CITS1231 Web Technologies

Introduction to Cascading Style Sheets
The Problems with HTML – Poor Coding

• No consistency in the way a document is developed.

• The same markup can often be written a number of ways.

• One can omit attributes, or include irrelevant or illegal attributes, with the browser usually ignoring the problems. It seemingly encourages poorly written code.

  – <table border>
  – <table border="1">
  – <table>
  – <table bdr> (wrong syntax)
  – <table border="0">
## XHTML vrs HTML

<table>
<thead>
<tr>
<th>XHTML</th>
<th>HTML</th>
</tr>
</thead>
<tbody>
<tr>
<td>All elements explicitly closed eg:</td>
<td>Not necessary eg:</td>
</tr>
<tr>
<td>&lt;br /&gt;</td>
<td>&lt;br&gt;</td>
</tr>
<tr>
<td>All elements/attributes must be <em>lower case</em></td>
<td>Case insensitive</td>
</tr>
<tr>
<td>Must have quotes for attributes eg:</td>
<td>Quotes not necessary eg:</td>
</tr>
<tr>
<td>&lt;body lang=&quot;en&quot;&gt;</td>
<td>&lt;body lang=en&gt;</td>
</tr>
<tr>
<td>Necessary to have:</td>
<td>Not necessary</td>
</tr>
<tr>
<td>&lt;html&gt;, &lt;head&gt;, or &lt;body&gt;</td>
<td></td>
</tr>
</tbody>
</table>

**Recommendation:** Use XHTML.
The Problems with HTML – Coding Design

- As appearance and design became a central issue (with disregard to rules) new design tags were created.

- `<font>` is the classic example. It defines typographic presentation of the page. However this tag is not easy to use. Every time you change an attribute of the font, you must close and re-open the tag.

- You must open and close them every time you apply this tag to sections of your document. For example, to define the font to be used table cells, the definition must be repeated in each cell, it cannot be done as a global setting.
The Problems with HTML - Compatibility

• The initial goal of HTML was across platforms. Unfortunately this goal has been degraded.

• Distinct differences between the main browsers, and how they interpret HTML.

• Proprietary tags introduced due to browser war. For example:
  – `<blink>` only works for Netscape
  – `<marquee>` only works for Internet Explorer.
The Problems with HTML – Editors

- The worst abuses of HTML have subsequently found themselves coded into editor software. One needs to understand the limitations of the software one uses.

- A document created with three HTML editors, will produce three very different markups, but each giving rise to the same document. Many are unnecessarily long and complicated.
Solution: Separate Structure from Appearance

- There is a return to the old school of thought, one which requires a language that clearly separates the definition of a document’s structure, from the definition of a document’s appearance.

- This separation reduces to the design or appearance being a transformation using a style language operated over the structure.

- The standard language for doing this is “Cascading Style Sheets” (CSS), like HTML 4.0, CSS is an official W3C recommendation: http://www.w3.org/TR/REC-CSS2
Cascading Style Sheets (CSS)

\[
\text{<font face=“Courier” size=“6” color=“black”>some text</font>}
\]

\[
\text{<head>}
\text{<style type=“text/css”>}
\text{<!--}
\text{.defined {font-family: Courier; background-color:red; color:black; font-size: medium}}
\text{-->}
\text{</style>}
\text{</head>}
\text{<body>}
\text{<p><span class=“defined”>some text</span></p>}
\text{</body>}
\]
Document Presentation

- Markup comes in two varieties: markup that represents *logical* structure and markup that represents *presentational* structure.

- The logical structure influences but does not control the layout. One logical structure can result in multiple layouts: e.g. text-based terminal, computer screen, WebTV, paper, mobile phone ... even audio or Braille.

- Different layout are possible on the same output device.

- Multiple layout can be generated for presentation for a single marked up document.
Text terminal rendering
Rendering without CSS
What a difference CSS can make!
Types of CSS

- There are three ways of specifying styles:
  - External style sheet (multiple pages can link to this).
  - Internal style sheet (applies only to specify web page).
  - Inline style (applies only to the specific element).
Inline Styles

• Each browser has default display style.

• Inline styles can be used to customise how elements are displayed. Examples:

  <body style="background-color:yellow">
  <h1 style="text-align:center">
  <p style="font-family:courier new; color:red; font-size:20px">

Disadvantages of Inline Styles

• If all h2 headings need to use a certain font and alignment, then every time we specify a h2, we need to duplicate presentation attributes:

   <h2 style="text-align:center; font-size:10px"> blah blah </h2>

• Each time we want to change the style for h2, need to search and replace everywhere.

