Week 4: Coding Assignment

URL to GitHub Repository: https://github.com/MCSquaredTech

URL to Your Coding Assignment Video: https://youtu.be/SLa4eTYzZ4U

Instructions (these are **required):**

- In Visual Studio Code, **write the code** that accomplishes the objectives listed below and ensure that the code you have written compiles and runs as directed.
- **Comment your code**, to prove that you have understand of what you have written.
- **Create a new repository on GitHub** for this week's assignments and push this document, with your project code, to the repository.
- **Include the URLs** for this week's repository and video where instructed.
- **Submit a document** containing the two URL links as a .PDF file **in the LMS**.

Coding Steps:

All Code: https://github.com/MCSquaredTech/week04/blob/main/week-04-coding-assignment/src/code-assignment.js

The answer to all questions must be printed to your Browser's console -- using console.log():

1. Create an array called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.

   The code starts at line 4.

   1a. Programmatically subtract the value of the first element in the array from the value in the last element of the array
   - Do not use numbers to reference the last element, find it programmatically,
   - **ages[7] – ages[0]** is not allowed!

   Create two function calls that copy the current function and pop's the appropriate value from the end of the array. The code starts at line 16.

Week 4: Coding Assignment

1b. Add a new age to your array and repeat the step above to ensure it is dynamic. (works for arrays of different lengths).

The code starts at line 74. Simple push to the array stack and repeat function call.

1c. Use a loop to iterate through the array and calculate the average age.

The code starts at line 93. The Plan: Create two variables, length, and sum. Length holds the array count and sum product of all the ages. A for loop will calculate the sum of all the ages. Returning the average by dividing the length by the sum.

2. Create an array called names that contains the following values: 'Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'.
   The code starts at line 116. The Plan - Create a function to calculate the average number of letters per name: Create two variables to calculate the array length and one to accumulate the sum of name length. Create a for loop to sum and add up the length of each name. Returning the average name length (sum/length)
   2a. Use a loop to iterate through the array and calculate the average number of letters per name.
   The code starts at line 124. The Plan - Create a function to calculate the average number of letters per name: Create two variables to calculate the array length and one to accumulate the sum of name length. Create a for loop to sum and add up the length of each name. Returning the average name length (sum/length)
   2b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces.

   The code starts at line 152. The Plan: is to create a string variable to hold the concatenation of all the elements in the array and return them to the calling function.

3. How do you access the last element of any array?
   The explanation starts at line 176. The last element can be accessed in many way here are a few.
   - array[array.length - 1]; // primitive

Week 4: Coding Assignment

- array.pop(); // function call from the array. Note: removes the element from the array
- array.slice(-1); // This function can remove an element from the array if you add a second parameter
- array.at(-1); // returns the last element.

4. How do you access the first element of any array?
   The explanation starts at line 185. Similarly, you can return the first element in the same manner as the explanation above. The first element can be accessed in many way here are a few:
   - array[0]; // primitive
   - Make a copy of the array and reverse the array and pop the first element.
   - array.slice(0); // This function can remove an element from the array if you add a second parameter
   - array.at(0); // returns the last element.

5. Create a new array called **nameLengths**. Write a loop to iterate over the previously created names array and add the length of each name to the **nameLengths** array.

   For example:

   ```
   let names = ["Kelly", "Sam", "Kate"];        //starting with this array
   let nameLengths = [5, 3, 4];                 //create this new array
   ```

   The code starts at 199. Created an empty array.

6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array. The first element can be accessed in many way here are a few:

   The code starts at line 229. Create a Do While loop to iterate over the array and calculate the sum.

PROMINEO TECH

Week 4: Coding Assignment

7. Write a function that takes two parameters, **word** and **n**, as arguments and returns the word concatenated to itself n number of times. (i.e. if I pass in 'Hello' and 3, I would expect the function to return 'HelloHelloHello').
   The code starts at line 253. Create a function called repeatToN(word, n) that takes two parameters. create a loop that will cycle n types and concatenates the parameter word together. Finally, return the concatenated word back to the calling code.

8. Write a function that takes two parameters, **firstName** and **lastName**, and returns a full name. *The full name should be the first and the last name separated by a space*.
   The code starts at line 277. Create a function that takes two parameters first and last and uses a template literal to return the full name

9. Write a function that takes an array of numbers and returns true if the sum of all the numbers in the array is greater than 100.
   The code starts at line 308. *Refactored code - Create a function that initiates a loop to calculate the sum total of all the numbers in an array. If the array is greater than 100, return true else, false. The variable sum will hold the accumulative total of the array. variable flag isGreater will be initialized to false and returned at the end of the function

10. Write a function that takes an array of numbers and returns the average of all the elements in the array.
    The code starts at line 338. *Refactored code - Create a function that returns the average of a number array. Utilize the loop from the code. Assignment 9. Return the sum divided by the array length.

11. Write a function that takes two arrays of numbers and returns true if the average of the elements in the first array is greater than the average of the elements in the second array.
    The code starts at line 371. Create a new function called isArrayAGreater - I will call pass two arrays into the function and return if Array 1 is greater than Array 2 - I am creating a helping function sumArray pass in the array and return the sum value. Return a true if the first array isGreaterThen or false if isLessThan - I will refactor my code from the previous two functions.

12. Write a function called **willBuyDrink** that takes a boolean **isHotOutside**, and a number **moneyInPocket**, and returns true if it is hot outside and if **moneyInPocket** is greater than 10.50.

PROMINEO TECH

Week 4: Coding Assignment

The code starts at line 392. Using a Ternary expression, return true or false based on the data provided in the function call. If both conditions are true, then it returns true else, false.

13. Create a function of your own that solves a problem. *In **comments**, write what the function does and why you created it.*
    The code starts at line 422. Sort alphanumerical license numbers using the array.sort() method.

Video Steps:

- Create a video, up to five minutes max, showing and explaining how your project works with an emphasis on the portions you contributed.
- This video should be done using screen share and voice over.
- This can easily be done using Zoom, although you don't have to use Zoom, it's just what we recommend.
    - You can create a new meeting, start screen sharing, and start recording.
    - This will create a video recording on your computer.
- This should then be uploaded to a publicly accessible site, such as YouTube.
    - Ensure the link you share is **PUBLIC** or **UNLISTED**!
    - If it is not accessible by your grader, your project will be graded based on what they can access.