



**INTERNET & WEB**  
**APPLICATION DEVELOPMENT**  
**SWE 444**

Fall Semester 2008-2009 (081)

**Module 2 (IV): Cascading Style  
Sheets (CSS) [Part I]**

**Dr. El-Sayed El-Alfy**

Computer Science Department  
King Fahd University of Petroleum and Minerals  
alfy@kfupm.edu.sa

## Objectives/Outline

### • Objectives

- Learn the basics of CSS
- Learn how to create and use inline styles
- Learn how to create and use embedded style sheets

### • Outline

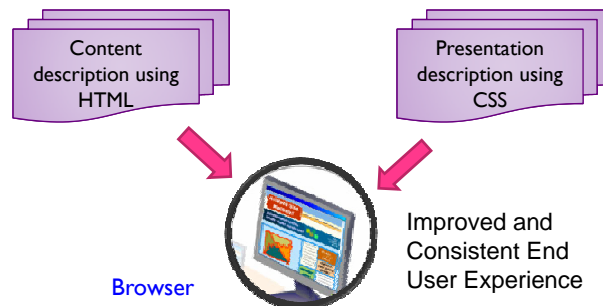
- Introduction to CSS?
- Benefits of Using CSS
- CSS Versions & Compatibility
- Types of CSS
- Conflicting Styles
- Inline Styles
- Embedded Styles

## Introduction to CSS

- Originally HTML was intended to describe the content of a document
- The formatting (style) of a document was control by the browser settings
  - So no need to have formatting tags in HTML
- This was a good engineering approach, but it didn't satisfy advertisers and "artists"
- As a result, HTML acquired more and more tags to control appearance
  - Content and appearance became more intertwined
  - Different browsers displayed things differently, which is a real problem when appearance is important

## Introduction to CSS (cont.)

- The solution was to specify the presentation of elements separately from the structure of the document
  - HTML describes what should be rendered (i.e. describes the hierarchy and relationships between parts of a document only), but not how it should be rendered
  - Cascading style sheets (CSS) describe the appearance of elements



## Benefits of Using CSS

- Gives finer and increased control on document formatting
- Consistency (uniformity) among different pages and elements
  - All pages in the site look the same
  - Rich design and layout
- Easier coding and maintenance
  - Due to separation of the document from the presentation, changes and reusing styles can be done easily
- Accessibility
  - PC browsers, mobiles, PDAs, printers, TVs, users with disabilities, etc...
  - No browser specific requirements, such as plug-ins
- Can be used for both HTML and XML pages

## CSS Versions & Compatibility

- CSS I
  - Released in 1996
  - IE 4.x was fully CSS I compliant
  - Spotty Netscape 4.x support
    - Netscape pushed their own style sheet language
    - Result: if you have users using Netscape 4.x then use CSSes with care!
  - Always test with both browsers!
  - Limitations of CSS I
    - Has almost no support for tables
    - Makes no provision for downloadable fonts
    - Lack of media types

## CSS Versions & Compatibility (cont.)

- CSS 2
  - Released in 1998
  - Extends CSS1
  - IE 5.x+ supports most, but not all CSS2 features
  - Netscape 6.x claims “unsurpassed support” for CSS1 and CSS2
  - Mozilla 1.x is generally considered to have the best CSS support
- CSS 2.1 and CSS 3 are currently under development

## CSS Versions & Compatibility (cont.)

- The only disadvantage of CSS is non-compatibility with all Internet browsers
- CSS1 was partially supported by browsers Internet Explorer 3, Internet Explorer 4, and Netscape Navigator 4.7
- CSS2 is fully supported by all new versions of popular Web browsers like: Internet Explorer 6, Netscape 6, Opera 5, and Micro Browsers for Mobiles
- If browser does not support CSS it will display page in HTML formatted form, ignoring the styles
  - i.e., the styles are themselves displayed

## Conflicting Styles

- What style will be used when there is more than one style specified for an HTML element?
  - Inline style
  - If no inline, embedded style is applied
  - If no embedded style, external style sheet applied
  - Any undefined attributes use web browser default

