**Java8 Magic using Stream**

**Author**: Rabi

**Source**: <http://www.dailycodebuffer.com/>

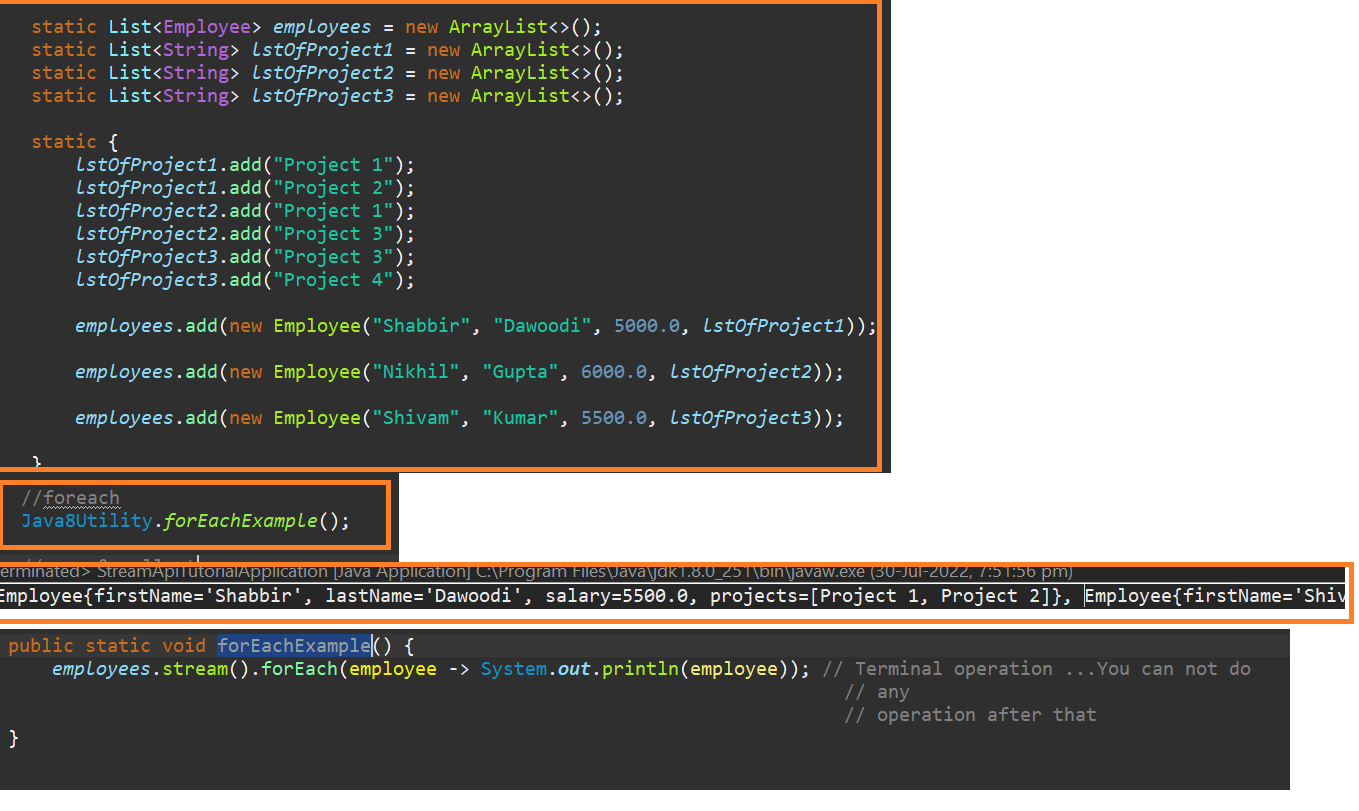
<https://www.youtube.com/c/DailyCodeBuffer>

