CSS
Cascading **S**tyle **S**heet

Introduction

- CSS stands for Cascading Style Sheets.
- CSS is a standard style sheet language used for describing the presentation (i.e. the layout and formatting) of the web pages.
- CSS was designed to enable the separation of presentation and content.
- Web designers can move the formatting information of the web pages to a separate style sheet which results in considerably simpler HTML markup, and better maintainability.

What can you do with CSS

- You can easily apply same style rules on multiple elements.
- You can control the presentation of multiple pages of a website with a single style sheet.
- You can present the same page differently on different devices.
- You can style dynamic states of elements such as hover, focus, etc. that isn't possible otherwise.
- You can change the position of an element on a web page without changing the markup.
- You can alter the display of existing HTML elements.
- You can transform elements like scale, rotate, skew, etc. in 2D or 3D space.
- You can create animations and transitions effects without using any JavaScript.
- You can create print friendly version of your web pages.

Advantages of Using CSS

- CSS Save Lots of Time
- Easy Maintenance
- Pages Load Faster
- Superior Styles to HTML
- Multiple Device Compatibility

Levels of Style sheet

- **Including CSS in HTML Documents/ Levels of CSS:**
  - **Inline styles**:
    - Using the style attribute in the HTML start tag.
  - **Embedded styles/ Document level style sheet**:
    - Using the <style> element in the **head** section of a document.
  - **External style sheets**:
    - Using the <link> element, pointing to an external CSS file.
    - <link rel="stylesheet" href="style.css">

Levels of Style sheet – Inline Style Sheet

- Inline styles are used to apply the unique style rules to an element by putting the CSS rules directly into the start tag.
- It can be attached to an element using the style attribute.
- The style attribute includes a series of CSS property and value pairs.
- Each "property: value" pair is separated by a semicolon (;). But it needs to be all in one line i.e. no line break after the semicolon,

Levels of Style sheet – Inline Style Sheet

- **Inline Style sheet:**
- The style attribute is used to apply styles to an element.

```
Tag
{
Property: value
}
```

Eg: -

- If we want to use more than one property then we have to use seperator such as semicolon.
- Eg:

```
<h1 style="font-size: 140%; color: #ff0000;">The Style Attribute</h1>
```

- Output

The Style Attribute

Levels of Style sheet – Inline Style Sheet

XHTML Document [InlineStyle.html]

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Inline Cascading Style Sheet</title>
    </head>
    <body>
        <p>This is simple text</p>
        <p style="font-size: 30pt ;font-family:Script">This text is different </p>
        <p style="font-size: 40pt ;color:#ff0000">This text is colored.</p>
    </body>
</html>
```

Levels of Style sheet – Inline Style Sheet

```
<html>
<head>
<title>TAG index</title>
</head>
<body>

<div style="background-color: #c0c0c0;">

<h1 style="color: red;">Heading</h1>
<p style="line-height: 120%;">Short paragraph.</p>
<img src="example.jpg" alt="Photo" style="border: 1px green solid;">

</div>

</body>
</html>
```

Advantages and Disadvantages of Inline Style Sheet

- Advantages:
  - Apply uniform style on tags for the whole document
- Disadvantages:
  - Actual contents of the web page are mixed with the presentation

Levels of Style sheet – Document Level Style Sheet

- Embedded or internal style sheets only affect the document they are embedded in.
- Embedded style sheets are defined in the <head> section of an HTML document using the <style> element.
- You can define any number of <style> elements in an HTML document but they must appear between the <head> and </head> tags.

Levels of Style sheet – Document Level Style Sheet

```
<!DOCTYPE html>
<html lang="en">
<head>
   <title>My HTML Document</title>
   <style>
      body { background-color: YellowGreen; }
      p { color: #fff; }
   </style>
</head>
<body>
   <h1>This is a heading</h1>
   <p>This is a paragraph of text.</p>
</body>
</html>
```

Levels of Style sheet – Document Level Style Sheet

- Advantages:
  - Helps to decide the layout of the web page.
  - Useful when we want to apply the unique style sheet for the web page.
- Disadvantages:
  - No use when we want to apply style to more than one documents at a time

Ways to include CSS

Way 1: Document level

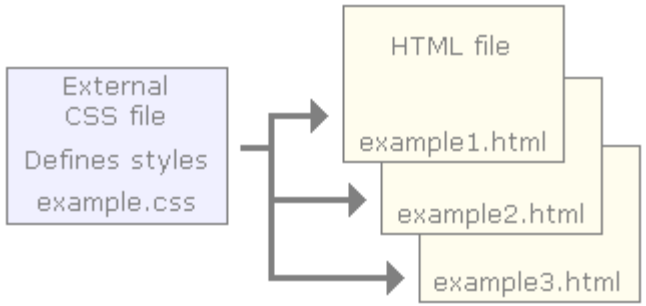
```
<html>
<head>
    <style type="text/css">
    <!--
    h3 { font-family: arial; font-style: italic; color:
    green }
    -->
  </style>
</head>

<body>
<h3>this is a green, italic, arial h3 header.</h3>
<p>
<h3>so is this.</h3>
</body>
</html>
```

Way 2: Inline

```
<html>
<body>
<h3 style="font-family: arial;
          font-style: italic;
          color: green">
this is a green, italic, arial h3 header.
</h3>
<p>
<h3>this is an h3 header, but it's not
green, italic, or arial.</h3>
</body>
</html>
```

Levels of Style sheet – External Level Style Sheet

- An external style sheet is ideal when the style is applied to many pages of the website.
- An external style sheet holds all the style rules in a separate document that you can link from any HTML file on your site.
- External style sheets are the most flexible because with an external style sheet, you can change the look of an entire website by changing just one file.



Levels of Style sheet – External Level Style Sheet

```
/* CSS file inside */

h1 { color: red; }
p { line-height: 120%; }
#container { background-color: #c0c0c0; }
.photo { border: 1px green solid; }
```

```
<html>
<head>
<title>TAG index</title>
<link rel="stylesheet" type="text/css" href="example.css">
</head>
<body>



</body>
</html>
```

```
<html>
<head>
<title>TAG index</title>
<link rel="stylesheet" type="text/css" href="body.css">
<link rel="stylesheet" type="text/css" href="list.css">
</head>
<body>



</body>
</html>
```

Levels of Style sheet – External Level Style Sheet

- <link rel="stylesheet" href="Style.css">
  - **Link** tells the browser some file must be linked to the page
  - **rel= "stylesheet"** tells the browser that this linked thing is a style sheet.
  - **href=" "** denotes the path name of style sheet file
  - **type= "text/css**" tells the browser that what it is reading is text which is affected by the CSS.

Levels of Style sheet – External Level Style Sheet

- **Style.css**

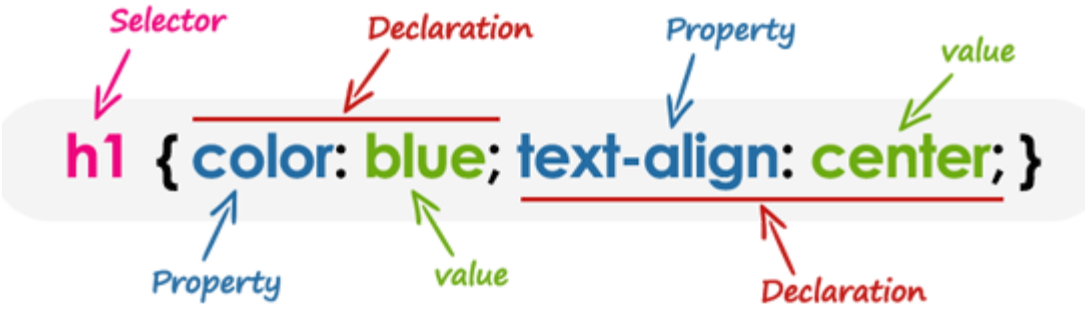
```
body
{
   background: lightyellow;
   font: 18px Arial, sans-serif;
}
h1
{
   color: orange;
}
```

- **My.html**

```
<html lang="en">
<head>
   <title>My Document</title>
   <link rel="stylesheet" href="Style.css">
</head>
<body>
   <h1>This is a heading</h1>
   <p>This is a paragraph of text.</p>
</body>
</html>
```

CSS basic Syntax and Structure

- A CSS stylesheet consists of a set of rules that are interpreted by the web browser and then applied to the corresponding elements such as paragraphs, headings, etc. in the document.
- A CSS rule have two main parts, a selector and one or more declarations:



CSS Selector

- **Selector:**
- A CSS selector is a pattern to match the elements on a web page.
- The style rules associated with that selector will be applied to the elements that match the selector pattern.
- Selectors are one of the most important aspects of CSS as they allow you to target specific elements on your web page in various ways so that they can be styled.

CSS Selectors: Simple Selector

- Selector { Property: Value }
  - The property and value are separated by a colon.

    color: red
  - **Multiple properties**: If you specify two or more properties for the selector, separate by a semi-colon.

```
h1 { padding: 3px; background-color: #f9f9f9; color: red; }

p {
margin: 10px;
color: green;
}
```

  - **Grouping selectors**: You can specify the same properties for multiple selectors by separating each selector with a comma.

```
h1, h2, h3 { color: red; }

#header, #footer {
border: 1px green solid;
padding: 10px;
}
```

CSS Selector – Class selector

- **Generic and Class Selectors**
- The class selectors can be used to select any HTML element that has a class attribute. All the elements having that class will be formatted according to the defined rule.
- The class selector is defined with a period sign (.) immediately followed by the class value.

CSS Selector – Class Selector

- **Generic selector:** The below style rules renders the text in blue of every element in the document that has class attribute set to blue.

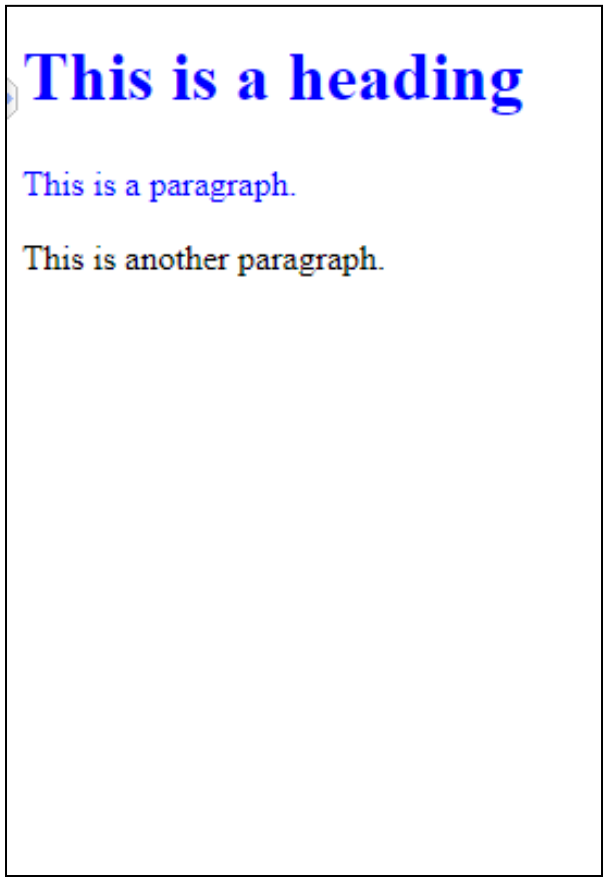
```
.blue {
    color: blue;
}
```

- **Class Selector:** The style rule inside the selector p.blue renders the text in blue of only those <p> elements that has class attribute set to blue, and has no effect on other paragraphs.

```
p.blue {
    color: blue;
}
```

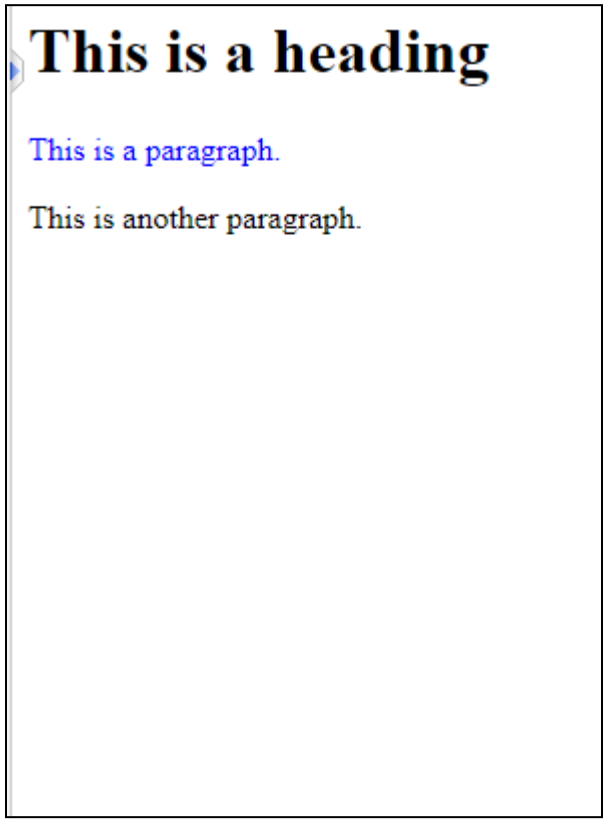
CSS Selector – Generic Selector

```
<html >
<head>
        <title>CSS class selector</title>
         <style>
                .blue {
                 color: #0000ff;
                }
          </style>
</head>

<body>
  <h1 class="blue">This is a heading</h1>
  <p class="blue">This is a paragraph.</p>
  <p>This is another paragraph.</p>
</body>

</html>
```

- 

CSS Selector – Class Selector

```
<html >
<head>
            <title>CSS class selector</title>
             <style>
                        p .blue {
                          color: #0000ff;
                         }
              </style>
</head>

<body>
  <h1 class="blue">This is a heading</h1>
  <p class="blue">This is a paragraph.</p>
  <p>This is another paragraph.</p>
</body>

