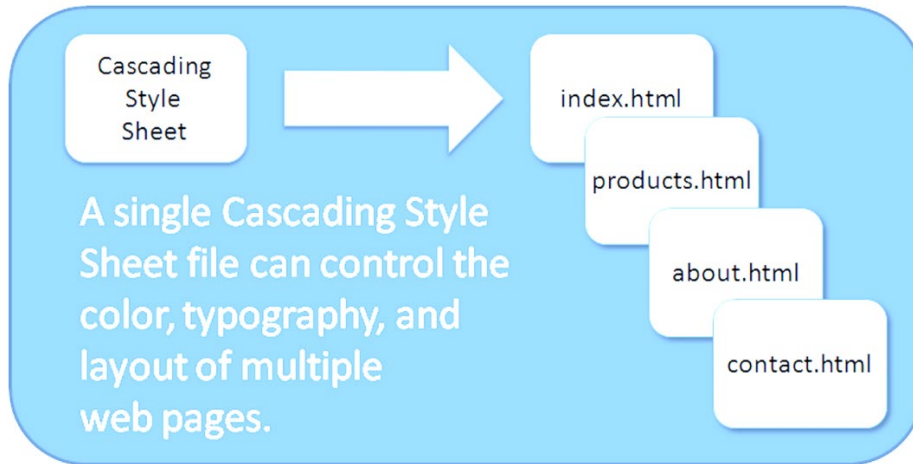Css – Cascading Style Sheets – allows developers to apply typographic styles (typeface, font size, style, etc.) and page layout instructions to cascade to all associated web pages associated with a website. This can result in an easy to manage website and ensure consistency of style, color scheme, and layout.

Cascading Style Sheet → index.html
products.html
about.html
contact.html

A single Cascading Style Sheet file can control the color, typography, and layout of multiple web pages.

Advantages of Cascading Style Sheets

❖ **Typography and page layout can be better controlled.** This includes font size, line spacing, letter spacing, indents, margins, and element positioning.

❖ **Style is separate from structure.** The format of the text and colors used on the page can be configured and stored separately from the body section of the web page document.

❖ **Styles can be stored.** Multiple styles (color, design, etc.) can be applied quickly as desire changes. The HTML coding remains intact so that if a client decides to completely change the look and layout of their website then one file change is needed instead of modifying the code of each web page.

❖ **Documents are potentially smaller.** The formatting is separate from the document; therefore, the actual documents should be smaller.

❖ **Site maintenance is easier.** When styles need to be changed then modifications can be completed by changing one line of code to associate the new CSS file. Do you think it would be easier to work together in teams also?

Methods to Incorporate CSS Technology

1. **Inline styles** are coded
   a. **in the body of the web page**
   b. as an **attribute of an HTML** tag.
   c. The style **applies only to the specific element** that contains it as an attribute.
   d. An inline style (inside an HTML tag) has the **highest priority**, which means that it **will override a style** defined inside the <head> tag, or in an external style sheet, or in a browser (a default value).
   e. Examples:
      <p style="color:orange; text-align:center">This is a paragraph.</p>
      <h1 style="color:#ff0000">Heading text is red</h1>
            Heading text is red
      Separate style rule declarations with ;
      <h1 style="color:#FF0000;background-color:#cccccc">
      This is displayed as a red heading with gray background</h1>
2. **Embedded styles** (*internal styles*) are
   a. defined within a style element **in the "HEAD" section** of a web page
   b. an HTML style element.
   c. These global rules may apply to the entire web page. They can be overridden for <h1> and <h2> elements by the h1 and h2 style rules.
   d. Example 1: **<style> body { background-color: #000000; color: #FFFFFF;} <style>**
      Example 2: **<head>**
                        <style type="text/css">
                           h2{color:blue;}
                           body{background-image:url('images/back40.gif');}
                        </style>
                        </head>
3. *External styles* are
   a. coded in a **SEPARATE TEXT FILE**.
   b. Text file with **.css** file extension which contains *only style rules* is
   c. associated with the web page by *configuring a link element in the head section.* It *does not contain any HTML tags.* (Relationship = rel)
   d. Example:
      <head>
              <link rel="stylesheet" type="text/css" href="styles1.css">
      </head>

4. **Imported styles** are like external styles in that they connect styles coded in a separate text file with a web page document. It can be imported into embedded styles or into another external style sheet by using the @import directive. We'll concentrate on the other three types of styles.

