

**In This Practical You Will:**

1. Identify and use different HTML formatting codes.
2. Create and modify HTML documents using a simple text editor (Notepad).
3. Write a series of web pages that present information, graphics, and provide hypertext links to other documents on the Internet.
4. Get familiar with “localhost”.
5. Maybe have some fun?

**HTML**

HTML (HyperText Markup Language) is a formatting language that tells a computer how to display a web page. It is not a programming language because it has no variables and you cannot write loops, etc. The documents themselves are plain text files (ASCII) with special “tags”, or codes, that a web browser knows how to interpret and display on your screen.

This tutorial teaches you how to create web pages the old-fashioned way – by hand. There are software tools that allow you to create web pages without touching any HTML (e.g. VS.NET!). But if you are serious about becoming a professional web developer, we believe grounding in the basics will greatly accelerate and enhance what you can do.

Everything you create in this tutorial is designed to run from any desktop computer – it does not depend on access to a web server – you only need a web browser.

NOTE: This tutorial will not teach you all of HTML. There are many more tags that are not covered here. You can look on the web for other tutorials on HTML or borrow a book on HTML from the Library to learn more about it. The full specification of HTML is available at the World Wide Web Consortium: <http://www.w3.org>. However, you will not need to know all of the HTML tags!

## **Part A – Creating Your First HTML Document**

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

When a web browser displays a page, it reads from a plain text file, and looks for special codes or “tags” that are marked by < and >. (These are less than and greater than signs, or sometimes called “angle brackets”.) The general format for a HTML tag is:

**<tag\_name>string of text</tag\_name>**

The syntax is important, if you forget the slash for instance, the text won’t display in the browser like you expect!

Note: HTML is case-insensitive. However, XHTML is case-sensitive and is all lower case

If this document was written in HTML, the sub-heading above might look like this:

**<h3>What are HTML tags?</h3>**

where the “h3” means Heading size 3. (There are six levels, with 1 being the highest and 6 the lowest.)

Your browser has only a small vocabulary! An interesting aspect of HTML is that if the browser does not know what to do with a given tag, it will ignore it! For example, when a browser tries to display the following HTML snippet:

```
<wigggle><h3>What are HTML tags?</h3></wigggle>
```

The “wigggle” tag would be ignored and it would look exactly the same as the heading above.

## Opening Up Your Workspace

We will use Notepad as our text editor and MS Internet Explorer (IE) as the web browser (although you can use other another browser if you like).

- Open the browser (IE) choosing Start → Internet Explorer.
- Open Notepad (Start Menu → All Programs → Accessories → Notepad).

## Creating Your HTML Document

An HTML document contains two distinct parts, the Head and the Body. The head contains information about the document that is not displayed on the screen. The body contains everything else that is displayed as part of the web page.

The basic structure of an HTML page is:

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns = "http://www.w3.org/1999/xhtml">
<head>
<!-- header info used to contain extra information about this
document,
      not displayed on the page -->
</head>
<body>
<!-- all the HTML for display -->

</body>
</html>
```

The second line: **<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">** is not technically required, but is a code that tells the browser what version of HTML the current page is written for. For more information, see the W3C Reference Specification.

The HTML page contains nested tags, i.e. the HTML content is contained within <html>...</html> tags. Inside are the <head>...</head> and then the <body>...</body> sections.

Also note the comment tags enclosed by <!-- blah blah blah -->. The text between the tags is NOT displayed in the web page but is for information that might be of use to you or anyone else who might look at the HTML code behind the web page. (You can see the HTML code for a web page you are currently viewing with the View → Source command in IE.)

- In the empty Notepad document window, type in the text below (or something like it). You can copy and paste the text. If you do so please try to understand what we are doing here.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title>Web Applications</title>
    <!-- written for Workshop 3 by Joe Bloggs -->
  </head>
  <body>
    This is plain text that the browser displays using its default font
    settings.
  </body>
</html>
```

- Select File → Save As and browse to the C:\Inetpub\wwwroot\ directory on the local computer. (This is the “root” directory for the web server running on your PC. More about this later.)
- Save the document as a file called "prac1.html" into the C:\Inetpub\wwwroot directory. By using the file name extension of ".html", a web browser will know to read this text file as HTML and properly display the web page. (You could also use ".htm")

### Displaying Your Document in a Web Browser

- Press and hold down the ALT key, then press the TAB key at the same time. Repeatedly press the TAB key (while holding down ALT) until the browser icon is highlighted in the pop-up window, and then release ALT. (This is how you switch between applications in Windows.)
- Select **Open...** from the File menu (or press CONTROL+O) in the browser.
- Click the **Browse...** button and choose the C:\Inetpub\wwwroot\prac1.html file you saved above, and click the Open button.

