UNIT

02 | *DESIGNING WEBSITES*

CSS stands for Cascading Style Sheet. It is a styling sheet language used to describe the look and feel of a website. Without CSS language, the webpage would be plain and dull. Imagine searching on the internet and scrolling through multiple websites with no colors, alignments, and proper design.

CSS can control how the HTML elements should be displayed on a browser. Formatting the layout of the webpage, changing the font colors, size of a font, and modifying the width and height of an element is all the CSS job—the complete presentation of a webpage.

Mostly CSS is combined with the markup languages like HTML or XHTML. The best way to describe the connection between CSS and HTML is a "house." The house would be the HTML, and the HTML elements would be the doors, windows, and the roofs. However, the house and its parts are unpainted. Now the paint would be the CSS; it adds colors or style to the house. To summarize it all, HTML is used to describe a website's content, while CSS is used to display the content of a website in a presentable and attractive way. This unit will discuss the most widely used style sheet over the web, the Cascading Style Sheet (CSS), and how it styles different HTML elements.

THINK BEFORE YOU PROCEED

Name: _____ **Date:** _____
Course & Section: _____ **Score:** _____

Answer the questions based on the student's current design skills and knowledge about designing and website. Use the 4-point Likert Scale below.

1 – Not at all   2 – Maybe
3 – Yes   4 – So well, that I could help others

	1	2	3	4
1. Students can define what pixel is.				
2. Students can design and layout creatively.				
3. Students know the different types of font styles.				
4. Students can explore appealing color combinations.				
5. Students can use at least one photo editor application.				
6. Students understand what a responsive website is.				
7. Students can describe the different parts of a website.				

If the student answered "yes" or "so well" in questions number 5 and 6, answer the questions below.

1. Define ideas about responsive websites.

   _____
   _____
   _____.

2. What are the different parts of a website?

   _____
   _____
   _____.

3. What do you consider a well-designed website?

   _____
   _____
   _____.

LESSON 1:

WHAT IS CSS?

OBJECTIVES:

At the end of the lesson, students shall be able to:

- Define what CSS is;

- Describe the importance of CSS and how it is connected to HTML;

- Control the layout of the webpage using CSS Selectors; and

- Use multiple style sheets to make unique and attractive website.

DURATION: 2 hours

Name: _____   **Date:** _____
Course & Section: _____   **Score:** _____

Creating Digital Birthday Invitation

Create a webpage, notably a digital birthday invitation using HTML, and write about its design plan. For example, the background is blue, and the font color is black.

Why Learn CSS?

Cascading Style Sheet (CSS) is a design or a style language intended to use for creating presentable and attractive websites.

Understanding of HTML and CSS are the foundation of being a great Web Developer and Web Designer. Therefore, it is a must skill for every IT student and professional. Moreover, once they understand the basics of HTML and CSS, they can easily understand other related languages like JavaScript and PHP and their frameworks like the LaravelPHP, Bootstrap, and JQuery.

> **DID YOU KNOW?** In June 1999, CSS3, the current and latest version of CSS, became the recommendation of W3C, and nowadays, it is the widely used style language over the web.

Prerequisite

Before learning CSS, the learner should have a solid understanding of Hypertext Markup Language (HTML) which was covered in the first unit of this module.

The CSS Selectors

The CSS syntax is composed of a set of rules. These rules consist of three different parts: the selector, property, and value.

Selector: A selector represents an HTML tag that the style will be applied. It could be any tag like a paragraph tag `<p>` or a heading tag `<h1>`.

Property: A property specifies what style to add to an HTML element. They could be a color, alignment, or a border.

Value: Values are given to a property. For example, a color property can have a value of either white or black.

```
1 selector { property: value; }
```

Example (paragraph with a font color of red):

```
1 p { color: value; }
```

CSS selectors are used in selecting an HTML element to add style to it. There are several types of CSS selectors:

1. **Type Selector** – This type of selector is for **selecting a specific HTML tag**. For example, selecting an `<h1>` tag and changing its color to blue. Using

this type of selector means all of the `<h1>` tags from the document will be colored blue, even the newly added `<h1>` tags.

HTML Syntax:

```
1 <tagName> Content goes here... </tagName>
```

CSS Syntax:

```
1 tagName { property: value; }
```

Examples:

```
1 <h1> Hello Students! </h1>
```

```
1 h1 { color: blue; }
```

Output:

Hello Students!

Example: (Adding another `<h1>` tag into the HTML file without changing the styles above.)

```
1 <h1> Hello Students! </h1>
2 <h1> Have a Good day! </h1>
```

Output:

Hello Students!

Have a Good day!

2. **Universal Selector** – The Universal selector **matches all the HTML elements** that you added on your HTML file. May it be a paragraph, button, or a heading tag. All styles will be applied in each HTML element.

HTML Syntax:

```
1 <tagName> Content goes here... </tagName>
```

CSS Syntax:

```
1 * { property: value; }
```

Examples:

```
1 <h1> Hello Students! </h1>
2 <h1> Have a Good day! </h1>
3
4 <p> Hello Students! </p>
```

```
5 <button> Have a Good day! </button>
```

```
1 * { color: red; }
```

Output:

Hello Students!

Have a Good day!

Hope you are all doing good today.

