Javascript

**Todays topic hoisting**

Hoisting is nothing but pull something up .for example you are calling a variable or function before declaring it ,javascript pulls the variable or function to the top of your code and execute it . this is known as hoisting

Example ;

Console.log(x);

Var x=10;

Output:

It does not shows any Error it just say its undefined .Because while executing the javascript program it strats creating the call stack with the global execution context .the call stack is noting but a container .it contains the global execution context and all the functions in it .

Function 2()

Function 1()

Global Execution   
context

For var only it shows as undefined ,in case of let and const it does not create a memory ,it cant able to store the undefined value.if goes for function it creates a variable and stores the function inside the memory so the can be accessed before declaration .

Use Strict:

It is use to maintain our code clean .without use sctrict we can able to declare like this

“use stric” 🡪 this is the strating point of use strict

x=9;

Let x =9;

Console.log(x);

(use strict); 🡪 this is the ending (deactivate )point of use strict

It show an error like **Cannot access 'x' before initialization.**

By default javascript takes var for the declaration.  
  
we can’t able to delete the variable in the use strict mode .and we cant able use duplicate parameter in the function

Example :

    "use strict";

    function a(*num1*, *num1*) {

      let ans = *num1* + *num1*;

      console.log(ans);

    }

    a(2, 3);

Uncaught SyntaxError: Duplicate parameter name not allowed in this context

In strict mode we cant write the object in the read only property.keywords cannot be used as variable name in use strict mode .