# Java Execution Context

- JS runs your program in two phase


JS code → Global EC ↕ this

in Browser (this = Window object)

[this ke andar GEC rakh diya jata hai

JS have 3 EC (2 mainly)
- Global EC
- Function EC
- Eval EC

- JS is single threaded
- The two phases are
  - → Memory Creation Phase (allocate Mem)
  - → Execution Phase

eg
```
let val1 = 10
let val2 = 3
function addNum (Num1, Num2) {
        let total = num1 + num2
        return total
};
let result1 = addNum (val1, val2)
let result2 = addNum (10, 2)
```

① → runs thru. → Global Execution ↕ allocate at this

② Memory Phase

val1 → undefined, val2 → ud, ..
addNum → defination
result1 → ud, result2 → ud

③ Execution Phase

val1 ← 10, val2 ← 3, [func addNum → kuch nahi hoga]

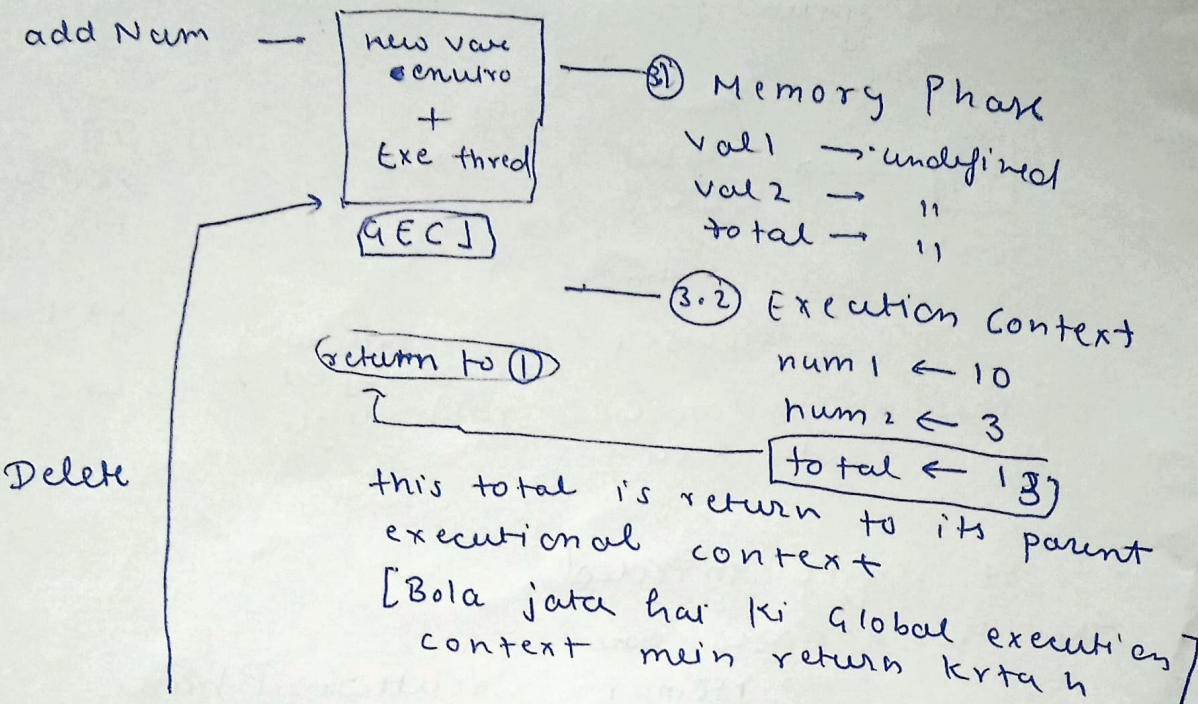
But when result1 call hoga toh

addNum → new execution ctx called → new variable Environment + Execution thread

{ jitni Baar func call honge utni baar yeh Box Create hoga }

as ③ Execution Phase

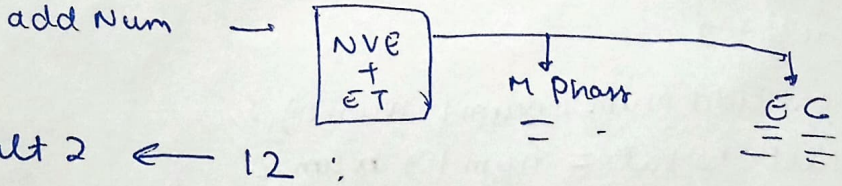
val1 ← 10 , val2 ← 3




Now GEC ⟶ Deleted

result 1 ← 13

: Now again for result 2




result 2 ← 12 ;

## Call Stack

```
let func one () {
        func two () {
                func three () {
                        3
                }
        }
}
```

in stack it works as