

CSC 551: Web Programming

Fall 2001

more HTML

- layout/positioning with tables
- layout/positioning with style sheets
- frames
- images
- more on style sheets

Last week...

basic HTML

- page structure (html, head, body)
- text layout (br, , center, align)
- text grouping (p, pre, blockquote, span)
- text separation (h1...h6, hr)
- text styles (b, i, tt, big, small, sub, sup, font, color)
- lists (ol, ul, dl)
- links (a, href, name)
- images (img)
- style sheets

this week...

- more complex layout with tables, style sheets, frames
- better integration of images

Tables

prior to style sheets, tables were the main technique 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  html18.html  8/22/01 -->

<head>
  <title>Title for Page</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](#)

Tables with borders

```
<html>
<!-- Dave Reed  html19.html  18/22/01 -->

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

<body>
  <table border=1 rules="rows">
    <tr>
      <td>foo</td> <td>bar</td>
    </tr>
    <tr>
      <td>bizbaz</td> <td>booboo</td>
    </tr>
  </table>
</body>
</html>
```

can have a border on tables using the **BORDER** attribute

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

increasing the number makes the border thicker

can be selective with the border

frame attribute controls the border type

```
<table border=1 frame="hsides">
<table border=1 frame="lhs">
```

rules attribute controls internal borders

```
<table border=1 rules="none">
<table border=1 rules="rows">
```

[view page in browser](#)

Example Web page

Consider the [CSC 551 Home Page](#)

- 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

Layout in a table

```
<html>
<!-- Dave Reed  html20.html  8/22/01 -->

<head>
<title>Title for Page</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 control the layout within specific rows and/or columns

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

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

for multiline columns, can control vertical alignment

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

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

can apply layout to an entire row

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

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

Table width

```
<html>
<!-- Dave Reed  html21.html  8/22/01 -->

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

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

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*

[view page in browser](#)

Spanning rows and columns

```
<html>
<!-- Dave Reed  html22.html  8/22/01 -->

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

<body>
  <table border=1>
    <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 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](#)

Other table options

```
<html>
<!-- Dave Reed  html23.html  8/22/01  -->

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

<body>
  <table border=1 cellpadding=4
        cellspacing=8 bgcolor="green">
    <caption align="bottom">
      Student data.
    </caption>
    <tr bgcolor="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>
  </table>
</body>
</html>
```

can add a caption to a table

```
<caption align="bottom">
```

can add headings

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

can control the space between cells & margins within cells

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

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

can change the background color using the BGCOLOR attribute

[view page in browser](#)

Example Web page revisited

Consider the [CSC 551 Home Page](#) again

- week number centered in first column
- weekly topic displayed as a table within a table
- midterm & final exam are multi-column
- file name and date are displayed as footer

Tables & style sheets

```
<html>
<!-- Dave Reed  html24.html  8/22/01 -->

<head>
  <title>Title for Page</title>
  <style type="text/css">
    table {font : Arial}
    th {background-color : red}
    caption {color : red;
             font-style : italic;
             vertical-align : bottom}
  </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>
  </table>
</body>
</html>
```

style sheets can be combined with tables to define general layout/presentation style

again, the advantage is in separating presentation info and content info

[view page in browser](#)

Absolute positioning with style sheets

```
<html>
<!-- Dave Reed  html25.html  8/22/01 -->

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

<body>
  <div style="position:absolute;
            top:20; left:20">
    
  </div>

  <p style="position:absolute;
          top:120; left:30;
          font-weight:bold">
    Dave Reed
  </p>
</table>
</body>
</html>
```

style sheets also provide the capability of placing elements at precise coordinates

```
<div style="position:absolute;
          top:20; left:20">
```

```
<p style="position:absolute;
        top:120; left:25">
```

can even overlay elements

e.g. text on top of an image

dangers?

[view page in browser](#)

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  html26.html  8/22/01  -->

<frameset cols="*,*">
  <frame src="html24.html">
  <frame src="html25.html">
</frameset>

</html>
```

