CP 160- Web Programming and Design

Lab 4: CSS Basics

Hands-On Practice 1: Configure Embedded CSS

The web page uses embedded styles to set the text color and background color of the web page document with the body element selector.

Launch a text editor and open the **starter.html** file. Save your page as **embedded.html** and test it in a browser. Your page should look similar to the one shown in **Figure 1**.

In this Hands-On Practice, you'll code embedded styles to configure selected background and text colors. You'll use the body element selector to configure the default background color (#F9F0FE) and default text color (#5B3256) for the entire page. You'll also use the h1 and h2 element selectors to configure different background and text colors for the heading areas. Edit the embedded.html file in a text editor and add the following code in the **head** section above the closing `</head>` tag:

```
<style>
body { background-color: #F9F0FE;    color: #5B3256; }
h1 { background-color: #833B83;      color: #F9F0FE; }
h2 { background-color: #AD77C3;      color: #F9F0FE; }
</style>
```

Save and test your file in a browser.

Figure 1 The web page without any styles.

Figure 2 displays the web page along with color swatches. A monochromatic color scheme was chosen. Notice how the repetition of a limited number of colors adds interest and unifies the design of the web page.

Figure 2 The web page after embedded styles are configured.

Hands-On Practice 2: Configure External CSS

The flexibility and power of CSS are best utilized when the CSS is external to the web page document. An external style sheet is a text file with a .css file extension that contains CSS style rules. The external style sheet file is associated with a web page using the link element. This provides a way for multiple web pages to be associated with the same external style sheet file. The external style sheet file does not contain any HTML tags—it only contains CSS style rules.

Let's practice using external styles. First, you'll create an external style sheet. Then you'll configure a web page to be associated with the external style sheet. **Create an External Style Sheet.** Launch a text editor and type style rules to set the background color of a page to blue and the text color to white. Save the file as color.css. The code follows:

```
body { background-color: #0000FF; color: #FFFFFF; }
```

Figure 3 shows the external color.css style sheet displayed in Notepad. Notice that there is no HTML in this file. HTML tags are not coded within an external style sheet. Only CSS rules (selectors, properties, and values) are coded in an external style sheet.

Figure 3 The external style sheet color.css.

Configure the Web Page. To create the web page shown in **Figure 4**, launch a text editor and open the **template.html** file. Modify the title element, add a link tag to the head section, and add a paragraph to the body section as indicated by the following highlighted code:

```
<!DOCTYPE html>

<html lang="en">

<head>

<title>External Styles</title>

<meta charset="utf-8">

<link rel="stylesheet" href="color.css">

</head>

<body>
```
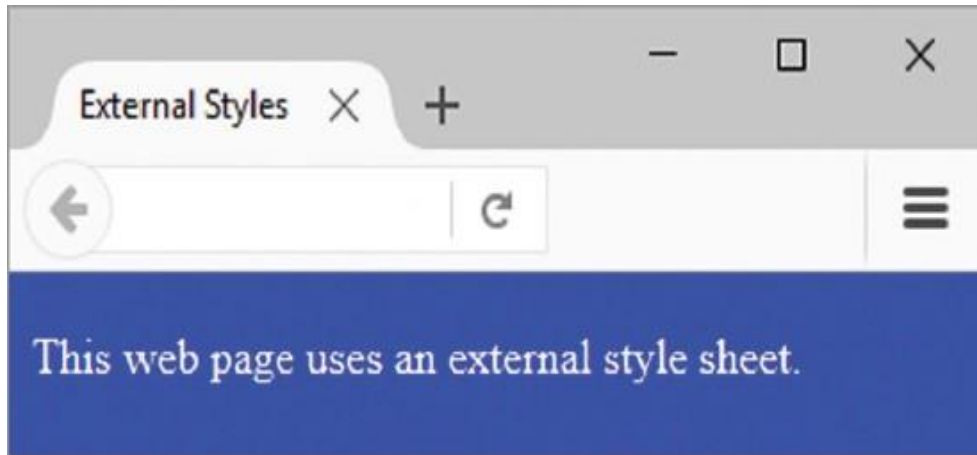
```
<p>This web page uses an external style sheet.</p>

</body>

</html>
```


Figure 4 This page is associated with an external style sheet.

