Paper Code: BCA-107 **L T C**

3 1 4

Paper: Web Technologies

UNIT – II-Part1

Cascading Style Sheet: Types of Style Sheets – Internal, inline and External style sheets, creating styles, link tag, CSS Properties, CSS Styling, Style Selector-Id, class name and Pseudo class.

[No. of Hrs: 11]

CSS:- Cascading Style Sheet

CSS is the language we use to style a Web page.

CSS Introduction (w3schools.com) for demo click on this link

What is CSS?

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

Why Use CSS? CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

CSS Solved a Big Problem/advantages of CSS over HTML:-

- HTML was NEVER intended to contain tags for formatting a web page!
- HTML was created to describe the content of a web page, like:

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

- When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.
- To solve this problem, the World Wide Web Consortium (W3C) created CSS.

- CSS removed the style formatting from the HTML page!
- The style definitions are normally saved in external .css files.
- With an external stylesheet file, you can change the look of an entire website by changing just one file!

CSS Example

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}
h1 {
  color: white;
  text-align: center;
}
p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>

<h1>My First CSS Example</h1>
<p>This is a paragraph.</p>
<p> deepika </p

</body>

</html>
```

A CSS rule consists of a selector and a declaration block.

CSS Syntax:

The selector points to the HTML element you want to style.

```
h1 {
  color: white;
  text-align: center;
}
```

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:** In this example all <p> elements will be center-aligned, with a red text color:

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  color: red;
  text-align: center;
}
</style>
</head>

<body>
<p>Hello World!</p>
<p>These paragraphs are styled with CSS.</p>
</body>
</html>
```

Example Explained

- 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

CSS Selectors:- A CSS selector selects the HTML element(s) you want to style.

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

1. **The CSS element Selector**

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

Example

Here, all <p> elements on the page will be center-aligned, with a red text color:

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

2. The CSS id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.
- The id of an element is unique within a page, so the id selector is used to select one unique element!
- To select an element with a specific id, write a hash (#) character, followed by the id of the element.

Example

The CSS rule below will be applied to the HTML element with id="para1":

```
<!DOCTYPE html>
<html>
<head>
    <style>
    #para1 {
      text-align: center;
      color: red;
    }
    </style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

Note: An id name cannot start with a number!

3. The CSS class Selector

- The class selector selects HTML elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the class name.

```
<!DOCTYPE html>
<html>
<head>
        <style>
        .center {
          text-align: center;
          color: red;
        }
        </style>
</head>
        <body>

        <h1 class="center">Red and center-aligned heading</h1>
        <p class="center">Red and center-aligned paragraph.</p>

        </body>
</html>
```

You can also specify that only specific HTML elements should be affected by a class. Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>

</body>
</html>
```

HTML elements can also refer to more than one class.

4. The Universal Selectors

Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type:

```
* { color: #000000; }
```

This rule renders the content of every element in our document in black.

The universal selector (*) selects all HTML elements on the page.

```
<!DOCTYPE html>
<html>
<head>
<style>
* {
  text-align: center;
  color: blue;
}
</style>
</head>
<body>
<h1>Hello world!</h1>
<p>Every element on the page will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>
</body>
</html>
```

Hello world!

Every element on the page will be affected by the style.

Me too!

And me!

5. **Grouping Selectors:** You can apply a style to many selectors if you like. Just separate the selectors with a comma, as given in the following example.
   - The same style definition is implemented over multiple elements.
   - Code size gets minimized.

- Commas are used to isolate every selector while grouping; hence it easily readable.

```
h1, h2, h3

{ color: #36C; font-weight: normal; letter-spacing: .4cm;

margin-bottom: 1cm; text-transform: lowercase;

}
```

The grouping selector selects all the HTML elements with the same style definitions. Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

```
h1 {
  text-align: center;
  color: red;
}

h2 {
  text-align: center;
  color: red;
}

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

It will be better to group the selectors, to minimize the code. To group selectors, separate each selector with a comma.

```
<!DOCTYPE html>
<html>
<head>
<style>
h1, h2, p {
  text-align: center;
  color: red;
}
```

```
</style>
</head>
<body>

<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>

</body>
</html>
```

Inclusion of CSS

When a browser reads a style sheet, it will format the HTML document according to the information in the style sheet.

Three Ways to Insert CSS

There are three ways of inserting a style sheet:

1. **External CSS**
2. **Internal CSS**
3. **Inline CSS**

1. External CSS/ external linking:

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.

Style sheets are a convenient way to create a document with a uniform theme. With external style sheets (i.e., separate documents that contain only CSS rules), you can provide a uniform look and feel to an entire website. Different pages on a site can all use the same style sheet. When changes to the styles are required, ***the author needs to modify only a single CSS file to make style changes across the entire website***. Note that while embedded style sheets separate content from presentation, both are still contained in a single file, preventing a web designer and a content author from working in parallel. *External style sheets solve this problem by separating the content and style into separate files.*

Example

External styles are defined within the <link> element, inside the <head> section of an HTML page:

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

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

</body>
</html>
```

An external style sheet can be written in any text editor, and must be saved with a .css extension. The external .css file should not contain any HTML tags. Here is how the "mystyle.css" file looks: **"mystyle.css"**

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

Note: Do not add a space between the property value and the unit (such as margin-left: 20 px;). The correct way is: margin-left: 20px;

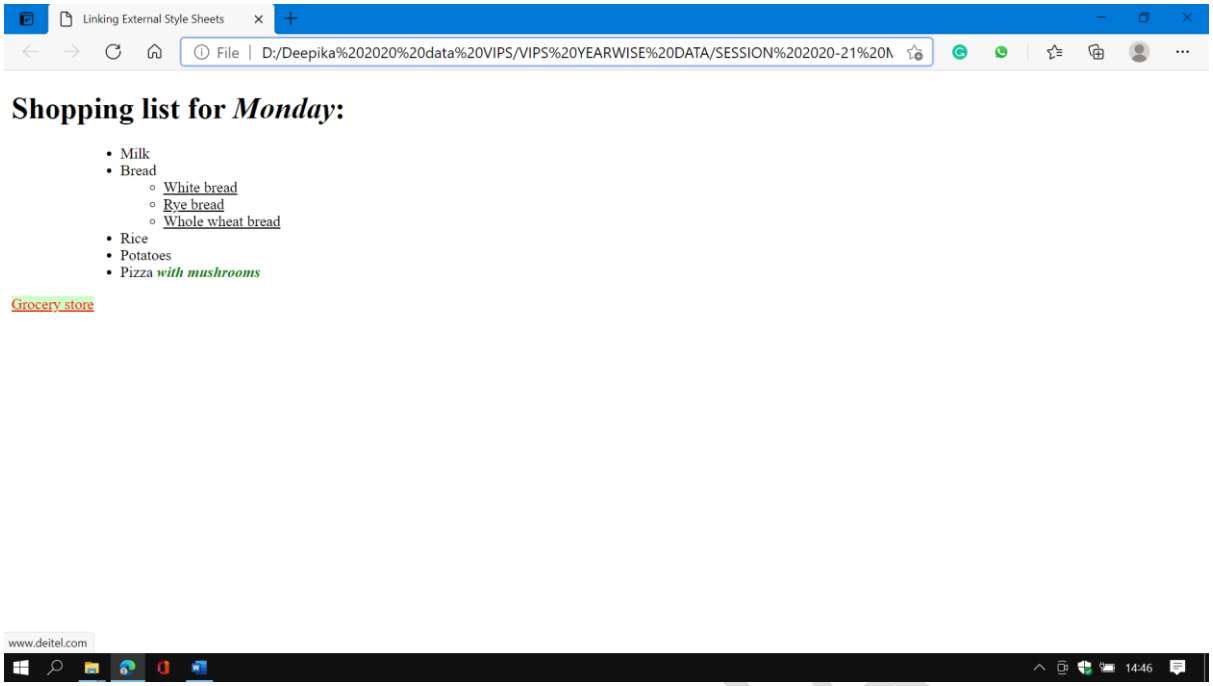
Example2:- External linking can give same uniform look to completed website. Separate pages on the site can utilize the same style sheet and you would have to modify a single file to make changes to styles across your whole web site. Lets create a file named styles.css as below:-

```
A       { text-decoration: none }

A:hover   { text-decoration: underline; color: red; background-color: #CCFFCC
}
```

```
LI EM        { color: green; font-weight: bold}

UL    { margin-left:2cm }

UL UL        { text-decoration: underline; margin-left: .15cm  }
```

