1. **Do the below programs in anonymous function & IIFE**
2. ***Print odd numbers in an array***

***SOLUTION :***

(function (arr){

let oddNum =[] // creating an empty array called oddNum

for(let i=0; i<arr.length; i++) // iterating the all array elements

{

if(arr[i]%2==1) // condition to check the odd numbers or not

{

oddNum.push(arr[i]) // if it is odd number pushing the values to that empty array

}

}

console.log(oddNum.join(" ")) // since the elements stored in a array, we are using join()

})([2,4,11,5,7,10])

**OUTPUT: 11 5 7**

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

1. ***Convert all the strings to title caps in a string array***

*(function (str){*

*str = str.toLowerCase().split(" ")*

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

*{*

*str[i] = str[i].charAt(0).toUpperCase() + str[i].slice(1)*

*}*

*console.log(str.join(" "))*

*})("welcome to guvi")*

##### **Output:**

Welcome To Guvi

1. **Sum of all numbers in an array**

*let add = function(arr)*

*{*

*let sum =0;*

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

*{*

*sum = sum + parseInt(arr[i]);*

*}*

*return sum;*

*}*

*console.log(add([1,2,3,4,5]))*

##### **Output:**

15

1. **Return all the prime numbers in an array**

const array = [...Array(20).keys()]

let results = function(num){

for(let start =2; num>start; start++)

{

if(num%start==0){

return false;

}

}

return num>1;

}

console.log(array.filter(results))

##### **Output:**

[ 2, 3, 5, 7, 11, 13, 17, 19 ]

--------------------------------------------------------------------

1. **Return all the palindromes in an array**

const arr = ['carecar', 1344, 12321, 'did', 'cannot'];

const isPalindrome = function(el) {

const str = String(el);

let i = 0;

let j = str.length - 1;

while(i < j) {

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

i++;

j--;

}

else {

return false;

}

}

return true;

};

console.log(arr.filter(isPalindrome));

##### **Output:**

[ 12321, 'did' ]

-----------------------------------------------------------------------------------------------------------------

1. Return median of two sorted arrays of same size

function getMedian(ar1, ar2, n)

{

var i = 0; /\* Current index of i/p array ar1[] \*/

var j = 0; /\* Current index of i/p array ar2[] \*/

var count;

var m1 = -1, m2 = -1;

/\* Since there are 2n elements, median will be average

of elements at index n-1 and n in the array obtained after

merging ar1 and ar2 \*/

for (count = 0; count <= n; count++)

{

/\*Below is to handle case where all elements of ar1[] are

smaller than smallest(or first) element of ar2[]\*/

if (i == n)

{

m1 = m2;

m2 = ar2[0];

break;

}

/\*Below is to handle case where all elements of ar2[] are

smaller than smallest(or first) element of ar1[]\*/

else if (j == n)

{

m1 = m2;

m2 = ar1[0];

break;

}

/\* equals sign because if two

arrays have some common elements \*/

if (ar1[i] <= ar2[j])

{

m1 = m2; /\* Store the prev median \*/

m2 = ar1[i];

i++;

}

else

{

m1 = m2; /\* Store the prev median \*/

m2 = ar2[j];

j++;

}

}

return (m1 + m2)/2;

}

/\* Driver program to test above function \*/

var ar1 = [1, 12, 15, 26, 38];

var ar2 = [2, 13, 17, 30, 45];

var n1 = ar1.length;

var n2 = ar2.length;

if (n1 == n2)

console.log("Median is "+ getMedian(ar1, ar2, n1));

else

console.log("Doesn't work for arrays of unequal size");

##### **Output:**

Median is 16

1. **Remove duplicates from an array**

(function(arr){

let uniqueChars = [...new Set(arr)]

console.log(uniqueChars)

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

##### **Output:**

**[ 1, 2, 3, 4, 5 ]**

1. **Rotate an array by k times**

(function(arr,k){

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

arr.unshift(arr.pop())

}

console.log(arr)

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

##### **Output:**

**[ 4, 5, 1, 2, 3 ]**

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