

# **CS120**

## **Introduction to Web Site Development**

### **Lecture 5 - CSS**

**Gabor Kondas**

**California State University, Los Angeles**

**(Based on slides by John Hurley)**

# Styles as Attributes

- Many elements have optional attributes for various aspects of appearance
- Examples:
  - `<body bgcolor = "red">`
  - `<table border = "1">`
- This involves two suboptimal practices
  - Setting presentation of individual elements in hard-to-find code in the document and can easily become inconsistent
  - Coding content and presentation together, where they are hard to separate when your html is used for some unanticipated purpose
- Better to use CSS for consistency and better control

# Cascading Style Sheets

## History

- Used to separate data and presentation
- 1996 – CSS 1.0
- 1998 – CSS 2.0
- 2011 – CSS 2.1
- CSS 3 currently being rolled out

## About

- CSS 2.1 contains 115 presentation properties
- Slight variation between browsers

# CSS Syntax

## Syntax:

- `STYLE = SELECTOR { RULES }`
- `RULES = PROPERTY;PROPERTY;...;PROPERTY;`
- `PROPERTY = NAME:VALUE`

## Example:

- ```
p {  
    color:red;  
    background-color:black;  
}
```

# CSS Syntax

- A **style** is a list of properties that can be applied to one or more HTML elements to change the way the element is presented or rendered
- A **selector** defines what HTML elements a style affects
- A **rule** is a list of properties separated by a semicolon
- A **property** is a value that can be altered to change the way an HTML element is presented

# The HTML STYLE Attribute

- All HTML elements have the **STYLE** attribute, for which we can apply one or more CSS style rules to these elements (inline style)
- To specify a single style rule:
  - **STYLE="CSS style rule;"**
- To specify multiple style rules, separate your rules with semicolons
  - **STYLE="CSS style rule 1;CSS style rule 2;"**

# Linking CSS to HTML

Multiple ways

1. Use an external style sheet by linking an element in the head of the HTML file
2. Embed the styles in the head section
3. Use inline styles per-element

# Inline Styles and HTML Style Attribute

## Examples:

```
<p style="color:red;">This is red text.</p>
```

```
<p style="color:red;background-color:black;">  
  This is red text over a black background.  
</p>
```



# Internal / Embedded Style Sheets

Styles are defined in the document's head section using the style element

Achieves consistent styling of a given type of element throughout the document

Example (put something in a <p> to test!):

```
<style>
  p {
    color: #00FF00;
  }
</style>
```

# External Style Sheets

- An **external stylesheet** contains styles that are defined in another document
- Separate content from presentation
  - Easily change all the styles for an entire site
- External sheets are stored in plain text files using the filename extension `.css`
- External sheets are typically placed in a directory called `style` or `styles`

# External Style Sheets

- Place link in the head section of the html document
  - `<link rel = "stylesheet" href = "style/sample.css" />`
- There are three types of external stylesheets
  - Permanent
    - `rel="stylesheet"`
  - Preferred
    - `rel="stylesheet"`
  - Alternate; overrides preferred if chosen
    - `rel="alternate stylesheet"`

# User-Defined Stylesheets

- Defined by browsers or other software

# Stylesheet Precedence

- Cascading Style Sheets “cascade” with these rules of precedence:
  - Descending order: inline, embedded, external, user-defined
- Most specific elements have most precedence
  - `<td style = "background-color: green">`  
    `<p style = "background-color: red">Hello, World</p></td>`
  - will display with red background color
- User agent defaults (eg, a browser's default font) usually have lower precedence than CSS but can be set to override CSS

# Color and Backgrounds

- **color**
- **background-color**
- **background-image**
- **background-repeat**
- **background-position**
- **background-attachment**

# Color Values

- Specified as 6-digit hexadecimal values (“#FF0000”)
- RRGGBB. Hex values A-F are capitalized
- This system can describe up to  $256^3$  colors; that’s more colors than we have concise terms for (“medium-light beige-avocado mauve”)
- Some color names (“red”) work but may not be consistent for different browsers
- rgb decimal values 0-255 also work, with this syntax:
- `(rgb(0,0,0))`

# Color

White = #FFFFFF; black = #000000

Many other RGB colors are less intuitive (yellow: #FFFF00) Use a color generator like the one at [www.colorschemedesigner.com](http://www.colorschemedesigner.com)



# Developer Tools

- All major browsers have available tools that can be used to experiment with changes to CSS without saving and rerendering
- Firebug is a free add-on for (at least) Firefox and Chrome that allows you to change html code and immediately see how it renders.  
Click Tools/Add-ons/Get Add-ons

# Background Images

Example, inside a stylesheet:

```
body {  
    background-image: url("../images/godzilla.jpg");  
}
```

- Need relative path **from the location of the stylesheet** to the location of the image file
- In this case, the html file was in demos, the external stylesheet was in demos/style, and the image was in demos/images

