



Week 3: Coding Assignment

URL to GitHub Repository:

<https://github.com/soccermoe14/promineo-week3/blob/main/codinglesson.js>

URL to Your Coding Assignment Video:

<https://youtu.be/tBf9ruhb5Us>

Instructions:

- In Visual Studio Code, write the code that accomplishes the objectives listed below and ensures that the code compiles and runs as directed.
- 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 this document as a .PDF file in the LMS.

Coding Steps:

All questions must be printed to your Browser's console:

1. Create an array called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
 - 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!
 - 1b. Add a new age to your array and repeat the step above to ensure it is dynamic. (works for arrays of different lengths).
 - 1c. Use a loop to iterate through the array and calculate the average age.



Week 3: Coding Assignment

2. Create an array called `names` that contains the following values: 'Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'.

- 2a. Use a loop to iterate through the array and calculate the average number of letters per name.

- 2b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces.

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

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

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.

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.



Week 3: Coding Assignment

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.

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.



Week 3: Coding Assignment

```
1 //1. Create an array called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
2
3 ages = [3, 9, 23, 64, 2, 8, 28, 93]; //creates a variable called ages and assigns it an array of numbers
4 console.log(ages); //prints the length of the array (8), and the values within the array
5
6 //1a. Programmatically subtract the value of the first element in the array from the value in the last element of the array
7 // Do not use numbers to reference the last element, find it programmatically,
8 // ages[7] - ages[0] is not allowed!
9
10 let firstElement = ages[0]; //creates a variable called firstElement and assigns it the value of the first element in the array
11 console.log("firstElement: ", firstElement); //prints the value of the firstElement variable to the console
12
13 let lastElement = ages[ages.length - 1]; //creates a variable called lastElement and assigns it the value of the last element in the array
14 console.log("lastElement: ", lastElement); //prints the value of the lastElement variable to the console
15
16 let difference = lastElement - firstElement; //creates a variable called difference and subtracts lastElement minus the firstElement
17 console.log("last element in the ages array:", difference); //printing last element minus the first element
18
19
20 //1b. Add a new age to your array and repeat the step above to ensure it is dynamic. (works for arrays of different lengths).
21 ages.push(44); //adds another element to the ages array
22 console.log(ages); //prints new array
23 console.log(difference); //prints last element minus first element
24 ages.sort(); //sorts the array with the new element
25 console.log(ages); //prints sorted array
26 console.log(difference); //prints last element minus first element **if I'm undersanding, the answer should have been 91 once the array was sorted, but it is still 90**
27
28 //1c. Use a loop to iterate through the array and calculate the average age.
29 let sum = 0;
30 ages.forEach(function(num) {sum += num}); //created a function that loops through numbers in array adding each to the sum
31 average = sum/ages.length; //divided sum of numbers in array by the number of elements in the array
32 console.log(average); //printed the average value of the array ages
33
34 //2. Create an array called names that contains the following values: 'Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob'.
35
36 names = ['Sam', 'Tommy', 'Tim', 'Sally', 'Buck', 'Bob']; //creates a variable called names and assigns it an array of strings
37 console.log(names); //prints all the elements in the array names
38
39 //2a. Use a loop to iterate through the array and calculate the average number of letters per name.
40 let lengthOfNames = 0; //declaring a variable for the length of the names in the array and initializes the sum as 0
41 for (person of names) {
42   | lengthOfNames += person.length
43 } //creates a for loop that loops through the integers (names) in the array counting integers in each name and adding each total to the sum
44 let averageLengthOfNames = lengthOfNames/names.length; //divided the sum of the integers in the array by the number of elements in the array to determine avg
45 console.log(averageLengthOfNames); //prints the average number of letters in the array
```



