Potential Issues

While we're on the topic of syntax, I'd like to talk to you about some of the issues that you might run across while you're trying to work with JavaScript for the first time. JavaScript is case sensitive. It also has comments and reserved words that you should be aware of. Knowing about these in advance will help you avoid some of the common problems that people run into when they're experimenting with a new language.

Case Sensitivity

Case sensitivity means that the names of the variables that you choose have to be consistent in terms of upper case and lower case letters. JavaScript pays attention to the exact spelling of variable names, including the case. So, a variable whose name is all in lower case is not the same variable as a variable whose name has an upper case letter.

Imagine that we've created a variable called `example`. And given it the value "coffee". And we've created another variable called `Example` with a capital "E". And given it the value "tea". Now, if we `console.log` each of these variables, we will see that these two variables contain different values, even though they're basically the same word.

This kind of variable naming is not very good practice when you're writing JavaScript. You want your variable names to represent what the content is actually going to be. So, programmers working with your code will understand what they mean, but this is a demonstration that capital letters and lower case letters are not interchangeable in variable names.

Comments

Being considerate about how people read and understand your code is very important. You're writing code so that it can be read and interpreted by a computer. But eventually, you're may need to go and edit your code yourself or you may want to share it with somebody, and you want them to be able to understand what you're writing about.

To help facilitate this, JavaScript lets you write **comments** in your code that aren't interpreted by the computer, but that can be read by somebody visually reading your code. A line in JavaScript that starts with two slashes is completely ignored by the JavaScript compiler, and this is a good way to make notes in your code.

```
// a comment
```

JavaScript will also let you create multi-line comments by typing a `/*`, and then as much text as you want on as many lines as you want and then ending them with a `*/`. Everything between `/*` and `*/` will be ignored by the compiler, just like single lines that begin with `//`.

```
/*
a comment
and here
*/
```

It's up to you how you choose to use comments in your code, but it's important to be consistent about where you put your comments, and how you expect them to be interpreted.

Variable Naming Rules

There are a number of other rules about how you name your variables in JavaScript, and these can trip you up when you're starting out, because you might choose to name your variable something that is not actually legally permitted by the JavaScript compiler.

- Variables may only contain letters, numbers, the dollar sign symbol, or the underscore. Any other character cannot be used in the name of a variable.
- Variables must also not begin with a number.
- Variables must not use words that have been reserved as JavaScript keywords.

```
var 2nd = 2;
// this name is invalid because it begins with a number

var this = "thing";
// this name is invalid because this is a reserved word
```

Reserved JavaScript Keywords

There are a few places online where you can look for complete lists of reserved keywords. This list may change as new versions of JavaScript are released.