

Web Development Club: Week 10



CSS transitions and
animations

1
0
0
1
0
1
0
1
0
1
0
1
0
1
0
0

1
0
1
0
0
1
0
1
0
1
0
0
1
0
1
0



Table of contents

01

Week 9 recap

02

CSS transitions

03

Animations

04

Practice

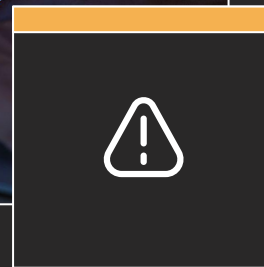
1
0
1
0
1
0
1
0
1
1
1
0
0
0
0
1
1
1
0

1
1
0
0
1
0
1
0
1
0
1
0
1
1
1
0

Week 09 Recap:

- In week 9 we did a winter break competition to build a website including a snowman made entirely out of HTML
- The winner would be awarded any item (within reason) from QT!
- We would then vote on the winner!

And the winner is...



1
0
0
1
0
1
0
1
0
1
0
1
0
1
0
0

1
0
1
0
0
1
0
1
0
1
0
0
1
0
1
0



Winner:

Jason

1
0
1
0
1
0
1
1
1
0
0
0
1
1
1
0

1
1
0
0
1
0
1
0
1
0
1
0
1
1
1
0

CSS Transitions

Transition

- Transition
 - Shorthand property that sets different properties to transition an element in a single declaration.
- Transition-delay
 - Specifies how long an element should wait before beginning a transition.
- Transition-duration
 - Specifies how long an element's transition should take to complete.
- Transition-property
 - Specifies the property or properties of an element that a transition effect should apply to.
- Transition-timing-function
 - Specifies the speed of a transition effect over the course of its duration.

CSS Transition Example

```
1 <style>
2 div {
3   width: 100px;
4   height: 100px;
5   background: red;
6   transition: width 2s;
7   Transition-timing-function:
8   linear;
9 }
10
11 div:hover {
12   width: 300px;
13 }
14 </style>
```

See this transition in
action:
[W3Schools HTML
editor](#)

The transition-timing-function property can have the following values:

- ease - specifies a transition effect with a slow start, then fast, then end slowly (this is default)
- linear - specifies a transition effect with the same speed from start to end
- ease-in - specifies a transition effect with a slow start
- ease-out - specifies a transition effect with a slow end
- ease-in-out - specifies a transition effect with a slow start and end

1
0
1
0
0
1
0
0
1
0
1
1
0
1
1
1

CSS Animations

CSS allows for animations without using Javascript!

- Keyframes hold what CSS styles the element will have at certain times. The animation will gradually change from the current style to the new style during the keyframe
- The animation-duration property defines how long an animation should take to complete.
- When an animation is finished it goes back to its original style
- You can also specify aspects like:
 - animation-name
 - animation-duration
 - animation-delay
 - animation-iteration-count
 - animation-direction
 - animation-timing-function
 - animation-fill-mode
 - animation

CSS Animations Example

```
1 <style>
0 div {
0   width: 100px;
1   height: 100px;
0   background-color: red;
1   position: relative;
0   animation-name: example;
0   animation-duration: 4s;
1 }
0 @keyframes example {
1   0%   {background-color:red; left:0px; top:0px;}
0   25%  {background-color:yellow; left:200px; top:0px;}
1   50%  {background-color:blue; left:200px; top:200px;}
0   75%  {background-color:green; left:0px; top:200px;}
1   100% {background-color:red; left:0px; top:0px;}
0 }
1 </style>
```

See this animation in action:
[W3Schools HTML editor](#)

1
0
1
0
0
1
0
0
1
0
1
0
1
1
1



Practice

**Time to Practice
on your own!**

Feel free to ask for
help!

1
0
0
1
0
1
0
1
0
0
1
0
1
0
1
0
1
0

1
0
1
0
0
1
0
0
1
0
1
1
0
1
1
1
1