

CSS Transitions and Animations Guide



CSS Transitions and Animations Guide	1
What Are CSS Transitions?	2
Syntax:	2
CSS Animations	2
Syntax:	2
Examples	3

Learn more *HTML*, *CSS*, *JavaScript* Web Development at <https://basescripts.com/> Laurence Svekis

Example 1: Button Hover Transition	3
Example 2: Simple Animation	3
CSS Transition and Animation Properties	4
Transition Properties:	4
Animation Properties:	4
Advanced Examples	4
Example 3: Bouncing Ball Animation	5
Example 4: Rotating and Scaling Animation	5
Exercises	6
Exercise 1: Create a Hover Effect	6
Exercise 2: Build a Loading Animation	6
Multiple-Choice Questions	7
Question 1:	7
Question 2:	7
Question 3:	8

CSS Transitions and Animations are powerful tools to enhance the user experience by adding smooth visual effects. This guide covers the basics, advanced features, and practical examples of CSS Transitions and Animations.

What Are CSS Transitions?

CSS Transitions allow changes in CSS properties to occur smoothly over a specified duration rather than happening instantly.

Syntax:

```
selector {
  transition: property duration timing-function delay;
}
```

- **property:** The CSS property to be transitioned (e.g., color, transform).
- **duration:** Time (e.g., 2s, 500ms) for the transition to complete.
- **timing-function:** Defines the speed curve (e.g., ease, linear, ease-in).
- **delay:** Delay before the transition starts.

CSS Animations

CSS Animations allow you to animate changes in CSS properties using keyframes.

Syntax:

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```

@keyframes animation-name {
  0% { /* Initial styles */ }
  100% { /* Final styles */ }
}
selector {
  animation: name duration timing-function delay iteration-count
direction;
}

```

Examples

Example 1: Button Hover Transition

```

<button class="btn">Hover Me</button>
<style>
  .btn {
    background-color: teal;
    color: white;
    padding: 10px 20px;
    border: none;
    cursor: pointer;
    transition: background-color 0.5s ease-in-out;
  }
  .btn:hover {
    background-color: coral;
  }
</style>

```

Explanation:

- `transition: background-color 0.5s ease-in-out;`: Smoothly changes the background color over 0.5 seconds when the button is hovered.

Example 2: Simple Animation

```

<div class="box"></div>
<style>
  .box {
    width: 100px;
    height: 100px;
    background: blue;
  }

```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```

        animation: move 2s infinite alternate;
    }
    @keyframes move {
        0% {
            transform: translateX(0);
        }
        100% {
            transform: translateX(200px);
        }
    }
}
</style>

```

Explanation:

- `@keyframes move`: Defines the animation's starting (0%) and ending (100%) states.
- `animation: move 2s infinite alternate;`: Applies the animation, making it repeat infinitely and reverse direction alternately.

CSS Transition and Animation Properties

Transition Properties:

- `transition-property`: Specifies which property to transition.
- `transition-duration`: Specifies the duration of the transition.
- `transition-timing-function`: Specifies the speed curve.
- `transition-delay`: Specifies a delay.

Animation Properties:

- `animation-name`: Specifies the name of the `@keyframes`.
- `animation-duration`: Specifies the time the animation takes.
- `animation-timing-function`: Specifies the speed curve.
- `animation-delay`: Delays the start of the animation.
- `animation-iteration-count`: Specifies the number of times the animation should run.
- `animation-direction`: Defines whether the animation runs forward, backward, or alternates.

Advanced Examples

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

Example 3: Bouncing Ball Animation

```
<div class="ball"></div>
<style>
  .ball {
    width: 50px;
    height: 50px;
    background: red;
    border-radius: 50%;
    position: relative;
    animation: bounce 1.5s infinite ease-in-out;
  }
  @keyframes bounce {
    0%, 100% {
      top: 0;
    }
    50% {
      top: 200px;
    }
  }
</style>
```

Explanation:

- The `@keyframes bounce` makes the ball move up and down by modifying the `top` position.

Example 4: Rotating and Scaling Animation

```
<div class="box"></div>
<style>
  .box {
    width: 100px;
    height: 100px;
    background: purple;
    animation: rotate-scale 3s infinite linear;
  }
  @keyframes rotate-scale {
    0% {
      transform: rotate(0deg) scale(1);
    }
  }
</style>
```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```

    }
    50% {
        transform: rotate(180deg) scale(1.5);
    }
    100% {
        transform: rotate(360deg) scale(1);
    }
}
</style>

```

Exercises

Exercise 1: Create a Hover Effect

1. Create a button that changes its text color and background color with a smooth transition over 1 second.

Solution:

```

<button class="hover-btn">Hover Over Me</button>
<style>
.hover-btn {
    padding: 10px 20px;
    background: lightblue;
    color: black;
    border: none;
    transition: background 1s, color 1s;
}
.hover-btn:hover {
    background: darkblue;
    color: white;
}
</style>

```

Exercise 2: Build a Loading Animation

1. Create a circle that rotates continuously like a loading spinner.

Solution:

```

<div class="spinner"></div>

```

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

```
<style>
.spinner {
  width: 50px;
  height: 50px;
  border: 5px solid lightgray;
  border-top: 5px solid blue;
  border-radius: 50%;
  animation: spin 1s infinite linear;
}
@keyframes spin {
  0% {
    transform: rotate(0deg);
  }
  100% {
    transform: rotate(360deg);
  }
}
</style>
```

Multiple-Choice Questions

Question 1:

What property is used to specify the duration of a CSS Transition?

1. animation-duration
2. transition-time
3. transition-duration
4. duration

Answer: 3. transition-duration

Explanation: transition-duration specifies the time a transition takes to complete.

Question 2:

What does animation: myAnimation 3s infinite alternate; do?

1. Runs the animation indefinitely in alternate directions.
2. Runs the animation 3 times and stops.
3. Alternates between two animations.

Learn more HTML, CSS, JavaScript Web Development at <https://basescripts.com/> Laurence Svekis

4. None of the above.

Answer: 1. Runs the animation indefinitely in alternate directions.

Explanation: The `infinite` keyword makes the animation repeat indefinitely, and `alternate` reverses its direction every iteration.

Question 3:

Which CSS property is NOT supported by transitions?

1. `background-color`
2. `transform`
3. `display`
4. `opacity`

Answer: 3. `display`

Explanation: The `display` property cannot be transitioned as it does not have intermediate states.