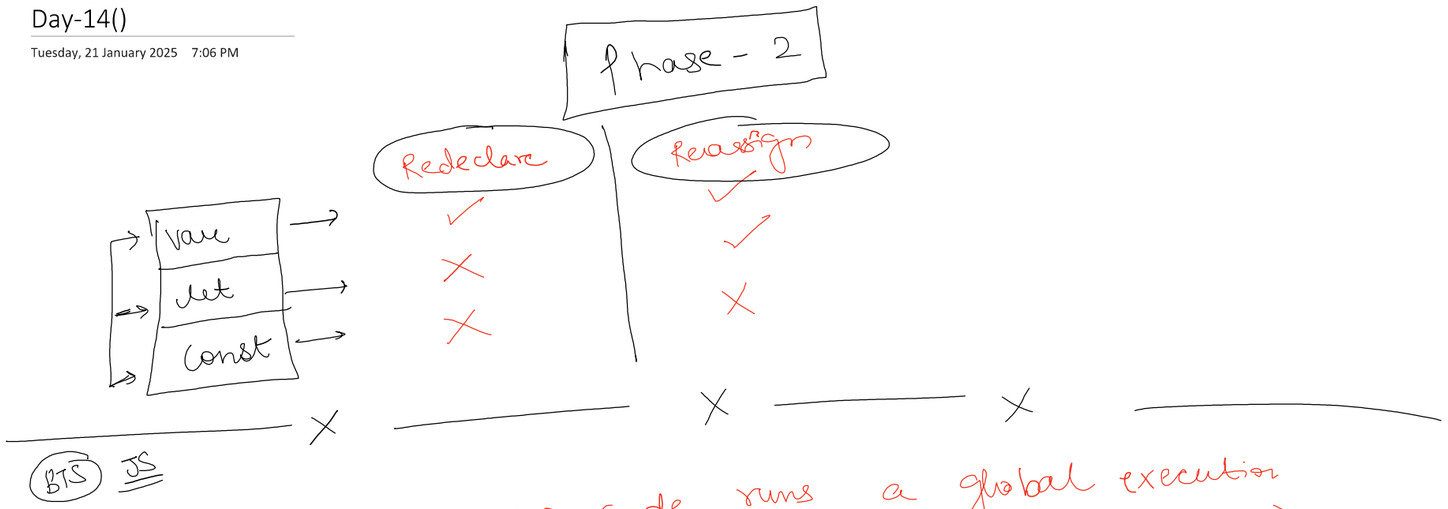
Day-14()

Tuesday, 21 January 2025 7:06 PM



1. Whenever a JS code runs a global execution context is created. (GEC)

2. Inside a GEC we have 2 phases
   - 2.1 Memory Creation phase (MCP) 1.
   - 2.2 Code Execution phase (CEP) 2.

3. Role of MCP is to allocate memory
   - 3.1 to all the variables ★
   - 3.2 to all the fn's ★

   Memory

MCP runs before CEP ★

4. CEP means running the code line by line (0 line)

after MCP is done. ★

... is called a new execution (EC)

process repeats.

