase, as this helps to make the tags stand out for ease of editing.

➤ Extra Space is Ignored

Browsers will always recognise the first space after a character, but any immediate additional spaces are ignored. For instance, if you want to indent the first word of a paragraph by three spaces, you have to specifically instruct the browser to include the spaces.

Similarly, tabs and carriage returns (hitting the enter key to move down a line) are also ignored.

Chapter 5: A Basic HTML Skeleton

When creating an HTML document there are a few tags that you must include in order for the rest of your document to appear as you intend it to.

Without the following tags that define the basic structure of an HTML document, you may find you get varying results when viewing your documents with different browsers.

The <HTML> Tag

The first tag you should consider mandatory is the <HTML> tag. <HTML> is a container tag that works together with the closing </HTML> tag to contain the document information. This simply tells the browser that everything between the tags is HTML.

For example:

```
<HTML>
... your document information...
</HTML>
```

The <HEAD> Tag

The document head is one of the two major sections that must be included between the <HTML> and </HTML> tags. The information found in the document head is essential to the inner workings of the document. With the exception of the title, all information placed in the document head ends up being completely transparent to the end user.

The <HEAD> tag defines the document head, and ends with the </HEAD> tag. The <HEAD> tag should immediately follow the <HTML> tag.

For example:

```
<HTML>

<HEAD>
...components of the document head
</HEAD>

...your document information...

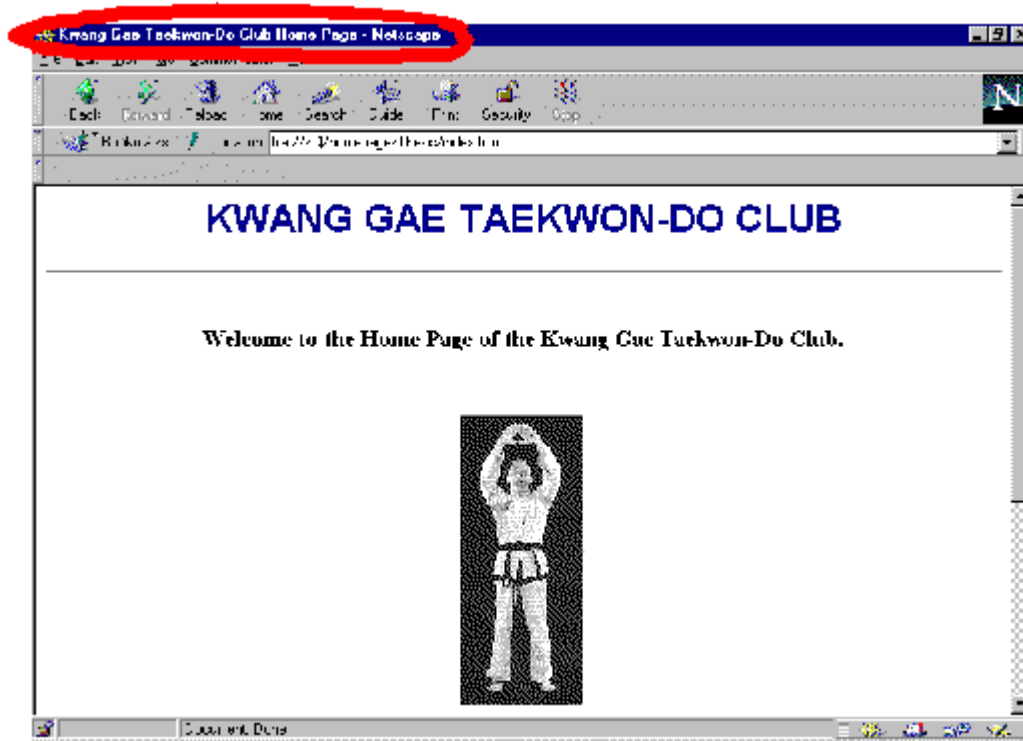
</HTML>
```


The <TITLE> Tag

Perhaps one of the most important elements of the document head is the title of the document, which is placed between the <TITLE> and </TITLE> tags.

The document title serves several important functions:

- *Titles show up at the top of the browser window.*
Titles are the one piece of information in the document head that end users actually see.



- *Titles are used when by your browser when adding bookmarks and creating history lists*
When you save a "Bookmark" or "Favourite" the page is stored by its title. If the author did not include a title the page is stored by it's URL, which is usually much less descriptive than a title.
- *Titles are used by robots to index a page.*
When you register your Web site with a search engine such as Yahoo or Webcrawler, robots scan your site obtaining information for databases to index the content of your pages. Nearly all of them look for a title, and most of them list your site by the title.

The title tag is therefore used like this:

```
<HTML>

<HEAD>
<TITLE>...Document title...</TITLE>
...other components of the document head
</HEAD>
...your document information...

</HTML>
```

The <BODY> Tag

The other major section of an HTML document is the body. After all, who ever heard of a head without a body?

The document body is, of course, contained between the <BODY> and </BODY> tags (HTML is beginning to look pretty simple about now, eh?). Anything you want users to see when they visit your site will be located between these two tags.

The <BODY> tag should come immediately after the </HEAD> tag and the </BODY> tag should be placed immediately before the </HTML> tag, thus your document now looks like this:

```
<HTML>

<HEAD>
<TITLE> ...Document title... </TITLE>
...other components of the document head
</HEAD>

<BODY>

...your document information...

</BODY>

</HTML>
```

A Suggestion of Style

While the above tags will soon become easy to use and remember, many HTML authors frequently forget the document structure tags.

It is a good idea to take a few minutes to set up the following basic template that you can use to start every new HTML document:

```
<HTML>

<HEAD>
<TITLE>    </TITLE>
</HEAD>

<BODY>

</BODY>

</HTML>
```

This gives you an initial starting point to which you can add your text and any other tags you need.

Chapter 6: Body Building

Most HTML tags are meant to be used in the document body, between the `<BODY>` and `</BODY>` tags. Many of them directly affect the text that is displayed in the browser, while others affect the general format of the document.

An important aspect of HTML publishing is remembering to account for every bit of formatting in the browser view.

Attributes of the `<BODY>` Tag

Aside from simply being the container for all of your viewable content, the `<BODY>` tag also has additional attributes that impact on the look of your page. The following table shows these additional attributes and describes their function:

Attribute	Function
BACKGROUND	Defines the URL of an image (GIF or JPEG) used as the document background (like wallpaper on your desktop)
BGCOLOR	Specifies the background colour of the document (Note the American spelling – this is necessary)
TEXT	Sets the colour of the body text
LINK	Sets the colour of unvisited hypertext links
VLINK	Sets the colour of visited hypertext links
ALINK	Sets the colour of active (as you click on them) hypertext links

These attributes can be included in the `<BODY>` tag like so:

```
<BODY BACKGROUND="spots.gif" LINK="#0000FF" VLINK="#FF0000">
```

This example uses the image file *spots.gif* as the background image of the document (the browser tiles the image so the whole window is covered), turns the colour of any unvisited links to blue (0000FF is hexadecimal for blue) and turns the colour of any visited links to red (FF0000 is hexadecimal for red).

Similarly, you can specify a single background colour. This example sets the background to white.

```
<BODY BGCOLOR="#FFFFFF">
```

Fortunately, come the end of the body section of your document, the closing tag remains simply `</BODY>` regardless of any other attributes specified in the opening tag.

Colour Your World

Incorporating colour into your Web pages can produce very effective results, and can also make your pages easier to read. Aspects of an HTML document that can be coloured are its background colour, and its text.

Computers produce colour on their monitors by using varying amounts of the primary colours: red, green and blue (RGB). To specify a colour in an HTML document, you need to define how much of each colour to use.

Colours on computer screens are quantified by using values between 0 and 255. A 0 value specifies no contribution from that particular colour, while a setting of 255 is the maximum contribution available. By choosing values between 0 and 255 you can vary the contribution of a colour, and combining various contributions from red, green and blue, up to 256 colours can be produced.

Finding the numerical red, green and blue values for a specific colour that you have in mind could become an interesting exercise in 'hit and miss' methodology. Fortunately, you can get help from your computer. The **Paint** application that comes bundled with Windows has a handy function called *Edit Colours* where you can define custom colours, which is under the *Options* menu on the toolbar.

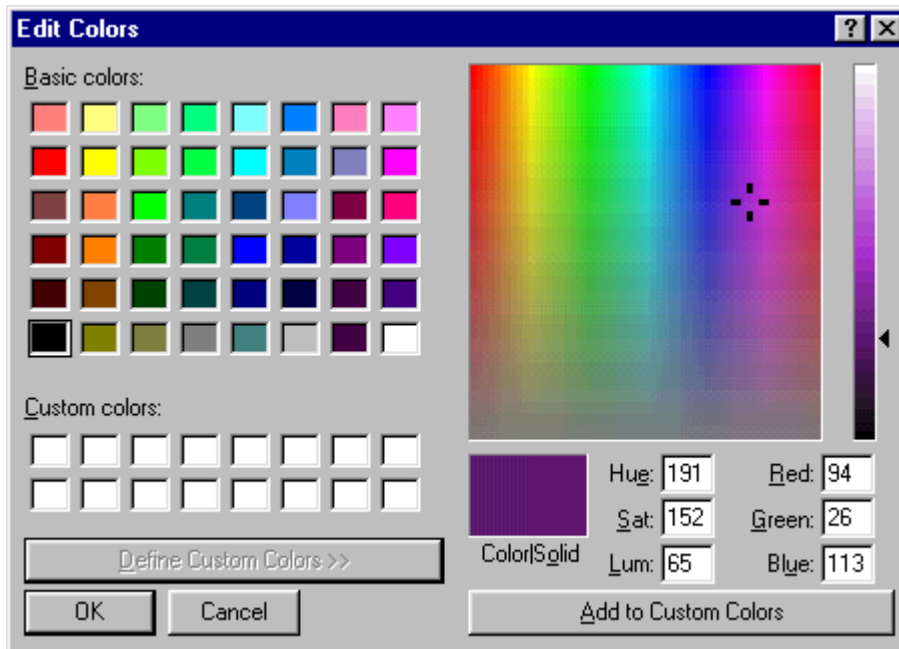


Figure 6.1
Defining Custom Colours

Clicking on the *Define Custom Colours* button displays the section to the right side. Adjusting the far-right scroll bar and dragging the cursor around in the large multicoloured box allows you to find the colour you want and usefully displays the Red, Green and Blue values of that colour.

However, defining colours in HTML must be done in hexadecimal numbers, rather than the decimal numbers we get from the above example. Once again, we can use another common Windows application to convert from hexadecimal to decimal numbers – the **Windows Calculator**.



Figure 6.2
Converting Decimal to
Hexadecimal

Setting the calculator to scientific mode (click on *View* on the toolbar and select *Scientific*) you can enter the decimal number and convert it to hexadecimal by clicking the *Hex* radio button. Once you have calculated each of the hexadecimal red, green and blue values, you can string them together into an *RGB hexadecimal triplet*. For example, a forest green colour has decimal values of 35 red, 142 green, and 35 blue. Converting these to hexadecimal results in 23 red, 8E green and 23 blue. The RGB triplet is formed by simply placing these values together like so:

238E23

When defining colours in HTML markup using an RGB triplet you should place speech marks ("") around the triplet and a hash (#) should also immediately precede the triplet, as in this example which sets the background colour of the HTML document to forest green:

```
<BODY BGCOLOR="#238E23">
```

For a great list of RGB hexadecimal triplets this site is worth a look:
<http://www.phoenix.net/~jacobson/rgb.html>

There are, however, 16 colours that have been built into HTML 3.2 with English names. These colours were selected because they are the 16 colours that make up the Windows VGA palette.

Table 6.1 - English Language Colour Names (HTML 3.2)

Colour Name	Colour Name
Aqua	Navy
Black	Olive
Blue	Purple
Fuchsia	Red
Gray	Silver
Green	Teal
Lime	White

These colours can be implemented this way:

```
<BODY BGCOLOR="OLIVE" TEXT="SILVER">
```

Creating Space

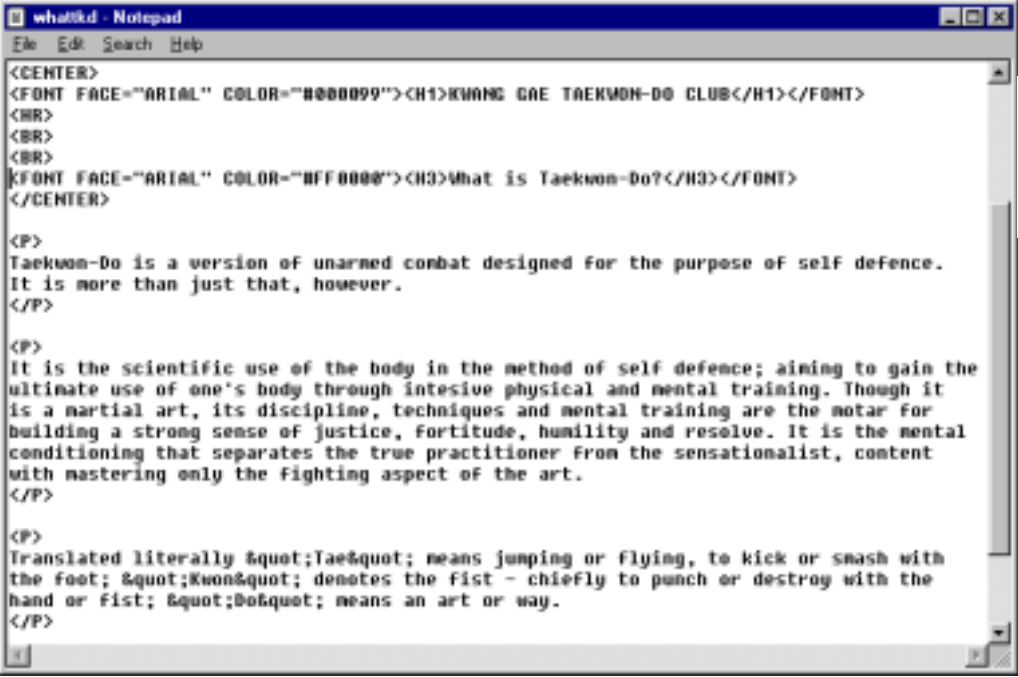
As mentioned earlier, browsers ignore extra white space in HTML documents, such as carriage returns. Simply because your HTML code is laid out on different lines doesn't mean the browser creates a similar type of view.

There are two types of basic line breaks that you will want to utilise in your Web pages, paragraphs and line breaks.

The <P> Tag

The <P> and </P> tags are used to contain a paragraph. Using the paragraph tags will cause the browser to break to a new line, skipping a little bit of space between the previous line and the new line. The <P> tag *does not* indent the first word in the new paragraph.

Figure 6.3 shows an example of how the <P> and </P> tags are used. Figure 6.4 shows how the same example looks in a browser. Note how the paragraphs look in the browser.

A screenshot of a Notepad window titled 'what00d - Notepad'. The window contains HTML code for a page about Taekwon-Do. The code includes a center-aligned header with a font face of 'ARIAL' and color of '000000', followed by a paragraph of text, a heading, and three more paragraphs of text. The code is as follows:

```
<CENTER>
<FONT FACE="ARIAL" COLOR="000000"><H1>KWANG GAE TAEKWON-DO CLUB</H1></FONT>
<BR>
<BR>
<BR>
<FONT FACE="ARIAL" COLOR="FF0000"><H3>What is Taekwon-Do?</H3></FONT>
</CENTER>

<P>
Taekwon-Do is a version of unarmed combat designed for the purpose of self defence.
It is more than just that, however.
</P>

<P>
It is the scientific use of the body in the method of self defence; aiming to gain the
ultimate use of one's body through intensive physical and mental training. Though it
is a martial art, its discipline, techniques and mental training are the motor for
building a strong sense of justice, fortitude, humility and resolve. It is the mental
conditioning that separates the true practitioner from the sensationalist, content
with mastering only the fighting aspect of the art.
</P>

<P>
Translated literally &quot;Tae&quot; means jumping or flying, to kick or smash with
the foot; &quot;Kwon&quot; denotes the fist - chiefly to punch or destroy with the
hand or fist; &quot;Do&quot; means an art or way.
</P>
```

Figure 6.3
Using the <P> and
</P> tags in the
HTML document.

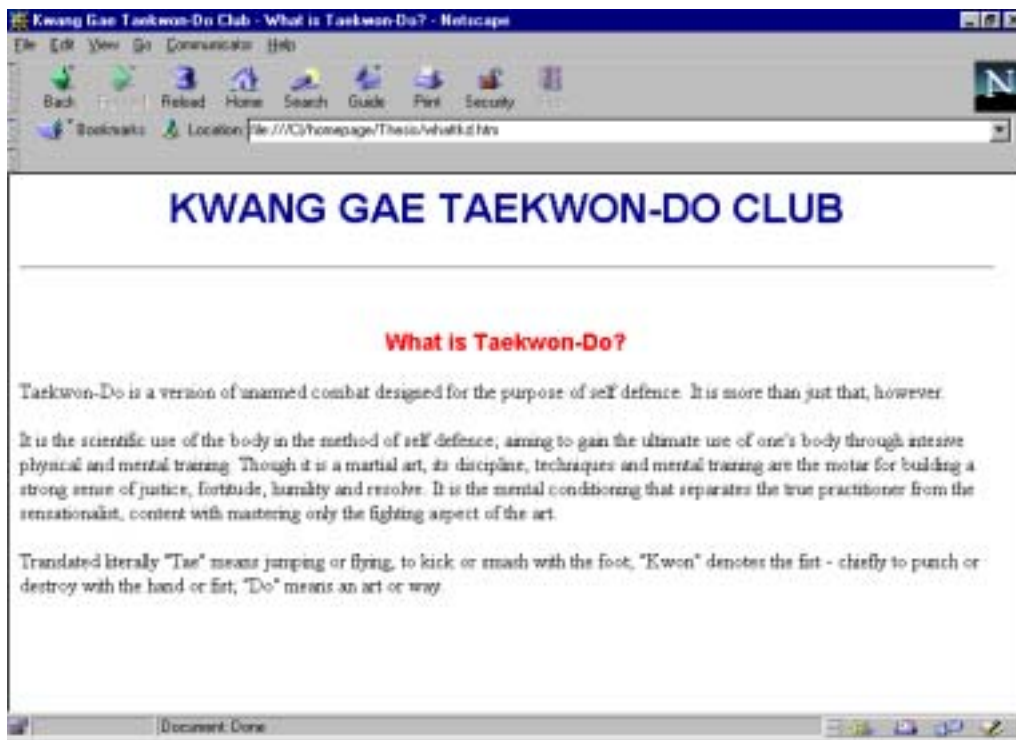


Figure 6.4
The effect of the <P>
and </P> tags.
Paragraphs are formed
with blank lines, but no
indenting

The <P> tag can also take on the ALIGN attribute. ALIGN can be set to LEFT, RIGHT or CENTER, and controls how the text in the paragraph is aligned. By default, paragraph text is left-aligned in the browser screen. To change to centred or right-aligned text, simply include the attribute in the <P> tag, like so:

```
<P ALIGN="LEFT">
Taekwon-Do is a version of unarmed combat designed for the purpose of
self defence. It is more than just that, however.
</P>
```

The text alignment remains until the </P> tag is encountered.

**
eaking out**

The
 tag is used where you need a carriage return (hitting the *Enter* key in most word processors).

For example,

```
White Belt<BR>
Yellow Belt<BR>
Green Belt<BR>
Blue Belt<BR>
Red Belt<BR>
Black Belt<BR>
```

would appear as in Figure 6.5

Without the
 tags, all of the belt colours would appear on the same line.

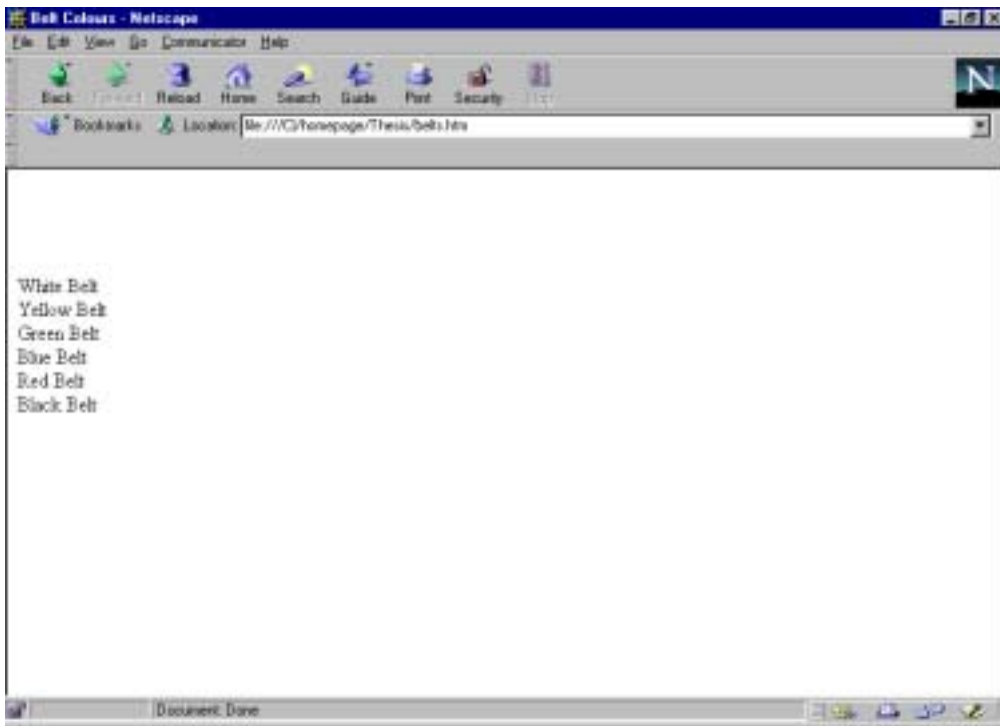


Figure 6.5
Using
 tags

Similarly, having the HTML text like so;

```
White Belt<BR>Yellow Belt<BR>Green Belt<BR>Blue Belt<BR>Red
Belt<BR>Black Belt<BR>
```

would also produce the same effect as in Figure 6.5 when viewed in a browser, due to the function of the
 tags.

Also note, browsers will recognise multiple
 tags, but will ignore multiple <P> tags.

Breaking Up Your Document

It can be difficult for readers to remain interested if they have to scroll through screen upon screen of information. Sometimes it is appropriate to have all of that information on the same page however, so there are some things you can do to break up the information within your page, making the appearance easier on the eye and more appealing to read.

Heading Tags

Heading tags come in six different types, H1 being the largest and H6 being the smallest. The heading tags are container tags, <Hn> (where n is the heading level you want to use, between 1 and 6) precedes the heading text and </Hn> immediately follows the text, as in the following example:

```
<H1>Heading 1</H1>
<H2>Heading 2</H2>
<H3>Heading 3</H3>
<H4>Heading 4</H4>
<H5>Heading 5</H5>
<H6>Heading 6</H6>
```

Figure 6.6 shows the effects of using the various heading tags.

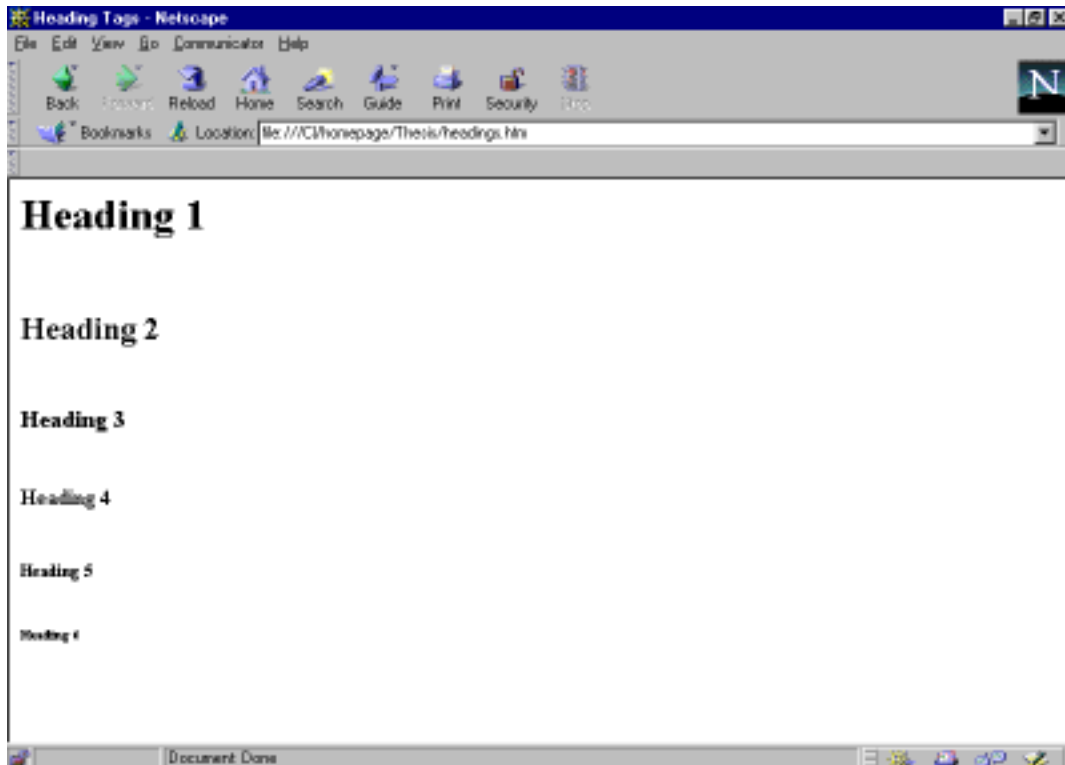


Figure 6.6
Heading styles 1
through 6 in action.

The heading tags can also take the ALIGN attribute. The default alignment is left-justified, but you can set ALIGN to CENTER or RIGHT as required:

```
<H1 ALIGN="CENTER">Heading 1</H1>
<H3 ALIGN="RIGHT">Heading 3</H3>
```

Ruling Lines

Another method of breaking up text in your pages is to add a horizontal line (or rule) between selected sections.

Add a horizontal rule with the <HR> tag. By default, the rule is set to the full width of the screen (with a small amount of space at each end), is one point in size, and has a shading effect to give it a three-dimensional appearance.

The <HR> tag also has some additional attributes, as outlined in Table 6.2.

Attribute	Function
-----------	----------



effect
n
the screen width. As
wide everyone's
75%> .
he right, depending on

Figure 6.7
Using the Horizontal Rule

⇐ Horizontal Rule

Formatting Text

When formatting your actual text in simple HTML, there are two styles of markup tags you can choose from - the logical or physical styles.

Physical HTML

Physical tags explicitly define (to a degree) the characteristics of the text they effect. All of these tags are container tags, as you need to be able to define specifically which text is to take on the tag's attributes. Table 6.3 lists each of the physical tags and their effect.

Table 6.3 - HTML Physical Styles

Tag	Effect
<code> </code>	Displays text in Boldface
<code><I> </I></code>	Displays text in Italics
<code><BIG> </BIG></code>	Renders text in a large font
<code><SMALL> </SMALL></code>	Renders text in a small font
<code><SUB> </SUB></code>	Displays text as a Subscript
<code><SUP> </SUP></code>	Displays text as a Superscript
<code><STRIKE> </STRIKE></code>	Displays Strikethrough text
<code><TT> </TT></code>	Displays text in a fixed-pitch typewriter font

Logical HTML

The interpretations of logical tags may vary from browser to browser, as they indicate the meaning of the text they contain in the context of the document. For instance, the tag, which indicates emphasised text, may render text in browser A as italics while browser B may display the text in boldface.

Table 6.4 lists available logical tags and their typical effect.

Table 6.4 – HTML Logical Styles

Style	Tag	Typical Effect
Emphasis	 	Displays text in Italics
Strong Emphasis	 	Displays text in Boldface
Code	<CODE> </CODE>	Displays text in a Fixed-width font
Definition	<DFN> </DFN>	Displays text in Bold or Bold Italics
Keyboarded	<KBD> </KBD>	Displays text in a Fixed-width font
Citation	<CITE> </CITE>	Displays text in Italics
Sample text	<SAMP> </SAMP>	Displays text in a Fixed-width font
Variable	<VAR> </VAR>	Displays text in Italics

Figure 6.8 shows a page containing text marked up using the listed physical styles, with the corresponding HTML shown in Figure 6.9.

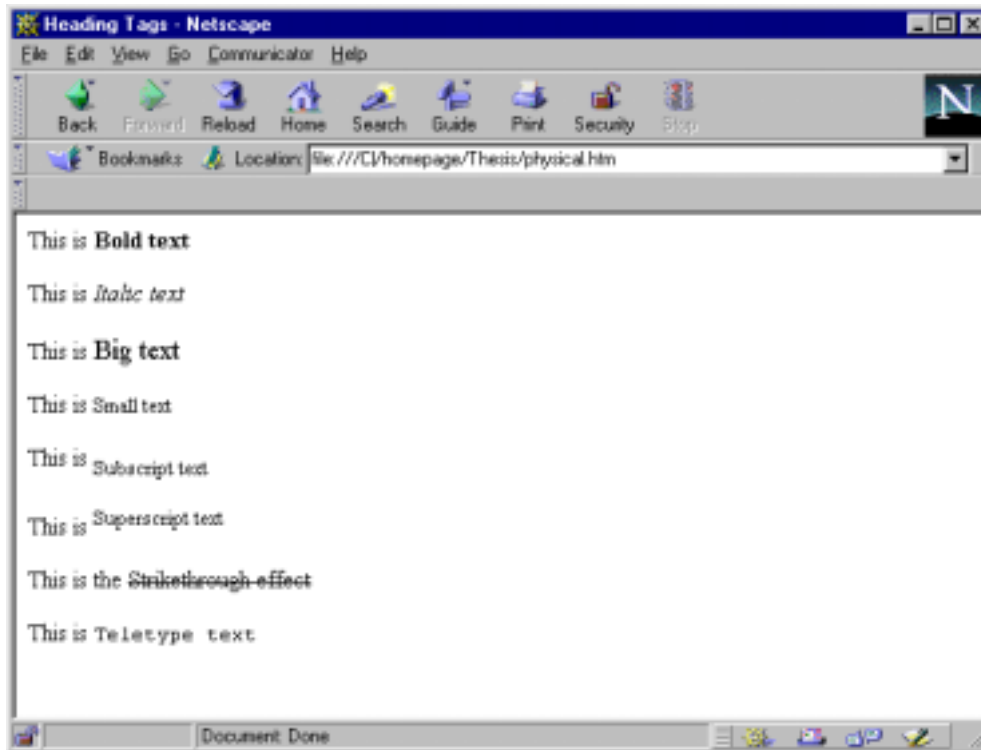


Figure 6.8
Physical Style
HTML Tag effects

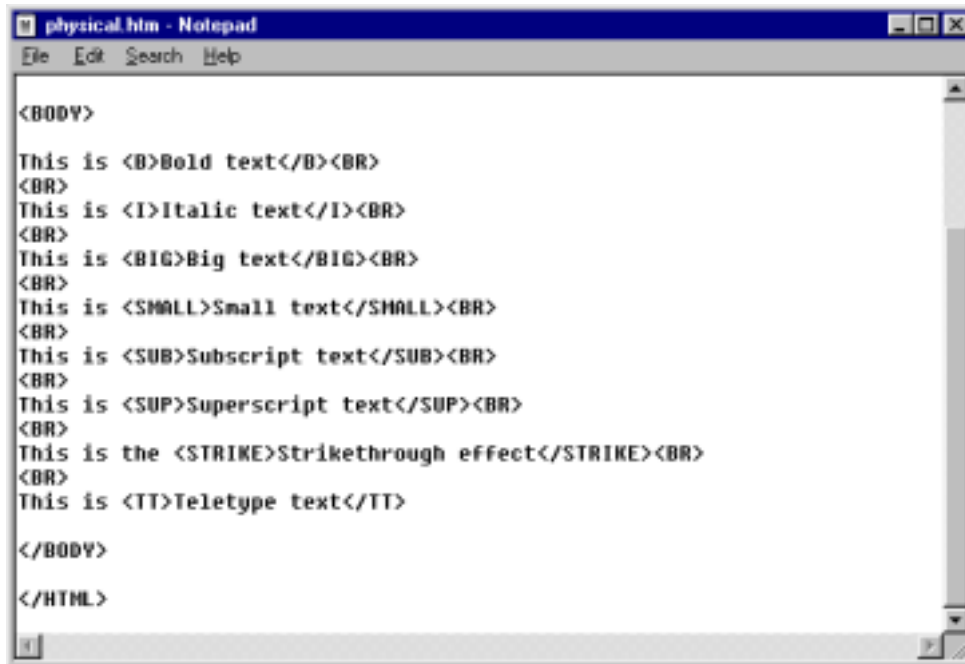


Figure 6.9
Physical Style HTML
Tags in use

Figure 6.10 shows a page containing text marked up using the listed logical styles, with the corresponding HTML shown in Figure 6.11.

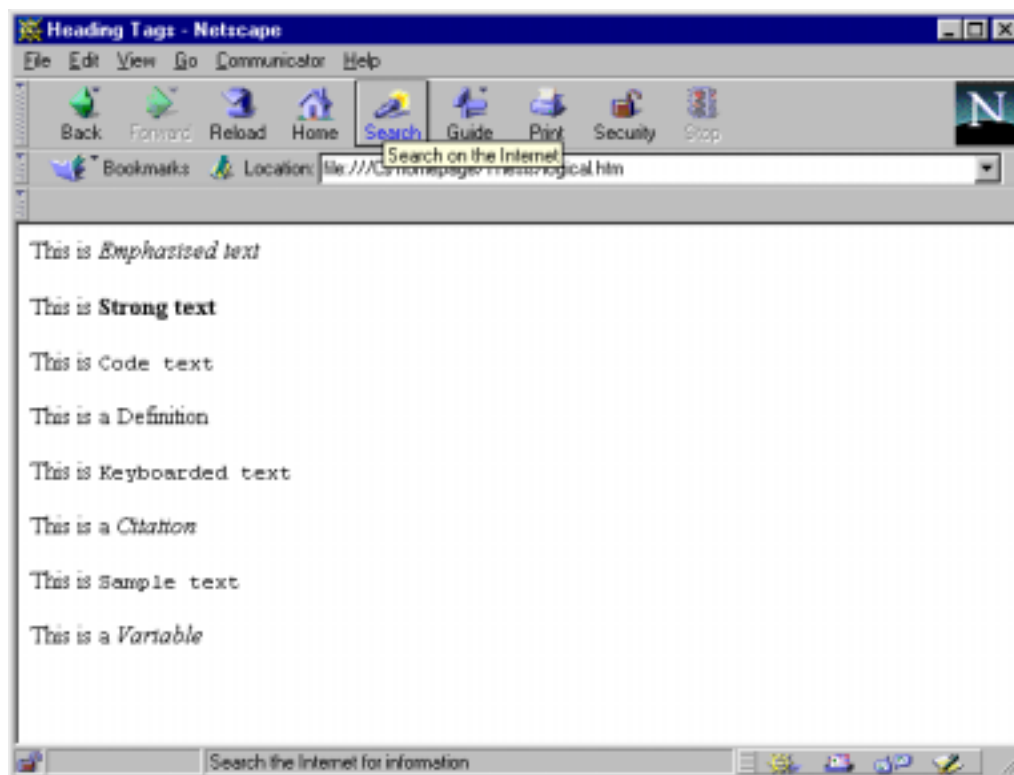


Figure 6.10
Logical Style
HTML Tag effects

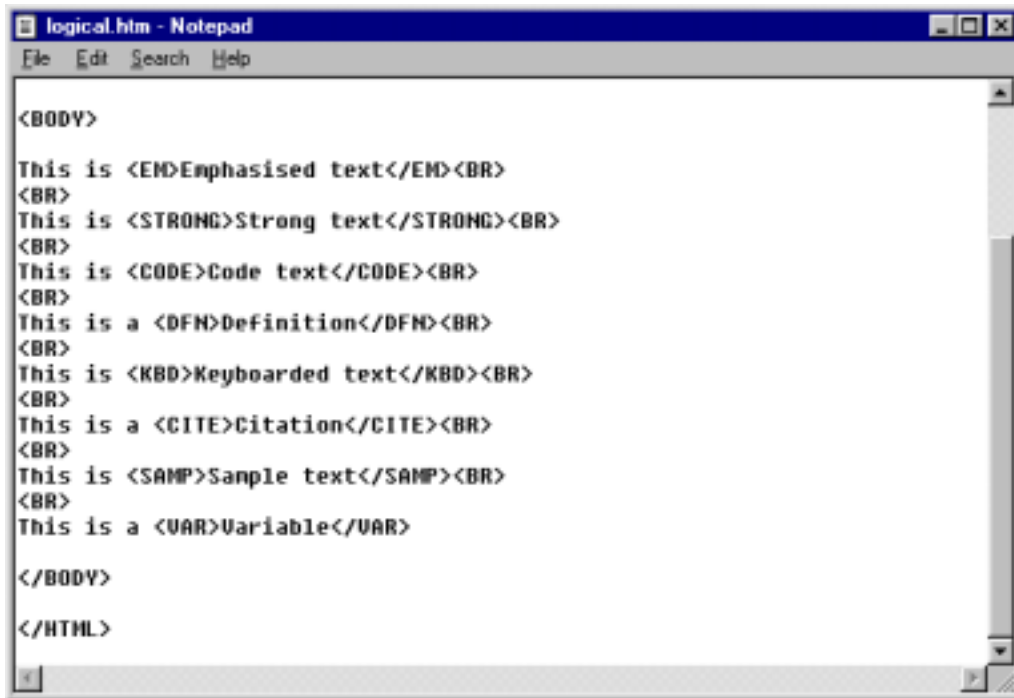


Figure 6.11
Logical Style
HTML Tags in use

Choosing which of the two styles to use depends on what you want to achieve in your markup. If you are simply looking for on-screen effect, stick with the physical style tags. If indicating the meaning of the text you are marking up is important, use the logical style tags.

Other General Text Effects

Besides the tags in the physical and logical styles, there are also other HTML tags that allow you to create other on-screen formatting effects.

Centring Text

You can centre most aspects of your Web pages using the <CENTER> container tag. Note the American spelling – yes, this is something that can take some getting used to, but it is necessary to spell it correctly or your tag will be ignored.

If you only need a specific part of a Web page centred, you may be better off using the alignment attribute of a <P> tag, but the <CENTER> and </CENTER> tags do achieve a general centring solution whereby they centre everything found between them – text, tables, images, horizontal rules, lists and any other page element.

Figure 6.12 shows the HTML of a document utilising the <CENTER> and </CENTER> tags.

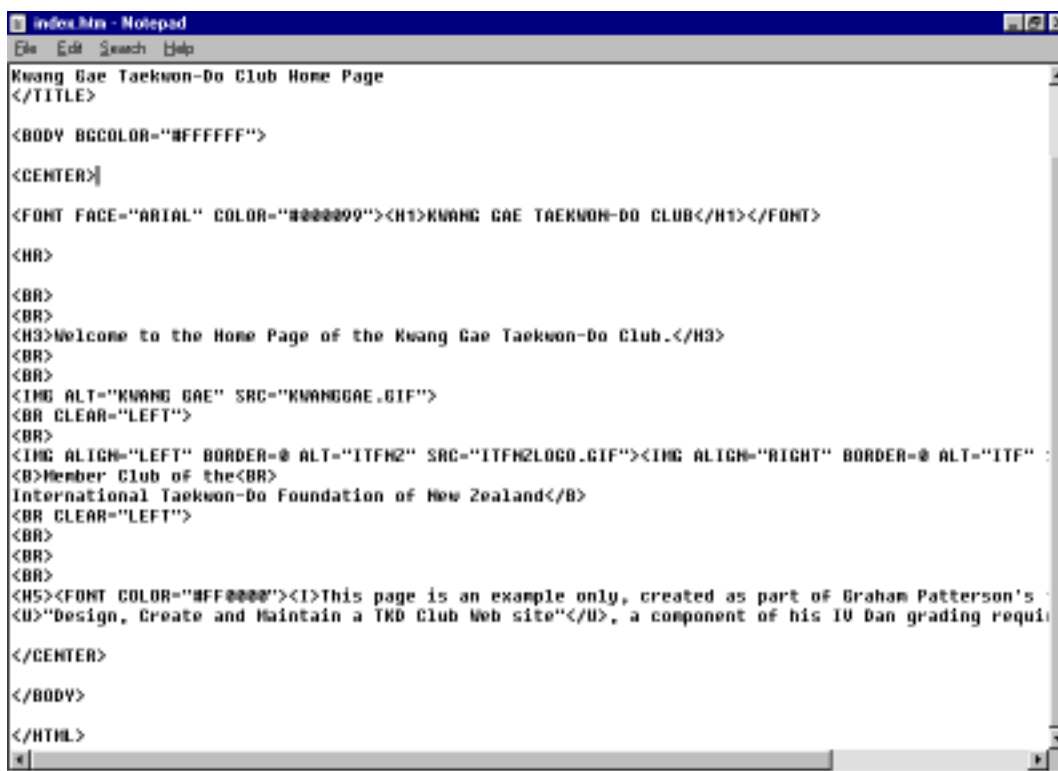


Figure 6.12
Using the
<CENTER> tag.

Underlining Text

You can underline text using the <U> and </U> container tags, for example:

```
<U>Really important phrase</U>
```

would produce an effect not dissimilar to:

Really important phrase

Altering Font Properties

There are a number of things you can do to fonts using HTML. The default font displayed depends on the browser that is being used. Usually the default font is Times Roman and is about 10 point.

Using attributes of the and tags you can control the properties of the fonts the viewers see.

Table 6.5 lists the attributes of the tag.

Table 6.5 – Attributes of the Tag

Attribute	Effect
SIZE= <i>n</i> or +/- <i>n</i>	Increase/Decrease the font size by <i>n</i>
COLOR	Define the colour of the enclosed text
FACE	Specify the font face

SIZE can be set to a value between 1 (smallest) and 7 (largest), or you can specify how many times larger or smaller than the base font size you want to make the enclosed text. The default base font size is 3, so using `` is the same as using ``

The colour of the text can be altered using the COLOR attribute. The colours are defined by specifying either the hexadecimal figure for the colour, or the name of a supported English language colour as discussed earlier in this chapter. For example, to change the text colour to red, you could use either `` or ``.

Using the FACE attribute you can specify alternate typefaces to be used, rather than just Times Roman or Courier. For a browser to be able to change the font, however, the selected font must be available on the viewers PC. With this being the case, it is a good idea to keep your list of fonts limited to the standard ones found on most PC's and Mac's. These include Helvetica, Arial, Times Roman, Courier, Palatino, and Garamond.

You can try using some different font faces, such as Brush Script, but if you do it is a good idea to provide the browser with alternative font faces to use, should the initial font not be available on a readers PC. This can be achieved by using the FACE property like so:

```
<FONT FACE="BRUSH SCRIPT,PALATINO,TIMES ROMAN">Text goes here</FONT>
```

If the first font is not available, in this instance Brush Script, the browser tries the second. If the second option is also unavailable, the third is tried and so on. It is a good idea to have one of the chosen fonts set as either Times Roman or Courier, as these fonts are sure to be available on any PC. While the viewer may not be able to view your page with the effect that you intended, being able to see something is better than not seeing anything at all, which can sometimes be the case if the browser cannot interpret your chosen fonts.

Figure 6.13 shows examples of HTML markup using the `` tag and its attributes, while Figure 6.14



Figure 6.14
Using differing
typefaces within a
Web page.

The <ADDRESS> Tag

You may want to disclose your postal or email address on your Web pages so that interested people and potential students can get in touch with you directly.

Enclosing your details in the <ADDRESS> and </ADDRESS> tags like so:

```
<ADDRESS>contact@isp.co.nz</ADDRESS>
```

can make it easier for robots or people viewing the HTML source of your document to get your contact information. The <ADDRESS> tag usually renders the contained text in italics.

The <BLOCKQUOTE> Tag

When you want to quote a passage from another source, you can use the <BLOCKQUOTE> tag to define the quoted text from the rest of the document. The <BLOCKQUOTE> tag produces indented margins on both the left and right sides.

Figure 6.15 shows an example of HTML markup using incorporating the <BLOCKQUOTE> tag, and Figure 6.16 shows the resulting effect.



Figure 6.15
The <BLOCKQUOTE>
Tag in HTML markup

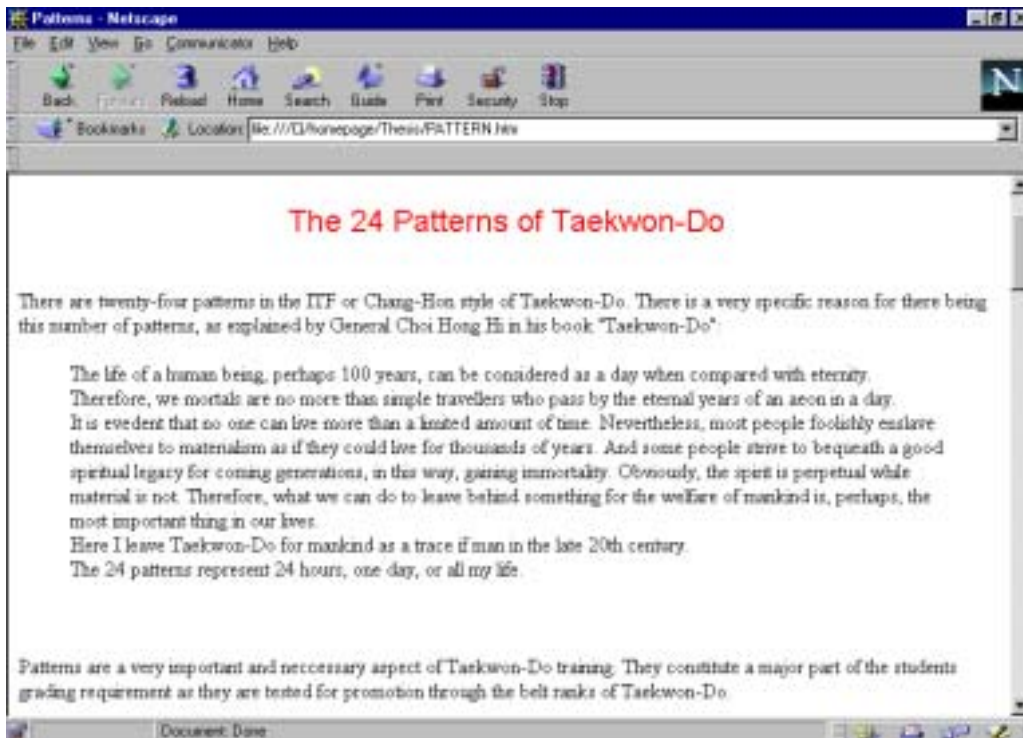


Figure 6.16
The resulting effect of
the <BLOCKQUOTE>
Tag.

Preformatted Text

Preformatted text is displayed in a fixed-width font (like courier) and includes any extra white space, such as spaces, tabs and carriage returns. This function made preformatted text an early approach for creating tables in HTML documents, as it was easy to line up text with a fixed-width font due to all characters having the same width.

Preformatted text is defined using the `<PRE>` and `</PRE>` tags, as in the following example:

```
<PRE>
DAY      TIME      VENUE
Monday   6:30-8:00pm  Town Hall,
Thursday 6:30-8:00pm  Main Street,
Saturday 9:00-11:30am  Suburbia
</PRE>
```

which would appear similar to the example in Figure 6.17.



Figure 6.17
Preformatted text can be used to layout text using tabs and carriage returns.

Special and Reserved Characters

HTML supports several special and reserved characters (non-textual characters) that can be handy.

The entities of these characters all begin with an ampersand (&) and end with a semicolon (;). Table 6.6 lists the reserved and special characters and their respective entities.

Table 6.6 – Special and Reserved Characters

Character	Description	Entity
>	Greater-than sign	>
<	Less-than sign	<
&	Ampersand	&
“	Quotation mark	"
©	Copyright symbol	©
®	Registered trademark	®
	Non-breaking space	

The non-breaking space () can be particularly useful when formatting your text. Use a non-breaking space between two words you don't want separated by an automatic line break, or use it to force extra spaces where you need them (remember, browsers ignore multiple keyboard spaces when processing HTML markup).

Chapter 7: Creating Lists

Lists are a great way to effectively display information in a logical way. They can improve the readability and enhance the general layout of your Web pages.

Currently supported list formats are:

- Unordered (bulleted) lists
- Ordered (numbered) lists
- Definition lists

Unordered Lists

Unordered lists are used for items that do not have a specific order to them. Unordered lists are also called bulleted lists, because, by default, the list items are displayed with some sort of bullet item in front of them.

Unordered lists start with the opening `` tag. Each item in the list is contained within `` and `` tags. The list is closed with the `` tag.

```
<H2>Grading Requirements</H2>

<UL>
<LI>Patterns</LI>
<LI>Step Sparring</LI>
<LI>Theory</LI>
<LI>Fundamental Exercises</LI>
</UL>
```

The default bullet type can be changed by incorporating the `TYPE=` attribute in either the `` or `` tags. There are three types of bullet you can use, `DISC` for a solid circle (usually the default), `CIRCLE` for an open circle, and `SQUARE` for a solid square.

Using the `TYPE` attribute within the `` tag changes the bullet type for the whole list. Using it within the `` tag allows you to change bullet types for each item. Examples of the HTML for both functions are below. Figure 7.1 shows the standard default list from above and both of the below lists also.

```
<H2>Grading Requirements</H2>

<UL TYPE="CIRCLE">
<LI>Patterns</LI>
<LI>Step Sparring</LI>
<LI>Theory</LI>
<LI>Fundamental Exercises</LI>
</UL>
```

```
<H2>Grading Requirements</H2>

<UL>
<LI TYPE="CIRCLE">Patterns</LI>
<LI TYPE="DISC">Step
Sparring</LI>
<LI TYPE="SQUARE">Theory</LI>
<LI>Fundamental Exercises</LI>
</UL>
```

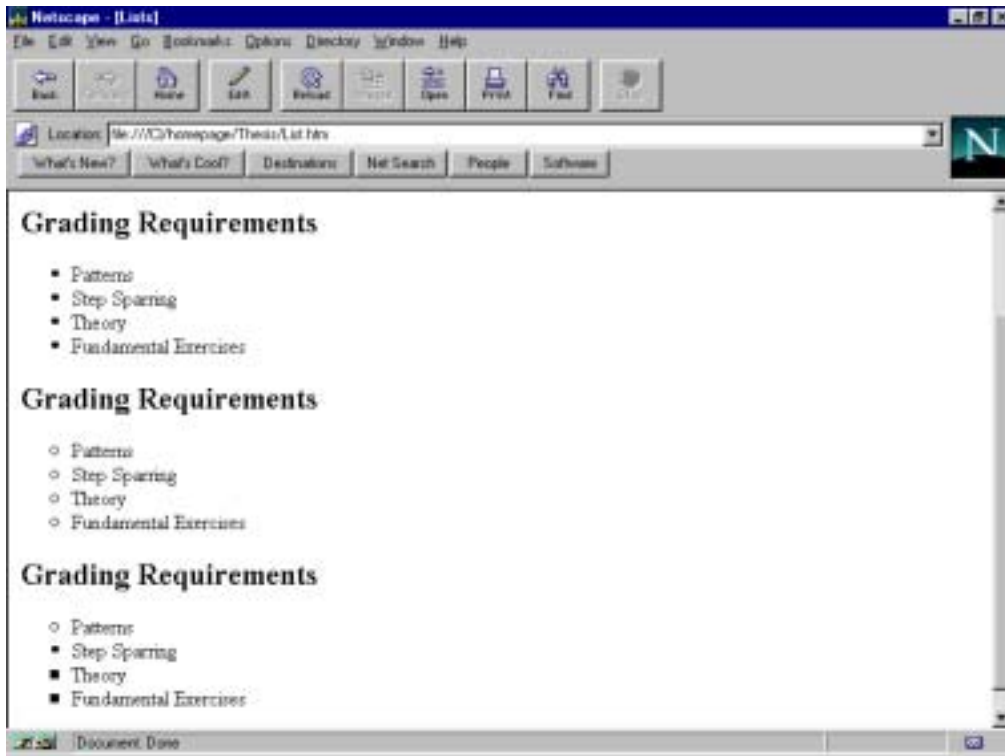


Figure 7.1
Unordered lists
with varying bullet
types.

Note in the last example, that by not specifying a bullet type in the last tag the bullet type has taken the attribute from the nearest above tag attribute.

Unfortunately, not all browsers support the TYPE attribute. For instance, the above HTML examples appear as per Figure 7.2 in Microsoft Internet Explorer 3.0.

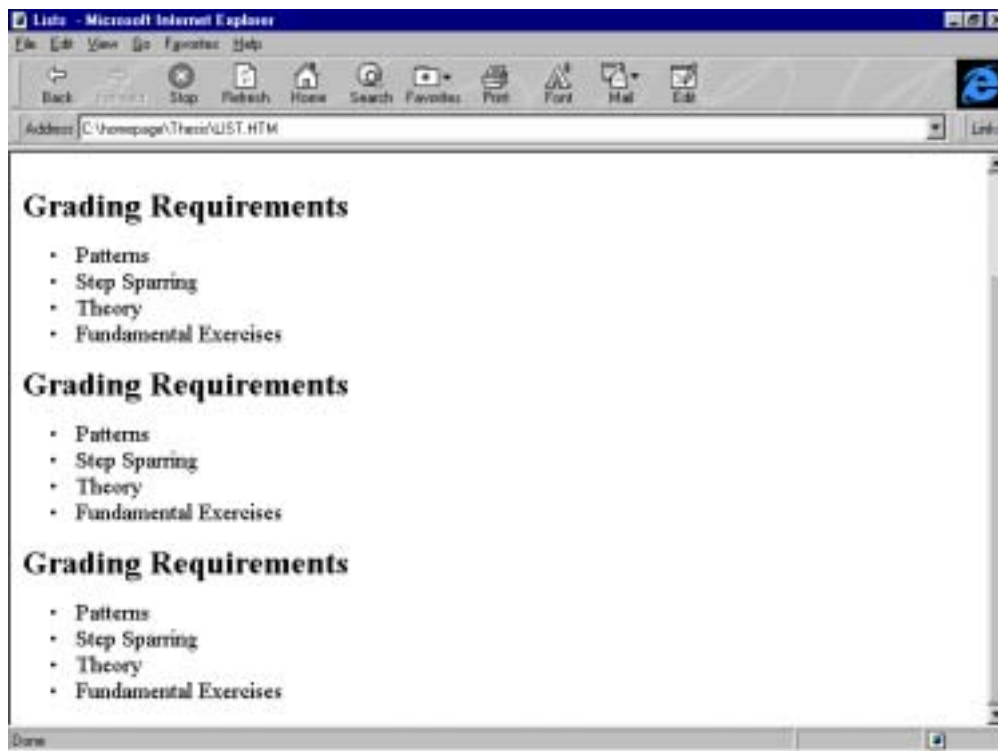


Figure 7.2
Some browsers do not
support the TYPE
attribute.

