THE UNIVERSITY OF WESTERN AUSTRALIA

HTML

CITS3403 and CITS5505 – Agile Web Development

Unit Coordinator: Matthew Daggitt

Semester 1, 2024

Key Web Technologies

THE UNIVERSITY OF WESTERN AUSTRALIA

- HTML - Lecture 2 - describes the semantic content of a web page and the logical relationships between content.
- CSS - Lectures 3 & 4 - describes the style and appearance of a web page.
- JavaScript - Lectures 5, 6 & 7 - an interpreted language that runs on the client device. It provides the functionality in a web page.

JavaScript

THE UNIVERSITY OF WESTERN AUSTRALIA

HTML5 basics

HTML (Hyper Text Markup Language)

THE UNIVERSITY OF WESTERN AUSTRALIA

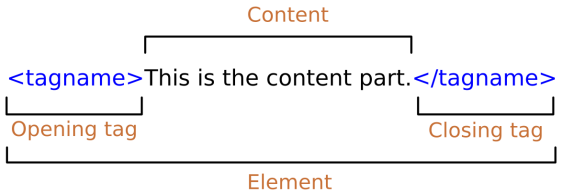
- HTML was originally defined as a type of SGML in 1990, by Tim Berners-Lee.
- HTML5 was a significant revamp of the language released in 2008.

The HTML5 philosophy

1. Interoperability – should be renderable on a wide variety of browsers.
2. Graceful error recovery – small errors should not stop the page from rendering.
3. Backwards compatible – new features should not break the web.
4. Prioritise users – User > Web Designer > Browser Implementer > Theorists.
5. Separation of concerns – describe the *type* of information, not how it displays.

HTML elements

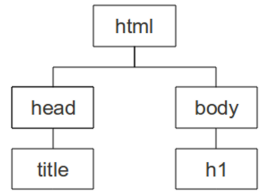
- An HTML document is made up of elements.
- An element is usually made up of opening and closing tags with content between.



- An element's content may contain further elements within it.
- HTML is therefore a tree data structure.

HTML document structure

- HTML5 documents begin with a `<!DOCTYPE html>` declaration.
- The document has a `<html>` tag 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.
- Visible elements are on `<body>` branch




```
<!DOCTYPE html>
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
    <h1>Welcome to Agile Web!</h1>
    <p> Ready to build a website? </p>
  </body>
</html>
```

Welcome to Agile Web!

Ready to build a website?

HTML validity

- In keeping with the HTML5 philosophy, web browser renderers are very relaxed!
- Browers ignore:
  - Unrecognised tags
  - Line breaks
  - Tabs, Multiple spaces
- No need to have key document tags such as `<html>`, `<head>`, and even `<body>`
- No need to have matching opening and closing tags (but you *really* should as it an easy source of bugs!)

- Tags are only suggestions to the browser, can be ignored (even if they are recognized by the browser!)

HTML text

- Text can be included as the content of many elements:
  - Headings: `<h1>`, `<h2>`, `<h3>`, `<h4>`
  - Paragraphs: `<p>`
  - Code: `<code>`
  - Emphasis text: `<em>`
- All these can be nested.
- Various special characters use `&...;`

Character	Entity	Meaning
&	`&`	Ampersand
<	`<`	Less than
>	`>`	Greater than
"	`&quot;`	Double quote
'	`&apos;`	Single quote (apostrophe)
¼	`&frac14;`	One quarter
½	`&frac12;`	One half
¾	`&frac34;`	Three quarters
°	`&deg;`	Degree
(space)	` `	Nonbreaking space

```
<!DOCTYPE html>
<html>
  <body>
    <h1> Coding 101 </h1>
    <p> Never <em> ever </em> use the name <code> thing </code> for a variable. </p>
  </body>
</html>
```

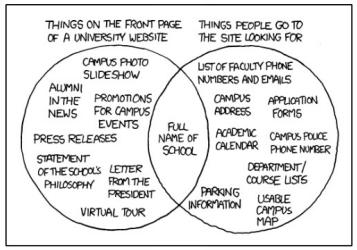
Coding 101

Never *ever* use the name `thing` for a variable.

