# Creating Transitions in CSS

We have learned about keyframes and animations to design the animation. Here we will learn what transitions are and how to create them. CSS transitions allow us to change property values smoothly over a given duration. In addition, it provides a way to control animation speed when changing CSS properties.

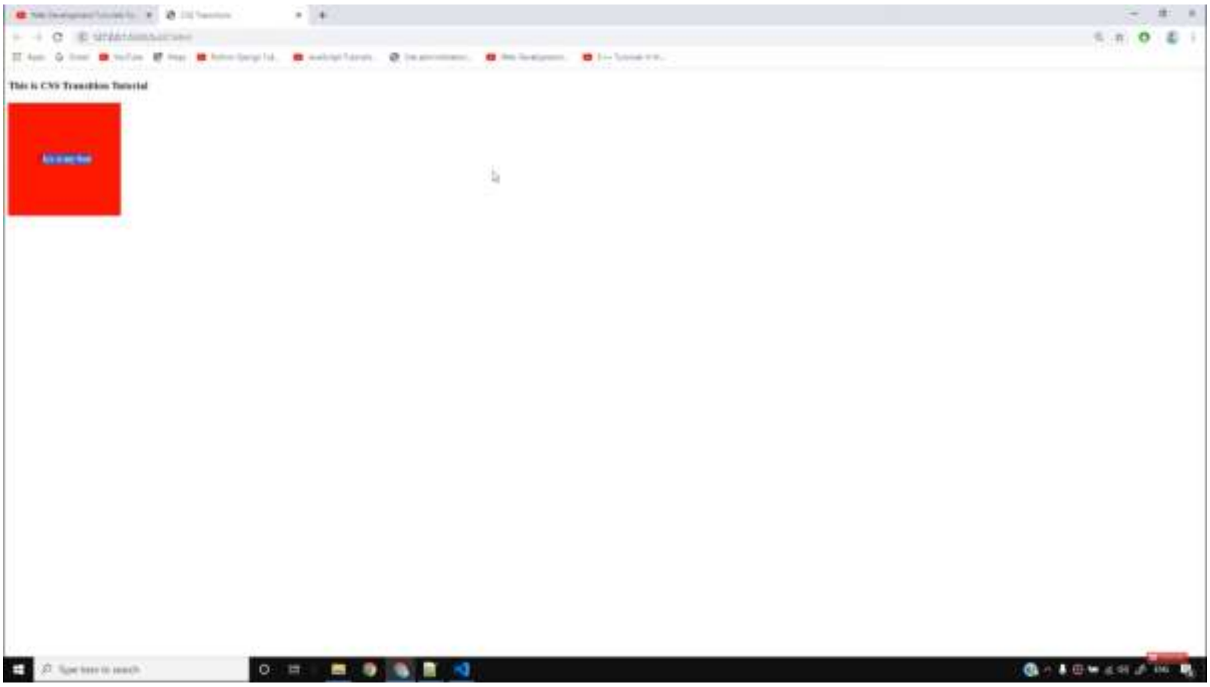
Make a new file add the boilerplate to get the basic HTML code. Give the title as **CSS Transitions** in the <title> tag. Let us now add some HTML code in the <body> tag to get started.

```
<body>
    <h3>This is CSS Transition Tutorial</h3>
    <div class="container">
        <div id="box">
            This is my box
        </div>
    </div>
</body>
```

Now we will add some CSS in the box to see some of the transitions effects-

```
body{
        background-color: black;
    }
#box{
        display: flex;
        height: 200px;
        width: 200px;
        background-color: red;
        justify-content: center;
        align-items: center;
}
```

The **align-items** as center is used here to place the text inside the box in the center as shown below.



This is CSS Transition Tutorial

Now we will make a hover effect which will change the properties when the mouse pointer will hover on the box.

```
     #box:hover{
          background-color: green;
}
```

Let us now discuss some of the transition properties-

1. **Transition-property-** It is used to decide which transition property we want to use. For example, if we want to transition background color, then we have to write-

```
transition-property: background-color;
```

2. **Transition-duration-** If we want to see the duration which is required to make the change, we can use this property. For example, if we set transition duration as 1seconds, then the transition will happen in 1 second only.

```
transition-duration: 1s;
```

3. **Transition-timing-function-** This property is used to decide the speed of transition from beginning to end. These are of three types as follows-

- **ease-in**

After applying this, the animation will start slowly and becomes fast towards the end.

- **ease-out**

After applying this, the animation will begin fastly and become slow towards the end.

