1. Counting duplicate characters: Write a program that counts duplicate characters from a given string.

let str = "hellowworld";

let count = 0;

for (let i = 0; i < str.length - 1; i++) {

for (let j = i + 1; j < str.length; j++) {

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

count++;

}

}

}

console.log("Number of duplicate characters in the string: " + count);

**2**. Finding the first non-repeated character: Write a program that returns the first non-repeated character from a given string.

function firstNonRepeatedCharacter(string) {

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

let c = string.charAt(i);

if (string.indexOf(c) == i && string.indexOf(c, i + 1) == -1) {

return c;

}

}

return null;

}

console.log(firstNonRepeatedCharacter('aabbccd'));

3.Reversing letters and words: Write a program that reverses the letters of each word and a program that reverses the letters of each word and the words themselves.

Var str=”compare”

Result= str.split(“ “).reverse().join(“”);

Console.log(result);

4. Checking whether a string contains only digits: Write a program that checks whether the given string contains only digits.

function mani(ma) {

return /^\d/.test(ma)

}

console.log(mani("d"))

5. Counting vowels and consonants: Write a program that counts the number of vowels and consonants in a given string. Do this for the English language, which has five vowels (a, e, i, o, and u).

let string = 'Hello World';

let countVowel = 0;

let countConsonant = 0;

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

let char = string.charAt(i).toLowerCase();

if(char == 'a' || char == 'e' || char == 'i' || char == 'o' || char == 'u') {

countVowel++;

}

else if(char >= 'a' && char <= 'z') {

countConsonant++;

}

}

console.log('Number of vowels: ' + countVowel);

console.log('Number of consonants: ' + countConsonant);

6. Counting occurrences of a certain character: Write a program that counts the occurrences of a certain character in a given string.

function countChar(str, char) {

let count = 0;

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

if (str.charAt(i) == char) {

count++;

}

}

return count;

}

let myString = 'Hello World!';

let myChar = 'l';

let charCount = countChar(myString, myChar);

console.log('The character "' + myChar + '" appears ' + charCount + ' times in "' + myString + '".');

7. Converting String into int, long, float, or double: Write a program that converts the given String object (representing a number) into int, long, float, or double.

let n = "hellow";

console.log(typeof n);

let n1 = parseInt("hellow");

console.log(typeof n1)

let n2 = parseFloat("hellow")

console.log(typeof n2);

let n3 = Boolean("hellow")

console.log(typeof n3);

8. Removing white spaces from a string: Write a program that removes all white spaces from the given string. using java script.

const removeWhiteSpaces = (str) => {

return str.replace(/\s/g, '');

}

console.log(removeWhiteSpaces("This is an example"));

9. Joining multiple strings with a delimiter: Write a program that joins the given strings by the given delimiter.

function joinStrings(strings, delimiter){

var result = '';

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

if(i > 0){

result += delimiter;

}

result += strings[i];

}

return result;

}

var strings = ['Hello', 'World', 'How', 'Are', 'You'];

var delimiter = ' ';

console.log(joinStrings(strings, delimiter));

10. Generating all permutations: Write a program that generates all of the permutations of a given string.

function getPermutations(str) {

let result = [];

if (str.length === 1) {

result.push(str);

return result;

}

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

let char = str[i];

let remainingChars = str.slice(0, i) + str.slice(i + 1);

let permutations = getPermutations(remainingChars);

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

result.push(char + permutations[j]);

}

}

return result;

}

// Example usage

let input = "abcd";

console.log("Input string: " + input);

console.log("Permutations: " + getPermutations(input));

11. Checking whether a string is a palindrome: Write a program that determines whether the given string is a palindrome or not.

function isPalindrome(str) {

var reversedString = str.split('').reverse().join('');

return str == reversedString;

}

console.log(isPalindrome('racecar')); // true

console.log(isPalindrome('hello')); // false

12. Removing duplicate characters: Write a program that removes the duplicate characters from the given string.

function removeDuplicates(string) {

let uniqueString = "";

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

if (uniqueString.indexOf(string[i]) == -1) {

uniqueString += string[i];

}

}

return uniqueString;

}

console.log(removeDuplicates("geeksforgeeks"));

13. Removing given characters: Write a program that removes the given character from the given string.

function removeChar(str, char) {

return str.replace(new RegExp(char, 'gi'), '');

}

console.log(removeChar('duplicate', 'd'));

14. Finding the character with the most appearances: Write a program that finds the character with the most appearances in the given string.

function findMostFrequentChar(str) {

let charMap = new Map(); // Create a map to store character frequencies

// Loop through the string and update character frequencies in the map

for (let char of str) {

if (charMap.has(char)) {

charMap.set(char, charMap.get(char) + 1);

} else {

charMap.set(char, 1);

}

}

let maxChar = '';

let maxCount = 0;

// Loop through the map to find the character with the highest frequency

for (let [char, count] of charMap.entries()) {

if (count > maxCount) {

maxChar = char;

maxCount = count;

}

}

return maxChar;

}

// Example usage

let str = "Hello World";

let mostFrequentChar = findMostFrequentChar(str);

console.log("The character with the most appearances is: " + mostFrequentChar);

15. Sorting an array of strings by length: Write a program that sorts by the length of the given array of strings.

let array = ["cocanet", "apple", "mangos", "cat", "dogs"];

array.sort(function (a, b) {

return a.length - b.length;

});

console.log(array);

Exercise 1: Basic Arithmetic (★✩✩✩✩)

Exercise 1a: Basic Arithmetic Operations (★✩✩✩✩)

Write method int calc(int, int) that multiplies two variables, m and n of type int, then divides the product by two, and outputs the remainder with respect to division by 7.

Examples

|  |
| --- |
| m n m \* n m \* n / 2 Result ((n \* m / 2) % 7) |
| 6 7 42 21 0 |
| 3 4 12 6 6 |
| 5 5 25 12 5 |

As a short reminder here: With an integer division, the remainder is truncated. Therefore 25/2 results in the value 12.

**Solution:**

function calc(m, n) {

let product = m \* n;

let quotient = Math.floor(product / 2);

let remainder = quotient % 7;

return remainder;

}

let m1 = 6, n1 = 7;

let result1 = calc(m1, n1);

console.log(`Result for m=${m1}, n=${n1}: ${result1}`);

let m2 = 3, n2 = 4;

let result2 = calc(m2, n2);

console.log(`Result for m=${m2}, n=${n2}: ${result2}`);

let m3 = 5, n3 = 5;

let result3 = calc(m3, n3);

console.log(`Result for m=${m3}, n=${n3}: ${result3}`);