Adobe Drewamweaver 2021

2 – HTML

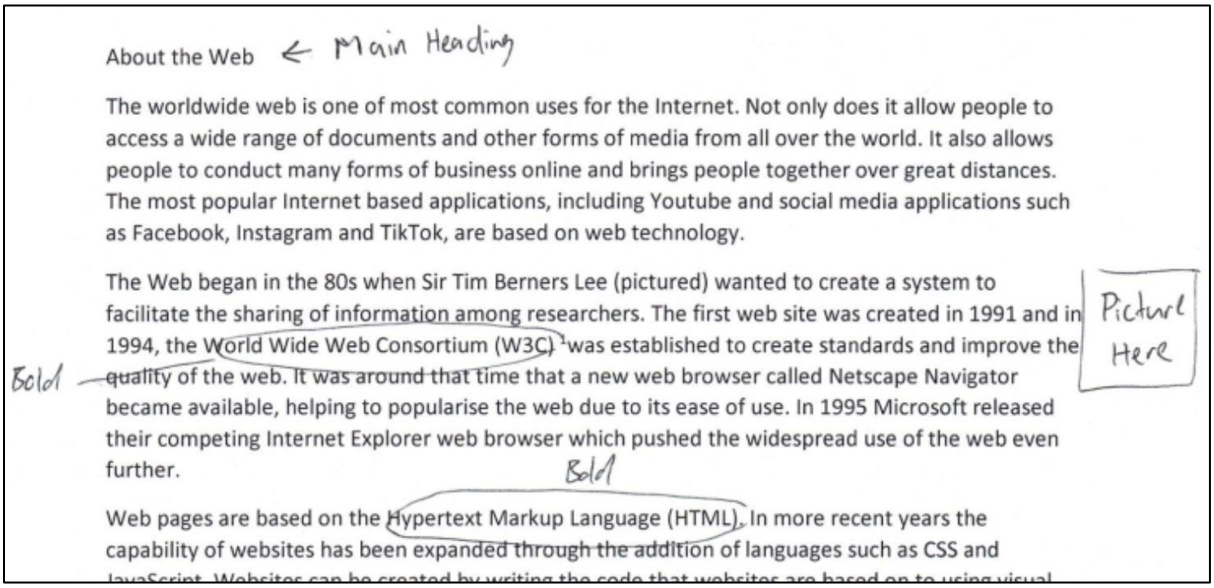
Web pages are created using **HTML** (Hypertext Markup Language). Dreamweaver can be used in Design view as a **WYSIWYG** (What You See Is What You Get) editor. This means that you can edit the document on screen while it's looking more or less the way it would appear as a web page. It is similar to the way that editing a document in a Word Processor (such as Microsoft Word) allows you to edit the document looking much the way it will appear when printed.

Dreamweaver also allows you to edit HTML code directly. In fact, in many cases, editing the code directly is preferable to working purely in the WYSIWYG view. Often, working with the code directly can make it easier to understand how your website works, including understanding what problems you may be having with your site along with your site along with how to fix those problems. Whichever way you work with Dreamweaver, it is a good tool for creating HTML code that conforms closely to W3C standards.

What is a Markup Language?

HTML is a **Markup language** which is a bit different from programming languages that you may have learned about. A programming language tells a computer what to do. E.g. add some numbers, store some information, respond to buttons pressed on the keyboard.

A Markup language on the other hand, tells a computer how to present a document. Consider the following example.



About the Web ← Main Heading

The worldwide web is one of most common uses for the Internet. Not only does it allow people to access a wide range of documents and other forms of media from all over the world. It also allows people to conduct many forms of business online and brings people together over great distances. The most popular Internet based applications, including Youtube and social media applications such as Facebook, Instagram and TikTok, are based on web technology.

The Web began in the 80s when Sir Tim Berners Lee (pictured) wanted to create a system to facilitate the sharing of information among researchers. The first web site was created in 1991 and in 1994, the World Wide Web Consortium (W3C) was established to create standards and improve the quality of the web. It was around that time that a new web browser called Netscape Navigator became available, helping to popularise the web due to its ease of use. In 1995 Microsoft released their competing Internet Explorer web browser which pushed the widespread use of the web even further.