HTML images

- All HTML tags can have a list of attributes which appear between its name and the right bracket of the opening tag.
- Images are inserted into a document with the `<img/>` tag with the `src` attribute.
- The `alt` attribute provides text for if the image can't be displayed or for screen readers.

```
<!DOCTYPE html>
<html>
  <body>
    <img
      src="https://imgs.xkcd.com/comics/university_website.png"
      alt="Pointed advice on university website design"
      style="width:500px"
    >
  </body>
</html>
```



HTML lists

- There are two types of lists:
  - An ordered list used the `<ol>` tag.
  - An unordered list uses the `<ul>` tag
- Elements of a list are added as contents of the `<li>` tag.
- The `type` attribute of `<ol>` can be used to change the list marker (e.g. to letters).

```
<!DOCTYPE html>
<html>
  <body>
    <h3> The best bubble tea places in Perth </h3>
    <ol type="1">
      <li> T4 </li>
      <li> Chatime </li>
      <li> Utopia </li>
    </ol>
    <h3> Runners up </h3>
    <ul>
      <li> Presotea </li>
      <li> Chaffic </li>
    </ul>
  </body>
</html>
```

The best bubble tea places in Perth

1. T4
2. Chatime
3. Utopia

Runners up

- Presotea
- Chaffic

HTML tables

- A table is a matrix of cells, each possibly having content
  - A table is specified as the content of a `<table>` tag
  - 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

```
<!DOCTYPE html>
<html>
  <body>
    <h1> Votes for our new kitten's name </h1>
    <table border=1 >
      <tr> <th> Suggested name </th> <th> Votes </th> <tr>
      <tr> <td> Whiskers </td> <td> 129 </td> <tr>
      <tr> <td> Schrodinger </td> <td> 2 </td> <tr>
      <tr> <td> Cleocatra </td> <td> 2 </td> <tr>
      <tr> <td> Sir Isaac Mewton </td> <td> 1 </td>
      <tr>
    </table>
  </body>
</html>
```

Votes for our new kitten's name

Suggested name	Votes
Whiskers	129
Schrodinger	2
Cleocatra	2
Sir Isaac Mewton	1

HTML table attributes

- The <table> element tag has various useful attributes:
  - The `cellspacing` attribute sets the distance between cells.
  - The `cellpadding` attribute sets the spacing between the cell's content and its inner walls.
  - The `border` attribute sets the width of the border between the cells. Without the border attribute, the table will have no visible cell borders.
- It can be *very* tempting to use borderless tables to layout a webpage.... don't!
  - Violates semantic intent – screen readers will read it as a table.
  - Browsers may decide to render it as a table regardless!
  - Quickly gets unmaintainable as you usually need (deeply) nested tables.
  - Can be slow to render in the browser for various reasons...

HTML hypertext links

- Hypertext was one of the key selling points of the Web!
- A link is created using an anchor tag <a> with a href (***h***ypertext ***ref***erence) attribute.
- The content of <a> is the visual link in the document (can be images, text etc.)

```
<!DOCTYPE html>
<html>
  <body>
    <a href="https://www.youtube.com/watch?v=dQw4w9WgXcQ">
      Agile Web Development exam answers
    </a>
  </body>
</html>
```

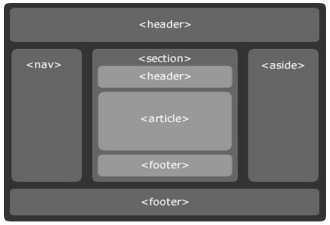
Agile Web Development exam answers

- When linking within the same page or same website use relative rather than absolute addressing, e.g. you can link to elements in the same document, use an id attribute:
- Links can point to any file, not just HTML.

```
<!DOCTYPE html>
<html>
  <body>
    <h2 id="Link"> Link to me! </h2>
    <a href="#Link"> linking... </a>
  </body>
</html>
```

HTML advanced layout elements

- The <header> element contains introductory information to a section or page.
- 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.
- 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.
- The <article> element represents a portion of a page which can stand alone such as: a blog post or a forum entry.
- The <aside> element represents content related to the main area of the document. Usually expressed in sidebars that contain elements like related posts, tag clouds.
- The <footer> element is for marking up the footer of, not only the current page, but each section contained in the page.




```
<!doctype html>
<html>
<head>
    <title>Page title</title>
</head>
<body>
    <header>
        <h1>Page title</h1>
    </header>
    <nav>
        <!-- Navigation -->
    </nav>
    <section id="intro">
        <!-- Introduction -->
    </section>
    <section>
        <!-- Main content area -->
    </section>
    <aside>
        <!-- Sidebar -->
    </aside>
    <footer>
        <!-- Footer -->
    </footer>

</body>
</html>
```

Span and div tags

- For various reasons it often useful to group arbitrary collections of elements, e.g. for common formatting, or drawing a box round them, etc.
- This the purpose of following two elements, which, by default, do not render to anything on the page:
  - - groups content on a single line, e.g. words in a sentence.
  - - groups content in a multi-line block, e.g. several paragraphs or images.

```
<!DOCTYPE html>
<html>
<body>
<div style="border-style:dotted">
  <p> Borders are cool... </p>
  <p> ... especially dotted ones </p>
</div>
<p>
  But not as cool as
  <span style="background-color:tomato">
    over-saturated highlighted
  </span>
  text.
</p>
</body>
</html>
```

Borders are cool...

