

# CSC435: Web Programming

## Lecture 5: Intro to CSS

Bei Xiao

American University

Jan 30, 2014

# Overview

- Introduction to CSS
- Selector syntax
- Multiple selectors
- Nested selectors
- Inserting CSS and precedence.

# What is CSS?

- CSS is an alternative to the “style” attribute  
`<div style="height:500px; width:200px"></div>`
- Advantage:
  - Can control the style of many different elements with a single command.
  - Separation of function: HTML can focus on content while CSS handles styling.

# What does CSS do?

- **Examples:**
- Setting text and background colors : Demo
- Specifying fonts, text size, and line space.
- Positioning elements

# Basic Syntax: writing the rules

- Selector {  
    declaration  
}

Example: for styling a paragraph:

```
P{  
Color: blue;  
}
```

# Selector and Properties

Selector

Declaration

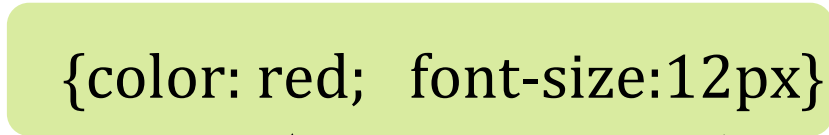
Declaration

H1

{color: red; font-size: 12px}

property value

property value



# Link to CSS

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

# CSS demo

## Heading 1

Paragraph 1

## Heading 2

Paragraph 2

## Heading 3

Paragraph 3

## Heading 4

Paragraph 4

## Heading 1

Paragraph 1

## Heading 2

Paragraph 2

## Heading 3

Paragraph 3

## Heading 4

Paragraph 4



# Multiple selectors

```
p, #heading1
{
    color:blue;
    font-family: Stencil;
}
```

# How to selectively style the element?

- **ID:** Can be used with a single, unique element to define a specific style

```
<h1 id="heading1">Heading 3</h1>
```

- **Class:** Can be used with several elements to use the same style with each of them. A single element may also have several class.

```
<p class="class-selector1 class-selector2 class-selector3"></p>
```

# Demo: IDs and Class

- **ID selector**.id

```
p.blue_paragraphs {  
    color: blue;  
    font-size: larger  
}
```

- **Class: #class name**

```
#heading1, #heading2 {  
    color: red;  
    font-family: Verdana;  
    font-size: 200%  
}
```

# Exercise: multiple selectors

## Heading 1

font-family: Verdana

PARAGRAPH 1

## HEADING 2

font-family: Stencil

Paragraph 2

## Heading 3

PARAGRAPH 3

## Heading 4

Paragraph 4

# Demo: Nested selector

- .html

```
<div id="special">
  <h1 id="heading1">Heading 1</h1>
  <p class="blue_paragraphs" > Paragraph 1</p>
</div>
<p class="blue_paragraphs"> Paragraphs 2</p>
```

- .CSS

```
#special p.blue_paragraphs
{
  color:blue;
  font-family: Stencil;
  text-indent: 2em;
}
```

# Nested selector can be arbitrarily long

- .html

```
<div id="special">  
  <h1 id="heading1">Heading 1</h1>  
  <p class="blue_paragraphs" > Paragraph 1</p>  
</div>  
<p class="blue_paragraphs"> Paragraphs 2</p>
```

- .CSS

```
#special p.blue_paragraphs #heading1, p.a  
div#b p.c  
{  
  color:blue;  
  font-family: Stencil;  
}
```

# Exercise: nested selector

## Heading 1

Font-family: Verdana

Color: blue

Paragraph 1

## Heading 2

Paragraph 2

Color: red

## Heading 3

Paragraph 3

## Heading 4

Font-family: Verdana

Color: red

Paragraph 4

# Summary of Selector

- Selector or selectors which group or groups of the following rules to apply to.
- Selectors are flexible by setting IDs and Class to HTML elements.



# Importing Styles: three methods

- **Inline:** add a “style” attribute containing the CSS rule directly to an HTML

- **Internal:** Define CSS Rules in the style tag in the HTML

```
<head>
  <style type="text/css">
    p{
      color:red;
    }
  </style>
</head>
```

- **External:** Add a link to an external stylesheet in the head of your HTML page

```
<head>
  <link rel="stylesheet" type="text/css" href="exercise1.css">
</head>
```

# Precedence (Hierarchy)

- Browser default (lowest)
- Linked External Style Sheet
- Internal Style Sheet
- Inline Style Sheet (style attribute)
- Any style rule marked !important by the author (highest)
- Exception: if a link a external style sheet is placed after an internal style sheet, the **external** will take precedence!

# Summary of Precedence

- The more specific rule wins (according to a well-defined set of specificity rules).
- If the two rule has the same specificity, the one that comes last win.

# Using !Important

- It is a hack, so use it sparingly.
- Example:

```
p{  
  Background-color:green  
  color:blue!important;  
}
```

# Quiz 1

- Which color will paragraphs be when this embedded style sheet is applied to a document?

```
<style type="text/css">
```

```
  p{color: green;}
```

```
  p{color:red}
```

```
  p{color:yellow}
```

```
</style>
```

# Quiz 2

- Which color will paragraph “Blue Paragraphs” appear?

```
<style type="text/css">
```

```
  p{color: blue;}
```

```
</style>
```

```
<body>
```

```
<p style="color:red"> Blue Paragraphs</p>
```

```
</body>
```

# Quiz 3

- Which color will paragraph “Blue Paragraphs” appear?

```
<style type="text/css">
```

```
  p.blue_paragraphs{color:green}
```

```
  p{color: blue;}
```

```
</style>
```

```
<body>
```

```
<p class="blue_paragraphs"> Blue Paragraphs</p>
```

```
<p> Another paragraph</p>
```

```
</body>
```

# Homework & Readings

Chapter 11, 12 and 13, Robbins.

Online:

<http://www.w3.org/TR/CSS1/#basic-concepts>

Code Academy, Web fundamentals:

[http://www.codecademy.com/courses/web-beginner-en-TlhFi/0/1?curriculum\\_id=50579fb998b470000202dc8b](http://www.codecademy.com/courses/web-beginner-en-TlhFi/0/1?curriculum_id=50579fb998b470000202dc8b)