Advanced Internet Technology Lab

Lab # 2

Cascading Style Sheet

Eng. Haneen El-masry

February, 2014
Objective

➢ To be familiar with CSS.

Introduction

HTML was never intended to contain tags for formatting a document. HTML was intended to define the content of a document.

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

What is CSS?

➢ CSS stands for Cascading Style Sheets.
➢ Styles define how to display HTML elements.
➢ Styles were added to HTML 4.0 to solve a problem.

There are three ways of inserting a style sheet:

➢ External style sheet.
➢ Internal style sheet.
➢ Inline style.

External Style Sheet

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section:

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```
An external style sheet can be written in any text editor. The file should not contain any html tags. Your style sheet should be saved with a .css extension.

**Internal Style Sheet**

An internal style sheet should be used when a single document has a unique style. You define internal styles in the head section of an HTML page, by using the `<style>` tag, like this:

```html
<head>
  <style>
    .....  
  </style>
</head>
```

**Inline Styles**

An inline style loses many of the advantages of style sheets by mixing content with presentation. Use this method sparingly!

To use inline styles, you use the **style attribute** in the relevant tag.

```html
<p style="....."> This is a paragraph. </p>
```

**CSS Syntax**

A CSS rule has two main parts: a selector, and one or more declarations:

- **The selector** is normally the HTML element you want to style.
- Each **declaration** consists of a property and a value.
- The property is the style attribute you want to change. Each property has a value.

A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly brackets “{}”.

```plaintext
Selector  { Property : value ; property : value }
```

The selector can be the tag name of HTML element, id selector or class selector.
The id Selector

The id selector is used to specify a style for a single, unique element.

The id selector uses the id attribute of the HTML element, and is defined with a "#".

```css
#para1 {
  text-align:center;
  color:red;
}
```

The class Selector

The class selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements.

This allows you to set a particular style for many HTML elements with the same class.

The class selector uses the HTML class attribute, and is defined with a ".".

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

CSS Comments

A CSS comment begins with "/*", and ends with "*/", comments are ignored by browsers.

CSS Properties & values

- **CSS Background**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Values</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>background-color</td>
<td>Sets the background color of an element.</td>
<td>The value can be: • a HEX value - &quot;#ff0000&quot;. • an RGB value - &quot;rgb(255,0,0)&quot;. • a color name - &quot;red&quot;.</td>
<td>h1{background-color:#6495ed;}</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
<td>Values</td>
<td>Example</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>background-image</td>
<td>Sets the background image for an element.</td>
<td>URL</td>
<td>body { background-image:url('2013.jpg'); }</td>
</tr>
<tr>
<td>background-repeat</td>
<td>Sets how a background image will be repeated.</td>
<td>No-repeat&lt;br&gt;Repeat-x&lt;br&gt;Repeat-y</td>
<td>body { background-image:url('2013.jpg');&lt;br&gt;background-repeat:repeat-x; }</td>
</tr>
<tr>
<td>background-attachment</td>
<td>Sets whether a background image is fixed or scrolls with the rest of the page.</td>
<td>Fixed scroll</td>
<td>Body{ background-image:url('2013.jpg');&lt;br&gt;background-repeat:repeat-x;&lt;br&gt;background-attachment:fixed; }</td>
</tr>
<tr>
<td>background-position</td>
<td>Sets the starting position of a background image.</td>
<td>Top, Bottom, Right, Left, right top, center, ....</td>
<td>body { background-image:url('2013.jpg');&lt;br&gt;background-repeat:repeat-x;&lt;br&gt;background-attachment:fixed;&lt;br&gt;background-position:top; }</td>
</tr>
<tr>
<td>background</td>
<td>Sets all the background properties in one declaration.</td>
<td></td>
<td>Body { Background: #fff00&lt;br&gt;url('2013.jpg') no-repeat top;}</td>
</tr>
</tbody>
</table>

**Note:** The order is (background-color background-image background-repeat background-attachment background-position)

---

### CSS Text

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Values</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>color</td>
<td>Sets the color of text</td>
<td>The value can be:</td>
<td>h1 {color:#6495ed;}</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- a HEX value - &quot;#ff0000&quot;.&lt;br&gt;- an RGB value - &quot;rgb(255,0,0)&quot;.&lt;br&gt;- a color name - &quot;red&quot;.</td>
<td></td>
</tr>
<tr>
<td>letter-spacing</td>
<td>Increases or decreases the space between characters in a text.</td>
<td>Number</td>
<td>body { letter-spacing:25; }</td>
</tr>
</tbody>
</table>
### CSS Text Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Values</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>text-align</td>
<td>Specifies the horizontal alignment of text.</td>
<td>Centered, Left, Right, justified.</td>
<td>body {text-align:Centered;}</td>
</tr>
<tr>
<td>text-decoration</td>
<td>Specifies the decoration added to text.</td>
<td>none, overline, line-through, underline, blink</td>
<td>Body{text-decoration:overline;}</td>
</tr>
<tr>
<td>text-indent</td>
<td>Specifies the indentation of the first line in a text-block.</td>
<td>Number</td>
<td>body {text-indent:5;}</td>
</tr>
<tr>
<td>text-transform</td>
<td>Controls the capitalization of text.</td>
<td>uppercase, lowercase, capitalize</td>
<td>p {text-transform:uppercase;}</td>
</tr>
<tr>
<td>word-spacing</td>
<td>Increases or decreases the space between words in a text</td>
<td>Number</td>
<td>p {word-spacing:2;}</td>
</tr>
</tbody>
</table>

#### CSS Font

**Property** | **Description** | **Values** | **Example** |
--- | --- | --- | --- |
Font-family | Specifies the font family for text | Times New Roman, Georgia, Serif, Arial, Verdana, Courier New, Lucida Console | p{font-family:"Times New Roman", Times, serif;} |
**Notes:**
- If the name of a font family is more than one word, it must be in quotation marks.
- More than one font family is specified in a comma-separated list.

Font-size | Specifies the font size of text | Number | h1 {font-size:40px;} |

Font-style | Specifies the font style for text | Normal, Italic, oblique | p.normal {font-style:normal;} |

Font-weight | Specifies the weight of a font. | normal, bold, bolder, lighter, 100-900 | p.normal {font-weight:normal;} |
Font
Sets all the font properties in one declaration

body {
  font: normal bold 40 "times new roman";
}

*Note:* The order is (font-style font-weight font-size font-family).

**CSS Links**

Links can be styled with any CSS property (e.g. color, font-family, background, etc.).

Special for links are that they can be styled differently depending on what state they are in.

The four links states are:

- **a:link**: a normal, unvisited link.
- **a:visited**: a link the user has visited.
- **a:hover**: a link when the user mouse over it.
- **a:active**: a link the moment it is clicked.

**Examples**

```css
a:link {color:#FF0000;}  /* unvisited link */
a:visited {color:#00FF00;} /* visited link */
a:hover {color:#FF00FF;}  /* mouse over link */
a:active {color:#0000FF;} /* selected link */
```

**CSS Lists**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Values</th>
<th>Example</th>
</tr>
</thead>
</table>
| list-style-type   | Specifies the type of list item marker   | Circle, Square, upper-roman, lower-roman, lower-alpha, upper-alpha     | Ol {list-style-type:upper-roman;}  
|                   |                                          |                                                                        | Ul(list-style-type:circle;)   |
| list-style-image  | specifies an image as the list item marker | URL                                                                    | ul {                          |
|                   |                                          |                                                                        | list-style-image: url('sqpurple.gif');} |
CSS Box Model

**Margin:** Clears an area around the border. The margin does not have a background color, it is completely transparent.

**Properties:**
Margin, Margin-top, Margin-right, Margin-bottom, and Margin-left.

**Border:** A border that goes around the padding and content. The border is affected by the background color of the box.

**Properties:**
Border, border-width, border-style, and border-color.

**Padding:** Clears an area around the content. The padding is affected by the background color of the box.

**Properties:**
Padding, padding-top, padding-right, padding-bottom, and padding-left.

**Content:** The content of the box, where text and images appear.
Positioning

There are four different positioning methods.

- **Static Positioning**

  HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the page. Static positioned elements are not affected by the top, bottom, left, and right properties.

- **Fixed Positioning**

  An element with fixed position is positioned relative to the browser window. It will not move even if the window is scrolled.

- **Relative Positioning**

  A relative positioned element is positioned relative to its normal position.

- **Absolute Positioning**

  An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is `<html>`.
Exercise:

Design a web page using HTML and CSS basics that studied.