CSS Animation

**CSS Animation property** is used *to create animation on the webpage*. It can be used as a replacement of animation created by Flash and JavaScript.

CSS3 @keyframes Rule

The animation is created in the @keyframe rule. It is used to control the intermediate steps in a CSS animation sequence.

What animation does

An animation *makes an element change gradually* from one style to another. You can add as many as properties you want to add. You can also specify the changes in percentage.0% specify the start of the animation and 100% specify its completion.

How CSS animation works

When the animation is created in the @keyframe rule, it must be bound with selector; otherwise the animation will have no effect.

The animation could be bound to the selector by specifying at least these two properties:

* The name of the animation
* The duration of the animation

CSS animation properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| @keyframes | It is used to specify the animation. |
| animation | This is a shorthand property, used for setting all the properties, except the animation-play-state and the animation-fill- mode property. |
| animation-delay | It specifies when the animation will start. |
| animation-direction | It specifies if or not the animation should play in reserve on alternate cycle. |
| animation-duration | It specifies the time duration taken by the animation to complete one cycle. |
| animation-fill-mode | it specifies the static style of the element. (when the animation is not playing) |
| animation-iteration-count | It specifies the number of times the animation should be played. |
| animation-play-state | It specifies if the animation is running or paused. |
| animation-name | It specifies the name of @keyframes animation. |
| animation-timing-function | It specifies the speed curve of the animation. |

CSS animation example: changing background color

Let's see a simple CSS animation example that changes background color of rectangle from RED to BLACK.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. width: 100px;
7. height: 100px;
8. background: red;
9. -webkit-animation: myfirst 6s; /\* Chrome, Safari, Opera \*/
10. animation: myfirst 5s;
11. }
12. /\* Chrome, Safari, Opera \*/
13. @-webkit-keyframes myfirst {
14. from {background: red;}
15. to {background: green;}
16. }
17. /\* Standard syntax \*/
18. @keyframes myfirst {
19. from {background: red;}
20. to {background: green;}
21. }
22. **</style>**
23. **</head>**
24. **<body>**
25. **<p><b>**Note:**</b>** The IE 9 and earlier versions don't support CSS3 animation property. **</p>**
26. **<div></div>**
27. **</body>**
28. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssanimation1)

CSS animation example: Moving Rectangle

Let's take another example to display animation with percentage value.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. width: 100px;
7. height: 100px;
8. background: red;
9. position: relative;
10. -webkit-animation: myfirst 5s; /\* Chrome, Safari, Opera \*/
11. animation: myfirst 5s;
12. }
13. /\* Chrome, Safari, Opera \*/
14. @-webkit-keyframes myfirst {
15. 0%   {background:red; left:0px; top:0px;}
16. 25%  {background:yellow; left:300px; top:0px;}
17. 50%  {background:blue; left:200px; top:300px;}
18. 75%  {background:green; left:0px; top:200px;}
19. 100% {background:red; left:0px; top:0px;}
20. }
21. /\* Standard syntax \*/
22. @keyframes myfirst {
23. 0%   {background:red; left:0px; top:0px;}
24. 25%  {background:yellow; left:300px; top:0px;}
25. 50%  {background:blue; left:300px; top:200px;}
26. 75%  {background:green; left:0px; top:200px;}
27. 100% {background:red; left:0px; top:0px;}
28. }
29. **</style>**
30. **</head>**
31. **<body>**
32. **<p><b>**Note:**</b>** The Internet Explorer 9 and its earlier versions don't support this example.**</p>**
33. **<div></div>**
34. **</body>**
35. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssanimation2)

# CSS Gradient

CSS gradient is used to display smooth transition within two or more specified colors.

## Why CSS Gradient

These are the following reasons to use CSS gradient.

* You don't have to use images to display transition effects.
* The download time and bandwidth usage can also be reduced.
* It provides better look to the element when zoomed, because the gradient is generated by the browser.

There are two types of gradient in CSS3.

1. Linear gradients
2. Radial gradients

## 1) CSS Linear Gradient

The CSS3 linear gradient goes up/down/left/right and diagonally. To create a CSS3 linear gradient, you must have to define two or more color stops. The color stops are the colors which are used to create a smooth transition. Starting point and direction can also be added along with the gradient effect.

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

## (i) CSS Linear Gradient: (Top to Bottom)

Top to Bottom Linear Gradient is the default linear gradient. Let's take an example of linear gradient that starts from top. It starts red and transitions to green.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. #grad1 {
6. height: 100px;
7. background: -webkit-linear-gradient(red, green); /\* For Safari 5.1 to 6.0 \*/
8. background: -o-linear-gradient(red, green); /\* For Opera 11.1 to 12.0 \*/
9. background: -moz-linear-gradient(red, green); /\* For Firefox 3.6 to 15 \*/
10. background: linear-gradient(red, green); /\* Standard syntax (must be last) \*/
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h3>**Linear Gradient - Top to Bottom**</h3>**
16. **<p>**This linear gradient starts at the top. It starts red, transitioning to green:**</p>**
17. **<div** id="grad1"**></div>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssgradient1)

Output:

### Linear Gradient - Top to Bottom

This linear gradient starts at the top. It starts red, transitioning to green:

## (ii) CSS Linear Gradient: (Left to Right)

The following example shows the linear gradient that starts from left and goes to right. It starts red from left side and transitioning to green.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. #grad1 {
6. height: 100px;
7. background: -webkit-linear-gradient(left, red, green); /\* For Safari 5.1 to 6.0 \*/
8. background: -o-linear-gradient(right, red, green); /\* For Opera 11.1 to 12.0 \*/
9. background: -moz-linear-gradient(right, red, green); /\* For Firefox 3.6 to 15 \*/
10. background: linear-gradient(to right, red , green); /\* Standard syntax (must be last) \*/
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h3>**Linear Gradient - Left to Right**</h3>**
16. **<p>**This linear gradient starts at the left. It starts red, transitioning to green:**</p>**
17. **<div** id="grad1"**></div>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssgradient2)

Output:

### Linear Gradient - Left to Right

This linear gradient starts at the left. It starts red, transitioning to green:

## (iii) CSS Linear Gradient: (Diagonal)

If you specify both the horizontal and vertical starting positions, you can make a linear gradient diagonal.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. #grad1 {
6. height: 100px;
7. background: -webkit-linear-gradient(left top, red , green); /\* For Safari 5.1 to 6.0 \*/
8. background: -o-linear-gradient(bottom right, red, green); /\* For Opera 11.1 to 12.0 \*/
9. background: -moz-linear-gradient(bottom right, red, green); /\* For Firefox 3.6 to 15 \*/
10. background: linear-gradient(to bottom right, red , green); /\* Standard syntax (must be last) \*/
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h3>**Linear Gradient - Diagonal**</h3>**
16. **<p>**This linear gradient starts at top left. It starts red, transitioning to green:**</p>**
17. **<div** id="grad1"**></div>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssgradient3)

Output:

### Linear Gradient - Diagonal

This linear gradient starts at top left. It starts red, transitioning to green:

## 2) CSS Radial Gradient

You must have to define at least two color stops to create a radial gradient. It is defined by its center.

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

## (i) CSS Radial Gradient: (Evenly Spaced Color Stops)

Evenly spaced color stops is a by default radial gradient. Its by default shape is eclipse, size is farthest- carner, and position is center.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. #grad1 {
6. height: 150px;
7. width: 200px;
8. background: -webkit-radial-gradient(blue, green, red); /\* Safari 5.1 to 6.0 \*/
9. background: -o-radial-gradient(blue, green, red); /\* For Opera 11.6 to 12.0 \*/
10. background: -moz-radial-gradient(blue, green, red); /\* For Firefox 3.6 to 15 \*/
11. background: radial-gradient(blue, green, red); /\* Standard syntax (must be last) \*/
12. }
13. **</style>**
14. **</head>**
15. **<body>**
16. **<h3>**Radial Gradient - Evenly Spaced Color Stops**</h3>**
17. **<div** id="grad1"**></div>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssgradient21)

Output:

### Radial Gradient - Evenly Spaced Color Stops

## (ii) Radial Gradient: (Differently Spaced Color Stops)

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. #grad1 {
6. height: 150px;
7. width: 200px;
8. background: -webkit-radial-gradient(blue 5%, green 15%, red 60%); /\* Safari 5.1 to 6.0 \*/
9. background: -o-radial-gradient(blue 5%, green 15%, red 60%); /\* For Opera 11.6 to 12.0 \*/
10. background: -moz-radial-gradient(blue 5%, green 15%, red 60%); /\* For Firefox 3.6 to 15 \*/
11. background: radial-gradient(blue 5%, green 15%, red 60%); /\* Standard syntax (must be last) \*/
12. }
13. **</style>**
14. **</head>**
15. **<body>**
16. **<h3>**Radial Gradient - Differently Spaced Color Stops**</h3>**
17. **<div** id="grad1"**></div>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssgradient22)

Output:

### Radial Gradient - Differently Spaced Color Stops

## Supporting Browsers

This table specify the first browser version that fully supports the gradient property.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **property** | **chrome** | **firefox** | **ie** | **opera** | **safari** |
| linear-gradient | 26.0  10.0-webkit- | 16.0  3.6-moz- | 10.0 | 12.1  11.1-o- | 6.1  5.1-webkit- |
| radial-gradient | 26.0  10.0-webkit- | 16.0 3.6-moz- | 10.0 | 12.1 11.1-o- | 6.1 5.1-webkit- |
| repeating-linear-gradient | 26.0  10.0-webkit- | 16.0 3.6-moz- | 10.0 | 12.1 11.1-o- | 6.1 5.1-webkit- |
| repeating-radial-gradient | 26.0 10.0-webkit- | 16.0 3.6-moz- | 10.0 | 12.1 11.1-o- | 6.1 5.1-webkit- |

# CSS Transition

The CSS transitions are effects that are added to change the element gradually from one style to another, without using flash or JavaScript.

You should specify two things to create CSS transition.

* The CSS property on which you want to add an effect.
* The time duration of the effect.

Let's take an example which defines transition effect on width property and duration of 3 seconds.

#### Note: If you don't specify the duration part, the transition will have no effect because its default value is 0. The transition effect is started when you move cursor over an element.

#### Note: The transition property is not supported by IE9 and earlier version.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div{
6. width: 100px;
7. height: 100px;
8. background: orange;
9. -webkit-transition: width 1s; /\* For Safari 3.1 to 6.0 \*/
10. transition: width 1s;
11. }
12. div:hover {
13. width: 300px;
14. }
15. **</style>**
16. **</head>**
17. **<body>**
18. **<div></div>**
19. **<p>**Move the cursor over the div element above, to see the transition effect.**</p>**
20. **</body>**
21. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csstransition1)

#### Note: When you take mouse cursor out of the element, it gains its original style gradually.

## CSS Multiple Transition Effect

It is used to add transition effect for more than one CSS property. If you want to add transition effect on more than one property, separate those properties with a comma.

Let's take an example. Here, the transition effects on width, height and transformation.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div {
6. padding:15px;
7. width: 150px;
8. height: 100px;
9. background: violet;
10. -webkit-transition: width 2s, height 2s, -webkit-transform 2s; /\* For Safari 3.1 to 6.0 \*/
11. transition: width 2s, height 2s, transform 2s;
12. }
13. div:hover {
14. width: 300px;
15. height: 200px;
16. -webkit-transform: rotate(360deg); /\* Chrome, Safari, Opera \*/
17. transform: rotate(360deg);
18. }
19. **</style>**
20. **</head>**
21. **<body>**
22. **<div><b>**JavaTpoint.com**</b><p>** (Move your cursor on me to see the effect)**</p></div>**
23. **</body>**
24. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csstransition2)

CSS Tooltips

CSS Tooltips are a great way to display extra information about something when the user moves the mouse cursor over an element.

Basic Tooltip Example

Let's create basic tooltip that appears when the cursor of the mouse moves over an element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: red;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. }
20. .tooltip:hover .tooltiptext {
21. visibility: visible;
22. }
23. **</style>**
24. **<body** style="text-align:center;"**>**
25. **<p>**Move the mouse over the text below:**</p>**
26. **<div** class="tooltip"**>**Hover over me
27. **<span** class="tooltiptext"**>**This is tooltip text**</span>**
28. **</div>**
29. **</body>**
30. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltips1)

By using tooltips, you can display the position of the tooltip information in four ways:

* Top of the element
* Left side of the element
* Right side of the element
* Bottom of the element

Top Tooltip

The top tooltip specifies that if you move your mouse cursor over the element, the tooltip information will be displayed on the top of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: red;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. bottom: 100%;
20. left: 50%;
21. margin-left: -60px;
22. }
23. .tooltip:hover .tooltiptext {
24. visibility: visible;
25. }
26. **</style>**
27. **<body** style="text-align:center;"**>**
28. **<h2>**Top Tooltip Example**</h2>**
29. **<p>**Move your mouse cursor over the below heading**</p>**
30. **<div** class="tooltip"**><strong>** Welcome to JavaTpoint**</strong>**
31. **<span** class="tooltiptext"**>**A solution of all technology.**</span>**
32. **</div>**
33. **</body>**
34. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltips2)

Bottom Tooltip

The bottom tooltip specifies that if you move your mouse cursor over the element, the tooltip information will be displayed on the bottom of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
10. .tooltip .tooltiptext {
11. visibility: hidden;
12. width: 120px;
13. background-color: red;
14. color: #fff;
15. text-align: center;
16. border-radius: 6px;
17. padding: 5px 0;
18. position: absolute;
19. z-index: 1;
20. top: 100%;
21. left: 50%;
22. margin-left: -60px;
23. }
24. .tooltip:hover .tooltiptext {
25. visibility: visible;
26. }
27. **</style>**
28. **<body** style="text-align:center;"**>**
29. **<h2>**Bottom Tooltip Example**</h2>**
30. **<p>**Move your mouse cursor over the below heading**</p>**
31. **<div** class="tooltip"**><strong>** Welcome to JavaTpoint**</strong>**
32. **<span** class="tooltiptext"**>**A solution of all technology.**</span>**
33. **</div>**
34. **</body>**
35. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltips3)

Left Tooltip

The left tooltip specifies that if you move your mouse cursor over the element, the tooltip information will be displayed on the left side of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: red;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. top: -5px;
20. right: 105%;
21. }
22. .tooltip:hover .tooltiptext {
23. visibility: visible;
24. }
25. **</style>**
26. **<body** style="text-align:center;"**>**
27. **<h2>**Left Tooltip Example**</h2>**
28. **<p>**Move your mouse cursor over the below heading**</p>**
29. **<div** class="tooltip"**><strong>** Welcome to JavaTpoint**</strong>**
30. **<span** class="tooltiptext"**>**A solution of all technology.**</span>**
31. **</div>**
32. **</body>**
33. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltips4)

Right Tooltip

The right tooltip specifies that if you move your mouse cursor over the element, the tooltip information will be displayed on the right side of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: red;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. top: -5px;
20. left: 105%;
21. }
22. .tooltip:hover .tooltiptext {
23. visibility: visible;
24. }
25. **</style>**
26. **<body** style="text-align:center;"**>**
27. **<h2>**Right Tooltip Example**</h2>**
28. **<p>**Move your mouse cursor over the below heading**</p>**
29. **<div** class="tooltip"**><strong>** Welcome to JavaTpoint**</strong>**
30. **<span** class="tooltiptext"**>**A solution of all technology.**</span>**
31. **</div>**
32. **</body>**
33. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltips5)

CSS Tooltip Animation/ Fade In Tooltips

The CSS fade in tooltip or tooltip animation is used to fade in the tooltip text when it is about to visible. The CSS3 "transition" property along with "opacity" property is used o attain fade in tooltip or tooltip animation. The animation time from being completely invisible to 100% visible is specified in second.

Let's take an example to demonstrate the CSS tooltip animation. In this example, the fade in time is 5 second.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: red;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. bottom: 100%;
20. left: 50%;
21. margin-left: -60px;
23. /\* Fade in tooltip - takes 1 second to go from 0% to 100% opac: \*/
24. opacity: 0;
25. transition: opacity 5s;
26. }
27. .tooltip:hover .tooltiptext {
28. visibility: visible;
29. opacity: 1;
30. }
31. **</style>**
32. **<body** style="text-align:center;"**>**
33. **<h2>**Fade In Tooltip Example**</h2>**
34. **<p>**Move your mouse cursor over the below heading**</p>**
35. **<div** class="tooltip"**><strong>** Welcome to JavaTpoint**</strong>**
36. **<span** class="tooltiptext"**>**A solution of all technology.**</span>**
37. **</div>**
38. **</body>**
39. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-tooltip-animation)

