TASK – 4:

**PROGRAMS IN ANONYMOUS AND IIFE FUNCTIONS**

**1)Print odd numbers in an array:**

1. **Anonymous function:**

//print odd numbers in an array

let oddNumbers = function(nums){

    for(let i = 0;i<nums.length;i++){

        if(nums[i]%2==1){

            console.log(nums[i])

        }

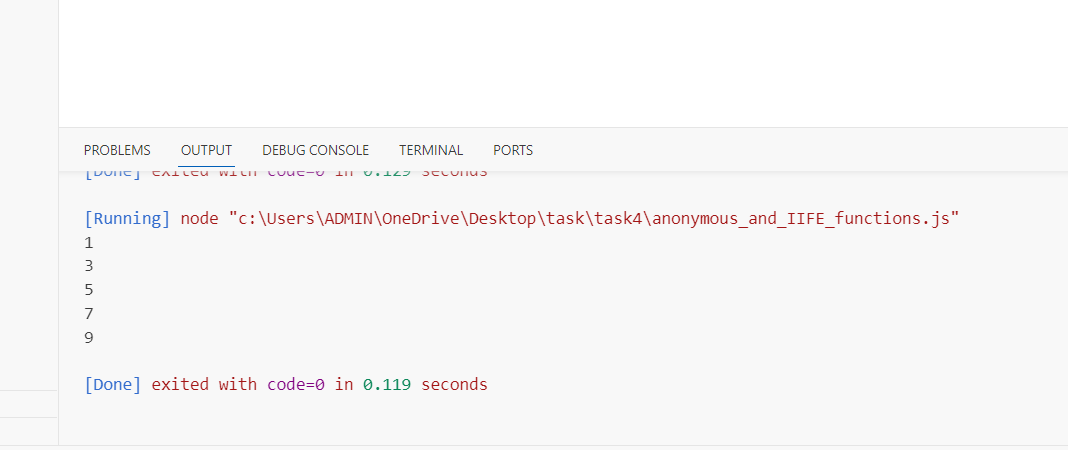
    }

}

let arr = [1,2,3,4,5,6,7,8,9,10];

oddNumbers(arr);

sample output:



**b) IIFE FUNCTION:**

let arr = [1,2,3,4,5,6,7,8,9,10];

((nums) => {

    for(let i = 0;i<nums.length;i++){

        if(nums[i]%2==1){

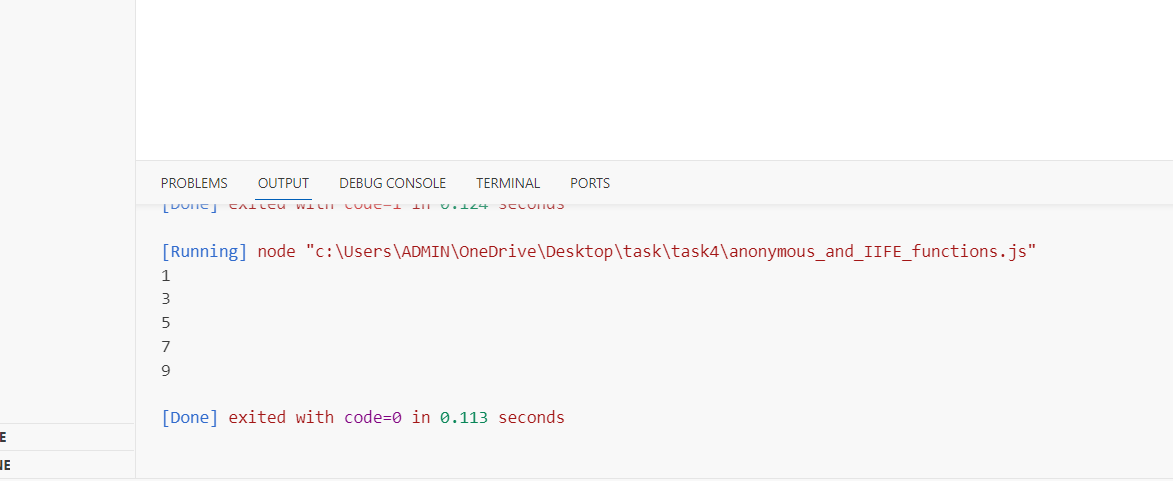
            console.log(nums[i])

        }

    }

})(arr);

**Sample output:**

****

**-----------------------------------------------------------------------------------------------------------------**

**2)Convert all strings into title caps in a string array:**

**A) Anonymous function:**

let  convertToTitleCase = function(stringArray) {

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

      let words = stringArray[i].split(' ');

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

        // Capitalize the first letter of each word and append the rest of the word

        words[j] = words[j].charAt(0).toUpperCase() + words[j].slice(1);

      }

      stringArray[i] = words.join(' ');

    }

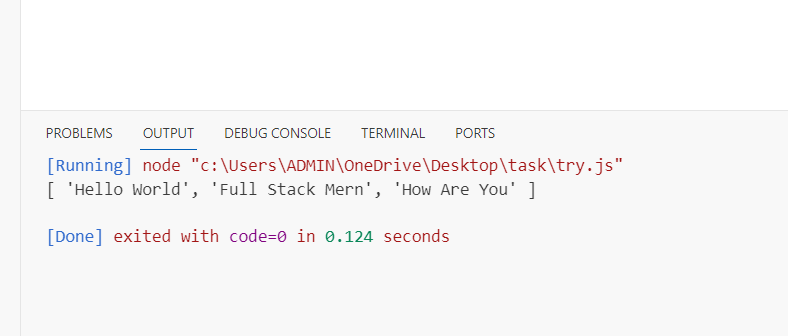
    return stringArray;

}

const stringArray = ["hello world", "full stack mern", "how are you"];

console.log(convertToTitleCase(stringArray));

**output:**

****

1. **IIFE FUNCTION:**

const stringArray = ["hello world", "full stack mern", "how are you"];

(function(strings) {

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

      let words = strings[i].split(' ');

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

        // Capitalize the first letter of each word and append the rest of the word

        words[j] = words[j].charAt(0).toUpperCase()+words[j].slice(1);

      }

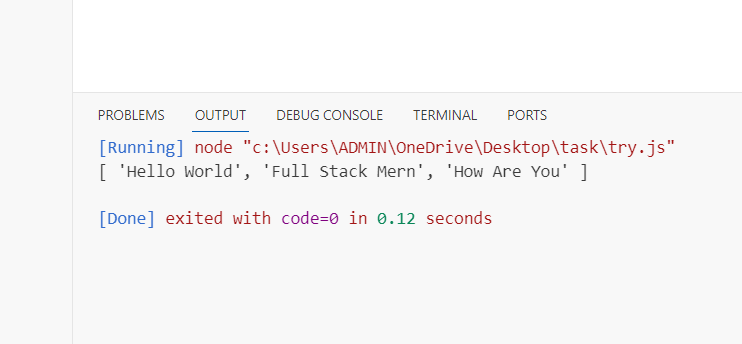
      strings[i] = words.join(' ');

    }

   console.log(strings);

})(stringArray)

**Output:**

****

**3) Sum of all numbers in an array:**

**A) Anonymous function:**

let sum = function(...nums){

    let total = 0;

    for (let i= 0;i<nums.length;i++){

        total += nums[i];

