Transitions & Animations

Using CSS

Transitions

In CSS, transitions allow elements to change from one state to another over a period of time. For example, when using the CSS :hover selector to make an element change color on hover, the color changes abruptly. With CSS transitions you can create a smooth change between colors by customizing the duration, and speed of the transition.

Example:
Live Site: https://puffy-petal-violin.glitch.me
Code: https://glitch.com/edit/#!/puffy-petal-violin
In this example, the background transitions from yellow to blue on hover.

The Four Components of CSS Transitions:

- Property, that will transition (background color, color, font size, width, height)
  - Here is a full list of properties that can use transitions.
- Duration, declares how long the transition will take
- Delay, defines the pause before the transition begins
- Timing, a function that specifies the acceleration of the transition

```
div {
transition-property: color;
transition-duration: 1s;
transition-delay: 400ms;
transition-timing-function: ease-out;
}
```

Example:
Live Site: https://knotty-sophisticated-october.glitch.me/
Code: https://glitch.com/edit/#!/knotty-sophisticated-october
In this example the text grows over a period of 5 seconds.

Example:
Live Site: https://crimson-small-settee.glitch.me
Code: https://glitch.com/edit/#!/crimson-small-settee
In this example the image grows on hover.

Transition (shorthand) Property:
You can also use the transition shorthand property to declare up four properties in one line.

```
div {
transition: background-color 0.5s ease-in 4s;
}
```

Summary:
You can use CSS transitions to change an element from one state to another. In order for this to work, you need to define a beginning state and an ending state, meaning, transitions only occur between 2 states.

References:

- Using CSS Transitions https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Transitions/Using_CSS_transitions
- Transition Shorthand Property:
  - https://developer.mozilla.org/en-US/docs/Web/CSS/transition
  - https://css-tricks.com/almanac/properties/t/transition/

More Examples:

- https://codepen.io/ajc100/pen/QNPyWa
- https://codepen.io/felipefialho/pen/AwYmMe

Animations

CSS animations allow you to change the states of your elements over a period of time, with much more control. Unlike CSS transitions, CSS animations allow you to have many changes happen between the start and the end of the animation. There is also no limit to which CSS properties you can animate.

You can animate your elements by using @keyframes. Inside of the **@keyframes** rule is where you define the state changes for the elements you want animated. The steps of the animation can be defined using **from** & **to** or **percentages**.

This example gives a result that is similar to a transition, because there are only two state changes.

```
@keyframes AnimateMe {
  from {color: yellow;}
  to {color: pink;}
}
```

In this example you can see the true power of CSS Animations. Multiple state changes are declared within each percentage.

```
@keyframes AnimateMe {
  0% {color: yellow; font-size: 15px; background: orange; transform: scale(.5);}
  25% {color: orange; font-size: 30px; background: blue; transform: scale(1);}
  50% {color: blue; font-size: 60px; background: pink; transform: scale(1.5);}
  75% {color: green; font-size: 90px; background: yellow; transform: scale(2);}
  100% {color: pink; font-size: 120px; background: black; transform: scale(2.5);}
}
```

In order for **@keyframes** to work, the **animation** property needs to be declared, and must contain a name for the animation. With the animation property, you can define the name of the animation, timing function, duration, number of repetitions, delay etc.

These are all of the animation attributes you can specify:

- animation-name
- animation-duration
- animation-timing-function
- animation-delay
- animation-iteration-count
- animation-direction
- animation-fill-mode
- animation-play-state

```
div {
    width: 400px;
    height: 400px;
    animation-name: AnimateMe;
    animation-duration: 10s;
    animation-timing-function: ease-in;
    animation-iteration-count: infinite;
}
```

Animation short-hand property

```
div {
    animation: AnimateMe 10s ease-in infinite;
}
/* name duration timing-function iteration-count */
```

Example:
Live Site: https://pouncing-lavish-tornado.glitch.me
Code: https://glitch.com/edit/#!/pouncing-lavish-tornado

References

- Transition vs Animation: https://cssanimation.rocks/transition-vs-animation/
- CSS Tricks: Animation:

https://css-tricks.com/almanac/properties/a/animation/

- Animation property: https://www.w3schools.com/cssref/css3_pr_animation.asp
- @keyframes: https://www.w3schools.com/cssref/css3_pr_animation-keyframes.asp