ANONYMOUS FUNCTIONS & IIFE

**1.a** let a=[0,1,2,3,4,5,6,7,8,9];

(function () {

let odd= a. filter(num => num %2 ==1);

return odd;

})();

**1.b** function sentenceCase(str) {

    if ((str === null) || (str === ''))

        return false;

    else

        str = str.toString();

    return str.replace(/\w\S\*/g,

        function (txt) {

            return txt.charAt(0).toUpperCase() +

                txt.substr(1).toLowerCase();

        });

}

console.log(sentenceCase(‘guvi’));

**1.c** let myNums = [1, 2, 3, 4, 5];

let sum = 0;

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

sum += myNums[i];

}

console.log(sum);

**1.d** function sort\_prime(num) {

var prime\_num1 = [],

prime\_num2 = [];

for (var i = 0; i <= num; i++) {

prime\_num2.push(true);

}

for (var i = 2; i <= num; i++) {

if (prime\_num2[i]) {

prime\_num1.push(i);

for (var j = 1; i \* j <= num; j++) {

prime\_num2[i \* j] = false;

}

}

}

return prime\_num1;

}

console.log(sort\_prime(5));

console.log(sort\_prime(11));

console.log(sort\_prime(19));

**1.e** function isPalindrome(s)

{

     let a = s;

     s = s.split('').reverse().join('');

return s == a;

}

function PalindromicStrings(arr,N)

{

    let ans = [];

    for (let i = 0; i < N; i++) {

   if (isPalindrome(arr[i])) {

  ans.push(arr[i]);

        }

    }

    return ans;

}

 let arr = [ "abc", "car", "ada", "racecar", "cool" ];

let N = arr.length;

 let s = PalindromicStrings(arr, N);

if(s.length == 0)

    document.write("-1");

for(let st of s)

    document.write(st," ");

**1.f** 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;

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

    {

        if (i == n)

        {

            m1 = m2;

            m2 = ar2[0];

            break;

        }

else if (j == n)

        {

            m1 = m2;

            m2 = ar1[0];

            break;

        }

        if (ar1[i] <= ar2[j])

        {

            m1 = m2;

            m2 = ar1[i];

            i++;

        }

        else

        {

            m1 = m2;

            m2 = ar2[j];

            j++;

        }

    }

    return (m1 + m2)/2;

}

 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)

    document.write("Median is "+ getMedian(ar1, ar2, n1));

else

    document.write("Doesn't work for arrays of unequal size");

**1.g** let arr = ["apple", "mango", "apple",

          "orange", "mango", "mango"];

function removeDuplicates(arr) {

    return arr.filter((item,

        index) => arr.indexOf(item) === index);

}

console.log(removeDuplicates(arr));

**1.h** function RightRotate(a, n, k)

{

   k = k % n;

  for (let i = 0; i < n; i++) {

        if (i < k) {

             document.write(a[n + i - k] + " ");

        }

        else {

   document.write((a[i - k]) + " ");

        }

    }

    document.write("<br>");

}

  let Array = [1, 2, 3, 4, 5];

let N = Array.length;

let K = 2;

  RightRotate(Array, N, K);

ARROW FUNCTIONS:

**2.a**

|  |
| --- |
| oddNumbers = (array) => { |
|  | for(var i = 0 ; i< array.length ; i++){ | |
|  | if(array[i]%2!=0){ | |
| console.log(array[i]) | |
|  | } | |
|  | } | |
|  | } | |

**2.b** function converttotitlecase(str) {

If (!str) {

Return “”

}

Return str.toLowerCase().replace(/\b\w\g,s => s.toUpperCase());

}

Console.log(converttotitlecase(‘this is my task4’));

**2.C** const sum = [1, 2, 3].reduce((partialSum, a) => partialSum + a, 0);

console.log(sum);

**2.d**

|  |
| --- |
|  |
|  | var a=[ 5,9, 63, 29, 35,6,55,23]  var prime = []; |
|  |
|  | function isPrime(item) { |
|  | var identifier = item / 2; |
|  | for (var j = 2; j <= identifier; j++) { |
|  | if ((item % j) == 0) { // modulous |
|  | return false; |
|  | } |
|  | } |
|  | return true; |
|  | } |
|  | for (var index = 0; index < a.length; index++) { |
|  | if (isPrime(a[index])) { |
|  | prime.push(a[index]) |
|  | } |
|  | } |
|  | console.log(prime); |

**2.e** const getAllPalindromes = (words) => {

return words.filter((word) => {

word.split("").reverse().join("") === word;

});

};

console.log(getAllPalindromes(["hello", "noon"]));