    }

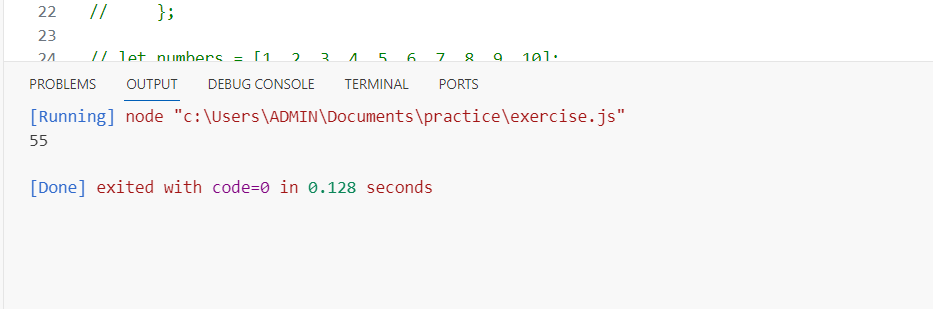
    console.log(total);

};

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

sum(...numbers);

**output:**

****

**B) IIFE FUNCTION:**

((numbers) => {

    let total = 0;

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

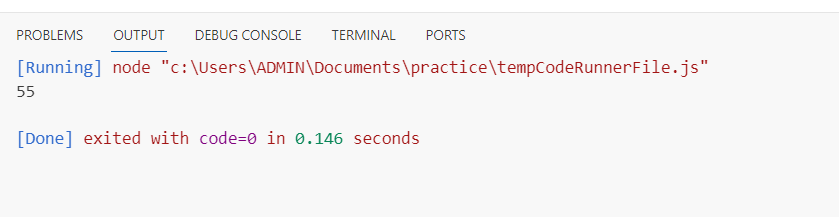
        total += numbers[i];

    }

    console.log(total);

})([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]);

**OUTPUT:**

****

**4) Return all the prime numbers in an array:**

1. **Anonymous function:**

let isPrime = function (num) {

  for (let i = 2; i <= Math.sqrt(num); i++) {

      if (num % i === 0) return false; //

  } // i=2 because prime numbers starts from 2 only

  return true;

}

const numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10,13];

let prime = [];

for (let index = 0; index < numbers.length; index++) {

  if (isPrime(numbers[index])) {

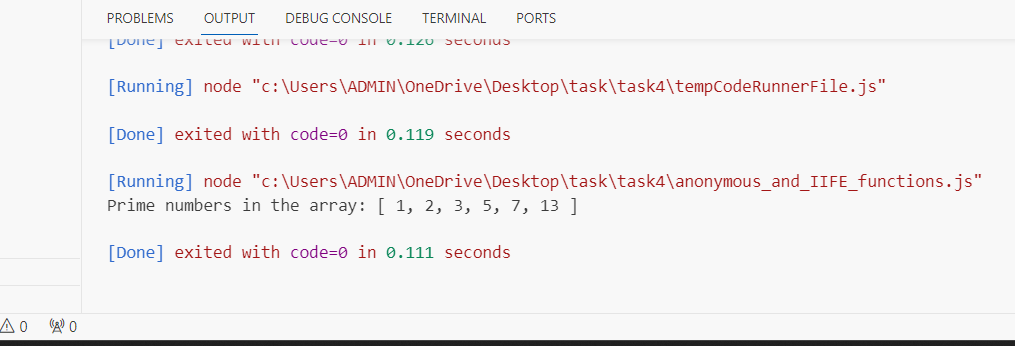
      prime.push(numbers[index])

  }

}

console.log("Prime numbers in the array:", prime);

**Output:**

****

**B) IIFE FUNCTION:**

(()=>{

let isPrime = (num) =>{

  for (let i = 2; i <= Math.sqrt(num); i++) {

      if (num % i === 0) return false; //

  }

  return true;

}

let numbers = [1,2,3,4,5,6,7,8,9,10];

const prime = [];

for (let index = 0; index < numbers.length; index++) {

  if (isPrime(numbers[index])) {

      prime.push(numbers[index])

