Revision Session

06 June 2022 20:12

- 2D arrays }
- Nested loops
- let and const in hoisting
- Pass by value and ref
- Arrays]
- itretive statements
- map
- filter sort reduce
- for of for in

✱ Arrays :- {

|10| |11| |12|

↳ Special variable }

↳ hold more than one value

```
arr = [10, 11, 12]
       0   1   2
```

arr

arr[1]   arr[0]

Syntax :-

```
arrayName = [val1, val2, ---valn]
```

✱ Creating an array:-

```
const cars = [ ];
```

①

```
cars[0] = "Bmw";
cars[2] = "Hero";
cars[1] = "Tesla";
cars[2] = "Honda";
```

```
cars = ["Bmw", "Hero", "Tesla", "Hond
```

②

```
cars.push("Audi");
```

③

```
const cars = new Array("Bmw", "Honda");
```

④ Accessing Arrays:-

cars[

arrayName[0]

↑ Index of arr

→ Starts from 0

Primitive & Non Primitive

Array → Object

{C

[

[

[

[1,0 "a"

✓

(*) Properties / Methods:-

(1) length:-

```
const Person = ["John", "Doe", 35];
```

Person.length // no. of elements

(2) Sort:-

// arrayname.sort((1))

Abhishek { Desc

// Class

Biber
Church

{ Biber ;
Abhishek
Church }

arr = [1, 2, 3, 4] // [4, 3, 2, 1]

arr.sort((a,b) => (b-a)) <0

new array:-

Map
filter
reduce
Sort

// access your last elem. 4 -> arr[3]

arr = [1, 2, 3, 4]

10 -> arr[9]

arr[arr.length - 1]

// loop through an array:

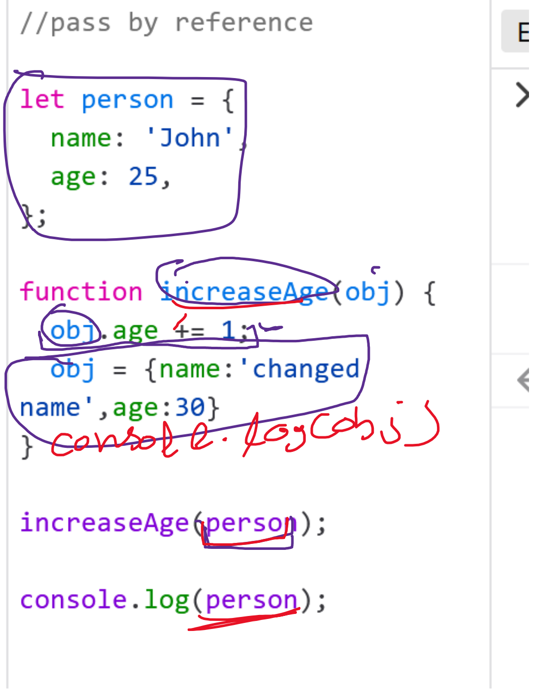
arr.foreach((elem) => {
elem
})

Mutable & Non-Mutable.

Numbered index { Const Person = ["John", "Doe",

named index → keys { Const Person = { (1) fName(1) :
"LName" :
"Age" :
}

//Person.fname Jawa object { " "
→ JSon object



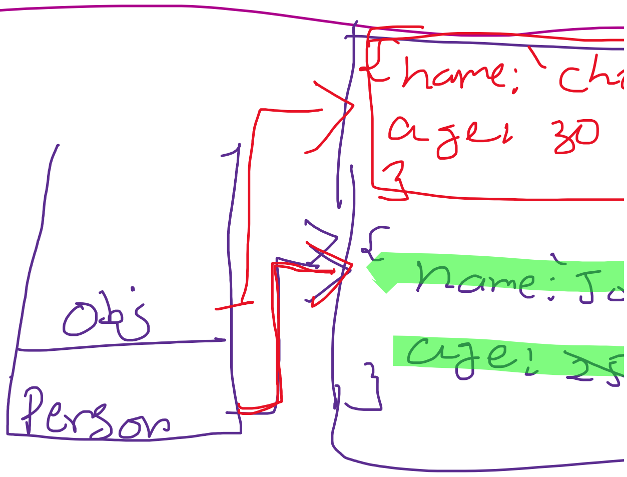
```
//pass by reference

let person = {
  name: 'John',
  age: 25,
};

function increaseAge(obj) {
  obj.age += 1;
  obj = {name:'changed
name',age:30}
} console.log(obj)

increaseAge(person);

console.log(person);
```


obj
Person
{name: 'ch
age: 30
}
{name: Jo
age: 25
}