CT053-3-1
Fundamentals of Web Design and Development

CSS3 Animations

What are CSS3 Animations?

- An animation lets an element gradually change from one style to another.
- You can change as many CSS properties you want, as many times you want.
- To use CSS3 animation, you must first specify some keyframes for the animation.
- Keyframes hold what styles the element will have at certain times.

The @keyframes Rule

- When you specify CSS styles inside the @keyframes rule, the animation will gradually change from the current style to the new style at certain times.

- To get an animation to work, you must bind the animation to an element.

Example #1

```
div {
    width: 500px;
    height: 500px;
    border-radius: 50%;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
}

@keyframes changeColor {
    from {background-color: red;}
    to {background-color: purple;}
}
```

Can you describe the shape of the ?

In this example we specified when the style will change by using the keywords "from" and "to" (which represents 0% (start) and 100% (complete)).

Example #2

```
div {
    width: 500px;
    height: 500px;
    border-radius: 50%;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
}

@keyframes changeColor {
    0%   {background-color: red;}
    25%  {background-color: yellow;}
    50%  {background-color: blue;}
    100% {background-color: green;}
}
```

It is also possible to use percent. By using percent, you can add as many style changes as you like.

Example #3

```
div {
    width: 100px;
    height: 100px;
    border-radius: 50%;
    background-color: red;
    position: relative;
    animation-name: changeColor;
    animation-duration: 4s;

}

@keyframes changeColor {
    0%   {background-color:red; left:0px; top:0px;}
    25%  {background-color:yellow; left:200px; top:0px;}
    50%  {background-color:blue; left:200px; top:200px;}
    75%  {background-color:green; left:0px; top:200px;}
    100% {background-color:red; left:0px; top:0px;}
}
```

This example will change both the background-color and the position of the element when the animation is 25% complete, 50% complete, and again when the animation is 100% complete

Delay an Animation

- The animation-delay property specifies a delay for the start of an animation.
- Example of usage:

```
div {
    width: 100px;
    height: 100px;
    position: relative;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
    animation-delay: 2s;
}
```

Set How Many Times an Animation Should Run

- The animation-iteration-count property specifies the number of times an animation should run.

```
div {
    width: 100px;
    height: 100px;
    position: relative;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
    animation-iteration-count: 3;
}
```

This example will run the animation 3 times before it stops:

```
div {
    width: 100px;
    height: 100px;
    position: relative;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
    animation-iteration-count: infinite;
}
```

This example uses the value "infinite" to make the animation continue for ever

Run Animation in Reverse Direction or Alternate Cycles

- The animation-direction property is used to let an animation run in reverse direction or alternate cycles.

```
div {
    width: 100px;
    height: 100px;
    position: relative;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
    animation-iteration-count: 3;
    animation-direction: reverse;
}
```

This example will run the animation in reverse direction.

```
div {
    width: 100px;
    height: 100px;
    position: relative;
    background-color: red;
    animation-name: changeColor;
    animation-duration: 4s;
    animation-iteration-count: 3;
    animation-direction: alternate;
}
```

This example uses the value "alternate" to make the animation first run forward, then backward, then forward.

Specify the Speed Curve of the Animation

- The animation-timing-function property specifies the speed curve of the animation.
- The animation-timing-function property can have the following values:
  - **ease** - specifies an animation with a slow start, then fast, then end slowly (this is default)
  - **linear** - specifies an animation with the same speed from start to end
  - **ease-in** - specifies an animation with a slow start
  - **ease-out** - specifies an animation with a slow end
  - **ease-in-out** - specifies an animation with a slow start and end
  - **cubic-bezier(n,n,n,n)** - lets you define your own values in a cubic-bezier function

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    width: 100px;
    height: 50px;
    background-color: red;
    font-weight: bold;
    position: relative;
    animation: mymove 5s infinite;
}

#div1 {animation-timing-function: linear;}
#div2 {animation-timing-function: ease;}
#div3 {animation-timing-function: ease-in;}
#div4 {animation-timing-function: ease-out;}
#div5 {animation-timing-function: ease-in-out;}

@keyframes mymove {
    from {left: 0px;}
    to {left: 300px;}
}
</style>
</head>
<body>

<div id="div1">linear</div>
<div id="div2">ease</div>
<div id="div3">ease-in</div>
<div id="div4">ease-out</div>
<div id="div5">ease-in-out</div>

</body>
</html>
```

Animation Shorthand Property

The example below uses six of the animation properties:

```
div {
    animation-name: changeColor;
    animation-duration: 5s;
    animation-timing-function: linear;
    animation-delay: 2s;
    animation-iteration-count: infinite;
    animation-direction: alternate;
}
```

The same animation effect as above can be achieved by shorthand animation property:

```
div {
    animation: changeColor 5s linear 2s infinite alternate;
}
```

Summary of @keyframes rule and all the animation properties

Property	Description
@keyframes	Specifies the animation code
animation	A shorthand property for setting all the animation properties
animation-delay	Specifies a delay for the start of an animation
animation-direction	Specifies whether an animation should play in reverse direction or alternate cycles
animation-duration	Specifies how many seconds or milliseconds an animation takes to complete one cycle
animation-fill-mode	Specifies a style for the element when the animation is not playing (when it is finished, or when it has a delay)
animation-iteration-count	Specifies the number of times an animation should be played
animation-name	Specifies the name of the @keyframes animation
animation-play-state	Specifies whether the animation is running or paused
animation-timing-function	Specifies the speed curve of the animation

Quick Exercise

Write a **CSS animation code** to change both the background-color and the position of the element when the animation is 50% complete, when the animation is 100% complete. Animation movement setting as follows:

Completion %	Transform	Background Color	Position from left
0%	-	Red	0px
50%	Rotate 20 deg	Yellow	50%
100%	Rotate - 360 deg	Green	0px

Animation duration is **5s** and make the **animation continue for ever**. Use **animated_div** as @keyframes rule.

Submit your answer for attendance.