CSS Selectors and Declarations

Style sheets are composed of style rules that describe the styling to be applied. Each *rule* has (2) parts: a **selector** and a **declaration**.

In this figure, the CSS rule sets the color of the text used on a web page to blue. The selector is the body tag, and the declaration sets the color property to the value of blue.

body { color: blue }

Selector   Declaration Property   Declaration Value

- ❖ **CSS Style Rule <u>Selector</u>.** The selector can be configured for:
  - o **HTML element name selector**
  - o **class name selector** – used for paragraphs, links, buttons, input boxes, etc.
    - ▪ apply a CSS style rule to a certain "class" of multiple elements
    - ▪ does not associate the style to a specific HTML element
    - ▪ Example: Configure with .classname (class is preceded by .(dot))

            <style>
                    .new { color: #FF0000;
                    font-style: italic;}
            </style>
      Above CSS code to create a class called "new" with red italic text.
      To apply the class: <p class="new">This is text is red and in italics</p>
      This is text is read and in italics
  - o **id name selector** – appear on page once like header, footer, menu, etc.
    - ▪ Apply a CSS rule style to ONE unique element on a web page.
    - ▪ Configure with #idname (id is preceded by #(hash)).
    - ▪ Ex: Code CSS to create an id called "new" with red, large, italic text.

            <style>
                    #new { color: #FF0000;
                    font-size:2em;
                    font-style: italic; }
            </style>
      Apply the id: <p id="new">This is text is red, large, and in italics</p>
      This is text is red, large, and in italics
  - o **descendant selector -** Specify an element within the context of its container (parent) element. AKA contextual selector
  - o Advantage of contextual selectors: Reduces the number of classes and ids you need to apply in the HTML
  - o The below example configures a green text color only for p tags located *within* an element assigned to the id named content

        <style>
                #content p { color: #00ff00; }
        </style>

❖ **CSS Style Rule** <u>Declaration.</u> The declaration indicates the CSS *property* being set (such as color) and the value you are assigning to the property.
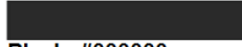

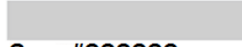## *Using Color on Web Pages*

Red: #FF0000

Green: #00FF00

Blue: #0000FF

Black: #000000

White: #FFFFFF

Grey: #CCCCCC

RGB Color - The values of red, green, and blue vary from 0 to 255.

Hexadecimal numbers (base 16) represent these color values.
　# indicates a hexadecimal value
　Hex value pairs range from 00 to FF
　Three hex value pairs describe an RGB color

Web Safe Colors - A collection of 216 colors. Display the most similar on the Mac and PCplatforms.
　Hex values: 00, 33, 66, 99, CC, FF
　Color Chart - http://webdevfoundations.net/color
　Monochromatic (HSL – Hue, Saturation, & Lightness important) – http://www.w3.org/TR/css3-color/#hsl-color

<u>**Another Embedded Example Using Color:**</u>

```
<style>
      body { background-color: #E6E6FA; color: #191970;}
      h1 { background-color: #191970; color: #E6E6FA;}
      h2 { background-color: #AEAED4; color: #191970;}
</style>
```

<u>**Another External Style Sheets Example:**</u> Multiple web pages can associate with the same external style sheet file.➔ Apply to index.html, clients.html, about.html, etc.

Below is the text within the site.css file:

```
      body {background-color:#E6E6FA;
            color:#000000;
            font-family: Arial, sans-serif;
            font-size:90%; }
      h2 {  color: #003366; }
      nav { font-size: 16px;
            font-weight: bold; }
```

Below is the text within the <head> section of the index.html file. A self-contained tag with the purpose of associating the site.css file with the website. Multiple .css style files can be created. Editing the below code can be an efficient means of completely changing styles.

