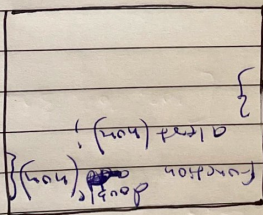


①

```
let numbers = [100, 200, 33, 45, 65, 76, 51];
function double(num) {
  alert(num);
  let doubleNum = num * 2;
}
```

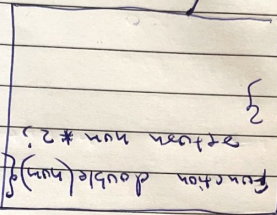


It will alert each element of numbers at it is called and after one

100, 200, 33, 45, 65, 76, 51

②

```
let doubleNum =
  numbers.forEach
    (double)
```

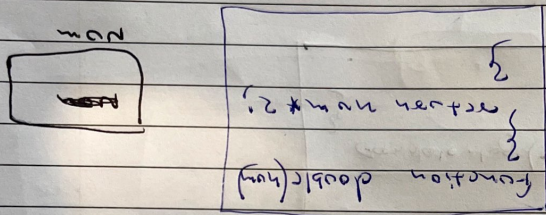


[200, 400, 66, 90, 130, 152, 102]
(garbage collected)

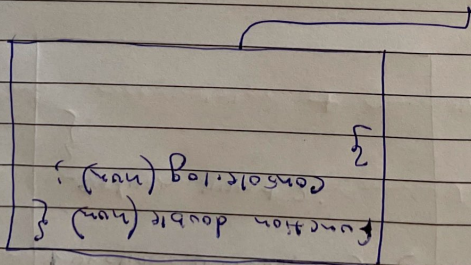
So it will be undefined as for each double return it only calls the all back function

doubleNumbers = [200, 400, 600, 800, 1000, 1200]

let doubleNumbers = numbers.map(double);



100
200
300
400
500
600
700
800
900
1000

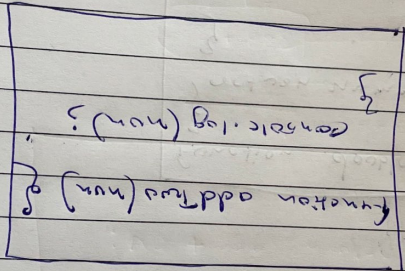


let doubleNum = numbers.forEach(double)

51
76
64
45
33
200

Double Numbers = 100

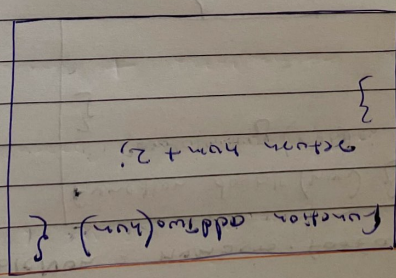
let doubleNumbers = numbers.map(addTwo)



6

doubleNumbers = [102, 202, 35, 47, 67, 78, 53]

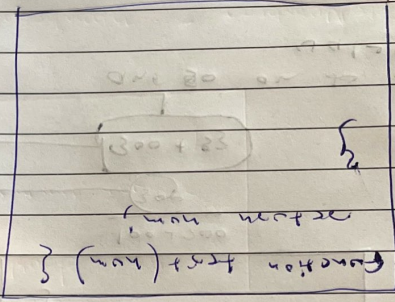
let doubleNumbers = numbers.map(addTwo);



5

Filtered Numbers = [100, 200, 33, 45, 65, 76, 51]

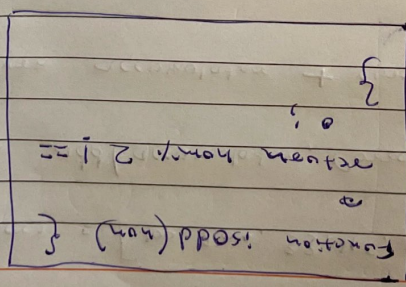
let FilteredNumbers = numbers.Filter(test);



Filter action new array, value based on have and false and in this case it is returning smaller array.

addNumbers = [33, 45, 65, 51]

let addNumbers = numbers.Filter(isodd);

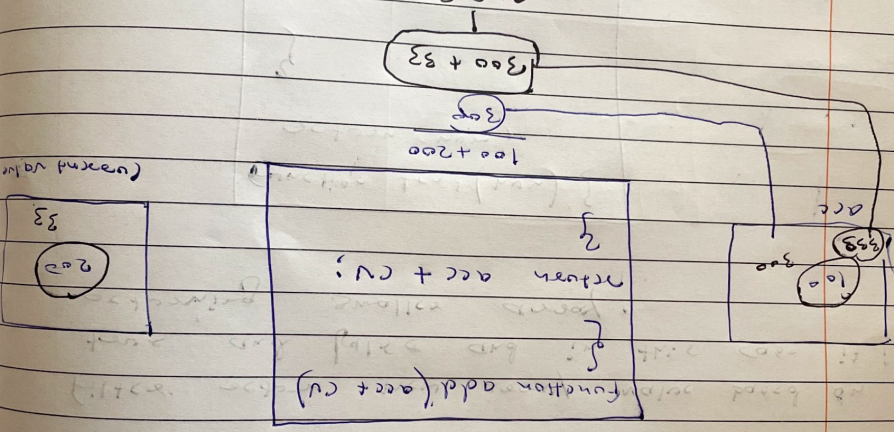


Q Let numbers = [100, 200, 33, 45, 65, 76, 51];

function add(accumulator, currentValue) {

return accumulator + currentValue;

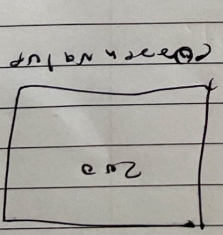
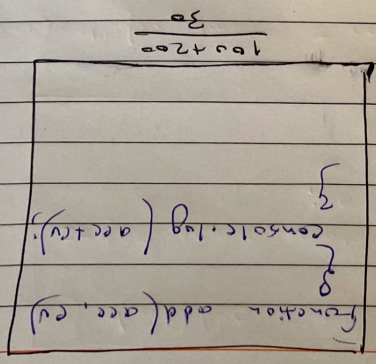
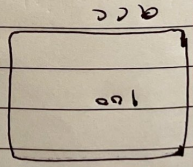
let sum = numbers.reduce(add);



And so on to the last value

Sum = 370

(13)



$$\begin{array}{r} 100 + 200 \\ 300 \end{array}$$

let sum = numbers.reduce(add);

console will print the first address

300;