CSC 551: Web Programming

Spring 2004

Basic HTML

- hypertext
- tags & elements
- text formatting
- lists, hyperlinks, images
- tables, frames

- cascading style sheets
  - inline, document, external

Hypertext & HTML

HyperText Markup Language (HTML) is the language for specifying the *static* content of Web pages

- *hypertext* refers to the fact that Web pages are more than just text
  can contain multimedia, provide links for jumping within & without
- *markup* refers to the fact that it works by augmenting text with special symbols (tags) that identify structure and content type

HTML is an evolving standard (as new technology/tools are added)

- HTML 1 (Berners-Lee, 1989): very basic, limited integration of multimedia
  in 1993, Mosaic added many new features (e.g., integrated images)
- HTML 2.0 (IETF, 1994): tried to standardize these & other features, but late
  in 1994-96, Netscape & IE added many new, divergent features
- HTML 3.2 (W3C, 1996): attempted to unify into a single standard
  but didn't address newer technologies like Java applets & streaming video
- HTML 4.0 (W3C, 1997): current standard
  attempted to map out future directions for HTML, not just react to vendors
- XHTML 1.0 (W3C, 2000): HTML 4.01 modified to conform to XML standards

Web development tools

many high-level tools exist for creating Web pages

e.g., Microsoft FrontPage, Netscape Composer, Adobe PageMill,
Macromedia DreamWeaver, HotDog, …
also, many applications have "save to HTML" options (e.g., Word)

for most users who want to develop basic, static Web pages, these are fine

assembly language vs. high-level language analogy

so, why are we learning low-level HTML using a basic text editor?

- may want low-level control
- may care about size/readability of pages
- may want to "steal" page components and integrate into existing pages
- may want dynamic features such as scripts or applets

Tags vs. elements

HTML specifies a set of *tags* that identify structure and content type

- tags are enclosed in < >

  `<img src="image.gif" />` specifies an image

- most tags come in pairs, marking a beginning and ending

  `<title>` and `</title>` enclose the title of a page

an HTML *element* is an object enclosed by a pair of tags

`<title>My Home Page</title>` is a TITLE element

`<b>This text appears bold.</b>` is a BOLD element

`<p>Part of this text is <b>bold</b>.</p>`
is a PARAGRAPH element that contains a BOLD element

HTML document is a collection of elements (text/media with context)

Structural elements

an HTML document has two main structural elements

- HEAD contains setup information for the browser & the Web page
  e.g., the title for the browser window, style definitions, JavaScript code, …
- BODY contains the actual content to be displayed in the Web page

```
<html>
<!-- Dave Reed  page01.html  1/16/04 -->
<!-- Demo web page                   -->

<head>
  <title>Title for Page</title>
</head>

<body>
  Text that appears in the page
</body>

</html>
```

HTML documents begin and end with <html> and </html> tags

Comments appear between <!-- and -->

HEAD section enclosed between <head> and </head>

BODY section enclosed between <body> and </body>

view page in browser

Text layout

```
<html>
<!-- Dave Reed  page02.html  1/16/04 -->
<!-- Demo web page                   -->

<head>
  <title>Text Layout</title>
</head>

<body>
  <p>
  This is a paragraph of text<br/>
  made up of two lines.
  </p>

  <p>
  This is another paragraph with a
  &nbsp; GAP &nbsp; between
  some of the words.
  </p>

  <p>
  &nbsp;&nbsp; This paragraph is<br/>
  indented on the first line<br/>
  but not on subsequent lines.
  </p>
</body>

</html>
```

for the most part, layout of the text must be left to the browser

- every sequence of whitespace is interpreted as a single space
- browser automatically wraps the text to fit the window size

can override some text layout

- can cause a line break using the <br/> tag (no closing tag)
- can specify a new paragraph (starts on a new line, preceded by a blank line) using <p>…</p>
- can force a space character using the symbol for a non-breaking space: &nbsp;

view page in browser

Separating blocks of text

```
<html>
<!-- Dave Reed  page03.html  1/16/04 -->
<!-- Demo web page                   -->

<head>
  <title>Blocks of Text</title>
</head>

<body>
  <h1>Major heading 1</h1>
  <p>
  Here is some text.
  </p>

  <h2>Subheading</h2>
  <p>
  Here is some subtext.
  </p>

  <hr/>

  <h1>Major heading 2</h1>
  <p>
  Here is some more text.
  </p>
</body>

</html>
```

can specify headings for paragraphs or blocks of text

- `<h1>…</h1>` tags produce a large, bold heading
- `<h2>…</h2>` tags produce a slightly smaller heading

