**Basic Animation**

The animation shorthand CSS property applies an animation between styles. It is a shorthand for **animation-name, animation-duration, animation-timing-function, animation-delay, animation-iteration-count, animation-direction, animation-fill-mode, and animation-play-state.**

**Constituent properties**

This property is a shorthand for the following CSS properties:

animation-delay

animation-direction

animation-duration

animation-fill-mode

animation-iteration-count

animation-name

animation-play-state

animation-timing-function

**Syntax**

CSS

/\* @keyframes duration | easing-function | delay |

iteration-count | direction | fill-mode | play-state | name \*/

animation: 3s ease-in 1s 2 reverse both paused slidein;

/\* @keyframes duration | easing-function | delay | name \*/

animation: 3s linear 1s slidein;

/\* two animations \*/

animation: 3s linear slidein, 3s ease-out 5s slideout;

**Description**

The order of time values within each animation definition is important: the first value that can be parsed as a <time> is assigned to the animation-duration, and the second one is assigned to animation-delay.

The order of other values within each animation definition is also important for distinguishing an animation-name value from other values. If a value in the animation shorthand can be parsed as a value for an animation property other than animation-name, then the value will be applied to that property first and not to animation-name. For this reason, the recommended practice is to specify a value for animation-name as the last value in a list of values when using the animation shorthand; this holds true even when you specify multiple, comma-separated animations using the animation shorthand.

While an animation name must be set for an animation to be applied, all values of the animation shorthand are optional, and default to the initial value for each long-hand animation component. The initial value of animation-name is none, meaning if no animation-name value is declared in the animation shorthand property, there is no animation to apply on any of the properties.

When the animation-duration value is omitted from the animation shorthand property, the value for this property defaults to 0s. In this case, the animation will still occur (the animationStart and animationEnd events will be fired) but no animation will be visible.

**Configuring an animation IMP!!**

To create a CSS animation sequence, you style the element you want to animate with the animation property or its sub-properties. This lets you configure the timing, duration, and other details of how the animation sequence should progress. This does not configure the actual appearance of the animation, which is done using the @keyframes at-rule as described in the Defining the animation sequence using keyframes section below.

**Defining animation sequence using keyframes**

After you've configured the animation's timing, you need to define the appearance of the animation. This is done by establishing one or more keyframes using the @keyframes at-rule. Each keyframe describes how the animated element should render at a given time during the animation sequence.

Since the timing of the animation is defined in the CSS style that configures the animation, keyframes use a <percentage> to indicate the time during the animation sequence at which they take place. 0% indicates the first moment of the animation sequence, while 100% indicates the final state of the animation. Because these two times are so important, they have special aliases: from and to. Both are optional. If from/0% or to/100% is not specified, the browser starts or finishes the animation using the computed values of all attributes.

**Using the animation shorthand**

The animation shorthand is useful for saving space. As an example, some of the rules we've been using through this article:

p {

animation-duration: 3s;

animation-name: slidein;

animation-iteration-count: infinite;

animation-direction: alternate;

}

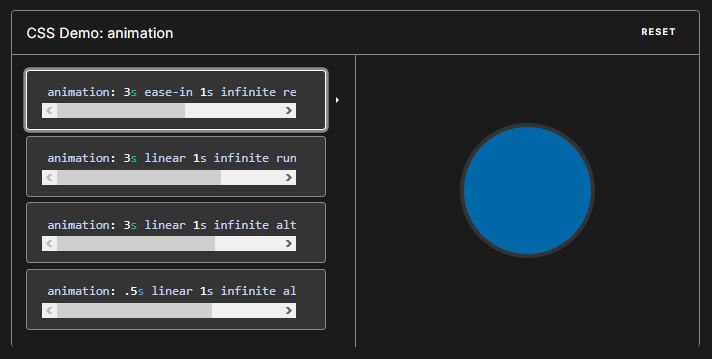
...could be replaced by using the animation shorthand.

p {

animation: 3s infinite alternate slidein;

}

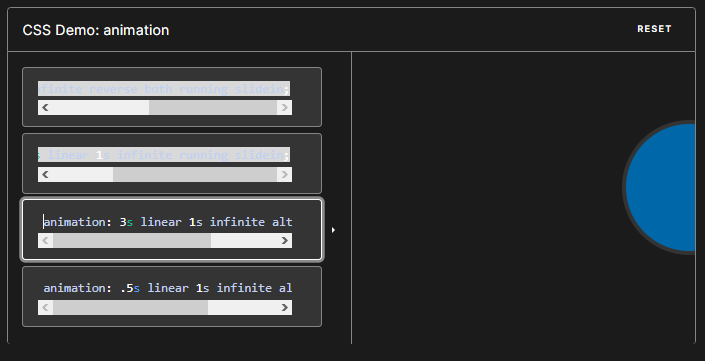
Example:



animation: 3s ease-in 1s infinite reverse both running slidein;



animation: 3s linear 1s infinite running slidein;



animation: 3s linear 1s infinite alternate slidein;



animation: .5s linear 1s infinite alternate slidein;