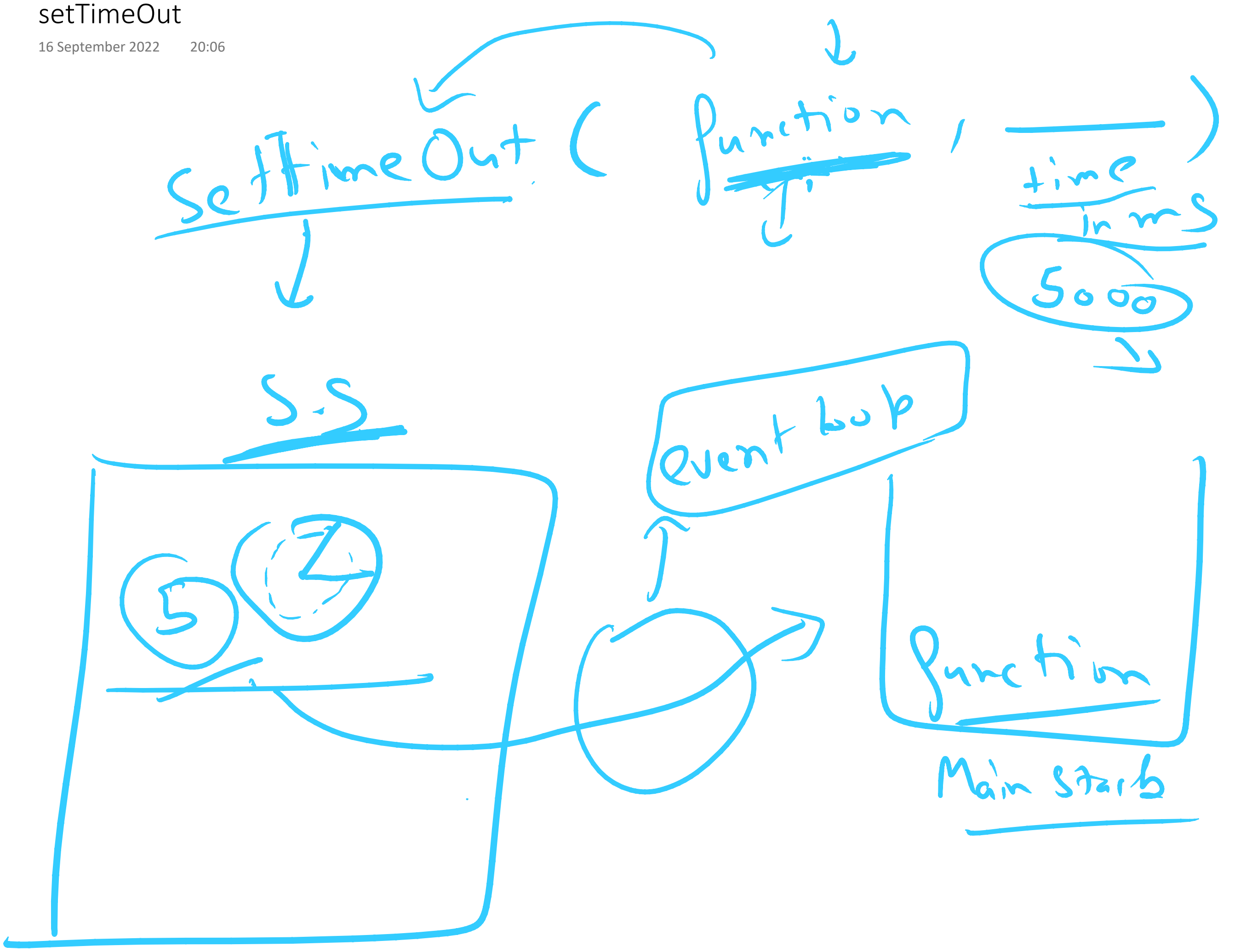
setTimeOut

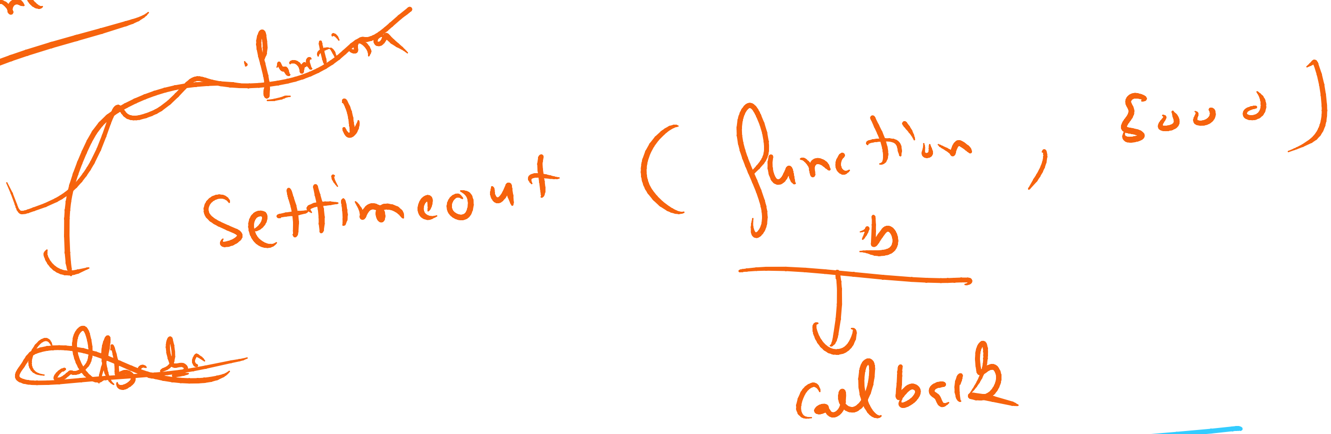
16 September 2022 20:06



got passed

Callback is a function which ~~takes