

Subbu Sir's
iGate Solutions
Near Manasa Theatre, Kavali.



**THE ULTIMATE
WEB DEVELOPMENT COURSE**
PART 1

HTML (Hypertext Markup Language)

History:

- Tim Berners-Lee from CERN Developed HTML to use and share research documents among the scientists.
- HTML is influenced by Standard Generalized Markup Language (SGML)
- It has been used as an internet programming language from late 1991
- Now, every web page on internet is developed using HTML and its associated technologies like CSS, Java script and XML etc.

HTML is a web based formatting language, where different tags are used to format different elements of the web page. Here the tag is a keyword place in angular brackets <>. However some tags have opening and closing tags called container tags and tags that have only opening tag with self-closure are called singular tags.

Example

```
<strong>
    .....
</strong>

<br/>           <!--self-closed singular tag-->
```

Structure of a HTML program

```
<!-- Structure of a html program-->
<!DOCTYPE ...>
<html>
    <head>
        <title>-----</title>
    </head>
    <body>
        -----
        </body>
</html>
```

HTML Tags

<!DOCTYPE>

It is an instruction to the web browser about what version of HTML the web page is currently being used. The browser evaluates the code according to the HTML version used within the page. It is to be written in the very first line of any HTML page. <!DOCTYPE> for different versions of HTML are

HTML 4.1 Strict	<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	This DTD contains all HTML elements and attributes, but does NOT INCLUDE deprecated elements (like font). Framesets are not allowed.
HTML 4.01 Transitional	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	This DTD contains all HTML elements and attributes, INCLUDING deprecated

		elements (like font). Framesets are not allowed.
HTML 4.01 Frameset	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">	This DTD is equal to HTML 4.01 Transitional, but allows the use of frameset content.
HTML 5.0	<!DOCTYPE html>	This DTD is equal to HTML 5.0 and it simply means "this page is written in HTML5"

 [Break]

The
 tag inserts a single line break. It is an empty tag which means that it has no end tag. So, it is self-closed.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
  <body>  
    Igate Solutions <br/> Near Manasa Theatre <br/> Kavali  
  </body>  
</html>
```

<p> [Paragraph]

The <p> tag defines a paragraph. Browsers automatically add some space (paragraph space) before and after each <p> element. The margins can be modified with CSS (with the margin properties)

ATTRIBUTE	VALUE	EXAMPLE
align	Center Right Left Justify	<p align="right">....</p>

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
  <body>  
    <p>igate solutions</p>  
    <p align="center">Near Manasa Theatre</p>  
    <p align="right">Kavali</p>  
  </body>  
</html>
```

Comments in HTML

```
<!--xxxxxxxxxxxxxxxxxxxxxx -->
```

<h1> to <h6> [Headings]

The <h1> to <h6> tags are used to define HTML headings. <h1> defines the most important heading, <h6> defines the least important heading.

Note: Search engines index the information on the webpage prioritizing the tags from <h1> to <h6>.

ATTRIBUTE	VALUE	EXAMPLE
Align	Center Right Left Justify	<h1 align="right">...</h1> <h4 align="center">...</h4>

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <h1 align="center">igate solutions</h1>
  </head>
  <body>
    <h1>Address</h1>
    <h2>Behind Manasa Theatre</h2>
    <h3>Kavali</h3>
    <h4>SPSR Nellore</h4>
    <h5>AP</h5>
    <h6>India</h6>
  </body>
</html>
```

Formatting Content Tags

: Bold (Deprecated in HTML5)
<i>: Italic (Deprecated in HTML5)
<u>: Underline (Deprecated in HTML5)
<strike> : Strikeout (Deprecated in HTML5)
 : Equals to Bold (Recommended in HTML5)
 : Equals to Italic (Recommended in HTML5)

Example

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    Formatting Content tags in HTML<br/>
    <b>Igate Solutions</b><br/>
    <i>Near Manasa Theatre</i><br/>
```

```
<u>Kavali</u><br/>
<strike>Hello World</strike><br/>
<strong>IGate Solutions</strong><br/>
<em>Near Manasa Theatre</em><br/>
</body>
</html>
```

Nesting of Tags:

One HTML tag can be placed within another HTML tag. So that multiple formats can be applied to the content. But Make sure that, first opened tag must be closed last.

Example

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    <strong><em>
      IGate Solutions
    </em></strong>
    <p><strong>IGate Solutions</strong></p>
  </body>
</html>
```

Color Schemes in HTML

In HTML we can set the color of an element using major four different types of color schemes that are

Color names

Red, green, blue, black, white,

Color mix with `rgb()` function

Here the combination of red, green and blue constants ranges from 0 to 255 need to be specified to create a mixed color

`Rgb(red,green,blue)`

Examples

`rgb(255,0,0)` - red ; `rgb(0,255,0)` - green ; `rgb(0,0,255)` - blue; `rgb(0,0,0)` - black ;
`rgb(255,255,255)` - white

Color mix with `rgba()` function

RGBA color values are an extension to RGB color values with an alpha channel - which specifies the opacity of the object. An RGBA color value is specified with: `rgba(red, green, blue, alpha)`. The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

Example:

`Rgba(255,0,0,0.3)`

Color mix with Hexadecimal constant

Here the combination of red, green and blue constants ranges from 00 to FF need to be specified to create a mixed color. A prefix character # must be preceded with the color mix

#**FFFFFF**

 Red green blue

Examples

#FF0000 – red ; #00FF00 – green ; #0000FF – blue ; #000000 – black ; #FFFFFF - white

To select a color visit : <http://www.colorhexa.com/>

<center>

It is a container tag used align the text or any object to the center of the container. It is deprecated by HTML5

```
<center>
Igate Solutions
</center>
```

 [Formatting font]

The tag is used to format the font name, size and color. The default font name is chosen by the browser itself if font face is not specified. Generally, browsers choose generic font families like 'Times New Roman' and 'Georgia' if we don't specify any font family.

The default size of the is '3' in abstract. It equals to '12pt' (3X4) in point units. If we specify the font size as 6 then it would result (6x4) pt.

Note: It is not supported in HTML5, instead using CSS is recommended.

ATTRIBUTE	VALUE	EXAMPLE
Color	rgb(x,x,x) #xxxxxx colorname (e.g., red,green,blue)	...
face	Font_family (e.g., Verdana, Courier)	...
Size	An integer in points	... Or ...

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<body>
<center>
  <font size="6" face="arial" color="green">
    Igate Solutions
  </font>
</center>
<p>
  <font face="verdana" size="3">
    Near Manasa Theatre<br/>
    Kavali
  </font>
</p>
</body>
</html>
```

Note: The font tag is no more supported by modern browsers

 [Ordered list]

The tag defines an ordered list. An ordered list can be numerical or alphabetical. We use the tag to define list items.

ATTRIBUTE	VALUE	EXAMPLE
reversed (HTML5) (not supported in IE browser)	Reversed	<ol reversed>...
start (supported by all browsers)	Any integer	<ol start="50">...
type	1, A, a, i, I	<ol type="I">
value ()	1, A, a, i, I	<li value="20">...

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <h1 align="center">igate solutions</h1>
  </head>
  <body>
    <ol start="10">
      <li>iphone</li>
      <li>ipad</li>
      <li value="20">iPod</li>
      <li>iMac</li>
    </ol>
    <ol type="A">
      <li>iphone</li>
      <li>ipad</li>
      <li>iPod</li>
      <li>iMac</li>
    </ol>
  </body>
</html>
```

 [Un-ordered List]

The tag defines an unordered (bulleted) list. Use the tag together with the tag to create unordered lists.

ATTRIBUTE	VALUE	EXAMPLE
Type (not supported in HTML 5)	Disc Square circle	<ul type="square">...

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
```

```

<ul type="disc">
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ul>
<ul type="square">
<li>iphone</li>
<li>ipad</li>
<li>iPod</li>
<li>iMac</li>
</ul>
</body>
</html>

```

<dl> [Definition Lists]

The `<dl>` tag defines a description/definition list. `<dt>` and `<dl>` are used in conjunction with `<dl>`. Here `<dt>` is used to define definition title and `<dd>` used to define definition details.

Example:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    <dl>
      <dt>Coffee</dt>
      <dd>Black hot drink</dd>
      <dt>Milk</dt>
      <dd>White cold drink</dd>
    </dl>
  </body>
</html>

```

<table>

It is a tag mechanism used to define a data table with rows and columns. Sometimes it is also used to design layout of a web page. Other tags like `<tr>`, `<td>`, `<th>` are used in conjunction with `<table>` tag.

Here `<tr>` tag is used to define a row, `<td>` is used to define a data element, `<th>` is used to define heading level data element in bold format.

A more complex HTML table may also include `<caption>`, `<col>`, `<colgroup>`, `<thead>`, `<tfoot>`, and `<tbody>` elements.

Note: All table attributes are discouraged in HTML5 as it encourages the usage of CSS instead.

ATTRIBUTE	VALUES	EXAMPLE
align	Left, Center, Right	<table align="right"> <tr align="right"> <td align="right">
bgcolor	Color name (e.g., red, green, magenta...) Hexa code (e.g., #009aae) RGB values (251,010,36)	<table bgcolor="#00FF00"> <tr bgcolor="#00FF00"> <td bgcolor="#00FF00">

Border	Numbers (e.g., 0,1,2,3...)	<table border="1">
Cellpadding	Pixels (e.g., 2px)	<table cellpadding="10">
Cellspacing	Pixels	<table cellspacing="10">
Rules	None, Groups, Rows, cols, All	<table rules="rows">
Width	Pixels(e.g., 200px, 720px) % (e.g., 40%, 80%)	<table width="400px">
Height	Pixels(e.g., 400px, 540px) % (e.g., 40%, 80%)	<table height="400px">

Example: Drawing basic table

Version	Year	Who
HTML 1.0	1991	Berners Lee
HTML 2.0	1995	W3C
HTML 3.0	1998	W3C
HTML 4.0	1999	W3C
XHTML 1.0	2000	W3C
HTML 5.0	2004	WHATWG
XHTML 2.0	2009	W3C

Time Line of HTML

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    <table align="center" cellpadding="7" width="600px" cellspacing="0"
border="1">
      <caption align="bottom">Time Line of HTML</caption>
      <tr bgcolor="#cccccc">
        <th>Version</th>
        <th>Year</th>
        <th>Who</th>
      </tr>
      <tr>
        <td>HTML 1.0</td>
        <td>1991</td>
        <td>Berners Lee</td>
      </tr>
      <tr>
        <td>HTML 2.0</td>
        <td>1995</td>
        <td>W3C</td>
      </tr>
      -----
    
```

```

        </table>
    </body>
</html>

```

Example 2: Drawing table with row and column spans.

Year	Operating Systems		Most Sales
	Android	iOS	
2008	Android 1.0	iPhone OS 1.0-2.0	Android
2009	Android 1.1-2.0	iPhone OS 3.0	iOS
2010	Android 2.1-2.2	iPhone OS 3.1-4.0	iOS
2011	Android 2.3-2.3.4	iOS 4.2-5.0	Android
2012	Android 2.3.5-4.2	iOS 6.0	Android
2013	Android 4.2-4.4.2	iOS 7.0	-Equal-
2014	Android 4.4.3-5.0.2	iOS 8.0	iOS
2015	Android 5.1-5.1.1	iOS 8.1.3-9.0 beta	Android

Timeline of Smartphone Operating Systems

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
    <body>
        <table align="center" cellpadding="7" width="600px" cellspacing="0"
border="1">
            <caption align="bottom"><br/>Timeline of Smartphone Operating
Systems</caption>
            <tr bgcolor="#cccccc">
                <th rowspan="2">Year</th>
                <th colspan="2">Operating Systems</th>
                <th rowspan="2">Most Sales</th>
            </tr>
            <tr align="center">
                <td>Android</td>
                <td>iOS</td>
            </tr>
            <tr>
                <td>2008</td>
                <td>Android 1.0</td>
                <td>iPhone OS 1.0-2.0</td>
                <td>Android</td>
            </tr>
            -----
            </table>
        </body>
    </html>

```

Example 3: Drawing table with ordered and unordered lists

iPhone	OSX	MacBook
5. iPhone 6. iPhone 2 7. iPhone 3 8. iPhone 3S 9. iPhone 4 10. iPhone 4S 11. iPhone 5 12. iPhone 5S 13. iPhone 6 14. iPhone 6 Plus	<ul style="list-style-type: none"> • Kodiak • Cheetah • Puma • Jaguar • Panther • Tiger • Leopard • Snow Leopard • Lion • Mountain Lion • Mavericks • Yosemite • El Captain 	MacBook 12-inch version MacBook Air 11-inch version MacBook Air 13-inch version MacBook Pro 13-inch, 2.3GHz version MacBook Pro 13-inch, 2.5GHz version MacBook Pro 15-inch, 2.5GHz version
hello	world	welcome

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
    <body>
        <table border="1" cellspacing="0px" cellpadding="3px" width="600px"
height="auto" align="center">
            <thead bgcolor="#dddddd">
                <th>iPhone</th>
                <th>OSX</th>
                <th>MacBook</th>
            </thead>
            <tbody>
                <tr>
                    <td>
                        <ol start="5" type="1">
                            <li>iPhone</li>
                            <li>iPhone 2</li>
                            <li>iPhone 6</li>
                            <li>iPhone 6 Plus</li>
                        </ol>
                    </td>
                    <td>
                        <ul type="disc">
                            <li>Kodiak</li>
                            <li>Cheetah</li>
                            <li>Mountian Lion</li>
                            <li>Mavericks</li>
                            <li>Yosemite</li>
                            <li>El Captain</li>
                        </ul>
                    </td>
                    <td>
                        <dl>
                            <dt>MacBook</dt>
                            <dd>12-inch version</dd>
                            <dt>MacBook Air</dt>
                            <dd>11-inch version</dd>
                        </dl>
                    </td>
                </tr>
            </tbody>
        </table>
    </body>
</html>
```

```

        <dt>MacBook Pro</dt>
            <dd>13-inch, 2.5GHz version</dd>
        <dt>MacBook Pro</dt>
            <dd>15-inch, 2.5GHz version</dd>
    </dl>
    </td>
</tr>
</tbody>
<tfoot align="center" bgcolor="yellow">
    <td>hello</td>
    <td>world</td>
    <td>welcome</td>
</tfoot>
</table>
</body>
</html>

```

Example: Table rules

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
    <body>
        <table align="center" cellpadding="7" width="600px" cellspacing="0"
border="1" rules="none">
            <caption align="bottom">Time Line of HTML</caption>
            <tr bgcolor="#cccccc">
                <th>Version</th>
                <th>Year</th>
                <th>Who</th>
            </tr>
            <tr>
                <td>HTML 1.0</td>
                <td>1991</td>
                <td>Berners Lee</td>
            </tr>
            <tr>
                <td>HTML 2.0</td>
                <td>1995</td>
                <td>W3C</td>
            </tr>
        </table>
    </body>
</html>

```

Version	Year	Who
HTML 1.0	1991	Berners Lee
HTML 2.0	1995	W3C

Time Line of HTML

Notes

The tag is used to embed an image in a HTML page. It has two required attributes that are **src** and **alt**. **src** specifies the source of image and **alt** is the text that would be displayed on hover on the image.

We can load images belongs to different formats like **.bmp**, **.jpg**, **.png**, **.gif** according to the required size and quality. Here the higher the color depth (that is, the more bits used to uniquely describe the color of each pixel), more the quality is.

Name	Color Depth	Compression	Animation	Transparency
Graphics Interchange Format (GIF)	8-bit (256 colors)	Lossless	Yes	Yes
Joint Photographic Experts Group (JPEG, JPG)	24-bit (1.6 million colors)	Lossy	No	No
Portable Network Graphics (PNG)	24-bit or 48-bit	Lossless	Yes	Yes

Note: Images are not technically inserted into an HTML page, images are linked to HTML pages. The tag creates a holding space for the referenced image.

ATTRIBUTE	VALUE	EXAMPLE
align (Not supported in HTML5)	Top, Bottom, Middle, Right, Left (aligns the image in respect to the text)	
alt	Text	
border (Not supported in HTML5)	Pixels (e.g., 2px)	
crossorigin (Introduced in HTML5)	anonymous use-credentials	
height	Pixels (e.g., 400px)	
width	Pixels (e.g., 300px)	
hspace , vspace(Not supported in HTML5)	Pixels (e.g., 20px) (Gives space vertically or horizontally between the text and image)	<p>This is some text. This is some text. This is some text.</p>
Ismap	Ismap (gives coordinates as output)	
Longdesc (Not supported in HTML5)	URL	
Src	URL	

Usemap	#mapname (e.g., #planetmap)	<pre> <map name="planetmap"> <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm"> <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm"> <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm"> </map></pre>
--------	--------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
<body>

</body>
</html>
```

<a>

The **<a>** tag defines a hyperlink, which is used to link from one page to another. The most important attribute of the **<a>** element is the **href** attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red (Appears on mouse down)

Tip: Attributes like download, hreflang, media, rel, target, and type cannot be present if the href attribute is not present.

Tip: A linked page is normally displayed in the current browser window, unless you specify another target.

Tip: Use CSS to style links.

ATTRIBUTE	VALUE	EXAMPLE
Charset (Not supported in HTML5)	Char_encoding (e.g., UTF-8)	<pre>www.codi ngfox.com</pre>
Coords (Not supported in HTML5)		<pre><map name="Map1"> The Sun Mercury Venus </map></pre>

Download	Filename (optional)	<code></code>
Href	URL	<code>Visit CodingFox</code>
Hreflang	Language_code (e.g.,en, ru, ph, jp, ...)	<code>CodingFox</code>
Media (Not supported in HTML5. Use the global id attribute instead.)	Media_query	<code><p>print</p></code>
Name (Not supported in HTML5. Use the global id attribute instead.)	Section_name	<code>See also Chapter 4<h2>Chapter 4</h2><p>This chapter explains ba bla bla</p></code>
Rel	Alternate, author, bookmark, help, license, next, nofollow, dofollow(default), noreferrer, prefetch, prev, search, tag	<code>Cheap Flights</code>
Rev (Not supported in HTML5.)	Text	<code>Cheap Flights</code>
Shape (Not supported in HTML5)	Default, rect, circle, poly	<code><map name="Map1">The SunMercuryVenus</map></code>
Target	_blank, _parent, _self, _top, Framename	<code>Visit CodingFox</code>
Type	Media_type	<code>CodingFox</code>
Title	Tool tip text	<code>CodingFox</code>

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<body>
  <a href="http://www.codingfox.com">CodingFox</a>
</body>
</html>
```

Absolute Address:

Writing the total physical path of a page is called absolute path

Example:

```
<a href="http://www.codingfox.com/Chapter-1/0.5.Arrays.html">CodingFox</a>
```

Relative address:

Writing the address of the page in related with the current location

Example:

```
<a href="10.6.Arraysinput.html">CodingFox</a>
```

Referring another page in the same folder

```
<a href=". /hello.html">CodingFox</a>
```

Referring a page in the parent folder

```
<a href=". /functions/11.1.functions-introduction.html">CodingFox</a>
```

Referring the page in another folder of the same level

Opening a page in new window

Use `_blank` value as the target attribute to open the page in a new window. In case of HTML5 we do the same using JavaScript

```
<a target="_blank" href="http://www.codingfox.com/about.html">CodingFox</a>
```

Creating an e-mail link

Add `mailto:` to your mail address and set as href. It will open the local mailer program with the given address. We can also set the subject automatically by concatenating `subject=` with concatenating operator `?`

```
<a href="mailto:nicesubbusir@gmail.com?subject=comment" title="comment">mail  
me</a>
```

Linking to anchors

We can link to different sections of the same page. First we need to create anchors using the `name` attribute and then these names can be referred with the `href` attribute

```
<h2><a name="first">Chapter 4</a></h2> // defining the anchor  
<a href="#first">Arrays</a> // referring the anchor
```

Linking other files

Like a web page, even other files like .doc, .docx, .zip, .pdf, .exe can be linked using `href` attribute.

