[Class will start
at
7:05]

Agenda

1. Js refershe r
2. premitive and non-primitives.
3. Var , let and const
4. code. Execution / Hoisting
5. shadowing : legal and illegal

1. js is dynamic type language

```
int a = 10        ;     // statically typed lang.
String b= "abc" ;
```

2. high level language

3. V8 Engine - C++

④ real name: Ecmascript

⑤ default value is undefined.

⑥ single and double quote works in same way.

Data type

Primitive : number, String, undefined, boolean
null

: (Newly added) Symbol , bigint

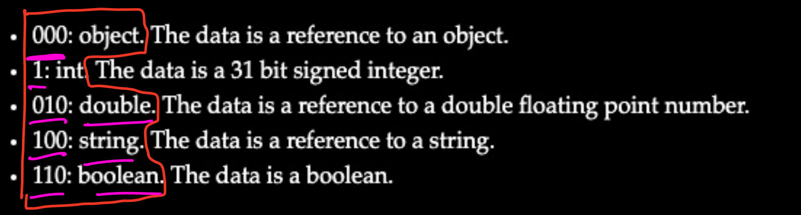
(*) can't store a character of string.
it is treated as string only.
→ In case you want to work with charalter
deal with ASCII value.

Non-primitive data type

→ array, object, function.
→ (newly added) map, set, weak map, weak set.

Array

```
Var a = [10, "abc", 10.5]
```



- 000: object. The data is a reference to an object.
- 1: int. The data is a 31 bit signed integer.
- 010: double. The data is a reference to a double floating point number.
- 100: string. The data is a reference to a string.
- 110: boolean. The data is a boolean.

value of null is 0.

Objects

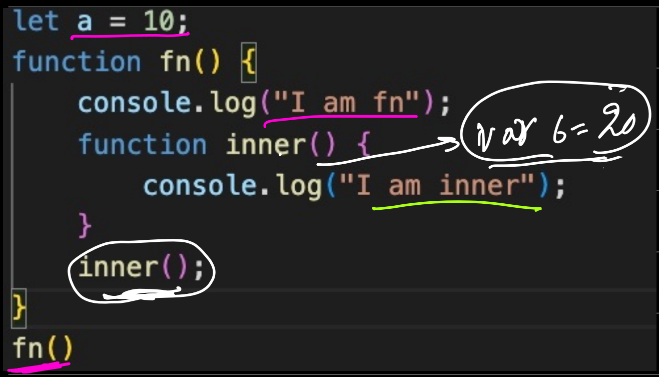
① Key - value pair

↳ Unique (number, String)

↳ (any data type of js)

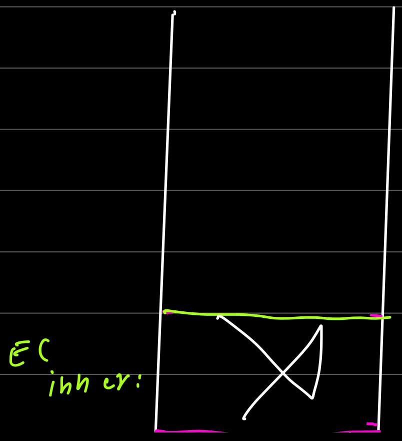
- [x] → Search for the value inside the variable
- .x → it will search for x.

code execution



```
let a = 10;
function fn() {
    console.log("I am fn");
    function inner() {
        console.log("I am inner");
    }
    inner();
}
fn()
```

var b=20

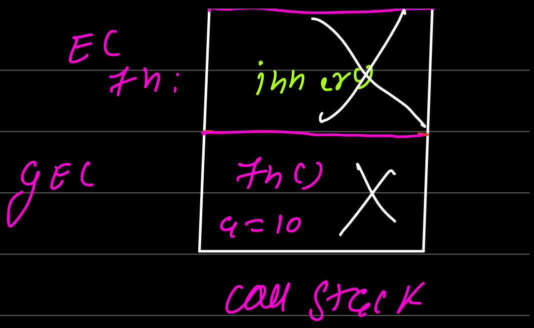


global Execution context

console.