**Date**: 30th July 2022



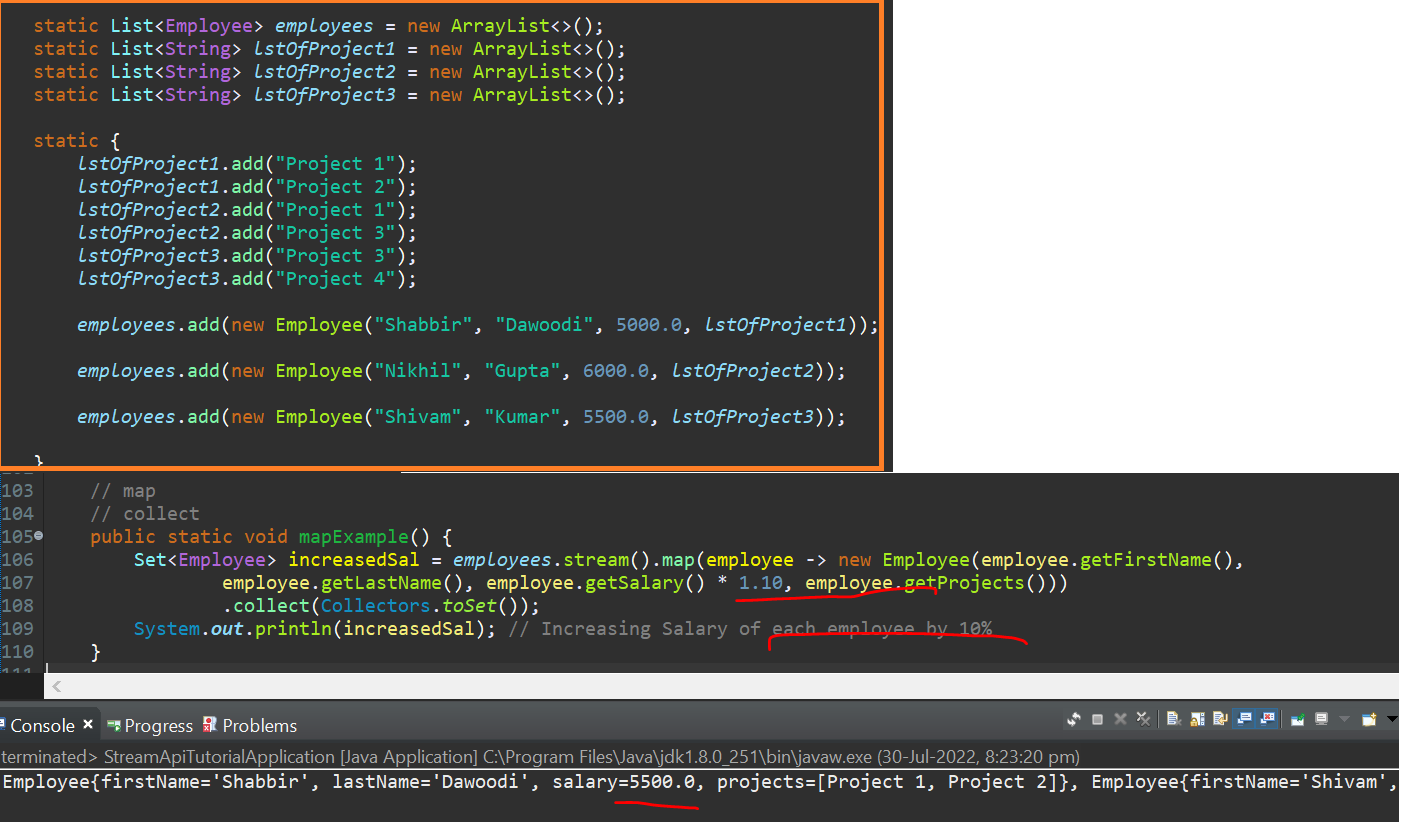
1. **forEach()** – Terminal operation . You can not do any operation after that.

(print object)



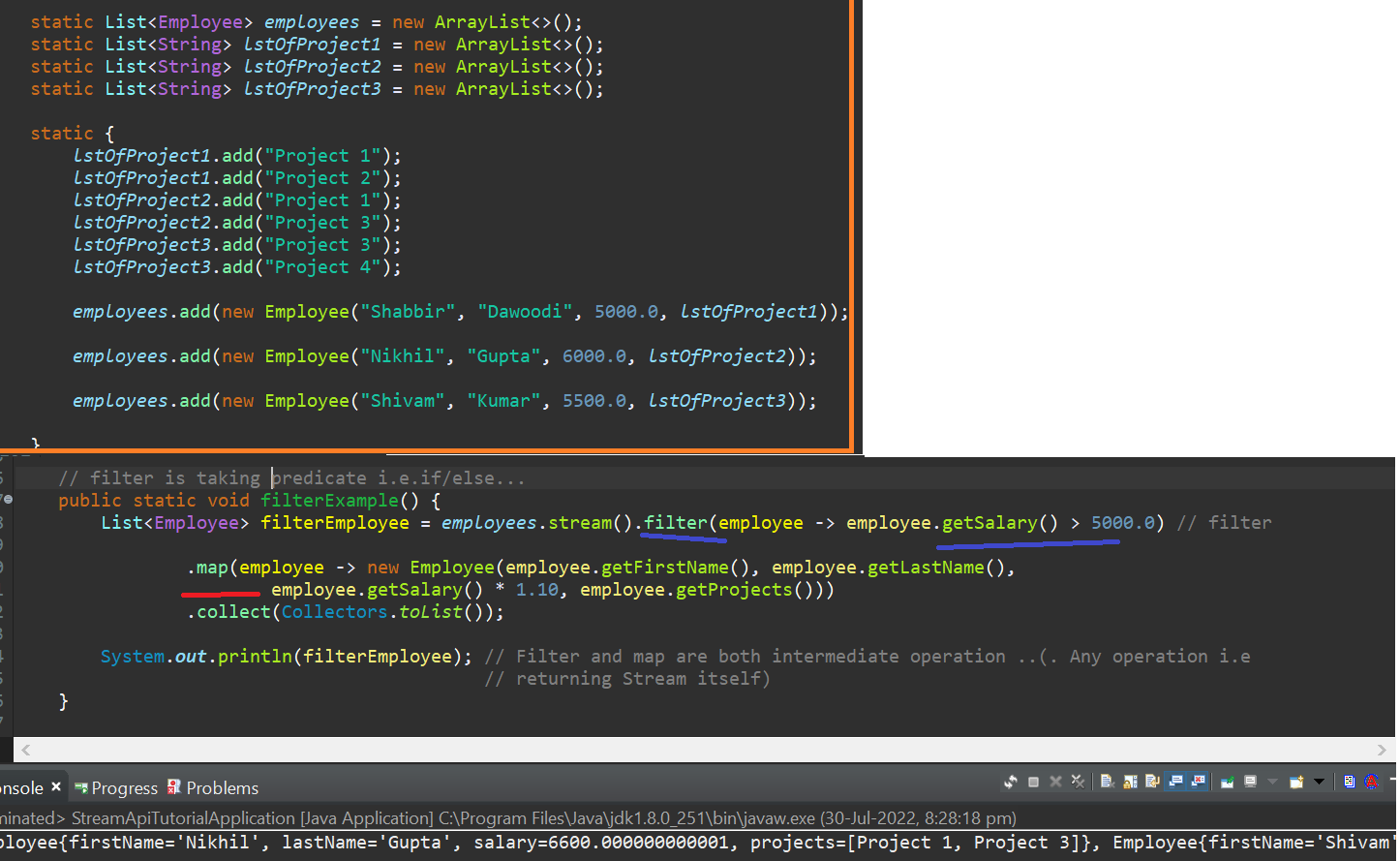
1. **map()** - intermediate operation

(increase salary by 10%)



