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?

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

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.

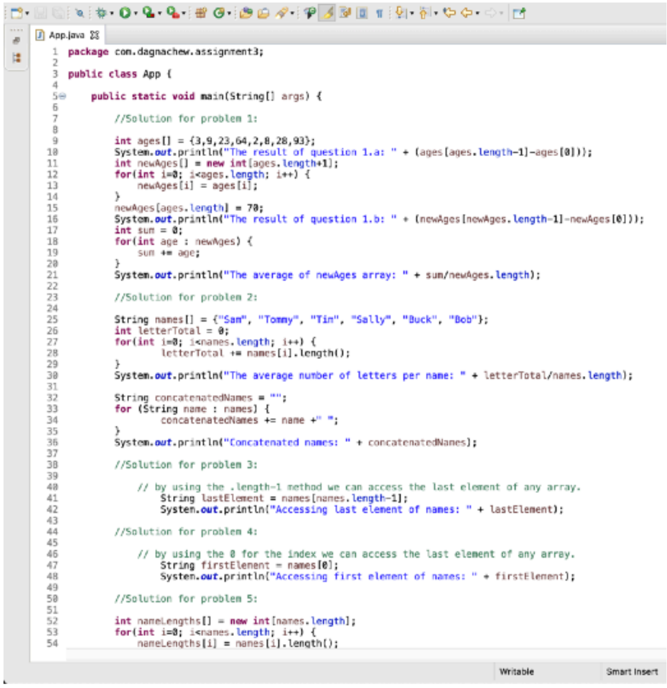
10. Write a method that takes an array of double and returns the average of all the elements in the array.

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.

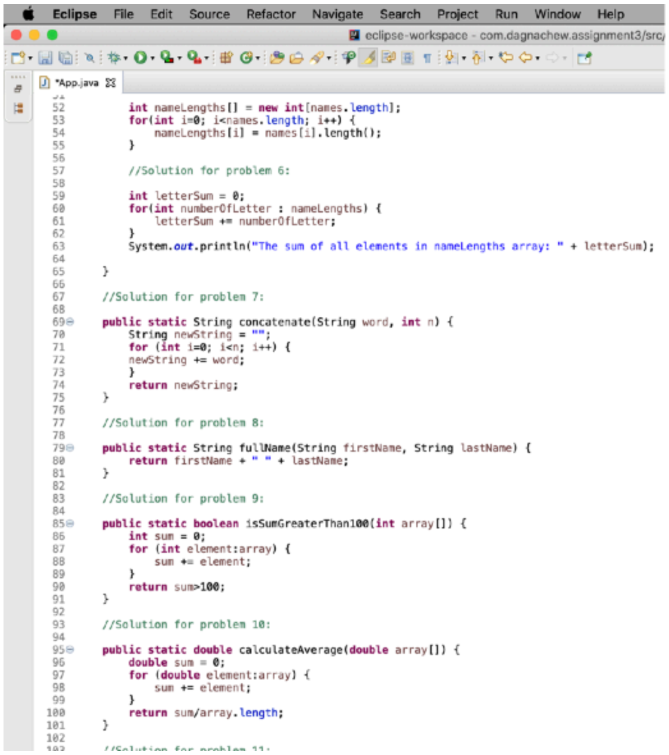
12. Write a method called willBuyDrink that takes a boolean isHotOutside, and a double moneyInPocket, and returns true if it is hot outside and if moneyInPocket is greater than 10.50.

