p.c.windridge@staffs.ac.uk

Introduction to HTML

Introduction

This tutorial provides an introduction to HTML5 (henceforth HTML) syntax. HTML is a mark-up language for web pages and, at the time of writing (September 2013), a W3C candidate recommendation.

This tutorial represents the 'least you need to know' in order to start using HTML.

It will introduce the syntax, the structure and how to check that a web page is valid HTML.

A Basic Example of HTML

Minimal structure

An example of a minimal HTML structure is:

```
<!DOCTYPE html><title>MyDoc</title><p>Hello
```

This is an example of the 'least you can get away with' but it should not be seen as good practice. The reason why this is valid is because the application that reads this will know what elements can be expected given the minimal structure. It can generate the missing parts of the structure and make assumptions about other aspects of the document so that it can do something with it (such as render it to a browser display).

Recommended template

It is much better to specify the details so that you, the developer, retain control.

This is an HTML 'skeleton' that you should use as the basis for your HTML pages.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>My First Web Page</title>
</head>
<body>
</body>
</html>
```

HTML Structure

Doctype

The first line in an HTML document is not actually part of HTML. It is the document type declaration (doctype). The declaration is formed as follows:

- Any case-insensitive match for the string "<!DOCTYPE"
- One or more space characters.
- Any case-insensitive match for the string "HTML".
- Optionally, one or more space characters.
- A ">" character.

However, while "<!DocTYpE htmL >" would be entirely valid, the preferred form is that given in the recommended template above.

Tags

Tags mark the start and end of markup elements. For some elements they are optional. Void elements only have a start tag and if it is a self-closing start tag then the closing tag is not allowed. For example:

- A paragraph element can have a start and end tag * … * or it can just have a start tag * …*
- The *<body>* element may be omitted entirely (but please don't)
- An *<img>* tag is void and cannot have an end tag.

All tags begin with a "<" character and end with a ">" character and have a tag name. A start tag may also contain attributes and, if the element is self-closing, it may contain a "/" character before the ">". An end tag will have the "/" character immediately following the "<" character.

Attributes

Attributes appear in the start tag only. They consist of a name and a value (in some cases the value is optional.

Name and value are separated by a "=" character. The value can optionally be enclosed by quotation marks " or apostrophes '.

A start tag can contain multiple attributes but they must be separated by a space character.

Examples:

- *disabled* is an empty attribute (it does not require a value)
- *type="checkbox"*, *type='checkbox'* or *type=checkbox* are all valid

Normal HTML Elements

All elements follow a set of syntactic rules:

- Elements must be nested correctly (elements cannot overlap)
- An HTML document can only have one root element *<html>*, all other elements are contained within the root element.

Parent, child and sibling

Each element within the mark-up can be described as being a child, parent or sibling of other elements in the document.

For example the *<head>* element contains the *<meta>* element and the *<title>* element (it can contain others that are not shown here). Both of these elements can be described as child elements of *<head>*. They can also be described as sibling elements of each other. *<head>*, in turn, can be described as a parent element of *<meta>* and *<title>*. *<head>* can also be described as a child element of *<html>*.

The relationships between elements (amongst other things) are defined by the rules of the mark-up language. Each mark-up will have differences in how the mark-up may be formed. The *DOCTYPE*, or document type definition, indicates to the browser which set of rules applies.

