**Css:**

Css stands for cascading style sheets.by using css we can apply the styles for the web pages. Here to apply styles for the html elements by using style sheets we have 3 types of style sheets.

They are:

1. Inline style sheets

2. Internal style sheets

3. External style sheets

**1. Inline style sheets:**

These are used to apply the styles for the specified tag in the webpage. These styles will applicable for only specified tag.

To apply inline style sheets we use style attribute.

**Syntax:**

<tagname style=”values”/>

**Example**:

<p style=”color:red”>any text</p>

**Pogram:**

**Example :1**

<html>

<head>

<title>inline styles</title>

</head>

<body>

<h1>heading 1</h1>

<h1 style="background-color:blue;width:200px;color:white">heading 1</h1>

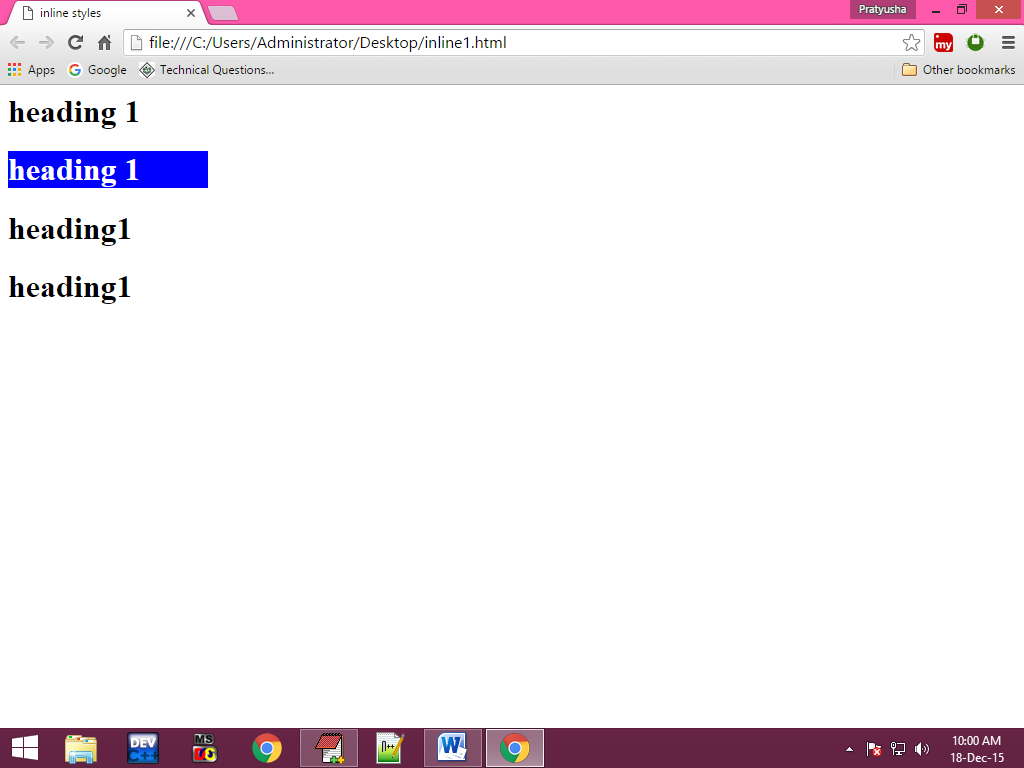
<h1>heading1</h1>

<h1>heading1</h1>

</body>

</html>

**Output:**



**Example :2**

<html>

<head>

<title> inline styles</title>

</head>

<body>

<p align="justify" style="word-spacing:10px;letter-spacing:2px;text-indent:20px;text-decoration:underline;font-style:italic;text-transform:small;background-color:blue;margin-left:30;margin-right:20px">

welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page

</p>

<p align="justify">

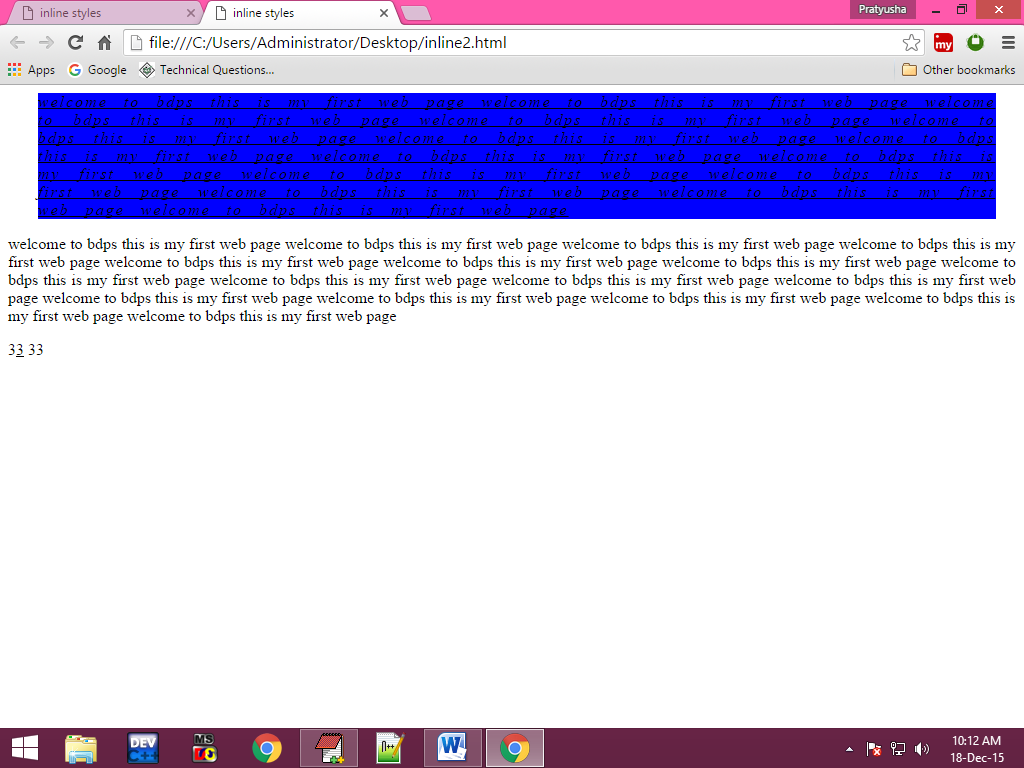
welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page welcome to bdps this is my first web page

</p>

html<abbr style="text-decoration:underline">hypertext markup lang</abbr>

</body>

</html>

**Output:**

**Example 3:**

<html>

<head>

<title>inline style </title>

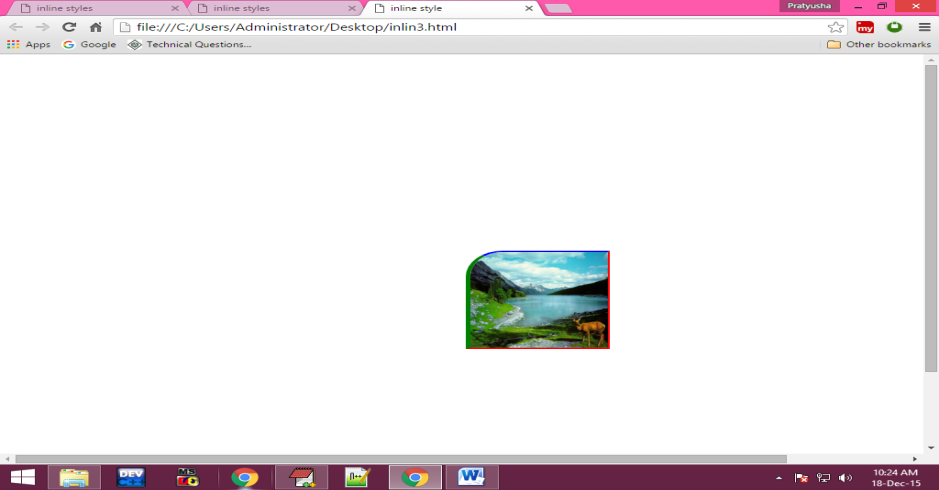
</head>

<body>

<img src="E:\my html notes\my notes\html images\77.jpg" height="150" width="150" border="2" hspace="500" vspace="300" style="border-color:red;border-left-color:green;border-top-color:blue;border-left-width:5px;border-right-style:double;border-top-left-radius:40px;border-right-radius:50px">

