CSS Transforms

CSS3 supports transform property. This transform property facilitates you to translate, rotate, scale, and skews elements.

Transformation is an effect that is used to change shape, size and position.

There are two type of transformation i.e. 2D and 3D transformation supported in CSS3.

CSS 2D Transforms

The CSS 2D transforms are used to re-change the structure of the element as translate, rotate, scale and skew etc.

Following is a list of 2D transforms methods:

**translate(x,y):** It is used to transform the element along X-axis and Y-axis.

**translateX(n):** It is used to transform the element along X-axis.

**translateY(n):** It is used to transform the element along Y-axis.

**rotate():** It is used to rotate the element on the basis of an angle.

**scale(x,y):** It is used to change the width and height of an element.

**scaleX(n):** It is used to change the width of an element.

**scaleY(n):** It is used to change the height of an element.

**skewX():** It specifies the skew transforms along with X-axis.

**skewY():**It specifies the skew transforms along with Y-axis.

**matrix():** It specifies matrix transforms.

Supporting browsers

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Property** | chrome browser**Chrome** | ie browser**IE** | firefox browser**Firefox** | opera browser**Opera** | safari browser**Safari** |
| transform | 36.04.0 -webkit- | 10.09.0 -ms- | 16.03.5 -moz- | 23.0  15.0 -webkit-  12.1  10.5 -o- | 9.03.2 -webkit- |
| transform-origin  (two-value syntax) | 36.0  4.0 -webkit- | 10.0  9.0 -ms- | 16.03.5 -moz- | 9.03.2 -webkit- | 23.0  15.0 -webkit-  12.110.5 -o- |

The translate() method

The CSS translate() method is used to move an element from its current position according to the given parameters i.e. X-axis and Y-axis.

Let's take an example to translate a

element 50 pixels right, and 100 pixels down from its current position.

**See this example:**

<!DOCTYPE html**>**

**<html>**

**<head><meta** http-equiv="Content-Type" content="text/html; charset=windows-1252"**>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightgreen;

    border: 1px solid black;

    -ms-transform: translate(100px,80px); /\* IE 9 \*/

    -webkit-transform: translate(100px,80px); /\* Safari \*/

    transform: translate(100px,80px); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This div element is translated 50 pixels right, and 100 pixels down from its current position by using translate () method.

**</div>**

**</body>**

**</html>**

The rotate() method

The CSS rotate() method is used to rotate an element clockwise or anti-clockwise according to the given degree.

Let's take an example to rotate a

element by 300.

See this example:

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

}

div#myDiv {

    -ms-transform: rotate(30deg); /\* IE 9 \*/

    -webkit-transform: rotate(30deg); /\* Safari \*/

    transform: rotate(30deg); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This a normal div element.

**</div>**

**<div** id="myDiv"**>**

This is the 30 degree clockwise rotated div element by using rotate() method.

**</div>**

**</body>**

**</html>**

The scale() method

The CSS scale() method is used to increase or decrease the size of an element according to the given width and height.

Let's take an example to increase the size of an

element by increasing its width and height two times.

strong>See this example:

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    margin: 150px;

    width: 200px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

    border: 1px solid black;

    -ms-transform: scale(2.5,2); /\* IE 9 \*/

    -webkit-transform: scale(2.5,2); /\* Safari \*/

    transform: scale(2.5,2); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This div element is scaled 2.5 times of its original width, and 2 times of its original height.

**</div>**

**</body>**

**</html>**

The skewX() method

The CSS skewX() method is used to skew an element along the X axis according to the given angle. Let?s take an example to skew an

element 30 degrees along the X-axis.

**See this example:**

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

}

div#myDiv {

    -ms-transform: skewX(30deg); /\* IE 9 \*/

    -webkit-transform: skewX(30deg); /\* Safari \*/

    transform: skewX(30deg); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This a normal div element.

**</div>**

**<div** id="myDiv"**>**

This div element is skewed 30 degrees along the X-axis.

