Intro to JavaScript Week 3 Coding Assignment

Points possible: 75

URL to Your GitHub Repository:

https://github.com/Kai-By-Design/Week3.git

URL to Your Coding Assignment Video:

https://drive.google.com/file/d/1CnAuhsGYGjHOVhFIU69eP6gyNLdAbK1u/view?usp=sharing

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.

PROMINEO TECH

JS PT-W3-CodingAssignment.js M ● <> PT-W3-CodingAssignment.html

JS PT-W3-CodingAssignment.js > probOne

```
/*
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.
*/

let ages = [3, 9, 23, 64, 2, 8, 28, 93];
let ages2 = [1, 3, 2, 4];

function probOne(array) {
  //access the first index of the ages array & assign to variable
  let first = array.shift();
  //access the last index of the ages array & assign to variable
  let last = array.pop();
  //
  let diff = last - first;
  console.log('difference: ' + diff);
  array.push(last);
  array.unshift(first);
  console.log('arraytest: ' + array);

  //c.
  let sum = 0;
  for (let i = 0; i < array.length; i++) {
    // loops through and adds the index to the 'sum' variable value
    sum += array[i];
  }
  //calculate the average number of an array by dividing the array value's sum by the length of the array
  let averageNum = sum / array.length;
  console.log('Average Age: ' + averageNum);
}

//a. Expected Output: 90
//c. Expected Average Age: 28.75
probOne(ages);

//b. Expected Output: 3
//c. Expected Average Age: 2.5
probOne(ages2);
```

```
difference: 90                          PT-W3-CodingAssignment.js:22
arraytest: 3,9,23,64,2,8,28,93          PT-W3-CodingAssignment.js:25
Average Age: 28.75                      PT-W3-CodingAssignment.js:35
difference: 3                           PT-W3-CodingAssignment.js:22
arraytest: 1,3,2,4                      PT-W3-CodingAssignment.js:25
Average Age: 2.5                        PT-W3-CodingAssignment.js:35
```

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.

```
/*
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.
*/

let names = ['Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob']
// Expected 23
let nameLengthTotal = 0;
// Expected 6
let nameCount = 0;
let nameString = '';
let nameLengths = [];
let nameLengthsTotal = 0;
function count(people) {
  for (let i = 0; i < people.length; i++) {
    nameLengthTotal += people[i].length;
    nameLengths.push(people[i].length);
    nameCount += 1;
  }
  // Expected 3.8333..
  console.log(nameLengthTotal / nameCount);
  // console.log(nameCount);

  // 5. Expected [3, 5, 3, 5, 4, 3]
  console.log(nameLengths);
  for (let i = 0; i < nameLengths.length; i++){
    nameLengthsTotal += nameLengths[i];
  }
  // 6. Expected Outcome: 23
  console.log(nameLengthsTotal);


  for (let i = 0; i < people.length; i++) {
    nameString += people[i];
  }
  console.log(nameString);
}
//a. Expected Output: 23
//b. Expected Output: Sam Tommy Tim Sally Buck Bob
count(names);
```

```
3.8333333333333335          PT-W3-CodingAssignment.js:70
▶ (6) [3, 5, 3, 5, 4, 3]    PT-W3-CodingAssignment.js:74
23                          PT-W3-CodingAssignment.js:79
SamTommyTimSallyBuckBob     PT-W3-CodingAssignment.js:85
```

PROMINEO TECH

3. How do you access the last element of any array?

   array[array.length -1]

   array.pop()

4. How do you access the first element of any array?

   array[0]

   array.shift()

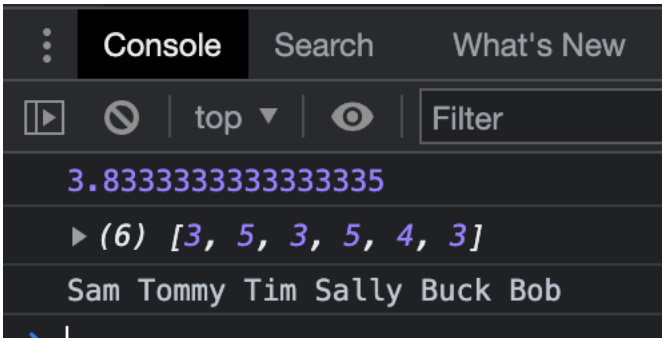
5. Create a new array called nameLengths. Write a loop to iterate over the previously created names array and add the length of each name to the nameLengths array. For example:

   namesArray = ["Kelly", "Sam", "Kate"] //given this array

   nameLengths = [5, 3, 4] //create this new array

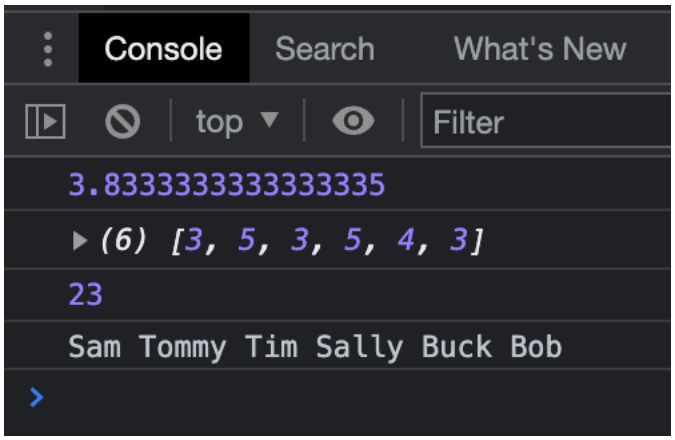
PROMINEO TECH

```
46  let names = ['Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob']
47  // Expected 23
48  let nameLengthTotal = 0;
49  // Expected 6
50  let nameCount = 0;
51  let nameString = '';
52  let nameLengths = [];
53  function count(people) {
54    for (let i = 0; i < people.length; i++) {
55      nameLengthTotal += people[i].length;
56      nameLengths.push(people[i].length);
57      nameCount += 1;
58    }
59    // Expected 3.8333..
60    console.log(nameLengthTotal / nameCount);
61    // console.log(nameCount);
62
63    // 5. Expected [3, 5, 3, 5, 4, 3]
64    console.log(nameLengths);
65
66    for (let i = 0; i < people.length; i++) {
67      nameString += people[i] + ' ';
68    }
69    console.log(nameString);
70  }
71  //a. Expected Output: 23
72  //b. Expected Output: Sam Tommy Tim Sally Buck Bob
73  count(names);
```



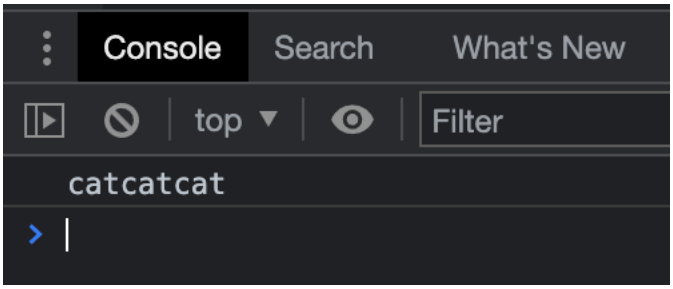
6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array. Print the result to the console.

```
let names = ['Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob']
// Expected 23
let nameLengthTotal = 0;
// Expected 6
let nameCount = 0;
let nameString = '';
let nameLengths = [];
let nameLengthsTotal = 0;
function count(people) {
  for (let i = 0; i < people.length; i++) {
    nameLengthTotal += people[i].length;
    nameLengths.push(people[i].length);
    nameCount += 1;
  }
  // Expected 3.8333..
  console.log(nameLengthTotal / nameCount);
  // console.log(nameCount);

  // 5. Expected [3, 5, 3, 5, 4, 3]
  console.log(nameLengths);
  for (let i = 0; i < nameLengths.length; i++){
    nameLengthsTotal += nameLengths[i];
  }
  // 6. Expected Outcome: 23
  console.log(nameLengthsTotal);


  for (let i = 0; i < people.length; i++) {
    nameString += people[i] + ' ';
  }
  console.log(nameString);
}
//a. Expected Output: 23
//b. Expected Output: Sam Tommy Tim Sally Buck Bob
count(names);
```



