Cascading Style Sheets 3

At the end of this lesson, students should be able to:

- Explain different ways of cascading style sheets to customize HTML elements and precisely control the formatting of a web page.
- Write commonly used style properties such as precise font, size, color and other properties of displayed text and their associated values.
- Apply external style sheets to standardize the style for all pages in a website
- Write CSS animations.

1 Introduction

CSS is the abbreviation for Cascading Style Sheet. A style sheet simply holds a collection of rules that we define to enable us to manipulate our web pages. CSS can be applied to our pages in many ways, however the most powerful way to employ CSS rules is from an external cascading style sheet. When used in this manner the full power of CSS can be brought to control the design and appearance of our work from a single controlling location, which makes it easy to update our site on a global basis.

In this lesson, you will experiment using a few styles, and get a glimpse of just some of the many style options Cascading Style Sheets or CSS affords you. A style is simply a set of formatting instructions that can be applied to a piece of text. There are three mechanisms by which styles can be applied to HTML documents:

- Inline Style Sheets - The style can be defined within the basic HTML tags.
- Embedded Style Sheets - Styles can be defined in the `<head>` section and applied to the whole document.
- External Style Sheets - Styles can be defined in external files with extension .css which can then be used/called in any HTML document by including the style sheet via a URL.

.

1.1 Different Ways of Cascading Style Sheets

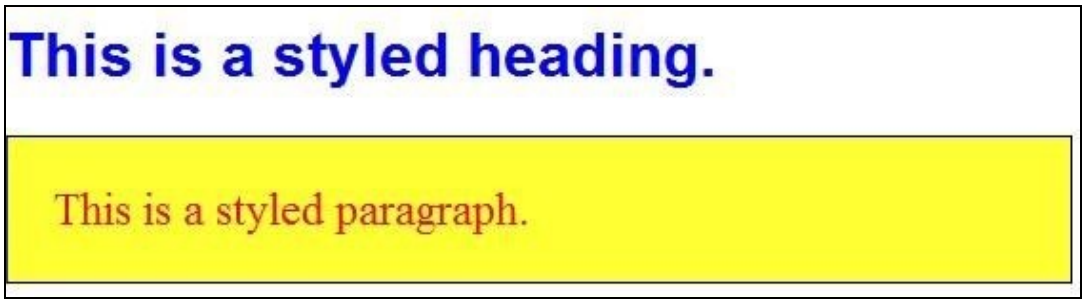
1.1.1 Defining Style Inline within HTML tags

1. With inline styles sheet, you can write CSS directly into an HTML tag. Inline style useful for once-off formatting and overriding previously defined properties in external or embedded style sheets. Enter the following HTML code in your text editor and save the file as **'inline.html'**.

```
<html>
<head>
<title>Inline Style</title>
</head>
<body>

<h1 style="color: blue; font-family: Arial;"> This is a styled heading.</h1>

<p style="background-color:#ff3; border:1px solid black; color:red; font-size:150%; padding:1em;">This is a styled paragraph.</p>

</body>
</html>
```

2. Modify and apply the rule to display the heading text in **green**. Save your file again and open it in the browser.

3. Test your document in the browser. The following screen capture shows the effect of the style on the '**inline.html'**.



1.1.2 Embedding a Style Sheet within HTML document

When you embed a style sheet, the browser will apply the style sheet's rules to the tags only on the page in which the style sheet declaration is inserted. You can apply CSS styles to any tag in a document.

1. The following is an example on how to set the colour of the heading text using CSS in an HTML document.

```
 1. <html>
 2. <head>
 3.   <title>Embed Style</title>
 4.   <style type="text/css">
 5.
 6. h1 {    font-family: Arial;
 7.         color: blue;     }
 8. p  {   background-color:#ff3;
 9.        border:1px solid black;
10.        color:red;
11.        font-size:150%;
12.        padding:1em;      }
13.
14.   </style>
15.
16. </head>
17.
18. <body>
19.
20. <h1> This is a styled heading.</h1>
21. <p>This is a styled paragraph.</p>
22.
23. </body>
24.
25. </html>
```

2. **Line 6-13**: You embed a style sheet, you need to insert the style sheet's rules between the start and end style tags (`<style>…</style>`) within the header section. The `type` attribute in the `<style>` tag tells the browser the type of style sheet to expect.

   Line 6 contains rules that set heading level 1 (`<h1>`) to blue colour.

