**Nested Arrays:**

const arr1 = [0, 1, 2, [3, 4]];

console.log(arr1.flat());

// output: [0, 1, 2, 3, 4]

const arr2 = [0, 1, 2, [[[3, 4]]]];

console.log(arr2.flat(2));

// output: [0, 1, 2, [3, 4]]

const arr6 = [1, 2, [3, 4, [5, 6, [7, 8, [9, 10]]]]];

console.log(arr6.flat(Infinity));

// [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

[**Splitting an array with slice()**](https://codepen.io/cferdinandi/pen/ExEGXdj) **:**

let wizards = ['Merlin', 'Gandalf', 'Ursula', 'Radagast'];

let wizardsFirst = wizards.slice(0, 2); //Merlin, Gandalf

let wizardsSecond = wizards.slice(2); //Ursula, Radagast

console.log(wizardsFirst, wizardsSecond);

**String.replace:**

let str="Obed Gnanam";

let newStr=str.replace("Gnanam", "Sundar");

console.log(newStr); // Obed Sundar

**Array replace/update:**

const arr = ["obed", "Stainz", "leela"];

arr[1] = "Sundar";

console.log(arr);

//Output: obed, Sundar, leela

**String Concatenation:**

let str1="Obed";

let Str2=" Gnanam";

let fullStr=str1.concat(Str2);

document.write(fullStr); // Obed Gnanam

**For Loop in Array:**

The JavaScript for loop is used to **iterate** through the array or **the elements for a specified number of times.**

**Example:**

const cars = ["BMW", "Volvo", "Saab", "Ford", "Fiat", "Audi"];

let text = "";

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

text += cars[i] + "<br>";

}

document.write(text);

**Ex 2:**

for (let i = 0; i < 5; i++) {

console.log(i);

} // Output: 0, 1, 2, 3, 4

**For each Loop in Array:**

The forEach() method is also used to loop through arrays, but it uses a function differently than the classic “for loop”. It passes a **callback function** for each element of an array.

**Example:**

const cars = ["BMW", "Volvo", "Saab", "Ford", "Fiat", "Audi"];

let text = "";

cars.forEach(myFunction);

function myFunction(i) {

text += i + "<br>";

}

document.write(text);

**Ex 2:**

const numbers = [1, 2, 3, 4, 5];

numbers.forEach(number => {

console.log(number);

}); // Output: 1, 2, 3, 4, 5

**For each Loop in Set:**

const cars = new Set (["BMW", "Volvo", "Saab", "Ford", "Fiat", "Audi"]);

let text = "";

cars.forEach(function(value){

text += value + "<br>";

})

document.write(text);

**For each Loop in Map:**

const cars = new Map([

["BMW", 500], ["Volvo", 300], ["Audi", 200]

]);

let text = "";

cars.forEach (function(value, key) {

text += key + " " + value + "<br>"

})

document.write(text);

**For of Loop:**

This for (..of) statement lets you loop over the data structures that are iterable such as **Arrays, Strings, Maps, Node Lists**, and more.

**Example:**

const cars = ["BMW", "Volvo", "Saab"];

let text = "";

var i;

for (i of cars) {

text += i + "<br>";

}

document.write(text);

**Ex 2:**

const numbers = [1, 2, 3, 4, 5];

for (const number of numbers) {

console.log(number);

} // Output: 1, 2, 3, 4, 5

**For in Loop:**

The loop will iterate over all enumerable properties of the **object** itself.

**Example:**

const cars = {1:"BMW", 2:"Volvo", 3:"Saab"};

let text = "";

var i;

for (i in cars) {

text += cars[i] + "<br>";

}

document.write(text);

**While Loop:**

Loops a code block while a **condition is true**.

let text="";

var i=0;

while(i<10){

text+= i + "<br>" ;

i++;

}

document.write(text);

**Do While Loop:**

Loops a code block **once**, and then **while a condition is true.**

let text="";

var i=0;

do{

text+= i + "<br>" ;

i++;

}

while(i<10)

document.write(text);

**Constructor Function:**

function Person(first, last, age) {

this.firstName = first;

this.lastName = last;

this.age = age;

}

const myFather=new Person("Sundar", "Muthusamy", "54");

//const myMother=new Person("Leela", "Sundar", "54");

document.write(+myFather.age);

// output:54

**Fibonacci Series Up to n Terms (For Loop)**

// program to generate fibonacci series up to n terms

// take input from the user

const number = parseInt(prompt('Enter the number of terms: '));

let n1 = 0, n2 = 1, nextTerm;

for (let i = 1; i <= number; i++) {

console.log(n1);

nextTerm = n1 + n2;

n1 = n2;

n2 = nextTerm;

}

## Fibonacci Sequence Up to a Certain Number

// program to generate fibonacci series up to a certain number

// take input from the user

const number = parseInt(prompt('Enter a positive number: '));

let n1 = 0, n2 = 1, nextTerm;

console.log('Fibonacci Series:');

console.log(n1); // print 0

console.log(n2); // print 1

nextTerm = n1 + n2;

while (nextTerm <= number) {

// print the next term

console.log(nextTerm);

}

n1 = n2;

n2 = nextTerm;

nextTerm = n1 + n2;

