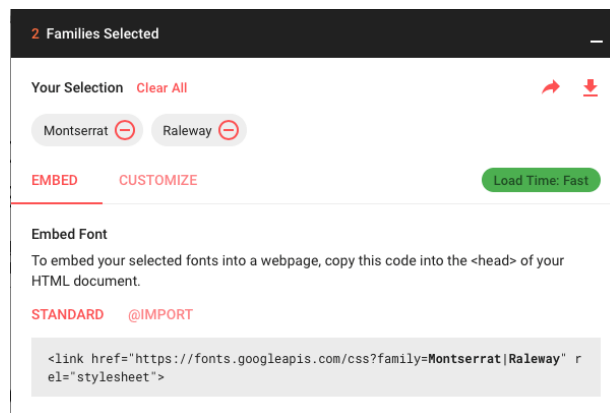


Adding Google Fonts to the webpage

1. Visit the google font official website fonts.google.com
2. Select the required fonts and change the **style, thickness and size** then add font using **+** symbol



3. Copy the fonts link from the google download box



4. Use the Google fonts in the web page using font-family CSS property

<head>

<link href="https://fonts.googleapis.com/css?family=Montserrat|Raleway" rel="stylesheet">

<style>

.container h1{

font:4vw Raleway;

}

.container p{

font:1.5vw Montserrat;

text-align: justify;

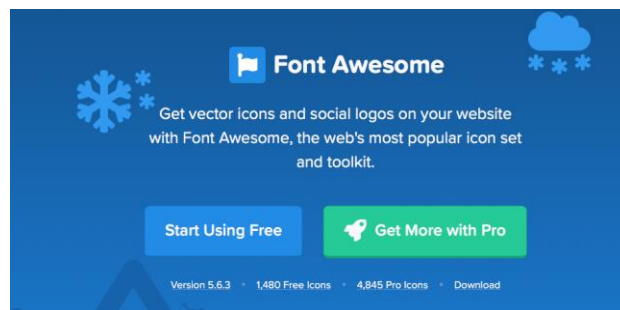
}

</style>

</head>

Using font awesome

1. These are the symbols that we can use for free. To use these symbols, visit the site <https://fontawesome.com>
2. Select start using free to get free fonts



3. Copy the link to clipboard

```
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css" >
```

4. Add the link to the head section of your page and start using font-awesome

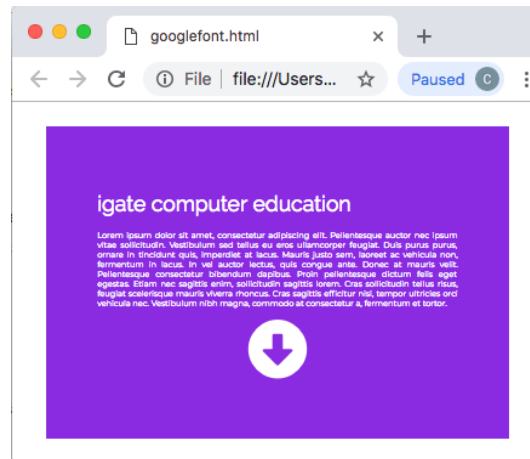
```
<head>
  <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css"
  integrity="sha384-UHRtZLI+pbxtHCWp1t77Bi1L4ZtiqrqD80Kn4Z8NTSRyMA2Fd33n5dQ8lWUE00s/"
  crossorigin="anonymous">
  <style>
    .container .social{
      text-align: center;
    }
  </style>
</head>
-----
<span style="font:50px white">
  <i class="fas fa-arrow-alt-circle-down"></i>
</span>
```

Putting google fonts and font-awesome together

```
<!DOCTYPE html>
<html>
  <head>
    <link href="https://fonts.googleapis.com/css?family=Montserrat|Raleway" rel="stylesheet">
```

```
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.3/css/all.css"
integrity="sha384-UHRtZLI+pbxtHCWp1t77Bi1L4ZtiqrqD80Kn4Z8NTSRyMA2Fd33n5dQ8IWUE00s/"
crossorigin="anonymous">
<style>
  .container{
    color: white;
    background-color: blueviolet;
    margin: 20px;
    padding: 10%;

  }
  .container h1{
    font:4vw Raleway;
  }
  .container p{
    font:1.5vw Montserrat;
    text-align: justify;
  }
  .container .social{
    text-align: center;
  }
</style>
</head>
<body>
  <div class="container">
    <h1>igate computer education</h1>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipiscing elit....
    </p>
    <div class="social">
      <span style="font:50px white">
        <i class="fas fa-arrow-alt-circle-down"></i>
      </span>
    </div>
  </div>
</body>
</html>
```



CSS3

- Initially the name of CSS was CHSS (Cascading HTML Style Sheets), Later it has been simplified as CSS
- The first version of CSS that is CSS 1.0 is released during 1996. It can be used with multiple markup languages
- CSS 2.0 was released during 1998.
- As many browsers was not supporting CSS 2.0, W3C has standardized CSS and released CSS 2.1 during 2004. Many browser manufacturers started supporting CSS 2.1.
- CSS3 is not a new specification of CSS but recommendations from the W3C to address device independent applications. CSS3 given freedom to the browser vendors to have their implementations to the new requirements.

