

Variables

Creating a Variable

Var Keyword

To create a variable, we usually use the `var` keyword and then give that variable a name. The `var` keyword may look a bit more self-explanatory for creating a variable since it is basically the shortened version of the word *variable*. However, there are other ways to create variables which are even considered as the best practice nowadays.

Let

The first alternative is by using the `let` keyword. The `let` keyword works almost the same as the `var` keyword, however there are some crucial differences. When using the `let` keyword to declare variables, **once a variable has been declared, it cannot be re-declared again.**

```
// Valid
var myName = "John Doe"
var myName = "Jane Doe"

// Invalid
let yourName = "John Doe"
let yourName = "Jane Doe"

// Valid
let myAge = "25"
myAge = "29"
```

Note that in the example above, the variable `yourName` is declared using the `let` keyword. `yourName` was initially declared and defined with the value of "John Doe", however it is re-declared again in the line just below it, this will cause an error. Now if we take a look at the `myAge` variable, it was initially declared and defined with the value of "25", and then it was re-defined with the value of "29". This will not cause an error since `myAge` was not re-declared, rather only re-defined.

Const

Another way of creating a variable is by using the `const` keyword. As the name suggests, `const` is a way of creating a **constant variable**, meaning it is never-changing. Just like a `let` variable, a `const` variable cannot be re-declared, but there are extra limitations to a `const` variable, it also cannot be re-defined.

```
// Valid
let city = "London"
city = "Paris"

// Invalid
const country = "Italy"
const country = "Sweden"

// Invalid
const island = "Java"
island = "Borneo"
```

Take a look at the example above, `country` was declared using the `const` keyword and defined with the value of "Italy". It was then re-declared and re-defined with the value of "Sweden", this will cause an error since a `const` variable cannot be re-declared nor re-defined. The `island` variable will also cause an error because it was re-defined from having the value of "Java" into having the value of "Borneo". Although it was not re-declared, it is still invalid because it was re-defined, and we now already know that a `const` variable cannot be re-defined.

Here are some references if you would like to explore more about this topic:

- https://www.w3schools.com/js/js_variables.asp
- https://www.w3schools.com/js/js_const.asp
- https://www.w3schools.com/js/js_let.asp
- https://www.tutorialspoint.com/es6/es6_variables.htm