Linking external sheet

```
<head>
<title>Linking External Style Sheets</title>
<link rel= “stylesheet” type= “text/css” href= “styles.css”>
</head>
<body>
 <h1>Shopping list for <em>Monday</em>:</h1>
 <ul>
 <li>Milk</li>
 <li>Bread
    <ul>
    <li>White bread</li>
    <li>Rye bread</li>
    <li>Whole wheat bread</li>
    </ul>
 </li>
 <li>Rice</li>
 <li>Potatoes</li>
<li>Pizza <em>with mushrooms</em></li>
</ul>
<a  href = "http://www.deitel.com"> Grocery store</a>

 </body>
 </html>
```



Here in the example:-***A:hover{----}*** , specify a style for hover, which is a pseudo-class that gives the author access to the content not specifically declare in the document. The hover pseudoclass gets dynamically activated when the user moves the mouse cursor over an A element.

Also, LI EM { color: green; font-weight: bold}

Declare a style for all EM elements that are the children of LI elements. Here Monday is not red and bold as it is not a list element as with mushrooms.

UL UL { text-decoration: underline; margin-left: .15cm }

In above line, it specifies that all the ***nested lists*** (UL elements that are children of UL elements) will be underlined and have a left-hand margin of 15 cm.

Unit of measuring pixel size is em,ex,percentage,cm,in,pt,pc,mm.

<LINK REL= "stylesheet" TYPE="text/css" HREF= "styles.css">

Here in above line, link element which specifies a relationship between the current document and another document using the REL attribute. We declare the linked document to be stylesheet for this document. Type attribute is to specify the MIME type as text/css and provide the URL for the stylesheet with the HREF attribute.

2. 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.

Example

Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

```
<!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>
```

3. 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.

Example

Inline styles are defined within the "style" attribute of the relevant element:

```
<!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>

<p style= “font-size:20pt; color: #0000FF”> Deepika with 20 point text size</p>

</body>
</html>
```

External Stylesheet	Internal CSS	Inline CSS

<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>
 	
 This is a heading
 This is a paragraph.
 	
</body>
</html>
 	
Here is how the "mystyle.css" file looks:
"mystyle.css"
body {
background-color: lightblue;
}
h1 {
color: navy;
margin-left: 20px;
}
	<!DOCTYPE html>
	<html>
	<head>
	<style>
	body {
	background-color: linen;
	}
	
	h1 {
	color: maroon;
	margin-left: 40px;
	}
	</style>
	</head>
	<body>
	
	 This is a heading
	 This is a paragraph.
	
	</body>
	</html>
		<!DOCTYPE html>
		<html>
		<body>
		 This is a heading
		 This is a paragraph.
		<p style= “font-size:20pt; color: #0000FF”> Deepika with 20 point text size
		
		</body>
		</html>

Creating styles sheets with the STYLE element

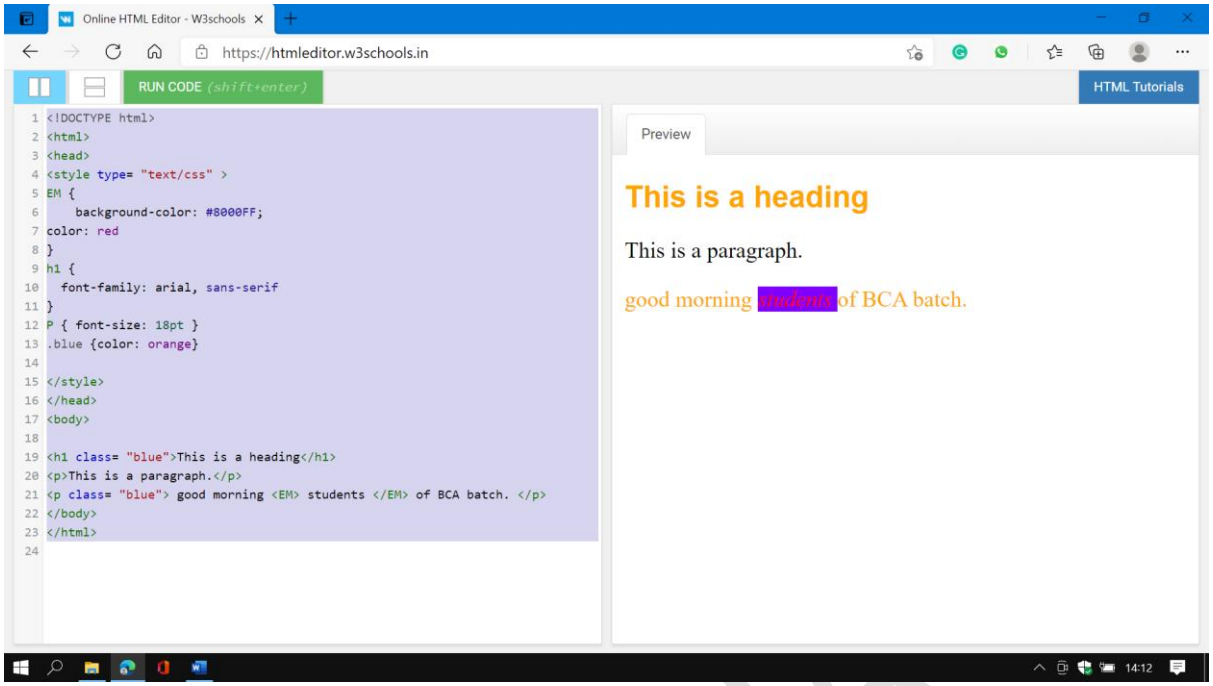
Styles that may be applied to entire document cane be given in header section of the document. The type attribute specifies the Internet media type (formerly known as MIME type) of the <style> tag.

The type attribute identifies the content between the <style> and </style> tags.

The default value is "text/css", which indicates that the content is CSS. Here in code below em, h1 and p are elements. Rules are given in { }. Class style named blue shows that all class declarations preceded with a period and are applied to elements only of that specific class. Arial font is name of the font and sans-serif

is a generic font-family. i.e here if arial font is not found on the system, the browser will instead display another sans-serif font such as verdana or Helvetica.

```
<!DOCTYPE html>
<html>
<head>
<style type= "text/css" >
EM {
        background-color: #8000FF;
color: red
}
h1 {
  font-family: arial, sans-serif
}
P { font-size: 18pt }
.blue {color: orange}

</style>
</head>
<body>

<h1 class= "blue">This is a heading</h1>
<p>This is a paragraph.</p>
<p class= "blue"> good morning <EM> students </EM> of BCA batch. </p>
</body>
</html>
```



CSS Comments

 Comments are used to explain your code, and may help you when you edit the source code at a later date. Comments are ignored by browsers.
 A CSS comment begins with "/*", and ends with "*/", like this:
 /*This is a comment*/

```
p
{
text-align:center;
/*This is another comment*/
color:black;
font-family:arial;
}
```

Background Properties in CSS:-

These helps give background effects to your HTML documents. These are:

- background-color
- background-image
- background-repeat
- background-attachment

- background-position

Setting Background of HTML element

You can set the following background properties of an element:

• The background-color property is used to set the background color of an element.

• The background-image property is used to set the background image of an element.

• The background-repeat property is used to control the repetition of an image in the background.

• The background-position property is used to control the position of an image in the background.

• The background-attachment property is used to control the scrolling of an image in the background.

• The background property is used as a shorthand to specify a number of other background properties.