Ordered Lists

Items in an ordered list have a specific sequential order. Each item is numbered or lettered in ascending or descending order.

As far as HTML markup goes, ordered lists work very much the same way that unordered lists work. All list items are contained within the `` and `` tags and each item is contained within the `` and `` tags. For example,

```
<H2>My Favourite Taekwon-Do Techniques</H2>
<OL>
<LI>Side Piercing Kick</LI>
<LI>Turning Kick</LI>
<LI>Forefist Punch</LI>
<LI>Knifehand Strike</LI>
</OL>
```

appears as in Figure 7.3. Note that the browser has inserted the numbers of the list items automatically. This is especially handy when you have to add, delete, or rearrange list items, as the numbering is done for you.



Figure 7.3
Ordered Lists
automatically
number list items.

You can change the numbering scheme by assigning the `TYPE` attribute to either the `` or `` tags. `TYPE` can be set to the values detailed in Table 7.1.

Table 7.1 – Possible Values of the TYPE Attribute

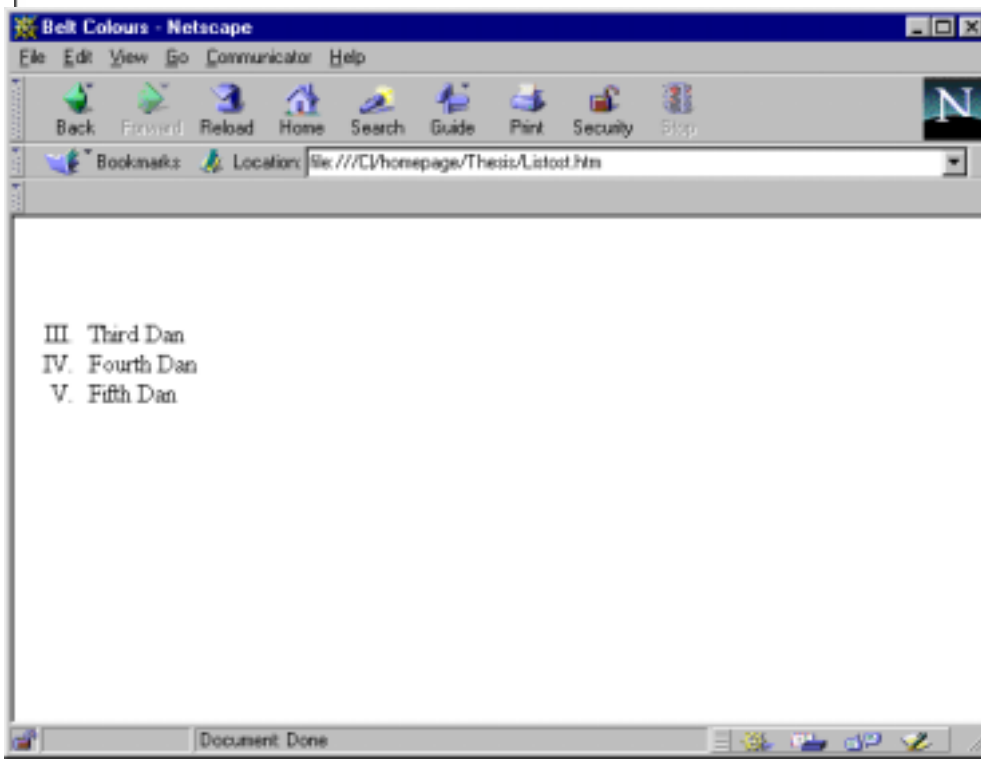
Value	Numbering Scheme
TYPE="1"	Numbers (1, 2, 3, ...)
TYPE="A"	Uppercase letters (A, B, C, ...)
TYPE="a"	Lowercase letters (a, b, c, ...)
TYPE="I"	Uppercase Roman Numerals (I, II, III, ...)
TYPE="i"	Lowercase Roman Numerals (i, ii, iii, ...)

Using the TYPE attribute in the tag effects all list items, while using it in the tag allows you to alter any give list item.

The numbering sequence can also be altered. You can change the value at which the numbering starts by using the START attribute of the tag. This can be useful when your ordered list is interrupted by a non-list element on the page. When you resume the list you can set the START attribute to the appropriate starting value, as in Figure 7.4.



Figure 7.4
Using the START attribute to resume an interrupted list.



eme. The correct value.

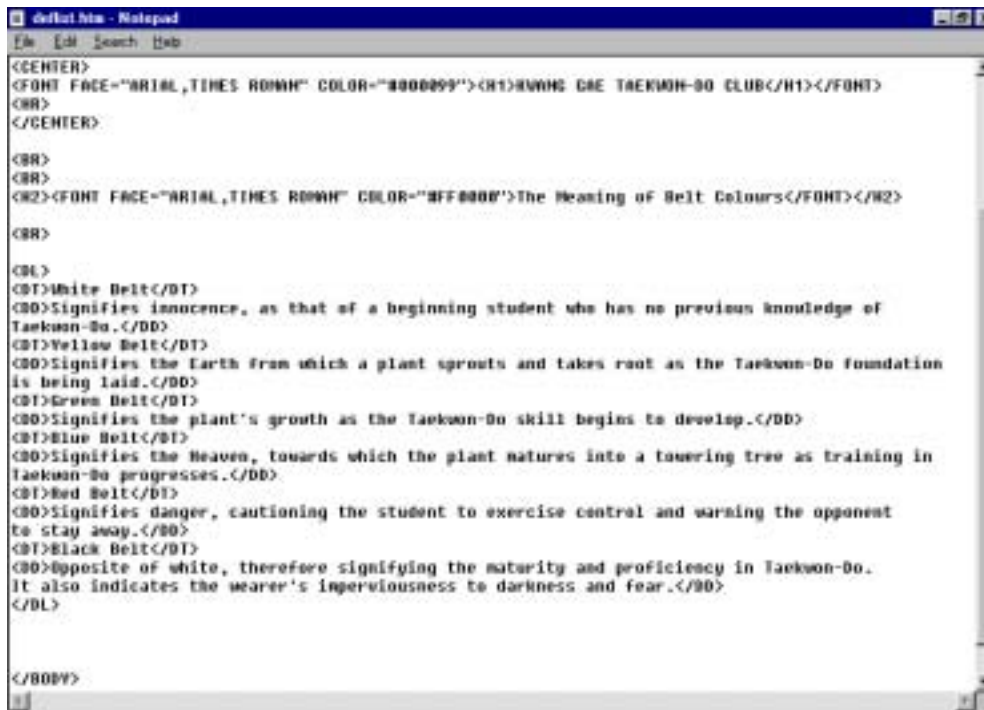
Figure 7.5
Using the TYPE and START tags together. The browser automatically determines which character to start numbering with based on the values for both of these attributes.

Definition Lists

Another type of list that is available in HTML is a definition list. Definition lists can be used to display information in the style of a glossary, listing a term followed by its definition.

Definition lists are found between the `<DL>` and `</DL>` tags. Each term in the list is marked up with `<DT>` and `</DT>` tags and each definition with `<DD>` and `</DD>` tags.

Figure 7.6 shows an example of the HTML markup of a definition list.



```
<CENTER>
<FONT FACE="ARIAL,TIMES ROMAN" COLOR="000099"><H1>WONG GAE TAEKWON-DO CLUB</H1></FONT>
<BR>
</CENTER>

<BR>
<BR>
<H2><FONT FACE="ARIAL,TIMES ROMAN" COLOR="FF0000">The Meaning of Belt Colours</FONT></H2>

<BR>

<DL>
<DT>White Belt</DT>
<DD>Signifies innocence, as that of a beginning student who has no previous knowledge of
Taekwon-Do.</DD>
<DT>Yellow Belt</DT>
<DD>Signifies the Earth from which a plant sprouts and takes root as the Taekwon-Do foundation
is being laid.</DD>
<DT>Green Belt</DT>
<DD>Signifies the plant's growth as the Taekwon-Do skill begins to develop.</DD>
<DT>Blue Belt</DT>
<DD>Signifies the Heaven, towards which the plant matures into a towering tree as training in
Taekwon-Do progresses.</DD>
<DT>Red Belt</DT>
<DD>Signifies danger, cautioning the student to exercise control and warning the opponent
to stay away.</DD>
<DT>Black Belt</DT>
<DD>Opposite of white, therefore signifying the maturity and proficiency in Taekwon-Do.
It also indicates the wearer's imperviousness to darkness and fear.</DD>
</DL>

</BODY>
```

Figure 7.6
Marking up a
Definition List.

Figure 7.7 shows how the above HTML appears in a browser. Note how the definitions are indented from the terms above them.



Figure 7.7
A definition list makes
it easy to read each
term and definition.

To make the terms stand out from the definitions a little more, you can use the container tag to render the terms within the list in bold. For example:

```
<DL>
<DT><B>White Belt</B></DT>
<DD>Signifies innocence, as that of a beginning student who has no
previous knowledge of Taekwon-Do.</DD>
</DL>
```

Chapter 8: Using Graphics and Pictures

One of the great things about the Web is that it supports graphical content. Graphics create visual appeal, and it is the use of graphics that keep viewers on-line for hours. Graphics can also be an effective means of conveying a message, at times better than plain text can. As they say, a picture is worth a thousand words.

Graphic Storage Formats

The Web currently supports two types of graphic storage formats, or images, that display inline (directly in the content of a Web page without the need for a helper application to display the picture) – GIF and JPEG.

GIF

GIF stands for *Graphics Interchange Format*. Originally developed for users of CompuServe as a standard for storing image files, GIF standards have had a few revisions since their inception. The current standard is GIF89a.

Graphics stored in GIF (*.gif) are limited to 256 colours. Because full-colour photos require more than 256 colours to look sharp, you shouldn't store them as GIFs. GIF is best used with logos, icons, and simple images.

In spite of having a limited number of colours, the GIF89a standard supports the following Web page effects:

➤ Interlacing

As the browser reads an interlaced GIF, the image appears to fade in over several passes. This can give the viewer an idea of what the image look like without having to wait for the entire image to load. This can be particularly handy if you use images that have a large file size, and may take some time to load.

➤ Transparency

In a transparent GIF, one of the colours is designated as transparent. This allows the background of the document to show through. Figure 8.1 shows the difference between a transparent and non-transparent GIF. Notice how the border around the non-transparent GIF is visible. By specifying the colour of the bordering box (in this case, black) to be transparent, the background colour shows through and the logo appears to just be sitting on the background.

There are a number of graphics programs that support the creation of transparent GIFs. LView Pro allows you to create transparent GIFs on the PC, and another program called Giftrans allows you to create transparent GIFs from existing images. Both of these programs can be downloaded from <http://www.shareware.com/>.

➤ Animation

Animated GIFs are created by storing the sequence of images used to produce the animation in one file. A browser that fully supports the GIF89a standard is designed to present the images in the file one after the other to produce the animation. The programs that let you store the multiple images in the GIF file also let you specify how much delay should occur before beginning the animation and how many times the animation should repeat. One of these animation programs is GIF Construction Set, and is also available for download from <http://www.shareware.com/>. For an example of an animated GIF, check out the animated reverse turning kick picture on my club's photos page at <http://khdnwltdk.wellington.net.nz>.



Figure 8.1
Transparent GIFs
make images appear
to blend into the
background.

JPEG

JPEG is an acronym for “Joint Picture Experts Group”, and refers to a set of formats that support full-colour images and stores them in a compressed form. JPEG is a 24-bit storage format that allows for 16 million colours.

While JPEG is great for full-colour images such as photos, it does not have some of the functionality of the GIF. Transparency is not possible with JPEG images, nor is animation.

However, a recently emerged format called p-JPEG (*Progressive JPEG*) gives the effect of an image fading in similar to an interlaced GIF.

Which Format Should You Use?

When deciding which image format to use in your Web pages, consider the following:

- Does your image require a transparency effect? If so, you will have to use a GIF because it is the only format that supports transparency.
- Do you have a number of images you want to animate? Again, the easiest method is to use an animated GIF.
- Is your graphic a full-colour image, like a photo? Full-colour images are best stored as JPEGs, which support 16 million colour.
- Does your graphic have any sharp colour changes or boundaries? Because of the mathematics behind the compression algorithm, JPEGs don’t cope well with sudden colour changes. A GIF would be a better choice.
- If you require a fade-in effect, you can use either an interlaced GIF or a p-JPEG.

A good rule of thumb when choosing your image format is to use JPEG for colour photos and use GIF for all other graphics and illustrations.

The Tag

Once you have a copy of the image you want to use, you need to use the tag to place the image on a page. is a stand-alone tag that can take on a number of other attributes, outlined in Table 8.1. Only SRC is mandatory, but you will soon want to use a few more of them.

Table 8.1 – Additional Attributes of the Tag.

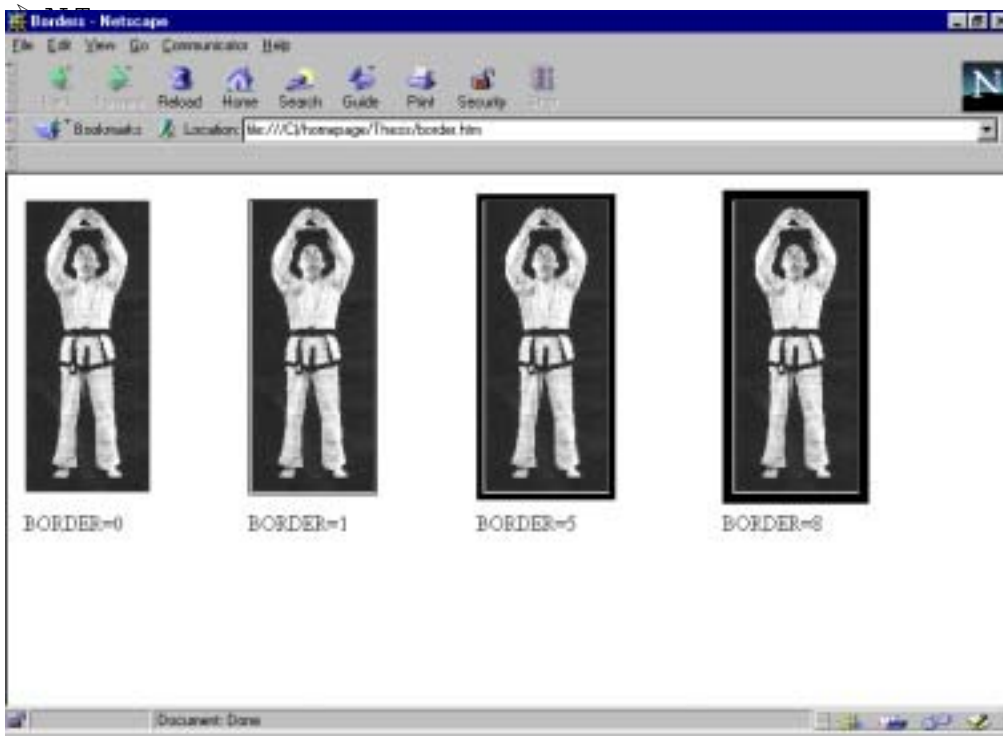
Attribute	Description
ALT	Provides a textual description of the image, should the image not be loaded.
ALIGN	Controls alignment of text following the image.
BORDER	Specifies the size of the border to be placed around the image.
HEIGHT	Specifies the height of the image in pixels.
HSPACE	Controls the amount of white space to the left and right of the image.
ISMAP	Denotes an image is to be used as part of a server-side image map.
SRC	Specifies the URL of the file where the image is stored.
USEMAP	Specifies an image-map to use with the image.
VSPACE	Controls the amount of white space above and below the image.
WIDTH	Specifies the width of the image in pixels.

Inserting a Basic Image

Though SRC is the only mandatory attribute, you should perhaps also get into the habit of also using the HEIGHT, WIDTH and ALT attributes.

➤ HEIGHT and WIDTH

By providing the HEIGHT and WIDTH information, you effectively speed up the page layout process and allow users to see your pages faster. Browsers use the HEIGHT and WIDTH values of the tag to reserve a space for the image, in which it places the image once it has finished downloading. If you don't use these two attributes, the browser has to download the entire image, compute it's size, place it on the page, and then continue loading the rest of the page. If you have a lot of graphical content, leaving these two attributes out can seriously delay the presentation of your page.



Browsers
idea to provide
b, because

3..

o the number

Figure 8.2

Use borders to create a framed look around your images. These borders are all of varying width.

Increasing Space Around Your Image

Increasing the white space around your image can make your images stand out better by reducing clutter. The attributes that control the extra white space are HSPACE and VSPACE. Each is set to the number of pixels of extra space to leave to the left and right of the image (HSPACE) or above and below the image (VSPACE). Figure 8.3 shows an example of using HSPACE, and Figure 8.4 shows an example of using VSPACE.

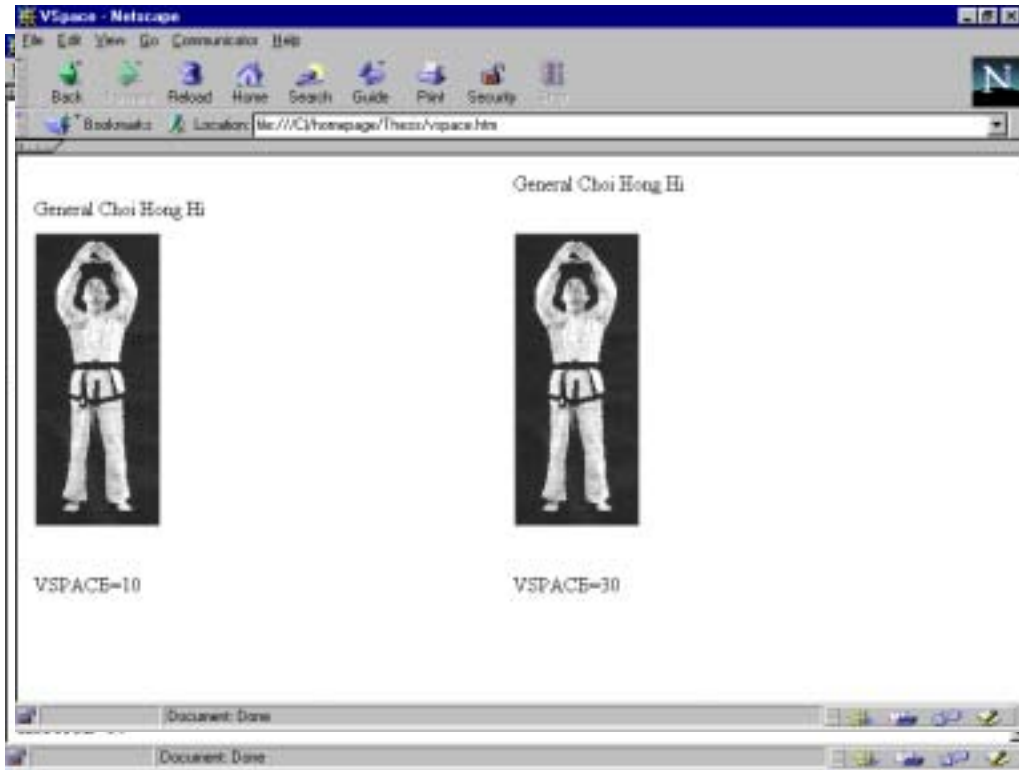


Figure 8.4
Increasing VSPACE puts more white space above and below an image.

