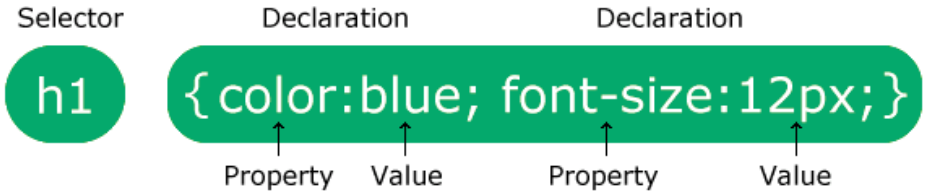
Compiled By:
Ranjan Raj Aryal
Additional Associate Professor
Cosmos College of Management and Technology

Unit 3- CSS(Cascading Style Sheets)

- CSS is the language we use to style an HTML document.
- It describes how HTML elements should be displayed.
- It can control the layout of multiple web pages all at once.
- CSS saves a lot of work.
- External stylesheets are stored in CSS files

CSS Syntax



- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a CSS property name and a value, separated by a colon.
- Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces

Example

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

Here

- `p` is a selector in CSS (it points to the HTML element you want to style: <p>).

- `color` is a property, and `red` is the property value
- `text-align` is a property, and `center` is the property value

Inserting CSS

There are three ways of inserting a style sheet:

- External CSS
- Internal CSS
- Inline CSS

External CSS

With an external style sheet, you can change the look of an entire website by changing just one file. Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.

Example:

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>

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

</body>
</html>
```

Here "**mystyle.css"** may look :

```
body {
        background-color: lightblue;
}

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

Internal CSS

- An internal style sheet may be used if one single HTML page has a unique style.
- The internal style is defined inside the <style> element, inside the head section

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}

h1 {
```

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

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

</body>
</html>
```

Inline CSS

- An inline style may be used to apply a unique style for a single element.
- To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;text-align:center;">This is a heading</h1>
<p style="color:red;">This is a paragraph.</p>

</body>
</html>
```

CSS Selectors

CSS selectors are used to "find" (or select) the HTML elements you want to style.

We can divide CSS selectors into five categories:

- **Simple selectors** (select elements based on name, id, class)
- **Combinator selectors** (select elements based on a specific relationship between them)
- **Pseudo-class selectors** (select elements based on a certain state)
- **Pseudo-elements selectors** (select and style a part of an element)
- **Attribute selectors** (select elements based on an attribute or attribute value)

The CSS element Selector

The element selector selects HTML elements based on the element name.

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

CSS Combinators

- A combinator is something that explains the relationship between the selectors.

A CSS selector can contain more than one simple selector. Between the simple selectors, we can include a combinator.

There are four different combinators in CSS:

- descendant selector (space)
- child selector (>)
- adjacent sibling selector (+)
- general sibling selector (~)

Descendant Selector

The descendant selector matches all elements that are descendants of a specified element.

The following example selects all <p> elements inside elements:

```
div p {
  background-color: red;
}
```

Example:

```
<html>
<head>
<style>
div p {
  background-color: yellow;
}
</style>
</head>
<body>

<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <section><p>Paragraph 3 in the div.</p></section>
```

```
</div>

<p>Paragraph 4. Not in a div.</p>
<p>Paragraph 5. Not in a div.</p>

</body>
</html>
```

Child Selector

The child selector selects all elements that are the children of a specified element.

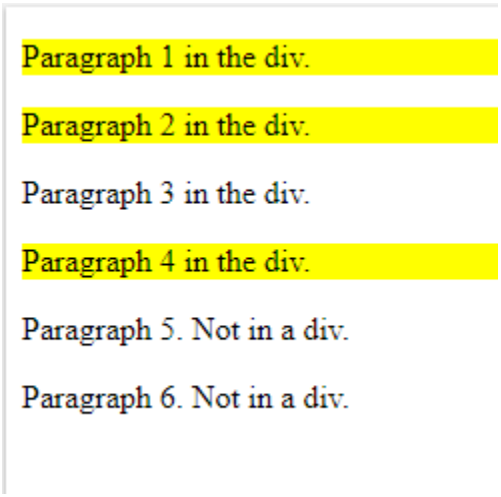
The following example selects all <p> elements that are children of a element:

```
div > p {
  background-color: yellow;
}
```

Example

```
<html>
<head>
<style>
div > p {
  background-color: yellow;
}
</style>
</head>
<body>

<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <section><p>Paragraph 3 in the div.</p></section> <!-- not Child but Descendant -->
  <p>Paragraph 4 in the div.</p>
</div>

<p>Paragraph 5. Not in a div.</p>
<p>Paragraph 6. Not in a div.</p>

</body>
</html>
```

Output



Adjacent Sibling Selector

The adjacent sibling selector selects all elements that are the adjacent siblings of a specified element.

Sibling elements must have the same parent element, and "adjacent" means "immediately following".