Features of CSS3

1. New selectors
2. Colors
3. Shadows
4. Gradient styles
5. Transitions and transformations
6. Responsive design
7. Multiple backgrounds
8. Transparency

Disadvantage:

- It is so difficult (impossible) to develop complete browser independent applications
- CSS3 is supported by Chrome, Opera and Safari but IE would not give full support

Vendor specific prefixes

Because CSS3 styles are not standardized, vendor specific prefixes must be used to have the same style in all the browsers. These may not be needed once the CSS3 is standardized.

- -webkit- (Chrome, newer versions of Opera.)
- -moz- (Firefox)
- -o- (Old versions of Opera)
- -ms- (Internet Explorer)

Example:

```
#box{
    -webkit-border-radius: 10px;
    -o-border-radius: 10px;
    -ms-border-radius: 10px;
    -moz-border-radius: 10px;
    border-radius: 10px;
}
```

Border-radius

The border-radius property allows to define rounded border corners. The curve of each corner is defined using one or two values.

Syntax:

Border-radius: radius in px em %

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      #box{
        width: 100px;
        height: 75px;
        border: 2px solid black;
        border-radius: 10px; /*sets radius to all the sides */
      }
    </style>
  </head>
  <body>
    <div id="box">
    </div>
```

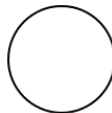
```
</body>  
</html>
```

Examples:

```
/* Radius is set for all 4 sides */  
border-radius: 10px;  
/* equals to */  
border-top-left-radius:10px;  
border-top-right-radius:10px;  
border-bottom-right-radius: 10px;  
border-bottom-left-radius: 10px;
```



```
/* Radius is set for all 4 sides */  
border-radius: 50%;  
/* equals to */  
border-top-left-radius:50%;  
border-top-right-radius:50%;  
border-bottom-right-radius: 50%;  
border-bottom-left-radius: 50%;
```

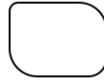


```
/* top-left-and-bottom-right | top-right-and-bottom-left */  
border-radius: 20px 10%;  
/* equals to */  
border-top-left-radius:20px;  
border-top-right-radius:10px;  
border-bottom-right-radius: 20px;  
border-bottom-left-radius: 10px;
```

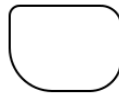


```
/* top-left | top-right-and-bottom-left | bottom-right */  
border-radius: 10px 30px 20px;  
/*equals to */  
border-top-left-radius:10px;
```

```
border-top-right-radius:30px;  
border-bottom-right-radius: 20px;  
border-bottom-left-radius: 30px;
```



```
/* top-left | top-right | bottom-right | bottom-left */  
border-radius: 10px 20px 30px 40px;  
/* equals to */  
border-top-left-radius:10px;  
border-top-right-radius:20px;  
border-bottom-right-radius: 30px;  
border-bottom-left-radius: 40px;
```



Setting unequal radius

Giving two values to a corner radius applies unequal and separate radius

```
border-top-left-radius:50px 20px;
```



Shortcut property for unequal radius

```
border-radius: 15px/30px;  
/* is equivalent to */  
border-top-left-radius: 15px 30px;  
border-top-right-radius: 15px 30px;  
border-bottom-right-radius: 15px 30px;  
border-bottom-left-radius: 15px 30px;
```



box-shadow

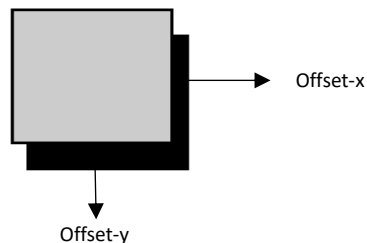
The box-shadow property is used to set shadow or drop shadow to a box or an image. It is also possible to set one or more shadows with comma separation. If a border-radius is specified on the element with a box shadow, the box shadow takes on the same rounded corners.

Syntax:

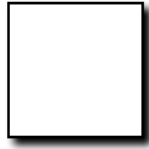
Box-shadow: offset-x | offset-y | blur-radius | spread-radius | color

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      #box{
        width: 100px;
        height: 100px;
        border: 2px solid black;
        background-color: #ccc;
        box-shadow:10px 20px black;
      }
    </style>
  </head>
  <body>
    <div id="box">
    </div>
  </body>
</html>
```



```
/* offset-x | offset-y | spread radius | color
box-shadow:5px 5px 5px black;
```

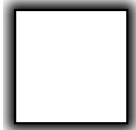
```
box-shadow:-5px -5px 5px black;
```



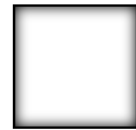
```
box-shadow:0px 0px 10px black;
```



```
box-shadow:0px 0px 10px 5px black;
```



```
box-shadow: inset 0px 0px 10px black;
```

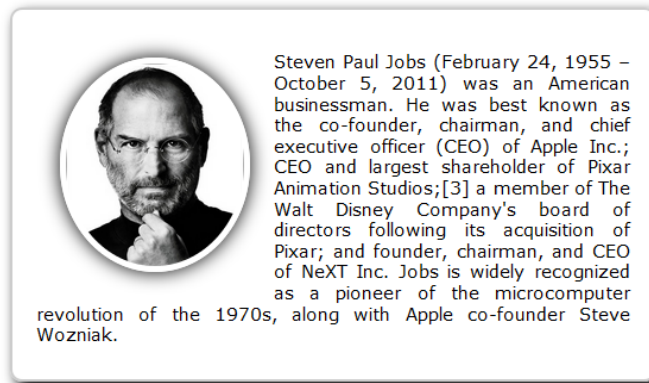


```
/* setting multiple borders */
```

```
box-shadow: inset 0px 0px 10px black, 0px 0px 15px red;
```