... especially dotted ones

But not as cool as over-saturated highlighted text.

HTML forms

HTML form elements

- A `<form>` element is the standard way to get information from the browser to a server.
- Within a `<form>` you can add `<input>` elements to automatically create widgets that gather information (e.g., text buttons, radio buttons and checkboxes).

```
<!DOCTYPE html>
<html>
  <body>
    <form action="/action_page.php">
      <fieldset>
        <legend> Poll </legend>
        <label for="kitten-name"> What should the kitten be called? </label><br>
        <input type="text" id="kitten-name" name="answer" value="Whiskers"><br>
        <input type="submit" value="Submit">
      </fieldset>
    </form>
  </body>
</html>
```

Poll
What should the kitten be called?
Whiskers
Submit

- `<input>` elements can be grouped with `<fieldset>` element.
- Labels for `<input>` elements can be created using `<label>` elements and linked using the `for` attribute whose value should match the `id` attribute on the corresponding input.

HTML form input attributes

- The `type` attribute on the input element sets how it renders: button, checkbox, color, date, email, file, image, month, number, password, radio, range, reset, search, submit, tel, text, time, url, week etc.

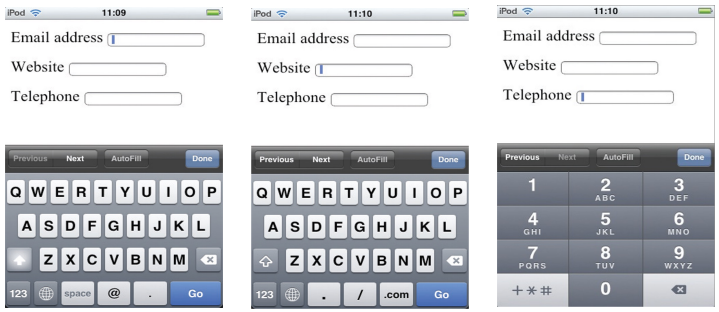
```
<!DOCTYPE html>
<html>
  <body>
    <form action="/action_page.php">
      Favourite colour: <input type="color" name="favcolor" value="#ff0000"> <br>
      Birthday: <input type="date" name="birthday"> <br>
      Email: <input type="email" name="email">
    </form>
  </body>
</html>
```

Favourite colour:
Birthday: dd/mm/yyyy
Email:

- The `value` attribute sets the input's initial value.
- The `readonly` attribute means the user can't edit it.
- The `placeholder` attribute provides a hint for what sort of value should be entered.
- The `required` attribute forces the user to enter a value.
- Lots of attributes are dependent on the type used, e.g. `autocomplete`, `min`, `max`

Using the right input type

- By using the correct semantic input type, browsers can make the user's life easier.
- For example, by using `email`, `url` and `tel` instead of a plain `text`, the on-screen keyboard changes accordingly:



Submitting a form

- Forms have a special input type called `submit` which generates a submission button.
- When the submission button of a form is clicked, the form's current values are sent to the server at the URL specified by the value of the `action` attribute of the `<form>` tag.
- Each input's value is sent as a key-value pair where the key is the value of `name` attribute.
- The `method` attribute on the <form> tag determines which protocol to use to transmit the values. The `get` method appends the form's values in the URL of the server request. The `post` method instead encodes the values in the body of the server request. See Lecture 9.

```
<!DOCTYPE html>
<html>
  <body>
    <form action="/action_page.php" method="get">
      Email: <input type="email" name="my-mail"> <br>
      <input type="submit">
    </form>
  </body>
</html>
```

