

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:

```
1 package week3CodeAssignment;
2
3
4
5 public class Week3CodeAssignment {
6
7     public static void main(String[] args) {
8
9         int[] ages = {3, 9, 23, 64, 2, 8, 28, 93, 45};
10
11         int firstNumber = 0;
12         int lastNumber = 0;
13
14         //subtracting the first number of the array from the last number
15
16         for(int i = 0; i < ages.length; i++) {
17             if(i < 1) {
18                 firstNumber = ages[i];
19             }
20         }
21
22         for(int i1 = 0; i1 < ages.length; i1++) {
23             if(i1 > -1) {
24                 lastNumber = ages[i1];
25             }
26         }
27
28         System.out.println("The difference of the last number and the first number is:" + " " + (lastNumber - firstNumber));
29
30         //calculating the average age in the array
31
32         int sum = 0;
33
34         for(int i = 0; i < ages.length; i++) {
35             sum = ages[i] + sum;
36         }
37     }
```

```

38
39     int average = sum / ages.length;
40     System.out.println("The average of my array is:" + " " + average);
41
42     //finding the average length of name in this array
43
44     String[] names = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
45
46     int sumOfLetters = 0;
47
48     for(int i = 0; i < names.length; i++) {
49
50         String allNames = names[i];
51         sumOfLetters += allNames.length();
52     }
53
54     System.out.println("The average length of these names is:" + " " + sumOfLetters / names.length);
55
56     //Concatenating all the names together with another loop
57
58     String namesTogether = "";
59
60     for(int i = 0; i < names.length; i++) {
61         namesTogether = names[i] + " " + namesTogether;
62     }
63     System.out.println("All of the names together are:" + " " + namesTogether);
64
65     //To access the last number of an array you would call the array and choose the first number which is always zero
66     // example : System.out.println(testArray[0]);
67
68     //To access the last number of an array you would call the length of the array -1
69     //example : System.out.println(testArray[testArray.length - 1]);
70
71     //add the number of letters in each name to a new string called NamesLengths
72
73     int[] namesLengths = new int[6];
74
75     for(int i = 0; i < namesLengths.length; i++) {
76         namesLengths[i] += names[i].length();
77     }
78
79
80     //calculate the sum of all the values in the namesLengths array
81     int sumOfLengths = 0;
82
83     for(int i = 0; i < namesLengths.length; i++) {
84         sumOfLengths = namesLengths[i] + sumOfLengths;
85     }
86     System.out.println("The sum of letters in all these names is: " + sumOfLengths);
87
88     //using a method to repeat a string a number of times depending on an int value
89
90     String repeat = duplicate("Hello", 4);
91     System.out.println(repeat);
92
93     //demo of first + last name method
94     String fullName = fullName("Jon", "Vinson");
95     System.out.println("My full name is: " + fullName);
96
97     //demo of array greater than 100 method
98     int[] randomNumbers = {1, 90, 14, 5, 7, 9};
99     boolean greaterThan100 = greaterThan100(randomNumbers);
100    System.out.println("The sum of this array greater than 100: " + greaterThan100);
101
102    //demo taking average of doubles
103    double[] randomDoublesTwo = {12.5, 14.7, 33.2, 9.22, 11.41};
104    double[] randomDoubles = {1.3, 2.3, 5.5, 77.5};
105    double averageDoubles = averageOfDouble(randomDoubles);
106    System.out.println("The average of this array is:" + averageDoubles);
107
108    //demo is double array 1 average greater than double array 2 average
109
110    boolean doublesGreaterThen = averageGreaterThan(randomDoubles, randomDoublesTwo);
111    System.out.println("array 1 average is greater than array 2 average: " + doublesGreaterThen);
112

```

```

112
113 //demo can I buy a drink method
114
115 boolean isHotOutside = true;
116 double moneyInPocket = 10.50;
117 boolean canIbuyDrink = willBuyDrink(isHotOutside, moneyInPocket);
118 System.out.println("Can I buy a drink? " + canIbuyDrink);
119
120 //demo stock counter method
121
122 int amountOfShirts = 50;
123 int amountOfPants = 50;
124 boolean stockCheck = stockAmount(amountOfShirts, amountOfPants);
125 System.out.println("Is there enough stock in the store? " + stockCheck);
126
127
128 }
129
130 //Method to print input word a number of times
131
132 public static String duplicate(String word, int n) {
133     String x = "";
134     for(int i = 0; i < n; i++) {
135         x += word;
136     }
137     return x;
138 }
139
140 //method that takes firstName and LastName and returns a full name
141
142 public static String fullName(String firstName, String lastName) {
143     String n = "";
144     n = firstName + " " + lastName;
145     return n;
146 }
147
148 //Takes array of int and returns true if sum is greater than 100
149
150 public static boolean greaterThan100(int[] array) {
151     boolean x = false;
152     int sum = 0;
153     for(int i = 0; i < array.length; i++) {
154         sum = array[i] + sum;
155     }
156     if(sum > 100) {
157         x = true;
158     }
159     else {
160         x = false;
161     }
162     return x;
163 }
164
165 //take a string of double and return an average
166
167 public static double averageOfDouble(double[] array) {
168     double x = 0;
169     double sum = 0;
170     for(int i = 0; i < array.length; i++) {
171         sum = array[i] + sum;
172     }
173     x = sum / array.length;
174
175     return x;
176
177 }
178
179
180 //is double 1 array average greater than double 2 array average
181
182 public static boolean averageGreaterThan(double[] array, double[] arrayTwo) {
183     boolean x = true;
184     double sum = 0;
185     double sum2 = 0;

```

```

186         double averageOne = 0;
187         double averageTwo = 0;
188
189         for(int i = 0; i < array.length; i++) {
190             sum = array[i] + sum;
191         }
192         averageOne = sum / array.length;
193
194         for(int i = 0; i < arrayTwo.length; i++) {
195             sum2 = arrayTwo[i] + sum2;
196         }
197         averageTwo = sum2 / arrayTwo.length;
198
199         x = averageOne > averageTwo;
200         return x;
201     }
202 }
203
204 //willBuyDrink method
205
206
207● public static boolean willBuyDrink(boolean isHotOutside, double moneyInPocket) {
208     boolean x = true;
209     double enoughMoney = 10.50;
210     x = isHotOutside && (moneyInPocket > enoughMoney);
211
212     return x;
213 }
214
215 //stock counting method for store
216
217● public static boolean stockAmount(int x, int y) {
218     boolean enoughStock = true;
219     enoughStock = (x >= 45) && (y >= 25);
220     return enoughStock;
221 }
222
223 }

```

Screenshots of Running Application:

```

<terminated> Week3CodeAssignment [Java Application] C:\Users\jmvle\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.
The difference of the last number and the first number is: 42
The average of my array is: 30
The average length of these names is: 3
All of the names together are: Bob Buck Sally Tim Tommy Sam
The sum of letters in all these names is: 23
HelloHelloHelloHello
My full name is: Jon Vinson
The sum of this array greater than 100: true
The average of this array is:21.65
array 1 average is greater than array 2 average: true
Can I buy a drink? false
Is there enough stock in the store? true

```

URL to GitHub Repository:

<https://github.com/JonVinson24/Week-3-Coding-Assignment>