Intro to Java Week 3 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: In Eclipse, 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 Java 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 of int 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 (i.e. do not use ages[7] in your code). 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 of String 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?

    The last element can be accessed by typing the array name followed by the length of the array minus 1 in brackets. For example: arr[arr.length - 1];
4. How do you access the first element of any array?

    The first element can be accessed by typing the array name followed by the number 0 in brackets. For example: arr[0];
5. Create a new array of int called nameLengths. Write a loop to iterate over the previously created names array and add the length of each name to the nameLengths 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 method that takes a String, word, and an int, 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 method to return “HelloHelloHello”).
8. Write a method that takes two Strings, firstName and lastName, and returns a full name (the full name should be the first and the last name as a String separated by a space).
9. Write a method that takes an array of int and returns true if the sum of all the ints in the array is greater than 100.
10. Write a method that takes an array of double and returns the average of all the elements in the array.
11. Write a method that takes two arrays of double 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 method called willBuyDrink that takes a boolean isHotOutside, and a double moneyInPocket, and returns true if it is hot outside and if moneyInPocket is greater than 10.50.
13. Create a method of your own that solves a problem. In comments, write what the method does and why you created it.

Screenshots of Code:

```java

public class Week3CodingAssignment {

    public static void main(String[] args) {

//      Question 1
        System.out.println("** Question 1 **\n");
        int[] ages = {3,9,23,64,2,8,28,93,72};

        System.out.println("Last Age - First Age: " + (ages[ages.length - 1] - ages[0]));

        double total = 0;

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

        double average = total/ages.length;

        System.out.printf("Average Age: %.2f%n", average);

//      Question 2
        System.out.println("\n** Question 2 **\n");

        String[] names = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};

        double totalNameLength = 0;

        for(String name: names) {
            totalNameLength += name.length();
        }

        double averageNameLength = totalNameLength / names.length;

        System.out.printf("Average Name Length: %.2f%n", averageNameLength);

        String allNames = "";

        for (String name: names) {
            allNames += name + " ";
        }

        System.out.println(allNames);

//      Question 5
        System.out.println("\n** Question 5 & 6 **\n");
        int[] nameLengths = new int[names.length];

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

//      Question 6
        int totalNameLengths = 0;

        for(int num: nameLengths) {
            totalNameLengths += num;
        }
```

```
59
60          System.out.println("Total Names Length: " + totalNameLengths);
61
62  //      Question 7
63          System.out.println("\n** Question 7 **\n");
64          String wordMultiplied = wordMultiplier("Hello", 3);
65          System.out.println(wordMultiplied);
66
67  //      Question 8
68          System.out.println("\n** Question 8 **\n");
69          String fullName = fullName("John", "Doe");
70          System.out.println(fullName);
71
72  //      Question 9
73          System.out.println("\n** Question 9 **\n");
74          int[] intArr1 = {10, 20, 30, 41};
75          int[] intArr2 = {10, 20, 30, 40};
76          System.out.println(greaterThan100(intArr1));
77          System.out.println(greaterThan100(intArr2));
78
79  //      Question 10
80          System.out.println("\n** Question 10 **\n");
81          double[] nums = {1.2, 5.1, 6.7, 8.324, 4.996};
82          double averageOfDoubles = averageOfDoubleArray(nums);
83          System.out.println(averageOfDoubles);
84
85  //      Question 11 (I used nums array above for example)
86          System.out.println("\n** Question 11 **\n");
87          double[] nums2 = {1.1, 4.9, 6.6, 8.234, 4.874};
88          System.out.println(firstArrayAverageGreaterThanSecond(nums, nums2));
89          System.out.println(firstArrayAverageGreaterThanSecond(nums2, nums));
90
91  //      Question 12
92          System.out.println("\n** Question 12 **\n");
93          boolean hot = true;
94          double money = 13.45;
95          boolean drinkBought = willBuyDrink(hot, money);
96          System.out.println(drinkBought);
97
98  //      Question 13
99          System.out.println("\n** Question 13 **\n");
100         double kilometers = 144;
101         double miles = kilometersToMiles(kilometers);
102         System.out.println(kilometers + " km = " + miles + " mi");
103
104     }
105
106 //  *** METHODS ****
107
108 //  For Q7
109     public static String wordMultiplier(String word, int n) {
110         String result = "";
111         for(int i = 0; i < n; i++) {
112             result += word;
113         }
114         return result;
115     }
```

```
116 //  For Q8
117     public static String fullName(String fname, String lname) {
118         return fname + " " + lname;
119     }
120 //  For Q9
121     public static boolean greaterThan100(int[] nums) {
122         int sum = 0;
123         for (int num: nums) {
124             sum += num;
125         }
126         return sum > 100;
127     }
128 //  For Q10
129     public static double averageOfDoubleArray(double[] arr) {
130         double total = 0;
131         for(double n: arr) {
132             total += n;
133         }
134         return total / arr.length;
135     }
136 //  For Q11
137     public static boolean firstArrayAverageGreaterThanSecond(double[] arr1, double[] arr2) {
138         double average1 = averageOfDoubleArray(arr1);
139         double average2 = averageOfDoubleArray(arr2);
140         return average1 > average2;
141     }
142 //  For Q12
143     public static boolean willBuyDrink(boolean isHotOutside, double moneyInPocket) {
144         if(isHotOutside && moneyInPocket > 10.50) {
145             return true;
146         } else {
147             return false;
148         }
149     }
150 /// For Q13. This method takes a double, kilometers, and returns a double that is the kilometers value converted to miles.  I created this method because I like to travel but most count
151 //  Please note that my method is an estimate conversion.
152     public static double kilometersToMiles(double kilometers) {
153         double halfKm = kilometers / 2;
154         return halfKm + (halfKm/4);
155     }
156
157 }
158
```

Screenshots of Running Application:

```
** Question 1 **

Last Age - First Age: 69
Average Age: 33.56

** Question 2 **

Average Name Length: 3.83
Sam Tommy Tim Sally Buck Bob

** Question 5 & 6 **

Total Names Length: 23

** Question 7 **

HelloHelloHello

** Question 8 **

John Doe

** Question 9 **

true
false

** Question 10 **

5.264

** Question 11 **

true
false

** Question 12 **

true

** Question 13 **

144.0 km = 90.0 mi
```

URL to GitHub Repository:

https://github.com/evan-m-jackson/PT-W3