</body>

</html>

**Output:**

**Example: 4**

<html>

<head>

<title>Inline Style Sheets</title>

</head>

<body>

<h1 align="center" style="font-size:80; color:orange; text-shadow:70px 50px 20px black">IRCTC</h1>

</body>

</html>

**Output:**



**Example: 5**

<html>

<head>

<title>inline style sheets</title>

</head>

<body>

<div style="height:200px;width:200px;border:solid;overflow:"auto\hidden\display">

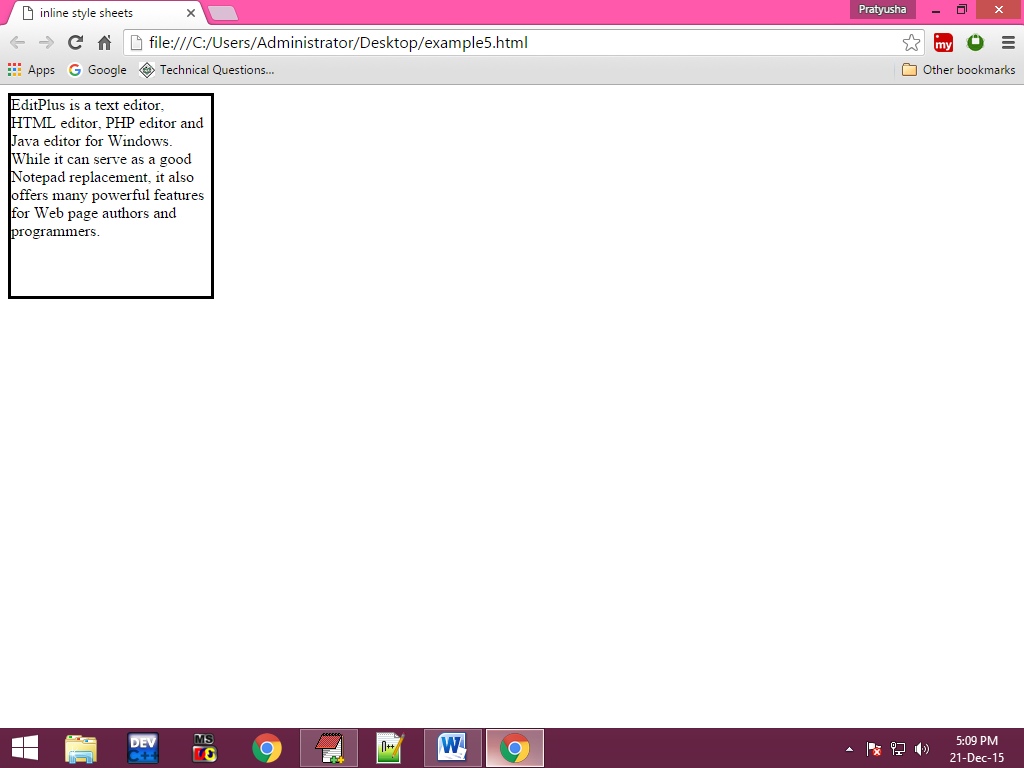
EditPlus is a text editor, HTML editor, PHP editor and Java editor for Windows. While it can serve as a good Notepad replacement, it also offers many powerful features for Web page authors and programmers.

</div>

</body>

</html>

**Output:**



**Example :6**

<html>

<head>

<title>inline styles</title>

</head>

<body>

<div style="height:100;width:200;border:solid;color:green;font-size:20px;padding-left:30px;padding-top:50px;background:blue">

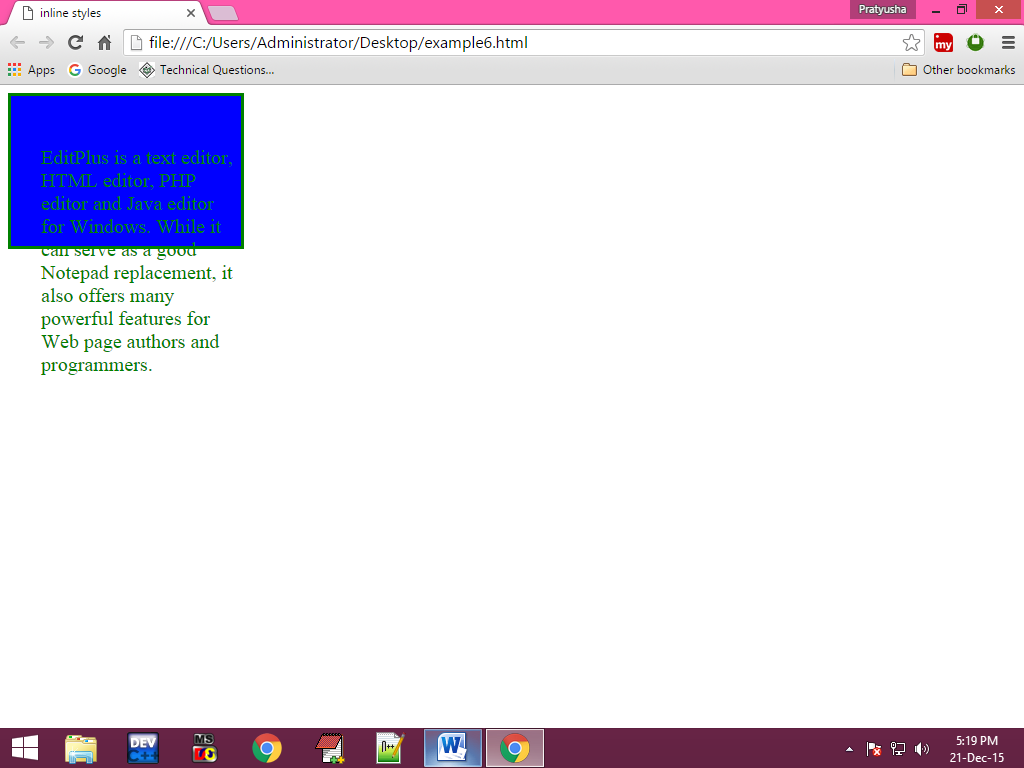
EditPlus is a text editor, HTML editor, PHP editor and Java editor for Windows. While it can serve as a good Notepad replacement, it also offers many powerful features for Web page authors and programmers.

