THE UNIVERSITY OF WESTERN AUSTRALIA

CITS3403 Agile Web Development

Topic 2: HTML

Unit Coordinator: Tim French

Semester 1, 2022

Anatomy of a URL

THE UNIVERSITY OF WESTERN AUSTRALIA

http://www.domain.edu.au:1000/path/to/file?parameters=true#fragment

- The protocol used. Typically http, ftp, https, ...
- The domain name. A domain name server maps this to an IP address
- The port number. Servers have ports 0-65535, but http defaults to port 80.
- The path (route) to the file to execute. The file is typically an html file, but it could also be php, text, pdf.
- The parameters of the request. These are specified as a set of key value pairs.
- The fragment. This anchors to a location in a page.
- There are also hidden parts of the request including the browser name and cookies.

Key Web Technologies

THE UNIVERSITY OF WESTERN AUSTRALIA

- HTML describes the semantic content of a web page and the logical relationships between content.
- CSS (cascading style sheets, next lecture) describes the style and appearance of a web page.
- Javascript is an interpreted language that runs on the client device. It provides the functionality in a web page.

Hyper Text Markup Language

THE UNIVERSITY OF WESTERN AUSTRALIA

- HTML was originally defined with as a type of SGML in 1990, by Tim Berners-Lee.
- Original intent of HTML: General layout of documents that could be displayed by a wide variety of computers.
- Along with Cascading Style Sheets and Javascript as the standard web technologies.

Philosophies behind HTML5

- Specifying current browser behaviour to ensure interoperability
  - Clear specifications on error handling
- Not breaking the web (backward compatiability)
- Programmatic rather than theoretic (HTML5 is not XML)
- User > Web Designer > Browser Implementer > Standard Theorists
- Aiming at easier authoring of Web Applications
- One of the key ideas behind the web and is the separation between *what sort of information* it is, and *how it should be displayed*

HTML Basic Syntax

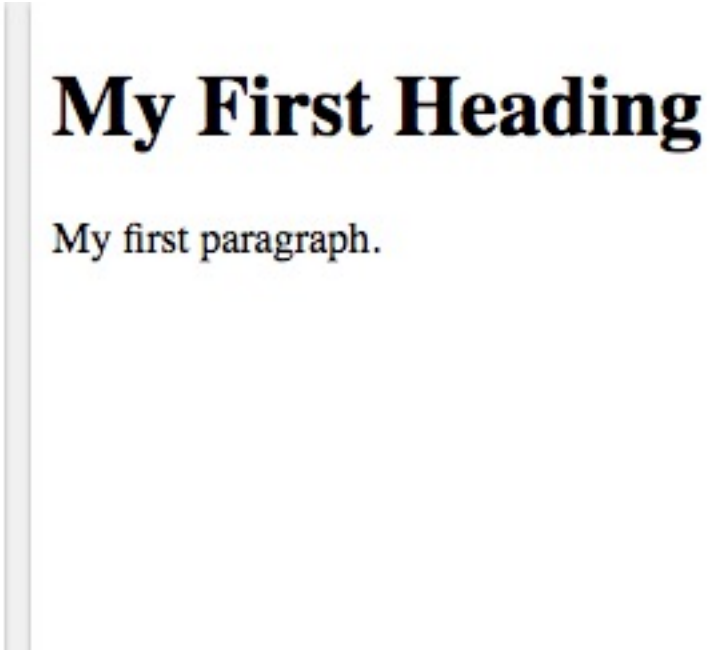
THE UNIVERSITY OF WESTERN AUSTRALIA

- Elements are defined by *tags* (markers)
  - Tag format: <name> ..content... </name>, or <name/>
  - The container and its content together are called an *element*

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```



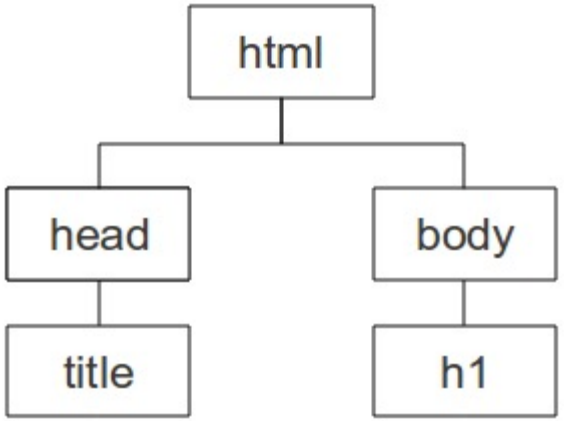
HTML Syntax

THE UNIVERSITY OF WESTERN AUSTRALIA

- If a tag has *attributes*, they appear between its name and the right bracket of the opening tag
  - eg. `<table border="1" cellpadding="1">...</table>`
- Comments:
  `<!-- ignore stuff here... -->`
- Browsers ignore comments, unrecognizable tags, line breaks, multiple spaces, and tabs
- Tags are suggestions to the browser, can be ignored (even if they are recognized by the browser)

HTML Document Structure

- Every HTML5 document should begin with a DOCTYPE declaration: <!DOCTYPE html>
- <html>, <head>, <title>, and <body> are required in every document
- The whole document must have <html> as its root
- A document consists of a head and a body
- The <title> tag is used to give the document a title, which is normally displayed in the browser's window title bar (at the top of the display)
- Document is a tree of elements
- Visible elements are on <body> branch



Highlighting and Special Characters

THE UNIVERSITY OF WESTERN AUSTRALIA

- Font Styles and Sizes (can be nested)
  - Boldface - <b>
  - Italics -
  - Larger - <big>
  - Smaller - <small>
  - Monospace - <tt>