Web pages are based on the Hypertext Markup Language (HTML). In more recent years the capability of websites has been expanded through the addition of languages such as CSS and JavaScript. Websites can be created by writing the code that websites are based on to using visual

If you had been asked to type this document on a computer, you would notice that the document has been **marked** with additional information. Make certain parts bold. Add a picture. Format the first line as a heading etc. A markup language performs a similar role. E.g. Where paragraphs and headings are. Where links and pictures are. Even which parts of the text should be emphasised. In HTML these instructions are referred to as **tags**.

Structure vs Presentation

HTML is a **markup language**. It stands for HyperText (meaning the links in web pages) Markup Language. in its original form was created to define the structure of a website. However, HTML didn't define how a web page would actually look in a web browser. The HTML code might say that part of the page was a heading, but it was up to the web browser to decide how big the text in that heading was, whether or not it was bold etc.

As the World Wide Web became more popular, people wanted more control over how their pages looked. The HTML language was expanded with the capability to control things like the size and colour of text. Adding extra elements to the language though, meant that HTML was no longer simply defining the structure of a web page as intended. It was now also handling the way a web page was presented. The way it was being done was not very good though and led to web pages with bloated code that often didn't work very well in different web browsers.

The solution was to create a new language called **Cascading Stylesheets** (**CSS**). Modern web page standards use HTML to define the structure of a web page as originally intended, while the CSS language controls how a web page appears. E.g. HTML will say what parts of the page are headings, while the CSS says how those headings actually look. As we move through these exercises you will see how having the Structure and Presentation of web pages handled by different languages leads to more versatile and flexible websites that are easier to edit along with a range of other advantages.

Typically, as we work through these exercises, we will start by using HTML to create the structure of a page and then will later use CSS to establish how those pages will be presented.

Note Web pages also often use a programming language such as **JavaScript** to provide additional functionality. JavaScript is not covered in these exercises other than a few small examples in the more advanced exercises.

HTML Tags

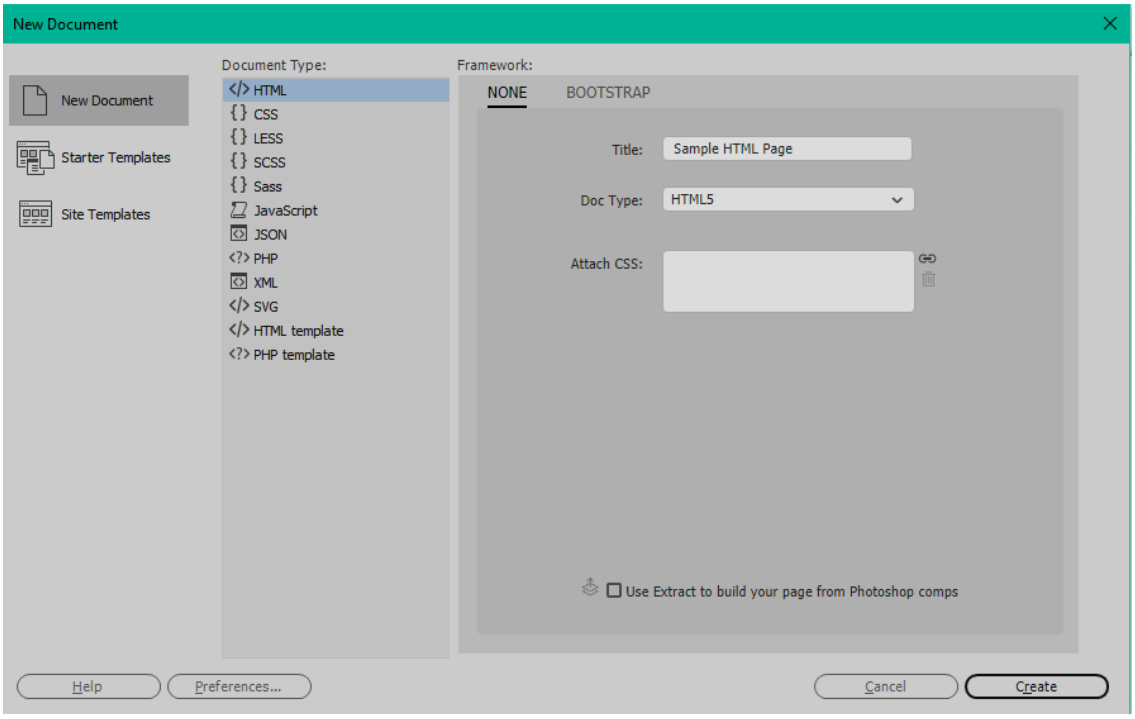
Web pages consist of various page **Elements** such as paragraphs and headings. These elements are created using **Tags**. Tags in HTML are instructions that are contained between **<** and **>** symbols. When a web browser sees anything between those symbols, it knows that it is a tag that needs to be interpreted in a certain way.

