

# V-EX TECH

## Web Development

Java / Node.js / PHP / .Net / Python

Certification Course

**Assured Placement Program  
With International Certificate**

### About V-Ex Tech....

**V-Ex Tech** is an elevated education platform providing rigorous industry-relevant programs

Designed and delivered on collaboration with industry professionals. It has been constantly

Into process of creating an immersive learning experience binding latest technologies, pedagogy

and services with enormous job placement opportunities too.

# Css Advance

## Rounded Corners

```
<!DOCTYPE html>
<html>
<head>
<style>
#rcorners1 {
  border-radius: 25px;
  background: #73AD21;
  padding: 20px;
  width: 200px;
  height: 150px;
}

#rcorners2 {
  border-radius: 25px;
  border: 2px solid #73AD21;
  padding: 20px;
  width: 200px;
  height: 150px;
}
```

```
}

#rcorners3 {
  border-radius: 25px;
  background: url(paper.gif);
  background-position: left top;
  background-repeat: repeat;
  padding: 20px;
  width: 200px;
  height: 150px;
}
</style>
</head>
<body>

<h1>The border-radius Property</h1>

<p>Rounded corners for an element with a specified background
color:</p>
<p id="rcorners1">Rounded corners!</p>
<p>Rounded corners for an element with a border:</p>
<p id="rcorners2">Rounded corners!</p>
<p>Rounded corners for an element with a background image:</p>
<p id="rcorners3">Rounded corners!</p>

</body>
</html>
```

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

**Four values - border-radius: 15px 50px 30px 5px;**

## Border Images

```
<!DOCTYPE html>
<html>
<head>
<style>
#borderimg {
  border: 10px solid transparent;
  padding: 15px;
  border-image: url(border.png) 40 round; // stretch
}
</style>
</head>
<body>

<h1>The border-image Property</h1>

<p id="borderimg">border-image: url(border.png) 30 round;</p>

<p>Here is the original image:</p>
```

```
<p><strong>Note:</strong> Internet Explorer 10, and earlier  
versions, do not support the border-image property.</p>  
  
</body>  
</html>
```

## Multiple Backgrounds

- background-size
- background-origin
- background-clip

```
• <!DOCTYPE html>  
• <html>  
• <head>  
• <style>  
• #example1 {  
•   background-image: url(img_flwr.gif), url(paper.gif);  
•   background-position: right bottom, left top;  
•   background-repeat: no-repeat, repeat;  
•   padding: 15px;  
• }  
• </style>  
• </head>  
• <body>  
•  
• <h1>Multiple Backgrounds</h1>  
• <p>The following div element has two background  
  images:</p>  
•  
• <div id="example1">
```

```
• <h1>Lorem Ipsum Dolor</h1>
• <p>Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed diam nonummy nibh euismod tincidunt ut laoreet
dolore magna aliquam erat volutpat.</p>
• <p>Ut wisi enim ad minim veniam, quis nostrud exerci
tation ullamcorper suscipit lobortis nisl ut aliquip ex ea
commodo consequat.</p>
• </div>
•
• </body>
• </html>
•
```

## Background Size

**background-size: 100px 80px;**

**background-size: contain;**

**background-size: cover;**

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-
scale=1">
```

```
<style>
body {
  margin: 0;
  font-family: Arial, Helvetica, sans-serif;
}

.hero-image {
  background: url(img_man.jpg) no-repeat center;
  background-size: cover;
  height: 500px;
  position: relative;
}

.hero-text {
  text-align: center;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  color: white;
}
</style>
</head>
<body>

<div class="hero-image">
  <div class="hero-text">
    <h1 style="font-size:50px">I am John Doe</h1>
    <h3>And I'm a Photographer</h3>
    <button>Hire me</button>
  </div>
```

```
</div>

<p>Page content..</p>
<p>Note that this technique will also make the image
responsive: Resize the browser window to see the effect.</p>

</body>
</html>
```

## background-origin Property

The property takes three different values:

- border-box - the background image starts from the upper left corner of the border
- padding-box - (default) the background image starts from the upper left corner of the padding edge
- content-box - the background image starts from the upper left corner of the content