</div>

</body>

</html>

**output:**



**Example : 7**

**<**html>

<head>

<title>inline style sheets</title>

<meta name="" content="width-device-width; initial-scale=1"/>

<body>

<table border="4" height="200" width="200" align="center">

<tr>

<td>1</td>

<td>abc</td>

<td>56</td>

</tr>

<tr bgcolor="gray" style="font-size:20px;text-align:center;font-style:italic">

<td>2</td>

<td>asp</td>

<td>89</td>

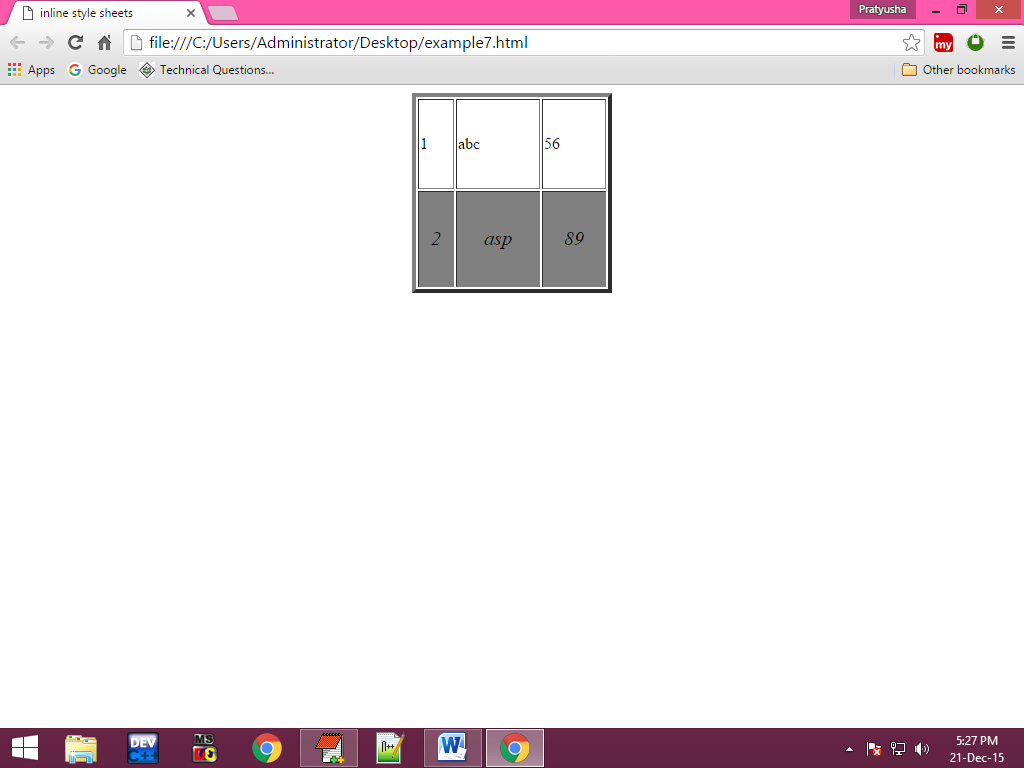
</tr>

</table>

</body>

</html>

**Output:**



**Example 8:**

<html>

<head>

<title>inline style sheets</title>

</head>

<body>

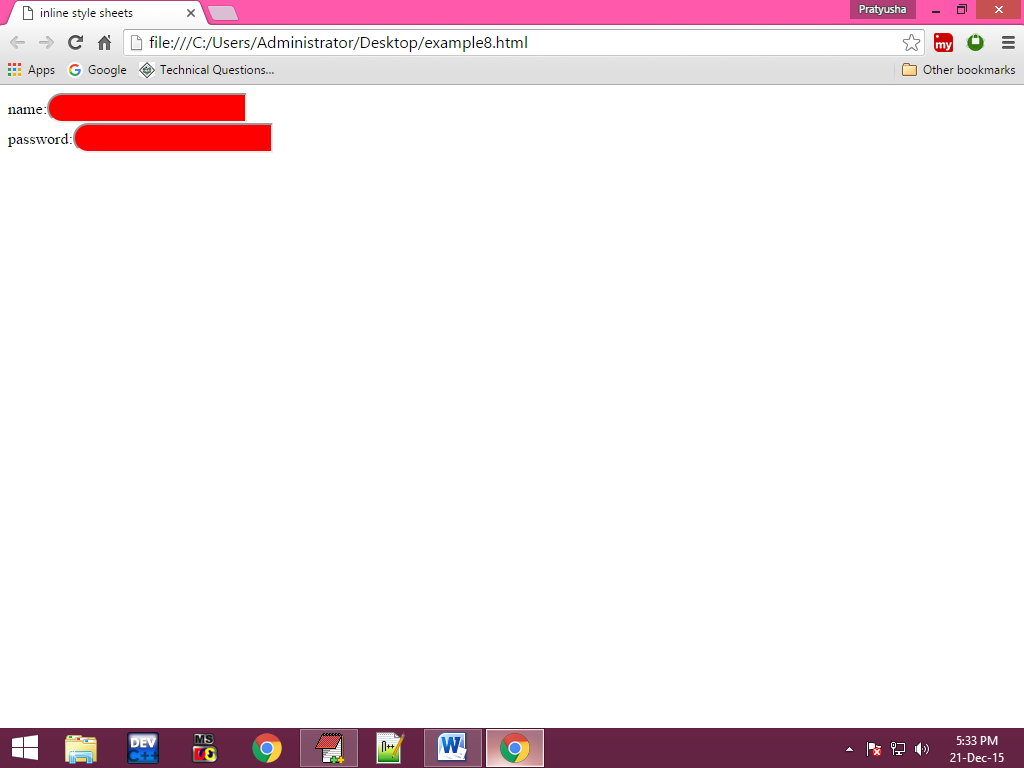
name:<input type="text" style="background-color:red;height:30px;width:200px;border-top-left-radius:20px;border-bottom-left-radius:20px;color:white;font-size:20px"><br/>

password:<input type="password" style="background-color:red;height:30px;width:200px;border-top-left-radius:20px;border-bottom-left-radius:20px;color:white;font-size:20px">

</body>

</html>

**Output:**



1. **Internal style sheets:**

These are used to apply the styles for specified tag in entire webpage. To apply internal style sheets we use style tag in html document. And internal styles are also called embedded style sheets.

**Selectors:**

Selectors are nothing but a pattern which are using to selects the html elements.

**Syntax:**

<style>

Selectors

{

Style value1;

Style value2;

---------------

---------------

}

</style>

In Internal Style Sheets we are using selectors. Here there are so many types of selectors are available some of them are

1. Generic Selector
2. Group Selector
3. Descendent Selector
4. Class Selector
5. Id Selector
6. **Generic selector:**

These type of selector can select only one html document.

**Syntax**

tagname

{

Style properties

}

1. **Group selector:**

Group selector can select more than one html at a time.

**Syntax:**

Tagname,tagname

{

Style properties

}

1. **descendent selector:**

Descendent selector can select the sub items.

**Syntax:**

Tagname1, tagname2

{

Style properties

}

1. **class selector:**

Class selector can group the style properties and it doesn’t select html document.

**Syntax:**

.Classname

{

Style properties

}

**Example:**

<tagname class>contents</tagname>

1. **id selector:**

Id selector can group the style properties and it doesn’t select any html document.

**Syntax:**

#id name{

Style properties

}

**Example:**

<tagname id=”idname”>contents</tagname>

**CSS Selectors**

The following is a list of the most common and well-supported CSS selectors. There are many, many more, but these are the ones you should know well.

* [Element Type Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#element)
* [Descendant Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#descendant)
* [Class selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#class)
* [Id Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#id)
* [Child Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#child)
* [Adjacent sibling selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#adjacent)
* [Pseudo Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#pseudo)
* [Universal Selectors](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#universal)
* [Additional Reading](http://web.simmons.edu/~grabiner/comm244/weekfour/selectors.html#additional)

**Element Type Selectors**

The most basic CSS selectors are Element Type Selectors. That's a fancy name for simply **using an HTML tag,** without the angle braces.

We've used this selector extensively already.

For example, if we wanted to make **all paragraphs** have green text, we would use the following CSS rule:

p { color: green; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/element.html)

**Descendant Selectors**

Match an element that is a descendant of another element.

This uses two separate selectors, separated by a space.

For example, if we wanted **all emphasized text in our paragraphs** to be green text, we would use the following CSS rule:

p em { color: green; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/descendant.html)

**Class Selectors**

Match an element that has the specified class.

To match a specific class attribute, we always start the selector with a period, to signify that we are looking for a class value. The period is followed by the class attribute value we want to match.

For example, if we wanted **all elements with a class of "highlight"** to have a different background color, we would use the following CSS rule:

.highlight { background-color: #ffcccc; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/class.html)

**Id Selectors**

Match an element that has the specified id.

To match a specific id attribute, we always start the selector with a hash symbol (#), to signify that we are looking for an id value. The hash is followed by the id attribute value we want to match. Remember, we can only use the same id attribute value once, so the id selector will always only match one element in our document.

For example, if we wanted **the element with an id of "content"**, we would use the following CSS rule:

#content { border: 2px solid green; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/id.html)

**Child selectors**

Match an element that is an immediate child of another element.

For example, if we wanted **all emphasized text in our paragraphs's** to have green text,**but not emphasized text in other elements**, we would use the following CSS rule:

p > em { color: green; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/child.html)

Note: This selector does not work in Internet Explorer 6

**Adjacent sibling selectors**

Match an element that is immediately after another element, but not a child of it.

For example, if we wanted **all paragraphs that immediately followed an h4** to have green text, **but not other paragraphs**, we would use the following CSS rule:

h4 + p { color: green; }

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/adjacent.html)

Note: This selector does not work in Internet Explorer 6

**Pseudo Selectors**

**An Aside About Link States**

Anchor elements are special. You can style the <a> element with an Element Type Selector, but it might not do exactly what you expect. This is because links have different states, that relate to how they are interacted with. The four primary states of a link are: link, visited, hover, active.

Pseudo selectors come in different sizes and shapes. By far the most common pseudo selectors are used to style our links. There are four different pseudo selectors to be used in conjunction with links:

**:link**

A link that has not been previously visited (visited is defined by the browser history)

**:visited**

A link that has been visited

**:hover**

A link that the mouse cursor is "hovering" over

**:active**

A link that is currently being clicked

a:link { color: red } /\* unvisited links \*/

a:visited { color: blue } /\*visited links\*/

a:hover { color: green } /\* user hovers \*/

a:active { color: lime } /\*active links \*/

For reasons of browser compatibility, you should always specify the pseudo selectors in this order. An easy way to remember this is by using the mnemonic: "**L**o**V**e **HA**!".

* [See Demo](http://web.simmons.edu/~grabiner/comm244/weekfour/pseudo.html)

**Note:** Touch screen devices do not have a hover state. See [*No Hover*](http://trentwalton.com/2010/07/05/non-hover/) by Trent Walton for more information regarding this interesting usability quandary.

You can read more about the other types of pseudo selectors on the [sitepoint page](http://reference.sitepoint.com/css/pseudoclasses).

**Universal Selector**

Matches every element on the page.

For example, if we wanted **every element** to have a solid 1px wide border, we would use the following CSS rule:

\* { border: 1px solid blue;}

* [See Demo 1](http://web.simmons.edu/~grabiner/comm244/weekfour/universal.html)

For reasons that are likely obvious after the previous example, you should be careful with universal selectors. When might you want to use them?

The answer is, not often. But an example would be to **set the margins and padding for all elements** on the page to zero. *We'll learn a better way to do this shortly.*

\* {

margin: 0;

padding: 0;

}

**Example: 1**

<html>

<head>

<title>internal style sheets</title>

<style>

h1

{

height:20px;

width:200px;

color:white;

background-color:blue;

}

h2,p

{

font-style:italic;

font-weight:normal;

font-size:20px;

}

</style>

</head>

<body>

<h1>BDPS</h1>

<h1 style="background-color-green">BDPS</h1>

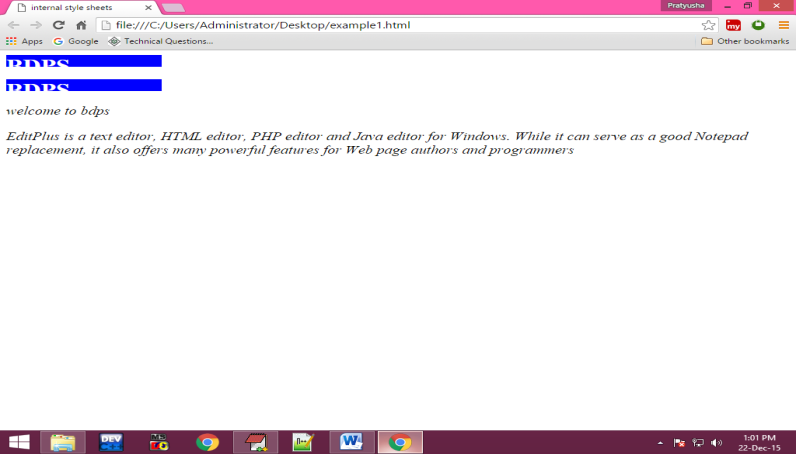
<h2>welcome to bdps</h2>

<p>

EditPlus is a text editor, HTML editor, PHP editor and Java editor for Windows. While it can serve as a good Notepad replacement, it also offers many powerful features for Web page authors and programmers</p>

</body>

</html>

**Output:**

**Example :2**

<html>

<head>

<title>internal style sheets</title>

<style>

div

{

height:150;

width:200;

border:solid;

overflow:auto;

transform:transform(300px,200px);

}

div:hover

{

background-color:rgba(255,0,0,0.3);