I am fn

I am inner



code Execution

① EC creation

(i) Hoisting : memory allocation

(variable : undefine

fn : ye it's memory]

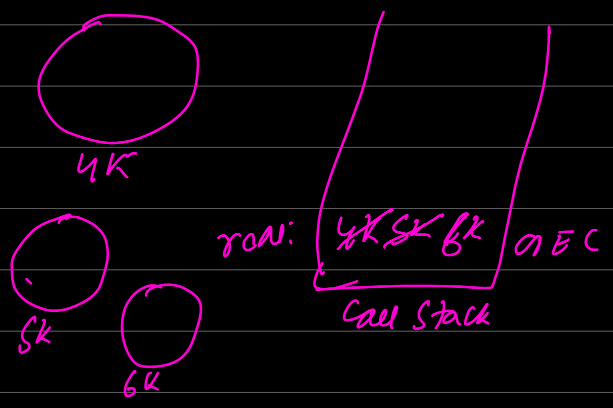
(ii) windows : browser global object.

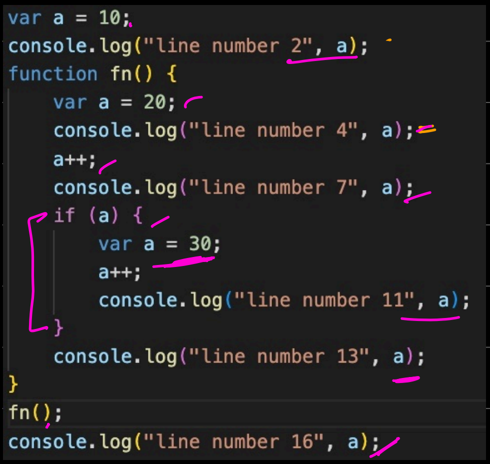
② EC code execution.

function real () : 4k

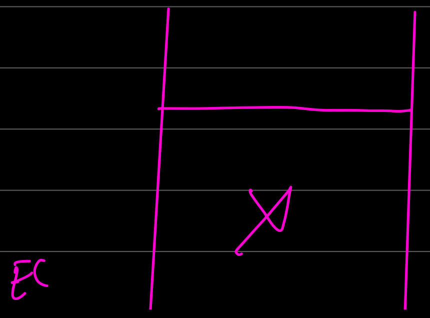
" real () : 5k

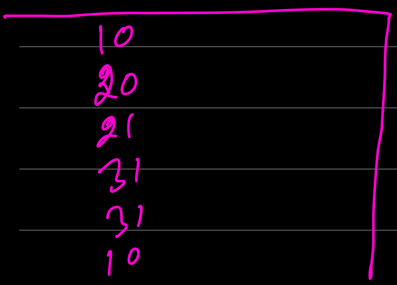
" real () : 6k

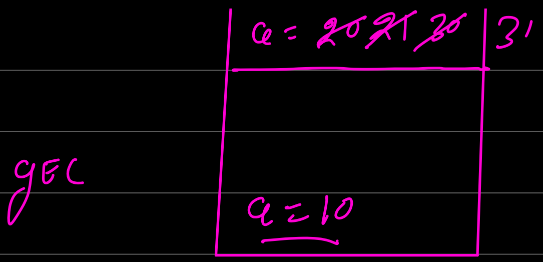


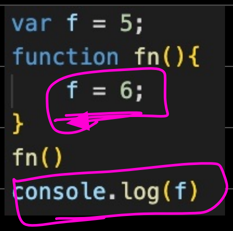


```
var a = 10;
console.log("line number 2", a);
function fn() {
    var a = 20;
    console.log("line number 4", a);
    a++;
    console.log("line number 7", a);
    if (a) {
        var a = 30;
        a++;
        console.log("line number 11", a);
    }
    console.log("line number 13", a);
}
fn();
console.log("line number 16", a);
```