```
<!DOCTYPE html>
<html>
<head>
<style>
#example1 {
  border: 10px solid black;
  padding: 35px;
  background: url(img_flwr.gif);
  background-repeat: no-repeat;
}

#example2 {
  border: 10px solid black;
  padding: 35px;
  background: url(img_flwr.gif);
  background-repeat: no-repeat;
  background-origin: border-box;
}

#example3 {
  border: 10px solid black;
  padding: 35px;
  background: url(img_flwr.gif);
  background-repeat: no-repeat;
  background-origin: content-box;
}
</style>
</head>
<body>

<h1>The background-origin Property</h1>
```

```
<p>No background-origin (padding-box is default):</p>
<div id="example1">
  <h2>Lorem Ipsum Dolor</h2>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
  <p>Ut wisi enim ad minim veniam, quis nostrud exerci tation
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo
consequat.</p>
</div>

<p>background-origin: border-box:</p>
<div id="example2">
  <h2>Lorem Ipsum Dolor</h2>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
  <p>Ut wisi enim ad minim veniam, quis nostrud exerci tation
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo
consequat.</p>
</div>

<p>background-origin: content-box:</p>
<div id="example3">
  <h2>Lorem Ipsum Dolor</h2>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
```

```
<p>Ut wisi enim ad minim veniam, quis nostrud exerci tation  
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo  
consequat.</p>  
</div>  
  
</body>  
</html>
```

## background-clip Property

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
#example1 {  
  border: 10px dotted black;  
  padding: 35px;  
  background: yellow;  
}  
  
#example2 {  
  border: 10px dotted black;  
  padding: 35px;  
  background: yellow;  
  background-clip: padding-box;  
}
```

```
#example3 {
  border: 10px dotted black;
  padding: 35px;
  background: yellow;
  background-clip: content-box;
}
</style>
</head>
<body>

<h1>The background-clip Property</h1>

<p>No background-clip (border-box is default):</p>
<div id="example1">
  <h2>Lorem Ipsum Dolor</h2>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
</div>

<p>background-clip: padding-box:</p>
<div id="example2">
  <h2>Lorem Ipsum Dolor</h2>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
</div>

<p>background-clip: content-box:</p>
<div id="example3">
```

```
<h2>Lorem Ipsum Dolor</h2>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna
aliquam erat volutpat.</p>
</div>

</body>
</html>
```

## CSS Colors

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  color: blue;
  border: 10px solid currentcolor;
  padding: 15px;
}
</style>
</head>
<body>
```

```
<h2>The currentcolor Keyword</h2>

<p>The currentcolor keyword refers to the current value of the
color property of an element.</p>

<div>
This div element has a blue text color and a blue border.
</div>

</body>
</html>
```

# Gradients

- **Linear Gradients** (goes down/up/left/right/diagonally)
- **Radial Gradients** (defined by their center)
- **Conic Gradients** (rotated around a center point)

## CSS Linear Gradients

`background-image: linear-gradient(direction, color-stop1, color-stop2, ...);`

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
  height: 200px;
  background-color: red; /* For browsers that do not support
gradients */
  background-image: linear-gradient(red, yellow);
}
</style>
</head>
<body>

<div id="grad1"></div>

</body>
</html>
```

to right

repeating-linear-gradien

# Radial Gradients

## Syntax

background-image: radial-gradient(*shape size at position, start-color, ..., last-color*);

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
  height: 150px;
  width: 200px;
  background-color: red; /* For browsers that do not support
gradients */
  background-image: radial-gradient(red, yellow, green);
}
</style>
</head>
<body>

<div id="grad1"></div>

</body>
</html>
```



# CSS Conic Gradients

## Syntax

background-image: conic-gradient([from *angle*]  
[at *position*,] *color* [*degree*], *color* [*degree*], ...);

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
  height: 200px;
  width: 200px;
  background-color: red; /* For browsers that do not support
gradients */
  background-image: conic-gradient(red, yellow, green);
}
</style>
</head>
<body>

<h1>Conic Gradient - Three Colors</h1>

<div id="grad1"></div>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {
  height: 200px;
  width: 200px;
  background-color: red; /* For browsers that do not support
gradients */
  background-image: conic-gradient(red 0deg, red 90deg, yellow
90deg, yellow 180deg, green 180deg, green 270deg, blue 270deg);
  border-radius: 50%;
}
</style>
</head>
<body>

<h1>Conic Gradient - Pie Chart</h1>

<div id="grad1"></div>

</body>
</html>
```