```
div + p {
  background-color: yellow;
}
```

Example:

```
<html>
<head>
<style>
div + p {
  background-color: yellow;
}
</style>
</head>
<body>

<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. Not in a div.</p>
```

```
<p>Paragraph 4. Not in a div.</p>

</body>
</html>
```

Output

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3. Not in a div.

Paragraph 4. Not in a div.

General Sibling Selector

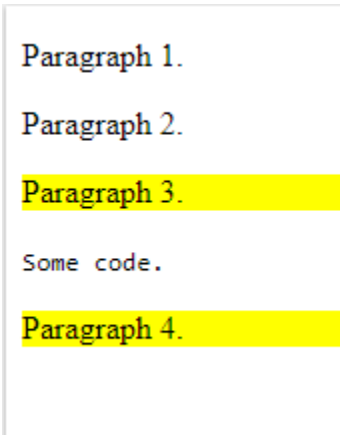
The general sibling selector selects all elements that are siblings of a specified element.

```
div ~ p {
  background-color: yellow;
}
```

Example

```
<html>
<head>
<style>
div ~ p {
  background-color: yellow;
}
</style>
</head>
<body>

<p>Paragraph 1.</p>

<div>
  <p>Paragraph 2.</p>
</div>

<p>Paragraph 3.</p>
<code>Some code.</code>
<p>Paragraph 4.</p>

</body>
</html>
```

Output



CSS Borders

The CSS `border` properties allow you to specify the style, width, and color of an element's border.

Border Style

The `border-style` property specifies what kind of border to display.

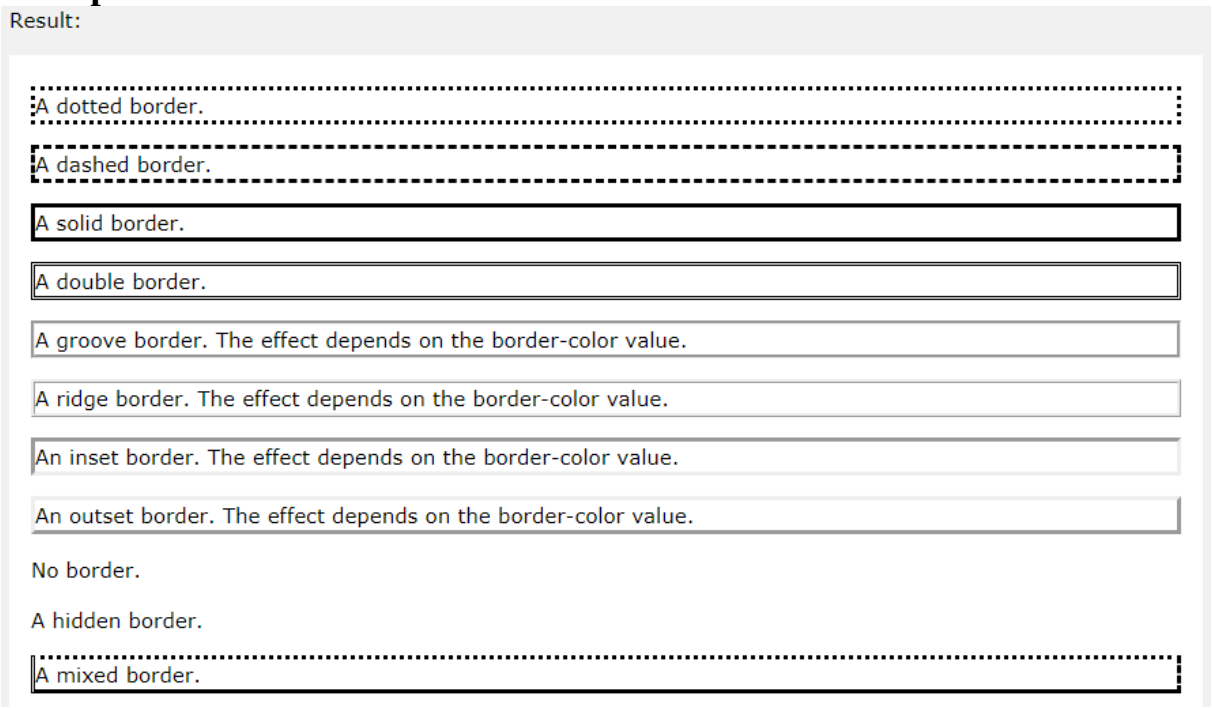
The following values are allowed:

- `dotted` - Defines a dotted border
- `dashed` - Defines a dashed border
- `solid` - Defines a solid border
- `double` - Defines a double border
- `groove` - Defines a 3D grooved border. The effect depends on the border-color value
- `ridge` - Defines a 3D ridged border. The effect depends on the border-color value
- `inset` - Defines a 3D inset border. The effect depends on the border-color value
- `outset` - Defines a 3D outset border. The effect depends on the border-color value
- `none` - Defines no border
- `hidden` - Defines a hidden border

The `border-style` property can have from one to four values (for the top border, right border, bottom border, and the left border).

Example

Result:



CSS Margins

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

CSS has properties for specifying the margin for each side of an element:

- margin-top
- margin-right
- margin-bottom
- margin-left

All the margin properties can have the following values:

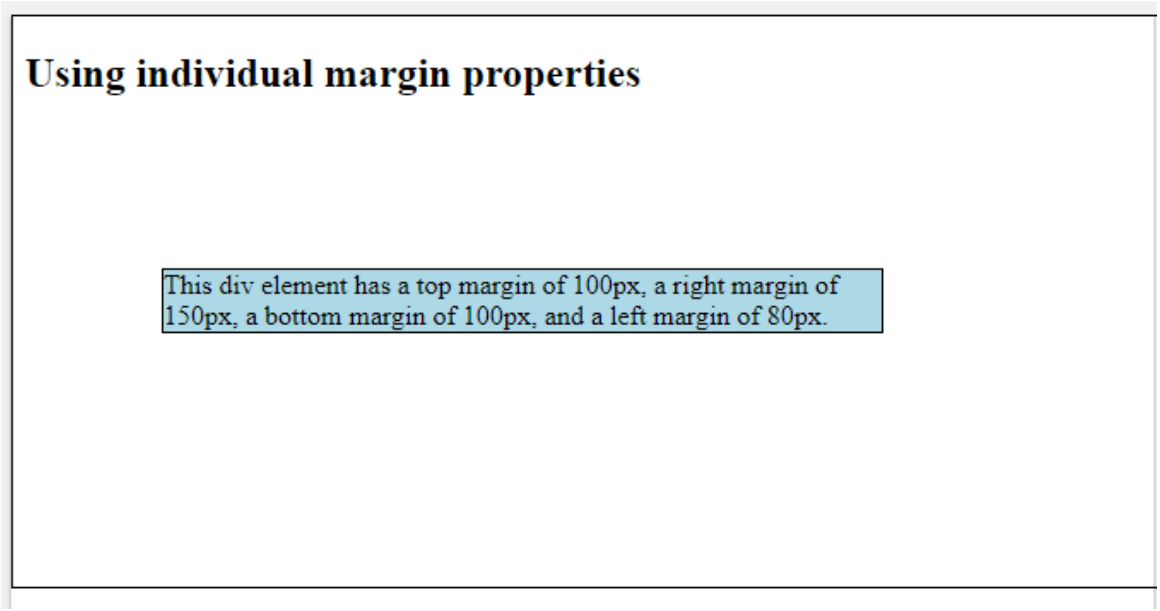
- auto - the browser calculates the margin
- *length* - specifies a margin in px, pt, cm, etc.
- *%* - specifies a margin in % of the width of the containing element
- inherit - specifies that the margin should be inherited from the parent element

Tip: Negative values are allowed.

Example

