**Do the below programs in anonymous function & IIFE**

## 1.a)Print odd numbers in an array using anonymous function:

let numbers = [66,22,65,79,90];

let odd = function(number)

{

let oddNumbers = [];

for(let index=0;index<number.length;index++)

{

if(number[index] %2 == 0)

{

oddNumbers.push(number[index]);

}

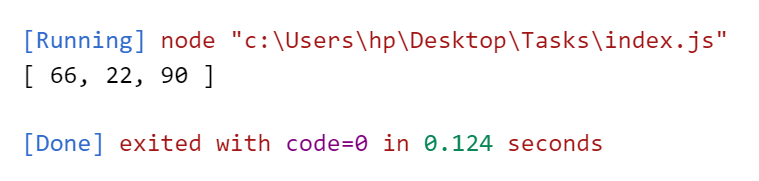
}

console.log(oddNumbers);

}

odd(numbers);

Output:



## 1.b)Print odd numbers in an array using IIFE function:

let numbers = [66,22,65,79,90];

(function(number)

{

let oddNumbers = [];

for(let index=0;index<number.length;index++)

{

if(number[index] %2 == 0)

{

oddNumbers.push(number[index]);

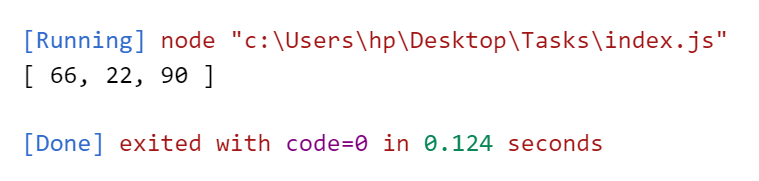
}

}

console.log(oddNumbers);

})(numbers);

OutPut:



## 2.a)Convert all the strings into a title caps in an array anonymous function:

let strings = ['apple','mangao','grapes'];

let cap = function(values)

{

let finalArray = [];

for(let string of values)

{

let letter = string.charAt(0).toUpperCase() + string.substring(1);

finalArray.push(letter);

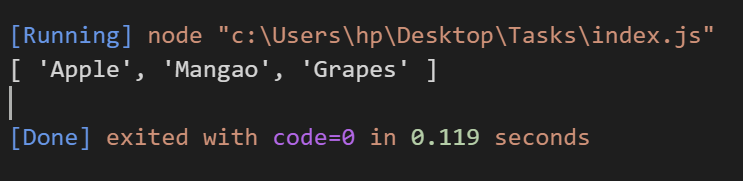
}

console.log(finalArray);

}

cap(strings)

Output:



## 2.b)Convert all the strings into a title caps in an array IIFE function:

let strings = ['apple','mangao','grapes'];

(function(values)

{

let finalArray = [];

for(let string of values)

{

let letter = string.charAt(0).toUpperCase() + string.substring(1);

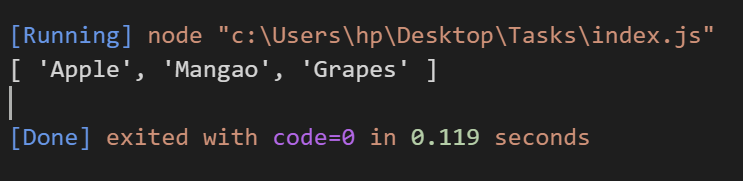
finalArray.push(letter);

}

console.log(finalArray);

})(strings);

Output:



## 3.a)Sum of all numbers in an array using anonymous function:

let numbers = [1,2,4,5];

let sum = function(values)

{

let result = 0;

for(let index in numbers)

{

result+=values[index];

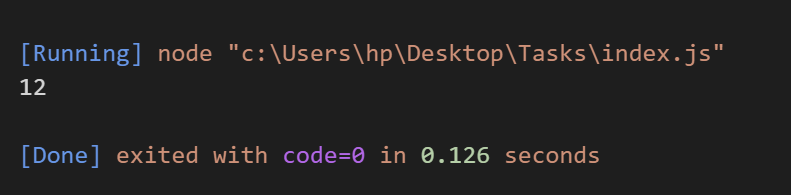
}

return result;

}

console.log(sum(numbers));

Output:



## 3.b)Sum of all numbers in an array using IIFE function:

let numbers = [1,2,4,5];

