

Javascript Execution Context

$\{ \}$ → Global EC
this

↳ Global Execution Context

↳ Function Execution Context

↳ Eval Execution Context

`{ }` → Memory Creation Phase
→ Execution Phase

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```



`{ }` → Memory Creation Phase
→ Execution Phase

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```



→ Execution Phase

① → Global Execution
↓
this

② Memory Phase

val1 → undefined
val2 → undefined
addnum → definition

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```



① → Global Execution
↓
this

② Memory Phase

val1 → undefined
val2 → undefined
addNum → definition
result1 → undefined
result2 → undefined

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```

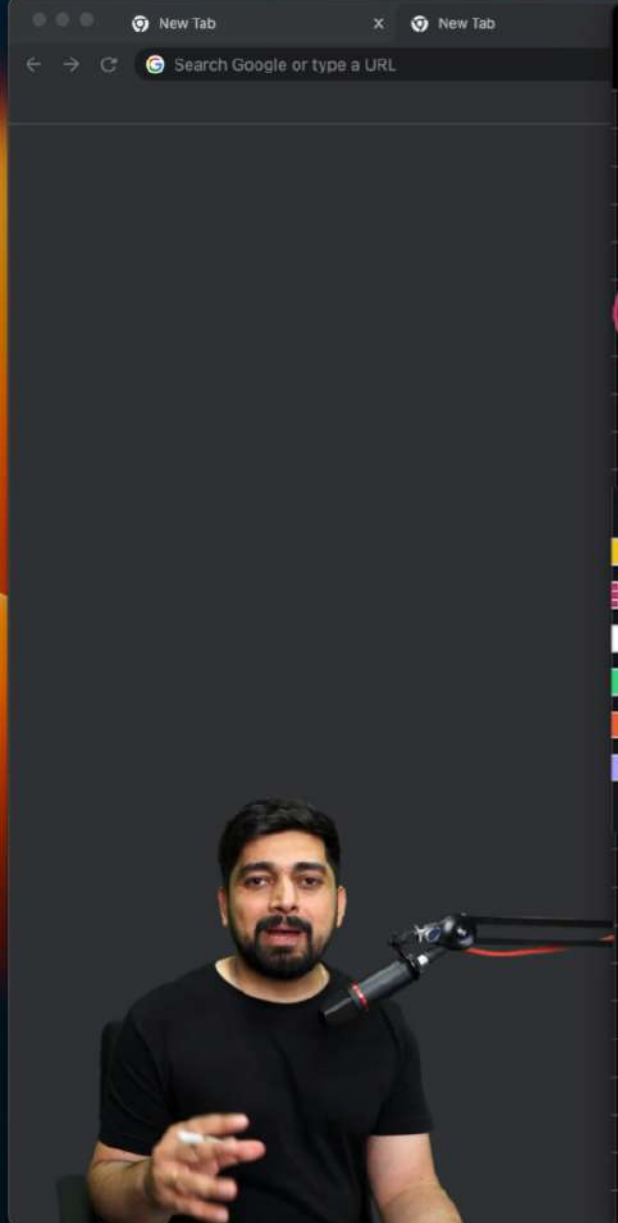
③ Execution Phase

val1 ← 10

val2 ← 5

addNum →

new variable
environment
+
Execution
thread



① → Global Execution

↓
this

② Memory Phase

val1 → undefined
val2 → undefined
addNum → definition
result1 → undefined
result2 → undefined

③ Execution Phase

val1 ← 10
val2 ← 5
addNum →

new variable environment
+
Execution thread

Memory Phase

val1 → undefined
val2 → undefined
total → undefined

Execution Context

num1 → 10
num2 → 5
total → 15

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```

```
1 let val1 = 10
2 let val2 = 5
3 function addNum(num1, num2){
4   let total = num1 + num2
5   return total
6 }
7 let result1 = addNum(val1, val2)
8 let result2 = addNum(10, 2)
```

① → Global Execution

↓
this

② Memory Phase

val1 → undefined
val2 → undefined
addNum → definition
result1 → undefined
result2 → undefined

③ Execution Phase

val1 ← 10
val2 ← 5
addNum →



Memory Phase

val1 → undefined
val2 → undefined
total → undefined

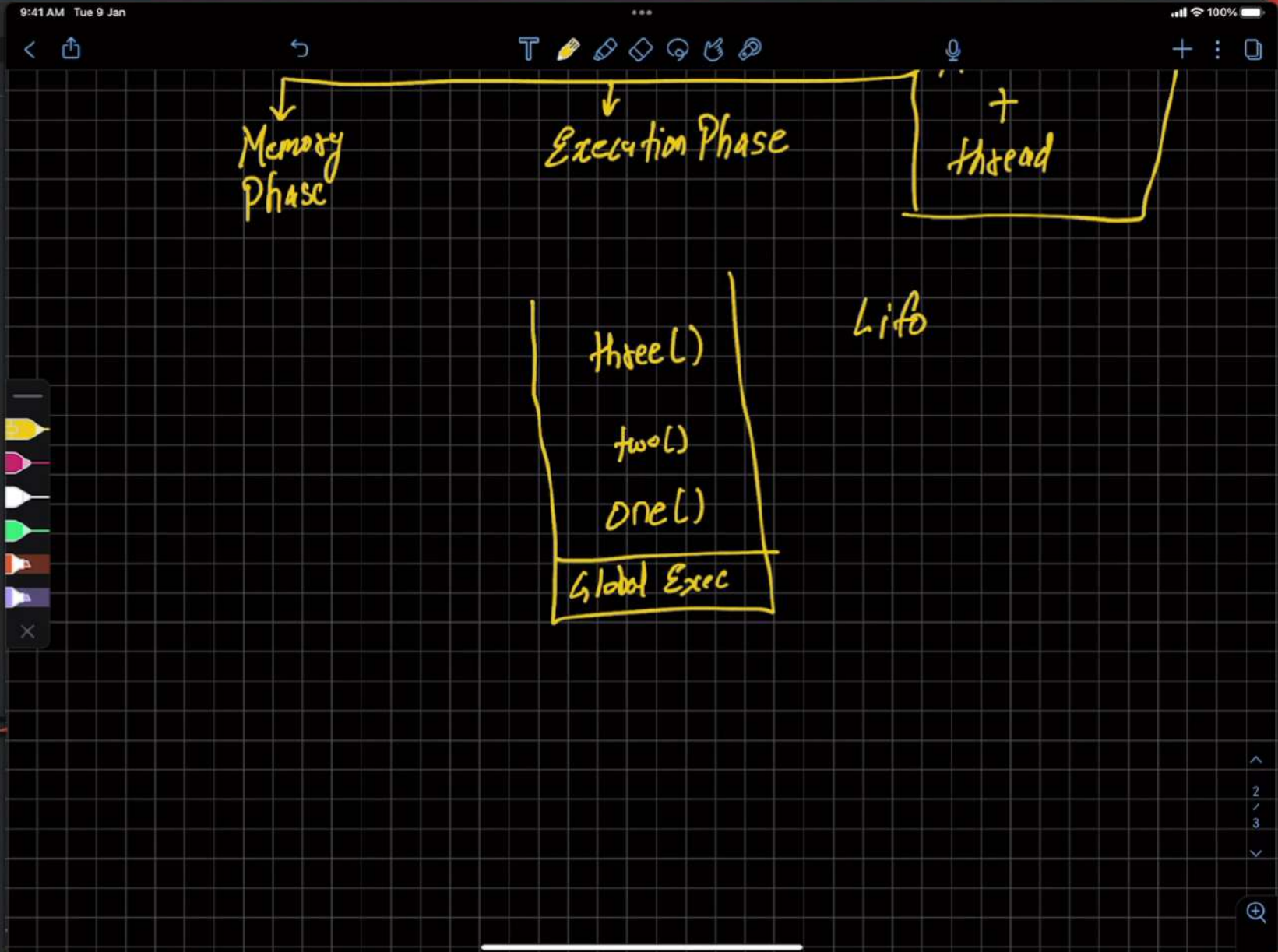
Execution Context

num1 → 10
num2 → 5
total → 15



New Tab X New Tab

Search Google or type a URL





IMG_20220730_105008.jpg



68%

