## Generate an API key with given link below https://openweathermap.org/guide Print the current weather data in console- By lat lang

**HTML :**

<!DOCTYPE html>

<html lang="en">

<head>

<title>function api call</title>

</head>

<body>

<h1>Weather data in console - Latitude and longitude</h1>

<div id="root"></div>

<script src="api\_key.js">

</script>

</body>

</html>

**Js:**

var request = new XMLHttpRequest()

var url\_string = 'http://api.openweathermap.org/data/2.5/weather?q=London,uk&APPID=cf10aa90261638e49ec62344d097b5ce';

// Open a new connection, using the GET request on the URL endpoint

request.open('GET',url\_string)

request.send();

request.onload = function() {

// Begin accessing JSON data here

//console.log(this.response);

var data = JSON.parse(this.response)

console.log(data.coord);

}

# programs in anonymous function & IIFE

## Print odd numbers in an array

**ANONYMOUS FUNCTION:**

let array = [1,2,3,4,5]

let newarr=[]

let oddNumber=function (array){

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

if(array[i] % 2 ===1)

newarr.push(array[i]);

}

return newarr;

}

console.log(oddNumber(array))

**IIFE:**

let newarr=[]

let oddNumber=function (array){

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

if(array[i] % 2 ===1)

newarr.push(array[i]);

}

console.log (newarr);

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

## Convert all the strings to title caps in a string array

**ANONYMOUS FUNCTION:**

var result ="";

var str = "guvi geeks coding";

let titleCase =function(str){

result= str.split(' ').map(w => w[0].toUpperCase() + w.substr(1).toLowerCase())

.join(' ');

return result;

}

console.log(titleCase(str));

**IIFE:**

let titleCase =function(str){

var result ="";

result= str.split(' ').map(w => w[0].toUpperCase() + w.substr(1).toLowerCase())

.join(' ');

console.log (result);

} ("guvi geeks coding")

## Sum of all numbers in an array

**ANONYMOUS FUNCTION:**

let array = [1,2,3,4,5]

let sumOfAllNum = function(array){

let sum = (accum,curr)=> accum + curr

console.log(array.reduce(sum))

}

sumOfAllNum(array);

**IIFE:**

let sumOfAllNum = function(array){

let sum = (accum,curr)=> accum + curr

console.log(array.reduce(sum))

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

## Return all the prime numbers in an array

**ANONYMOUS FUNCTION:**

let array = [1,2,3,4,5];

let isPrime = function(array){

let result =[]

result = array.filter(num => {

if(num < 2) return false;

for (let k = 2; k < num; k++){

if( num % k === 0){

return false;

}

}

return true;

});

return result;

}

console.log(isPrime(array));

**IIFE:**

let isPrime = function(array){

let result =[]

result = array.filter(num => {

if(num < 2) return false;

for (let k = 2; k < num; k++){

if( num % k === 0){

return false;

}

}

return true;

});

console.log(result) ;

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

## Return all the palindromes in an array

**ANONYMOUS FUNCTION:**

let arr = ['guvi','geekeeg',123,909,'pop'];

let isPalindrome = function(arr){

let result= [];

result = arr.filter(word => {

const str = String(word);

let i = 0;

let j = str.length - 1;

while(i < j) {

if(str[i] === str[j]) {

i++;

j--;

}

else {

return false;

}

}

return true;

});

return result

}

console.log(isPalindrome(arr));

**IIFE:**

let isPalindrome = function(arr){

let result= [];

result = arr.filter(word => {

const str = String(word);

let i = 0;

let j = str.length - 1;

while(i < j) {

if(str[i] === str[j]) {

i++;

j--;

}

else {

return false;

}

}

return true;

});

console.log(result)

}(['guvi','geekeeg',123,909,'pop'])

## Remove duplicates from an array

ANONYMOUS FUNCTION:

let unique = function (array){

const uniqueSet = new Set(array);

const backToArray = [...uniqueSet];

return backToArray ;

}

console.log(unique( [1,2,3,4,1,2,3,4,11,33,55,22,33]));

IIFE:

let unique = function (array){

const uniqueSet = new Set(array);

const backToArray = [...uniqueSet];

console.log(backToArray) ;

}( [1,2,3,4,1,2,3,4,11,33,55,22,33])

## Rotate an array by k times and return the rotated array

**ANONYMOUS FUNCTION:**

let rotate = function (array,k){

let splicedArray = array.splice(0,k);

let rotatedArray = array.concat(splicedArray);

return rotatedArray;

}

console.log(rotate([1,2,3,4,5,6,7,8,9],7))

**IIFE:**

let rotate = function (array,k){

let splicedArray = array.splice(0,k);

let rotatedArray = array.concat(splicedArray);

console.log(rotatedArray)

}([1,2,3,4,5,6,7,8,9],8)

## Return median of two sorted arrays of same size

**ANONYMOUS FUNCTION:**

let median = function(array1,array2){

if(array1.length === array2.length){

let array = array1.concat(array2);

let sorted = array.sort(function(a,b){return a-b});

let value="";

let result =0;

let arrlen = sorted.length;

let middleElement = arrlen /2;

if(middleElement %1 !== 0)

{

value= Math.round(middleElement);

//console.log(value);

return sorted[value-1];

}

if(middleElement %1 === 0)

{

result = (sorted[middleElement]+ sorted[middleElement-1]) / 2;

return result

}

}

else

return -1

}

console.log(median([5,7,9],[9,11]))

**IIFE:**

let median = function(array1,array2){

if(array1.length === array2.length){

let array = array1.concat(array2);

let sorted = array.sort(function(a,b){return a-b});

let value="";

let result =0;

let arrlen = sorted.length;

let middleElement = arrlen /2;

if(middleElement %1 !== 0)

{

value= Math.round(middleElement);

//console.log(value);

console.log (sorted[value-1]);

}

if(middleElement %1 === 0)

{

result = (sorted[middleElement]+ sorted[middleElement-1]) / 2;

console.log (result)

}

}

else

console.log("-1");

}([5,7,9],[11,13,15])