}

</style>

</head>

<body>

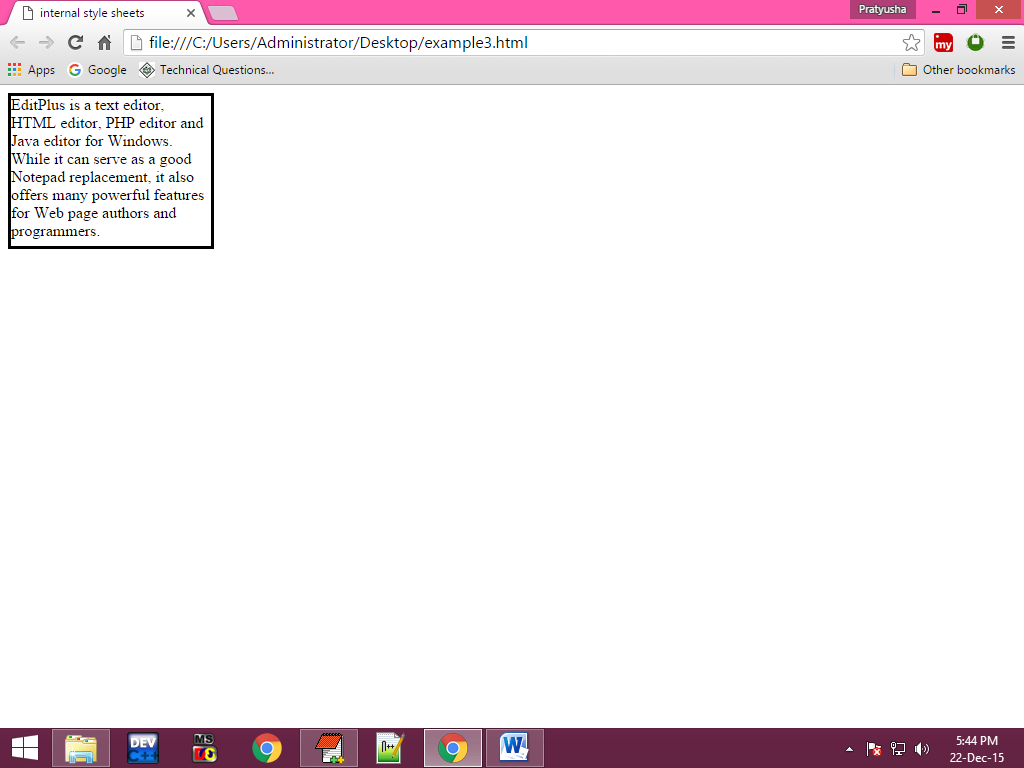
<div>

EditPlus is a text editor, HTML editor, PHP editor and Java editor for Windows. While it can serve as a good Notepad replacement, it also offers many powerful features for Web page authors and programmers.

</div>

</body>

</html>

**Output:**

**Example :4**

<html>

<head>

<title>internal style sheets</title>

<style>

#odd

{

background-color:wheat;

}

.even

{

background-color:blue;

}

.styles

{

font-size:20px;

text-align:center;

font-style:italic;

}

</style>

</head>

<body>

<table border="2" height="200" width="200" align="center">

<tr id="odd" class="styles">

<td>1</td>

<td>2</td>

<td>3</td>

</tr>

<tr class="even styles">

<td>1</td>

<td>2</td>

<td>3</td>

</tr>

<tr id="odd" class="styles">

<td>1</td>

<td>2</td>

<td>3</td>

</tr>

<tr class="even styles">

<td>1</td>

<td>2</td>

<td>3</td>

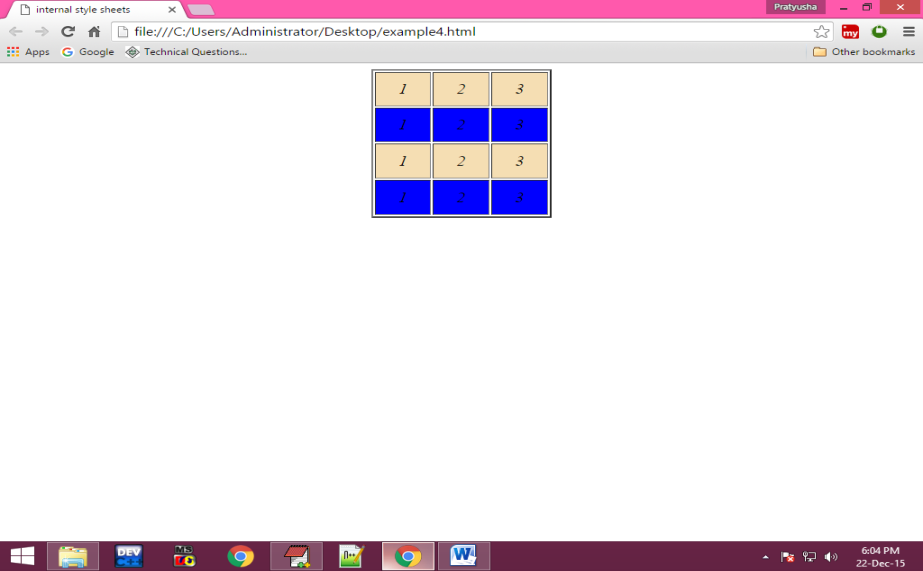
</tr>

</table>

</body>

</html>

**Output:**



**Example: 5**

<html>

<head>

<title>internal style sheets</title>

<style>

.second li

{

font-size:20px;

display:inline;

}

</style>

</head>

<body>

<ol>

<li>Home</li>

<li>About</li>

<li>Contact</li>

<li>register</li>

<ol>

<ul class="second">

<li>Home</li>

<li>About</li>

<li>Contact</li>

<li>register</li>

</ul>

<ul>

<li>Home</li>

<li>About</li>

<li>Contact</li>

<li>register</li>

</ul>

<ul>

<li>Home</li>

<li>About</li>

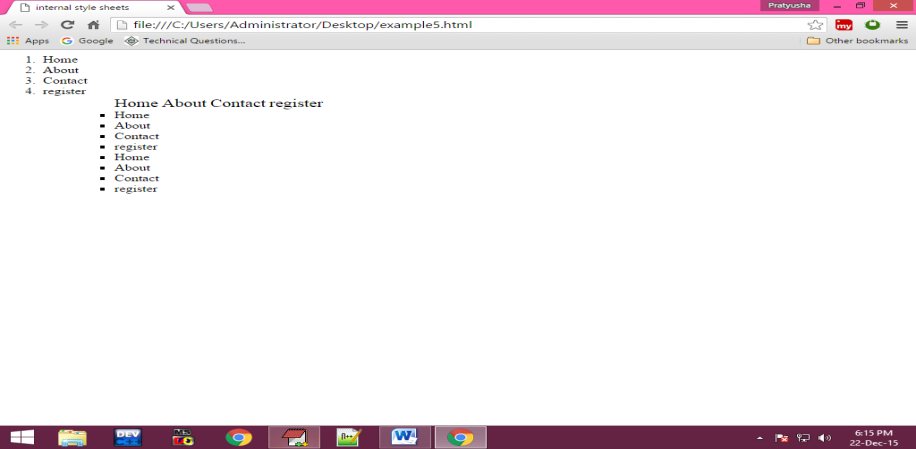
<li>Contact</li>

<li>register</li>

</ul>

</body>

</html>

**Output:**

**Example :6**

<html>

<head>

<title>internal style sheets</title>

<style>

#text1

{

position:absolute;

left:50px;

top:150px;

}

#il

{

box-shadow:5px 5px 5px black;

}

</style>

</head>

<body>

<div style="height:100px;width:100px;border:solid" id="il">

<img src="E:\my html notes\my notes\html images\2.jpg" height="200%" width="200%"/>

</div>

<div id="text1">

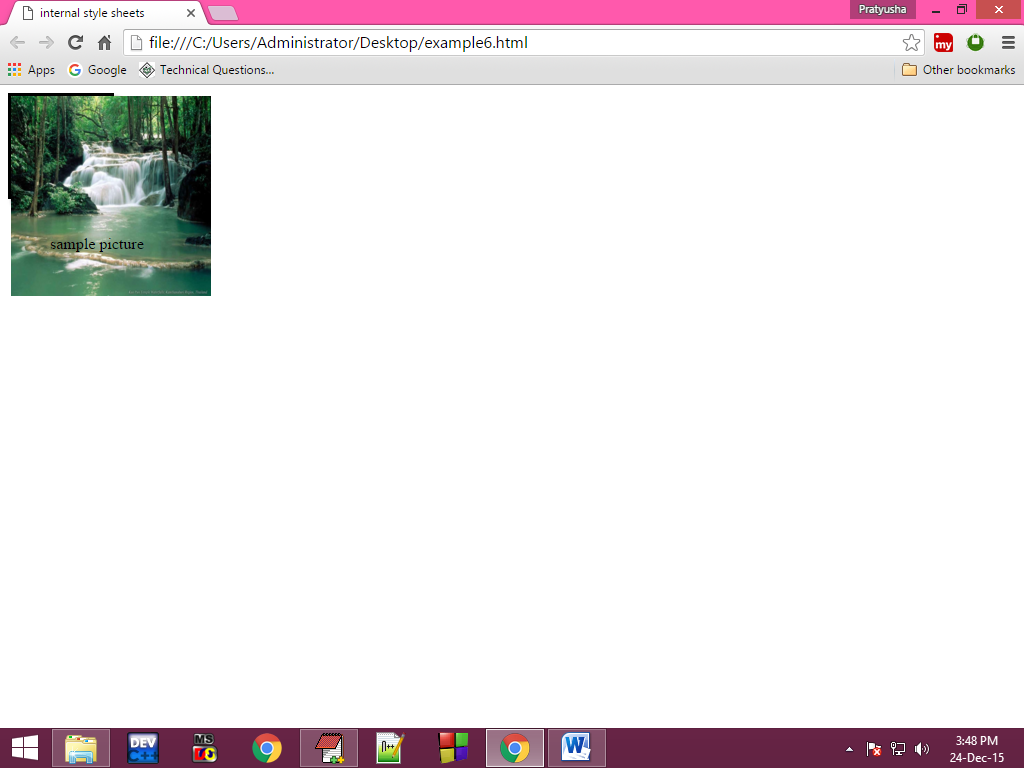
sample picture

</div>

</body>

</html>

**Output:**



**Example :7**

<html>

<head>

<title>internl style sheets</title>

</head>

<body>

<div style="height:200px;width:200px;border:solid;position:absolute;right:30px;top:300px">

<div style="background-color:blue">

<p>chat</p>

</div>

<div style="background-color:gray;height:100px;overflow:auto">

<ul>

<li>1</li>

<li>2</li>

<li>3</li>

<li>4</li>

<li>5</li>

<li>6</li>

<li>7</li>

<li>8</li>

<li>9</li>

<li>10</li>

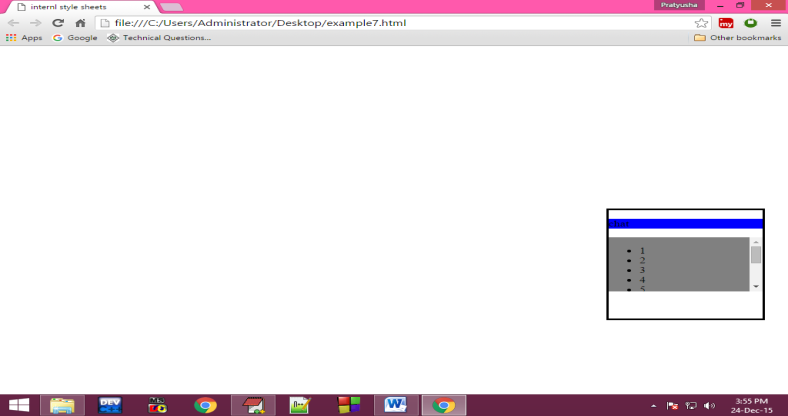
</ul>

</div>

</body>

</html>

**Output:**



**Example 8:**

<html>

<head>

<title>internal style sheets</title>

<style>

#back

{

height:200px;

width:250px;

border:solid orange;

position:absolute;

left:300px;

top:100px;

background-color:red

}

#front

{

height:20;

width:20;

border:solid orange;

transform:rotate(45deg);

position:absolute;

left:480px;

top:90px;

background-color:orange;

}

</style>

</head>

<body>

<div id="front">

</div>

<div id="back">

welcome to web page

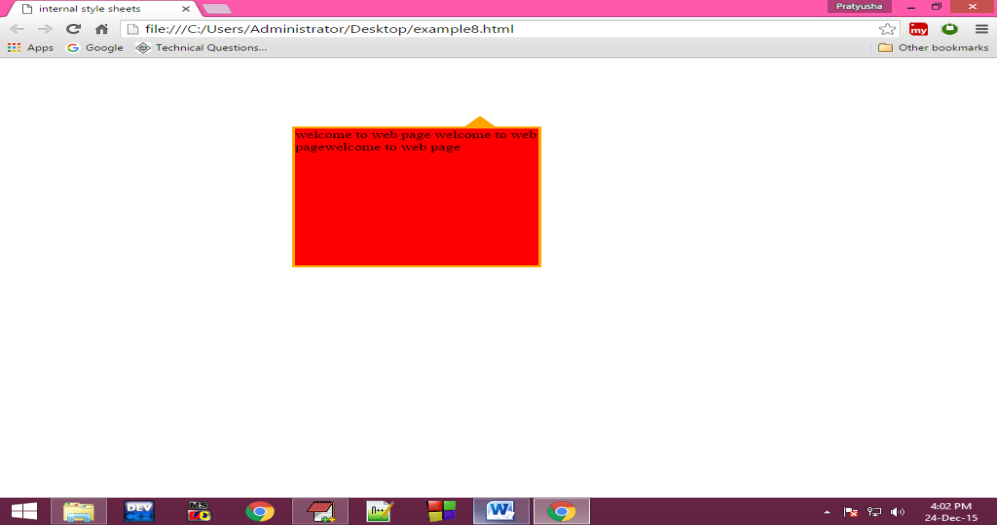
welcome to web pagewelcome to web page

</div>

</body>

</html>

**Output:**

  
**External style sheets:**

This is also used to apply the common style values in all the web pages then we use external style sheets. Here we can write the common style values in one css file and then use that css file in the html document.

To use css document we can use a tag ie; <link> tag.

**Attributes of the link tag:**

**Href:**

This attribute is to specify the destination css file.

**Syntax:**

Href=”path of the file”

**Rel:**

This attribute is used to specify the relation between css document and html document.

**Syntax:**

rel=”stylesheet”

**Type:**

By using this we can specify mime [multipurpose internet mail extension] type or internet media type.

