Day25(closure , prototypes , constructor funcn)

Monday, 10 April 2023   7:44 PM

Closures

1. Closure ka istemaal hum classes se pehle karte the.
2. taki hum data members ko private bana sake jisse bahar ka banda uski value ko directly change na kar sake.

Def: → func bundled along with the reference to its lexical env.

or

reference to its surrounding state | lexical env.

Prototypes

Sachin ✗

object ke ander se agar hum uss key ko access kare jo uske ander nahi hai toh hume undefined milega.

if you try to access the property of an object, then JS will firstly find it inside that object, then, if not found, it will go to its Prototype.

fail

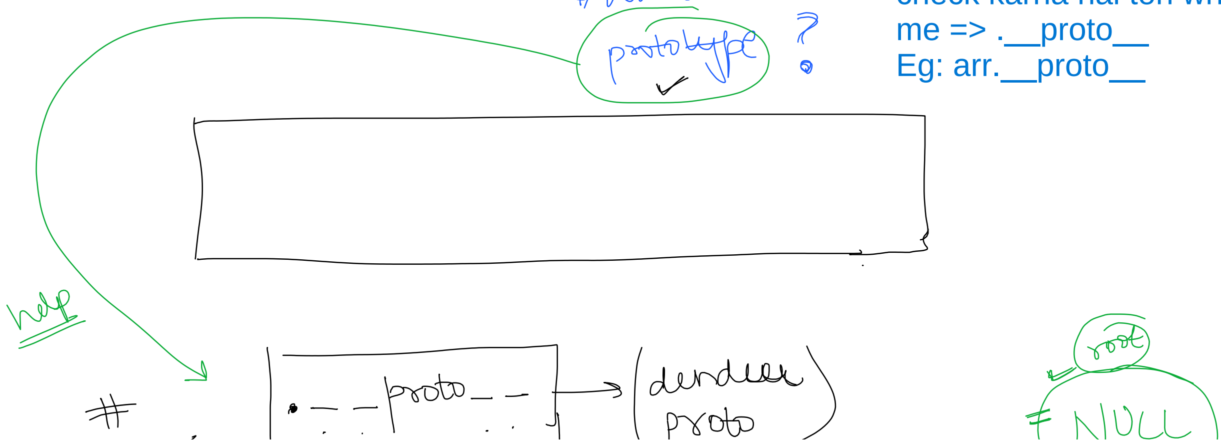
Prototype → it is also another object which is used as a fallback source of properties

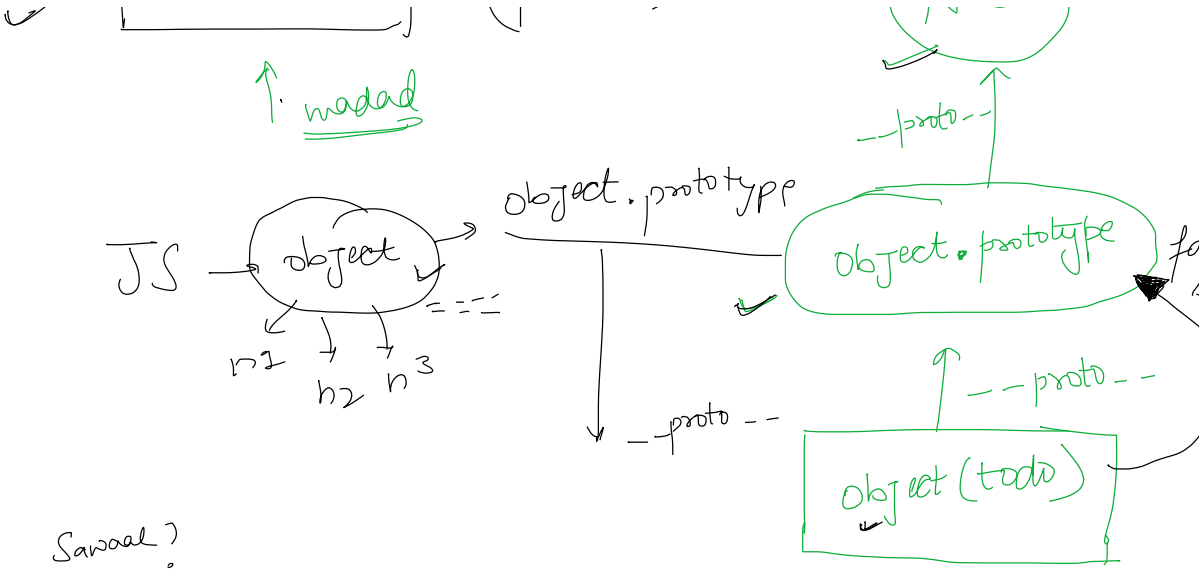
fallback means => agar key, object m nahi milta hai toh fallback karega matlab parent object m jaake dekhega.