function~~ as an argument and returns anything

to another function



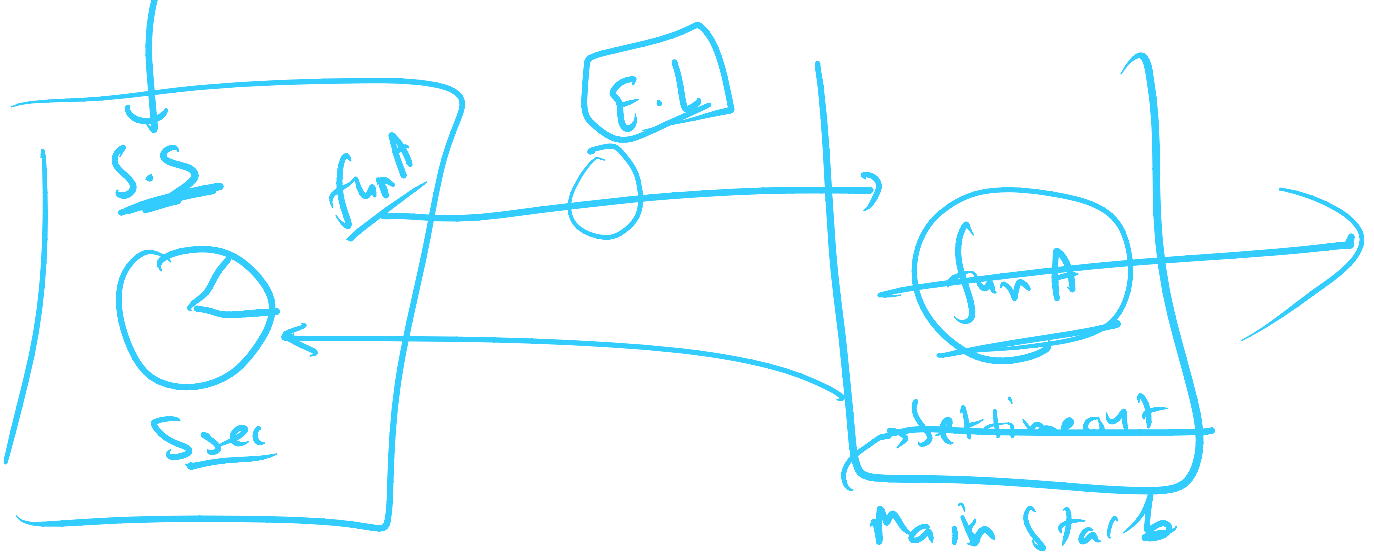
function
↓
get passed as an argument
to some another function