Most HTML tags have an opening tag and a closing tag. In the example below, we have a <p> tag to indicate where a paragraph will begin and a closing tag which indicates where the paragraph will end. Note that the closing tag has a / symbol inside it.

```
<p>This text will appear in a web browser as a paragraph. </p>
```

Exercise 1. HTML Page Elements

1. **Open Dreamweaver** if it isn't already open.
2. From the **startup** screen if it is showing click the **Create New** button to create a new HTML document. You can also select **New** from the **File** menu or press **[Ctrl] [N]**.

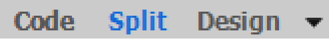


Title: Sample HTML Page

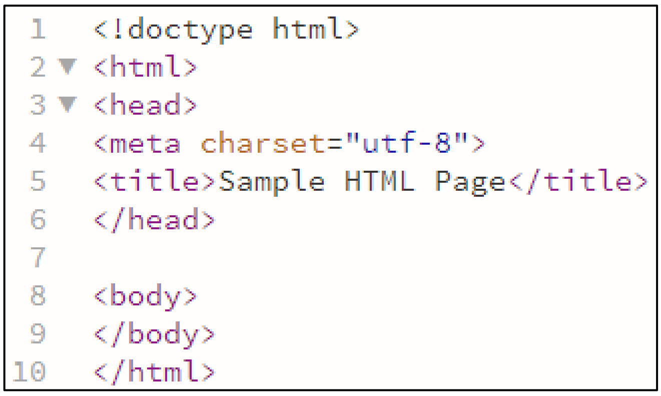
Doc Type: HTML5

☐ Use Extract to build your page from Photoshop comps

3. Enter *Sample HTML Page* for the page **Title** and then click **Create**.
4. Switch to **Split** view so that one half of the editing window shows the HTML code.
5. Change the other half of the screen to **Design** view.



A document will be created which includes the basic elements that each web page should have.



```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Sample HTML Page</title>
</head>

<body>
</body>
</html>
```

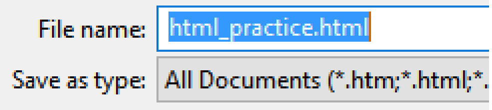
Each of these elements are explained below.

Note Dreamweaver shows HTML tags in different colours to make it easier for you while you are editing code. The colours that are used in code editing can be changed in Dreamweaver's preferences. You can even change to editing with light text on a dark background if you prefer.

<!doctype html>	Tells the web browser that this document uses the HTML language
<html> </html>	Tells the web browser where the HTML code starts and finishes. All other parts of your HTML should be between this opening and closing tag.
<head> </head>	The head section of your document contains information about your document such as a page title, keywords and description of the page. This information will not appear on the page itself.
<meta charset="utf-8">	This tag tells the browser that the page uses a Unicode character set. A commonly used collection of letters, numbers and symbols.
<title> </title>	The title should contain a brief description of the page (approximately 3 to 5 words). A page title is used in a few important ways: • It appears in the title bar of a web browser when the page loads • It is used in web search results • It can be used as a label when someone adds a web site to their bookmarks or favourites
<body> </body>	The content of a page is placed in the body section. Anything that is intended to be visible in a web page should be between the body tag and closing body tag.

Exercise 2. Saving a Page

1. From the **File** menu select **Save** or press **[Ctrl] [S]** to save the file.
2. Select a location to save the file and enter *html_practice.html* for the file name.



File name: html_practice.html

Save as type: All Documents (*.htm;*.html;*.

A few things to remember about filenames in websites (this applies not only to HTML files but to other files that will be used in your site, such as images and stylesheets).