- **ease-in-out**

After applying this, the animation will start slowly, then become fast in the midway, and ends slowly.

```
transition-timing-function: ease-in-out;
```

4. **Transition-delay-** It is that particular time interval after which the transition effect will start. For example, if we set it as 2s, then the transition effect will start after 2 seconds only.

```
transition-delay: 2s;
```

Also, there is one *short hand property* that allows us to write all these transitions in a single line. It can be written as follows-

```
transition: background-color 1s ease-in-out 2s;
```

If we want all the properties should go under transition, then we can write-

```
transition: all 1s ease-in-out 2s;
```

Now, we can add some more properties in the hover effect as follows-

```
#box:hover{
        background-color: green;
        height: 400px;
        width: 400px;
        border-radius: 100px;
        font-size: 45px;
        }
```

Here, all the properties will get changed accordingly and you witness some good transitions. All those properties that can change their values like colors, can show different transition properties. You can try different such properties and view the effects of these transitions.

## Transform property in CSS

The **transform** property applies a 2D or 3D transformation to an element. This property allows you to rotate, scale, move, skew., elements. This property is also used in creating animations.

We will start by making a new file and create an instant boilerplate through emmet abbreviation to get the basic HTML template. Give the title as **CSS Transform** in the <title> tag. Then we will write our HTML code to get started as follows-

```
<body>
    <div class="container">
        <div class="box">This is a box</div>
    </div>
</body>
```

Let us now design our box with some CSS as shown below-

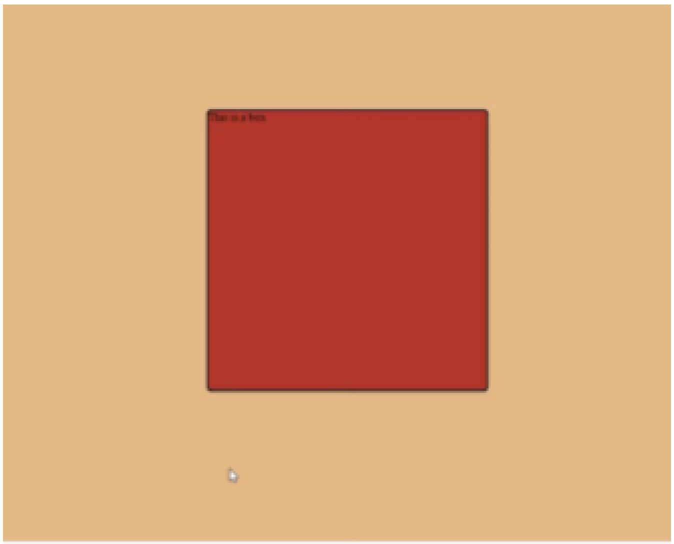
```
.box{
            background: brown;
            border: 2px solid black;
            border-radius: 8px;
            height: 400px;
            width: 400px;
}
```

We will also customize our *container* with the following code as shown below-

```
.container{
            height: 80vh;
            background-color: burlywood;
```

```
                display: flex;
                justify-content: center;
                align-items: center;
            }
```

The box and the container will look as follows-



This is a box.

The box will come to the center. We can reset the margin and paddings already applied as follows-

```
*{
             margin: 0px;
             padding: 0px;
        }
```

We are doing this because you must know what all obstacles occur while creating a website and how to tackle them.

Now we can play with the box and apply different transform properties to it. However, we can add more properties to the box to make it look better.

```
.box{
             display: flex;
             align-items: center;
             justify-content: center;
}
```

We can add transform property directly to the box itself. For example, if we write

```
transform: rotate(45deg);
```

After writing the above code in the box element, it will look like-

We have to apply the transition effect in the box so that it looks like a complete transition effect. For that, we have to add the below code in the box element.-

```
transition: all 0.5s ease-in-out;
```

Now, we can ad the hover effect to add the various transition effects-

- If we want to rotate the box, we can write-

```
.box:hover{
            transform: rotate(360deg);
}
```

It will completely rotate the box to 360 degrees.

- We can also **skew** the box through certain degrees. The skew property is used sometimes when we want to put the content on one side or we want to show the 3D effect. The code is as follows-

```
.box:hover{
            transform: skew(40deg);
}
```

- We can also **scale** the box. The box will become large depending on the values we provide.

```
.box:hover{
              transform: scale(2);
}
```

In the above example, the box will grow 2 times the initial size.

- We can also **translate** or move the box in the x or y axis respectively by providing some values. The code for them is as follows-

```
.box:hover{
            transform: translateX(123px);
            transform: translateY(123px);
}
```

We can also write them in the same line as follows-

```
transform: translate(123px, 123px);
```

So I hope that you must have understood the concept of transform property. You can now create your own awesome websites and practice more to get perfect.