MCP ✓

CEP ✓

Whenever a fn … the same … context is created &

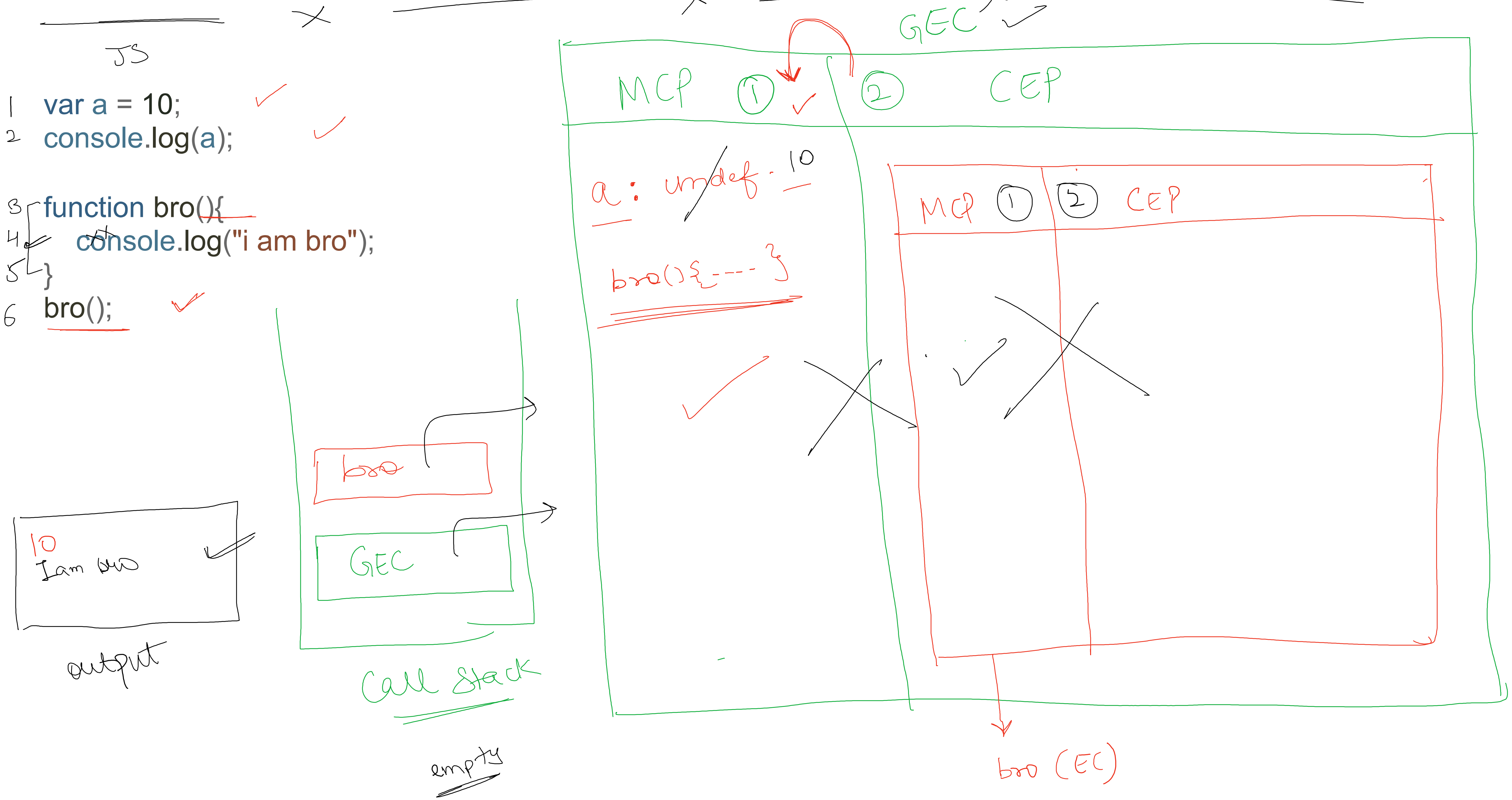
Whenever GEC is created, a global object (window) ☆ along with it is also created.