- **Avoid uppercase letters**. If you name a file with uppercase letters such as *Document.html*, you will have to type it exactly the same every time you refer to it. Referring to it as *document.html* won’t work in a website as your web browser will treat *“Document.html”* and *“document.html”* as different files. Keeping names all lowercase makes things a bit easier and reduces the chance of broken links.
- **Don’t use spaces** since some web servers don’t allow filenames with spaces. File names made up of more than one word are fine but use underscores (_) instead of spaces. Holding down the **Shift** key and pressing the dash/minus sign key **–** will give you an underscore. Alternatively use **CamelCase** (first letter of each word uppercase).
- When you are saving the main page in a website, it is best to call it **index.html**. This is the standard name for the main page on a site. If you type http://www.oneil.com.au/pc/ in to your web browser it will go to http://www.oneil.com.au/pc/index.html since it will look in that location for a file named index.html. Basically, if your main page is called index, it allows you to drop the filename off your site’s address, making it shorter and easier to type.

3. Click **Save** to save the file.

Exercise 3. Additional Page Elements

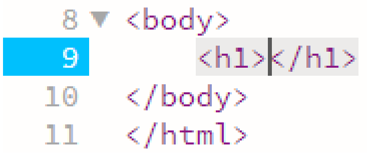
Now that we have created the main structure of the page, we will begin adding some additional elements within the body section. Remember that the parts of the page you want to appear within your web browser need to be in the body section of your HTML.

1. In the HTML code, create a blank line between the <body> and </body> tags.
2. Type <h1>. h1 indicates a level 1 heading which should be the main headings in your document. Other numbers are used for subheadings (from h1 through to h6).

You may have noticed as you began typing the tag that a list of tags appeared. This is to help you with HTML by allowing you to select a tag from the list. If the tag you want is already selected in the list, you can press **[Enter]** to complete the tag. h1 isn't a lot to type though so you probably didn't need the help this time.

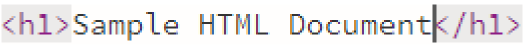
When you type the > to close the tag, Dreamweaver will usually add the closing tag for you. If it doesn't, you can always type it yourself. Most tags require a closing tag to tell a web browser where to end that page element (in this case where the end of the heading is). If you forgot to add a closing tag or mis-typed the closing tag, a web browser might think that everything until the end of the document is still part of the heading and display it accordingly. Incorrectly closing a tag is a common mistake, but fortunately an easy mistake to spot and correct.

Your HTML should now look like the following example.



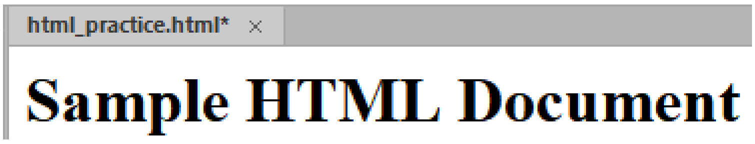
```
<body>
    <h1></h1>
</body>
</html>
```

3. Click between the opening and closing heading tag as shown above to make sure you are ready to type between them.
4. Type *Sample HTML Document* between the opening and closing h1 tag.



```
<h1>Sample HTML Document</h1>
```

While you are editing in split screen view, any changes you make in the **Code** part of the screen will also be shown in the other part of the screen (the **Design** or **Live** view).



Sample HTML Document

5. Add two lines of HTML after your heading as shown.

```
<h1>Sample HTML Document</h1>
<h2>Subheading</h2>
<p>A paragraph of text with a link to a website and an image which is aligned to the right
of the page</p>
```

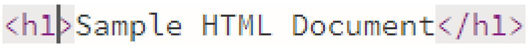
6. Press **Ctrl S** to save your changes

Tip While you have a tag selected you can press **Ctrl K** to open **Quick Docs**. A handy little reference that will provide some information about the selected tag. Try it!

Exercise 4. Element Attributes

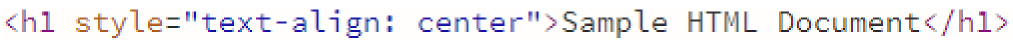
Many HTML elements have additional options that can be set using attributes within the tags.

