

cis3.5 spring2009 lecture 1.3

topics:

- whirlwind tour of HTML
 - pages, links
 - lists, tables, images
 - image maps
- whirlwind tour of CSS (cascading style sheets)

resources:

- see class web page for a long list of on-line tutorials and references, including:
 - <http://www.w3.org/MarkUp/Guide/>
 - <http://www.w3.org/MarkUp/Guide/Advanced.html>
 - <http://www.w3.org/MarkUp/Guide/Style.html>

what is HTML?

- “HTML” stands for Hyper-Text Mark-up Language.
- HTML is the language with which web servers communicate with other computers via web browsers.
- Web page content is delivered in *pages*, consisting of *plain text* interspersed with *tags* that control how the content of the page is displayed.
- Web pages are stored in *files* on computers referred to as *servers*, because they “serve” (i.e., deliver) the content to the computers that want to look at the files.
- Web content pages, “documents”, are stored in files with names ending in: **.html**¹
For example: **sklar-home.html**
- *Style sheets* are files that can be used to control visual aspects of web pages. A single style sheet can be used for multiple web pages, so they are useful for maintaining a consistent “look-and-feel” across an entire web site.
- Style sheets are stored in files with names ending in: **.css**
For example: **sklar-style.css**

¹Note that web content can be delivered in other types of files with different naming conventions, but discussion of such types is beyond the scope of this class.

writing HTML

- There are a wide range of WYSIWYG² applications that can be used to create HTML documents.
- No matter which application you use, the basic underlying HTML is the same.
- I want you to understand the basic HTML because that will help you learn to use any HTML-writing application and will help you understand other people’s HTML code.³
- So for this class, you will use a **text editor** in order to write basic HTML.
 - on a PC, this is **Notepad** (not Wordpad)
 - on a Mac, this is **TextEdit** (in plain text mode, not rtf)
 - on Linux, this is **pico** or **emacs** or **vi**...
- For this class, you will create a file using one of these text editors and type content and HTML tags into the file. **Follow the lab sheet for detailed instructions!**
- In the remainder of these notes, you will find a quick reference to basic HTML tags and Style Sheet rules.

²What You See Is What You Get

³It is common practise to look at (“view”) the HTML code (“source”) of pages you find on the web in order to replicate aspects that you like. But don’t plagiarize content! And don’t copy your classmates’ code!

page tags

- opening *tag*: `<html>`
- closing *tag*: `</html>`
- header:
`<head>... </head>`
- body:
`<body>... </body>`
- title (goes inside the header):
`<title>My first HTML document</title>`
this text appears as the title of the *browser window*
- “heading” text:
`<h1>An important heading</h1>`
`<h2>A slightly less important heading</h2>`
there are other sizes, from `<h1>` (largest) to `<h7>` (smallest)
- note the difference between **header** and **heading**: header appears at the top of the page, between `<head>... </head>` tags; whereas headings appear in the content of a page

paragraph tags

- paragraphs:
`<p>This is the first paragraph.</p>`
`<p>This is the second paragraph.</p>`
- line breaks:
This is the first line.

This is the second line.

character tags

- emphasis:
This is a really interesting topic!
- italics:
This is a really <i>interesting</i> topic!
- bold:
This is a boldly interesting topic!
- preformatted text:
`<pre>`
hello world
hello
again
`</pre>`

font tags

- font size:
`<big>this text is very big</big>`
`<small>this text is very small</small>`
- another way to set font size:
`<font face="size=1">this text is in font size 1 (smallest)</font>`
`<font face="size=7">this text is in font size 7 (largest)</font>`
- font face:
`<font face="courier">this text is in courier font face</font>`
- font color:
`<font color="red">this text is red</font>`

links

- a *link* is a tag that directs your browser to another page if the user clicks on the link
- the content of the link tag consists of:
 1. the URL where you want the browser to go if the user clicks on the link, and
 2. the text that you want the user to see (i.e., the text that the user will click on to activate the link)
- example:
This a link to `<a href="http://www.google.com">google</a>`.