</html>
```



CSS Selector – Generic Selector

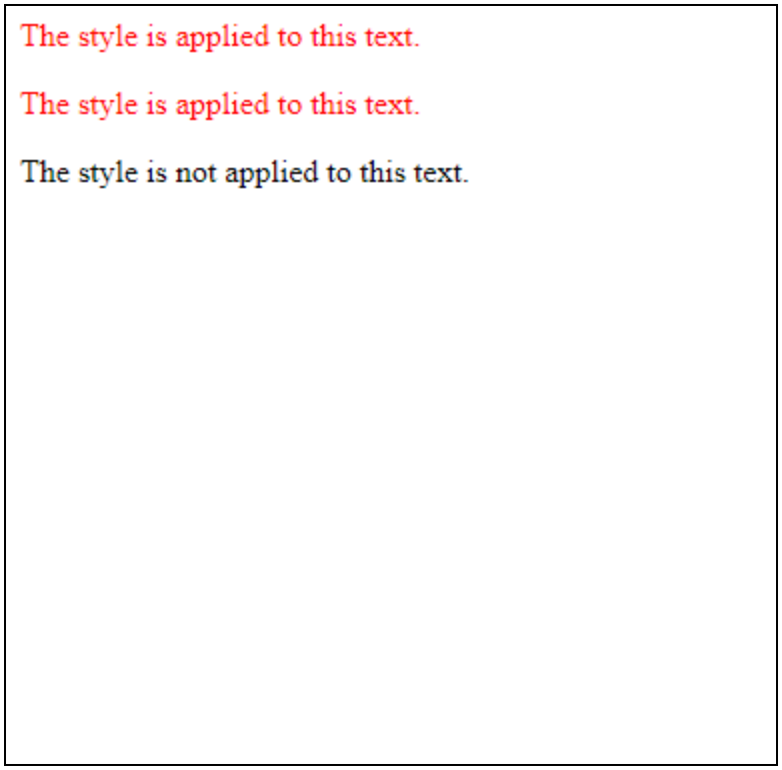
- **Multiple classes**

```
<html>
<head>
        <title>TAG index</title>
        <style type="text/css">
                        .example1 { color: red; }
                        .example2 { text-decoration: underline; }
        </style>
</head>
<body>

<p class="example1">Only the color style.</p>

<p class="example2">Only the decoration style.</p>

<p class="example1 example2">The color and decoration styles.</p>

</body>
</html>
```

CSS Selector – Generic Selector

- **Nesting classes:**

```
<html>
<head>
<title>TAG index</title>
        <style type="text/css">
                .example p { color: red; }
        </style>
</head>
<body>

<div class="example">
        <p>The style is applied to this text.</p>
        <p>The style is applied to this text.</p>
</div>

<p>The style is not applied to this text.</p>

</body>
</html>
```

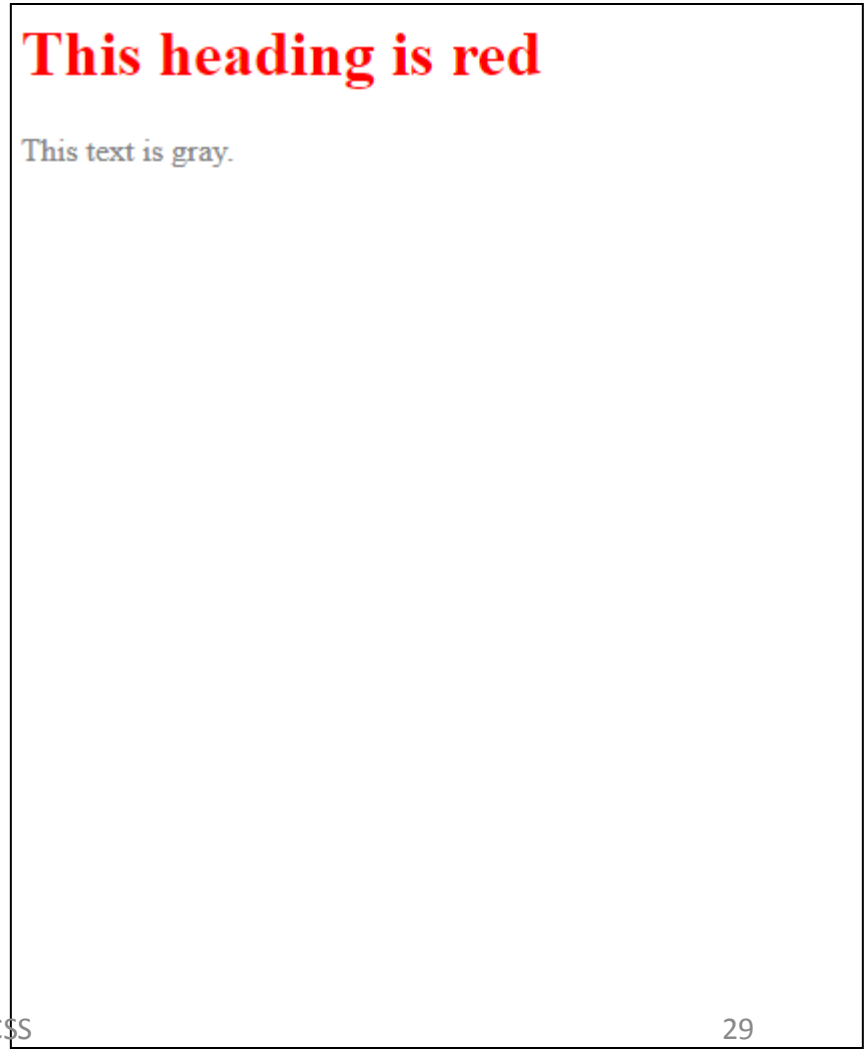


CSS Selector – id Selector

- You can apply styles to one identified element by using the ID selector.
- An ID selector is a name preceded by a hash character (#).
- #example { color: red; }
  - In the above example, the style is applied to one identified element that have the ID "example".
- p#example { color: red; }
  - The ID can also be used with a particular element type.

CSS Selector – id Selector

```
<html>
<head>
        <title>TAG index</title>

        <style type="text/css">

                #example1 { color: red; }
                #example2 { color: gray; }

        </style>

</head>
<body>

        <h1 id="example1">This heading is red</h1>
        <p id="example2">This text is gray.</p>

</body>
</html>
```



CSS Selector – id Selector

- **Nesting**

```
#example p { color: red; }
```

```
<html>
<head>
        <title>TAG index</title>
        <style type="text/css">
                #example p { color: red; }
        </style>
</head>
<body>
<div id="example">
        <p>The style is applied to this text.</p>
        <p>The style is applied to this text.</p>
</div>
<p>The style is not applied to this text.</p>

</body>
</html>
```

The style is applied to this text.

The style is applied to this text.

The style is not applied to this text.

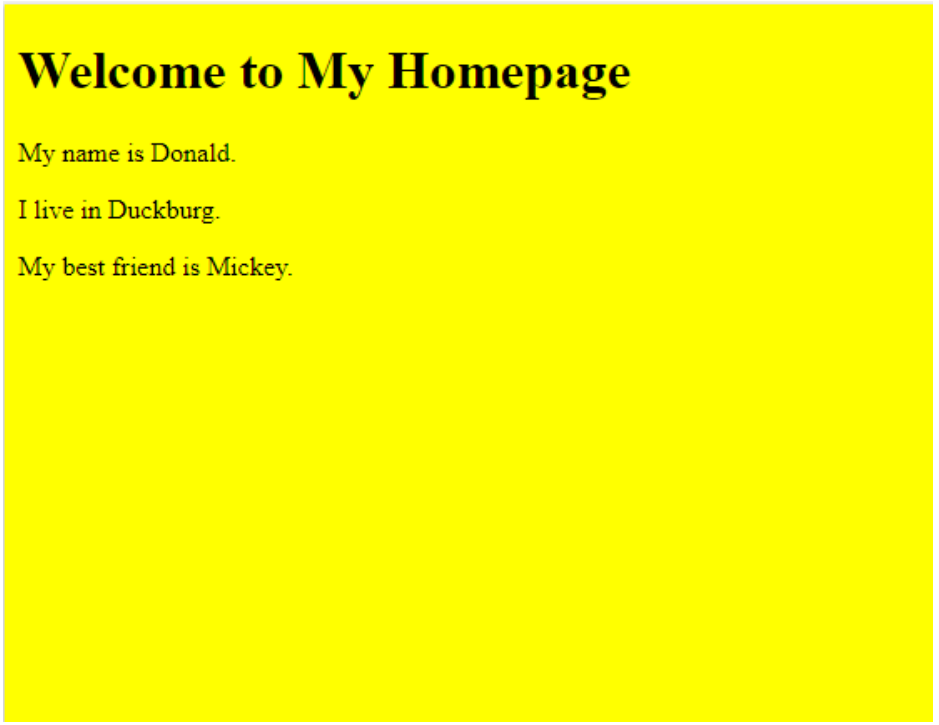
CSS Selector – Universal Selector

- **Universal Selector**
- The universal selector, denoted by an asterisk (*), matches every single element on the page.
- Applied to all the elements in the document.
- Eg:

```
* {
   css declarations;
}
```

CSS Selector – Universal Selector

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

<h1>Welcome to My Homepage</h1>

<div class="intro">
  <p id="firstname">My name is Donald.</p>
  <p id="hometown">I live in Duckburg.</p>
</div>

<p>My best friend is Mickey.</p>

</body>
</html>
```



CSS Selector – Pseudo Class Selectors

- pseudo-class is used to define a special state of an element.
- For example, it can be used to:
  - Style an element when a user mouses over it
  - Style visited and unvisited links differently
  - Style an element when it gets focus
- **Syntax**

```
selector:pseudo-class {property: value}
```

- CSS classes can also be used with pseudo-classes –

```
selector.class:pseudo-class {property: value}
```

CSS Selector – Pseudo Class Selectors

- **Anchor Pseudo-classes**

- **/* unvisited link */**
```
a:link {
  color: #FF0000;
}
```

- **/* visited link */**
```
a:visited {
  color: #00FF00;
}
```

- **/* mouse over link */**
```
a:hover {
  color: #FF00FF;
}
```

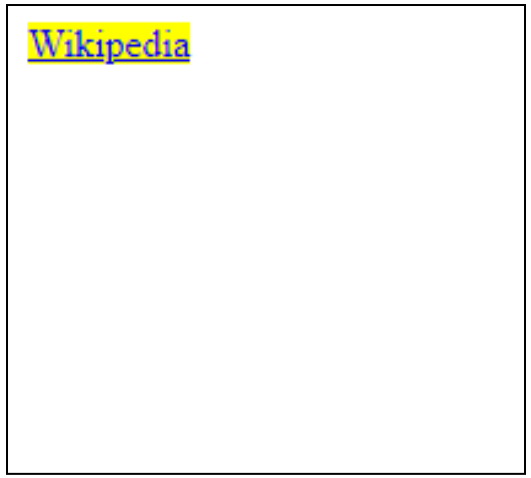
- **/* selected link */**
```
a:active {
  color: #0000FF;
}
```

CSS Selector – Pseudo Class Selectors

```
<!DOCTYPE html>
<html>
<head>
	<style>
	a:hover {
	  background-color: yellow;
	}
	</style>
</head>

<body>

<a href="https://www.schools.com">schools.com</a>
<br>
<a href="https://www.colleges.org">colledges.org</a>

</body>
</html>
```

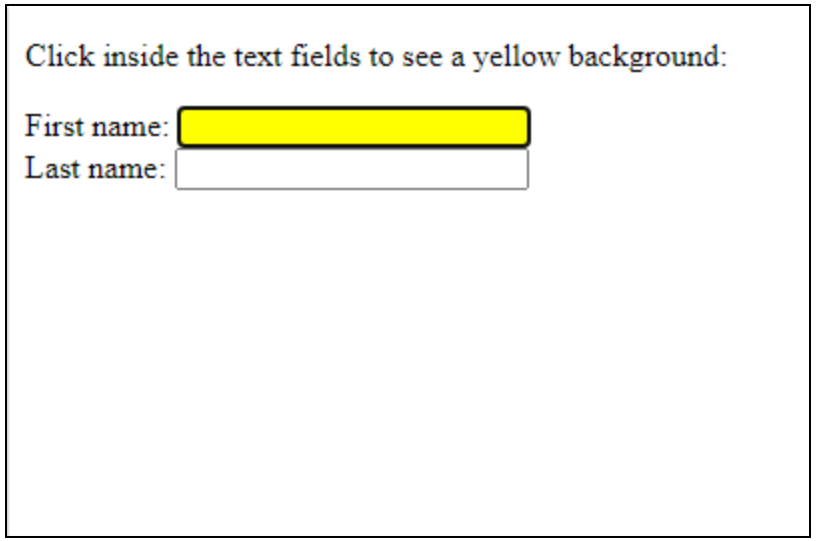
CSS Selector – Pseudo Class Selectors

```
<!DOCTYPE html>
<html>
<head>
	<style>
	a:link {
	  background-color: yellow;
	}
	</style>
</head>
<body>

<a href="http://www.wikipedia.org">Wikipedia</a>


</body>
</html>
```



CSS Selector – Pseudo Class Selectors

```
<!DOCTYPE html>
<html>
<head>
	<style>
	input:focus
	{
	  background-color: yellow;
	}
	</style>
</head>
<body>

<p>Click inside the text fields to see a yellow background:</p>

<form>
  First name: <input type="text" name="firstname"><br>
  Last name: <input type="text" name="lastname">
</form>

</body>
</html>
```



Categories of properties

- There are 7 categories of properties:
  1. Fonts
  2. List
  3. Alignment of text
  4. Color
  5. Margins
  6. Background
  7. borders

Background Properties

- 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.

