* - [Instructor] In this chapter, we'll take a step back and look at some fundamentals of JavaScript including variables, data types, and various operators.
* For this purpose, have created the most basic of exercise files, a blank HTML document calling in a single JavaScript file.
* This gives us a good playground to experiment in the console.
* We'll start with variables.
* The classic variable container in JavaScript is simply called **var**, and it's defined with the var statement.
* ***You assign a value to a var by giving it first a name then using the assignment symbol, the equal symbol and then adding a value to it.***
* In the console, I can create a new var by saying var, and then give it a name, container, and then use the equal symbol to assign a value to it.

Graphical user interface, application, Teams

Description automatically generated

* So I'll assign it to value of five.
* Now I can call in the container at any time and when I do, I get the value.
* So, five.
* So I can say container and the browser will automatically tell me, if you do this, you get the value number five.

Graphical user interface, text, application, email

Description automatically generated

* And if I actually execute, I get five back.
* Now, **var is mutable.**
* *Meaning you can change its value at any time and assign it a new value.*
* So I can now grab container and set the value equal to red.
* And this is a string of texts.
* Hit return.

Chart

Description automatically generated with medium confidence

* And when I now call container back, I get the string red in it's place.
* Five is gone and will never reappear unless I reassign container to the number five.
* ***Var was the default for all variables for decades***.
* So you'll come across it in tutorials and examples and in code all the time.
* It is also the default if we forget to say that we are creating a new variable.
* So look at this.
* If I just create a random variable and assign a name to it and assign a value to it without using the var keyword, it'll still appear as a variable but the browser will not be happy.

Graphical user interface, application

Description automatically generated

* So I'll say something equals seven and hit return.
* Now we have a variable called something, and if I hit return, I get a seven.
* Now this variable is a var, we just never declared it.
* And if you run this through your code editor, your code editor will tell you, hey, hey, hey, you forgot to use the var statement.
* So I don't really know what you're trying to do.
* **One more thing, you can define multiple different variables at the same time using the var statement.**
* So if you know that you're going to have several different variables, you can use a comma separated list to declare them all at the same time.
* Var, so if I make var x equals four, y equals five and z equals blue and hit return.
* I now have three variables; x, which is four.
* y which is five and z which is blue.
* Graphical user interface, text, application, email

  Description automatically generated
* **Finally, I can create variables that don't have any content yet.**
* So I can say var empty and just hit return.
* Now we have undefined variables.

Graphical user interface, text, application

Description automatically generated

* That means the variable exists but it contains nothing or it is undefined.
* It has nothing assigned to it.
* So when I call back empty now, you'll see it says undefined.
* That means it's there, but to use it, I need to put something inside it.