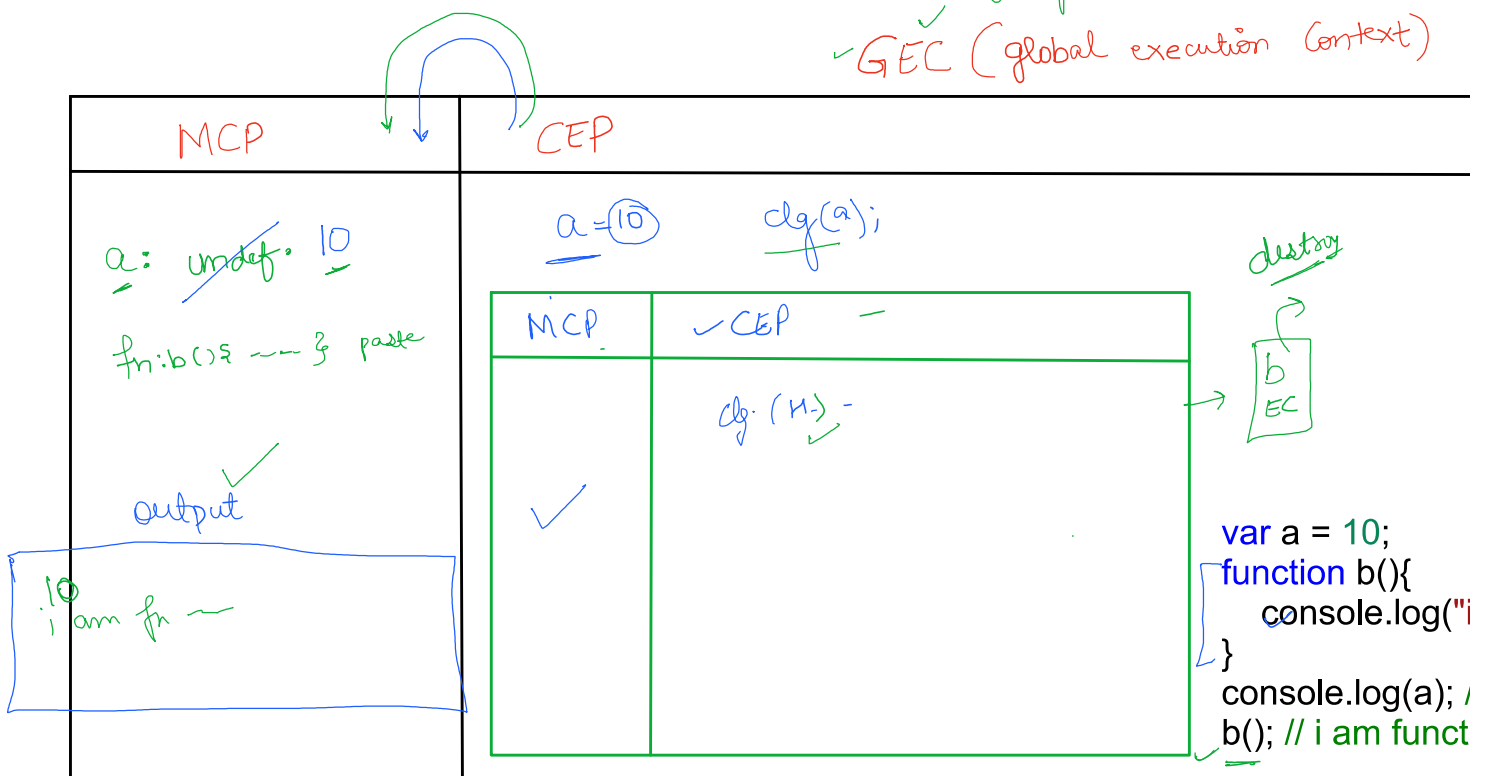


Day-13

Monday, 8 July 2024 12:10 PM

Loops?methodsString? \Rightarrow toUpperCase \checkmark Math \checkmark \rightarrow Array \rightarrow methods \times

Main interviews

 \checkmark GEC (global execution Context) ^{destroy}

Whenever a JS code is run, a GEC is created. inside that GEC, 2 phases

- ① MCP \rightarrow memory creation phase.
- ② CEP \rightarrow code execution phase.

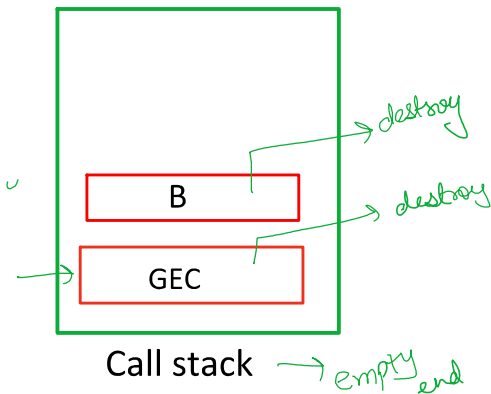
1st MCP \rightarrow ① variables (var, let, const) \rightarrow memory allocate
 \rightarrow ② fn definition unlike as it store kaise, memory all

Line by line code execute / run
 1 by 1 → CEP

1 bhi line of code ~~khahi chala~~, MCP ✓ → CE

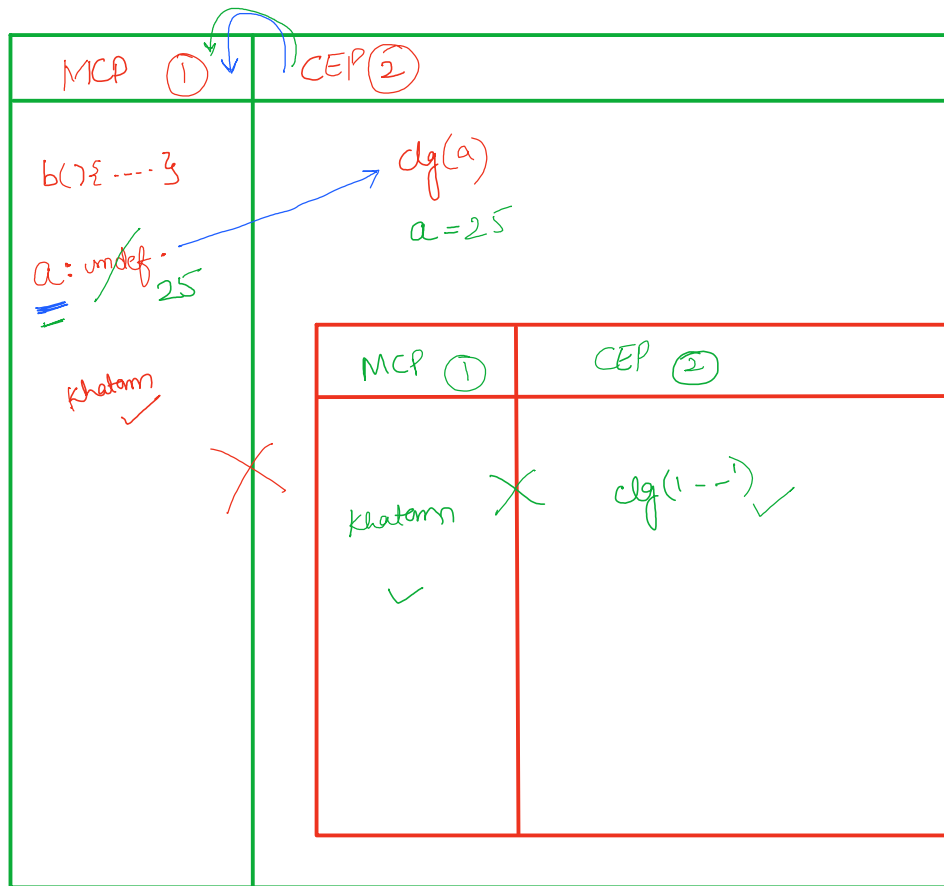
⇒ Whenever a fn is called a new EC is created
 execution → fn room
 context

```
1 console.log(a); ✓
2 b(); ✓ → fn()
3 function b(){
4   console.log("hello")
5 }
6 var a = 25; ✓
```



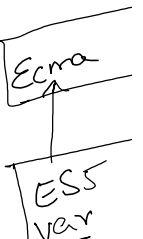
undefined ✓
 Hello ✓

output



Hoisting → accessing a variable / fn even before its declar. is called H.

3 declaratives



	var	let	const
redeclare	✓	X	X
reinitialize	✓	✓	X

R.D. ✓ { var a = 10 ;
var a = "Sam" ; }

let a = 10
let a = true X error X
a = true ✓ X

R.IV. ✓ a = true

Hoisting { var }
{ fn }

access ✓
Basd mei
declare
karna ✓

<u>var</u> <u>undef.</u>	<u>let & const</u> <u>error</u>
-----------------------------	--

✓ var ✓
let ✓
const ✓