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

Common CSS Properties, including:

background-color

color

font-family – configures the font typeface of text

font-size – configures the size of text – text values, em units, px units, pt units, and percentage are the options – for accessibility use em or percentage which can easily be enlarged by all browsers

font-style – configures text to an italic style

font-weight – configures the boldness of the text

line-height – configures the height of the line of text

margin

text-align – configures alignment of text within a block display element

text-decoration – modifies the appearance of text with an underline, overline, or line-through

width

text-align – configures alignment of text within a block display element

text-indent – configures the indentation of the first line of text

text-transform – configures the capitalization of text

letter-spacing – configures space between text characters

word-spacing – configures space between words

text-shadow – configures a drop shadow on text

container – centers page content - example #container {margin-left: auto; margin-right: auto; width: 80%;}

span – configures a specially formatted area displayed in-line with other elements, such as within a paragraph. There is no additional empty space above or below a span – it is inline display.

Example: Embedded CSS:

```
<style>
        .companyname { font-weight: bold;
                font-family: Georgia, "Times New Roman", serif;
                font-size: 1.25em; }
</style>
```

HTML: <p>Your needs are important to us at Acme Web Design.
We will work with you to build your Web site.</p>
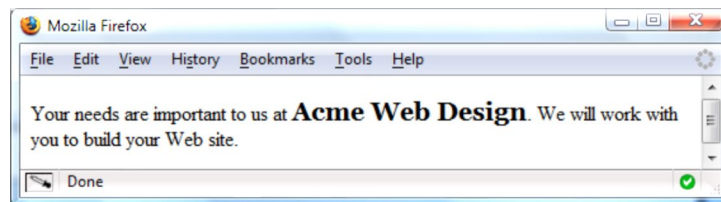
Using Fonts on Web Pages

Font-family Category	Font Typeface
serif	Times New Roman, Georgia, Times
sans-serif	Arial, Verdana, Geneva
monospace	Courier New, Lucida Console
cursive	Brush Script MT, Comic Sans MS
fantasy	Jokerman, Curlz MT

Not everyone has the same fonts installed in their computer
Configure a list of fonts and include a generic family name

p { font-family: Arial, Verdana, sans-serif; }

The "Cascade"
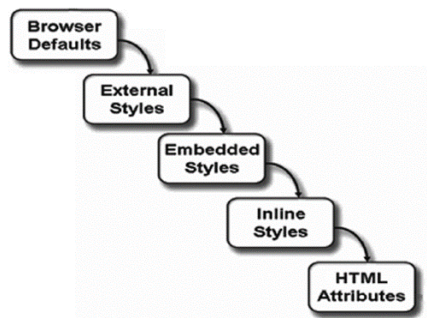
External styles can apply to multiple pages. If a web page contains both a link to an external style sheet followed by embedded styles, the external styles will be applied first, and then the embedded styles will be applied. This approach allows a web developer to override global external styles on selected pages.

If a web page contains both embedded styles and inline styles, the embedded styles are applied first, and then the inline styles are applied. This approach allows a web developer to override pagewide styles for particular HTML tags or classes.

Any HTML tag or attribute will override styles. For example, a <strong> tag will override corresponding font-related styles configured for an element. If no attribute or style is applied to an element, the browser's default is applied. However, *the appearance of the browser's default may vary by browser,* and you might be disappointed with the result. Use CSS to specify the properties of your text and web page elements. Avoid depending on the browser's default.

CSS Border Property

Configures a border on the **top, right, bottom, and left** sides of an element
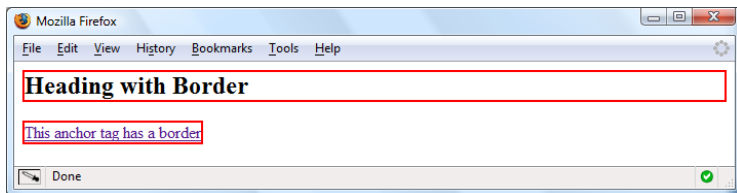
Consists of
> border-width
> border-style
> border-color

h2 { border: 2px solid #ff0000 }

Block display element
- ◦ Default width of element content extends to browser margin (or specified width)
- ◦ h2 { border: 2px solid #ff0000; }

Inline display element
- ◦ Border closely outlines the element content
- ◦ a { border: 2px solid #ff0000; }

Use CSS to configure a line on **one or more sides** of an element
border-bottom
border-left
border-right
border-top

Example: **h2 { border-bottom: 2px solid #ff0000 }**   **Heading with Border**

CSS padding Property - Configures empty space between the content of the HTML element and the border. Set to 0px by default.