  . . .
- `<h6>…</h6>` tags produce a tiny heading

can insert a horizontal rule to divide sections

- `<hr/>` draws line across window
- `<hr width="50%" />` sets width
- `<hr size=10 />` sets thickness

view page in browser

Aligning text

```
<html>
<!-- Dave Reed  page04.html  1/16/04 -->
<!-- Demo web page                   -->

<head>
  <title>Text Alignment</title>
</head>

<body>
  <h1 style="text-align:center">Centered Heading</h1>
  <p>
  Here is some left-justified text
  (which is the default in HTML).
  </p>

  <p style="text-align:center">
  Here is some centered text.
  </p>

  <div style="text-align:right">
  <h2>Right-justified Heading</h2>
  <p>Here is some right-justified text.</p>
  </div>
</body>

</html>
```

can specify how elements should be aligned (default is left-justified)

- utilize STYLE attribute of tag

to justify more than one element as a group, use DIV tags

- ell elements enclosed in DIV are formatted similarly

view page in browser

Example Web page

consider the CSC 551 Home Page

- page title and section information is centered
- paragraphs/sections for different topics
  (course description, required work, collaboration policy, schedule)
- headings to identify each section
- breaks and spaces used to align text, improve layout

Text styles

```
<html>
<!-- Dave Reed  page05.html  1/16/04 -->
<!-- Demo web page                   -->

<head>
  <title>Text Styles</title>
</head>

<body>
  <p>
  Text can be emphasized using
  <b>bold</b>, <i>italics</i>, or even
  <big>resizing</big>.  <br/>
  The typewriter font is good for
  displaying code:
  <tt>sum = sum + i;</tt>  <br />
  And remember: <span style="color:red">
  <small>2<sup>10</sup></small> =
  1024</span>
  </p>
</body>

</html>
```

view page in browser

can specify styles for fonts

- `<b>… </b>` specify bold
- `<i>… </i>` specify italics
- `<tt>… </tt>` specify typewriter-like (fixed-width) font

- `<big>… </big>` increase the size of the font
- `<small>… </small>` decrease the size of the font

- `<sub>… </sub>` specify a subscript
- `<sup>… </sup>` a superscript

- `<p style="color:red">…</p>` for paragraphs
- `<span style="color:blue">…</span>` for inline text

Note: if elements are nested, the order of opening/closing is important! (LIFO)

More text grouping

```
<html>
<!-- Dave Reed  page06.html  1/16/04 -->

<head>
  <title>More Text Grouping</title>
</head>

<body>
  <p>
  <tt><pre>
    for (i = 0; i < 10; i++) {
        sum = sum + i;
    }
  </pre></tt>
  </p>

  <p>
  Eagleson's Law states that:
  <blockquote>
  Any code of your own that you haven't
  looked at for six or more months
  might as well have been written by
  someone else.
  </blockquote>
  </p>
</body>

</html>
```

- `<pre>…</pre>` specify text that is to be displayed as is (line breaks and spacing are preserved)

  useful for code or whenever you want text to fit a specific layout

- `<blockquote>…</blockquote>` specify text that is to be indented on both margins

  useful for quotations or for indenting text in subsections

view page in browser

Lists

```
<html>
<!-- Dave Reed  page07.html  1/16/04 -->

<head>
  <title>Simple Lists</title>
</head>

<body>
  <p>
  <ol>
    <li>First thing.
    <li>Second thing.
    <li>Third thing.
  </ol>
  </p>

  <p>
  <dl>
    <dt>HTML
    <dd>HyperText Markup Language
    <dt>HTTP
    <dd>HyperText Transfer Protocol
  </dl>
  </p>
</body>

</html>
```

there are 3 different types of list elements

- `<ol>…</ol>` specifies an ordered list (using numbers or letters to label each list item)
  `<li>` identifies each list item

  can set type of ordering, start index

- `<ul>…</ul>` specifies unordered list (using a bullet for each)
  `<li>` identifies each list item