# Background Images

- Background-position (right top, etc)
- Background-repeat: no-repeat, repeat-x, repeat-y, or just repeat (default)
- Background-attachment
  - Determines whether background scrolls with other content; scroll or fixed

# Background Images

- Background images can make it hard to read text.
- You can set background color for elements to keep the image in the margins
- Background color of the element will be superimposed over the image and will be opaque

# Background Images

```
<style type="text/css">
  body {
    background-image: url("images/godzilla.jpg");
  }
  article {
    background-color: #AAAAAA;
    margin-left: 5%;
    margin-right: 5%;
  }
</style>
```

# Examples

- Page Background Color
- Background Color of different elements
- Image Background
- Background color and image

# Margins

May be used with many elements, but most often with div or html5 semantic elements

- eg `margin-left: 15%; margin-right: 15%; margin-top: 5%; margin-bottom: 5%;`
- These will be margins outside the element

# Margins and Padding

- margin, margin-left, margin-right, margin-top, margin-bottom
- padding, padding-left, padding-right, padding-top, padding-bottom
- Shorthand for these t r b l
  - margin: 0 0 15% 15%;
  - top, right, bottom, left



# Selectors

- A **selector** determines what HTML elements are affected by the list of style rules in a style
- There are many types of selectors

# Type Selectors

A **type selector** specifies a specific HTML element as the selector

- Example:

```
p { font-size:24px; }
```

Example

# Group Selectors

A **group selector** specifies a comma-separated list of HTML elements as the selector

- Example:

```
p, pre, code, a { font-size:24px; }
```

Example

# Universal Selectors

A **universal selector** specifies all HTML elements as the selector

- Example:

```
* { font-size:24px; }
```

# Descendant Selectors

- A **descendant selector** specifies a nested HTML element as the selector
- All parents and the child for whom the style is to be applied must be listed in order of appearance, in a space-separated list

- Examples:

```
p { font-size:12px; }
```

```
p strong { font-size:24px; }
```

```
p strong em { font-size:36px; }
```

# Child Selectors

- A **child selector** specifies a parent and direct child HTML element as the selector
- The style is applied to the child element
- Examples:  
`body > p { color:#FF0000; }`
- Does not apply if the child element is not a direct child (ie, inside a div, etc)

# ID Selectors

Select elements identified by specific IDs

```
#special { font-weight: bold; color: red; }
```

```
<p id="special">Special paragraph</p>
```

```
<p>Regular paragraph</p>
```

# Class Selectors

- A period followed by a class name defines a **class selector**
- The style is applied to any HTML element where you specify the attribute CLASS = classname
- Examples:

```
.contactInfo { color:#000077; }
```

```
<p class="contactInfo">
```

```
    Don't call us, we'll call you!
```

```
</p>
```

Example



# The Div Element

- Div is an html element container
  - Holds child elements
  - Operates at the block level
  - Can be used to apply styles to all div children
- Use when no HTML5 semantic element is correct

# The SPAN Element

- A Span is an HTML element container
  - Holds child elements
  - Content is phrasing content
  - Can be used to apply styles to all span children

# Div vs Span

- Div operates on flow (block) content.
- Span operates on phrasing (inline) content.
- Div must go outside p (div is flow content)
- Span must go inside p (span is phrasing content).

[Example](#)

# Fonts

- font-family
- font-size
- font-style
- font-variant
- font-weight
- font

# Font Size

- Pixels
  - Dicey since users may read page on mobile devices
  - Example: `font-size: 10px`
- Names
  - Example: `xx-large`
- Percent
  - Example: `font-size: 200%`
- Em
  - Relative units; 1= standard size
  - Example: `font-size: 10em`

# Fonts

- Font-variants
  - Example: `font-variant: small-caps`
- Font-weight
  - names, eg thick
  - Relative values, eg 300. Normal is 400.
- Font-style
  - Normal, bold, italic
  - This is completely different from using html elements to set font-style

# Examples

- Font
- Fontsize (in % and em)
- Font Style
- Font Variant

# Text

- **text-align**
  - e.g. right, left (default), center
- **text-decoration**
  - e.g. blink, line-through
- **text-indent**
  - e.g. 5% or 20 px
- **text-transform**
  - e.g. text-transform:uppercase



# Text

- **letter-spacing**
  - e.g. `letter-spacing:2px`
  - Relative to standard: can be negative
- **word-spacing**
  - analogous to letter spacing
- **white-space**
  - Controls wrapping rules
  - e.g. `white-space:nowrap;`

# Examples

- Align
- Decorate
- Indent
- Letter Spacing

# Borders

- border-style
- border-left-style, border-right-style, border-top-style, border-bottom-style
- border-color
- border-left-color, border-right-color, border-top-color, border-bottom-color
- border-width
- border-left-width, border-right-width, border-top-width, border-bottom-color

## Examples

# Looking Up Styles

- You will never be able to memorize all the properties of all the styles
- One good source for looking them up is <http://www.w3schools.com>