toString() → todo object ✗

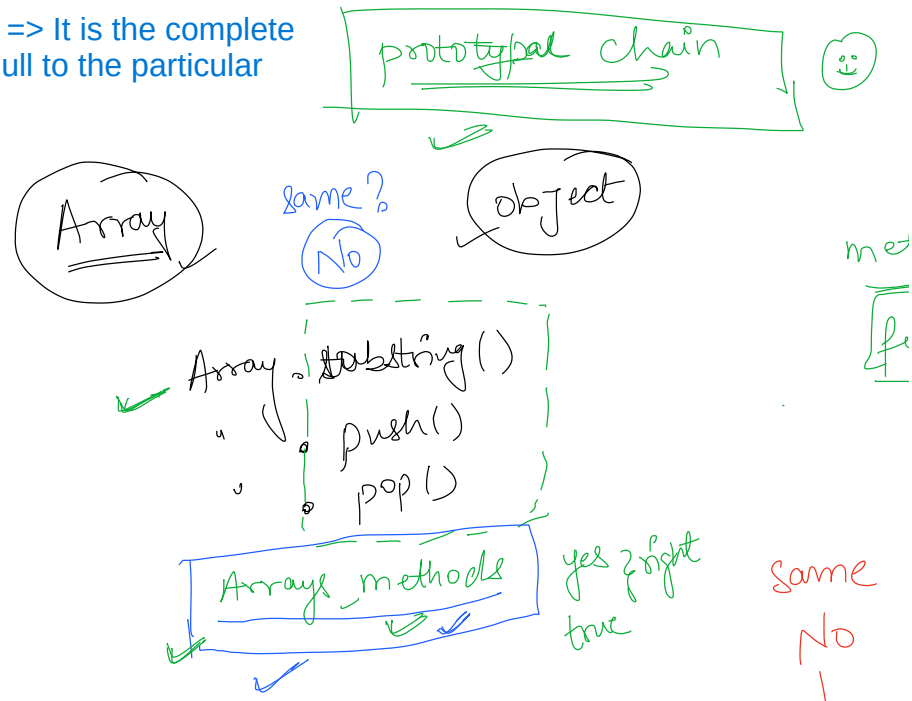
↓ what is its prototype ?

Agar mujhe kisi ka prototype check karna hai toh who will help me => .__proto__
Eg: arr.__proto__

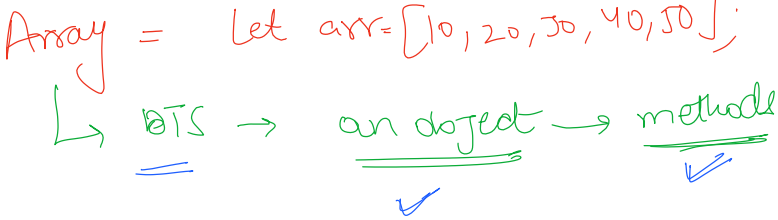




prototypal chain => It is the complete chain from the null to the particular object.



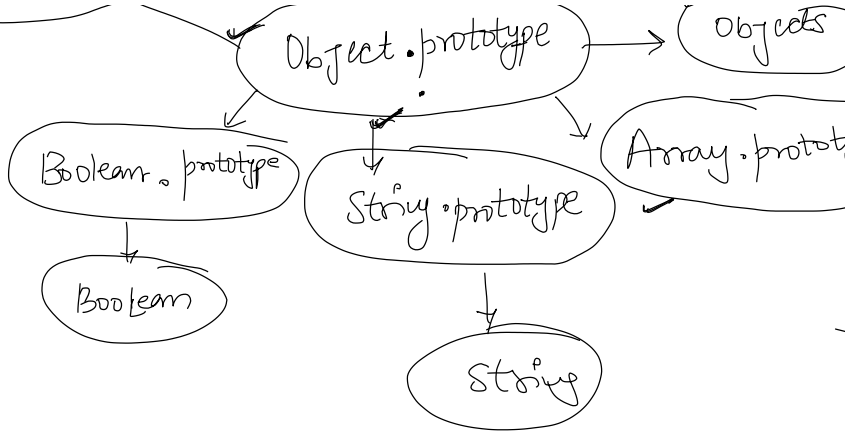
Array behind the scene converting itself into an object so that's why we are able to use the methods of the array.



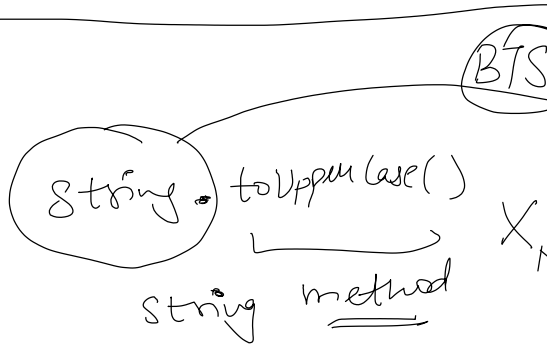
arr.__proto__ == Array.prototype
=> true

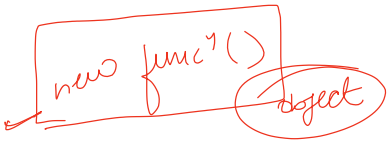
That's why we use all the methods which are defined inside Arrays prototype.

Prototype concept is same as the inheritance concept.



We can override the predefined methods of the object like arr.reverse = function(){ return 10 + 20}; Here the reverse method of array is override with out defined function, but the changes will be tempory not permanent





jab hum function ke aage new keyword lagake function ko call karte hai toh constructor calling hoti hai jiski wajeh se ek naya object create hota hai.

Constructor func"s

new Keyword before the func" Calling is creating this func" as a constructor func" whose task/job is to create objects.

Convention → Behtar tarika maana jaata hai

Compulsory ✗

→ Capitalize (Capital letter)

Kya hum iss object ke ander properties ko add kar sakte hai toh answer hai
=> Yes add kar sakte hai.

Kaise add karenge ?
1. this keyword ki help se hum object m properties add kar sakte hai.
2. this keyword ki help se kaise add ho paa raha hai. => this keyword point karvayega uss object ko jo ki constructor calling ki wajeh se ban raha hai.,

constructor function are used to create blueprint for creating object.

Agar hum chahate hai ki same function jo ki har object m same task perform karta hai toh hum uss function ko object ke prototype m add kara sakte hai taki rather then creating same function inside every object we put it into the prototype of that object so that we can directly access it from its parent / prototype object.

Jab bhi m contructor call karta hu toh uske prototype ka name kya hota hai exactly object ka name hi uska prototype hua.

```
function person(name,age){
    console.log("Inside Person object");
    console.log(this);
    this.name = name;
    this.age = age;
}

let a = new person("Amit",20);
console.log(a);

a.__proto__ == person.prototype
true

let a = [10,20,30];
a.__proto__ == Array.prototype
// true

a.__proto__.__proto__ == Object.prototype
// true
```