**University of Wolverhampton**

**School of Engineering, Computational and Mathematical Sciences**

**5CS020 Human-Computer Interaction**

**Workshop 1 - Introduction to HTML Part 1**

As you will be creating and experimenting with user interface via web pages, it is important that you understand HTML

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.

Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.

HTML is a Markup Language which means you use HTML to mark-up a text document with tags that tell a Web browser how to structure it to display.

HTML was originally developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

**Basic HTML Document**

In its simplest form, following is an example of a simple HTML document :

<!DOCTYPE html>

<html>

<head>

<title>This the document title</title>

</head>

<body>

<h1>This is a heading</h1>

<p>Document content goes here.</p>

</body>

</html>

Type or copy this into a new file in a text editor (Notepad, Notepad++, or VSCode, or any other text editor) and same it as webpage.html.

Then open this fie in a web browser (Edge or Chrome), and you should get something similar to he following:

Text

Description automatically generated

**HTML Tag Description**

<!DOCTYPE...>

This tag defines the document type and HTML version.

<html>

This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.

<head>

This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.

<title>

The <title> tag is used inside the <head> tag to mention the document title.

<body>

This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc.

<h1>

This tag represents the heading.

<p>

This tag represents a paragraph.

**Heading Tags**

A document starts with a heading. HTML has six levels of headings, which use the elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

Type or copy the following HTML into your webpage file and open it in a web browser:

<!DOCTYPE html>

<html>

<head>

<title>Heading Example</title>

</head>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

<h4>This is heading 4</h4>

<h5>This is heading 5</h5>

<h6>This is heading 6</h6>

</body>

</html>

You should see the following:

Text

Description automatically generated

**Paragraph Tag**

The <p> tag offers a way to structure your text into different paragraphs.

Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below:

<!DOCTYPE html>

<html>

<head>

<title>Paragraph Example</title>

</head>

<body>

<p>Here is a first paragraph of text.</p>

<p>Here is a second paragraph of text.</p>

<p>Here is a third paragraph of text.</p>

</body>

</html>

Enter the above and view the webpage. You should see :

A picture containing text, indoor, screenshot

Description automatically generated

**Line Break Tag**

Whenever you use the <br /> element, anything following it starts from the next line. This tag is an example of an empty element, where you do not need opening and closing tags, as there is nothing to go in between them.

Enter and view the following example in your web browser:

<!DOCTYPE html>

<html>

<head>

<title>Line Break Example</title>

</head>

<body>

<p>Hello<br />

You submitted your assignment on time.<br />

Thanks<br />

Jeffrey</p>

</body>

</html>

You should get:

Text

Description automatically generated with medium confidence

Centring Content

You can use <center> tag to put any content in the center of the page or any table cell.

<!DOCTYPE html>

<html>

<head>

<title>Centring Content Example</title>

</head>

<body>

<p>This text is not in the centre.</p>

<center>

<p>This text is in the centre.</p>

</center>

</body>

</html>

You should get :

Text

Description automatically generated

**Horizontal Lines**

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

For example, you may want to give a line between two paragraphs as in the given example below

<!DOCTYPE html>

<html>

<head>

<title>Horizontal Line Example</title>

</head>

<body>

<p>This is paragraph one and should be on top</p>

<hr />

<p>This is paragraph two and should be at bottom</p>

</body>

</html>

You should get:

Graphical user interface, text

Description automatically generated

**Preserve Formatting**

Sometimes, you want your text to follow the exact format of how it is written in the HTML document. In these cases, you can use the preformatted tag <pre>.

Any text between the opening <pre> tag and the closing </pre> tag will preserve the formatting of the source document.

<!DOCTYPE html>

<html>

<head>

<title>Preserve Formatting Example</title>

</head>

<body>

<pre>

function testFunction( strText ){

alert (strText)

}

</pre>

</body>

</html>

You should get

Text

Description automatically generated with medium confidence

Now try removing the <pre> and </pre> tags and see what happens.

**HTML Attributes**

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts − a name and a value

The name is the property you want to set. For example, the paragraph <p> element in the example carries an attribute whose name is align, which you can use to indicate the alignment of paragraph on the page.

The value is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: left, centre and right.