This a link to
`<a href="http://www.google.com" target=_blank>google</a>`
that opens up another window.

lists

- un-ordered list:

```
<ul>
<li>the first list item</li>
<li>the second list item</li>
<li>the third list item</li>
</ul>
```

- ordered list:

```
<ol>
<li>the first list item</li>
<li>the second list item</li>
<li>the third list item</li>
</ol>
```

- lists can be *nested*, which means you can put one list inside another list:

```
this is my shopping list:
<ul>
<li>bread</li>
<li>apples</li>
<li>ice cream</li>
  <ul>
    <li>vanilla</li>
    <li>chocolate</li>
    <li>mango</li>
  </ul>
<li>peanut butter</li>
</ul>
```

images

- the *image* tag allows you to display an image within a web page

- example:

```

```

- you can specify properties of the image:

- width, height
- border
- alignment

```

This text will flow around the right side of the graphic.
<p>

This text will flow around the left side of the graphic.
```

- you can include images in tables (see next topic)

tables

- begin a table with <table>
- end a table with </table>
- begin each row with <tr> and end each row with </tr>
- begin each column with <td> and end each column with </td>
- options:
 - borders
 - *cellpadding* (padding within a cell)
 - *cellspacing* (spacing between cells)
 - width and height (in pixels)
- tricks:
 - empty cells — use
 - *spanning* multiple rows or columns
- coloring cells:

```
<td bgcolor="red">ASDF</td>
```

- aligning cell content:
 - horizontally: left, center, right
 - vertically: top, middle, bottom
- advanced formatting:
 - cell padding (extra space inside the cells)
 - cell spacing (space between the cells)
 - width of table, of cells
 - alignment of cell content
 - empty cells ()
 - multi-row and multi-column cells
 - borders
 - coloring cells
 - headings

image maps

- example:


```

<map name="mymap">
  <area shape="rect" coords="0, 0,100,100" href="blue.html">
  <area shape="rect" coords="100,0,200,100" href="purple.html">
  <area shape="rect" coords="200,0,300,100" href="grey.html">
</map>
```
- origin is upper left corner (pixel)
- for **rect**, coords are: left-x, top-y, right-x, bottom-y
- for **circle**, coords are: center-x, center-y, radius
- for **poly**, coords are: x1,y1, x2,y2, ... xn,yn

style sheets

- text color and background color:


```
<style type="text/css">
  body { color: black; background: white; }
</style>
```
- linking to a separate style sheet (in another file):


```
<link type="text/css" rel="stylesheet" href="style.css">
```

and the "style file" looks like this:

```
/* style.css - a simple style sheet */
body {
  color: black; background: white;
}
```

- margins, left and right indents:


```
body { margin-left: 10%; margin-right: 10%; }
h1 { margin-left: -8%; }
h2,h3,h4,h5,h6 { margin-left: -4%; }
p { text-indent: 2em; margin-top: 0; margin-bottom: 0; }
```
- white space above and below:


```
h2 { margin-top: 8em; margin-bottom: 3em; }
```
- fonts:
 - styles:


```
em { font-style: italic; font-weight: bold; }
strong { text-transform: uppercase; font-weight: bold; }
```
 - text-transform can be: uppercase, lowercase
 - font families: Verdana, Garamond, "Times New Roman", sans-serif, e.g.


```
body { font-family: Verdana, sans-serif; }
```

- link colors:

```
:link { color: rgb(0, 0, 153) } /* for unvisited links */
:visited { color: rgb(153, 0, 153) } /* for visited links */
a:active { color: rgb(255, 0, 102) } /* when link is clicked */
a:hover { color: rgb(0, 96, 255) } /* when mouse is over link */
```

- divisions:

- name the divisions using “class” or “id”
“class” can be used for multiple elements;
“id” can only be used for one element

- class example:

in the style file:

```
div.box { border: solid; border-width: thin; width: 100% }
```

in the html file:

```
<div class="box">
```

The content within this DIV element will be enclosed in a box.

```
</div>
```

- id example:

in the style file:

```
#box2 { border: solid; border-width: thick; width: 100% }
```

in the html file:

```
<div id="box2">
```

The content within this DIV element will be enclosed in another box.

```
</div>
```