CSS

gradients, shadows, transitions, transformations and animations

Cherio Stefano and Galizia Nicolò

CSS gradients

CSS gradients consist in progressive transitions between two or more colors

MDN: gradients

CSS gradients: linear gradient

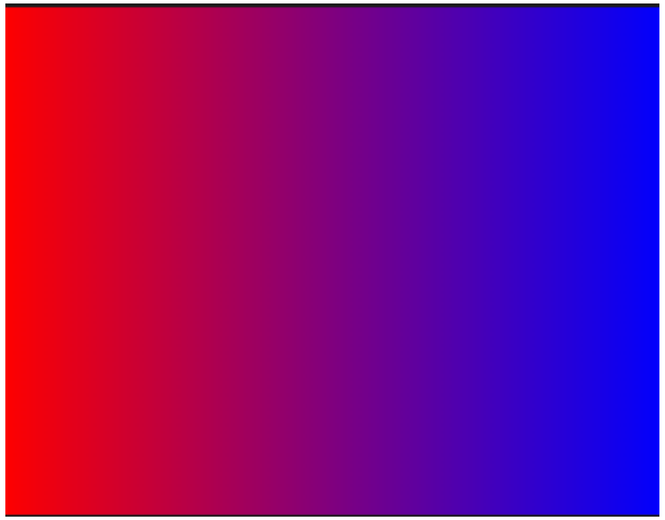
Linear gradients transition colors progressively along an imaginary line

Syntax:

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

Example:

```
.box {
        background: linear-gradient(to right, red, blue);
}
```



MDN: linear gradient

CSS gradients: radial gradients

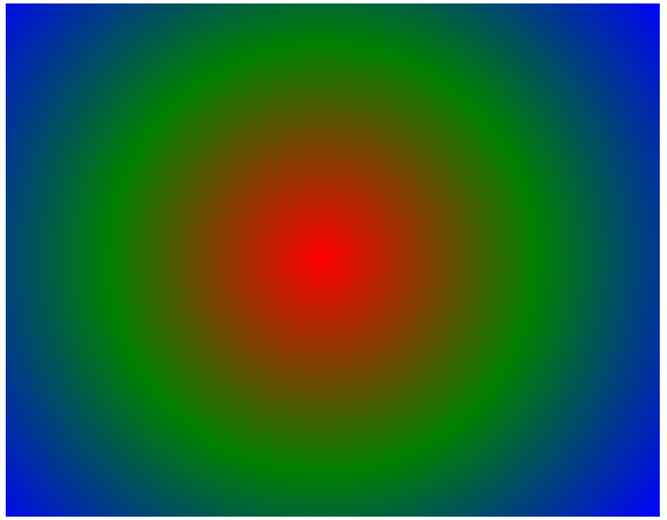
Radial gradients transition colors progressively from a center point (origin).

Syntax:

background: radial-gradient(shape, start-color, ..., last-color);

Example:

```
.box {
        background: radial-gradient(circle, red, green, blue);
}
```



MDN: radial gradient

CSS gradients: conic gradients

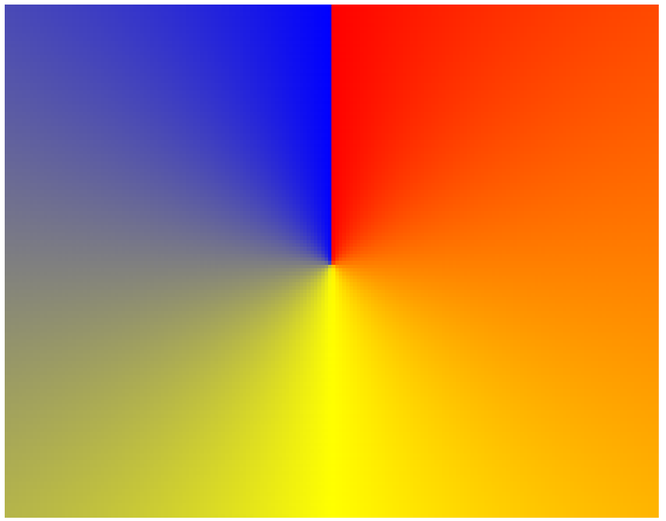
Conic gradients creates an image consisting of a gradient with color transitions rotated around a center point (rather than radiating from the center).

Syntax:

background: conic-gradient([from angle] [at position,] color1, color2, ...);

Example:

```
.box {
        background: conic-gradient(red, yellow, blue);
}
```



MDN: conic gradient

CSS shadows

Shadows add depth and dimension to the elements

CSS shadow: box shadow

The box-shadow adds shadow effects around an element's frame.

Syntax:

box-shadow: [offset-x] [offset-y] [blur-radius] [spread-radius] [color] [inset];

Parameters:

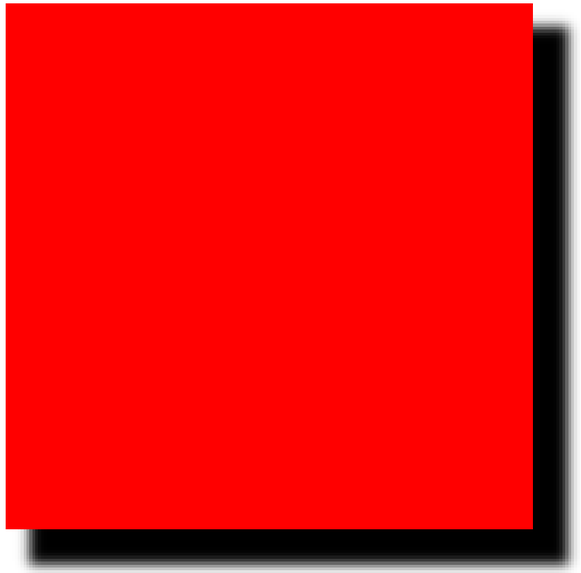
- **offset-x**: horizontal distance of the shadow (positive values move the shadow to the right, negative values to the left).
- **offset-y**: vertical distance of the shadow (positive values move the shadow down, negative values move up).
- **blur-radius**: blur of the shadow (optional: default is 0).
- **spread-radius**: shadow extension (optional: positive values increase the shadow dimension, negative values decrease it).
- **color**: color of the shadow (optional: current text color is the default).
- **inset**: the shadow is placed inside the box (optional)

MDN: box-shadow

CSS shadow: box shadow

Example:

```
.box {
        background-color: red;
        box-shadow: 10px 10px 4px 2px black;
}
```



MDN: box-shadow

CSS shadow: text shadow

The text-shadow CSS property adds shadows to text.

Syntax:

box-shadow: [offset-x] [offset-y] [blur-radius] [color];

Parametri:

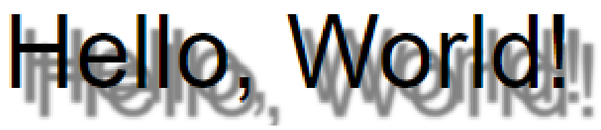
- **offset-x**: horizontal distance of the shadow (positive values move the shadow to the right, negative values to the left).
- **offset-y**: vertical distance of the shadow (positive values move the shadow down, negative values move up).
- **blur-radius**: blur of the shadow (optional: default is 0).
- **color**: color of the shadow (optional: current text color is the default).

MDN: text-shadow

CSS shadow: text shadow

Example:

```
h1 {
        text-shadow: 7px 7px 2px grey, 14px 14px 2px grey;
}
```



MDN: text-shadow

CSS transitions

CSS transitions provide a way to control animation speed when changing CSS properties.

MDN: CSS transitions

CSS transitions: transition-property

Specifies the name or names of the CSS properties to which transitions should be applied.

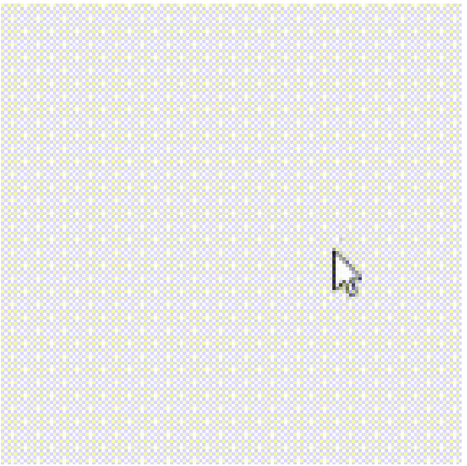
Syntax:

transition-property: none/all/property;

Example:

```
#box {
        transition-property: background-color;
        background-color: red;
}

#box:hover,
#box:focus {
                background-color: #eee;
}
```



MDN: transition-property

CSS transitions: transition-duration

Specifies the duration over which transitions should occur.