## Shadow Effects

- `text-shadow`
- `box-shadow`

`text-shadow: 2px 2px color;`

`box-shadow: 10px 10px color;`

`box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);`

## Text Effects

- `text-overflow`
- `word-wrap`
- `word-break`
- `writing-mode`
- 

```
• p.test2 {  
•   white-space: nowrap;  
•   width: 200px;  
•   border: 1px solid #000000;  
•   overflow: hidden;  
•   text-overflow: ellipsis;  
• }
```

```
<!DOCTYPE html>
<html>
<head>
<style>
p.test {
  width: 11em;
  border: 1px solid #000000;
  word-wrap: break-word;
}
</style>
</head>
<body>

<h1>The word-wrap Property</h1>

<p class="test">This paragraph contains a very long word:
thisisaveryveryveryveryveryverylongword. The long word will
break and wrap to the next line.</p>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
p.test1 {
  width: 140px;
  border: 1px solid #000000;
  word-break: keep-all;
}

p.test2 {
  width: 140px;
  border: 1px solid #000000;
  word-break: break-all;
}
</style>
</head>
<body>

<h1>The word-break Property</h1>

<p class="test1">This paragraph contains some text. This line
will-break-at-hyphens.</p>

<p class="test2">This paragraph contains some text. The lines
will break at any character.</p>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
p.test1 {
  writing-mode: horizontal-tb;
}

span.test2 {
  writing-mode: vertical-rl;
}

p.test2 {
  writing-mode: vertical-rl;
}
</style>
</head>
<body>
<h1>The writing-mode Property</h1>

<p class="test1">Some text with default writing-mode.</p>

<p>Some text with a span element with a <span
class="test2">vertical-rl</span> writing-mode.</p>

<p class="test2">Some text with writing-mode: vertical-rl.</p>

</body>
</html>
```

# 2D Transforms

## 2D Transforms Methods

- `translate()`
- `rotate()`
- `scaleX()`
- `scaleY()`
- `scale()`
- `skewX()`
- `skewY()`
- `skew()`
- `matrix()`



## translate() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
  transform: translate(50px,100px);
}
</style>
</head>
<body>

<h1>The translate() Method</h1>
<p>The translate() method moves an element from its current position:</p>

<div>
This div element is moved 50 pixels to the right, and 100 pixels down from its
current position.
</div>

</body>
</html>
```



## rotate() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
}

div#myDiv {
  transform: rotate(20deg);
}
</style>
</head>
<body>

<h1>The rotate() Method</h1>

<p>The rotate() method rotates an element clockwise or counter-clockwise.</p>

<div>
This a normal div element.
</div>

<div id="myDiv">
This div element is rotated clockwise 20 degrees.
</div>

</body>
</html>
```

## The scale() Method

The `scale()` method increases or decreases the size of an element (according to the parameters given for the width and height).

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  margin: 150px;
  width: 200px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
  transform: scale(1,3);
}
</style>
</head>
<body>

<div>
This div element is two times of its original width, and three times of its
original height.
</div>

</body>
</html>
```

## The scaleX() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  margin: 150px;
  width: 200px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
  transform: scaleX(4);
}
</style>
</head>
<body>

<div>
This div element is two times of its original width.
</div>

</body>
</html>
```

## The scaleY() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  margin: 150px;
  width: 200px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
  transform: scaleY(3);
}
</style>
</head>
<body>

<div>
This div element is three times of its original height.
</div>

</body>
</html>
```

## The skewX() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
}

div#myDiv {
  transform: skewX(20deg);
}
</style>
</head>
<body>

<div>
This a normal div element.
</div>

<div id="myDiv">
This div element is skewed 20 degrees along the X-axis.
</div>

</body>
</html>
```