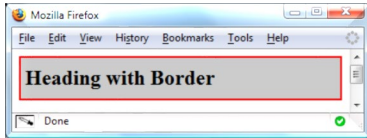h2 { border: 2px solid #ff0000; padding: 5px; }

Use CSS to configure padding on one or more sides of an element
> padding-bottom
> padding-left
> padding-right
> padding-top

```
h2 { border: 2px solid #ff0000;
        background-color: #cccccc;
        padding-left: 5px;
        padding-bottom: 10px;
        padding-top: 10px; }
```

CSS padding Property Shorthand: Two numeric values or percentages
first value configures top and bottom padding
the second value configures left and right padding

```
h2 { border: 2px solid #ff0000;
     background-color: #cccccc;
     padding: 20px 10px;  }
```

CSS padding Property Shorthand: Four numeric values or percentages
Configure top, right, bottom, and left padding

```
h2 { border: 2px solid #ff0000;
         width: 250px;
     background-color: #cccccc;
     padding: 30px 10px 5px 20px; }
```
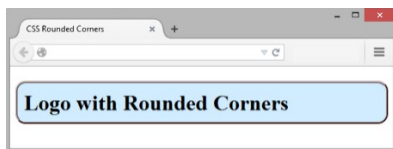
Configure CSS to eliminate the border
```
img { border-style: none; }
```

CSS3 Rounded Corners - border-radius property
        Configures the horizontal radius and vertical radius of the corner
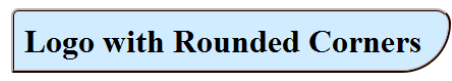   Numeric value(s) with unit (pixel or em) or percentage
```
h1 {  border-radius: 15px; }
```


Four values for border-radius configure each corner separately
Ordered by top left, top right, bottom right, bottom left
```
        border-radius: 15px 30px 100px 5px;
```

Organizing Your Site – Creating links to Files within Folders

- Place images in their own folder
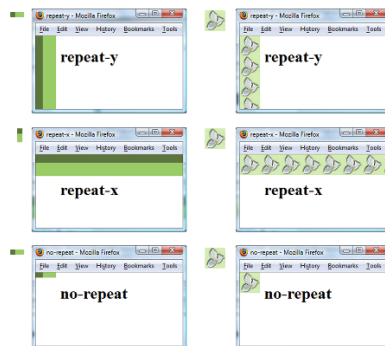- Code the path to the file in the src attribute

<img src="images/home.gif" alt="Home" height="100" width="200">

CSS Imaging

CSS background-image Property

Configures a background-image

By default, background images tile (repeat)

body { background-image: url(background1.gif); }


trilliumbullet.gif:

h2 { background-color: #d5edb3;
    color: #5c743d;
    font-family: Georgia, "Times New Roman", serif;
    padding-left: 30px;
  background-image: url(trilliumbullet.gif);
  background-repeat: no-repeat; }

```css
body { background-color: #f4ffe4; color: #333333;
        background-image: url(trilliumgradient.png);
        background: url(trilliumfoot.gif)
              no-repeat  bottom right,
              url(trilliumgradient.png); }
```
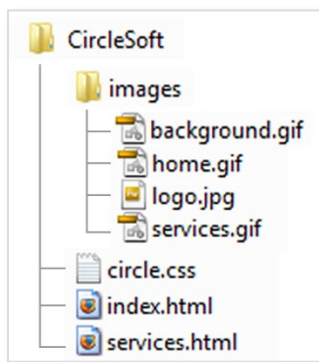
Now that we progressed from a web page to a website, it is a good idea to keep the files organized. When organizing your website, place all of your images and CSS files into separate folders.

• Place images in their own folder
• Code the path to the file in the src atttribute

<img src="images/home.gif" alt="Home" height="100" width="200">

You will be allowed some class time to complete this assignment. You may need to also set aside some time outside of the class time to complete the entire assignment.

