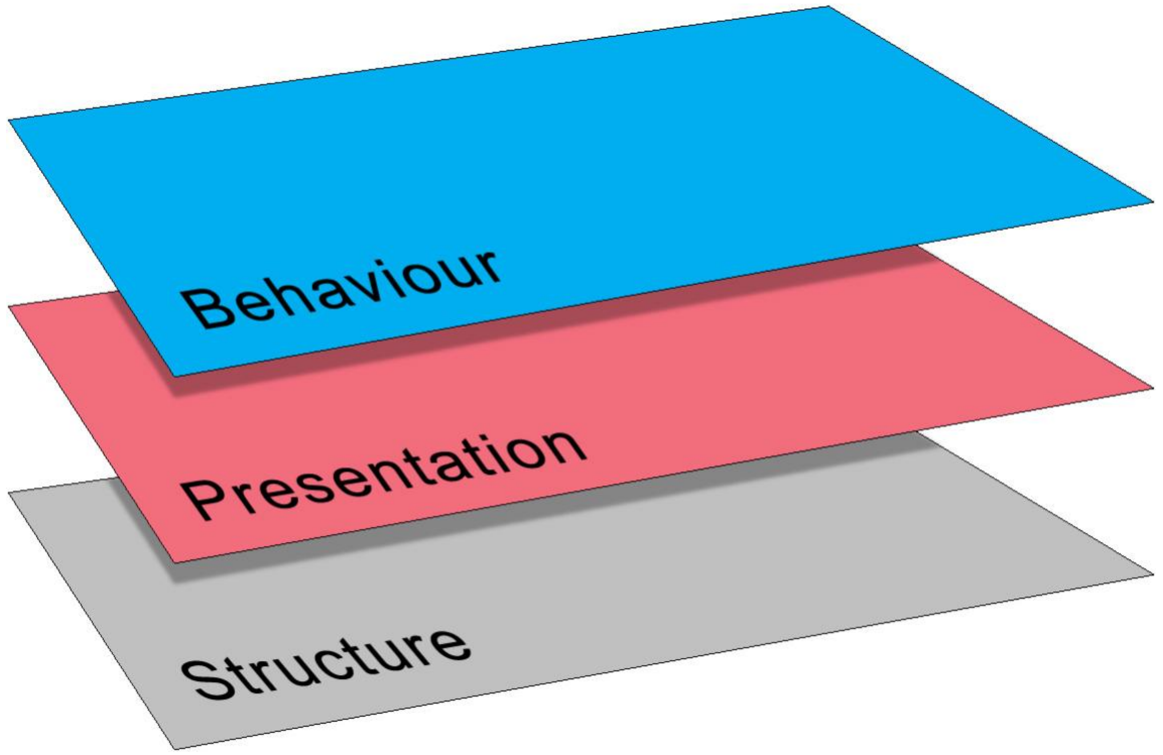
The Structural Layer (Hypertext Markup Language)

Webpage Design

Anatomy of a webpage

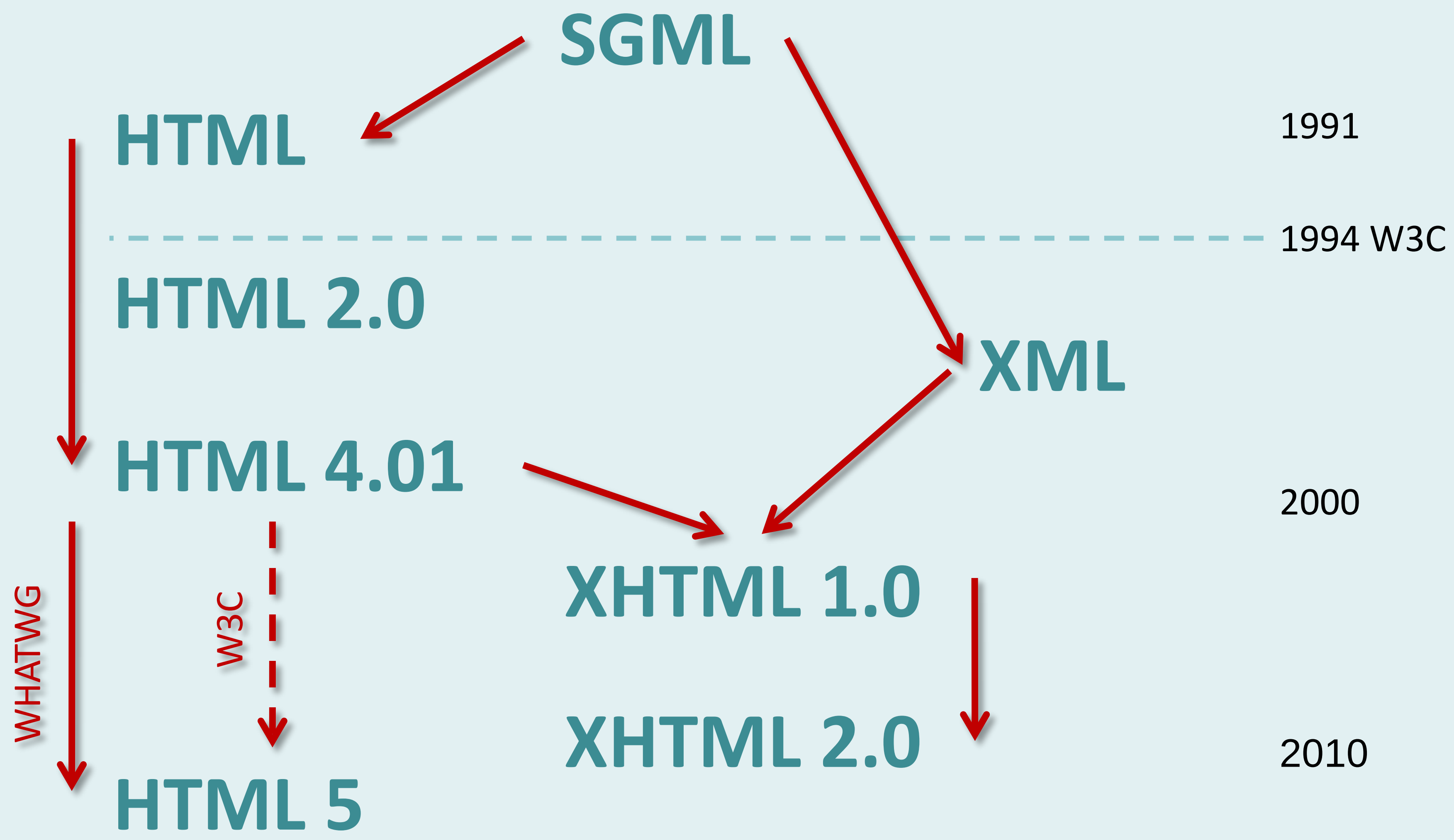
The structure of a webpage can be broken down into 3 layers. This is sometimes referred to as the Web Standards Model. The aim is to separate structure, presentation and behaviour. Each layer is defined by a different technology or language; HTML for structure, CSS for presentation and JavaScript for behaviour.



Document structure is defined using a *markup language* such as HTML

The Web Standards Model

A *very* brief history of markup



Standard Generalized Markup Language
HyperText Markup Language
eXtensible Markup Language
eXtensible HyperText Markup Language

World Wide Web Consortium
Web Hypertext Application Technology Working Group

A brief history of markup

A simple question

```
<p>What is the
<strong>purpose</strong>
of markup?</p>
```

What is the **purpose** of markup?

Answer

```
<p>To add structure
to documents and
<em>meaning</em>
to content.</p>
```

To add structure to documents and *meaning* to content.

What are tags for?

```
<p>content</p>
```

Tags are used to describe the type of content they contain. They are **not** used to describe how the content should look.

Another question

<p>If HTML5 is the current version of markup, why are we learning XHTML 1.0?</p>

If HTML 5 is the current version of markup, why are we learning XHTML 1.0?

Answer

```
<p>Because current best practice is to use HTML5 elements with XHTML syntax.</p>
```

Because current best practice is to use HTML5 elements with XHTML syntax.

XHTML has a strict syntax, HTML5 does not.

My Preferred Syntax Style for HTML5 Markup

XHTML*

Elements, attributes and values

* eXtensible Hyper-Text Markup Language

Introduction to XHTML

Elements

```
<p>This is a paragraph. It
contains some text.</p>
```

```
<img src="bird.jpg" alt="eagle" />
```

Most XHTML elements are defined by opening and closing tag pairs; a few (like the image element) are defined using a single, self-closing tag.

Note: there is a difference in the way XHTML and HTML5 treat self-closing elements (such as images). In XHTML, the closing slash character (/) is mandatory but in HTML 5 it is optional.

