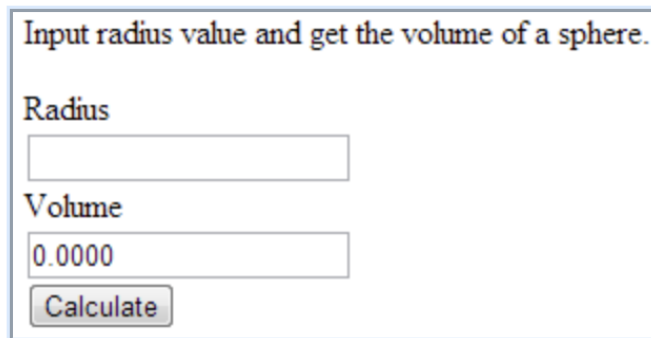


Practise tasks: Mar 28 | March

JS Basics

1. Write a JavaScript program to calculate the volume of a sphere.

Sample Output of the form :



Input radius value and get the volume of a sphere.

Radius

Volume

Calculate

2. There are two arrays with individual values, write a JavaScript program to compute the sum of each individual index value from the given arrays.

Sample array :

array1 = [1,0,2,3,4];

array2 = [3,5,6,7,8,13];

Expected Output :

[4, 5, 8, 10, 12, 13]

3. Write a JavaScript program to check two given numbers and return true if one of the number is 50 or if their sum is 50.
4. Write a JavaScript program to check whether a given integer is within 20 of 100 or 400.
5. Write a JavaScript program to remove a character at the specified position of a given string and return the new string.
6. Write a JavaScript program to create a new string from a given string changing the position of first and last characters. The string length must be greater than or equal to 1.

7. Write a JavaScript program to check whether a given positive number is a multiple of 3 or a multiple of 7.
8. Write a JavaScript program to create a new string from a given string taking the last 3 characters and added at both the front and back. The string length must be 3 or more.
9. Write a JavaScript program to check whether three given integer values are in the range 50..99 (inclusive). Return true if one or more of them are in the said range.
10. Write a JavaScript program to check whether the last digit of the three given positive integers is same.

DSA Basics

1. Check whether the two Binary Search Trees are Identical or Not

Given the root nodes of the two binary search trees. The task is to print 1 if the two Binary Search Trees are identical else print 0. Two trees are identical if they are identical structurally and nodes have the same values.

2. Calculate the height of a binary tree

Write an efficient algorithm to compute the binary tree's height. The height or depth of a binary tree is the total number of edges or nodes on the longest path from the root node to the leaf node.