**CSS animation**

By now you should know the basics of CSS animation.

It's time to explore

some more advanced properties and rules.

In this video, you will learn

how to use the animation property in

combination with the @keyframes rule

to create advanced animations.

When developers want to create complex animations,

they use the animation property.

But to control the execution

of the steps of the animation,

they use the @keyframe rule.

Next, let's explore the syntax of the @keyframe rule,

and the animation property.

The @keyframe rule specifies the name of the animation,

and the individual keyframes for the animation.

You define each keyframe with

a keyframe selector and CSS styles for that keyframe.

The @keyframe rule can also use the optional keywords,

from and to, to show transitions.

The animation name links

the animation to a specific keyframe rule.

The animation property has a number of

subproperties like animation name,

animation delay, and so on.

You can use a shorthand CSS syntax to

set all the subproperties of the animation property.

But say you want one or more

of them to have the default settings,

you simply leave them out.

However, each of the subproperties

can be listed individually as well.

To understand animation cues,

let's explore a simple animation of

a clock with hour and minute hands.

The HTML code for the clock will

consist of only three basic elements,

which are the div class called clock, and inside it,

a div class called minutes,

and another div class called hours.

Take note how both hours and minutes

have an additional common class called hand.

In order to let the clock work,

you need this additional hand class

with its own CSS rules.

But since the focus of this lesson is to

understand the animation part of the CSS,

the basic styling properties of

the clock are not covered now.

All that you need to know is that you create two classes,

clock and hand,

and insert style rules that specify

the basic layout and appearance of the clock and hands.

So far, the output of the CSS is

a static blue clock face and

red hands with white borders.

Now, you can configure the animation

for both the hour and minutes classes.

The only difference between the two will be

the speed at which they change positions.

For the minutes class, you add the animation property

with the value of six seconds, infinite linear.

For the hours class,

you add the animation property with the value

of cycle 60 seconds infinite linear.

These two rules set

the animation name cycle to complete iterations for

respective timings over an infinite time duration

and linear timing function.

Since the other animation properties are not mentioned,

they are set to the default values.

Finally, you can add the @keyframe rule.

Type @keyframes cycle,

and then the two special keywords,

namely from and to.

For the from keyword,

add the transform property with

the value of rotate zero degrees.

For the to keyword,

add the transform property with

the value of rotate 360 degrees.

Since keyframes is a type of rule in CSS,

we will be adding these details inside curly braces.

There will be a follow-up reading

about keyframes after this video.

What exactly does this CSS animation do?

Well, it uses the transform property

to rotate the cycle associated with

both the hours and minute classes from

the value of 0-360 degrees.

The combination of properties and rules will give

a seamless appearance of a clock in continuous motion.

In this video, you've learned how

to use the animation property and

the @keyframe rule together to get a specific effect.

CSS offers endless creativity when it comes to animation.

The properties and rules you covered

can be configured in many more ways.

For instance, you can

combine them with other elements like

images to achieve amazing animations using purely CSS.