**Statements vs Expressions**

We have looked at the console, and I thought this will be a good time to explain a relatively advanced topic in JavaScript.

Don't worry, I'll explain it as simply as I can, in a way that I hope makes sense.

You ready?

Here it is:

In JavaScript, every piece of code is either an *expression* or a *statement*.

And when it comes the console:

**Expressions always return a value.**

**Statements always return undefined.**

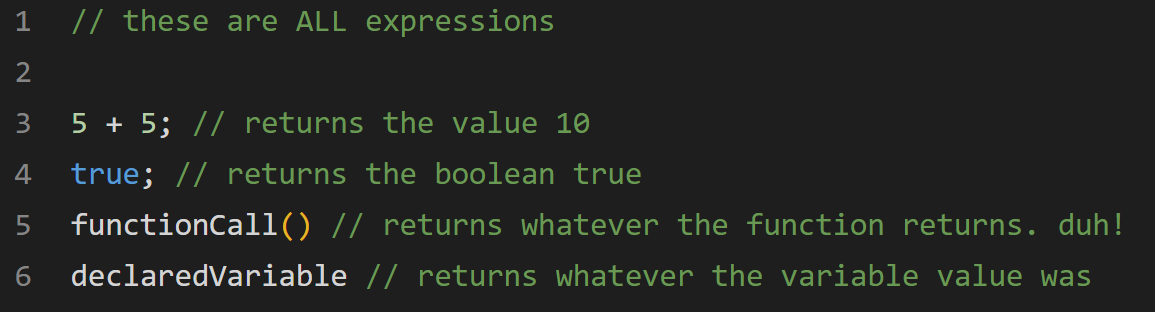
Now you’re probably asking, “what is a statement and what is an expression?” That is a great question, and the differences are defined in the [original JavaScript specification from 1997](http://www.ecma-international.org/publications/files/ECMA-ST-ARCH/ECMA-262,%201st%20edition,%20June%201997.pdf). You don’t have to worry about wrapping your head around this right now, but if you’re curious to know how I think about an expression vs a statement, then read on …

**What is an expression in JS**

An expression is any valid unit of code that resolves to a value.

Simple enough.

Things like these are all expressions (remember, they return a value):

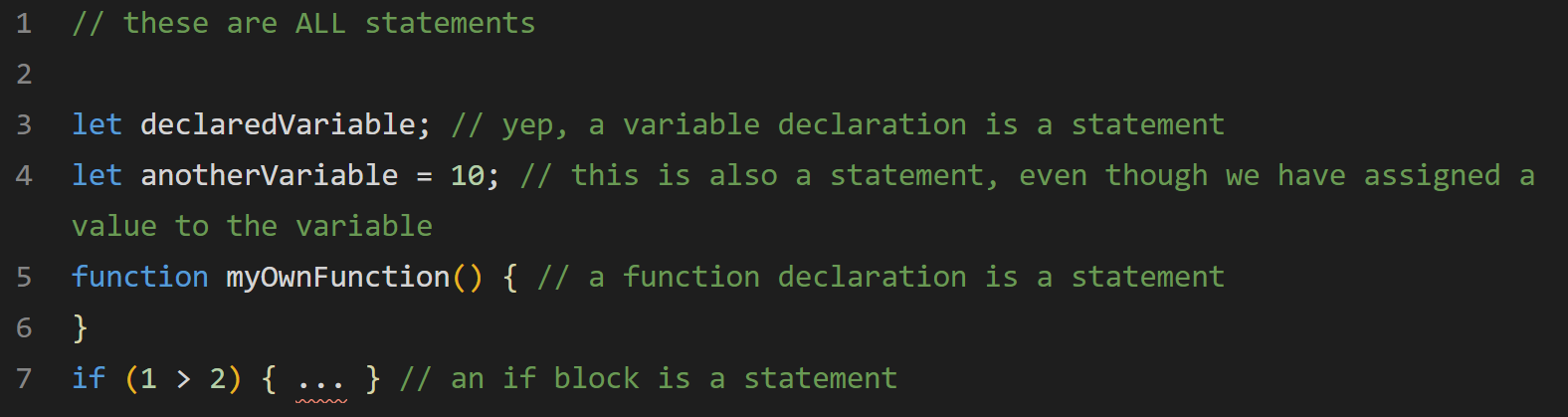


**What is a statement in JavaScript?**

As we’ve mentioned before, a JavaScript program consists of statements.