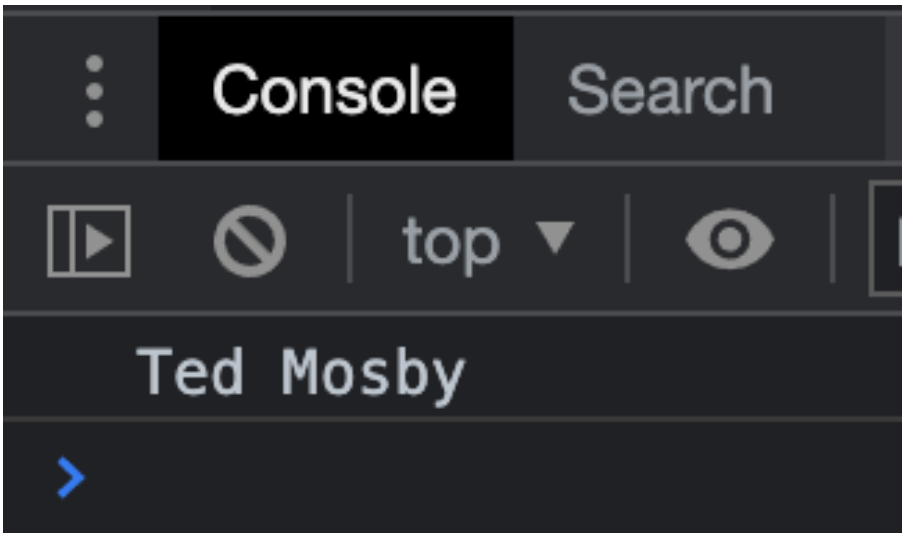
7. Write a function that takes two parameters, word and n, as arguments and returns the word concatenated to itself n number of times. (i.e. if I pass in ‘Hello’ and 3, I would expect the function to return ‘HelloHelloHello’).

```
/*
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').
*/

let output = '';
function repeated (word, n) {
  for (let i = 0; i < n; i++) {
    output += word;
  }
  console.log(output);
}
repeated('cat', 3);
```



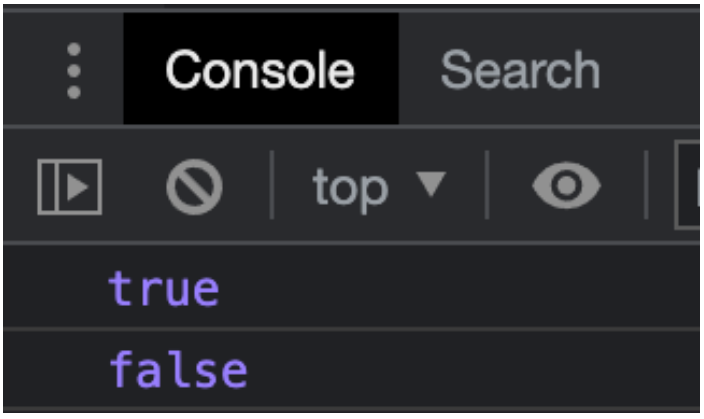
8. Write a function that takes two parameters, firstName and lastName, and returns a full name (the full name should be the first and the last name separated by a space).

```
/*
8. Fullname
*/

function name(first, last){
  console.log(first + ' ' + last);
}
name('Ted', 'Mosby');
```



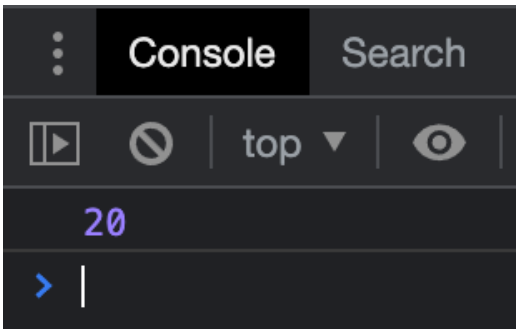
9. Write a function that takes an array of numbers and returns true if the sum of all the numbers in the array is greater than 100.

```
/*
9. Sum over 100 Boolean
   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.
*/

//  Expected Output: True
let numbers = [ 60, 10, 30, 40, 50];
//  Expected Output: False
let numbers2 = [ 10, 10, 10, 20, 40];

function sumTest(array){
  let sum = 0;
  for (let i = 0; i < array.length; i++) {
    sum += array[i];
    if (sum === 100) {
      return true;
    }
  }
  return false;
}
console.log(sumTest(numbers));
console.log(sumTest(numbers2));
```