**Syntax:**

<link href=”css file path” rel=”stylesheet” type=”text/css”/>

**Example:**

**Menu.html:**

**<**html>

<head>

<title>external style sheets</title>

</head>

<link href="menu.css" rel="stylesheet" type="text/css">

<body>

<center>

<ul>

<li><a href="#">file</a></li>

<li><a href="#">edit</a></li>

<li><a href="#">firstdocument</a></li>

<li><a href="#">search</a></li>

<li><a href="#">view</a></li>

<li><a href="#">project</a></li>

</ul>

</center>

</body>

</html>

**Main.css**

ul li

{

display:inline;

padding:60;

}

ul

{

background-color:black;

color:white;

padding-top:10;

height:30;

}

a:link

{

color:white;

text-decoration:none;

}

a:hover

{

background-color:orange;

padding:5;

}

a:active,visited

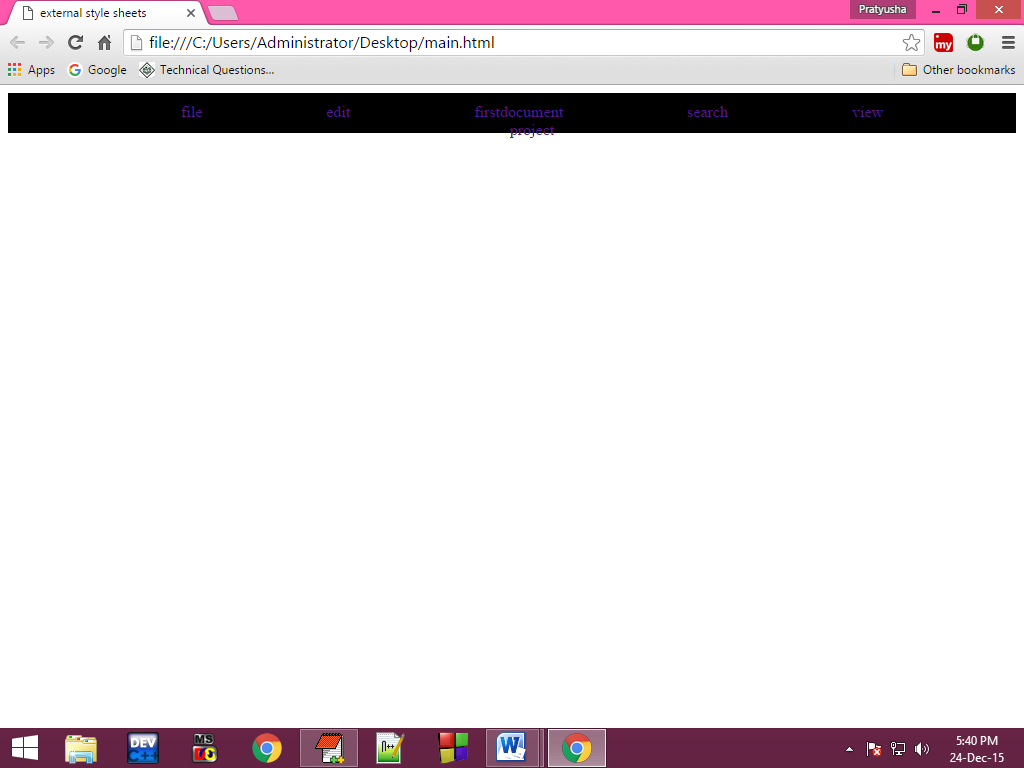
{

color:red;

font-size:25;

}

**Output:**



**Program:**

<html>

<head>

<title>animations</title>

<style>

div

{

height:75px;

width:100px;

border:solid;

animation:myanimation 10s;

-webkit-animation:second se 2;

}

@keyframes myanimation

{

from{background-color:red;}

to{background-color:gray:height:100px;width:200px;font-size:28px;}

}

@-webkit-keyframes second

{

from{background-color:blue;}

to{background-color:red;height:100px;width:200px;font-size:28px;}

}

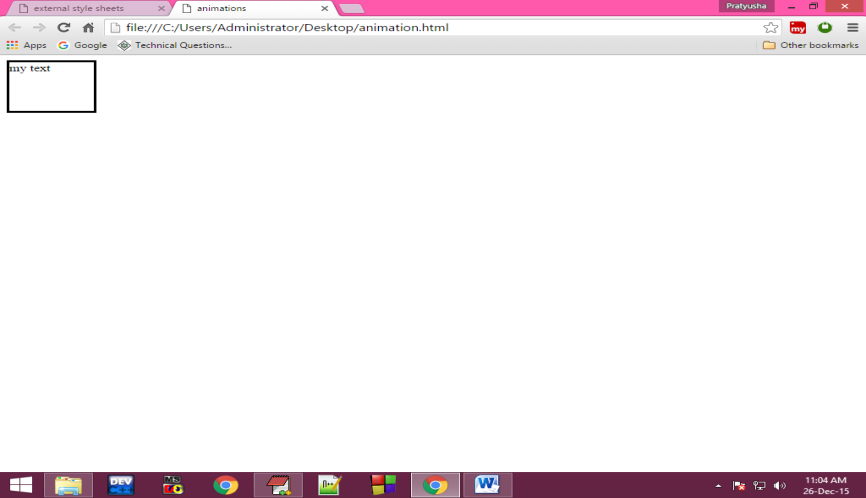
</style>

</head>

<body>

<div>

my text

</div>

</body>

</html>

**Output:**

**CSS BOX Model: -**

The **CSS box model** module defines the rectangular boxes, including their padding and margin, that are generated for elements and laid out according to the visual formatting model.

**Box model overview**

A box in CSS consists of a content area, which is where any text, images, or other HTML elements are displayed. This is optionally surrounded by padding, a border, and a margin, on one or more sides. The box model describes how these elements work together to create a box as displayed by CSS. To learn more about it read Introduction to the CSS box model.

### [**Box-edge keywords**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#box-edge_keywords)

The Box Model specification defines a set of keywords that refer to the edges of each part of the box, these are used as keyword values in CSS including as a value for the [**box-sizing**](https://developer.mozilla.org/en-US/docs/Web/CSS/box-sizing) property, to control how the box model calculates its size.

### [**content-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#content-box)

The edge of the content area of the box.

### [**padding-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#padding-box)

The edge of the padding of the box, if there is no padding on a side then this is the same as content-box.

### [**border-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#border-box)

The edge of the border of the box, if there is no border on a side then this is the same as padding-box.

### [**margin-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#margin-box)

The edge of the margin of the box, if there is no margin on a side then this is the same as border-box.

### [**stroke-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#stroke-box)

In SVG refers to the stroke bounding box, in CSS treated as content-box.

### [**view-box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#view-box)

In SVG refers to the nearest SVG viewport element's origin box, which is a rectangle with the width and height of the initial SVG user coordinate system established by the [viewBox](https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/viewBox) attribute for that element. In CSS treated as border-box.

### [**Properties for controlling the margin of a box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#properties_for_controlling_the_margin_of_a_box)

Margins surround the border edge of a box, and provide spacing between boxes.

### [**margin**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin)

### [**margin-bottom**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin-bottom)

### [**margin-left**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin-left)

### [**margin-right**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin-right)

### [**margin-top**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin-top)

### [**margin-trim**](https://developer.mozilla.org/en-US/docs/Web/CSS/margin-trim)

### [**Properties for controlling the padding for a box**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_box_model#properties_for_controlling_the_padding_for_a_box)

Padding is inserted between the content edge and border edge of a box.

### [**padding**](https://developer.mozilla.org/en-US/docs/Web/CSS/padding)

### [**padding-bottom**](https://developer.mozilla.org/en-US/docs/Web/CSS/padding-bottom)

### [**padding-left**](https://developer.mozilla.org/en-US/docs/Web/CSS/padding-left)

### [**padding-right**](https://developer.mozilla.org/en-US/docs/Web/CSS/padding-right)

### [**padding-top**](https://developer.mozilla.org/en-US/docs/Web/CSS/padding-top)

### **box-shadow**

The box-shadow [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS) property adds shadow effects around an element's frame. You can set multiple effects separated by commas. A box shadow is described by X and Y offsets relative to the element, blur and spread radius, and color.