- The <big> sleet <big> in <big> Crete <br /> lies </big> completely </big> in </big> the street

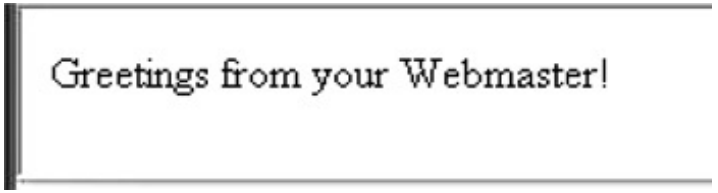
The sleet in *Crete*
lies completely in the street

Character	Entity	Meaning
&	&amp;	Ampersand
<	&lt;	Less than
>	&gt;	Greater than
"	&quot;	Double quote
'	&apos;	Single quote (apostrophe)
1/4	&frac14;	One quarter
1/2	&frac12;	One half
3/4	&frac34;	Three quarters
°	&deg;	Degree
(space)		Nonbreaking space

The HTML5 Way

- Document is a *tree* of elements
- No need to have <html>, <head>, and even <body>
  - No need to close <p> elements (but you should)
  - This is valid HTML5 code

```
<!DOCTYPE html>
<html lang = "en">
  <head>
     <meta charset="utf-8"/>
     <title> Our first document </title>
  </head>
  <body>
    <p>
      Greetings from your Webmaster!
    </p>
  </body>
</html>
```



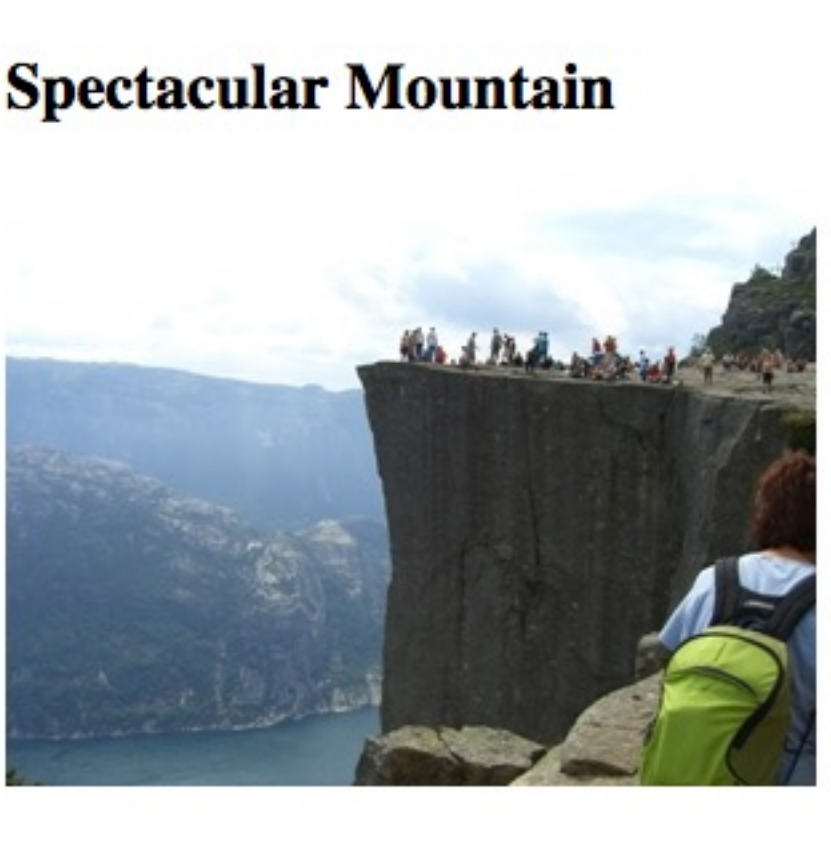
Images

- Images are inserted into a document with the `<img />` tag with the `src` attribute

```
<!DOCTYPE html>
<html>
<body>

<h2>Spectacular Mountain</h2>
<img src="pic_mountain.jpg" alt="Mountain View"
style="width:304px;height:228px;">

</body>
</html>
```



Hypertext Links

THE UNIVERSITY OF WESTERN AUSTRALIA

- Hypertext is the essence of the Web!
- A link is specified with the `href` (*h*ypertext *ref*erence) attribute of `<a>` (the anchor tag)
  - The content of `<a>` is the visual link in the document
  - Note: Relative addressing of targets is often easier to maintain and more portable than absolute addressing
  - You can link to elements in the same document, use an id attribute:
    - `<H2 id="Link">Link to me!</H2>`
    - `<a href="#Link">linking...</a>`
  - The `href` can be any file, not just html.

Hypertext Links

```
<html>
 <head> <title> Links </title>
  </head>
  <body>
    <h1> Aidan's Airplanes
  </h1>
    <h2> The best in used
  airplanes </h2>
    <h3> "We've got them by the
  hangarful" </h3>
    <h2> Special of the month
  </h2>
    <p>
     1960 Cessna 210 <br />
     <a href = "C210data.html">
        Information on the
  Cessna 210
     </a>
    </p>
  </body>
</html>
```

Lists

- *Unordered lists*
  - The list is the content of the `<ul>` tag
  - List elements are the content of the `<li>` tag
  - Use `<ol>` for *ordered* lists
  - type attribute can change from numbers to letters etc.

```
<h3> Some Common Single-Engine Aircraft </h3>
 <ul>
   <li> Cessna Skyhawk </li>
   <li> Beechcraft Bonanza </li>
   <li> Piper Cherokee </li>
 </ul>
```