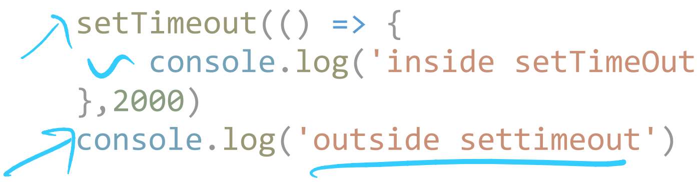
callback

fun A   fun B

fun A (fun B)

SettimeOut (functionA, time in ms)

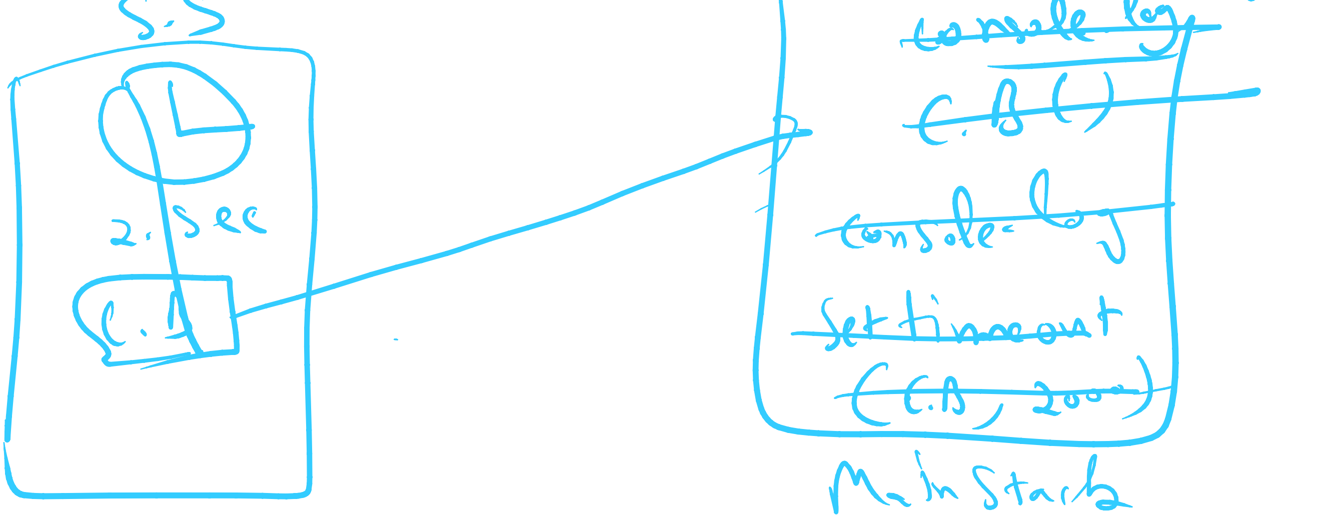


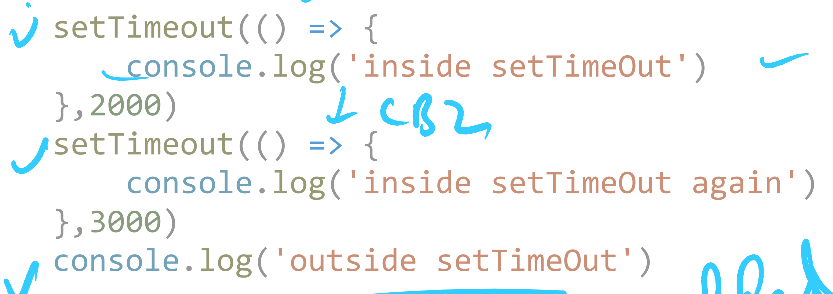


```
setTimeout(() => {
  console.log('inside setTimeOut')
},2000)
console.log('outside settimeout')
```

outside settimeout

inside set--





```
setTimeout(() => {
    console.log('inside setTimeOut')
},2000)
setTimeout(() => {
    console.log('inside setTimeOut again')
},3000)
console.log('outside setTimeOut')
```

