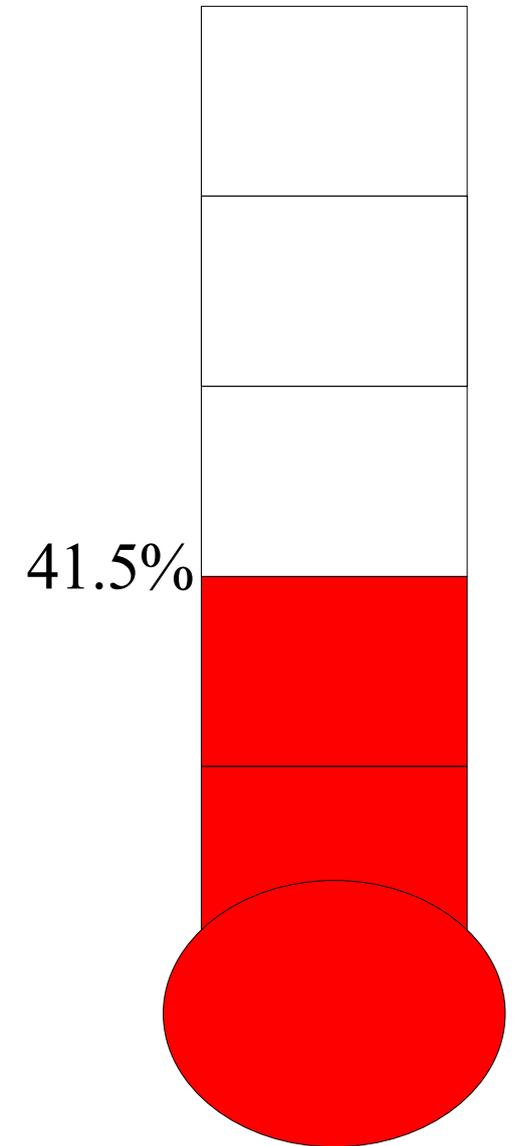


## Comp-206 : Introduction to Software Systems Lecture 23

Alexandre Denault  
Computer Science  
McGill University  
Fall 2006

# Course Evaluation - Mercury

22 / 53



# Assignment 3

- Artistic Bonus
- There will be no extension

# Protecting a directory

- For assignment 3, you will need to password protect your ~/public\_html/cs206 directory.
- The first step is to put an .htaccess file in the directory.

```
AuthUserFile /home/user/adenau/public_html/cs206/.htpasswd
AuthGroupFile /dev/null
AuthName EnterPassword
AuthType Basic
```

```
require user alex
```

- You then need to create a .htpasswd file using the htpasswd command (found on [troy.cs.mcgill.ca](http://troy.cs.mcgill.ca)).

```
htpasswd -c .htpasswd alex
```

- You can also add users.

```
htpasswd .htpasswd bob
```

- Forms allow users to input data into web page
- They are delimited using the `<form>` tag.
- When a form is submitted, data contained inside the form is sent to the action page.
- There are two ways to send this data : post and get
- Sub-components of forms (the place where you write data) are mostly `<input>` tags.

# Post vs Get

- When a form transfers input data to the web server, it can do so in two ways.
- The first, GET, transfers the data inside the query string.
  - ◆ This is usually easy to recognize :  
`http://www.google.ca/search?q=test&ie=utf-8&oe=utf-8&rls=org.mozilla:en-US:official&client=firefox-a`
  - ◆ Allow the easy use of “Back” buttons.
  - ◆ Easier to debug.
  - ◆ However, less secure since text is transferred in the query (and logged by the server).

# Post vs Get

- When a form transfers input data to the web server, it can do so in two ways.
- The second, POST, transfers the data a part of the query packet.
  - ◆ Doesn't change the query, thus more secure
    - data not automatically logged
  - ◆ Doesn't always work well with back buttons
    - Warning messages about data needed to be posted again
  - ◆ More difficult to debug
    - Need special software to read "POST"

- As mentioned, most input components use the `<input>` tag.
  - ◆ Text fields:
    - `<input type="text" name="firstname">`
  - ◆ Radio buttons
    - `<input type="radio" name="sex" value="male"> Male <br />`
    - `<input type="radio" name="sex" value="female"> Female`
  - ◆ Checkboxes:
    - `<input type="checkbox" name="bike">`
- Components that must be grouped together should all use the same name.