Hands on: Boxed text with profile avatar picture

**Solutions:**

```

<html>
<head>
  <meta charset="UTF-8">
  <title>Styling table</title>
  <style>
    #box{
      margin: 100px;
      width: 500px;
      height: 270px;
      border: 2px solid #ccc;
      border-radius: 10px;
      padding: 20px;
      box-shadow: 5px 5px 10px black;
    }
    #img1{
      float: left;
      width: 150px;
      height: 170px;
      margin: 20px;
      border-radius: 50%;
      border: 5px solid white;
      box-shadow: 0px 0px 15px black;
    }
    #box p{
      font:11pt verdana;
      text-align: justify;
      line-height: 1.2em;
    }
  </style>
</head>
<body>

```

```

<div id="box">
  
  <p>
    Steven Paul Jobs (February 24, 1955 – October 5, 2011) was an American businessman. He was
    best known as the co-founder, chairman, and chief executive officer (CEO) of Apple Inc.; CEO and largest
    shareholder of Pixar Animation Studios;[3] a member of The Walt Disney Company's board of directors
    following its acquisition of Pixar; and founder, chairman, and CEO of NeXT Inc. Jobs is widely recognized
    as a pioneer of the microcomputer revolution of the 1970s, along with Apple co-founder Steve Wozniak.
  </p>
</div>
</body>
</html>

```

Text-shadow

The text-shadow property adds shadows to text. It accepts a comma-separated list of shadows to be applied to the text and text-decorations of the element. Each shadow is specified as an offset from the text, along with optional color and blur radius values. Multiple shadows are applied front-to-back, with the first-specified shadow on top.

Syntax: offset-x | offset-y | blur-radius | spread-radius | color

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      #box{
        width:300px;
        height: 50px;
        margin: 100px;
        padding: 20px;
        border: 2px solid #ccc;
      }
      #box p{
        margin: 0px;
        font:bolder 30pt Arial;
        text-shadow: 0.1em 0.1em 0.1em rgba(0,102,102,0.4);
      }
    </style>
  </head>
  <body>
    <div id="box">

```



```

        color:#CCAA3F; /* text color*/
    }
</style>
</head>
<body>
    <div id="box">
        <p class="raised">Igate Solutions</p>
    </div>
</body>
</html>

```



Inset Effect

```

/* dark color upper shadow + back ground drop down shadow*/
text-shadow:-1px -1px #7F744D, 1px 1px #FFE79B;

```



Hands On: Textbox with upper stripe, profile picture and text



Solutions:

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Styling table</title>
  <style>
    #box{
      width:350px;
      margin: 100px;
      border: 2px solid #ccc;
      border-radius: 30px 30px 0px 0px;
      box-shadow: 5px 5px 5px black;
    }
    #box #top{
      border: 2px solid #ccc;
      border-radius: 30px 30px 0px 0px;
      background-color: #EAEAEA;
      font:bold 25pt Arial;
      color:#A4A8AC;
      text-shadow:-1px -1px #8995A2, 1px 1px #EAEAEA;
      padding: 5px 0px 10px 35px;
    }
    #box #img1{
      width: 150px;
      height: 170px;
      border-radius: 50%;
      border: 5px solid white;
      box-shadow: 0px 0px 15px black;
      margin: 20px 0px 10px 100px;
    }
    #box p{
      font:10pt verdana;
      text-indent: 100px;
      text-align: justify;
      line-height: 1.2em;
    }
  </style>
</head>
<body>
  <div id="box">
    <div id="top">
      Apple Computers
    </div>
```

```


<div style="padding: 0px 20px 20px 20px;"><p>
  Steven Paul Jobs (February 24, 1955 – October 5, 2011) was an American businessman. He was
  best known as the co-founder, chairman, and chief executive officer (CEO) of Apple Inc.; CEO and largest
  shareholder of Pixar Animation Studios; a member of The Walt Disney Company's board of directors
  following its acquisition of Pixar; and founder, chairman, and CEO of NeXT Inc.
  <br/><p style="text-align: right;font:bold 1.5em Arial;">-Igate Solutions</p>
</p></div>
</div>
</body>
</html>

```

Generated content

Generating Leading images

By using content attribute and ::before pseudo selector we can set an icon image to either list item or heading

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      body{
        font:12pt Verdana;
        line-height: 1.5em;
      }
      h2.aicon::before{
        content: url(..images/apple.png) ' Mac';
        font:bold 18pt Arial;
        display: inline;
      }
      h2.wicon::before{
        content: url(..images/windows.png) ' Windows';
        font:bold 18pt Arial;
        display: inline;
      }
    </style>
  </head>
  <body>
    <h2 class="aicon"></h2>

```

