# ***THE BACKGROUND PROPERTY***

[MUSIC]

Up until now,

we've been using the background property just to set the background color.

But the truth is, there's many more properties we could set.

So let's jump into the code editor and take a look.

Play video starting at ::15 and follow transcript0:15

Okay, so here I am in Sublime Text and I'm looking at the file called

background-before.html and it's located in the examples, Lecture20 folder.

And the structure of this HTML document is pretty simple.

We have the h1 just to basically tell us what the title of this whole thing is

be visible.

And we have the div with an id called bg, and so

the content just says, background properties are fun.

And the bg is styled with an id selector and width being 500 and height being 500.

So if we resize our editor a little bit and expose the browser, so

we see that's basically the page that's being represented.

Let's go ahead and close the sidebar.

Okay, and let's do what we have done previously,

which is set the background color of this element, and let's set it to blue.

And if we refresh the page,

we see that we have 500 by 500 pixels square that we've created.

So having a background as a color, as a solid color, is nice, but

what if we wanted an image as a background color?

Well, there's a property to take care of that and

the property is called background-image.

And if we do that, we could set up, supply a URL with parentheses and inside there,

and it's best practice just to put it in quotation marks, you could specify a file.

So let me take a look here at my file explorer here and I see I have yaakov.png.

And if we take a look at it, ye,s that's a caricature of me.

And so I want to display that as the background image.

I'll specify yaakov.png right here.

Now, before we go on, notice that the URL that you specify right here,

let me close this real quick.

In fact, let me open it back up and make this bigger a little bit.

Notice that the URL that you supply is a relative variable, but

it has to be relative to your CSS.

So if this CSS that we're writing, which is right now we're just writing it inside

the style tag, but if we were to put it inside of a subfolder like CSS or

styles subfolder, the URL would be different.

The URL would have to be relative to the CSS file, or actually,

to be very technically correct,

the URL would have to be relative to the way the CSS file is served in the browser.

So slash something something, that's the URL that it would need to be relative to.

So let's go back to the browser and we'll close up the sidebar,

save this and let's reload this.

And voila, we have our background image.

And you can see that it's being repeated, and that's actually yet

another property than you can set.

So let's go ahead and put a semicolon here and

say background-repeat, and here we have a few options.

We have no-repeat, repeat, repeat-x, repeat-y.

So if we do repeat-x, or was it repeat-y,

what all it's going to do is it's just going to repeat it vertically.

Or we could just say no-repeat.

And no-repeat will just put one image in the top left corner by default.

Also notice that we're using a background-color property and

a background-image property at the same time.

And you can see that image wins on terms of priority.

It is on top of the color.

We can also specify the position of the image within the element.

So let's go ahead and do that right now.

We'll say background-position and we'll give it two values.

First is a horizontal value and then the vertical value.

So let's say we want it to be top and right.

If we save that and refresh, now we're top and right.

Let's make it a little bit more interesting, bottom.

Bottom and right.

Now, specifying only one of these will make the other one default to center.

So if I just specify right and

refresh, you'll see that it's now vertically in the center.

And similarly, if I specify bottom and refresh,

you'll see it'll be at the bottom, but horizontally now it will be in the center.

So these properties, background-color, background-image, repeat, and so on,

they're all individual properties,

but the truth is, you can combine them all into one.

And let's go ahead and do that now.

Let's go ahead and erase this and specify just the background.

And if you specify just the background, you can now list these same properties

right here without having to specify them one by one.

Here I'll specify my background-image,

I'll specify that I don't want it to be repeated, no-repeat.

And I can also specify its location, so

let's say I want it to be right and center.

If I save that and

refresh, you can see that my image did move to the right and it's in the center.

Now, something strange, though, happened, right?

I mean, where did the color: blue disappear to?

I thought it was both, we could have both the color: blue and

at the same time have a background image.

Well, the reason it disappeared is because my background property is

being specified after the background-color property.

And since choose to used the background without the dash for

a particular property, it's overriding every single property except

the ones that I'm actually specifying.

So if I wanted to, I could go ahead and say blue and refresh and my blue is back.

So the point is, if you specify background-color specifically and

then you go ahead and override it with the background, but you don't specify

the color or you don't specify whatever the specific property is, whether

it's background-image or background-repeat or background-position and so on.

If you went ahead then and specified background without any dash subproperty,

whatever properties that are inherited with a dash will be overridden unless

you actually call them out directly, straight in the background property.

Play video starting at :6:3 and follow transcript6:03

Okay, so to summarize, we spoke about the background property and its subproperties,

image, color, position, etc.

And the background property can be pretty simple, but

it can also get pretty complex, and it's used for

adjusting image, size, and resolution based on screen size.

Now, I know we haven't seen that, because we haven't seen media queries yet, but

we will in just a few lectures.

Next, we're going to talk about positioning elements by floating.