Javascript Core

1. var , let and const

var, let and const are different ways of defining variables in Javascript. let and const were introduced in the ES6 version of Javascript. Traditionally, only var was the way of defining variables.

The problem in var was that it was functional scope. So anytime we wanted to define a new variable inside a block scope (for loops, if-else, while), we could run into the chance of overriding any existing variable with the same name. This was because of the functional scope nature of var.

For example :

```
function func() {
    var i = 2;
    if (i === 2) {
        var i = 3;
        console.log(i); // Prints 3
    }
    console.log(i); // Prints 3. 2 has been replaced by 3 here
}
```

To tackle this issue, let was introduced which is block scoped unlike functionally scoped var. const also behaves as let with the difference that const variables cannot be muted or reinitialised.

PS: A function is also a block. Global is also a block.

All the three variable definitions are hoisted on the top of their respective scope. The difference comes in the initialisation and accessing of the variables.

var is hoisted and initialised to undefined if not explicitly initialised, and it can be accessed by the program. It is initialised to undefined.

let is hoisted and not initialised in the memory unless explicitly initialised. Therefore, it cannot be accessed by the program before its initialisation. The area between the defining of let and the initialisation of the let is called TDZ (Temporal Dead Zone)

const deals with immutability. It is also not possible to leave a const uninitialised.

That means,

```
const x;
console.log(x); // Throws an error
```

PS: Although const deals with immutability and non re-initialisation, yet it is possible to mutate an array or object because non primitive data types like arrays and objects are passed by reference in the memory and not passed by value.

For example -:

```
const arr = [1,3,4,5,6]
arr.push(7);
console.log(arr); // [1,3,4,5,6,7]
```

==========Page End==========