

Shadows, gradients & animations

Gabriele Speciale - Valentin Sorohan

Gradients

Shadows

Transitions

Transformations

Animations

Gradients

Gradients

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

Linear-gradient: creates a linear gradient that develops along an imaginary line drawn between two or more colors.

Radial-gradient: creates a radial gradient that expands from a central point outward in all directions.

Conic-gradient: creates a conical gradient that unfolds around a central point, like a color wheel.

Gradients - linear-gradient

Creates a linear gradient that develops along an imaginary line drawn between two or more colors.

Syntax: `linear-gradient(directions, color1, color2, ...)`

Example: `background: linear-gradient(to bottom, teal, purple);`

Common directions:

- **to right:** from left to right.
- **to left:** from right to left.
- **to top:** from bottom to up.
- **to bottom:** from top to bottom.
- **corners:** 45deg, 90deg, etc. (45deg is diagonally from the bottom left corner to the top right corner).



Gradients - radial-gradient

Creates a radial gradient that expands from a central point outward in all directions.

Syntax: `radial-gradient(format, color1, color2, ...)`

Example: `background: radial-gradient(circle, cyan, magenta);`

Common formats:

- **circle:** creates a circular gradient.
- **ellipse:** creates an elliptical gradient.
- **positions:** at center, at top, at bottom right, ecc. (at center is default).



Gradients - conic-gradient

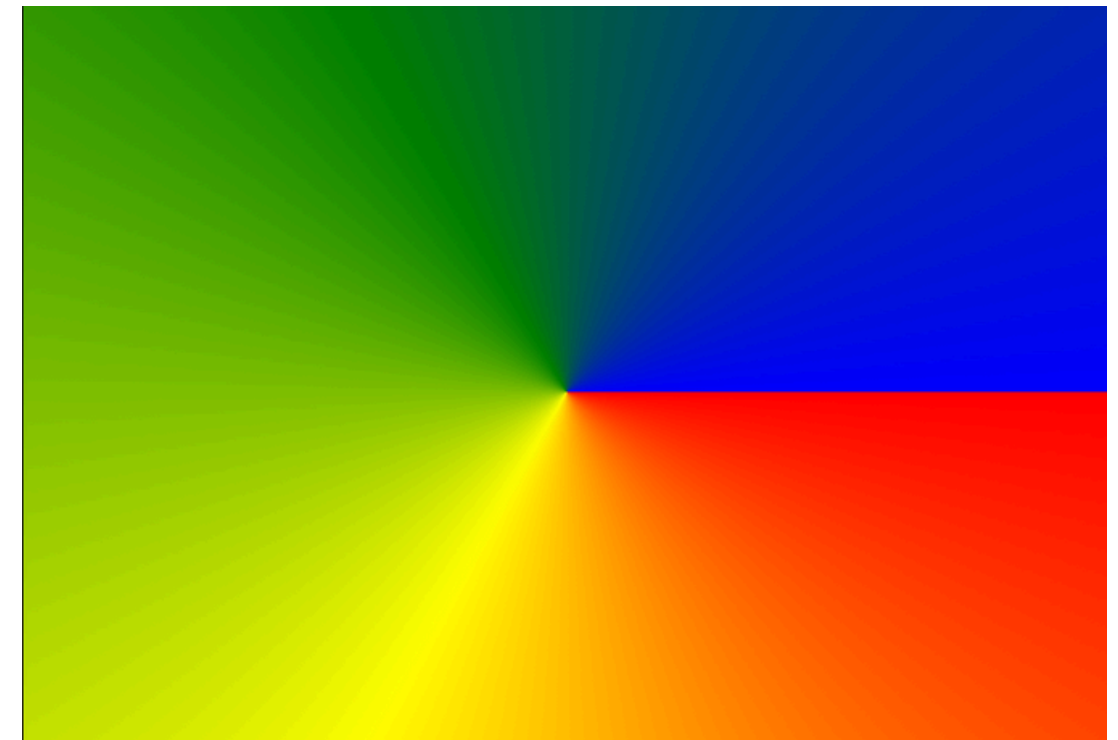
Creates a conical gradient that unfolds around a central point, like a color wheel.

Syntax: `conic-gradient(from corner, color1, color2, ...)`

Example: `background: conic-gradient(from 90deg, red, yellow, green, blue);`

Corner and colors: you can **specify angles** to control where colors change.

For example: `conic-gradient(red 0deg, yellow 90deg, green 180deg, blue 270deg);`



Box-shadows

[MDN | box-shadow](#)

box-shadows

is a property which adds shadow effects around an element's frame, can be set multiple effects separated by commas.

```
div {  
  box-shadow: 50px 20px 5px 8px rgba(255, 0, 0);  
}
```

Diagram illustrating the components of the `box-shadow` property:

- offsets X**: 50px (horizontal offset)
- offsets Y**: 20px (vertical offset)
- blur**: 5px (blur radius)
- spread radius**: 8px (spread radius)
- color**: `rgba(255, 0, 0)` (color of the shadow)



Text-shadows

Text-shadows

property in CSS is used to apply shadow effects to text, enhancing its visual appearance

```
p {  
  text-shadow: 3px 4px 1px red;  
}
```

Diagram illustrating the components of the `text-shadow` property:

- `3px` (blue) is labeled **offsets X** (blue) with an upward arrow.
- `4px` (orange) is labeled **offsets Y** (orange) with a downward arrow.
- `1px` (green) is labeled **blur** (green) with an upward arrow.
- `red` is labeled **color** (black) with a downward arrow.

Hello World!

Shadow - offset X (both)

is the horizontal distance of the shadow(**first length value**).

a positive value offsets the shadow to the right of the element, while a negative value offsets it to the left.

Shadow - offset Y (both)

is the vertical distance of the shadow(**second length value**).

a positive value offsets the shadow below the element, while a negative value offsets it above.

Shadow – blur(both)

is the radius of the shadow's blur(**third length value**).

a larger value results in a more blurred (and thus softer) shadow, a value of 0 produces a sharp shadow.

Shadow – spread radius (only box-shadows)

defines the spread radius(**fourth length value**).

A positive value increases the size of the shadow, a negative value decreases the size of the shadow.

Shadow – color(both)

specifies color for the shadow.

this parameter can be place in any position, if omitted, the default color of the shadow is the current text-color of the element

Transformations

[MDN | transform](#)

Transform

The **transform** CSS property lets you **rotate**, **scale**, **skew**, or **translate** an element. It modifies the coordinate space of the CSS visual formatting model.

Translate: move an item from its original position.

Scale: resizes an element by a scale factor.

Rotate: Rotates an element around its origin point.

Skew: skews an element along the x and/or y axes.

Transform-origin: sets the origin point for element transformations. this point is the center of rotation, scaling and skewing.

Transform - **translate**

The **translate()** CSS function repositions an element in the horizontal and/or vertical directions

Syntax: `translate(x, y)`

Example: `transform: translate(50px, 100px);`

Variants:

- **translateX(x):** move only on the x axis (es. `translateX(50px)`).
- **translateY(y):** move only on the y axis (es. `translateY(100px)`).

Transform - **scale**

The **scale()** CSS function defines a transformation that resizes an element on the 2D plane.

Syntax: `scale(scaleX, scaleY)`

Example: `transform: scale(1.5, 2);`

Variants:

- **scaleX(sx):** scale only on the x axis (es. `scaleX(1.5)`).
- **scaleY(sy):** scale only on the y axis (es. `scaleY(2)`).

Transform - rotate

The `rotate()` CSS function defines a transformation that rotates an element around a fixed point on the 2D plane, without deforming it.

Syntax: `rotate(corner)`

Example: `transform: rotate(45deg);`

Transform - skew

The skew() CSS function defines a transformation that skews an element on the 2D plane.

Syntax: skew(skewX, skewY)

Example: transform: skew(30deg, 20deg);

Variants:

- **skewX(ax):** only tilts along the x-axis (ex. skewX(30deg)).
- **skewY(ay):** only tilts along the y-axis (ex. skewY(20deg)).

