1)Print odd numbers in an array

anonymous :

var res = function(arr){

var temp=[];

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

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

temp.push(arr[i]);

}

}

return temp;

}

console.log(res([12,13,14,15,16,17,18]));

IIFE :

(function(array){

var temp = []

for(var i = 0 ; i< array.length ; i++){

if(array[i]%2!=0){

temp.push(array[i]);

}

}console.log(temp)

})([12,13,14,15,16,17,18])

Arrow Function:

var oddNumbers = (array) => {

var temp =[];

for(var i = 0 ; i< array.length ; i++){

if(array[i]%2!=0){

temp.push(array[i]);

}

}

return temp;

}

console.log(oddNumbers([12,13,14,15,16,17]))

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

anonymous :

var res = function(str){

str = str.toLowerCase().split(" ");

for (var i = 0; i < str.length; i++) {

str[i] = str[i].charAt(0).toUpperCase() + str[i].slice(1);

}

return str.join(" ");

}

console.log(res("dinesh kumar"));

IIFE :

(function (str) {

str = str.toLowerCase().split(" ");

for (var i = 0; i < str.length; i++) {

str[i] = str[i].charAt(0).toUpperCase() + str[i].slice(1);

}

console.log(str.join(" "));

})("dinesh kumar");

Arrow Function :

var titleCase = (str) => {

str = str.toLowerCase().split(" ");

for (var i = 0; i < str.length; i++) {

str[i] = str[i].charAt(0).toUpperCase() + str[i].slice(1);

}

return str.join(" ");

}

console.log(titleCase("dinesh kumar"));

3)Sum of all numbers in an array

anonymous :

var a =function(array){

var sum = 0;

for(var i = 0 ; i< array.length ; i++){

sum =sum+array[i];

}

return sum;

}

console.log(a([2,3,4,5]));

IIFE :

var a =(function(array){

var sum = 0;

for(var i = 0 ; i< array.length ; i++){

sum = sum + array[i];

}

return sum;

})

console.log(a([1,2,3,4]));

Arrow:

var res = (array)=>{

var sum = 0;

for(var i = 0 ; i< array.length ; i++){

sum = sum + array[i];

}

return sum;

}

console.log(res([1,2,3,4]));

4)Return all the prime numbers in an array

Anonymous Function:

function(numArray){

numArray = numArray.filter((number) => {

for (var i = 2; i <= Math.sqrt(number); i++) {

if (number % i === 0) return false;

}

return true;

});

console.log(numArray);

}

IIFE

(

function(numArray){

numArray = numArray.filter((number) => {

for (var i = 2; i <= Math.sqrt(number); i++) {

if (number % i === 0) return false;

}

return true;

});

console.log(numArray);

})([1,2,3,4])

Arrow Function :

primeNumber = (numArray) => {

numArray = numArray.filter((number) => {

for (var i = 2; i <= Math.sqrt(number); i++) {

if (number % i === 0) return false;

}

return true;

});

console.log(numArray);

}

5) Return all the palindromes in an array

function isPalindrome(N)

{

let str = "" + N;

let len = str.length;

for (let i = 0; i < parseInt(len / 2, 10); i++)

{

if (str[i] != str[len - 1 - i ])

return false;

}

return true;

}

Anonymous Function : function (arr, n)

{

// Traversing each element of the array

// and check if it is palindrome or not

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

{

let ans = isPalindrome(arr[i]);

if (ans == false)

return false;

}

return true;

}

IIFE :

( function (arr, n)

{

// Traversing each element of the array

// and check if it is palindrome or not

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

{

let ans = isPalindrome(arr[i]);

if (ans == false)

return false;

}

return true;

})([1,2,3] , 3)

Arrow :

Palindrome = (arr, n) =>

{

// Traversing each element of the array

// and check if it is palindrome or not

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

{

let ans = isPalindrome(arr[i]);

if (ans == false)

return false;

}

return true;

}

Q.Return median of two sorted arrays of same size

Q.Remove duplicates from an Array

Anonymous Function : function(array){

let dup = [...new Set(array)];

console.log(dup);

}

IIFE : (function(array){

let dup = [...new Set(array)];

console.log(dup);

})([1,1,2,3,4])

Q.Rotate an array by K times

function reverse(array , li , ri){

while(li < ri){

int temp = a[li];

a[li]= a[ri];

a[ri] = temp;

li++;

ri--;

}

}

Anonymous function : function(array , k){

k = k % a.length;

if(k < 0){

k += a.length;

}

reverse(a, 0, a.length - k - 1);

reverse(a, a.length - k, a.length - 1);

reverse(a, 0, a.length - 1);

}

IIFE : (function(array , k){

k = k % a.length;

if(k < 0){

k += a.length;

}

reverse(a, 0, a.length - k - 1);

reverse(a, a.length - k, a.length - 1);

reverse(a, 0, a.length - 1);

})([1,2,3,4] , 2)