These statements are executed one by one, in the same order as they are written. And here’s the important point, **statements can contain other things like expressions, keywords, operators and more.**

Want examples? Sure.

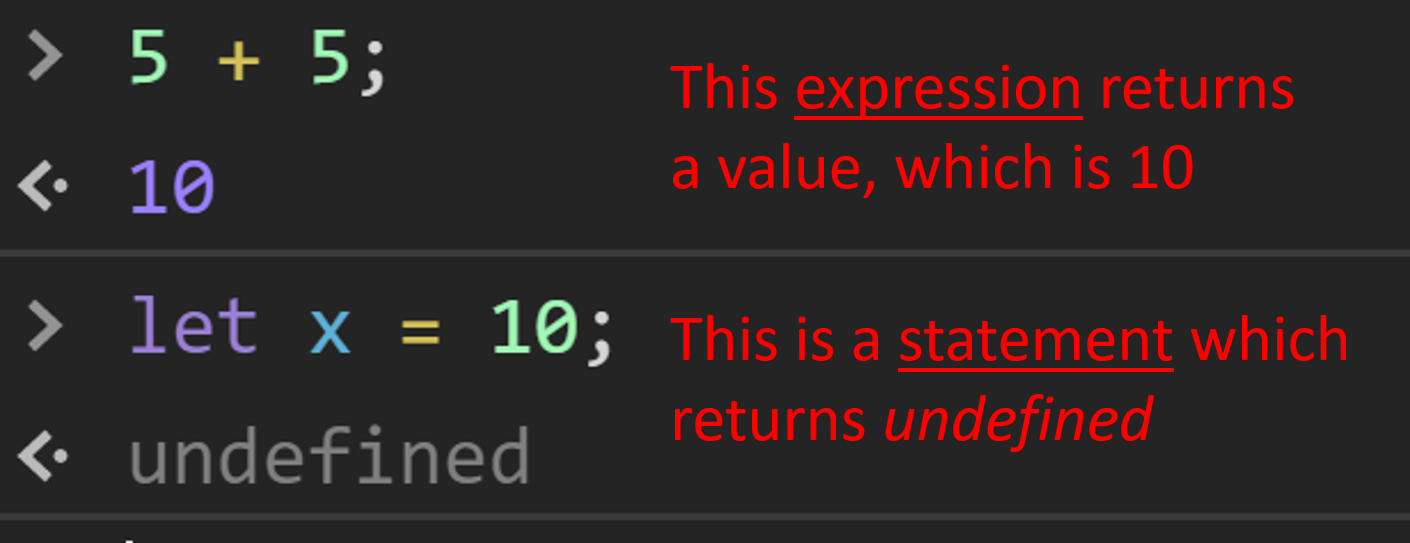


Remember what I said above?

**Expressions always return a value.**

**Statements always return undefined.**

This is why, for example, if you type 5 + 5, you get the value returned to you. But if you type let x = 10, then we get undefined.



But things can get more complicated. Sometimes expressions can also be statements. Lets take this expression:

10;

This is a statement.

It’s useless.

But as you can see it consists of one expression. I know, 10 is already a value, but for the language interpreter it's an expression that resolves to the value 10. Another expression that resolves to the value 10 is 5 + 5, for example. In case you were wondering, statements like 10; are called expression statements.

**Conclusion**

At this point in your coding career, it may seem pointless to distinguish between statements and expressions.

You may be right.

But I wanted you to at least be aware of it because trust me, sometimes you’ll be scratching your head for hours trying to understand why the console is giving you a certain result. And at least now you have another feather in your cap when trying to dig through various solutions.

I can’t wait for the next section 😊