Save your file as external.html in the same folder as your color.css file. Launch a browser and test your page. It should look similar to the page shown in **Figure 4**.

The color.css style sheet can be associated with any number of web pages. If you ever need to change the style of formatting, you only need to change a single file (color.css) instead of multiple files (all of the web pages). As mentioned earlier, this technique can boost productivity on a large site. This is a simple example, but the advantage of having only a single file to update is significant for both small and large websites.

Hands-On Practice 3: Configure a class and an id

In this Hands-On Practice, you will modify the Trillium Media Design page while you practice configuring a class and an id. Launch a text editor and open the **embedded.html** file. Save the file as **classid.html**.

Configure the CSS. Edit the embedded CSS in the head section of the web page as you configure a class named `feature` and an id named `new`.

1. Create a class named `feature` that configures red (#B33939) text. Add the following code to the embedded styles in the head section of the web page:

```
.feature { color: #B33939; }
```

2. Create an id named `new` that configures a medium blue text color. Add the following code to the embedded styles in the head section of the web page:

```
#new { color: #227093; }
```

Configure the HTML. Associate HTML elements with the class and id you just created.

1. Modify the last two <li> tags in the unordered list. Add a class attribute that associates the <li> with the feature class as follows:

```
<li class="feature">Usability Studies</li>

<li class="feature">Search Engine Optimization</li>
```

2. Modify the second opening paragraph tag. Add an id attribute that associates the paragraph with the id named `new`:

```
<p id="new">
```

Save your classid.html file and test it in a browser. Your page should look similar to the image shown in **Figure 5**. Notice how the class and id styles are applied.

Figure 5 CSS class and id selectors are used on this page.

Hands-On Practice 4: Span Element

In this Hands-On Practice, you will experiment with span elements in the Trillium Media Design home page. Launch a text editor and open the **starter.html** file. Save your file as **span.html** and test it in a browser. Your page should look similar to the one shown in **Figure 1**. Open span.html in a text editor and view the source code. In this Hands-On Practice, you'll code embedded styles to configure selected background and text colors. You'll also add `<span>` tags to the web page.

When you are finished with the first part of this Hands-On Practice, your web page will be similar to **Figure 6**.

Figure 6 This page uses the span element.

Configure the Embedded Styles. Edit span2.html in a text editor and add embedded styles in the head section above the closing `</head>` tag. You will configure styles for a class named `companyname` and for the body, h1, h2, nav, and footer element selectors. The code is

```
<style> body { background-color: #F7F7F7;        color: #191970; } h1
{ background-color: #833B83;        color: #F9F0FE; } h2 { color:
#AD77C3; } nav { background-color: #EAEAF2; } footer { color:
#666666; }

.companyname { color: #833B83; }

</style>
```

Configure the Company Name. View **Figure 6** and notice that the company name, Trillium Media Design, is displayed in a different color than the other text within the first paragraph. You've already created a class named `companyname` in the CSS. You'll use a span to apply this formatting. Find the text "Trillium Medium Design" in the first paragraph. Configure a span element to contain this text. Assign the span to the class named `companyname`. A sample code excerpt follows:

```
<p><span class="companyname">Trillium Media Design</span> will bring
```

Save your file and test in a browser. Your page should look similar to the one shown in **Figure 6**.

Hands-On Practice 5: Cascade

Figure 7 shows the "cascade" (**order of precedence**) that applies the styles from outermost (external styles) to innermost (inline styles).
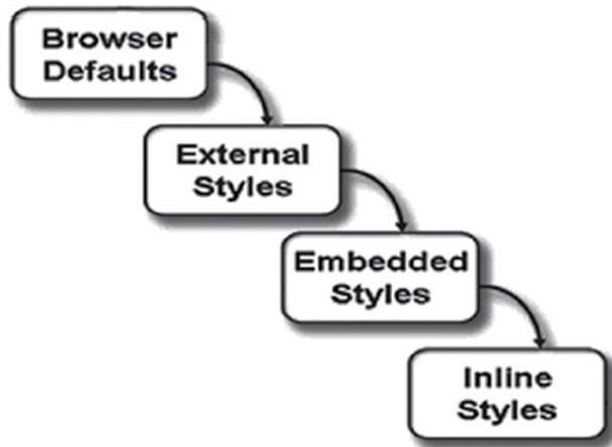

Figure 7 The cascade.