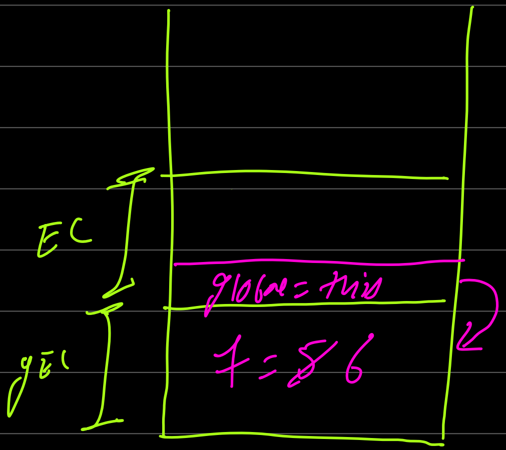




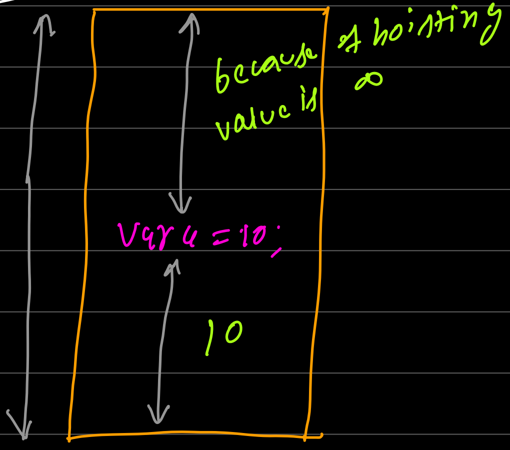


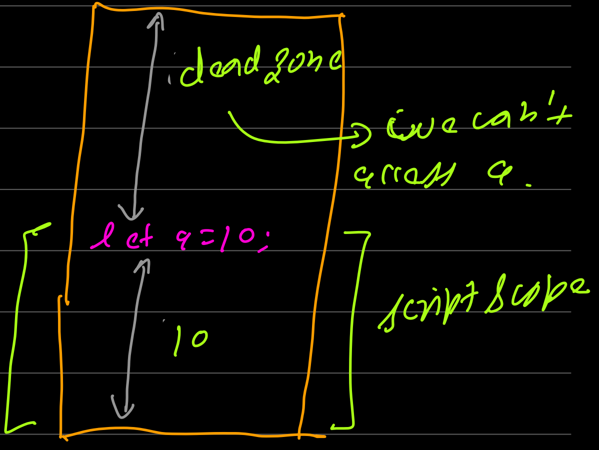


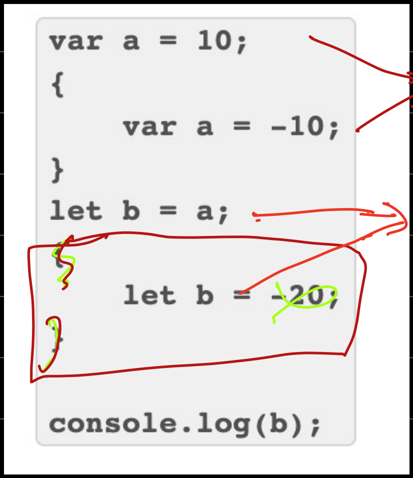
```
var f = 5;
function fn(){
    f = 6;
}
fn()
console.log(f)
```



var and let





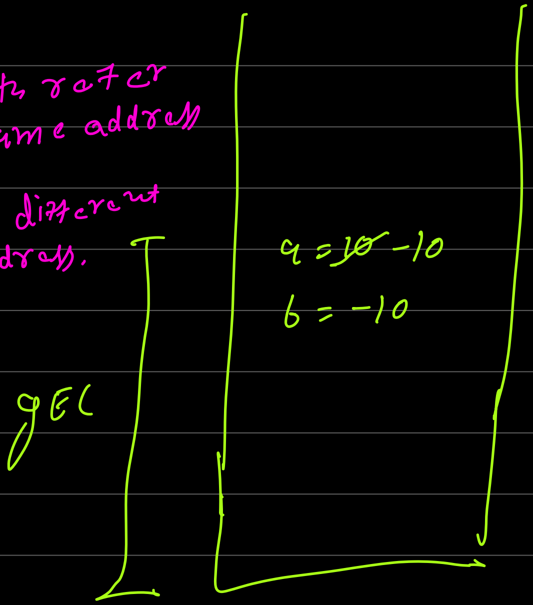


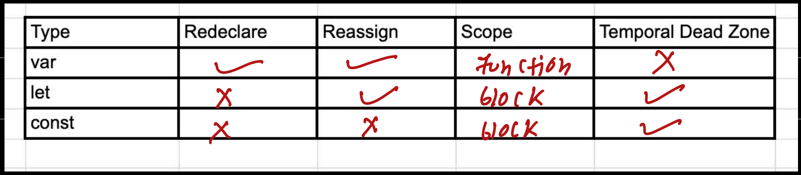
```
var a = 10;
{
    var a = -10;
}
let b = a;
{
    let b = -20;
}

console.log(b);
```