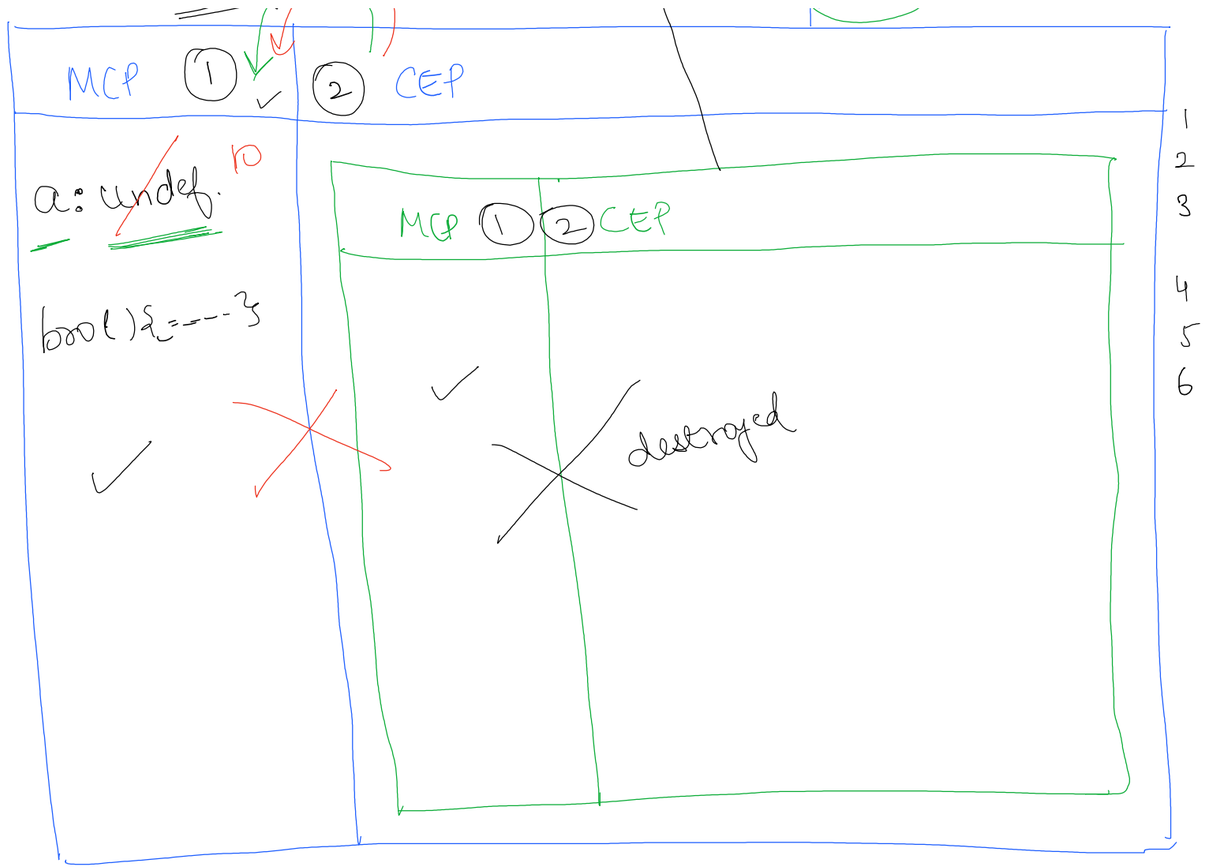
5

later on 6

JS

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

function bro(){
    console.log("i am bro");
}
bro();
```

