Cascading Style Sheets: Basics

What is CSS?

- A set of rules for displaying markup content

What is CSS?

- A set of rules for displaying markup content
- Cascading:
  - Display rules “cascade” down
  - The most specific rule is used

What is CSS?

- A set of rules for displaying markup content
- Cascading:
  - Display rules “cascade” down
  - The most specific rule is used
- Styles Sheet:
  - Rules are created as styles

The Need for CSS

- Fixing kludges in HTML

The Need for CSS

- Fixing kludges in HTML
- Separation of content and display

The Need for CSS

- Fixing kludges in HTML
- Separation of content and display
- More options for displaying content

The Need for CSS

- Fixing kludges in HTML
- Separation of content and display
- More options for displaying content
- Efficiency

CSS Levels & CSS History

- CSS level 1 - CSS1 is the first official release of the style recommendations set forth in 1996.
- CSS level 2 - CSS2 is the 1998 release of recommended specifications to CSS.
- CSS level 2 revision 1 - CSS2.1 is a revision that replaces the mechanics of CSS2 that lead to problems. Recommended as specification June 2011 by W3C.
- CSS level 3 - Drafts to CSS3 started appearing in 1999. CSS3 is made more modular and extensible for browser software vendors than previous levels of CSS.
- CSS level 4 - Drafts of CSS4 began September 2009. None of its modules are supported yet by browser software.

Visual Display in HTML

```
<h1><font color=‘red’ face=“Georgia, Times New Roman, Times, serif”>
This will be a heading 1 in red Georgia font</font></h1>
```

```
<h1><font color=‘red’ face=“Georgia, Times New Roman, Times, serif”>
Every time I want my text to look the same,
I have to retype or cut and paste all of this markup</font></h1>
```

Style Rules

- CSS style rules look like this:

```
Selector {
            property0:value0;
            property1:value1;
            propertyZ:valueZ
            }
```

CSS Reference

- http://www.w3schools.com/css/css_reference.asp

Inline Style Sheet

- Using a Style Attribute

```
<h1 style=“color:red; font-family: Georgia, Times New Roman, Times, serif”>
This also shows up as heading 1 in red Georgia font.  I don’t have to use
the font tag any more. </h1>
```

```
<h1 style=“color:red; font-family: Georgia, Times New Roman, Times, serif”>
However, I still have to retype or cut and paste the markup every time I
want to use the same style.</h1>
```

Notice that we switch from the attribute “face” for the “font” element to the property “font-family” in CSS. Names of attributes in HTML do not necessarily exactly match the names of properties in style rules.

Embedded Style Sheet

- Using a Style Element

```
<html>
<head><title>Embedded style sheet</title>
<style type=“text/css”>
h1 {color:red; font-family: Georgia, Times New Roman, Times, serif;}
</style>
</head><body>
<h1>This will also appear as a heading 1 in red Georgia font. In the head of this document, I
    declared an embedded style sheet that will make all h1 tags in this document red with
    Georgia font</h1>
<p>But how do I make some h1 tags use this style and others look different?
</p>
```

Class Selectors

```
<html>
<head><title>Embedded style sheet</title>
<style type=“text/css”>
.red {color:red; font-family: Georgia, Times New Roman, Times, serif;}
</style>
</head><body>
<h1 class=“red”>This will also appear as a heading 1 in red Georgia font. In the inline style sheet
    I declared a generic class style that applies red Georgia to ANY element declared as class
    red. I simply change a value in the style element in the head and all elements using that
    style will change to match</h1>
<p class=“red”>
This will also be in red Georgia font. Now I can be very specific when I apply styles, but I still
    have to add the same markup in the HEAD to each of my pages. How do I make all my
    pages use the same styles without copying markup to each page?
</p>
```

Linked Style Sheets

```
<html>
<head><title>Linked style sheet</title>
<link rel=“stylesheet” type=“text/css” href=“mystyle.css”>
</head><body>
<h1 class=“red”>This will also appear as a heading 1 in red Georgia font. In the head of this document I declared a link to an outside stylesheet named “mystyle.css”. In that external stylesheet I place all of the style declarations I was using in my inline stylesheet</h1>
<p>
Now I can change the way all of my pages display simply by modifying a value in my single external stylesheet. How efficient!
</p>
```

Example External Stylesheet

```
h1.redH1 {
        color:red;
        font-family: Georgia, Times New Roman, Times, serif
        }
body {
        background-color: #333333;
        }
.blueback {
        background-color: blue;
        color: white
        }
p {
        font-size: 12px;
        text-align: center;
        }
```

Linked Style Sheet

- Using an External Stylesheet

In-Class exercise:

Create an external stylesheet that defines a new style named phunky that makes text pink, centered, and bold. The sheet should also make the background color of the page ”antiquewhite”

```
.phunky {
            color:pink;
            text-align: center;
            font-weight: bold;
            }
body{
            background-color: antiquewhite
            }
```

Cascading revisited

- When using multiple styles that conflict, which will be displayed?

Cascading revisited

- When using multiple styles that conflict, which will be displayed?
- Order:

1. Inline style sheet
2. Embedded style sheet
3. External style sheet
4. Browser default

Style Rule Values

- Colors:
  - Names for some:
    - blue, red, green, pink
  - Hexadecimal
    - #0000FF, #FF0000, #00FF00, #FF3399
  - RGB
    - rgb(0,0,255), rgb(255,0,0), rgb(0,255,0)

Style Rule Values

- Font size:
  - px for pixels (a dot on the screen)
    - font-size: 12px
  - pt for point (1/72 of an inch)
    - font-size: 12pt
  - pc for pica (12 points)
    - font-size: 2pc