Attributes and values

```
<img src="bird.jpg" width="25" height="25" alt="Eagle" />
```

Attributes give the browser more information about an element. The attributes in the example above are telling the browser how big the image is and where to find it. Like many elements, the image has a range of valid attributes (e.g. width and height). Each attribute must have a quoted value in the form: attribute="value".

In XHTML, some attributes are mandatory. For example, all image elements must have an alt attribute (text alternative) for accessibility and also a src (source) so the browser knows where to find the image file . Some elements are discretionary (height and width).

Structure and relationships

(talking the right language)

Block-level elements

```
<p>This is a paragraph. It
contains some
<strong>important</strong> text.
The paragraph is a block-level
element.</p>
```

The paragraph is a *block-level* element. Effectively, this means that it begins and ends with a line-break and forms a distinctive “block” of content within the page.

Inline elements

```
<p>This is a paragraph. It
contains some
<strong>important</strong> text.
The important text is an inline
element.</p>
```

<strong> is an *inline* element. It does not begin and end with a line-break, it runs inline with any surrounding text. Images are inline elements.

Parent elements

```
<p>This is a paragraph. It
contains some
<strong>important</strong> text.
The paragraph is the parent of the
bold text.</p>
```

The paragraph is said to be the *parent* of the strong element because it is “nested” inside the paragraph.

Child elements

```
<p>This is a paragraph. It
contains some
<strong>important</strong> text.
The bold text is a child of the
paragraph element.</p>
```

The strong element is said to be the *child* of the paragraph.

Sibling elements

```
<ul>
   <li>This is a list item.</li>
   <li>This is an adjacent sibling.</li>
</ul>
```

The markup above defines an unordered list <ul>. Each list item <li> is the child of the unordered list. Each list item is also the *sibling* of the other list items contained within the same unordered list. The unordered list is the parent of all the list items it contains.

Semantic markup

(the important, controversial stuff)

Incorrect use of markup

```
<h1>Page Heading</h1>

<p>Some introductory text</p>

<p><big>A sub-heading</big></p>

<p>This is a paragraph. It contains some
<strong>important</strong> text and
content relating to the sub-heading
above.</p>
```

There are two things wrong with the markup above. A sub-heading should not be marked up as a paragraph because it is **not** a paragraph, it is a heading. Although <big> is not deprecated in XHTML, it should be avoided because it is used to control presentation (i.e. it makes text bigger).

Semantically correct markup

```
<h1>Page Heading</h1>

<p>Some introductory text</p>

<h2>A sub-heading</h2>

<p>This is a paragraph. It contains some
<strong>important</strong> text and
content relating to the sub-heading
above.</p>
```

The markup above is correct – each tag is used to describe the content it contains.

Why is this important?

```
<h2>A sub-heading</h2>

<p><big>A sub-heading</big></p>
```

These two lines of HTML may *look* the same when rendered in a browser but they have very different meanings. Remember, the purpose of HTML is to add *meaning* to our content, not to control the way it looks (the presentation). That job is for CSS, not HTML.

When designing websites, we need to be aware that our content is often read by non-visual agents (e.g. bots and screen readers). So the way our content looks is not the best way to convey meaning.

Although presentational elements such as <big> are still part of the XHTML specification, they have been removed from HTML5.

Incorrect use of markup

```
<h2>This is a list of colours</h2>

<p>Red</p>

<p>Green</p>

<p>Blue</p>
```

The markup above is non-semantic. The content is clearly a list but it is marked up as three paragraphs. Visually, it may look like a list, but it is not.

Semantically correct markup

```
<h2>This is a list of colours</h2>
<ul>
  <li>Red</li>
  <li>Green</li>
  <li>Blue</li>
</ul>
```

Correctly marked up as an unordered list. Although this code is more verbose, we have added a lot more meaning to the content and we have made relationships between the different elements (parent, child etc.).

Why is this important?

```
<ul>
   <li>Red</li>
   <li>Green</li>
   <li>Blue</li>
</ul>
```

In addition to telling non-sighted agents that this is a list, making the correct relationships between different elements will help us a great deal when we come to style those elements with CSS. For example, we can easily select all the child elements of a specific list and style them consistently.

Hypertext links

Making links

```
<a href="file.html">Text Link</a>
```

The opening and closing anchor tags are used to form a hyperlink. Any content between the tags will be clickable. The anchor element has one mandatory attribute, href (hypertext reference) which points to the linked file.

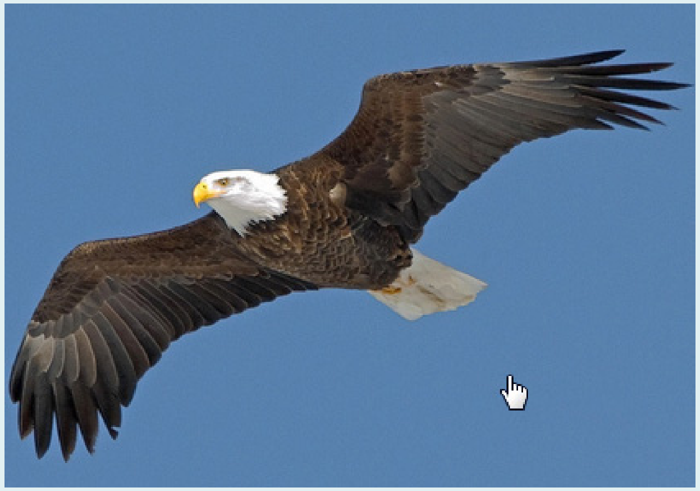
```
<a href="file.html" target="_blank">Text Link</a>
```

Using the target attribute is a usability issue and considered bad practice. In general, we should allow the user control over link actions.

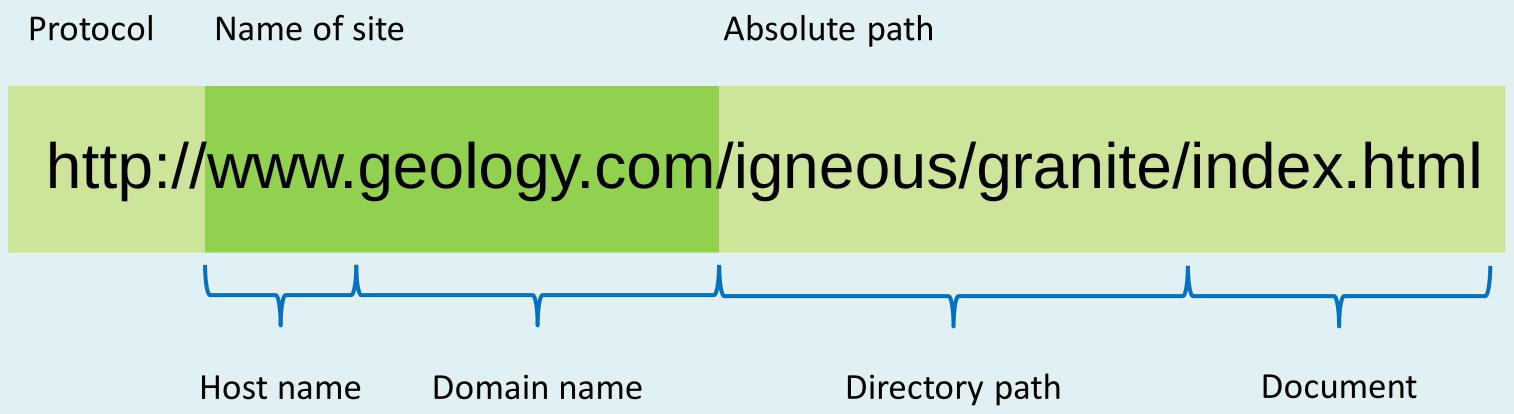
Making image links

```
<a href="file.html"><img src="bird.jpg" alt="Eagle" /></a>
```

Image links are made in exactly the same way as text links. In the example above, the whole image would be clickable.



Universal resource locator (URL)



A URL is a sub-class of a URI (Universal Resource Identifier), just as the Web could be said to be a sub-class of the Internet. The two should not be confused – they are not the same thing.

The protocol is the HyperText Transport Protocol. You may also see https where the "s" stands for *secure* and is common on sites where financial transactions take place.

Uniform resource identifier

Absolute links

```
<a href="http://www.mysite/design/file.html">Text Link</a>
```

Absolute links are always used for links to *external* pages (i.e. those on other websites). The absolute path **must** begin with “http”. They will also work for local files but are not usually used because they are verbose and not portable – there’s a neater way to make local links...

Relative links

```
<a href="../design/file.html">Text Link</a>
```

Relative links are used for links to local pages (i.e. those on the same website). In the example above, the two little dots mean “go up one level in the folder structure”. From there the link is to a file called *file.html* in the folder called *design*. Relative links are useful because they will work when developing sites locally (on a PC or Mac) **and** when they are uploaded to the server.

Example relative links

```
<a href="filename.html">Text Link</a>
```

A link to a file in the same folder.

```
<a href="folder/file.html">Text Link</a>
```

A link to a file in a folder one level below.

```
<a href="../file.html">Text Link</a>
```

A link to a file in a folder one level above.

```
<a href="../folder/file.html">Text Link</a>
```

A link to a file in another folder at the same level.

Relative to document root

```
<a href="/design/file.html">Text Link</a>
```

The *document root* is a special folder on a web server. Links made relative to the document root, characterised by a leading slash character (/), are very useful for dynamic websites because they work no matter where the link is placed. However, they **do not work on your local computer** and are therefore difficult to check until the website is uploaded.

As a beginner, you should stick with *absolute* paths for links to external files and *relative* paths for links to internal files. This means you can check that all your links are working before uploading your site to the server.

Special characters

&

Character escaping

Some characters that have special meaning in (X)HTML **must** be *escaped* in order to display correctly as content to and validate.
For example "<" and ">" will be interpreted as the start and end of HTML tags unless they are escaped.

< = &lt;	less than
> = &gt;	greater than
& = &amp;	ampersand
" = &quot;	plain quote

```
<p>MA Web Design &amp; Content Planning</p>
```

Character escaping

Alternatively, special characters can also be encoded using their decimal ASCII numbers:

< = &#60;	less than
> = &#62;	greater than
& = &#38;	ampersand
" = &#34;	plain quote

[HTML Codes – Characters and Symbols]

Character escaping

One of the benefits of this is that we can display characters that do not exist on our keyboard. For example:

```
<p>&copy; David Watson</p>

<p>&#169; David Watson</p>
```

Both of the elements above would render as:

© David Watson

Common structural elements

Headings

```
<h1>Heading level 1</h1>

<h2>Heading level 2</h2>

<h3>Heading level 3</h3>
```

=

Heading level 1

Heading level 2

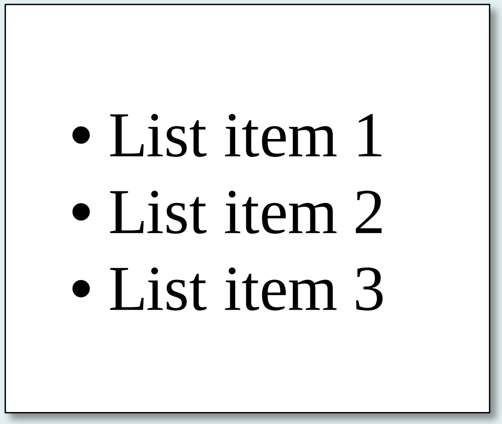
Heading level 3

Headings are used to define hierarchy within a document, **not** to specify the text height.

The unordered list

```
<ul>
    <li>List item 1</li>
    <li>List item 2</li>
    <li>List item 3</li>
</ul>
```

=



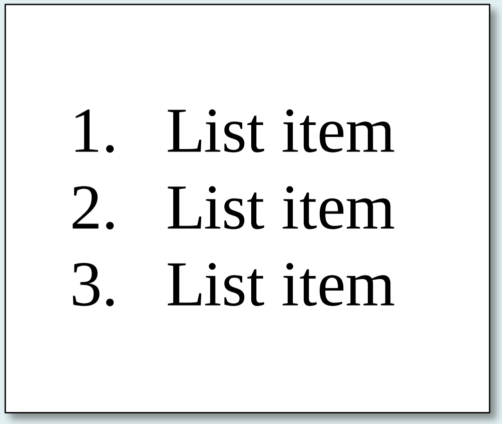
- List item 1
- List item 2
- List item 3

Unordered lists are very useful structural elements and are commonly used to define navigation on a web page (amongst other things).

The ordered list

```
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ol>
```

=

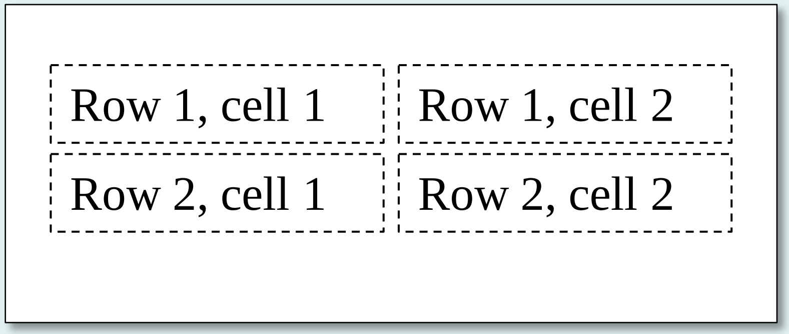

1. List item
2. List item
3. List item

Ordered lists have a fairly specific use case and are used where the list items have a specific order. The definition list is a third list type in HTML, it also has a specific use case.

Tables

```
<table>
  <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>
  </tr>
</table>
```

=



Row 1, cell 1	Row 1, cell 2
Row 2, cell 1	Row 2, cell 2

The poor, old table has been badly misused in the past but these days it is correctly used only for its specific purpose i.e. as a container for tabular data.

Structure of a webpage

(X)HTML file structure

```
<html>
<head>
        <title>Webpage Design</title>
</head>
<body>
        <p>My first web page!</p>
</body>
</html>
```

Basic structure of a webpage

(X)HTML file structure

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en">

<head>

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

<title>Webpage Design</title>

</head>

<body>...
```

Although the basic file structure is simple, we must add some gobbledygook in order to make it work properly. You’re not expected to remember this stuff – just copy it from a reliable source.

http://www.w3.org/TR/html401/struct/global.html

XHTML DOCTYPE

- XHTML 1.0 Transitional

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

- XHTML 1.0 Frameset

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

- XHTML 1.0 Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

On this programme we will only ever use the 1.0 Strict doctype for XHTML

Doctypes

XHTML rules & good practice

- All documents must have a document declaration (doctype)
- All documents must be properly formed
- All tags must be in lower case
- Deprecated tags (e.g. <font>) are excluded
- All tags must be closed (even standalone tags) <br />
- All attributes must have explicit quoted values width="25"
- The id attribute should be used in place of name
- All tags must be properly nested

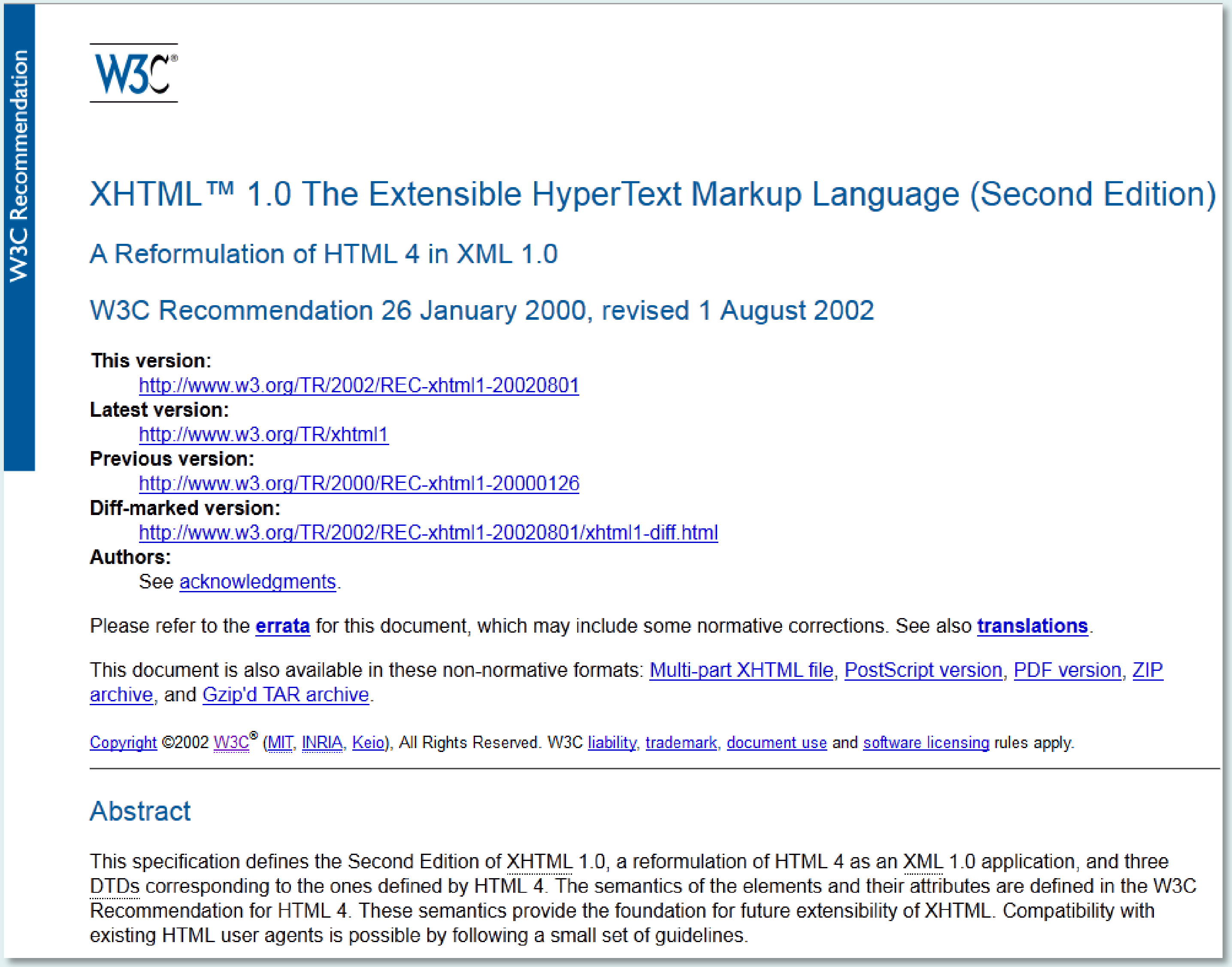
http://www.devguru.com/Technologies/xhtml/quickref/xhtml_coding_rules.html

How HTML5 differs

```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Webpage Design</title>

</head>

<body>...
```

HTML5 is a much less verbose markup language. Much of the unnecessary baggage has been stripped away. Whereas in XHTML, every setting must be made explicitly, in HTML5, many settings assume a common default value. HTML5 does not have the strict syntax of XHTML but best practice is to use strict syntax anyway. Notice that the doctype doesn’t even include the number “5”. Why do you think that is?

Who controls XHTML?



W3C Recommendation

XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition)

A Reformulation of HTML 4 in XML 1.0

W3C Recommendation 26 January 2000, revised 1 August 2002

This version:
http://www.w3.org/TR/2002/REC-xhtml1-20020801
Latest version:
http://www.w3.org/TR/xhtml1
Previous version:
http://www.w3.org/TR/2000/REC-xhtml1-20000126
Diff-marked version:
http://www.w3.org/TR/2002/REC-xhtml1-20020801/xhtml1-diff.html
Authors:
See acknowledgments.

Please refer to the **errata** for this document, which may include some normative corrections. See also **translations**.

This document is also available in these non-normative formats: Multi-part XHTML file, PostScript version, PDF version, ZIP archive, and Gzip'd TAR archive.

Copyright ©2002 W3C® (MIT, INRIA, Keio), All Rights Reserved. W3C liability, trademark, document use and software licensing rules apply.

Abstract

This specification defines the Second Edition of XHTML 1.0, a reformulation of HTML 4 as an XML 1.0 application, and three DTDs corresponding to the ones defined by HTML 4. The semantics of the elements and their attributes are defined in the W3C Recommendation for HTML 4. These semantics provide the foundation for future extensibility of XHTML. Compatibility with existing HTML user agents is possible by following a small set of guidelines.

http://www.w3.org/TR/xhtml1/

Deprecated tags and attributes

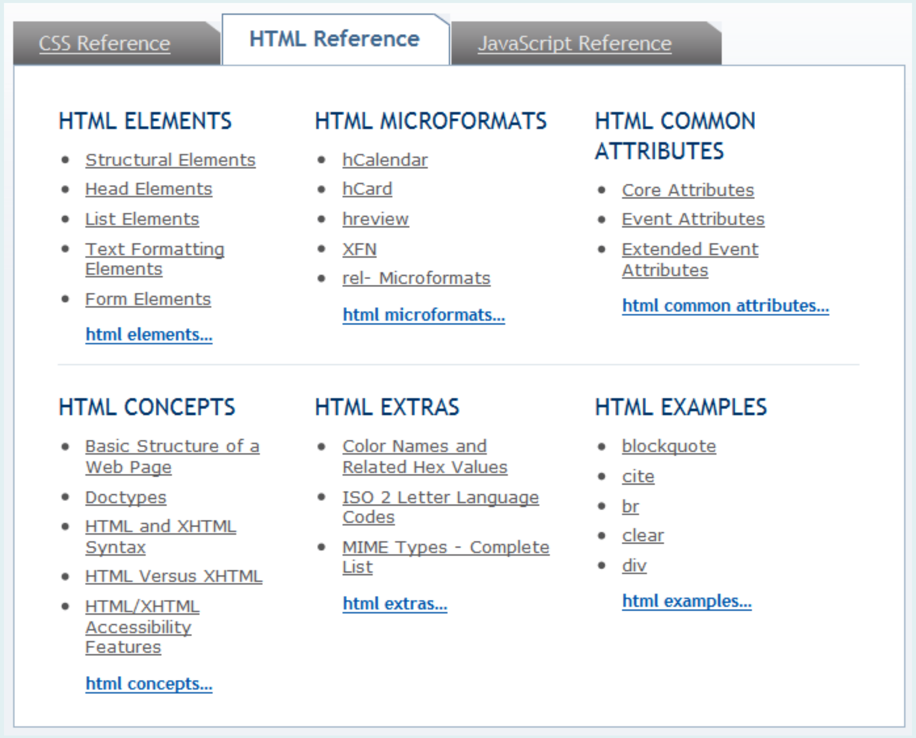
```
<p><font size="5">big text</font></p>
```

Deprecated tags are those that are considered obsolete in the current version of the markup language. However, they are still recognised by browsers and will be rendered accordingly, irrespective of the doctype you are using. The use of deprecated tags and/or attributes will invalidate your markup. You should not use deprecated tags.

In XHTML, most of the deprecated tags are those that were used in HTML4 to describe presentation. Examples include <font>, <center> and <u> (underline). Tags deprecated in XHTML are also deprecated in HTML5.

Deprecated elements

Which tags can I use?



CSS Reference | HTML Reference | JavaScript Reference

HTML ELEMENTS
- Structural Elements
- Head Elements
- List Elements
- Text Formatting Elements
- Form Elements

html elements...

HTML MICROFORMATS
- hCalendar
- hCard
- hreview
- XFN
- rel- Microformats

html microformats...

HTML COMMON ATTRIBUTES
- Core Attributes
- Event Attributes
- Extended Event Attributes

html common attributes...

HTML CONCEPTS
- Basic Structure of a Web Page
- Doctypes
- HTML and XHTML Syntax
- HTML Versus XHTML
- HTML/XHTML Accessibility Features

html concepts...

HTML EXTRAS
- Color Names and Related Hex Values
- ISO 2 Letter Language Codes
- MIME Types - Complete List

html extras...

HTML EXAMPLES
- blockquote
- cite
- br
- clear
- div

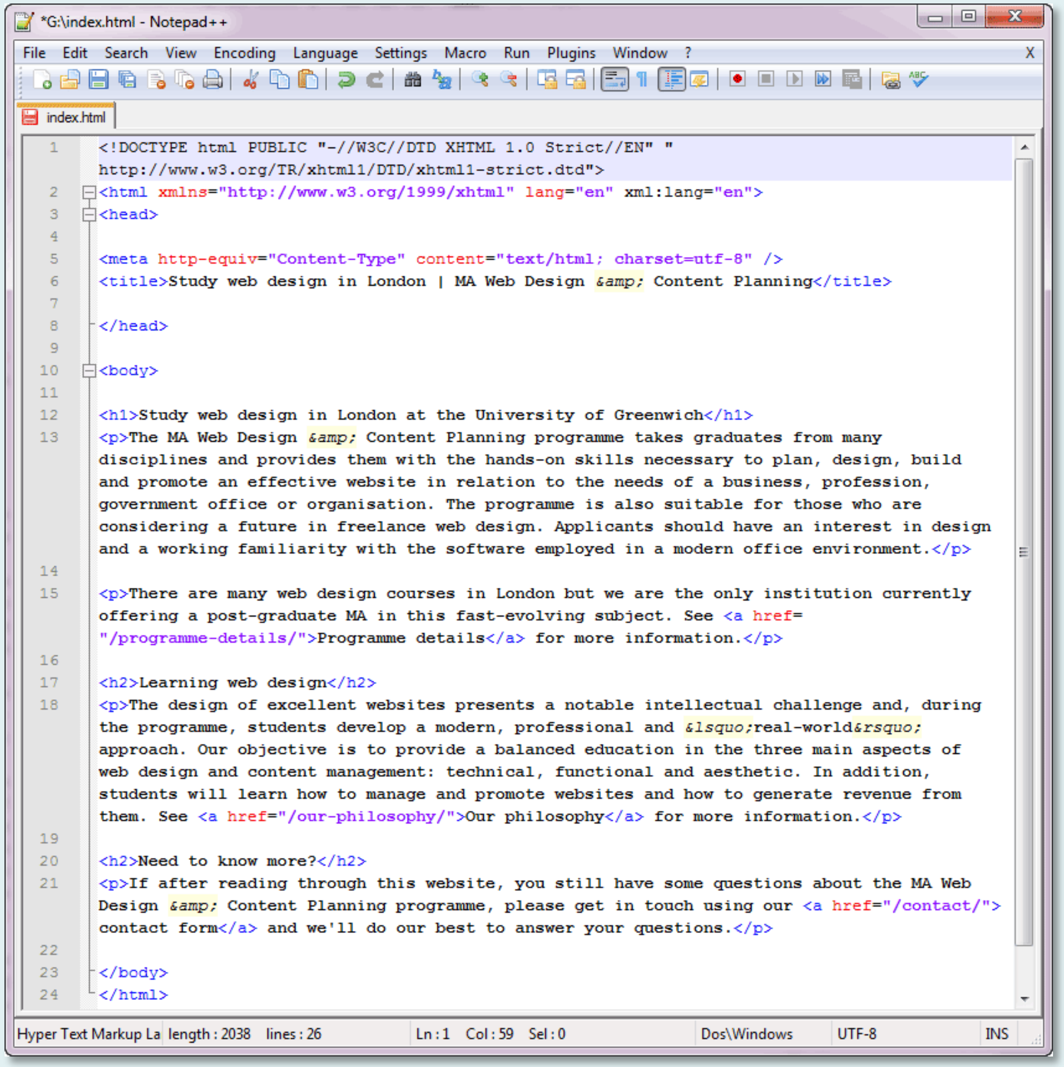
html examples...

Use a reference. The excellent online reference at Sitepoint is a good option. You can even use a book if you prefer that medium.

HTML Elements

Writing (X)HTML

- Use a simple text editor to write your markup
- Get one with code highlighting to make life easier
- Notepad++ or Sublime Text are good choices but any text editor will do


*G:\index.html - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
index.html
Hyper Text Markup La length : 2038 lines : 26 Ln : 1 Col : 59 Sel : 0 Dos\Windows UTF-8 INS

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "
http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Study web design in London | MA Web Design &amp; Content Planning</title>

</head>

<body>

<h1>Study web design in London at the University of Greenwich</h1>
<p>The MA Web Design &amp; Content Planning programme takes graduates from many
disciplines and provides them with the hands-on skills necessary to plan, design, build
and promote an effective website in relation to the needs of a business, profession,
government office or organisation. The programme is also suitable for those who are
considering a future in freelance web design. Applicants should have an interest in design
and a working familiarity with the software employed in a modern office environment.</p>

<p>There are many web design courses in London but we are the only institution currently
offering a post-graduate MA in this fast-evolving subject. See <a href=
"/programme-details/">Programme details</a> for more information.</p>

<h2>Learning web design</h2>
<p>The design of excellent websites presents a notable intellectual challenge and, during
the programme, students develop a modern, professional and &lsquo;real-world&rsquo;
approach. Our objective is to provide a balanced education in the three main aspects of
web design and content management: technical, functional and aesthetic. In addition,
students will learn how to manage and promote websites and how to generate revenue from
them. See <a href="/our-philosophy/">Our philosophy</a> for more information.</p>

<h2>Need to know more?</h2>
<p>If after reading through this website, you still have some questions about the MA Web
Design &amp; Content Planning programme, please get in touch using our <a href="/contact/">
contact form</a> and we'll do our best to answer your questions.</p>

</body>
</html>
```

