CSS animation-timing-function Property

❮ Reference ❯

Example

Play an animation with the same speed from beginning to end:

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

Try it Yourself »

More "Try it Yourself" examples below.

Definition and Usage

The `animation-timing-function` specifies the speed curve of an animation.

The speed curve defines the TIME an animation uses to change from one set of CSS styles to another.

The speed curve is used to make the changes smoothly.

Animatable:	no. Read about *animatable*
Version:	CSS3
JavaScript syntax:	*object*.style.animationTimingFunction="linear" Try it

Browser Support

The numbers in the table specify the first browser version that fully supports the property.

Numbers followed by -webkit-, -moz-, or -o- specify the first version that worked with a prefix.

Property					
animation-timing-function	43.0 4.0 -webkit-	10.0	16.0 5.0 -moz-	9.0 4.0 -webkit-	30.0 15.0 -webkit- 12.0 -o-

CSS Syntax

```
animation-timing-function: linear|ease|ease-in|ease-out|ease-in-
out|step-start|step-end|steps(int,start|end)|cubic-
bezier(n,n,n,n)|initial|inherit;
```

The animation-timing-function uses a mathematical function, called the Cubic Bezier curve, to make the speed curve. You can use your own values in this function, or use one of the pre-defined values:

Property Values

Value	Description	Demo

	before it ends slowly	
ease-in	The animation has a slow start	Play it »
ease-out	The animation has a slow end	Play it »
ease-in-out	The animation has both a slow start and a slow end	Play it »
step-start	Equivalent to steps(1, start)	
step-end	Equivalent to steps(1, end)	
steps(int,start|end)	Specifies a stepping function, with two parameters. The first parameter specifies the number of intervals in the function. It must be a positive integer (greater than 0). The second parameter, which is optional, is either the value "start" or "end", and specifies the point at which the change of values occur within the interval. If the second parameter is omitted, it is given the value "end"	
cubic-bezier(*n*,*n*,*n*,*n*)	Define your own values in the cubic-bezier function Possible values are numeric values from 0 to 1	
initial	Sets this property to its default value. Read about *initial*	
inherit	Inherits this property from its parent element. Read about *inherit*	

Tip: Try the different values in the "More Examples" section below.

More Examples

Example

To better understand the different timing function values;
Here are five different elements with five different values:

```
#div1 {animation-timing-function: linear;}
#div2 {animation-timing-function: ease;}
#div3 {animation-timing-function: ease-in;}
```

Example

Same as the example above, but the speed curves are defined with the cubic-bezier function:

```
#div1 {animation-timing-function: cubic-bezier(0,0,1,1);}
#div2 {animation-timing-function: cubic-bezier(0.25,0.1,0.25,1);}
#div3 {animation-timing-function: cubic-bezier(0.42,0,1,1);}
#div4 {animation-timing-function: cubic-bezier(0,0,0.58,1);}
#div5 {animation-timing-function: cubic-bezier(0.42,0,0.58,1);}
```

Try it Yourself »

Related Pages

CSS tutorial: CSS Animations

HTML DOM reference: animationTimingFunction property

COLOR PICKER

Top Tutorials