In this course, you are expected to do some analytical thinking as well as trouble-shooting of your code (debugging). Attempt to resolve you problem then ask for assistance.

When learning CSS coding, you will need to take some time outside of the time-restricted lecture to look into the code to understanding it. Most of the time, it does not automatically resonate with every student the first or second time they see it. I know that you can do it and will earn a sense of accomplishment when you do! ;-)

Page Layout using CSS

```
wrapper
┌──────────────────────┐
│ header               │
├──────┬───────────────┤
│ nav  │ main          │
│      │               │
│      │               │
│      │               │
├──────┴───────────────┤
│ footer               │
└──────────────────────┘
```
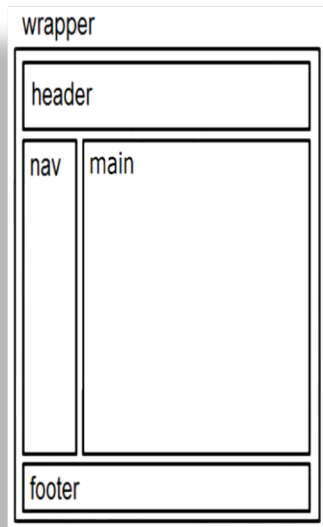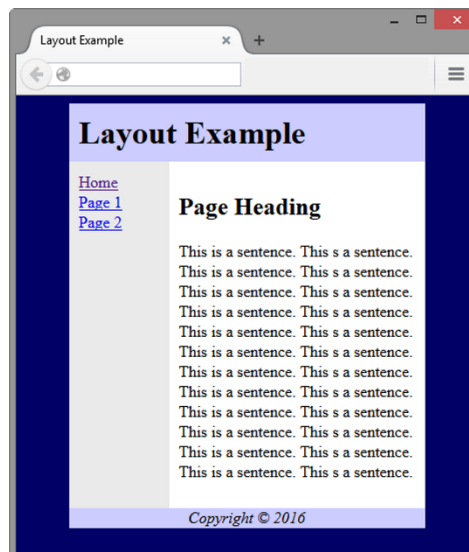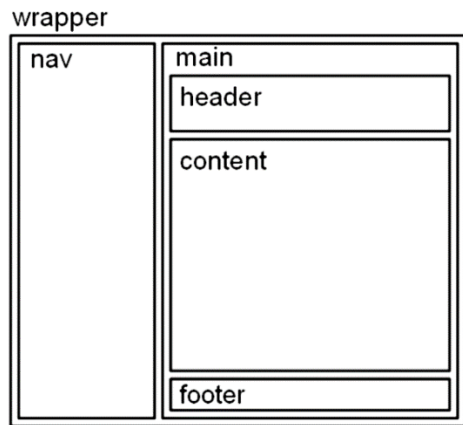
```css
#wrapper { width: 80%;
        margin-left: auto;
            margin-right: auto;
            background-color: #EAEAEA; }
header { background-color: #CCCCFF; }
h1 { margin: 0; padding: 10px; }
nav { float: left;
        width: 90px;
        padding: 10px; }
main { margin-left: 100px;
        padding: 10px;
        background-color:  #FFFFFF; }
footer { text-align: center;
            font-style: italic;
            background-color: #CCCCFF; }
```

```html
<body>
<div id="wrapper">
 <header> <header>
 <nav> </nav>
 <main> </main>
 <footer> </footer>
</div>
</body>
```

Layout Example

Home
Page 1
Page 2

Page Heading

This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.
This is a sentence. This s a sentence.

Copyright © 2016

HTML:
```html
<nav>
<ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="menu.html">Menu</a></li>
    <li><a href="directions.html">Directions</a></li>
    <li><a href="contact.html">Contact</a></li>
</ul>
</nav>
```

wrapper

nav

main

header

content

footer

Vertical Navigation
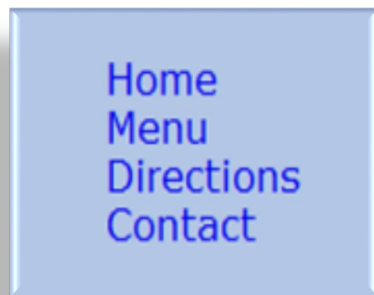

Home
Menu
Directions
Contact

CSS removes the list marker, removes the underline, adds padding, and configures the list items for inline display.