CSS Arrow

The CSS arrow is used to add an array along with tooltip. It makes the tooltip like a speech bubble. It can also be represented in four ways:

* Top arrow
* Bottom arrow
* Left arrow
* Right arrow

CSS Top Arrow

The top arrow is used to add an arrow like structure on the top of the tooltip when you move your mouse cursor over the element. It will display the tooltip on the bottom of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: black;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. top: 150%;
20. left: 50%;
21. margin-left: -60px;
22. }
23. .tooltip .tooltiptext::after {
24. content: "";
25. position: absolute;
26. bottom: 100%;
27. left: 50%;
28. margin-left: -5px;
29. border-width: 5px;
30. border-style: solid;
31. border-color: transparent transparent black transparent;
32. }
33. .tooltip:hover .tooltiptext {
34. visibility: visible;
35. }
36. **</style>**
37. **<body** style="text-align:center;"**>**
38. **<h2>**Top Arrow Example**</h2>**
39. **<p>**Move your mouse cursor over the below heading**</p>**
40. **<div** class="tooltip"**><strong>**Welcom to JavaTpoint**</strong>**
41. **<span** class="tooltiptext"**>**A solution of all technology**</span>**
42. **</div>**
43. **</body>**
44. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-arrow1)

CSS Bottom Arrow

The bottom arrow is used to add an arrow like structure on the bottom of the tooltip when you move your mouse cursor over the element. It will display the tooltip on the top of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: black;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. bottom: 150%;
20. left: 50%;
21. margin-left: -60px;
22. }
23. .tooltip .tooltiptext::after {
24. content: "";
25. position: absolute;
26. top: 100%;
27. left: 50%;
28. margin-left: -5px;
29. border-width: 5px;
30. border-style: solid;
31. border-color: black transparent transparent transparent;
32. }
33. .tooltip:hover .tooltiptext {
34. visibility: visible;
35. }
36. **</style>**
37. **<body** style="text-align:center;"**>**
38. **<h2>**Bottom Arrow Example**</h2>**
39. **<p>**Move your mouse cursor over the below heading**</p>**
40. **<div** class="tooltip"**><strong>**Welcom to JavaTpoint**</strong>**
41. **<span** class="tooltiptext"**>**A solution of all technology**</span>**
42. **</div>**
43. **</body>**
44. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-arrow2)

CSS Left Arrow

The left arrow is used to add an arrow like structure on the left of the tooltip when you move your mouse cursor over the element. It will display the tooltip on the right of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: black;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. top: -5px;
20. left: 110%;
21. }
22. .tooltip .tooltiptext::after {
23. content: "";
24. position: absolute;
25. top: 50%;
26. right: 100%;
27. margin-top: -5px;
28. border-width: 5px;
29. border-style: solid;
30. border-color: transparent black transparent transparent;
31. }
32. .tooltip:hover .tooltiptext {
33. visibility: visible;
34. }
35. **</style>**
36. **<body** style="text-align:center;"**>**
37. **<h2>**Left Arrow Example**</h2>**
38. **<p>**Move your mouse cursor over the below heading**</p>**
39. **<div** class="tooltip"**><strong>**Welcom to JavaTpoint**</strong>**
40. **<span** class="tooltiptext"**>**A solution of all technology**</span>**
41. **</div>**
42. **</body>**
43. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-arrow3)

CSS Right Arrow

The right arrow is used to add an arrow like structure on the right of the tooltip when you move your mouse cursor over the element. It will display the tooltip on the left of the element.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<style>**
4. .tooltip {
5. position: relative;
6. display: inline-block;
7. border-bottom: 1px dotted black;
8. }
9. .tooltip .tooltiptext {
10. visibility: hidden;
11. width: 120px;
12. background-color: black;
13. color: #fff;
14. text-align: center;
15. border-radius: 6px;
16. padding: 5px 0;
17. position: absolute;
18. z-index: 1;
19. top: -5px;
20. right: 110%;
21. }
22. .tooltip .tooltiptext::after {
23. content: "";
24. position: absolute;
25. top: 50%;
26. left: 100%;
27. margin-top: -5px;
28. border-width: 5px;
29. border-style: solid;
30. border-color: transparent transparent transparent black;
31. }
32. .tooltip:hover .tooltiptext {
33. visibility: visible;
34. }
35. **</style>**
36. **<body** style="text-align:center;"**>**
37. **<h2>**Right Arrow Example**</h2>**
38. **<p>**Move your mouse cursor over the below heading**</p>**
39. **<div** class="tooltip"**><strong>**Welcom to JavaTpoint**</strong>**
40. **<span** class="tooltiptext"**>**A solution of all technology**</span>**
41. **</div>**
42. **</body>**
43. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-arrow4)

Next Topic[CSS3 Flexbox](https://www.javatpoint.com/css-flexbox)

CSS Flexbox

CSS3 Flexible boxes also known as CSS Flexbox, is a new layout mode in CSS3.

