

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.

- How do you access the last element of any array?
- How do you access the first element of any array?
- 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.
- 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.
- 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").
- 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).
- 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.
- Write a method that takes an array of double and returns the average of all the elements in the array.
- 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.
- 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.
- 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:

```

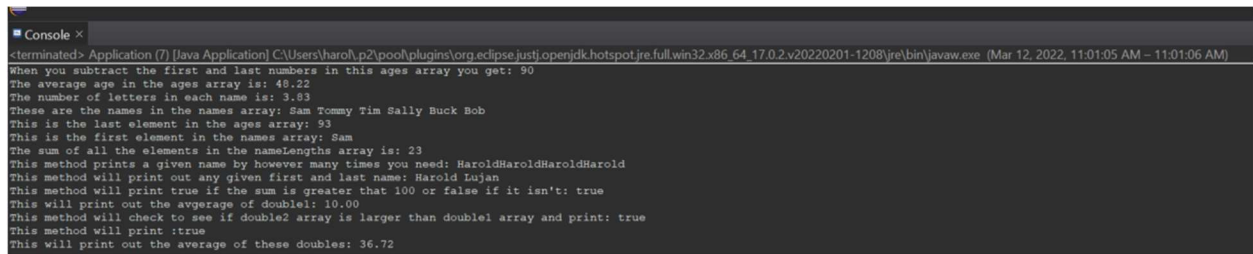
Application.java
Week3CodingAssignment/src/Application.java
1  // TODO Auto-generated method stub
2  // Question 1 Create an array of int called ages that contains the following values: 3, 9, 23, 64, 2, 8, 28, 93.
3  //Question 1b. added a new age to the array 204.
4  int [] ages = {3, 9, 23, 204, 64, 2, 8, 28, 93};
5
6  // Question 1a. Subtract the first number in the array from the last number. took the
7  // subtractAgeValues = ages[ages.length -1] - ages[0];
8  System.out.println("When you subtract the first and last numbers in this ages array you get: " + subtractAgeValues);
9
10 //Question 1c. Use a loop to iterate through the array and calculate the average age.
11 double avgSum = 0;
12 for (int age : ages) {
13     avgSum += age;
14 }
15 System.out.println("The average age in the ages array is: " + avgSum / ages.length);
16
17 //Question 2 Create an array of String called names that contains the following values: "Sam", "Tommy", "Tim", "Sally", "Buck", "Bob".
18 String [] names = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
19
20 //Question 2a. Use a loop to iterate through the array and calculate the average number of letters per name. Print the result to the console.
21 double avgChar = 0;
22 for (String name : names) {
23     avgChar += name.length();
24 }
25 System.out.println("The number of letters in each name is: " + avgChar / names.length);
26
27 //Question 2b. 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.
28 String name1 = "";
29 for (String name : names) {
30     name1 = name1 + name + " ";
31 }
32 System.out.println("These are the names in the names array: " + name1);
33
34 //Question 3 How do you access the last element of any array?
35 System.out.println("This is the last element in the ages array: " + ages[ages.length - 1]);
36
37 //Question 4 How do you access the first element of any array?
38 System.out.println("This is the first element in the names array: " + names[0]);
39
40 //Question 5 Create a new array of int called nameLengths.
41 //Write a loop to iterate over the previously created names array and add the length of each name to the nameLengths array.
42 int [] nameLengths = new int [names.length];
43
44 for (int n = 0; n < names.length; n++) {
45     String name = names[n];
46     nameLengths[n] = name.length();
47 }
48
49 //Question 6. Write a loop to iterate over the nameLengths array and calculate the sum of all the elements in the array.
50 //Print the result to the console.
51 int sumOfNames = 0;
52 for (int sum : nameLengths) {
53     sumOfNames += sum;
54 }
55 System.out.println("The sum of all the elements in the nameLengths array is: " + sumOfNames);
56
57 // the output of question 7
58 System.out.println("This method prints a given name by however many times you need: " + greeting("Harold", 4));
59
60 //the output of question 8
61 System.out.println("This method will print out any given first and last name: " + fullName("Harold", "Lujan"));
62
63 //the output of question 9
64 System.out.println("This method will print true if the sum is greater than 100 or false if it isn't: " + sumInts(ages));
65
66 // creating the first double array and printing out its average for question 10.
67 double[] double1 = {2.99, 4.75, 11.29, 20.99};
68 System.out.println("This will print out the average of double1: " + sumAvg(double1));
69
70 //creating the second array that will get compared to the first array to print out true or false is its greater than 100 for question 11.
71 double[] double2 = {4.99, 2.12, 9.00, 12.25};
72