```
<a href="file:///d:/Women shoppe.docx">mail me</a>
```

Referring the local document using file access protocol

```
<a href="http://www.qc.com/docs/whitepapers.pdf">mail me</a>
```

Referring the document from the web

```
<a href="http://www.qc.com/docs/whitepapers.pdf" download>mail me</a>
```

Allowing to download the file

```
<a href="http://www.qc.com/resources/installer.exe">start install</a>
```

Allows to install an application from web server to the local machine

Example

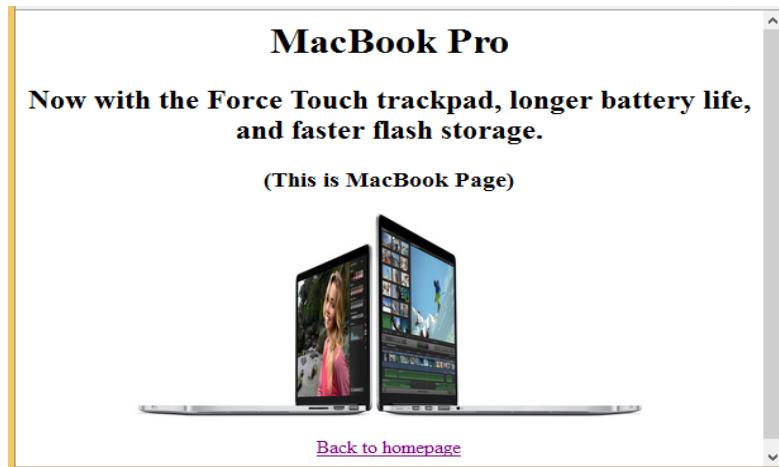
The following are the products of Apple

(This is homepage)

Click on the following links to go to that specific product page and know more details about it

- [Mac](#)
- [iPhone](#)
- [iPad](#)
- [Apple.com](#)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <title>Example for anchor tag(homepage)</title>
  </head>
  <body>
    <h1 align="center">The following are the products of Apple</h1>
    <h3 align="center">(This is homepage)</h3>
    <p>Click on the following links to go to that specific
       product page and know more details about it
    <ul>
      <li><font size="5"><a href="mac.html">Mac</a></font></li>
      <li><font size="5"><a href="iphone.html">iPhone</a></font></li>
      <li><font size="5"><a href="ipad.html">iPad</a></font></li>
      <li><font size="5">
          <a href="http://www.apple.com">Apple.com</a></font></li>
    </ul>
    </p>
  </body>
</html>
```



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <title>Details about Mac</title>
  </head>
  <body>
    <h1 align="center">MacBook Pro</h1>
    <h2 align="center">Now with the Force Touch trackpad, longer battery
life, and faster flash storage.</h2>
    <h3 align="center">(This is MacBook Page)</h3>
    </img>
    <p align="center"><a href="anchor_homepage.html">Back to
homepage</a></p>
  </body>
</html>
```

An image as a hyperlink

By keeping an `` tag within `<a>` tag, we can make an image as a hyperlink

Example:

```
<a href="http://www.apple.com">

</a>
```

Image map

Image map is an overlay on an image that defines number of hotspots according to cords we supply. These hotspots can be different dimensions like rectangle, circle and polygon. Each hot spot refers different web pages or different sections of the same page.

The position of a rectangular hotspot is defined by two points: its upper-left and lower right Corners.

```
<area shape="rect" coords="284,170,352,314" href="enter.htm">
```

To define a circular hotspot, we use three coordinates: two for the circle's center point (horizontal and vertical values), and one for the radius of the circle.

```
| <area shape="circle" coords="270,364,144" href="index.htm">
```

To define a poly hotspot, we use as many coordinates as are needed to define all the vertexes of the shape. Poly hotspots consist of straight lines that connect each of the points we define.

```
| <area shape="poly" coords="287,71,413,286,314,446,188,267" href="index.htm">
```

Image map can be defines in two steps that are

1. Defining map with hotspots with `<area>`
2. Attaching the map with image using "usemap" attribute

Example

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <body>
    
    <map name="navbar">
      <area shape="rect" coords="0,0,60,30" href="home.htm">
      <area shape="rect" coords="70,0,155,30" href="tips.htm">
      <area shape="rect" coords="165,0,250,30" href="problem.htm">
      <area shape="rect" coords="260,0,325,30" href="products.htm">
      <area shape="rect" coords="335,0,400,30" href="about.htm">
      <area shape="rect" coords="410,0,490,30" href="contact.htm">
    </map>
  </body>
</html>
```

Home

Tip & Tricks

Fix Problems

Products

About Us

Contact Us

```
<body>
```

The `<body>` tag defines the document's body. It contains all the contents of an HTML document, such as text, hyperlinks, images, tables, lists, etc.

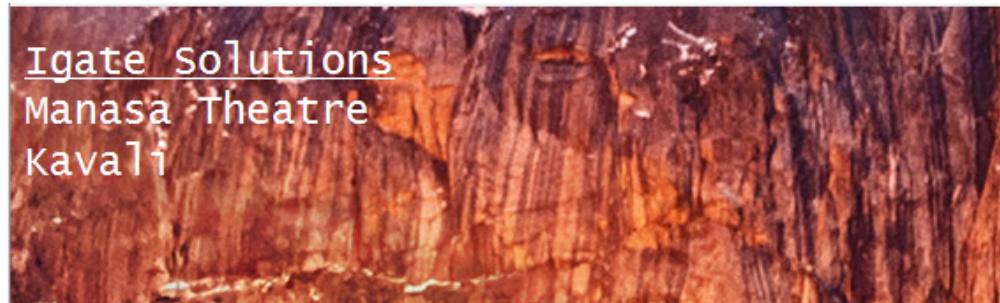
Note: All layout attributes are removed in HTML5.

ATTRIBUTE	VALUE	EXAMPLE
Alink (Not supported in HTML5)	<code>color_name hex_number rgb_number</code>	<code><body alink="green"></code>
Background (Not supported in HTML5)	<code>URL of the image</code>	<code><body background="/images/bgimage.jpg"></code>
Bgcolor (Not supported in HTML5)	<code>color_name hex_number rgb_number</code>	<code><body bgcolor="#E6E6FA"></code>
Link (Not supported in HTML5)	<code>color_name hex_number rgb_number</code>	<code><body link="blue"></code>

text (Not supported in HTML5)	<i>color_name hex_number rgb_number</i>	<body text="green">
Vlink (Not supported in HTML5)	<i>color_name hex_number rgb_number</i>	<body vlink="red">

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
    <body alink="green" background="/images/bgimage.jpg" bgcolor="#E6E6FA"
 link="white" text="white" vlink="red">
        <p> Igate Solutions <br/> Near Manasa Theatre <br/> Kavali</p>
    </body>
</html>
```



✍ Notes

Notes

Meta tags:

Metadata is the data about data. The `<meta>` tag provides metadata about the HTML document. It will not be displayed on the page, but will be used by the browsers and search engines. Meta elements are typically used to specify page description, keywords, author of the document, last modified etc.

Points to remember:

- `<meta>` tags always goes inside the `<head>` element.
- Metadata is always passed as name/value pairs.
- The content attribute MUST be defined if the name or the http-equiv attribute is defined. If none of these are defined, the content attribute CANNOT be defined.

```
| <meta charset="UTF-8">
```

It is to specify that the character set to be used by the browser while displaying the web page

```
| <meta name="description" content="Keyboards used with some platforms were missing  
some of these characters. So it was not possible to write C code on these  
machines. ">
```

It would be displayed with the search result

[Trigraph characters in C language - CodingFox](#)

codingfox.com/4-5-trigraph-characters-in-c-language/ ▾

Sep 12, 2013 - Keyboards used with some platforms were missing some of these characters. So it was not possible to write C code on these machines.

```
| <meta name="keywords" content="C-language tutorials, C, D in depth ">
```

Search engines search for the required pages by looking for the search criteria on the web based on the keywords specified with meta tag

```
| <meta name="author" content="Subbu">
```

It is the name of the author of the page

```
<div>
```

Initially Tables and framesets were used to lay out a web page, but now `<div>` is used to layout the page. `<div>` is a blocked HTML Document Division Element used to define a style to a section or a division of a web page using Cascading Style Sheets (CSS)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
  <body>  
    <div>  
      Igate Solutions  
    </div>  
    <div>  
      Near Manasa Theatre  
    </div>
```

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

Notes

Miscellaneous tags

`<sup> [super script] , <sub> [sub script]`

The `<sup>` tag defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font. Superscript text can be used for footnotes, like WWW.

The `<sub>` tag defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas, like H₂O.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<body>
  <p>(a+b)2=a2+b2+2ab</p>
  <p>CH4+O2 --> CO2+H2O</p>
</body>
</html>
```

Output:

(a+b)²=a²+b²+2ab

CH₄+O₂ --> CO₂+H₂O

Monospaced Text

Generally the text we type in any word processor follows proportional font that is different characters take different sizes according to their shapes. In HTML there are some tags used to format the text monospaced.

Note The `<tt>` tag was widely used for monospace text in a document in earlier HTML versions, but it is not supported in HTML5.

`<kbd>` (Keyboard) : The tag used for monospaced text to indicate something a user should type on a keyboard

`<code>` (Code) The tag used for monospaced text applied to programming code

`<samp>` (Sample) The tag used for sample text, which is largely the same thing as `<code>`

`<pre>` (Preformatted) The tag preserves the spaces and next line characters

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<body>
  MMMMMMMMMMMMMMM<br/>
  IIIIIIIIIIIII<br/>
  <tt>
```

```
MMMMMMMMMM  
IIIIIIIIIIII<br/>
</tt>
<code>
    #include<stdio.h><br/>
    int main()<br/>
    {<br/>
        int x,y;<br/>
        printf("Enter Two integers:\n");<br/>
        scanf("%d%d",&x,&y);<br/>
        printf("Sum %d\n",x+y);<br/>
        return 0;<br/>
    }<br/>
</code>
<kbd>
55<br/>
11<br/>
Sum 66
</kbd>
<pre>
sdjf hkj dg dfgdfgsdfg
          sdfj g h k l j df gh j s d f g j h s l d f s d f g
          s d f g k s j f h s d g f
</pre>
</body>
</html>
```

Output:

```
oooooooooooooooooooo
oooooooooooooooooooo
#include
int main()
{
int x,y;
printf("Enter Two integers:\n");
scanf("%d%d", &x, &y);
printf("Sum %d\n", x+y);
return 0;
}
55
11
Sum 66
```

<blockquote>

When quoting blocks from other sources, it is customary on Web pages to indent those blocks from the main body of the text. The `<blockquote>` tag does exactly that and you can use `<blockquote>` for any text that you want to indent, not just quotations.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
  <body>  
    <h4>W3C</h4>
```

```

<blockquote
cite="https://en.wikipedia.org/wiki/World_Wide_Web_Consortium">
<p>
The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web (abbreviated WWW or W3). Founded and currently led by Tim Berners-Lee, the consortium is made up of member organizations which maintain full-time staff for the purpose of working together in the development of standards for the World Wide Web.
As of 10 April 2015, the World Wide Web Consortium (W3C) has 397 members. W3C also engages in education and outreach, develops software and serves as an open forum for discussion about the Web.
</p>
</blockquote>
</body>
</html>

```

Output:

W3C

The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web (abbreviated WWW or W3). Founded and currently led by Tim Berners-Lee, the consortium is made up of member organizations which maintain full-time staff for the purpose of working together in the development of standards for the World Wide Web. As of 10 April 2015, the World Wide Web Consortium (W3C) has 397 members. W3C also engages in education and outreach, develops software and serves as an open forum for discussion about the Web.

Special Characters

Char	Number	Entity	Description
©	©	©	COPYRIGHT SIGN
®	®	®	REGISTERED SIGN
€	€	€	EURO SIGN
™	™	™	TRADEMARK
←	←	←	LEFTWARDS ARROW
↑	↑	↑	UPWARDS ARROW
→	→	→	RIGHTWARDS ARROW
↓	↓	↓	DOWNWARDS ARROW
&	&	&	AMPERSAND
>	>	>	GREATE THAN
<	<	<	LESS THAN
µ	µ	µ	MICRON
¿	¿	¿	INVERTED QUESTION MARK

–	–	–	EN DASH
—	—	—	EM DASH

For more special symbol HTML interpretations, refer this link:

<https://www.utexas.edu/learn/html/spchar.html>

Note:

- The " entity was mistakenly omitted from the HTML 3.2 specification. While use of " generates error reports when validating against 3.2, browsers have continued to recognize the entity and its use is generally safe (sticklers may wish to use " instead). The omission has been corrected in the HTML 4.0 specification.
- The non-breaking space (s; or) can be used not only to prevent the separation of words by line wraps, but also to force a space equal in size to a keyboard/spacebar space (useful for indentation or wider word separation) or to "hold" space in the empty cell of a table (as in the table above).
- The middle dot (· or ·) can be used as a bullet and embedded anywhere in text. Because it is equal in size to a period, however, it may be necessary to apply **** or tags to enhance its graphic effect (use of **** or **<BIG></BIG>** elements is not recommended, as these will alter the character's vertical spacing relative to other characters in the same line).

<hr>

The **<hr>** tag defines a thematic break in an HTML page (e.g. a shift of topic). It is used to separate content (or define a change) in an HTML page.

Note:

- In HTML5, the **<hr>** tag defines a thematic break.
- In HTML 4.01, the **<hr>** tag represents a horizontal rule.
- However, the **<hr>** tag may still be displayed as a horizontal rule in visual browsers, but is now defined in semantic terms, rather than presentational terms.
- All the layout attributes are removed in HTML5. Use CSS instead.
- In HTML, the **<hr>** tag has no end tag.
- In XHTML, the **<hr>** tag must be properly closed, like this: **<hr />**.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    <h1>HTML</h1>
    <p>HTML is a language for describing web pages.</p>
    <hr color="red" width="50%" size="5" align="left"/>
    <h1>CSS</h1>
    <p>CSS defines how to display HTML elements.</p>
  </body>
</html>
```

<marquee>

The HTML <marquee> tag is used for scrolling piece of text or image displayed either horizontally across or vertically down your webpage.

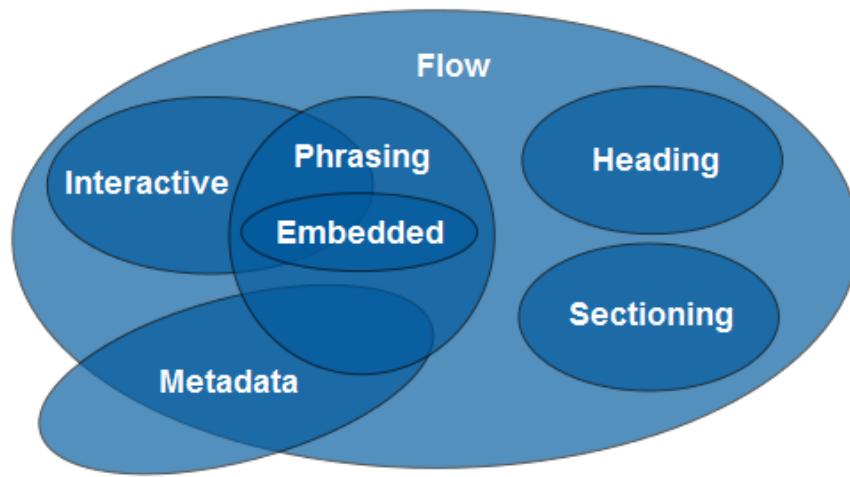
Note: The usage of <marquee> tag is discouraged from HTML5 as it is obsolete and can be removed at any time. It may work on some browser but, still not quite stable.