• For large web pages/sites this leads to complexity and maintenance issues. Could be tens or hundreds of occurrences across tens of web pages.

• How to cope with on-screen display versus printed

• How to display same page on mobile phone or text browser?

• Not easy to copy the presentation style from one web page to another.
Embedded Style

• They specify uniform styles for each element in given page.

• Are specified in the head of your HTML document:

```html
<head>
  <style type="text/css">
    h2 style="text-align:center; font-size:10px">
  -->
  </style>
</head>
```

• Older browsers do not support embedded style hence use of comments.
Style Rule Syntax

- Each style rule is of the form:

  \[
  \text{selector} \{ \text{property}: \text{value}; \text{property}: \text{value}; \ldots; \text{property}: \text{value}; \}
  \]

- Selectors determine which markup elements the style properties and values apply to. Example:
  
  \[
  \text{h1} \{ \text{color}: \text{white}; \text{background-color}: \text{orange} \}
  \]
Embedded Style

Advantages:

– Uniform display of each element type within a given web page.
– Easy to change – one location at head section.

Disadvantages:

– To make several web pages look same, need to duplicate the embedded style in each one.
External Style Sheet

- Style rules stored in a separate file (eg mystyle.css).
- Each page wishing to use this style links to this external file:
  
  ```html
  <head>
    <link rel="stylesheet" type="text/css" href="mystyle.css" />
  </head>
  
  </head>
  
  - External style sheet contains list of style rules.
  - Advantages:
    - Many pages can share same style easily.
    - Easy maintenance – just change external style sheet.
How does CSS work?

• CSS defines an extensive set of presentation properties, including color, font size, font family, margins, borders, and many more.

• Web browsers assume a default set of values for each property, which may vary from element to element. For example, browsers will render `<p>` text differently from `<em>` text or `<strong>` text.

• A CSS style sheet changes that default rendering by naming an element or set of elements, and the new property values:

  - `p {color: blue; font-size: small}`
What does it all mean?

• Each line in the style sheet looks something like:
  
  - p {color: blue}

• p is the selector. This is the element to apply the style to.

• color is the property.

• blue is the value that we wish to set the property to.
Cascading?

- Style sheets may have three different origins: author, user and browser.
  - The author specifies style sheets for a source document, defining a preferred presentation;
  - The user may also specify style information for a particular document;
  - The web browser must effectively provide a default style sheet.
- These rules combine together to form the final set of presentations.
Understanding Cascading Order

- You can link a single style sheet to multiple documents in your Web site by using the link element or the @import element.
- You can also link a single document to several style sheets.
Resolving Rule Conflict

1. By default, rules in user style sheets have higher priority than author style rules. Use !important to override.
   
   ```p {color: blue !important}
   ```

2. All user rules and author rules have more weight than the default supplied by the web browser.

3. If two rules of the same origin conflict, the more specific rule applies.

4. If the rules are equally specific, the last-specified rule is chosen.
Style precedence for one document

- Three ways to apply a style to an HTML or XHTML document:
  - Inline Styles
  - Embedded Styles
  - External Styles

- Style precedence
  - Inline > embedded > external
Inheritance

- Property values are normally *inherited*.

- Suppose there is a `<h1>` element with an emphasizing element (`<em>`) inside:
  - `<h1>The headline <em>is</em> important!</h1>

- If no color has been assigned to `<em>` element, the emphasized “is” will inherit the color of the parent element, so if `<h1>` has the color blue, the content enclosed in the `<em>` element will be in blue as well.
HTML <div> Tag

- The <div> element, to mark logical blocks within a document.
- Used to group block-elements to format them with styles.

```html
<div style="color:red">
  <h3>This is a header</h3>
  <p>This is a paragraph.</p>
</div>
```
HTML <span> Tag

• Used to mark sub-sequences of text.
• Styles can be applied to text marked with <span>:

   <p>
     I have <span style=color:brown;">brown</span> eyes and my wife has <span style="color:blue;">blue</span> eyes.
   </p>
HTML class Attribute

- Marks scattered members of a particular conceptual group.

```html
<html>
<head>
<style type="text/css">
span.blue {color:lightskyblue; font-weight:bold;}
span.brown {color:brown; font-weight:bold;}
</style>
</head>
<body>
<p> I have <span class="brown">brown</span> eyes and my wife has <span class="blue">blue</span> eyes. </p>
</body>
</html>
```