Background Color

- Sets the background color of an element
- Value:
  - RGB
  - HEX
  - name
  - Transparent
- The W3C CSS standard Supports only 16 names:
  - Aqua, black, blue, fuchia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white and yellow.
- Eg:
  - body

```
{
background-color: gray;
}
```

Background Images

- Sets an image in the background
- Value: URL or none
- **Syntax:**

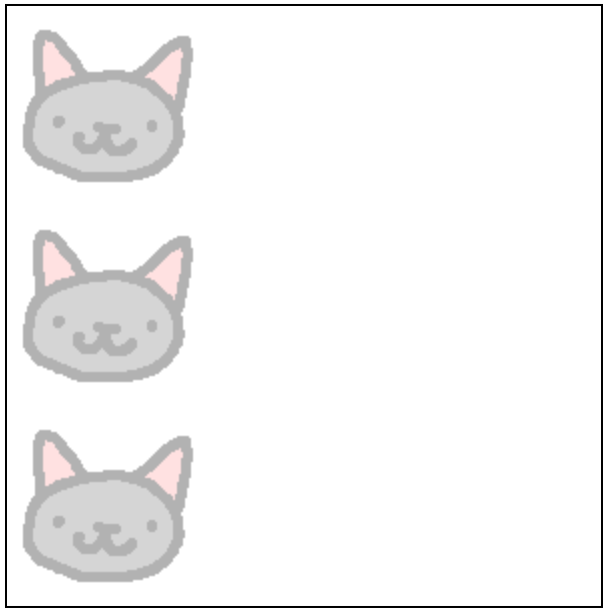
```
body
{
background-image:  url(image/back.gif);
}
```

Background- repeat

- Example:

```
body
 {
background-image: url(image/back.gif);
background-repeat: repeat-y;
}
```

background-repeat	repeat	the image is repeated both horizontally and vertically (**default**)
	repeat-x	the image is only repeated horizontally
	repeat-y	the image is only repeated vertically
	no-repeat	the image is displayed only once



Background-position

- **Example:**
- body

```
 {
background-image: url(image/back.gif);
background-repeat:  no-repeat;
background-position:  center;
}
```

Horizontal position	left	center	right
Vertical position	top	center	bottom

- **For example :**
  - background-position: right bottom; : The right bottom position.
  - background-position: center center; : The center position.
- **If you only specify one keyword, the other value becomes "center".**
- **For example :**
  - background-position: right; : The right center position.
  - background-position: bottom; : The bottom center position.
  - background-position: center; : The center position.

Background-position

- **Position** (by %, pixels, or other units)
- The first value is the horizontal position (The distance from the left side) and the second value is the vertical position (The distance from the top side).
  - For example :
    background-position: 100px 40px; : The position to 100px from the left and 40px from the top.
    background-position: 30% 60%; : The position to 30% from the left and 60% from the top.
    background-position: 100px 60%; : The position to 100px from the left and 60% from the top.
- If you only specify one value, the other value becomes "50%".
  - For example :
    background-position: 100px; : The position to 100px from the left and 50% from the top.
    background-position: 50%; : The position to 50% from the left and 50% from the top.
- If only specify one value, other value will be 50%.
- Default is 0% 0%

Background-attachment

- The background-attachment property specifies whether a background image is fixed or scrolls.

Property	Value	Explanation
background-attachment	fixed	specifies the fixed
	scroll	specifies the scrolling (**default**)

Background

- The background property is a shorthand property for setting all background properties.
- Eg:

```
body
{
background: #87ceeb url(image/back.gif) repeat-y fixed right top;
}
```

Property	Value	Explanation
background	each value	sets color, image, repeat, attachment, and position

Property	Description	Values
Background	Sets all background properties in one declaration	as RGB
Background-attachment	Sets whether image moves with page when scrolled	scroll or fixed
background-color	Sets the background color of an element	RGB, hex, name or transparent
background-image	Sets an image in the background	URL or none
Background-position	Sets the starting position of an image in the background	top left top center top right center left center center center right bottom left bottom center bottom right x-% y-% x-pos y-pos
background-repeat	Sets the repetition of an image used in the background	repeat repeat-x repeat-y no-repeat

→ If only specify one value, other value will be 50%. Default 0% 0%.

Manipulating Text

- **Text Properties:**
  - 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 **line-height** property specifies the height of a text line.

Text color

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

   <body>
      <p style = "color:red;">
         This text will be written in red.
      </p>
   </body>
</html>
```

It will produce the following result –

This text will be written in red.

Text Direction

- Syntax:
  - Direction: value
- Values: ltr or rtl

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

   <body>
      <p style = "direction:rtl;">
         This text will be rendered from right to left
      </p>
   </body>
</html>
```

It will produce the following result –

This text will be renedered from right to left

Space between Characters

- The letter-spacing property is used to set extra spacing between the characters of text.
- Possible values are *normal or a number specifying space.*.

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

   <body>
      <p style = "letter-spacing:5px;">
         This text is having space between letters.
      </p>
   </body>
</html>
```

It will produce the following result –

```
This text is having space between letters.
```

Spacing between words

- The word-spacing property is used to specify additional spacing between the words.
- This property can accept a length value in pixels, ems, etc. Negative values are also allowed.

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

   <body>
      <p style = "word-spacing:5px;">
         This text is having space between words.
      </p>
   </body>
</html>
```

This will produce following result –

This text is having space between words.

Text Indent

- The text-indent property is used to set the indentation of the first line of text within a block of text.
- It is typically done by inserting the empty space before the first line of text.
- Possible values are *% or a number specifying indent space*.

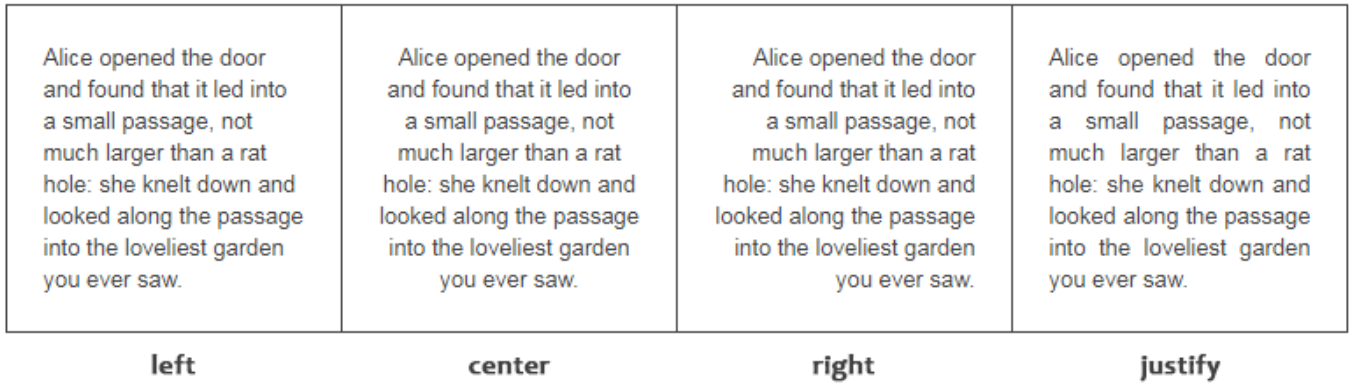
```
<html>
   <head>
   </head>

   <body>
      <p style = "text-indent:1cm;">
         This text will have first line indented by 1cm and this
         its actual position this is done by CSS text-indent prop
      </p>
   </body>
</html>
```

It will produce the following result –

This text will have first line indented by 1cm and this line will remain at its
this is done by CSS text-indent property.

Text Alignment

- Possible **values** are *left, right, center, justify*.




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

   <body>
      <p style = "text-align:right;">
         This will be right aligned.
      </p>

      <p style = "text-align:center;">
         This will be center aligned.
      </p>

      <p style = "text-align:left;">
         This will be left aligned.
      </p>
   </body>
</html>
```

This will produce following result −

This will be right aligned.

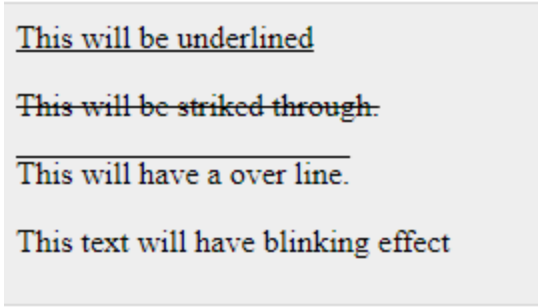
This will be center aligned.

This will be left aligned.

Text-decoration: Decorating the text

Property	Value	Explanation
text-decoration	underline	horizontal line under the text
	overline	horizontal line over the text
	line-through	horizontal line through the text
	blink	blinking text
	none	normal text (**default**)

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

   <body>
      <p style = "text-decoration:underline;">
         This will be underlined
      </p>

      <p style = "text-decoration:line-through;">
         This will be striked through.
      </p>

      <p style = "text-decoration:overline;">
         This will have a over line.
      </p>

      <p style = "text-decoration:blink;">
         This text will have blinking effect
      </p>
   </body>
</html>
```



Set the Text Cases: Text transformation

- The text-transform property controls whether or not the text is capitalized.

Property	Value	Explanation
text-transform	capitalize	capitalized the first letter of each word
	uppercase	using only uppercase letters
	lowercase	using only lowercase letters
	none	normal text (**default**)

text-transform: Eg

```
<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

Font

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

Font-family

- The font is specified by the font family names or generic family names.

Generic Font Name	Example
sans-serif	Arial, Helvetica, Futura
cursive	Zapf-chancery
fantasy	Critter, Cottonwood
monospace	Courier, Prestige
serif	Times New Roman,Garamond

- **The list of the fonts**
  - If you set two or more fonts, separate by a comma.
    - font-family: "Font1","Font2","Font3";
  - When Font1 cannot be used in the user's computer, Font2 is used.
- **Use of quotation marks**
  - If a font name contains space, it must be placed within quotation marks.
  - font-family: "Century Gothic";
    font-family: Century;

Font-family: Eg

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

   <body>
      <p style = "font-family:georgia,garamond,serif;">
         This text is rendered in either georgia, garamond, or the
         default serif font depending on which font  you have at your system.
      </p>
   </body>
</html>
```

This will produce following result −

This text is rendered in either georgia, garamond, or the default serif font depending on which font you have at your system.

Font-style: Set the Font Style

Property	Value	Explanation
font-style	italic	italic font
	oblique	oblique font
	normal	normal font (**default**)

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example1 { font-style: italic; }
#example2 { font-style: oblique; }
#example3 { font-style: normal; }

</style>

</head>
<body>

<p>The font style is <span id="example1">italic</span></p>
<p>The font style is <span id="example2">oblique</span></p>
<p><em>The font style is <span id="example3">normal</span></em></p>


</body>
</html>
```

Output

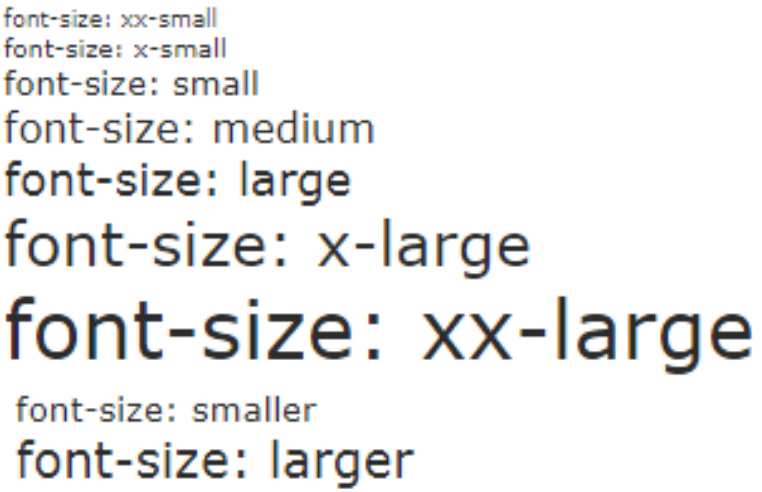
The font style is *italic*

The font style is *oblique*

The font style is normal

Font-size

- **The font size keywords: 7**
- The seven sizes :
  - [small] xx-small, x-small, small,
  - medium,
  - large, x-large, xx-large [large]
- The default is "medium".
- The keyword medium is equivalent to the browsers default font-size, which is normally 16px.
  - Likewise, xx-small is the equivalent of 9 pixels,
  - x-small is 10 pixels,
  - small is 13 pixels,
  - large is 18 pixels,
  - x-large is 24 pixels,
  - xx-large is 32 pixels.



Font-size: Eg

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

p { font-size: 100%; }

#example1 { font-size: 200%; }
#example2 { font-size: 1.5em; }
#example3 { font-size: x-small; }
#example4 { font-size: larger; }

</style>

</head>
<body>

<p>The font size is <span id="example1">200%</span></p>
<p>The font size is <span id="example2">1.5em</span></p>
<p>The font size is <span id="example3">x-small</span></p>
<p>The font size is <span id="example4">larger</span></p>

</body>
</html>
```

Output

The font size is 200%

The font size is 1.5em

The font size is x-small

The font size is larger

Font-variants

Property	Value	Explanation
font-variant	small-caps	The lowercase letters is displayed in small uppercase.
	normal	normal font (**default**)

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example { font-variant: small-caps; }

</style>

</head>
<body>

<p>TAG index html TAG</p>
<p id="example">TAG index html TAG</p>

</body>
</html>
```

Output

TAG index html TAG

TAG INDEX HTML TAG

Font-weight

Property	Value	Explanation
font-weight	bold	bold font
	normal	normal font (**default**)

- The font weight can also be specified by using the following keywords or numbers.
- The nine weight numbers :
  [light] 100, 200, 300, 400, 500, 600, 700, 800, 900 [bold]
- The default is "400".
- Eg:
  - font-weight: 100
    font-weight: 900
  - font-weight: lighter
    font-weight: bolder

Font-weight Eg

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example1 { font-weight: bold; }
#example2 { font-weight: normal; }

</style>

</head>
<body>

<p>The font weight is <span id="example1">bold</span></p>
<p><strong>The font weight is <span id="example2">normal</span></strong></p>

</body>
</html>
```

Output

The font weight is **bold**

The font weight is normal

Line- height

- You can set the distance between lines by using this property.