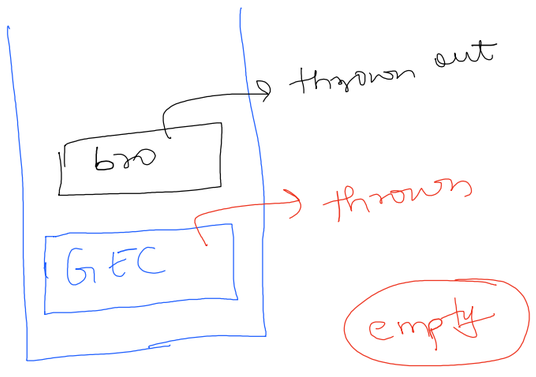


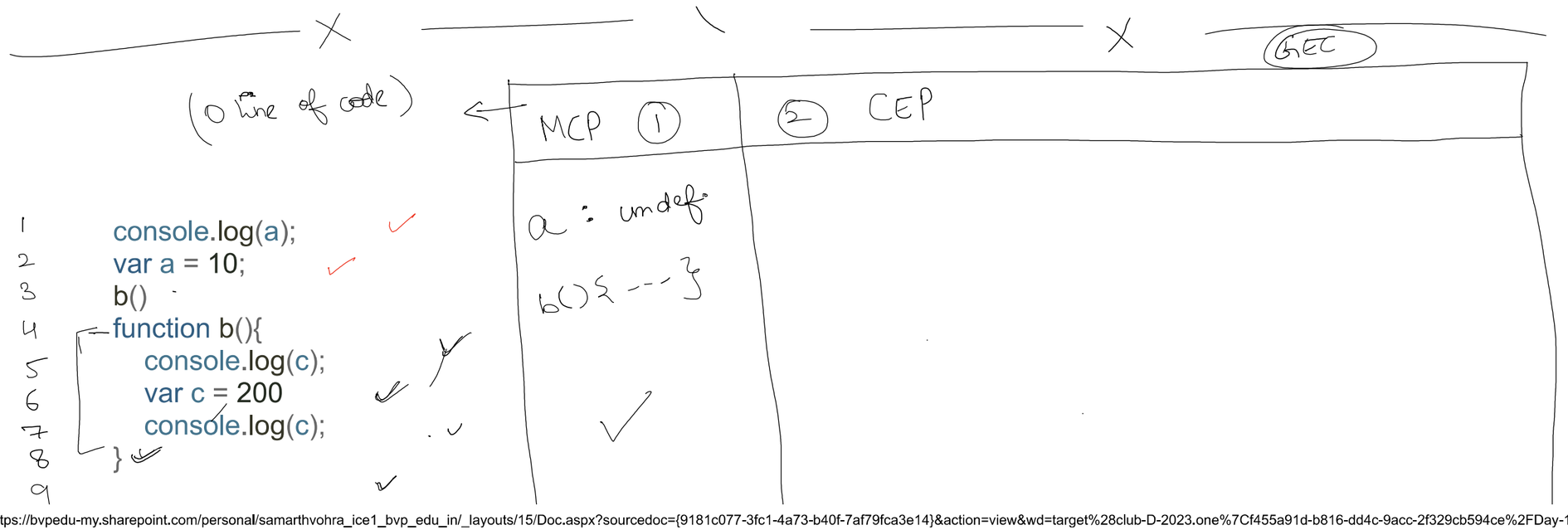


I am bro

output

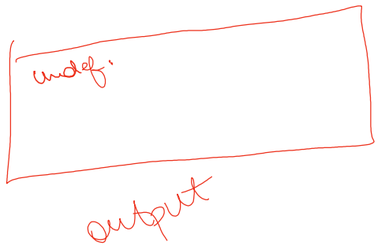
```
1  console.log(a);
2  bro();
3  var a = 10;