<!DOCTYPE html>

<html>

<head>

<title>Align Attribute Example</title>

</head>

<body>

<p align = "left">This is left aligned</p>

<p align = "center">This is centre aligned</p>

<p align = "right">This is right aligned</p>

</body>

</html>

Does it do what you expect?

Text

Description automatically generated with low confidence

**Core Attributes**

The four core attributes that can be used on the majority of HTML elements (although not all) are

* Id
* Title
* Class
* Style

**The Id Attribute**

The id attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an element

* If an element carries an id attribute as a unique identifier, it is possible to identify just that element and its content.
* If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

**The title Attribute**

The title attribute gives a suggested title for the element. The behaviour of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

<!DOCTYPE html>

<html>

<head>

<title>The title Attribute Example</title>

</head>

<body>

<h3 title = "Hello HTML!">Titled Heading Tag Example</h3>

</body>

</html>

This will produce the following



Bring your mouse cursor over "Titled Heading Tag Example" and you will see that whatever title you used in your code is coming out as a tooltip of the cursor.

**The class Attribute**

The class attribute is used to associate an element with a style sheet, and specifies the class of element. The value of the attribute may also be a space-separated list of class names. For example

class = "className1 className2 className3"

**The style Attribute**

The style attribute allows you to specify Cascading Style Sheet (CSS) rules within the element.

<!DOCTYPE html>

<html>

<head>

<title>The style Attribute</title>

</head>

<body>

<p style = "font-family:arial; color:#FF0000;">Some text...</p>

</body>

</html>

Will produce



**Basic HTML Formatting**

**Bold Text**

Anything that appears within <b>...</b> element, is displayed in bold

<!DOCTYPE html>

<html>

<head>

<title>Bold Text Example</title>

</head>

<body>

<p>The following word uses a <b>bold</b> typeface.</p>

</body>

</html>

This will produce



**Italic Text**

Anything that appears within <i>...</i> element is displayed in italicized

<!DOCTYPE html>

<html>

<head>

<title>Italic Text Example</title>

</head>

<body>

<p>The following word uses an <i>italicized</i> typeface.</p>

</body>

</html>

This will produce the following



**Underlined Text**

Anything that appears within <u>...</u> element, is displayed with underline

<!DOCTYPE html>

<html>

<head>

<title>Underlined Text Example</title>

</head>

<body>

<p>The following word uses an <u>underlined</u> typeface.</p>

</body>

</html>

This will produce the following

A picture containing text

Description automatically generated

**Strike Text**

Anything that appears within <strike>...</strike> element is displayed with strikethrough, which is a thin line through the text

<!DOCTYPE html>

<html>

<head>

<title>Strike Text Example</title>

</head>

<body>

<p>This uses a <strike>strikethrough</strike> typeface.</p>

</body>

</html>

This produces



**Monospaced Font**

The content of a <tt>...</tt> element is written in monospaced font. Most of the fonts are known as variable-width fonts because different letters are of different widths (for example, the letter 'm' is wider than the letter 'i'). In a monospaced font, however, each letter has the same width.

<!DOCTYPE html>

<html>

<head>

<title>Monospaced Font Example</title>

</head>

<body>

<p>The following word uses a <tt>monospaced</tt> typeface.</p>

</body>

</html>

Produces



**Superscript Text**

The content of a <sup>...</sup> element is written in superscript; the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters.

<!DOCTYPE html>

<html>

<head>

<title>Superscript Text Example</title>

</head>

<body>

<p>The following word uses a <sup>superscript</sup> typeface.</p>

</body>

</html>

This produces



**Subscript Text**

In a similar way, the content of a <sub>...</sub> element is written in subscript; the font size used is the same as the characters surrounding it, but is displayed half a character's height beneath the other characters:

<!DOCTYPE html>

<html>

<head>

<title>Subscript Text Example</title>

</head>

<body>

<p>The following word uses a <sub>subscript</sub> typeface.</p>

</body>

</html>

This produces

A picture containing text

Description automatically generated

**Grouping Content**

The <div> and <span> elements allow you to group together several elements to create sections or subsections of a page.

For example, you might want to put all of the footnotes on a page within a <div> element to indicate that all of the elements within that <div> element relate to the footnotes. You might then attach a style to this <div> element so that they appear using a special set of style rules.