html, head and body

At its highest levels an HTML document is comprised of the *<html>* element. The *<html>* element contains one *<head>* element and one *<body>* element as immediate children.

The *<head>* element contains information about the document that will generally not be displayed to the user. This commonly contains the document title, meta information, scripts, information on styling and linked pages.

The *<body>* element contains all the content that will be presented to the user. For example, in a browser, this element contains everything that will be displayed in the browser window.

Block and Inline Elements

There are two types of elements in HTML, —block level elements and inline elements.

Block level elements contain content that needs to be treated separately from the content around it; although there may be varying levels of separation represented by the block level element. You could think of paragraphs which, though linked, are still separate blocks of information or a heading which has a slightly greater degree of separation from its surrounding content. Typically, in a browser, a block level element will force a new line.

Inline elements are contained within block level elements. They mark-up content that is part of their surrounding content but should be treated differently. Hyperlinks are a good example of an inline element where the link appears within a body of text. You could also consider highlighted text **. Typically, in a browser, an inline element will not cause a new line to appear.

Common HTML Body Elements

There are several other elements (tags) that can be added to the body of an HTML document. Some of the more common ones are:

Tag	Example	Content marked up
	`<p>This is my first HTML page </p>` `<p>and this is a new paragraph</p>`	Paragraph

(can be any number 1-6)
	`<h1>This is a heading</h1>`
	`<h2>This is a heading</h2>`
	`<h3>This is a heading</h3>`
		Headings

<a>	`<a href="http://www.bbc.co.uk">This is a link</a>`	Hyperlink. The href (the attribute) is the destination of the link.
<img>	`<img src="url" alt="some_text"/>`	Image Tag. The src attribute is the location of the image. The alt attribute is the alternative text which is used by screen readers.

<table>, <tr>, <td>	`<table>`
	`<tr>`
	`<th>Header 1</th>`
	`<th>Header 2</th>`
	`</tr>`
	`<tr>`
	`<td>row 1, cell 1</td>`
	`<td>row 1, cell 2</td>`
	`</tr>`
	`<tr>`
	`<td>row 2, cell 1</td>`
	`<td>row 2, cell 2</td>`
		These are the tags for a table. The <tr> element starts a new row. The <td> element is cells within the row, and makes up the columns. The <th> is a header cell. The example will show
		Header 1 | **Header 2**

Tag	Example	Content marked up
	`</tr>` `</table>`	row 1, cell 1 | row 1, cell 2 row 2, cell 1 | row 2, cell 2

,	`<ul>`
	`<li>Coffee</li>`
	`<li>Milk</li>`
	`</ul>`
		This is an un-ordered list. The starts a new list. The is a new list item. This is also known as bullet points, so the example shown will show:
		• Coffee
		• Milk

,	`<ol>`
	`<li>Coffee</li>`
	`<li>Milk</li>`
	`</ol>`
		This is an ordered list. The starts a new list. The is a new list item. This is also known as numbered lists, so the example shown will show:
		1. Coffee
		2. Milk

<dl>, <dt>, <dd>	`<dl>`
	`<dt>Coffee</dt>`
	`<dd>- black hot drink</dd>`
	`<dt>Milk</dt>`
	`<dd>- white cold drink</dd>`
	`</dl>`
		This is a definition list. The example will show
		Coffee
		- black hot drink
		Milk
		- white cold drink

	`<div>`
	`  <h3>This is a header</h3>`
	`  <p>This is a paragraph.</p>`
	`</div>`
		This is a section or division in a XHTML document. is often used to format sections with style

	`<p>My mother has <span class="blue">light blue</span> eyes.</p>`	The tag also describes a section. However it provides no visual change by itself.(i.e. does not force a new block)

Tag	Example	Content marked up
<header>	<header> <img src="logo.jpg" alt="logo"> Welcome to the company </header>	The <header> element dictates that this is the header content of a section.