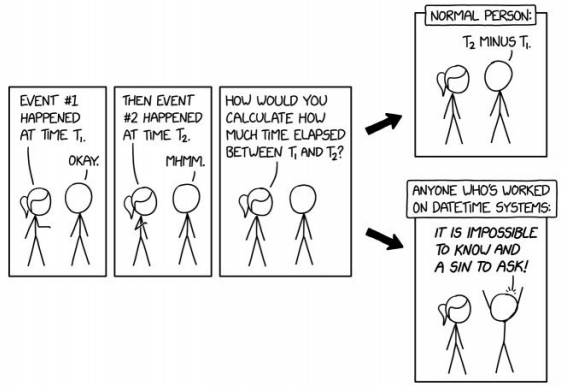
Email:
whiskers@gmail.com
Submit

- Will send the request `"/action_page.php?my-mail=whiskers@gmail.com"`

HTML form validation

- How to ensure the user enters valid data in the form?
- Approach 1: use the native validation provided by HTML.
  - Automatic basic validation on email, url and tel input types.
  - Various type-dependent validation attributes available:
    - For numeric inputs: `max`, `min`, `step`
    - For textual inputs: `maxlength`, `pattern`
- Approach 2: the more flexible approach is to use JavaScript in the browser to write code to check your inputs (see Lecture 5).
- Regardless of the approach, client-side validation is trivial to bypass! Therefore, always validate your values on the server as well!

HTML dates and times



HTML time element

- The `<time>` element can represent durations where the contents is the value shown to the user and the `datetime` attribute encodes length.
  - Prefix "P" for period, "D" for days, "H" for hours, "M" for minutes and "S" for seconds, e.g. <time datetime="P4D"> is a duration of 4 days,
  - using a "T" after the "P" marker allows you to be more precise, e.g. `<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 that the `<time>` element represents when the page was 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 datetime inputs

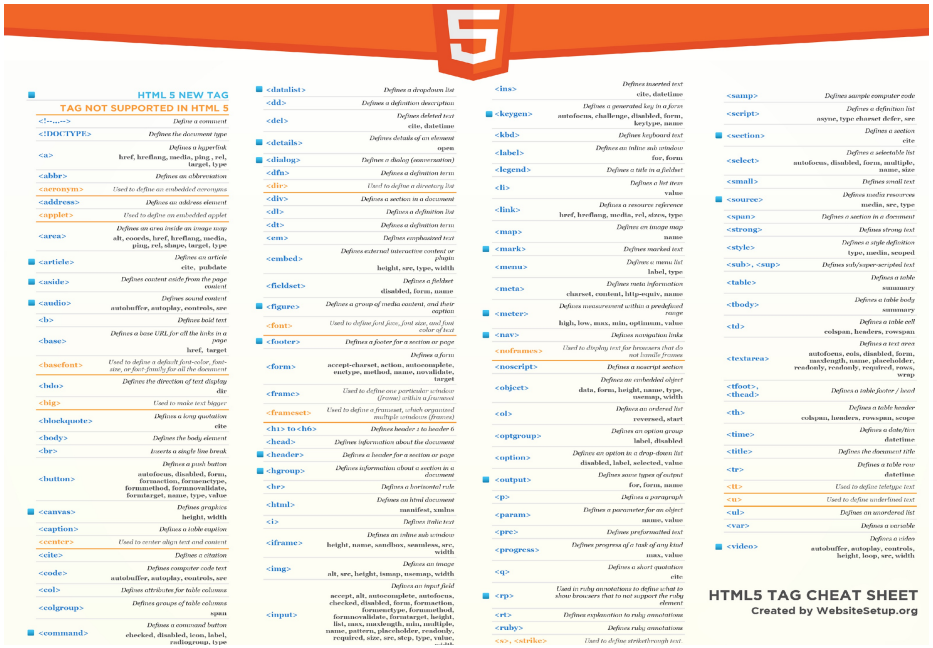
- HTML has various `<input>` types specifically for dates and times.
- These input types record timestamps with some subset of the standardized format

  YYYY-MM-DDThh:mm:ss.Z

  where Y = year, M = month, D = day, h = hour, m = minute, s = second, Z = timezone.
- Take, for example, the start time of Taylor Swift's Era's tour in Australia, 7:30pm on February 16th, 2024:
  - `datetime` - 2024-02-16T19:30:00.+11 - a year, month, and day in combination with hours, minutes, and seconds and time zone information.
  - `datetime-local` - 2024-02-16T19:30:00 - is the same but without the time zone information.
  - `date` - 2024-02-16 - a year, month, and day.
  - `month` - 2024-02 - a year and a month but without a day.
  - `time` - 19:30:00 - hours, minutes, and seconds.
- Weirdly there is no `year` type....

THE UNIVERSITY OF WESTERN AUSTRALIA

Useful HTML resources



HTML 5 NEW TAG

TAG NOT SUPPORTED IN HTML 5

<!--...--> Define a comment
<!DOCTYPE> Define the document type
<a> Defines a hyperlink. href, hreflang, media, ping, rel, target, type
<abbr> Defines an abbreviation
<acronym> Used to define an acronym
<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, width
<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 type 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

THE UNIVERSITY OF WESTERN AUSTRALIA