```
p {
  margin-top: 100px;
  margin-bottom: 100px;
  margin-right: 150px;
  margin-left: 80px;
}
```

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  border: 1px solid black;
  margin-top: 100px;
  margin-bottom: 100px;
  margin-right: 150px;
  margin-left: 80px;
  background-color: lightblue;
}
</style>
</head>
<body>

<h2>Using individual margin properties</h2>

<div>This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div>

</body>
</html>
```



Margin - Shorthand Property

To shorten the code, it is possible to specify all the margin properties in one property.

The `margin` property is a shorthand property for the following individual margin properties:

- `margin-top`
- `margin-right`
- `margin-bottom`
- `margin-left`

So, here is how it works:

If the `margin` property has four values:

- **margin: 25px 50px 75px 100px;**
  - top margin is 25px
  - right margin is 50px
  - bottom margin is 75px
  - left margin is 100px
- **margin: 25px 50px 75px;**
  - top margin is 25px
  - right and left margins are 50px
  - bottom margin is 75px
- **margin: 25px 50px;**

    - top and bottom margins are 25px
    - right and left margins are 50px
- **margin: 25px;**
    - all four margins are 25px

Example

```
p {
    margin: 25px 50px 75px 100px;
}
```

The auto Value

You can set the margin property to `auto` to horizontally center the element within its container.

The element will then take up the specified width, and the remaining space will be split equally between the left and right margins:

Example:

```
div {
  width: 300px;
  margin: auto;
  border: 1px solid red;
}
```

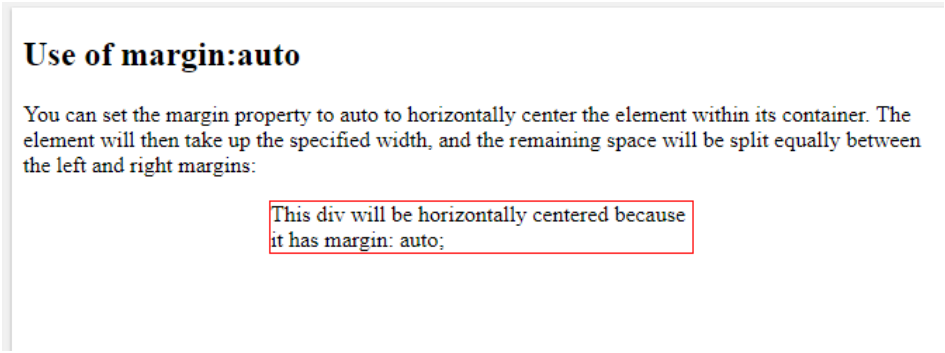
Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width:300px;
  margin: auto;
  border: 1px solid red;
}
</style>
</head>
<body>