Tables

- A table is a matrix of cells, each possibly having content
  - The cells can include almost any element
  - Some cells have row or column labels and some have data
  - A table is specified as the content of a `<table>` tag
  - A `border` attribute in the `<table>` tag specifies a border between the cells
  - The `border` attribute can be set to a number, which will be the border width
  - Without the `border` attribute, the table will have no lines
  - Tables are given titles with the `<caption>` tag, which can immediately follow `<table>`
  - Each row of a table is specified as the content of a `<tr>` tag
  - The row headings are specified as the content of a `<th>` tag
  - The contents of a data cell is specified as the content of a `<td>` tag

Tables (continued)

```
<!DOCTYPE html>
<html>
<body>
<table style="width:100%" border="border">
  <tr>
    <th>Firstname</th> <th>Lastname</th> <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td> <td>Smith</td> <td>50</td>
  </tr>
  <tr>
    <td>Eve</td> <td>Jackson</td> <td>94</td>
  </tr>
  <tr>
    <td>John</td> <td>Doe</td> <td>80</td>
  </tr>
</table>
</body>
</html>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80

- The `cellspacing` attribute of `<table>` is used to specify the distance between cells in a table

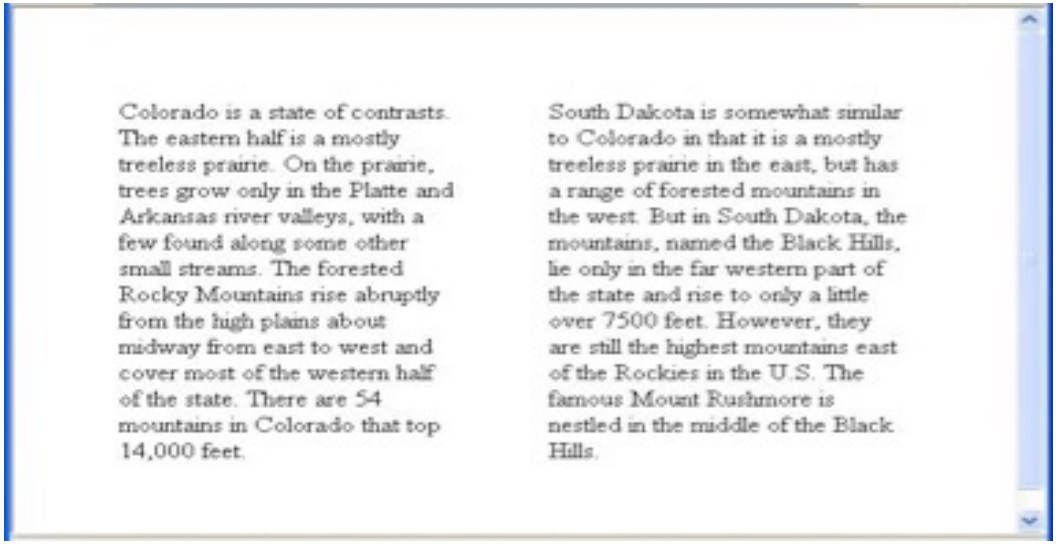
The `cellpadding` attribute of `<table>` is used to specify the spacing between the content of a cell and the inner walls of the cell

Tables for layout?

Tables have frequently been used to layout a webpage.

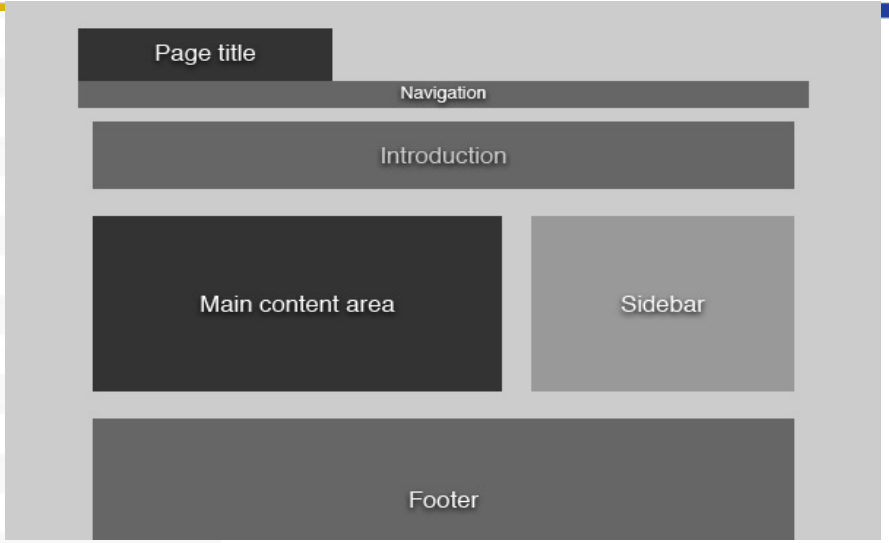
Why is this not a great idea?

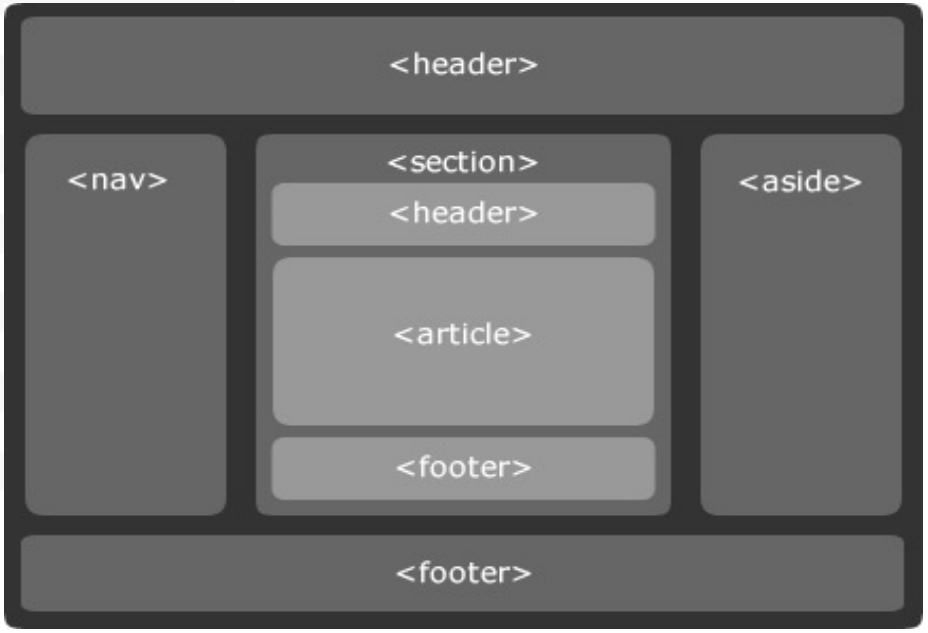
```
<table cellspacing = "50">
      <tr>
         <td> Colorado is a
state of …
         </td>
         <td> South Dakota is
somewhat…
         </td>
      </tr>
</table>
```



Layout in HTML5

