**Basic JavaScript**

JavaScript is a scripting language you can use to make web pages interactive. It is one of the core technologies of the web, along with HTML and CSS, and is supported by all modern browsers.

In this course, you'll learn fundamental programming concepts in JavaScript. You'll start with basic data structures like numbers and strings. Then you'll learn to work with arrays, objects, functions, loops, if/else statements, and more.

# Explore Differences Between the var and let Keywords

One of the biggest problems with declaring variables with the var keyword is that you can easily overwrite variable declarations:

var camper = "James";

var camper = "David";

console.log(camper);

In the code above, the camper variable is originally declared as James, and is then overridden to be David. The console then displays the string David.

In a small application, you might not run into this type of problem. But as your codebase becomes larger, you might accidentally overwrite a variable that you did not intend to. Because this behavior does not throw an error, searching for and fixing bugs becomes more difficult.

A keyword called let was introduced in ES6, a major update to JavaScript, to solve this potential issue with the var keyword. You'll learn about other ES6 features in later challenges.

If you replace var with let in the code above, it results in an error:

let camper = "James";

let camper = "David";

The error can be seen in your browser console.

So unlike var, when you use let, a variable with the same name can only be declared once.

Update the code so it only uses the let keyword.

# Declare a Read-Only Variable with the const Keyword

The keyword let is not the only new way to declare variables. In ES6, you can also declare variables using the const keyword.

const has all the awesome features that let has, with the added bonus that variables declared using const are read-only. They are a constant value, which means that once a variable is assigned with const, it cannot be reassigned:

const FAV\_PET = "Cats";

FAV\_PET = "Dogs";

The console will display an error due to reassigning the value of FAV\_PET.

You should always name variables you don't want to reassign using the const keyword. This helps when you accidentally attempt to reassign a variable that is meant to stay constant.

**Note:** It is common for developers to use uppercase variable identifiers for immutable values and lowercase or camelCase for mutable values (objects and arrays). You will learn more about objects, arrays, and immutable and mutable values in later challenges. Also in later challenges, you will see examples of uppercase, lowercase, or camelCase variable identifiers.