<h2>Use of margin:auto</h2>
```

You can set the margin property to auto to horizontally center the element within its container. The element will then take up the specified width, and the remaining space will be split equally between the left and right margins:

This div will be horizontally centered because it has margin: auto;

</body>
</html>



CSS Colors

Colors are specified using predefined color names, or RGB, HEX, HSL (Hue, Saturation, Lightness) , RGBA (red, green, blue, alpha), HSLA values.

HTML supports 140 standard color names.

Background Color

You can set the background color for HTML elements:

<h1 style="background-color:Blue;">Hello World</h1>
This is cosmos college

Text Color

You can set the color of text:

<h1 style="color:Tomato;">Hello World</h1>
cosmos college
Tutepani Lalitpur

Border Color

You can set the color of borders:

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

CSS Text-align

Example

Set the text alignment for different elements

```
div.a {
  text-align: center;
}

div.b {
  text-align: left;
}

div.c {
  text-align: right;
}

div.c {
```

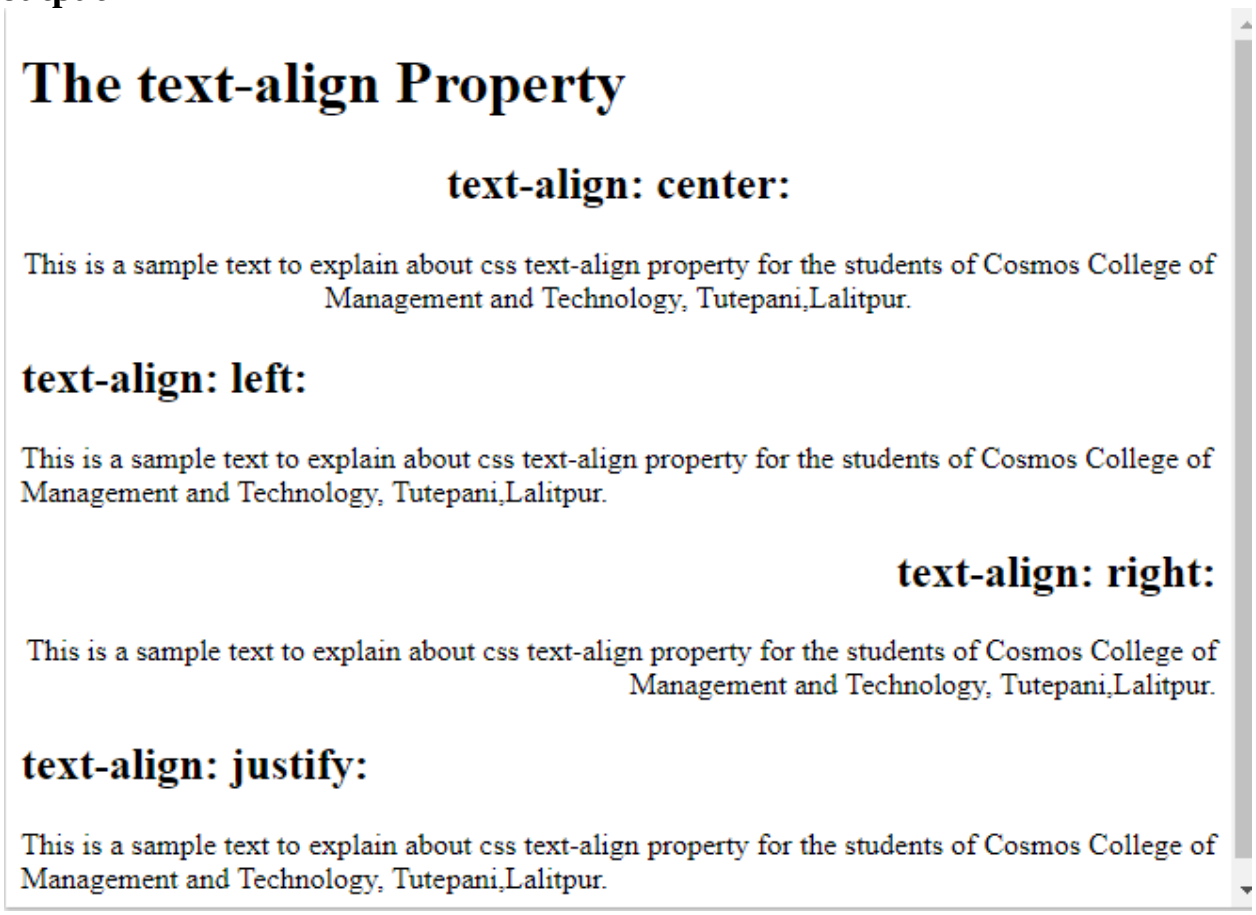
```
  text-align: justify;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
div.a {
   text-align: center;
}

div.b {
   text-align: left;
}

div.c {
   text-align: right;
}

div.d {
   text-align: justify;
}
</style>
</head>
<body>
<h1>The text-align Property</h1>

<div class="a">
<h2>text-align: center:</h2>
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>

<div class="b">
<h2>text-align: left:</h2>
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>

<div class="c">
<h2>text-align: right:</h2>
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>
```

```
<div class="d">
<h2>text-align: justify:</h2>
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>
</body>
</html>
```

output



CSS text-decoration

Example

Set different text decorations for <h1>, <h2>, and <h3> elements:

```
h1 {
  text-decoration: overline;
}

h2 {
  text-decoration: line-through;
```

```
}

h3 {
  text-decoration: underline;
}

h3 {
  text-decoration: underline overline;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    text-decoration: overline;
}

h2 {
    text-decoration: line-through;
}

h3 {
    text-decoration: underline;
}

h4 {
    text-decoration: underline overline;
}
</style>
</head>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
</body>
</html>
```

Output

This is heading 1

~~This is heading 2~~

This is heading 3

This is heading 4

CSS text-indent

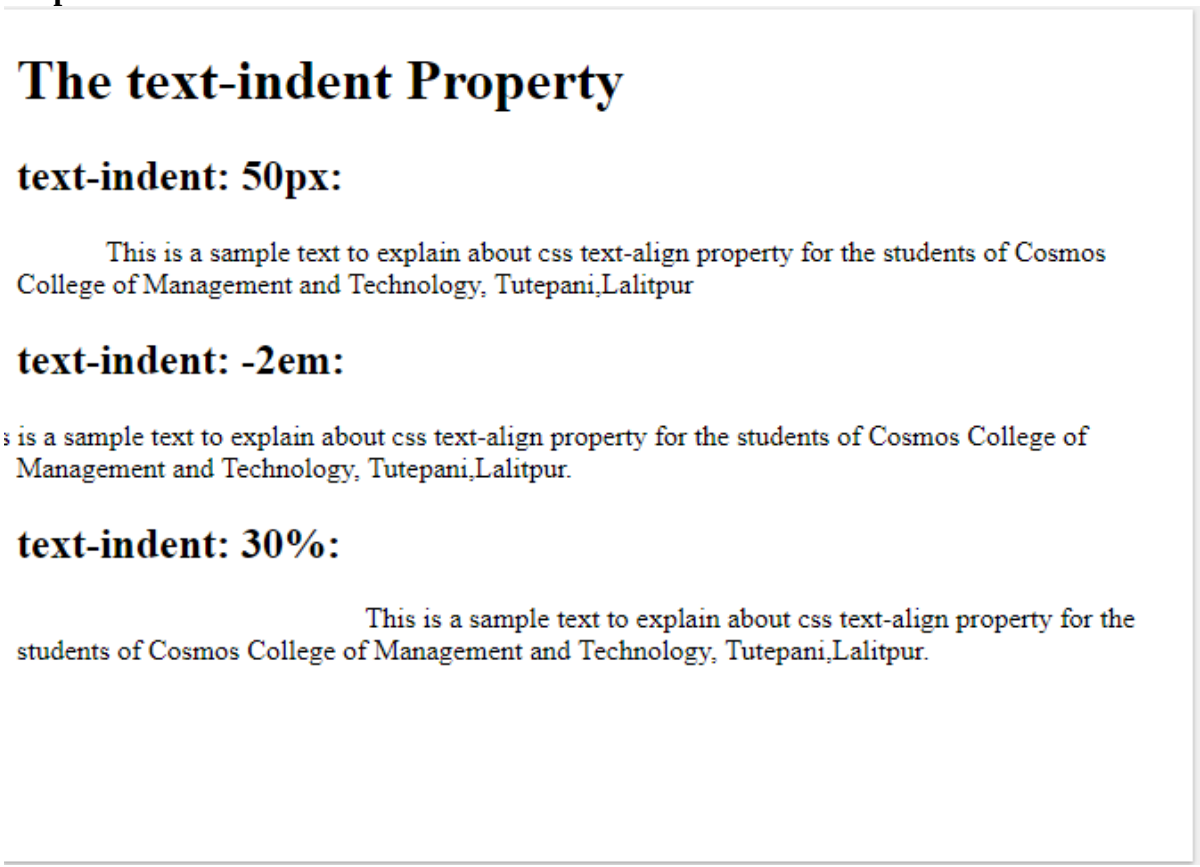
```
div.a {
  text-indent: 50px;
}

div.b {
  text-indent: -2em;
}

div.c {
  text-indent: 30%;
}
```

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
div.a {
    text-indent: 50px;
}

div.b {
    text-indent: -2em;
}

div.c {
    text-indent: 30%;
}
</style>
```

```
</head>
<body>
<h1>The text-indent Property</h1>

<h2>text-indent: 50px:</h2>
<div class="a">
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur</p>
</div>

<h2>text-indent: -2em:</h2>
<div class="b">
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>

<h2>text-indent: 30%:</h2>
<div class="c">
<p>This is a sample text to explain about css text-align property for the students of Cosmos College of Management and Technology, Tutepani,Lalitpur.</p>
</div>
</body>
</html>
```

Output



CSS text-justify

Set the justification method to "inter-word" when text-align is set to "justify":

```
div {
  text-align: justify;
  text-justify: inter-word;
}
```