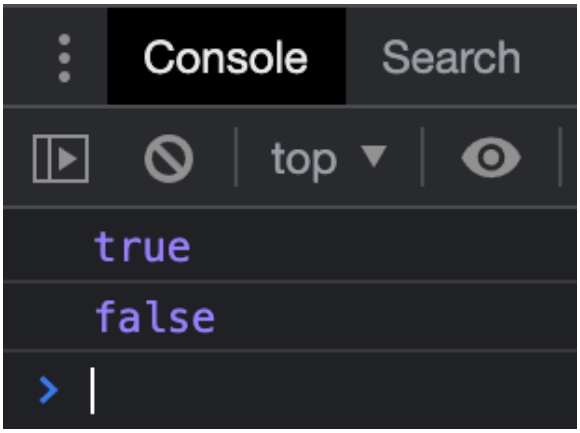
10. Write a function that takes an array of numbers and returns the average of all the elements in the array.

```
/*
10. Write a function that takes an array of numbers and returns the average of all the elements in the array.
*/

let numbers2 = [ 10, 10, 10, 20, 50];
function sumAvg (array) {
  let sum = 0;
  for (let i = 0; i < array.length; i++) {
    sum += array[i];
  }
  return sum / (array.length);
}
// Expected Output: 20
console.log(sumAvg(numbers2));
```



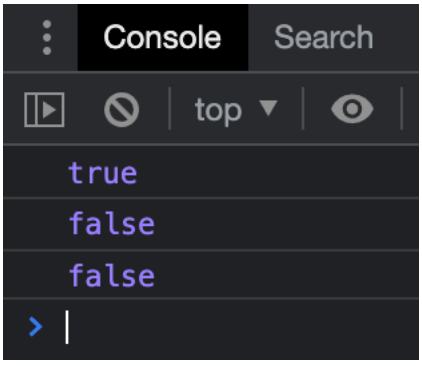
11. Write a function that takes two arrays of numbers and returns true if the average of the elements in the first array is greater than the average of the elements in the second array.

```
/*
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.
*/

let numbers1 = [ 20, 20, 20, 40, 50];
let numbers2 = [ 10, 10, 10, 20, 50];
function sumAvg (array1, array2) {
  let sum1 = 0;
  let sum2 = 0;
  for (let i = 0; i < array1.length; i++) {
    sum1 += array1[i];
  }
  for (let i = 0; i < array2.length; i++) {
    sum2 += array2[i];
  }
  let avg1 = sum1 / (array1.length);
  let avg2 = sum2 / (array2.length);
  if (avg1 > avg2) {
    return true;
  }
  return false;
}
// Expected Output: true
console.log(sumAvg(numbers1, numbers2));

// Expected Output: false
console.log(sumAvg(numbers2, numbers1));

```



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.

```
/*
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.
*/

function willBuyDrink (isHotOutside, moneyInPocket) {
  if (isHotOutside && (moneyInPocket > 10.50)) {
    return true;
  }
  return false;
}
// Expected Output: true
console.log(willBuyDrink(true, 11));
// Expected Output: false
console.log(willBuyDrink(true, 10));
// Expected Output: false
console.log(willBuyDrink(false, 11));
```



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:

```
/*
13. Create a function of your own that solves a problem.
    In comments, write what the function does and why you created it.
*/

/*
I created a function that takes an array of arrays
that contain room names and their width and length dimensions
and outputs a room count and the room names with their corresponding square footage.

This is has value for architects and contractors that have to calculate square footage after taking measurements onsite.
*/

// input - array of arrays
let rooms = [['bedroom_1', 3, 2], ['kitchen', 4,3], ['bath', 4,5], ['bedroom_2', 3, 2], ['living room', 4,3], ['garage', 4,5]];

const floorArrowArray = (buildingInfo) => {
  let totalSqft = 0;
  let output = 'Building Info: ';

  for (i = 0; i < buildingInfo.length; i++){
  let area = (buildingInfo[i][1] * buildingInfo[i][2]);
  let roomName = buildingInfo[i][0];
  let roomCount = i+1;
  output += '\n    ' + 'Room ' + roomCount + ': ' + roomName + '    Area: ' + area + ' SqFt';
  totalSqft += area;
  }
  return output + '\nTotal Building Area: ' + totalSqft + ' Sqft';
}
console.log(floorArrowArray(rooms));
```

Screenshots of Running Application: