**Q1: Create an array fruits containing "apple", "banana", and "cherry". Add "orange" to the end of the array and log the updated array.**

**Solution:**

const fruits= ['apple','banana','cherry','orange'];

console.log(fruits);

fruits.push('Kiwi');

console.log(fruits);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q2: Given an array numbers = [10, 20, 30, 40, 50],**

**Write a function that returns the sum of all numbers in the array without using a for or while loop.**

**Solution:**

const arr=[10,20,30,40,50]

function sumofarray(arr) //arr here is Parameter, acts as a placeholder for value that will be passed when function is called

{

    let sum= arr.reduce((acc,val) => acc+val,10);

    console.log(`Sum of Array is: ${sum}`);

}

sumofarray(arr); //arr here is Argument, actual value passed to function when it is called.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q3: Create a function removeDuplicates that takes an array and returns a new array without duplicate elements.**

Example: removeDuplicates([1, 2, 2, 3, 4, 4]) should return [1, 2, 3, 4].

**Solution 1 – ONLY FOR LOGIC BUILDING:**

const arr = [1, 2, 2, 3, 4, 4];

function removeDuplicates(arr) {

    const newArray = []; // Initialize an empty array to store unique elements

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

        // Check if arr[i] is not already in newArray

        let isDuplicate = false;

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

            if (arr[i] === newArray[j]) {

                isDuplicate = true; // Mark as duplicate

                break;

            }

        }

        // If it's not a duplicate, add it to newArray

        if (!isDuplicate) {

            newArray.push(arr[i]);

        }

    }

    console.log(newArray); // Print the array with unique elements

}

removeDuplicates(arr);

**Solution 2 – Easier and JS6:**

const removeDuplicates = (arr) => [...new Set(arr)];

console.log(removeDuplicates([1, 2, 2, 3, 4, 4])); // Output: [1, 2, 3, 4]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**MY TRY**

const arr=[1,1,2,2]

const removee= (arr) => [...new Set(arr)];

console.log(removee(arr));

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_