nav ul { list-style-type: none;}
nav a { text-decoration: none;
          padding-right: 10px; }
nav li { display: inline; }

Horizontal Navigation

Home   Menu   Directions   Contact

CSS Debugging Tips

Manually check syntax errors
Use W3C CSS Validator to check syntax errors
http://jigsaw.w3.org/css-validator/
Configure temporary background colors
Configure temporary borders
Use CSS comments to find the unexpected
/* the browser ignores this code */
Don't expect your pages to look exactly the same in all browsers!
Be patient!

CSS Assignment Instructions

All HTML files and CSS files should be created using a text editor such as Notepad or Notepad++ only. All HTML tags must be in lowercase and properly nested and closed. You should not use any graphical web design packages to develop your page.

Complete the following steps within each Section for your CSS Assignment.

A. Update your HTML coding to add more coding we have learned since the first assignment (within **Section A** instructions – 15 points)

B. Create (3) HTML webpages to link within this website (**Section B**) – 15 points).

C. Implement (3) CSS methods (inline, embedded, and external) in a manner which will not over-rule any of the other CSS coding involving various background colors, font colors, fonts, images, etc. (**Section C** instructions – 30 points)

D. Develop (2) more CSS external files for this website. (**Section D** – 30 points)

E. Answer the questions within a Word document of how to efficiently utilize your creations to manage this website (within **Section E** instructions – 10 points).

Section A – Updating with More HTML Features

Open your file from the previous Personal HTML Webpage Assignment within Notepad or Notepad++. Save the file as **index.html**.

Add the following HTML coding to your file. Remember to put quotes " " around the attribute values.

1. HTML5 DTD   **<!DOCTYPE html>**
2. Indicate English. **<html lang = "en">**
3. Within the title tags, add meta code of   **<meta charset= "utf-8">**
4. Add accessibility to your picture code **alt = "Image Description"** within the <img src code.
5. Add picture size to the image source code. For example**, height = "273" width "273"**
6. Add a block quote element **- <blockquote> …. Text   </blockquote>**
7. Add a contact email hyperlink code.
8. **me@gmail.com**
9. Add the following: **<em> , <mark>, <footer>, <small>**

10. Add a description list to display such things as skill set, definitions, etc. For example:
<dl>

    <dt> IP </dt>

        <dd> Internet Protocol </dd>

    <dt> TCP </dt>

        <dd> Transmission Control Protocol </dd>

</dl>

11. Add multiple pictures of your favorite foods or other types of images using the **div** to separate the images.
12. Add borders around the pictures.
13. Add at least one Horizontal Line somewhere in your webpage. **<hr>**
14. Add a thumbnail image.

<img src="small.jpg" alt= ski vacation" width="200" height="100">

15. Add a Meter Element or a Progress Element from Chapter 4. For example, **<progress value="5000" max="10000"> 5000</progress> Progress Towards Completion of My Goal**

Section B – Creating Other Web Pages

1. Gather the content you need to make at least (2) pages of information concerning you! For example: pictures, information, files, resume', etc.

2. Open, create, and develop (3) other .html files for your Personal HTML Web Page. One of the pages should be a "Contact Me" page.

3. Add hyperlinks to the new webpages and images within your index.html code. * If you copy/paste the below coding examples, you may need to delete and retype the "quotation marks".

   For example, **<a href="contact.html" Contact Me**

   <img src="images/home.gif" alt="Home" height="100" width="200">

4. Consider adding hyperlinks on all of the new HTML file to go back to index.html. However, this can be addressed later via the nav – navigate bar listed within CSS file. **<a href="index.html">Home>/a>**

5. Consider adding specific HTML elements to these pages also.

6. Save these files into the same folder as the index.html file

7. Save all of the corresponding images into an image folder (saved within parent folder).

Section C – Implementing (3) CSS Methods

1. Create an external style sheet (named **mystyles.css**), an embedded style sheet, and implement inline CSS to format your web page *based on the guidelines below*. Remember, you will need to link your mystyles.css file to your html files.

