HTML v XHMTL:

HTML and XML are derived from **SGML (Standard Generalised Markup Language)** – standards for defining generalised markup language for documents

* Define document structure and attributes, rather than processing techniques
* Can be processed by the same techniques as programs and databases

HTML – Hypertext Markup Language

* The markup language for all webpages on the Web
* Used for text, images, objects, interactive forms
* Use of *tags* <> ; paired and self-closing
* How content is written is based on modularising content and using tags that give meaning to content on the page
  + **structural** – how content is structured on the page e.g. <div>, <id>, <p>
  + **semantic markup** – how content is written to give it meaning e.g. <h1> is because it’s the most important heading, not because its larger)

XML – Extensible Markup Language

* Information sharing – human and machine readable
* Strict, provides warnings and won’t validate until all fixed
* Elements must be closed
* No inbuilt element names
* XML is a document formatting skeleton

XHTML – Extensible Hypertext Markup Language

* Enables non-text markup languages part of the XML suite
  + Scalable Vector Graphics (SVG)
  + Math Markup (MathML)
  + Geographical Markup (GeographyML)
  + RSS
  + http://en.wikipedia.org/wiki/List\_of\_XML\_markup\_languages

|  |  |
| --- | --- |
| **XHTML** | **HTML** |
| XHTML uses XML   * All elements must be CLOSED * Elements and attributes CASE-SENSITIVE * Attribute values MUST be included * All elements must be included | HTML uses pseudo SGML   * Elements can be left OPEN * Not case-sensitive * Attribute values/quotes can be omitted * Elements can be inferred if not included |
| Parsing errors will stop the program loading | Will still parse with minor errors |
| CSS selectors CASE SENSITIVE | CSS selectors NOT CASE SENSITIVE |
| JavaScript CASE SENSITIVE   * Document.write() not available | JavaScript NOT CASE SENSITIVE |
| Supports the full range of XML languages |  |
| All elements within <body> must be **block level** | Elements can be **inline level** or **block level** |

DOM:

DOM (Document Object Model)

* Representation of the whole document as nodes and attributes
  + Change or remove nodes or attributes
  + Create new nodes or add attributes to existing nodes
* Cross-platform modelling for representing objects in HTML, XML and XHTML
* Tree structure represents the page internally in applications – XHTML and HTML are two different ways of representing that model using different markup (serialisations)

Legacy DOM

* First generation scripting for early browsers (Navigator 2.0 and Internet Explorer 3.0)
* Limited facilities for detecting user-generated events

Intermediate DOM

* Navigator 4.0 and Internet Explorer 4.0 – adding support for Dynamic HTML (DHTML)
* DHTML required extensions to rudimentary document objects available in Legacy DOM

DOM Tree



* Browser sees a HTML document as a list of nested nodes, attributes and child nodes
* Nodes can be images, text, comments, links etc.

DOM Retrieval Methods

* document.getElementById(ID) – returns element with the ‘ID’
* document.getElementsByTagName(p) – returns an array of all the elements with the <p> tag

DOM child nodes and properties

* Child nodes are subsequent elements of a node in a nested sequence – to access them;

x = document.getElementById(“formatlist”);

* Initialise the variable ‘x’ and access its methods (similar to OOP), or places in the array  
    
  x.firstChild;  
  x.lastChild;  
  x[2];  
  x[3];
* When you have a node (or child node) you can read its properties.
* document.getElementById**(**nav**).**getElementByTagName**(**li**)[**1**].**firstChild**;**

<ul id=**"nav"**>

nodeName : A  
nodeType : 1  
nodeValue : null

<li><a href=act.html>**ACT**</a></li>

<li><a href=nsw.html>**NSW**</a></li>

<li><a href=nt.html>**NT**</a></li>

</ul>

* nodeName – element in uppercase (e.g. A, P, H1) or ‘#text’ for text
* nodeType – 1 for element, 3 for text
* nodeValue – ‘null’ for element, ‘text’ for text

Event Listeners:

Substitute event attributes in HTML code, greater separation of HTML and JavaScript. There are two options to ADD and REMOVE an event listener.

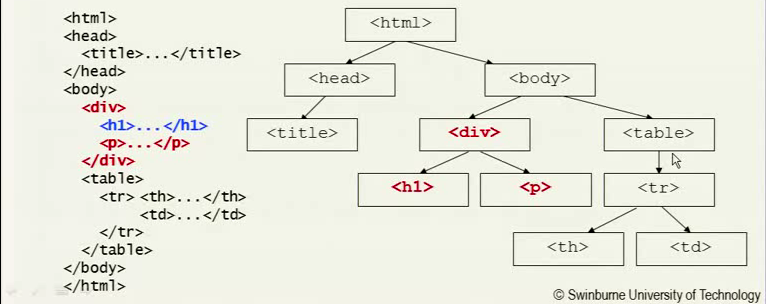
* object.addEventListener(event, function, capture);
* object.removeEventListener(event, function, capture);

The three arguments for the body of the event listener are;

* **event** – the action that calls the function (refer to full list of event attributes), the “on” must be omitted (e.g. ‘~~on~~click’, ‘~~on~~mouseover’)
* **function** – the function that’s called when the event is triggered, omit () (e.g. clearText~~()~~, FunctionCall~~()~~)
* **capture** – Boolean expression denoting whether the action occurs in the true or false state

Add the event listener into the .js file after the function definition.CSS Hierarchy:

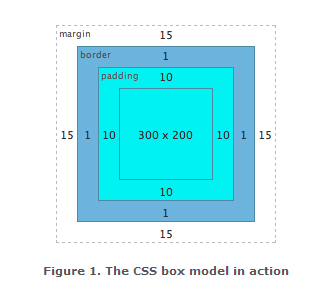
* **Last property style applied wins**



div**{color: red;}**

h1 **{color: blue;}**

CSS Box Model:



Definitions:

Internet general overview

**TCP/IP (Transmission Control Protocol/Internet Protocol)** – end-to-end connectivity specifying how data should be formatted, addressed, transmitted, routed and received

* Link layer (Ethernet) – local device, direct connection technologies
* Internet layer (IP) – connects local networks using IP addresses
* Transport layer (TCP) – host-to-host communication
* Application layer (HTTP) – protocols for specific web related applications and services operating on a process-to-process level (web browser to a server)

Basic transfer protocols

**FTP (File Transfer Protocol)** – transfer files from one host to another, defined in the TCP/IP Application layer

Security

**HTTP (Hypertext Transfer Protocol)** – the protocol for the transfer of hypertext documents

**HTTP Secure (HTTPS)** – protocol for secure transmission, an extension of HTTP on top of the SSL/TLS protocol

**SSH (Secure Shell)** – secure data communication over a network, encrypts data sent across the Internet (unlike Telnet which sends plain-text information, easily distinguished if intercepted)

**SSL (Secure Socket Layer)** – encrypts segments of network connections of Application layer for the Transport layer

Client/Server technologies

**Server side technologies**

* **Server-side Includes (SSI)** – interpreted scripting language to include one or more files into a web page on a web server <!--#include file="**something.ssi**" -->, executed via server
* **Common Gateway Interface (CGI)** – gateway interface to the server, reads file extension of the client request and determines the process, gateway between other programs/files/databases located on the server
* **Server-side Scripting** – some languages used on servers
  + ASP (Active Server Pages)
    - ASP.NET – larger language support, controls, XML components
  + PHP (Personal Hypertext Processor)
  + Python
  + Ruby
  + JavaScript
* **PHP** – server-side scripting, supports databases

**Client side technologies**

* **Client-side Scripting** **–** dynamic content delivery (DHTML), web page content can change depending on user input, location etc.