```

<p>
  Works from Mac Mountain lion onwards
</p>
<h2 class="wicon"></h2>
<p>
  Works from Windows 7 onwards
</p>
</body>
</html>

```

Output:



Works from Mac Mountain lion onwards



Works from Windows 7 onwards

Generating a triangle

We can create a triangle by setting the width and height of an element to 0, and giving it a wide border. If we give each border a different color, they intersect as triangles. Make two of the borders transparent, and we will left with a triangle.

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      .content{
        position: relative;
        top:50px;
        width: 100%;
        height: 200px;
        background-color: #A4A4A4;
      }
      .triag{
        position: absolute;
        top: -19px;
        left: 50px;
        width:0px;
        height:0px;

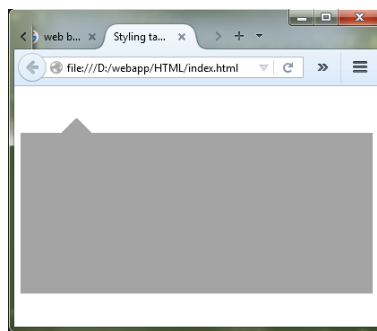
```



```
border-left:20px solid transparent;
border-right:20px solid transparent;
border-bottom: 20px solid #A4A4A4;
}
</style>
</head>
<body>
  <div class="content">
    <div class="triag">

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

Output:



Ribbon effect

Program:

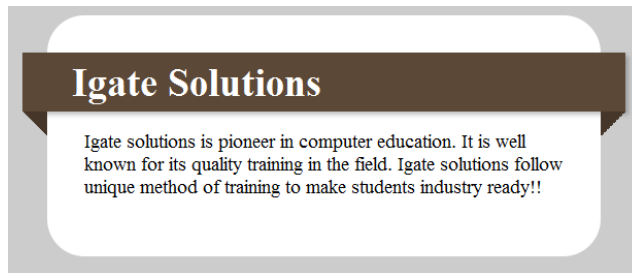
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      body{
        background-color: #ccc;
      }
      #container{
        width: 60%;
        background-color: white;
        border-radius: 30px;
        margin: 0px auto;
        padding: 30px;
```

```

    }
    h1 {
        background-color: #5C4837;
        color: #FFF;
        padding: 5px 40px;
        box-shadow: 3px 3px 3px rgba(51,51,51,0.3);
        margin: 0 -50px;
        position: relative;
    }
    h1::before, h1::after {
        content: "";
        height: 0;
        width: 0;
        border-top: 20px solid #453629;
        position: absolute;
        bottom: -20px;
    }
    h1::before {
        border-left: 20px solid transparent;
        left: 0;
    }
    h1::after {
        border-right: 20px solid transparent;
        right: 0;
    }
</style>
</head>
<body>
    <div id="container">
        <h1>Igate Solutions</h1>
        <p>
            Igate solutions is pioneer in computer education. It is well known for its quality training in the field.
            Igate solutions follow unique method of training to make students industry ready!!
        </p>
    </div>
</body>
</html>

```

Output:



Pagination

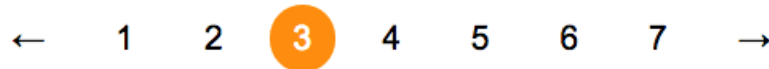
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
    .page{
      list-style: none;

    }
    .page li{
      display: inline;
    }
    .page a{
      display:inline-block;
      text-decoration: none;
      color:black;
      font: 2vw Arial;
      padding:9px 15px;
      margin: 5px;
      background-color:white;
      border-radius: 50%;
      transition:all 0.5s ease 0s;
    }
    .page a:hover{
      background-color: lightgrey;
    }
    .page a.active{
      color:white;
      background-color:darkorange;
    }
  </style>
</head>
```

```

<body>
  <div style="text-align: center">
    <ol class="page">
      <li><a href="#">&leftarrow;</a></li>
      <li><a href="#">1</a></li>
      <li><a href="#">2</a></li>
      <li><a href="#" class="active">3</a></li>
      <li><a href="#">4</a></li>
      <li><a href="#">5</a></li>
      <li><a href="#">6</a></li>
      <li><a href="#">7</a></li>
      <li><a href="#">&rightarrow;</a></li>
    </ol>
  </div>
</body>
</html>

```



Gradient Styling

A gradient creates a smooth transition between two or more colors, and is often used in graphic design to create an illusion of depth or a 3D effect. Before CSS3 background images were used to create gradient effects. With CSS3, we can generate a gradient simply by specifying its colors and shape. CSS supports both linear gradients—in which the colors fade smoothly from one to the next in a straight line—and radial gradients, which emerge from a single point with the colors spreading out in a circle or ellipse. We can also create gradients that repeat the same sequence of colors indefinitely.

Function	Description
linear-gradient()	Sets the direction and colors of a gradient that fades from one color to the next in a straight line. The direction can be set using keywords or an angle. If the direction is omitted, the gradient is drawn from the top to the bottom.
radial-gradient()	Creates a circular or elliptical gradient, and optionally sets its shape, size, and position. If all optional features are omitted, an elliptical gradient is centered in the element's background.
repeating-linear-gradient()	Creates a linear gradient, in which the colors are repeated infinitely in both directions.
repeating-radial-gradient()	Creates a radial gradient that repeats infinitely.

Linear-gradient()

The linear gradient takes two or more arguments that are

1. The angle or direction of the gradient (Optional)
2. A comma separated list of two or more colors

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      #box{
        width: 200px;
        height: 100px;
        margin: 20px;
        background: linear-gradient(red,yellow); /* by default it sets top to bottom */
      }
    </style>
  </head>
  <body>
    <div id="box">
    </div>
  </body>
</html>
```



```
#box{
  width: 200px;
  height: 100px;
  margin: 20px;
  background: linear-gradient(to top left,red,yellow); /* to top left from bottom right */
}
```

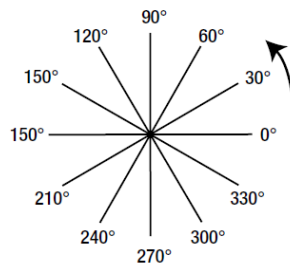


```
#box{
  width: 200px;
  height: 100px;
```

```
margin: 20px;
background: linear-gradient(to top left,red,yellow,green);
}
```



We can also set the angle to specify the gradient direction



```
#box{
width: 200px;
height: 100px;
margin: 20px;
background: linear-gradient(135deg, red, yellow);
}
```



Radial-gradient()

A radial gradient emerges from a single point with the colors spreading outward in a circular or elliptical shape. The radial-gradient() function takes the following arguments:

Shape: The accepted values are circle and ellipse.