# Submitting Input

- To submit a form, you will need a submit button.
- Buttons are also defined using `<input>` tag.
  - ◆ `<input type="submit" value="Submit">`
- Once pressed, a form will send the input to the URL specified in its action attribute.

# Example of a Form

```
<form name="input" action="script.php" method="get">
  Username: <input type="text" name="user"> <br />
  Type of student: <br />
  <input type="radio" name="student" value="ugrad">
    Undergraduate<br />
  <input type="radio" name="student" value="grad">
    Graduate<br />
  Graduating : <input type="checkbox"
    name="graduating"> <br />
  <input type="submit" value="Submit">
</form>
```

# Colors

- Colors are defined using a hexadecimal notation for the combination of Red, Green, and Blue color values (RGB).
- The lowest value that can be given to one light source is 0 (hex #00). The highest value is 255 (hex #FF).

Color	Name	Hex	RGB
	White	#FFFFFF	255, 255, 255
	Black	#000000	0, 0, 0
	Red	#FF0000	255, 0, 0
	Lime (Green)	#00FF00	0, 255, 0
	Blue	#0000FF	0, 0, 255
	Yellow	#FFFF00	255, 255, 0
	Fuschia	#FF00FF	255, 0, 255
	Aqua	#00FFFF	0, 255, 255
	Gray	#808080	128,128,128

# Not all browser are equal

- Although HTML is a “strict” standard, not all browsers will interpret HTML the same way.
- It's important to test your HTML in different browsers.



# November 2006

■ Microsoft IE	85.24%
■ Mozilla Firefox	12.15%
■ Apple Safari	1.61%
■ Opera	0.69%
■ Netscape	0.11%



- CSS stands for Cascading Style Sheets.
- Styles define how to display HTML elements.
- Styles were added to HTML 4.0 to solve a problem :
  - ♦ A centralized and uniform way to define the style of a website.
  - ♦ Styles are saved in an external file (.css).
  - ♦ Style can be shared by several pages.
- The introduction to CSS is based on the W3C tutorial [http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp)

# Color by number



1 - blue  
2 - black  
3 - grey  
4 - orange

■ Content : Lines

■ HTML : Numbers

■ CSS : Instructions

# CSS syntax

- A CSS rule is composed of three elements: a selector, properties and values.

```
selector {property: value}
```

- ♦ The selector is normally the HTML element/tag you wish to define.
- ♦ The property is the attribute you wish to change
- ♦ Each property can take a value.
- ♦ For example, the following rule will draw the body of your document black.

```
body {color: black}
```

- ♦ If a value has multiple words, the value should be quoted.

```
p {font-family: "sans serif"}
```

- ♦ Multiple properties can be defined in a particular rule.

```
p {text-align:center;color:red}
```

- You can group selectors :

```
h1,h2,h3,h4,h5,h6
```

```
{
```

```
  color: green
```

```
}
```

# Specific Classes

- With the class selector you can define different styles for the same type of HTML element.

```
p.right {text-align: right}
```

```
p.center {text-align: center}
```

- You have to use the class attribute in your HTML document:

```
<p class="right">
```

This paragraph will be right-aligned.

```
</p>
```

```
<p class="center">
```

This paragraph will be center-aligned.

```
</p>
```

# Specific Classes

- You can also omit the tag name in the selector to define a style that will be used by all HTML elements that have a certain class.

```
.center {text-align: center}
```

```
.title {color: black}
```

- Again, you need to use the class attribute in your HTML document:

```
<p class="center title">
```

```
This is a paragraph.
```

```
</p>
```

- Note that you can assign more than one class per given element.

# External Style Sheet

- As previously mentioned, an external style sheet is ideal when the style is applied to many pages.
- Each page must link to the style sheet using the <link> tag (located in the header).

```
<head>
```

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

```
</head>
```

- The browser will read the style definitions from the file mystyle.css, and format the document according to it.

```
hr {color: sienna}
```

```
p {margin-left: 20px}
```

# Internal Style Sheet

- An internal style sheet should be used when applying a style element to a single document.
- You define internal styles in the head section by using the `<style>` tag.

```
<head>
```

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

```
  hr {color: sienna}
```

```
  p {margin-left: 20px}
```

```
</style>
```

```
</head>
```

- Note the absence of space between 20 and px. Some browser will not display the style correctly if the value is “20 px”, as opposed to “20px”.

# Inline Styles

- An inline style loses many of the advantages of style sheets by mixing content with presentation.
- Use this method sparingly, such as when a style is to be applied to a single occurrence of an element.

```
<p style="color: sienna; margin-left: 20px">
```

This is a paragraph

```
</p>
```

# Order of priority

- Style are resolved in the following order:
  1. Browser default
  2. External style sheet
  3. Internal style sheet (inside the <head> tag)
  4. Inline style (inside an HTML element)
- When faced with overlapping styles, the last style will take effect (highest number).
  - ♦ For example, if an external style sheet defines texts as green, but an internal style sheet defines it as red, the text will be red.

# CSS Text Properties

- Allows you to control the appearance of text.
  - ♦ `color` : Sets the color of a text : *color*
  - ♦ `text-align`: aligns the text in an element : left, right, center, justify
  - ♦ `text-decoration` : Adds decoration to text : none, underline, overline, line-through, blink
  - ♦ `text-transform` : Controls the letters in an element : none, capitalize, uppercase, lowercase

# CSS Font Properties

- Allows you to change the font family, boldness, size, and the style of a text.
  - ♦ font-family :A prioritized list of font family names and/or generic family names for an element : *family-name, generic-family*
  - ♦ font-size : Sets the size of a font : xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, *length*
  - ♦ font-style : Sets the style of the font : normal, italic, oblique

# CSS Border Properties

- Allow you to specify the style and color of an element's border.
- You can create lots of effect with borders.
  - ◆ `border-bottom-color` : Sets the color of the bottom border : *border-color*
  - ◆ `border-bottom-style` : Sets the style of the bottom border : *border-style*
  - ◆ `border-bottom-width` : Sets the width of the bottom border  
thin : medium thick *length*
- You have the same options for `border-left-`, `border-right-` and `border-top-` .
- You also have shorthand version `border-left`, `border-right`, `border-top`, `border-bottom` and `border`.

# CSS Margin Properties

- Define the space around elements.
- It is possible to use negative values to overlap content.
  - ♦ `margin-bottom` : Sets the bottom margin of an element : `auto`, *length*, %
  - ♦ `margin-left` : Sets the left margin of an element : `auto`, *length*, %
  - ♦ `margin-right` : Sets the right margin of an element : `auto`, *length*, %
  - ♦ `margin-top` : Sets the top margin of an element : `auto`, *length*, %
- A shorthand margin property can also be used to change all of the margins at once.
- The CSS padding property is similar, it defines the space between the element border and the element content.

# And many others ...

- CSS Background
- CSS List Properties
- CSS Dimension Properties
- CSS Classification Properties
- etc

# Setting the color of text

```
<html>
  <head>
    <style type="text/css">
      h1 {color: #00ff00}
      h2 {color: #dda0dd}
      p {color: rgb(0,0,255)}
    </style>
  </head>

  <body>
    <h1>This is header 1</h1>
    <h2>This is header 2</h2>
    <p>This is a paragraph</p>
  </body>
</html>
```

# Setting the font family

```
<html>
  <head>
    <style type="text/css">
      h3 {font-family: times}
      p {font-family: courier}
      p.sansserif {font-family: sans-serif}
    </style>
  </head>

  <body>
    <h3>This is header 3</h3>
    <p>This is a paragraph</p>
    <p class="sansserif">This is a paragraph</p>
  </body>
</html>
```