}

## Pass by Value in JavaScript

let num1 = 70

let num2 = num1

console.log(num1) *// 70*

console.log(num2) *// 70*

num1 = 40

console.log(num1) *// 40*

console.log(num2) *// 70*

## Pass by Reference in JavaScript

let obj1 = {website: "Scaler Academy"}

let obj2 = obj1;

console.log(obj1) *// {website: "Scaler Academy"}*

console.log(obj2) *// {website: "Scaler Academy"}*

obj1.website = "Scaler Topics"

console.log(obj1) *// {website: "Scaler Topics"}*

console.log(obj2) *// {website: "Scaler Topics"}*

**Closure Function:**

<p id="demo"></p>

<script>

myFunction();

function myFunction() {

let a = 4;

document.getElementById("demo").innerHTML = a \* a;

}

</script>//output: 16

**Print the occurrence of the characters in a string:**

**Method 1:**

*//print occurrence of chars*

function print(ans){

for (let [key, value] of ans){

console.log(`${key} occurs ${value} times`);

}

}

*//count occurrence of char*

function count(str,let\_map){

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

let k= let\_map.get(str[i]);

let\_map.set(str[i], k+1);

}

print(let\_map);

}

*//map*

function count\_occurs(test, callback){

if(test.length === 0){

console.log("empty");

return;

}

else {

let ans = new Map();

for (let i=0; i<test.length; i++)

{

ans.set(test[i], 0);

}

callback (test, ans);

}

}

let test = "javascript is programming";

count\_occurs ( test, count);

**Method 2:**

function count\_occur(str){

if (str.length == 0){

console.log("invalid")

return;

}

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

let count = 0;

for (let j=0; j<str.length; j++){

if(str[i] == str[j] && i>j)

{

break;

}

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

count++;

}

}

if (count > 0)

console.log(`${str[i]} occurs ${count} times`);

}

}

let test\_str = "javascript is programming";

count\_occur(test\_str);

**Method 3: Easy way to find one character:**

function test(){

let str= "javascript is programming";

*//let j= (str.match(/j/g)}.length;*

console.log(

*//(str.match(/j/g)).length);*

(str.match(/a/g)).length);

*//(str.match(/v/g)).length);*

}

test();

**Replace Even Letters with Dash:**

function replaceEvenLettersWithDash(inputString) {

if (typeof inputString !== 'string') {

return "Invalid input. Please provide a string.";

}

let result = '';

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

if (i % 2 === 0) {

result += '-';

} else {

result += inputString[i];

}

}

return result;

}

*// Test the function with an example string*

const inputString = "abcdefghijklmnopqrstuvwxyz";

const output = replaceEvenLettersWithDash(inputString);

console.log(output);

**Check Anagram:**

function areAnagrams(str1, str2) {

*// Remove spaces and convert to lowercase*

str1 = str1.replace(/\s/g, '').toLowerCase();

str2 = str2.replace(/\s/g, '').toLowerCase();

*// Sort the characters in both strings and compare*

const sortedStr1 = str1.split('').sort().join('');

const sortedStr2 = str2.split('').sort().join('');

return sortedStr1 === sortedStr2;

}

*// Example usage:*

const str1 = "listen";

const str2 = "silent";

console.log(areAnagrams(str1, str2)); *// Output: true*

**Ascending / Sorting an Array of data by id or age:**

We can sort the array by id or age.

const users = [

{ id: 1, name: "shankar", isActive: true, age: 55 },

{ id: 2, name: "raju", isActive: true, age: 15 },

{ id: 3, name: "pavan", isActive: false, age: 76 },

{ id: 4, name: "shankar", isActive: true, age: 55 },

{ id: 5, name: "ajay", isActive: false, age: 24 },

{ id: 6, name: "bhuvan", isActive: true, age: 77 }

];

const usersSortedAscending = users.slice().sort((a, b) => a.id - b.id);

// const usersSortedAscending = users.slice().sort((a, b) => a.age - b.age);

console.log(usersSortedAscending);

**Coding Challenge:**

(function() {

console.log(1);

setTimeout(function(){console.log(2)}, 1000);

setTimeout(function(){console.log(3)}, 0);

console.log(4);

})();

Output:

1

4

3

2

**Coding Challenge:**

for (var i = 0; i < 3; i++) {

setTimeout(function() { console.log(i); }, 1000 + i);

}

Output:

3

3

3

**Coding Challenge:**

for (var i = 0; i < 3; i++) {

(function (index) {

setTimeout(function() { console.log(index); }, 1000 + index);

})(i);

}

Output:

0

1

2

**To find the second Most Array element:**

function secondMost(arr) {

const frequencyMap = {};

arr.forEach((element)=> {frequencyMap[element] = (frequencyMap[element] || 0)+ 1;

});

const frequencyArray = Object.entries(frequencyMap);

frequencyArray.sort((a,b) => b-a);

if(frequencyArray.length > 1){

return parseInt(frequencyArray[1][0]);

} else {

return "Array doesn't have second largest value"

}

}

const myArray = [5, 2, 8, 1, 2, 6, 8, 8];

const result = secondMost(myArray);

console.log(result);

Output :

2