You will experiment with the "cascade" in this Hands On Practice as you work with a web page that uses external, embedded, and inline styles.

1. Create a new folder named **mycascade**.
2. Launch a text editor. Open a new file. Save the file as **site.css** in the mycascade folder. You will create an external style sheet that sets the background color of the web page to a shade of yellow (#FFFFCC) and the text color to black (#000000). The code follows:

   ```
   body { background-color: #FFFFCC; color: #000000; }
   ```

   Save and close the site.css file.

3. Open a new file in the text editor and save it as index.html in the mycascade folder. The web page will be associated with the external style sheet site.css, use embedded styles to set the global text color to blue, and use inline styles to configure the text color of the second paragraph. The file index.html will contain two paragraphs of text. The code for index.html follows:

   ```
   <!DOCTYPE html>

   <html lang="en">

   <head>

   <title>The Cascade in Action</title>

   <meta charset="utf-8">

   <link rel="stylesheet" href="site.css">
   ```

```
<style>

body { color: #0000FF; }

</style>

</head>

<body>

<p>This paragraph applies the external and embedded styles &mdash;
note how the blue text color that is configured in the embedded
styles takes precedence over the black text color configured in the
external stylesheet.</p>

<p style="color: #FF0000">Inline styles configure this paragraph to
have red text and take precedence over the embedded and external
styles.</p>

</body>

</html>
```

4. Save index.html and display it in a browser. Your page should look similar to the sample shown in **Figure 8**.
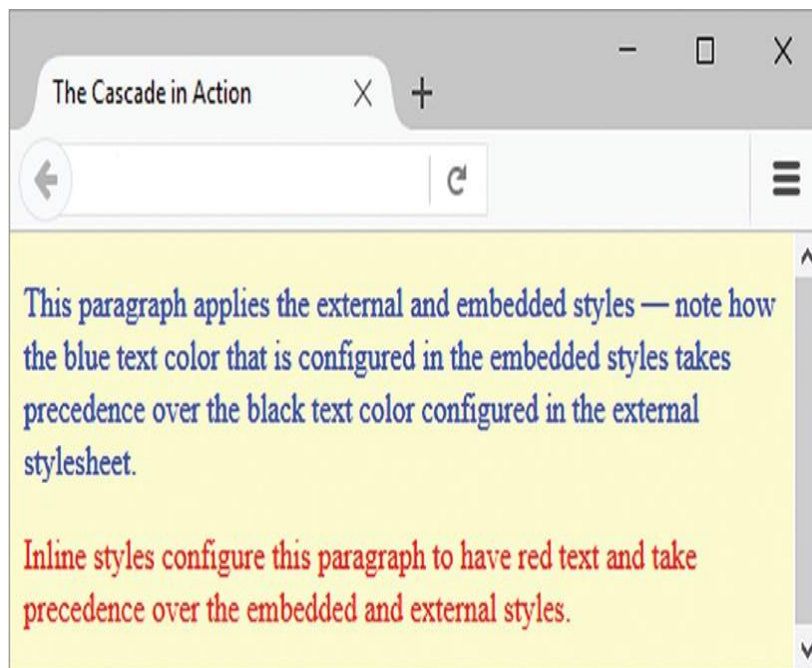

Figure 8 The cascade in action.

The web page picked up the yellow background from the external style sheet. The embedded style configured the text to be the color blue, which overrides the black text color in the external style sheet. The first paragraph in the web page does not contain any inline styles, so it inherits the style rules in

the external and embedded style sheets. The second paragraph contains an inline style of red text color; this setting overrides the corresponding external and embedded styles.