4  function bro(){
5      console.log("i am bro");
6  }
```

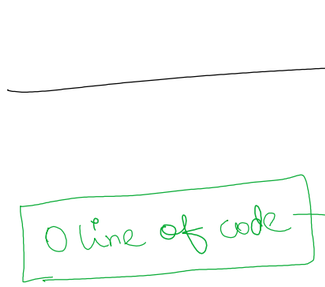




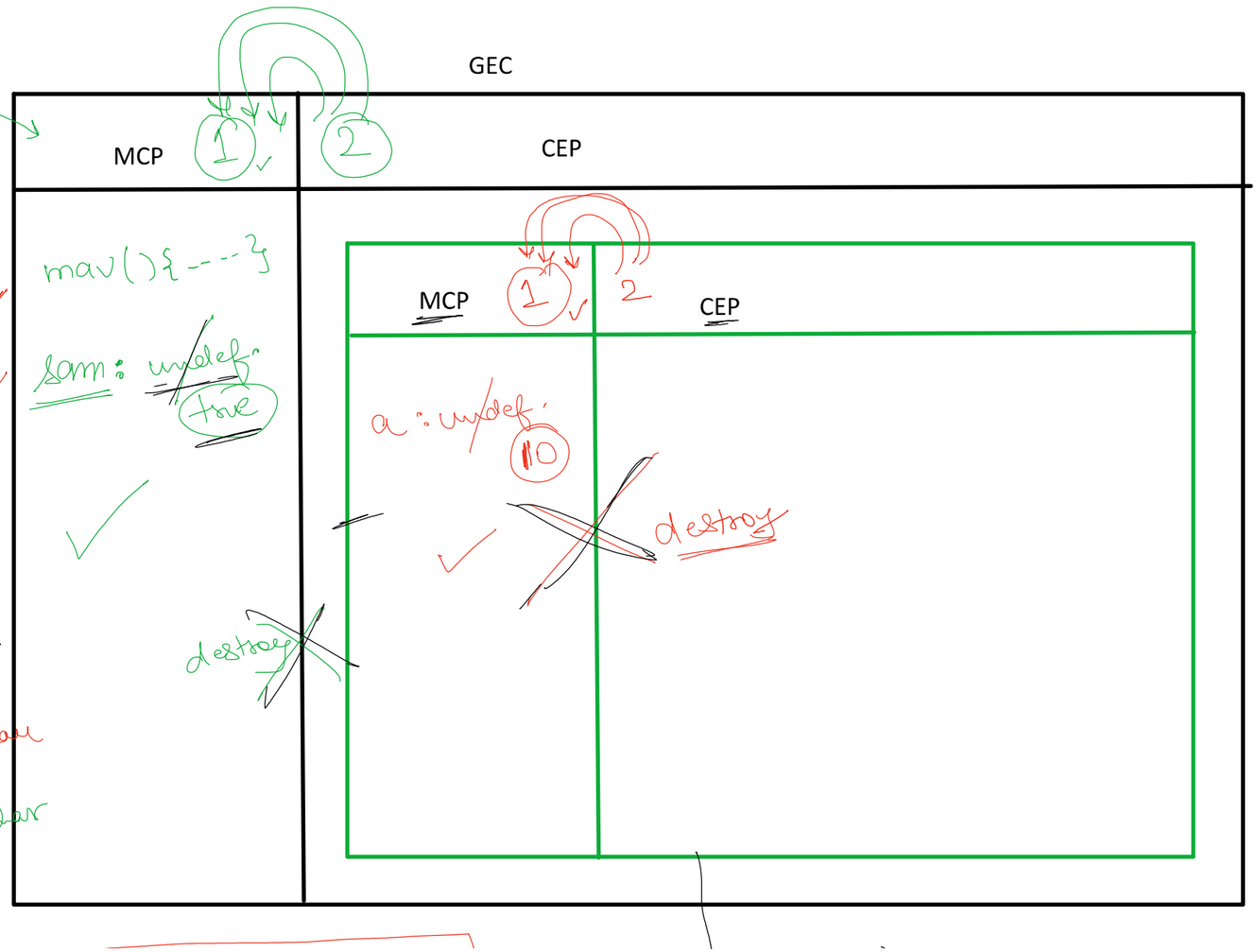

```
1      console.log(a);
2      var a = 10;
3      b()
4    function b(){
5        console.log(c);
6        var c = 200
7        console.log(c);
8    }
9
```

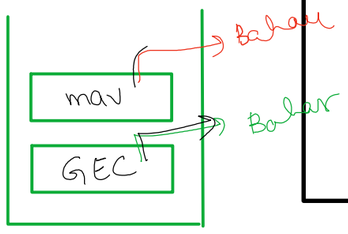
```
console.log(a);
```



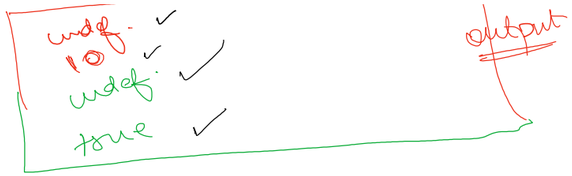



```
mav() ✓

function mav(){
    console.log(a);
    var a = 10;
    console.log(a);
}

console.log(sam);

var sam = true;

