Assessment # 3

Control Structures, Arrays and Functions

Due Date 1 Day before the Lab at 11:59 PM

1. Using a ***While*** loop and arrays, write a JavaScript program that saves the first 100 prime numbers into an Array called **primeNumbers**.

2. Display all the elements inside the **primeNumbers** array using one for the array built-in functions.

```
  2   3   5   7  11  13  17  19  23  29
 31  37  41  43  47  53  59  61  67  71
 73  79  83  89  97 101 103 107 109 113
127 131 137 139 149 151 157 163 167 173
179 181 191 193 197 199 211 223 227 229
233 239 241 251 257 263 269 271 277 281
283 293 307 311 313 317 331 337 347 349
353 359 367 373 379 383 389 397 401 409
419 421 431 433 439 443 449 457 461 463
467 479 487 491 499 503 509 521 523 541
```

3. Consider the following array declaration: *let countries = ["Qatar", "Germany", "China"];*
    - Add "**Kuwait**" to the end of the array
    - Add "**Japan**" to the beginning of the array.
    - Create a ***displayCountries*** function that takes an array as an argument and prints the array elements individually using a ***For-of*** loop. Call the function to display the cars array.
    - Sort the array alphabetically in reverse order and print it again.
4. Write a javaScript code that find the average of the following array. Also, it should show the total number of elements inside [It should be 12] . Hint : First, flatten the array before calculating the average. Also, you should use the built-in Array functions.

```
const nums = [[2,3] , [34,89, [3,4,5, [3,4,5]]] , [55,10]];
```

5. Write and test a function named **dayOfWeek**. It takes a day number and return the day name (e.g., *dayOfWeek(1)* return Monday). Tip: use a switch to implement this function.

6. Write and test function named **drawTriangle** to draw a Triangle. For example, if you call *drawTriangle(5)* you get the following output:

```
5 5 5 5 5
4 4 4 4
3 3 3
2 2
1
```

Tip: use a nested loop and array **push** and **join** functions.

7. Consider the following variables:

```
let string = 'Hello', int = 254;
let float = 25.4;
let arr = [1, 2, 3];
let object = {course: 'JS', part: 1};
let func = function(){return;};
let nullValue = null;
let undefinedValue;
let boolean = true;
```

Add these variables to an array named ***variables*** then loop through the array using for-of loop to display the value of each variable and its data type. The output should be as follows:

```
Hello is string
254 is number
25.4 is number
1,2,3 is object
[object Object] is object
function (){return;} is function
null is object
undefined is undefined
true is boolean
```

8. Write a function named **greet** that takes three parameters **firstName**, **lastName** and **gender** then returns the following greeting: *Hello, my name is …. …. I am a ….* .

The function should replace …. with the function parameters using a **string template literal**.

If you test the function using **greet('Ali', 'Faleh' , 10)**; then you should get back: Hello, my name is Ali Faleh I am 10 years old.

9. Use a **map** to store then display the following data:

Key	Value
1	'Monday'
2	'Tuesday'
3	'Wednesday'
4	'Thursday'
5	'Friday'
6	'Saturday'
7	'Sunday'

10. Consider the following array:

```
let colors = ["white", "blue", "yellow", "black", "red", "green"]
```

Using array destructuring, assign the first two elements to firstColor and secondColor variables and assign the remaining elements to otherColors variable. Display the values of these 3 variables.

11. Use the **spread** operator to produce an array named **seasons** by concatenating the following 2 arrays and adding extra elements "Cool!" and "Super Hot Summer!" as seasons.

Input arrays:

```
let cold = ['autumn', 'winter'];
let warm = ['spring', 'summer'];
```

Output seasons array:

```
["Cool!", 'autumn', 'winter', 'spring', 'summer', "Super Hot Summer!"]
```

12. Call the ***Math***.max() function and pass the **nums** array to it. Do not forget to use **the spread** operator to convert the array to multiple arguments expected by the max function.

```
let nums = [1, 2, 3, 4, 45, 5, 6]
```

13. Consider the following the ***powerOf*** function and **nums** array.

```
let numbers = [6, 2, 3, 3, 8, 9, 12, 3, 8, 9, 5];

function powerOf(x, p) {
  return x ** p;  //returns X to the power of P
}
```

a. Use the ***powerOf*** function along with the array **map** function to power the elements of ***numbers*** array and assign the results to **poweredNums**. Then Display **poweredNums**.

b. Enhance the implementation done in the previous question by removing the ***powerOf*** function and using an anonymous **arrow** function.

```
let numbers = [1, 2, 3, 4, 4, 5, 6, 7, 8, 9, 10];
```

14. Consider the following array:

```
let nums = [1, 2, 3, 4, 5, 6, 7, 8];
```

- Use **filter** then **map** array functions to filter even numbers then square them. Assign the results to a variable named **squaredEvenNumbers** then display it.

  The output should be: **squaredEvenNumbers**: [4, 16, 36, 64]

- Use the **reduce** array function to compute the of **sums** array. The output should be: Sum of array elements: 36

You must use arrow functions in this exercise.

15. Square and sum the elements of this array using arrow functions and in 1 line of code. Then find the average of the array.

```
let nums = [2, 4, 5];
```

16. Sort the in ascending, descending order [18,57,86,65,14,13,21,91]

Note that you must use **JavaScript features and capabilities** such as **arrow functions**, array functions **(.map, .reduce, .filter, .splice, .sort…) and spread operator**.

After you complete the lab, fill in the ***Lab3-TestingDoc-Grading-Sheet.docx*** and save it inside ***Lab3-JavScript*** folder under the assessment. Sync your repository to push your work to Github.