[view page in browser](#)

can divide vertically

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

or, can divide horizontally

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

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

Frame defaults

by default, each frame is an independent page, scrollable

- the relative size of the frames can be changed by dragging the border in between

```
<html>
<!-- Dave Reed  html27.html  8/22/01  -->

<frameset rows="35%,*"  frameborder=0>
  <frame src="html24.html">
  <frame src="html25.html">
</frameset>

</html>
```

[view page in browser](#)

can specify whether you want a border

```
frameborder=1  (default)
```

```
frameborder=0  no border
```

can specify whether you want scrolling

```
scrolling="auto"  (default)
```

```
scrolling="yes"  always
                  displays scroll bars
```

```
scrolling="no"  never
```

Nested frames

can even nest frames to obtain more complex layouts of pages

- simply nest FRAMESET elements

```
<html>
<!-- Dave Reed  html28.html  8/22/01  -->

<frameset rows="40%,*">
  <frameset cols="30%,*">
    <frame src="reed.gif">
    <frame src="html24.html">
  </frameset>
  <frame src="html23.html">
</frameset>

</html>
```

[view page in browser](#)

if the frame is simply to hold an image, can specify the image file as SRC

```
<frame src="foo.gif">
```

important questions:

why have a frame that just stores an image?

are frames really worth it?

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/~csc551/CGI/grades.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

```
<html>
<!-- Dave Reed  html29.html  8/22/01  -->

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

<frameset cols="30%,*">
  <frame src="menu29.html"  name="menu">
  <frame src="html01.html"  name="main">
</frameset>

</html>
```

[view page in browser](#)

```
<html>
<!-- Dave Reed  menu29.html  8/22/01  -->

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

<body>
  Links to demo pages

  <p>
  <a href="html01.html"
    target="main">html 1</a><br>
  <a href="html02.html"
    target ="main">html 2</a><br>
  <a href="html03.html"
    target ="main">html 3</a><br>
  <a href="html04.html"
    target ="main">html 4</a><br>
  <a href="html05.html"
    target ="main">html 5</a><br>
  <a href="html06.html"
    target ="main">html 6</a>
  </p>
</body>
</html>
```

Leaving frames

there are times you want to break out of frames

e.g., have a link in the menu to an outside page

- can specify "_top" as TARGET

```
<html>
<!-- Dave Reed  html30.html  8/22/01  -->

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

<frameset cols="30%,*">
  <frame src="menu30.html"  name="menu">
  <frame src="html01.html"  name="main">
</frameset>

</html>
```

[view page in browser](#)

```
<html>
<!-- Dave Reed  menu30.html  8/22/01  -->

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

<body>
  Links to demo pages

  <p>
  <a href="html01.html"
    target="main">html 1</a><br>
  <a href="html02.html"
    target="main">html 2</a><br>
  <a href="html03.html"
    target="main">html 3</a><br>
  <a href="html04.html"
    target="main">html 4</a><br>
  <a href="html05.html"
    target="main">html 5</a><br>
  <a href="html06.html"
    target="main">html 6</a><br>
  <br>
  <a href="http://www.creighton.edu"
    target="_top">Creighton</a>
  </p>
</body>
</html>
```

Handy frame trick

in fact, you may see the following in the HEAD of Web pages

```
<script language="JavaScript">
  if (self != top) {
    top.location = self.location;
  }
</script>
```

this is JavaScript code that

- looks at the current location (self)
 - sees if it is on top in the browser (i.e., not embedded in a frame)
 - if not, then it sets the top location to the current location
- ➔ ensures that the page is displayed at the top level (not embedded in frames)

note: causes some problems with BACK button in browsers

Images revisited

recall from last week,

- an image can be placed in a Web page using the IMG tag

```
<html>
<!-- Dave Reed  html31.html  8/22/01  -->

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

<body>
  <center>
    
  </center>
</body>
</html>
```

it is good practice to always define the ALT attribute for an image

- ALT specifies text that will be displayed in the place of the image if it is not viewable or while it is loading

when is this needed?

