IS 242 Web Application Development 1

Lecture 5: Introduction to CSS (Part 1)

Outlines of today's lecture

- What is CSS?
- CSS syntax
- How to use CSS
- Different CSS properties

But first … what we have addressed so far ..

- HTML form elements

- HTML5 new layout elements

Introduction to CSS

- HTML5 was designed to specify the **content and structure** of a document and was NEVER intended to contain tags for formatting a document.

```
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
```

- Though **HTML5 has some attributes that control presentation**, it's better not to mix presentation with content.
  - 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.

Introduction to CSS (Cont.)

- CSS stands for **Cascading Style Sheets**

- We use CSS to specify the **presentation** of elements on a web page (e.g., fonts, spacing, sizes, colors, positioning) **separately** from the document's **structure and content** (section headers, body text, links, etc.).

http://www.w3schools.com/css/demo_default.htm

Welcome to My Homepage

Use the menu to select different Stylesheets

- Stylesheet 1
- Stylesheet 2
- Stylesheet 3
- Stylesheet 4
- No Stylesheet

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: Stylesheet1, Stylesheet2, Stylesheet3, Stylesheet4.

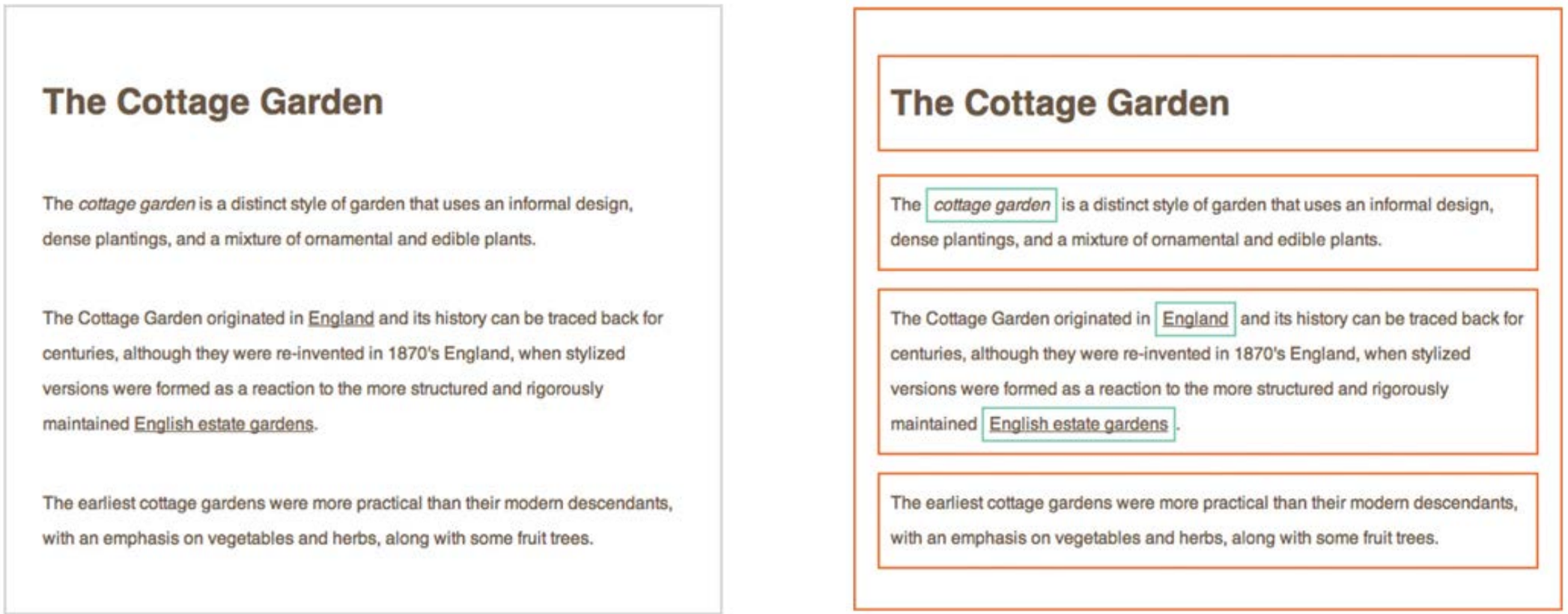
No Styles

This page uses DIV elements to group different sections of the HTML page. Click here to see how the page looks like with no stylesheet:
No Stylesheet.