Lab Completion / Submission:

Complete all the lab practices. Take the screenshots of your completed webpages; put them into a single word file and submit it to **Blackboard -> CP160 -> Assessments -> Lab 4**; due date: today

After-lab Assignment:

1. Create a one-paragraph conclusion of what you have learned during the lab today.

2. Website design evaluation

    Choose a website to evaluate from the following list of URLs or any other one that you like:
    https://www.confederationcollege.ca/
    https://www.canada.ca/en/department-national-defence.html
    https://www.cbc.ca/
    https://www.thunderbay.ca/
    https://www.tbaytel.net/
    https://www.toysrus.ca/en/home
    https://www.usnews.com/
    http://www.arm.gov
    http://www.telework.gov
    https://www.w3.org/

    Visit the website you are evaluating. Write a short paper (1-2 pages) that includes the following information:
    a. URL of the website
    b. Name of the website
    c. Target audience
    d. Describe how the design principles of contrast, repetition, alignment, and proximity are applied.
    e. Recommend improvements for the website.

Submission:

Put 1 and 2 into a single word or pdf file and submit it to **Blackboard -> CP160 -> Assessments -> Lab 4**; due date: one week

3. **Keep working on your Project Proposal/Plan**

The purpose of this Web Project is to design a website using recommended design practices and skills. Your website might be about a favorite hobby or subject, your family, a church or club you belong to, a company that a friend owns, the company you work for, and so on.

Your website will contain a home page and at least six content pages. It is a semester-long project in which you design, create and publish an original website.

By the due date of **Oct.21**, you should have accomplished the following milestones:

- **Web Project Topic Selection**
- **Web Project Planning Analysis**
- **Web Project Site Map**
- **Web Project Page Layout Design**

In the proposal document that you will submit to Blackboard, you should include the items listed above. Use the following as your reference when you work on the website and prepare the document:

1) **Web Project Topic.** Write 1-2 pages in your document with a discussion of the following items

   ▪ What is the name and purpose of the site?

   List the website name and the reasons you are creating the site.

   ▪ What do you want the site to accomplish?

   Explain the goal you have for the site. Describe what needs to happen for you to consider your site a success.

   ▪ Who is your target audience?

   Describe your target audience by age, gender, socioeconomic characteristics, and so on.

   ▪ What opportunity or issue is your site addressing?

   Note: Your site might be addressing the opportunity of providing information about a topic to others, creating an initial web presence for a company, and so on.

   ▪ What type of content might be included in your site?

   Describe the type of text, graphics, and media you will need for the site.

   ▪ List at least two related or similar sites found on the Web.

2) **Web Project Planning Analysis.** Write 1-2 pages in your document with a discussion of the following items.

  ■ **What results do I want to see?**

  List the working title of each page on your site. A suggested project scope is seven to eleven pages.
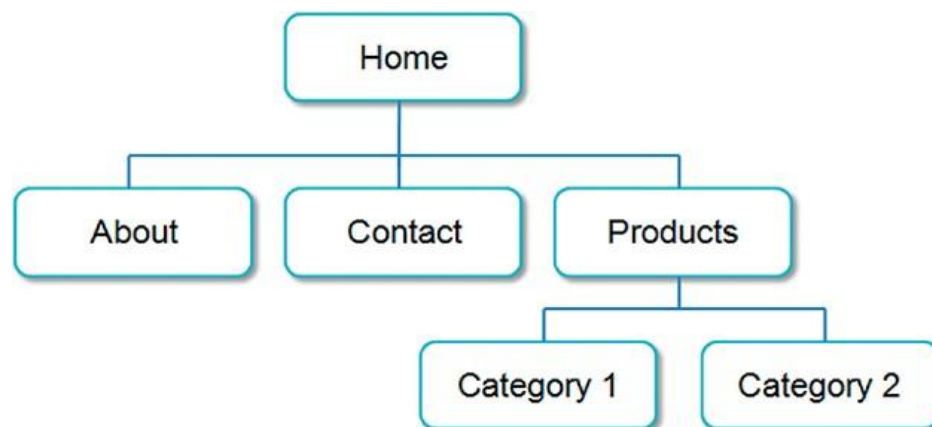
  ■ **What information do I need?**

  List the sources of the content (facts, text, graphics, sounds and video) for the web pages you listed. While you should write the text content yourself, you may use outside sources for royalty-free images and multimedia. Review copyright considerations.