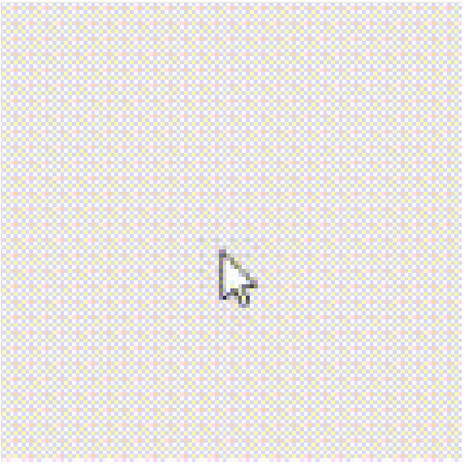
Syntax:

transition-duration: time;

Example:

```
#box {
        transition-property: background-color;
        transition-duration: 2s;
        background-color: red;
}

#box:hover,
#box:focus {
                background-color: #eee;
}
```



MDN: transition-duration

CSS transitions: transition-timing-function

Specifies a function to define how intermediate values for properties are computed.

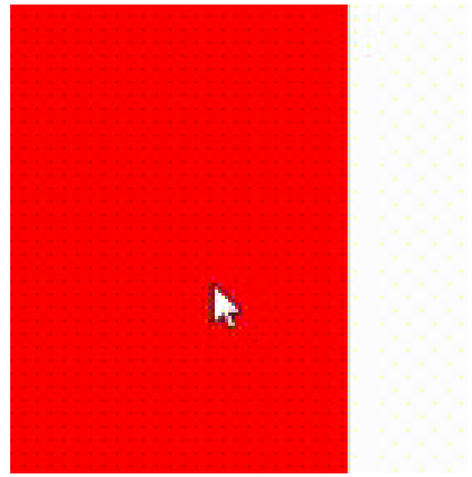
Syntax:

transition-timing-function: easing-function (ease/linear/ease-in/ease-out etc...);

Example:

```
#box {
        width: 200px;
        height: 200px;
        transition-property: width;
        transition-duration: 5s;
        transition-timing-function: ease-out;
}

#box:hover,
#box:focus {
                width: 100px
}
```



MDN: transition-timing-function

CSS transitions: transition-delay

Defines how long to wait between the time a property is changed and the transition actually begins.

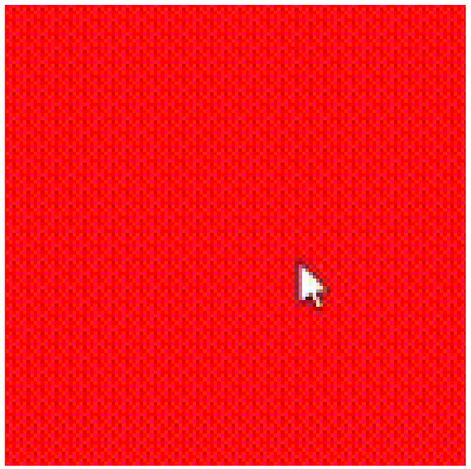
Syntax:

transition-delay: time;

Example:

```
#box {
        transition-property: background-color;
        transition-duration: 1s;
        transition-delay: 5s;
        background-color: red;
}

#box:hover,
#box:focus {
            background-color: #eee;
}
```



MDN: transition-delay

CSS transformations

CSS transformations allow to edit the aspect of elements through rotation, scale, translation or deformation (skew).

MDN: transform

CSS transformation: transform: rotate(angle)

Rotate defines a transformation that rotates an element around a fixed point on the 2D plane, without deforming it.

Syntax:

transform: rotate(a)

Parameter:

a: represents the angle of the rotation. In a left-to-right context, a positive angle denotes a clockwise rotation, a negative angle a counter-clockwise one. In a right-to-left context, a positive angle denotes a counter-clockwise rotation, a negative angle a clockwise one.

MDN: rotate

CSS transformation:transform: scale(x, y)

Scale defines a transformation that resizes an element on the 2D plane.

Syntax:

transform: scale(sx, sy);

Parameters:

- sx: a number or percentage representing the abscissa (horizontal, x-component) of the scaling vector.
- sy: a number or percentage representing the ordinate (vertical, y-component) of the scaling vector. (optional: if not defined, its default value is sx, resulting in a uniform scaling that preserves the element's aspect ratio.

MDN: scale

CSS transformation: transform: translate(x, y)

Translate repositions an element in the horizontal and/or vertical directions.

Syntax:

transform: translate(single-length-percentage, double-length-percentage);

Parameters:

- **Length-percentage**: a length or percentage representing the abscissa (horizontal, x-component) of the translating vector [tx, 0]. The ordinate (vertical, y-component) of the translating vector will be set to 0.
- **Double-length-percentage**: two length or percentage values representing both the abscissa (horizontal, x-component) and the ordinate (vertical, y-component) of the translating vector [tx, ty]. A percentage as first value refers to the width, as second part to the height of the reference box defined by the transform-box property.

MDN: translate

CSS transformation: transform: skew(x, y)

Skew defines a transformation that skews an element on the 2D plane.

Syntax:

transform: skew(ax, ay);

Parameters:

- ax: an angle representing the angle to use to distort the element along the x-axis.
- ay: an angle representing the angle to use to distort the element along the y-axis.
  (Optional: if not defined, its default value is 0, resulting in a purely horizontal skewing.

MDN: skew

CSS animations

CSS animations make it possible to animate transitions from one CSS style configuration to another.

MDN: animations

CSS animation: @keyframes

@keyframes control the intermediate steps in a CSS animation sequence by defining styles for keyframes (or waypoints) along the animation sequence.

Parameters:

- **custom-ident**: A name identifying the keyframe list. This must match the identifier production in CSS syntax
- **from**: starting offset of 0%
- **to**: ending offset of 100%
- **percentage**: A percentage of the time through the animation sequence at which the specified keyframe should occur.
- **timeline-range-name percentage**: A percentage of the time through the specified animation-range at which the specified keyframe should occur

Example:

```
@keyframes slidein {
  from {
    transform: translateX(0%);
  }

  to {
    transform: translateX(100%);
  }
}
```

MDN: @keyframe

CSS animation: animation-name

Animation-name specifies the names of one or more @keyframes at-rules that describe the animation to apply to an element.

Syntax:

animation-name: animation-id/none;

Parameters:

- **none: a special keyword denoting no keyframes. It can be used to deactivate an animation without changing the ordering of the other identifiers, or to deactivate animations coming from the cascade.**
- **animation-id: a name identifying the animation. This identifier is composed of a combination of case-sensitive letters a to z, numbers 0 to 9, underscores (_), and/or dashes (-).**

MDN: animation-name

CSS animation: animation-duration

Animation-duration sets the length of time that an animation takes to complete one cycle.

Syntax:

animation-duration: time/auto;

Parameters:

- auto: **for time-based animations, auto is equivalent to a value of 0s. For CSS scroll-driven animations, auto fills the entire timeline with the animation.**
- time: **the time that an animation takes to complete one cycle. This may be specified in either seconds (s) or milliseconds (ms). The value must be positive or zero and the unit is required. If no value is provided, the default value of 0s is used**

MDN: animation-duration

CSS animation: animation-timing-function

The animation-timing-function CSS property sets how an animation progresses through the duration of each cycle.

Syntax:

animation-timing-function: easing-function;

Example:

animation-timing-function: step-start;

MDN: animation-timing-function

CSS animation: animation-delay

The animation-delay CSS property specifies the amount of time to wait from applying the animation to an element before beginning to perform the animation.

Syntax:

animation-delay: time;

Parameter:

- time: The time offset, from the moment at which the animation is applied to the element, at which the animation should begin. This may be specified in either seconds (s) or milliseconds (ms). A positive value indicates that the animation should begin after the specified amount of time has elapsed. A negative value causes the animation to begin immediately, but partway through its cycle.

MDN: animation-delay

Questions?