JAVASCRIPT LAB 3 - STUDENT SUBMISSIONS

Task: Create an array of objects representing student submissions. Define a variety of functions for working with such an array. Also call each of the functions at least once to test it.

Build Specifications:

1. Declare a variable named `submissions` that is initialized to an array with the following objects:

name	score	date	passed
Jane	95	2020-01-24	true
Joe	77	2018-05-14	true
Jack	59	2019-07-05	false
Jill	88	2020-04-22	true

2. Declare a function named `addSubmission`
    - Parameter(s): `array`, `newName`, `newScore`, `newDate`
    - Functionality: construct an object and push it into the `array`. The object must have the same properties as the objects already in the array. Use conditional statements to set the value for the `passed` property to `true` if the score is greater than or equal to 60 and `false` otherwise.

3. Declare a function named `deleteSubmissionByIndex`
    - Parameter(s): `array`, `index`
    - Functionality: remove the object from the `array` at the specified `index` using the splice method.

4. Declare a function named `deleteSubmissionByName`
    - Parameter(s): `array`, `name`
    - Functionality: remove the object from the array that has the provided `name`. Incorporate the findIndex method and the splice method.

5. Declare a function named `editSubmission`
    - Parameter(s): `array`, `index`, `score`
    - Functionality: update an object's `score` in the `array` at the specified `index`. Use conditional statements to set the value for the `passed` property to `true` if the score is greater than or equal to 60 and `false` otherwise.

continued on the next page...

6. Declare a function named `findSubmissionByName`
    - Parameter(s): `array`, `name`
    - Functionality: return the object in the `array` that has the provided `name`. Use the find method.

7. Declare a function named `findLowestScore`
    - Parameter(s): `array`
    - Functionality: return the object in the `array` that has the lowest score. Use the forEach method to loop through the whole array.

8. Declare a function named `findAverageScore`
    - Parameter(s): `array`
    - Functionality: return the average quiz score. Use a for...of loop.

9. Declare a function named `filterPassing`
    - Parameter(s): `array`
    - Functionality: return a new array using the filter method. The filter method should find objects in the `array` that have passing scores.

10. Declare a function named `filter90AndAbove`
    - Parameter(s): `array`
    - Functionality: return a new array using the filter method. The filter method should find objects in the `array` that have scores greater than or equal to 90.

Tests: Same as build specifications.

Extended Challenges:

1. Create a function named `createRange`
    - Parameter(s): `start`, `end`
    - Functionality: construct and return an array of integers starting with the start parameter and ending at the end parameter. e.g `createRange(2, 5)` returns `[2, 3, 4, 5]`.
2. Create a function named `countElements`
    - Parameter(s): `array` (an array of strings)
    - Functionality: construct and return an object with the array values as keys and the number of times that key appears in the array as values. e.g. `countElements(['a', 'b', 'a', 'c', 'a', 'b'])` returns `{ a: 3, b: 2, c: 1 }`.