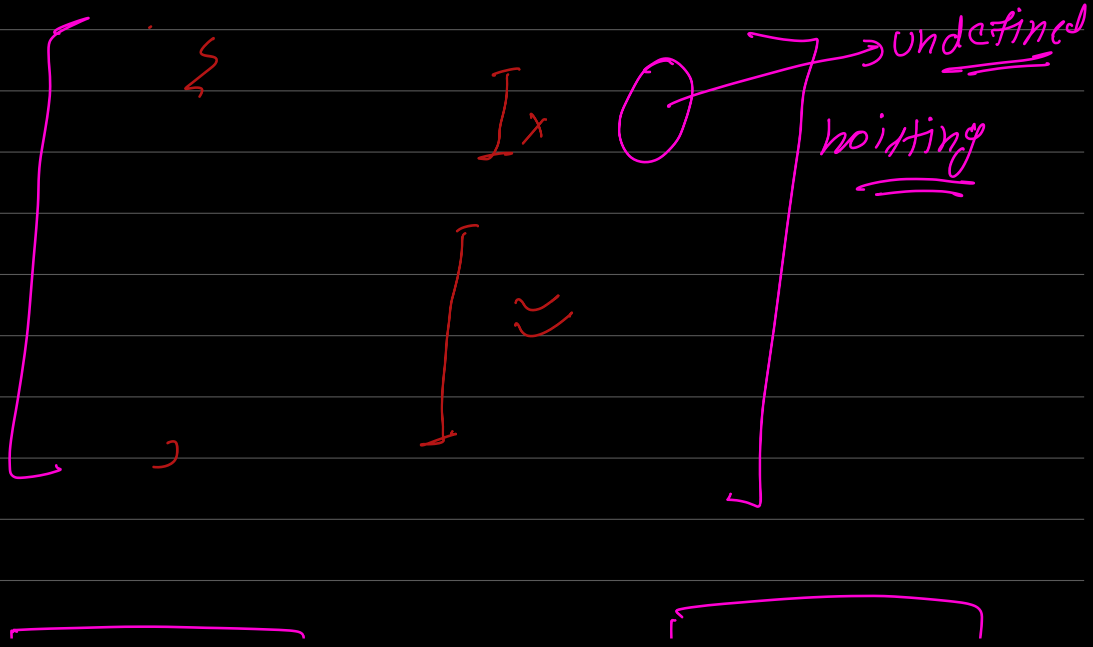
both refer same address

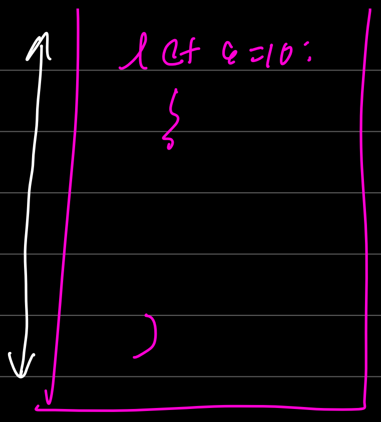
two different address.

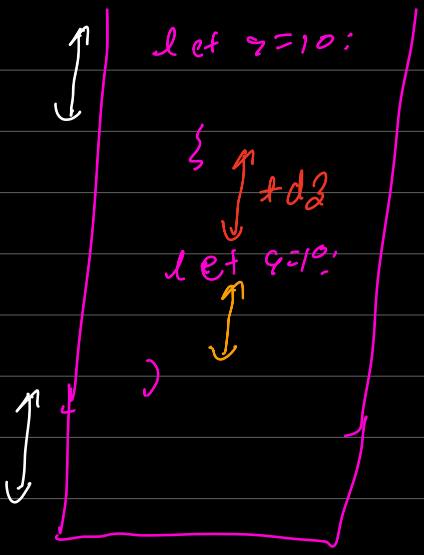




Type	Redeclare	Reassign	Scope	Temporal Dead Zone
var	✓	✓	function	✗
let	✗	✓	block	✓
const	✗	✗	block	✓







Q1 : string str : "abcDEFG"

output : "ABCdefg"

(ASCII)

$A \longrightarrow a'$

$G \longrightarrow g$

$\boxed{'Z'} - 'A' = \boxed{'z'} - 'a'$

$\boxed{'Z' = 'z' - 'a' + 'A'}$

$$\boxed{'q' = ('a' - 'A' + 's')}$$