**Overview**

The objective of this lab is to give you practice using:

* Creating an array
* Adding values to an array
* Getting values from an array
* Calling array methods to do special operations on the array

**Part 1: Array Exercises**

A web page, *ArrayExercises.html*, has been written for you that contains code to call functions that you will write in a file named *ArrayExercises.js*. The instructions for writing your functions and the code to test your functions are *ArrayExercises.html*, but all the code you write will go in *ArrayExercises.js.*

Here is a screenshot of the finished ArrayExercises:

A screenshot of a social media post

Description automatically generated

**Part 2: Web Apps**

You will create two web apps. The HTML page for each of these has already been written for you. You will just write the JavaScript file.

Web App I for Group A – Average Score Calculator

You have been provided with a web page for entering scores, calculating an average, and finding the highest score. You will need to create a file named *AverageOfScores.js* in which you will:

1. Declare an empty array named *scores*.
2. Define a function named *calcAverage* that has no parameters and that returns the average of the values in the scores array.
3. Define a function named *findHighest* that has no parameters and returns the highest value in the scores array.

Use loops to calculate the average and to find the highest score.

Here is a screenshot of the finished web app:

A screenshot of a cell phone

Description automatically generated

Web App II for Group A – Price List

This web app displays a list of products and prices. A user can add items to the list and enter prices.

Implementation:

1. Declare two one-dimensional arrays:
   1. *products*
   2. *prices*
2. Write three functions:
   1. *addProduct*
      * Has one parameter: a product name.
      * Returns true if the product was found.
      * Adds a product to the *products* array and a price to the *prices* array
   2. *removeProduct*
      * Has one parameter: a product name.
      * Returns true if the product name was found.
      * Use the *indexOf* method to get the array index, then use the *splice* method to remove the correct elements from the two arrays.
   3. *changePrice*
      * Has two parameters: the array index for the product, and the price.
      * Returns true if the index is valid.
      * Change the price in the *prices* array.

This is a screenshot of a completed web app:

A screenshot of a cell phone

Description automatically generated

**Submitting your lab work on Moodle**

Beta Version

Post the following in the *Lab Beta forum*:

1. The web pages you created for part 2.  
   (Zip the files for you web pages and attach them to the post.)
2. A code review of your lab partner’s web page for part 2.   
   (Review the part 2 web apps for one of your lab partners using the Code Review Form provided.)

Code Review

1. Submit a copy of the code review above to the *Lab Code Review assignment*.

Production Version  
You may revise your beta version before submitting the production version. On the code review form you received from your lab partner, complete the “Production” column to show what you did or did not revise.

Upload the following to the *Lab Production Version* assignment:

1. A zip file containing the two files (.html and .js) for part 1.
2. A zip file containing the four files for part 2.
3. The code review from your lab partner with the “Prod” column filled in by you.