Example:-

```
<html>
<head>
 <style>
 body {
 background-color: lightblue;
 background-image:
url("https://www.google.co.in/images/branding/googlelogo/2x/googlelogo_color_272x92dp.png");
 background-repeat: no-repeat;
 background-position: right bottom;
 margin-right: 200px;
background-attachment: fixed; //To specify that the background image should be fixed
 }

 h1 {
```

```
background-color: aqua;
}

p {
background-color: lightyellow;
}
</style>
</head>
<body>
<h1>Sample text</h1>
<p>This page has a background color!</p>
<p> background no-repeat, set position example.</p>
<p>Now the background image is only shown once, and positioned away from the text.</p>
<p>In this example we have also added a margin on the right side, so the background image will never
disturb the text.</p>
</body>
</html>
```

FONTS properties in CSS

How to set fonts of a content, available in an HTML element. We can set the following font properties of an element:

- ***The font-family property is used to change the face of a font.***
- ***The font-style property is used to make a font italic or oblique.***
- ***The font-variant property is used to create a small-caps effect.***

font-variant: normal| small-caps| initial| inherit;

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

• The font-weight property is used to increase or decrease how bold or light a font appears.

font-weight: normal|bold|bolder|lighter|number|initial|inherit;

• The font-size property is used to increase or decrease the size of a font.

• The font property is used as shorthand to specify a number of other font properties.

All CSS Font Properties

Property	Description
font	Sets all the font properties in one declaration
font-family	Specifies the font family for text
font-size	Specifies the font size of text
font-style	Specifies the font style for text
font-variant	Specifies whether or not a text should be displayed in a small-caps font
font-weight	Specifies the weight of a font

```
<html>
<head>
  <style>
    p.normal {
      font-style: normal;
      font-family: "Times New Roman";
      font-size: 40px;
      font-weight: normal;
    }

    p.italic {
      font-style: italic;
      /*oblique - The text is "leaning" (oblique is very similar to italic, but less supported)*/
      font-family: Arial;
      font-size: 14px;
      font-weight: bold;
      font-variant: small-caps;
    }
  </style>
</head>

<body>

  <p class="normal">This is a paragraph in normal style.</p>
  <p class="normal">Normal text continued.</p>
  <p class="italic">This is a paragraph in italic style.</p>

</body>

</html>
```

Shorthand Property

You can use the *font* property to set all the font properties at once. For example:

```
<p style="font:italic small-caps bold 15px georgia;">
Applying all the properties on the text at once.
 </p>
```

It will produce the following result:

APPLYING ALL THE PROPERTIES ON THE TEXT AT ONCE.

Manipulating text using CSS properties

You can set the following text properties of an element:

- The **color** property is used to set the color of a text.
- The **direction** property is used to set the text direction.
- The **letter-spacing** property is used to add or subtract space between the letters that make up a word.
- The **word-spacing** property is used to add or subtract space between the words of a sentence.
- The **text-indent** property is used to indent the text of a paragraph.
- The **text-align** property is used to align the text of a document.
- The **text-decoration** property is used to underline, overline, and strikethrough text.
- The **text-transform** property is used to capitalize text or convert text to uppercase or lowercase letters.
- The **white-space** property is used to control the flow and formatting of text.
- The **text-shadow** property is used to set the text shadow around a text.

CSS text-shadow property (w3schools.com)

```
<!DOCTYPE html>
<html>

<head>
  <style>
        p.letterbody {
          color: blue;
                    /*text-transform: capitalize; */
           text-transform:uppercase;
                     /* can have uppercase lowercase
                     inherit or initial */
          text-indent: 4px;
                    /*text-indent: 50px;/-2em / 30%
                    length can be in px,pt,cm,em etc, default
                    is 0 */
          letter-spacing: 5px;
          line-height: 1.8;
          word-spacing: 8px;
                    text-decoration-style: solid;
                    /*wavy,double can be style too*/
```

```
        }

    h1 {
                color: green;
                text-align: center;
                    /* can have values left right justify also */
                text-decoration: overline;
                    /*can have values line-through,underline,
                    underline overline also */
                letter-spacing: -3px;
                text-shadow: 2px 2px red;
              }

    p.signature {
                direction: rtl;
                word-spacing: 10px;
     }
  </style>
</head>

<body>

  <h1>Sample letter</h1>

  <p class="letterbody"> It is with great interest, Deepika, that I am applying for the position of chief accountant. When I read the job description of your ad in the New York Times on August 12th, I felt that it was an ideal match with my career aspirations. I have always wanted to work for an oustanding company in the Fortune 500 such as Global Answers.    </p>

 <p class="signature">Sincerely, Ken Jacobs </p>


</body>

</html>
```

The `text-indent` property specifies the indentation of the first line in a text-block.

he `text-shadow` property adds shadow to text.This property accepts a comma-separated list of shadows to be applied to the text.

CSS Syntax

```
text-shadow: h-shadow v-shadow blur-radius color|none|initial|inherit;
```

Note: To add more than one shadow to the text, add a comma-separated list of shadows.

(-ve values are allowed in h and v shadow. In blur, by default is 0

Property Values

Value	Description
h-shadow	Required. The position of the horizontal shadow. Negative values are allowed
v-shadow	Required. The position of the vertical shadow. Negative values are allowed
blur-radius	Optional. The blur radius. Default value is 0
color	Optional. The color of the shadow. Look at CSS Color Values for a complete list of possible color values

none	Default value. No shadow
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

White-space property: The `white-space` property specifies how white-space inside an element is handled.

CSS Syntax

white-space: normal|nowrap|pre|pre-line|pre-wrap|initial|inherit;

```
p.a {
  white-space: nowrap;
}
```

Property Values

Value	Description
normal	Sequences of whitespace will collapse into a single whitespace. Text will wrap when necessary. This is default
nowrap	Sequences of whitespace will collapse into a single whitespace. Text will never wrap to the next line. The text continues on the same line until a tag is encountered
pre	Whitespace is preserved by the browser. Text will only wrap on line breaks. Acts like the <pre> tag in HTML
pre-line	Sequences of whitespace will collapse into a single whitespace. Text will wrap when necessary, and on line breaks

pre-wrap	Whitespace is preserved by the browser. Text will wrap when necessary, and on line breaks
initial	Sets this property to its default value. Read about *initial*
inherit	Inherits this property from its parent element. Read about *inherit*

```
text-transform: none|capitalize|uppercase|lowercase|initial|inherit;
```

CSS for Images

Images play an important role in any webpage. Though it is not recommended to include a lot of images, but it is still important to use good images wherever required.

CSS plays a good role to control image display. You can set the following image properties using CSS.

- The border property is used to set the width of an image border.
- The height property is used to set the height of an image.
- The width property is used to set the width of an image.
- The opacity property is used to set the opacity of an image.

```
<!DOCTYPE html>
<html>
<head>
   <style>
      img {
         border-radius: 15px 50px;
         -webkit-filter: grayscale(20%);
         height: auto;
         opacity: .5;
         width: auto;
      }
   </style>
</head>
```

```
<body>
   <h2>Rounded Images</h2>
   <p>Use the border-radius property to create circled images:</p>
   <img src="paris.jpg" alt="Paris">
</body>      </html>
```

Rounded Images

Use the border-radius property to create circled images:

The border-radius property is actually a shorthand property for the border-top-left-radius, border-topright-radius, border-bottom-right-radius and border-bottom-left-radius properties.

The border-radius property is used to add rounded corners to an element.

The border-radius property is a shorthand property for setting the four border-*-radius properties.

If you specify only one value for the border-radius property, this radius will be applied to all 4 corners.