Transform - transform-origin

Sets the origin point for element transformations. This point is the center of rotation, scaling and skewing.

Syntax: transform-origin: x y

Example: transform-origin: 50% 50%;

Common values:

- **Percentages:** 50% 50% (centro), 0% 0% (top left corner).
- **Position:** top, bottom, left, right, center.

Transitions

Transitions

Transitions enable you to define the transition between two states of an element

transition-property: Specifies which CSS property(s) will gradually change when a transition occurs

transition-duration: Indicates the duration of the transition

transition-delay: Specifies the delay before the transition begins.

transition-timing-function: Defines the speed curve of the transition during its duration.

transition: A shortcut that allows you to set all the properties mentioned above in a single declaration

Transitions – transition-property

transition-property: Specifies which CSS property(s) will gradually change when a transition occurs

```
div {  
    transition-property: width;  
}
```

Transitions – transition-duration

transition-duration: Indicates the duration of the transition

```
div {  
    transition-duration: 2s;  
}
```

Transitions - transition-delay

transition-delay: Specifies the delay before the transition begins.

```
div {  
  transition-delay: width;  
}
```

Transitions - transition-timing-function

transition-timing-function: Defines the speed curve of the transition during its duration.

```
div {  
  transition-timing-function: ease;  
}
```


Transitions - transition

transition: A shortcut that allows you to set all the properties mentioned above in a single declaration

```
div {  
    transition-timing-function: ease-in-out;  
}
```

```
div {  
    animation: [transition-property] [transition-duration][transition-delay]  
                [transition-timing-function];  
}
```

Animations

Animations

CSS animations enable the transition from one style configuration to another.
They have **two main components**:

@keyframes

allows you to specify one or more key points (keyframes) that describe the state of the element at different moments of the animation.

animation

is a shorthand that allows you to combine several animation-related properties into a single declaration.

Animations

```
/* Define the animation */

@keyframes example {
  0% {
    background-color: red;
  }
  50% {
    background-color: yellow;
  }
  100% {
    background-color: green; }
}
```

@keyframes

animation-name

animation-duration

animation-delay

animation-direction

animation-timing-function

animation-fill-mode

animation

Animations - animation-name

`animation-name:` allows us to define which keyframe to give to our element

```
div {  
  animation-name: example;  
}
```

Animations - animation-duration

`animation-duration:` allows us to specify how long the animation lasts

```
div {  
  animation-duration: 4s;  
}
```

Animations - animation-timing-function

animation-timing-function: allows us to set the type of animation

```
div {  
  animation-timing-function: linear;  
}
```

Animations - animation-delay

animation-delay: how much time must pass before the animation begins

```
div {  
  animation-delay: 2s;  
}
```

Animations - animation-iteration-count

animation-iteration-count: how many times the animation must occur

```
div {  
  animation-iteration-count: 3;  
}
```

Animations - animation-direction

animation-direction: it allows us to change the direction of the animation (start-end, end-start)

```
div {  
  animation-iteration-count: 5;  
}
```


Animations – animation-fill-mode

animation-fill-mode: sets how a CSS animation applies styles to its target before and after its execution.

```
div {  
    animation-fill-mode: forwards;  
}
```

Animations – animation-play-state

animation-play-state: allows us to stop the animation when it changes state, example hover

```
div {  
    animation-play-state: paused; /*running to resume*/  
}
```

Animations - animation

animation: is the shorthand, to set all the properties in one:

```
div {  
  animation: [animation-name] [animation-duration] [animation-timing-function]  
               [animation-delay] [animation-iteration-count] [animation-direction]  
               [animation-fill-mode] [animation-play-state];  
}
```

Generator

Generator

Gradients Generator

Css Gradient
Color Space
Josh Comeau

Shadows generator

Box-shadow
CSS matic
Neumorphism.io

Animations generator

Animista
Web Code Tools
CSS Portal