The ALIGN Attribute and Floating Images

The ALIGN attribute of the tag can be set to one of five different values. These values are listed in Table 8.2.

Table 8.2 – Values of the ALIGN Attribute

Value	Description
TOP	Aligns the top of subsequent text to the top of the image
MIDDLE	Aligns the baseline of subsequent text with the middle of the image
BOTTOM	Aligns the baseline of subsequent text with the bottom of the image
LEFT	Floats the image in the left margin and allows text to wrap around the right side of the image.
RIGHT	Floats the image in the right margin and allows text to wrap around the left side of the image.

➤ TOP, MIDDLE, and BOTTOM Alignment

These three values refer to how text is aligned following the image. Examples of each are in Figure 8.5.

It is important to note with the TOP and MIDDLE alignments that if the text reaches a point where it needs to break, it begins the text after the break below the image, leaving some white space between the broken text (see Figure 8.6).

➤ LEFT and RIGHT Alignment

Using these values of ALIGN creates a floating image in the left or right margin of the page.

Floating images were adopted as part of the HTML 3.2 standard, and opened the door to many creative and interesting layouts. Figure 8.7 shows examples of both, while Figure 8.8 shows the corresponding HTML



Figure 8.5

Setting ALIGN to TOP, MIDDLE, or BOTTOM controls where any subsequent text will be placed in relation to the image.

document.

Figure 8.6

The ALIGN value here has been set to TOP, but when the browser breaks the text, it breaks to a point below the image and does not fill in the extra white space.



Figure 8.7
Using the LEFT and RIGHT values of ALIGN to create floating images. Notice how the text continues to wrap around the image, unlike the example in Figure 8.6.

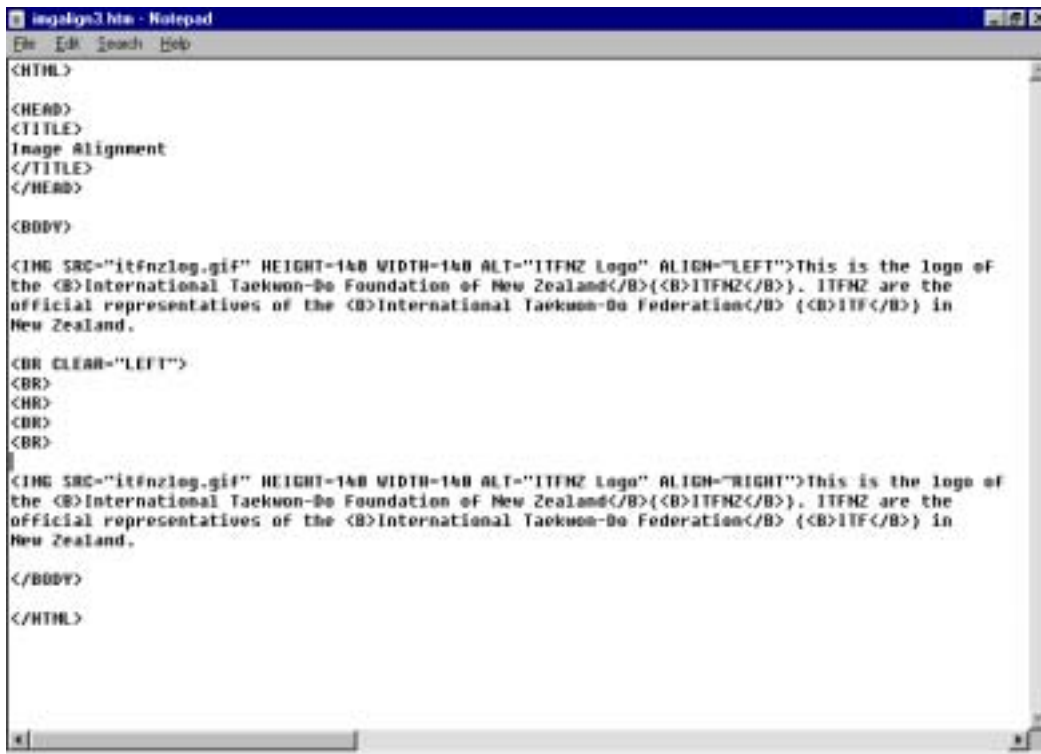


Figure 8.8

The HTML markup that creates the above Web page.

To make the text move down the image (rather than having it aligned to the top of the image), simply insert
 tags until the desired point is reached.

Because of the way text is wrapped around a floating image, it was necessary to create a method of breaking to the first left or right margin that is clear of a floating image. The CLEAR attribute was added to the
 tag to fulfil this requirement. Specifying <BR CLEAR="LEFT"> breaks to the first instance of a left margin that is clear of floating images. Using <BR CLEAR="RIGHT"> does the same thing, except it breaks to the first clear right margin. You can clear both margins with <BR CLEAR="ALL">. See Figures 8.7 and 8.8 for an example of using the CLEAR attribute of
.



Figure 8.7

Breaking clear of a floating image using the CLEAR attribute of the
 tag. This example breaks to the first instance of a clear left margin.

ISMAP & USEMAP

The ISMAP and USEMAP attributes of the tag are used to define aspects of an *Image Map*. Image maps allow you to create multiple clickable links to other locations within an image, called *hot regions*. Where the link takes you depends on what part of the image you click on.

ISMAP is a stand-alone attribute that instructs the browser that the image is to be used as part of a server-side image. Server-side maps were the first type of image map available to Web authors. A server-side image map determines which document should be downloaded, based on where the viewer clicks on the image.

USEMAP is set equal to the name of a client-side image map. With client-side image maps, map information is named and sent directly to the browser. Setting USEMAP equal to a map name tells the browser to use the map information associated with that name.

The challenge in creating an image map is determining the shape of the clickable area for a particular link, and the co-ordinates that define the shape. This can be done manually, but it can be a little painstaking calculating and typing in all of the co-ordinates needed to define hot regions. Fortunately there are applications available that can do all of this for you. These programs allow you to load your image and then, using a variety of selected shapes, you can trace the hot regions on your screen using your mouse and enter where each hot region links to. The program then writes the appropriate map file or HTML code for you. Some good examples of these types of program are:

➤ Advanced ImageMap Designer by Altersoft

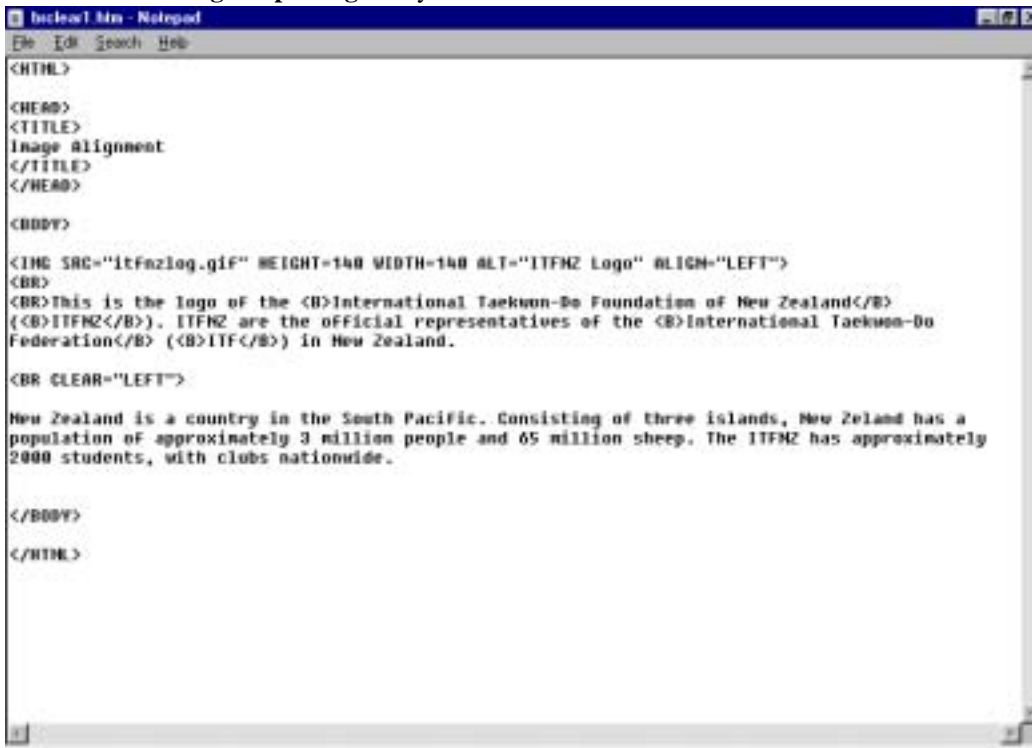


Figure 8.8
The HTML document
behind the above
example.

- **Mapedit** by Boutell.Com Inc
- **Map This!** By Todd C. Wilson

These programs, plus many more similar applications, are readily available as shareware.

To give you an idea of what is involved with creating an image map using such an application, here is **Advanced Imagemap Designer** in action:

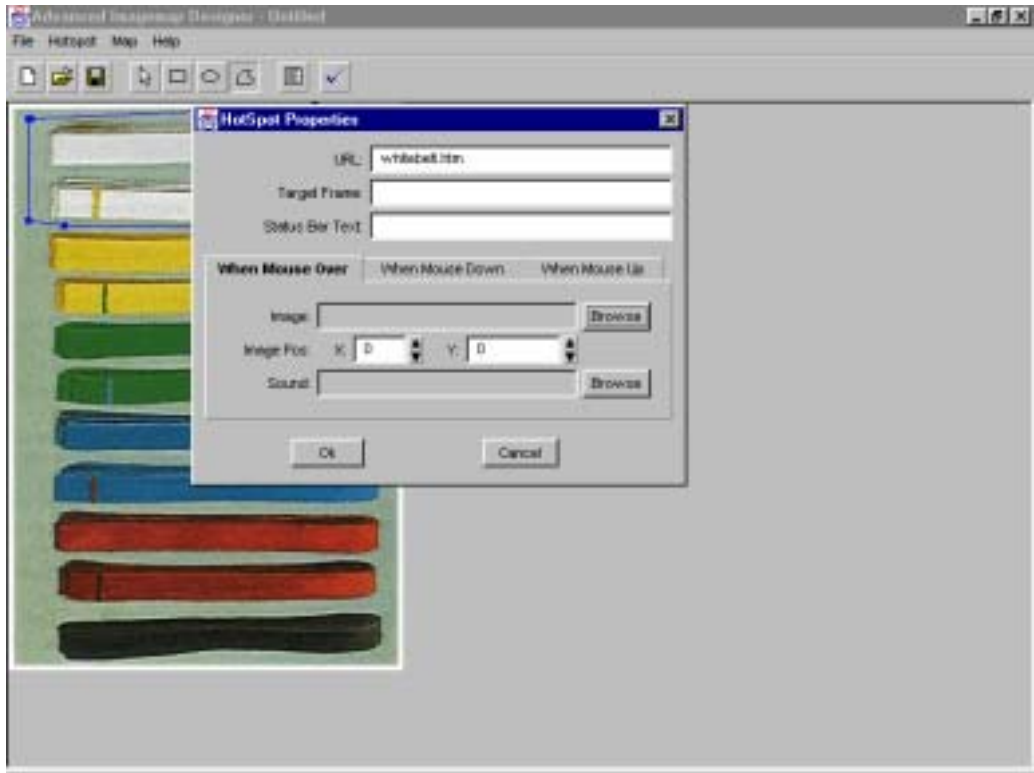
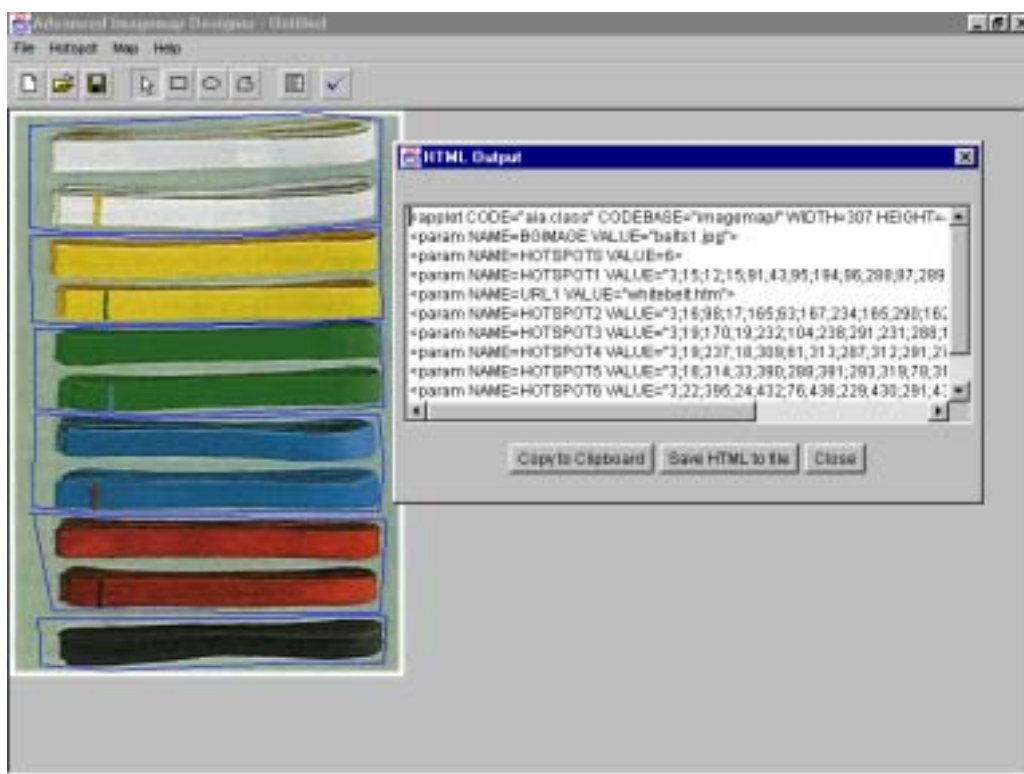


Figure 8.10
Selecting the HotSpot Properties allows you to enter the location the hot region will take you to once clicked on.

Figure 8.11
Choosing *Map* on the toolbar reveals an option to *Generate HTML*. The HTML for the mapped hot regions is automatically prepared. From here you can copy it to the Clipboard for manual insertion into an HTML document, or have it written directly into an existing document.



Chapter 9: Linking HTML Documents

The real power of HTML is its ability to link pieces of text and images to other Web pages and Internet resources. The browser usually highlights linked areas, called hyperlinks, by underlining or changing the colour of the text or image border.

You can create various types of links in your HTML document, including links to the following:

- Another Web page
- Another section of the same Web page
- A specific location within another document
- Another computer
- A downloadable file
- A mail program
- A newsgroup

We will only look at linking to Web pages and mail programs as these are the most common types of links, and you probably won't need to use the other types of links in your club's Web site.

Uniform Resource Locators (URLs)

In order to be able to create links, you must know the location, or *URL*, of whatever it is that you want to link to.

A URL is the unique Internet address for any type of Internet resource. The most common forms of URLs are the addresses of Web sites. For instance, the URL of the ITFNZ Web site is <http://www.itfnz.org.nz>.

Every URL is made up of the following four parts:

- Protocol
- Server name
- Port
- Directory path and file name

These elements are strung together in the form:

`protocol://server_name:port/directory_path_and_file_name`

Protocol

The protocol portion of a URL tells the browser which set of rules to use when retrieving the resource. The most common protocols are listed in Table 9.1.

Table 9.1 – Common Internet Protocols

Protocol	Name
http	Hypertext Transport Protocol
https	Secure Hypertext Transport Protocol
ftp	File Transfer Protocol
gopher	File Search
wais	Wide Area Information Service
telnet	Telnet session
news	UseNet newsgroup protocol
mailto	Email

The news and mailto protocols are slightly different from the rest. A typical news URL looks like

```
news:ma.tkd.itf.patterns
```

The news protocol is simply followed by a colon (:) and the name of the newsgroup.

A typical mailto URL looks like

```
mailto:fred@ihug.co.nz
```

A colon follows the mailto (:) and then an email address.

Server Name

Once the browser knows which protocol to use, it needs to know on what machine the resource resides. The server name can be the dotted English language name of the machine (e.g. www.ihug.co.nz or yahoo.com) or the machines Internet Protocol (IP) address (e.g. 202.156.18.153). Technically, machines use IP addresses to find each other on the Internet, but it usually doesn't matter whether you use the English language name or the IP address, as most Internet Service Providers have a Domain Name Service (DNS) that translates English language names into IP addresses.

Port

The port refers to the port number on the server where the client should connect. The port element is optional and, if omitted, the default port is used.

Directory Path and File Name

The server name directs the browser to a certain machine and, once there, the browser needs to know in which directory it can find the desired resource and the name of the file that contains the resource. Directory path and file name information is specified in much the same way as it is in DOS, but you must use forward slashes (/) instead of back slashes (\). Case is also sensitive, meaning that you must use correct lower and upper case characters. This is due to the Internet being based on the UNIX operating system. An example of the directory path and filename might be:

```
photos/demos/may98.htm
```

This directs the browser to the file named may98.htm that is in the demos subdirectory of the photos directory.

Some URLs may specify a directory path, but not a file name. In cases like these, the server automatically delivers a default file (usually named index.htm, index.html, home.htm, home.html, or default.htm). If no such file exists, the browser shows a list of the sub-directories and files found in the current directory that can be opened by clicking on them.