[view page in browser](#)

Image resizing

you can resize an image by specifying its height & width in pixels

```
<html>
<!-- Dave Reed  html32.html  8/22/01 -->
<head>
  <title>Title for Page</title>
</head>
<body>
  <center>
    
  </center>
</body>
</html>
```

[view page in browser](#)

be very careful, since resizing may mess up aspect ratio

- first, determine image dimensions
can right-click on an image, then select properties to determine its width x height
- then, resize proportionately

Image layout

```
<html>
<!-- Dave Reed  html33.html  8/22/01 -->
<head>
  <title>Title for Page</title>
</head>
<body>
  <p>
    
    text aligns with bottom of image (default)
  </p>
  <p>
    
    text aligns with center of image
  </p>
  <p>
    
    image is right-justified, text wraps
  </p>
</body>
</html>
```

[view page in browser](#)

you can position an image relative to other elements

hspace attribute specifies the amount of space to the left & right of the image

hspace=50

vspace attribute specifies the amount of space to top & bottom of the image

vspace=10

ALIGN attribute specifies the way the image is placed relative to surroundings

```
align="left"
align="right"
align="top"
align="center"
align="bottom"
```

Images & style sheets

```
<html>
<!-- Dave Reed  html34.html  8/22/01  -->
<head>
  <title>Title for Page</title>
</head>

<body>
  

</body>
</html>
```

[view page in browser](#)

most image formatting and layout can now be done using style sheets

options include:

- absolute positioning
- clip a rectangle
- various IE filters

again, be careful with Netscape 4.7

Style sheet properties include...

font properties

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

color & background properties

- color:red color:#000080
- background-color:white background-color:#000000

text properties

- 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

box properties

- width:10em
- margin-left:5em margin-left:10%
- margin-top:0
- border-width:0.2in border-width:thin border-width:medium border-width:thick
- border-style:solid border-style:double border-style:inset
- border-color:red

units of length

em: height of current font
ex: height of 'x' in current font
px: pixels, relative to resolution

in: inches
cm: centimeters
pt: points (1 pt = 1/72 in)

%: percentage, relative to current

Images + JavaScript (*sneak peek*)

```
<html>
<!-- Dave Reed  html35.html  8/22/01  -->

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

<body>
  <a href="javascript:alert('Do NOT click on me!');">
    </a>

  <p><br>

  
</body>
</html>
```

[view page in browser](#)

as we will see later, with the addition of JavaScript, images can be dynamic within the page

- can embed an image inside a link
→ clickable image
- newer attributes make this even easier (but not handled by Netscape 4.7)

ONMOUSEOVER specifies an action to take place when the mouse goes over the image

ONMOUSEOUT specifies an action to take place when the mouse leaves the image

ONCLICK specifies an action to take place when the mouse clicks on the image

On to programming...

you now know enough HTML to be able to design interfaces for Web-based programs

- semester countdown at www.creighton.edu/~davereed
utilizes a JavaScript program to determine current time, time until semester
user interface includes HTML text areas, tables for alignment, style sheet for font
- substitution ciphers at www.creighton.edu/~csc551/JavaScript/cipher.html
utilizes a JavaScript program to encode/decode messages
user interface includes HTML text boxes, tables for alignment, style sheet for font
- grade checker at www.creighton.edu/~csc551/CGI/grades.html
utilizes a JavaScript program to get user data, CGI program for accessing grades
user interface includes HTML text boxes, frames for input/output
- Othello game at www.creighton.edu/~csc551/JavaScript/Othello.html
utilizes a JavaScript program to play the game
user interface includes HTML text boxes, tables for alignment

Next week...

Internet & Web protocols

- TCP/IP, packet switching, routers
- HTTP, Web servers
- networking issues

read online materials

as always, be prepared for a quiz on

- today's lecture (moderately thorough)
- the reading (superficial)