Intro to JavaScript Week 3 Coding Assignment

Points possible: 75

URL to Your GitHub Repository: https://github.com/harberts01/week_3.git

Instructions: In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

1. Create an array called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
    a. 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). Print the result to the console.
    b. Add a new age to your array and repeat the step above to ensure it is dynamic (works for arrays of different lengths).
    c. Use a loop to iterate through the array and calculate the average age. Print the result to the console.
2. Create an array called names that contains the following values: 'Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'.
    a. Use a loop to iterate through the array and calculate the average number of letters per name. Print the result to the console.
    b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces, and print the result to the console.
3. How do you access the last element of any array?
   You can access the last element of an array by using array.length-1. The end of an array will always be one less than total elements in the array since an array is 0 indexing.

4. How do you access the first element of any array?
   You can access the first element of an array by using array[0]. The first element of the array will always be at index 0. So you can call the element using the specific index.

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:

   namesArray = ["Kelly", "Sam", "Kate"] //given this array
   nameLengths = [5, 3, 4] //create this new array

6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array. Print the result to the console.
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’).
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).
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.
10. Write a function that takes an array of numbers and returns the average of all the elements in the array.
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.
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.
13. Create a function of your own that solves a problem. In comments, write what the function does and why you created it.

Screenshots of Code:

PROMINEO TECH

JS week3.js > ...

```
// Week 3 Coding Assignment

// Step 1<---Programmatically subtract the 1st element
//from the last element in the "ages" array.--->

let ages = [3, 9, 23, 64, 2, 8, 28, 93];
//Step 1a.
function subtractFirstLast(a) {
    let x = a[a.length - 1]
    let y = a[0];
    console.log(x - y)
}
subtractFirstLast(ages);

//Step 1b.
ages.push(16);
subtractFirstLast(ages);

//Step 1c.

let sum = 0

for (let number of ages) {
    sum += number;
}
let average = sum / ages.length
console.log(average);


//Step 2 <---Create an array called names that contains the following values:
// 'Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'.--->

let names = ['Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'];
//Step 2a <---Use a loop to iterate through the array and calculate the
//average number of letters per name. Print the result to the console.--->
let total = 0

for (let i = 0; i < names.length; i++) {
    total += names[i].length
}

let avrg = total / names.length
console.log(avrg)

//Step 2b <--- Use a loop to iterate through the array again and concatenate all
//the names together, separated by spaces, and print the result to the console --->
let newNames = '';

for (let i = 0; i < names.length; i++) {
    newNames = newNames.concat(names[i]);
    if (i < names.length-1){
        newNames = newNames.concat(' ')
    }
}

console.log(newNames);


//Step 3 <--- How do you access the last element of any array? --->

// You access the last element of an array by using .length -1. The end of the array
// will always be one less than the total of elements since Arrays are zero indexing.

//Step 4 <--- How do you access the first element of an array? --->

// You access the first element of an array by using array[0]. The first element will always be
// at index 0. So you can call the element by using the specific position.

//Step 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.--->

let nameLength = []

for (let i = 0; i < names.length; i++) {
    nameLength.push(names[i].length);

}
console.log(nameLength)

```

PROMINEO TECH

JS week3.js > ...

```
72
73  //Step 6<---Write a loop to iterate over the nameLengths array and calculate the sum of all the elements
74  // in the array. Print the result to the console.--->
75
76  let sumOne = 0
77  for (let i = 0; i < nameLength.length; i++) {
78      sumOne += nameLength[i]
79  } console.log(sumOne);
80
81  //Step 7 <---Write a function that takes two parameters, word and n, as arguments and returns
82  //the word concatenated to itself n number of times.--->
83
84
85  function concatSelf(word, n) {
86      let a = word
87      let b = n
88      return a.repeat(b)
89  }
90  console.log(concatSelf('Hello', 4));
91
92  //Step 8 <---Write a function that takes two parameters,
93  //firstName and lastName, and returns a full name--->
94
95  function fullName(firstName, lastName) {
96      return firstName + ' ' + lastName;
97  }
98
99  console.log(fullName('Linda', 'Smith'));
100
101 //Step 9 <---Write a function that takes an array of numbers and
102 // returns true if the sum of all the numbers in the array is greater than 100
103
104 let plusHundred = [10, 20, 30, 40]
105 let anyNum = [460, 20, 30]
106
107 function hundredPlus(x) {
108     let a = 0
109     for (i = 0; i < x.length; i++) {
110         a += x[i]
111         }
112 if(a > 100){
113     return true
114 }else{
115     return false
116 }
117
118 }
119 console.log(hundredPlus(plusHundred));
120 console.log(hundredPlus(anyNum));
121 // I tested my function 2x. to prove both results. Also had to write the
122 // if statement outside the for loop in order for it to print to the console correctly
123
124 //Step 10 Write a function that takes an array of numbers and returns the average of all
125 //the elements in the array.
126 let array1 = [10, 20, 20, 30, 40]
127
128
129 function avgArray(arry){
130     let a = 0
131         for(let i = 0; i < arry.length; i++) {
132             a += arry[i]
133     } return a / arry.length
134 }
135 console.log(avgArray(array1));
136
```

PROMINEO TECH

JS week3.js > ...

```
137  //Step 11 Write a function that takes two arrays of numbers and returns true if the average of the elements
138  // in the first array is greater than the average of the elements in the second array
139
140  function doubleArry (arry1, arry2){
141      let a = 0
142      let b = 0
143      let aa = 0
144      let bb = 0
145          for(let i =0; i < arry1.length; i++){
146              a += arry1[i]
147              aa = a/arry1.length
148          }
149          for(let i = 0; i < arry2.length; i++){
150              b += arry2[i]
151              bb = b/arry2.length
152          }
153  if(aa > bb){
154      return true
155  }else{
156      return false
157  }
158  }
159  console.log(doubleArry(plusHundred, array1));
160
161  //Step 12 Write a function called willBuyDrink that takes a boolean isHotOutside,
162  // and a number moneyInPocket, and returns true if it is hot outside and if moneyInPocket
163  // is greater than 10.50.
164
165  let isHotOutside = true;
166  let moneyInPocket = 5.00;
167
168  function willBuyDrink(a, b){
169          if(a === true && b > 10.50){
170              return true
171          }else{
172              return false
173          }
174  }
175
176  console.log(willBuyDrink(isHotOutside, moneyInPocket));
177
178  //Step 13 Create a function of your own that solves a problem.
179  // In comments, write what the function does and why you created it
180
181
182  let stocks = ['GME', 30, 'TSLA', 35, 'AMD', 45, 'AMZN', 47, 'GOOGL', 50];
183
184  function buyStock(array){
185
186      for (let i = 1; i < array.length; i += 2){
187          if(array[i] > 40){
188              console.log(array[i-1]);
189          }
190      }
191  }
192  buyStock(stocks);
193
194  // The stocks array listed above represents a stock and the number after the stock ticker (string)
195  // represents the percent that the price of the stock is off of its
196  // high(which is where investors look for "good buys"). This function
197  // will take the array and will print the stocks that are greater than 40% off their highs
198  // by looping every other element from position 1 and then if that number is greater than 40
199  // it will print the previous element which represents the ticker that should be bought.
200
```

Screenshots of Running Application: