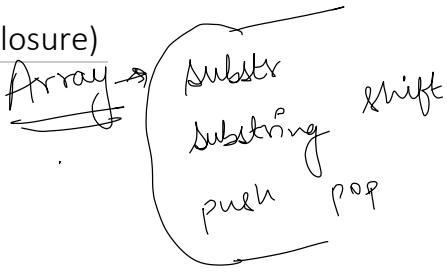
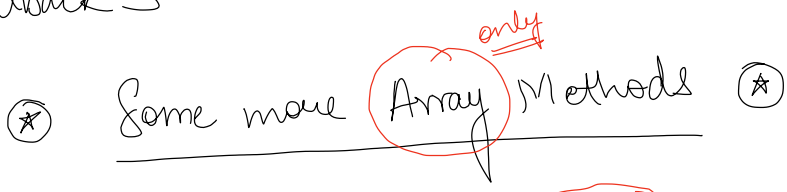
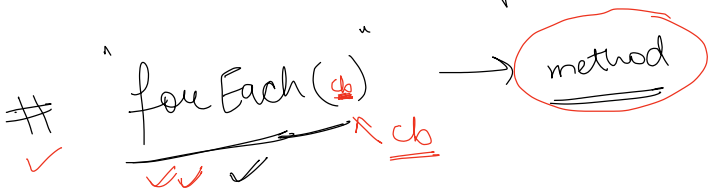
Day24(map, forEach, filter, sort, comparator , closure)

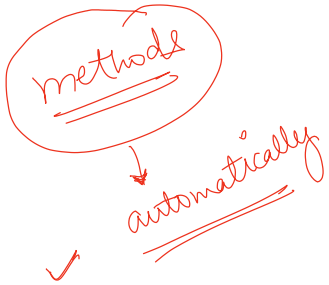
Friday, 7 April 2023 7:46 PM



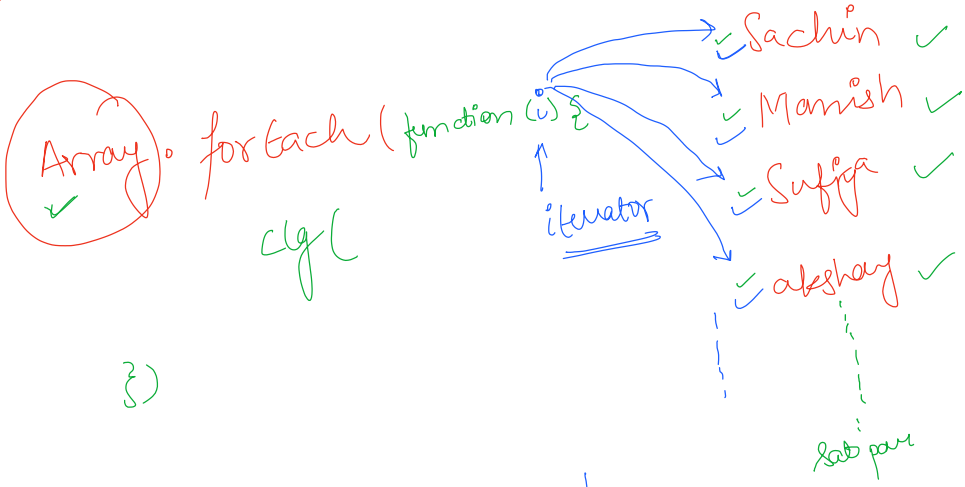
HOF & Callback

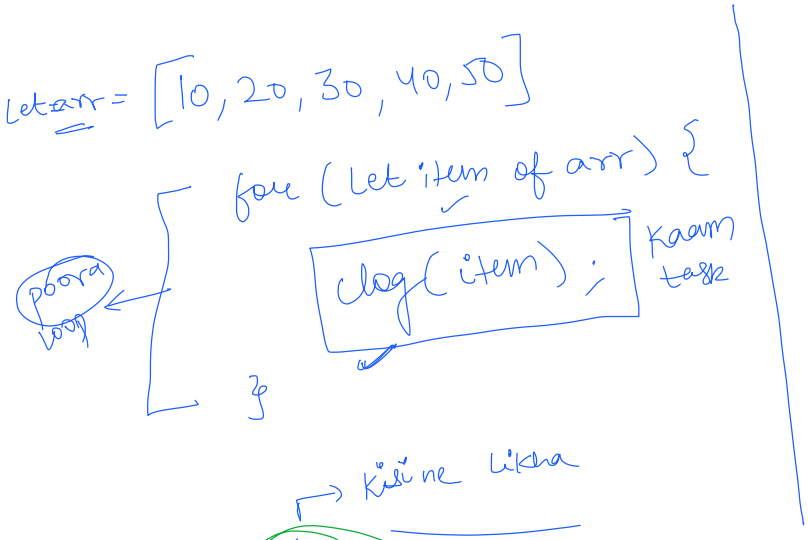


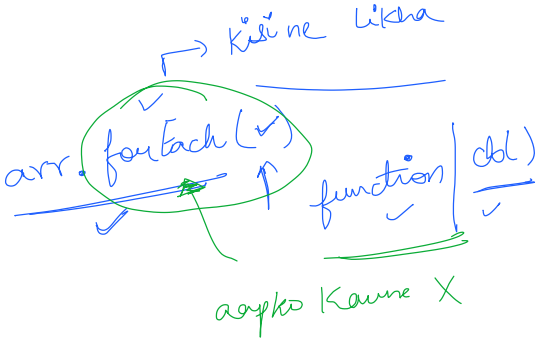




1. it accepts a callback func







array

① forEach — return X
② map() — return ✓

map() (automatic call)

① it accepts a callback func.
② it returns a new array.

React

filter()

① it accepts a callback (cb)
② it returns a new array.
③ it only sends the truthful value to the new Array.

(condition) → Each (T) ✓ add (value)
→ Short (F) ✗ nahi

sort

sort the value lexiographically.

Lexiographically

homework
Comparator
5-7 mins
youtube

Sort
asc.
123
desc-
321
2, 1
a, b
a - b > 0
1, 2
return

a - b = 0 => No changes
a - b > 0 => a is greater than b means a comes after b.
a - b < 0 => a is less than b means b comes after a.

closures

A function bundled together with reference to its surrounding state or lexical environment that is called Closure.

Todd → closure

jab bhi hum koi function return karte hai toh simple function return nahi hota woh apne sath sath unn variables ko bhi lata hai (kon se variable ?) jinka istemaal uss function ke ander kahi na kahi ho rakha hai (lexical environment m se variables ko lekar jata hai agar woh variables uske ander present ya declared hai toh taki woh unhe baad m istemaal kar sake.

A closure is a function that has access to its outer function scope even after the function has returned. Meaning, A closure can remember and access variables and arguments reference of its outer function even after the function has returned.

A closure is the combination of a function bundled together (enclosed) with references to its surrounding state (the lexical environment). In other words, a closure gives a function access to its outer scope. In JavaScript, closures are created every time a function is created, at function creation time.

```
function init() {

  var name = "Mozilla"; // name is a local variable created by init

  function displayName() {

    // displayName() is the inner function, that forms a closure
    console.log(name); // use variable declared in the parent function

  }

  displayName();
}
init();
```

init() creates a local variable called name and a function called displayName(). The displayName() function is an inner function that is defined inside init() and is available only within the body of the init() function. Note that the displayName() function has no local variables of its own. However, since inner functions have access to the variables of outer scopes, displayName() can access the variable name declared in the parent function, init().

Closure Detailed explanation:

https://chatgpt.com/c/30da716d-c4ea-42cf-a2d0-43f2a0653ff2