# ANONYMOUS FUNCTION/ IIFE TASK

Answer

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

(function(arr){

var arr=[];

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

{

if(a[i]%2==0)

arr.push(a[i]);

}

console.log(“ODD numbers" + arr);

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

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

(function (tit) {

var stt = tit.split(' ');

var arst = [];

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

{

arst.push(stt[i][0].toUpperCase() + stt[i].slice(1, stt[i].length));

}

console.log("Capital string" + arst.join(' '));

})("i have a beautiful flower”);

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

(function (arr) {

var sum=0;

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

{

sum=sum+arr[i];

}

console.log("Sum of array elements" + sum);

} )([2,4,6,8]);

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

(function (arr) {

var prime=[];

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

{

var flag = true;

if (arr[i] == 1 || arr[i] == 0 || arr[i]<0) {

flag = false;

}

else {

for(let j = 2; j <= arr[i]/2; j++) {

if (arr[i] % j == 0) {

flag = false;

break;

}

}

}

if(flag==true)

prime.push(arr[i]);

}

console.log("Prime elements of array " +prime);

})([2,4,6,8]);

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

(function (arr) {

var pal=[];

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

{

var flag = true;

for(let j = 0; j < arr[i].length / 2; j++) {

if(arr[i][j] !== arr[i][arr[i].length - j - 1]){

flag = false;

break;

}

}

if(flag==true)

pal.push(arr[i]);

}

}

console.log("Palindrome elements of array" + pal);

})(['abc','AAA','ABBA','lkjd']);

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

(function (arr,arr2) {

var med=0;

for(let i=0;i<arr2.length;i++)

{

arr.push(arr2[i]);

}

arr.sort();

if(arr.length%2 ==0)

med=(arr[arr.length/2-1] +arr[arr.length/2])/2;

else

med = arr[(arr.length)/2-1];

console.log("Median of elements of array " +med);

} )([2,4,6,8],[0,1,2,3]);

1. **Remove duplicates from an array.**

(function (arr) {

var dist=[];

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

{

if(dist.indexOf(arr[i]) == -1)

dist.push(arr[i]);

}

console.log("Distinct elements of array " + dist);

})([2,8,5,4,6,8]);

1. **Rotate an array by K times and return the array.**

(function (arr,n) {

for(let i=0;i<n;i++)

{

var temp = arr[0];

for (let j = 0; j < arr.length - 1; j++)

arr[j] = arr[j + 1];

arr[arr.length-1] = temp;

}

console.log("Rotated array" + arr);

})([2,10,5,4,6,8],6);

# 2. <https://medium.com/@reach2arunprakash/guvi-zen-class-javascript-warm-up-programming-problems-15973c74b87f>

**Problem**:

Write a function called “addFive”.  
Given a number, “addFive” returns 5 added to that number.  
  
Input:

addFive(5);  
addFive(0);  
addFive(-5);

Output:

10  
5  
0

var num = 10;

function addFive(num) {

**return(num+5);**}

var result = addFive(num)

**Problem**:

Write a function called “getOpposite”.  
Given a number, return its opposite

Input:

getOpposite(5);  
getOpposite(0);  
getOpposite(-5);  
getOpposite(“5a”);  
getOpposite(5.5);

Output:

-5  
0  
5  
-1  
-1

var num = 5;

function getOpposite(num) {

**if(num%1 == 0}**

**retrun num\*-1;**

**else**

**retrun -1;**

var result = getOpposite(num)

**Problem**:

Fill in your code that takes an number minutes and converts it to seconds.

Examples  
toSeconds(5) ➞ 300

toSeconds(3) ➞ 180

toSeconds(2) ➞ 120

var min = 5;

function toSeconds(min) {

**return min\*60**

}

var secs = toSeconds(min)

**Problem**  
Create a function that takes a string and returns it as an integer.

Examples  
toInteger(“6”) ➞ 6

toInteger(“1000”) ➞ 1000

toInteger(“12”) ➞ 12

var mystr = "5";

function toInteger(mystr) {

**return Number(mystr);**

}

var myint = toInteger(mystr)

**Problem**

Create a function that takes a number as an argument, increments the number by +1 and returns the result.

Examples  
nextNumber(0) ➞ 1

nextNumber(9) ➞ 10

nextNumber(-3) ➞ -2

var myint = 0;

function nextNumber(myint) {

**return myint+1;**

}

var myNextint = nextNumber(myint)

**Problem**

Create a function that takes an array and returns the first element.

Examples  
getFirstElement([1, 2, 3]) ➞ 1

getFirstElement([80, 5, 100]) ➞ 80

getFirstElement([-500, 0, 50]) ➞ -500

var arr = [1, 2, 3];

function getFirstElement(arr) {

**return arr[0];**

}

var data = getFirstElement(arr)

**Problem**

Convert Hours into Seconds

Write a function that converts hours into seconds.

Examples  
hourToSeconds(2) ➞ 7200

hourToSeconds(10) ➞ 36000

hourToSeconds(24) ➞ 86400

var arr = [1, 2, 3];

function hourToSeconds(arr) {

**return arr.map((e)=>e\*3600));**

}

var data = hourToSeconds(arr)

**Problem**

Find the Perimeter of a Rectangle  
Create a function that takes height and width and finds the perimeter of a rectangle.

Examples  
findPerimeter(6, 7) ➞ 26

findPerimeter(20, 10) ➞ 60

findPerimeter(2, 9) ➞ 22

function findPerimeter(num1,num2) {

**retrun 2\*(num1+num2);**

}

var peri = findPerimeter(6,7)

**Problem**

Less Than 100?  
Given two numbers, return true if the sum of both numbers is less than 100. Otherwise return false.

Examples  
lessThan100(22, 15) ➞ true  
// 22 + 15 = 37

lessThan100(83, 34) ➞ false  
// 83 + 34 = 117

function lessThan100(num1,num2) {

**return num1+num2 < 100;**

}

var res = lessThan100(22,15)

**Problem**

There is a single operator in JavaScript, capable of providing the remainder of a division operation. Two numbers are passed as parameters. The first parameter divided by the second parameter will have a remainder, possibly zero. Return that value.

Examples  
remainder(1, 3) ➞ 1

remainder(3, 4) ➞ 3

remainder(-9, 45) ➞ -9

remainder(5, 5) ➞ 0

function remainder(num1,num2) {

**return num1%num2;**

}

var res = remainder(1,3)

**Problem**

Old macdonald had a farm:

MacDonald is asking you to tell him how many legs can be counted among all his animals. The farmer breeds three species:

turkey = 2 legs  
horse = 4 legs  
pigs = 4 legs

The farmer has counted his animals and he gives you a subtotal for each species. You have to implement a function that returns the total number of legs of all the animals.

Examples  
CountAnimals(2, 3, 5) ➞ 36

CountAnimals(1, 2, 3) ➞ 22

CountAnimals(5, 2, 8) ➞ 50

function CountAnimals(tur,horse,pigs) {

return **(tur\*2 + horse\*4 + pigs\*4);**

}

var legs = CountAnimals(2,3,5)

**Problem**

Frames Per Second  
Create a function that returns the number of frames shown in a given number of minutes for a certain FPS.

Examples  
frames(1, 1) ➞ 60

frames(10, 1) ➞ 600

frames(10, 25) ➞ 15000

function frames(num1,num2) {

**return (num1\*num2\*60);**

}

var fps = frames(1,2)

**Problem**

Check if an Integer is Divisible By Five  
Create a function that returns true if an integer is evenly divisible by 5, and false otherwise.

Examples  
divisibleByFive(5) ➞ true

divisibleByFive(-55) ➞ true

divisibleByFive(37) ➞ false

function divisibleByFive(num1) {

**return num1%5 ==0;**

}

var divisible = divisibleByFive(5)

**Problem**:

Write a function called “isEven”.  
Given a number, “isEven” returns whether it is even.  
  
Input:  
isEven(12);  
isEven(0);  
isEven(11);  
isEven(“11h”);

Output:

true  
true  
false  
-1

function isEven(num){  
 // your code here

**if(num%1 ==0)**

**return num%2==0;**

**else**

**return false;**  
}var even = isEven(5)

**Problem**:  
Write a function called “areBothOdd”.  
Given 2 numbers, “areBothOdd” returns whether or not both of the given numbers are odd.  
  
Input:  
areBothOdd(1, 3);  
areBothOdd(1, 4);  
areBothOdd(2, 3);  
areBothOdd(0, 0);

Output:

true  
flase  
flase  
flase

function areBothOdd(num1, num2){  
 // your code here

**return (num1%2==0 && num2%2==0);**  
}

**Problem**:  
Write a function called “getFullName”.  
Given a first and a last name, “getFullName” returns a single string with the given first and last names separated by a single space.  
  
Input:

getFullName(“GUVI”, “GEEK”);  
getFullName(“GUVI”, );  
getFullName(, “GEEK”);  
getFullName(“”, “”);

Output:

“GUVI GEEK”  
“GUVI”  
“GEEK”  
“”

function getFullName(firstName, lastName){  
 // your code here

**return firstName+lastName;**  
}

**Problem**:  
Write a function called “getLengthOfWord”.  
Given a word, “getLengthOfWord” returns the length of the given word.  
Input:

getLengthOfWord(“GUVI”);  
getLengthOfWord(“”);  
getLengthOfWord();  
getLengthOfWord(9);

Output:

4  
0  
-1  
-1

function getLengthOfWord(word1){

// your code here

**if( word1 ==’’|| typeof(word1)== ‘number’)**

**return -1;**

**else**

**return word1.length;**  
   
}

**Problem**:  
Write a function called “isSameLength”.  
Given two words, “isSameLength” returns whether the given words have the same length.  
Input:  
isSameLength(“GUVI”, “GEEK”);  
Output:  
true

function isSameLength(word1, word2){  
 // your code here

**return word1.length == word2.length;**  
}

**Problem**:

Create a function to calculate the distance between two points defined by their x, y coordinates

**console.log(getDistance(100, 100, 400, 300));**

**function getDistance(x1, y1, x2, y2)  
{  
 return Math.sqrt((Math.pow(x2-x1,2))+(Math.pow(y2-y1,2)));  
}**

**Problem**:

Write a function called “getNthElement”.  
Given an array and an integer, “getNthElement” returns the element at the given integer, within the given array. If the array has a length of 0, it should return ‘undefined’.  
Input:  
getNthElement([1, 3, 5], 1);  
Output:  
3

**function getNthElement(array,n){  
 // your code here**

**if(array.length ==0 || n>array.length-1)**

**return undefined;**

**else**

**return array.slice(n,n+1).join(‘’);  
}**

**Problem**:

Write a function called “getLastElement”.  
Given an array, “getLastElement” returns the last element of the given array. If the given array has a length of 0, it should return ‘-1’.  
Input:  
getLastElement([1, 2, 3, 4]);  
Output:  
4

function getLastElement(array){  
 // your code here

**if(array.length ==0)**

**return -1;**

**else**

**return array[array.length-1];**  
}

**Problem**:

Write a function called “getProperty”.   
Given an object and a key, “getProperty” returns the value of the property at the given key. If there is no property at the given key, it should return undefined.  
  
var obj = {  
mykey: “value”  
};

Input:  
getProperty(obj,’mykey’);  
getProperty(obj,’dummykey’);  
Output:  
value  
NA

var obj = {  
 mykey: “value”  
};

function getProperty(obj, key) {  
 // your code here

**if(obj[key])**

**return obj.key;**

**else**

**return ‘NA’;**  
}

**Problem**:

Write a function called “addProperty”.  
Given an object and a key, “addProperty” adds a new property on the given object with a value of true.  
  
var obj = {  
}  
Input:  
addProperty(obj, “mykey”);

Output:

{  
mykey: true  
}

var obj = {  
 mykey: “value”  
};

function addProperty(obj, key){  
 // your code here

**obj[key] = “value”;**

**return obj;**

}

**Problem**:

Write a function called “removeProperty”.  
Given an object and a key, “removeProperty” removes the given key from the given object.  
Input:  
removeProperty(obj, “name”);  
Output:  
undefined

function removeProperty(obj, key){  
 // your code here

**delete obj[key]**;  
}

**Problem**:

Return an array, where the first element is the count of positives numbers and the second element is sum of negative numbers.

var arr = [-5, 10, -3, 12, -9, 5, 90, 0, 1];

var ar2 = function countPositivesSumNegatives(arr) {  
 **let count1=0,coun2=0;**

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

**{**

**if(arr[i]>0)**

**count1++;**

**else(arr[i]<0)**

**count2++;**

**}**

**return ([count1,count2]);**  
}

console.log(ar2);

**Problem**:

Create a function that receives an array of numbers and returns an array containing only the positive numbers

function getPositives(ar)  
{  
 // your code here

**let arr=[];**

**arr=ar.filter((e)=> e>0);**

**return arr;**  
}

var ar = [-5, 10, -3, 12, -9, 5, 90, 0, 1];  
var ar2 = getPositives(ar);

console.log(ar2);

**Problem**:

Write a function `powersOfTwo` which will return list of all powers of 2 from 0 to n (where n is an exponent).

n = 0 -> 2⁰ -> [1]

n = 1 -> 2⁰, 2¹ -> [1,2]

n = 2 -> 2⁰, 2¹, 2² -> [1,2,4]

Input:  
powersOfTwo(0)  
powersOfTwo(1)  
powersOfTwo(2)  
Output:  
1  
1,2  
1,2,4

function powersOfTwo(n){

**let res=[];**

**for(let i=0;i<=n;i++)**

**{**

**res.push(Math.pow(2,i));**

**}**  
 return res;  
}

**Problem**:

Find the maximum number in an array of numbers

function findMax(ar)  
{  
// your code here

**return Math.max(…ar);**

}

var ar = [-5, 10, -3, 12, -9, 5, 90, 0, 1];  
var max = findMax(ar);  
console.log(“Max: “, max);

**Problem**:

Print the first 100 prime numbers

printPrimes(100);

// Function prints the first nPrimes numbers

function printPrimes(nPrimes){  
 var n = 0;  
 var i = 2;  
   
 while(n < nPrimes) {  
 if (isPrime(i)){  
 console.log(n, “ → “, i);  
 n++; }  
 i++; }  
}// Returns true if a number is prime  
function isPrime(n){  
 // your code here

if(n==1||n==0)

 return false;

 for(let i=2; i\*i<=n; i++){

   if(n%i==0)return false;

 }

 return true;

}

**Problem**:

Create a function that will return in an array the first “nPrimes” prime numbers greater than a particular number “startAt”

console.log(getPrimes(10, 100));

function getPrimes(nPrimes, startAt)  
{// your code here

**let arr=[],i=startAt+1;**

**while(arr.length<nPrimes){**

**if(isPrime(i)){**

**arr.push(i);**

**}**

**i++;**

**}**

**return arr;**  
}

// Returns true if a number is prime  
function isPrime(n)  
{  
 // your code here

if(n==1||n==0)

 return false;

 for(let i=2; i\*i<=n; i++){

   if(n%i==0)return false;

 }

 return true;

}

**Problem**:

Reverse a string

var s = reverseString("JavaScript");  
console.log(s);

function reverseString(s)  
{  
 // your code here

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

}

**Problem**:

Create a function that will merge two arrays and return the result as a new array

var ar1 = [1, 2, 3];  
var ar2 = [4, 5, 6];

var ar = mergeArrays(ar1, ar2);  
console.log(ar);

function mergeArrays(ar1, ar2)  
{  
 var result = [];//this will add the first array to the result array  
for(let el of ar1)  
 {  
 result.push(el);  
 }  
 **result=[…result,…ar2];**  
 //Some piece of code goes here   
   
 return result;  
}

**Problem**:

Calculate the sum of numbers received in a comma delimited string

console.log(sumCSV(“1.5, 2.3, 3.1, 4, 5.5, 6, 7, 8, 9, 10.9”));

function sumCSV(s)  
{  
 // your code here

**let arr=s.split(',');**

**return arr.reduce((a,b)=> Number(a)+Number(b),0);**

}

# 3. Arrow functions

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

var odd\_num => (a){

var arr=[];

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

{

if(a[i]%2==0)

arr.push(a[i]);

}

return arr;

}

console.log("ODD numbers" + odd\_num([1,2,3,4,5,6,7]));

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

var title\_str => (tit) {

var stt = tit.split(' ');

var arst = [];

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

{

arst.push(stt[i][0].toUpperCase() + stt[i].slice(1, stt[i].length));

}

return arst.join(' ');

}

console.log("Capital string" + title\_str("i have a beautiful flower"));

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

var arr\_num => (arr) {

var sum=0;

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

{

sum=sum+arr[i];

}

return (sum);

}

console.log("Sum of array elements" + arr\_num([2,4,6,8]));

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

var prime\_num => (arr) {

var prime=[];

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

{

var flag = true;

if (arr[i] == 1 || arr[i] == 0 || arr[i]<0) {

flag = false;

}

else {

for(let j = 2; j <= arr[i]/2; j++) {

if (arr[i] % j == 0) {

flag = false;

break;

}

}

}

if(flag==true)

prime.push(arr[i]);

}

return (prime);

}

console.log("Prime elements of array" + prime\_num([2,4,6,8]));

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

var pal\_el => (arr) {

var pal=[];

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

{

var flag = true;

for(let j = 0; j < arr[i].length / 2; j++) {

if(arr[i][j] !== arr[i][arr[i].length - j - 1]){

flag = false;

break;

}

}

if(flag==true)

pal.push(arr[i]);

}

}

return (pal);

}

console.log("Palindrome elements of array" + pal\_el(['abc','AAA','ABBA','lkjd']));