```
1.  <!doctype html>
2.  <html>
3.  <head>
4.      <title>Page title</title>
5.  </head>
6.  <body>
7.      <header>
8.          <h1>Page title</h1>
9.      </header>
10.     <nav>
11.         <!-- Navigation -->
12.     </nav>
13.     <section id="intro">
14.         <!-- Introduction -->
15.     </section>
16.     <section>
17.         <!-- Main content area -->
18.     </section>
19.     <aside>
20.         <!-- Sidebar -->
21.     </aside>
22.     <footer>
23.         <!-- Footer -->
24.     </footer>
25.
26. </body>
27. </html>
```





HTML5 Elements

THE UNIVERSITY OF WESTERN AUSTRALIA

- <header>
  The header element contains introductory information to a section or page.
- <nav>
  The nav element is reserved for a section of a document that contains links to other pages or links to sections of the same page.
- <section>
  The section element represents a **generic document or application section**. It acts much the same way a does by separating off a portion of the document.
- <article>
  The article element represents a portion of a page which can stand alone such as: a blog post or a forum entry.
- <aside>
  Aside, represents content related to the main area of the document. Usually expressed in sidebars that contain elements like related posts, tag clouds.
- <footer>
  The footer element is for marking up the footer of, not only the current page, but each section contained in the page.

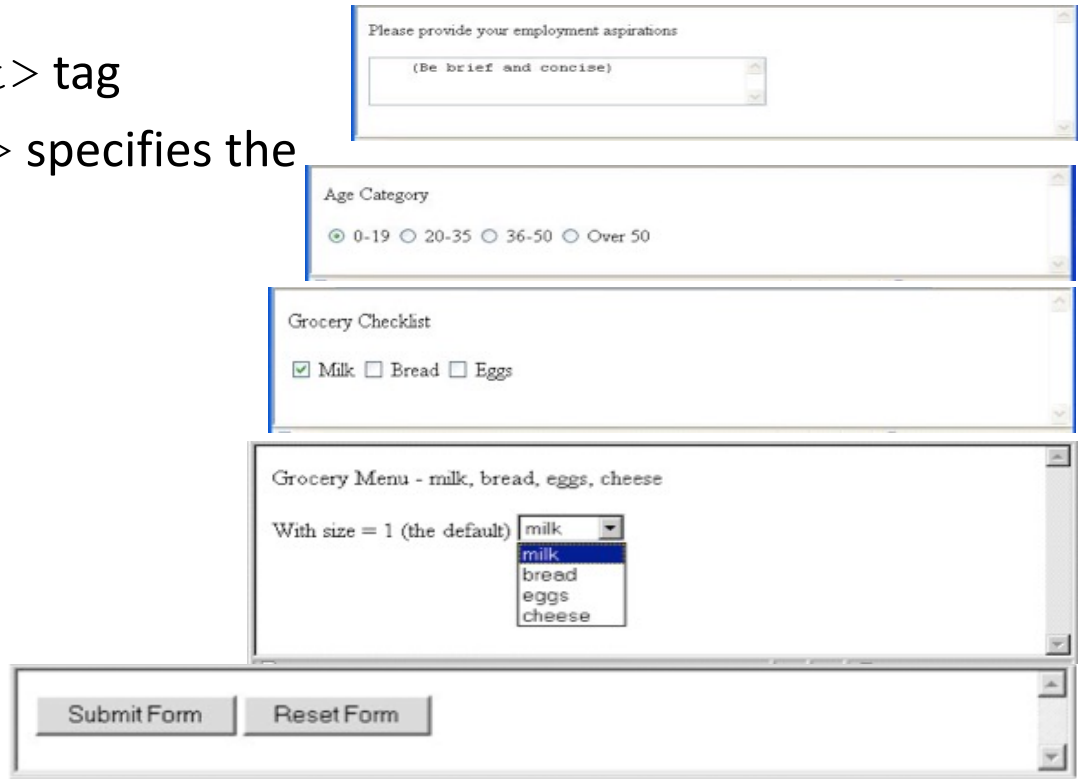
Forms

- A form is the usual way to get information from a browser to a server
- HTML has tags to create a collection of objects that implement this information gathering
  - The objects are called *widgets* (e.g., radio buttons and checkboxes)
- When the Submit button of a form is clicked, the values are sent to the server
- All of the widgets, or components of a form are defined in the content of a `<form>` tag
  - The only required attribute of `<form>` is `action`, which specifies the URL of the application that is to be called when the Submit button is clicked
  - `action = "http://www.cs.ucp.edu/cgi-bin/survey.pl"`
    - » If the form has no action, the value of `action` is the empty string

Type	Description
<input type="text">	Displays a single-line text input field
<input type="radio">	Displays a radio button (for selecting one of many choices)
<input type="checkbox">	Displays a checkbox (for selecting zero or more of many choices)
<input type="submit">	Displays a submit button (for submitting the form)
<input type="button">	Displays a clickable button

Forms (continued)

- The `method` attribute of `<form>` specifies one of the two possible techniques of transferring the form data to the server, `get` and `post`
  - you might recognise these in the topic on *protocols*..
- *Widgets*
  - Many are created with the `<input>` tag
    - » The `type` attribute of `<input>` specifies the kind of widget being created
    - Text
    - Text Areas
    - Checkboxes
    - Radio buttons
    - Menus
    - Reset and submit buttons



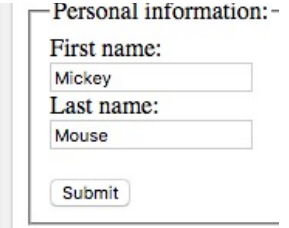
Form actions

Forms have an *action* attribute which is the script that is executed when the form is submitted.
Named fields are *post*ed to the script.

```
<!DOCTYPE html>
<html>
<body>

<form action="/action_page.php">
  <fieldset>
    <legend>Personal information:</legend>
    First name:<br>
    <input type="text" name="firstname" value="Mickey">
    <br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse">
    <br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>

</body>
</html>
```



Submit has two actions:

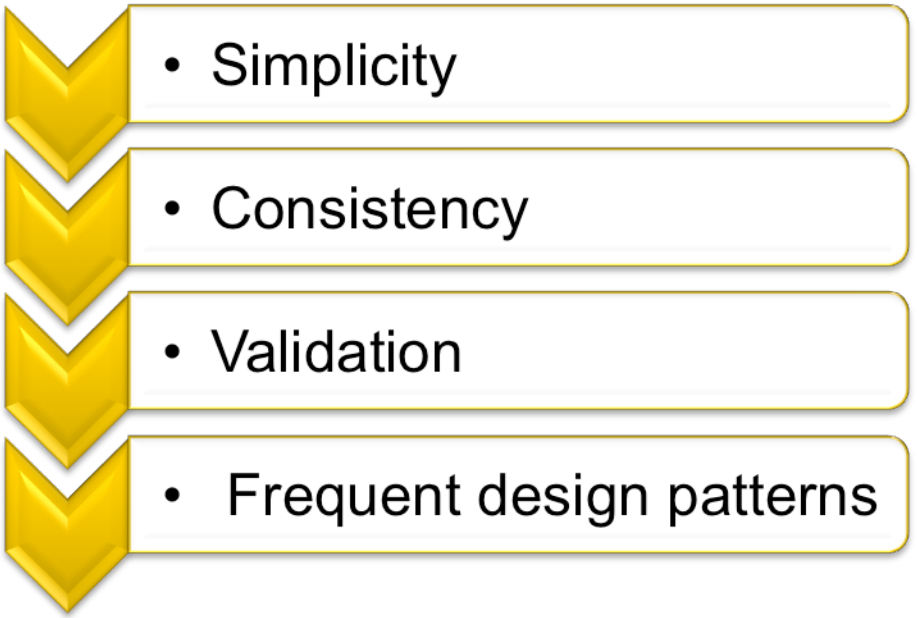
1. Encode the data of the form
2. Request that the server execute the server-resident program specified as the value of the `action` attribute of `<form>`

– A Submit button is required in every form

HTML5 Forms

THE UNIVERSITY OF WESTERN AUSTRALIA

- New Attributes
  - placeholder
  - autocomplete (on, off)
  - required
  - autofocus
- Form controls
  - datalist
- Input types
  - search
  - Contacts (email, url, tel)
  - Slider: range
  - Spinner: number
  - Data/time: date, datetime, datetime-local, month, week



HTML5 Forms - Summary

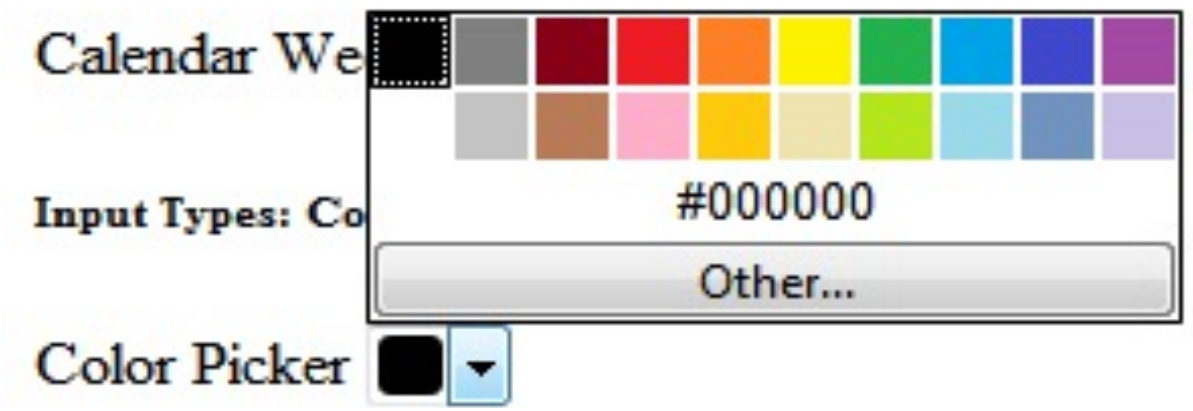
THE UNIVERSITY OF WESTERN AUSTRALIA

type="text"
type="radio"
type="check"

type="submit"
type="reset"
type="file"

<textarea>
<fieldset>
<select> <option>

type="email"
type="tel"
type="url"
type="date"
type="search"
type="range"
type="number"
type="color"



type="date | datetime | datetime-local | week | month"
<output>

The HTML 5 Way

THE UNIVERSITY OF WESTERN AUSTRALIA

- The Mobile Safari changes on-screen keyboard according to different contact input types:

HTML 5 Forms: Dates and times

THE UNIVERSITY OF WESTERN AUSTRALIA

- One of the most popular JavaScript widgets is the calendar picker.
- These calendar widgets all do the same thing, but you'll find that they're implemented slightly differently on each site. A native calendar widget would smooth away the inconsistencies and reduce cognitive load during the date-picking process.
- HTML5 introduces a raft of input types specifically for dates and times
  - **date** is for a year, month, and day.
  - **datetime** is for a year, month, and day in combination with hours, minutes, and seconds and time zone information.
  - **datetime-local** is the same but without the time zone information.
  - **time** is for hours, minutes, and seconds.
  - **month** is for a year and a month but without a day.

HTML 5 Forms: Dates and times

THE UNIVERSITY OF WESTERN AUSTRALIA

- All of these input types will record timestamps with some subset of the standardized format YYYY-MM-DDThh:mm:ss.Z (Y is year, M is month, D is day, h is hour, m is minute, s is second, and Z is timezone). Take, for example, the date and time at which World War One ended, 11:11am on November 11th, 1918:
  - **date**: 1918-11-11
  - **datetime**: 1918-11-11T11:11:00+01
  - **datetime-local**: 1918-11-11T11:11:00
  - **time**: 11:11:00
  - **month**: 1918-11
- There is no **year** input type, although there is a **week** input type that takes a number between 1 and 53 in combination with a year.

HTML 5 Forms – Build-in Validation

THE UNIVERSITY OF WESTERN AUSTRALIA

- Native validation without scripting.
- HTML5 browsers support basic validation on email, url and tel input types.
- HTML5 has made it even more friendly for web authoring
  - The pattern attribute that allows you to use regular expression to specify required format
  - For example:

```
<input id="phone" name="phone" pattern="\d{8}" type="tel">
```

The form output element

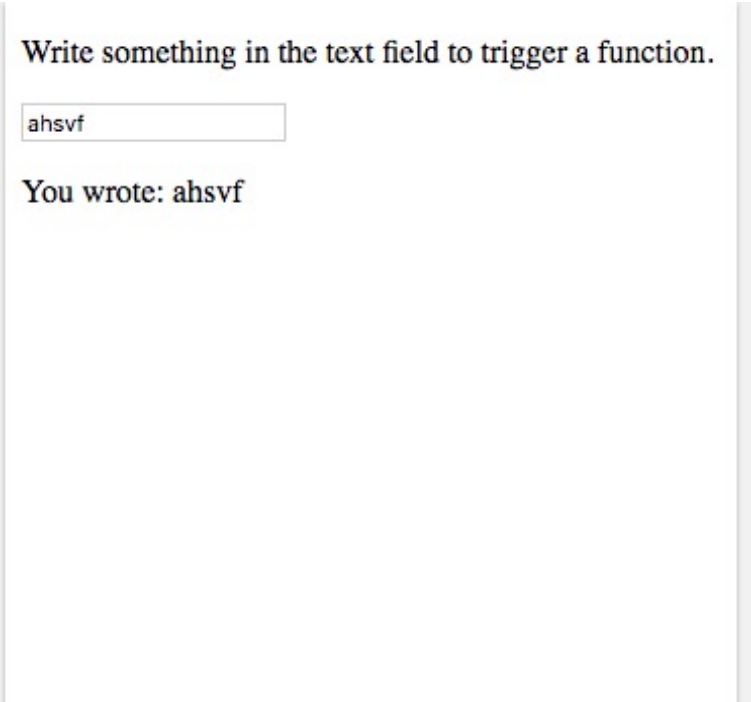
- Represent the results of some calculation

```
<form oninput="result.value=parseInt(a.value)+parseInt(b.value)”>
0<input type="range" name="b" value="50" />100 +
<input type="number" name="a" value="10" /> =
<output name="result"></output>
</form>
```

```
<!DOCTYPE html>
<html>
<body>

<p>Write something in the text field to trigger a function.</p>
<input type="text" id="myInput" oninput="myFunction()">
<p id="demo"></p>
<script>
function myFunction() {
    var x = document.getElementById("myInput").value;
    document.getElementById("demo").innerHTML = "You wrote: " + x;
}
</script>