<!DOCTYPE html>

<html>

<head>

<title>Div Tag Example</title>

</head>

<body>

<div id = "menu" align = "middle" >

<a href = "/index.htm">HOME</a> |

<a href = "/about/contact\_us.htm">CONTACT</a> |

<a href = "/about/index.htm">ABOUT</a>

</div>

<div id = "content" align = "left" >

<h5>Content Articles</h5>

<p>Actual content goes here.....</p>

</div>

</body>

</html>

Produces

A picture containing logo

Description automatically generated

The <span> element, on the other hand, can be used to group inline elements only. So, if you have a part of a sentence or paragraph which you want to group together, you could use the <span> element as follows.

<!DOCTYPE html>

<html>

<head>

<title>Span Tag Example</title>

</head>

<body>

<p>This is the example of <span style = "color:green">span tag</span>

and the <span style = "color:red">div tag</span> alongwith CSS</p>

</body>

</html>

This will produce



**Meta Tags**

HTML lets you specify metadata - additional important information about a document in a variety of ways. The META elements can be used to include name/value pairs describing properties of the HTML document, such as author, expiry date, a list of keywords, document author etc.

The <meta> tag is used to provide such additional information. This tag is an empty element and so does not have a closing tag but it carries information within its attributes.

You can include one or more meta tags in your document based on what information you want to keep in your document but in general, meta tags do not impact physical appearance of the document so from appearance point of view, it does not matter if you include them or not.

You can add metadata to your web pages by placing <meta> tags inside the header of the document which is represented by <head> and </head> tags. A meta tag can have following attributes in addition to core attributes

* Name
  + Name for the property. Can be anything. Examples include, keywords, description, author, revised, generator etc.
* content
  + Specifies the property's value.
* scheme
  + Specifies a scheme to interpret the property's value (as declared in the content attribute).
* http-equiv
  + Used for http response message headers. For example, http-equiv can be used to refresh the page or to set a cookie. Values include content-type, expires, refresh and set-cookie.

**Specifying Keywords and description**

You can use <meta> tag to specify important keywords related to the document and later these keywords are used by the search engines while indexing your webpage for searching purpose.

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

If you display this in your web browser, you won't see visible difference to one without the meta tags, but search engines will note the additional information.

You can use <meta> tag to give a short description about the document. Again can be used by search engines while indexing your webpage for searching purpose.

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

<meta name = "description" content = "Learning about Meta Tags." />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

**Document Revision Date**

You can use <meta> tag to give information about when last time the document was updated. This information can be used by various web browsers while refreshing your webpage.

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

<meta name = "description" content = "Learning about Meta Tags." />

<meta name = "revised" content = "3/7/2022" />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

**Document Refreshing**

Whilst the previous <meta> tags don't seem to do much visible, this <meta> tag has an action which can affect HCI. It can be used to specify a duration after which your web page will keep refreshing automatically. If you want your page keep refreshing after every 5 seconds then use the following

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

<meta name = "description" content = "Learning about Meta Tags." />

<meta name = "revised" content = "3/7/2022" />

<meta http-equiv = "refresh" content = "5" />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

Try it in your web browser. If used incorrectly, it can be rather annoying.

**Page Redirection**

You can use the same <meta> tag to redirect your page to any other webpage, which can be more useful than constantly refreshing the same page. Following is an example of redirecting current page to another page after 5 seconds.

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

<meta name = "description" content = "Learning about Meta Tags." />

<meta name = "revised" content = " 3/7/2022" />

<meta http-equiv = "refresh" content = "5; url=http://www.wlv.ac.uk" />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

**Specify Character Set**

You can use <meta> tag to specify character set used within the webpage. By default, Web servers and Web browsers use ISO-8859-1 (Latin1) encoding to process Web pages. Following is an example to set UTF-8 encoding

<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name = "keywords" content = "HTML, Meta Tags, Metadata" />

<meta name = "description" content = "Learning about Meta Tags." />

<meta http-equiv = "Content-Type" content = "text/html; charset = UTF-8" />

</head>

<body>

<p>Hello HCI!</p>

</body>

</html>

**End of Part 1**

This is the end of this workshop. There will be more TML next week.