Absolute URLs

An absolute URL is made up of a protocol, server name, port (if required), and a directory path and file name. In other words it is spelled out in full, containing all necessary information about the location of the resource. An example of an absolute URL is

```
http://itfnz.org.nz/tech/tia.htm
```

You will need to use absolute URLs when creating links to locations outside your own Web site.

Relative URLs

Relative URLs are useful when linking to locations within your own Web site. Because all files in your Web site are located on the same server, it becomes tedious and unnecessary to have to type the protocol and server name each time a URL is specified.

A relative URL can include directory and file name information, specifying a location in your Web site relative to your current location. For example, consider the following directory structure in Figure 9.1:

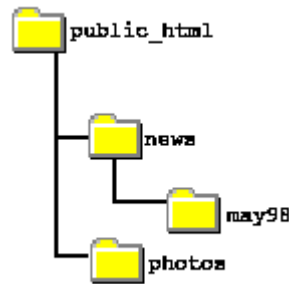


Figure 9.1
A sample directory structure on a Web server.

Say, for instance, that your homepage, `index.htm`, is located in the `public_html` directory. You are currently working on the News page for your Web site, and want to create a link from the main homepage to the news page called `news.htm` that is located in the `news` directory. The relative link from `index.htm` to `news.htm` would be declared as:

```
news/news.htm
```

Likewise, if you wanted to create a link from the News page to the May '98 newsletter (`may98.htm`), located in the `may98` subdirectory, the relative link would be:

```
may98/may98.htm
```

If you wanted to create a link from the News page back to the main homepage, you would use the relative link:

```
../index.htm
```

The two dots followed by a forward slash (`../`) instructs the browser to go up one directory level. If you wanted to link from the `news.htm` file to the picture file `team.jpg` in the `photos` directory, the relative URL would be expressed as:

```
../photos/team.jpg
```

The browser is instructed to move up one directory from the `news` directory, which is the `public_html` directory. The browser then goes to the `photos` subdirectory and loads the `team.jpg` file.

Wherever possible, you should use relative links when linking to documents within your Web site because

- you have less to type
- it is easy to move your Web site to another location (domain) as your relative links will still be valid (so long as the directory structure remains the same).

The <A>ncor Tag

Linking from one location to another is facilitated through the <A> and container tags. These tags enclose the text or image the viewer clicks on to follow the link. Linked text is known as *hypertext*.

Two attributes that the <A> tag can take are

- HREF (short for Hypertext Reference)
- NAME

The most common attribute is HREF, which is set equal to the URL being linked to.

NAME is useful for setting up links within a document. It establishes a named anchor within the document that can be targeted by an HREF.

Linking to Other HTML Documents

To establish a hypertext link to the “Clubs” page of the ITFNZ Web site you could use the following HTML:

```
Go to the <A HREF="http://itfnz.org.nz/i-clubs.htm" >ITFNZ Club  
Directory</A>.
```

Note the quotation marks around the URL – don’t forget them, or the link will not work.

Figure 9.2 shows how the above looks in a browser.

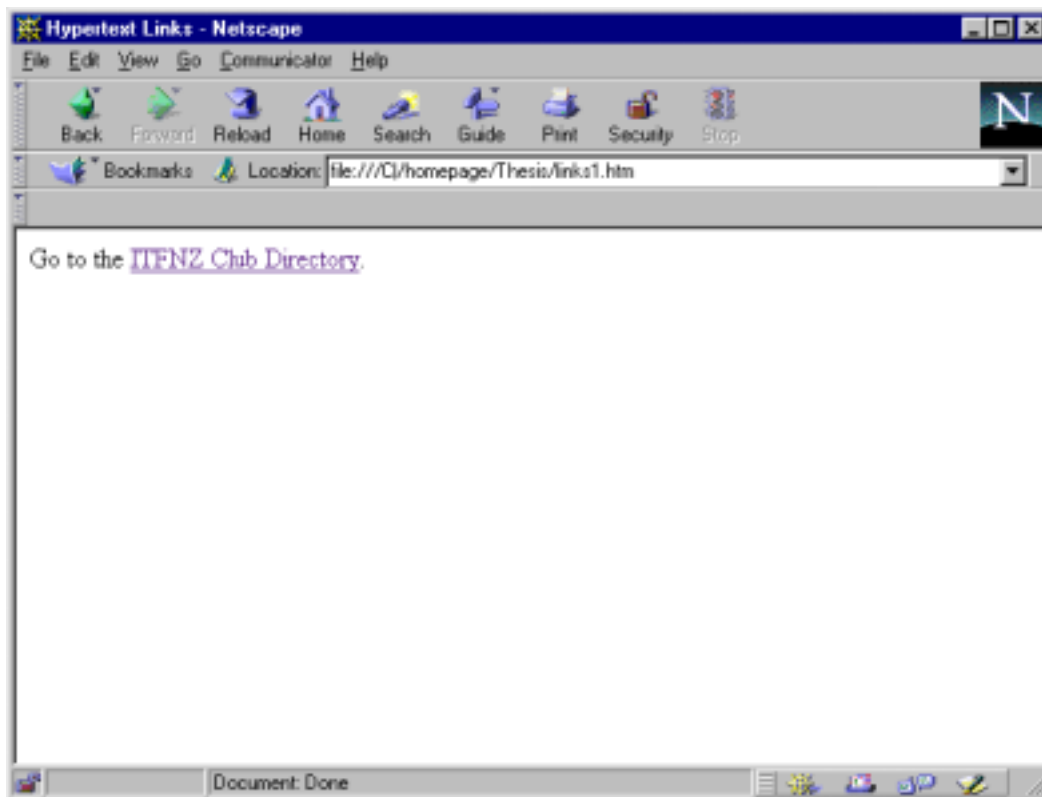


Figure 9.2

A basic hypertext link. The actual text that is linked is usually displayed underlined and in a different colour to the normal text. When the mouse pointer passes over the link it becomes an upward pointing hand.

You can make any text on your pages a hypertext link. Text within the document body marked up with either physical or logical styles, list items, headings, preformatted text, blockquotes, and addresses can all be contained between the <A> and tags to create hypertext.

Hypertext anchors should be limited to a few key words that relate the current document to the linked document. Making anchors that are large blocks of text can be visually distracting to viewers.

Images can also be used as hyperlink anchors. To establish the link, simply enclose the tag that places the image on the page with <A> and tags. For example, say we have an image named *itfnzlog.gif* that you want to link to the ITFNZ Web site. The following HTML would be used:

```
<A HREF="http://www.itfnz.org.nz"><IMG SRC="itfnzlog.gif" WIDTH=140  
HEIGHT=140 ALT="ITFNZ"></A>
```

The linked image would appear as in Figure 9.3. Notice the border around the image, even though there is no BORDER attribute specified. Hyperlinked images automatically get a border coloured with the same colours you set up for hypertext links using the LINK, VLINK, and ALINK attributes of the <BODY> tag. To remove the border created by the hyperlink, set the BORDER attribute of the tag equal to 0.

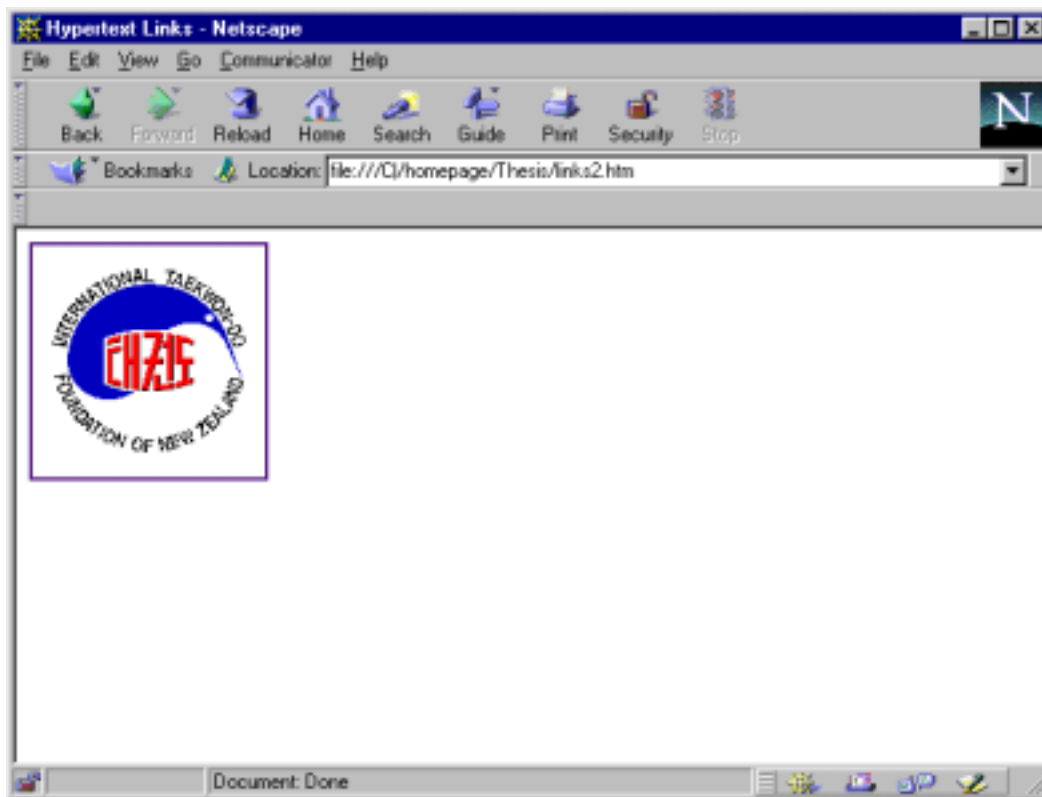


Figure 9.3
A linked image will be given a coloured border. The border reveals the boundaries of the image. To remove the border, set the BORDER attribute of the tag to 0.

Linking Within an HTML Document

Usually, when a user clicks a hyperlink, the linked document loads into their browser, starting at the top of the page. Another method of using hyperlinks allows you to create links to specific locations within a document.

Using the NAME attribute of the <A> tag you can set up named anchors that can be linked to using the HREF attribute. This is particularly handy in making your longer Web pages easier to navigate. For example, you can set up named anchors at the start of each major section of the document. You can then create hypertext links to each major section at the top of the page, in perhaps a menu-type format.

Consider the document in Figure 9.4. As it is rather long, users are forced to scroll through the document to the pattern they require. Named anchors could be set up so that the titles of each pattern could be linked to,

Figure 9.4
Long documents force viewers to scroll through them to find the required information.

as in Figure 9.5.

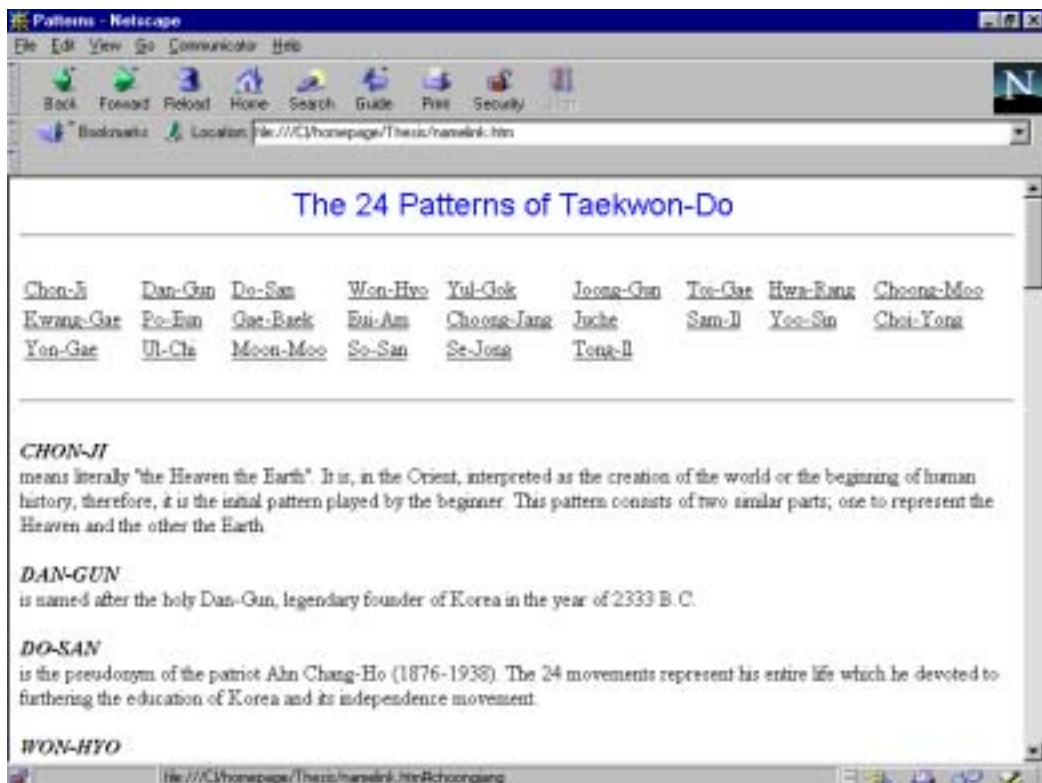
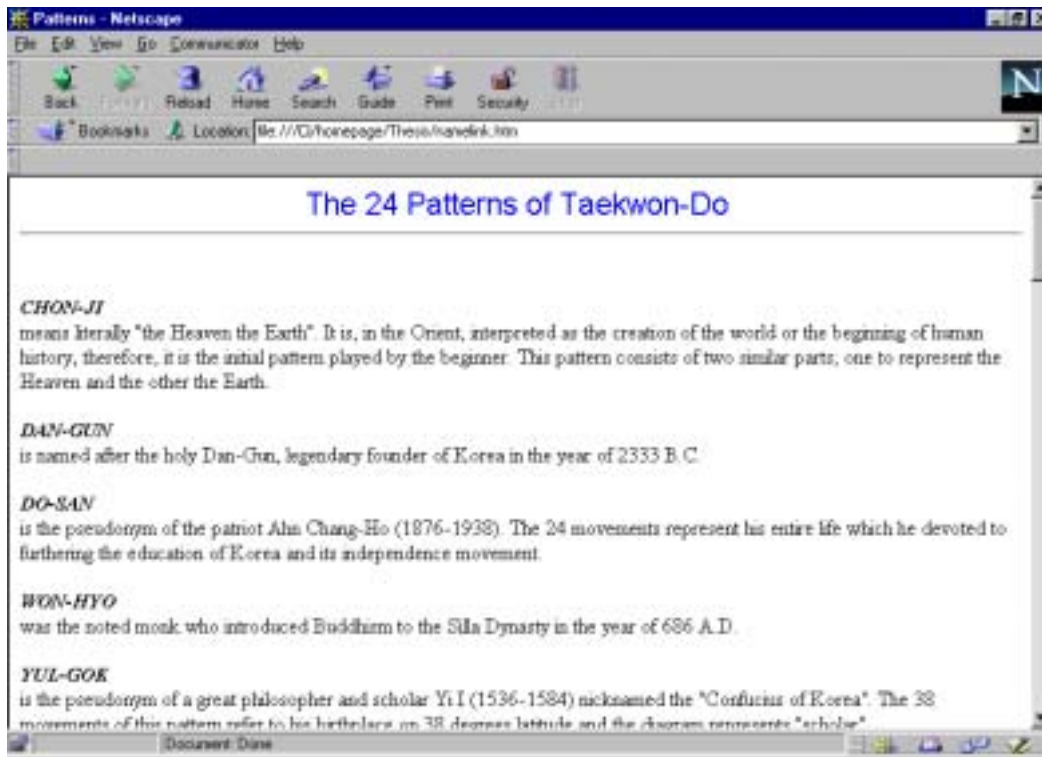


Figure 9.5

← This “table of contents” lists the headings that feature further down the page. Clicking on these links takes the reader directly to the selected heading, saving the reader from having to manually scroll to the location.

Figure 9.6 shows the HTML of the Web page in Figure 9.5. Using the Chon-Ji heading as an example, the named anchor is set up with this HTML:

```
<A NAME="chonji"><I><B>CHON-JI</B></I></A>
```

The <A> tag using the NAME attribute has no apparent effect on the enclosed bolded and italicised text. It simply sets up an anchor that is linked to with the following HTML that is found in the table of contents at the top of the page:

```
<A HREF="#chonji">Chon-Ji</A>
```

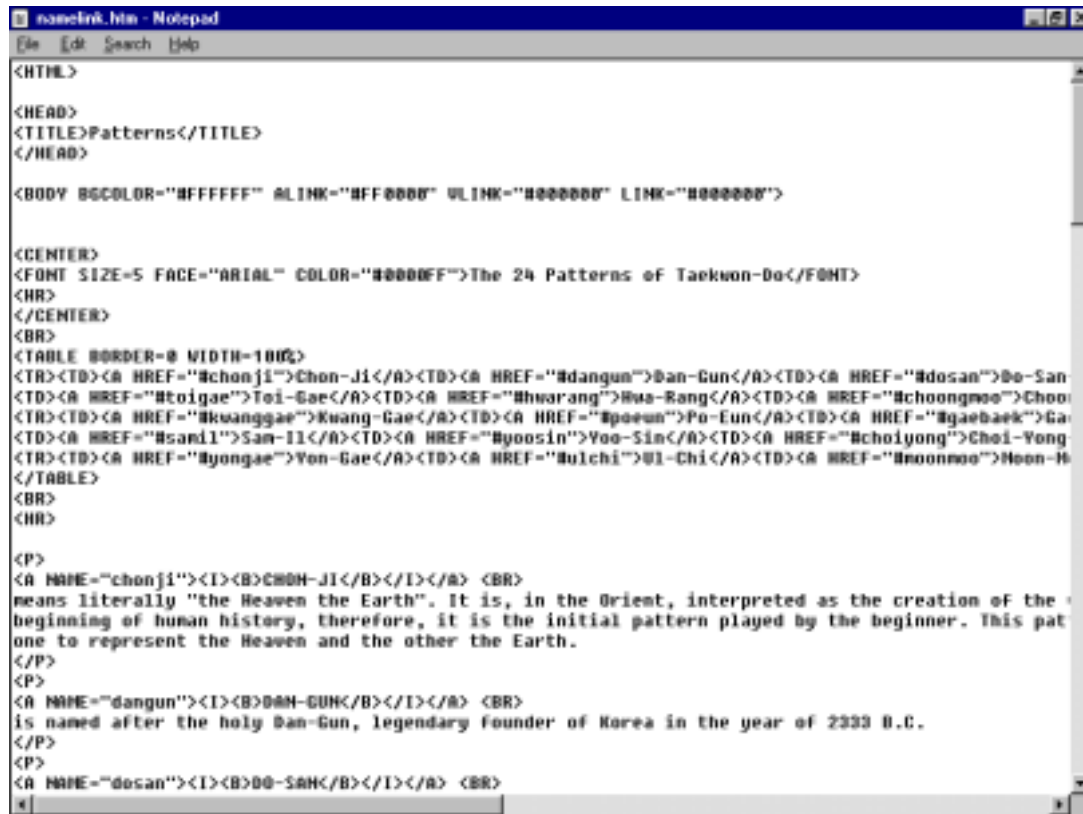


Figure 9.6
The HTML used to create the example in Figure 9.5. Notice the two types of anchor – HREF and NAME. Clicking on an HREF link moves the corresponding NAME anchor to the top of the page.