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

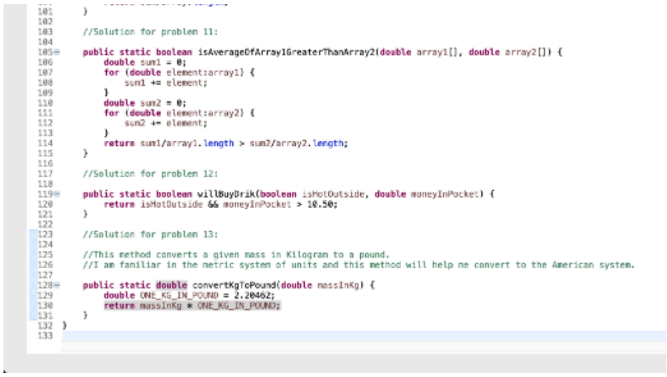
Screenshots of Code:



```
package com.dagnachew.assignment3;

public class App {

    public static void main(String[] args) {

        //Solution for problem 1:

        int ages[] = {3,9,23,64,2,8,28,93};
        System.out.println("The result of question 1.a: " + (ages[ages.length-1]-ages[0]));
        int newAges[] = new int[ages.length+1];
        for(int i=0; i<ages.length; i++) {
            newAges[i] = ages[i];
        }
        newAges[ages.length] = 70;
        System.out.println("The result of question 1.b: " + (newAges[newAges.length-1]-newAges[0]));
        int sum = 0;
        for(int age : newAges) {
            sum += age;
        }
        System.out.println("The average of newAges array: " + sum/newAges.length);

        //Solution for problem 2:

        String names[] = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
        int letterTotal = 0;
        for(int i=0; i<names.length; i++) {
                letterTotal += names[i].length();
        }
        System.out.println("The average number of letters per name: " + letterTotal/names.length);

        String concatenatedNames = "";
        for (String name : names) {
                concatenatedNames += name +" ";
        }
        System.out.println("Concatenated names: " + concatenatedNames);

        //Solution for problem 3:

            // by using the .length-1 method we can access the last element of any array.
                String lastElement = names[names.length-1];
                System.out.println("Accessing last element of names: " + lastElement);

        //Solution for problem 4:

            // by using the 0 for the index we can access the last element of any array.
                String firstElement = names[0];
                System.out.println("Accessing first element of names: " + firstElement);

        //Solution for problem 5:

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

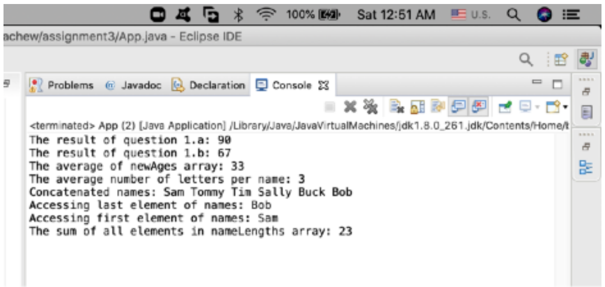


```
        int nameLengths[] = new int[names.length];
        for(int i=0; i<names.length; i++) {
            nameLengths[i] = names[i].length();
        }

        //Solution for problem 6:

        int letterSum = 0;
        for(int numberOfLetter : nameLengths) {
            letterSum += numberOfLetter;
        }
        System.out.println("The sum of all elements in nameLengths array: " + letterSum);

    }

    //Solution for problem 7:

    public static String concatenate(String word, int n) {
        String newString = "";
        for (int i=0; i<n; i++) {
        newString += word;
        }
        return newString;
    }

    //Solution for problem 8:

    public static String fullName(String firstName, String lastName) {
        return firstName + " " + lastName;
    }

    //Solution for problem 9:

    public static boolean isSumGreaterThan100(int array[]) {
        int sum = 0;
        for (int element:array) {
            sum += element;
        }
        return sum>100;
    }

    //Solution for problem 10:

    public static double calculateAverage(double array[]) {
        double sum = 0;
        for (double element:array) {
            sum += element;
        }
        return sum/array.length;
    }

    //Solution for problem 11:
```



```
    }

    //Solution for problem 11:

    public static boolean isAverageOfArray1GreaterThanArray2(double array1[], double array2[]) {
        double sum1 = 0;
        for (double element:array1) {
            sum1 += element;
        }
        double sum2 = 0;
        for (double element:array2) {
            sum2 += element;
        }
        return sum1/array1.length > sum2/array2.length;
    }

    //Solution for problem 12:

    public static boolean willBuyDrik(boolean isHotOutside, double moneyInPocket) {
        return isHotOutside && moneyInPocket > 10.50;
    }

    //Solution for problem 13:

    //This method converts a given mass in Kilogram to a pound.
    //I am familiar in the metric system of units and this method will help me convert to the American system.

    public static double convertKgToPound(double massInKg) {
        double ONE_KG_IN_POUND = 2.20462;
        return massInKg * ONE_KG_IN_POUND;
    }
}
```

Screenshots of Running Application:



```
<terminated> App (2) [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_261.jdk/Contents/Home/t
The result of question 1.a: 90
The result of question 1.b: 67
The average of newAges array: 33
The average number of letters per name: 3
Concatenated names: Sam Tommy Tim Sally Buck Bob
Accessing last element of names: Bob
Accessing first element of names: Sam
The sum of all elements in nameLengths array: 23
```

URL to GitHub Repository:

https://github.com/dwold/Week3Assignment.git