CSS 2D Transforms

## **CSS 2D Transforms**

CSS transforms allow you to move, rotate, scale, and skew elements.

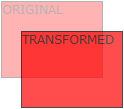
Mouse over the element below to see a 2D transformation:

## **CSS 2D Transforms Methods**

With the CSS transform property you can use the following 2D transformation methods:

* translate()
* rotate()
* scaleX()
* scaleY()
* scale()
* skewX()
* skewY()
* skew()
* matrix()

## **The translate() Method**



The translate() method moves an element from its current position (according to the parameters given for the X-axis and the Y-axis).

The following example moves the <div> element 50 pixels to the right, and 100 pixels down from its current position:

<!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>

## **The rotate() Method**



The rotate() method rotates an element clockwise or counter-clockwise according to a given degree.

The following example rotates the <div> element clockwise with 20 degrees:

<!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>

Using negative values will rotate the element counter-clockwise.

The following example rotates the <div> element counter-clockwise with 20 degrees:

<!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">

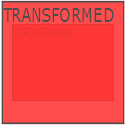
This div element is rotated counter-clockwise with 20 degrees.

</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).

The following example increases the <div> element to be two times of its original width, and three times of its original height:

<!DOCTYPE html>

<html>

<head>

<style>

div {

margin: 150px;

width: 200px;

height: 100px;

background-color: yellow;

border: 1px solid black;

transform: scale(2,3);

}

</style>

</head>

<body>

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

<p>The scale() method increases or decreases the size of an element.</p>

<div>

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

</div>

</body>

</html>

The following example decreases the <div> element to be half of its original width and height:

<!DOCTYPE html>

<html>

<head>

<style>

div {

margin: 150px;

width: 200px;

height: 100px;

background-color: yellow;

border: 1px solid black;

transform: scale(0.5,0.5);

}

</style>

</head>

<body>

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

<p>The scale() method increases or decreases the size of an element.</p>

<div>

This div element is decreased to be half of its original width and height.

</div>

</body>

</html>

## **The scaleX() Method**

The scaleX() method increases or decreases the width of an element.

The following example increases the <div> element to be two times of its original width:

<!DOCTYPE html>

<html>

<head>

<style>

div {

margin: 150px;

width: 200px;

height: 100px;

background-color: yellow;

border: 1px solid black;

transform: scaleX(2);

}

</style>

</head>

<body>

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

<p>The scaleX() method increases or decreases the width of an element.</p>

<div>

This div element is two times of its original width.

</div>

</body>

</html>

## **The skewX() Method**

The skewX() method skews an element along the X-axis by the given angle.

The following example skews the <div> element 20 degrees along the X-axis:

<!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>

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

<p>The skewX() method skews an element along the X-axis by the given angle.</p>

<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**

The skewY() method skews an element along the Y-axis by the given angle.

The following example skews the <div> element 20 degrees along the Y-axis:

<!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>

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

<p>The skewY() method skews an element along the Y-axis by the given angle.</p>

<div>

This a normal div element.

</div>

<div id="myDiv">

This div element is skewed 20 degrees along the Y-axis.

</div>

</body>

</html>

## **The skew() Method**

The skew() method skews an element along the X and Y-axis by the given angles.

The following example skews the <div> element 20 degrees along the X-axis, and 10 degrees along the Y-axis:

<!DOCTYPE html>

<html>

<head>

<style>

div {

width: 300px;

height: 100px;

background-color: yellow;

border: 1px solid black;

}

div#myDiv {

transform: skew(20deg,10deg);

}

</style>

</head>

<body>

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

<p>The skew() method skews an element into a given angle.</p>

<div>

This a normal div element.

</div>

<div id="myDiv">

This div element is skewed 20 degrees along the X-axis, and 10 degrees along the Y-axis.

</div>

</body>

</html>

## **The matrix() Method**



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

The matrix() method take six parameters, containing mathematic functions, which allows you to rotate, scale, move (translate), and skew elements.

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>