3. Based on above style declaration, redefine the style attributes of the `<body>` tag. To do this, enter the above html tags into new file called '**embed.html**'. Using any text editor such as notepad, declare the CSS styles for `<body>` tag that support following property values:

   - **Font-family: Verdana, Helvetica, Courier**
   - **Font-size:12pt**
   - **Font color: yellow**
   - **Background-color: #800000**

To set multiple properties in a single rule, use the following form of style rule:

```
Selector {prop1:val1; prop2:val2; ……propN:valN;}
```

How do we ensure that we write the properties and values correctly?

- *Refer to CSS development books or websites like https://w3schools.com*

4. Test your document in a browser.

1.1.3 Attaching an External Style Sheet to HTML documents

---

1. The CSS styles you've created so far have only applied to one document. Internal style sheets apply only to the document in which they were created. Now, you'll learn how to create an external style sheet which contains the styles you defined in the document you created in before.

2. Rewrite all the style sheet information in **Exercise 1.1.2** in an external style sheet. Save your external file **mystyle.css**. Remember you do not need to copy the start and end `<style>` tags.

3. Validate your external style sheet with the W3C CSS Validation Services at http://jigsaw.w3.org/css-validator/.

4. Create a new html file called '**external.html'** and use the **<link>** tag to insert external style sheet on to your document. The link tag must be written in the following way

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

5. Open your document '**external.html**' in the browser. The browser should display the output of implementing the embedded style is the same as the **embed.html** in previous exercise.

1.1.4 Applying CSS ID Selectors and Classes

In the previous exercises, you have looked solely at HTML selectors - those that represent a HTML tag. You can also define your own selectors in the form of ID selectors and classes.

1. Open your Notepad and add the code below. Save the file as **selectors.html**.

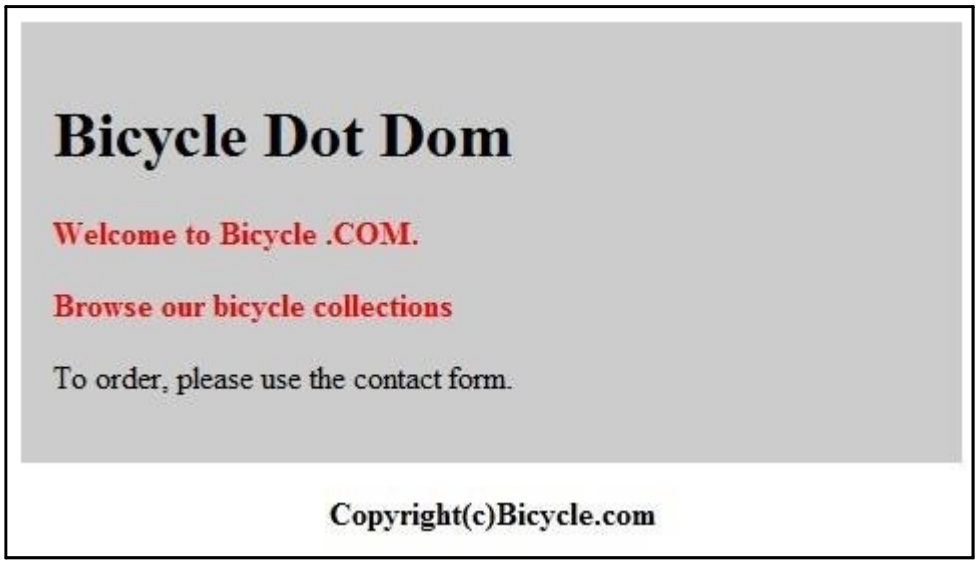
```
<html>
<head>
   <title>Selectors</title>
   <link rel="stylesheet" href="selectstyle.css" type="text/css">
</head>
<body>
   <div id="main">
   <h1>Bicycle Dot Com</h1>
   <p class="red">Welcome to Bicycle.com.  </p>
   <p class="red">Browse our bicycle collections </p>
   <p >To order, please use the contact form.</p>
   </div>
   <p class="center">Copyright(c)Bicycle.com</p>
</body>
</html>
```

2. Create a new file and insert the styles below. Save it as **selectstyle.css.**

```
#main { background-color: #ccc; padding: 1em }
```

```
.red {  color: red;  font-weight: bold;  }
.center{  textalign:center;  fontweight: bolder;  }
```