Linking to named anchors is virtually the same as linking to other documents, but the hash (#) symbol is placed in front of the anchor name. Again, the quotation marks around the URL of the link are compulsory.

You can also link to named anchors in another document. This can be useful when you want one Web page to act as an index or contents page for another. The named anchors are set upon the destination document as normal, but the link requires the URL of the document where the named anchor is located to be placed before the anchor name. For instance, if we have a long document named `info.htm` that has a named anchor called `section1` within it, the HREF link would be something like:

```
<A HREF="info.htm#section1">Section One</A>
```

Clicking on the above link would load the `info.htm` file and put the anchor with the name `section1` at the top of the page, perhaps halfway through the document.

Linking to an Email Program

Using a `mailto` URL creates a link on your pages that, when clicked on, launches the readers default mail program with the email address already inserted. Such a link might be:

```
<A HREF="mailto:tkdclub@ihug.co.nz">Send Your Comments</A>
```

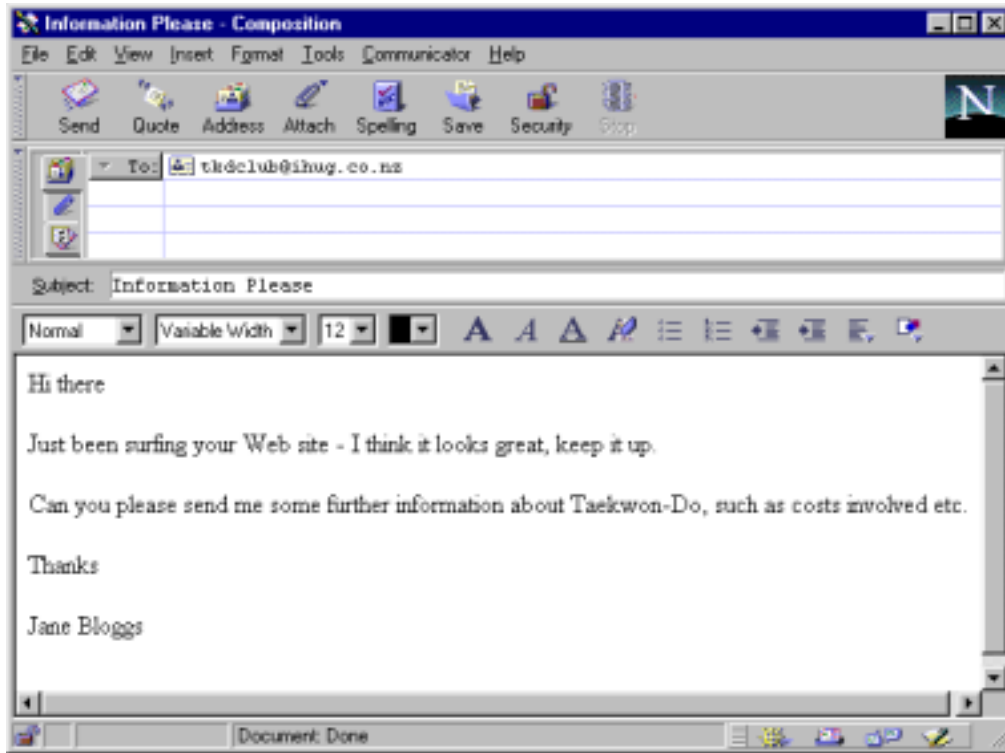


Figure 9.7
Clicking on a link with a `mailto` URL brings up the viewers mail program, with the specified email address already inserted.

Providing a link with a `mailto` attribute is a great way to collect feedback on your site. Even if you don't want feedback, it is still a good idea to give visitors away to contact you should they have any queries or need further information.

Chapter 10: Tables

Tables were intended to be used to display columnar data in a neat format. They have also developed into an effective page design tool, assisting with the alignment of the elements in your Web pages.

Tables are made up of *cells*, which contain data elements or a heading for a column of data. Related cells are grouped together to create *rows*, and the rows combine to form the table.

Figure 10.1 offers a graphical description of the concepts of tables.

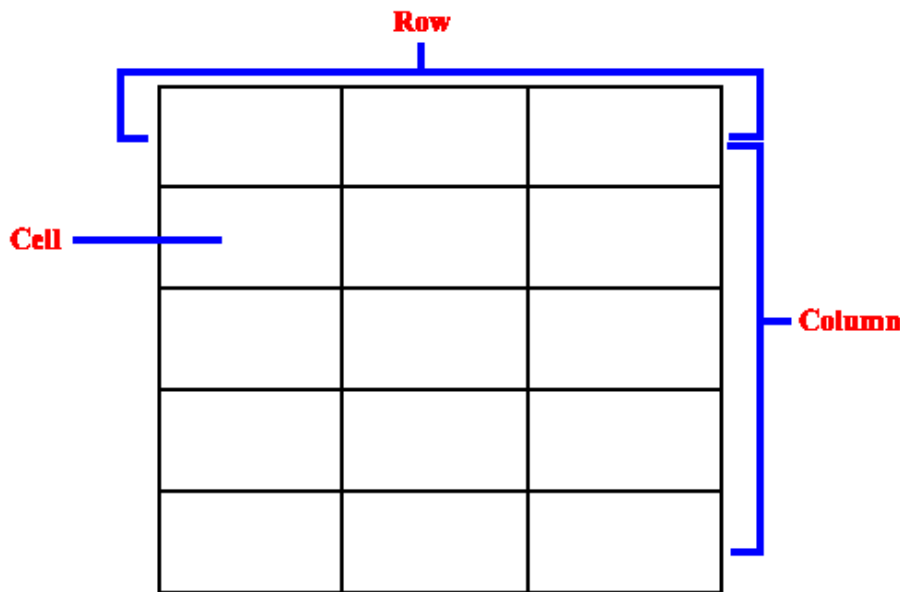


Figure 10.1
The make up of
tables.

The <TABLE> Tag

Tables are defined between the `<TABLE>` and `</TABLE>` tags. Any table related tags that are not between these tags are ignored.

A good habit to adopt is to put the `</TABLE>` tag in your HTML document when you put the `<TABLE>` tag in. If you forget to use the `</TABLE>` tag your table will not be displayed. This is because browsers read through all of the code used to produce a table and determine the space needed to display the table. Once the space required to display the table has been determined, the browser goes back and fills the cells. Without the `</TABLE>` tag, the browser won't know where the table ends, and won't display it.

Defining Table Rows

Tables are formed from rows of cells. The `<TR>` and `</TR>` tags define the table rows that contain the individual data cells. You can have as many `<TR>` and `</TR>` tag pairs as you require in a table, each pair identifying one row.

Defining Table Cells

There are two types of table cell: *header cells* that contain column headers and *data cells* that contain the individual table entries.

Header Cells

Table header cells are defined with the `<TH>` and `</TH>` tags. The contents of a table header cell are automatically centred and rendered in bold.

In a standard table, header cells usually make up the first row of the table.

Data Cells

Data cells usually make up the major part of the table. The `<TD>` and `</TD>` tags define data cells. By default, text within data cells is left justified and has no special formatting applied. If special formatting is required, you can enter the appropriate tags between the `<TD>` and `</TD>` tags.

Figure 10.2 shows an example of the HTML markup of a table:



```
<HTML>
<HEAD>
<TITLE>
Tables
</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
<CENTER>
<FONT FACE="ARIAL, TIMES ROMAN" COLOR="#000099"><H1>KIMNG GAE TAEKWON-DO CLUB</H1></FONT>
<HR>
</CENTER>
<H2>Events Calendar</H2>
<BR>
<TABLE>
<TR><TH>Date/Time</TH><TH>Event</TH><TH>Venue</TH></TR>
<TR><TD>15 May, 9:30am</TD><TD>Kup Grading</TD><TD>South School Hall, Tean.</TD></TR>
<TR><TD>22 June, 10am</TD><TD>Fun Run</TD><TD>Waterfront Hill Road</TD></TR>
<TR><TD>19 July, 9am</TD><TD>National Tournament</TD><TD>Big Stadium, Wellington</TD></TR>
<TR><TD>25 July, 6:30 - 8pm</TD><TD>Combined Training</TD><TD>XVZ Club, School Hall</TD></TR>
</TABLE>
</BODY>
</HTML>
```

Figure 10.2
Creating a table with
HTML.

I prefer to layout my HTML table markup in the format the table will be rendered in the browser, i.e. placing the cell contents of each row side-by-side.

Because of the way browsers interpret the table tags, ignoring carriage returns, it is also possible to layout the HTML markup down the document, as in figure 10.3. This can make it easier to see each individual cell entry but can also make it harder to envisage what goes where when creating the table.

```

<H2>Events Calendar</H2>
<BR>

<TABLE>
<TR>
<TH>Date/Time</TH>
<TH>Event</TH>
<TH>Venue</TH>
</TR>
<TR>
<TD>15 May, 9:30am</TD>
<TD>Kup Grading</TD>
<TD>South School Hall, Town.</TD>
</TR>
<TR>
<TD>22 June, 10am</TD>
<TD>Fun Run</TD>
<TD>Waterfront Hill Road</TD>
</TR>
<TR>
<TD>19 July, 9am</TD>
<TD>National Tournament</TD>
<TD>Big Stadium, Wellington</TD>
</TR>
<TR>
<TD>25 July, 6:30 - 8pm</TD>
<TD>Combined Training</TD>
<TD>XYZ Club, School Hall</TD>
</TR>
</TABLE>

```

Figure 10.3
Laying out the table HTML entities down the page can make it easier to read each individual cell item, but can make it harder to picture what the table will look like when you are creating it.

Regardless of the method you choose to format the HTML of your table, the results are the same. Figure 10.4 shows how either of the examples from figures 10.2 or 10.3 would appear in a browser.



Figure 10.4
Either of the HTML examples from figures 8.2 or 8.3 would produce a table like this. Note how the header row is automatically centered and rendered in bold, while the normal cell data is left justified.

This default formatting of the table tags is not all-together tidy though. The data is a little difficult to read, as it is hard to determine at a glance where one cell ends and another begins. Fortunately, the tags that make up a table can take additional attributes that allow you to have control over virtually every aspect of the table's format.

Alignment

It is possible to control the alignment of both the content in the individual cells, and also the entire table itself. There are two types of alignment that can specify:

Horizontal alignment allows you to control the alignment of a cell's contents across the width of the cell, or the alignment of the entire table across the width of the page. The attribute that specifies the horizontal alignment is `ALIGN`, which can be set equal to `LEFT`, `CENTER`, or `RIGHT`.

Vertical alignment refers to the alignment of a cell's contents between the top and bottom of the cell. Vertical alignment is set with the `VALIGN` attribute, which can be set equal to `TOP`, `MIDDLE`, or `BOTTOM`. Vertical alignment cannot be set for an entire table, because a page's length isn't fixed like the width is.

Aligning the Table

You can use the `ALIGN` attribute in the `<TABLE>` tag to specify how the table should be aligned in relation to the width of the browser window. Setting the alignment to the left or right, `<TABLE ALIGN="LEFT">` or `<TABLE ALIGN="RIGHT">`, floats the table in the left or the right margin respectively. Floating tables mean that you can wrap text around them.

Using the `CENTER` value of `ALIGN` centres the table in the browser window, but not all browsers support this function yet. To ensure that the table is centred across the page, you may want to use the `<CENTER>` and `</CENTER>` container tags around the table, for example:

```
<CENTER>
<TABLE>
...table contents...
</TABLE>
</CENTER>
```

Aligning a Row

If you want to set the horizontal or vertical alignment to be the same for every cell in a given row, you can use the `ALIGN` and `VALIGN` attributes in the row's `<TR>` tag. For example:

```
<TABLE>
<TR ALIGN="CENTER" VALIGN="CENTER"><TD>cell data</TD>"><TD>cell
data</TD></TR>
</TABLE>
```

Any alignment specified in a `<TR>` tag will override any default alignments.

Alignment within a Cell

You can control the horizontal and vertical alignment of entities within an individual cell in much the same way that you control the alignment of an entire row.

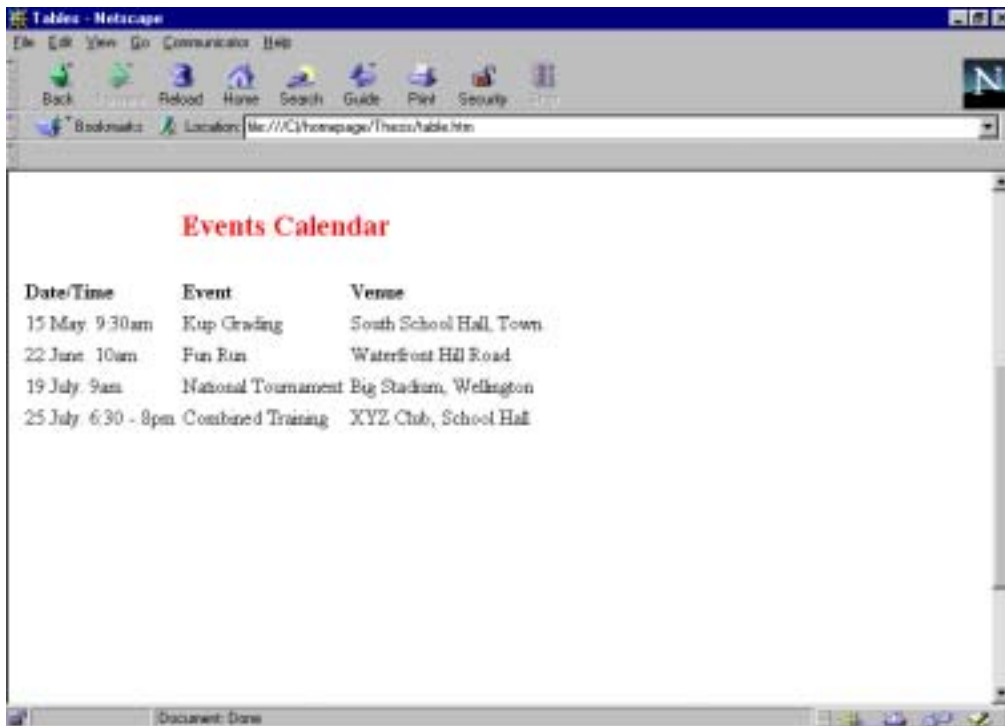
Using the ALIGN and VALIGN attributes in <TH> or <TD> tags will override any default alignments and any alignments specified in the <TR> tag.

Adding a Caption

If you want to include a title for your table, you can easily do so by incorporating the <CAPTION> container tag. Captions appear centred over the table and, if the caption's length will exceed the width of the table, are broken to match the table's width. You can also use the physical style tags to format the caption text. For example:

```
<TABLE>
<CAPTION><H2><FONT COLOR="FF0000">Events
Calendar</FONT></H2></CAPTION>
<TR ALIGN="LEFT"><TH>Date/Time</TH><TH>Event</TH><TH>Venue</TH></TR>
<TR><TD>15 May. 9:30am</TD><TD>Kup Grading</TD><TD>South School Hall,
Town.</TD></TR>
<TR><TD>22 June. 10am</TD><TD>Fun Run</TD><TD>Waterfront Hill
Road</TD></TR>
<TR><TD>19 July. 9am</TD><TD>National Tournament</TD><TD>Big Stadium,
Wellington</TD></TR>
<TR><TD>25 July. 6:30 - 8pm</TD><TD>Combined Training</TD><TD>XYZ Club,
School Hall</TD></TR>
</TABLE>
```

would appear as in Figure 10.5.



The screenshot shows a Netscape browser window with the title 'Tables - Netscape'. The address bar shows 'file:///C:/homepage/Themes/table.htm'. The main content area displays a table with the caption 'Events Calendar' in red, bold text. The table has three columns: 'Date/Time', 'Event', and 'Venue'. The data rows are as follows:

Date/Time	Event	Venue
15 May. 9:30am	Kup Grading	South School Hall, Town
22 June. 10am	Fun Run	Waterfront Hill Road
19 July. 9am	National Tournament	Big Stadium, Wellington
25 July. 6:30 - 8pm	Combined Training	XYZ Club, School Hall

Figure 10.5
Using a caption and aligning the table header row immediately improves the look of our table.

If you prefer to have the caption below the table, you can include the `ALIGN="BOTTOM"` attribute in the `<CAPTION>` tag.

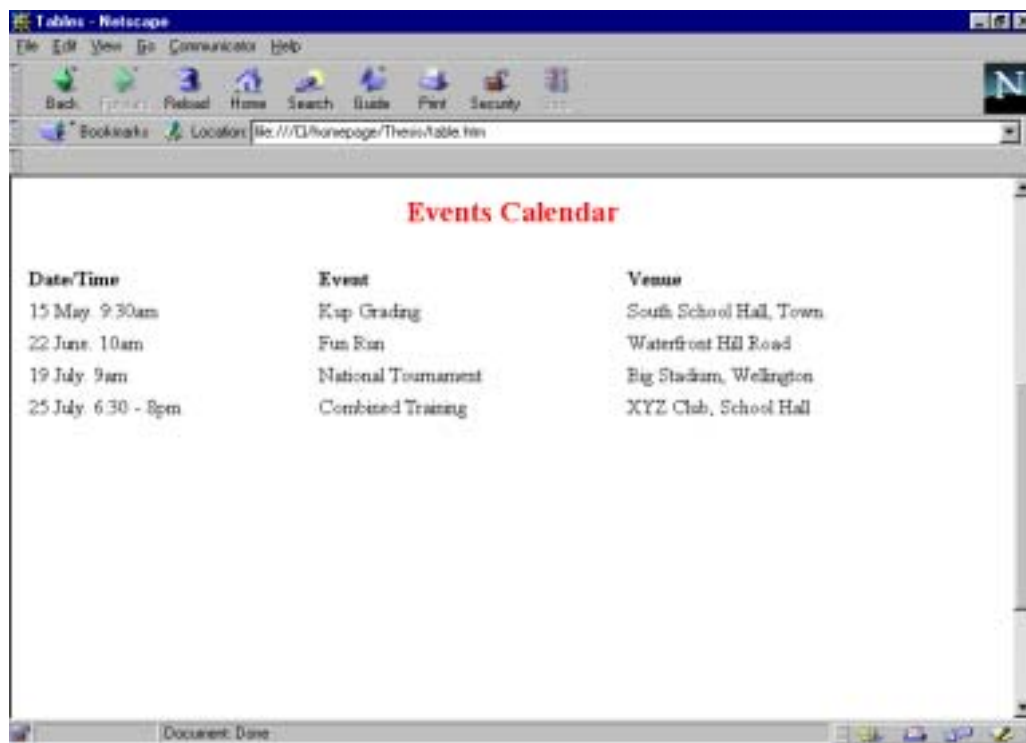
Specifying the Width

The `WIDTH` attribute of the `<TABLE>` tag allows you to specify how wide the table will appear in the browser window. `WIDTH` can be set to a specific number of pixels or to a percentage of the available screen width. Wherever possible set `WIDTH` to a percentage, as specifying a number of pixels can produce mixed results due to users having their screens configured differently.