ATTRIBUTE	VALUE	EXAMPLE
behavior	scroll slide alternate	<marquee behavior="scroll">This is basic example of marquee</marquee>
bgcolor	rgb(x,x,x) #xxxxxx colorname	<marquee bgcolor="yellow">This is basic example of marquee</marquee>
direction	up down left right	<marquee behavior="scroll" direction="down">This is basic example of marquee</marquee>
hspace	pixels	Specifies horizontal space around the marquee.
loop	number	Specifies how many times to loop. The default value is INFINITE, which means that the marquee loops endlessly.
scrolldelay	seconds	Defines how long to delay between each jump.
scrollamount	number	Defines how far to jump.
width	pixels or %	Defines the width of marquee.
vspace	pixels	Specifies vertical space around the marquee.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <body>
    <marquee bgcolor="yellow" height="50%">This is basic example of
marquee</marquee>
  </body>
</html>
```

Content Model of any HTML Page



Heading content

`h1, h2, h3, h4, h5, h6`

Phrasing content

`a*, abbr, area*, audio, b, bdi, bdo, br, button, canvas, cite, code, data, date, datalist, del*, dfn, em, embed, i, iframe, img, input, ins*, kbd, keygen, label, map*, mark, math, meter, noscript, object, output, progress, q, ruby, s, samp, script, select, small, span, strong, sub, sup, svg, textarea, time, u, var, video, wbr, Text*`

Sectioning content

`article, aside, nav, section`

Embedded content

`audio, canvas, embed, iframe, img, math, object, svg, video`

Interactive content

`a, audio*, button, embed, iframe, img*, input*, keygen, label, object*, select, textarea, video*`

Metadata content

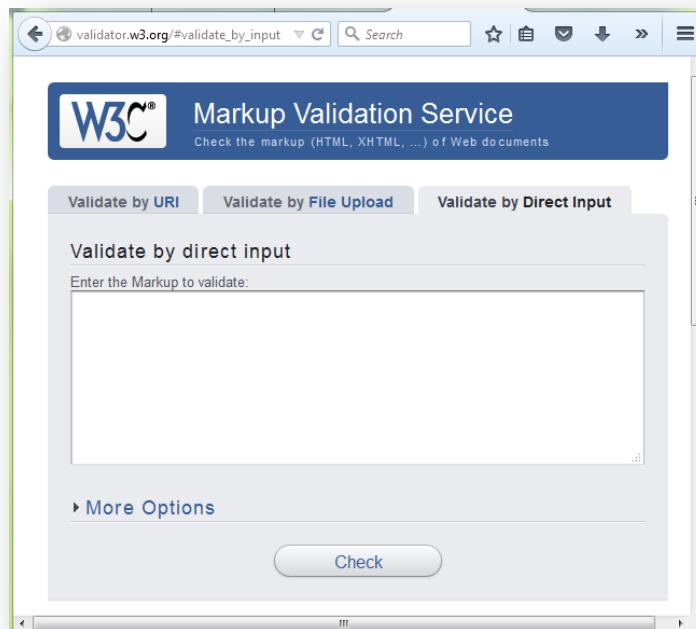
`base, link, meta, noscript, script, style, title`

Flow content

a, abbr, address, area*, article, aside, audio, b, bdi, bdo, blockquote, br, button, canvas, cite, code, data, datalist, del, dfn, div, dl, em, embed, fieldset, figure, footer, form, h1, h2, h3, h4, h5, h6, header, hr, i, iframe, img, input, ins, kbd, keygen, label, main, map, mark, math, meter, nav, noscript, object, ol, output, p, pre, progress, q, ruby, s, samp, script, section, select, small, span, strong, style*, sub, sup, svg, table, textarea, time, u, ul, var, video, wbr, Text*

Validating a HTML program

"validator.w3.org" is the online tool developed and maintained by W3C to validate a web page. It accepts either url or the total code of a webpage and shows the validation result according to the <!doctype> dtd we write at the top of the web page.



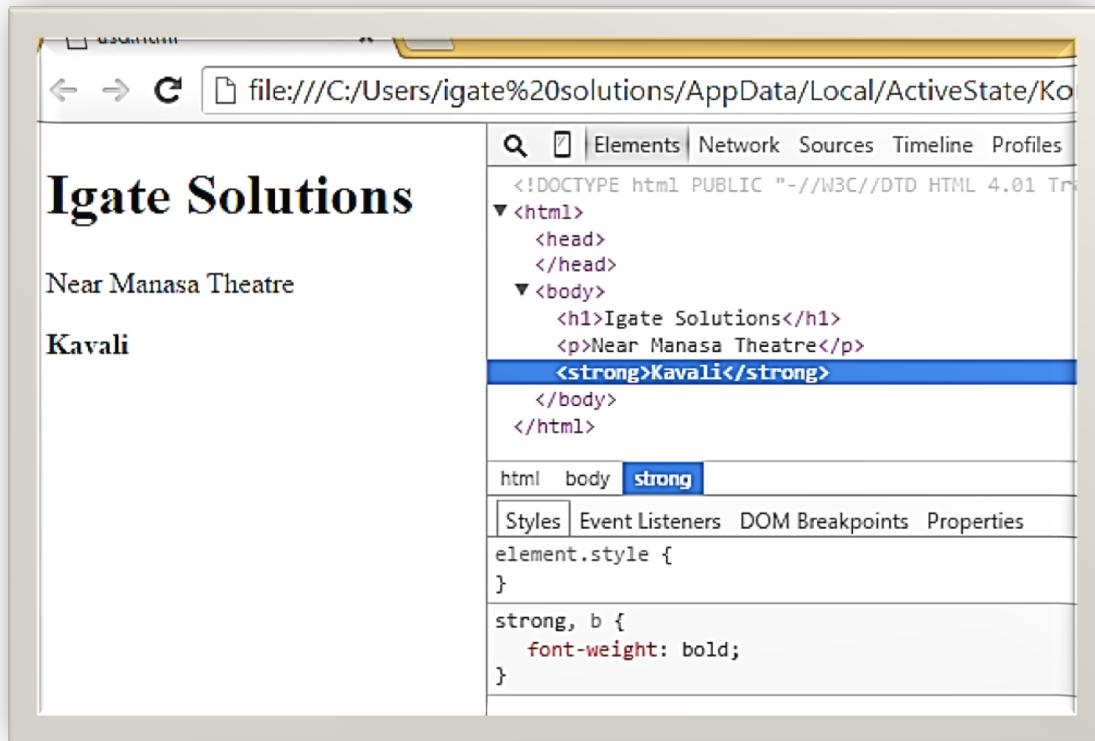
✍ Notes

Notes

Cascading Style Sheets (CSS2)

Need of CSS:

Generally every HTML tag is predefined with some styles, these styles are defined with a default style sheet (file). Say for example **Webkit** is a default style sheet for Google Chrome web browser.



Cascading Style Sheets (CSS) is a style sheet language used to override the existing styles, look and formatting of a document written in HTML. It is used to define styles for web pages, including the design, layout and variations in display for different devices and screen sizes.

How to define a style?

One or more styles can be defined at different levels or scope. That are

- Inline styles
- Internal styles
- External styles

Inline styling

Here we define or overwrite a style of a tag using "style" attribute. It will not apply the style to other tags of the same type.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
      "http://www.w3.org/TR/html4/loose.dtd">
<html>
```

```
<body>
<p style="color:red;font-size:12pt">
    Hello Igate
</p>
<p>
    Igate World <!-- style won't applies-->
</p>
</body>
</html>
```

Internal styling

Here we define styles in the `<style>` tag of the `<head>` section. These styles can be accessed along the web page.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <style>
        p
        {
            color:red;
            font-size:12pt;
        }
    </style>
</head>
<body>
<p>Hello Igate</p>
<p>Igate World </p>
</body>
</html>
```

External styling

The main disadvantage with the internal styling is that, the styles defined in one page can't be applied to other pages. So the same styles must be redefined in every page. It leads to a problem when a style needs to be changed along the web site because it needs modifying every page.

In case of external style sheets, we define all the styles in one or more files and these are linked to other pages. The change of style in a style sheet reflects the change in multiple dependent web pages.

Mystyle.css

```
p{
    color:red;
    font-size:15pt;
}
h1{
    text-align: center;
    background-color: #cccccc;
}
```

Demo.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
  <link href="mystyle.css" rel="stylesheet"/>
</head>
<body>
<h1>Igate Solutions</h1>
<p>
  Near Manasa Theatre
  Kavali
</p>
</body>
</html>
```

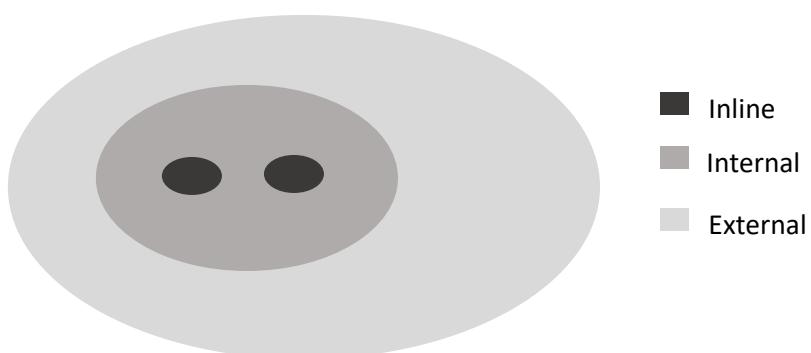
Scope of style:

Scope is an effective area of style.

- The scope of external style sheet is among multiple pages.
- The scope of internal style sheet is along a web page
- The scope of inline style is only in the tag (context) in which the style is defined.

Overriding of Styles:

If same style is defined at different scope levels then one scope would override another scope. Say for example if the same style definition is defined in inline and as well in internal then inline style overrides internal style. If the same style definition is defined as external and internal then internal style overrides external style.



Notes

CSS Selectors

Selector is a pattern used to define a style. Different selectors are used to define styles according to the requirement.

Element or Tag selector

It is a selector used to change the style of a tag.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  p{
    background-color: yellow;
  }
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p>igate solutions</p>
<p>Kavali</p>
</body>
</html>
```

element, element selector

It is used to style several elements with the same style, separate each element name with a comma.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  h1,p{
    background-color: yellow;
    font-family: "lucida console";
  }
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p>igate solutions</p>
<p>Kavali</p>
</body>
</html>
```

Here the same style applies to both the tags <h1> and <p>

element element selector

This selector is used to define style to a tag only when used within a particular tag (Nesting of tags)

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p strong{
    background-color: yellow;
    color:red;
    display: block;
}
</style>
</head>
<body>
<p><strong>Igate Solutions</strong></p>
<strong>Near Manasa Theatre</strong>
<strong>Kavali</strong>
</body>
</html>
```

Here the style of `` changes only when used in `<p>` tag

element > element Selector

This selector is also used to define style to a tag only when used within a particular tag, but only applies at the first descendent level.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p>strong{
    background-color: yellow;
    display: block;
}
</style>
</head>
<body>
<p>
    <strong>Igate Solutions</strong>
    <em>
        <strong>Newr Manasa Theatre</strong>
        <strong>Kavali</strong>
    </em>
</p>
</body>
</html>
```

Here the style of `` applies to “Igate Solutions” because it is in the first descendent level of `<p>` tag, but would not apply in the `` tag.

element+element selector

The element+element selector is used to define a style to the elements that is placed immediately after (not inside) the first specified element.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
P+em {
    background-color: yellow;
}
</style>
</head>
<body>
<p>
  <em>My name is Igate</em>
</p>
  <em>I live in Kavali.</em>  <!-- style applies to this --&gt;
  &lt;em&gt;My best friend is Subbu.&lt;/em&gt;
  &lt;em&gt;I will not be styled.&lt;/em&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
```

Class selector

It is a selector by which we can define a style and can be used anywhere and with any tag using **class** attribute.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
  <style>
    .igate {
        background: yellow;
    }
  </style>
  </head>
  <body>
    <p>igate solutions.</p>
    <p class="igate">Kavali.</p>
    <h6 class="igate">Gateway to IT Carre..</h6>
  </body>
</html>
```

Here we are applying the same style to both the `<h6>` and `<p>` tags.

#id selector

A style defined with # selector can be used only once with any tag. Here the name of selector need to be specified with "id" attribute of a tag.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
#igateid {
    background-color: yellow;
}
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p id="igateid">igate solutions</p>
<p>Kavali</p>
</body>
</html>
```

* Selector

The style defined with * selector will be applied to all the tags and even including with descending levels.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
*{
    width:500px;
    margin-left: 100px;
}
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p>
    <strong>igate solutions</strong>
    <em>Kavali</em>
</p>
</body>
</html>
```

[attribute] selector

The [attribute] selector is used to style any tag with the specified attribute.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
[type] {
    margin-left: 100px;
}
</style>
</head>
<body>
<ol type="A">
    <li>igate solutions</li>
    <li>Cloud 99</li>
</ol>
<ul type="square">
    <li>Igate Solutions</li>
    <li>Clud 99</li>
</ul>
</body>
</html>
```

[attribute=value] Selector

The [attribute=value] selector is used to style a tag with the specified attribute and value.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
[target=_blank] {
    background-color: #cccccc;
    color: white;
    border: 1px solid black;
    padding: 5px;
    text-decoration: none;
    border-radius: 5%;
}
</style>
</head>
<body>
<h1> My favorite links </h1>
<a href="http://www.codingfox.com">codingfox.com</a> <br/><br/>
<!-- style applies to the following two links--&gt;
&lt;a href="http://www.disney.com" target="_blank"&gt;disney.com&lt;/a&gt; &lt;br/&gt;&lt;br/&gt;&lt;a
href="http://www.wikipedia.org" target="_blank"&gt;wikipedia.org&lt;/a&gt; &lt;br/&gt;&lt;br/&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
```

[attribute \sim =value] Selector

The [attribute \sim =value] selector is used to select tags with an attribute value containing a specified word.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
[title $\sim$ =flower] {
    border: 5px solid yellow;
}
</style>
</head>
<body>
 <!-- applies
only to this-->


</body>
</html>
```

[attribute|=value] Selector

The [attribute|=value] selector is used to select elements with the specified attribute starting with the specified value. The value has to be a whole word, either alone, like lang="en", or followed by a hyphen(-), like lang="en-us".

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
[lang|=en] {
    background: yellow;
}
</style>
</head>
<body>
<p lang="en">Hello!</p>
<p lang="en-us">Hi!</p>
<p lang="en-gb">Ello!</p>
<!-- applies to all the above tags -->
<p lang="us">Hi!</p>
<p lang="no">Hei!</p>
</body>
</html>
```

Notes

Styling Anchor tags

:active selector

It is used to style an element when we click on it. Though it can be used with all the elements, it is generally used with hyperlinks to format the active hyperlink

Note: active MUST come after :hover (if present) in the CSS definition in order to be effective!

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
a:active {
            color: #cccccc;
            text-decoration: none;

}
p:active{
            border:2px solid black;
            background-color: black;
            color:white;
}
</style>
</head>
<body>
<a href="http://www.codingfox.com">codingfox.com</a><br/>
<a href="http://www.wikipedia.org">wikipedia.org</a><br/>
<p>hello igate</p>
</body>
</html>
```

:hover selector

The *:hover* selector is used to select elements when you mouse over them. It can be used on all elements, not only on links.

Note: Use the *:link* selector to style links to unvisited pages, the *:visited* selector to style links to visited pages, and the *:active* selector to style the active link.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
a:hover {
            color: #cccccc;
            text-decoration: none;

}
p:hover{
            border:2px solid black;
            background-color: black;
            color:white;
```

```
    }
</style>
</head>
<body>
<a href="http://www.codingfox.com">codingfox.com</a><br/>
<a href="http://www.wikipedia.org">wikipedia.org</a><br/>
<p>hello igate</p>
</body>
</html>
```

:visited selector

The *:visited* selector is used to select visited links.

Note: Use the *:link* selector to style links to unvisited pages, the *:hover* selector to style links when you mouse over them, and the *:active* selector to style links when you click on them.

Browsers limits the styles that can be set for *a:visited* links, due to security issues.

Allowed styles are:

- color
- background-color
- border-color (and border-color for separate sides)
- outline color
- column-rule-color
- the color parts of fill and stroke

:link selector

The *:link* selector is used to select unvisited links. It will not style links you have already visited.

Note: Use the *:visited* selector to style links to visited pages, the *:hover* selector to style links when you mouse over them, and the *:active* selector to style links when you click on them.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
a{
    font:1em sans-serif;
}
/* unvisited link */
a:link {
    text-decoration: none; /* will be inherited to others */
    color: green;
}
/* visited link */
a:visited {
    color: green;
}
/* mouse over link */
a:hover {
```

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

Points to remember:

1. Styles of :link would be inherited to :visited, :hover and :active
2. The order of styles to hyperlinks are :link, :visited, :hover and :active.

::after selector

The `::after` selector inserts content after the selected element(s). The `content` property is used to specify the content to insert.

Note: IE8 and Opera 4-6 only support the old, single-colon CSS2 syntax (`:after`). Newer versions support the standard, double-colon CSS3 syntax (`::after`).

::before selector

The `::before` selector inserts content before the content of the selected element(s). The `content` property is used to specify the content to insert.

Note: IE8 and Opera 4-6 only support the old, single-colon CSS2 syntax (`:after`). Newer versions support the standard, double-colon CSS3 syntax (`::after`).

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
h3:before {
    content: " Quotes on ";
    color:green;
}
ul:after {
    content: " - Gandhi.";
    color: red;
    font:oblique;
}
</style>
</head>
<body>
<h3>Love and Inclusion</h3>
```

```
<ul>
    <li>The good man is the friend of all living things.</li>
    <li>Civilization is the encouragement of differences.</li>
</ul>
<h3>Change and Service</h3>
<ul>
    <li>Action expresses priorities.</li>
    <li>To lose patience is to lose the battle.</li>
</ul>
</body>
</html>
```

:first-child selector

The *:first-child* selector is used to define a style to the tag so that the style applies only when it is at the first descendent level of any container along the page.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p:first-child {
    color: magenta;
    font:15pt sans-serif;

}
</style>
</head>
<body>
<p>The story begins from here...</p> <!-- first child of body-->
<h1>Welcome to My Homepage</h1>
<div>
    <p>Igate Solutions.</p> <!-- first child of div-->
    <p>Near Manasa Theatre.</p>
    <p>Kavali.</p>
</div>
</body>
</html>
```

:first-letter selector

The *::first-letter* selector is used to add a style to the first letter of the specified selector.

Note: The following properties can be used with *::first-letter*:

- font properties
- color properties
- background properties
- margin properties
- padding properties
- border properties
- text-decoration

- vertical-align (only if float is 'none')
- text-transform
- line-height
- float
- clear

Note: The `::first-letter` selector can only be used with block-level elements.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p::first-letter {
    font-size: 200%;
    color: #8A2BE2;
}
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p>My name is Donald.</p>
<p>I live in Duckburg.</p>
<p>My best friend is Mickey.</p>
<p><b>Note:</b> For this selector to work in IE 5.5-8, you must specify the old,
single-colon CSS2 syntax (:first-letter instead of ::first-letter).</p>
</body>
</html>
```

:first-line selector

The `::first-line` selector is used to add a style to the first line of the specified selector.

Note: The following properties can be used with `::first-line`:

- font properties
- color properties
- background properties
- word-spacing
- letter-spacing
- text-decoration
- vertical-align
- text-transform
- line-height
- clear

Note: The `::first-line` selector can only be used with block-level elements.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p::first-line {
    background-color: yellow;
}
</style>
</head>
<body>
<h1>WWF's Mission Statement</h1>
<p>To stop the degradation of the planet's natural environment and to build a future in which humans live in harmony with nature, by; conserving the world's biological diversity, ensuring that the use of renewable natural resources is sustainable, and promoting the reduction of pollution and wasteful consumption.</p>
<p><b>Note:</b> For this selector to work in IE 5.5-8, you must specify the old, single-colon CSS2 syntax (:first-line instead of ::first-line).</p>
</body>
</html>
```

Nested pseudo specification

We can use multiple pseudo selectors in one precise nested specification. Here the order of the pseudo specification is important. For example, you can specify *:first-letter* after *:first-child* but not vice versa. It then applies styles to the first letter of first child of the specified element.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p:first-child:first-letter{
    font-size: 200%;
    color: #8A2BE2;
}
</style>
</head>
<body>
<p> <!--Applies only to this -->
    Alphabet is about businesses prospering through strong leaders and independence.In general, our model is to have a strong CEO who runs each business, with Sergey and me in service to them as needed. We will rigorously handle capital allocation and work to make sure each business is executing well.
</p>
<p>
    Alphabet is about businesses prospering through strong leaders and independence.In general, our model is to have a strong CEO who runs each business, with Sergey and me in service to them as needed. We will rigorously handle capital allocation and work to make sure each business is executing well.
</p>
</body>
</html>
```

:focus selector

The `:focus` selector is used to define a style that must appear on the element on getting the focus. The `:focus` selector is allowed on elements that accept keyboard events or other user inputs. For example, `<input>`, `<button>`.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  p:focus {
    background-color: yellow;
  }
</style>
</head>
<body>
First name: <input type="text"><br/>
Last name: <input type="text">
</body>
</html>
```

CSS Units

- CSS has several different units to express the length.
- Many CSS properties take "length" values, such as width, margin, padding, font-size, border-width, etc.
- Length is a number followed by a length unit, such as 10px, 2em, etc.
- A whitespace cannot appear between the number and the unit. However, if the value is 0, the unit can be omitted.
- For some CSS properties, negative lengths are allowed.
- There are two types of length units: relative and absolute.

Relative lengths

Relative length units specify a length relative to another length property. Relative length units scales better between different rendering mediums.

Unit	Description
em	Relative to the font-size of the element (2em means 2 times the size of the current font)
ex	Relative to the x-height of the current font (rarely used)
ch	Relative to width of the "0" (zero)
rem	Relative to font-size of the root element
vw	Relative to 1% of the width of the viewport
vh	Relative to 1% of the height of the viewport
vmin	Relative to 1% of viewport's smaller dimension

vmax	Relative to 1% of viewport's larger dimension
%	Relative to the parent element.

Note: The em and rem units are practical in creating perfectly scalable layout!

Absolute lengths

The absolute length units are fixed and a length expressed in any of these will appear as exactly that size. Absolute length units are not recommended for use on screen, because screen sizes vary so much. However, they can be used if the output medium is known, such as for print layout.

Unit	Description
cm	centimeters
mm	millimeters
in	inches ($1\text{in} = 96\text{px} = 2.54\text{cm}$)
px	pixels ($1\text{px} = 1/96\text{th of 1in}$)
pt	points ($1\text{pt} = 1/72 \text{ of 1in}$)
pc	picas ($1\text{pc} = 12 \text{ pt}$)

Note: Pixels (px) are relative to the viewing device. For low-dpi devices, 1px is one device pixel (dot) of the display. For printers and high resolution screens 1px implies multiple device pixels.

✍ Notes

Notes

CSS Properties

Font properties:

Font-style

The *font-style* property specifies the font style of the text.

Syntax:

```
font-style: normal|italic|oblique|initial|inherit;
```

Property values:

Value	Description
normal	The browser displays a normal font style. This is default
italic	The browser displays an italic font style
oblique	The browser displays an oblique font style
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Font-variant

The *font-variant* property is used to display small-caps font. For example, if lower case letters are specified with this property, they would be converted to capitals but in the same size as the lower case letters i.e., smaller than regular capital letters.

Syntax:

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

Property values:

Value	Description
normal	The browser displays a normal font. This is default
small-caps	The browser displays a small-caps font
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Font-weight

The *font-weight* property sets the thickness of the font.

Syntax:

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

Property values:

Value	Description
normal	Defines normal characters. This is default
bold	Defines thick characters
bolder	Defines thicker characters
lighter	Defines lighter characters
100, 200, 300, 400, 500, 600, 700, 800, 900	Defines from thin to thick characters. 400 is the same as normal, and 700 is the same as bold
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Font-size

The *font-size* property sets the size of the font.

Syntax:

```
font-size:medium|xx-small|x-small|small|large|x-large|xx  
large|smaller|larger|length|initial|inherit;
```

Property values:

Value	Description	Units
medium	Sets the font-size to a medium size. This is default	-
xx-small	Sets the font-size to an xx-small size	-
x-small	Sets the font-size to an extra small size	-
small	Sets the font-size to a small size	-
large	Sets the font-size to a large size	-
x-large	Sets the font-size to an extra-large size	-
xx-large	Sets the font-size to an xx-large size	-
smaller	Sets the font-size to a smaller size than the parent element	-
larger	Sets the font-size to a larger size than the parent element	-
length	Sets the font-size to a fixed size in px, cm, etc.	16px, 12pt, 11cm, 2em
%	Sets the font-size to a percent of the parent element's font size	50%, 120%, 200%
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Line-height

The *line-height* property sets the space between the lines of text.

Note: *line-height* spacing is different from paragraph spacing. However, the *line-height* will affect the paragraph spacing indirectly.

Syntax:

```
line-height: normal | number | length | initial | inherit;
```

Property values:

Value	Description	Units
normal	A normal line height. This is default	
number	A number that will be multiplied with the current font size to set the line height	1, 2, 3...
length	A fixed line height in px, pt, cm, etc.	25px, 12px, 15pt, 20cm
%	A line height in percent of the current font size	20%, 50%, 100%
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Font-family

The *font-family* property specifies the font for an element. Here, more than one font name is to be specified separated by commas. The preferred font is to be specified in single/double quotes in order to hint the web browsers to consider it as first prioritized value. If the prioritized value is not supported by the browser, then it will take the next specified font (which is comma separated) to style the element.

For example, in *font-family: "Times New Roman", Georgia, Serif;* specification, the default font family style applied will be *Times New Roman*. If the web browser doesn't support the font-style, then it will opt for the very next specified value *Georgia* and if it is also not supported, the browser will consider the next specification *Serif*.

Note: If the web browser doesn't support any of the font-family specified, then it will apply the browser defaults.

Syntax:

```
font-family: font | initial | inherit;
```

Property values:

Value	Description	Units
<i>family-name</i> <i>generic-family</i>	A prioritized list of font family names and/or generic family names	<i>Times New Roman,</i> <i>Georgia,</i> <i>Serif, calibri</i>
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Font

The *font* shorthand property sets all the above discussed font properties in one declaration. The properties that can be set, are (in order): "*font-style font-variant font-weight font-size/line-height font-family*"

Note: *font-size* and *font-family* values must be declared in this shorthand declaration. The rest of the values may or may not be declared. If not declared, browser defaults will be applied.

Syntax:

font: *font-style font-variant font-weight font-size/line-height font-family|initial|inherit*;

Example:

Font: 12pt Arial;

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p{
    text-align: justify;
    text-indent: 50px;
    width: 70%;
    margin: 0 auto;
    font: 12pt "Segoe UI",Georgia;
}
h1{
    text-align: center;
    text-decoration: underline;
    color: #cc00cc;
    font: 20pt arial;
}
</style>
</head>
<body>
<h1>Apple Products</h1>
<p>
The Apple Store app makes it easy to shop, compare, and read reviews.
You can even check out with Apple Pay. You can buy iPhone with a contract and
carrier discount,
with a plan that lets you upgrade your iPhone more often, or with a pre-paid plan.
When you turn on iPhone for the first time, Setup Assistant walks you through
activating
your phone number, connecting to Wi-Fi, and more.
We will show you some of the features of iOS 8 and built-in apps.
</p>
</body>
</html>
```

Notes

Browser font defaults:

Web browsers will have some of their own font-family style specifications which can be overridden by the using CSS. If not CSS font-family styles are not specified or not compatible with them, web browsers would apply their own default font-family styles to the HTML elements. The following are the list of default font-family styles of web browsers based on the operating system.

	Edge/Internet Explorer	Mozilla Firefox	Google Chrome	Safari	Opera
Windows	Arial, Times, New Roman, Courier New	Arial, Times, New Roman, Courier New	Arial, Times, New Roman, Courier New	Times, New Roman, Courier New	Arial, Times, New Roman, Courier New
Mac OS X	-	Helvetica, Helvetica Neue, Times, Courier	Helvetica, Helvetica Neue, Times-Roman, Courier	Times, Courier	Helvetica, Helvetica Neue, Times, Courier, Lucida Grande,
Linux	-	sans-serif, serif, monospace	-	-	Bitstream Vera Serif, Bitstream Sans Mono

Text Properties:

Color

The color property specifies the color of text.

Syntax:

```
color: color|initial|inherit;
```

Property values:

Value	Description	Units
color	Specifies the text color.	Red, green, rgb(0,200,180), #cccccc
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Text-align

The text-align property is used to align the text horizontally on the webpage. By default, the text will be aligned to the left.

Syntax:

Text-align:left|center|right|justify;

Property values:

Value	Description
Left	Aligns the text to the left
Center	Aligns the text to the center
Right	Aligns the text to the right
Justify	Stretches the line of text to equal the left and right margins of the paragraph(s).

Text-indent

The *text-indent* property is used to set the indentation of the first line of text of a paragraph.

Syntax:

Text-indent: *length*;

Property values:

Value	Description	Units
Length	Horizontal length from left	50px, 10pt, 5em, 12%

Text-decoration

The *text-decoration* property is used to set or remove text decorations of text.

Syntax:

Text-decoration: none|overline|underline|line-through

Property values:

Value	Description
None	Removes the text-decoration if any
Overline	Inserts a line above the text
Underline	Inserts a text under the text
Line-through	Inserts a strike through line to the text

Text-transform

The *text-transform* property is used to transform the text on the page to *lowercase*, *uppercase* and *capitalize* automatically.

For example, in the following code, we are giving the text in `<h1>` tag in lowercase and when it is run on browser, it transforms automatically to uppercase as we are giving `text-transform:uppercase` for the `<h1>` tag in CSS specification.

Syntax:

`Text-transform: lowercase | uppercase | capitalize`

Property values:

Value	Description
Lowercase	Converts the text into lowercase
Uppercase	Converts the text into uppercase
Capitalize	Capitalizes the first letter of each word in the line of text

Word-spacing

The `word-spacing` property increases or decreases the length of the white space between words.

Syntax:

`word-spacing: normal | length | initial | inherit;`

Property values:

Value	Description	Units
normal	Defines normal space between words . This is default	-
<i>length</i>	Defines an extra space between words in px, pt, cm, em, etc. Negative values are allowed	30px, 20pt, 25cm, 2em
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

White-space

The `white-space` property specifies how the white-space between the words is handled.

Syntax:

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

Property values:

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

pre-line	Sequences of whitespace will collapse into a single whitespace. Text will wrap when necessary, and on line breaks
pre-wrap	Whitespace is preserved by the browser. Text will wrap when necessary, and on line breaks
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Letter-spacing

The *letter-spacing* property increases or decreases the space between characters in a text.

Syntax:

```
letter-spacing: normal|length|initial|inherit;
```

Property values:

Value	Description	Units
normal	No extra space between characters. This is default	-
<i>length</i>	Defines an extra space between characters (negative values are allowed)	2px, 5px, -3px
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  p{
    text-align: justify;
    text-indent: 50px;
    width: 70%;
    margin: 0 auto; /* aligns to center */
  }
  h1{
    text-align: center;
    text-decoration: underline;
    color: #cc00cc;
    text-transform: uppercase;
  }
</style>
</head>
<body>
<h1>apple products</h1>
<p>
The Apple Store app makes it easy to shop, compare, and read reviews.
</p>

```

```
You can even check out with Apple Pay. You can buy iPhone with a contract and
carrier discount,
with a plan that lets you upgrade your iPhone more often, or with a pre-paid plan.
When you turn on iPhone for the first time, Setup Assistant walks you through
activating
your phone number, connecting to Wi-Fi, and more.
We will show you some of the features of iOS 8 and built-in apps.
</p>
</body>
</html>
```

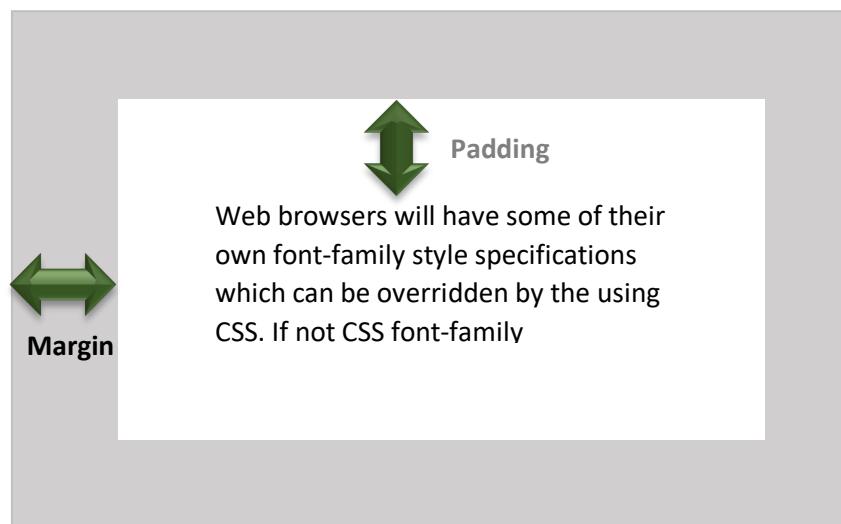
Output:



Notes

Notes

Margin and Padding properties



Margin properties:

Margin-top

The *margin-top* property sets the top margin of an element with respect to its container.

Syntax:

```
Margin-top: length | auto | initial | inherit;
```

Margin-right

The *margin-right* property sets the right margin of an element with respect to its container.

Syntax:

```
margin-right: length | auto | initial | inherit;
```

Margin-bottom

The *margin-bottom* property sets the bottom margin of an element with respect to its container.

Syntax:

```
Margin-bottom: length | auto | initial | inherit;
```

Margin-left

The *margin-left* property sets the left margin of an element with respect to its container.

Syntax:

```
Margin-left: length | auto | initial | inherit;
```

Margin

The margin shorthand property sets all the margin properties in one declaration. This property can have from one to four values.

Examples:

- **margin:10px 5px 15px 20px;**
 - Sets top margin to 10px
 - Sets right margin to 5px
 - Sets bottom margin to 15px
 - Sets left margin to 20px
- **margin:10px 5px 15px;**
 - Sets top margin to 10px
 - Sets right and left margins to 5px
 - Sets bottom margin to 15px
- **margin:10px 5px;**
 - Sets top and bottom margins to 10px
 - Sets right and left margins to 5px
- **margin:10px;**
 - Sets all four margins to 10px

Syntax:

margin: length | auto | initial | inherit;

Property values:

Value	Description	Units
length	Specifies a fixed top margin in px, pt, cm, etc. Default value is 0px	150px, 100pt, 60cm,
%	Specifies a top margin in percent of the width of the containing element	10%,
Auto	The browser calculates a top margin	-
Initial	Sets this property to its default value.	-
Inherit	Inherits this property from its parent element.	-

Padding properties:

Padding-top

The *padding-top* property sets the top padding (space) of an element.

Syntax:

```
padding-top: length | initial | inherit;
```

Padding-right

The *padding-right* property sets the right padding (space) of an element.

Syntax:

```
padding-right: length | initial | inherit;
```

Padding-bottom

The *padding-bottom* property sets the bottom padding (space) of an element.

Syntax:

```
padding-bottom: length | initial | inherit;
```

Padding-left

The *padding-left* property sets the left padding (space) of an element.

Syntax:

```
padding-left: length | initial | inherit;
```

Padding

The padding shorthand property sets all the padding properties in one declaration. This property can have from one to four values.

Examples:

- **padding:10px 5px 15px 20px;**
 - Sets top padding to 10px
 - Sets right padding to 5px
 - Sets bottom padding to 15px
 - Sets left padding to 20px

- **padding:10px 5px 15px;**
 - Sets top padding to 10px
 - Sets right and left padding to 5px
 - Sets bottom padding to 15px

- **padding:10px 5px;**
 - Sets top and bottom padding to 10px
 - Sets right and left padding to 5px

- **padding:10px;**
 - Sets all four paddings to 10px

Property values:

Value	Description	Units
length	Specifies a fixed top margin in px, pt, cm, etc. Default value is 0px	150px, 100pt, 60cm,
%	Specifies a top margin in percent of the width of the containing element	10%,
Initial	Sets this property to its default value.	-
Inherit	Inherits this property from its parent element.	-

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
html{
    background-color: #cccccc;

}
.textbox{
    margin:5%;
    padding:5%;
    background-color: white;
}
</style>
</head>
<body>
<p class="textbox">
Web browsers will have some of their own font-family style specifications
which can be overridden by the using CSS.
If not CSS font-family
</p>
</body>
</html>
```

Centering a container

Many times in web development it is needed to center an element to the container. To do so, we specify top and bottom margins as some value, left and right margins as **auto**

```
#container{  
    width:80%;  
    margin:0 auto;  
}
```

Box properties:

Width

The *width* property sets the width of the container or element.

Note: The *width* property does not include padding, borders, or margins; it sets the width of the area apart from the padding, border, and margin of the element.

Syntax:

```
width: auto | length | initial | inherit;
```

Height

The *height* property sets the height of the element or container.

Note: The *height* property does not include padding, borders, or margins; it sets the height of the area apart from the padding, border, and margin of the element!

Syntax:

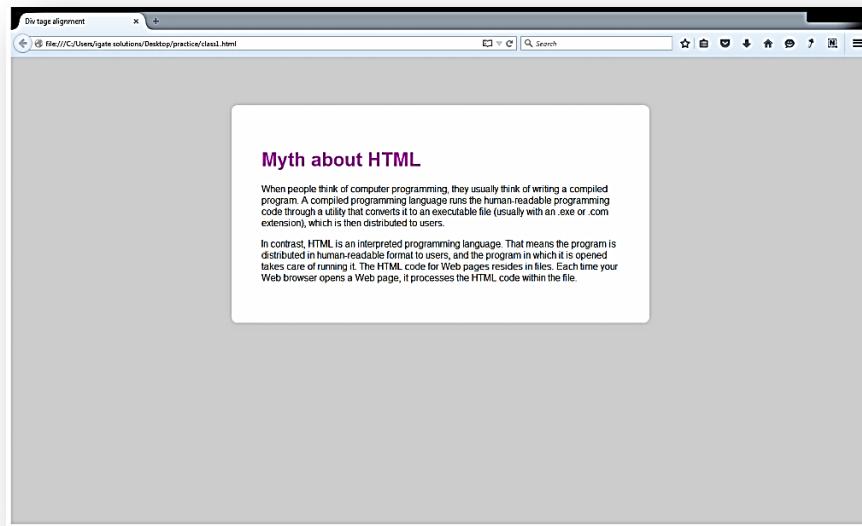
```
height: auto | length | initial | inherit;
```

Property Values:

Value	Description	Units
auto	Default value. The browser calculates the width	-
Length	Defines the width in px, cm, etc.	200px, 100cm,
%	Defines the width in percent of the containing block	20%,33%,60%
Initial	Sets this property to its default value.	-
Inherit	Inherits this property from its parent element.	-

Example 1:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
<head>
    <title>Div tage alignment</title>
    <style>
        body{
            background-color: #cccccc;
            margin:0;
            padding:0;
            font-family:arial;
        }
        h1{
            color:#cc00cc;
        }
        #textbox
        {
            background-color: white;
            width:600px;
            border-radius: 10px;
            margin:5em auto; /*applies top, bottom margins and alignes to center*/
            padding:50px; /* inner border of text box for all sides*/
        }
    </style>
</head>
<body>
<div id="textbox"> <!--Container starts from here-->
<h1> Myth about HTML </h1>
<p> When people think of computer programming, they usually think of writing a compiled program. A compiled programming language runs the human-readable programming code through a utility that converts it to an executable file (usually with an .exe or .com extension), which is then distributed to users.
</p>
<p> In contrast, HTML is an interpreted programming language. That means the program is distributed in human-readable format to users, and the program in which it is opened takes care of running it. The HTML code for Web pages resides in files. Each time your Web browser opens a Web page, it processes the HTML code within the file.
</p>
</div>
</body>
</html>
```



Example 2:

```
<!doctype html>
<html lang="en">
<head>
    <title>Demo of width</title>
    <style>
        #wdemo
        {
            height: 200px;
            width: 300px;
            background-color:#ccc;
            font:12pt arial;
            overflow: scroll;
            margin: 50px auto;
            padding: 10px;
        }
    </style>
</head>
<body>
    <div id="wdemo">
        Some text here....Some text here....Some text here....Some text
        here....Some text here....Some text here.....
    </div>
</body>
</html>
```



Display

The *display* property sets the type of the box to be used.

Syntax:

`display: value;`

Property values:

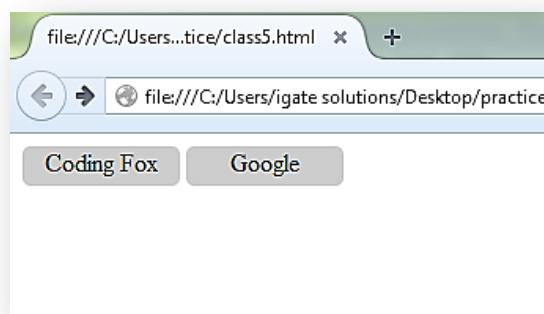
Value	Description
Inline	Displays the element as an inline element
Block	Displays an element in a block.
Inline-block	Displays an element as an inline-level block container. The inside of this block is formatted as block-level box, and the element itself is formatted as an inline-level box
List-item	Displays the item like a item
None	The display will be hidden/not shown at all
Initial	Sets the property to its default value
Inherit	Inherits the property from its parent element

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
a:link,a:visited{
    display: inline-block;
    width:100px;
    height: 25px;
    background-color: #cccccc;
    border-radius: 5px;
    text-decoration: none;
    color: black;
    text-align: center;
}
a:active,a:hover{
    color:white;
    background-color: black;
```

```
}

#hide{
    display: none;
}
</style>
</head>
<body>
    <a href="http://www.codingfox.com">Coding Fox</a>
    <a href="http://www.google.in">Google</a>
    <p id="hide">
        This text would not be displayed
    </p>
</body>
</html>
```



Position

The *position* property is used to set the position of an element.

Syntax:

```
position: static|absolute|fixed|relative|initial|inherit;
```

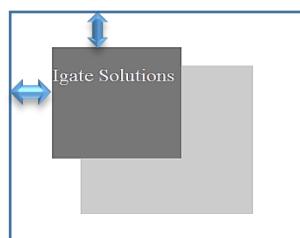
Property values:

Value	Description
static	It is the default value. The elements render as a normal flow. Left, top, bottom and right properties are not applicable
absolute	The element is positioned relative to its parent element. If the parent element is static then positioned according to the body otherwise positioned according to the parent
fixed	The element is positioned relative to the browser window. It will not leave any gap in the page where it would have been.
relative	The element is positioned relative to its normal position. Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.
Sticky	A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met. -webkit-

sticky is used for safari and at least one property like top, bottom, left or right must be used

Case1: If the position of inner element is **absolute** and container is not a positional element (static) then it is aligned according to the windows top and left.

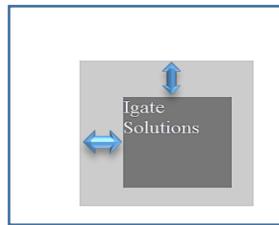
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
#container{ /* static container*/
    margin: 75px;
    width:200px;
    height:200px;
    background-color: #CCC;
}
#inner{
    position:absolute;
    top:50px;
    left:50px;
    width: 150px;
    height: 150px;
    background-color: #777;
}</style>
</head>
<body>
    <div id="container">
        <div id="inner">
            <p style="font-size:1.5em;color:white;">Igate Solutions</p>
        </div>
    </div>
</body>
</html>
```



Case2: If the position of inner element is **relative** and container is static then it will move relative to its original position

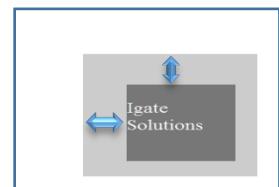
```
#container{ /* static container*/
    margin: 75px;
    width:200px;
    height:200px;
    background-color: #CCC;
}
#inner{
    position:relative;
    top:50px;
```

```
left:50px;  
width: 150px;  
height: 150px;  
background-color: #777;  
}
```



Case3: If the container positioned as **relative** and child's position is set as **absolute** then child is set as relative to the container.

```
#container{  
    position: relative;  
    margin: 75px;  
    width:200px;  
    height:200px;  
    background-color: #CCC;  
}  
#inner{  
    position: absolute;  
    top:50px;  
    left:50px;  
    width: 125px;  
    height: 125px;  
    background-color: #777;  
}
```



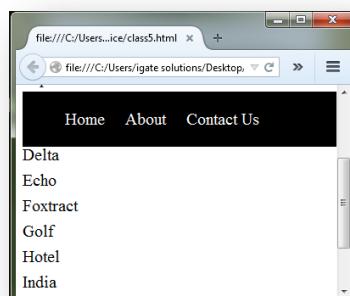
Case4: If we want to keep anything fixed on the screen while other part is moving then it's position must be set as fixed and its container's position must be set as relative.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>  
<head>  
<style>  
#container{  
    position: relative;  
}  
#menu{  
    position: fixed;  
    background-color: black;  
    width: 100%;
```

```
}

#content{
    position: absolute;
    top:75px;
}
ul li{
    list-style: none;
    display: inline-block;
    font-size: 15pt;
    padding: 5px 10px;
    color: white;
}

</style>
</head>
<body>
<div id="container">
    <div id="menu">
        <ul>
            <li>Home</li>
            <li>About</li>
            <li>Contact Us</li>
        </ul>
    </div>
    <div id="content">
        <p style="font-size: 15pt;line-height: 1.5em">
            Alpha<br/>Bravo<br/>Charlee<br/>Delta<br/>Echo<br/>Foextract<br/>Golf<br/>
            Hotel<br/>India<br/>Juliat<br/>Kilo
        </p>
    </div>
</div>
</body>
</html>
```



Case 5: A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met. -webkit-sticky is used for safari and **at least one property like top, bottom, left or right must be used**

```
#one{
    background-color: aquamarine;
    border: 1px solid green;
    padding: 10px;
    margin: 10px auto;
    position: sticky;
```

```
    position: -webkit-sticky;  
    top:10px;  
}
```

Notes

Top

The *top* property sets the element to a specified position from the top of the page.

Note: This property is mostly recommended to use along with the “*position: absolute*” property-value. The top property has no effect on “*position:static*” elements.

Syntax:

```
top: auto|length|initial|inherit;
```

Right

The *right* property sets the element to a specified position from the right side of the page.

For *absolutely positioned* elements, the right property sets the right edge of an element to the left/right of the right edge of its container.

For *relatively positioned* elements, the right property sets the right edge of an element to the left/right to its normal position.

Note: The right property has no effect for “*position:static*” elements.

Syntax:

```
right: auto|length|initial|inherit;
```

Bottom

The *bottom* property sets the element to a specified position from the bottom of the page.

For *absolutely positioned* elements, the bottom property sets the bottom edge of an element above/below the bottom edge of its container.

For *relatively positioned* elements, the bottom property sets the bottom edge of an element above/below its normal position.

Note: The bottom property has no effect on “*position:static*” elements.

Syntax:

```
bottom: auto|length|initial|inherit;
```

left

The *left* property sets the element to a specified position from the left side of the page.

For *absolutely positioned* elements, the left property sets the left edge of an element to the left/right of the left edge of its container.

For *relatively positioned* elements, the left property sets the left edge of an element to the left/right to its normal position.

Note: The left property has no effect on “*position:static*” elements.

Property values:

Value	Description	Units
auto	Lets the browser calculate the top edge position. This is default value	-
length	Sets the top edge position in px, cm, etc. Negative values are allowed	25px, 10cm
%	Sets the top edge position in % of containing element. Negative values are allowed	2%,10%
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Float

The *float* property is set to float the element or container to one side of its parent container. The floated child element will stay as it is set through all the screen sizes.

Syntax:

```
float: none|left|right|initial|inherit;
```

Property values:

Value	Description
none	The element is not floated, and will be displayed just where it occurs in the text. This is default value.
left	The element floats to the left of the container
right	The element floats the right of the container
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Clear

The *clear* property specifies which floating elements are not allowed to float and on which side(s).

Syntax:

```
clear: none|left|right|both|initial|inherit;
```

Property values:

Value	Description
none	This is the default value. Allows floating elements on both sides

left	No floating elements allowed on the left side
right	No floating elements allowed on the right side
both	No floating elements allowed on either the left or the right side
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Example:

```

<!doctype html>
<html lang="en">
<head>
  <title>Class2</title>
  <style>
    body{
      background-color: #cccccc;
      padding:0;
      font:100% arial;
      margin:30px;
    }
    .textbox
    {
      background-color: white;
      width:400px;
      padding:10px 15px 10px 15px;
      border-radius: 10px;
      margin: 15px 0 0 30px;
      float:left;
    }
    .textbox h1{
      color:#cc00cc;
    }
    .clearing
    {
      clear:both;
    }
  </style>
</head>
<body>
  <div class="textbox">
    <h1>HTML</h1>
    <p>
      Some text here..... Some text here.....Some text here.....Some text here.....Some text
here.....<br/>
    </p>
  </div>
  <div class="textbox">
    <h1>CSS</h1>
    <p>
      Some text here..... Some text here.....Some text here.....Some text here.....Some text
here.....<br/>
    </p>
  </div>
</body>

```

```
</p>
</div>
<div class="clearing"></div>
<div class="textbox" style="width:88%">
<h1>JQuery</h1>
<p>
    Some text here..... Some text here.....Some text here.....Some text here.....Some text
here.....</p>
</div>
</body>
</html>
```

The diagram illustrates the effect of the `visibility` property across three different contexts:

- HTML:** Shows a block of text where all elements are visible.
- CSS:** Shows a block of text where all elements are visible.
- JQuery:** Shows a block of text where all elements are visible.

In all three cases, the text consists of multiple lines of "Some text here....." repeated several times.

Visibility

The `visibility` property is used to show or hide an element.

Note: Hiding elements on the page will not release the space taken by them on the page. Using `display` is recommended if the space is also to be removed.

Syntax:

```
visibility: visible|hidden|collapse|initial|inherit;
```

Property values:

Value	Description
visible	This is the default value. The element will be visible
hidden	The element will be hidden/invisible (but still takes up space)
collapse	Only for table elements. Collapse removes a row or column, but it does not affect the table layout. The space taken up by the row or column will be available for other content.

	If collapse is used on other elements, it renders as "hidden"
initial	Sets this property to its default value.
inherit	Inherits this property from its parent element.

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
<head>
<title>Class3</title>
<style>
#hide{
  visibility: hidden;
}
</style>
</head>
<body>
<p>igate solutions</p>
<p id="hide">Near Manasa Theatre</p>
<p>Kavali</p>
</body>
</html>
```

Vertical-align

The *vertical-align* property is used to align the element vertically in that line-height limit.

Syntax:

```
vertical-align: baseline | length | sub | super | top | text-top | middle | bottom | text-bottom | initial | inherit;
```

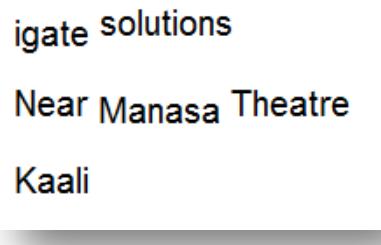
Property values:

Value	Description	Units
baseline	Align the baseline of the element with the baseline of the parent element. This is the default value	-
length	Raises or lower an element by the specified length. Negative values are allowed	10px, 1cm, -12px
%	Raises or lower an element in a percent of the "line-height" property. Negative values are allowed	10%, 50%
sub	Aligns the element as subscript	-
super	Aligns the element as superscript	-
top	The top of the element is aligned with the top of the tallest element on the line	-
text-top	The top of the element is aligned with the top of the parent element's font	-
middle	The element is placed in the middle of the parent element	-

bottom	The bottom of the element is aligned with the lowest element on the line	-
text-bottom	The bottom of the element is aligned with the bottom of the parent element's font	-
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
<head>
<title>Class3</title>
<style>
p{
  font:15pt "Arial";
  line-height: 2.0;
}
.sup{
  vertical-align: super;
}
.sub{
  vertical-align: sub;
}
</style>
</head>
<body>
<p>igate <span class="sup">solutions</span><br/>Near <span class="sub">Manasa</span>
Theatre<br/>Kaali</p>
</body>
</html>
```



igate solutions
Near Manasa Theatre
Kaali

Notes

Z-index

The z-index property is used to specify the order of an element in stack style. The element with greater value will always be in front and an element with a lower value will stay background. The front element will overlap the back thrown element(s).

Syntax:

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

Note: z-index only works on positioned elements (position:absolute, position:relative, or position:fixed).

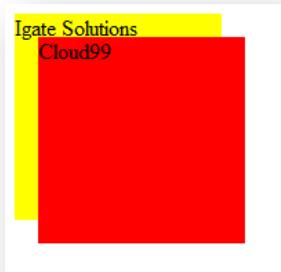
Property values:

Value	Description	Units
auto	Sets the stack order equal to its parents. This is the default value	-
number	Sets the stack order of the element. Negative numbers are allowed	1,2,3,999
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

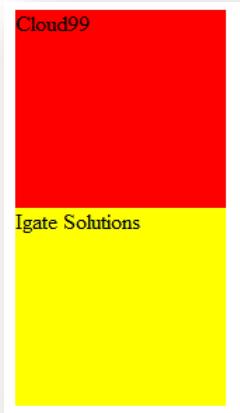
Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
<head>
<title>Igate Solutions</title>
<style>
.box{
  position: absolute;
  width:150px;
  height:150px;
}
.box.light{
  background-color: yellow;
  z-index: 1;
}
.box.dark{
  top:25px;
  left:25px;
  background-color: red;
  z-index: 2;
}
</style>
</head>
<body>
<div class="box dark">
  Cloud99
</div>
```

```
<div class="box light">  
  Igate Solutions  
</div>  
</body>  
</html>
```



If the position is not set to the absolute then the output appears as



Background properties

Background-color

The background-color property sets the background color of an element. The background of an element is the total size of the element, including padding and border (but not the margin).

Syntax: background-color: *color|transparent|initial|inherit*;

Property values:

Value	Description	Units
<i>color</i>	Specifies the background color.	Red, rgb(255,255,255), #cccccc
<i>transparent</i>	Specifies that the background color should be transparent. This is default	-
<i>initial</i>	Sets this property to its default value.	-
<i>inherit</i>	Inherits this property from its parent element.	-

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  html{
    background-color: #CCC;
  }
  body{
    width:500px;
    background-color: white;
    padding:2%;
    margin:30px auto;
    font:0.8em sans-serif;
  }
  #title{
    color:#CC00CC;
    font:1.5em sans-serif;
  }
</style>
</head>
<body>
  <h2 id="title">Bahubali</h2>
  <p>
    The Beginning was released worldwide on 10 July 2015 to positive reviews from critics,
    and it recorded the highest-grossing opening for an Indian film at the box-office.
    Baahubali became the first non-Hindi film to net.
  </p>
</body>

```

```
    gross over Rs.104 crore (US$16 million) in dubbed Hindi version in India
</p>
</body>
</html>
```

Bahubali

The Beginning was released worldwide on 10 July 2015 to positive reviews from critics, and it recorded the highest-grossing opening for an Indian film at the box-office. Baahubali became the first non-Hindi film to net. gross over Rs.104 crore (US\$16 million) in dubbed Hindi version in India

Background-attachment

The background-attachment property sets whether a background image is fixed or scrolls with the rest of the page.

Syntax: background-attachment: scroll|fixed|local|initial|inherit;

Property values:

Value	Description	Units
scroll	The background scrolls along with the element. This is default	-
fixed	The background is fixed with regard to the viewport	-
local	The background scrolls along with the element's contents	-
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
      "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
body{
    background-image: url("./images/paper.png");
    background-repeat: no-repeat;
    background-attachment: fixed;
}
</style>
</head>
<body>
```

```
<p>Some Text Here... </p>
</body>
</html>
```

Background-image

The background-image property sets one or more background images for an element. The background of an element is the total size of the element, including padding and border (but not the margin).

By default, a background-image is placed at the top-left corner of an element, and repeated both vertically and horizontally.

Syntax: background-image: *url|none|initial|inherit*;

Note: Always set a background-color to be used if the image is unavailable.

Property values:

Value	Description	Units
url('URL')	The URL to the image. To specify more than one image, separate the URLs with a comma	url(URL), url(URL,URL)
none	No background image will be displayed. This is default	-
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Background-position

The background-position property sets the starting position of a background image.

Syntax: background-position: *value*;

Note: By default, a background-image is placed at the top-left corner of an element, and repeated both vertically and horizontally.

Property values:

Value	Description	Units
left top, left center, left bottom, right top, right center, right bottom, center top, center center, center bottom	If you only specify one keyword, the other value will be "center"	-
x% y%	The first value is the horizontal position and the second value is the vertical. The top left corner is 0% 0%. The right bottom corner is 100% 100%. If you only specify one value,	(20%,45%),(0%,0%)

	the other value will be 50%. . Default value is: 0% 0%	
xpos ypos	The first value is the horizontal position and the second value is the vertical. The top left corner is 0 0. Units can be pixels (0px 0px) or any other CSS units. If you only specify one value, the other value will be 50%. You can mix % and positions	200px 90px, 20% 500px
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

Background-repeat

The background-repeat property sets how a background image will be repeated. By default, a background-image is repeated both vertically and horizontally.

Syntax: background-repeat: repeat|repeat-x|repeat-y|no-repeat|initial|inherit;

Note: The background image is placed according to the background-position property. If no background-position is specified, the image is always placed at the element's top left corner.

Property values:

Value	Description	Units
repeat	The background image will be repeated both vertically and horizontally. This is default	-
repeat-x	The background image will be repeated only horizontally	-
repeat-y	The background image will be repeated only vertically	-
no-repeat	The background-image will not be repeated	-
initial	Sets this property to its default value.	-
inherit	Inherits this property from its parent element.	-

background

The background shorthand property sets all the background properties in one declaration. The properties that can be set, are: *background-color*, *background-image*, *background-position*, *background-size(CSS3)*, *background-repeat*, *background-origin(CSS3)*, *background-clip(CSS3)*, and *background-attachment*.

Syntax:

```
background: color image position/size repeat origin clip attachment initial|inherit;
```

Note: There is no need to mention all the properties in every declaration. It does not matter if one of the values above are missing.

Property values:

Value	Description	Units
<i>background-color</i>	Specifies the background color to be used	Red,rgb(0,0,0),#dadada
<i>background-image</i>	Specifies ONE or MORE background images to be used	url('URL of the image')
<i>background-position</i>	Specifies the position of the background images	left top, left center, left bottom, right top, right center, right bottom, center top, center center, center bottom
<i>background-size</i>	Specifies the size of the background images	(Refer CSS3 material)
<i>background-repeat</i>	Specifies how to repeat the background images	Scroll, fixed, local, initial, inherit
<i>background-origin</i>	Specifies the positioning area of the background images	(Refer CSS3 material)
<i>background-clip</i>	Specifies the painting area of the background images	(Refer CSS3 material)
<i>background-attachment</i>	Specifies whether the background images are fixed or scrolls with the rest of the page	Scroll, fixed, local, initial, inherit
<i>initial</i>	Sets this property to its default value.	-
<i>inherit</i>	Inherits this property from its parent element.	-

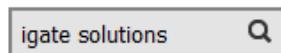
Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
#search{
    margin:10px auto;
    background-image: url("./images/magnifier.png");
    background-position: right;
```

```
background-repeat: no-repeat;
background-color: rgb(228,228,228);
padding: 5px;
}
</style>
</head>
<body>
<input id="search" type="text"/>
</body>
</html>
```

The same style can be defined using shortcut property

```
#search{
margin:10px auto;
background: rgb(228,228,228) url("./images/magnifier.png") right no-repeat;
padding: 5px;
}
```



Border Properties

Border

The border shorthand property sets all the border properties in one declaration. The properties that can be set, are (in order): border-width, border-style, and border-color.

Syntax:

```
border: border-width border-style border-color | initial | inherit;
```

Note: There is no need to give all the shorthand property values in every declaration.

Property values:

Value	Description	Units
<i>border-width</i>	Specifies the width of the border. Default value is "medium"	Thin, medium, thick, 5px
<i>border-style</i>	Specifies the style of the border. Default value is "none"	None, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset, initial, inherit
<i>border-color</i>	Specifies the color of the border. Default value is the color of the element	Red, green, rgb(0,0,0),#dcdcdc
initial	Sets this property to its default value.	-

inherit	Inherits this property from its parent element.	-
---------	-------------------------------------------------	---

Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
          "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
p {
    border: 5px solid red;
}
</style>
</head>
<body>
<p>This is some text in a paragraph.</p>
</body>
</html>
```

Border-bottom / Border-left / Border-right / Border-top

The border-bottom shorthand property sets all the bottom border properties in one declaration. The properties that can be set, are (in order): border-bottom-width, border-bottom-style, and border-bottom-color.

Syntax:

```
border-bottom: border-width border-style border-color|initial|inherit;
```

Note: There is no need to give all the shorthand property values in every declaration.

Example:

```
border-bottom: thick dotted #ff0000;
```

border-bottom-color / border-left-color / border-right-color / border-top-color

The border-bottom-color property sets the color of an element's bottom border.

Syntax:

Note: Always declare the border-style property before the border-bottom-color property. An element must have borders before you can change the color.

```
border-bottom-color: #ff0000;
```

border-bottom-style / border-left-style / border-right-style / border-top-style

The border-bottom-style property sets the style of an element's bottom border.

Syntax:

```
border-bottom-
style:none|hidden|dotted|dashed|solid|double|groove|ridge|inset|outset|initial|inherit;
```

border-bottom-width / border-left-width / border-right-width /border-top-width

The border-bottom-width property sets the width of an element's bottom border.

Syntax: border-bottom-width: medium | thin | thick | *length* | initial | inherit;

Note: Always declare the border-style property before the border-bottom-width property. An element must have borders before you can change the width.

```
border-style: solid;  
border-bottom-width: 15px;
```

border-color

The border-color property sets the color of an element's four borders. This property can have from one to four values.

Definitions:

border-color:red green blue pink;

- top border is red
- right border is green
- bottom border is blue
- left border is pink

border-color:red green blue;

- top border is red
- right and left borders are green
- bottom border is blue

border-color:red green;

- top and bottom borders are red
- right and left borders are green

border-color:red;

- all four borders are red

Syntax: border-color: *color* | transparent | initial | inherit;

Note: Always declare the border-style property before the border-color property. An element must have borders before you can change the color.

Example:

```
border-style: solid;  
border-color: #ff0000 #00ff00 #0000ff rgb(250,0,255);
```

Example:

```
<!doctype html>
<html lang="en">
<head>
    <title>Paragraph and Font formatting</title>
    <style>
        html{
            background-color: #cccccc;
        }
        body{
            background-color: #ffffff;
            width:70%;
            margin:auto; /* center align*/
            padding:50px;
            border-top: 20px solid #cc00cc;
            border-bottom: 20px solid #cc00cc;
        }
        h1{
            font-family: arial;
            font-size: 1.5em;
            color:#cc00cc
        }
        p{
            font-family:Grande,arial;
            font-size:12pt;
            line-height: 1.7;
            text-align: justify;
        }
    </style>
</head>
<body>

<h1>HTML4</h1>
<p>A very stable, ...
</p>
<h1>XHTML</h1>
<p>A strict, ....
</p>
<h1>HTML5</h1>
<p>A revised code ...
<p> For example...
<p>
A good Web ...
</body>
</html>
```

The screenshot shows a web browser window with a purple header bar. The address bar displays 'file:///C:/Users/igate solutions/Desktop/practice/lab1.html'. The main content area contains three sections: 'HTML4' (with a note about compatibility), 'XHTML' (with a note about XML), and 'HTML5' (with notes about new features like multimedia tags and improved application handling). The browser interface includes standard toolbar icons.

HTML4
A very stable, universally accepted code set, which is also fairly forgiving of small coding errors. Using HTML4 codes is desirable when compatibility with all browsers is important.

XHTML
A strict, standards-based implementation of HTML4 created with XML (eXtensible Markup Language). XHTML coding uses the same codes as HTML4, so it is compatible with the same browsers as HTML4. (See the sidebar about XML on the next page for more information.)

HTML5
A revised code set that builds upon HTML4 to add new capabilities. HTML5 offers many dramatic improvements in the areas of application handling and multimedia, but a lot of those features are beyond the scope of this book. In terms of basic coding, which is what this book teaches, the biggest difference is that there are new specific codes for different types of content that were previously handled with more general codes.
For example, HTML5 has tags for inserting multimedia content, whereas HTML4 inserts all types of multimedia content via a generic tag. Since this is a book about HTML5, it might seem like an obvious decision to do your coding using HTML5 tags, but it is not quite as simple as that in real-world situations.
A good Web browser should ideally support every tag and every version of HTML it can, because the various HTML version differences should be completely invisible to the Web site visitor. However, HTML5 is so new that not all browsers have caught up to it yet, and people who use older computers may not have the latest version of a browser even if an

Cursor

It is to set the mouse pointer that would appear when we place the mouse pointer on an element

Syntax:

Cursor: value

Values: auto, default , none , context-menu , help , pointer , progress , wait , cell , crosshair , text , vertical-text , alias , copy , move , no-drop , not-allowed , e-resize , n-resize , ne-resize , nw-resize , s-resize , se-resize , sw-resize , w-resize , ew-resize , ns-resize , nesw-resize , nwse-resize , col-resize , row-resize , all-scroll , zoom-in , zoom-out , grab , grabbing

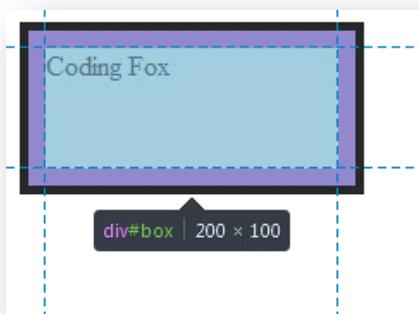
Example:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
  input[type="text"]{
    cursor: pointer;
  }
</style>
</head>
<body>
  <input type="text"/>
</body>
</html>
```

Box-sizing

Generally when we add the padding to the div the size of div would change. We set "box-sizing" property to "border-box" to add padding without changing the actual size of div

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<style>
#box{
    background-color: #ccc;
    height:100px;
    width:200px;
    border: 5px solid black;
    padding:10px;
    box-sizing: border-box;
}
</style>
</head>
<body>
    <div id="box">
        Coding Fox
    </div>
</body>
</html>
```

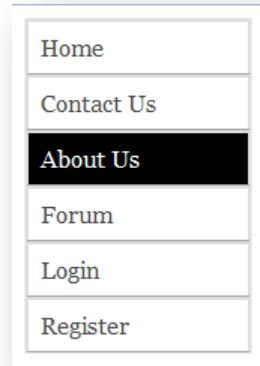


Notes

Notes

PUTTING ALL TOGETHER

Challenger 1: Creating vertical menu



Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
.menu{
    list-style: none;
    padding: 0;
    margin: 0;
    width: 150px;
    border: 1px solid #eee;
}
.menu li a{
    display: block;
    padding: 8px;
    border: 2px solid #ddd;
    border-bottom: 1px solid #aaa; /* to get 3D effect */
    font: 1em Georgia;
    color:#555;
    text-decoration: none;
    zoom: 1;
}
.menu li a:hover{
    background: #000;
    color: #fff;
}
        </style>
    </head>
    <body>
        <div>
            <ul class="menu">
                <li><a href="#">Home</a></li>
                <li><a href="#">Contact Us</a></li>
                <li><a href="#">About Us</a></li>
                <li><a href="#">Forum</a></li>
                <li class='selected'><a href="#">Login</a></li>
            </ul>
        </div>
    </body>
</html>
```

```
<li><a href="#">Register</a></li>
</ul>
</div>
</body>
</html>
```

Challenger 2: Horizontal menu

Igate Solutions

HOME CONTACT US ABOUT US FORUM LOGIN REGISTER

Solution:

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <style>
      body{
        background: #eee;
      }
      #container{
        width: 960px;
        margin: 0 auto;
      }
      h1{
        font: 2em Georgia;
      }
      .menu{
        font: 1em Georgia;
        list-style: none;
        padding: 0;
        margin: 0;
        background: #fff;
        text-transform: uppercase;
        border-bottom: 3px solid #000;
        border-top: 1px solid #ddd;
      }
      .menu li{
        display: inline-block;
        zoom: 1;
      }

      .menu li a{
        display: block;
        color: #000;
        text-decoration: none;
        line-height: 1.6em;
        padding: 5px 15px 4px 15px;
      }
    </style>
  </head>
  <body>
    <div id="container">
      <h1>Igate Solutions</h1>
      <ul class="menu">
        <li><a href="#">Register</a></li>
        </ul>
      </div>
    </body>
  </html>
```

```
}

.menu li a:hover{
    background-color: black;
    color: white;
}
</style>
</head>
<body>
    <div id="container">
        <h1>Igate Solutions</h1>
        <ul class="menu">
            <li><a href="#">Home</a></li>
            <li><a href="#">Contact Us</a></li>
            <li><a href="#">About Us</a></li>
            <li><a href="#">Forum</a></li>
            <li class='selected'><a href="#">Login</a></li>
            <li><a href="#">Register</a></li>
        </ul>
    </div>
</body>
</html>
```

Challenger 3: Creating a button



Don't have account? Sign up now

Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
            .button{
                margin: 0 5px;
                background: #eddede;
                padding: 5px 15px;
                border: 1px solid #dcdcdc;
                border-radius: 6px;
                color: #777;
                text-decoration: none;
                font: bold 0.8em arial,sans-serif;
            }
            .button:hover{
                background: #ddd;
                text-decoration: underline;
                text-decoration: none;
            }
        </style>
    </head>
    <body>
        <p>Don't have account? <a href="#" class='button'>Sign up now</a></p>
    </body>
</html>
```

Challenger 4: Creating a boxed content



Solutions:

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <style>
      .comment{
        width:300px;
        padding:15px;
        border:1px solid #777;
        font: 12pt Arial;

      }
      .comment img{
        width:150px;
        height:50px;
        float: left;
        margin-right: 10px;
        margin-bottom: 5px;
      }
      .text{
        margin-bottom: 25px;
        text-align: justify;
      }
      .comment .links{
        list-style: none;
        float: right;
      }
      .comment .links li{
        display: inline;
        margin-left: 15px;
      }
      .clear{
        clear: both;
      }
    </style>
  </head>
  <body>
    <div class='comment'>
      
      <div class="text">
```

Hello, my name is Subbu, I'm making tutorials on CSS floating. Not too internationally I know, but it's really useful for any developer to know this stuff.

```
</div>
<ul class="links">
    <li>Reply</li>
    <li>Delete</li>
    <li>Report</li>
</ul>
<div class="clear"/>
</div>
</div>
</body>
</html>
```

Challenger 5: Styling the lists

- ⇒ HTML
- ⇒ CSS
- ⇒ JavaScript
- ⇒ XML
- ⇒ PHP

Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
            ul{
                list-style: none;
                list-style-position: inside;
                list-style-image: url("arrow.png");
                line-height: 1.5em;
            }
        </style>
    </head>
    <body>
        <ul>
            <li>HTML</li>
            <li>CSS</li>
            <li>JavaScript</li>
            <li>XML</li>
            <li>PHP</li>
        </ul>
    </body>
</html>
```

Challenger 6: Transparent Box



Solution:

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <style>
      img{
        position: absolute;
      }
      #text{
        position: absolute;
        background-color:white;
        width:500px;
        top:75px;
        left: 70px;
        opacity: 0.4; /* from 0.1 to 1.0 */
        font: 15pt Arial black;
        padding: 10px;
        border: 5pt solid yellow;
      }
    </style>
  </head>
  <body>
    
    <div id="text">
      Don't Kill Your Self<br/>
      Not at least others..<br/>
    </div>
  </body>
</html>
```

Challenger 7: Sprites

Original Picture



Output

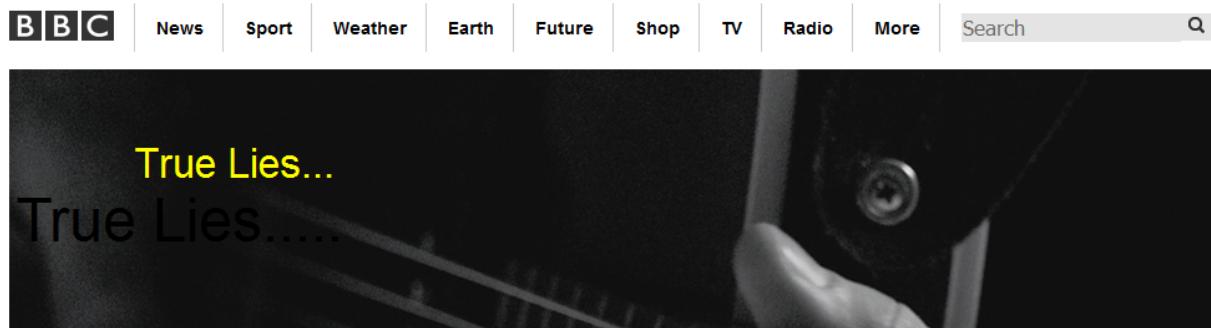


Check mark must appear whenever mouse pointer is placed on a part of image

Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <style>
.button{
    list-style: none;
    margin: 0;
    padding: 0;
}
.button li{
    width: 50px;
    height: 50px;
    display: inline-block;
    background-color: #aaa;
    margin-right: 10px;
}
.button li.info{
    background: url("./images/sprite.png) no-repeat;
}
.button li.multi{
    background: url("./images/sprite.png) no-repeat -50px;
}
.button li.fsat{
    background: url("./images/sprite.png) no-repeat -100px;
}
.button li:hover{
    background: url("./images/sprite.png) no-repeat -100px;
}
        </style>
    </head>
    <body>
        <ul class="button">
            <li class="info"></li>
            <li class="multi"></li>
            <li class="fast"></li>
        </ul>
    </body>
</html>
```

Challenger 8: BBC Header



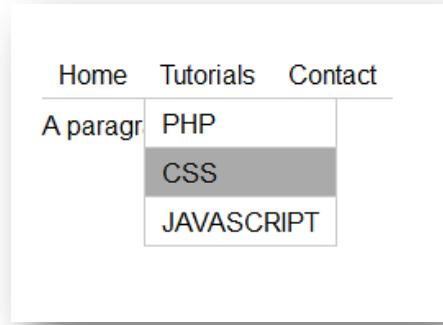
Solution:

```
<!doctype html>
<html lang="en">
<head>
    <title>
        Demo of simple div
    </title>
    <style>
        .fixedwidth
        {
            width:1000px;
            margin:0 auto;
        }
        #header
        {
            position: relative;
            background-color: #ffffff;
            height:45px;
            width:100%;
            margin: 0 auto;
        }
        table td
        {
            padding:5px 15px 5px 15px;
            font:bold 0.9em arial;
            border-right: 1px solid #cccccc;
        }
        #searchbox
        {
            font-size: 13pt;
            width:200px;
            border-style: none;
            background-color: rgb(228,228,228);
            background-image: url("images/magnifier.png");
            background-repeat: no-repeat;
            background-position: right;
        }
        .clear
        {
            clear: both;
        }
        #himage
```

```
{  
    position: absolute;  
    margin: 0 auto;  
}  
#htext  
{  
    position: absolute;  
    padding: 93px 0px 0px 165px;  
    font: 3em sans-serif;  
    color: black;  
    z-index: 10;  
}  
  
#headimage  
{  
    position: relative;  
    left: 18px;  
    top: 10px;  
}  
#imgtext{  
    position: absolute;  
    color: yellow;  
    top: 55px;  
    left: 100px;  
    font: 25pt arial;  
}  
}  
</style>  
</head>  
<body>  
    <div class="fixedwidth">  
        <div id="header">  
            <table>  
                <tr>  
                    <td>  
                          
                    </td>  
                    <td>  
                        News  
                    </td>  
                    <td>  
                        Sport  
                    </td>  
                    <td>  
                        Weather  
                    </td>  
                    <td>  
                        Earth  
                    </td>  
                    <td>  
                        Future  
                    </td>  
                    <td>  
                        Shop  
                    </td>  
                    <td>  
                        TV  
                    </td>  
                </tr>  
            </table>  
        </div>  
    </div>
```

```
<td>
    Radio
</td>
<td>
    More
</td>
<td style="border-right: none;">
    <input type="text" placeholder="Search" id="searchbox"/>
</td>
</tr>
</table>
</div>
</div>
<div class="clear"/>
<div class="fixedwidth">
    <div id="headimage">
        
        <div id="imgtext">
            True Lies...
        </div>
    </div>
</div>
<div class="clear"/>
</body>
</html>
```

Challenger 9: Pure CSS Dropdown Menu



Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
#container{
    margin: 50px;
}

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

```
    font: 12pt Arial;
}
ul#nav li{
    background: #fff;
    float: left;
}
ul#nav li a{
    display: block;
    padding: 5px 10px;
    color: #000;
    text-decoration: none;
    border-bottom: 1px solid #ccc;
}
ul#nav li a:hover{
    background: #aaa;
}
ul#nav li ul li{
    float: none;
    border-left: 1px solid #ccc;
    border-right: 1px solid #ccc;
}
ul#nav li ul{
    position: absolute;
    display: none;
}
ul#nav li:hover ul{
    display: block;
}
</style>
</head>

<body>
    <div id="container">
        <ul id="nav">
            <li>
                <a href="#">Home</a>
            </li>
            <li>
                <a href="#">Tutorials</a>
                <ul>
                    <li><a href="#">PHP</a></li>
                    <li><a href="#">CSS</a></li>
                    <li><a href="#">JAVASCRIPT</a></li>
                </ul>
            </li>
            <li>
                <a href="#">Contact</a>
                <ul>
                    <li><a href="#">My Email</a></li>
                    <li><a href="#">My Telephone</a></li>
                </ul>
            </li>
        </ul>

        <div>
            <br>
            <p style="font:12pt Arial;">
                A paragraph of text here.
            </p>
        </div>
    </div>

```

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

Challenger 10: Simple website layout



Solution:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <style>
body{
    font: 0.8em Arial,sans-serif;
}
h1, ul{
    margin: 0;
    padding: 0;
}

.clear{
    clear: both;
}

#wrap{
    width: 940px;
    padding: 10px;
    margin: 0 auto;
}
#header{
    margin-bottom: 10px;
}
#menu ul{
    list-style: none;
    border-bottom: 1px solid black;
    margin-bottom: 5px;
    float: left;
}
```

```
}

#menu ul li{
    float: left;
}
#content{
    width: 680px;
    float: left;
    padding: 10px;
}
#side{
    width: 219px;
    float: right;
    padding: 10px;
    border-left: 1px solid #aaa;
}
#content,#side{
    min-height: 200px;
}
#menu ul li a{
    text-decoration: none;
    color: #000;
    display: block;
    padding: 0 30px;
    line-height: 30px;
}
#menu ul li a:hover, #menu ul li a.selected{
    background: #000;
    color: #fff;
}

#content h1{
    font-size: 1.6em;
}
#footer{
    background: #000;
    color: #fff;
    padding: 10px 10px;
}
</style>
</head>
<body>
    <div id="wrap">
        <div id="header">
            <div id="logo"><h2>Igate Solutions</h2></div>
        </div>
        <div id="menu">
            <ul>
                <li> <a href="#">Home</a></li>
                <li><a href="#" class="selected">About Us</a></li>
                <li><a href="#">Contact Us</a></li>
                <li><a href="#">Refer a friend</a></li>
            </ul>
            <div class="clear"></div>
        </div>

        <div id="main">
            <div id="content">
                <h1>This is our site</h1>
            </div>
        </div>
    </div>
```

```
<div id="side"> This is a side bar.</div>
<div class="clear"/>
</div>
<div id="footer">
    This is footer
</div>
</div>
</body>
</html>
```

 **Notes**

Notes

Box-shadow (CSS3)

It is used to set the shadow to the div or any object

Syntax:

Box-shadow: h-shadow v-shadow blur spread color

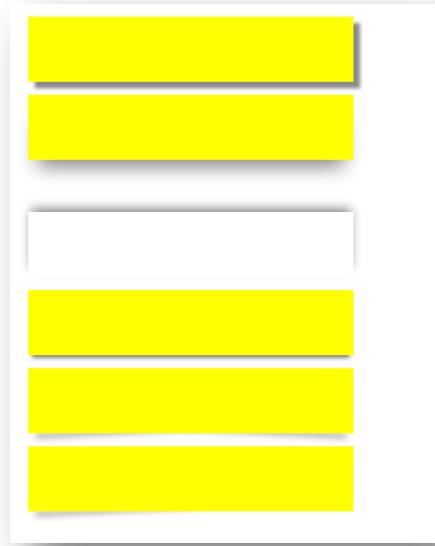
Values	Description
h-shadow	Required. The position of the horizontal shadow. Negative values are allowed
v-shadow	Required. The position of the vertical shadow. Negative values are allowed
blur	Optional. The blur distance
spread	Optional. The size of shadow. Negative values are allowed
color	Optional. The color of the shadow. The default value is black. Look at CSS Color Values for a complete list of possible color values.
Inset	Optional. Changes the shadow from an outer shadow (outset) to an inner shadow

Examples

```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        .box{
            width: 500px;
            height: 100px;
            background-color: yellow;
            margin: 20px;
        }
        .shadow1{
            box-shadow: 10px 10px 5px #888;
        }
        .shadow2{
            box-shadow: 0px 25px 30px -15px rgba(128,128,128,1);
        }
        .shadow3{
            box-shadow: 0px -9px 16px -5px #777;
        }
        .shadow4{
            box-shadow: 0 10px 6px -6px #777;
        }
        .shadow5{
            position: relative;
        }
        .shadow5:before, .shadow5:after {
            z-index: -1;
            position: absolute;
            content: "";
            bottom: 15px;
```

```
left: 10px;
width: 50%;
top: 80%;
max-width:700px;
background: #777;
box-shadow: 0 15px 10px #777;
-webkit-transform: rotate(-3deg);
-moz-transform: rotate(-3deg);
-o-transform: rotate(-3deg);
-ms-transform: rotate(-3deg);
transform: rotate(-3deg);
}
.shadow5:after {
-webkit-transform: rotate(3deg);
-moz-transform: rotate(3deg);
-o-transform: rotate(3deg);
-ms-transform: rotate(3deg);
transform: rotate(3deg);
right: 10px;
left: auto;
}
.shadow6 {
position: relative;
}
.shadow6:before {
z-index: -1;
position: absolute;
content: "";
bottom: 15px;
left: 10px;
width: 50%;
top: 80%;
max-width:300px;
background: #777;
-webkit-box-shadow: 0 15px 10px #777;
-moz-box-shadow: 0 15px 10px #777;
box-shadow: 0 15px 10px #777;
-webkit-transform: rotate(-3deg);
-moz-transform: rotate(-3deg);
-o-transform: rotate(-3deg);
-ms-transform: rotate(-3deg);
transform: rotate(-3deg);
}
</style>
</head>
<body>
<div class="box shadow1">
</div>
```

```
<div class="box shadow2">
</div>
<br><br>
<div class="box shadow3" style="background-color: white">
</div>
<div class="box shadow4">
</div>
<div class="box shadow5">
</div>
<div class="box shadow6">
</div>
</body>
</html>
```



Text-shadow (CSS3)

It is a CSS attribute used to set the shadow to the text

Syntax:

`text-shadow: h-shadow v-shadow blur-radius color, another style...`

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

Example:

```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        .box{
            width:300px;
            height:250px;
            float: left;
            margin: 10px;
        }
        .style1 h1 {
            font:8em Arial;
            /*color: rgba(0,0,0,0.6);
            text-shadow: 2px 2px 3px rgba(255,255,255,0.1);*/
            text-shadow: 0px -2px 0px #333, 0px 2px 3px #666;
            /*color: #111;
            text-shadow: 0px 2px 3px #666;*/
        }
        .style2 h1{
            font:8em Arial;
            color: rgba(0,0,0,0.6);
            text-shadow: 2px 8px 6px rgba(0,0,0,0.2),0px -5px 35px rgba(255,255,255,0.3);
        }
        .style3 h1{
            font:8em Arial;
            color: rgba(10,60,150, 0.8);
            text-shadow: 1px 4px 6px #def, 0 0 0 #000, 1px 4px 6px #def;
        }
        .style4 h1{
            font:8em Arial;
            color: white;
            text-shadow: 0 0 10px #fff,
            0 0 20px #fff,
            0 0 30px #fff,
            0 0 40px #ff00de,
            0 0 70px #ff00de,
            0 0 80px #ff00de,
            0 0 100px #ff00de,
            0 0 150px #ff00de;
        }
    </style>
</head>
<body>
    <div class="box style1" style="background-color:#444;">
        <h1>igate</h1>
    </div>
    <div class="box style2" style="background-color:#666;">
```

```
<h1>igate</h1>
</div>
<div class="box style3" style="background-color:#def;">
<h1>igate</h1>
</div>
<div class="box style4" style="background-color:#000;padding: 50px;text-align: center;">
<h1>igate</h1>
</div>
</body>
</html>
```



Drawing shapes

We can draw different shapes by using the combination of properties like width, height, border-top, border-bottom, border-left, border-right and transparent etc..

Example:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
<style>
.shape{
  float: left;
  margin: 10px;
}
.circle{
  width:100px;
  height:100px;
  border-radius: 50%;
  background-color: gray;
```

```
        }
.ovel{
    width:150px;
    height:100px;
    border-radius: 50%;
    background-color: gray;
}
.up-triangle{
    width:0px;
    height:0px;
    border-left: 50px solid transparent;
    border-right: 50px solid transparent;
    border-bottom: 100px solid gray;
}
.top-left-triangle{
    width: 0;
    height: 0;
    border-top: 100px solid gray;
    border-right: 100px solid transparent;
}
</style>
</head>
<body>
    <div class="circle shape">
    </div>
    <div class="ovel shape">
    </div>
    <div class="up-triangle shape">
    </div>
    <div class="top-left-triangle shape">
    </div>
</body>
</html>
```

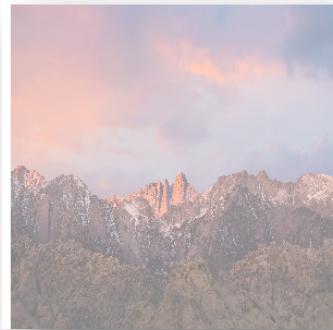


Opacity

We can set the opacity of an image using opacity property

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
</head>
<style>
.o{
  width: 300px;
  height: 300px;
  opacity: 0.5;
}
</style>
<body>
  
</body>
</html>
```



Transform (CSS3)

It is the way to change the size, scale and skew of a shape. CSS3 supports both 2D and 3D transforms.

2D transform: CSS3 provides the following transform methods to change the size, scale and skew of a shape.

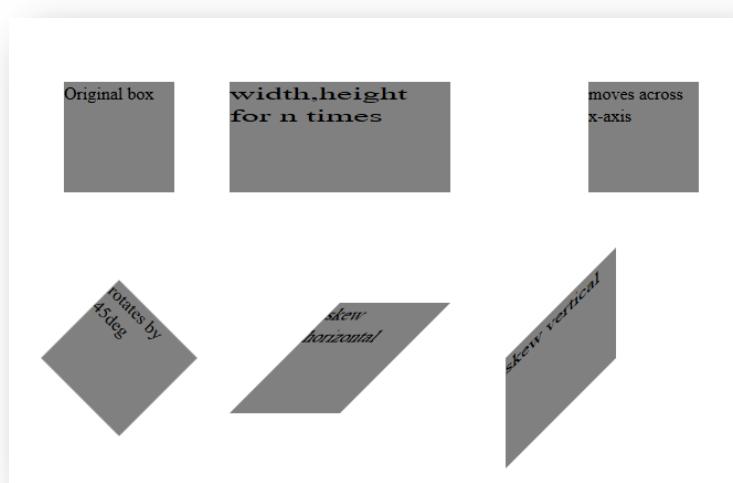
Function	Description
scale(x,y)	Defines a 2D scale transformation, changing the elements width and height
scaleX(n)	Defines a 2D scale transformation, changing the element's width
scaleY(n)	Defines a 2D scale transformation, changing the element's height
translate(x,y)	Defines a 2D translation, moving the element along the X- and the Y-axis
translateX(n)	Defines a 2D translation, moving the element along the X-axis

<code>translateY(n)</code>	Defines a 2D translation, moving the element along the Y-axis
<code>rotate(angle)</code>	Defines a 2D rotation, the angle is specified in the parameter
<code>skew(x-angle,y-angle)</code>	Defines a 2D skew transformation along the X- and the Y-axis
<code>skewX(angle)</code>	Defines a 2D skew transformation along the X-axis
<code>skewY(angle)</code>	Defines a 2D skew transformation along the Y-axis
<code>matrix(n,n,n,n,n,n)</code>	Defines a 2D transformation, using a matrix of six values

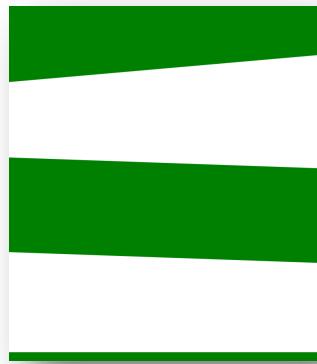
Example:

```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        .box{
            width: 100px;
            height: 100px;
            background-color: gray;
            margin: 50px;
            float: left;
        }
        .style1{
            -webkit-transform:scale(2,1); /* width,height for n times*/
            -moz-transform: scale(2,1);
            transform:scale(2,1);
        }
        .style2{
            -webkit-transform:translate(75px,0px); /* moves across x-axis*/
            -moz-transform:translate(75px,0px);
            transform:translate(75px,0px);
        }
        .style3{
            -webkit-transform:rotate(45deg); /* rotates by 45deg */
            -moz-transform:rotate(45deg);
            transform:rotate(45deg);
        }
        .style4{
            -webkit-transform: skewX(-45deg); /*skew horizontal */
            -moz-transform: skewX(-45deg);
            transform: skewX(-45deg);
        }
        .style5{
            -webkit-transform: skewY(-45deg); /*skew vertical */
            -moz-transform: skewY(-45deg);
            transform: skewY(-45deg);
        }
    </style>
```

```
</head>
<body>
  <div class="box">
    Original box
  </div>
  <div class="box style1">
    width,height for n times
  </div>
  <div class="box style2">
    moves across x-axis
  </div>
  <div class="box style3">
    rotates by 45deg
  </div>
  <div class="box style4">
    skew horizontal
  </div>
  <div class="box style5">
    skew vertical
  </div>
</body>
</html>
```



Example: Skew web page layout with transform



```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        html,body{
            margin: 0;
            padding: 0;
            width: 100%;
        }
        .section{
            position: relative;
            width: 100%;
            padding: 15% 0;
        }
        .trans1{
            margin-top: -10%;
            background-color: green;
            -moz-transform: skew(0,-5deg);
        }
        .trans2{
            background-color: green;
            -moz-transform: skew(0,2deg);
        }
        .trans3{
            background-color: green;
            padding: 2% 0%;
        }
    </style>
</head>
<body>
    <div class="section trans1">
        </div>
    <div class="section trans2">
        </div>
    <div class="section trans3">
        </div>
    </div>
</body>
</html>
```

```
<div class="section">  
</div>  
<div class="section trans2">  
</div>  
<div class="section">  
</div>  
<div class="section trans3">  
</div>  
</body>  
</html>
```

3D Transform: CSS3 provides the following transform methods to change the size, scale and skew of a shape.

Function	Description
<code>rotateX(angle)</code>	Defines a 3D rotation along the X-axis
<code>rotateY(angle)</code>	Defines a 3D rotation along the Y-axis
<code>rotateZ(angle)</code>	Defines a 3D rotation along the Z-axis

Example:

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Page Title</title>  
    <style>  
        .box{  
            width: 100px;  
            height: 100px;  
            background-color: gray;  
            margin: 50px;  
            float: left;  
            box-shadow: 10px 10px 5px 5px red;  
        }  
        .style1{  
            -webkit-transform: rotateX(45deg);  
            -moz-transform: rotateX(45deg);  
            transform: rotateX(45deg);  
        }  
        .style2{  
            -webkit-transform: rotateY(45deg);  
            -moz-transform: rotateY(45deg);  
            transform: rotateY(45deg);  
        }  
    </style>  
</head>  
<body>  
    <div class="box" style="background-color: red; width: 100px; height: 100px; margin: 50px; float: left; box-shadow: 10px 10px 5px 5px red; -webkit-transform: rotateX(45deg); -moz-transform: rotateX(45deg); transform: rotateX(45deg);></div>  
    <div class="box" style="background-color: blue; width: 100px; height: 100px; margin: 50px; float: left; box-shadow: 10px 10px 5px 5px red; -webkit-transform: rotateY(45deg); -moz-transform: rotateY(45deg); transform: rotateY(45deg);></div>  
</body>
```

```
.style3{  
    -webkit-transform: rotateZ(45deg);  
    -moz-transform: rotateZ(45deg);  
    transform:rotateZ(45deg);  
}  
</style>  
</head>  
<body>  
    <div class="box">  
        Original box  
    </div>  
    <div class="box style1">  
  
    </div>  
    <div class="box style2">  
  
    </div>  
    <div class="box style3">  
  
    </div>  
</body>  
</html>
```

Transitions: (CSS3)

Transition is the change in the state of an element while performing events like **:hover**, **:focus**, **:active**, **:target**, **:checked**, **:disabled**, etc.

According to <https://www.w3.org/TR/css3-transitions/> transitions can be applied to the following properties

background-color	as color
background-position	as repeatable list of simple list of length, percentage, or calc
border-bottom-color	as color
border-bottom-width	as length
border-left-color	as color
border-left-width	as length
border-right-color	as color
border-right-width	as length
border-spacing	as simple list of length

border-top-color	as color
border-top-width	as length
bottom	as length, percentage, or calc
clip	as rectangle
color	as color
font-size	as length
font-weight	as font weight
height	as length, percentage, or calc
left	as length, percentage, or calc
letter-spacing	as length
line-height	as either number or length
margin-bottom	as length
margin-left	as length
margin-right	as length
margin-top	as length
max-height	as length, percentage, or calc
max-width	as length, percentage, or calc
min-height	as length, percentage, or calc
min-width	as length, percentage, or calc
opacity	as number
outline-color	as color
outline-width	as length
padding-bottom	as length
padding-left	as length
padding-right	as length
padding-top	as length
right	as length, percentage, or calc
text-indent	as length, percentage, or calc
text-shadow	as shadow list
top	as length, percentage, or calc
vertical-align	as length
visibility	as visibility
width	as length, percentage, or calc
word-spacing	as length
z-index	as integer

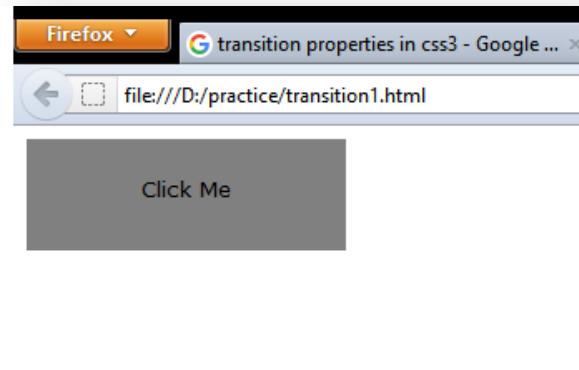
Make sure that always transition declaration is added to initial state, The changing properties would be given with the events like **:hover, :focus, :active, :target, :checked, :disabled**

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
  <style>
    .btn{
```

```
width: 150px;  
height: 25px;  
background-color: gray;  
text-align: center;  
font:10pt Verdana;  
padding: 20px;  
border: 1px solid gray;  
/* transition declarations */  
-moz-transition-property: background-color;  
-moz-transition-duration: 1s;  
}  
.btn:hover{  
background-color: #ddd;  
}  
</style>  
</head>  
<body>  
<div class="btn">  
Click Me  
</div>  
</body>  
</html>
```

Output:



While defining transition definition vendor prefixes and generic property of transition need to be specified to support across all the browsers.

Example:

```
-moz-transition-property: background-color; //for mozilla firefox  
-o-transition-property: background-color; //for opera  
-ms-transition-property: background-color; //for Microsoft internet explorer  
-webkit-transition-property: background-color; //for google chrome
```

```
transition-property: background-color; //generic
```

Transition behavioral properties:

A transition behavior can be decided by different properties like

1. transition-property
2. transition-duration
3. transition-timing-function
4. transition-delay

1. transition-property:

We can specify one or more properties that need to be changed on an event with comma (,) separation

```
/* changing one property */
```

```
transition-property: background-color;
```

```
/* changing multiple properties */
```

```
transition-property: background-color, color;
```

2. transition-duration:

It specifies the length of time that a transition takes. It must be specified in seconds.

```
/* changing one property */
```

```
transition-property: background-color;  
transition-duration:1s;
```

```
/* changing multiple property*/
```

```
transition-property: background-color, color;  
transition-duration:1s, 2s;
```

3. transition-delay:

It specifies when the transition must be started after event got fired. If it is zero then transition starts immediately otherwise wait for specified number of seconds.