(function(values)

{

let result = 0;

for(let index in numbers)

{

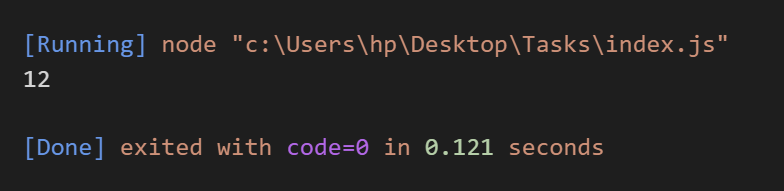
result+=values[index];

}

console.log(result);

})(numbers);

Output:



## 4.a)Return all prime numbers in an array using anonymous function:

let numbers = [12,13,14,17];

let prime = function(values)

{

let isPrime =[];

for(let num of numbers)

{

let factor = 0;

for(let x=0;x<=num;x++)

{

if(num%x == 0)

{

factor++

}

}

if(factor == 2)

{

isPrime.push(num);

}

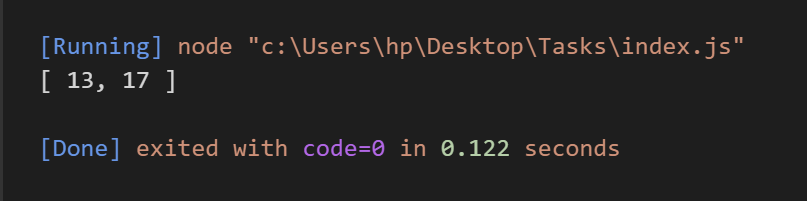
}

return isPrime;

}

console.log(prime(numbers));

Output:



## 4.b)Return all prime numbers in an array using IIFE function:

let numbers = [12,13,14,17];

(function(values)

{

let isPrime =[];

for(let num of numbers)

{

let factor = 0;

for(let x=0;x<=num;x++)

{

if(num%x == 0)

{

factor++

}

}

if(factor == 2)

{

isPrime.push(num);

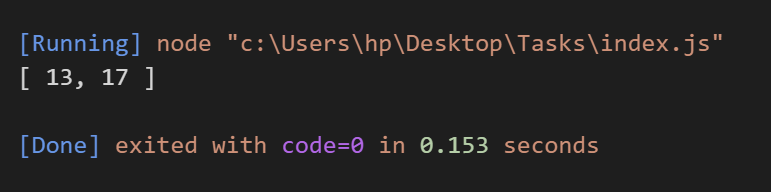
}

}

Return isPrime;

})(numbers);

Output:



## 5.a)Return all palindrome in an array using Anonymous function:

let array = ['malayalam','hello','madam'];

let palindrome = function(values)

{

let palinArray =[];

for(let string of values)

{

let arrayValues = string.split('');

let arrayRev = arrayValues.reverse().join('');

if(string == arrayRev)

{

palinArray.push(string);

}

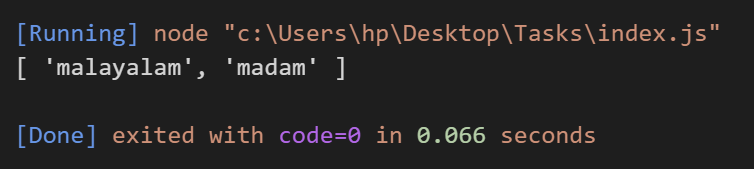
}

return palinArray;

}

console.log(palindrome(array));

Output:



## 5.b)Return all palindrome in an array using IIFE function:

let array = ['malayalam','hello','madam'];

(function(values)

{

let palinArray =[];

for(let string of values)

{

let arrayValues = string.split('');

let arrayRev = arrayValues.reverse().join('');

if(string == arrayRev)

{

palinArray.push(string);

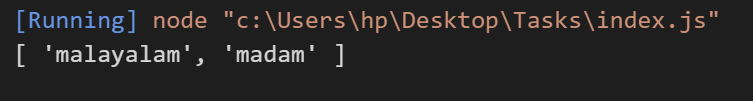
}

}

return palinArray;

})(array);

Output:



## 6.a)Return the median of two sorted arrays of same size using Anonymous function:

let array1 = [1,5,3,9];

let array2 = [2,6,4,8];

let median = function(value1,value2)