The text-justify property specifies the justification method of text when text-align is set to "justify"

CSS text-transform

Transform text in different elements:

The text-transform property controls the capitalization of text.

```
div.a {
  text-transform: uppercase;
}

div.b {
```

```
  text-transform: lowercase;
}

div.c {
  text-transform: capitalize;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
div.a {
   text-transform: uppercase;
}

div.b {
   text-transform: lowercase;
}

div.c {
   text-transform: capitalize;
}
</style>
</head>
<body>
<h1>The text-transform Property</h1>

<h2>text-transform: uppercase:</h2>
<div class="a">This is cosmos college of management and technology.</div>

<h2>text-transform: lowercase:</h2>
<div class="b">This is cosmos college of management and technology.</div>

<h2>text-transform: capitalize:</h2>
<div class="c">This is cosmos college of management and technology.</div>

</body>
</html>
```

Output

CSS Font Family

The font-family property specifies the font for an element.

The font-family property can hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.

There are two types of font family names:

- **family-name** - The name of a font-family, like "times", "courier", "arial", etc.
- **generic-family** - The name of a generic-family, like "serif", "sans-serif", "cursive", "fantasy", "monospace".

Start with the font you want, and always end with a generic family, to let the browser pick a similar font in the generic family, if no other fonts are available.

Note: Separate each value with a comma.

Note: If a font name contains white-space, it must be quoted. Single quotes must be used when using the "style" attribute in HTML.

Example

```
p.a {
  font-family: "Times New Roman", Times, serif;
}

p.b {
  font-family: Arial, Helvetica, sans-serif;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
p.a {
  font-family: "Times New Roman", Times, serif;
}

p.b {
  font-family: Arial, Helvetica, sans-serif;
}
</style>
</head>
<body>
<h1>The font-family Property</h1>

<p class="a">This is a paragraph, shown in the Times New Roman font.</p>

<p class="b">This is a paragraph, shown in the Arial font.</p>

</body>
</html>
```

Output

CSS Font Size

The font-size property sets the size of a font

Example

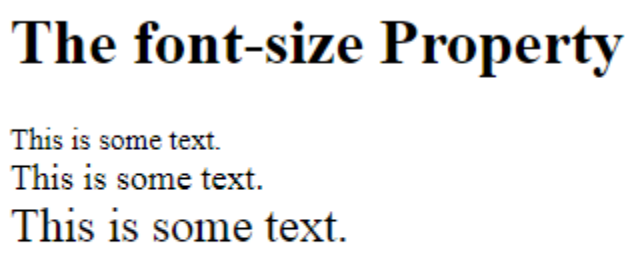
```
div.a {
  font-size: 15px;
}

div.b {
  font-size: large;
}

div.c {
  font-size: 150%;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
div.a {
  font-size: 15px;
}

div.b {
  font-size: large;
}

div.c {
```

```
  font-size: 150%;
}
</style>
</head>
<body>
<h1>The font-size Property</h1>

<div class="a">This is some text.</div>

<div class="b">This is some text.</div>

<div class="c">This is some text.</div>

</body>
</html>
```

output



CSS Font Style

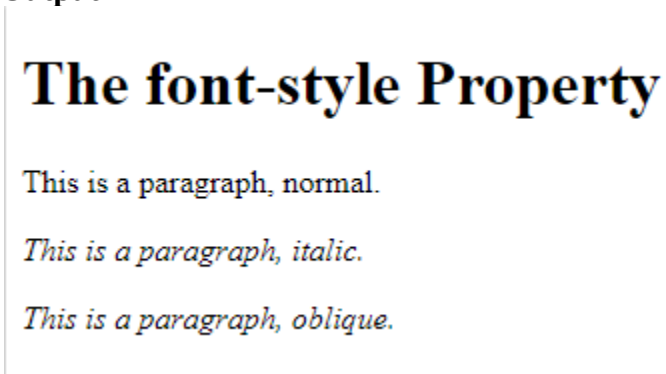
The font-style property specifies the font style for a text.

```
p.a {
  font-style: normal;
}

p.b {
  font-style: italic;
}

p.c {
  font-style: oblique;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
p.a {
  font-style: normal;
}

p.b {
  font-style: italic;
}

p.c {
  font-style: oblique;
}
</style>
</head>
<body>
<h1>The font-style Property</h1>

<p class="a">This is a paragraph, normal.</p>
<p class="b">This is a paragraph, italic.</p>
<p class="c">This is a paragraph, oblique.</p>

</body>
</html>
```

Output



CSS font-variant

In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

The font-variant property specifies whether or not a text should be displayed in a small-caps font.