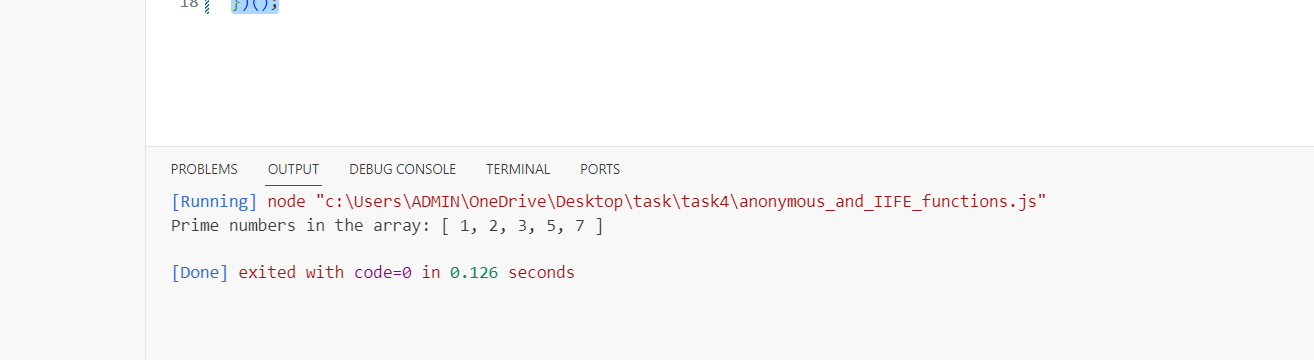
  }

}

console.log("Prime numbers in the array:", prime);

})();

**Output:**

****

**--------------------------------------------------------------------------------------------------------**

**5)Return all the palindromes in an array**

1. **Anonymous function:**

let ispalindrome = function(words){

  let palindromes = [];

for (let word of words){

   let reverse = '';

   for ( let index =word.length-1; index>=0; index--){

       reverse += word[index];

   }

   if (word === reverse){

       palindromes.push(word);

   }

 }

  return palindromes;

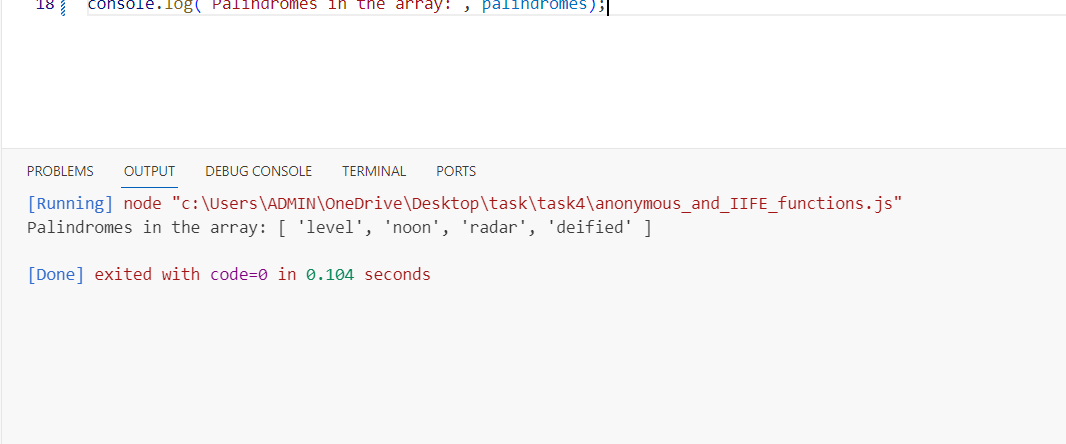
}

const strings = ["level", "hello", "noon", "radar", "world", "deified"];

const palindromes = ispalindrome(strings);

console.log("Palindromes in the array:", palindromes);

**Output:**

****

**B) IIFE FUNCTION:**

console.log((function(words){

   let palindromes = [];

 for (let word of words){

    let reverse = '';

    for ( let index =word.length-1; index>=0; index--){

        reverse += word[index];

    }

    if (word === reverse){

        palindromes.push(word);