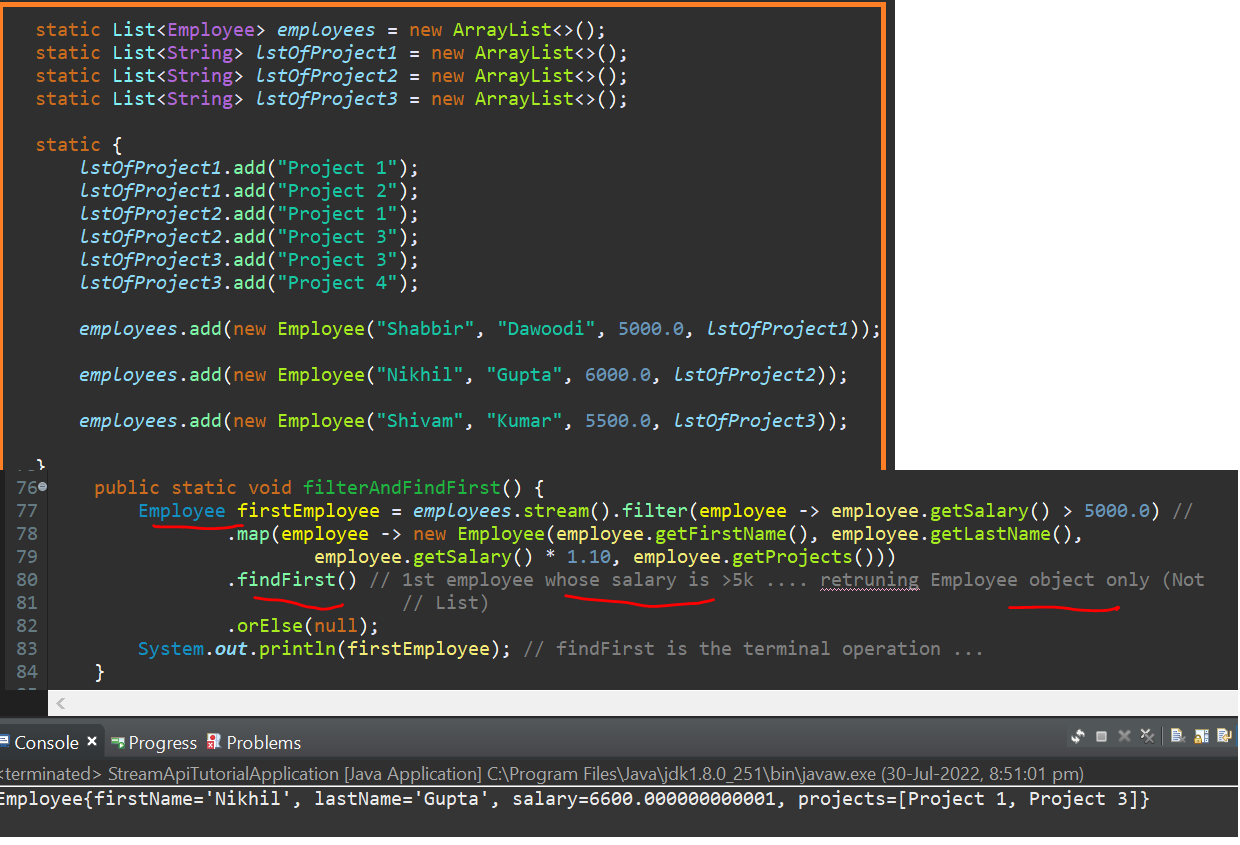
1. **Filter() +map()** - Both are intermediate operation and returning Stream only

(more than >5k and 10% salary increase)

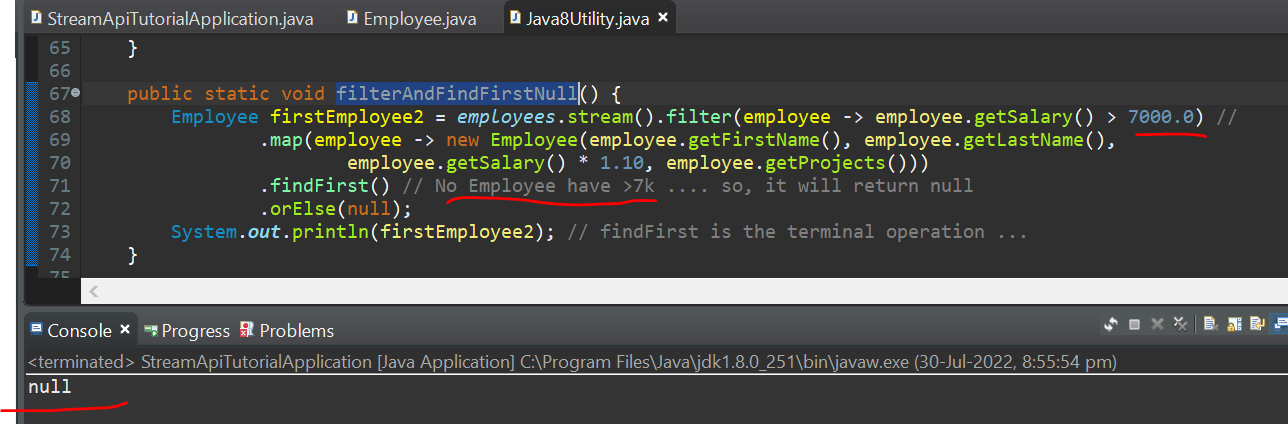


1. **findFirst()** – **Terminal** operation

(1st employee whose salary is 10% and return Employee object NOT LIST )