- `<dl>…</dl>` specifies a definition list
  `<dt>` identifies each term
  `<dd>` identifies its definition

view page in browser

Hyperlinks

```
<html>
<!-- Dave Reed  page08.html  1/16/04 -->

<head>
  <title>Hyperlinks</title>
</head>

<body>
  <p>
  <a href="http://www.creighton.edu">
  Creighton University</a>
  <br>
  <a href="page07.html" target="_blank">
  Open page07 in a new window</a>
  </p>
</body>

</html>
```

view page in browser

perhaps the most important HTML element is the hyperlink, or ANCHOR

- `<a href="URL">…</a>`

  where URL is the Web address of the page to be displayed when the user clicks on the link

  if the page is accessed over the Web, must start with `http://`

  if not there, the browser will assume it is the name of a local file

- `<a href="URL" target="_blank">…</a>`

  causes the page to be loaded in a new window

Hyperlinks (cont.)

```
<html>
<!-- Dave Reed  page09.html  1/16/04 -->

<head>
  <title>Internal Links in a Page</title>
</head>

<body>
  <p align="center">
  [ <a href="#HTML">HTML</a> |
    <a href="#HTTP">HTTP</a> |
    <a href="#IP">IP</a> |
    <a href="#TCP">TCP</a> ]
  </p>
  <p>
  Computer acronyms:
  <dl>
    <a name="HTML"></a><dt>HTML
    <dd>HyperText Markup Language
    <a name="HTTP"></a><dt>HTTP
    <dd>HyperText Transfer Protocol
    <a name="IP"></a><dt>IP
    <dd>Internet Protocol
    <a name="TCP"></a><dt>TCP
    <dd>Transfer Control Protocol
  </p>
</body>
</html>
```

for long documents, you can even have links to other locations in that document

- `<a name="ident">…</a>`

  where ident is a variable for identifying this location

- `<a href="#ident">…</a>`

  will then jump to that location within the file

- `<a href="URL#ident">…</a>`

  can jump into the middle of another file just as easily

view page in browser

Images

can include images using `IMG`

- by default, browsers can display GIF and JPEG files
- other image formats may require plug-in applications for display

<img src="*filename*" alt="*alternate text*" />

again, if file is to be accessed over the Web, must start with http:// (if not, will assume local file)

```
<html>
<!-- Dave Reed  page10.html  1/16/04 -->

<head>
  <title>Images</title>
</head>

<body>
  <div style="text-align:center">
    <img src="http://www.creighton.edu/~davereed/Images/reed.gif"
         alt="Dave Reed" />
  <p>Dave Reed</p>
  </div>
</body>
</html>
```

view page in browser

Example Web page revisited

consider the CSC 551 Home Page again

- bold, italics, and font size changes are used throughout to embellish text
- does not have colored fonts (use sparingly!)
- uses an unordered list to present the course goals
- has links to University page, department page, instructor page
- links to lecture notes bring up a new window

Tables

tables are common tools for arranging complex layout on a Web page

- a table divides contents into rows and columns
- by default, column entries are left-justified, so provide for alignment

```
<html>
<!-- Dave Reed  page11.html  1/16/04 -->

<head>
  <title>Tables</title>
</head>

<body>
  <table>
    <tr>
      <td>foo</td> <td>bar</td>
    </tr>
    <tr>
      <td>bizbaz</td> <td>booboo</td>
    </tr>
  </table>
</body>
</html>
```

`<table>…</table>` specify a table element

`<tr>…</tr>` specify a row in the table

`<td>…</td>` specify table data (i.e., each column entry in the table)

view page in browser

Layout in a table

```
<html>
<!-- Dave Reed  page12.html  1/16/04 -->

<head>
  <title>Table Layout</title>
</head>

<body>
  <table border=1>
    <tr align="center">
      <td>foo<br>foo</td>
      <td valign="top">bar</td>
    </tr>
    <tr>
      <td>bizbaz</td>
      <td>booboo</td>
    </tr>
  </table>
</body>
</html>
```

view page in browser

can have a border on tables using the BORDER attribute

```
<table border=1>
```

increasing the number makes the border thicker

can control the horizontal & vertical layout within cells

```
<td align="center">
<td align="right">

<td valign="top">
<td valign="bottom">
```

can apply layout to an entire row

```
<tr align="center">
<tr valign="top">
```

Table width

```
<html>
<!-- Dave Reed  page13.html  1/16/04 -->

<head>
  <title>Table Width</title>
</head>

<body>
  <table width="100%">
    <tr>
      <td>left-most
      <td align="right">right-most</td>
    </tr>
</table>
</body>
</html>
```

view page in browser

by default, the table is sized to fit the data

can override & specify the width of a table relative to the page

```
<table width="60%">
```

useful for page footer –
set table width to 100%
1st column: left-justified
2nd column: right-justified