However, you can specify each corner separately if you wish. Here are the rules:

• Four values: first value applies to top-left, second value applies to top-right, third value applies

to bottom-right, and fourth value applies to bottom-left corner

• Three values: first value applies to top-left, second value applies to top-right and bottom-left,

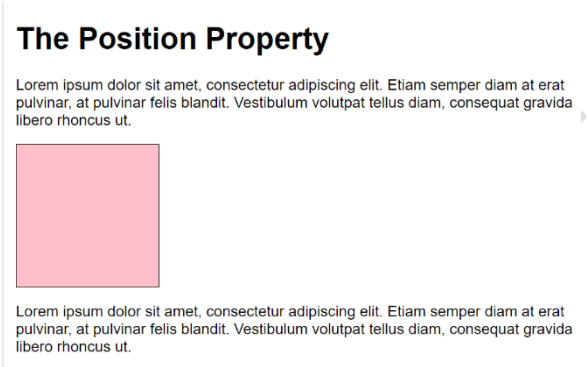
and third value applies to bottom-right

• Two values: first value applies to top-left and bottom-right corner, and the second value applies to top-right and bottom-left corner

• One value: all four corners are rounded equally

Example2:-

```
<!DOCTYPE html>
<html>
<head>
<style>
img {
   border-radius: 50%;
}
</style>
</head>
<body>
<h2>Rounded Images</h2>
<p>Use the border-radius property to create circled images:</p>
<img src="paris.jpg" alt="Paris" width="400" height="300">
</body>
</html>
```



Positioning Elements in CSS and Z index Property:

Earlier browser positioned the elements in an HTML document. Now you can position and control how our documents are displayed.

Example of position property in CSS:-

```
<!DOCTYPE html>
<html>
<head>
<title> CSS positions </title>
<style>
body {  font-family: arial; }
#test {
width: 150px;
```

```
height: 150px;
border: 1px solid #000;
background: pink;
}
</style>
</head>
<body>
<h1>The Position Property</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.</p>

<div id="test"> </div>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.</p>
</body>
</html>
```

We can set the position of the box created with div tag using different types of positioning as below:-

```
position: absolute;
    top: 30px;
    left:100px;
```

In absolute position, the box will move according to the screen. And in relative it will move as per its original position. Position fixed is fixed. It will not change as per the scroll if length of paragraph is increased, in absolute we can see after scrolling the page and box is absolute as per screen…..

Example with position property:-

```
<!DOCTYPE html>
<html>
<head>
 <style>
 img {
 position: relative;
 /*position: absolute;*/
```

```
left: 0px;
top: 0px;
z-index: -1;
}
</style>
</head>
<body>
<h2>Position property and Z index property</h2>
<p>Use the position property </p>
<img src=" paris.jpg" alt="Paris" height="300px" width="500px">
</body>
```

Position property

The position Property

The position property specifies the type of positioning method used for an element.

There are five different position values:

• static

• relative

• fixed

• absolute

- sticky

✓ Elements are then positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the position value.

position: static;

✓ HTML elements are positioned static by default.

✓ Static positioned elements are not affected by the top, bottom, left, and right properties.

- ✓ An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page (element will stick to the normal page flow)

position: relative;

- ✓ An element's original position remains in the flow of the document, just like the static value. But now left/right/top/bottom/z-index will work. The positional properties "nudge" the element from the original position in that direction. In relative mode, element will first be lay out on the page, then offset the element by the specified top, left, bottom or right values. Unlike absolute positioning, relative positioning keeps elements in general flow of the elements on the page.

position: absolute;

- ✓ An element with position: absolute; absolute property set element absolutely followed by the absolute offset from top, right, bottom or left. The element is removed from the flow of the document

position: fixed;

- ✓ The element is removed from the flow of the document like absolutely positioned elements. In fact, they behave almost the same, only fixed positioned elements are always relative to the document, not any particular parent, and are unaffected by scrolling.

The z-index property:-

It is used to displace elements on the z-axis i.e in or out of the screen. It is used to define the order of elements if they overlap on each other.

Syntax:
z-index: auto|number|initial|inherit;

Property values:

- **auto:** The stack order is equal to that of the parent(default).
- **number:** The stack order depends in the number.
- **initial**: Sets the property to its default value.
- **inherit:** Inherits the property from the parent element.

Overlapping Elements

When elements are positioned outside the normal flow, they can overlap other elements.

The z-index property specifies the stack order of an element (which element should be placed in front of, or

behind, the others).

An element can have a positive or negative stack order:

Example

```
img
{
position:absolute;
left:0px;
top:0px;
z-index:-1
}
```

An element with greater stack order is always in front of an element with a lower stack order.

Note: If two positioned elements overlap, without a z-index specified, the element positioned last in the HTML code will be shown on top.

Setting properties for links in CSS

To set different properties of a hyper link using CSS. You can set the following properties of a hyperlink:

We will revisit the same properties when we will discuss Pseudo-Classes of CSS.

- The :link signifies unvisited hyperlinks.
- The :visited signifies visited hyperlinks.
- The :hover signifies an element that currently has the user's mouse pointer hovering over it.
- The :active signifies an element on which the user is currently clicking.

Usually, all these properties are kept in the header part of the HTML document.

Remember a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective. Also, a:active MUST come after a:hover in the CSS

definition as follows:

```
<!DOCTYPE html>
```

```
<html>
<head>
<style>
/* unvisited link */
a:link {
 color: green;
}
/* visited link */
a:visited {
 color: blue;
}
/* mouse over link */
a:hover {
 color: red;
}
/* selected link */
a:active {
 color: yellow;
}
</style>
</head>
<body>
<p>Mouse over and click the link:
<a href="http://www.w3schools.com">w3schools.com</a></p>
</body>
</html>
```

CSS properties for tables

The following describes properties of an HTML table using CSS. You can set the following properties of a table:

• The border-collapse specifies whether the browser should control the appearance of the adjacent borders that touch each other or whether each cell should maintain its style.

• The border-spacing specifies the width that should appear between table cells.

Setting Table properties:

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Berglunds snabbkp	Christina Berglund	Sweden
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Koniglich Essen	Philip Cramer	Germany
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy
North/South	Simon Crowther	UK
Paris spcialits	Marie Bertrand	

• The caption-side captions are presented in the <caption> element. By default, these are rendered above the table in the document. You use the caption-side property to control the placement of the table caption.

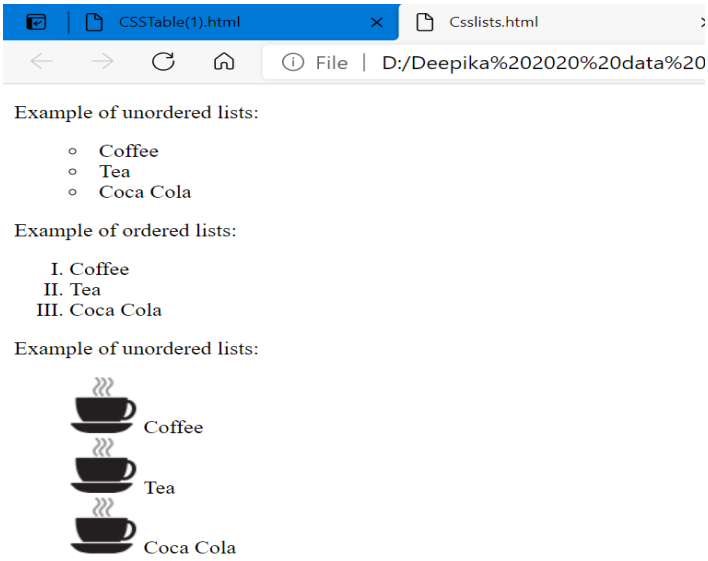
• The empty-cells specifies whether the border should be shown if a cell is empty.