http://www.sublimetext.com/

http://notepad-plus-plus.org/

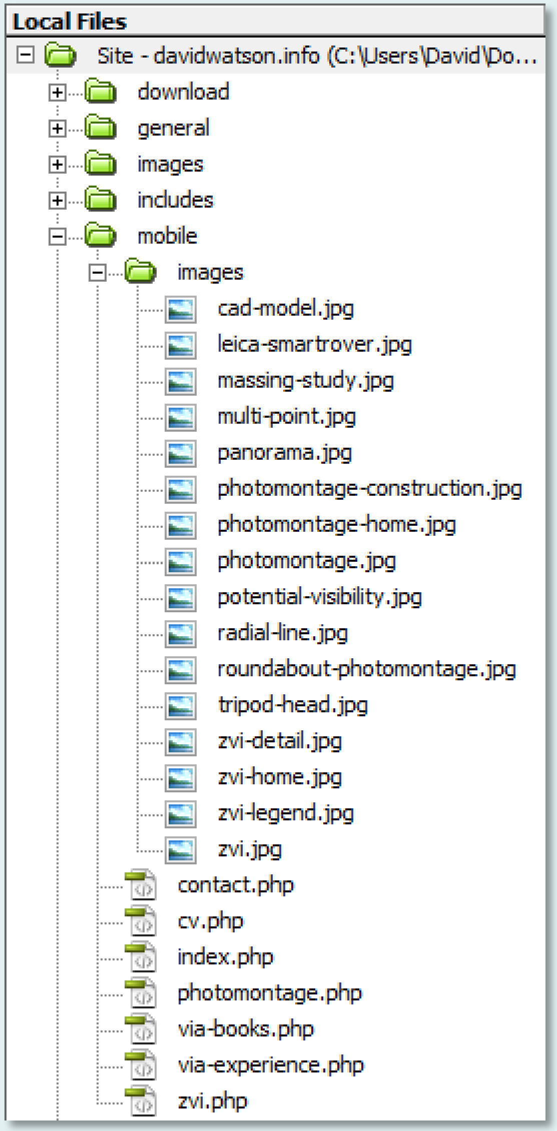
Comments in HTML

It's **always** a good idea to comment your markup in order to remind yourself (or to let someone else know) what your code is intended to do or to clarify the document structure.

```
<!-- start of main page content -->
<h2>This week's book review</h2>
<p>The Return of the Native by…</p>
```

File naming and folder structure

File naming



```
Local Files
⊟ Site - davidwatson.info (C:\Users\David\Do...
  ⊞ download
  ⊞ general
  ⊞ images
  ⊞ includes
  ⊟ mobile
    ⊟ images
        cad-model.jpg
        leica-smartrover.jpg
        massing-study.jpg
        multi-point.jpg
        panorama.jpg
        photomontage-construction.jpg
        photomontage-home.jpg
        photomontage.jpg
        potential-visibility.jpg
        radial-line.jpg
        roundabout-photomontage.jpg
        tripod-head.jpg
        zvi-detail.jpg
        zvi-home.jpg
        zvi-legend.jpg
        zvi.jpg
      contact.php
      cv.php
      index.php
      photomontage.php
      via-books.php
      via-experience.php
      zvi.php
```

- Files & folder names should all be in **lower case**
- Avoid using spaces or other none alphanumeric characters
- The hyphen* or underscore should be used if you need a separator
- Make sure file naming is descriptive and systematic
- HTML filenames should be descriptive (SEO)
- The homepage is always called “index.html”

*Has SEO implications (Google)

Hyphens vs. Underscores

It’s not a good idea to include spaces in URLs because they are automatically encoded by the browser (%20) and look odd to users.

my first web page.html

my%20first%20web%20page.html

You may use underscores but be aware that Google does not recognise the underscore as a separator and will interpret the name as a single search term. This may be a way to hide content.

my_first_web_page.html

Google interprets the hyphen (dash) as a separator, so the name below gives 4 search terms.

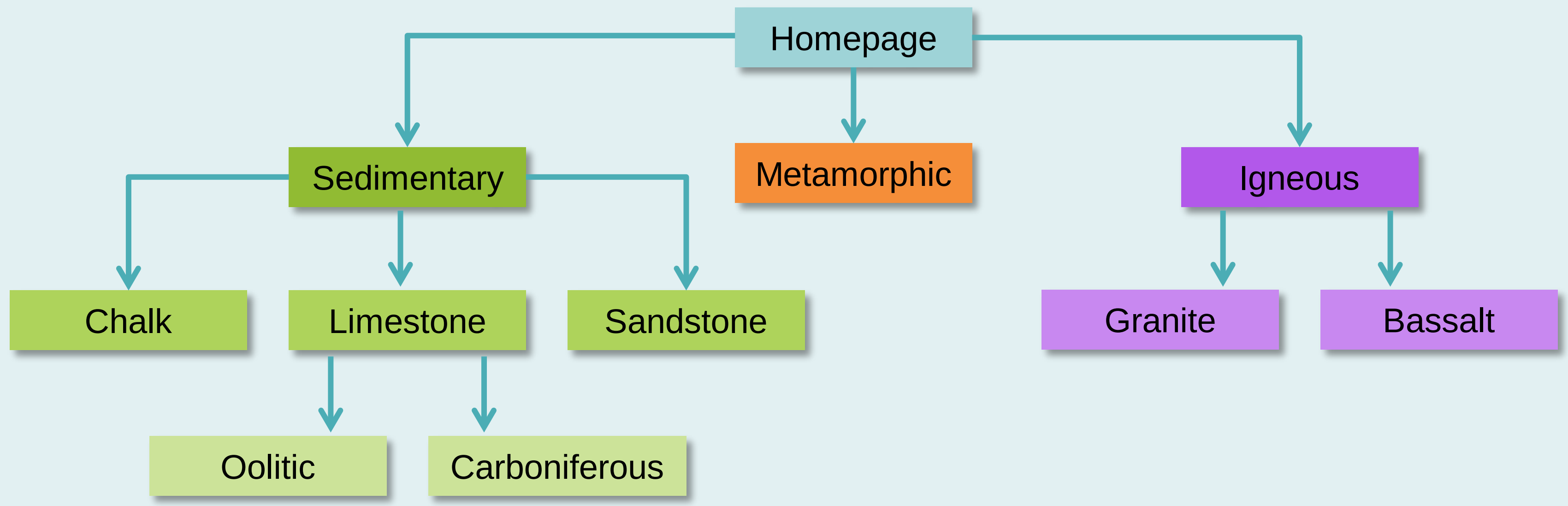
my-first-web-page.html

Underscores vs. dashes in URLs – Google Webmaster video   URL Encoding

Website structure

- Categorise and arrange your information (this is known as Information Architecture).
- Always design your website on paper first; a quick sketch at the start of the process can save time later.
- Most sites use a hierarchical structure for complex content but simple sites may be flat.
- Limit first order categories to 6 or fewer – remember Hick's Law.
- Limit structure depth to 4 unless there is a good reason – allow users to create mental maps.

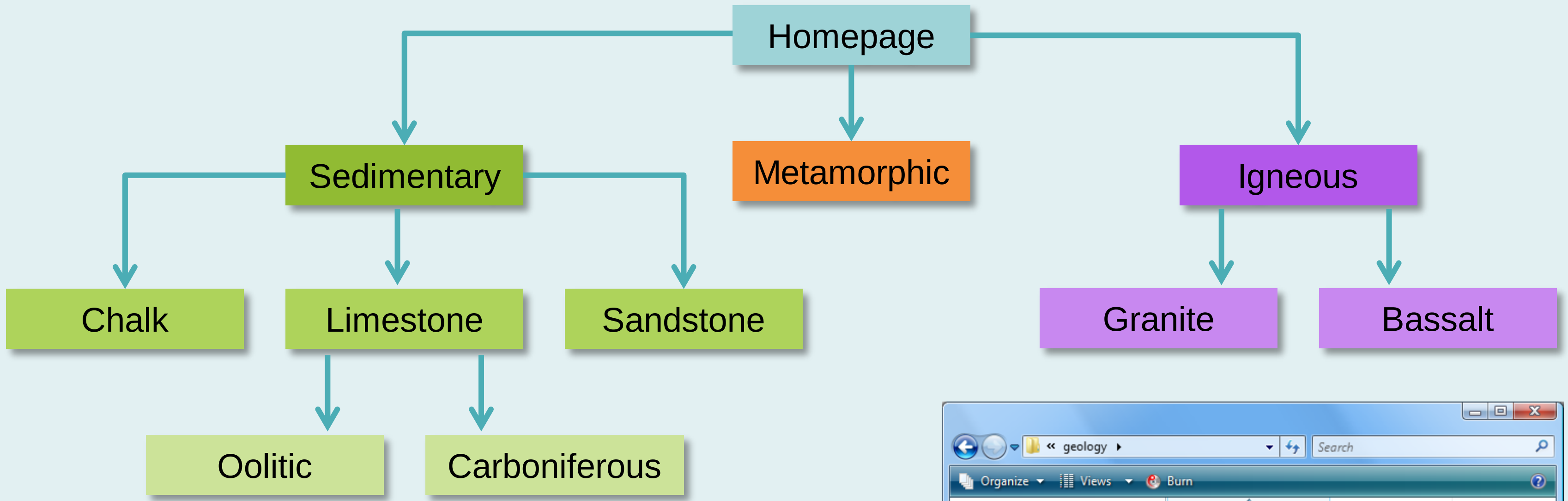
A website about rocks



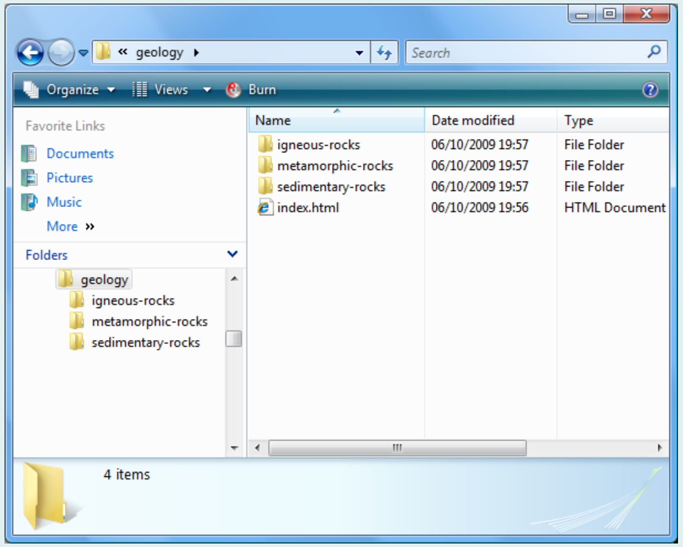
Folder structure

- The home page is always called “index” although the extension may vary.
- The index.html file is placed in the **root** folder (*public_html* on apache servers).
- Use a folder structure that mirrors site structure.
- Give your HTML files and folders descriptive names (for ease of use as well as SEO).
- Current trend is to use SEO names for folders and then “index” for the main file within that folder but this is personal choice.

A website about rocks



There are many different ways to organise the files and folders in a website like this but be logical and consistent in your approach.

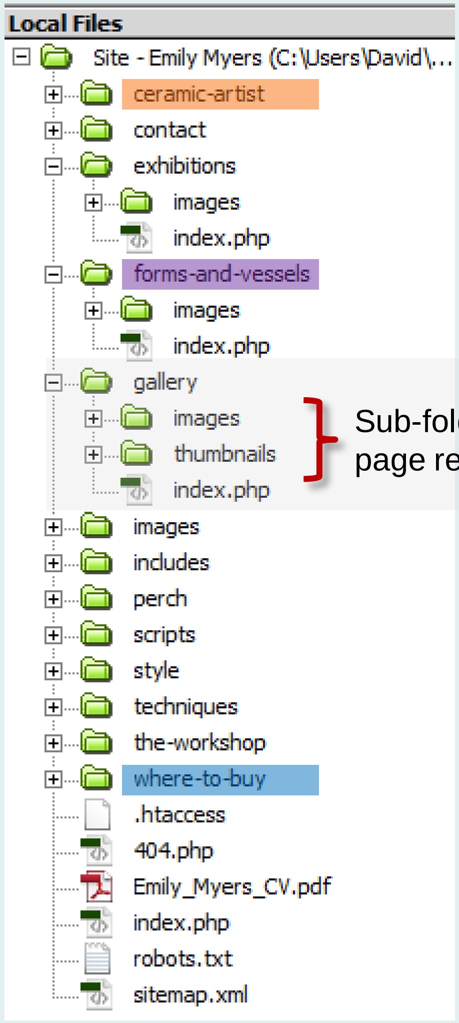


geology
- igneous-rocks
- metamorphic-rocks
- sedimentary-rocks

Name	Date modified	Type
igneous-rocks	06/10/2009 19:57	File Folder
metamorphic-rocks	06/10/2009 19:57	File Folder
sedimentary-rocks	06/10/2009 19:57	File Folder
index.html	06/10/2009 19:56	HTML Document

4 items

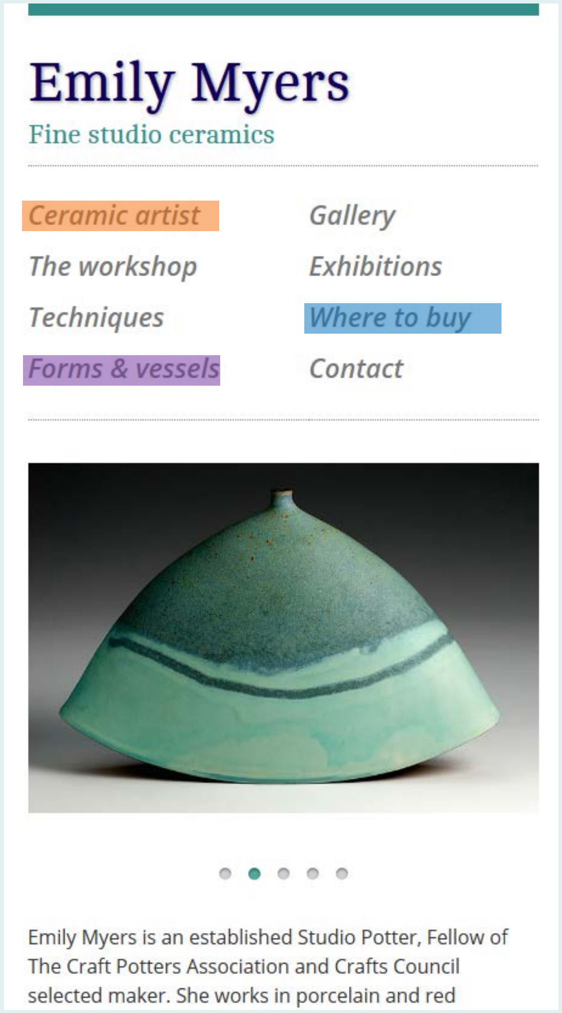
Organising your files



Local Files

- Site - Emily Myers (C:\Users\David\...
  - ceramic-artist
  - contact
  - exhibitions
    - images
    - index.php
  - forms-and-vessels
    - images
    - index.php
  - gallery
    - images
    - thumbnails
    - index.php
  - images
  - includes
  - perch
  - scripts
  - style
  - techniques
  - the-workshop
  - where-to-buy
  - .htaccess
  - 404.php
  - Emily_Myers_CV.pdf
  - index.php
  - robots.txt
  - sitemap.xml

Sub-folders for page resources



Emily Myers

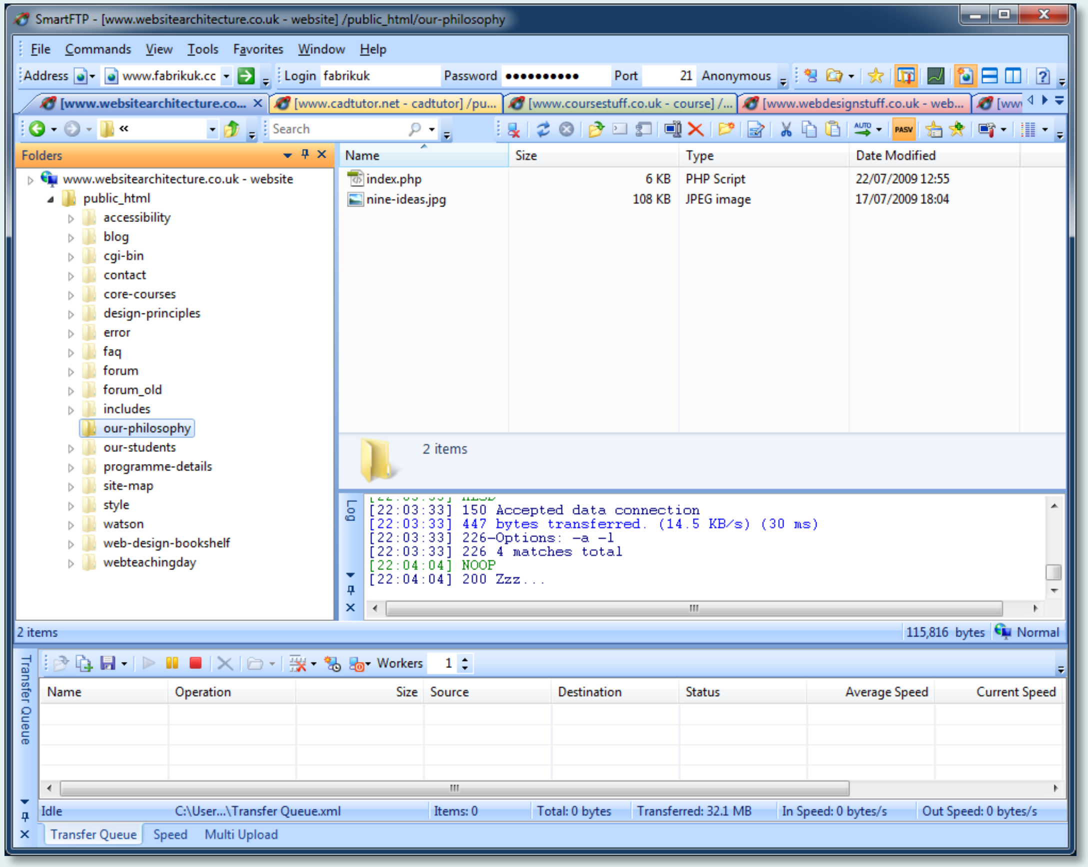
Fine studio ceramics

Ceramic artist	Gallery
The workshop	Exhibitions
Techniques	Where to buy
Forms & vessels	Contact

Emily Myers is an established Studio Potter, Fellow of The Craft Potters Association and Crafts Council selected maker. She works in porcelain and red

FTP Client

- Use a FTP client to upload files from your local computer to the web server
- Any FTP software will do; FileZilla is a good choice, as is Fire FTP, SmartFTP or Cyberduck
- *Always* maintain a backup copy of your website locally



SmartFTP - [www.websitearchitecture.co.uk - website] /public_html/our-philosophy

Address: www.fabrikuk.cc | Login: fabrikuk | Password: •••••••••• | Port: 21 | Anonymous

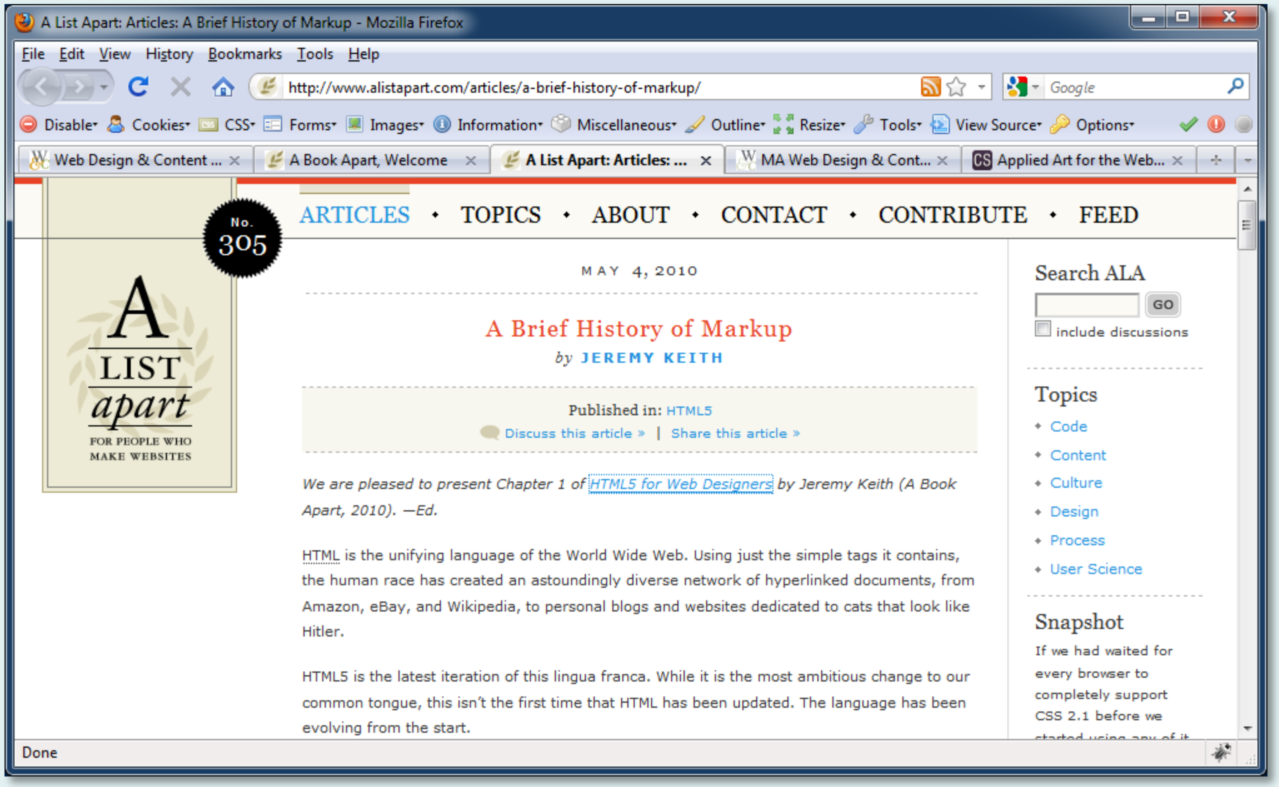
Name	Size	Type	Date Modified
index.php	6 KB	PHP Script	22/07/2009 12:55
nine-ideas.jpg	108 KB	JPEG image	17/07/2009 18:04

```
[22:03:33] 150 Accepted data connection
[22:03:33] 447 bytes transferred. (14.5 KB/s) (30 ms)
[22:03:33] 226-Options: -a -l
[22:03:33] 226 4 matches total
[22:04:04] NOOP
[22:04:04] 200 Zzz...
```

Idle | C:\User...\Transfer Queue.xml | Items: 0 | Total: 0 bytes | Transferred: 32.1 MB | In Speed: 0 bytes/s | Out Speed: 0 bytes/s

http://filezilla-project.org/ | http://www.smartftp.com/

Web Browser


A List Apart: Articles: A Brief History of Markup - Mozilla Firefox

File Edit View History Bookmarks Tools Help
http://www.alistapart.com/articles/a-brief-history-of-markup/
Google
Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options
Web Design & Content ... | A Book Apart, Welcome | A List Apart: Articles: ... | MA Web Design & Cont... | Applied Art for the Web...
ARTICLES • TOPICS • ABOUT • CONTACT • CONTRIBUTE • FEED
No. 305
A LIST apart FOR PEOPLE WHO MAKE WEBSITES

MAY 4, 2010

A Brief History of Markup

by JEREMY KEITH

Published in: HTML5

Discuss this article » | Share this article »

We are pleased to present Chapter 1 of HTML5 for Web Designers by Jeremy Keith (A Book Apart, 2010). —Ed.

HTML is the unifying language of the World Wide Web. Using just the simple tags it contains, the human race has created an astoundingly diverse network of hyperlinked documents, from Amazon, eBay, and Wikipedia, to personal blogs and websites dedicated to cats that look like Hitler.

HTML5 is the latest iteration of this lingua franca. While it is the most ambitious change to our common tongue, this isn't the first time that HTML has been updated. The language has been evolving from the start.

Search ALA

GO

include discussions

Topics

- Code
- Content
- Culture
- Design
- Process
- User Science

Snapshot

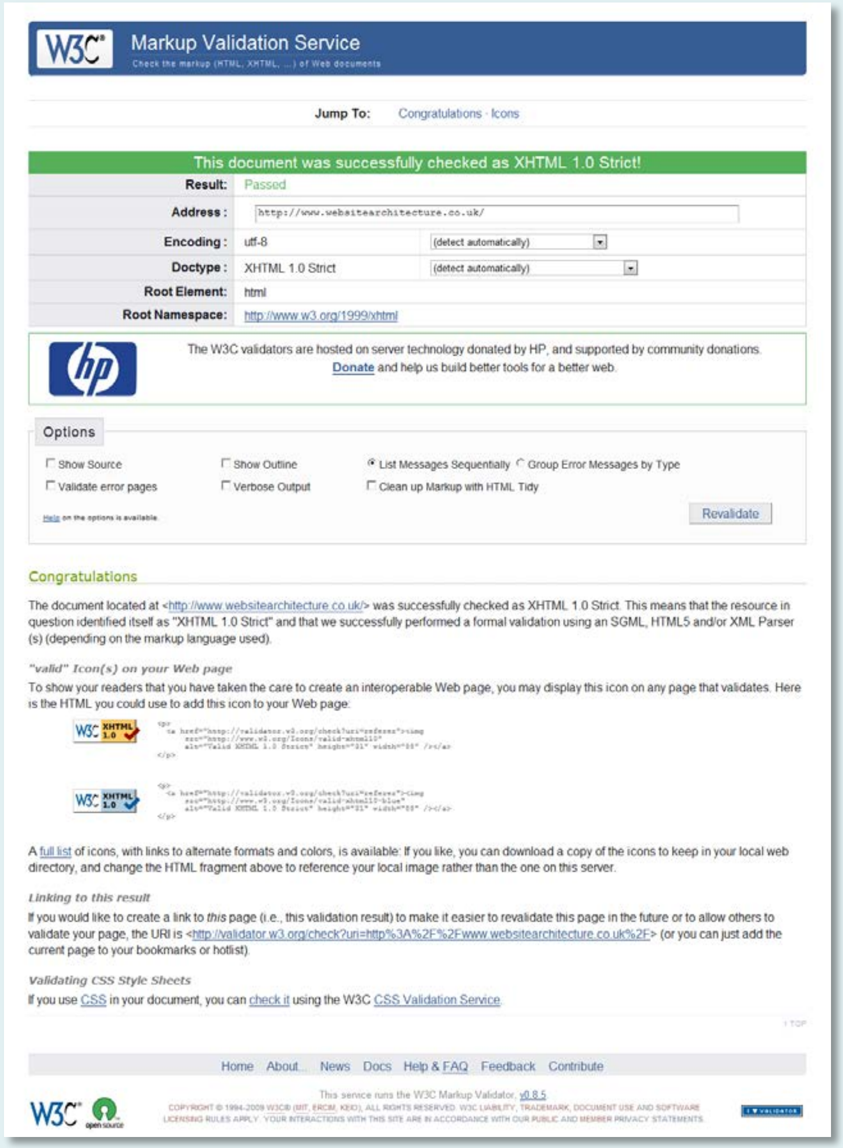
If we had waited for every browser to completely support CSS 2.1 before we

Done

- Check your webpage using a reliable (standards compliant) browser; Firefox is a good choice
- Consider adding the Firebug and Web Developer plug-ins for better functionality

Code Validation

- Always validate your XHTML using the W3C Markup Validation Service
- The service uses the doctype to determine what type of markup you are using
- Correct any errors in your markup



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Congratulations · Icons

This document was successfully checked as XHTML 1.0 Strict!

Result:	Passed	
Address :	http://www.websitearchitecture.co.uk/	
Encoding :	utf-8	(detect automatically)
Doctype :	XHTML 1.0 Strict	(detect automatically)
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

The W3C validators are hosted on server technology donated by HP, and supported by community donations. **Donate** and help us build better tools for a better web.

Options

☐ Show Source ☐ Show Outline ● List Messages Sequentially ○ Group Error Messages by Type

☐ Validate error pages ☐ Verbose Output ☐ Clean up Markup with HTML Tidy

Help on the options is available.

Revalidate

Congratulations

The document located at <http://www.websitearchitecture.co.uk/> was successfully checked as XHTML 1.0 Strict. This means that the resource in question identified itself as "XHTML 1.0 Strict" and that we successfully performed a formal validation using an SGML, HTML5 and/or XML Parser (s) (depending on the markup language used).

"valid" Icon(s) on your Web page

To show your readers that you have taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the HTML you could use to add this icon to your Web page:

```
<p>
  <a href="http://validator.w3.org/check?uri=referer"><img
      src="http://www.w3.org/Icons/valid-xhtml10"
      alt="Valid XHTML 1.0 Strict" height="31" width="88" /></a>
</p>
```

```
<p>
  <a href="http://validator.w3.org/check?uri=referer"><img
      src="http://www.w3.org/Icons/valid-xhtml10-blue"
      alt="Valid XHTML 1.0 Strict" height="31" width="88" /></a>
</p>
```

A full list of icons, with links to alternate formats and colors, is available: If you like, you can download a copy of the icons to keep in your local web directory, and change the HTML fragment above to reference your local image rather than the one on this server.

Linking to this result

If you would like to create a link to *this* page (i.e., this validation result) to make it easier to revalidate this page in the future or to allow others to validate your page, the URI is <http://validator.w3.org/check?uri=http%3A%2F%2Fwww.websitearchitecture.co.uk%2F> (or you can just add the current page to your bookmarks or hotlist).

Validating CSS Style Sheets

If you use CSS in your document, you can check it using the W3C CSS Validation Service.

↑ TOP

Home About... News Docs Help & FAQ Feedback Contribute

This service runs the W3C Markup Validator, v0.8.5

COPYRIGHT © 1994-2009 W3C® (MIT, ERCIM, KEIO), ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.

http://validator.w3.org/

<end type="slideshow" />