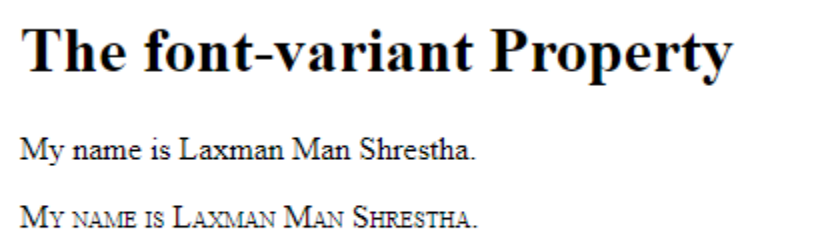
Example

```
p.small {
  font-variant: small-caps;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
p.normal {
  font-variant: normal;
}

p.small {
  font-variant: small-caps;
}
</style>
</head>
<body>
<h1>The font-variant Property</h1>

<p class="normal">My name is Laxman Man Shrestha.</p>
<p class="small">My name is Laxman Man Shrestha.</p>

</body>
</html>
```

Output



CSS font-weight

The font-weight property sets how thick or thin characters in text should be displayed.

```
p.normal {
  font-weight: normal;
}

p.thick {
  font-weight: bold;
}

p.thicker {
  font-weight: 900;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
p.normal {
  font-weight: normal;
}

p.light {
  font-weight: lighter;
}

p.thick {
  font-weight: bold;
}

p.thicker {
  font-weight: 900;
}
</style>
</head>
<body>

<h1>The font-weight Property</h1>

<p class="normal">This is a paragraph.</p>
<p class="light">This is a paragraph.</p>
<p class="thick">This is a paragraph.</p>
<p class="thicker">This is a paragraph.</p>
```

```
</body>
</html>
```

Output

The font-weight Property

This is a paragraph.

This is a paragraph.

This is a paragraph.

This is a paragraph.

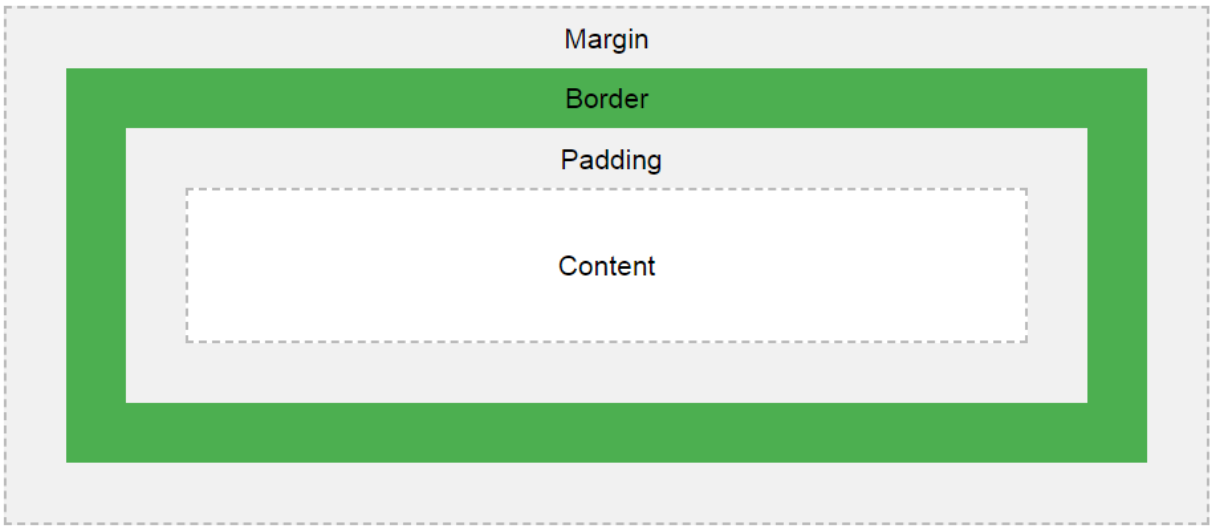
The CSS Box Model

All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model:

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content
- **Margin** - Clears an area outside the border. The margin is transparent

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




```
div {
  width: 300px;
  border: 25px solid green;
  padding: 25px;
  margin: 25px;
}
```

Important: When you set the width and height properties of an element with CSS, you just set the width and height of the **content area**. To calculate the full size of an element, you must also add padding, borders and margins.

How it works ?

Assume we want to style a element to have a total width of 350px:

```
div {
  width: 320px;
  padding: 10px;
  border: 5px solid gray;
  margin: 0;
}
```

Here is the calcultion:

320px (width)
+ 20px (left + right padding)
+ 10px (left + right border)
+ 0px (left + right margin)
= 350px

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 320px;
  padding: 10px;
  border: 5px solid gray;
  margin: 0;
}
</style>
</head>
<body>

<h2>Calculate the total width:</h2>

<img src="klematis4_big.jpg" width="350" height="263" alt="Klematis">
<div>The picture above is 350px wide. The total width of this element is also 350px.</div>

</body>
</html>
```

Output

Calculate the total width:

The picture above is 350px wide. The total width of this element is also 350px.

CSS Lists

Unordered Lists:

- Coffee
- Tea
- Coca Cola

- Coffee
- Tea
- Coca Cola

Ordered Lists:

1. Coffee
2. Tea
3. Coca Cola

I. Coffee
II. Tea
III. Coca Cola

HTML Lists and CSS List Properties

In HTML, there are two main types of lists:

- unordered lists (<ul>) - the list items are marked with bullets
- ordered lists (<ol>) - the list items are marked with numbers or letters

The CSS list properties allow you to:

- Set different list item markers for ordered lists
- Set different list item markers for unordered lists
- Set an image as the list item marker
- Add background colors to lists and list items

Different List Item Markers

The list-style-type property specifies the type of list item marker.

Example