Click me!

3. **Class Selector** – The class selector selects all HTML elements depending on the particular **class attribute**. Class selectors are formed by a period and selector name.

HTML Syntax:
```
1 <tagName class="selectorName" > Content... </tagName>
```

CSS Syntax:
```
1 .selectorName { property: value; }
```

Examples:
```
1 <h1 class="heading-center"> Hello Students! </h1>
2 <p> Have a Good day! </p>
3 <p class="heading-center"> Hope you are all doing
4 good today. </p>
```

```
1 .heading-center {
2     color: red;
3     font-style: italic;
4     text-align: center;
5 }
```

Output:

Hello Students!

Have a Good day!

Hope you are all doing good today.

4. **ID Selector** – The id selector selects one **unique HTML element** depending on the **id attribute**. The id should **only be used once**, unlike the class that can be applied to multiple HTML elements. Id selectors are formed by a hash sign and selector name.

HTML Syntax:

```
1 <tagName id="selectorName" > Content... </tagName>
```

CSS Syntax:

```
1 #selectorName { property: value; }
```

Examples:

```
1 <h1 id="heading-center"> Hello Students! </h1>
```

```
1 .heading-center {
2     color: gray;
3     font-family: Arial, Helvetica, sans-serif;
4     text-align: center;
5 }
```

Output:

Hello Students!

5. **Grouping Selectors** – These are used to apply styles to **multiple CSS selectors**. Separate each selector with a comma.

HTML Syntax:

```
1 <tagName id="selectorName" > Content... </tagName>
2 <tagName class="selectorName" > Content... </tagName>
3 <tagName> Content... </tagName>
```

CSS Syntax:

```
1 #selectorName, .selectorName, tagName {
2     property: value;
3 }
```

Example:

```
1 <h1 id="heading-gray"> Hello Students! </h1>
2 <button> Click Me! </button>
3 <p class="parag-gray"> Web 1 </p>
```

```
1 #heading-gray, .parag-gray, button {
2     color: gray;
3     font-family: Arial, Helvetica, sans-serif;
4 }
```

Output:

Hello Students!

Click Me!

Web 1

The CSS Selectors

There are different ways to add CSS styles to HTML documents.

1. **Inline CSS** – Inline CSS uses style attribute to add an **inline style** to a particular HTML element.

```
1  <tagName style="property: value" > Content... </tagName>
```

Example:

```
1  <h1 style="color: purple"> Hello Students! </h1>
```

2. **Internal CSS** – Internal CSS can be found at the head section of a webpage inside the `<style>` tag. It is used to add style for a single HTML page.

```
1  <head>
2     <style>
3         .selectorName { property: value; }
4     </style>
5  </head>
```

Example:

```
1  <style>
2      h1 { color: purple }
3  </style>
```

3. **External CSS** – External CSS is a separate file with the extension of *.css. The external style sheet file referenced inside the `<link>` tag is found at the head section of the webpage.

```
1  <head>
2     <link rel="stylesheet" href=="locationOfCSSFile.css">
3  </head>
```

Example:

```
1   <link rel="stylesheet" href="styles.css" >
```

Output:

Hello Students!

| TIPS & TRICKS | Use an external style sheet to style different webpages using one or a few CSS files. It is efficient and saves time, especially for large web development projects. |

YOU CAN ANSWER THESE!

Name: _____   **Date:** _____
Course & Section: _____   **Score:** _____

1. What is CSS?
_____
_____
_____.

2. Describe the difference between HTML and CSS?
_____
_____
_____.

3. What is the definition of selector, property, and value?
_____
_____
_____.

4. What is the use of a Universal Selector?
_____
_____
_____.

5. If you are going to create multiple webpages, what is the most efficient way of inserting CSS styles?
_____
_____
_____.

6. Which HTML tag is used to insert an external style sheet?
_____
_____
_____.

7. HTML tag that is used to specify an internal style sheet.
_____
_____
_____.

TRY THIS ON YOUR OWN

78

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

Create different HTML elements and apply a font color of your choice using different types of CSS selectors. Take note to use all of the CSS selectors discussed previously.

LESSON 2:

CSS BACKGROUND, TEXT, FONT, AND IMAGE PROPERTIES

OBJECTIVES:

At the end of the lesson, students shall be able to:

- Use different CSS Background properties;

- Weigh different font styles using CSS Text and Font properties; and

- Design an image using CSS Image properties.

DURATION: 3 hours

ASSESS YOURSELF

Name: _____   **Date:** _____
Course & Section: _____   **Score:** _____

From the last activity (**Unit 2 Lesson 1 – Assess Yourself**), add the different types of CSS selectors into the HTML file. Make sure to use all the CSS selectors you have learned from the previous lesson.

CSS Background Properties

The CSS background properties are used to control the background of an HTML element. A single HTML element can have multiple background properties, and these are the following:

Background Properties
background-color
background-image
background-repeat
background-attachment
background-position
background

1. **The Background Color** – The background-color property is used to set the background color of an HTML element.

```
1  selectorName {
2      background-color: value;
3  }
```

Example:

```
1  <body>
2      <h1> Background Color is pink. </h1>
3  </body>
```

```
1  body {
2      background-color: pink;
3  }
```

Output:

Background Color is pink.