Week 3: Coding Assignment

```
JS codinglesson.js > ...
46
47
48 //2b. Use a loop to iterate through the array again and concatenate all the names together, separated by spaces.
49 let concatenatedNames = names.reduce(function(accumulator, names){
50 |   return accumulator + " " + names; /**not sure if I did this correctly**; used reduce funtion to remove the commas and then return with the elements concatenated with a space
51 | }
52 console.log(concatenatedNames); //prints all names separated by spaces and no commas
53
54 //3. How do you access the last element of any array?
55 //Use the name of the array and add.length and subtract 1 (ex: names.length -1)
56
57 //4. How do you access the first element of any array?
58 //The first element of an array starts at zero (the index). You can access by naming array and the index (ex: names[0])
59
60 //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.
61 //For example:
62
63 //namesArray = ["Kelly", "Sam", "Kate"] //given this array
64 //nameLengths = [5, 3, 4] //create this new array
65
66 namesArray = ["Kelly", "Sam", "Kate"]; //declares a variable called namesArray and assigns it with strings
67 let nameLengths = namesArray.map(function(element){
68 |   return element.length;
69 | }); /**not sure if I did this correctly** declared a new array using the map function to callback the length of each element and create a new array with those lengths
70 console.log(nameLengths); //prints new array
71
72 //6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array.
73 total = 0; //declares variable total initializes 0
74 nameLengths.forEach(function addNumbers(value){
75 |   total += value;
76 | }); //for each element in the loop adds the value of number of integers to previous total
77 console.log(total); //prints the sum of all the integers in the array
78
79 //7. Write a function that takes two parameters, word and n, as arguments and returns the word concatenated to itself n number of times.
80 // (i.e. if I pass in 'Hello' and 3, I would expect the function to return 'HelloHelloHello').
81 function wordMultiplier(word, multiplier) {
82 |   console.log (word*multiplier);
83 | }
84 wordMultiplier("Hello", 3); //tried to wrriet a function that would return a value of a word repeated n times, but can't get it to work
85
86
87 //8. Write a function that takes two parameters, firstName and lastName, and returns a full name.
88 //The full name should be the first and the last name separated by a space
89 function fullName(firstName, lastName) {
90 |   console.log (firstName + " " + lastName);
91 | }
92 fullName("Marin", "Brouillard");
```



Week 3: Coding Assignment

```
// codingLesson.js > ...
93
94 //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.
95
96 let array2 = [2, 8, 22, 44, 61];
97 function add(accumulator, a) {
98   return accumulator + a;
99   if (sum > 100);
100 }
101 console.log(sum);
102
103 //10. Write a function that takes an array of numbers and returns the average of all the elements in the array.
104 const values = [2, 6, 9, 10, 33, 90]; //declares a variable called values, assigns array of numbers
105 values.forEach(function(num) {sum += num}); //created a function that loops through numbers in array adding each to the sum
106 average = sum / values.length; //divided sum of numbers in array by the number of elements in the array to get the average
107 console.log(average); //printed the average value of the array values
108
109 //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.
110 const arrayA = [30, 22, 71, 100]; //declares a variable, assigns array of numbers
111 const arrayB = [1, 2, 3, 4, 5]; //declares a variable, assigns array of numbers
112 arrayA.forEach(function(num) {sum += num}); //created a function that loops through numbers in array adding each to the sum
113 average1 = sum / arrayA.length; //divided sum of numbers in array by the number of elements in the array to get the average
114 arrayB.forEach(function(num) {sum += num}); //created a function that loops through numbers in array adding each to the sum
115 average2 = sum / arrayB.length; //divided sum of numbers in array by the number of elements in the array to get the average
116 console.log((average1) > (average2)); //prints boolean value
117
118 //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.
119
120 function willBuyDrink(isHotOutside, moneyInPocket) {
121   console.log(isHotOutside && moneyInPocket > 10.5);
122 }
123 willBuyDrink(true, 11); // I'm not sure if I'm using return and console.log correctly. I feel like I'm mixing something up.
124 //created function that returns true if it is hot outside and we have more than 10.50 in our pocket
125
126 //13. Create a function of your own that solves a problem.
127 //In comments, write what the function does and why you created it.
128 function giveCatATreat(hasEaten, goodBehavior) {
129   if (hasEaten == true && goodBehavior == true) {
130     console.log("Good kitty! You deserve a tasty treat!");
131   } else {
132     console.log("Sorry turd muffin! No treats for you.");
133   }
134 }
135 giveCatATreat(true, true);
136 //this is a function I created for to determine whether or not my cat deserves a treat.
137 //created a function with 2 parameters that results in different responses
138 //depending on the boolean value of the arguments passed in
139
140
```