```
ul.a {
  list-style-type: circle;
}

ul.b {
  list-style-type: square;
}

ol.c {
  list-style-type: upper-roman;
}

ol.d {
  list-style-type: lower-alpha;
}
```

Example Program

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

```
ul.a {
  list-style-type: circle;
}

ul.b {
  list-style-type: square;
}

ol.c {
  list-style-type: upper-roman;
}

ol.d {
  list-style-type: lower-alpha;
}
</style>
</head>
<body>

<p>Example of unordered lists:</p>
<ul class="a">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

<ul class="b">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

<p>Example of ordered lists:</p>
<ol class="c">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ol>

<ol class="d">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ol>

</body>
```

```
</html>
```

Output

Example of unordered lists:

- Coffee
- Tea
- Coca Cola

- Coffee
- Tea
- Coca Cola

Example of ordered lists:

I. Coffee
II. Tea
III. Coca Cola

a. Coffee
b. Tea
c. Coca Cola

An Image as The List Item Marker

The list-style-image property specifies an image as the list item marker:

Example

```
ul {
  list-style-image: url('sqpurple.gif');
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
ul {
  list-style-image: url('sqpurple.gif');
}
</style>
</head>
<body>

<ul>
```

```
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ul>

</body>
</html>
```

Output

- Coffee
- Tea
- Coca Cola

Position The List Item Markers

The list-style-position property specifies the position of the list-item markers (bullet points).

"list-style-position: outside;" means that the bullet points will be outside the list item. The start of each line of a list item will be aligned vertically. This is default:

- Coffee - A brewed drink prepared from roasted coffee beans...
- Tea
- Coca-cola

"list-style-position: inside;" means that the bullet points will be inside the list item. As it is part of the list item, it will be part of the text and push the text at the start:

- Coffee - A brewed drink prepared from roasted coffee beans...
- Tea
- Coca-cola

Example

```
ul.a {
  list-style-position: outside;
}

ul.b {
  list-style-position: inside;
}
```

Example Program

```
<!DOCTYPE html>
<html>
<head>
<style>
ul.a {
  list-style-position: outside;
}

ul.b {
  list-style-position: inside;
}
</style>
</head>
<body>

<h1>The list-style-position Property</h1>

<h2>list-style-position: outside (default):</h2>
<ul class="a">
  <li>Coffee - A brewed drink prepared from roasted coffee beans</li>
  <li>Tea - An aromatic beverage commonly prepared by pouring hot or boiling water over cured leaves of the Camellia sinensis</li>
  <li>Coca Cola - A carbonated soft drink </li>
</ul>

<h2>list-style-position: inside:</h2>
<ul class="b">
  <li>Coffee - A brewed drink prepared from roasted coffee beans</li>
  <li>Tea - An aromatic beverage commonly prepared by pouring hot or boiling water over cured leaves of the Camellia sinensis</li>
  <li>Coca Cola - A carbonated soft drink</li>
</ul>

</body>
</html>
```

Output

The list-style-position Property

list-style-position: outside (default):

- Coffee - A brewed drink prepared from roasted coffee beans
- Tea - An aromatic beverage commonly prepared by pouring hot or boiling water over cured leaves of the Camellia sinensis
- Coca Cola - A carbonated soft drink

list-style-position: inside:

- Coffee - A brewed drink prepared from roasted coffee beans
- Tea - An aromatic beverage commonly prepared by pouring hot or boiling water over cured leaves of the Camellia sinensis
- Coca Cola - A carbonated soft drink

Remove Default Settings

The list-style-type:none property can also be used to remove the markers/bullets. Note that the list also has default margin and padding. To remove this, add margin:0 and padding:0 to <ul> or <ol>:

Example

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
```

List - Shorthand property

The list-style property is a shorthand property. It is used to set all the list properties in one declaration:

Example

```
ul {
  list-style: square inside url("sqpurple.gif");
}
```

When using the shorthand property, the order of the property values are:

- list-style-type (if a list-style-image is specified, the value of this property will be displayed if the image for some reason cannot be displayed)
- list-style-position (specifies whether the list-item markers should appear inside or outside the content flow)
- list-style-image (specifies an image as the list item marker)

If one of the property values above are missing, the default value for the missing property will be inserted, if any.

Styling List With Colors

We can also style lists with colors, to make them look a little more interesting.

Anything added to the <ol> or <ul> tag, affects the entire list, while properties added to the <li> tag will affect the individual list items:

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
ol {
  background: #ff9999;
  padding: 20px;
}

ul {
  background: #3399ff;
  padding: 20px;
}

ol li {
  background: #ffe5e5;
  padding: 5px;
  margin-left: 35px;
}

ul li {
  background: #cce5ff;
  margin: 5px;
}
</style>
</head>
<body>
```

```
<h1>Styling Lists With Colors:</h1>

<ol>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Coca Cola</li>
</ol>

<ul>
```

Output