Side-Bar

The CSS Box Model

- The key to understanding how CSS works is to imagine that there is an **invisible box around every HTML element.**



- CSS allows you to **create rules** that **control** the way that each **individual box** (and the **contents** of that box) is **presented.**

The CSS Box Model (Cont.)

- The CSS box model is essentially a box that wraps around HTML elements, and it consists of: **margins**, **borders**, **padding**, and the actual **content**.

- The box model allows us to add a border around elements, and to define space between elements.

The CSS Box Model (Cont.)

Explanation of the different parts:

- **Content** - The content of the box, where text and images appear

- **Padding** - Clears an area around the content. The padding is transparent

- **Border** - A border that goes around the padding and content

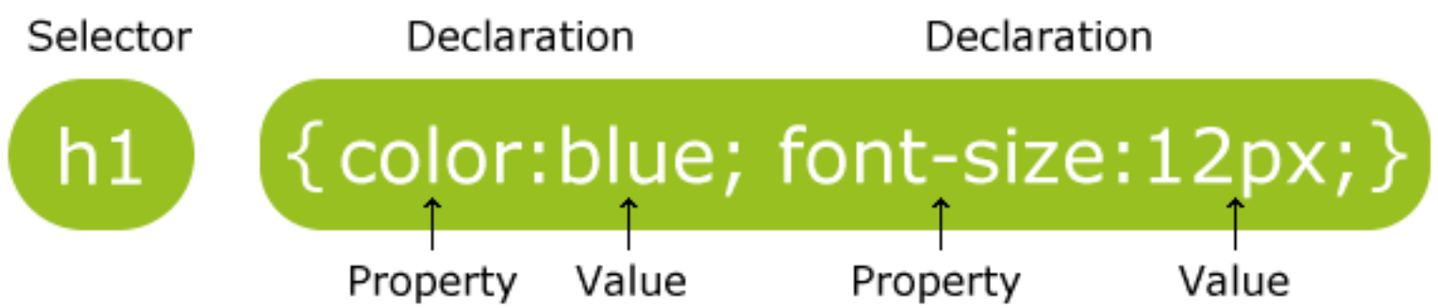
- **Margin** - Clears an area outside the border. The margin is transparent

From *Head first HTML and CSS*, by Robson, E., & Freeman, E., 2012, O'Reilly Media, Inc

Each statement in CSS consists of **a location** (like bedroom), a **property** in that location (like drapes or carpet), and a **style** to apply to that property (like the color blue, or 1 inch tiles).

CSS Syntax

- Each CSS consists of **a list of rules or rule sets.**
- A CSS rule set consists of a **selector** and a **declaration block**:



- The **selector** points to the HTML element you want to style.
- The **declaration block** contains one or more **declarations** separated by semicolons.
- Each declaration includes a **property name** and a **value**, separated by a **colon.**
- A CSS declaration always ends with a **semicolon**, and declaration groups are surrounded by curly braces.
- To make the CSS code more readable, you can put one declaration on each line.

CSS Selectors

- A CSS **selector** is the part of CSS rule set which **allows you to select** the content you want to style.

- CSS selectors are used to "find" (or select) HTML elements based on their id, class, type, attribute, and more.

The element Selector

- The **element/element type/type** selector selects elements based on the element **name/type.**

- **Example:** You can select all <p> elements on a page like this: (all <p> elements will be center-aligned, with a red text color)

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

The id Selector

- The **id selector** uses the **id attribute** of an HTML element to select a specific element.

- An **id** should be **unique within a page**, so the id selector is used if you want to **select a single, unique element.**

- To select an element with a specific id, write **a hash character**, followed by the id of the element.

- **Example:** The style rule below will be applied to the HTML element with **id="para1"**:

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

The class Selector

- The class selector **selects elements with a specific class attribute.**

- To select elements with a specific class, write a **period character**, followed by the **name of the class**:

- In the example below, all HTML elements with class="center" will be center-aligned:

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

- You can also specify that only specific HTML elements should be affected by a class.

- In the example below, all <p> elements with class="center" will be center-aligned:

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

Universal Selector

- Applies to **all HTML elements** in the document
- The universal selector is declared using an asterisk

```
* {
    color: green;
    font-size: 20px;
}
```

Grouping Selectors

- If you have elements with the **same style definitions**, like this:

```
h1 {
      text-align: center;
      color: red;
}
h2 {
      text-align: center;
      color: red;
}
p {
      text-align: center;
      color: red;
}
```

- You can group the selectors, to minimize the code.
- To group selectors, separate each selector with a comma.
- In the example below we have grouped the selectors from the code above:

```
h1, h2, p {
      text-align: center;
      color: red;
}
```

CSS Comments

- Comments are **used to explain your code**, and may help you when you edit the source code at a later date. Comments are ignored by browsers.