`WIDTH` is often used to force a table to take the entire width of the browser window. Using the previous example but changing the `<TABLE>` tag to:

```
<TABLE WIDTH=100%>
```

renders the table as shown in Figure 10.6.

A screenshot of a Netscape browser window. The title bar says 'Tables - Netscape'. The address bar shows 'file:///C:/homepage/Thesis/Tables.htm'. The main content area displays a table with the title 'Events Calendar' in red. The table has three columns: 'Date/Time', 'Event', and 'Venue'. It contains four rows of event data.

Date/Time	Event	Venue
15 May 9:30am	Kup Grading	South School Hall, Town
22 June 10am	Fun Run	Waterfront Hill Road
19 July 9am	National Tournament	Big Stadium, Wellington
25 July 6:30 - 8pm	Combined Training	XYZ Club, School Hall

Figure 10.6

You can alter the size of a table using the `WIDTH` attribute of the `<TABLE>` tag. When set to 100%, this example becomes much easier to read.

Adding a Border

You can place border around your table by including the `BORDER` attribute of the `<TABLE>` tag. `BORDER` is set to how wide you want the border to be in number of pixels. Building on the previous example again, adding the `BORDER` attribute like so:

```
<TABLE WIDTH=100% BORDER=2>
```

produces the effect shown in Figure 10.7.

You can also set `BORDER` equal to 0. This will prevent a border from being used.

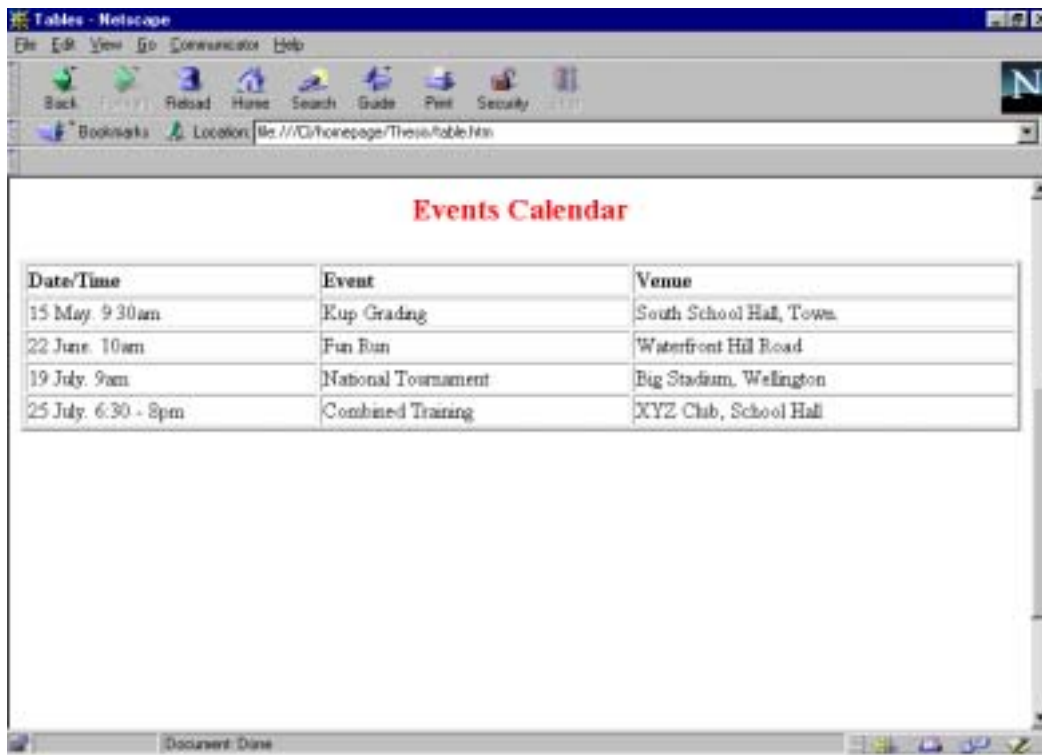


Figure 10.7
Using a border can aid readability by clearly separating cells from one another.

Specifying Spacing Within a Cell

The distance between an element in a cell and the boundaries of the cell is called *cell padding*. CELLPADDING is an attribute of the `</TABLE>` tag. Usually, Web page authors increase the cell padding from its default value of 1 to put a little more white space between the contents and edges of a cell.

Figure 10.8 shows the effect of adding CELLPADDING to the `<TABLE>` tag like so:

```
<TABLE WIDTH=100% BORDER=2 CELLPADDING=5>
```

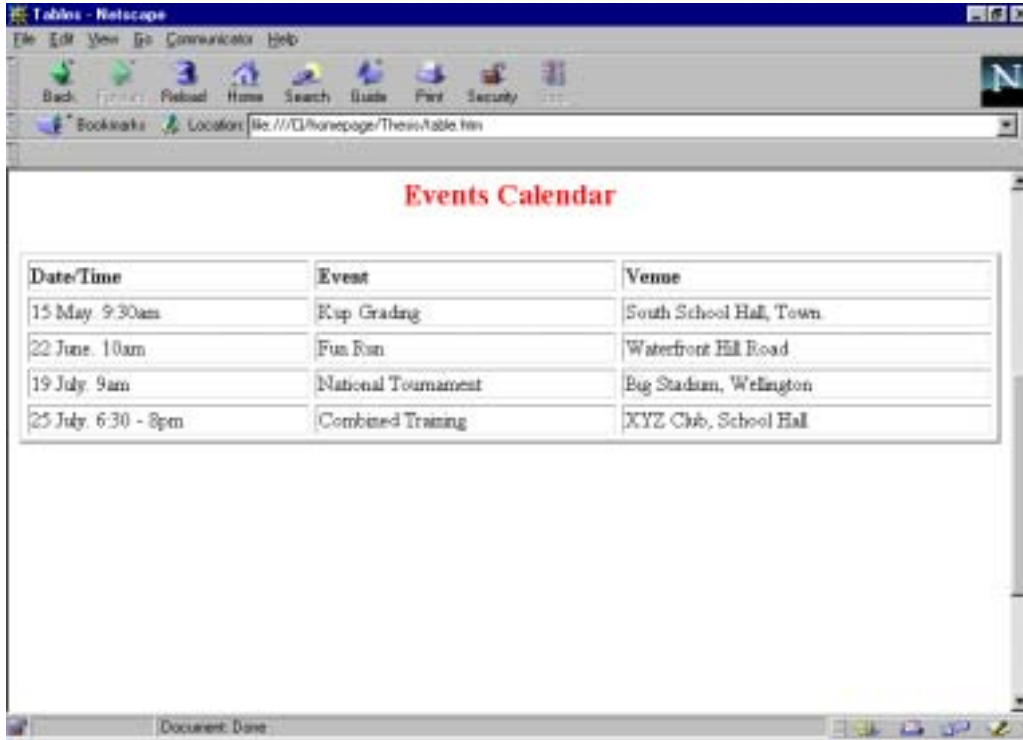


Figure 10.8
Using the CELLPADDING attribute of the `<TABLE>` tag you can create more room between the contents of a cell and the edge of the cell itself.

Specifying Space Between Cells

The `CELLSPACING` attribute of the `<TABLE>` tag allows you to increase the space between cells. Using the below `<TABLE>` tag produces the effect in Figure 10.9.

```
<TABLE WIDTH=100% BORDER=2 CELLSPACING=5>
```



Date/Time	Event	Venue
15 May. 9:30am	Kup Grading	South School Hall, Town
22 June. 10am	Fun Fun	Waterfront Hill Road
19 July. 9am	National Tournament	Big Stadium, Wellington
25 July. 6:30 - 8pm	Combined Training	XYZ Club, School Hall

Figure 10.9
The `CELLSPACING` attribute of the `<TABLE>` tag allows you to increase the space between cells. Notice how the border also increases in thickness when using this attribute.

Using the COLSPAN Attribute

`COLSPAN` is an attribute of the `<TH>` or `<TD>` tags. Using `COLSPAN` on one of the cells instructs the browser to make the cell take up more than one column. `COLSPAN` is set to the number of columns you want the cell to occupy.

`COLSPAN` can be useful when using tables as a page layout tool if you have one row of the table that is forcing the table to be a certain number of columns wide while the content of the other rows don't require so many columns.

Chapter 11: Other Web Page Facilities

Guestbooks and hit counters can be useful additions to any Web site. While creating either of these facilities yourself requires you to be familiar with a programming language, and perhaps have a discussion with your Web server provider in order to have them host the programs, there is an easy way to include both to your pages through online providers.

Guestbooks

Guestbooks allow visitors to record their comments about your Web site, and view comments left by other visitors. Signing is almost always voluntary, but there are usually links to the guestbook from the home page that encourage visitors to sign.

There are a number of suppliers of free guestbook facilities. You simply register with the supplier, who then provides you with the appropriate HTML to add to your Web site. When your visitors click on the guestbook link, they are taken to your guestbook located on the server of your guestbook provider. The visitor leaves their comments, and can click on a link back to your site or the back button of their browser to return to your actual site again. Most guestbook providers ask for a little credit, in the form of a small logo displayed somewhere on your site.

Probably the most popular guestbook provider is **GuestWorld**, which can be found at <http://www.lpage.com>.

For a list of other guestbook providers, have a look at **Web Resource's** list at <http://www.userworld.com/users/bwportne/books.html>

Page Counters

Page counters keep a track of the number of 'hits' (visitors) a particular page has had. While counters can be annoying if they made to be the focus of a page, they can be useful if they are built into a page in an unobtrusive way. Counters are only really needed on the home page of a Web site.

Some of the better online counters can also supply you with useful statistics about your visitors. Information such as how many hits occurred during a week or a day, which hours of the day are busiest, which URL brought the viewer to the page with the counter on it and other like statistics can help you keep a close eye on traffic.

Like the guestbook providers, counters can be obtained by registering your site with a provider who in turn supplies you with the appropriate HTML to link you to their service. Again, many of the providers request that you display a link or small graphic to their site.

There are a great number of free counter providers. Even your own Web server provider may have a standard counter available to its users. A popular counter service is SiteFlow.com, <http://www.siteflow.com>.

For a list of other counters have a look at **Web Resource's** list at <http://www.userworld.com/users/bwportne/counter.html>

Search Engines

In order to have your Web site visited, people must know where to go. As you probably want to extend your list of visitors beyond the number of people you remember to give the actual URL to, you will want to register your URL with a few popular search engines.

Search engines provide an indexing system of Web sites. Users enter criteria for a search, and the search engine provides a list of Web sites that contain some information relating to the criteria the user specified.

Sometimes the search engine robots may have found your site already. If you find you are not yet registered with a search engine you use, you can register with them by going to the Web site of the search engine itself and, usually, they have a link you can follow where you enter the details about your site (something like “Add URL”). The search engine then schedules its robot to index the content of your site, and categorises your content in its list.

A few of the popular search engines are below.

Table 11.1 – Popular Search Engines

Site Name	URL
Yahoo	http://www.yahoo.com
Webcrawler	http://www.webcrawler.com
Alta Vista	http://www.altavista.digital.com
Lycos	http://www.lycos.com
Answers	http://www.answers.com.au
Infoseek	http://www.infoseek.com

Chapter 12: Setting Your Creation Free

Once you have finished creating the documents that make up your Web site, you need to transfer them to a Web server in order for everyone else to be able to view them also.

Before transferring the files though, test your Web site on your local PC. Make sure all of the pages appear as you intend them to, in several browsers too if possible. This is possible by using the *Open* option of the *File* menu and selecting the files on your PC.

Once you are satisfied that your Web site is just how you want it, you will need a File Transfer Protocol (FTP) program, and some space on a Web server where you can store your files.

Some ISP's grant their customers a limited amount of Web server space as part of their account package. Accounts like this are really useful, as it means that your Web site is effectively free. In most cases like this, the URL of your allotted space is an extension of the ISP's own URL. For example, some of Ihug's accounts have such a facility. The URL of these user sites are Ihug's own URL followed by a tild (~) and the users login name. If your login name was zak, the URL of your personal Web site would be <http://hompages.ihug.co.nz/~zak>. Free space is also occasionally available for sports clubs or community groups. Check with your local Council or ISP to see if they have such an arrangement.

Web server space can also be leased from many ISP's for a modest fee, depending on the ISP.

Placing Your Site on a Web Server

When you have set up your Web server space you will need to transfer the necessary files from your system to the remote Web server. To do this you will probably need to use an FTP program.

Some Web server space providers supply a copy of an FTP program when you sign up with them, but you can also download FTP programs from <http://www.shareware.com>. A popular FTP program is WS_FTP, which is a windows based application that simply allows the copying, moving and deleting of files to and from a remote server.

Your Web server space provider should supply you with details of the address of the location you should store your files and a corresponding password to access that location.

Maintain Your Site

Try to keep your information as current as possible. Obviously if you are using the Web site as a medium for club newsletters you will want to update that part of the site fairly regularly. This involves the same process as actually creating your Web pages. Simply update and edit the appropriate pages and FTP the amended file, plus any additional files such as new images, to your Web server.

People like to visit a site that is regularly updated, especially with current information about the club, Taekwon-Do in general, and photos of TKD in action. If someone finds your site interesting they will probably visit again, and may even suggest your site to other people. People can get a real "feel" for your club from your Web site, and a well designed site will encourage visitors to find out more.

Requesting a Link from the ITFZN Web Site

The ITFZN Web site, <http://www.itfnz.org.nz>, is arguably one of the best Taekwon-Do Web sites in the world. Over 100 hits per day are received from all over the World.

One of the pages in the site, <http://itfnz.org.nz/clubs.htm>, lists all of the clubs in New Zealand. Links to approved club Web sites are found on this page.

To request that your site be linked to from this page, email Messieurs McPhail and Breen at itfnz@itfnz.org.nz.

To be eligible for a link, you must comply with the following requirements specified by the ITFZN:

Requirements for ITFZN Club Pages and other pages using the name and symbols of the International Taekwon-Do Foundation of New Zealand

The Constitution of The International Taekwon-Do Foundation of New Zealand requires that any published material claiming affiliation to the Foundation be duly authorised.

If your pages satisfy the requirements and are otherwise deemed appropriate you may request a link to be publicised on the ITFZN home pages.

1. "International Taekwon-Do Foundation of New Zealand, ABC Branch (or Team or whatever) " will be displayed in full at least once on each page claiming affiliation. Subsequent references to "ITFZN" on that page are ok. (Pages may be found in isolation by people using searches, it must be possible to identify a page without any assumptions about who is looking at it.)

2. Your "Home Page" will contain a link to the "Official ITFZN Home Page" - not just from a sub-page. (The central pages are a promotional resource for ALL branches)

3. Do not put material for interests or activities other than the officially affiliated body on your home page. Put a link to a sub-page. A discrete reference to a sponsor is acceptable. (A direct link from the ITFZN homepage has an implication of support by ITFZN. This will only be given where the link is to a page dedicated to ITFZN activities.)

4. Nothing inappropriate to be contained in any pages containing any reference to ITFZN. Apart from obviously objectionable material, this includes such things as unwarranted negative comment on any other martial art or persons associated therewith. (This requirement to be interpreted with common sense.)

5. Pages to be of a presentable standard. This does not mean they have to be flash - but if you have trouble with spelling, get a friend to check, or ask the webmaster to look it over. Some HTML functions can also make a mess if all aspects of their use are not considered. Not everyone uses Netscape, or has graphics always on - ensure that there are text alternatives to allow navigating your pages.

Email request for a check on your pages, or for assistance. (nbreen@monstar.co.nz)

*N Breen
ITFZN Internet Co-ordinator.*

Other ITF Web Sites

Here is a list of other ITF Web sites on the Net. Check them out, use them for inspiration, and maybe “borrow” an image from them for use on your site. If you do want to use an image from another Web site, it may be worth it to request permission.

Table 12.1 – ITF Affiliated Web Sites

Site Name	URL
International Taekwon-Do Federation (ITF)	http://www.itf-taekwondo.com
United States Taekwon-Do Federation (USTF)	http://ucsu.colorado.edu/~tkd/USTF/USTF_home.html
Jamaica Taekwon-Do Association	http://www.geocities.com/Tokyo/Towers/3543/jatkd_home.html
Lu’s Taekwon-Do (Canada)	http://www.pc-comp-sol.ca/~lu-taekwon-do
Tornado Taekwon-Do (ITF Russia)	http://www.tornado.metrocom.ru
Stenungsunds Taekwon-Do (ITF Sweden)	http://www.taekwon-do.com/eng.htm
Republic of Ireland Taekwon-do Association (ITF)	http://www.geocities.com/Colosseum/Field/8947
Oregon/Washington USTF	http://members.aol.com/ustfnw/index.html
Jue’s Taekwon-Do (California USTF)	http://www.jps.net/juestkd/index.html

The Final Word

Creating a Web site for your club can be very useful for promotion of the club, both externally and internally, and promotion of Taekwon-Do in general. It is also very rewarding and fulfilling - you should have an enormous amount of fun creating a Web site. When you start to get a good grip on the HTML covered in this document, you may want to take a trip to your local bookstore and investigate some more advanced HTML or Web authoring techniques to really liven up your pages.

Thanks for your time, if you have any queries please email me at gray_p@ihug.co.nz. I am not a Web page guru by any stretch of the imagination, but I am happy to offer my assistance. Good luck!