**Assignment vs. comparison**

Selecting transcript lines in this section will navigate to timestamp in the video

- [Instructor] You've probably noticed in JavaScript, the equal symbol is not used the same way we use it in math. In JavaScript, the equal symbol means assignment. The value on the right side is assigned to the container on the left. Anytime you see a single equal symbol, it's an assignment of a value to a variable. We also use equal symbols for comparison and that's what you see down here. We have two equal symbols next to one another, And that says, we're looking to see if the value of A equals the value of B. Are they the same or not? This code example in the exercise files is a really basic example to demonstrate how these different comparisons work. What we're doing here is just assigning a value to A and B and then we're console logging out, what the values are and also the type of data that's inside, each of these variables. Why we're doing that will become apparent in the second. Then we use a basic conditional statement. We'll cover conditions later on, but for now just know we're asking if this statement here is true. A is the same as B, then console log out, match. If not, then console log out no match. Right now, the values are five and four. If we save this and go to the browser, we get the output five as a number four as a number, and there is no match. If I go and change this to five and five, save it, go back in the browser. Five is a number five is a number and we have a match. The key to all this is this comparison. The two equal symbols tell us that we are doing a loose comparison. Meaning we're saying, are the values the same? But not absolutely the same. What I mean by that is I can actually change one of these values to a string. Technically speaking, we now have a number and a string. But if I run this in the browser, you'll see even though five is a number and five is a string, we still have a match. That's because these two values look the same. I call this semiotic equivalent, because the two symbols look the same and carry the same meaning right now. However, if I were to put in this text five and save it, we would not have a match because even though the number five and the text five, both save five, they're not the same thing. So JavaScript is trying to be clever here in saying that if you're just quoting a single number, you're probably trying to pass me a real number. So I'm going to tell you that these are the same even though they're not. If you want to test for that, to make sure you actually have two numbers, and not a number and a string with a number inside it, you can do an absolute equivalence test by adding an extra equal symbol. Now we have three equal symbols. This looks for absolute equality. The value on one side, and the other side has to be identical. Exactly the same. You can see it doesn't work here. There's no match because we have a number on a string. If we put in just the number, we get a match. But if I then do something else like I'll say five and five, there's no match, because the two values are not exactly the same. What you see here is any type of comparison like this, Either with two equal symbols or three equal symbols, returns either true or false. And then we can use that output in a conditional statement to say if the statement is true, then do something. If it's not true, then do something else. We also have some other comparison tools. We have the larger than. So if we say A is larger than B. If that's true, then we should get the true statement here. I'm going to have to actually make it true. So I'll say four, now A is larger than B. We get a true statement. If we turn it the other way and say A is smaller than B, then we get a false statement. We can also use larger than or smaller than equals two. Meaning if A is either larger, than or equal to B, then it's true. Otherwise it's false. And we can do the same thing with smaller than. And finally, if we want to see if something is not equal to, we can use a bang. An exclamation point and say, if A is not equal to B or if A is not absolutely equal to B, then do something. That's effectively just reversing the statement. Instead of testing whether it's true, we're testing whether it's false.