PROMINEO TECH

Intro to Java Weeks 3-4 Coding Assignment

Points possible: 75

URL to GitHub Repository: https://github.com/McKinney93/Week3_4CodingAssignment

URL to Public Link of your Video: https://www.youtube.com/watch?v=lmaqGtE_dug

——

Instructions:

1. Follow the **Coding Steps** below to complete this assignment.

- 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.

- Create a new repository on GitHub for this week's assignment and push your completed code to this dedicated repo.

- Create a video showcasing your work:

    - In this video: record and present your project verbally while showing the results of the working project.

    - Easy way to Create a video: Start a meeting in Zoom, share your screen, open Eclipse with the code and your Console window, start recording & record yourself describing and running the program showing the results.

    - Your video should be a maximum of 5 minutes.

    - Upload your video with a public link.

    - Easy way to Create a Public Video Link: Upload your video recording to YouTube with a public link.

2. In addition, please include the following in your Coding Assignment Document:

- The URL for this week's GitHub repository.

- The URL of the public link of your video.

3. Save the Coding Assignment Document as a .pdf and do the following:

- Push the .pdf to the GitHub repo for this week.

- Upload the .pdf to the LMS in your Coding Assignment Submission.

——

Intro to Java Weeks 3-4 Coding Assignment

Coding Steps — Arrays and Methods

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.

```java
14          int[] ages = {3, 9, 23, 64, 2, 8, 28, 93};
15          //System.out.println(ages[ages.length -1]);
16
17          //1a.
18          int firstLastAgeDifference = (ages[ages.length -1] - ages[0]);
19          System.out.println(firstLastAgeDifference);
20          //1b.
21          ages[ages.length - 1] = 100;
22          System.out.println(ages[ages.length -1] - ages[0]);
23          //1c.
24          double sum = 0;
25              for (int number : ages) {
26                  sum += number;
27              }
28              System.out.println(sum / ages.length);
```

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.

```java
34          String[] names = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
35
36          int sumOfLetters = 0;
37          for (String name : names) {
38              sumOfLetters += name.length();
39          }
40          //System.out.println(sumOfLetters);
41          //2a.
42          int averageLetters = sumOfLetters / names.length;
43          System.out.println(averageLetters);
44          //2b.
45          String allNames = "";
46          for (String names2 : names) {
47              allNames = allNames + names2 + " ";
48          }
49          System.out.println(allNames);
```

3. How do you access the last element of any array? **system.out.println(array(array.length -1));**

4. How do you access the first element of any array? **system.out.println(array(array[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.

```
55          int[] nameLengths = new int [names.length];
56
57          for (int i = 0; i < names.length; i++) {
58              nameLengths[i] = names[i].length();
59          }
60
61          for (int name : nameLengths) {
62              System.out.println(name);
63          }
64
```

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.

```
66
67          int sumOfNames = 0;
68          for (int name : nameLengths) {
69              sumOfNames += name;
70          }
71          System.out.println(sumOfNames);
72
```

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 expect the method to return "HelloHelloHello").

```
99      public static String multiplyString(String str, int numReps) {
100         String result = "";
101         for (int i = 0; i < numReps; i++) {
102             result += str;
103         }
104         return result;
105     }
106
75          String duplicateStr = multiplyString("Hello", 3);
76          System.out.println(duplicateStr);
```

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).

```
107     public static String fullName(String firstName, String lastName) {
108         String result = "";
109         result = firstName + " " + lastName;
110         return result;
111     }
```

```
79            String firstName = "Sally";
80            String lastName = "Lewis";
81            System.out.println(fullName(firstName, lastName));
```

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.

```
140    public static boolean isGreaterThan100(int[]nameLengths) {
141        int namesum = 0;
142        for (int nmLgth : nameLengths) {
143            namesum += nmLgth;
144        }
145        if (namesum > 100) {
146            return true;
147        } else {
148            return false;
149        }
150    }
151
```

```
91            System.out.println(isGreaterThan100(nameLengths));
```

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

```
121    public static double groceryAverage(double[] avgCostOfGroceries) {
122        double sumOfGroceries = 0;
123        for (double cost : avgCostOfGroceries) {
124            sumOfGroceries += cost;
125        }
126        return sumOfGroceries / avgCostOfGroceries.length;
127    }
128
```

```
95            double[] avgCostOfGroceries = {76.45, 65.33, 86.29, 70.96};
96            System.out.println(groceryAverage(avgCostOfGroceries));
```

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.

```
145    public static boolean isGroceriesMore(double[] avgCostOfGroceries, double[] avgCostOfTakeOut) {
146        double groceriesMore = 0;
147        double takeOutLess = 0;
148        for (double x : avgCostOfGroceries) {
149            groceriesMore += x;
150        }
151        for (double y : avgCostOfTakeOut) {
152            takeOutLess += y;
153        }
154        if (groceriesMore / avgCostOfGroceries.length > takeOutLess / avgCostOfTakeOut.length) {
155            return true;
156        }
157        else {
158            return false;
159        }
```

```
100            double[] avgCostOfTakeOut = {57.89, 65.93, 47.50, 54.36};
101            System.out.println(isGroceriesMore(avgCostOfGroceries, avgCostOfTakeOut));
```

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.

```java
163    public static boolean willBuyDrink(boolean isHotOutside, double moneyInPocket) {
164        if (isHotOutside == true && moneyInPocket > 10.50) {
165            return true;
166        }
167        return false;
168    }
```

```java
105        boolean isHotOutside = true;
106        double moneyInPocket = 10.00;
107        System.out.println(willBuyDrink(isHotOutside, moneyInPocket));
108        System.out.println("Question 13");
```

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

```java
197    public static int spentInOneDay(int[] store1, int[] store2) {
198        int cost1 = 0;
199        int cost2 = 0;
200        for (int item : store1) {
201            cost1 += item;
202        }
203        for (int item2 : store2) {
204            cost2 += item2;
205        }
206        return cost1 + cost2;
207    }
```

```java
110        int[] store1 = {54, 22};
111        int[] store2 = {100, 56, 64};
112        System.out.println(spentInOneDay(store1, store2));
113
```