2. The page should utilize good contrast between the background color you choose and the text on the page. Feel free to use a color charts such as http://webdevfoundations.net/color or other color scheme resources such as:
   ◦ Monochromatic - http://meyerweb.com/eric/tools/color-blend
   ◦ Choose from a photograph or other image - http://www.colr.org
   ◦ Begin with a favorite color and a site below to choose other colors
      http://colorsontheweb.com/colorwizard.asp
      https://color.adobe.com/create/color-wheel
      http://paletton.com

3. Within your <u>external style sheet</u>, format the following:
   a. All H1 headings should be utilize the font family in the following order: "Verdana", "Helvetica", and sans-serif. H1 headings should be a dark color of your choosing. Choose an alignment like centered, right or left. Add a border to these headings
   b. All paragraphs should be centered. You should pick a font to use other than the standard font. The font size of paragraphs should be 16 pixels and paragraphs should be in a dark color font (you may pick the color).
   c. Borders around H1 headings.
   d. Background color and Background-image Property as described above in this document.
   e. Include a document header, nav, main (does not replace body tags), footer, etc.(review above code as an example – Chapter 6 Page Layout)

4. Within your <u>embedded style sheet</u> format the following:
   a. H2s utilize the font families in the following order, "Arial", "Helvetica Bold" and sans-serif. H2's color different should be different than H1.
   b. Place a 2-inch colored image border (ie. around the original image on the HTML assignment file)
   c. Create a division (div) for (3) images include colored backgrounds or colored borders. (ie. could do images of your favorite foods).
   d. Add example for a size for picture sizing –

```
<head>
 <style> html, body { height: 100%; }
 img.one { height: auto; width: auto; }
 img.two { height: 50%; width: 50%; }
 </style>
```

5.   Within the inline CSS include the following:
   a. The table on your page should have a colored 5px border (you may pick the color). Pick a dark background color for the row that is your table heading and use a light font color for your headings. Also, alternate the table rows using different shades of the same color. Pick a color, where there is still good contrast between the text and color. Feel free to use table padding if you wish. You can use the default font for your table text or you may choose a different font family.
   b. Have (1) H2 heading to be a especially different color than then ones formatted within the embedded CSS.
   c. Apply <strong> tag to particular text. Will override all other previous CSS format settings?

Section D – Preparing (2) More CSS Files

1. Create (2) new CSS files. Research and analyze which color combinations and fonts, etc. you wish to implement. We you get finished your should have three different color schemes. Options can include: monochromatic with HSLA Color – (Hue, Saturation, Light, Alpha), analogous, complimentary, split complementary, triadic, tetradic, colorful/balanced, natural/earth tones, cool/fresh, etc. Page 216.
2. Add a type of border code to one of the new CSS files as well.

You are free to add any other formatting that you wish. 😉

Section E – Efficiently Managing a Website FAQ

Consider hosting your Personal HTML Web Page or at least linking it onto your Blog.

Answer the below questions within your website.

1. How would you easily change the look of your website by using a different CSS file? Refer to your specific file names utilized within your assignment.

2. How does using CSS assist you and/or a team of programmers to efficiently manage a website and customer requests?

3. What are (5) benefits to implementing CSS? Which CSS method overwrites all other methods of CSS?

Advanced Page Layout - If you have time to spare before the due date, challenge yourself to use the Advanced Page Layout PowerPoint notes to add a **two-column layout with a navigation bar. (optional and not part of this assignment grade.)**

To submit this assignment do the following:

1. Validate your HTML and CSS files using the W3C Validation Service. Save the result in a file within the folder as your HTML and CSS files.
2. Create a folder for your homework called yourlastnameCSSAsmt. Create a subfolder named images for any images and a subfolder names styles for any external style sheets.
3. Create a zipped folder for yourlastnameHW3 and upload this folder into eCourses by the due date. To do this:
   o Go to File Explorer (yellow folder on taskbar)
   o Locate the folder labeled yourlastnameCSSAsmt.
   o Right-click it – select "Send To" – Compress
4. Upload the zipped folder containing your html files, style sheet file, images, and a copy of the validation reports for your CSS.