<nav>	<nav> Navigation 1 2 3 </nav>	The <nav> element represents a navigation section of a page that links to other pages or to parts within the page.
<section>	<section> Granny Smith These juicy, green apples make a great filling for apple delicious pies. </section>	The *<section>* element groups thematically related content together – think of an introduction, main body and conclusion as separate sections, or chapters in a book, or numbered sections of a report
<article>	<article> This information would make sense without the rest of the document. </article>	The *<article>* element represents self-contained content so that it could be understood independently of the main page content, such as a post, magazine or newspaper article, a blog entry.
<aside>	<aside> something unrelated to the main content	The *<aside>* represents content that is related to the main content but which could

Tag	Example	Content marked up
	`</aside>`	be considered separate from that content.

<footer>	`<footer>`
	`<p>Copyright info</p>`
	`</footer>`
		A *<footer>* contains information similar to footnotes in a book.

Form Elements

XHTML forms are used to pass data to a server. A form can contain input elements including text boxes, check boxes, radio buttons, drop down lists, and submit buttons.

Tag	Example	Meaning
<form>	`<form id="input" action="html_form.php" method="get">` `</form>`	This is the main tag of a XHTML form. The id attribute specifies the unique name for the form. The action attribute specifies where to send the form data when the form is submitted. The method attribute specifies how that data is sent. Common methods are 'get' and 'post'

<input type="text" >	`<form action="test.html" method="get">`
	`<p>First name:`
	`<input type="text" id="firstname" />`
	`</p>`
	`<p>Last name:`
	`<input type="text" id="lastname" />`
	`</p>`
	`</form>`
		This is a text box. This is shown as follows
		First name:
		Last name:

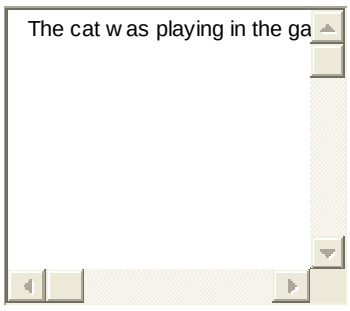
<input type="password" >	`<form action="test.html" method="get">`	This is a password box. It looks the same as a text box, but

Tag	Example	Meaning
	 Password: <input type="password" id="pword" /> </form>	when you type in it you can't see what you typed.
<input type="radio" >	<form action="test.html" method="get"> <input type="radio" id="rbutton" value="Oranges" /> Oranges <input type="radio" id="rbutton" value="Apples" /> Apples </form>	Radio Button. This appears like this. ○ Oranges ○ Apples
<input type="checkbox">	<form action="test.html" method="get"> <input type="checkbox" id="rbutton" value="Oranges" /> Oranges <input type="checkbox" id="rbutton" value="Apples" /> Apples 	Check Box. This appears like this ☐ Oranges ☐ Apples

Tag	Example	Meaning
	`</form>`	

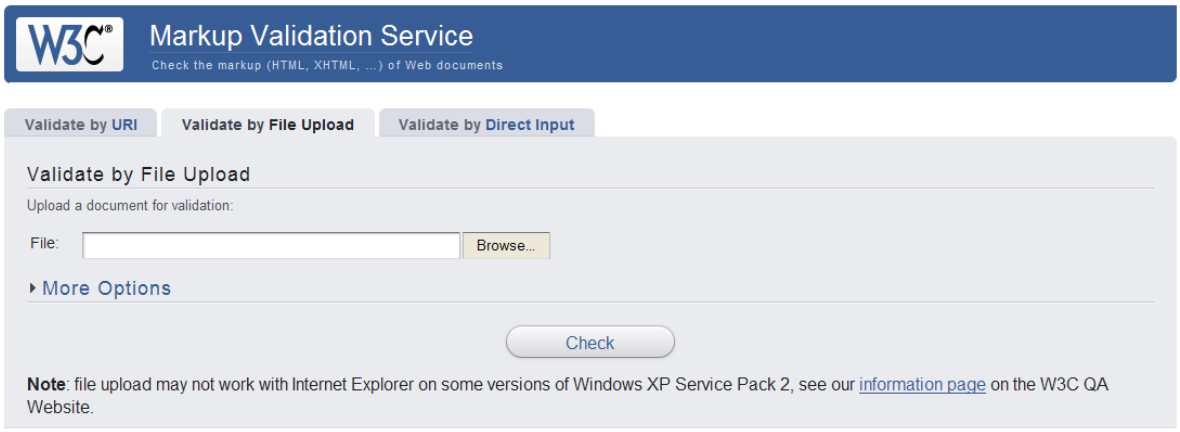
<input type=”submit” >	`<form action="test.html" method="get">`
	`<p>First name:`
	`<input type="text" id="firstname" />`
	`<input type="submit" value="Submit this form" />`
	`</p>`
	`</form>`
		This is the button that submits the form to the destination of the action in the form. The value is the text on the button. This will look like this 

<select>, <option>	`<select id="cars">`
	`<option value="1">Mini</option>`
	`<option value="2">Fiat</option>`
	`<option value="3">Ford</option>`
	`<option value="4">Audi</option>`
	`</select>`
		This is a drop down list. The <select> starts a new list. The <option> is choices in the list.

<textarea>	`<textarea rows="10" cols="30">`
	`The cat was playing in the garden`
	`</textarea>`
		This is multi-lined text area. This would appear like this 

Validating Your Documents

A validating tool, such as the W3C validator, will read the DTD in order to apply an appropriate set of rules when validating a web page. Documents should be validated regularly while being developed. The validator supplied by the World Wide Web Consortium (W3C) can be accessed at http://validator.w3.org/.



Local files can be validated by file upload or by direct input if the mark-up is copied and pasted into the tool directly.