console.log(sam);
```





CALL STACK
empty

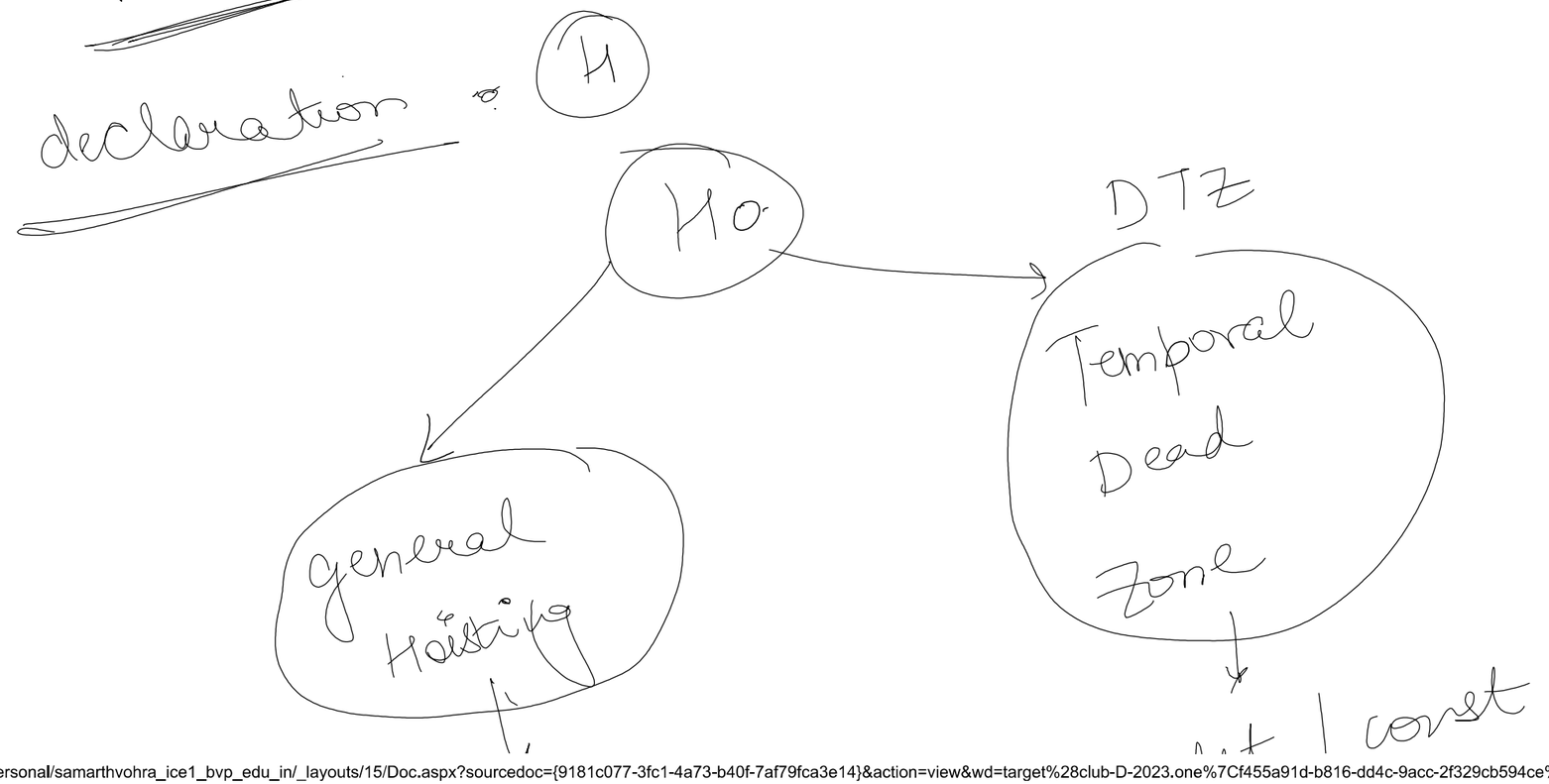


mav (FC)

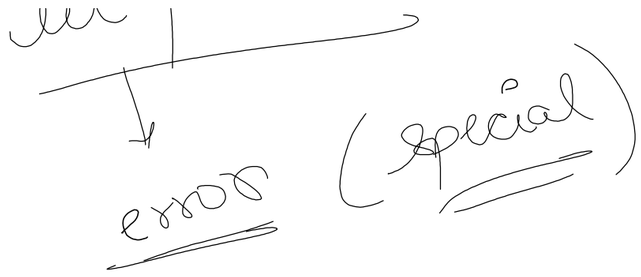
Hoisting

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

→ Whenever you are trying to access a variable or a fn even before its declaration




undefi


error
(special