1. In the HTML, click in the heading tag right after the number 1 as shown.



```
<h1>Sample HTML Document</h1>
```

2. Type a space and then type style="text-align: center"[1] so that it appears as below.

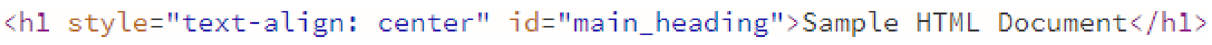


```
<h1 style="text-align: center">Sample HTML Document</h1>
```

A tag attribute contains two parts. The **name** of the attribute and the **value** of the attribute separated by an **=** equal sign. The value should be surrounded by double quotes. Single quotes can also be used but double quotes are more common. It is important to make sure the ones on both sides are the same type. In the above example, style is the name of the attribute while text-align: center is the value of the attribute. It is not necessary to include attributes in a closing tag.

Note For people outside USA, note the US spelling. HTML only recognises US spelling so putting 'centre' won't do anything. The same also applies for words like COLOR (colour) and GRAY (grey).

3. An element can have more than one attribute. Change the line of HTML so it looks like the following (with an underscore _ instead of a blank space. Holding shift while pressing the – key on your keyboard will add an underscore).



```
<h1 style="text-align: center" id="main_heading">Sample HTML Document</h1>
```

The id attribute doesn't change the heading. It gives it a name. This is useful later when it can become important for HTML elements to be referred to from elsewhere.

Note It doesn't matter which order attributes appear in so
<h1 style="text-align: center" id="main_heading">
is the same as
<h1 id="main_heading" style="text-align: center">

4. Add additional tags to your HTML so that it looks like the following.

```
<h1 style="text-align: center" id="main_heading">Sample HTML Document</h1>
<h2>Subheading</h2>
<p><img src="http://www.oneil.com.au/pc/dw_logo.png" align="right" height="47"
width="47" alt="Dreamweaver Logo">A paragraph of text with a
<a href="http://www.oneil.com.au/pc/">link to a website</a> and an image which is
aligned to the right of the page</p>
```

Note When you have elements nested inside of other elements then the last tag should be the first one closed. E.g. instead of <h2><strong>Bold Heading</h2></strong>
it should be <h2><strong>Bold Heading</strong></h2>

[1] This Style attribute places some CSS formatting in your HTML. You could also use the older align= "center" attribute to get the same result, but using the more modern CSS code is recommended.

Exercise 5. HTML Comments

Like most computer languages, HTML allows for comments in the code. Comments are text that can be seen while viewing the code and will not show up in the actual web page.

HTML comments begin with <!-- and end with -->. Anything in between will be ignored by the web browser.

1. Create a new blank line after the <body> tag.
2. Enter the following, adding in your own name.

```
<body>
    <!-- This HTML code was written by Steve -->
    <h1 style="text-align: center" id="main_heading">Sample HTML Document</h1>
```

3. Preview the page in the browser. The comment will not appear unless you right-click on the page and choose to view source.

Comments can be used to add notes for the developer or other people who want to view the code of a site. Since anything inside comments is ignored by the browser, they can also be used to temporarily disable blocks of HTML during testing.

4. Add the following line to your **HTML**. This code uses the abbr tag which allows you to add an abbreviation. In many browsers, hovering the mouse over the abbreviation will show the text in the title attribute.

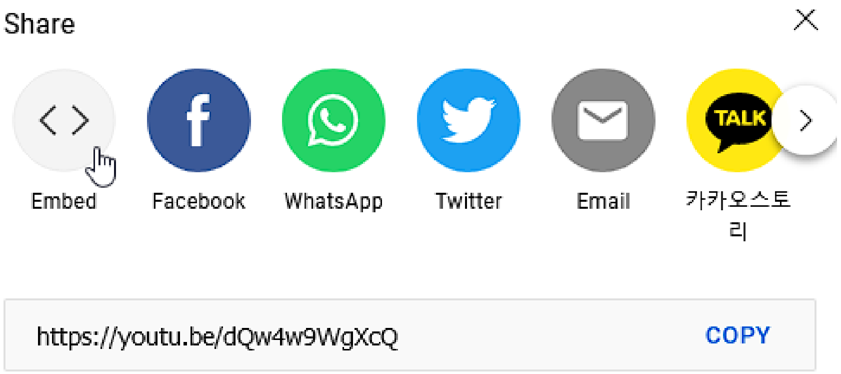
```
<p>HTML follows standards set by the <abbr title="World Wide Web
Consortium">W3C</abbr>.</p>
```