You should now see in the title bar of the browser window the text "Web Applications" and in the web page below, the one sentence of <body> text you wrote, "This is plain ..."

- So, the page displays correctly in your browser, but is it valid **XHTML**? Check using the XHTML validation service located at <http://validator.w3.org/>. If it's not valid, what do you need to change to make it valid XHTML?

## Part B – Formatting the Text in Your HTML Document

- ALT+TAB back to your prac1.html document in Notepad. In the body text, press the RETURN key a few times to insert a few line breaks. Add a few more spaces between some words. Save the document (File → Save).
- Switch back to the browser and refresh the page (in IE press F5).

The browser has ignored the extra line breaks and spaces that you placed in the HTML. This is because HTML is specifically designed to ignore “whitespace” and “wrap up” as many words as possible into each line on the page.

- Back in Notepad, add a line break tag `<br />` in the body text somewhere. Replace the repeated spaces you put in with the special\* non-breaking space character, `&nbsp;`. You should have something like `This is plain &nbsp;&nbsp;&nbsp;&nbsp; text<br />`  
Save the document.
- Switch back to the browser and refresh the page (press F5).

\*Note: Many characters that appear in normal text may not be rendered correctly by the browser – including spaces, quotes, bullets, copyright symbols, etc. There are special encodings for these characters – try these out: `&quot;`; `&amp;`; `&copy;`; `&acute;`; `&frac14;`; `&sect;`; `&aacute;`; `&auml;`; `&aelig;`

## HTML Headings

- In Notepad, add a new line just after the `<body>` tag, and type in the heading something like this: `<h1>Heading 1</h1>`.
- Save the document in Notepad, and reload in the browser.

There are six levels of headings in HTML – h1, h2, h3, h4, h5 and h6.

## Breaking it up into paragraphs

We've seen earlier that a web browser will ignore all of the CARRIAGE RETURNS typed into your text editor. But, wherever a browser sees the paragraph tag, it inserts a blank line and starts a new paragraph. The HTML code for forcing a paragraph break is `<p>` and the closing tag is `</p>`.

- Using Notepad, enclose some text with the paragraph tag, save the file and refresh the page in the browser.

You have probably already noticed that the heading tags will automatically place the heading text on a new separate line. You can also use the horizontal rule tag (<hr />) to insert a section break.

### Styled Text

Just like a word processor, HTML can tell a web browser to display certain portions of text in Italic or Bold style or even a combination of several styles.

HTML	What it looks like in the browser
<code>&lt;b&gt;This is Bold...&lt;/b&gt;</code>	<b>This is Bold ...</b>
<code>&lt;i&gt;This is Italic...&lt;/i&gt;</code>	<i>This is Italic ...</i>
<code>&lt;tt&gt;This is Typewriter...&lt;/tt&gt;</code>	<code>This is Typewriter...</code>

- Using Notepad, enclose some text with the tags in the table above. Save the file, and refresh the page in the browser.

## Graphics and Images in HTML

Images are an indispensable part of the www. So let's put in some!

There are two main image formats that all browsers can handle: GIF and JPEG. The strengths and weaknesses of these formats will be covered in lectures.

- First we need to get some images to use ... Open a new browser window, either by double clicking the icon on the desktop again, or by File → New → Window in the browser.
- Go to <http://www.google.com> and select the Images tab. Type in a keyword of your choice, and select an image from the result. Download the image, saving it in the same directory as your prac1.html file. (C:\inetpub\wwwroot)

The HTML format for the inline image tag is: ``

Substitute **filename.gif** with the name of the file you downloaded, which should reside in the same directory as your HTML document. By "inline", this means that a web browser will display the image in between text. (You might want to experiment a little with where you place the tag to see how the text wraps around the image.)