2. **The background-image Property** – The background-image property is used to insert a background image into an HTML element.

```
1  selectorName {
2      background-image: url("path.extension");
3  }
```

Example:

```
1  <body>
2      <h1> Background Image is Set. </h1>
3  </body>
```

```
1  body {
2      background-image:  url("bg-image.jpg");
3  }
```

Output:

3. **The** `background-repeat: repeat` **Property** – Use the **background-repeat** property and **repeat** value if the background image is too small.

```
1  selectorName {
2      background-image: url("path.extension");
3      background-repeat: repeat;
4  }
```

Example:
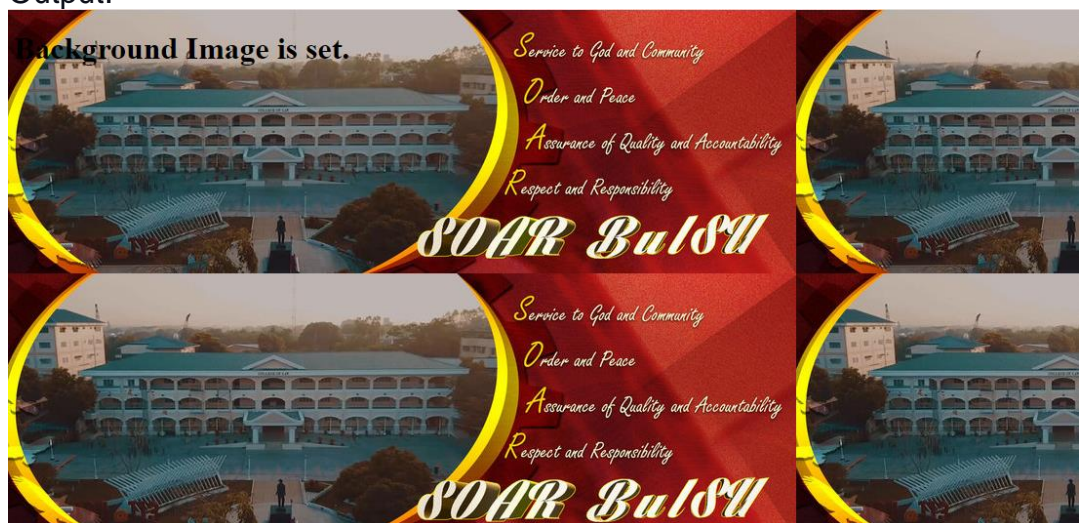
```
1  <body>
2      <h1> Background Image is Set. </h1>
3  </body>
```

```
1  body {
2      background-image:  url("bg-image.jpg");
3      background-repeat:  repeat;
4  }
```

Output:

To repeat the background vertically, use **background-repeat** property and **repeat-y** value, and for horizontal background, use **background-repeat** property and **repeat-x**.

```
1  selectorName {
```

```
2        background-image: url("path.extension");
3        background-repeat: repeat-y;
4 }
```

Example:

```
1 <body>
2        <h1> Background Image is Set. </h1>
3 </body>
```

Change the **background-repeat** value to **repeat-x** for the horizontal background image.

```
1 body {
2        background-image:  url("bg-image.jpg");
3        background-repeat:  repeat-y;
4 }
```

Output: (vertical)

(horizontal)

And to display the background image once, use **background-repeat** property and **no-repeat** value.

```
1 selectorName {
2        background-image: url("path.extension");
```

```
3     background-repeat: no-repeat;
4 }
```

Example:

```
1 <body>
2     <h1> Background Image is Set. </h1>
3 </body>
```

```
1 body {
2     background-image:  url("bg-image.jpg");
3     background-repeat:  no-repeat;
4 }
```

Output:

4. **The** `background-position` **Property** – Use the **background-position** property to determine the position of the background image.

```
1 selectorName {
2     background-image: url("path.extension");
3     background-repeat: no-repeat;
4     background-position: right-top;
5 }
```

Example:

```
1 <body>
2     <h1> Background Image is Set. </h1>
3 </body>
```

```
1 body {
2     background-image:  url("bg-image.jpg");
3     background-repeat:  no-repeat;
4     background-position:  right-top;
5 }
```

Output:

5. **The** `background-attachment` **Property** – The **background-attachment** property is used to specify if the background image is fixed or scrolls within the webpage.

```
1 selectorName {
2     background-image: url("path.extension");
3     background-repeat: no-repeat;
4     background-attachment: fixed;
5 }
```

Example: (fixed background)

```
1 <body>
2     <h1> Background Image is Fixed. </h1>
3     <p> Scroll the page. </p>
4     <p> Scroll the page. </p>
5     <p> Scroll the page. </p>
6     <p> Scroll the page. </p>
7     <p> Scroll the page. </p>
8     <p> Scroll the page. </p>
9     ...
  </body>
```

```
1 body {
2     background-image:  url("bg-image.jpg");
3     background-repeat:  no-repeat;
4     background-attachment:  fixed;
5 }
```

Output:

Scroll the page.

Scroll the page.

Scroll the page.

Scroll the page.

Scroll the page.

Scroll the page.

