Transition property can give a dynamic effect on web pages.

But there are a few disadvantages with transitions.

1. Transition starts only when some event like hover, focus occurs.
2. We can specify only the initial state and final state for transitions, but we cannot specify an intermediate state.

To overcome these disadvantages we use animations of CSS3.

CSS3 Animations can be used to replace animated images, scripts, and flash animations on most of the web pages.

Animations are the same as transitions, but we can specify intermediate states also and animations can start without events

Observe the below loader created using CSS Animations

Animation in CSS3 is done using keyframes. Within each keyframe, we define the start and end of a smooth transition.

We can specify when the change will happen using from & to keywords.

1. @keyframes anim
2. {
3. to {color: white;background:black;}
4. from {color: black;background:white;}
5. }

Also, we can specify when the change will happen by specifying values in percentages: 0% is the beginning, 100% is when the animation is complete.

1. @keyframes anim
2. {
3. 0% {color: white;background:black;}
4. 100% {color: black;background:white;}
5. }

**Best Practice:**

Do not have more than one animation to happen at the same time. Schedule the animations so that they will not get messed up in the page view.

Animation is a short-hand property to specify the properties mentioned below

|  |  |
| --- | --- |
| **animation property** | **Description** |
| animation-name | Every animation is given a name using which it is attached to an element. |
| animation-duration | The time span for an animation to complete one iteration. |
| animation-timing | Controls the pace/ speed of the animation. The key terms used are linear, ease, ease-in, ease-out, ease-in-out. |
| animation-delay | Specifies the delay before starting the animation. |
| animation-iteration-count | Specifies the number of iterations the animation will play.  The Default value is 1. |
| animation-direction | The default value is normal which plays the animation in the forward direction. Takes the value reverse which plays the animation in backward direction Takes the value alternate which plays the animation in the forward direction every odd time and backward direction even time  Takes value alternate-reverse which plays the animation in reverse direction every odd time and forward direction even time |
| play-state | Explains the state of animation. Can take values ‘paused’ or ‘running’.  Paused will pause the animation. Running will resume the animation from where it was paused. |
| fill-mode | The property is used to describe what occurs before the animations start & after the animation ends. |

|  |  |
| --- | --- |
| **fill-mode property** | **Description** |
| none | Default value. The properties remain in their original until the animation starts and once the animation ends the properties return to their original state. |
| forwards | Once the animation ends (determined by animation-iteration-count), the properties retain the final keyframe values. |
| backwards | The animation will apply the property values defined in the keyframe that will start the first iteration of the animation, during the period defined by animation-delay. These are either the values of the from keyframe (when animation-direction is "normal" or "alternate") or those of the to keyframe (when animation-direction is "reverse" or "alternate-reverse") |
| both | The animation will follow the rules of both forwards and backwards properties. That is, it will extend the animation properties in both directions |