

# Intro to Java Week 4 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.

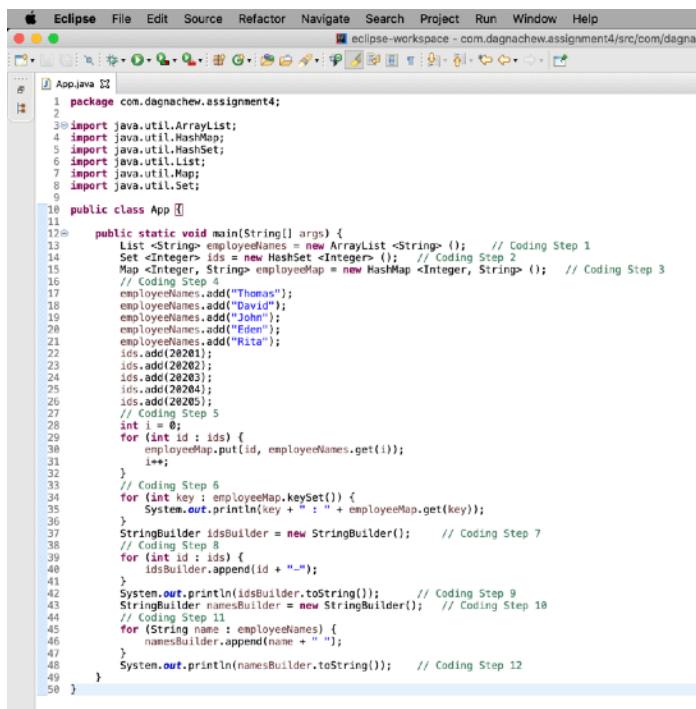
## Coding Steps:

1. Create an instance of an ArrayList of String called employeeNames
2. Create an instance of a HashSet of Integer called ids
3. Create an instance of a HashMap of Integer, String called employeeMap
4. Add at least five entries to the employeeNames and ids (make sure both collections have the same number of entries).
5. Create a variable **int i = 0;** then iterate over ids using an enhanced for loop. Inside the enhanced for loop use **employeeMap.put()** to add a new entry to the map. The entry should consist of a key that is the id in the enhanced loop's current iteration, and a value

that is the `employeeName` at position `i` of the `employeeNames` `ArrayList`. Increment `i` so that each iteration grabs the next element in the `ArrayList`.

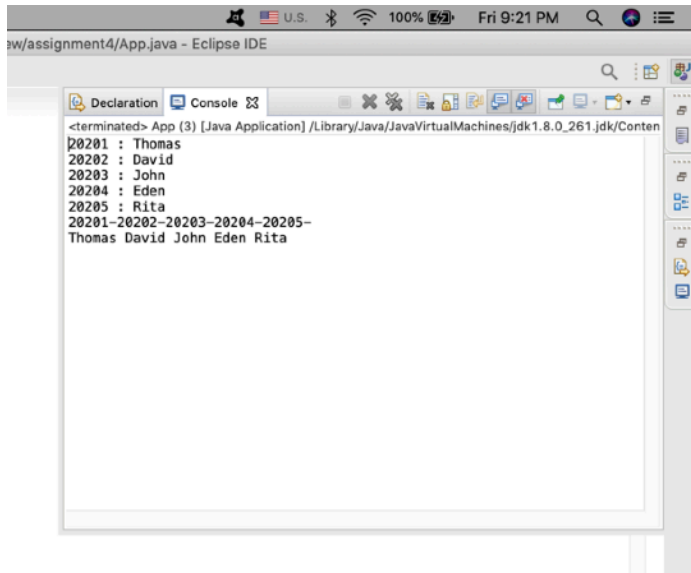
6. Once the `employeeMap` is fully populated, use another enhanced for loop to iterate over the **`employeeMap.keySet()`**, and use the key for each current iteration to print to the console both the current key and its associated value in the map.
7. Create a `StringBuilder` called `idsBuilder`.
8. Iterate over the `ids` `HashSet` and append each id, followed by a dash “-” to `idsBuilder`.
9. Print the result of **`idsBuilder.toString()`** to the console.
10. Create another `StringBuilder` called `namesBuilder`.
11. Iterate over the `employeeNames` `ArrayList` and append each name, followed by a space “ ” to the `namesBuilder`.
12. Print the result of **`namesBuilder.toString()`** to the console.

## Screenshots of Code:



```
1 package com.dagnachew.assignment4;
2
3 import java.util.ArrayList;
4 import java.util.HashMap;
5 import java.util.HashSet;
6 import java.util.List;
7 import java.util.Map;
8 import java.util.Set;
9
10 public class App {
11
12     public static void main(String[] args) {
13         List<String> employeeNames = new ArrayList<String>(); // Coding Step 1
14         Set<Integer> ids = new HashSet<Integer>(); // Coding Step 2
15         Map<Integer, String> employeeMap = new HashMap<Integer, String>(); // Coding Step 3
16         // Coding Step 4
17         employeeNames.add("Thomas");
18         employeeNames.add("David");
19         employeeNames.add("John");
20         employeeNames.add("Eden");
21         employeeNames.add("Rita");
22         ids.add(28201);
23         ids.add(28202);
24         ids.add(28203);
25         ids.add(28204);
26         ids.add(28205);
27         // Coding Step 5
28         int i = 0;
29         for (int id : ids) {
30             employeeMap.put(id, employeeNames.get(i));
31             i++;
32         }
33         // Coding Step 6
34         for (int key : employeeMap.keySet()) {
35             System.out.println(key + " : " + employeeMap.get(key));
36         }
37         StringBuilder idsBuilder = new StringBuilder(); // Coding Step 7
38         // Coding Step 8
39         for (int id : ids) {
40             idsBuilder.append(id + "-");
41         }
42         System.out.println(idsBuilder.toString()); // Coding Step 9
43         StringBuilder namesBuilder = new StringBuilder(); // Coding Step 10
44         // Coding Step 11
45         for (String name : employeeNames) {
46             namesBuilder.append(name + " ");
47         }
48         System.out.println(namesBuilder.toString()); // Coding Step 12
49     }
50 }
```

## Screenshots of Running Application:



## URL to GitHub Repository:

<https://github.com/dwold/Week4Assignment>