Other table options

```
<html>
<!-- Dave Reed  page14.html  1/16/04 -->

<head>
  <title>Table Formatting</title>
</head>

<body>
  <table border=1 cellspacing=4 cellpadding=8>
    <tr>
      <th>HEAD1</th> <th>HEAD2</th> <th>HEAD3</th>
    </tr>
    <tr>
      <td>one</td> <td>two</td> <td>three</td>
    </tr>
    <tr>
      <td rowspan=2 align="center"> four </td>
      <td colspan=2 align="center"> five </td>
    </tr>
    <tr>
      <td> six </td> <td> seven </td>
    </tr>
</table>
</body>
</html>
```

can control the space between cells & margins within cells

```
<table cellspacing=5>
<table cellpadding=5>
```

can add headings

`<th>` is similar to `<td>` but displays heading centered in bold

can have data that spans more than one column

```
<td colspan=2>
```

similarly, can span more than one row

```
<td rowspan=2>
```

view page in browser

Example Web page revisited

consider the CSC 551 Home Page again

- section names, times, and rooms are aligned (3-column table)
- office hours are aligned, with "Office hours:" to the left (2-column table)
- grading categories in a table with a border
- class schedule in a big table
  - week number centered in first column
  - weekly topic displayed as a table within a table
  - spring break & final exam are multi-column
- file name and date are displayed as footer

Frames

frames provide the ability to split the screen into independent pages

- must define a FRAMESET that specifies the layout of the pages
- actual pages to be displayed must be in separate files

```
<html>
<!-- Dave Reed   page15.html   1/16/03 -->

<frameset cols="*,*">
  <frame src="page01.html">
  <frame src="page02.html">
</frameset>

</html>
```

view page in browser

can divide vertically

```
<frameset cols="50%,50%">
```

or, horizontally

```
<frameset rows="30%,*,*">
```

** causes the browser to divide the remaining space evenly*

by default, each frame scrollable

- can drag the border to resize
- can hide the border with frameset attribute `frameborder=0`

Frame controversy

frames are probably the most controversial HTML feature

- some people love them, some people hate them

2 reasonable uses for frames

- as a navigational aid:

  can divide the screen into a static menu frame and the main frame for navigating a site

  e.g., www.creighton.edu/~davereed

- as a means of separating program input from output:

  can divide the screen into a static man input form frame and the main frame for displaying output

  e.g., www.creighton.edu/~davereed/csc551/JavaScript/story.html

Menu frame

to create a menu, need to be able to direct links to the main frame

- name the frames in the FRAMESET
- specify the frame name as TARGET in the link
- specify _top as target to return to top level of browser

```
<html>
<!-- Dave Reed  page16.html  1/16/04 -->

<head>
  <title>Demo Browser</title>
</head>

<frameset cols="30%,*">
    <frame src="menu16.html" name="menu">
    <frame src="page01.html" name="main">
</frameset>

</html>
```

view page in browser

```
<html>
<!-- Dave Reed menu16.html 1/16/04 -->

<head>
  <title>Menu of Demos</title>
</head>

<body>
Links to demo pages

<p>
<a href="page01.html"
   target="main">Demo 1</a><br/>
<a href="page02.html"
   target ="main">Demo 2</a><br/>
<a href="page03.html"
   target ="main"> Demo 3</a><br/>
<a href="page04.html"
   target ="main"> Demo 4</a><br/>
<a href="page05.html"
   target ="main"> Demo 5</a><br/>
<a href="page06.html"
   target ="main"> Demo 6</a><br/>
<a href="http://www.creighton.edu"
   target="_top" >Creighton</a>
</p>
</body>
</html>
```

Content vs. presentation

most HTML tags define content type, independent of presentation

- exceptions?