- A CSS comment **starts with /*** and **ends with */**. Comments can also span multiple lines:

```
p {
    color: red;
    /* This is a single-line comment */
    text-align: center;
}

/* This is
a multi-line
comment */
```

Three Ways of Inserting Style Sheets

There are three ways of inserting a style sheet:

- Internal style sheet
- External style sheet
- Inline style

Internal Style Sheet

- You define internal styles in the **head section** of an HTML page, inside the <style> tag, like this:

```
<head>
  <style>
   body {
     background-color: linen;
    }
    h1 {
     color: maroon;
     margin-left: 40px;
    }
  </style>
</head>
```

External Style Sheet

- You can create **a CSS file** to contain the style rule sets.
- The **<link> element** can be used in the HTML document **to tell the browser where to find the CSS file** used to style the page.
- **<link>** should use three attributes:
  - href

    This **specifies the path** to the CSS file (which is often placed in a folder called css or styles).
  - type

    This attribute **specifies the type of document** being linked to. The value should be text/css.
  - rel

    This **specifies the relationship between the HTML page and the file** it is linked to. The value should be **stylesheet** when linking to a CSS file.

External Style Sheet (Cont.)

- Each page must include a link to the style sheet with the **<link>** tag. The <link> tag goes **inside the head section**:

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

An example of a style sheet file called "myStyle.css", is shown below:

```
body {
    background-color: lightblue;
}
h1 {
    color: navy;
    margin-left: 20px;
}
```

Inline Style

- An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly!
- To use inline styles, add the **style attribute** to the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a h1 element:

```
<h1 style="color:blue;margin-left:30px;">This is a heading.</h1>
```

Why Use External Style Sheets?

- When building a website there are **several advantages** to placing your CSS rules in a separate style sheet.

- All of your web pages can **share the same style sheet.** This is achieved by using the <link> element on each HTML page of your site to link to the same CSS document.

- This means that the same code **does not need to be repeated** in every page (which results in less code and **smaller HTML pages**).

- Therefore, once the user has downloaded the CSS stylesheet, the rest of the site will **load faster.**

- If you want to make a change to how your site appears, you only need to **edit the one CSS file and all of your pages will be updated.**

- It is generally considered good practice to have the content of the site separated from the rules that determine how it appears.

When to Use Internal?

- If you have a **single document with a unique style.**
- If you are **just creating a single page**, you might decide to put the rules in the same file to keep everything in one place. (However, many authors would consider it better practice to keep the CSS in a separate file.)
- If you have **one page which requires a few extra rules** (that are not used by the rest of the site), you might consider using CSS in the same page. (Again, most authors consider it better practice to keep all CSS rules in a separate file.)

CSS Properties

Background Color

- The **background-color** property **specifies the background color** of an element.

- The background color of a page is set like this:

```
body {
    background-color: #b0c4de;
}
```

- With CSS, a color is most often specified by:
  - a HEX value - like "#ff0000"
  - an RGB value - like "rgb(255,0,0)"
  - a color name - like "red"

Background Image

- The **background-image** property **specifies an image to use as the background** of an element.

- By default, the **image is repeated** so it covers the entire element.

- The background image for a page can be set like this:

```
body {
    background-image: url("paper.gif");
}
```

Text Color

- The **color** property is used to **set the color of the text.**

Example

```
body {
    color: blue;
}

h1 {
    color: #00ff00;
}

h2 {
    color: rgb(255,0,0);
}
```

Text Alignment

- The **text-align** property is used to **set the horizontal alignment of a text.**

- Text can be **centered**, or aligned to the **left** or **right**, or **justified.**

- When text-align is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight.

Example

```
h1 {
     text-align: center;
}
```

Text Indentation

- The **text-indent** property is used to specify the **indentation of the first line of a text.**

Example

```
p {
      text-indent: 50px;
}
```

References

- www.w3schools.com
- Robson, E., & Freeman, E. (2012). *Head first HTML and CSS*. O'Reilly Media, Inc.
- Lazaris, L. (2013). *Jump Start CSS*. Sitepoint.
- Duckett, J. (2011). *HTML and CSS: Design and Build Websites*. John Wiley & Sons.
- Deitel & Deitel (2011). *Internet and World Wide Web How to Program, 5th Edition, Harvey & Paul Deitel & Associates.*