```

Screenshots of Code Continued:

```
Application.java ×
73 System.out.printf("This method will check to see if double2 array is larger than double1 array and print: %s\n", sumAvgDouble(double1, double2));
74
75 //this is the output for question 12
76 boolean isHotOutside = true;
77 double moneyInPocket = 10.50;
78 System.out.printf("This method will print :%s\n", willBuyDrink(isHotOutside, moneyInPocket));
79
80 //this is the output for question 13.
81 double x = 29.75;
82 double y = 34.50;
83 double z = 45.90;
84 System.out.printf("This will print out the average of these doubles: %.2f", doubleAvg(x, y, z));
85 } // end of main method
86
87 //Question 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.
88 //i.e. if I pass in "Hello" and 3, I would expect the method to return "HelloHelloHello".
89 public static String greeting(String word, int n) {
90     String wordConcat = "";
91     for(int i = 0; i < n; i++) {
92         wordConcat += word;
93     }
94     return wordConcat;
95 }
96
97 //Question 8 Write a method that takes two Strings, firstName and lastName, and returns a full name.
98 //the full name should be the first and the last name as a String separated by a space).
99 public static String fullName(String firstName, String lastName) {
100     return firstName + " " + lastName;
101 } // end of fullName method
102
103 //Question 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.
104 public static boolean sumInts(int[] num) {
105     int total = 0;
106     for(int num1 : num) {
107         total += num1;
108     }
109
110     if (total > 100) {
111         return true;
112     }
113     else {
114         return false;
115     }
116 } //end of sumInts method
117
118 //Question 10 Write a method that takes an array of double and returns the average of all the elements in the array.
119 public static double sumAvg(double[] double1) {
120     double total = 0;
121     for(double num1 : double1) {
122         total += num1;
123     }
124     return total/double1.length;
125 }
126
127 //Question 11 Write a method that takes two arrays of double
128 //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.
129 public static boolean sumAvgDouble(double[] firstDouble, double[] secondDouble) {
130     if(sumAvg(firstDouble) > sumAvg(secondDouble)) {
131         return true;
132     }
133     return false;
134 } //end of sumAvgDouble method
135
136 //Question 12 Write a method called willBuyDrink that takes a boolean isHotOutside, and a double moneyInPocket,
137 //and returns true if it is hot outside and if moneyInPocket is greater than 10.50.
138 public static boolean willBuyDrink(boolean isHotOutside, double moneyInPocket) {
139     if(isHotOutside && moneyInPocket >= 10.50) {
140         return true;
141     }
142     return false;
143 } //end of willBuyDrink method
144
145
146 //Question 13 Create a method of your own that solves a problem. In comments, write what the method does and why you created it.
147 // please see added java doc for information on what this method does.
148 /**
149  * takes the sum of 3 doubles and returns the average.
150  * @param x the first double
151  * @param y the second double
152  * @param z the third double
153  * @return takes the sum of all 3 doubles and divides them by 3.
154  */
155 public static double doubleAvg(double x, double y, double z) {
156     return (x + y + z)/3;
157 }
158 } // end of application
```

Screenshots of Running Application:

A screenshot of a Java application's console output. The window title is "Console x". The output text is as follows:

```
<terminated> Application (7) [Java Application] C:\Users\harol\AppData\Local\Temp\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (Mar 12, 2022, 11:01:05 AM - 11:01:06 AM)
When you subtract the first and last numbers in this ages array you get: 90
The average age in the ages array is: 48.22
The number of letters in each name is: 3.83
These are the names in the names array: Sam Tommy Tim Sally Buck Bob
This is the last element in the ages array: 93
This is the first element in the names array: Sam
The sum of all the elements in the nameLengths array is: 23
This method prints a given name by however many times you need: HaroldHaroldHaroldHarold
This method will print out any given first and last name: Harold Lujan
This method will print true if the sum is greater than 100 or false if it isn't: true
This will print out the average of double1: 10.00
This method will check to see if double2 array is larger than double1 array and print: true
This method will print :true
This will print out the average of these doubles: 36.72
```

URL to GitHub Repository:

<https://github.com/HaroldLujan/Week-3-Coding-Assignment.git>