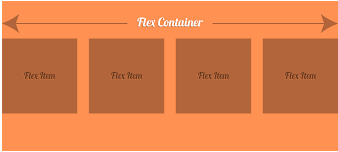
The CSS3 flexbox is used to make the elements behave predictably when they are used with different screen sizes and different display devices. It provides a more efficient way to layout, align and distribute space among items in the container.

It is mainly used to make CSS3 capable to change its item?s width and height to best fit for all available spaces. It is preferred over block model.

The CSS3 flexbox contains flex containers and flex items.

**Flex container:**The flex container specifies the properties of the parent. It is declared by setting the display property of an element to either flex or inline-flex.

**Flex items:**The flex items specify properties of the children. There may be one or more flex items inside a flex container.



**Note:**Flexbox specifies how flex items are set inside a flex container. It sets the flex items inside a flex container along a flex line. By default, there is only one flex line per flex container. Everything outside a flex container and inside a flex item is considered as usual.

Let's take an example to show three flex items within a flex container. By default, they are set along the horizontal flex line, from left to right:

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head><meta** http-equiv="Content-Type" content="text/html; charset=windows-1252"**>**
4. **<style>**
5. .flex-container {
6. display: -webkit-flex;
7. display: flex;
8. width: 400px;
9. height: 200px;
10. background-color: lightpink;
11. }
12. .flex-item {
13. background-color: brown;
14. width: 100px;
15. height: 100px;
16. margin: 10px;
17. }
18. **</style>**
19. **</head>**
20. **<body>**
21. **<div** class="flex-container"**>**
22. **<div** class="flex-item"**>**flex item 1**</div>**
23. **<div** class="flex-item"**>**flex item 2**</div>**
24. **<div** class="flex-item"**>**flex item 3**</div>**
25. **</div>**
26. **</body>**
27. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-flexbox1)

You can also change the direction of the flex line by using direction property. If you want to set the direction line right-to-left then set direction property to rtl.

**See this example:**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. direction: rtl;
7. }
8. .flex-container {
9. display: -webkit-flex;
10. display: flex;
11. width: 400px;
12. height: 200px;
13. background-color: lightpink;
14. }
15. .flex-item {
16. background-color: brown;
17. width: 100px;
18. height: 100px;
19. margin: 10px;
20. }
21. **</style>**
22. **</head>**
23. **<body>**
24. **<div** class="flex-container"**>**
25. **<div** class="flex-item"**>**flex item 1**</div>**
26. **<div** class="flex-item"**>**flex item 2**</div>**
27. **<div** class="flex-item"**>**flex item 3**</div>**
28. **</div>**
29. **</body>**
30. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=css-flexbox2)

CSS3 Flexbox Properties

Following is a list of CSS3 Flexbox properties:

|  |  |
| --- | --- |
| **property** | **description** |
| display | it is used to specify the type of box used for an html element. |
| flex-direction | it is used to specify the direction of the flexible items inside a flex container. |
| justify-content | it is used to align the flex items horizontally when the items do not use all available space on the main-axis. |
| align-items | it is used to align the flex items vertically when the items do not use all available space on the cross-axis. |
| flex-wrap | it specifies whether the flex items should wrap or not, if there is not enough room for them on one flex line. |
| align-content | it is used to modify the behavior of the flex-wrap property. it is similar to align-items, but instead of aligning flex items, it aligns flex lines. |
| flex-flow | it specifies a shorthand property for flex-direction and flex-wrap. |
| order | it specifies the order of a flexible item relative to the rest of the flex items inside the same container. |
| align-self | it is used on flex items. it overrides the container's align-items property. |
| flex | it specifies the length of a flex item, relative to the rest of the flex items inside the same container. |