style sheets associate presentation formats with HTML elements

- CSS1: developed in 1996 by W3C
- CSS2: released in 1998, but not fully supported by browsers

- HTML style sheets are known as *Cascading Style Sheets*, since can be defined at three different levels

  1. *inline* style sheets apply to the content of a single HTML element
  2. *document* style sheets apply to the whole BODY of a document
  3. *external* style sheets can be linked and applied to numerous documents

  lower-level style sheets can override higher-level style sheets

Inline style sheets

```
<html>
<!-- Dave Reed  page17.html  1/16/04 -->

<head>
  <title>Inline Style Sheets</title>
</head>

<body>
  <p style="font-family:Arial,sans-serif;
            text-align:right">This is a
  right-justified paragraph in a sans serif
  font (preferably Arial), with some
  <span style="color:green">green text</span>.
  </p>

  <p>And <a style="color:red;
                   text-decoration:none;
                   font-size:larger;"
         href="page01.html">here</a>
     is a formatted link.
  </p>
</body>
</html>
```

view page in browser

Using the `style` attribute, can specify presentation style for a single HTML element

- within tag, list sequence of property:value pairs

font-family:Courier,monospace
font-style:italic
font-weight:bold
font-size:12pt font-size:large font-size:larger

color:red color:#000080
background-color:white

text-decoration:underline
 text-decoration:none
text-align:left text-align:center
 text-align:right text-align:justify
vertical-align:top vertical-align:middle
 vertical-align:bottom

text-indent:5em text-indent:0.2in

Inline style sheets (cont.)

```
<html>
<!-- Dave Reed  page18.html  1/16/04 -->

<head>
  <title>Inline Style Sheets</title>
</head>

<body>
  <p>Here is an image
    <img src="reed.gif" alt="Dave Reed"
         style="margin-left:0.3in;
                margin-right:0.3in;
                vertical-align:middle;
                border-style:double;
                border-color:yellow">
    embedded in text.
  </p>

  <ol style="list-style-type:upper-alpha">
    <li> one thing
    <li> or another
    <ul style="list-style-type:square;
               whitespace:pre">
      <li> with  this
      <li> or    that
    </ul>
  </ol>
</body>
</html>
```

more style properties & values

margin-left:0.1in margin-right:5%
margin:3em
padding-top:0.1in padding-bottom:5%
padding:3em

border-width:thin border-width:thick
border-width:5
border-color:red
border-style:dashed border-style:dotted
border-style:double border-style:none

whitespace:pre

list-style-type:square
list-style-type:decimal
list-style-type:lower-alpha
list-style-type:upper-roman

view page in browser

Inline style sheets (cont.)

```
<html>
<!-- Dave Reed  page19.html  1/16/04 -->

<head>
  <title> Inline Style Sheets </title>
</head>

<body>
  <table style="font-family:Arial,sans-serif">
    <caption style="color:red;
                    font-style:italic;
                    text-decoration:underline">
       Student data. </caption>
    <tr style="background-color:red">
      <th> name </th> <th> age </th>
    </tr>
    <tr>
      <td> Chris Smith </td> <td> 19 </td>
    </tr>
    <tr>
      <td> Pat Jones </td> <td> 20 </td>
    </tr>
    <tr>
      <td> Doogie Howser </td> <td> 9 </td>
    </tr>
</table>
</body>
</html>
```

style sheets can be applied to tables for interesting effects

view page in browser

Document style sheets

inline style sheets apply to individual elements in the page

- can lead to inconsistencies as similar elements are formatted differently
  e.g., might like for all H1 elements to be centered
- inline definitions mix content & presentation
  ➔ violates the general philosophy of HTML

alternatively, document style sheets allow for a clean separation of content and presentation

- style definitions are placed in the HEAD of the page (within STYLE tags)
- can apply to all elements, or a subclass of elements, throughout the page