The box-shadow property enables you to cast a drop shadow from the frame of almost any element. If a [border-radius](https://developer.mozilla.org/en-US/docs/Web/CSS/border-radius) is specified on the element with a box shadow, the box shadow takes on the same rounded corners. The z-ordering of multiple box shadows is the same as multiple [text shadows](https://developer.mozilla.org/en-US/docs/Web/CSS/text-shadow) (the first specified shadow is on top).

**1.text-shadow**

**2.box-shadow**

[**Box-shadow generator**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_backgrounds_and_borders/Box-shadow_generator) is an interactive tool allowing you to generate a **box-shadow.**

## [Syntax](https://developer.mozilla.org/en-US/docs/Web/CSS/box-shadow#syntax)

**CSS**

/\* Keyword values \*/

box-shadow: none;

/\* A color and two length values \*/

/\* <color> | <length> | <length> \*/

box-shadow: red 60px -16px;

/\* Three length values and a color \*/

/\* <length> | <length> | <length> | <color> \*/

box-shadow: 10px 5px 5px black;

/\* Four length values and a color \*/

/\* <length> | <length> | <length> | <length> | <color> \*/

box-shadow: 2px 2px 2px 1px rgb(0 0 0 / 20%);

/\* inset, length values, and a color \*/

/\* <inset> | <length> | <length> | <color> \*/

box-shadow: inset 5em 1em gold;

/\* Any number of shadows, separated by commas \*/

box-shadow:

3px 3px red inset,

-1em 0 0.4em olive;

/\* Global values \*/

box-shadow: inherit;

box-shadow: initial;

box-shadow: revert;

box-shadow: revert-layer;

box-shadow: unset;

# **display**

The display [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS) property sets whether an element is treated as a [**block** or **inline** box](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_flow_layout) and the layout used for its children, such as [**flow layout**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_flow_layout), [**grid**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_grid_layout) or [**flex**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_flexible_box_layout).

Formally, the display property sets an element's inner and outer ***display types***. The outer type sets an element's participation in [**flow layout**](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_flow_layout)**;** the inner type sets the layout of children. Some values of display are fully defined in their own individual specifications; for example, the detail of what happens when display: flex is declared is defined in the CSS Flexible Box Model specification.

## [Syntax](https://developer.mozilla.org/en-US/docs/Web/CSS/display#syntax)

/\* pre composed values \*/

display: block;

display: inline;

display: inline-block;

display: flex;

display: inline-flex;

display: grid;

display: inline-grid;

display: flow-root;

/\* box generation \*/

display: none;

display: contents;

/\* multi-keyword syntax \*/

display: block flex;

display: block flow;

display: block flow-root;

display: block grid;

display: inline flex;

display: inline flow;

display: inline flow-root;

display: inline grid;

/\* other values \*/

display: table;

display: table-row; /\* all table elements have an equivalent CSS display value \*/

display: list-item;

/\* Global values \*/

display: inherit;

display: initial;

display: revert;

display: revert-layer;

display: unset;

# **Basic concepts of flexbox**

The flexible box layout module, usually referred to as flexbox, was designed as a one-dimensional layout model, and as a method that could offer space distribution between items in an interface and powerful alignment capabilities. This article gives an outline of the main features of flexbox, which we will be exploring in more detail in the rest of these guides.

When we describe flexbox as being one-dimensional we are describing the fact that flexbox deals with layout in one dimension at a time — either as a row or as a column. This can be contrasted with the two-dimensional model of CSS Grid Layout, which controls columns and rows together.

## The two axes of flexbox

When working with flexbox you need to think in terms of two axes — the main axis and the cross axis. The main axis is defined by the [flex-direction](https://developer.mozilla.org/en-US/docs/Web/CSS/flex-direction) property, and the cross axis runs perpendicular to it. Everything we do with flexbox refers back to these axes, so it is worth understanding how they work from the outset.

### **The main axis**

The main axis is defined by flex-direction, which has four possible values:

* row
* row-reverse
* column
* column-reverse

# **opacity**

The opacity [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS) property sets the opacity of an element. Opacity is the degree to which content behind an element is hidden, and is the opposite of transparency.

# [**Syntax**](https://developer.mozilla.org/en-US/docs/Web/CSS/opacity#syntax)

**CSS**

opacity: 0.9;

opacity: 90%;

/\* Global values \*/

opacity: inherit;

opacity: initial;

opacity: revert;

opacity: revert-layer;

opacity: unset;

# [**Values**](https://developer.mozilla.org/en-US/docs/Web/CSS/opacity#values)

[**<alpha-value>**](https://developer.mozilla.org/en-US/docs/Web/CSS/opacity#alpha-value)

A [**<number>**](https://developer.mozilla.org/en-US/docs/Web/CSS/number) in the range 0.0 to 1.0, inclusive, or a [**<percentage>**](https://developer.mozilla.org/en-US/docs/Web/CSS/percentage) in the range 0% to 100%, inclusive, representing the opacity of the channel (that is, the value of its alpha channel). Any value outside the interval, though valid, is clamped to the nearest limit in the range.

| **Value** | **Meaning** |
| --- | --- |
| 0 | The element is fully transparent (that is, invisible). |
| Any [<number>](https://developer.mozilla.org/en-US/docs/Web/CSS/number) strictly between 0 and 1 | The element is translucent (that is, content behind the element can be seen). |
| 1 (default value) | The element is fully opaque (visually solid). |

# [**Description**](https://developer.mozilla.org/en-US/docs/Web/CSS/opacity#description)

**opacity** applies to the element as a whole, including its contents, even though the value is not inherited by child elements. Thus, the element and its children all have the same opacity relative to the element's background, even if they have different opacities relative to one another.

To change the opacity of a background only, use the [**background**](https://developer.mozilla.org/en-US/docs/Web/CSS/background) property with a [**color** value](https://developer.mozilla.org/en-US/docs/Web/CSS/color_value) that allows for an alpha channel. For example:

CSs

**background: rgb(0 0 0 / 40%);**

# [**Transitioning opacity**](https://developer.mozilla.org/en-US/docs/Web/CSS/opacity#transitioning_opacity)

When [transitioning](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_transitions) the opacity of elements as you add them to the page when content was formerly hidden with [visibility: hidden](https://developer.mozilla.org/en-US/docs/Web/CSS/visibility#hidden), [display: none](https://developer.mozilla.org/en-US/docs/Web/CSS/display#none), or [content-visibility: hidden](https://developer.mozilla.org/en-US/docs/Web/CSS/content-visibility#hidden), you need to include both a [@starting-style](https://developer.mozilla.org/en-US/docs/Web/CSS/@starting-style) and [transition-behaviour: allow-discrete](https://developer.mozilla.org/en-US/docs/Web/CSS/transition-behavior#allow-discrete):

.card {

transition:

opacity 5s,

display 5s;

background-color: orange;

transition-behavior: allow-discrete;

@starting-style {

opacity: 0;

}

}

.card.hidden {

display: none;

opacity: 0;

}

To enable first-style transitions, @starting-style rules are needed. In the above code, setting opacity: 0 in @starting-style provides a starting point for the transition when the element receives its initial style update. For more details, see [@starting-style](https://developer.mozilla.org/en-US/docs/Web/CSS/@starting-style).

Setting transition-behavior: allow-discrete is required to transition to display: none. See the [transition-behavior](https://developer.mozilla.org/en-US/docs/Web/CSS/transition-behavior) property for more details.