**JAVASCRIPT TASK – 3**

1. **Print odd numbers in an array**

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var arr1= [3,4,5,6,7,5,4,3,2]**

**var oddnum = function(array){**

**for(var i in array){**

**if(array[i]%2 !== 0)**

**{**

**console.log(array[i]);**

**}**

**}**

**}**

**oddnum([1,2,3,4,5]);**

**});**

**Output:**

1

3

5

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

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var uppercase= str =>{**

**return str.split(" ").map(word =>**

**{**

**return (word.charAt(0).toUpperCase() + word.slice(1));**

**}).join(" ");**

**}**

**console.log(uppercase("guvi javascript classes"));**

**});**

##### Output:

Guvi Javascript Classes

##### Sum of all numbers in an array

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var sum = arr =>**

**{return arr.reduce((accm,curntvalue)=> {**

**return accm + curntvalue;**

**},0);**

**}**

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

**});**

##### Output:

15

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

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var prime = arr => arr.filter(a => {**

**for(var i=2; i<a; i++){**

**if(a%i === 0){**

**return false;**

**}**

**}**

**return a>1;**

**});**

**console.log(prime([2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]))**

**});**

##### Output:

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

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

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var palindrome = arr => arr.filter(a => {**

**var str = String(a);**

**var i=0;**

**var j= str.length-1;**

**while(i<j){**

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

**i++;**

**j--;**

**}**

**else {**

**return false;**

**}**

**}**

**return true;**

**});**

**console.log(palindrome(["malayalam" , 99 , "mom", "happy", 909, 198]))**

**});**

##### Output:

[ 'malayalam', 99, 'mom', 909 ]

1. **Return median of two sorted arrays of same size**

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**let med = (arr,arr1) =>**

**{**

**arr2 = arr.concat(arr1).sort(function(a,b){return (a-b)});**

**var str = arr2.length;**

**if((arr2.length%2)==0)**

**{**

**var i = arr2.length/2;**

**var median = (arr2[i]+arr2[i-1])/2;**

**}**

**else if((arr2.length%2)!== 0)**

**{**

**median= arr[arr.length/2];**

**}**

**return median;**

**}**

**console.log(med([1, 12, 15, 26, 38],[2, 13, 17, 30, 45]).toFixed(2));**

**});**

##### Output:

16.00

1. **Remove duplicates from an array**

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var nodup = function(a)**

**{**

**var b = a.slice(0,a.length)**

**var c=[];**

**for (var i in a)**

**{**

**var count=0;**

**for( var j in b)**

**{**

**if(a[i]==b[j])**

**{**

**count+=1;**

**delete b[j]**

**if(count==1)**

**{**

**c.push(a[i])**

**}**

**}**

**}**

**}**

**return c ;**

**}**

**console.log(nodup ([1,1,1,2,2,3,4,5,6,6,7]))**

**});**

##### Output:

[ 1, 2, 3, 4, 5, 6, 7 ]

1. **Rotate an array by k times and return the rotated array**

**// Getting input via STDIN**

**const readline = require("readline");**

**const inp = readline.createInterface({**

**input: process.stdin**

**});**

**const userInput = [];**

**inp.on("line", (data) => {**

**userInput.push(data);**

**});**

**inp.on("close", () => {**

**var rotate = function(a,k)**

**{**

**for(var i=1 ; i<=k ; i++)**

**{**

**a.unshift(a[(a.length)-1])**

**a.pop();**

**}**

**return a;**

**}**

**console.log(rotate([1,2,4,5,6],2))**

**});**

##### Output:

[ 5, 6, 1, 2, 4 ]