**Cookies** – small piece of data sent by a website to a client browser to store state, or user preference information about the website, notifies the website about the user’s previous activity on the website (used because HTTP is stateless)

**Sessions** – difference between ‘state’ and ‘stateless’ connections

* Stateless – only maintains connection for a single request, if user wants another file must create a new session (connection)
* State – keep track of state via cookies, when user connects server can read previous information about the user from the cookies stored from pervious connections

Finding websites

**DNS (Domain Name System)** – translates domain names to IP addresses

* **FQDN (Full Qualified Domain Name)** – hostname.department.organisation.orgtype.country

**URI (Uniform Resource Identifier)** – identifying location of web resources i.e. webpages, made up from locators (URLs) and names (URNs) which allow access and interaction to pages over the internet

* **URL (Uniform Resource Locator)** – used for getting the location of a web space, [protocol://server.domainnaame:port/pathandencodingstuff]
  + Protocol (scheme name)
  + Server (e.g. www)
  + Domain name (or an IP address)
* **URN (Uniform Resource Name)** – used for identification of a web space, location independent resource identifiers, encode character data in a form that can be sent over existing protocols (keyboards)

**MIME (Multipurpose Internet Mail Extensions)** – international standard for exchanging information with ‘non-textual’ parts – support expanded the standards for email and the Web which allows; text in character sets other than ASCII, non-text attachments, message bodies with multiple parts, non-ASCII header information

Other

**WCAG (Web Content Accessibility Guidelines)**

1. Perceivable

* Text alternatives – provide text alternatives for any ‘non-text’ based content
* Time-based media – provide alternatives for multimedia
* Adaptable – can be delivered and received across different devices, content can be presented in different ways without losing information or structure
* Distinguishable – make it easier for users to see and hear content

2. Operable

* Keyboard accessible
* Enough time – to read and use content
* Seizures – do not design content in ways that can cause seizures
* Navigable – help users navigate, find content, determine location

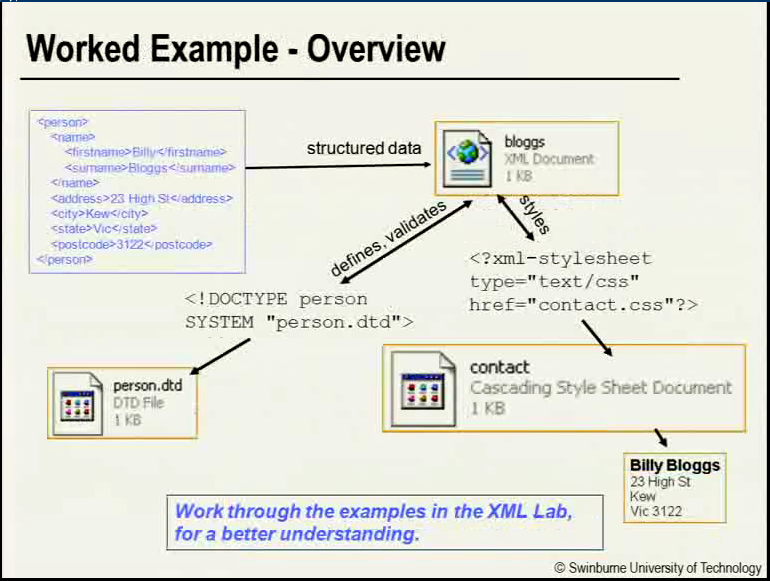
3. Understandable

* Readable
* Predictable – pages appear and operate as they predicted
* Input assistance – help to avoid and correct mistakes

4. Robust

* Compatible – maximise compatibility with current and future technologies

**DTD (Document Type Definition)** – markup declarations that define a document type for an SGML-family markup language (SGML, XML, HTML), formally specify rules of a document type (which element types are allowed, combination, attributes, values etc.)



SEARCH ENGINE – finds files stored on a computer

WEB SEARCH ENGINE – automatically scans the Web and stores results into an indexed database, databases can then be queried by users when searching for specific Web domains

WEB DIRECTORY – a database which lists a collection of links to websites ordered by categories or topics, **managed by people instead of an automated search engine**

**CMS (Content Management System)** – general term to describe web application software that supports collaborative content creation and management

**AJAX (Asynchronous JavaScript and XML)** **/ XHR (XML HTTP Request)** – JS that runs in the browser behind the scenes, sends HTML requests to the web server without user input and returns data without reloading the page (e.g. search suggestions when entering a term in the search bar)

**JSON (JavaScript Object Notation)** – lightweight format for creating JS objects, subset of JS, language independent