Size: This determines the extent of the radial gradient, either using specific dimensions or keywords.

Position: This determines the center of the circle or ellipse. The value is expressed in the same way as preceded by the keyword at. If omitted, it defaults to at center.

Color stops: A comma-separated list of colors with optional positions along the gradient ray starting at the center.

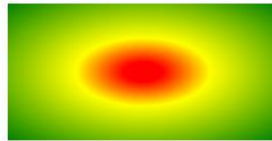
The only required argument is the list of color stops. All other arguments are optional.

```
#box{
width: 200px;
height: 100px;
margin: 20px;
background: radial-gradient(circle at 100px 50px,red 10%,yellow 35%,green);
}
```

```
}
```



```
#box{
  width: 200px;
  height: 100px;
  margin: 20px;
  background: radial-gradient(ellipse at 100px 50px,red 10%,yellow 35%,green);
}
```



Repeating gradients

```
#box{
  width: 200px;
  height: 100px;
  margin: 20px;
  background: repeating-linear-gradient(135deg, #426A77, #FFF 6px);
}
```



```
#box{
  width: 200px;
  height: 100px;
  margin: 20px;
  background: repeating-linear-gradient(to right, #D9CC3C, #FFEB79 5%, #D9CC3C 10%);
}
```



```
#box{
  width: 200px;
  height: 100px;
  margin: 20px;
```

```
background: repeating-radial-gradient(circle at 50% 100%, #FFEB79, #D9CC3C 5%);
}
```



```
#box{
  width: 100px;
  height: 100px;
  margin: 20px;
  background-color: #C51D31;
  border-radius: 50%;
  background-image: radial-gradient(circle 12px at 25px 30px, rgba(255,255,255,0.8),
transparent),radial-gradient(circle at 30% 30%, #C51D31, #921524, #450A11);
}
```



2D Transforms and Transitions

CSS3 provides Transforms and transitions and making it possible to shift, rotate, scale, and skew elements. Originally, there were three separate modules: for two- and three-dimensional transforms, and for scalable vector graphics (SVG). These have been merged into a single module, which is currently only a working draft. However, browsers have made significant progress with two-dimensional transforms. These're supported by all mainstream browsers except IE 8 and earlier.

Property	Initial Value	Description
Transform	None	Specifies how the element is to be transformed. Takes as its value a space-separated list of transform functions
transform-origin	50% 50%	Specifies the anchor point of the transform. Takes one or two lengths, percentages, or keywords representing the horizontal and vertical positions, respectively. If only one value is given, the vertical position defaults to center.

Transform

Specifies how the element is to be transformed. Takes as its value a space-separated list of transform functions. The commonly used functions are.

Function	Description
translateX()	Moves the element along the X axis by the specified length or percentage.

translateY()	Moves the element along the Y axis by the specified length or percentage.
translate()	Moves the element from its default position. Takes two comma separated lengths or percentages. The first value represents the horizontal offset and the second the vertical offset. If only one value is given, the element is moved only horizontally.
rotate()	Rotates the element around its anchor point by the specified angle.
scaleX()	Scales the element along the X axis by the specified scaling factor.
scaleY()	Scales the element along the Y axis by the specified scaling factor.
scale()	Scales the element. Takes one or two numbers as the scaling factor. If one number is given, both axes are scaled by the same multiplier. If two comma separated numbers are given, the first applies to the X axis and the second to the Y axis.
skewX()	Skews the element along the X axis by the specified angle.
skewY()	Skews the element along the Y axis by the specified angle.
matrix()	Specifies a 2D transform using a transform matrix of six values.

Experimental example

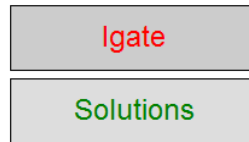
```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Styling table</title>
    <style>
      #first,#second{
        width: 150px;
        font: 15pt Arial;
        padding: 10px;
        text-align: center;
        border: 1pt solid black;
        margin: 5px;
      }
      #first{
        color: red;
        background-color: #ccc;
      }
      #second{
        color: green;
        background-color: #ddd;
      }
    </style>
  </head>
  <body>
    <div id="first">
      Igate
    </div>
    <div id="second">

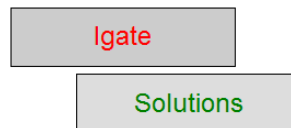
```

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

Output:



```
#second{
  color: green;
  background-color: #ddd;
  transform:translateX(50px); /* moves 50px towards right */
}
```



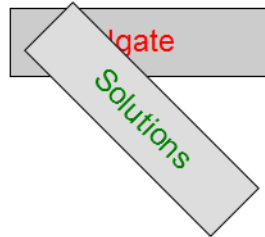
```
#second{
  color: green;
  background-color: #ddd;
  transform:translateY(50px); /* moves 50px towards down */
}
```