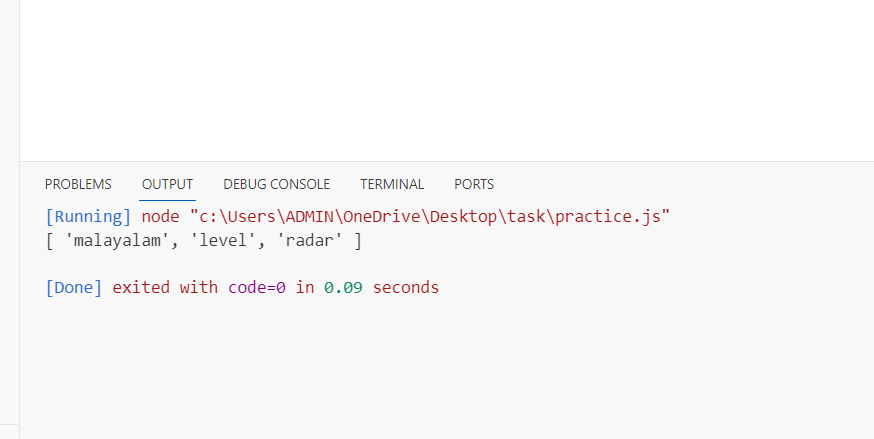
    }

  }

   return palindromes;

})(['malayalam','level','hi','radar']));

**Output:**

****

**-------------------------------------------------------------------------------------------------------**

**6) Return median of two sorted arrays of the same size:**

**A) Anonymous function:**

let MedianOfArr = function(nums1,nums2){

  let mergearr = nums1.concat(nums2).sort((a,b) => a-b)

  const len = mergearr.length;

  let median = 0;

  if(len%2===0){

    let mid1 = len / 2 - 1;

    let mid2 = len / 2;

    median = (mergearr[mid1]+mergearr[mid2])/2

  }

  else{

    let mid = Math.floor(len/2);

   median = mergearr[mid];

    }

return (median);

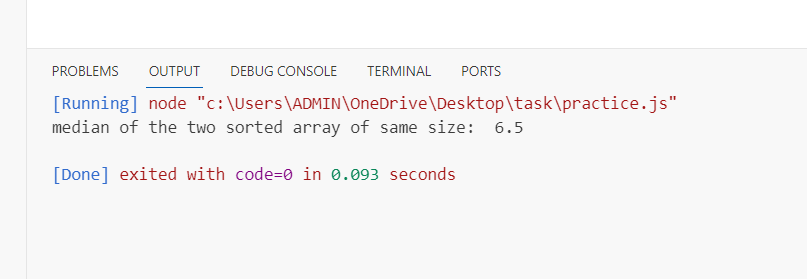
}

let OriginalArr1 = [3,2,1,9];

let OriginalArr2 = [8,10,11,5];

console.log('median of the two sorted array of same size: ',MedianOfArr(OriginalArr1,OriginalArr2));

**Output:**

****

**B) IIFE FUNCTION:**

medianarr = ((nums1,nums2)=>{

  let mergearr = nums1.concat(nums2).sort((a,b) => a-b)

  const len = mergearr.length;

  let median = 0;

  console.log('merge array: ',mergearr)

  if(len%2===0){

    let mid1 = len / 2 - 1;

    let mid2 = len / 2;

    median = (mergearr[mid1]+mergearr[mid2])/2

  }

  else{

    let mid = Math.floor(len/2);

   median = mergearr[mid];

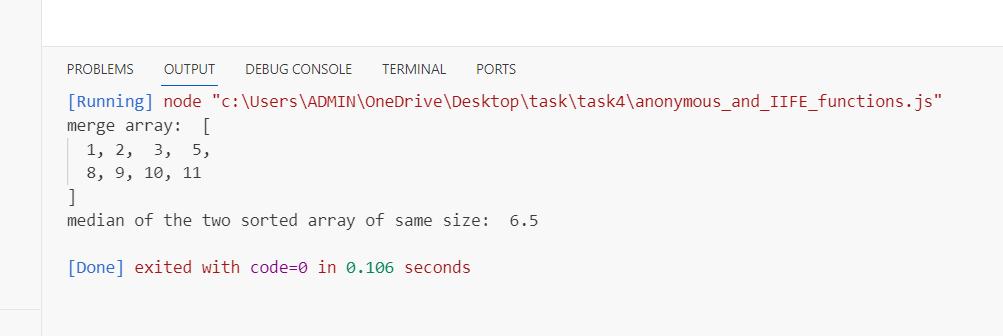
    }

return median;