Property	Value	Explanation
line-height	number, length, %, or normal	the height of a text line

Line-height: Eg

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example { line-height: 200%; }

</style>

</head>
<body>

<p>
This is example text.<br>
some text some text some text some text<br>
some text some text some text some text<br>
some text some text some text some text
</p>

<p id="example">
This is example text.<br>
some text some text some text some text<br>
some text some text some text some text<br>
some text some text some text some text
</p>

</body>
</html>
```

Output

This is example text.
some text some text some text some text
some text some text some text some text
some text some text some text some text

This is example text.

some text some text some text some text

some text some text some text some text

some text some text some text some text

Font

Property	Value	Explanation
font	each value	sets style, small-caps, weight, size, line height, & font family

- font: italic normal bold 80% / 150% "Century Gothic“ ;

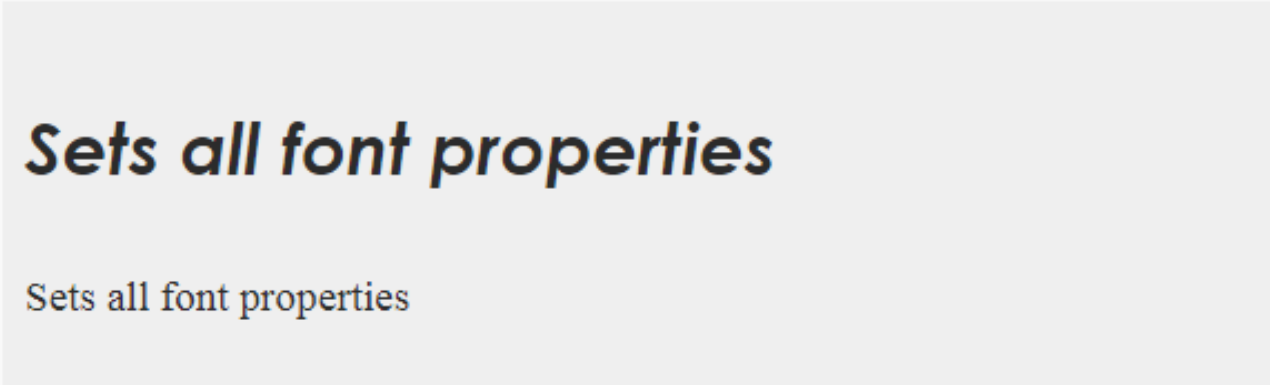
font:	italic	normal	bold	80%	/	150%	"Century Gothic"	;
	font-style	font-variant	font-weight	font-size		line-height	font-family	

- The values other than the font-size and font family properties can be omitted.
  - font: 80% "Century Gothic";

Font

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example1 { font: italic normal bold 200%/150% "Century Gothic",sans-serif; }
#example2 { font: 120% serif; }

</style>

</head>
<body>

<p id="example1">Sets all font properties</p>
<p id="example2">Sets all font properties</p>

</body>
</html>
```

Output



List properties

- These CSS list properties typically allow you to:
  - Control the shape or appearance of the marker.
  - Specify an image for the marker rather than a bullet point or number.
  - Set the distance between a marker and the text in the list.
  - Specify whether the marker would appear inside or outside of the box containing the list items.

List properties

- We have the following CSS properties, which can be used to control lists –
  - The **list-style-type** allows you to control the shape or appearance of the marker.
  - The **list-style-position** specifies whether a long point that wraps to a second line should align with the first line or start underneath the start of the marker.
  - The **list-style-image** specifies an image for the marker rather than a bullet point or number.
  - The **list-style** serves as shorthand for the preceding properties.

List-style-type

- The list-style-type property sets the type of the list item marker.
- This property can apply to the UL, OL, and LI element.

- ul

```
{
        list-style-type: square;
 }
```

- ol

```
{
        list-style-type: upper-alpha;
 }
```

Property	Value	Example
list-style-type	IE Fx O	
	none	First list item Second list item Third list item
	disc	• First list item • Second list item • Third list item
	circle	◦ First list item ◦ Second list item ◦ Third list item
	square	▪ First list item ▪ Second list item ▪ Third list item
	decimal	1. First list item 2. Second list item 3. Third list item
	upper-alpha	A. First list item B. Second list item C. Third list item
	lower-alpha	a. First list item b. Second list item c. Third list item
	upper-roman	I. First list item II. Second list item III. Third list item
	lower-roman	i. First list item ii. Second list item iii. Third list item

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

ul { list-style-type: square; }
ol { list-style-type: upper-alpha; }

</style>

</head>
<body>

<ul>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ul>

<ol>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ol>

</body>
</html>
```

Output

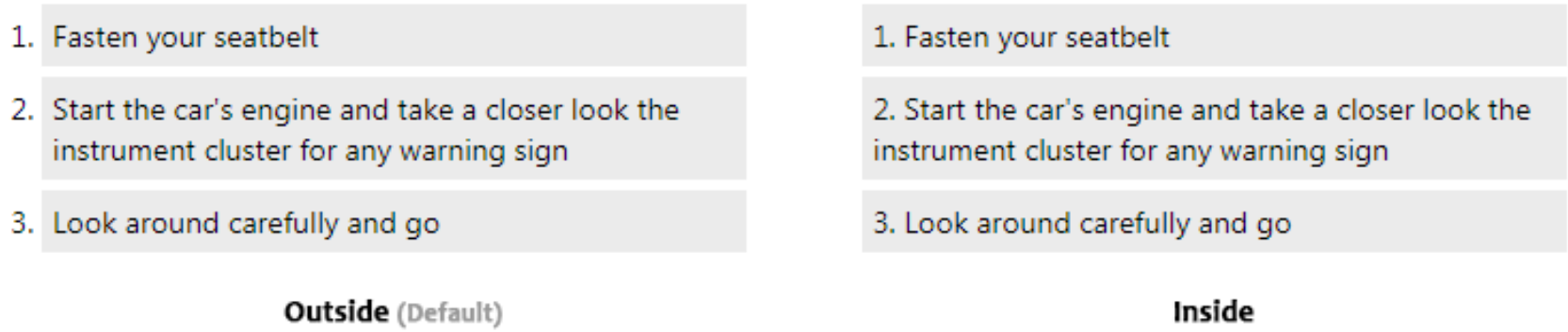
- First list item
- Second list item
- Third list item

A. First list item
B. Second list item
C. Third list item

list-style-position

- The list-style-position property specifies where the list item marker is placed.

Property	Value	Explanation
list-style-position	outside	the marker is placed outside the list item (**default**)
	inside	the marker is placed inside the list item



```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example1 { list-style-position: outside; }
#example2 { list-style-position: inside; }

</style>

</head>
<body>

<ul id="example1">
<li>First list item<br>The marker is placed outside</li>
<li>Second list item<br>The marker is placed outside</li>
<li>Third list item<br>The marker is placed outside</li>
</ul>

<ul id="example2">
<li>First list item<br>The marker is placed inside</li>
<li>Second list item<br>The marker is placed inside</li>
<li>Third list item<br>The marker is placed inside</li>
</ul>

</body>
</html>
```

Output

- First list item
  The marker is placed outside
- Second list item
  The marker is placed outside
- Third list item
  The marker is placed outside

  - First list item
  The marker is placed inside
  - Second list item
  The marker is placed inside
  - Third list item
  The marker is placed inside

list-style-image

- The list-style-image property sets an image as the list item marker.

Property	Value	Explanation
list-style-image	url(URL)	the URL of the image to display

- The default is "none".

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

ul { list-style-image: url(image/mark.gif); }

</style>

</head>
<body>

<ul>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ul>

</body>
</html>
```

Output

- First list item
- Second list item
- Third list item

list-style

- The list-style property is a shorthand property for setting all list-style properties.

Property	Value	Explanation
list-style	each value	sets type, position, and image

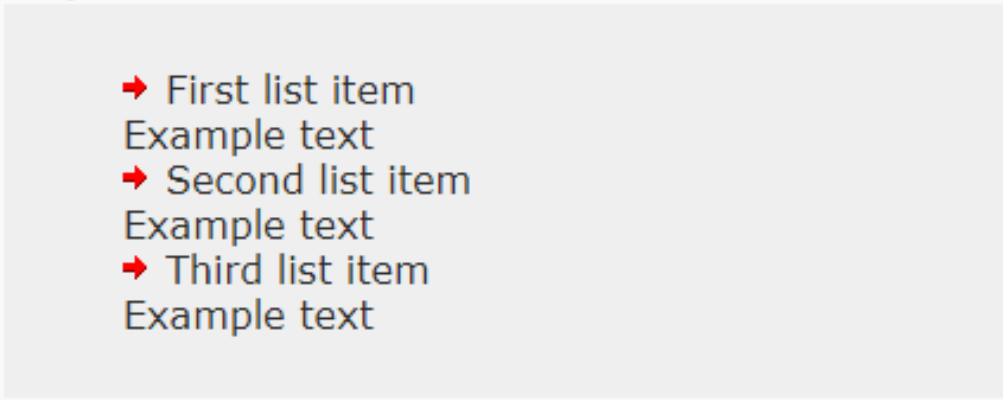
- Eg:
  - list-style: square inside url(image/mark.gif);

list-style:	square	inside	url(image/mark.gif)	;
list-style-	type	position	image	

- The unnecessary values can be omitted.
  Eg:
  - list-style: square inside;

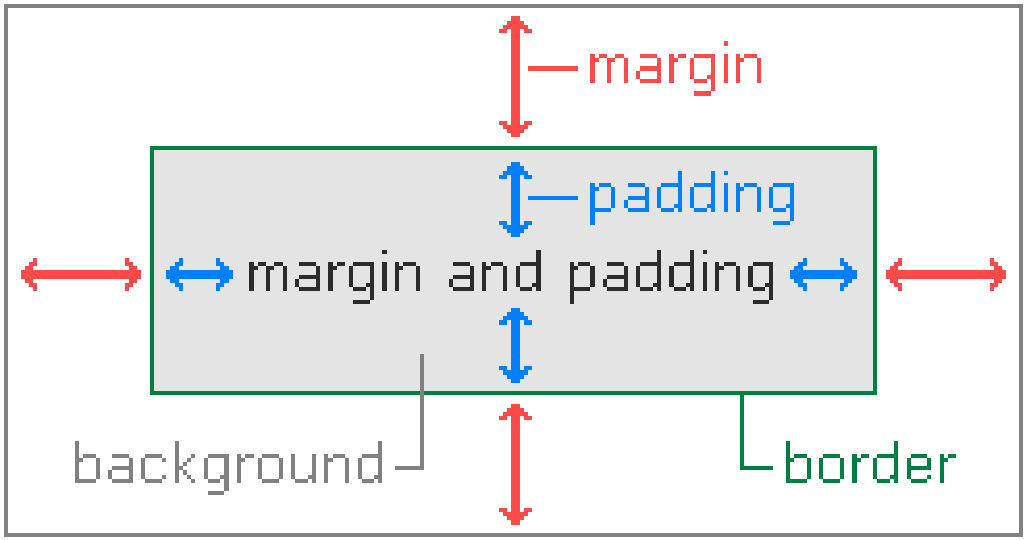
```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

ul { list-style: square inside url(image/mark.gif); }

</style>

</head>
<body>

<ul>
<li>First list item<br>Example text</li>
<li>Second list item<br>Example text</li>
<li>Third list item<br>Example text</li>
</ul>

</body>
</html>
```

Output



Boxes: Borders, Margins and Padding

- Every element that can be displayed on a web page is comprised of one or more rectangular boxes.
- CSS box model typically describes how these rectangular boxes are laid out on a web page.
- These boxes can have different properties and can interact with each other in different ways, but every box has a ***content area*** and optional surrounding ***padding***, ***border***, and ***margin areas***.



Margin

- Margin is the space between one element and another, in addition to and outside of any padding or border around an element.
- The margin- top/ bottom/left/ right property sets the top, bottom, left, and right margins of an element.

Property	Value	Explanation
margin-top	length, %, or auto	the top margin
margin-right	length, %, or auto	the right margin
margin-bottom	length, %, or auto	the bottom margin
margin-left	length, %, or auto	the left margin
margin	length, %, or auto	the top, bottom, left, and right margins

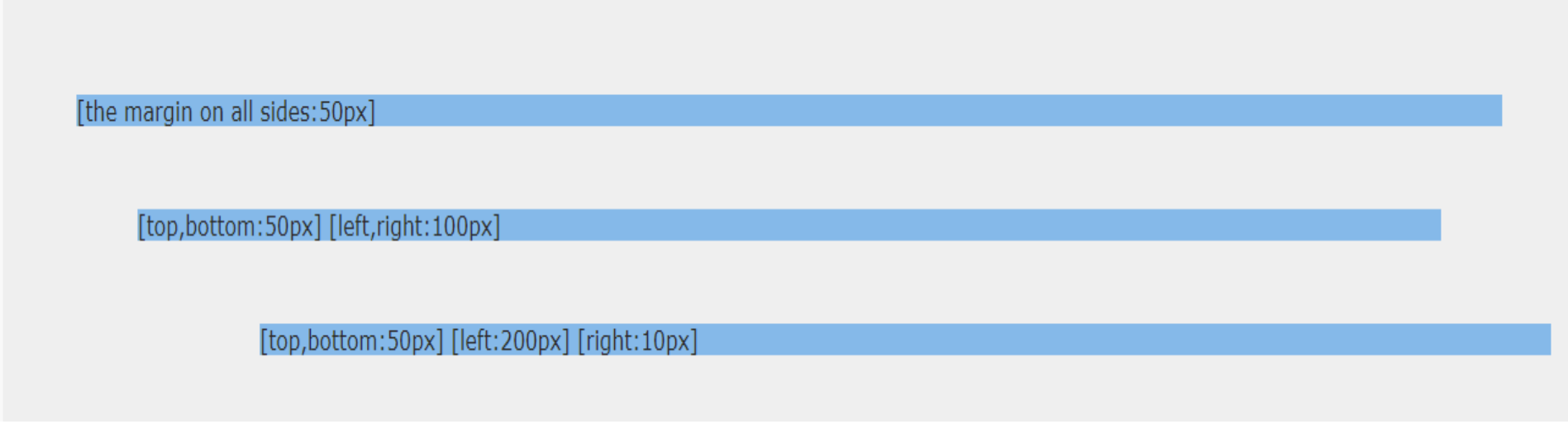
  - length specifies a margin in px, pt, cm, etc.
  - auto to horizontally center the element within its container.