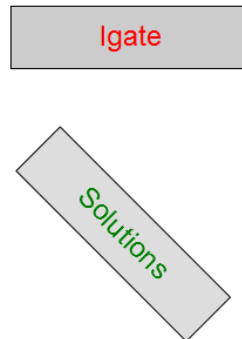
```
#second{
  color: green;
  background-color: #ddd;
  transform:translate(150px,25px); /* moves 150px right and 25px down */
}
```



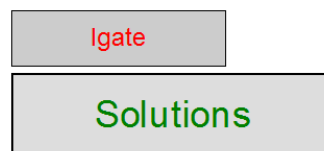
```
#second{
  color: green;
  background-color: #ddd;
  transform: rotate(45deg); /* rotates 45deg from the center */
}
```



```
#second{
  color: green;
  background-color: #ddd;
  transform: rotate(45deg) translateX(50px); /* rotates from top left and 50px away from border */
  transform-origin: left top;
}
```



```
#second{
  color: green;
  background-color: #ddd;
  transform: scale(1.5); /* zooms horizontally and vertically by 150% */
  transform-origin: left top;
}
```

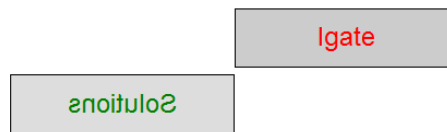


```
#second{
  color: green;
  background-color: #ddd;
```

```

    transform:scaleX(-1);    /* flips horizontally */
    transform-origin: left top;
}

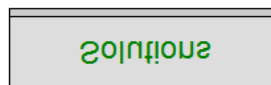
```



```

#second{
    color: green;
    background-color: #ddd;
    transform:scaleY(-1);    /* flips vertically */
    transform-origin: left top;
}

```



```

#second{
    color: green;
    background-color: #ddd;
    transform:skew(45deg); /* skews horizontally by 45deg */
    transform-origin: left top;
}

```



```

<!DOCTYPE html>
<html>
  <head>
    <metta charset="UTF-8">
    <title>Styling table</title>
    <style>
      fieldset{
        width: 500px;
        padding: 10px;
      }
    </style>
  </head>
  <body>
    <fieldset>
      <legend>igate solutions</legend>
      <p>Some text Here.....</p>
    </fieldset>
  </body>
</html>

```

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

Media queries

Media query is a CSS3 technique of rendering element styles according to the size of viewport of device. The following are the standard split points to different devices

```
/*
Device = Desktops
Screen = 1281px to higher resolution desktops
*/
```

```
@media (min-width: 1281px) {
```

```
}
```

```
/*
Device = Laptops, Desktops
Screen = B/w 1025px to 1280px
*/
```

```
@media (min-width: 1025px) and (max-width: 1280px) {
```

```
}
```

```
/*
Device = Tablets, Ipads (portrait)
Screen = B/w 768px to 1024px
*/
```

```
@media (min-width: 768px) and (max-width: 1024px) {
```

```
}
```

```
/*
Device = Tablets, Ipads (landscape)
Screen = B/w 768px to 1024px
*/
```

```
@media (min-width: 768px) and (max-width: 1024px) and (orientation: landscape) {
```

```
}

/*
Device = Low Resolution Tablets, Mobiles (Landscape)
Screen = B/w 481px to 767px
*/

@media (min-width: 481px) and (max-width: 767px) {

}

/*
Device = Most of the Smartphones Mobiles (Portrait)
Screen = B/w 320px to 479px
*/

@media (min-width: 320px) and (max-width: 480px) {

}
```

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <style>
    .box{
      width: 100px;
      height: 100px;
      background-color: red;
      margin: 40px;
    }
    @media (max-width:767px){
      .box{
        background-color: yellow;
      }
    }
  </style>
</head>
```

```

<body>
  <div class="box">

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

```

Result: box color changes from red to yellow on changing the screen size

Flex Box System

This module allows to design flexible responsive layout easily without using float and positioning. In this system items called flex-items are placed in flex-container.

display: flex;

Makes a container flex-container. The following properties are used to further format the flex-container and flex-items

Flex-container properties:

Property	Description	
flex-direction	It defines the direction in which flex-items has to be arranged	
	column	Arranges vertically
	row	*Arranges horizontally
	column-reverse	Arranges vertically in reverse order
	row-reverse	Arranges horizontally down to top
flex-wrap	It specifies whether flex-items must be wrapped or not	
	wrap	*Arranges flex items to down when page size reduced
	nowrap	default
	wrap-reverse	wraps in reverse order
flex-flow	It is the shortcut property with the combination of both flex-direction and flex-wrap Ex: flex-flow: row wrap;	
justify-content	It helps to align the flex items	
	center	*Aligns the flex-items to center
	flex-start	Aligns the flex items from left to right
	flex-end	Aligns the flex items from right to left
	space-around	Occupies the total width by giving additional space around flex items
	space-between	Occupies the total width by giving additional space around flex items
align-items	It is used to align the flex-items vertically	
	center	Middle (sizes the item to the size of contents)
	flex-start	Top
	flex-end	Bottom
	stretch	*Stretches the item vertically to fill the container

	baseline	Arranges to the base line
align-content	It is used to align the flex lines	
	space-between	to specify the space between items
	space-around	to specify the space around the items
	stretch	
	center	
	flex-start	
	flex-end	

Flex-items properties

Property	Description
order	It used to specify the order in which items must be arranged Ex: <code><div style="order: 3">1</div></code>
flex-grow	specifies how much a flex item will grow relative to the rest of the flex items. Ex: <code><div style="flex-grow: 8">3</div></code>
flex-shrink	specifies how much a flex item will shrink relative to the rest of the flex items. Ex: <code><div style="flex-shrink: 0">3</div></code>
flex-basis	specifies the initial length of a flex item. Ex: <code><div style="flex-basis: 200px">3</div></code>
flex	It is the short hand property of flex-grow, flex-shrink and flex-basis Ex: <code><div style="flex: 0 0 200px">3</div></code>
align-self	The align-self property specifies the alignment for the selected item inside the flexible container. The align-self property overrides the default alignment set by the container's align-items property. Ex: <code><div style="align-self: center">3</div></code>

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <style>
      *{
        box-sizing: border-box;
      }
      body{
        background-color: cadetblue;
        margin: 0px;
        padding: 0px;
        width: 100%;
      }
      .container{
        display: flex;