}) ([3,2,1,9],[8,10,11,5]);

console.log('median of the two sorted array of same size: ',(medianarr));

**Output:**

****

**----------------------------------------------------------------------------------------------------------------**

**7)** **Remove duplicates from an array:**

**A) Anonymous function:**

let getUniqueArray = function(array) {

    const uniqueArray = [];

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

        if (!uniqueArray.includes(array[i])) {

            uniqueArray.push(array[i]);

        }

    }

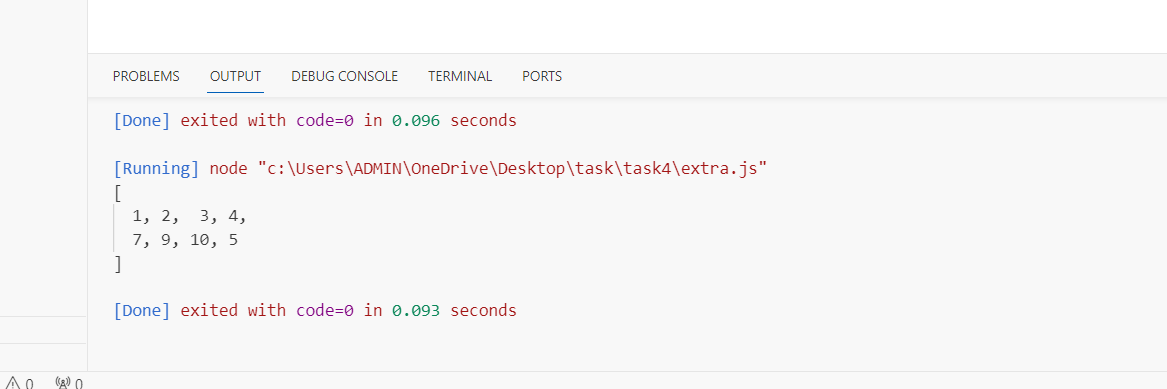
    return uniqueArray;

}

const array = [1, 2, 3, 3, 4,7,9,10 ,5, 5];

console.log(getUniqueArray(array));

**Output:**

****

**B) IIFE FUNCTION:**

let ArrAfterDuplicate = ((array) =>{

    const uniqueArray = [];

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

        if (!uniqueArray.includes(array[i])) {

            uniqueArray.push(array[i]);

        }

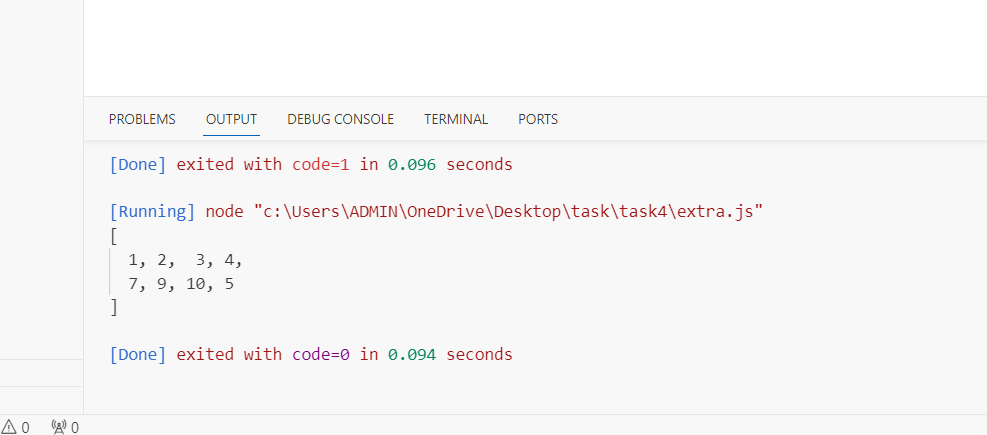
    }

    return uniqueArray;