- Type the following HTML into the body section of your prac1.html document:

```

```

- Save the document in Notepad, and reload in the browser.

If you have typed the filename correctly, and the image is in the same directory as your Notepad document, then you will see the image in the browser.

Along with the SRC attribute, the image tag has many more ... including ALT, HEIGHT, WIDTH, ALIGN, etc. We will leave it up to you to investigate these attributes! Look up the web for tutorials on HTML or refer to the W3C (World Wide Web Consortium) web site, [www.w3.org](http://www.w3.org).

## Hyperlinking with Anchors

The real power of the web is the ability to create hypertext links to related information. That other information may be other web pages, graphics, sounds, digital movies, animations, software programs, the contents of a file server, a log-in session to a remote computer, a software archive, or an "ftp" site.

The World Wide Web uses an addressing scheme known as URLs, or Uniform Resource Locators (sometimes also called "Universal Resource Locator"), to indicate the location of such items. These hypertext links, the ones [usually underlined and coloured blue](#), are known as anchors.

Now, let's create a link to a local file (e.g. a file from the same website) ...

- You need to create a new, separate HTML file. To avoid retyping the whole structure, the simplest thing to do is switch to Notepad, and use File → Save As to save the current document under a new filename (which will create another copy of the file). Save this new document in the same place as the original (C:\inetpub\wwwroot), and call it prac2.html

- Change the body text of the new page – so your two pages aren't identical! (Otherwise you won't see the difference.) Maybe put in a new heading (h1, h2...).

Note: The <title> tag in the head section of the HTML page defines what appears as the window name in the blue title bar at the top of the window. You should try to create a short but unique name for each page.

The simplest anchor link is one that opens another HTML file in the same directory as the presently displayed web page. The HTML format for doing this is:

```
<a href="filename.html">text that responds to a click</a>
```

Think of it as "a" for anchor link and "href" for "hypertext reference".

The filename must be another HTML file. Whatever text occurs after the first > and before the closing </a> symbols will be the "hypertext" that appears underlined and "hyper."

- Switch to Notepad if required, and make sure you have the prac2.html document window open. Type the following as the last line of the body section of the page:

```
<a href="prac1.html">Go to First Page</a>
```

- Save the prac2.html document in Notepad, switch back to the browser (remember ALT+TAB) and open the document you just edited. (File → Open → Browse ...)

You should be able to see the hyperlinked text Go to First Page at the bottom of the browser window. If you click the link, the browser will jump to your original HTML document (prac1.html). If this doesn't work, you have made a syntax error (typo!).

- Switch back to Notepad and Open the original prac1.html document. Type the following as the last line of the body section of the page:

```
<a href="prac2.html">Go to Second Page</a>
```

- Save the prac1.html document in Notepad, and switch (ALT+TAB) back to the browser, and refresh the browser window.

You should now be able to jump back and forth between your two HTML documents (prac1.html and prac2.html). If this doesn't work, you have made syntax error/s (typo!).

## Hyperlinking to files in other directories

The anchor tags can also link to an HTML document or graphic file in another directory/folder in relation to the document that contains the anchor. For example, it is recommended practice to keep all of the graphics in a separate directory/folder called images. As you create more and more HTML files, keeping things organised is increasingly important.

- Create a new directory in the C:\inetpub\wwwroot\ directory and call it "images". (Right mouse click in the file Explorer window and choose New → Folder)
- Save a graphic file into this directory. You can either download a new graphic, or copy the one you already used.

- Switch to Notepad if required, and make sure you have the prac2.html document window open. Type the following somewhere in the body section of the page:

```
<a href="images/insert_your_image_name.gif" >Show Picture</a>
```

Note: We have placed the directory name (images) into the URL for the hyperlink! You are also creating a link to an image file rather than an HTML document.

- Save the prac2.html document in Notepad, and switch (ALT+TAB) back to the browser, and jump to prac2.html. You may need to REFRESH (remember F5?) the browser window to see the new version of the HTML document.

You should now see a link ([Show Picture](#)) that will load the image by itself into the browser window. If this doesn't work, you have made syntax error/s (typo!). You need to use the browser BACK button to go back.

### Anchor Links to a Higher Level Directory

The types of links we have constructed here are known as RELATIVE URLs, meaning a web browser can construct the full URL based upon the current location of the HTML page and the link information in the `<a href=...>` tags. This is very powerful because you can build all your web pages on one computer, test them, and move them to another computer – all the relative links will stay intact.

We constructed a hyperlink to a document that is stored in a directory lower than the working HTML page. Note that you can also construct a link that connects to a higher level directory as well by using this HTML:

```
<a href="../../../default.html">return to home</a>
```

Each instance of "../" in the URL of an anchor link tells the web browser to go up one level higher in the directory structure relative to the current page; in this case to go up two directory/folder levels and look for a file called default.html

**Note:** You cannot link to files that are at a higher level than the server root directory. The root directory is the highest level directory that the web server has access to, e.g. <http://localhost/default.asp> looks for the file default.asp in the C:\inetpub\wwwroot\ (root) directory on your local computer.

Sometimes you may need to specify a page absolutely, so that the link points to the same file no matter which site directory it is in. These ABSOLUTE URLs specify a file path from the web server root directory. They start with a forward slash "/" which tells the web server to "begin at the top". For example, the link: `<a href="/default.html">return to home</a>` would work anywhere.

### URLs – Connecting to Files on the Internet

When linking files within your website or web application, you should generally use RELATIVE URLs. These use the syntax we described above (e.g. `<a href="../dir_name/file_name">`). You can also link to a document on any other (accessible) website on the www, by using a FULL URL.



The structure of a full URL is:

```
protocol://server.sub-domain.domain/directory/sub-  
directory/.../filename
```

(protocol is generally http, ftp or file)

- Switch to Notepad if required, and make sure you have the prac1.html document window open. Type the following somewhere in the body section of the page:

```
<a href="http://www.scitech.qut.edu.au/">FaST website</a>
```

- Save the prac1.html document in Notepad, and switch back to the browser, and refresh the window.

You should now be able to jump to the FaST home page.

- Switch to Notepad if required, and make sure you have the prac1.html document window open. Modify the FaST home page link you just created like this:

```
<a href="http://www.scitech.qut.edu.au/" target="_blank">FaST  
website</a>
```

- Save the prac1.html document in Notepad, and switch back to the browser, and refresh the window.

Now when you click the link to the FaST home page it will load into a new browser window! The target attribute allows you to specify which browser window you want the URL to be load into. Target recognises the “\_blank” keyword (among others) to tell the browser to create a new window. This should be used with great care – people hate windows popping up unexpectedly!

## Links to Sections of a Page

You have seen how to link to other web pages, whether they are ones you created or have found elsewhere on the Internet. What if you wanted to connect to a specific section within a document? YOU CAN!

You need to create named anchors within your HTML document. These are hidden reference markers, or “bookmarks”, for a particular section of your HTML file. This might be used to link to a different section of the same page if it is long, or to a marked section of another page.

For example, if your HTML document was very long, you may want to provide a link back to the TOP of the page. You would create a hidden marker at the line in your document you wish to jump to called (for example) “top”. Then, you write an anchor link that connects to this section of this document. Once you click on a link to this named anchor, your web browser will jump so this line.

- Switch to Notepad if required, and make sure you have the prac1.html document window open. Type the following as the first line in the body section of the page:

```
<a name="top">Top Anchor</a>
```

Note: You may make the anchor invisible by using &nbsp; instead of the text “Top Anchor” or even leaving the anchor text empty. Also note that this is a “name” anchor, not an “href” anchor.



- At the bottom (last line) of the document, create a link to the “top” anchor:

```
<a href="#top">Top of Page</a>
```

Note: The hash (#) is important! It tells the browser that the hyperlink points to a named anchor.
--

- Save the **prac1.html** document in Notepad, and switch back to the browser, and refresh the window. (To get the full effect, you might need to add some more paragraphs to your page to make it longer.) The Top of Page link should jump to the top of the same page.

You can also create a link that jumps to a section of another HTML document that is marked by a named anchor, i.e. jump right into the middle of a page. The HTML for building a link to a named anchor in another local HTML document is:

```
<a href="file.html#NAME">Text to activate link</a>
```