Example (scroll background):

```
 1  <body>
 2      <h1> Background Image is Fixed. </h1>
 3      <p> Scroll the page. </p>
 4      <p> Scroll the page. </p>
 5      <p> Scroll the page. </p>
 6      <p> Scroll the page. </p>
 7      <p> Scroll the page. </p>
 8      <p> Scroll the page. </p>
 9      ...
10  </body>
```

```
 1  body {
 2      background-image:  url("bg-image.jpg");
 3      background-repeat:  no-repeat;
 4      background-attachment:  scroll;
 5  }
```

Output:

```
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
Scroll the page.
```

6. **The CSS Background Shorthand** – A shorthand property defines all background properties in one property to have lesser code.

Originally you can write,

```
 1  selectorName {
 2      background-color: pink;
 3      background-image: url("path.extension");
 4      background-repeat: no-repeat;
 5      background-position: top;
 }
```

Or use shorthand background property.

```
 1  selectorName {
 2      background: pink url("path.extension") no-repeat top;
 3  }
```

TIPS & TRICKS

Adjust the text color or choose a background image that will go well with the text to ensure that the text is readable on top of the background image.

The Text and Font Properties

The CSS text and font properties allow the formatting of HTML text and font elements. These may customize the font color, the font style, alignment, and many more. Below is the list of various text and font properties:

Text Properties	Font Properties
color	font-family
direction	font-style
letter-spacing	font-variant
word-spacing	font-weight
text-indent	font-size
text-align	font
text-decoration	
text-transform	
white-space	
text-shadow	

CSS Text Properties

1. **The color Property** – The **color** property specifies the color of a text.

```
1  selectorName {
2      color: value;
3  }
```

Example:

```
1  <h1> Good day! </h1>
2  <p> Hello everyone. </p>
```

```
1  h1 {
2      color: green;
3  }
4
5  p {
6      color: brown;
7  }
```

Output:

Good day!

Hello everyone.

TIPS & TRICKS Make use of the different color values to have more options of colors to choose from. Ex. HEX colors: #ff0000, RGB colors: rgb(255, 0, 0, 0) and pre-defined colors: red, brown, yellow, and more.

2. **The** `direction` **Property** – The **direction** property is used to change the direction of a text. Right to the left (rtl) and Left to the right (ltr) are the two possible values.

```
1  selectorName {
2      direction: value;
3  }
```

Example:

```
1  <h1> Good day! </h1>
2  <p> Hello everyone. </p>
```

```
1  h1 {
2      color: green;
3      direction: ltr;
4  }
5
6  p {
7      color: brown;
8      direction: rtl;
9  }
```

Output:

Good day!

.Hello everyone

3. **The** `letter-spacing` **Property** – The **letter-spacing** property allows us to have specified space between characters. Use **normal** value or number value to set the space.

```
1  selectorName {
2      letter-spacing: value;
3  }
```

Example:

```
1  <h1> Good day! </h1>
2  <p> Hello everyone. </p>
```

```
1  h1 {
2      letter-spacing: normal;
3  }
4
5  p {
6      letter-spacing: 5px;
7  }
```

Output:

Good day!

Hello everyone.

4. **The** `text-indent` **Property** – The **text-indent** property specifies the indention of a text. It is mostly used for paragraph tags `<p>`.

```
1  selectorName {
2      text-indent: value;
3  }
```

Example:

```
1  <p> This is an example paragraph that will have
2          a 5cm indention at the first line. </p>
```

```
1  p {
2      text-indent: 5cm;
3  }
```

Output:

This is an example paragraph that will have a 5cm indention at the first line.

5. **The** `text-align` **Property** – The **text-align** property is used to change the alignment of a text. Example values are left, right, center, and justify.

```
1  selectorName {
2      text-align: value;
3  }
```

Example:

```
1  <p> This is an example paragraph for the
2          demonstration of text alignment. </p>
```

```
1  p {
2      text-align: right;
3  }
```

Output:

This is an example paragraph for the demonstration of text alignment.

6. **The** `text-decoration` **Property** – The **text-decoration** property changes the decoration of a text. Example values are none, underline, overline, line-through, and blink.

```
1 selectorName {
2     text-decoration: value;
3 }
```

Example:

```
1 <p> This is an example paragraph. </p>
```

```
1 p {
2     text-decoration: line-through;
3 }
```

Output:

~~This is an example paragraph.~~

7. **The** `text-transform` **Property** – The **text-transform** property specifies the text case of a text. Example values are none, capitalize, uppercase, and lowercase.

```
1 selectorName {
2     text-transform: value;
3 }
```

Example:

```
1 <p> This paragraph will be in uppercase. </p>
2 <h1> This heading will be in uppercase. </h1>
```

```
1 p, h1{
2     color: #7a0004;
3     text-transform: uppercase;
4 }
```

Output:

THIS PARAGRAPH WILL BE IN UPPERCASE.

THIS PARAGRAPH WILL BE IN UPPERCASE.

8. **The** `white-space` **Property** – The **white-space** property allows the text to have white spaces. Example values are normal, pre, and nowrap.

```
1 selectorName {
2     white-space: value;
3 }
```

Example:

```
1  <p> This paragraph will be in uppercase. </p>
2  <h1>        This heading has whitespaces. </h1>
```

```
1  h1{
2      white-space: pre;
3  }
```

Output:

This paragraph will be in uppercase.

This heading has whitespaces.

9. **The** `text-shadow` **Property** – The **text-shadow** property sets the shadow of a text. Note that not all browsers support this property.

```
1  selectorName {
2      text-shadow: h-shadow v-shadow blur-radius color;
3  }
```

Example:

```
1  <h1> Heading with a shadow. </h1>
```

```
1  h1{
2      color: pink;
3      text-shadow: 2x 2x 0 black;
   }
```

Output:

Heading with a shadow.

CSS Font Properties

1. **The** `font-family` **Property** – The **font-family** property specifies the font face of a text. There are two kinds of font-family names. First is the generic family, which is a group of fonts with a similar class. Typical generic font families include **serif**, **sans-serif**, and **monospace**. Second is a specific font family such as Times New Roman, Arial, and a lot more.

```
1  selectorName {
2      font-family: value;
3  }
```

Example:

```
1  <h1> Web Systems and Technologies</h1>
```

```
1  h1 {
2      color: #131033;
3      font-family: #Arial, Helvetica, sans-serif;
4  }
```

Output:

Web Systems and Technologies

TIPS & TRICKS   In some cases, the browser might not support your chosen font family. Therefore, it is advised that the font-family property should have several font names.

2. **The font-style Property** – The **font-style** property defines the font style of a text. This property is commonly used to italicize a text. Example values are normal, italic, and oblique.

```
1  selectorName {
2      font-style: value;
3  }
```

Example:

```
1  <h1> Web Systems and Technologies. </h1>
```

```
1  h1 {
2      color: #131033;
3      font-family: #Arial, Helvetica, sans-serif;
4      font-style: italic;
5  }
```

Output:

Web Systems and Technologies.

3. **The font-variant Property** – The **font-variant** property is used to set the font-variant of a text. Example values are normal and small-caps.

```
1  selectorName {
2      font-variant: value;
3  }
```

Example:

```
1  <h1> Web Systems and Technologies. </h1>
```

```
1  h1 {
2      color: #131033;
3      font-family: #Arial, Helvetica, sans-serif;
4      font-variant: small-caps;
5  }
```

Output:

WEB SYSTEMS AND TECHNOLOGIES.

4. **The** font-weight **Property** – The **font-variant** controls how bold a text is. Example values are normal, bold, bolder, lighter, 100, 200, and 300.

```
1  selectorName {
2      font-weight: value;
3  }
```

Example:

```
1  <p style="font-weight: normal"> This text is normal. </p>
2  <p style="font-weight: bold"> This text is bold. </p>
3  <p style="font-weight: bolder"> This text is bolder. </p>
```

Output:

This font is normal.

This text is bold.

This text is bolder.

5. **The** font-size **Property** – The **font-variant** property defines the size of a text. Example values are xx-small, x-small, small, medium, large, xx-large, in pixels, and percentage.

```
1  selectorName {
2      font-size: value;
3  }
```

Example:

```
1  <p style="font-size: 30px"> This font size is 10px. </p>
2  <p style="font-size: large"> This font size is large. </p>
3  <p style="font-size: 50%"> This font size is 50%. </p>
```

Output:

This font size is 10px.

This font size is large.

This font size is 50%.

7. **The CSS Font Shorthand** – A shorthand property defines all font properties in one property to have lesser code.

```
1  selectorName {
2      font: value;
3  }
```

Example:

```
1  <h1> This is a text with CSS font shorthand property. </h1>
```

```
1  h1 {
2      font: italic bold 50px Georgia;
3  }
```

Output:

This is a text with CSS font shorthand property.

CSS Image Properties

      People love to visualize rather than reading alone. Images help to give a deep understanding of a story or content. Hence, pictures on websites deliver an immersive experience to the visitors. CSS provides different image properties to customize and control the appearance of the images.

Image Properties
border
height
width

1. **The** border **Property** – The **border** property is used to define the border width of an image. Specify values in length or percentage.

```
1  selectorName {
2      border: length line-style color;
3  }
```

Example:

```
1  <img src="bulsu.png">
```

```
1  img {
2      border: 5px solid gray;
3  }
```

Output:

2. **The** `height` **Property** – The **height property** is used to change the height of an image. Specify values in length or percentage.

```
1  selectorName {
2      height: length-value;
3  }
```

Example:

```
1  <img src="bulsu.png">
```

```
1  img {
2      border: 5px solid gray;
3      height: 50px;
4  }
```

Output:

3. **The** `width` **Property** – The **width property** is used to change the width of an image. Specify values in length or percentage.

```
1  selectorName {
2      width: length-value;
3  }
```

Example:

```
1  <img src="bulsu.png">
```

```css
1  img {
2      border: 5px solid gray;
3      height: 100px;
4      width: 300px;
5  }
```

Output:

YOU CAN ANSWER THESE!

Name: _____   **Date:** _____
Course & Section: _____   **Score:** _____

1. What is the use of CSS shorthand property?
_____
_____.

2. Explain why is it recommended to set several font families into a one text element?
_____
_____.

3. Differentiate the background fixed and a scroll background image.
_____
_____.

4. What is the use of a Universal Selector?
_____
_____.

5. If you are going to create multiple webpages, what is the most efficient way of inserting CSS styles? Why?
_____
_____
_____.