```
/*changing one property*/
```

```
-moz-transition-property: background-color;  
-moz-transition-duration: 0.5s;  
-moz-transition-delay: 0.1s;
```

```
/*changing multiple properties*/  
-moz-transition-property: background-color,color;  
-moz-transition-duration: 0.5s,0.5s;  
-moz-transition-delay: 0.1s,0.1s;
```

4. Transition-timing-function

The ‘transition-timing-function’ property describes how the intermediate values used during a transition will be calculated. It allows for a transition to change speed over its duration. These effects are commonly called easing functions. In either case, a mathematical function that provides a smooth curve is used.

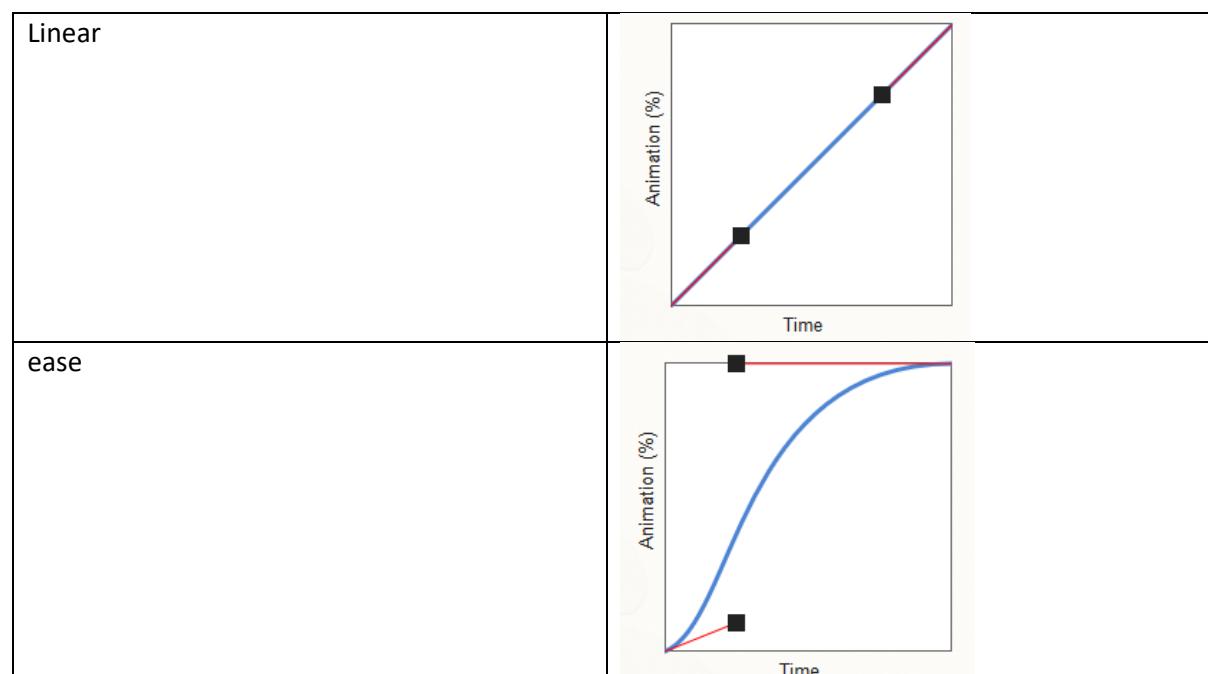
Note: To find custom transition timing functions visit <https://matthewlein.com/ceaser/>

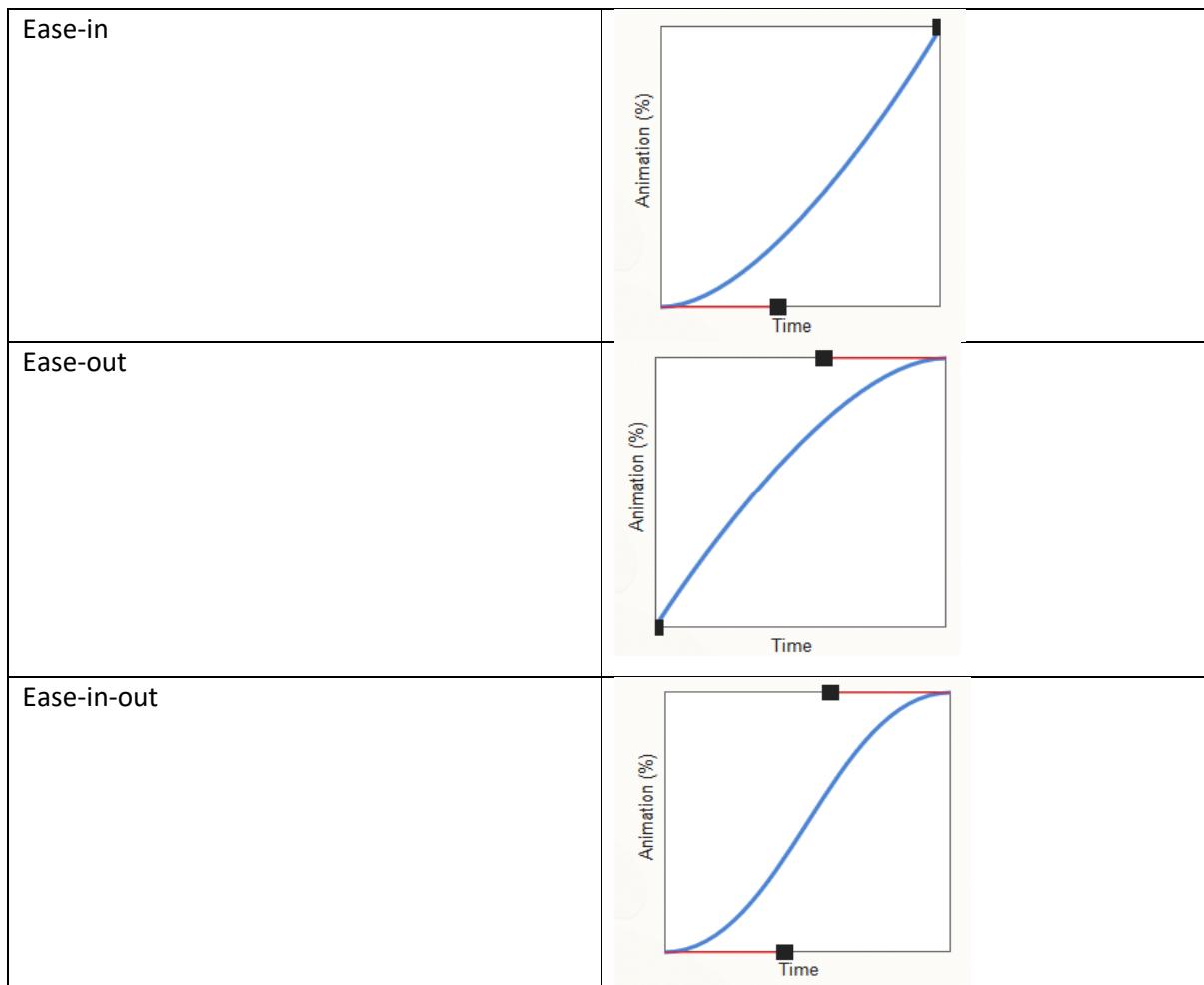
```
/*changing one property*/
```

```
-moz-transition-property: background-color;  
-moz-transition-duration: 0.5s;  
-moz-transition-timimg-funtion:linear;  
-moz-transition-delay: 0.1s;
```

```
/*changing multiple properties*/
```

```
-moz-transition-property: background-color,color;  
-moz-transition-duration: 0.5s,0.5s;  
-moz-transition-timimg-funtion:linear,linear;  
-moz-transition-delay: 0.1s,0.1s;
```





Transition short hand property:

Transition: property | duration | timing-function | delay,

Note: Multiple transitions can be set with comma separation.

```
/*changing one property*/
```

```
-moz-transition:background-color 0.3s linear 0.1s;
```

```
/*changing multiple properties*/
```

```
-moz-transition-property: -moz-transition: background-color 0.3s linear 0s,color 0.3s linear 0s;
```

Using all:

All can be used to change multiple properties at a time if they have same duration, time function and delay values

```
-moz-transition: all 0.3s linear 0s;
```

Example: Hovering button effect

Before



After :



background would be changed as black and text color would be changed as white on focus

```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        html,body{
            margin: 0px;
            padding: 0px;
            width: 100%;
        }
        .b-style{
            top: 100px;
            width: 100px;
            height: 25px;
            padding: 10px;
            margin: 10px;
            display: inline-block;
            text-decoration: none;
            background-color: gray;
            color: black;
            border: 1px solid black;
            border-radius: 5px;
            text-align: center;
            font: 12pt Arial;
            cursor: pointer;

            /* -moz-transition-property: background-color,color;
            -moz-transition-duration: 0.3s,0.3s;
            -moz-transition-timing-function: linear,linear;
            -moz-trasition-delay:0s,0s; */

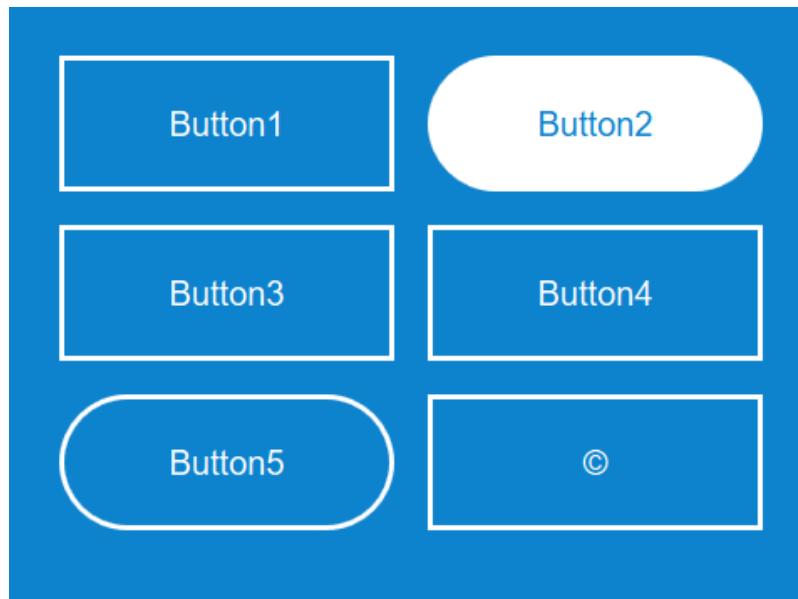
            -moz-transition:all 0.3s linear 0s;

        }
    </style>
</head>
<body>
    <a href="#" class="b-style">Hello</a>
</body>

```

```
.b-style:hover{  
    background-color: black;  
    color: white;  
}  
</style>  
</head>  
<body>  
    <a class="b-style" href="#">  
        Hello  
    </a>  
</body>  
</html>
```

Example: Button animations



Button1: background changes to white and text color changes to blue on hover

Button2: background color changes to transparent and text color changes to white on hover

Button3: Appears box-shadow on hover

Button4: white background will slide from top and fill on hover

Button5: A ">" symbol will slide out beside "button5" as "button5>"

Button6: Symbol will slide-out and "Button6" will appear

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Page Title</title>  
    <meta charset="utf-8">  
    <style>  
        #section1{
```

```
background-color: rgb(14,131,205);
width: 100%;
height: 300px;
padding: 150px;
position: relative;
}

.btn{
    width: 200px;
    border: 3px solid white;
    font: 15pt Arial;
    padding: 25px;
    text-align: center;
    text-decoration: none;
    box-sizing: border-box;
    float: left;
    margin: 10px;
    cursor: pointer;
    position: relative;
}
/* button1 */

.btn1{
    background-color: transparent;
    color: white;
    -webkit-transition:all 0.3s;
    -moz-transition:all 0.3s;
    transition: all 0.3s;
}
.btn1:hover{
    background-color: white;
    color: rgb(14,131,205);
}

/* button 2 */

.btn2{
    border-radius: 50px;
    background-color: white;
    color: rgb(14,131,205);
    -webkit-transition:all 0.3s;
    -moz-transition:all 0.3s;
    transition: all 0.3s;
}
.btn2:hover{
    color: white;
```

```
background-color:transparent;
}

/* button3 */
.btn3{
    background-color: transparent;
    color: white;
    -webkit-transition:all 0.3s;
    -moz-transition:all 0.3s;
    transition: all 0.3s;
}
.btn3:hover{
    -webkit-box-shadow: 0px 0px 20px 5px rgba(0,0,0,0.4);
    -moz-box-shadow:0px 0px 20px 5px rgba(0,0,0,0.4);
    box-shadow:0px 0px 20px 5px rgba(0,0,0,0.4);
}

/* button4 */
.btn4{
    color: white;
    z-index: 1;
    -webkit-transition: all 0.3s;
    -moz-transition: all 0.3s;
    transition: all 0.3s;
}
.btn4:hover{
    color:rgb(14,131,205);
    z-index: 1;
}
.btn4:after{
    content: "";
    position: absolute;
    background-color: white;
    width: 100%;
    left: 0;
    top:0;
    height:0;
    z-index: -1;
    -webkit-transition: all 0.3s;
    -moz-transition: all 0.3s;
    transition: all 0.3s;
}
.btn4:hover:after{
    height: 100%;
    z-index: -1;
}
/* button5 */

```

```
.btn5{  
    border-radius: 50px;  
    color: white;  
    background-color: rgb(14,131,205);  
    -webkit-transition: all 0.3s;  
    -moz-transition: all 0.3s;  
    transition: all 0.3s;  
}  
.btn5:hover{  
    background-color: rgb(14,131,150);  
}  
.btn5:before{  
    content: '>';  
    font:20pt "Courier New";  
    line-height:1;  
    position: absolute;  
    height: 100%;  
    color: #fff;  
    left: 60%;  
    opacity: 0;  
    -webkit-transition: all 0.3s;  
    -moz-transition: all 0.3s;  
    transition: all 0.3s;  
}  
.btn5:hover:before{  
    left:70%;  
    opacity: 1;  
}  
/* button6 */  
.btn6{  
    background: rgb(14,131,205);  
    color: #fff;  
    overflow: hidden;  
    -webkit-backface-visibility: hidden;  
    -moz-backface-visibility: hidden;  
    backface-visibility: hidden;  
}  
.btn6:before {  
    content: "Button";  
    position: absolute;  
    height: 100%;  
    width: 100%;  
    left: 0;  
    top: -100%;  
    line-height:3.5;  
    -webkit-transition: all 0.3s;  
    -moz-transition: all 0.3s;  
    transition: all 0.3s;
```

```
        }
    .btn6:hover:before {
        top: 0;
    }
    .btn6 span{
        display: inline-block;
        width: 100%;
        height: 100%;
        -webkit-backface-visibility: hidden;
        -moz-backface-visibility: hidden;
        backface-visibility: hidden;
        -webkit-transition: all 0.3s;
        -moz-transition: all 0.3s;
        transition: all 0.3s;

    }
    .btn6:hover span {
        -webkit-transform: translateY(300%);
        -moz-transform: translateY(300%);
        -ms-transform: translateY(300%);
        transform: translateY(300%);
    }
</style>
</head>
<body>
    <div id="section1">
        <a class="btn btn1" href="http://www.codingfox.com">
            Button1
        </a>
        <a class="btn btn2">
            Button2
        </a>
        <a class="btn btn3">
            Button3
        </a>
        <a class="btn btn4">
            Button4
        </a>
        <a class="btn btn5">
            Button5
        </a>
        <a class="btn btn6">
            <span>
                &copy;
            </span>
        </a>
    </div>
```

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

Notes

Notes

Example: Image block transitions

Image effect 1: zooming inside (overflow:hidden)

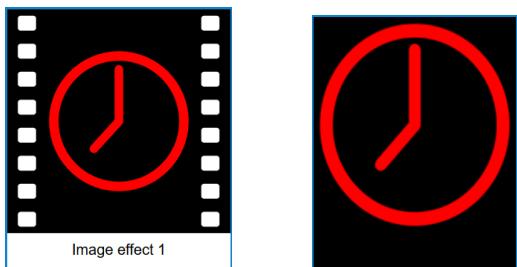


Image effect 2: zooming outside

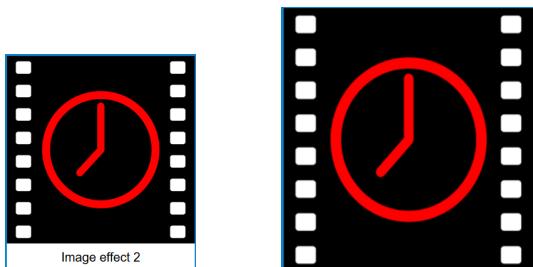


Image effect 3:

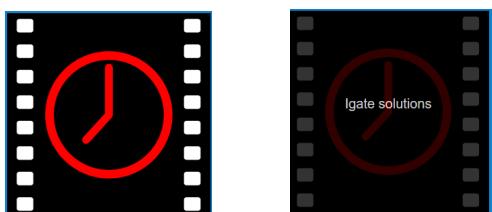
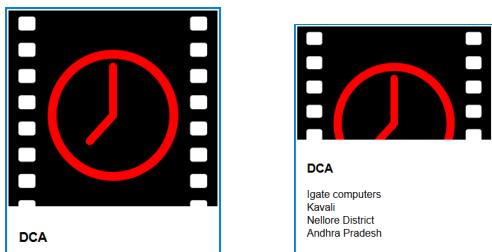


Image effect 4: Sliding up panel



```
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
  <style>
    body{
      width: 100%;
      padding: 0;
      margin: 0;
    }
  </style>
</head>
<body>
```

```
#section1{
    background-color: rgb(14,131,205);
    padding: 50px 10px;
    height: 500px;
}
/* image effects 1&2*/
.img-con{
    max-width: 300px;
    border: 2px solid transparent;
    margin: 40px;
    overflow: hidden;
    float: left;
    cursor: pointer;
}
.img-con img{
    width: 300px;
    height: auto;
    margin: 0;
    -webkit-transition: all ease 0.3s;
    -moz-transition: all ease 0.3s;
    transition: all ease 0.3s;
}
.img-con img:hover{
    -webkit-transform: scale(1.5,1.5);
    -moz-transform: scale(1.5,1.5);
    transform: scale(1.5,1.5);
}
.img-con .footer-con{
    margin-top: -10px;
    background-color: white;
    padding: 15px;
    font: 15pt arial;
    text-align: center;
}
/* image 3 effect */

.effect3-con{
    position: relative;
    width: 300px;
    height: 300px;
    border: 2px solid transparent;
    margin: 40px;
    overflow: hidden;
    float: left;
    cursor: pointer;
}
.effect3-con img{
```

```
position: absolute;
z-index: 1;
width: 100%;
height: auto;
left: 0px;
top: 0px;
}
.effect3-con .block{
position: absolute;
z-index: 2;
background-color: black;
opacity: 0;
width: 100%;
height: 100%;
padding-top: 35%;
left: 0px;
top: 0px;
-moz-transition: all 0.5s ease-in-out;
-webkit-transition: all 0.5s ease-in-out;
transition: all 0.5s ease-in-out;
}
.effect3-con .block p{
color: white;
opacity: 1;
font: 15pt Arial;
text-align: center;
}
.effect3-con:hover .block{
opacity: 0.8;
}

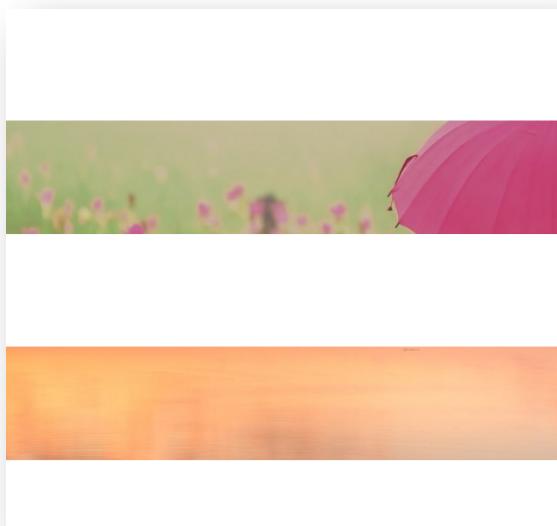
/* image 4 effect */
.effect4-con{
position: relative;
width: 300px;
height: 350px;
border: 2px solid white;
margin: 40px;
overflow: hidden;
float: left;
cursor: pointer;
}
.effect4-con img{
position: absolute;
z-index: 1;
width: 100%;
height: auto;
left: 0px;
```

```
    top: 0px;
}
.effect4-con .block{
    position: absolute;
    z-index: 2;
    background-color: white;
    font: 12pt Arial;
    padding: 5%;
    width: 100%;
    height:150px;
    top:80%;
    -moz-transition: all 0.3s ease;
    -webkit-transition: all 0.3s ease;
    transition: all 0.3s ease;
}
.effect4-con:hover .block{
    top:50%;
}

</style>
</head>
<body>
<div id="section1">
    <div class="img-con">
        
        <div class="footer-con">
            Image effect 1
        </div>
    </div>
    <div class="img-con" style="overflow: visible;">
        
        <div class="footer-con">
            Image effect 2
        </div>
    </div>
    <div class="effect3-con">
        
        <div class="block">
            <p style="color: white;">
                Igate solutions
            </p>
        </div>
    </div>
    <div class="effect4-con">
        
        <div class="block">
            <h3>DCA</h3>
            Igate computers<br>
        </div>
    </div>
</div>
```

```
Kavali<br>
Nellore District<br>
Andhra Pradesh
</div>
</div>
<div style="clear: both;"></div>
</div>
</body>
</html>
```

Parallax Background: Content moves while background images stands still



```
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
    <style>
        html,body{
            margin: 0;
            padding: 0;
            width: 100%;
        }
        .section{
            position: relative;
            width: 100%;
            padding: 10% 0%;
        }
        .parallax{
            width: 100%;
```

```
background-repeat: no-repeat;
background-position: top center;
background-size: cover;
background-attachment: fixed;
}
</style>
</head>
<body>
<div class="section">

</div>
<div class="section parallax" style="background-image: url(./lovers.jpg);>

</div>
<div class="section">

</div>
<div class="section parallax" style="background-image: url(./sunset.jpg);>

</div>
<div class="section">

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

Notes

Notes

Notes