4. Test your HTML document in the browser.

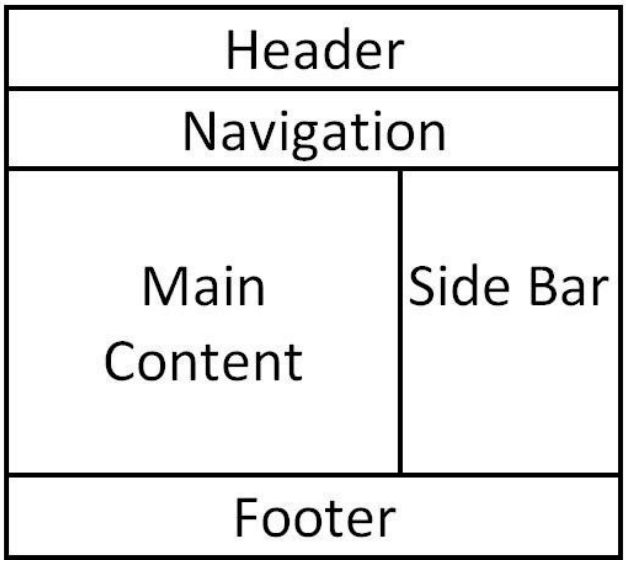


5. Analyze the HTML code. In CSS, an ID selector is a name preceded by a hash character (#) while a class selector is a name preceded by a full stop (.)The id selector is used to specify a style for a single, unique element. The class selector is used to specify a style for a group of elements.

1.1.5 Controlling Web Page Layout with HTML5 Elements and CSS

Every site or blog contains similar HTML basic body structure. All sites roughly contain the following sections to their coding: **Header, Navigation, Main Content, Sidebar,** and **Footer**. If you break down any site to its core, you will begin to see the structure are almost the same. It's the graphics, styling, and implementation of that structure that makes the website/blog all unique. View some examples here:

- http://www.1malaysia.com.my/
- http://www.headfirstlabs.com/



Header – Usually comprises of big pictures, graphics, or a logo of your site. This part is what most people like to exaggerate on as it's usually very appealing.

Navigation –The navigation part of your site. Your main links that will help navigate your audience through your site.

Main Content – The content section is usually where your main area will be that holds your content, posts, galleries and such forth.

Sidebar – Although not really a necessity (it's been a popular thing throughout many blogs), it contains additional information that may appeal to the user or even additional navigation items. (i.e Twitter Feed, Subscribe links, Search, Chatbox etc…)

Footer – And finally, the footer is usually the place where the copyright is held although lately it's been a trend where footers begin to hold extra information like recent posts, affiliates, short bios, footer links and the list goes on.

1. Open your Notepad and add the code below that consists of <header> and <footer> tags for the HTML document. Save the file as **layout.html**.

```
<!DOCTYPE html>
<html>
<head>
</head>
```

```
<body>
<div class="wrapper">
<header>
<h1>Yoko's Kitchen</h1> <nav>
<ul>
<li><a href="" class="current">home</a></li>
<li><a href="">classes</a></li>
<li><a href="">catering</a></li>
<li><a href="">about</a></li>
<li><a href="">contact</a></li>
</ul>
</nav>
</header>
<footer>&copy; 2023 Yoko's Kitchen</footer>
</div>
</body>
</html>
```

2. Test your HTML document in your browser. You'll get result as below.

3. Now, add 2 articles to the Web page as follows. Each article consists of a figure (`figure`) - an image with figure caption, a heading group (`hgroup`), and a paragraph (`p`). Add the code at the end of the `header` element.

```
<section class="courses">
<article>
     <figure>
     <img src="images/bok-choi.jpg" alt="Bok Choi" />
```

```
      <figcaption>Bok Choi</figcaption>
      </figure>
      <hgroup>
      <h2>Japanese Vegetarian</h2>
      <h3>Five week course in London</h3>
      </hgroup>
      <p>A five week introduction to traditional Japanese vegetarian meals, teaching you a selection of rice and noodle dishes.</p>
</article>
<article>
      <figure>
      <img src="images/teriyaki.jpg" alt="Teriyaki sauce" />
      <figcaption>Teriyaki Sauce</figcaption>
      </figure>
      <hgroup>
      <h2>Sauces Masterclass</h2>
      <h3>One day workshop</h3>
      </hgroup>
      <p>An intensive one-day course looking at how to create the most delicious sauces for use in a range of Japanese cookery.</p>
</article>
</section>
```

These articles need images to be displayed on the Web page. So, download the lab source file, and unzip the source into an images folder in your lab3 working folder (for example: lab3/images).

Save the layout.html, and check the output on your browser.

Output:

Yoko's Kitchen

- home
- classes
- catering
- about
- contact

Bok Choi

Japanese Vegetarian

Five week course in London

A five week introduction to traditional Japanese vegetarian meals, teaching you a selection of rice and noodle dishes.

Teriyaki Sauce

Sauces Masterclass

One day workshop

An intensive one-day course looking at how to create the most delicious sauces for use in a range of Japanese cookery.

