



Intro to Java Weeks 3-4 Coding Assignment

Points possible: 75

URL to GitHub Repository: <https://github.com/oscarc257/First-Coding-Assignment.git>

URL to Public Link of your Video:

<https://www.dropbox.com/s/gvr1opwsd5bemqk/video2297771951.mp4?dl=0>

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.
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 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.



PROMINEO TECH

Intro to Java Weeks 3-4 Coding Assignment

```
1 package firstCoding;
2
3 import java.util.Arrays;
4
5 import java.util.OptionalDouble;
6
7 public class CodingAssign {
8
9     public static void main(String[] args) {
10
11         // 1. =====
12
13         int[] ages = {3, 9, 23, 64, 2, 8, 28, 93}; // Array Literal; since size and variables are known.
14
15         // a. subtract the value of the first element in the array from the value in the last element of the array
16
17         int lengthOfArr = ages.length;
18         int difference = Math.abs(ages[0] - ages[ages.length - 1]);
19
20         System.out.println("Difference of first and last element in ages[] = " + difference);
21
22         int[] ages2 = new int[ages.length];
23
24         lengthOfArr = ages2.length;
25         ages2[ages2.length - 1] = 100;
26         difference = Math.abs(ages2[0] - ages2[ages2.length - 1]);
27
28         System.out.println("Difference of first and last element in ages2[] = " + difference);
29
30         // b. find the average of age
31
32         int sum = 0;
33
34         int i;
35         for (i = 0; i < ages2.length; i++) {
36             sum += ages2[i];
37         }
38     }
39 }
```

Console Output:

```
<terminated> CodingAssign [Java Application] C:\Users\oscar\p2\pool\plugins\org.eclipse.justi.openjdk hotspot.jre.full.win32.x86_64.17.0.4\j20220805-1047\jre\bin\javaw.exe (Oct 1, 2022, 7:28:57 PM) [pid: 41208]
Difference of first and last element in ages[] =90
```



Intro to Java Weeks 3-4 Coding Assignment

```
34 // b. find the average of age
35
36 int sum = 0;
37
38 int i;
39 for (i = 0; i < ages2.length; i++) {
40     sum += ages2[i];
41 }
42
43 double average = sum / ages2.length;
44 System.out.println("The average age in ages2[] = " + average);
45
46
47
48
49
50 // 2. =====
51 String[] everyName = {"Sam", "Tommy", "Tim", "Sally", "Buck", "Bob"};
52
53 // a. create a loop to iterate through the array and calculate average number of letters
54
55 sum = 0;
56
57 for (i = 0; i < everyName.length; i++) {
58     sum += everyName[i].length();
59 }
60
61 average = sum / everyName.length;
62 System.out.println("The average number of letters in everyName[] = " + average);
63
64 // b. create a loop to iterate through array and concatenate all names at once, separated by spaces.
65
66 String Names = "";
67
68 for (i = 0; i < everyName.length; i++) {
69     Names += everyName[i] + " ";
70 }
71
72 System.out.println(Names);
73
74
```

Package Explorer X

- Week 3_Practice
- Week 1 Github
- Week 2t
- Week 3t
- Week 4 Coding Assignment
 - JRE System Library (JavaSE-17)
 - src
 - firstCoding
 - CodingAssign.java
 - package-info.java
 - Week 4t
 - Week Java Practice

Git Repositories X

- beginner-repository (main) - C:\Users\oscar\p2\pool\plugin\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220805-1047\jre\bin\java.exe (Oct 1, 2022, 7:28:57 PM) [pid: 41208]
- beginner-repository (main) - C:\Users\oscar\p2\pool\plugin\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220805-1047\jre\bin\java.exe (Oct 1, 2022, 7:28:57 PM) [pid: 41208]
- beginner-repository (main) - C:\Users\oscar\p2\pool\plugin\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220805-1047\jre\bin\java.exe (Oct 1, 2022, 7:28:57 PM) [pid: 41208]

Outline X

- firstCoding
 - CodingAssign
 - main(String[]) : void
 - greaterThan100(int[]) : boolean
 - getAverageOfDoubleArr(double[]) : double
 - isFirstArrAvgGreater(double[], double[]) : boolean
 - willBuyDrink0 : boolean
 - [...]
 - equals(double[], double[]) : boolean

Console X

```
terminated> CodingAssign [Java Application] C:\Users\oscar\p2\pool\plugin\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4.v20220805-1047\jre\bin\java.exe (Oct 1, 2022, 7:28:57 PM) [pid: 41208]
Difference of first and last element in ages[] -90
```



