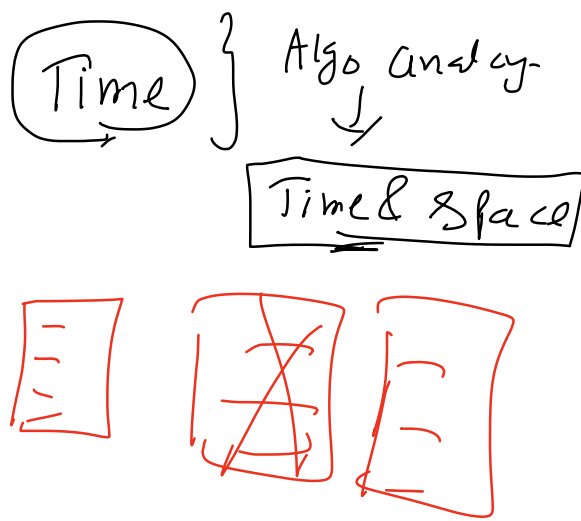


02-09 Revision Class

Friday, 2 September 2022 8:11 PM

- ✓ 1. Closure
- ✓ 2. Currying
- ✓ 3. Pass by value / Pass by ref



⊕ Closure:-

Revision class

+
Doubt session

```
> // return fun
  debugger;
  function callGreet(personName){
    debugger;
    function printName(){
      console.log(`Hello ${personName}`)
    }
    return printName;
  }
  callGreet("Usen");
```

Entire fun is returned

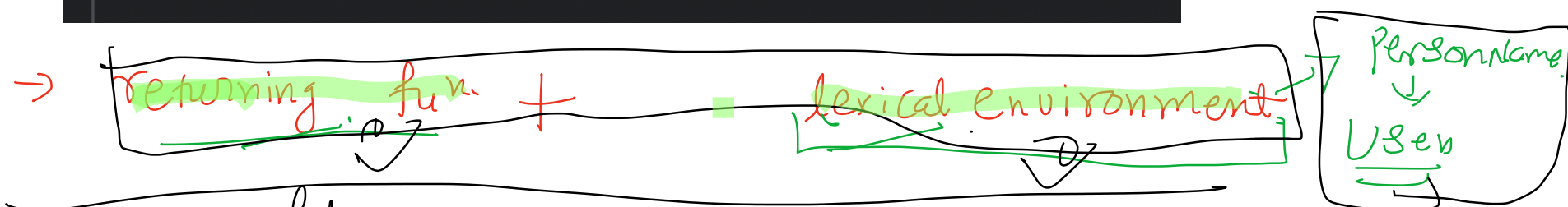
```
1 // return fun
2 debugger;
3 function callGreet(personName){
4   debugger;
5   function printName(){
6     console.log(`Hello ${personName}`)
7   }
8   return printName;
9 }
10 let res = callGreet("Usen");
11 console.log(res);
12 res();
```

fn name
res() ()

→ anonymous fn

```
1 // return fun
2 debugger;
3 function callGreet(personName){
4   debugger;
5   function printName(){
6     console.log(`Hello ${personName}`)
7   }
8   return printName;
9 }
10 let res = callGreet("Usen");
11 console.log(res);
12 res();
```

// inner



currying:-

getPen()

add(1, 2)
→ add(1)(2)

function getPen() {
 → getCar()
 → goShop()
}

Difficult to track error

getPen
→ getCar()

$fun(a, b, c) \Rightarrow fun(\underline{a}, \underline{b})(\underline{c})$

```
debugger;
const curryAdd = (sum) => { sum = 1
  return (a) => { // a = 1
    return (b) => { // b = 2
      return (c) => {
        return a+b+c;
      }
    }
  }
}
console.log(curryAdd(1)(2)(3));
```

fn a b
↳ (2)
fn not c
(3)

```
function add(x){
  x=2;
  return x+3;
}

let y = 7;
add(y);
add(y);
console.log(y)
```