Margin

- Eg:
  - margin: 2px; : [all sides] margins
  - margin: 2px 4px; : [top, bottom] [left, right] margins
  - margin: 2px 4px 6px; : [top] [left, right] [bottom] margins
  - margin: 2px 4px 6px 8px; : [top] [right] [bottom] [left] margins

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

p { background-color: #85b9e9; }

#example1 { margin: 50px; }
#example2 { margin: 50px 100px; }
#example3 { margin: 50px 10px 50px 200px; }

</style>

</head>
<body>

<p id="example1">[the margin on all sides:50px]</p>
<p id="example2">[top,bottom:50px] [left,right:100px]</p>
<p id="example3">[top,bottom:50px] [left:200px] [right:10px]</p>

</body>
</html>
```

Output



Padding

- Padding is the transparent space between the element's content and its border (or edge of the box, if it has no border), whereas margin is the transparent space around the border.
- The padding property sets the paddings of an element.

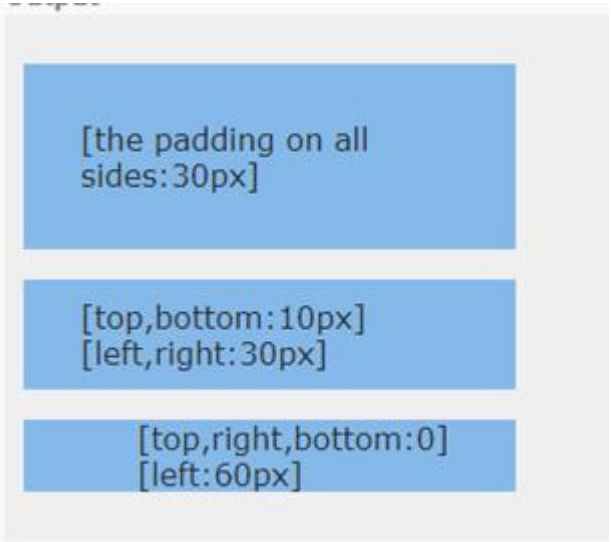
Property	Value	Explanation
padding-top	length, %, or auto	the top padding
padding-right	length, %, or auto	the right padding
padding-bottom	length, %, or auto	the bottom padding
padding-left	length, %, or auto	the left padding
padding	length, %, or auto	the top, bottom, left, and right paddings

Padding (conti..)

- Eg:
- padding: 2px; : [all sides] paddings
- padding: 2px 4px; : [top, bottom] [left, right] paddings
- padding: 2px 4px 6px; : [top] [left, right] [bottom] paddings
- padding: 2px 4px 6px 8px; : [top] [right] [bottom] [left] paddings

Padding (conti..)

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

p {
width: 200px;
background-color: #85b9e9;
}

#example1 { padding: 30px; }
#example2 { padding: 10px 30px; }
#example3 { padding: 0 0 0 60px; }

</style>

</head>
<body>

<p id="example1">[the padding on all sides:30px]</p>
<p id="example2">[top,bottom:10px] [left,right:30px]</p>
<p id="example3">[top,right,bottom:0] [left:60px]</p>

</body>
</html>
```



Border

- The CSS border properties allow you to define the border area of an element's box.
  - The **border-style** property sets the style of a box's border
  - The **border-width** property specifies the width of the border area.
  - The **border-color** property specifies the color of the border area.

- The default border style is none
- The default border width or thickness is medium,
- The default border color is the same as the text color.

Border (conti..)

- **border-width** : This is a shorthand property for setting the width for the four borders.

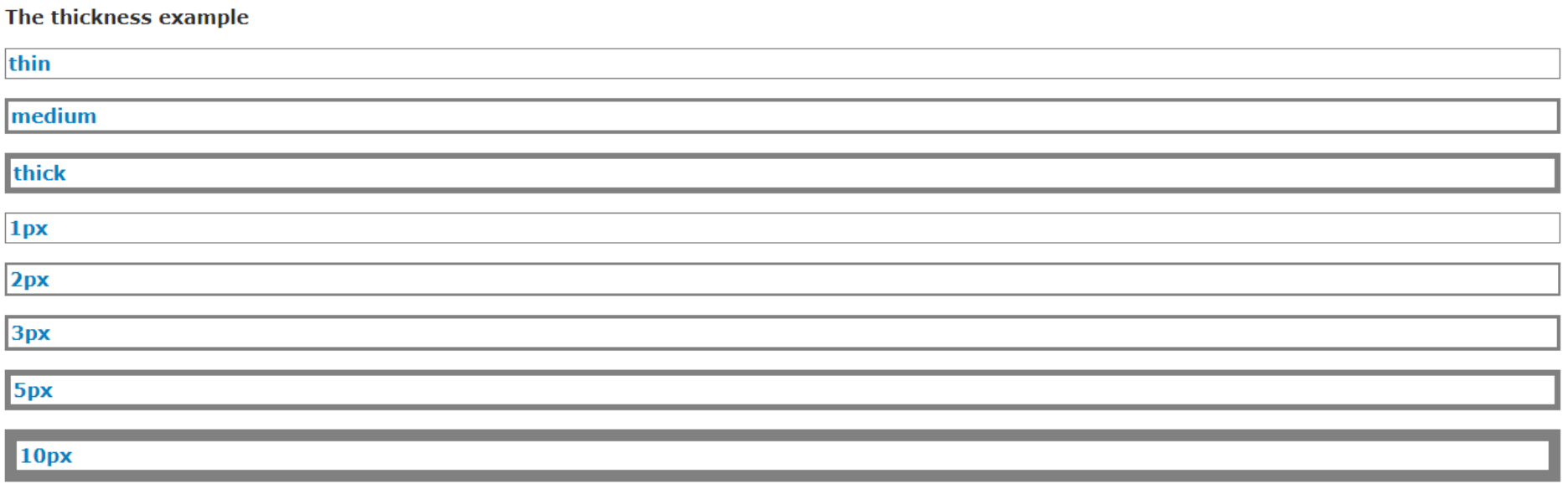
Property	Value	Explanation
border-width	thin, medium , thick, or length	the top, bottom, left, and right borders

- The default is "medium".
  - border-width: 5px; // [all sides] borders
  - border-width: 5px 10px; // [top, bottom] [left, right] borders
  - border-width: 5px 10px 15px; // [top] [left, right] [bottom] borders
  - border-width: 5px 10px 15px 20px; // [top] [right] [bottom] [left] borders

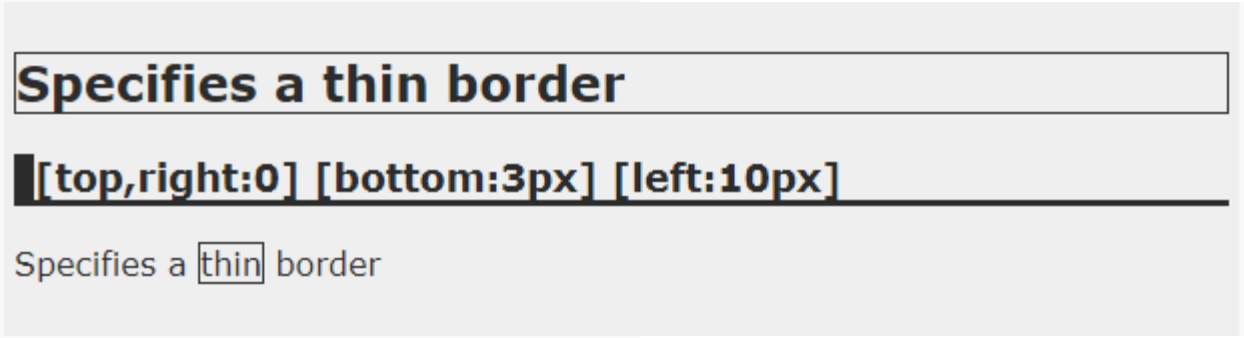
Border (conti..)

Property	Value	Explanation
border-top-width	thin, medium , thick, or length	the top border
border-right-width	thin, medium , thick, or length	the right border
border-bottom-width	thin, medium , thick, or length	the bottom border
border-left-width	thin, medium , thick, or length	the left border

Border (conti..)



```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

h2, h3, span {
border-color: #2b2b2b;
border-style: solid;
}

.example1 {
border-width: thin;
}
.example2 {
border-top-width: 0;
border-right-width: 0;
border-bottom-width: 3px;
border-left-width: 10px;
}

</style>

</head>
<body>

<h2 class="example1">Specifies a thin border</h2>
<h3 class="example2">[top,right:0] [bottom:3px] [left:10px]</h3>
<p>Specifies a <span class="example1">thin</span> border</p>

</body>
</html>
```



Border (conti..)

- **border-color** : This is a shorthand property for setting the color for the four borders.

Property	Value	Explanation
border-color	color code or name	the top, bottom, left, and right borders

- border-color: #85b9e9;

  [all sides] borders
- border-color: #85b9e9 #bde9ba;

  [top, bottom] [left, right] borders
- border-color: #85b9e9 #bde9ba #ffd37d;

  [top] [left, right] [bottom]
- border-color: #85b9e9 #bde9ba #ffd37d #d1d1d1;

  [top] [right] [bottom] [left]

Border (conti..)

Property	Value	Explanation
border-top-color	color code or name	the top border
border-right-color	color code or name	the right border
border-bottom-color	color code or name	the bottom border
border-left-color	color code or name	the left border

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

h2, h3, span {
border-width: thick;
border-style: solid;
}

.example1 {
border-color: red;
}
.example2 {
border-top-color: #85b9e9;
border-right-color: #bde9ba;
border-bottom-color: #ffd37d;
border-left-color: #d1d1d1;
}

</style>

</head>
<body>

<h2 class="example1">Specifies the red border</h2>
<h3 class="example2">Specifies four colors</h3>
<p>Specifies the <span class="example1">red</span> border</p>

</body>
</html>
```

Border (conti..)

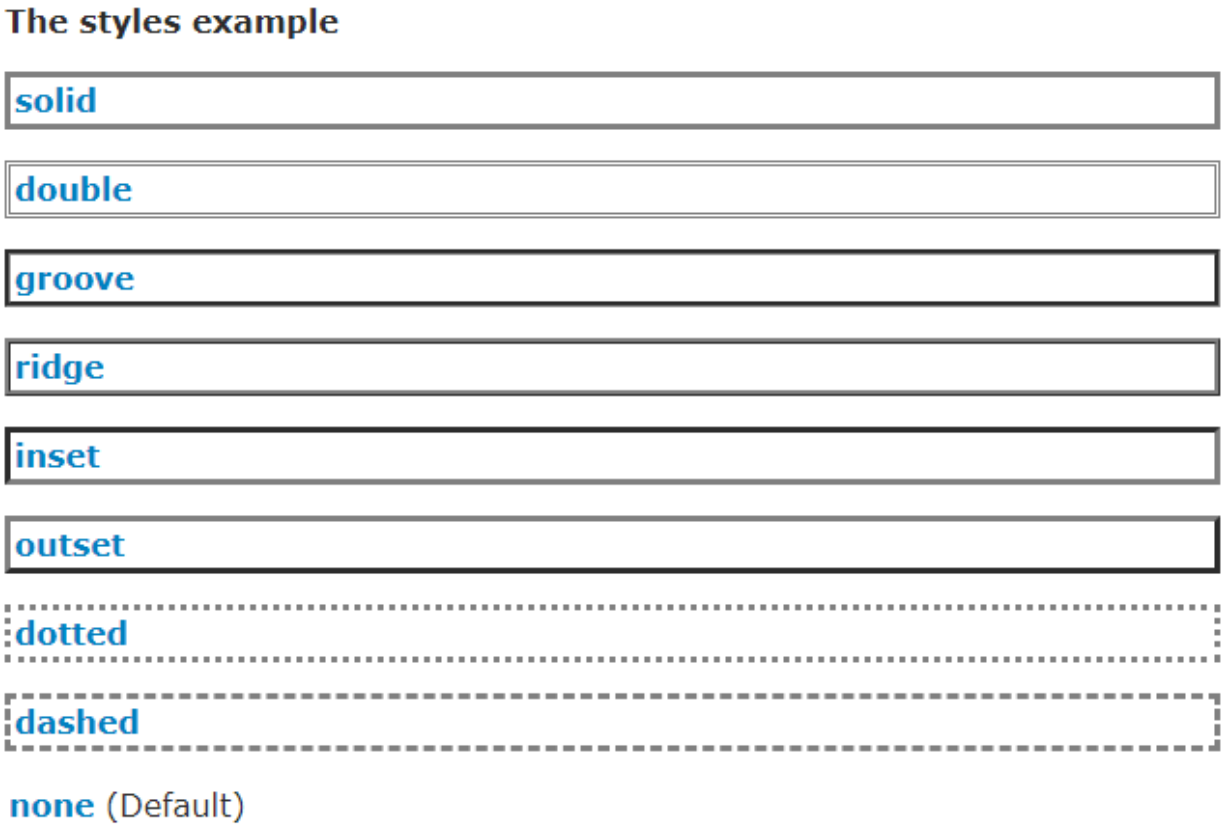
- **border-style** : This is a shorthand property for setting the style for the four borders.

Property	Value	Explanation
border-style	solid, double, groove, ridge, inset, outset, dotted, dashed, none	the top, bottom, left, and right borders

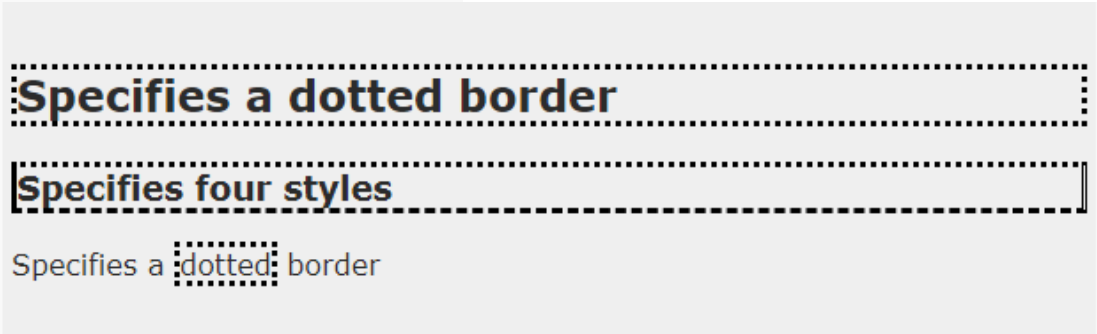
- The default is "none".
  - border-style: solid; // [all sides] borders
  - border-style: dashed double; //[top, bottom] [left, right] borders
  - border-style: dashed double dotted; // [top] [left, right] [bottom] borders
  - border-style: solid dashed double dotted; // [top] [right] [bottom] [left] borders

Border (conti..)

Property	Value	Explanation
border-top-style	solid, double, groove, ridge, inset, outset, dotted, dashed, none	the top border
border-right-style		the right border
border-bottom-style		the bottom border
border-left-style		the left border