5. Preview the page to view the new line.
6. Now place the line inside a comment so it looks like the following.

```
<!-- <p>HTML follows standards set by the <abbr title="World Wide Web
Consortium">W3C</abbr>.</p> -->
```

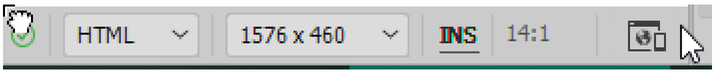
7. Preview it again in the browser. This time, the line inside the comment will not appear.

Tip Many web applications allow you to embed content in your own site's HTML. For example, on Youtube or Google maps, when you choose to share a video or map location one of the options will be **embed** which gives you code you can copy and paste into your own site. This lets you embed videos, maps and other online content in your own pages. Try it out!



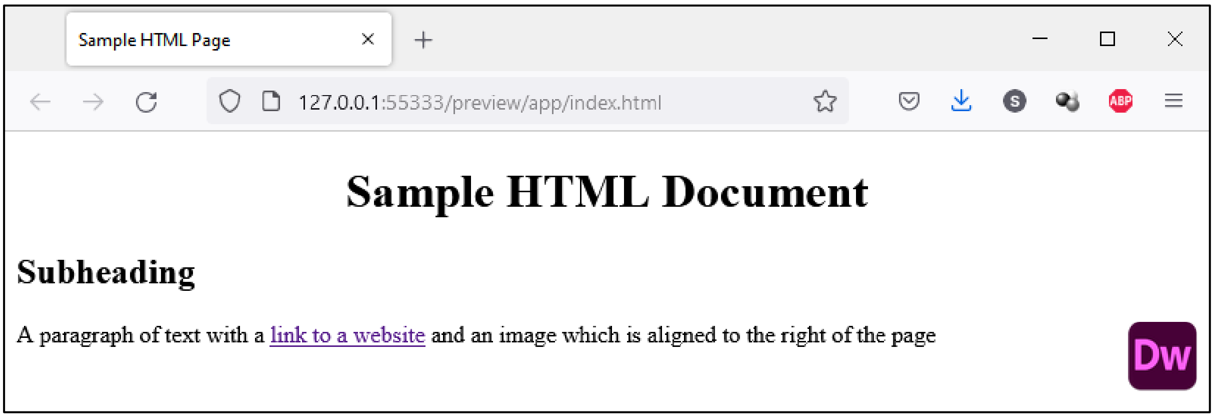
Exercise 6. Previewing the Document

1. Click the **Preview** icon in the bottom corner of your editing window.



A list of web browsers will appear. Testing how your web page looks in more than one browser is important as there can sometimes be small differences in how things look from one browser to another.

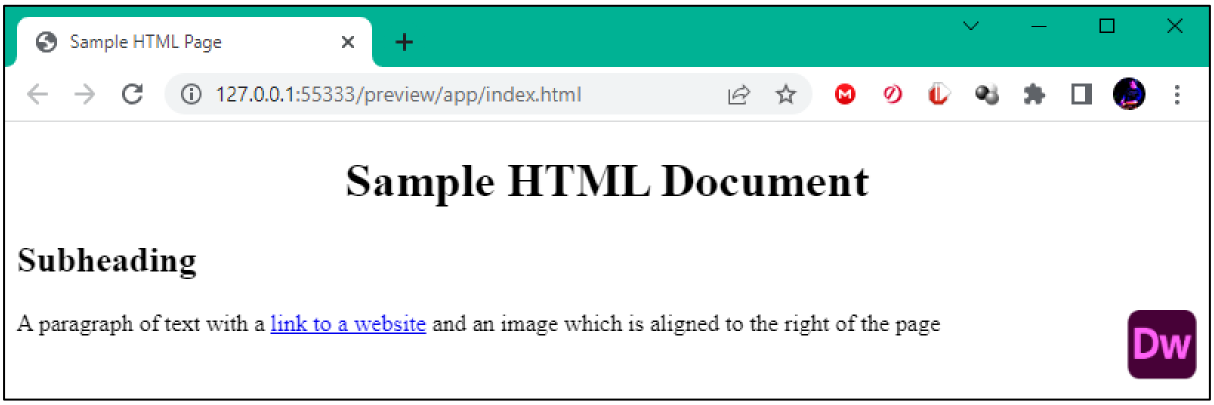
2. Preview the page in your primary browser. Below are examples of how the page looks in **Mozilla Firefox** and **Google Chrome**. Note the title appearing in the browser tab at the top.



