



@CODE.CLASH

translateZ()



# ANIMATIONS

translate3d()



scaleZ()



scale3d()



NEXT →



**Hey Everyones** 🖐️

CSS allows animation of HTML elements without using JavaScript or Flash!

Do Like, save and Share This Post If You Found This Helpful.

In this post you will learn about the following properties:

- `@keyframes`
- `animation-name`
- `animation-duration`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-timing-function`
- `animation-fill-mode`
- `animation`

- `@keyframes` : controls the intermediate steps in an animation sequence by styles for keyframes along the animation sequence.
- `animation-name` : specifies the name of a keyframe animation, and must correspond to a `@keyframes` rule.
- `animation-duration` : set the overall amount of time over which the animation executes.
- `animation-delay` : pause before executing an animation, using the same time values as for duration.

- `animation-iteration-count` : sets the number of times the animation plays.
- `animation-direction` : allows you to play the animation in normal or reverse order.
- `animation-timing-function` : controls the speed of progression between each keyframe.
- `animation-fill-mode` : preserves an animation's start state before a delayed animation executes, its end state after its final iteration, or both.
- `animation-play-state`: pause/play the animation.

values each of these sub-properties can take:

**animation-timing-function**

ease, ease-out, ease-in, ease-in-out, linear, cubic-bezier(x1, y1, x2, y2)

**animation-duration**

Xs or Xms

**animation-delay**

Xs or Xms

**animation-iteration-count**

X

**animation-fill-mode**

forwards, backwards, both, none

**animation-direction**

normal, alternate

**animation-play-state**

paused, running

These sub-properties can then be used like so:

```
.element {  
  background-color: red;  
  animation-name: stretch;  
  animation-duration: 1.5s;  
  animation-timing-function: ease-out;  
  animation-delay: 0;  
  animation-direction: alternate;  
  animation-iteration-count: infinite;  
  animation-fill-mode: none;  
  animation-play-state: running;  
}
```

## @keyframes

controls the intermediate steps in a CSS animation sequence by defining styles for keyframes along the animation sequence.

```
@keyframes stretch {  
  0% {  
    transform: scale(.3);  
    background-color: red;  
    border-radius: 100%;  
  }  
  50% {  
    background-color: orange;  
  }  
  100% {  
    transform: scale(1.5);  
    background-color: yellow;  
  }  
}
```



@CODE.CLASH

**THANKS FOR YOUR ATTENTION**



CODE.CLASH



IMTIYAZ NANDASANIYA

**LIKE AND SAVE IT  
FOR LATER**