```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

h2, h3, span {
border-width: medium;
border-color: #000000;
}

.example1 {
border-style: dotted;
}
.example2 {
border-top-style: dotted;
border-right-style: double;
border-bottom-style: dashed;
border-left-style: groove;
}

</style>

</head>
<body>

<h2 class="example1">Specifies a dotted border</h2>
<h3 class="example2">Specifies four styles</h3>
<p>Specifies a <span class="example1">dotted</span> border</p>

</body>
</html>
```



Border (conti..)

- **border** : This is a shorthand property for setting all border properties for the four borders.

Property	Value	Explanation
Border	each value	sets width, color, and style

- border: thick red solid;

border:	thick	red	solid	;
border-	width	color	style	

Border (conti..)

Property	Value	Explanation
border-top	each value	the top border
border-right	each value	the right border
border-bottom	each value	the bottom border
border-left	each value	the left border

Property	Description	Values
Border	Sets all properties for four borders in one declaration	border-width border-style border-color
border-bottom	Sets all properties for bottom border in one declaration	border-bottom-width border-style border-color
border-bottom-color	Sets bottom border color	border-color
border-bottom-style	Sets bottom border style	border-style
border-bottom-width	Sets bottom border width	thin, medium, thick length
border-color	Sets the color of the borders	color
border-left	Sets all properties for left side	border-left-width border-style border-color
border-left-color	Sets left border color	border-color
border-left-style	Sets left border style	border-style
border-left-width	Sets left border width	thin, medium, thick length
border-right	Sets all properties for right side	border-right-width border-style border-color
border-right-color	Sets right border color	border-color
border-right-style	Sets right border style	border-style
border-right-width	Sets right border width	thin, medium, thick length

Table 4-2 The border property

Border (conti..)

-

Property	Description	Values
border-style	Sets style of the four borders	none hidden dotted dashed solid double groove ridge inset outset
border-top	Sets all properties for top border	border-top-width border-style border-color
border-top-color	Sets color for top border	border-color
border-top-style	Sets style for top border	border-style
border-top-width	Sets width for top border	thin, medium, thick length
border-width	Sets width for all borders	thin, medium, thick length

Table 4-2 (continued)

```
h2 {
border-top: medium red dotted;
border-right: thick blue double;
border-bottom: medium green dashed;
border-left: thick gray groove;
}

h3, span {
border: medium red dotted;
}

</style>

</head>
<body>

<h2>Specifies four different borders</h2>
<h3>Specifies the red dotted border</h3>
<p>Specifies the <span>red dotted</span> border</p>

</body>
</html>
```

Output

Position

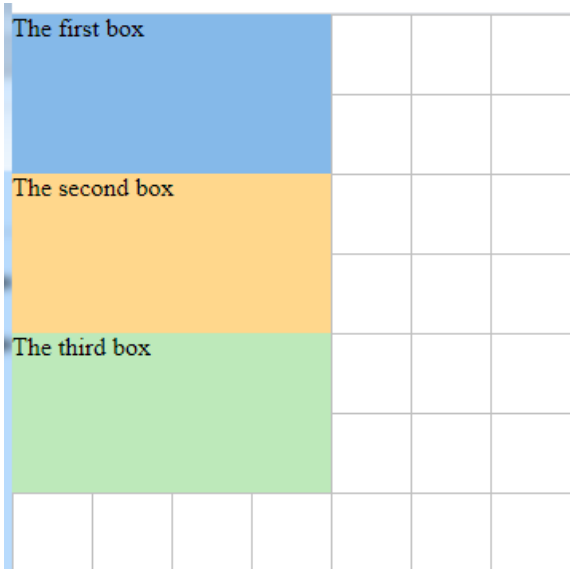
- The position property sets how a box will be positioned.

Property	Value	Explanation
position	static	the normal position (**default**)
	relative	the relative position from the normal position
	absolute	the absolute position from the parent element
	fixed	the absolute and fixed position from the window

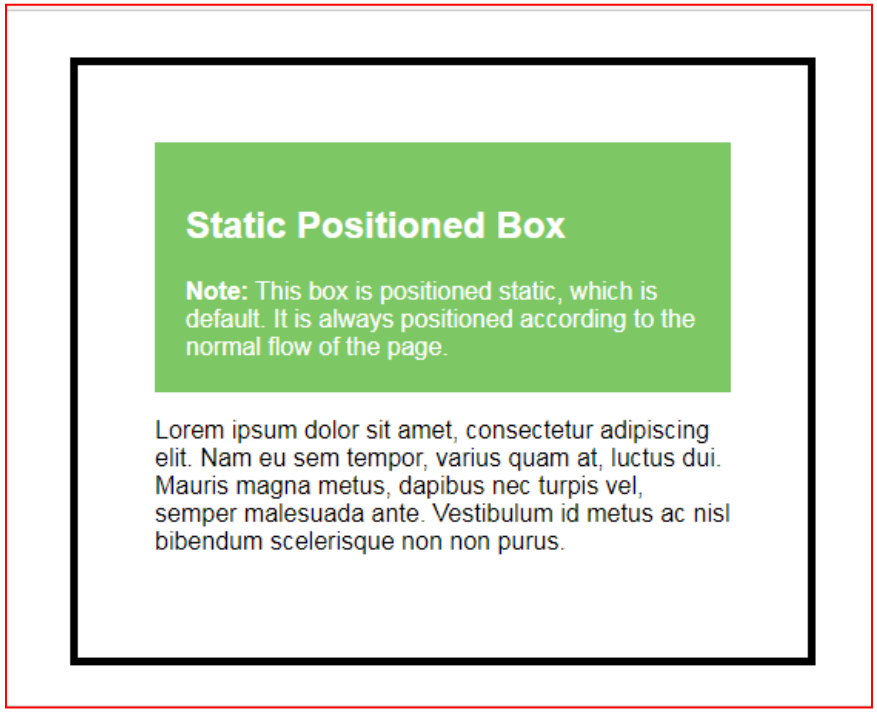
Position

- **static** (The normal position)
- A static positioned element is always positioned according to the normal flow of the page. HTML elements are positioned static by default.
- The static value ignores the top, bottom, left, or right properties.

```
div {
width: 200px;
height: 100px;
}

<div style="background-color: #85b9e9;">The first box</div>
<div style="background-color: #ffd78c; position: static;">The second box</div>
<div style="background-color: #bde9ba;">The third box</div>
```



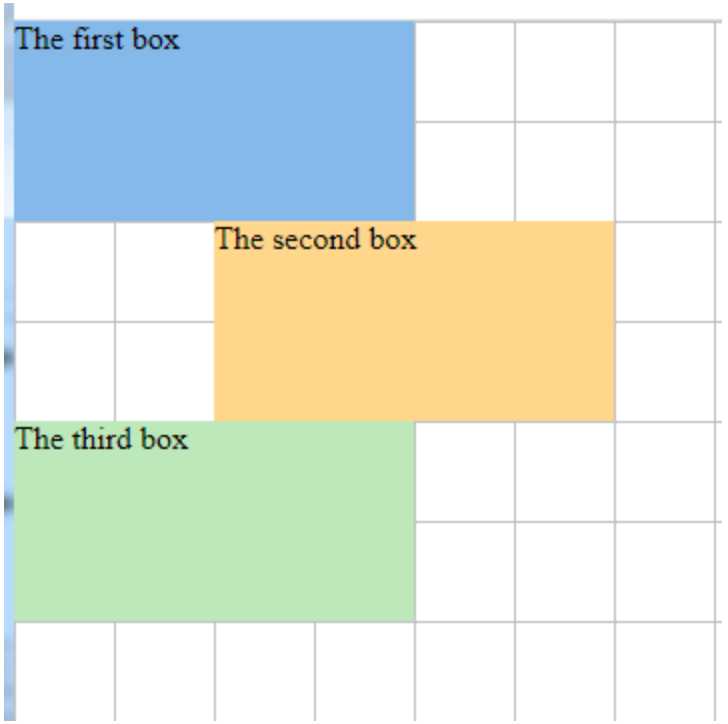
```
<html>
<head>
<title>Example of CSS Static Positioning</title>
<style>
    .box{
        color: #fff;
        background: #7dc765;
        padding: 20px;
    }
    .container{
        padding: 50px;
        margin: 50px;
        position: relative;
        border: 5px solid black;
        font-family: Arial, sans-serif;
    }

</style>
</head>
<body>
    <div class="container">
        <div class="box">
            <h2>Static Positioned Box</h2>
            <div><strong>Note:</strong> This box is positioned
static, which is default. It is always positioned according to the
normal flow of the page.</div>
        </div>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Nam eu sem tempor, varius quam at, luctus dui. Mauris magna metus,
dapibus nec turpis vel, semper malesuada ante. Vestibulum id metus
ac nisl bibendum scelerisque non non purus. </p>
    </div>
</body>
</html>
```



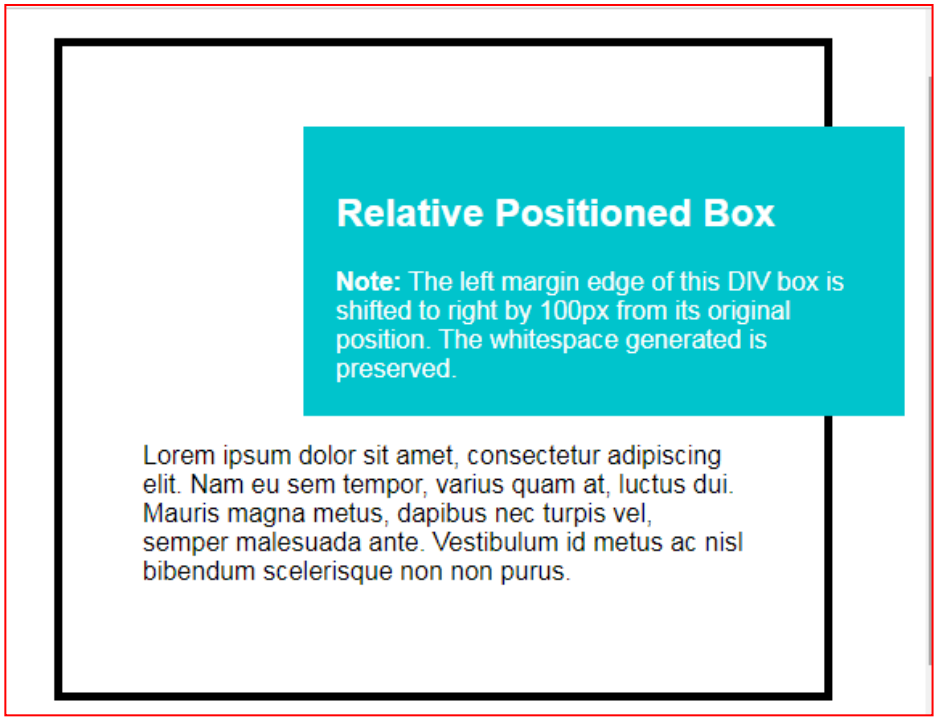
Position (conti..)

- **relative** (The relative position from the normal position)
- In the relative positioning scheme the element's box position is calculated according to the normal flow. Then the box is shifted from this normal position according to the properties — top or bottom and/or left or right.