## The skewY() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
}

div#myDiv {
  transform: skewY(20deg);
}
</style>
</head>
<body>

<div>
This a normal div element.
</div>

<div id="myDiv">
This div element is skewed 20 degrees along the Y-axis.
</div>

</body>
</html>
```

**transform:** skew(20deg, 10deg);

## The matrix() Method

The `matrix()` method combines all the 2D transform methods into one.

**The parameters are as follow: `matrix(scaleX(), skewY(), skewX(), scaleY(), translateX(), translateY())`**

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
}

div#myDiv1 {
  transform: matrix(1, -0.3, 0, 1, 0, 0);
}

div#myDiv2 {
  transform: matrix(1, 0, 0.5, 1, 150, 0);
}
</style>
</head>
<body>

<h1>The matrix() Method</h1>

<p>The matrix() method combines all the 2D transform methods into one.</p>

<div>
This a normal div element.
</div>

<div id="myDiv1">
```

```
Using the matrix() method.  
</div>  
  
<div id="myDiv2">  
Another use of the matrix() method.  
</div>  
  
</body>  
</html>
```

# 3D Transforms

## CSS 3D Transforms Methods

With the CSS **transform** property you can use the following 3D transformation methods:

- **rotateX()**
- **rotateY()**
- **rotateZ()**



## The rotateX() Method

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  background-color: yellow;
  border: 1px solid black;
}

#myDiv {
  transform: rotateX(150deg);
}
</style>
</head>
<body>

<div>
This a normal div element.
</div>

<div id="myDiv">
This div element is rotated 150 degrees.
</div>

</body>
</html>
```

# CSS Transitions

- `transition`
- `transition-delay`
- `transition-duration`
- `transition-property`
- `transition-timing-function`

```
• <!DOCTYPE html>
• <html>
• <head>
• <style>
•   div {
•     width: 100px;
•     height: 100px;
•     background: red;
•     transition: width 2s, height 2s;
•   }
•
•   div:hover {
•     width: 300px;
•     height: 300px;
•   }
• </style>
• </head>
• <body>
•   <div>
•
•   </div>
•
• </body>
• </html>
•
```

**ease** - specifies a transition effect with a slow start, then fast, then end slowly (this is default)

- **linear** - specifies a transition effect with the same speed from start to end
- **ease-in** - specifies a transition effect with a slow start
- **ease-out** - specifies a transition effect with a slow end
- **ease-in-out** - specifies a transition effect with a slow start and end
- **cubic-bezier(n,n,n,n)** - lets you define your own values in a cubic-bezier function

```
• <!DOCTYPE html>
• <html>
• <head>
• <style>
• div {
•   width: 100px;
•   height: 100px;
•   background: red;
•   transition: width 2s;
• transition-delay: 1s;
• }
•
• #div1 {transition-timing-function: linear;}
• #div2 {transition-timing-function: ease;}
• #div3 {transition-timing-function: ease-in;}
• #div4 {transition-timing-function: ease-out;}
• #div5 {transition-timing-function: ease-in-out;}
•
• div:hover {
•   width: 300px;
• }
• </style>
• </head>
• <body>
•
• <div id="div1">linear</div><br>
• <div id="div2">ease</div><br>
• <div id="div3">ease-in</div><br>
• <div id="div4">ease-out</div><br>
• <div id="div5">ease-in-out</div><br>
•
```

- `</bod</html>`

## Transition + Transform

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 100px;
  height: 100px;
  background: red;
  transition: width 2s, height 2s, transform 2s;
  /* transition-delay: 1s; */
}

div:hover {
  width: 300px;
  height: 300px;
  transform: rotate(180deg);
}
</style>
</head>
<body>

<div></div>

</body>
</html>
```

```
div {  
  transition-property: width;  
  transition-duration: 2s;  
  transition-timing-function: linear;  
  transition-delay: 1s;  
}
```

# CSS Animations

- @keyframes
- animation-name
- animation-duration
- animation-delay
- animation-iteration-count
- animation-direction
- animation-timing-function
- animation-fill-mode
- animation