Name: _____ **Date:** _____

Course & Section: _____ **Score:** _____

Creating a Digital Resume
Create a digital resume using the different background, text, font, and image properties of CSS.

Sample Output:
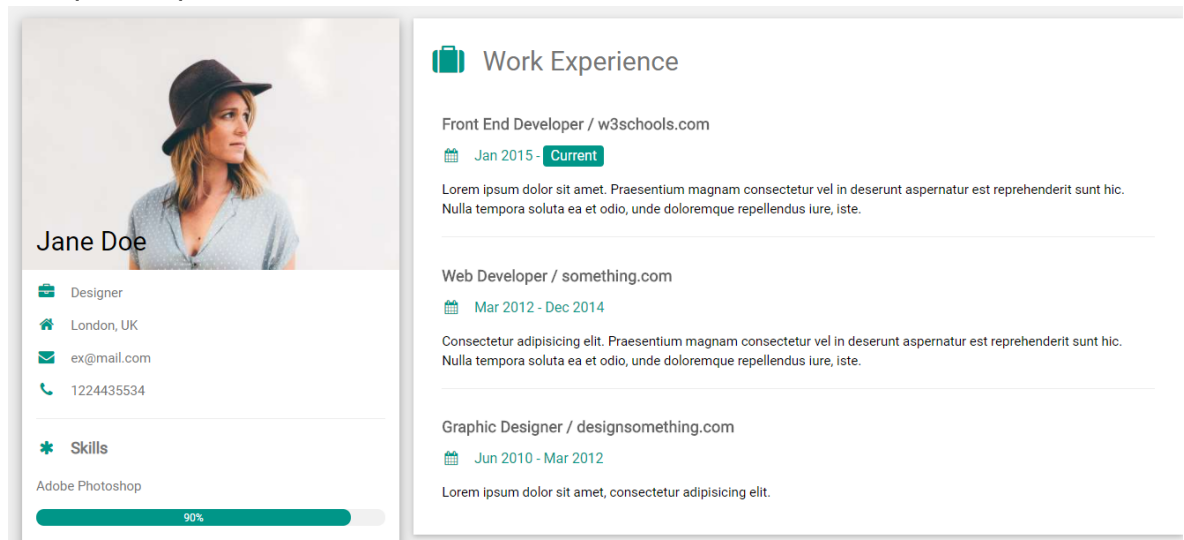
https://www.w3schools.com/w3css/tryw3css_templates_cv.htm

LESSON 3:

CSS POSITION AND DISPLAY PROPERTIES

OBJECTIVES:

At the end of the lesson, students shall be able to:

- Organize the layout of a webpage using position property;

- Learn the difference between block and inline HTML elements; and

- Hide an HTML element using display property.

DURATION: 3 hours

ASSESS YOURSELF

Name: _____ **Date:** _____
Course & Section: _____ **Score:** _____

Digital Birthday Invitation: Update
Improve and enhance the appearance of your digital birthday invitation by adding different types of CSS properties (background, text, font, and image properties) learned from the previous lesson (Lesson 2).

CSS Layout Properties

Designing and customizing a website is not limited to changing backgrounds, styling fonts, and adding images. When developing a website, one of the crucial things to learn is to layout the webpage which means, to decide the placement of the HTML elements.

In CSS, there are several different approaches to layout a website using other properties.

Position Properties	Display Properties
relative	block
absolute	inline
fixed	none

CSS Position Property

The position property specifies which type of position is used for an HTML element.

1. **The** `position: static` **Property** – The static position is the default position of an HTML element. Static positioning places the HTML elements according to the normal flow of the webpage.

```
1  selectorName {
2      position: static;
3  }
```

Examples:

```
1  <h2> The position:static; property </h2>
2  <p> Static position property. </p>
3  <div class="static"> > This div element has static
4      position property. </div>
```

```
1  .static { position: static; }
```

Output:

The position: static; property

Static position property.

This div element has static position property.

2. **The** `position: relative` **Property** – Relative positioning places the HTML elements close to where it usually appears. Additionally, it allows us to

change the place of an HTML element by defining the top, right, button, and left an HTML element. Take note: changing the position of an element using a relative position does not disturb the other elements.

```
1 selectorName {
2     position: relative;
3 }
```

Examples:

```
1 <h2> The position:relative; property </h2>
2 <p> Relative position property. </p>
3 <div class="relative"> > This div element has relative
4     position property. </div>
```

```
1 .static {
2     position: relative;
3     left: 30px;
4 }
```

Output:

The position: relative; property

Relative position property.

This div element has relative position property.

3. **The `position: absolute` Property** – The absolute position removes the HTML element from the document flow while other elements will render as if it does not exist. It also enables the placement of an HTML element using the top, right, button, and left. However, changing the position of an element using the absolute position initially disturb the position of other elements.

```
1 selectorName {
2     position: absolute;
3 }
```

Examples:

```
1 <p class="absolute"> This paragraph has
2     absolute position. </p>
3 <h2> The position:absolute; property </p>
```

```
1 .absolute {
2     position: absolute;
3     left: 30px;
4 }
```

Output:

The position: absolute; property
This paragraph has absolute position.