Document style sheets

```
<html>
<!-- Dave Reed  page20.html  1/16/04 -->

<head>
  <title>Document Style Sheets</title>
  <style type="text/css">
     h1 {color:blue;
         text-align:center}
     p.indented {text-indent:0.2in}
  </style>
</head>

<body>
  <h1>Centered Title</h1>

  <p class="indented">This paragraph will
have the first line indented, but
subsequent lines will be flush.</p>

  <p>This paragraph will not be indented.
  </p>

 <h1>The End</h1>

</body>
</html>
```

document style sheets ensure that similar elements are formatted similarly

- can even define subclasses of elements and specify formatting

`p.indented` defines subclass of paragraphs
- inherits all defaults of <p>
- adds new features

to specify this newly defined class, place class="ID" attribute in tag

note how "clean" the BODY is

view page in browser

Document style sheets (cont.)

```
<html>
<!-- Dave Reed  page21.html  1/16/04 -->

<head>
  <title> Inline Style Sheets </title>
 <style type="text/css">
     table {font-family:Arial,sans-serif}
     caption {color:red;
              font-style:italic;
              text-decoration:underline}
     th {background-color:red}
  </style>
</head>

<body>
  <table>
    <caption> Student data. </caption>
    <tr><th> name </th>          <th> age</th></tr>
    <tr><td> Chris Smith </td>   <td> 19 </td></tr>
    <tr><td> Pat Jones </td>     <td> 20 </td></tr>
    <tr><td> Doogie Howser </td> <td> 9 </td></tr>
</table>
</body>
</html>
```

view page in browser

document style sheets are especially useful in formatting tables

effectively separates content from presentation

what if you wanted to right-justify the column of numbers?

what if you changed your mind?

Pseudo-elements

```
<html>
<!-- Dave Reed  page22.html  1/16/04 -->

<head>
  <title>Title for Page</title>
  <style type="text/css">
      a {color : red;
         text-decoration : none;
         font-size : larger}
      a:visited {color : black}
      a:active {color : orange}
      a:hover {color : blue}
      p:first-letter {font-size : large;
              color : white;
              background-color : darkblue}
  </style>
</head>

<body>
  <p>Welcome to my Web page.  I am so
  happy you are here.
  </p>
  <p>Be sure to visit
  <a href="http://www.cnn.com">CNN</a>
  for late-breaking news.
  </p>
</body>
</html>
```

pseudo-elements are used to address sub-parts of elements

- can specify appearance of link in various states
  `:visited    :active    :hover`
- can specify format of first line in page or paragraph
  `:first-line`
- can specify format of first letter in page or paragraph
  `:first-letter`

Danger : changing the look of familiar elements is confusing

Careful : current browsers do not support all CSS2 features

view page in browser

External style sheets

modularity is key to the development and reuse of software

- design/implement/test useful routines and classes
- package and make available for reuse

- saves in development cost & time
- central libraries make it possible to make a single change and propogate

external style sheets place the style definitions in separate files

- multiple pages can link to the same style sheet, consistent look across a site
- possible to make a single change and propagate automatically

- represents the ultimate in content/representation separation

Modularity & style sheets

```
<html>
<!-- Dave Reed  page23.html  1/16/04 -->

<head>
  <title>Title for Page</title>
  <link rel="stylesheet"
        type="text/css"
        href="myStyle.css"
        title="myStyle">
</head>

<body>
 <h1>Centered Title</h1>

 <p class="indented">This paragraph will
have the first line indented, but
subsequent lines will be flush.</p>

 <p>This paragraph will not be indented.
 </p>

 <h1>The End</h1>

</body>
</html>
```

```
<!-- myStyle.css  Dave Reed  1/16/04 -->
h1 {color : blue; text-align : center}
p.indented {text-indent:0.2in}
```

ideally, the developer(s) of a Web site would place all formatting options in an external style sheet

all Web pages link to that same style sheet for a uniform look

- simplifies Web pages since only need to specify structure/content tags

view page in browser

Web rules of thumb

HTML provides for lots of neat features,
but just because you can add a feature doesn't mean you should!

don't add features that distract from the content of the page

- use color & fonts sparingly and be careful how elements fit together
  e.g, no purple text on a pink background, no weird fonts
- use images only where appropriate
  e.g., bright background images can make text hard to read
  e.g., the use of clickable images instead of buttons or links can slow access
- don't rely on window or font size for layout
  e.g., font size may be adjusted by viewer, window constrained
- don’t be annoying
  e.g., no pop-up windows, excessive advertising, silly music
- break large document into smaller or provide a menu (either internal or frame)
- stick to standard features and test using both IE and Netscape
- utilize style sheets to make changes easy & ensure consistency