</body>
</html>
```



The <time> element

THE UNIVERSITY OF WESTERN AUSTRALIA

- Encode time and date in formats that are both Machine and Human readable

  <time datetime=2011-8-12> 12 August 2011</time>

  <time datetime=2011-8-12> 12 th August Last Year</time>

  <time datetime=2012-8-12>UWA Expo 2012</time>

  <time datetime=2012-8-12T14:00Z>2PM on UWA Expo 2012</time>

  <time datetime=20:00> 8PM</time>

- Previously, you could only mark up precise dates, which could be a problem (e.g. historians)
  - 13 November 1905 could be expressed in HTML but not November 1905

    <time datetime="1905-11-13">
- Now, "fuzzy dates" are possible:

  <time datetime="1905"> means the year 1905

  <time datetime="1905-11"> means November 1905

  <time datetime="11-13"> means 13 November (any year)

  <time datetime="1905-W21"> means week 21 of 1905

The <time> element - Durations

- In HTML5 <time>, you can represent durations, with the prefix "P" for "period".
  - The datetime attribute value: "D" for days, "H" for hours, "M" for minutes and “S" for seconds.
- You can separate them with spaces (but you don't have to).
  - <time datetime="P4D"> is a duration of 4 days, same as
  - <time datetime="P 4 D">
- Using a "T" after the "P" marker allows you to be more precise:
  - <time datetime="PT23H 9M 2.345S"> is a duration of 23 hours, 9 minutes and 2.345 seconds.
- The pubdate attribute is a boolean to indicate when a page is published

