Today's mini-project is all about building a library of functions that might be useful in the future. Your task is to create a single file called **functions-array-practice.js** that contains the functions described below.

You should build the file using the code below, and then give two examples of how the function is used. This is done by calling the function in the **functions-array-practice.js** file.

Function Name: squareArea
Inputs: side (length of the side of a square), unit (the unit used to measure the side length)
Outputs: string containing the area and units
Code:

```
function squareArea(side,unit){

var area = side*side;
var areaUnit = unit + '^2';

return area + ' '+ areaUnit;

}
```

Example function calls:
squareArea(3,'m')--> "9 m^2"
squareArea(100,'in')--> "10000 in^2"

Function Name: fullName
Inputs: firstName (string containing first name), lastName (string containing last name)
Outputs: string containing the first and last name together with a space
Code:

```
function fullName(firstName,lastName){

return firstName + ' ' + lastName;


}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: circleInformation
Inputs: radius (radius of the circle)
Outputs: array containing the area and circumference
Code:

```
function circleInformation(radius){

var circleArea = Math.PI*radius*radius; //formula for circle area is Pi*r*r
var circleCircumference = 2*Math.PI*radius; //formula for circle circumference is 2*Pi*r

return [circleArea,circleCircumference];

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: stringToArray
Inputs: string (This is a string like “jim,millie,wayne,tony,liz”)
Outputs: array of names separated ([“jim”,”millie”,”wayne”,”tony”,”liz”])
Code:

```
function stringToArray(string){

var nameArray = string.split(',');
return nameArray;


}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: removeSpaces
Inputs: string (“this is a sentence with spaces”)
Outputs: string (“thisisasentencewithspaces”)
Code:

```
function removeSpaces(string){


var splitString = string.split(" ");
var newString = '';

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

    newString+=splitString[i];

}

return newString;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: dayOfWeek
Inputs: dayNumber (a number between 0 and 6)
Outputs: string (name of the day with the number dayNumber)
Code:

```
function dayOfWeek(dayNumber){

days = ['Sunday','Monday','Tuesday','Wednesday','Thursday','Friday','Saturday']; //Store the strings for each day in an array

var currentDayName = days[dayNumber];

return currentDayName;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: searchArray
Inputs: array (any array of elements), element (the array element to be found)
Outputs: index (a number indicating which numbered element contains the search element, or -1 if the element is not found.)
Code:

```
function searchArray(array,element){

foundIndex = -1;

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

if(array[i]==element){

foundIndex = i;

}

}

return foundIndex;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: searchArray
Inputs: array (any array of elements), element (the array element to be found)
Outputs: index (a number indicating which numbered element contains the search element, or -1 if the element is not found.)
Code:

```
function searchArray(array,element){

foundIndex = -1;

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

if(array[i]==element){

foundIndex = i;

}

}

return foundIndex;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: maxArray
Inputs: array (any array of elements)
Outputs: max (greatest numerical value contained in the array)
Code:

```
var maxArray = function(array){

var max = array[0]; //Set max to the first element of the array
var arraySize = array.length; //arraySize is the number of elements in the array

for(var i=0;i<array.length;i++){ //compare each element of the array to the max

if(array[i]>max){ //If the current element of the array is greater than max,

    max = array[i];    //Make max equal to the current value of the array

}


}

return max;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: minArray
Inputs: array (any array of elements)
Outputs: min (least numerical value contained in the array)
Code:

```
var minArray = function(array){

var min = array[0]; //Set min to the first element of the array
var arraySize = array.length; //arraySize is the number of elements in the array

for(var i=0;i<array.length;i++){ //compare each element of the array to the minimum value

if(array[i]<min){ //If the current element of the array is less than min,

    min = array[i];    //Make min equal to the current value of the array

}


}

return min;

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Function Name: isPrime
Inputs: number
Outputs: numberIsPrime (true or false, depending on whether or not the number is prime)
Code:

```
function isPrime(number){
numberIsPrime = true;

if(number==1){
return false;

}

else{
for(var i=2;i<=Math.sqrt(number);i++){


if(number%i==0){

numberIsPrime = false;

}
}

return numberIsPrime;

}

}
```

Example function calls: (Include these in your **functions-array-practice.js** file.)

Grading Rubric:

5	All functions correctly built inside **functions-array-practice.js**, with two calls to each function
4	All functions correctly built inside **functions-array-practice.js**, with at most two missing function calls **OR** no more than one function is incorrectly entered
3	More than one function is missing or incorrect, or more than two function calls are missing