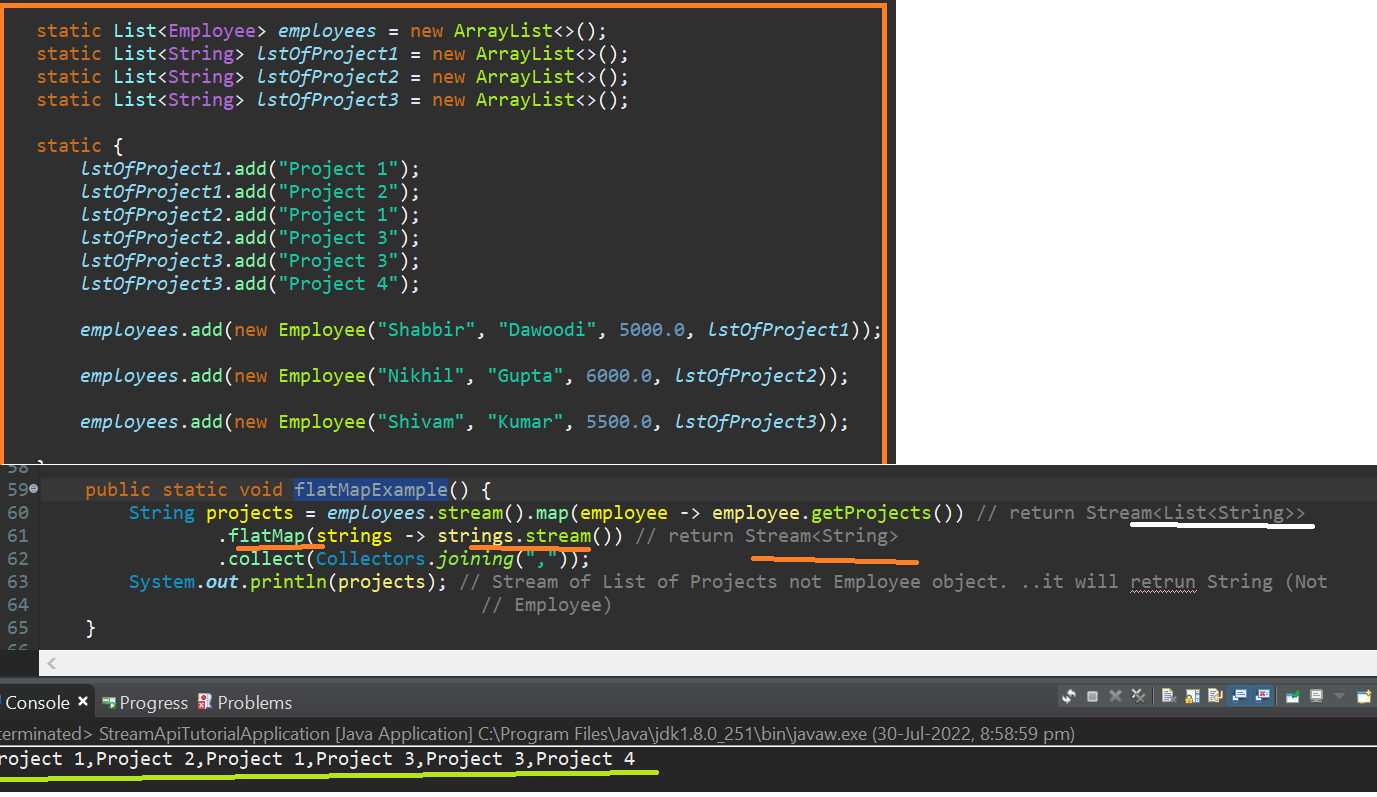


**4a. findFirst() and orElse()** 🡪No Employee have >7k salary ..so, it is null

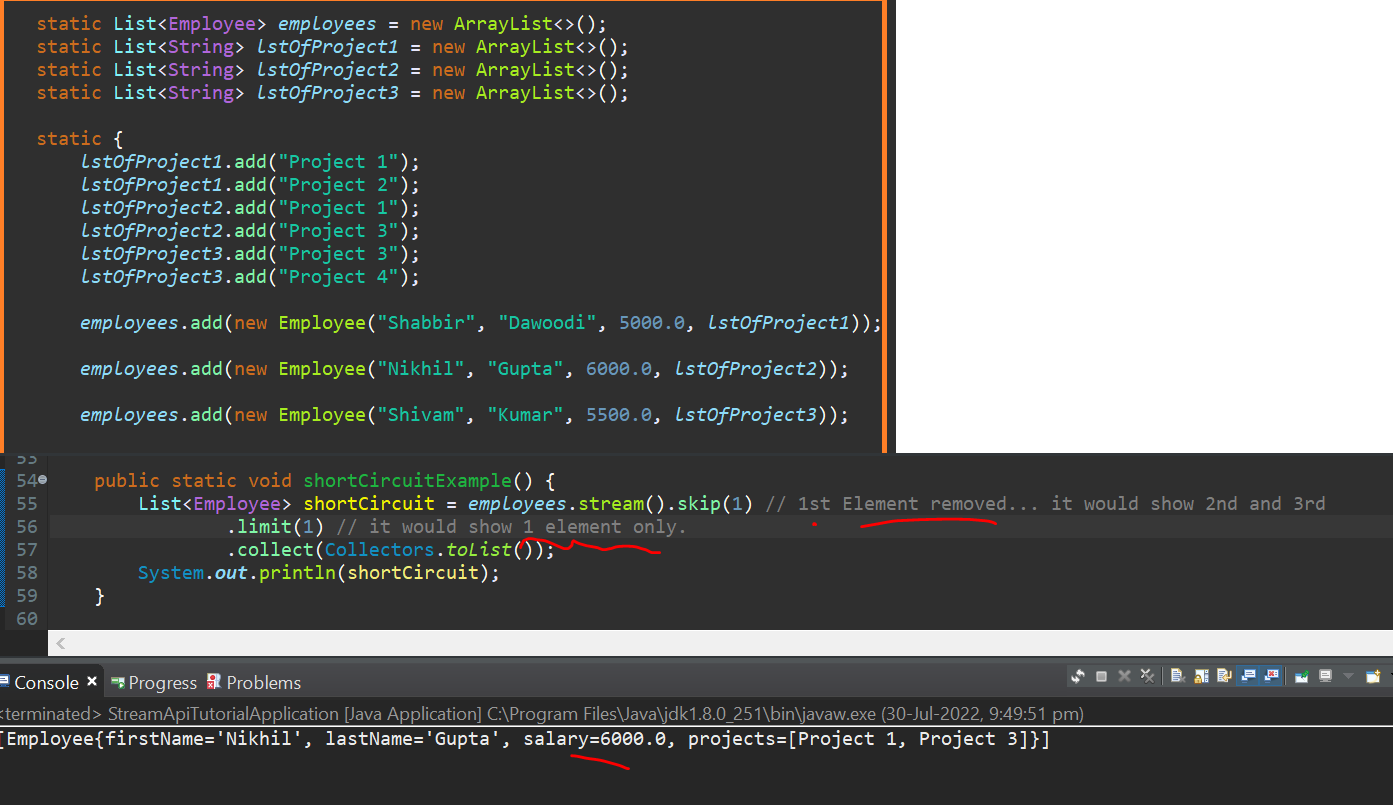