• The table-layout allows browsers to speed up the layout of a table by using the first width properties it comes across for the rest of a column rather than having to load the whole table before rendering it.


```
<!DOCTYPE html>
<html>
<head>
 <style>
 table {
 empty-cells: hide;
 /* border-collapse: collapse;*/
 }
```

```
table, td, th {
border: 4px solid black;
border-spacing: 20px;
}
caption {
caption-side: bottom;
}
 #customers tr: nth-child(even) {
background-color: lightgoldenrodyellow;
}
th {
background-color: #4CAF50;
color: white;
}
</style>
</head>
<body>
<h2>Setting Table properties:</h2>
<table id="customers">
<tr>
<th>Company</th>
<th>Contact</th>
<th>Country</th>
</tr>
<tr>
<td>Alfreds Futterkiste</td>
<td>Maria Anders</td>
<td>Germany</td>
</tr>
<tr>
<td>Berglunds snabbkp</td>
<td>Christina Berglund</td>
<td>Sweden</td>
</tr>
<tr>
<td>Centro comercial Moctezuma</td>
<td>Francisco Chang</td>
<td>Mexico</td>
</tr>
<tr>
<td>Ernst Handel</td>
<td>Roland Mendel</td>
<td>Austria</td>
```

```
</tr>
<tr>
<td>Island Trading</td>
<td>Helen Bennett</td>
<td>UK</td>
</tr>
<tr>
<td>Koniglich Essen</td>
<td>Philip Cramer</td>
<td>Germany</td>
</tr>
<tr>
<td>Laughing Bacchus Winecellars</td>
<td>Yoshi Tannamuri</td>
<td>Canada</td>
</tr>
<tr>
<td>Magazzini Alimentari Riuniti</td>
<td>Giovanni Rovelli</td>
<td>Italy</td>

</tr>
<tr>
<td>North/South</td>
<td>Simon Crowther</td>
<td>UK</td>
</tr>
<tr>
<td>Paris spcialits</td>
<td>Marie Bertrand</td>
</tr>
</table>
</body>
```



Lists in CSS

```
<!DOCTYPE html>
<html>
<head>
 <style>
 ul.a {
 list-style-position: inside;
 list-style-type: circle;
```

```
/* disc or square types also*/
 }
  ol.b {
 list-style-position: outside;
 list-style-type: upper-roman;
 }
 ul.c {
 list-style-position: inside;
 list-style-image: url("cof.png");
 }
</style>
</head>
<body>
 <p>Example of unordered lists:</p>
 <ul class="a">
 <li>Coffee</li>
 <li>Tea</li>
 <li>Coca Cola</li>
 </ul>
 <p>Example of ordered lists:</p>
 <ol class="b">
 <li>Coffee</li>
 <li>Tea</li>
 <li>Coca Cola</li>
 </ol>
 <p>Example of unordered lists:</p>
 <ul class="c">
 <li>Coffee</li>
 <li>Tea</li>
 <li>Coca Cola</li>
 </ul>
</body>
</html>
```

Element dimensions in CSS:

Dimension of each element on the page can be set using CSS.

You have seen the border that surrounds every box ie. element, the padding that can appear inside each box and the margin that can go around them. Here is, how we can change the dimensions of boxes.

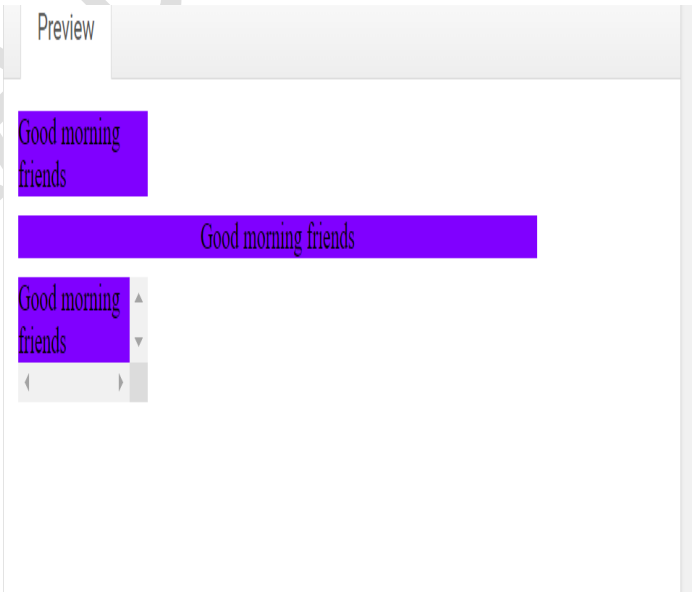
We have the following properties that allow you to control the dimensions of a box.

- The **height** property is used to set the height of a box.
- The **width** property is used to set the width of a box.

- The **line-height** property is used to set the height of a line of text.(no,percentage, a length)..its is space between lines of text.
- The **max-height** property is used to set a maximum height that a box can be.
- The **min-height** property is used to set the minimum height that a box can be.
- The **max-width** property is used to set the maximum width that a box can be.
- The **min-width** property is used to set the minimum width that a box can be.

Example:-

```
<html>
  <head>
  </head>
  <body>
    <p style="width:400px; height:100px;border:1px solid red; padding:50px; margin:20px; line-height:30px;
">
      This paragraph is 400pixels wide and 100 pixels high
    </p>
</body>
</html>
```




```
<!DOCTYPE html>
<html>
<head>
<style type= "text/css" >
DIV  {
  background-color: #8000FF;
margin-bottom: .5em }
```

```
</style>
</head>
<body>
<DIV STYLE= "WIDTH: 20%">Good morning friends </DIV>
<DIV STYLE= "WIDTH: 80%; text-align: center">Good morning friends
</DIV>
<DIV STYLE= "WIDTH: 20%; height: 30%; overflow: scroll">Good morning
friends </DIV>

</body>
</html>
```

Here, <DIV STYLE= "WIDTH: 20%">Good morning friends </DIV>
Shows how to set width of an element on the screen, DIV element should occupy 20% of the screen width.
Width: 10em
Means, that element's width will be equal to 10 times the size of the font.
Overflow property to Scroll value adds scrollbars if the text overflows the boundaries.

DIV tag in CSS:

HTML Tag
It does not affect the content or layout and is used to group HTML elements to be styled with CSS or manipulated with scripts.

The tag can NOT be inside <p> tag, because the paragraph will be broken at the point, where the tag is entered.
To apply styles inside a paragraph use tag, which is used with inline elements.
The div tag is known as Division tag. The div tag is used in HTML to make divisions of content in the web page like (text, images, header, footer, navigation bar, etc). Div tag has both open( ) and closing ( ) tag and it is mandatory to close the tag. The Div is the most usable tag in web development because it helps us to separate out data in the web page and we can create a particular section for particular data or function in the web pages.

- Div tag is Block level tag
- It is a generic container tag
- It is used to the group of various tags of HTML so that sections can be created and style can be applied to them.

The HTML tag is used for defining a section of your document. With the div tag, you can group large sections of HTML elements together and format them with CSS.

The difference between the div tag and the span tag is that the div tag is used with block level elements whilst the span tag is used with inline elements.