**</div>**

**</body>**

**</html>**

The skewY() method

The CSS skewY() method is used to skew an element along the Y axis according to the given angle. Let's take an example to skew an

element 30 degrees along the Y-axis.

**See this example:**

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

}

div#myDiv {

    -ms-transform: skewY(30deg); /\* IE 9 \*/

    -webkit-transform: skewY(30deg); /\* Safari \*/

    transform: skewY(30deg); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This a normal div element.

**</div>**

**<div** id="myDiv"**>**

This div element is skewed 30 degrees along the Y-axis.

**</div>**

**</body>**

**</html>**

The skew() method

The CSS skew() method is used to skew an element along with X-axis and Y-axis according to the given angle.

Let's take a <div> element and skew it 30 degree along the X-axis and 20 degree along the Y-axis.

**See this example:**

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

}

div#myDiv {

    -ms-transform: skew(30deg,20deg); /\* IE 9 \*/

    -webkit-transform: skew(30deg,20deg); /\* Safari \*/

    transform: skew(30deg,20deg); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

**<div>**

This a normal div element.

**</div>**

**<div** id="myDiv"**>**

This div element is skewed 30 degrees along the X-axis, and 20 degrees along the Y-axis.

**</div>**

**</body>**

**</html>**

The matrix() method

The CSS matrix() method combines all the six 2D transform methods altogether. It allows you to rotate, scale, translate, and skew elements.

**Syntax:**

The parameters of matrix method: matrix(scaleX(),skewY(),skewX(),scaleY(),translateX(),translateY())

**See this example:**

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

div {

    width: 300px;

    height: 100px;

    background-color: lightpink;

    border: 1px solid black;

}

div#myDiv1 {

    -ms-transform: matrix(1, -0.3, 0, 1, 0, 0); /\* IE 9 \*/

    -webkit-transform: matrix(1, -0.3, 0, 1, 0, 0); /\* Safari \*/

    transform: matrix(1, -0.3, 0, 1, 0, 0); /\* Standard syntax \*/

}

div#myDiv2 {

    -ms-transform: matrix(1, 0, 0.5, 1, 150, 0); /\* IE 9 \*/

    -webkit-transform: matrix(1, 0, 0.5, 1, 150, 0); /\* Safari \*/

    transform: matrix(1, 0, 0.5, 1, 150, 0); /\* Standard syntax \*/

}

**</style>**

**</head>**

**<body>**

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

**<div>**

This  is 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>**

CSS 3D Transforms

The CSS 3D transforms facilitates you to move an element to X-axis, Y-axis and Z-axis. Following is a list of 3D transforms methods:

|  |  |
| --- | --- |
| **Function** | **Description** |
| matrix3D(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n) | It specifies a 3D transformation, using a 4x4 matrix of 16 values. |
| translate3D(x,y,z) | It specifies a 3D translation. |
| translateX(x) | It specifies 3D translation, using only the value for the X-axis. |
| translateY(y) | It specifies 3D translation, using only the value for the Y-axis. |
| translateZ(z) | It specifies 3D translation, using only the value for the Z-axis. |
| scale3D(x,y,z) | It specifies 3D scale transformation |
| scaleX(x) | It specifies 3D scale transformation by giving a value for the X-axis. |
| scaley(y) | It specifies 3D scale transformation by giving a value for the Y-axis. |
| scaleZ(z) | It specifies 3D scale transformation by giving a value for the Z-axis. |
| rotate3D(X,Y,Z,angle) | It specifies 3D rotation along with X-axis, Y-axis and Z-axis. |
| rotateX(angle) | It specifies 3D rotation along with X-axis. |
| rotateY(angle) | It specifies 3D rotation along with Y-axis. |
| rotateZ(angle) | It specifies 3D rotation along with Z-axis. |
| perspective(n) | It specifies a perspective view for a 3D transformed element. |