```
div {
width: 200px;
height: 100px;
}

<div style="background-color: #85b9e9;">The first box</div>
<div style="background-color: #ffd78c; position: relative; top: 0;
left:100px;">The second box</div>
<div style="background-color: #bde9ba;">The third box</div>
```



```
<html>
<head>
<title>Example of CSS Relative Positioning</title>
<style>
    .box{
        position: relative;
        left: 100px;
        color: #fff;
        background: #00c4cc;
        padding: 20px;
    }
    .container{
        padding: 50px;
        margin: 50px;
        border: 5px solid black;
        font-family: Arial, sans-serif;
    }

</style>
</head>
<body>
    <div class="container">
        <div class="box">
            <h2>Relative Positioned Box</h2>
            <div><strong>Note:</strong> The left margin edge of this
DIV box is shifted to right by 100px from its original position. The
whitespace generated is preserved.</div>
        </div>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Nam eu sem tempor, varius quam at, luctus dui. Mauris magna metus,
dapibus nec turpis vel, semper malesuada ante. Vestibulum id metus
ac nisl bibendum scelerisque non non purus.</p>
    </div>
</body>
</html>
```



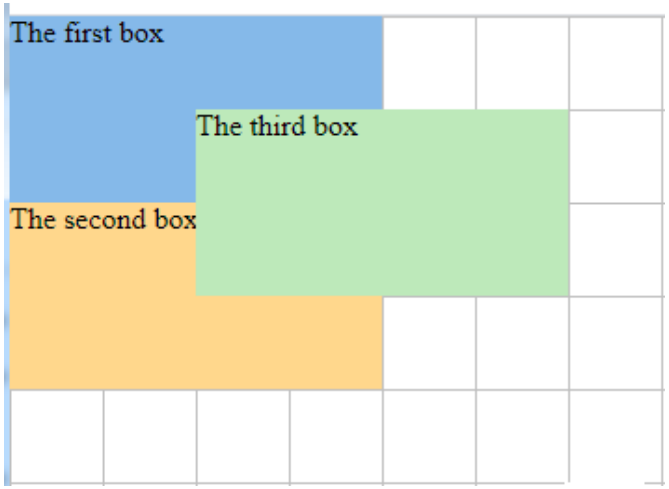
Position (conti..)

- **absolute** (The absolute position from the parent element)
- An absolutely positioned element is positioned relative to the first parent element that has a position other than static.
- If no such element is found, it will be positioned on a page relative to the 'top-left' corner of the browser window.
- The box's offsets further can be specified using one or more of the properties top, right, bottom, and left.
- Absolutely positioned elements are taken out of the normal flow entirely and thus take up no space when placing sibling elements.
- However, it can overlap other elements depending on the z-index property value.
- Also, an absolutely positioned element can have margins, and they do not collapse with any other margins.

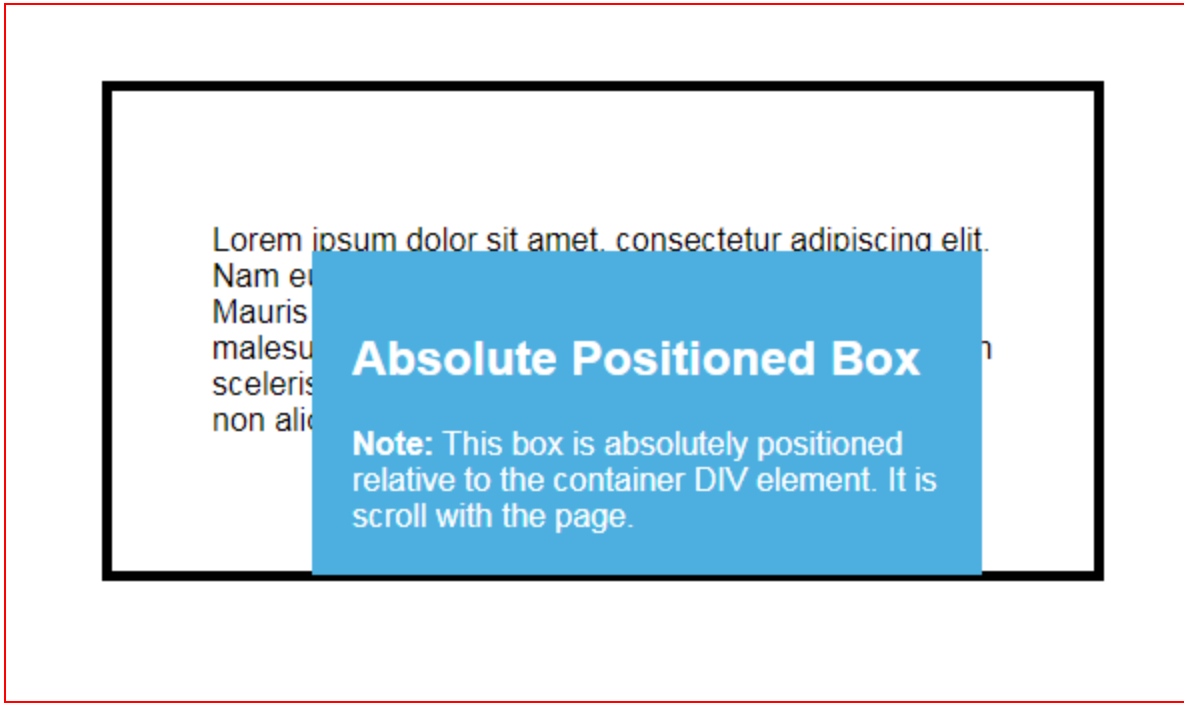
Position (conti..)

- absolute example

```
div {
width: 200px;
height: 100px;
}

<div style="background-color: #85b9e9;">The first box</div>
<div style="background-color: #ffd78c;">The second box</div>
<div style="background-color: #bde9ba; position: absolute; top: 50px;
left:100px;">The third box</div>
```



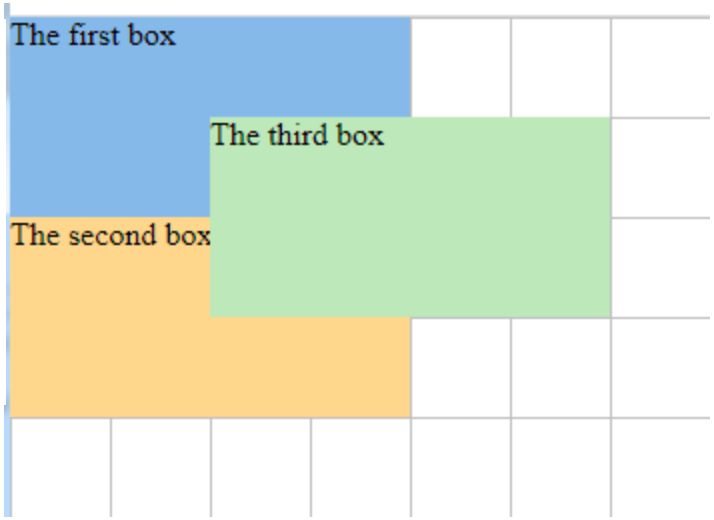
```
<html>
  <head>
<style>
    .box{
        position: absolute;
        top: 100px;
        left: 100px;
        color: #fff;
        width: 60%;
        background: #4cafdf;
        padding: 20px;
    }
    .container{
        padding: 50px;
        margin: 50px;
        position: relative;
        border: 5px solid black;
        font-family: Arial, sans-serif;
    }

</style>
</head>
<body>
    <div class="container">
        <div class="box">
            <h2>Absolute Positioned Box</h2>
            <div><strong>Note:</strong> This box is absolutely
positioned relative to the container DIV element. It is scroll with
the page.</div>
        </div>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Nam eu sem tempor, varius quam at, luctus dui. Mauris magna metus,
dapibus nec turpis vel, semper malesuada ante. Vestibulum id metus
ac nisl bibendum scelerisque non non purus. Suspendisse varius nibh
non aliquet sagittis.</p>

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



Position (conti..)

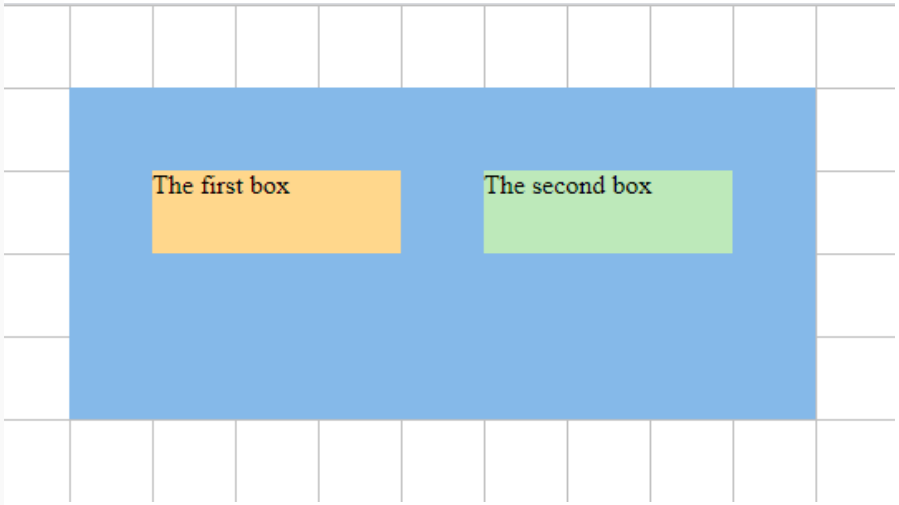
- **fixed** (The absolute and fixed position from the window)
- The box's position can be moved using the top, bottom, left, or right properties. (The box is fixed to the specified position)
- Fixed positioning is a subcategory of absolute positioning.
- The only difference is, a fixed positioned element is fixed with respect to the browser's viewport and does not move when scrolled.

```
div {
width: 200px;
height: 100px;
}

<div style="background-color: #85b9e9;">The first box</div>
<div style="background-color: #ffd78c;">The second box</div>
<div style="background-color: #bde9ba; position: fixed; top: 50px;
left:100px;">The third box</div>
```



```html
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#example { /* The parent box */
width: 450px;
height: 200px;
background-color: #85b9e9;
position: absolute;
top: 50px;
left: 100px;
}

#box1 { /* The first box */
width: 150px;
height: 50px;
background-color: #ffd78c;
position: absolute;
top: 50px;
left: 50px;
}

#box2 { /* The second box */
width: 150px;
height: 50px;
background-color: #bde9ba;
position: absolute;
top: 50px;
left: 250px;
}

</style>

</head>
```

```html
<body>

<div id="example">

<div id="box1">The first box</div>
<div id="box2">The second box</div>

</div>

</body>
</html>
```



z-index property

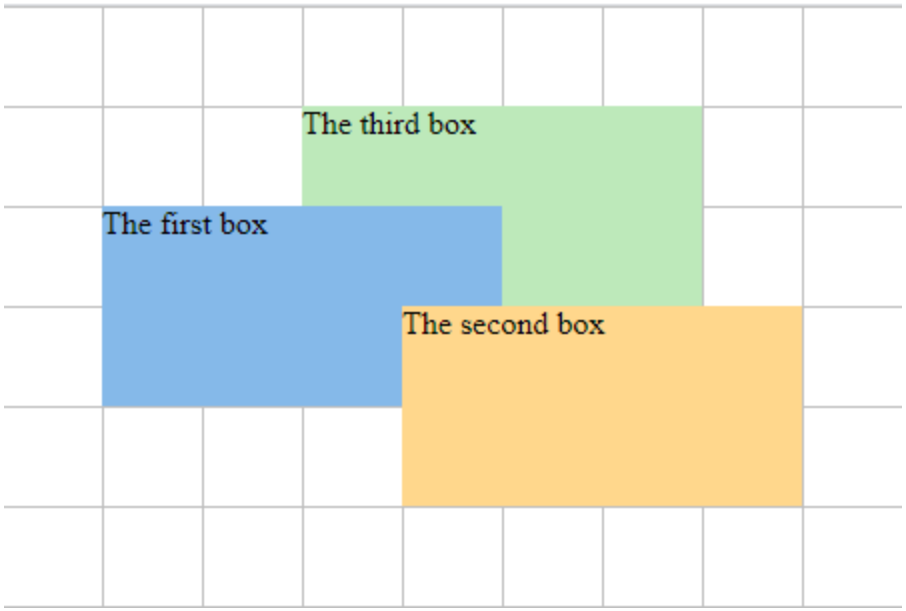
- The CSS z-index property can be used in conjugation with the position property to create an effect of layers like Photoshop.
- Usually HTML pages are considered two-dimensional, because text, images and other elements are arranged on the page without overlapping.
- However, in addition to their horizontal and vertical positions, boxes can be stacked along the z-axis as well i.e. one on top of the other by using the CSS z-index property.
- This property specifies the stack level of a box whose position value is one of absolute, fixed, or relative.
- The z-axis position of each layer is expressed as an integer representing the stacking order for rendering.
- An element with a larger z-index overlaps an element with a lower one.

Z-index (conti..)

- The z-index property sets the stack order of the boxes.