{

let sort1 = array1.sort();

let sort2 = array2.sort();

let merge = sort1.concat(sort2);

if(merge.length%2 == 0)

{

return ((merge[merge.length/2] + merge[merge.length/2-1]));

}

else{

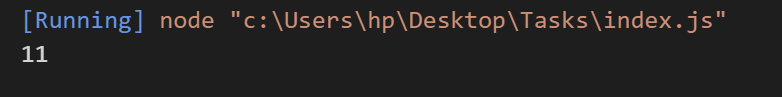
return merge[Math.floor(merge.length/2)]

}

}

console.log(median (array1,array2));

Output:



## 6.b)Return the median of two sorted arrays of same size using IIFE function:

let array1 = [1,5,3,9];

let array2 = [2,6,4,8];

( function(value1,value2)

{

let sort1 = array1.sort();

let sort2 = array2.sort();

let merge = sort1.concat(sort2);

if(merge.length%2 == 0)

{

return ((merge[merge.length/2] + merge[merge.length/2-1]));

}

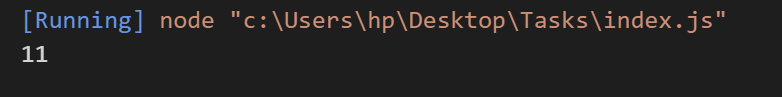
else{

return merge[Math.floor(merge.length/2)]

}

}) (array1,array2);

Output:



})

console.log(duplicateArrays);

}

removeDuplicates(numbers);

7.a)Remove Duplicates from an array using Anonymous function:

let numbers = [12,13,14,13,14,1,3,5,1,2];

let removeDuplicates = function(values)

{

let duplicateArrays = [];

numbers.forEach((item)=>

{

if(!duplicateArrays.includes(item))

{

duplicateArrays.push(item);

}

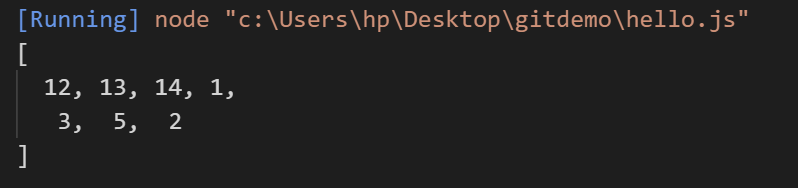
})

console.log(duplicateArrays);

}

removeDuplicates(numbers);

Output:



7.a)Remove Duplicates from an array using IIFE function:

let numbers = [12,13,14,13,14,1,3,5,1,2];

(function(values)

{

let duplicateArrays = [];

numbers.forEach((item)=>

{

if(!duplicateArrays.includes(item))

{

duplicateArrays.push(item);

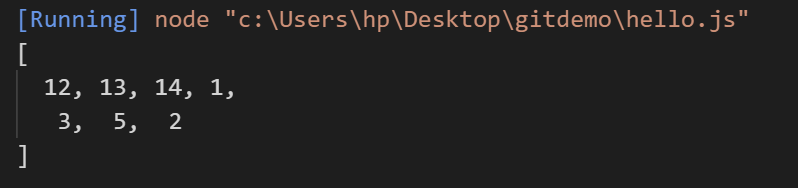
}

})

console.log(duplicateArrays);

})(numbers);

Output:



8.a)Rotate an array by k times using Anonymous function:

let rotate = function (arr, numberOfShifts) {

let tmp = 0;

const leng = arr.length;

numberOfShifts = numberOfShifts % leng;

for (let i = 0; i < numberOfShifts; i++) {

tmp = arr.pop();

arr.unshift(tmp);

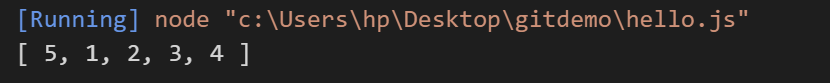
}

return arr;

};

console.log(rotate([1,2,3,4,5],1))

Output:



8.b)Rotate an array by k times using IIFE function:

(function (arr, numberOfShifts) {

let tmp = 0;

const leng = arr.length;

numberOfShifts = numberOfShifts % leng;

for (let i = 0; i < numberOfShifts; i++) {

tmp = arr.pop();

arr.unshift(tmp);

}

console.log(arr);

})(rotate([1,2,3,4,5],3));

Output:

