1. Do the below programs in anonymous function & IIFE
   1. Print odd numbers in an array
   2. Convert all the strings to title caps in a string array
   3. Sum of all numbers in an array
   4. Return all the prime numbers in an array
   5. Return all the palindromes in an array
   6. Return median of two sorted arrays of same size
   7. Remove duplicates from an array
   8. Rotate an array by k times
2. Do the below programs in arrow functions
   1. Print odd numbers in an array
   2. Convert all the strings to title caps in a string array
   3. Sum of all numbers in an array
   4. Return all the prime numbers in an array
   5. Return all the palindromes in an array

1)

a) Print odd numbers in an array

var a = function (arr) {

*for* (var i = 0; i <= arr.length - 1; i++)

*if* (arr[i] % 2 == 0) {

console.log(+arr[i] + "even")

}

*else* {

console.log(+arr[i] + "odd")

}

}

a([1, 2, 3, 4, 5])

b.Convert all the strings to title caps in a string array

var newarr = []

var a = function (arr) {

*for* (var i = 0; i <= arr.length - 1; i++) {

var val = arr[i][0].toUpperCase() + arr[i].slice(1)

newarr.push(val)

}

}

a(["hi", "hello", "sleep", "eating", "running"])

console.log(newarr)

c.Sum of all numbers in an array

var sum = 0

var a = function (arr) {

*for* (var i = 0; i <= arr.length - 1; i++) {

sum = sum + arr[i]

}

}

a([1, 2, 3, 4, 5])

console.log(sum)

d)Return all the prime numbers in an array

var a = function (number) {

*for* (var i = 0; i <= number.length - 1; i++) {

var isPrime = true;

*if* (number[i] === 1) {

console.log("1 is neither prime nor composite number.");

}

*else* *if* (number[i] > 1) {

*for* (let j = 2; j < number[i]; j++) {

*if* (number[i] % i == 0) {

isPrime = false;

*break*;

}

}

*if* (isPrime) {

console.log(`${number[i]} is a prime number`);

} *else* {

console.log(`${number[i]} is a not prime number`);

}

}

}

}

a([5, 6, 4, 2, 1])

Return all the palindromes in an array:

function check\_palindrome( str )

{

let j = str.length -1;

*for*( let i = 0 ; i < j/2 ;i++)

{

let x = str[i] ;

let y = str[j-i];

*if*( x != y)

{

*return* false;

}

}

*return* true;

}

function is\_palindrome( str )

{

let ans = check\_palindrome(str);

*if*( ans == true )

{

console.log("passed string is palindrome ");

}

*else*

{

console.log("passed string not a palindrome");

}

}

let test = "racecar";

is\_palindrome(test);

Return median of two sorted arrays of same size

function getMedian(ar1, ar2, n) {

var i = 0;

var j = 0;

var count;

var m1 = -1, m2 = -1;

*for* (count = 0; count <= n; count++) {

*if* (i == n) {

m1 = m2;

m2 = ar2[0];

*break*;

}

*else* *if* (j == n) {

m1 = m2;

m2 = ar1[0];

*break*;

}

*if* (ar1[i] <= ar2[j]) {

m1 = m2;

m2 = ar1[

i++]

}

*else* {

m1 = m2;

m2 = ar2[j];

j++;

}

}

*return* (m1 + m2) / 2;

}

var ar1 = [1, 12, 15, 26, 38];

var ar2 = [2, 13, 17, 30, 45];

var n1 = ar1.length;

var n2 = ar2.length;

*if* (n1 == n2)

console.log("Median is " + getMedian(ar1, ar2, n1));

*else*

console.log("Doesn't work for arrays of unequal size");

7)Remove duplicates from an array

var array=["x","w","d","w","x"];

function filterDuplicates(new\_array){

*return* new\_array.filter((value,index) =>new\_array.indexOf(value)===index);

}

console.log(filterDuplicates(array));

8)Rotate an array by k times

function leftRotate(arr , d , n) {

*for* (i = 0; i < d; i++)

leftRotatebyOne(arr, n);

}

function leftRotatebyOne(arr , n) {

var i, temp;

temp = arr[0];

*for* (i = 0; i < n - 1; i++)

arr[i] = arr[i + 1];

arr[n - 1] = temp;

}

function printArray(arr , n) {

*for* (i = 0; i < n; i++)

document.write(arr[i] + " ");

}

var arr = [ 1, 2, 3, 4, 5, 6, 7 ];

leftRotate(arr, 2, 7);

printArray(arr, 7);

1)Print odd numbers in an array

let arr = [1,2,3,4,5,6,7,8,9,10,11,12]

let odds = arr.filter(n => n%2)

console.log(odds)

2)Convert all the strings to title caps in a string array

let arr = ["joe", "hep", "hens", "fruit"]

let odds = arr.map(n => n[0].toUpperCase() + n.slice(1))

console.log(odds)

3) Sum of all numbers in an array

let arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

let odds = arr.reduce((accumulator, currentValue) => accumulator + currentValue);

console.log(odds)

4)Return all the prime numbers in an array

const arr = [1, 3, 2, 5, 10];

const isPrime = num => {

*for* (let i = 2; i < num; i++) {

*if* (num % i === 0) *return* false;

}

*return* num !== 1;

};

let odds = arr.filter(element => isPrime(element));

console.log(odds)