

Usage

To use animate.css in your website, simply drop the stylesheet into your document's `<head>` , and add the class `animated` to an element, along with any of the animation names. That's it! You've got a CSS animated element. Super!

```
<head>
  <link rel="stylesheet" href="animate.min.css">
</head>
```

or use a CDN hosted version by [CDNJS](#)

```
<head>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/3.7.2/animate.min.css">
</head>
```

Animations

To animate an element, add the class `animated` to an element. You can include the class `infinite` for an infinite loop. Finally you need to add one of the following classes to the element:

Class Name			
bounce	flash	pulse	rubberBand
shake	headShake	swing	tada
wobble	jello	bounceIn	bounceInDown
bounceInLeft	bounceInRight	bounceInUp	bounceOut
bounceOutDown	bounceOutLeft	bounceOutRight	bounceOutUp
fadeIn	fadeInDown	fadeInDownBig	fadeInLeft
fadeInLeftBig	fadeInRight	fadeInRightBig	fadeInUp
fadeInUpBig	fadeOut	fadeOutDown	fadeOutDownBig
fadeOutLeft	fadeOutLeftBig	fadeOutRight	fadeOutRightBig
fadeOutUp	fadeOutUpBig	flipInX	flipInY
flipOutX	flipOutY	lightSpeedIn	lightSpeedOut
rotateIn	rotateInDownLeft	rotateInDownRight	rotateInUpLeft
rotateInUpRight	rotateOut	rotateOutDownLeft	rotateOutDownRight
rotateOutUpLeft	rotateOutUpRight	hinge	jackInTheBox
rollIn	rollOut	zoomIn	zoomInDown
zoomInLeft	zoomInRight	zoomInUp	zoomOut
zoomOutDown	zoomOutLeft	zoomOutRight	zoomOutUp
slideInDown	slideInLeft	slideInRight	slideInUp
slideOutDown	slideOutLeft	slideOutRight	slideOutUp
heartBeat			

Full example:

```
<h1 class="animated infinite bounce delay-2s">Example</h1>
```

[Check out all the animations here!](#)

It's possible to change the duration of your animations, add a delay or change the number of times that it plays:

```
.yourElement {  
  animation-duration: 3s;  
  animation-delay: 2s;  
  animation-iteration-count: infinite;  
}
```

Usage with Javascript

You can do a whole bunch of other stuff with animate.css when you combine it with Javascript. A simple example:

```
const element = document.querySelector('.my-element')  
element.classList.add('animated', 'bounceOutLeft')
```

You can also detect when an animation ends:

```
const element = document.querySelector('.my-element')  
element.classList.add('animated', 'bounceOutLeft')  
  
element.addEventListener('animationend', function() { doSomething() })
```

You can use this simple function to add and remove the animations:

```
function animateCSS(element, animationName, callback) {  
  const node = document.querySelector(element)  
  node.classList.add('animated', animationName)  
  
  function handleAnimationEnd() {  
    node.classList.remove('animated', animationName)  
    node.removeEventListener('animationend', handleAnimationEnd)  
  
    if (typeof callback === 'function') callback()  
  }  
  
  node.addEventListener('animationend', handleAnimationEnd)  
}
```

And use it like this:

```
animateCSS('.my-element', 'bounce')  
  
// or  
animateCSS('.my-element', 'bounce', function() {  
  // Do something after animation  
})
```

Notice that the examples are using ES6's `const` declaration, dropping support for IE10 and some aging browsers. If you prefer, switch the `const` to `var` declarations and IE10 and some old browsers will get support (they still have to provide `classList` support, so do your [research](#)).

Setting *Delay* and *Speed*

Delay Class

It's possible to add delays directly on the element's class attribute, just like this:

```
<div class="animated bounce delay-2s">Example</div>
```

Class Name	Delay Time
delay-2s	2s
delay-3s	3s
delay-4s	4s
delay-5s	5s

Note: The default delays are from 1 second to 5 seconds only. If you need custom delays, add it directly to your own CSS code.

Slow, Slower, Fast, and Faster Class

It's possible to control the speed of the animation by adding these classes, as a sample below:

```
<div class="animated bounce faster">Example</div>
```

Class Name	Speed Time
slow	2s
slower	3s
fast	800ms
faster	500ms

Note: The `animated` class has a default speed of `1s`. If you need custom duration, add it directly to your own CSS code.

Custom Builds

Animate.css is powered by [gulp.js](#), which means you can create custom builds pretty easily. First of all, you'll need Gulp and all other dependencies:

```
$ cd path/to/animate.css/  
$ sudo npm install
```

Next, run `gulp` to compile your custom builds. For example, if you want only some of the "attention seekers", simply edit the `animate-config.json` file to select only the animations you want to use.

```
"attention_seekers": {  
  "bounce": true,  
  "flash": false,  
  "pulse": false,  
  "shake": true,  
  "headShake": true,  
  "swing": true,  
  "tada": true,  
  "wobble": true,  
  "jello": true  
}
```

--