****Problem****:

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

function addfive(num)

{

let sum=5;

return sum+=num;

}

console.log(addfive(5))

console.log(addfive(-5))

console.log(addfive(0))

##### **Output:**

10

0

5

****Problem****:

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

function getOpposite(num){

return(-num)

}

console.log(getOpposite(5))

##### **Output:**

-5

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

****Problem****:

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

var min = 5;

function toSeconds(min){

return min\*60

}

var secs = toSeconds(min)

console.log(secs)

##### **Output:**

300

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

var mystr = "5";

function toInteger(mystr) {

return parseInt(mystr)

}

var myint = toInteger(mystr)

console.log(myint)

##### **Output:**

5

****Problem****

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

var myint = 3;

function nextNumber(myint) {

return ++myint

}

var myNextint = nextNumber(myint)

console.log(myNextint)

##### **Output:**

4

****Problem****

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

var arr = [1, 2, 3];

function getFirstElement(arr) {

return arr[0]

}

var data = getFirstElement(arr)

console.log(data)

##### **Output:**

1

**Convert Hours into Seconds**

**Write a function that converts hours into seconds.**

var hour = 2;

function hourToSeconds(data) {

const sec = 3600;

return data\*sec

}

var data = hourToSeconds(hour)

console.log(data)

##### **Output:**

**7200**

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

function findPerimeter(num1,num2) {

return 2\*(num1+num2)

}

var perimeter = findPerimeter(6,7);

console.log(perimeter)

##### **Output:**

26

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

function lessThan100(num1,num2) {

if(num1+num2>100){

return false

}else return true

}

var res = lessThan100(22,15);

console.log(res)

##### **Output:**

True

**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.**

function remainder(num1,num2) {

if((num1 / num2)===0 ){

return 0;

}else{

return num1;

}

}

var res = remainder(1,3)

console.log(res)

##### **Output:**

1

**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.**

function CountAnimals(tur,horse,pigs) {

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

}

var legs = CountAnimals(2,3,5)

console.log(legs)

##### **Output:**

36

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

function frames(num1,num2) {

return (num1\*60)\*num2;

}

var fps = frames(1,2)

console.log(fps)

##### **Output:**

120

**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.**

function divisibleByFive(num1) {

return num1/5==1;

}

var divisible = divisibleByFive(5)

console.log(divisible)

##### **Output:**

True

**Write a function called “isEven”.  
Given a number, “isEven” returns whether it is even.**

function isEven(num){

return (num%2==0)===num

}

var even = isEven(5)

console.log(even)

##### **Output:**

**False**

**Problem**:  
**Write a function called “areBothOdd”.  
Given 2 numbers, “areBothOdd” returns whether or not both of the given numbers are odd.**

**//solution**

function areBothOdd(num1, num2){

if(num1%2!==0 && num2%2!==0){

return true;

}else{

return false;

}

}

console.log(areBothOdd(1,3));

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

**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.**

//solution

function getFullName(firstName, lastName){

return firstName+" "+lastName;

}

console.log(getFullName("GUVI","GEEK"));

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

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

//solution

function getLengthOfWord(word1){

if(word1==Number){

return -1;

}else{

return word1.length;

}

}

console.log(getLengthOfWord("GUVI"));

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

**Problem**:  
**Write a function called “isSameLength”.  
Given two words, “isSameLength” returns whether the given words have the same length**.

//solution

function isSameLength(word1, word2){

if(word1.length==word2.length){

return true;

}else{

return false;

}

}

console.log(isSameLength("GUVI","GEEK"));

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

**Problem**:

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

**//solution**

function getDistance(x1,y1,x2,y2){

this.x = x1;

this.y = y1;

this.x1 = x2;

this.y1 = y2;

this.distanceTo = function() {

return Math.sqrt((Math.pow(this.x1-this.x,2))+(Math.pow(this.y1-this.y,2)))

};

}

var points = new getDistance (100,100,400,300);

console.log(points.distanceTo());

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

**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’**.

//solution

function getNthElement(array,n){

return array[n];

}

console.log(getNthElement([1,3,5],1));

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

**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’.**

**//solution**

function getLastElement(array){

if(array.length!==0){

return array[array.length-1];

}if(array.length===0){

return -1;

}

}

console.log(getLastElement([1,2,3,4]));

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

**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.**

**//solution**

var obj = {

mykey: "value"

};

function getProperty(obj, key) {

if(key==Object.keys(obj)){

return obj.mykey;

}else{

return "NA";

}

}

console.log(getProperty(obj,"mykey"))

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

**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.**

**//solution**

var obj = {

mykey: "value"

};

function addProperty(obj, key){

return obj[key]=true;

}

console.log(addProperty(obj,"mykey"));

console.log(obj)

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

**Problem**:

**Write a function called “removeProperty”.  
Given an object and a key, “removeProperty” removes the given key from the given object.**

**//solution**

var obj={

name:"nk"

};

function removeProperty(obj, key){

return (delete obj[key]);

}

console.log(removeProperty(obj,"name"))

console.log(obj)

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

**Problem**:

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

**//solution**

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

function countPositivesSumNegatives(arr) {

var pos=0;

var sum=0;

var a=[];

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

if(arr[i]>0){

pos++;

if(arr[i]<0){

sum +=arr[i];

}

}

}

a.push(pos);

a.push(sum);

return a;

}

console.log(countPositivesSumNegatives(arr));

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

**Problem**:

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

**//solution**

function getPositives(ar){

var a=[];

for(var i=0;i<ar.length;i++){

if(ar[i]>0){

a.push(ar[i]);

}

}

return a;

}

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).**

**//solution**

function powersOfTwo(n){

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

var res=Math.pow(2,i);

}

return res;

}

console.log(powersOfTwo(2))

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

**Problem**:

**Find the maximum number in an array of numbers**

**//solution**

function findMax(ar)

{

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**

**//solution**

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)

{

for(let i = 2; i <= n/2; 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”

//solution

printPrimes(10);

// Function prints the first nPrimes numbers

function printPrimes(nPrimes)

{

var n = 0;

var i = 100;

while(n < nPrimes)

{

if (isPrime(i))

{

console.log(n, " → ", i);

n++;

}

i++;

}

}

// Returns true if a number is prime

function isPrime(n)

{

for(let i = 2; i <= n/2; i++){

if(n % i === 0){

return false;

}

}

return true;

}

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

**Problem**:

Reverse a string

//solution

var s = reverseString("JavaScript");

console.log(s);

function reverseString(s)

{

let newString="";

for(let i=s.length -1;i>=0;i--){

newString +=s[i];

}

return newString;

}

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

**Problem**:

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

//solution

var ar1 = [1, 2, 3];

var ar2 = [4, 5, 6];

var ar = mergeArrays(ar1, ar2);

console.log(ar);

function mergeArrays(ar1, ar2)

{

var result = [];

for(var el of ar1)

{

result.push(el);

}

for(var el of ar2){

result.push(el)

}

return result;

}

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

**Problem**:

Calculate the sum of numbers received in a comma delimited string

//solution

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

function sumCSV(s){

var sep=s.split(",");

var sum=0;

for (var i=0;i<sep.length;i++){

sum += parseFloat(sep[i]);

}

return sum;

}

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