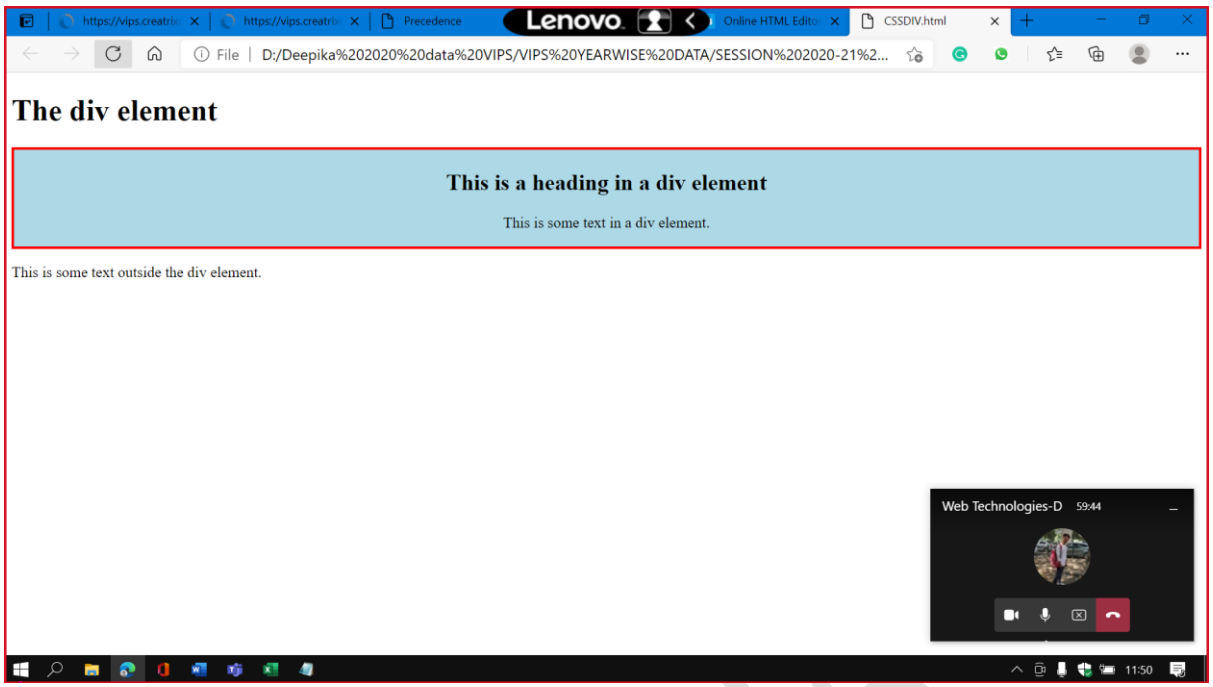
We can create boxes using division.

Syntax

The tag comes in pairs. The content is written between the opening ( ) and closing ( ) tags.

```
<!DOCTYPE html>
<html>
<head>
<style>
.myDiv {
  border-color:red;
  background-color: lightblue;
  text-align: center;
border-style:solid;
border-width:medium;

}
</style>
</head>
<body>

<h1>The div element</h1>

<div class="myDiv">
  <h2>This is a heading in a div element</h2>
  <p>This is some text in a div element.</p>
</div>

<p>This is some text outside the div element.</p>

</body>
</html>
```

Output:-



Multiple Styles Will Cascade into One

Styles can be specified:
- inside an HTML element
- inside the head section of an HTML page
- in an external CSS file

Tip: Even multiple external style sheets can be referenced inside a single HTML document.

Cascading order/Precedence of style rules/Conflicting styles

What style will be used when there is more than one style specified for an HTML element?

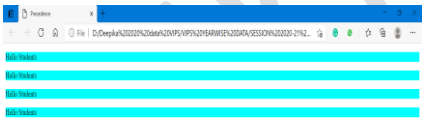
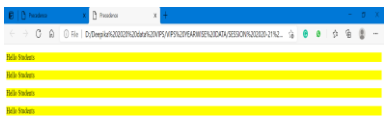
Generally speaking, we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number four has the highest priority:

1. **Browser default……lowest priority**
2. **External style sheet**
3. **Internal style sheet (in the head section)**
4. **Inline style (inside an HTML element)---------highest priority**

So, an inline style (inside an HTML element) 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).

Note: If the link to the external style sheet is placed after the internal style sheet in HTML <head>, the external style sheet will override the internal style sheet!

Internal or embedded CSS	External CSS	External and embedded	Internal,inline,external

<!DOCTYPE html>
<html>
<head>
<style>
p{
background-color: cyan;
}
</style>
</head>
<body>
 Hello Students
 Hello Students
 Hello Students
 Hello Students
</body>
</html>
Output:-
 	
Hello Students
 	
Hello Students
 	
Hello Students
 	
Hello Students
 	
Embedded styles i.e internal are placed within header tag. These override external and browser default styles.
	<!DOCTYPE html>
	<html>
	<head>
	<title>Precedence
	</title>
	<link href="main.css" type="text/css" rel="stylesheet"/>
	</head>
	<body>
	 Hello Students
	 Hello Students
	 Hello Students
	 Hello Students
	</body>
	</html>
	
	File "main.css" content is:-
	p
	{
	background-color:cyan;
	}
	 
	
	External styles are place in css files and override browser default styles. If more than one CSS files linked then the order in which they are placed are considered.
		<!DOCTYPE html>
		<html>
		<head>
		<title>Precedence
		</title>
		<link href="main.css" type="text/css" rel="stylesheet"/>
		
		<style>
		p{
		background-color: yellow;
		}
		</style>
		
		</head>
		<body>
		 Hello Students
		 Hello Students
		 Hello Students
		 Hello Students
		</body>
		</html>
		 
		 
			<!DOCTYPE html>
			<html>
			<head>
			<title>Precedence
			</title>
			<link href= "main.css" type= "text/css" rel= "stylesheet"/>
			
			<style>
			p{
			background-color: yellow;
			}
			</style>
			
			</head>
			<body>
			 Hello Students
			 Hello Students
			 Hello Students
			 Hello Students
			</body>
			</html>
			
			Output :- first line is magenta and others yellow
			
			Inline style overrides other styles as external, internal and browser defaults styles.

Important note:-nearest or closest style rule to the tag or element wins.

CSS precedence level of selectors

There are various types of CSS precedence levels of selectors.

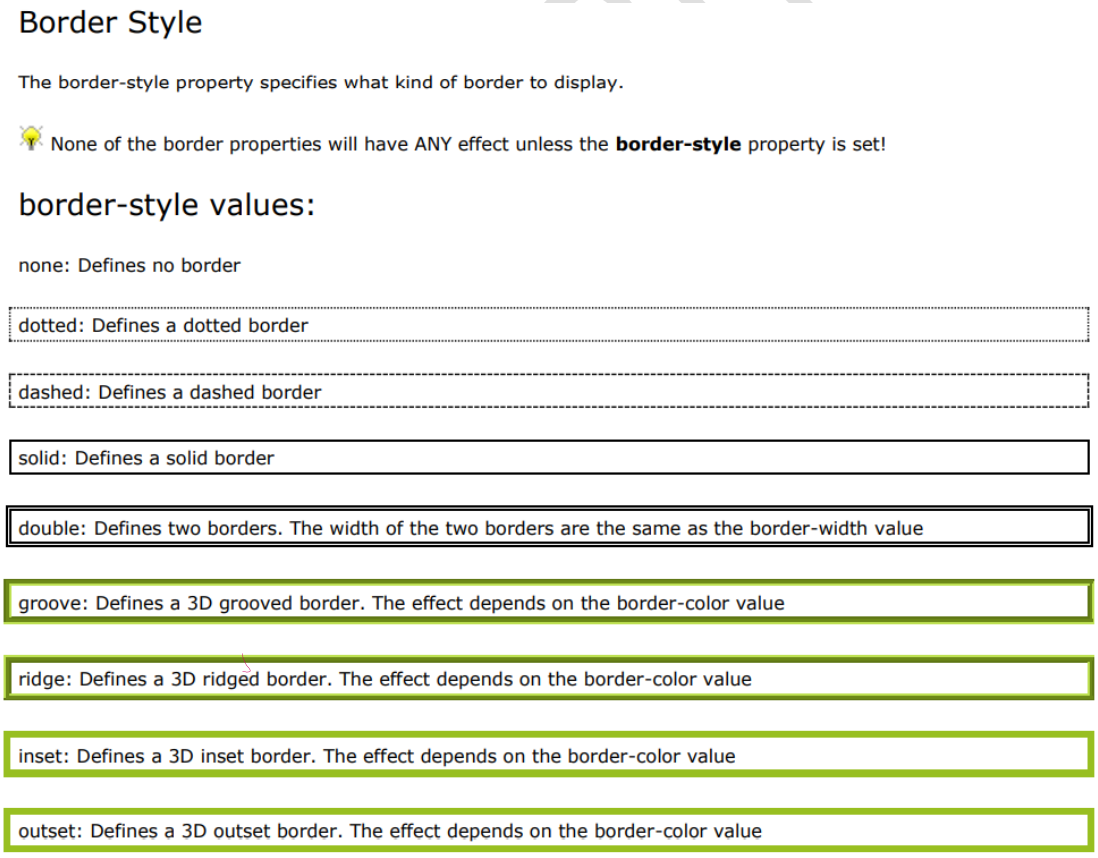
- ID selector
- Class selector
- Type Selector
- Element Selector

ID selector

ID selector can be used to override all the selectors.