CB1

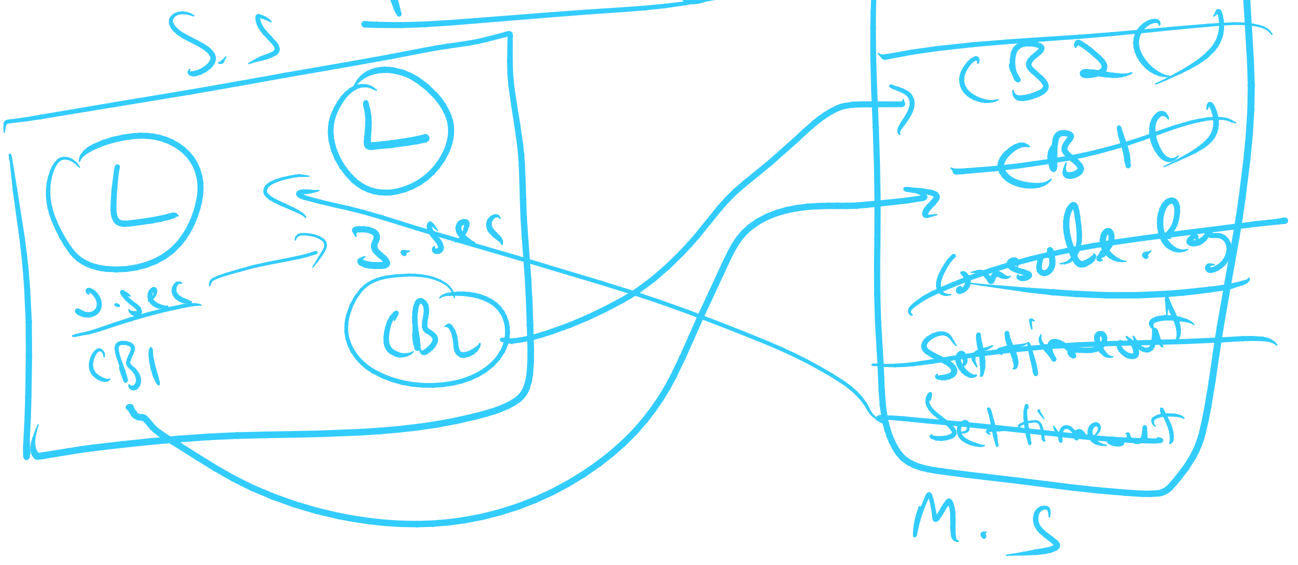
CB2

O.S

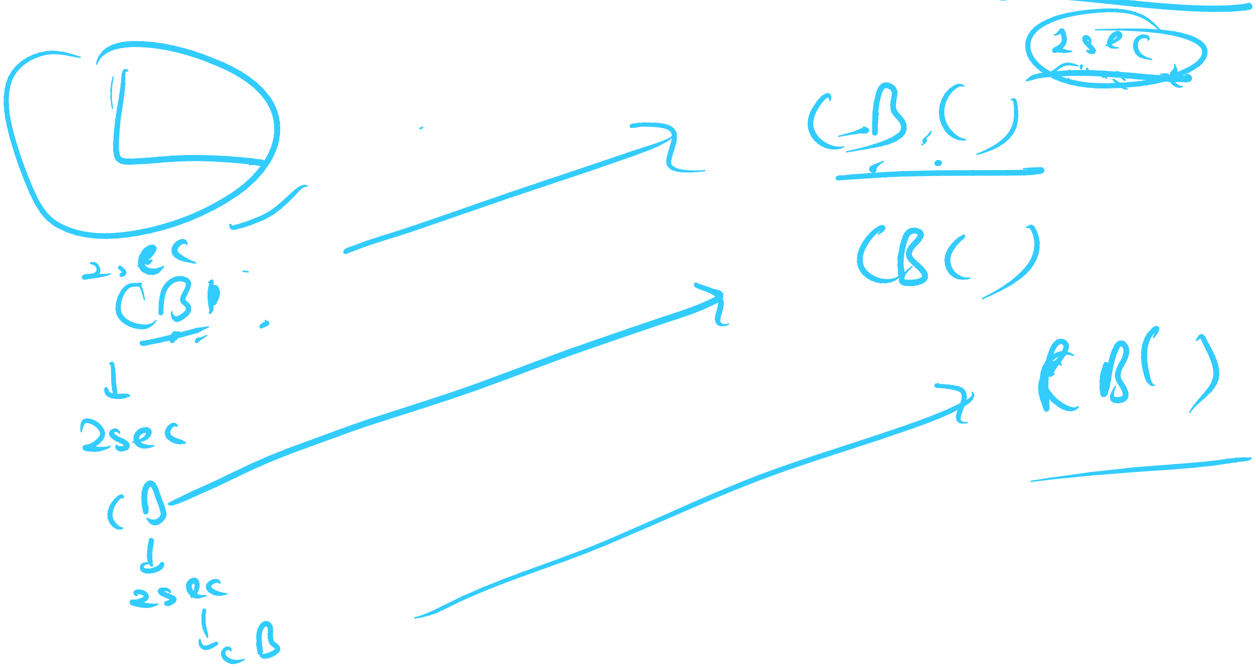
I.S

I.S again

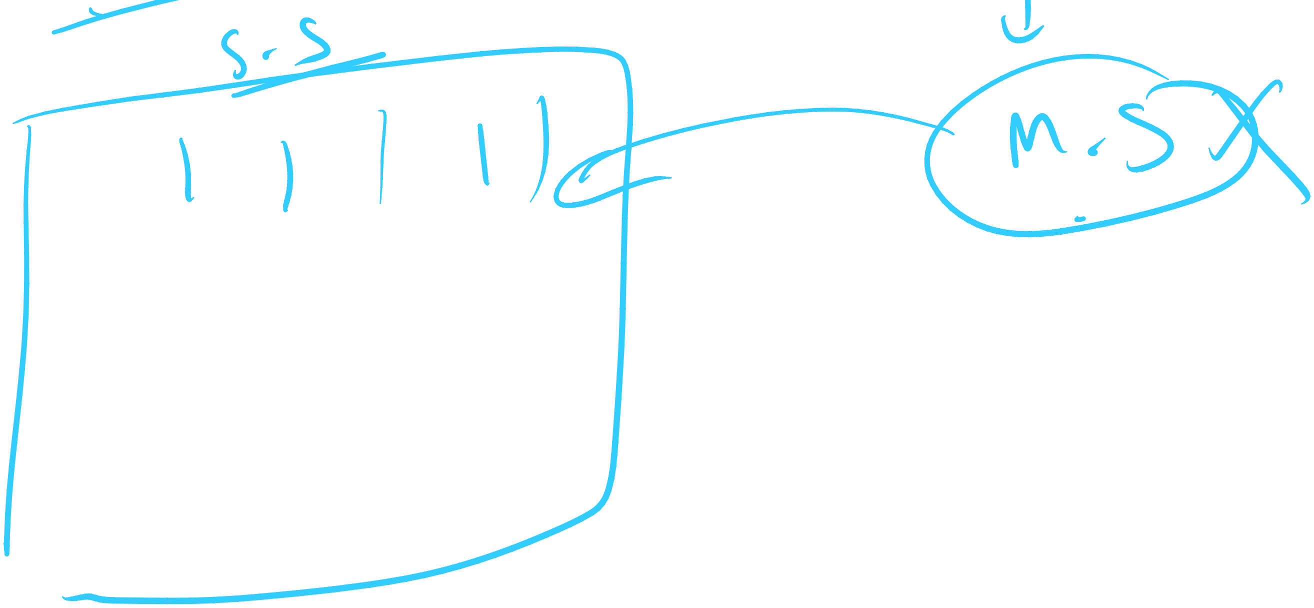
Parallely

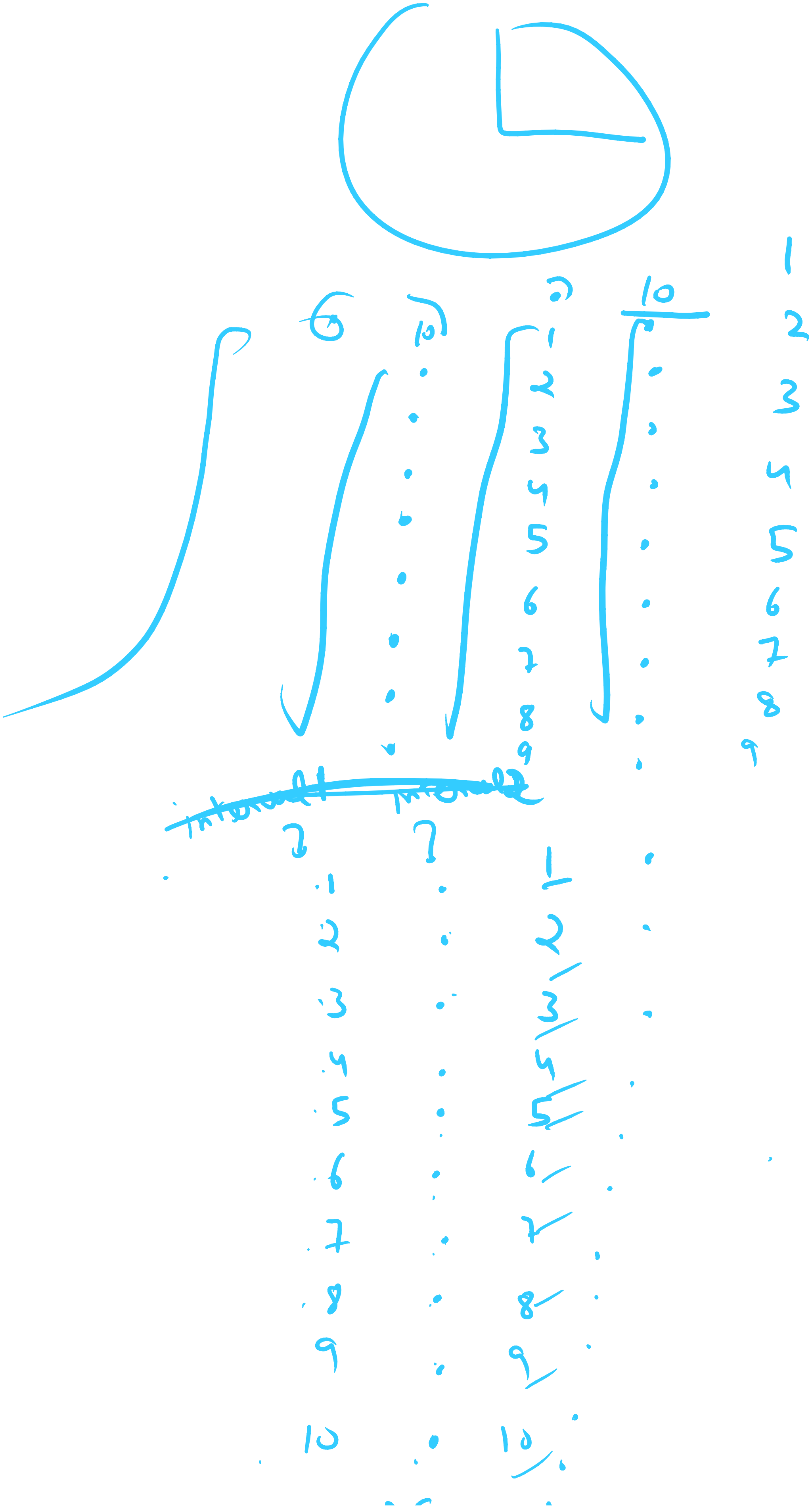


Setinterval (function , time in ms 2000)



Settimeout -- Setinterval





1. Print Flag = 0

2. Count = 1

interval = 1sec

10sec

10sec

6 7 8 .. 10

10 : 10

PF = true

Count = ~~1~~ ~~2~~ ~~3~~ ... ~~10~~ ~~0~~ ~~1~~ ~~2~~ ... ~~10~~ ~~0~~ ~~1~~ 2

```
let printFlag = true
let count = 1
let interval = setInterval(() => {
    if(printFlag) {
        console.log(count)
    }
    if(count === 10) {
        printFlag = !printFlag
        count = 0
    }
    count++
},1000)
```

~~true~~ ~~false~~ true

true

! false

1

true

1	:	1	.
2	:	2	.
3	:		
10	:	10	