Best Practices: CSS/HTML

ACM Webmonkeys @ UIUC

Review of a basic HTML webpage

```
<html>
  <head>
    <title>My Website</title>
  </head>

  <body>
    Content, content, content. Veni, vidi, vici.
  </body>
</html>
```

Best Practice #1: the DOCTYPE

- Like any format or specification, there are multiple versions of HTML.
  - HTML4, XHTML, HTML5, etc. etc.
- For a browser to really know how to *exactly* render your webpage, you should specify a doctype.
  - Doesn't always matter -- but theoretically, web browsers should parse each version accordingly.
- The DOCTYPE goes at the very top of your HTML page, and tells the browser what version of HTML you're using.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

It's pretty long-winded. I would never recommend memorizing it, but you should know what it means and to include it.

With a doctype (to show where it goes):

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html>
  <head>
    <title>My Website</title>
  </head>

  <body>
    Content, content, content. Veni, vidi, vici.
  </body>
</html>
```

Best Practice #2: Stylesheets

- Those of you who were at last week's tutorial might remember how long our inline style tags were getting.
- Here's an example:

```
<div style="width:95%;margin:8px;margin-left:
0px;background:#EDF3FE;height:90%;font-family:
Verdana;font-size:14px;padding:8px;">
```

- Also, it gets repetitive when multiple items need the same styles.
- What we need is a way to move styles out of tags, and into a common place...

External Stylesheets

- An external stylesheet fulfills the last 'S' in CSS: it defines a list of styles grouped into **classes**.
- The word "external" is because this file is typically saved in a file like "styles.css" and *included* with a special tag from your HTML page.
  - We'll go over how to create and load a stylesheet later.
- To define a class, first decide what elements you want to apply the class to. Then, give it a name, and define your class like so:

```
div.bigred {
  color:red;
  font-size:42px;
}
```

Anatomy of a class

You can omit tagname if you want it to apply to any type of tag.

Likewise, you can omit classname (along with the period) if you want to change all instances of this tag.

```
tagname.classname {

    styles:go here;

}
```

Separate each style definition with a semicolon and, for readability, a linebre

Note the brackets which surround the class definition!

More on classes

- If an element has a particular class (and an element can have multiple classes), it gets all the styles that were defined in that class.
- So if we have a class named "bigred" which makes text big and red, and a class named "emphasized" which makes text italicized, then the following text would be big, red, and italicized:

```
<span class="bigred italicized">Text goes here.</span>
```

Note: a SPAN is a generic inline tag, like a freeform version of <b> or <i>. Use it when you want to change fonts.

Including a stylesheet

- Attaching a stylesheet to a page is something that shouldn't be considered part of the page's content -- so, we put inside the HEAD tag of the HTML webpage.
- The tag to associate a stylesheet with a webpage is the LINK tag (which is a generic tag, so we need to tell it explicitly to link a stylesheet):

```
<link rel="stylesheet" href="styles.css" />
```

Our stylesheet

- If we were to make a stylesheet for the website from last time, it might look like the one on the following page.
- Don't worry about the exact styles I have; just note how its laid out.
- The next-next page has the HTML, modified to use classes instead of inline styles.

```
body, html {
    font-family: Verdana; /* This style will drop down. */
    margin:0px;
    padding:0px;
    background:#DBE8FD; /* This one won't.
                        (by default, some styles do, some don't). */
}

h1 {
    margin:5px;
    padding:0px;
    font-weight:normal;
    color:white;
    font-size:43px;
}

h2 {
    font-size:31px;
    font-weight:normal;
    margin-top:0px;
}

td.topbanner {
    background:#006699;
    border-bottom:7px solid white;
}
```

[this is all in styles.css]

```
div.box {
    background:#EDF3FE;
    margin:8px;
    font-size:12px;
    padding:8px;
    height:90%;
}

div.leftbox {
    width:140px;
}

div.rightbox {
    width:95%;
    margin-left:0px;
    font-size:14px;
}
```

Yeah, stylesheets get long quick. There are tricks to do more with less, but I'll leave those for another day. However, a longer stylesheet means shorter HTML:

```
<html>
<head>
    <title>Example</title>
    <link rel="stylesheet" href="styles.css"/>
</head>

<body>

<table width="100%" height="100%">
  <tr>
    <td height="97px" valign="bottom" colspan="2"class="topbanner">
        <h1>My Blog</h1>
    </td>
  </tr>
  <tr>
    <td width="167px" valign="top">
        <div class="box leftbox">
            <center>
            <img src="myimage.jpg" alt="Picture of me"/>
            </center>
            <br/>
            Left col.
        </div>
    </td>
    <td valign="top">
        <div class="box rightbox">
            <h2>Dolorem Ipsum</h2>
            Right col.
        </div>
    </td>
  </tr>
</table>

</body>
</html>
```

[and this is in index.html]

More to come

- This was a short tutorial, as I have another coming up on website deployment, but I will probably do more "best practices" throughout the semester.