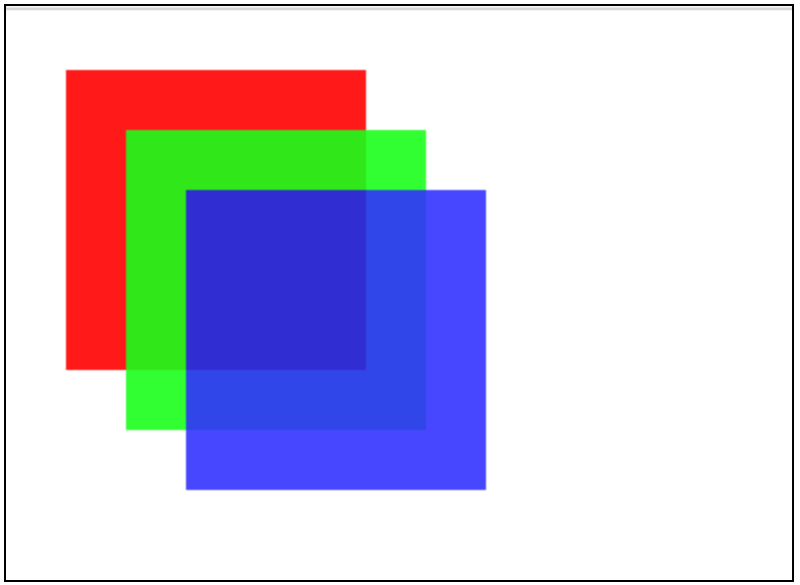
Property	Value	Explanation
z-index	number or auto	the box with the higher number will appear in front

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

div {
width: 200px;
height: 100px;
position: absolute;
}

#example1 {
z-index: 1;
background-color: #85b9e9;
top: 100px;
left: 50px;
}
#example2 {
z-index: 2;
background-color: #ffd78c;
top: 150px;
left: 200px;
}
#example3 {
z-index: 0;
background-color: #bde9ba;
top: 50px;
left: 150px;
}

</style>

</head>
<body>

<div id="example1">The first box</div>
<div id="example2">The second box</div>
<div id="example3">The third box</div>

</body>
</html>
```



```html
<html >
<head>
  <style>
      .box{
          width: 150px;
          height: 150px;
          opacity: 0.9;
          position: absolute;
          top: 30px;
          left: 30px;
      }
      .red{
          background: #ff0000;
          z-index: 1;
      }
      .green{
          background: #00ff00;
          z-index: 2;
      }
      .blue{
          background: #0000ff;
          z-index: 3;
      }
  </style>
</head>
```

```html
<body>
  <div class="box red">
      <div class="box green">
          <div class="box blue"></div>
      </div>
  </div>
</body>
</html>
```



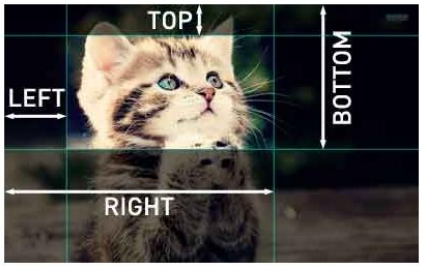
clip

- The **clip property** specify at defining what portion of an absolutely positioned element you want to make visible.
- Except for the specified region, all are hidden.
- The clip property applies only to absolutely positioned elements — that is, elements with **position: absolute** or **position: fixed**.

clip: auto|shape|initial|inherit;

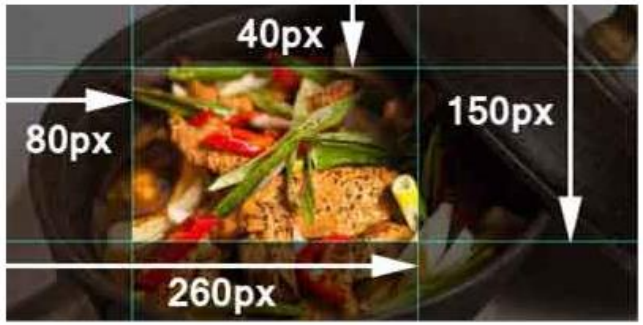
  - **auto:** It is default value, there won't be any clipping. The element is shown as it is.
  - **initial:** initial sets the default value i.e. there won't be any clipping as the default value is auto.
  - **inherit:** inherit receives the property from the parent element. When it is used with the root element then the initial property will be used.
  - **shape:** shape clips the defined portion of the element. The rect(top, right, bottom, left) is used to define visible portion.

- **Eg:**
  - **clip: auto;**
  - **clip: rect(<top>, <right>, <bottom>, <left>);**



Clip: rect()

- **clip: rect(40px, 260px, 150px, 80px);**



- The rect() function accepts **auto** as a value for each one of the four parameters. Basically it means “100%”.
- Let’s say you clip something to
  rect(0, 50px, 50px, 0) and want to display it in full size (100×100 let’s say).
  You could either do,
  rect(0, 100px, 100px, 0) or
  rect(0, auto, auto, 0).

Clip: inherit - Eg

```
<html>

<head>
<title>
        CSS | clip Property
</title>
<style>
        .shape {
                position: absolute;
                background: #0F9D58;
                width: 200px;
                height: 200px;
                color: #ffffff;
                text-align: center;
        }

        .shape1 {
                border: solid;
                border-color: black;
                position: absolute;
                background: #ffffff;
                width: 200px;
                height: 200px;
                color: #0F9D58;
                text-align: center;
        }

        #clip_property {
                clip: rect(0px, 120px, 100px, 0px);
        }

        #clip_property1 {
                clip: inherit;
        }
</style>
</head>

<body>

<div class="shape" id="clip_property">
        <p>
        Good Morning
        </p>
        <div class="shape1" id="clip_property1">
                <p>
                Hello World
                </p>
        </div>
</div>
</body>

</html>
```

Float

- The float property creates a floating box.
- You can float elements to the left or right, but only applies to the elements that generate boxes that are not absolutely positioned.
- Any element that follows the floated element will flow around the floated element on the other side.

Property	Value	Explanation
float	left	the box floats to the left (the text wraps to the right of the box)
	Right	the box floats to the right (the text wraps to the left of the box)
	none	the box doesn't float (**default**)

Float (conti..)

- Elements are floated horizontally, which means that an element can only be floated left or right, not up or down.
- If several floating elements are placed adjacently, they will float next to each other if there is horizontal room.
- If there is not enough room for the float, it is shifted downward until either it fits or there are no more floating elements present.

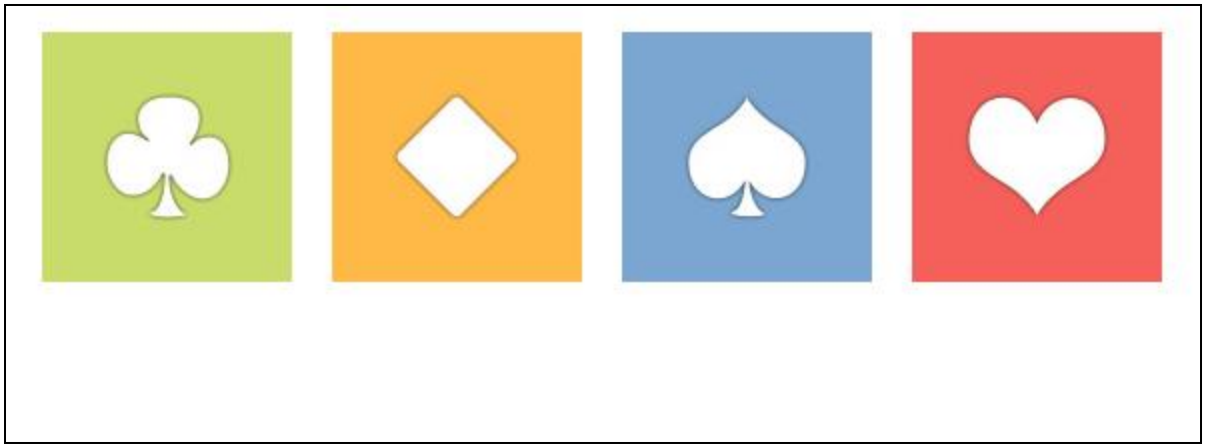
Float (conti..)

- The float property creates a floating box.
- When you apply this property to the LI element, the horizontal list can be created.
- float (Box)
- float (List)
- float (Image)
- float (Table)
- float (Frame)

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Example of Floating Elements</title>
<style>
    img {
        float: left;
        width: 150px;
        height: 150px;
        margin-right: 20px;
    }
</style>
</head>
<body>
    <p><img src="/examples/images/kites.jpg" alt="Flying Kites"> Lorem
ipsum dolor sit amet, consectetur adipiscing elit. Nam eu sem tempor,
varius quam at, luctus dui. Mauris magna metus, dapibus nec turpis
vel, semper malesuada ante. Vestibulum id metus ac nisl bibendum
scelerisque non non purus. Suspendisse varius nibh non aliquet
sagittis. </p>
</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam eu sem tempor, varius quam at, luctus dui. Mauris magna metus, dapibus nec turpis vel, semper malesuada ante. Vestibulum id metus ac nisl bibendum scelerisque non non purus. Suspendisse varius nibh non aliquet sagittis.

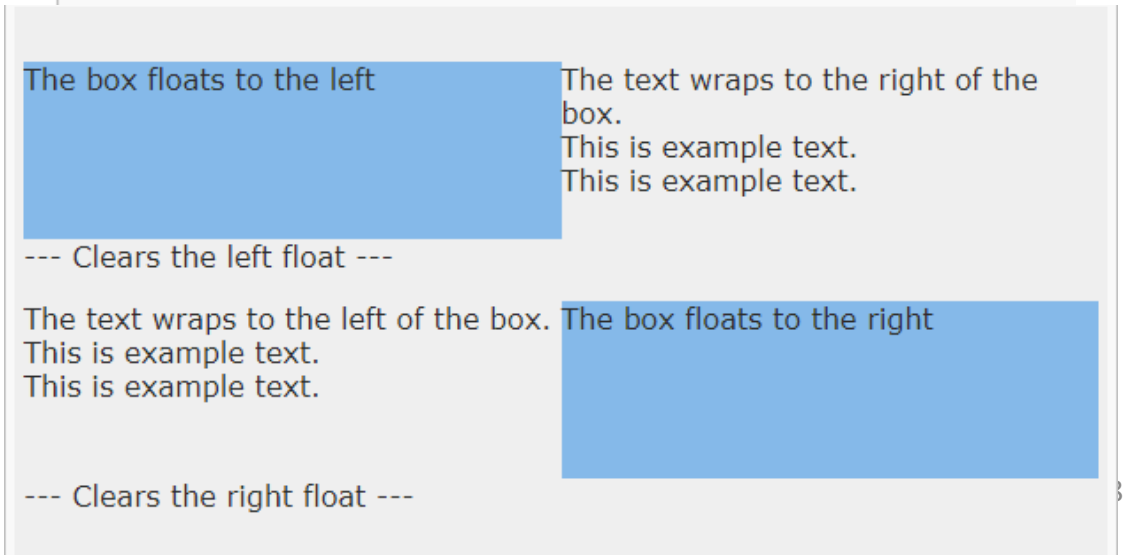
```
<html>
<head>
<style>
    ul{
        margin: 0;
        padding: 0;
        list-style: none;
    }
    .thumbnail {
        float: left;
        width: 125px;
        height: 125px;
        margin: 10px;
    }
</style>
</head>
```



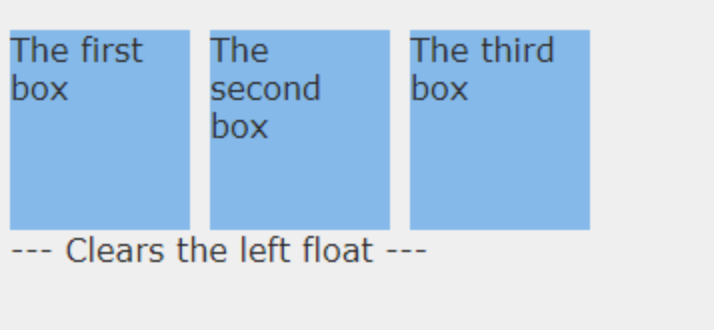
```
<body>
    <ul>
        <li class="thumbnail"><img src="/examples/images/club.jpg"
alt="Club Card"></li>
        <li class="thumbnail"><img
src="/examples/images/diamond.jpg" alt="Diamond Card"></li>
        <li class="thumbnail"><img src="/examples/images/spade.jpg"
alt="Spade Card"></li>
        <li class="thumbnail"><img src="/examples/images/heart.jpg"
alt="Heart Card"></li>
    </ul>
</body>
</html>
```

```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#exampleA {
width: 100%;
}

#example1 {
width: 50%;
height: 100px;
background-color: #85b9e9;
float: left;
}

#example2 {
width: 50%;
height: 100px;
background-color: #85b9e9;
float: right;
}

</style>
```

```
<body>

<div id="exampleA">

<div id="example1">The box floats to the left</div>
<p>
The text wraps to the right of the box.<br>
This is example text.<br>
This is example text.
</p>

<p style="clear: left;">--- Clears the left float ---</p>

<div id="example2">The box floats to the right</div>
<p>
The text wraps to the left of the box.<br>
This is example text.<br>
This is example text.
</p>

<p style="clear: right;">--- Clears the right float ---</p>

</div>

</body>
</html>
```

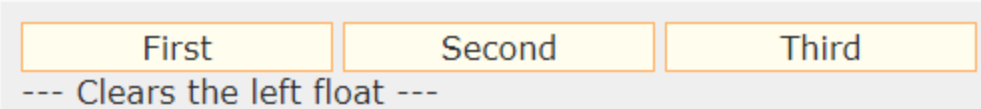


```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

#exampleB {
width: 300px;
}

.example3 {
width: 90px;
height: 100px;
background-color: #85b9e9;
margin-right: 10px;
float: left;
}

</style>

</head>
<body>

<div id="exampleB">

<div class="example3">The first box</div>
<div class="example3">The second box</div>
<div class="example3">The third box</div>
<p style="clear: left;">--- Clears the left float ---</p>

</div>

</body>
</html>
```



```
<html>
<head>
<title>TAG index</title>

<style type="text/css">

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

li {
float: left;
width: 150px;
margin-right: 5px;
padding: 2px;
border: 1px #ffb366 solid;
background-color: #fffdee;
text-align: center;
}

</style>

</head>
<body>

<ul>
<li>First</li>
<li>Second</li>
<li>Third</li>
</ul>

<p style="clear: left;">--- Clears the left float ---</p>

</body>
</html>
```

Output



Unit of length that can be used in CSS

Unit	Unit Name	Explanation
Absolute length units		
mm	millimeter	10mm = 1cm
cm	centimeter	1cm = 10mm
in	inch	1in = 2.54cm
pt	point	72pt = 1in
pc	pica	1pc = 12pt
Relative length units		
em	em	1em = the current height of the font
ex	x-height	1ex = the current height of the "x" character
px	pixels	1px = one dot on the screen
Percentage unit		
%	percentage	-

In CSS, some unit must be specified for length values. However, only when a value is zero, the unit can be omitted.

About "em"

1em is the current font size (It is based on "M" character). If the current font size is 14pt, 1em becomes 14pt.

1em = 14pt

2em = 28pt

0.5em = 7pt

THANK YOU!