Date: / /

How Var, let and const keywords wooks in Javascript

Earlier, pre-ES6 era, only var keyword was introduced for declaration of variable

With ES6, the let and const keyword introduced.

How to declare Variables in JavaScript

```
// without keywords. It is same as var
   and not allowed in 'strict' mode.

name = 'Jack';

// using var
var price = 100;

// using let
let isPermanent = False;

// using const
const PUBLICATION = 'Jack';
```

We'll disuss
- Scope
- Reassigning New Value
- When you access a variable before declaring it.

Variable Scope in JavaScript

The variable may exist in a block, inside function or outside function.

A block is section of code inside { }

Eg →
```
{
    let name = 'deepa';
}
```

* It has block Scope

A function is bunch of code you want to place logicall together.
It is declared using Function keyword

```
function test() {
    let name = 'deep';
}
```

* It has function scope
* Everything declared outside block and function is global Scope

So there are three types of Scope

* Block Scope
* Function Scope
* Global Scope

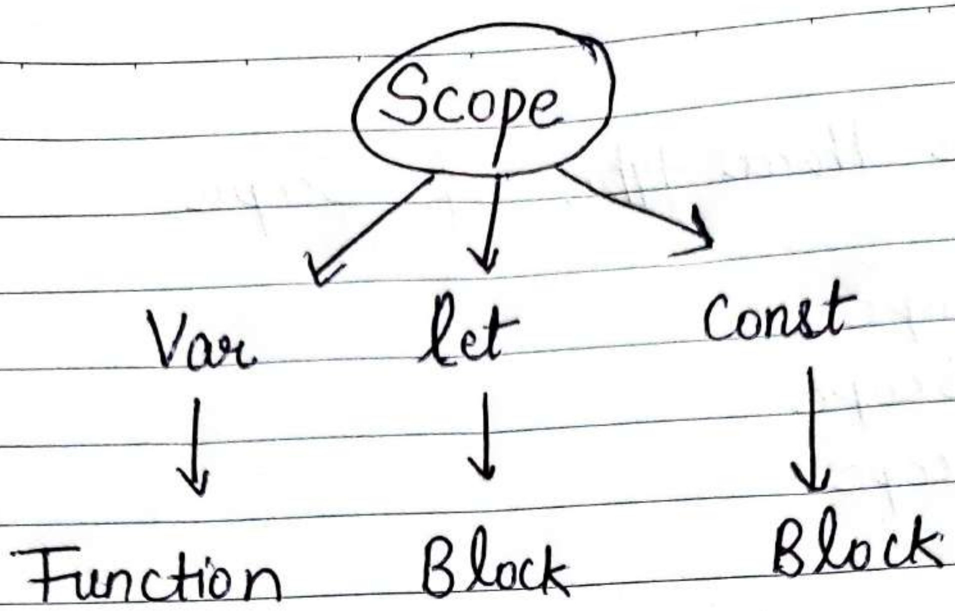
The three keyword var, let and const work around these scopes.

How to Use Javascript Variable in Global Scope

We can use var, let and const to declare global variable.
But it is recommended not to do it.
By doing this, variable are accessible everywhere.

So to restrict scope of variable using var, let and const keywords, here's order of accessiblity in scope starting with lowest:

- var : The functional Scope level
- let : The block Scope level
- const : The block Scope level



How to Reassign a New value to Variable in Javascript

You can reassign var or let variables, but you cannot reassign a new value to const variable

Const — (Constant) - ~~Always same. cannot change~~

One Tricky part
When object is declared and assigned value with const, you CAN STILL CHANGE VALUE OF ITS PROPERTIES

But you cannot reassign any object value to same variable

Date: ____

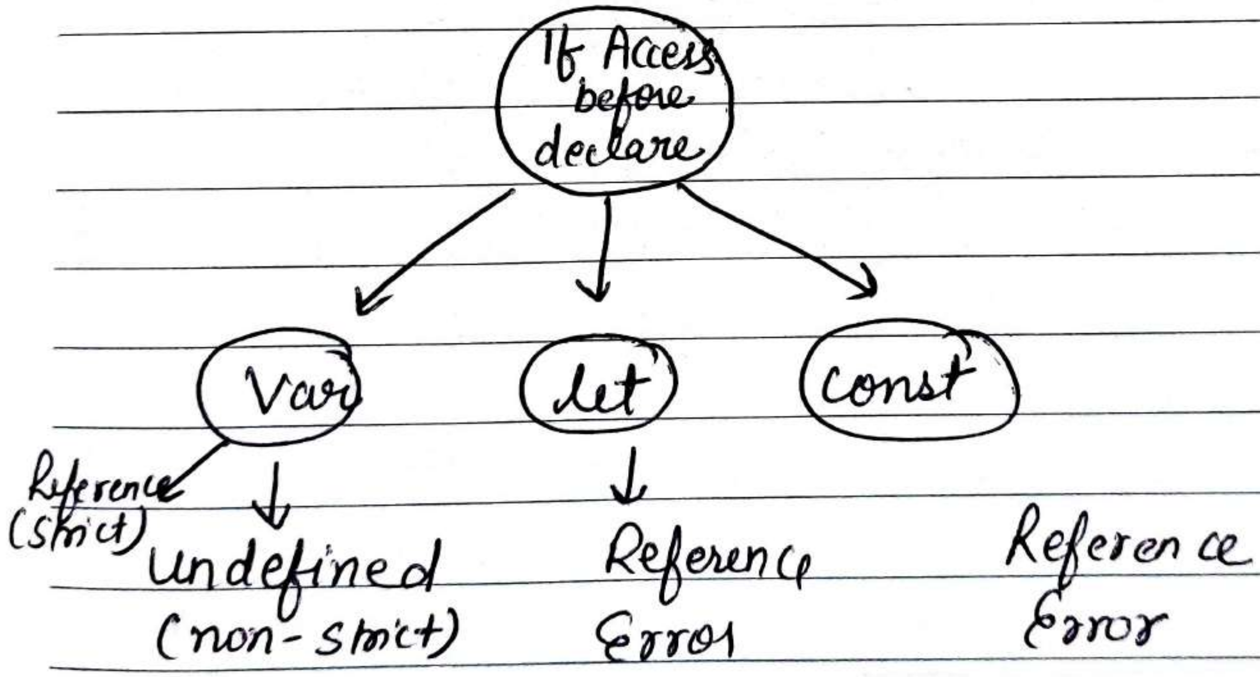
When You Access a Variable before declaring

With var in non-strict mode, the variable will have an undefined value.

This means variable declared but not assigned

In strict mode, you will get Reference Error that variable is not declared.

With let and const, you will always get Reference Errol



* Don't use Var
* Use let or const
* Use const more often.
* Use let, when you need to reassign