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.

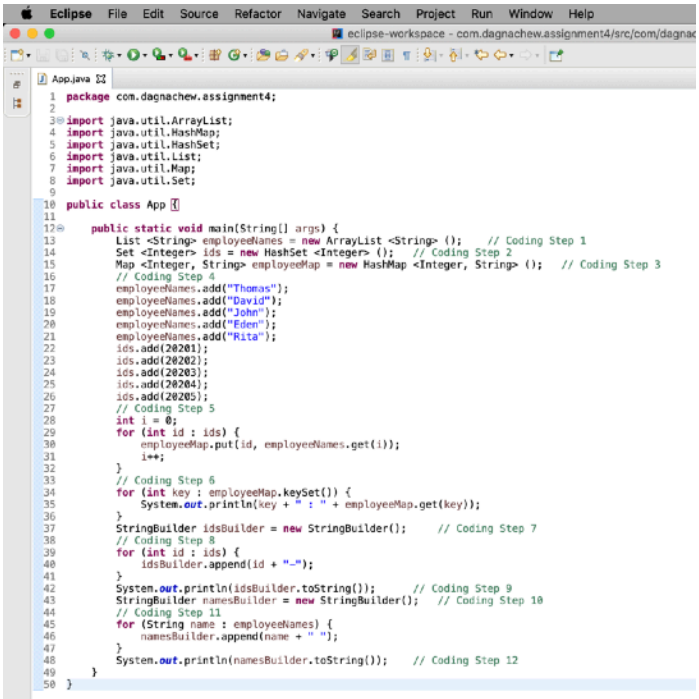
Coding Steps:

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

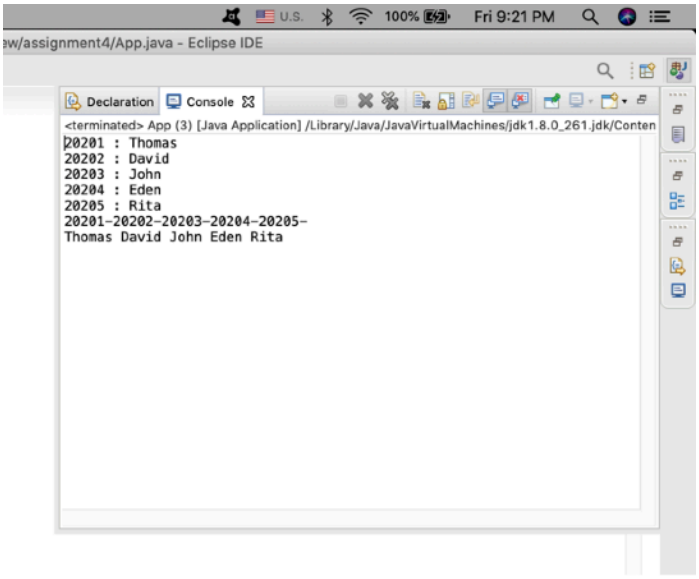
that is the employeeName at position **i** of the employeeNames ArrayList. Increment i so that each iteration grabs the next element in the ArrayList.

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:



Screenshots of Running Application:



URL to GitHub Repository:

https://github.com/dwold/Week4Assignment