CSS Border
CSS Border Properties

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



Border Width:

The border-width property is used to set the width of the border.

The width is set in pixels, or by using one of the three pre-defined values: thin, medium, or thick.
Note: The "border-width" property does not work if it is used alone. Use the "border-style" property to set the
borders first.

Example

```
p.one
{
border-style:solid;
border-width:5px;
}
p.two
{
border-style:solid;
border-width:medium;
}
```

Border Color

The border-color property is used to set the color of the border. The color can be set by:

- name - specify a color name, like "red"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- Hex - specify a hex value, like "#ff0000"

You can also set the border color to "transparent".

Note: The "border-color" property does not work if it is used alone. Use the "border-style" property to set the
borders first.

Example

```
p.one
{
border-style:solid;
border-color:red;
}
p.two
{
border-style:solid;
border-color:#98bf21;
```

}

University question:-

Q. Explain Text alignment, indentation, decoration, spacing and transformation using CSS giving appropriate code.

Q. Explain types of CSS and Link tag.

Text indentation property:-

The text-indent property specifies the indentation of the first line in a text-block.

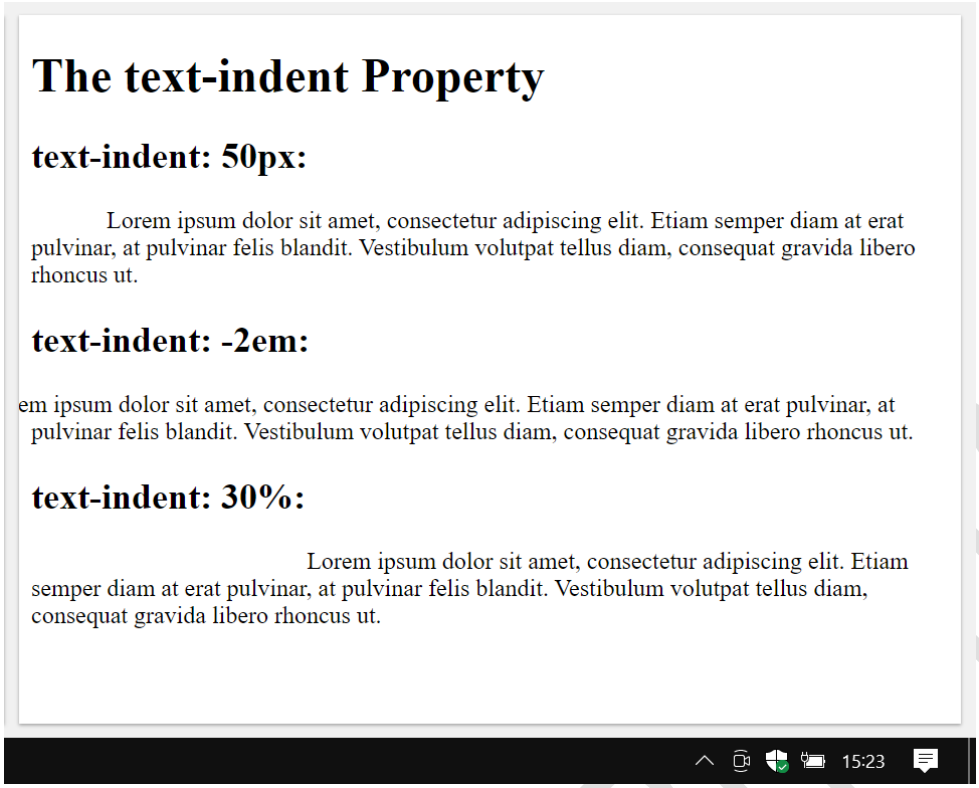
Note: Negative values are allowed. The first line will be indented to the left if the value is negative.

```
<style>
p {
  text-indent: 15%;
}
article {
  background-color: darkgray;
  margin: 2%;
}
```

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>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.</p>
</div>

<h2>text-indent: -2em:</h2>
<div class="b">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.</p>
</div>

<h2>text-indent: 30%:</h2>
<div class="c">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam semper diam at erat pulvinar, at pulvinar felis blandit. Vestibulum volutpat tellus diam, consequat gravida libero rhoncus ut.</p>
</div>

</body>
</html>
```



Text transformation properties:-

Set the Text Cases

The following example demonstrates how to set the cases for a text. Possible values are *none*, *capitalize*, *uppercase*, *lowercase*.

```
<html>
  <head>
  </head>

  <body>
    <p style = "text-transform:capitalize;">
      This will be capitalized
    </p>

    <p style = "text-transform:uppercase;">
      This will be in uppercase
    </p>

    <p style = "text-transform:lowercase;">
      This will be in lowercase
```

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

This will produce following result –

This Will Be Capitalized

THIS WILL BE IN UPPERCASE

this will be in lowercase

BOX MODEL AND TEXT FLOW IN CSS:-

All block-level XHTML elements have a virtual box drawn around them based on what is known as the box model. When the browser renders elements using the box model, the content of each element is surrounded by padding, a border and a margin.

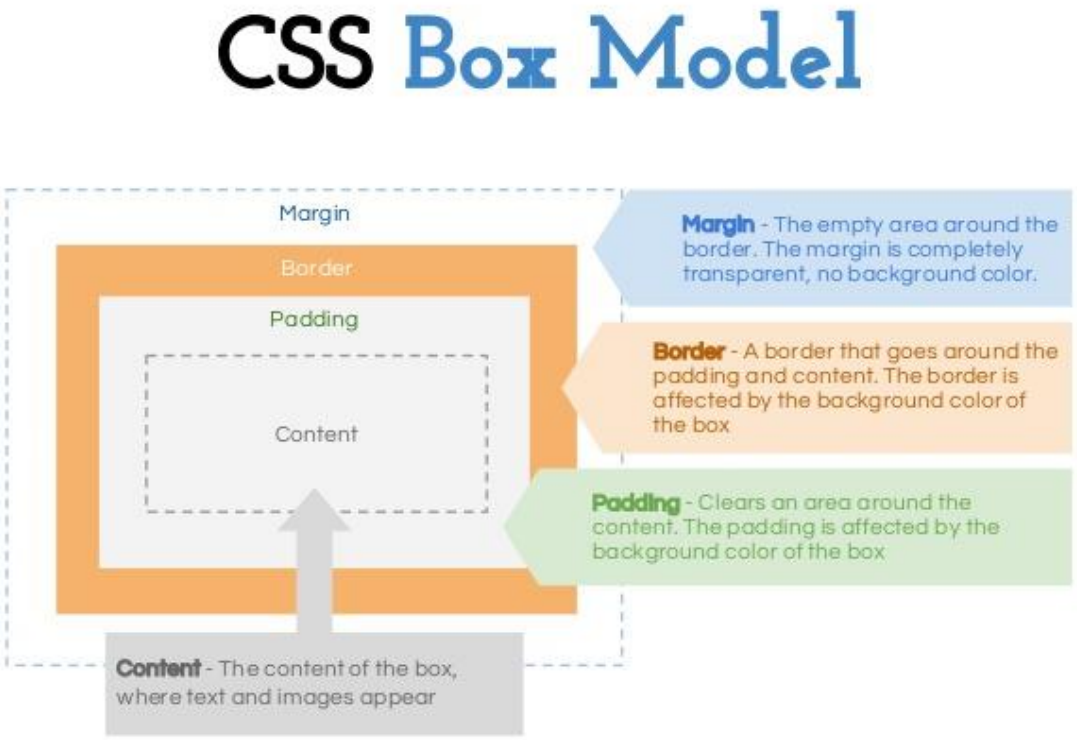
CSS controls the border using three properties: border-width, border-color and border-style. We illustrate these three properties below:

Border Area: It is the area between the box's padding and margin. Its dimensions are given by the width and height of border.

Margin Area: This area consists of space between border and margin. The dimensions of Margin area are the margin-box width and the margin-box height. It is useful to separate the element from its neighbors.

Padding Area: It includes the element's padding. This area is actually the space around the content area and within the border box. Its dimensions are given by the width of the padding-box and the height of the padding-box.

Content Area: This area consists of content like text, image, or other media content. It is bounded by the content edge and its dimensions are given by content box width and height.



- *Property border-width may be set to any valid CSS length (e.g., em, ex, px, etc.) or to the predefined value of thin, medium or thick.*
- *The border-color property sets the color. [Note: This property has different meanings for different style borders.]*
- *The border-style options are none, hidden, dotted, dashed, solid, double, groove, ridge, inset and outset. Borders groove and ridge have opposite effects, as do inset and outset. When border-style is set to none, no border is rendered.*