Intro to Java Weeks 3-4 Coding Assignment

```
76
77
78 // 3. =====
79
80 System.out.println("You must use a[arr.length] to access the last element of an array.");
81
82
83
84
85 // 4. =====
86
87
88 System.out.println("You must use a[0] to access the last element of an array.");
89
90
91
92
93 // 5. =====
94
95 int[] nameLengths = new int[everyName.length];
96 for (i = 0; i < everyName.length; i++) {
97     nameLengths[i] = everyName[i].length();
98 }
99
100 // 6. =====
101
102 sum = 0;
103
104
105 for ( i = 0; i < nameLengths.length; i++) {
106     sum += nameLengths[i];
107 }
108
109 System.out.println("The total of all lengths in nameLengths[] = " + sum);
110
111
112 // 7. =====
113
114 String str = "Oscar";
115
116 System.out.println( str.repeat(4) );
```

firstCoding.CodingAssign - Week 4 Coding Assignment/src

Console: Difference of first and last element in ages[] =90



Intro to Java Weeks 3-4 Coding Assignment

```
eclipse-workspace - Week 4 Coding Assignment/src/firstCoding/CodingAssign.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

CodingAssign.java x
85 // 4. =====
86
87
88 System.out.println("You must use a[0] to access the last element of an array.");
89
90
91
92
93 // 5. =====
94
95 int[] nameLengths = new int[everyName.length];
96 for (i = 0; i < everyName.length; i++) {
97     nameLengths[i] = everyName[i].length();
98 }
99
100 // 6. =====
101
102 sum = 0;
103
104
105 for ( i = 0; i < nameLengths.length; i++) {
106     sum += nameLengths[i];
107 }
108
109 System.out.println("The total of all lengths in nameLengths[] = " + sum);
110
111 // 7. =====
112
113 String str = "Oscar";
114
115 System.out.println( str.repeat(4) );
116
117
118
119
120 // 8. =====
121
122 String firstName = "Oscar";
123 String lastName = "Carrillo";
124 String fullName = firstName + " " + lastName;
125
126 System.out.println(fullName);
127
128 }
129
130
```

```
eclipse-workspace - Week 4 Coding Assignment/src/firstCoding/CodingAssign.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

CodingAssign.java x
133 // Calls for following methods
134
135 int [] newAge = {105, 129, 124, 113};
136 System.out.println(greaterThan100 (newAge));
137
138 double[] doubleArr = {0.5, 9.0, 245.4, 78.1};
139 System.out.println("getAverageOfDoubleArr() returns => " + getAverageOfDoubleArr(doubleArr));
140
141 double[] anotherDoubleArr = {7.2, 9.3, 8.4, 7.5};
142 System.out.println("isFirstArrAvgGreater returns => " + isFirstArrAvgGreater(doubleArr, anotherDoubleArr));
143
144
145 boolean isHotOutside = true;
146 double moneyInPocket = 153.20;
147
148 String arr3 = new String();
149 double[] arr1 = { 100.0, 14.0, 16.5, 11.1 };
150 double[] arr2 = { 90.0, 12.9, 16.5, 11.1 };
151 System.out.println(Arrays.equals(arr1, arr2));
152
153 }
154
155
156
157 // 9. =====
158
159
160
161
162 static boolean greaterThan100(int [] newAge) {
163
164     int arraySum = 0;
165     boolean greater100;
166
167     for (int i = 0; i < newAge.length; i++) {
168         arraySum += newAge[i];
169     }
170
171     if (arraySum > 100) {
172         return greater100 = true;
173     }else {
174         greater100 = false;
175         return greater100;
176     }
177 }
178
```



Intro to Java Weeks 3-4 Coding Assignment

```
eclipse-workspace - Week 4 Coding Assignment/src/firstCoding/CodingAssign.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

CodingAssign.java x
201
202
203
204
205
206
207
208
209 // 12. =====
210 boolean willBuyDrink() {
211     return true;
212 } {
213     boolean isHotOutside = true;
214     double moneyInPocket = 153.20;
215 }
216
217
218 //13. =====
219
220 //This method is to test if all values in both arrays are equal.
221
222
223 static boolean equals(double[] arr1, double[] arr2) {
224     if (arr1.length != arr2.length)
225         return false;
226
227     for (int i = 0; i < arr1.length; ++i) {
228         if (Math.abs(arr1[i] - arr2[i]) < 10.2) {
229             return false;
230         }
231     }
232
233     return true;
234 }
235
236
237
238
239
240
241
242
243
244
245 }
246
< | Writable | Smart Insert | 8:1:117 |
```