})([1, 2, 3, 3, 4,7,9,10 ,5, 5]);

console.log(ArrAfterDuplicate)

**Output:**

****

**8)Rotate an array by k times**

**A)Anonymous function:**

let RotateArr = function(nums,k){

  for (i=0;i<nums.length;i++){

   return nums.slice(k).concat(nums.slice(0, k));

  }

}

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

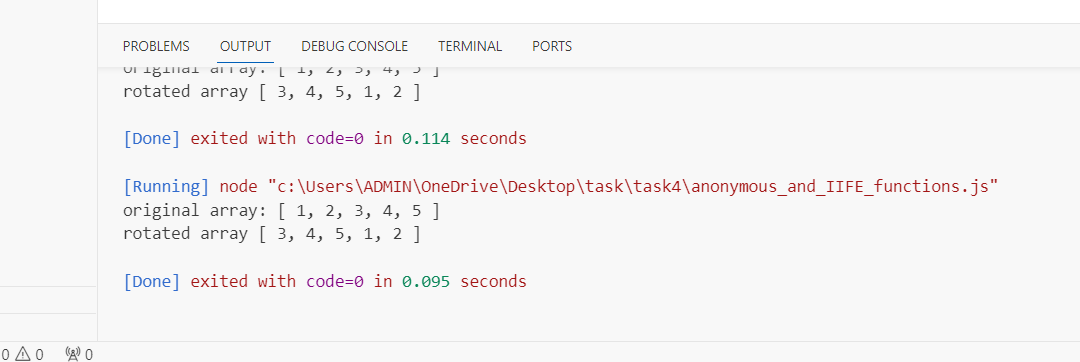
let k = 2;

let RotatedArray = RotateArr(OriginalArr,k);

console.log('original array :',OriginalArr);

console.log('rotated array :',RotatedArray);

**OUTPUT:**

****

**B) IIFE FUNCTION:**

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

let k = 2;

((nums,k)=>{

  for (i=0;i<nums.length;i++){

    RotatedArray= nums.slice(k).concat(nums.slice(0, k));

    return RotatedArray;

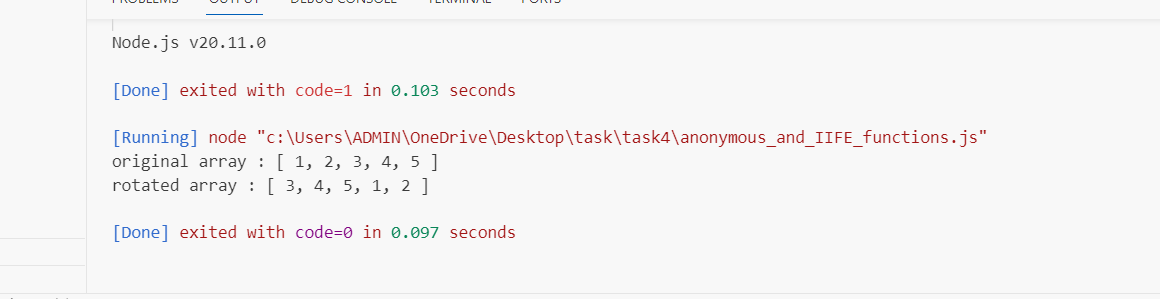
  }

})(OriginalArr,k)

console.log('original array :',OriginalArr);

console.log('rotated array :',RotatedArray);

**Output:**

****

**-----------------------------------------------------------------------------------------------------------------**

**6) Return median of two sorted arrays of the same size:**