```
<!DOCTYPE html>
    <head>
        <style>
        .main {
            font-size:32px;
            font-weight:bold;
            text-align:center;
        }
        #box {
```

```
            padding-top:40px;     /*or padding:4px 2px 10px 7px; for top
right bottom left*/
            width: 400px;
            height: 100px;
            border: 50px solid green;
            margin: 50px;
            text-align:center;
            font-size:32px;
            font-weight:bold;
        }
        </style>
    </head>

    <body>
        <div class="main">CSS Box-Model Property</div>
        <div id="box">Web Technology</div>
    </body>
</html>
```

Example2:-

```
<?xml version = "1.0" encoding = "utf-8"?>
 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
 <!-- borders.html -->
 <!-- Borders of block-level elements. -->
 <html xmlns= "http://www.w3.org/1999/xhtml">
 <head>
<title>Borders</title>
 <style type = "text/css">
        div
        {
        text-align: center;
        width: 50%;
        position: relative;
        left: 25%;
        border-width: 4px
        }
.medium { border-width: medium }
.thin { border-width: thin }
.solid { border-style: solid }
.double { border-style: double }
.groove { border-style: groove }
.inset { border-style: inset }
```

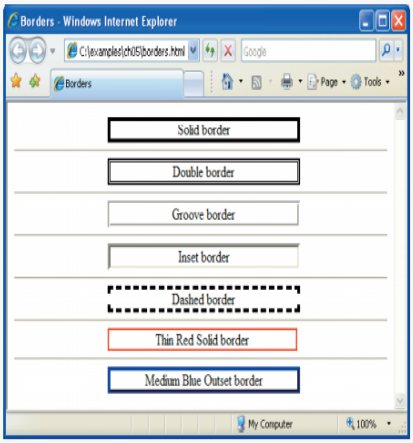


Fig. 5.11 | Borders of block-level elements. (Part 2 of 2.)

```
.outset { border-style: outset }
.dashed { border-style: dashed }
.red { border-color: red }
.blue { border-color: blue
</style>
</head>
<body>
<div class = "solid">Solid border</div><hr />
<div class = "double">Double border</div><hr />
<div class = "groove">Groove border</div><hr />
<div class = "inset">Inset border</div><hr />
<div class = "dashed">Dashed border</div><hr />
<div class = "thin red solid">Thin Red Solid border</div><hr />
<div class = "medium blue outset">Medium Blue Outset border</div>
</body>
</html>
```

- Each border property may be set for an individual side of the box (e.g., border-top style or border-left-color). Note that we assign more than one class to an XHTML element by separating multiple class names with spaces, as shown in lines in above code.
- ***As we have seen with absolute positioning, it is possible to remove elements from the normal flow of text. Floating allows you to move an element to one side of the screen; other content in the document then flows around the floated element.*** Figure below, demonstrates how floats and the box model can be used to control the layout of an entire page.

How to Create Floating Box Effect using HTML and CSS ?

A floating box is a box that can display the content of an element in left or right direction with respect to the content of another element.

For example, you can display a box containing the content of a paragraph in left direction and an image in the right direction. You can float an element by using the float property.

The syntax to use the float property is given as follows :

```
float: left | right | none
```

An example of using the float property is given as follows :

```
img { float: left; }
```

-
- Looking at the XHTML code, we can see that the general structure of this document consists of a header and two main sections. Each section contains a subheading and a paragraph of text.
- Block-level elements (such as divs) render with a line break before and after their content, so the header and two sections will render vertically one on top of another. In the absence of our styles, the subheading divs would also stack vertically on top of the text in the p tags. However, in line 24 we set the float property to right in the class floated, which is applied to the subheadings. This causes each subheading div to float to the right edge of its containing element, while the paragraph of text will flow around it.
- Line 17 assigns a margin of .5em to all paragraph tags. The margin property sets the space between the outside of the border and all other content on the page. In line 21, we assign .2em of padding to the floated divs. The padding property determines the distance between the content inside an element and the inside of the element's border. Margins for individual sides of an element can be specified (lines 22–23) by using the properties margin-top, margin-right, margin-left and margin-bottom. Padding can be specified in the same way, using padding-top, padding-right, padding-left and padding-bottom.

If the margin property has four values:

- margin: 10px 5px 15px 20px;
  - top margin is 10px
  - right margin is 5px
  - bottom margin is 15px
  - left margin is 20px

If the margin property has three values:

- margin: 10px 5px 15px;
  - top margin is 10px
  - right and left margins are 5px
  - bottom margin is 15px

If the margin property has two values:

- margin: 10px 5px;

- top and bottom margins are 10px
- right and left margins are 5px

If the margin property has one value:

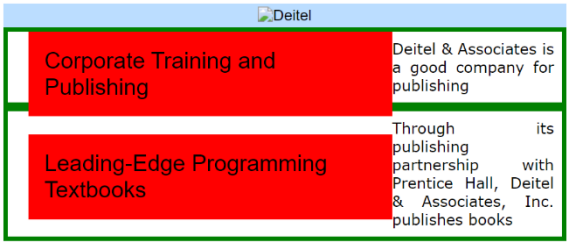
- margin: 10px;
  - all four margins are 10px

Note: Negative values are allowed.

Default is zero value.

To see the effects of margins and padding, try putting the margin and padding properties inside comments and observing the difference.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-
//W3C//DTD XHTML 1.0 Strict//EN"
```




```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!-- floating.html -->
<!-- Floating elements. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
<head>
<title>Flowing Text Around Floating Elements</title>
<style type = "text/css">
      div.heading { background-color: #bbddff;
                    text-align: center;
                    font-family: arial, helvetica, sans-serif;
                    padding: .2em }
       p { text-align: justify;
           font-family: verdana, geneva, sans-serif;
           margin: .5em }
div.floated { background-color: red;
```

```
        font-size: 1.5em;
        font-family: arial, helvetica, sans-serif;
        padding: .7em;
        margin-left: .9em;
        margin-bottom: .8em;
        float: left;
        text-align: left;
        width: 60% }
 div.section { border: 5px solid green }
</style>
</head>
<body>
<div class = "heading"><img src = "https://in.pinterest.com/pin" alt = "Deitel" />
</div>
<div class = "section">
<div class = "floated">Corporate Training and Publishing</div>
<p>Deitel &amp;amp; Associates is a good company for publishing</p>
 </div>
 <div class = "section">
 <div class = "floated">Leading-Edge Programming Textbooks</div>
 <p>Through its publishing  partnership with Prentice Hall, Deitel &amp;amp; Associates, Inc. publishes books</p>
 </div>
</body>
```

In line above, we assign a border to the section boxes using a shorthand declaration of the border properties. CSS allows shorthand assignments of borders to allow you to define all three border properties in one line. The syntax for this shorthand is border: <width> <style> <color>. Our border is five pixel thick, solid, and the same color as the background-color property of the heading div (line 11). This allows the border to blend with the header and makes the page appear as one box with a line dividing its sections.

****