4. **The** `position: fixed` **Property** – The fixed position moves the HTML elements as it scrolls.

```
1 selectorName {
2     position: fixed;
3 }
```

Examples:

```
1  <h2> The position:fixed; property </h2>
2  <p> Fixed position property. </p>
3  <div class="fixed"> This div element has fixed
4      position property. </div>
5  <p> Scroll the page. </p>
6  <p> Scroll the page. </p>
7  <p> Scroll the page. </p>
8  <p> Scroll the page. </p>
9  <p> Scroll the page. </p>
10 <p> Scroll the page. </p>
11 ...
```

```
1  .fixed { position: fixed; }
```

Output:

The position: relative; property

Relative position property.

This div element has relative position property.

CSS Display Property

The `display` **Property** – The display property determines how HTML elements are going to be placed on a webpage. The default HTML display value is block or inline.

Block level elements are the any HTML elements that uses the full width of a page. A block level element occupies the entire space and always begin on a new line. These are some examples of block level elements:

- `<p>`
- `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` and `<h6>`
- `<form>`
- `<header>`, `<footer>` and `<section>`

The **inline level elements** are the any HTML elements that do not force a new line and only occupy space as needed. These are some examples of inline level elements:

- `<span>`
- `<img>`
- `<a>`
- `<input>`
- `<button>`

As stated above, every HTML element has a default display value that is always changeable.

```
1  selectorName {
2      display: display-value;
3  }
```

Examples:

```
1  <span> Exchanging the default display value. </span>
2  <p> Span and paragraph tag. </p>
3  <p> Another paragraph tag. </p>
```

```
1  span {
2      display: block;
3  }
4
5  p {
6      display: inline;
7  }
```

Output:

Exchanging the default display value.
Span and paragraph tag. Another paragraph tag.

Another use of display property is hiding an HTML element using the **none** value.

```
1  selectorName {
2      display: none;
3  }
```

Examples:

```
1  <span> None display value. </span>
2  <p> This is the only HTML element that will show. </p>
```

```
1  span {
2      display: none;
```

```
3  }
4
5  p {
6      display: inline;
7  }
```

Output:

This is the only HTML element that will show.

YOU CAN ANSWER THESE!

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

1. What is the difference between position and display properties?
_____
_____
_____
_____
_____.

TRY THIS ON YOUR OWN

Name: _____   **Date:** _____
Course & Section: _____   **Score:** _____

Digital Resume: Update
Design your digital resume using position and display properties as discussed in this lesson.

LESSON 4:

CSS BOX MODEL

OBJECTIVES:

At the end of the lesson, students shall be able to:

- Describe the CSS Box Model;

- Classify the difference between padding, border, and margin;

- Compare the effects of padding, border, and margin; and

- Design elements following the rules of CSS box model.

DURATION: 2 hours

ASSESS YOURSELF

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

Digital Birthday Invitation: Update

Improve and enhance the appearance of your digital birthday invitation by adding different types of CSS position and display properties you have learned from the previous lesson.

What is the Box Model?

Everything in CSS is a box. Whether it is a button, image, or text, all HTML elements have a box model. Refer to the image below for the illustration of a box model. The box model consists of content, margin, padding, and a border.
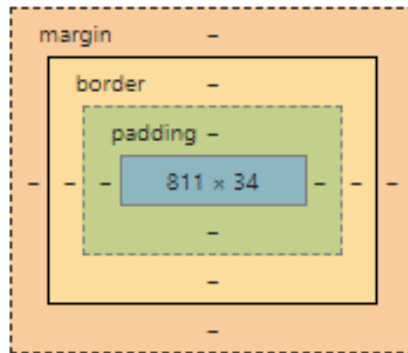
The blue color represents the content of the box or the content of an element, e.g., a paragraph tag with "Hello Students!" content.

```
<p> Hello Student! </p>.
```

The Padding, Border, and Margin Properties

The `padding` **Property** – The padding property set how much space should appear inside the border and around the content of an HTML element.
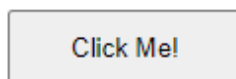
```
1  selectorName {
2      padding-top: value;
3      padding-right: value;
4      padding-bottom: value;
5      padding-left: value;
6      padding: top-value right-value bottom-value left-value;
7  }
```

Examples:

```
1  <button> Click Me! </button>
```

```
1  button { padding: 10px 30px 10px 30px; }
```

Output:
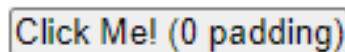
The `border` **Property** – The border property specifies the outer edge of an HTML element. Modify the border of an element using its three properties:

1. The **border-color** sets the color of the border.
2. The **border-style** sets the style of the border. Example values are solid, dashed, double, and dotted.

3. The **border-width** sets width of a border.

```
1  selectorName {
2      border-color: red;
3      border-style: dashed;
4      border-width: 10px;
5  }
```

Examples:

```
1  <button> Click Me! </button>
```

```
1  button {
2      border-color: red;
3      border-style: dashed;
4      border-width: 10px;
5  }
```

Output:

The `margin` **Property** – The margin property sets how much space should appear outside the border. Margin is usually used to specify space between the elements.
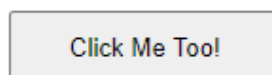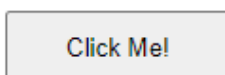
```
1  selectorName {
2      margin-top: value;
3      margin-right: value;
4      margin-bottom: value;
5      margin-left: value;
6      margin: top-value right-value bottom-value left-value;
7  }
```
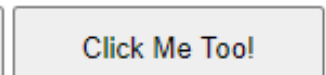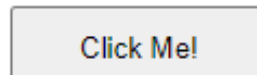
Examples:

```
1  <button> Click Me! </button>
2  <button> Click Me Too! </button>
```

```
1  button {
       padding: 10px 30px 10px 30px;
       margin: 10px 10px 10px 10px;
   }
```

Output (with margin):   Output (without margin):

YOU CAN ANSWER THESE!

Name: _____ **Date:** _____
Course & Section: _____ **Score:** _____

1. What is the difference between a padding and a margin?
_____
_____
_____.

2. What is the use of CSS Box Model?
_____
_____
_____.

3. What will happen if you set a padding: 0 to an HTML element?
_____
_____
_____.

4. What will happen if you set a margin: 0 to an HTML element?
_____
_____
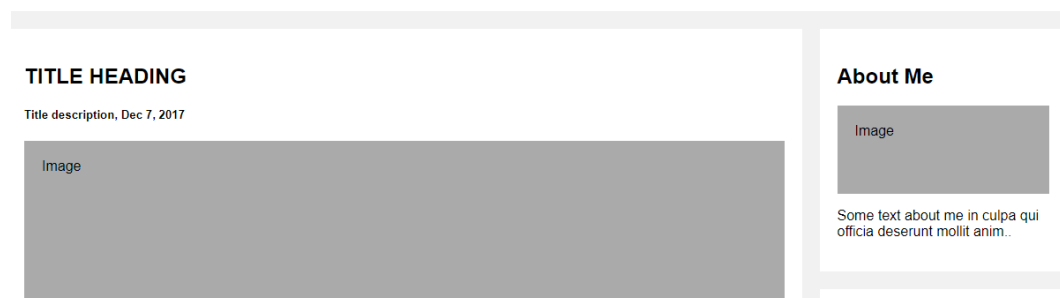_____.

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

Make your own attractive personal blog webpage using HTML elements and CSS properties. Make sure to use as many HTML elements as possible and make them visually appealing using different CSS properties.

Sample Output:

Blog Name

https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_blog_layout

EVALUATE WHAT YOU LEARNED

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

I. MULTIPLE CHOICE. Answer the following by encircling the letter of the best answer that fits the given question.

1. Which of the following CSS selectors chooses one unique HTML element?
      a. Universal Selector   c. Class Selector
      b. ID Selector   d. All of the above

2. CSS stands for _____.
      a. Cascading Styles Sheets   c. Cascading Style Sheet
      b. Creative Styles Sheet   d. Cascading Styling Sheet

3. Which is a correct CSS syntax?
      a. h1 { color: white; }   c. h1 black: color; }
      b. h1 { color= black; }   d. h1 color: white

4. Which property is used to make an italic font?
      a. font-family   c. font-weight
      b. font-italic   d. font-style

5. Which property is used to set shadow over the text?
      a. text-shadow   c. font-style
      b. text-decoration   d. color

6. How do you add a font color to all the buttons in an HTML document?
      a. button { color: red; }   c. .button { color: red; }
      b. all-button { color: red; }   d. #button { color: red; }

7. Which property is used to control the font size of an element?
      a. font-style   c. font-size
      b. text-size   d. text-style

8. What property is used to set space outside of an element?
      a. padding   c. border
      b. margin   d. CSS Box Model

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

9. How do you select HTML element with a class of "hello"?
    a. hello   c. .hello
    b. class:hello   d. #hello

10. Which property is used to define the border width of an element?
    a. bg   c. background
    b. border   d. width

11. Which is the default value or property position?
    a. absolute   c. static
    b. relative   d. none

12. To group HTML selectors, separate each selector with _____.
    a. space   c. plus size
    b. equal sign   d. comma

13. Which property is used to define the background image of an element?
    a. background   c. background-attachment
    b. background-image   d. background-repeat

14. "rtl" and "ltf" is the value of what property?
    a. font-family   c. position
    b. font-italic   d. direction

15. How would you reference your CSS across multiple HTML documents?
    a. Inline Style   c. External Style Sheet
    b. Internal Style Sheet   d. It is not possible

16. How can you add space between the border and the content of the element?
    a. padding   c. border
    b. margin   d. CSS Box Model

17. p { padding: 5px 10px 20px 15px; }. 20px is padding _____?
    a. top   c. right
    b. left   d. bottom

Name: _____   **Date:** _____

Course & Section: _____   **Score:** _____

18. It is a group of fonts with a similar class.
- a. generic font family
- c. Sans Serif
- b. font-family
- d. font

19. Setting the position of an element using _____ does not disturb the other elements.
- a. static
- c. relative
- b. absolute
- d. none

20. _____ is used to add an external style sheet in HTML document.
- a. <link rel="stylesheet" type="text/css" href="*.css">
- b. <styles rel="stylesheet" type="text/css" href="*.css">
- c. <link-rel="stylesheet" type="text/css" href="*.css">
- d. None of the above