```



```

    flex-direction:row;
    flex-wrap: wrap;
    justify-content:center;
    align-items: stretch;
    background-color: aqua;
    width:100%;
    height:auto;
    padding: 10px;
}
.container .item{
    flex:0 0 22%;
    height: auto;
    background-color: white;
    border-radius: 10px;
    margin: 10px;
    padding: 20px;
    font: 1vw Arial;
}

@media (min-width: 481px) and (max-width: 767px) {
    .container .item{
        flex:0 0 47%;
    }
}

@media (min-width: 320px) and (max-width: 480px) {
    .container .item{
        flex:0 0 98%;
    }
}

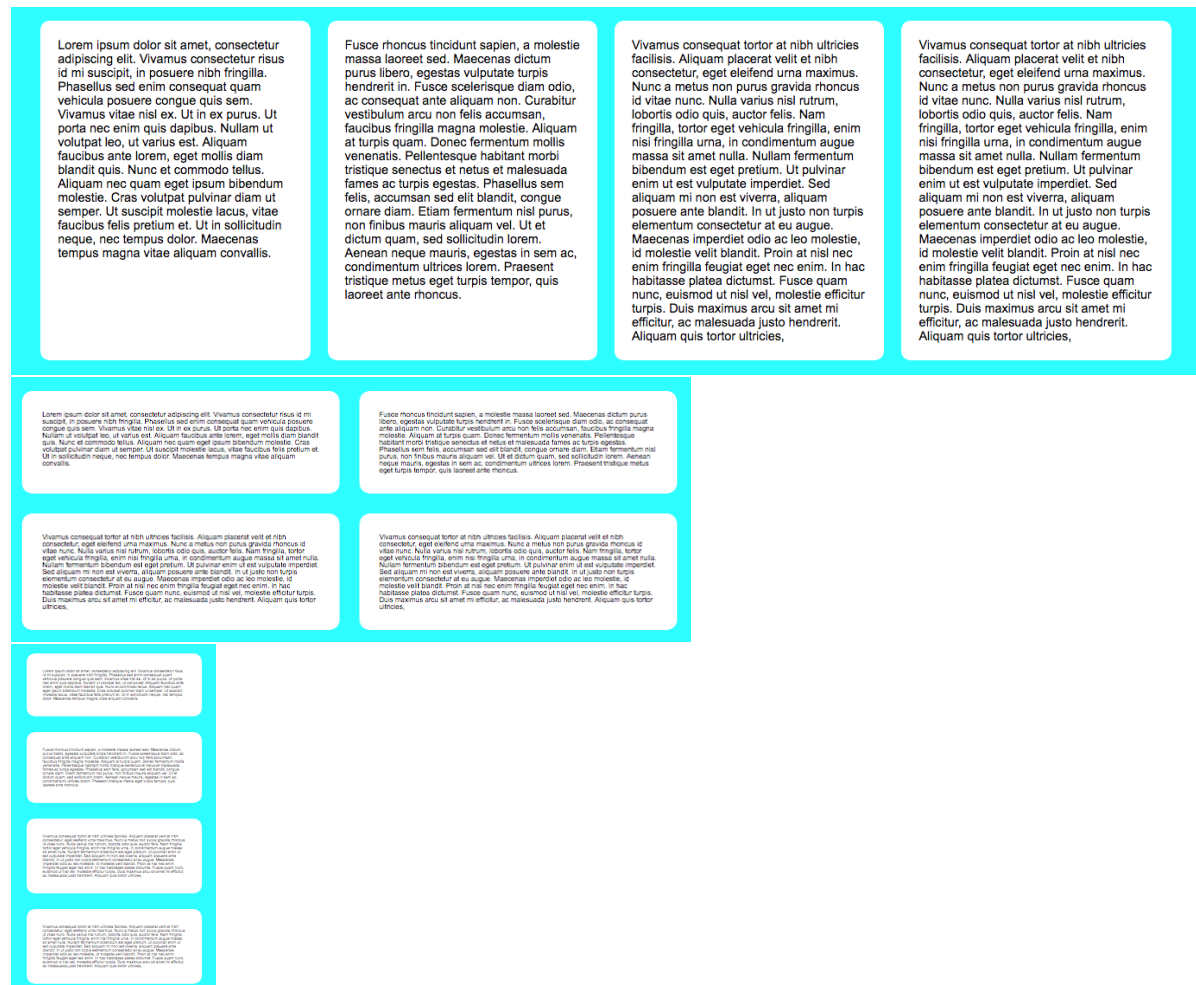
</style>
</head>
<body>
    <section>
        <div class="container">
            <div class="item">
                Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus consectetur risus id mi
                suscipit, in posuere nibh fringilla.
            </div>
            <div class="item">
                Fusce rhoncus tincidunt sapien, a molestie massa laoreet sed. Maecenas dictum purus
                libero, egestas vulputate turpis hendrerit in
            </div>
        </div>
    </section>
</body>

