**map() Array method**

* map method is use to create a new array from existing one by using a function to each elements of the existing array.
* map() does not execute the function for empty elements.
* map() does not change the original array.

Note

Map method takes a callback function and the callback function take three arguments first as a current Element, second as a index of current elements and third as a array it self.

**filter() Array method**

* filter method is use to create a new array by applying the condition of each element of existing array if the condition is true then it pushed the element in output array. Otherwise not
* map() does not execute the function for empty elements.
* map() does not change the original array.

Note

Filter method takes a callback function and the callback function take three arguments first as a current Element, second as a index of current elements and third as a array it self.

**reduce() Array method**

The reduce() method executes a reducer function for array element . and it returns a single value that is the function's accumulated result.

* reduce() does not execute the function for empty elements.
* reduce() does not change the original array.
* If user forget to give initial value in that case reduce set first element of array as a initial value automatically.

Note

The reduce method takes two arguments first arguments takes as callback function and the second arguments takes as initial value for accumulator.

*array*.reduce(function(total, currentValue, currentIndex, arr), initialValue)

First arguments also takes four arguments accumulator, current value, index, array itself

Accumulator: it store the previous result

const arr=[10,20,30,40,50];

let toatalSum=arr.reduce((accumulator,item)=>{

    return accumulator+item;

    // Accumulator     arrayVal     Return

    //     0       +    10        = 10

    //     10      +    20        = 30

    //     30      +    30        = 60

    //     60      +    40        = 100

    //     100     +    500       = 150

}, 0)// zero is the initial value for  accumulator.

console.log(toatalSum);

**forEach() Array method**

* The forEach() method calls a function for each element in an array.
* The forEach() method takes two arguments first arguments takes as a callback function and the second arguments is thisValue (thisValue Optional. Default undefined. A value passed to the function as its this value.)
* First arguments also takes three arguments current value, index, array itself
* forEach() not return anything
* The forEach() method itself does not modify the original array, but it provides the ability to do modefy

numbers.forEach(num => {

  console.log(num \* 2); // This just logs the doubled value, doesn't modify the array

});

console.log(numbers); // [1, 2, 3, 4] - The original array remains unchanged

//////////////////////////////////////////////////////////////////////////////

const numbers2 = [1, 2, 3, 4];

numbers.forEach((num, index, arr) => {

  arr[index] = num \* 2; // This modifies the original array

});

console.log(numbers2); // [2, 4, 6, 8] - The original array is modified

***array*.forEach(function(currentValue, index, arr), thisValue)**