

# Variables in JavaScript

---

JavaScript is a dynamically typed language, and is loosely typed which means there little to no need to define script types to its variables.

## The `var` keyword

In Js, there are numerous ways to define variables, one of which is `var`.

The `var` keyword is used to define a variable.

An uninitialized var variable is undefined.

### Example

```
var x = 10;  
console.log(x);
```

Output:

10

```
var a = "Hello World";
```

The above code will create a variable called `a` and assign the value `Hello World` to it.

### Variable Scope

Variables defined with the `var` keyword are created in the global scope.

```
var a = "Hello World";  
var b = "Hello World";
```

The above code will create two variables called `a` and `b` and assign the value `Hello World` to them.

```
console.log(a);  
console.log(b);
```

The above code will print the value of `a` and `b` in the console.

Now the thing with var is that it creates a variable in the global scope.

### Quick Quiz

```
var a;
```

- What is the value of `a`?

```
undefined
```

## The `let` keyword

The `let` keyword is used to define a variable. The difference between `var` and `let` is that `let` creates a variable in the scope of the block it is defined in.

A variable defined using the `let` keyword is created in the block it is defined in and is not available outside of that block.

### Example

```
let x = 10;
console.log(x);

function myFunction() {
  console.log(x);
  let x = 20;
  console.log(x);
}

myFunction();
```

Output:

10

Uncaught ReferenceError: Cannot access 'x' before initialization  
at myFunction (REPL8:2:15)

20

- Note that the above code will throw an error because `x` is not defined in the global scope. It is only defined in the `myFunction` function.
- Functions are not hoisted, so `myFunction` will not be defined until the end of the code.
- Functions as a concept will be discussed later, for now consider them as a block of code that can be executed on being called.

## The `const` keyword

The `const` keyword is used to define a constant variable. A constant variable is a variable that cannot be reassigned.

A constant variable is created in the global scope, and is available everywhere.

### Example

```
const x = 10;
console.log(x);
function myFunction() {
  console.log(x);
  const x = 20;
  console.log(x);
}
myFunction();
```

Output:

10

10

20

- Here `x` is defined as a constant variable in the global scope, hence it can be accessed anywhere even in the `myFunction` function.

## The `let` keyword vs `const` keyword

The `let` keyword is used to define a variable. The difference between `var` and `let` is that `let` creates a variable in the scope of the block it is defined in, and `const` creates a constant variable in the global scope.