```

```

</div>
<div class="item">
    Vivamus consequat tortor at nibh ultricies facilisis. Aliquam placerat velit et nibh
consectetur, eget eleifend urna maximus.
</div>
<div class="item">
    Vivamus consequat tortor at nibh ultricies facilisis. Aliquam placerat velit et nibh
consectetur, eget eleifend urna maximus.
</div>
</div>
</section>
</body>
</html>

```



Responsive image gallery

```

<!DOCTYPE html>
<html lang="en">

```

```
<head>
  <style>
    *{
      box-sizing: border-box;
    }
    header h1{
      text-align: center;
      font:2.5vw 'san-serif',Arial;
    }
    .container{
      display:flex;
      flex-wrap: wrap;
      align-items:top;
      width:100%;
      background-color: gray;
    }
    .container .item{
      flex:30%;
      margin:10px;
    }
    .container .item img{
      width: 100%;
      margin-top: 10px;
    }
    @media (min-width: 481px) and (max-width: 767px) {
      .container .item{
        flex:0 0 47%;
      }
    }
    @media (min-width: 320px) and (max-width: 480px) {
      .container .item{
        flex:0 0 95%;
        align-items: center;
      }
    }
  </style>
</head>
<body>
  <header>
    <h1>SCDC in media</h1>
  </header>
  <div class="container">
```

```

<div class="item">
  <img src='./images/1.jpg'>
  <img src='./images/2.jpg'>
  <img src='./images/3.jpg'>
  <img src='./images/4.jpg'>
</div>
<div class="item">
  <img src='./images/5.jpg'>
  <img src='./images/6.jpg'>
  <img src='./images/7.jpg'>
  <img src='./images/8.jpg'>
</div>
<div class="item">
  <img src='./images/9.JPG'>
  <img src='./images/10.jpg'>
  <img src='./images/11.jpg'>
  <img src='./images/12.jpg'>
</div>
</div>
</body>
</html>

```



Multiple columns

We can arrange the text in multiple columns easily using column property

column-count	Specifies the number of columns an element should be divided into
column-fill	Specifies how to fill columns
column-gap	Specifies the gap between the columns
column-rule	A shorthand property for setting all the column-rule-* properties
column-rule-color	Specifies the color of the rule between columns
column-rule-style	Specifies the style of the rule between columns

column-rule-width	Specifies the width of the rule between columns
column-span	Specifies how many columns an element should span across
column-width	Specifies a suggested optimal width for the columns
columns	A shorthand property for setting column-width and column-count

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<style>
```

```
.newspaper {
```

```
-webkit-column-count: 3;
```

```
-moz-column-count: 3;
```

```
column-count: 3;
```

```
column-rule: 1px solid green;
```

```
column-gap: 25px;
```

```
}
```

```
.newspaper h2{
```

```
-webkit-column-span: all;
```

```
column-span: all;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<article class="newspaper">
```

```
<h2>Igate computer education</h2>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean dictum congue eros vel varius. In hac habitasse platea dictumst.

```
</article>
```

```
</body>
```

```
</html>
```

Igate computer education

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean dictum congue eros vel varius. In hac habitasse platea dictumst. Praesent ac nunc pellentesque, laoreet nunc non, venenatis nunc. Integer ac pretium purus. Fusce vel magna suscipit, fringilla erat id, facilisis justo. Aenean tortor velit, ultricies eget augue sit amet, sodales fringilla tellus. Nam ligula mi, consectetur vel dolor a, sodales vulputate magna. Cras et ante eu velit consectetur consequat. Donec mattis tincidunt libero sed dapibus. Proin lobortis scelerisque tellus, at volutpat mauris ultricies et. In sed lectus diam. Phasellus id laoreet ligula, vel maximus lacus. Ut ex elit, gravida id nisi vitae, pulvinar accumsan diam. Duis ut nulla id mi

placemat rhoncus ut id magna. Nam ut blandit purus, et viverra ante. Vestibulum nibh libero, scelerisque pretium ante id, ullamcorper blandit libero. Nullam facilisis sit amet enim eget dictum. Integer ultricies ornare augue, in finibus dui tempor id. Quisque efficitur pulvinar efficitur. Duis ac dapibus est, in rhoncus risus. Curabitur sed nunc nulla. Aenean sed iaculis tellus. Nunc dapibus orci ac lacus molestie, sed egestas orci hendrerit. Mauris porttitor porta enim sit amet molestie. Maecenas tempus rhoncus erat, eu elementum quam molestie eget. Etiam dignissim sed nisl at posuere. Vestibulum id nisl eget mi tincidunt porttitor ut venenatis lorem. Nunc fringilla turpis vitae augue pretium

pellentesque. Mauris rhoncus feugiat felis, ut rutrum ex vulputate ut. Nullam placerat luctus magna, vestibulum lobortis lectus eleifend eu. Nunc ac malesuada libero. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla nec porttitor sapien, maximus pharetra ligula. Donec id viverra arcu. Morbi mollis augue eu sapien rutrum, ac interdum velit commodo. Suspendisse pretium ex eget suscipit tristique. Aliquam ut nisi sem. Nullam vel euismod sapien.