Sample HTML Document

Subheading

A paragraph of text with a link to a website and an image which is aligned to the right of the page

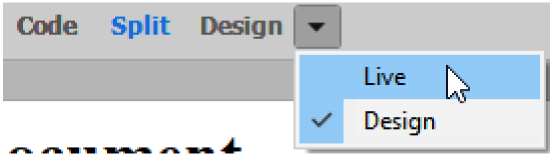


Sample HTML Document

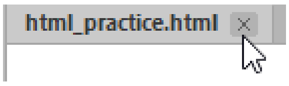
Subheading

A paragraph of text with a link to a website and an image which is aligned to the right of the page

3. You can also preview your page to see how it would look in a browser by changing from **Design** view to **Live** view. This gives you a browser preview without having to leave Dreamweaver. Some features of Dreamweaver require you to be in Live view but we will mostly work in Design view.



4. Change back to **Design** view and close the file when done, **saving** changes when prompted.



Exercise 7. Quick Revision

1. Write **True** or **False** next to each of the following statements.
   a) It is essential for attribute values to have quotation marks around them.

   b) Only US spelling is recognised in HTML.

   c) It doesn't matter if tags are written in upper case or lower case letters.

   d) The contents of the Head section will not appear in the web browser's main window.

2. For each of the following lines of HTML, correct the error and explain why you believe it is an error.

```
<p><strong><em><a href="prices.html">Our price list</strong></a></em></p>
```

Correction ____

Explanation ____

____

____

____

```
<img src="portrait.jpg" alt="Photo" width="300" height="150" border="0>
```

Correction ____

Explanation ____

____

____

____

```
<h1 style="text-align:centre">Bob's Garden Services</h1>
```

Correction ____

Explanation ____

____

____

____

```
<p><align="center">Low prices every day!</align></p>
```

Correction ____

Explanation ____

____

____

____

Exercise 8. Quick Revision - Answers

3. Write **True** or **False** next to each of the following statements.
   e) It is essential for attribute values to have quotation marks around them. **False** – it isn't essential but it is recommended
   f) Only US spelling is recognised in HTML. **True**
   g) It doesn't matter if tags are written in upper case or lower case letters. **True**, and it used to be common for people to write tags in uppercase. These days though, the HTML specification prefers lower case.
   h) The contents of the Head section will not appear in the web browser's main window. **True** – Though the title might appear in the browser tab at the top. The head section contains information that is used elsewhere. E.g. page title or keywords that might be used in internet searches.
4. For each of the following lines of HTML, correct the error and explain why you believe it is an error.

<p><strong><em><a href="prices.html">Our price list</strong></a></em></p>

Correction <p><strong><em><a href="prices.html">Our price list</a></em></strong></p>

Explanation - Tags should be closed in reverse of the order in which they were opened. I.e. the last tag opened should be the first one closed. The first one open should be the last one closed

<img src="portrait.jpg" alt="Photo" width="300" height="150" border="0>

Correction <img src="portrait.jpg" alt="Photo" width="300" height="150" border="0”>

Explanation - In the border attribute there was a closing “ before the number but there wasn't one after it. Leaving the quotes out altogether is ok (but not recommended). Putting quotes before and after is also ok. Having quotes on only one side is not ok.

<h1 style="text-align:centre">Bob's Garden Services</h1>

Correction <h1 style="text-align:center">Bob's Garden Services</h1>

Explanation - This would only work if US spelling was used for center. If you spelled it as centre, your web browser would not recognise it and the alignment would be ignored. So the heading would probably be displayed as left aligned instead.

<p><align="center">Low prices every day!</align></p>

Correction <p style="text-align:centre">Low prices every day!</p>

Explanation - There is no align tag in HTML. Instead, paragraphs are aligned using an attribute in the <p> tag. When the paragraph tag is closed with </p> it also closes any of the attributes within that tag.