  ■ **Schedule**

  By when to complete what? Include your planned schedule in this page.

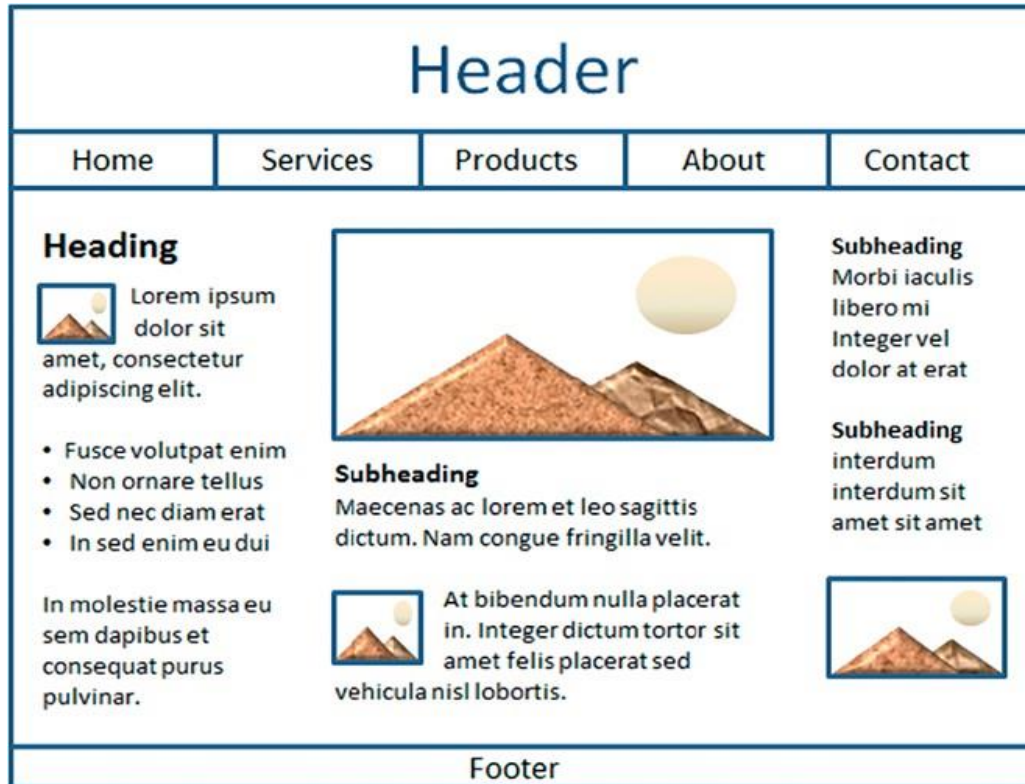3) **Web Project Site Map.** Use the drawing features of a word processing program, or other graphic application to create a site map of your website that shows the hierarchy of pages and relationships between pages. Refer to the following site map example:

In this document, you can have a high-level/draft site map which may not include every detail. In future you can continue to update/improve it.

4) **Web Project Page Layout Design.** Use the drawing features of a word processing program or a graphic application to create wireframe page layouts for the home page and content pages of your site. Indicate where the logo, navigation, text, and images will be located. Do not worry about exact wording or exact images.

An example of style for page layout composition is shown in the following:

Header

| Home | Services | Products | About | Contact |

Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

- Fusce volutpat enim
- Non ornare tellus
- Sed nec diam erat
- In sed enim eu dui

In molestie massa eu sem dapibus et consequat purus pulvinar.

Subheading
Maecenas ac lorem et leo sagittis dictum. Nam congue fringilla velit.

At bibendum nulla placerat in. Integer dictum tortor sit amet felis placerat sed vehicula nisl lobortis.

Subheading
Morbi iaculis libero mi Integer vel dolor at erat

Subheading
interdum interdum sit amet sit amet

Footer

Again, in this document, you may not include detail in the layout design. A draft layout design is fine in this phase.