4. Next, add <aside> element. The <aside> element has two purposes, depending on whether it is inside an <article> element or not. When the <aside> element is used inside an <article> element, it should contain information that is related to the article but not essential to its overall meaning. For example, a pullquote or glossary might be considered as an aside to the article it relates to. When the <aside> element is used outside of an <article> element, it acts as a container for content that is related to the entire page. For example, it might contain links to other sections of the site, a list of recent posts, a search box, or recent tweets by the author.

```
<aside>
        <section class="popular-recipes">
        <h2>Popular Recipes</h2>
        <a href="">Yakitori (grilled chicken)</a>
        <a href="">Tsukune (minced chicken patties)</a>
        <a href="">Okonomiyaki (savory pancakes)</a>
        <a href="">Mizutaki (chicken stew)</a>
         </section>
        <section class="contact-details">
        <h2>Contact</h2>
        <p>Yoko's Kitchen<br />
            27 Red Street<br />
            Shoreditch<br />
            London E2 7DP</p>
        </section>
</aside>
```

5. Now, you're going to style the page using CSS. Insert the following CSS in the head section of the HTML document.

```
<style type="text/css">
      header, section, footer, aside, nav, article, figure, figcaption {
        display: block;}
      body {
        color: #666666;
        background-color: #f9f8f6;
        background-image: url("images/dark-wood.jpg");
        background-position: center;
        font-family: Georgia, times, serif;
        line-height: 1.4em;
        margin: 0px;}
      .wrapper {
        width: 940px;
        margin: 20px auto 20px auto;
        border: 2px solid #000000;
        background-color: #ffffff;}
      header {
        height: 160px;
        background-image: url(images/header.jpg);}
      h1 {
        text-indent: -9999px;
        width: 940px;
        height: 130px;
        margin: 0px;}
      nav, footer {
```

```
      clear: both;
      color: #ffffff;
      background-color: #aeaca8;
      height: 30px;}
    nav ul {
      margin: 0px;
      padding: 5px 0px 5px 30px;}
    nav li {
      display: inline;
      margin-right: 40px;}
    nav li a {
      color: #ffffff;}
    nav li a:hover,
    nav li a.current {
      color: #000000;}
    section.courses {
      float: left;
      width: 659px;
      border-right: 1px solid #eeeeee;}
    article {
      clear: both;
      overflow: auto;
      width: 100%;}
    hgroup {
      margin-top:40px;}
    figure {
      float: left;
      width: 290px;
      height: 220px;
      padding: 5px;
      margin: 20px;
      border: 1px solid #eeeeee;}
    figcaption {
      font-size: 90%;
      text-align: left;}
    aside {
      width: 230px;
      float: left;
      padding: 0px 0px 0px 20px;}
    aside section a {
      display: block;
      padding: 10px;
      border-bottom: 1px solid #eeeeee;}
    aside section a:hover {
      color: #985d6a;
      background-color: #efefef;}
    a{
      color: #de6581; text-decoration: none;}
    h1, h2, h3 {
      font-weight: normal;}
    h2 {
      margin: 10px 0px 5px 0px;
      padding: 0px;}
    h3 {
      margin: 0px 0px 10px 0px;
      color: #de6581;}
    aside h2 {
      padding: 30px 0px 10px 0px;
      color: #de6581;}
    footer {
      font-size: 80%;
      padding: 7px 0px 0px 20px;}
  </style>
```

6. Save your HTML document and test it in your browser.

Output:

1.1.6 CSS Animation

An animation lets an element gradually change from one style to another. To use CSS animation, you must first specify some keyframes for the animation. Keyframes hold what styles the element will have at certain times.

CSS Animation Properties
• @keyframes
• animation-name
• animation-duration
• animation-delay
• animation-iteration-count
• animation-direction
• animation-timing-function
• animation-fill-mode
• animation

tiger.html

```
<!DOCTYPE html>
<html>
<head>
```

```
<link rel="stylesheet" href="tiger.css">
</head>
<body>
<h1>CSS Animation</h1>
<div id='sprite_anim'></div>
</body>
</html>
```

tiger.css

```
#sprite_anim {
  width: 180px;
  height: 158px;
  background: url("tiger1.jpeg") 0px 0px;
  animation-name: walking;
  animation-duration: 1s;
  animation-timing-function: steps(6);
  animation-delay: 0;
  animation-iteration-count: infinite;
  animation-direction: normal;
}

@keyframes walking {
  100% { background-position: -1075px;}
}
```

tiger1.jpeg

Output:

CSS Animation