## Types of CSS

- The style can be controlled
  - For individual elements using “inline styles”
  - Globally by defining style sheets which can be either embedded style sheets or external style sheets
- Styles have their own peculiar syntax
- External style sheets are defined in separate files (the filename extension is likely to be .css)
- Multiple style definitions will cascade into one
- CSS Validation: [jigsaw.w3.org/css-validator/](http://jigsaw.w3.org/css-validator/)

## Inline Styles

- Define a style for an individual element by using the `style` attribute in the element's start tag
- Each CSS *property* is followed by a *colon* and the *value* of the attribute
- Multiple property declarations are separated by a *semicolon*
- Example

```
<p style = "font-size: 20pt">This text has the  
<em>font-size</em> style applied to it, making it 20pt.  
</p>  
  
<p style = "font-size: 20pt; color: #6666ff">  
This text has the <em>font-size</em> and  
<em>color</em> styles applied to it, making it  
20pt. and light blue.</p>
```

## Inline Styles (cont.)

- Advantage
  - Useful if you only want a small amount of markup
- Limitations of inline styles
  - do not truly separate presentation from content and structure
    - mixes display information into HTML
    - clutters up HTML code
  - to apply similar styles to multiple elements, you need to redefine them for each element
- Solution
  - Use embedded styles sheets or external style sheets
    - External style sheets can save a lot of work

## Style Sheets Syntax

- A style sheet consists of a set of rules; each of which has the following syntax

➤ Or `selector {property: value}` descriptor

```
selector, ..., selector {  
  property: value;  
  ...  
  property: value  
}
```

Allows one style to be applied simultaneously to many tags

- where
  - selectors are html or xml tags to be affected (the selector is case-sensitive if and only if the document language is case-sensitive); CSS also defines some ways to combine tags
  - property and value describe the appearance of that tag; CSS has quite a long list of them
  - spaces after colons and semicolons are optional
  - a semicolon must be used between property:value pairs, but a semicolon after the last pair is optional
  - if the value is multiple words, put quotes around the value

## Examples

- `h1 {color: green; font-family: Verdana} /* says that everything included in h1 tags should be in the Verdana font and colored green*/`
- `h1, h2, h3 {font-family: Arial, sans-serif;} /* use 1st available font */`
- `p, table, li, address { /* apply to all these tags */  
 font-family: "Courier New"; /* quote values containing spaces */  
 margin-left: 15pt; /* specify indentation */  
}`
- `p, li, th, td {font-size: 80%;} /* 80% of size in containing element */`
- `th {background-color: #FAEBD7} /* colors can be specified in hex */`
- `body { background-color: #ffffff;}`
- `h1, h2, h3, hr {color: brown;} /* adds to what we said before */`

## Embedded Styles

- A style sheet can be embedded anywhere into an HTML document using the `style` element
- However it is common to have it
  - within the `head` element and
  - within a comment to hide it from older browsers that don't understand CSS
- The general form is:

```
<style TYPE="text/css">
  <!--
    CSS Style Sheet
  -->
</style>
```

## Example

```
<head>
  <title>Cascading Style Sheets</title>
  <style>
    h2, h3 {color: green; font-family: sans-serif}
    h4, h5, h6 {color: blue; font-family: sans-serif}
  </style>
</head>
```



## Q & A



## References

- Some useful links with examples and other resources:
  - *Internet and World Wide Web How to Program*, 4/e, H. M. Deitel, P. J. Deitel, and A. B. Goldberg, Pearson Education Inc., 2008. Chapter 5.
  - *Web Development and Design Foundations with XHTML*, 4/e, Pearson Education Inc. 2009. Chapter 3.
  - W3C
    - Cascading Style Sheets: <http://www.w3.org/Style/CSS/>
    - CSS2 Specification: <http://www.w3.org/TR/CSS2/>
  - W3 Schools CSS Tutorial:  
<http://www.w3schools.com/css/default.asp>
  - CSS Editors
    - Best CSS stand alone editor is Topstyle Pro – <http://www.bradsoft.com>
  - CSS Validation:
    - <http://jigsaw.w3.org/css-validator/>
  - Index DOT CSS (The Advanced CSS Reference)
    - <http://www.blooberry.com/indexdot/css/index.html>