```
<section>
    <article>
        <header>
            <h1>Seminar: What is ARIA?</h1>
            <p><time datetime="2012-08-12T11:00">12 August 2012 11:00am</time></p>
        </header>
        <p>This seminar is about accessibility.</p>
        <footer>
            Published at: <time datetime="2012-08-08T20:00" pubdate>8 August 2012 8:00pm</time>
        </footer>
    </article>
</section>
```

HTML 5 NEW TAG

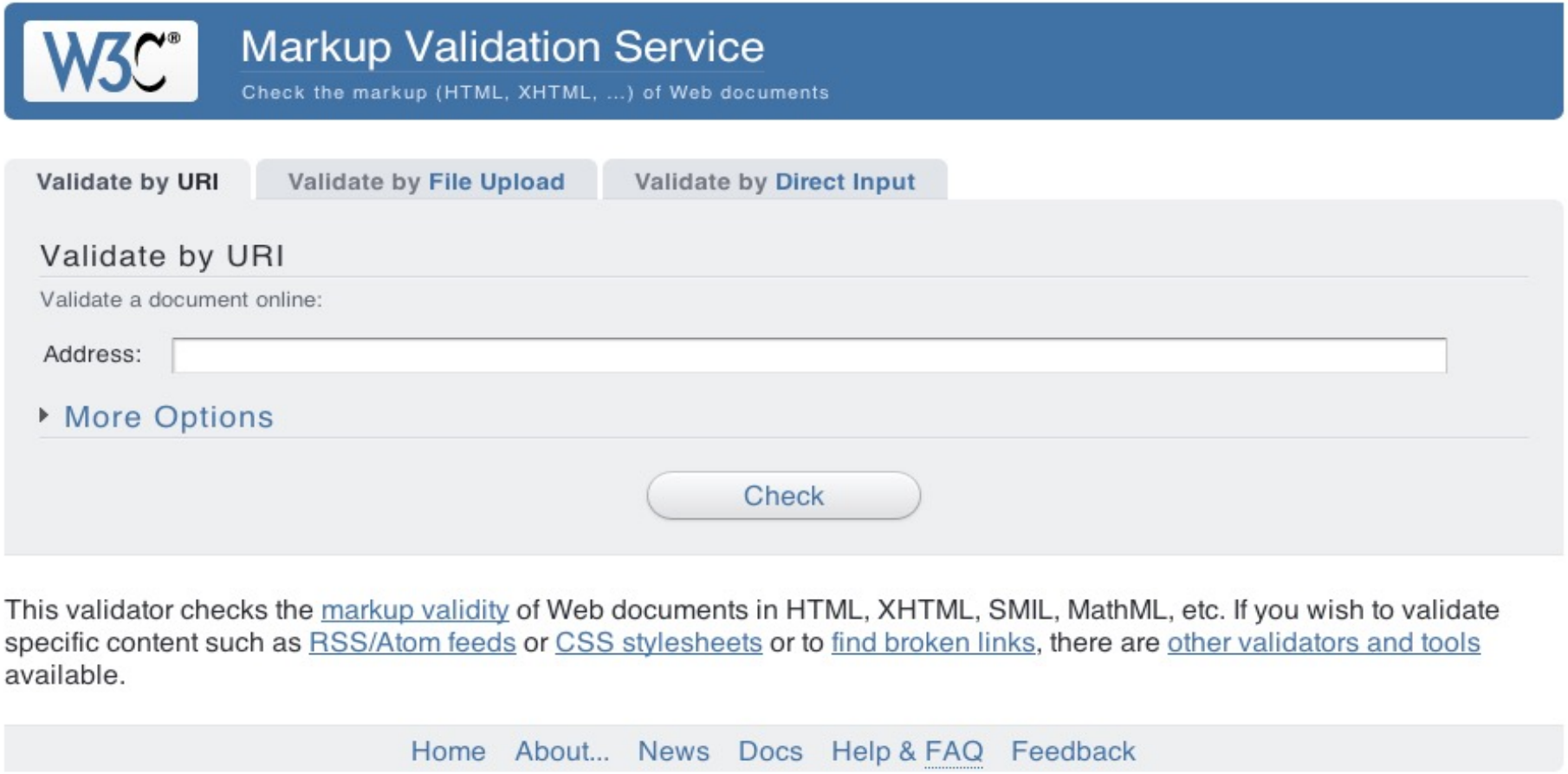
TAG NOT SUPPORTED IN HTML 5

<!--...-->	Define a comment
<!DOCTYPE>	Defines the document type
<a>	Defines a hyperlink href, hreflang, media, ping , rel, target, type
<abbr>	Defines an abbreviation
<acronym>	Used to define an embedded acronyms
<address>	Defines an address element
<applet>	Used to define an embedded applet
<area>	Defines an area inside an image map alt, coords, href, hreflang, media, ping, rel, shape, target, type
<article>	Defines an article cite, pubdate
<aside>	Defines content aside from the page content
<audio>	Defines sound content autobuffer, autoplay, controls, src
<b>	Defines bold text
<base>	Defines a base URL for all the links in a page href, target
<basefont>	Used to define a default font-color, font-size, or font-family for all the document
<bdo>	Defines the direction of text display dir
<big>	Used to make text bigger
<blockquote>	Defines a long quotation cite
<body>	Defines the body element
 	Inserts a single line break
<button>	Defines a push button autofocus, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, name, type, value
<canvas>	Defines graphics height, width
<caption>	Defines a table caption
	Used to center align text and content
<cite>	Defines a citation
<code>	Defines computer code text autobuffer, autoplay, controls, src
<col>	Defines attributes for table columns
<colgroup>	Defines groups of table columns span
<command>	Defines a command button checked, disabled, icon, label, radiogroup, type
<datalist>	Defines a dropdown list
<dd>	Defines a definition description
<del>	Defines deleted text cite, datetime
<details>	Defines details of an element open
<dialog>	Defines a dialog (conversation)
<dfn>	Defines a definition term
<dir>	Used to define a directory list
	Defines a section in a document
<dl>	Defines a definition list
<dt>	Defines a definition term
	Defines emphasized text
<embed>	Defines external interactive content or plugin height, src, type, width
<fieldset>	Defines a fieldset disabled, form, name
<figure>	Defines a group of media content, and their caption
<font>	Used to define font face, font size, and font color of text
<footer>	Defines a footer for a section or page
<form>	Defines a form accept-charset, action, autocomplete, enctype, method, name, novalidate, target
<frame>	Used to define one particular window (frame) within a frameset
<frameset>	Used to define a frameset, which organized multiple windows (frames)
<h1> to <h6>	Defines header 1 to header 6
<head>	Defines information about the document
<header>	Defines a header for a section or page
<hgroup>	Defines information about a section in a document
<hr>	Defines a horizontal rule
<html>	Defines an html document manifest, xmlns
<i>	Defines italic text
<iframe>	Defines an inline sub window height, name, sandbox, seamless, src, width
<img>	Defines an image alt, src, height, ismap, usemap, width
<input>	Defines an input field accept, alt, autocomplete, autofocus, checked, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, height, list, max, maxlength, min, multiple, name, pattern, placeholder, readonly, required, size, src, step, type, value,
<ins>	Defines inserted text cite, datetime
<keygen>	Defines a generated key in a form autofocus, challenge, disabled, form, keytype, name
<kbd>	Defines keyboard text
<label>	Defines an inline sub window for, form
<legend>	Defines a title in a fieldset
	Defines a list item value
<link>	Defines a resource reference href, hreflang, media, rel, sizes, type
<map>	Defines an image map name
<mark>	Defines marked text
<menu>	Defines a menu list label, type
<meta>	Defines meta information charset, content, http-equiv, name
<meter>	Defines measurement within a predefined range high, low, max, min, optimum, value
<nav>	Defines navigation links
<noframes>	Used to display text for browsers that do not handle frames
<noscript>	Defines a noscript section
<object>	Defines an embedded object data, form, height, name, type, usemap, width
	Defines an ordered list reversed, start
<optgroup>	Defines an option group label, disabled
<option>	Defines an option in a drop-down list disabled, label, selected, value
<output>	Defines some types of output for, form, name
<p>	Defines a paragraph
<param>	Defines a parameter for an object name, value
<pre>	Defines preformatted text
<progress>	Defines progress of a task of any kind max, value
<q>	Defines a short quotation cite
<rp>	Used in ruby annotations to define what to show browsers that to not support the ruby element
<rt>	Defines explanation to ruby annotations
<ruby>	Defines ruby annotations
<s>, <strike>	Used to define strikethrough text.
<samp>	Defines sample computer code
<script>	Defines a definition list async, type charset defer, src
<section>	Defines a section cite
<select>	Defines a selectable list autofocus, disabled, form, multiple, name, size
<small>	Defines small text
<source>	Defines media resources media, src, type
	Defines a section in a document
	Defines strong text
<style>	Defines a style definition type, media, scoped
<sub>, <sup>	Defines sub/super-scripted text
<table>	Defines a table summary
<tbody>	Defines a table body summary
<td>	Defines a table cell colspan, headers, rowspan
<textarea>	Defines a text area autofocus, cols, disabled, form, maxlength, name, placeholder, readonly, readonly, required, rows, wrap
<tfoot>, <thead>	Defines a table footer / head
<th>	Defines a table header colspan, headers, rowspan, scope
<time>	Defines a date/tim datetime
<title>	Defines the document title
<tr>	Defines a table row datetime
<tt>	Used to define teletype text
<u>	Used to define underlined text
	Defines an unordered list
<var>	Defines a variable
<video>	Defines a video autobuffer, autoplay, controls, height, loop, src, width

HTML5 TAG CHEAT SHEET

Created by WebsiteSetup.org

Validation


Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents
Validate by URI
Validate by File Upload
Validate by Direct Input
Validate by URI
Validate a document online:
Address:
More Options
Check
This validator checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as RSS/Atom feeds or CSS stylesheets or to find broken links, there are other validators and tools available.
Home About... News Docs Help & FAQ Feedback