

CSS Advanced

CSS Rounded Corners

With the CSS border-radius property, you can give any element "rounded corners".

CSS border-radius Property

The CSS border-radius property defines the radius of an element's corners.

Here are three examples:

1. Rounded corners for an element with a specified background color:

Rounded corners!

2. Rounded corners for an element with a border:

Rounded corners!

3. Rounded corners for an element with a background image:

Rounded corners!

Here is the code:

Example

```
<!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>
```

NOTE: 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.

CSS border-radius - Specify Each Corner

The border-radius property can have from one to four values. Here are the rules:

Four values - border-radius: 15px 50px 30px 5px; (first value applies to top-left corner, second value applies to top-right corner, third value applies to bottom-right corner, and fourth value applies to bottom-left corner):

Three values - border-radius: 15px 50px 30px; (first value applies to top-left corner, second value applies to top-right and bottom-left corners, and third value applies to bottom-right corner):

Two values - border-radius: 15px 50px; (first value applies to top-left and bottom-right corners, and the second value applies to top-right and bottom-left corners):

One value - border-radius: 15px; (the value applies to all four corners, which are rounded equally):

Here is the code:

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
#rcorners1 {
```

```
  border-radius: 15px 50px 30px 5px;
```

```
  background: #73AD21;
```

```
  padding: 20px;
```

```
width: 200px;  
height: 150px;  
}
```

```
#rcorners2 {  
border-radius: 15px 50px 30px;  
background: #73AD21;  
padding: 20px;  
width: 200px;  
height: 150px;  
}
```

```
#rcorners3 {  
border-radius: 15px 50px;  
background: #73AD21;  
padding: 20px;  
width: 200px;  
height: 150px;  
}
```

```
#rcorners4 {  
border-radius: 15px;  
background: #73AD21;  
padding: 20px;  
width: 200px;  
height: 150px;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

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

```
<p>Four values - border-radius: 15px 50px 30px 5px:</p>
```

```
<p id="rcorners1"></p>
```

```
<p>Three values - border-radius: 15px 50px 30px:</p>
```

```
<p id="rcorners2"></p>
```

```
<p>Two values - border-radius: 15px 50px:</p>
```

```
<p id="rcorners3"></p>
```

```
<p>One value - border-radius: 15px:</p>
```

```
<p id="rcorners4"></p>
```

```
</body>
```

```
</html>
```

CSS border-image Property

The CSS border-image property allows you to specify an image to be used instead of the normal border around an element.

The property has three parts:

1. The image to use as the border
2. Where to slice the image
3. Define whether the middle sections should be repeated or stretched

We will use the following image (called "border.png"):



The border-image property takes the image and slices it into nine sections, like a tic-tac-toe board. It then places the corners at the corners, and the middle sections are repeated or stretched as you specify.

Note: For border-image to work, the element also needs the border property set!

Here, the middle sections of the image are repeated to create the border:

An image as a border!

Here is the code:

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
#borderimg {
  border: 10px solid transparent;
  padding: 15px;
  border-image: url(border.png) 30 round;
}
</style>
</head>
<body>

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

<p>Here, the middle sections of the image are repeated to create the border:</p>
<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>
```

CSS Multiple Backgrounds

In this section you will learn how to add multiple background images to one element.

You will also learn about the following properties:

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

CSS Multiple Backgrounds

CSS allows you to add multiple background images for an element, through the background-image property.

The different background images are separated by commas, and the images are stacked on top of each other, where the first image is closest to the viewer.

The following example has two background images, the first image is a flower (aligned to the bottom and right) and the second image is a paper background (aligned to the top-left corner):

Example

```
<!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>
```

CSS Gradients

Gradient Background

CSS gradients let you display smooth transitions between two or more specified colors.

CSS defines two types of gradients:

- **Linear Gradients** (goes down/up/left/right/diagonally)
- **Radial Gradients** (defined by their center)

CSS Linear Gradients

To create a linear gradient you must define at least two color stops. Color stops are the colors you want to render smooth transitions among. You can also set a starting point and a direction (or an angle) along with the gradient effect.

Syntax

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

Linear Gradient - Top to Bottom (this is default)

The following example shows a linear gradient that starts at the top. It starts red, transitioning to yellow:

Example

```
#grad {  
  background-image: linear-gradient(red, yellow);  
}
```

Linear Gradient - Left to Right

The following example shows a linear gradient that starts from the left. It starts red, transitioning to yellow:

Example

```
#grad {  
  background-image: linear-gradient(to right, red , yellow);  
}
```

Linear Gradient - Diagonal

You can make a gradient diagonally by specifying both the horizontal and vertical starting positions.

The following example shows a linear gradient that starts at top left (and goes to bottom right). It starts red, transitioning to yellow:

Example

```
#grad {  
  background-image: linear-gradient(to bottom right, red, yellow);  
}
```

Using Angles

If you want more control over the direction of the gradient, you can define an angle, instead of the predefined directions (to bottom, to top, to right, to left, to bottom right, etc.).

Syntax

background-image: linear-gradient(*angle*, *color-stop1*, *color-stop2*);

The angle is specified as an angle between a horizontal line and the gradient line.

CSS Radial Gradients

A radial gradient is defined by its center. To create a radial gradient you must also define at least two color stops.

Syntax

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

By default, shape is ellipse, size is farthest-corner, and position is center.

Radial Gradient - Evenly Spaced Color Stops (this is default)

The following example shows a radial gradient with evenly spaced color stops:

Example

```
<!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); /* Standard syntax (must be last) */
}
</style>
</head>
<body>

<h1>Radial Gradient - Evenly Spaced Color Stops</h1>

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

<p>Note: Internet Explorer 9 and earlier versions do not support gradients.</p>

</body>

</html>

CSS Shadow Effects

With CSS you can add shadow to text and to elements.

- text-shadow
- box-shadow

CSS Text Shadow

The CSS text-shadow property applies shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

Text shadow effect!

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
h1 {
```

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

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Text-shadow effect!</h1>
```

```
</body>
```

```
</html>
```

CSS box-shadow Property

The CSS box-shadow property applies shadow to elements.

In its simplest use, you only specify the horizontal shadow and the vertical shadow:

```
This is a yellow <div> element with a black box-shadow
```

Example

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 300px;
  height: 100px;
  padding: 15px;
  background-color: yellow;
  box-shadow: 10px 10px;
}
</style>
</head>
<body>

<h1>The box-shadow Property</h1>

<div>This is a div element with a box-shadow</div>

</body>
</html>
```

Image slider using HTML and CSS code

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}
.slider{
width:671px;
margin-left:25%;
overflow:hidden;
}
.slide{
position: relative;
width:1435px;
/* animation-name:BSIT;*/
animation:10s BSIT infinite;
/* animation-iteration-count:infinite;*/
}
.slide img{
float:left;
}
@keyframes BSIT{
0% {left:0px;}40% {left:0px;}
50% {left:-671px;}
/*90% {left:-671px;}
100% {left:-1435px;}*/}
</style></head>
```

```
<body>
<div class="slider">
<div class="slide">


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