1. **map and flatMap** -- intermediate operation

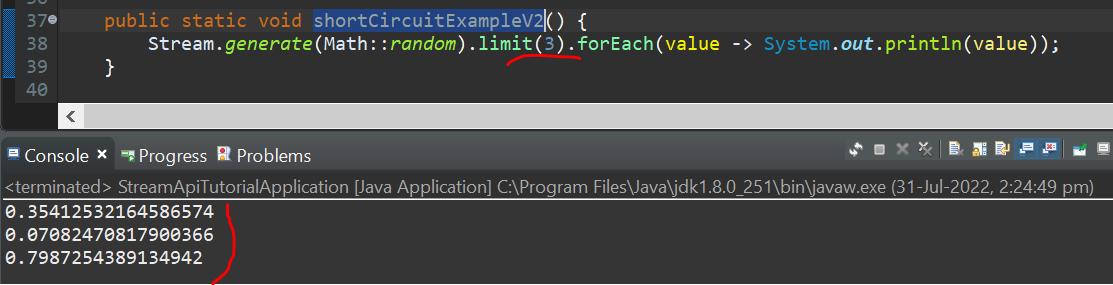
(Employee🡪list<String> )



1. **skip(1) and limit(1)** – short circuit operation



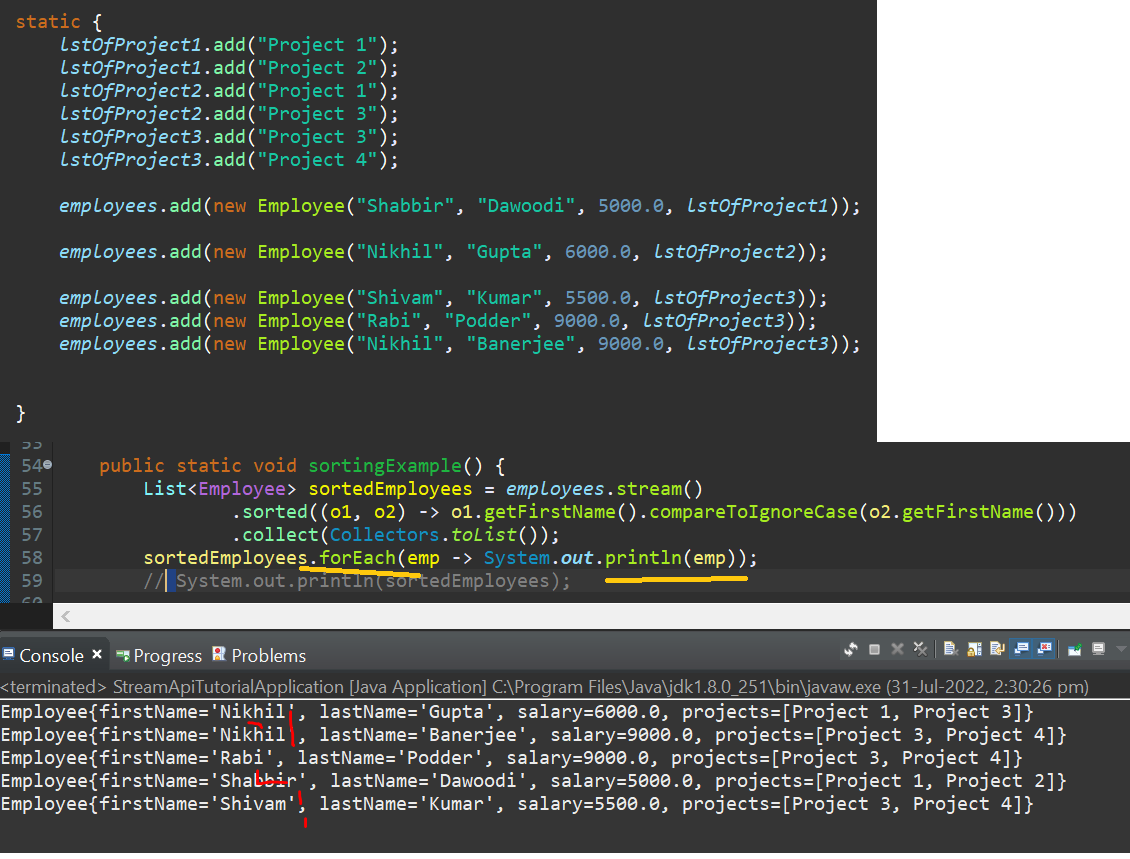
6A.**limit() –** the no. of random number will be limited ( 3)!!



1. Sorting using **compareTo()** (Based on Single Property i.e. “First Name” )

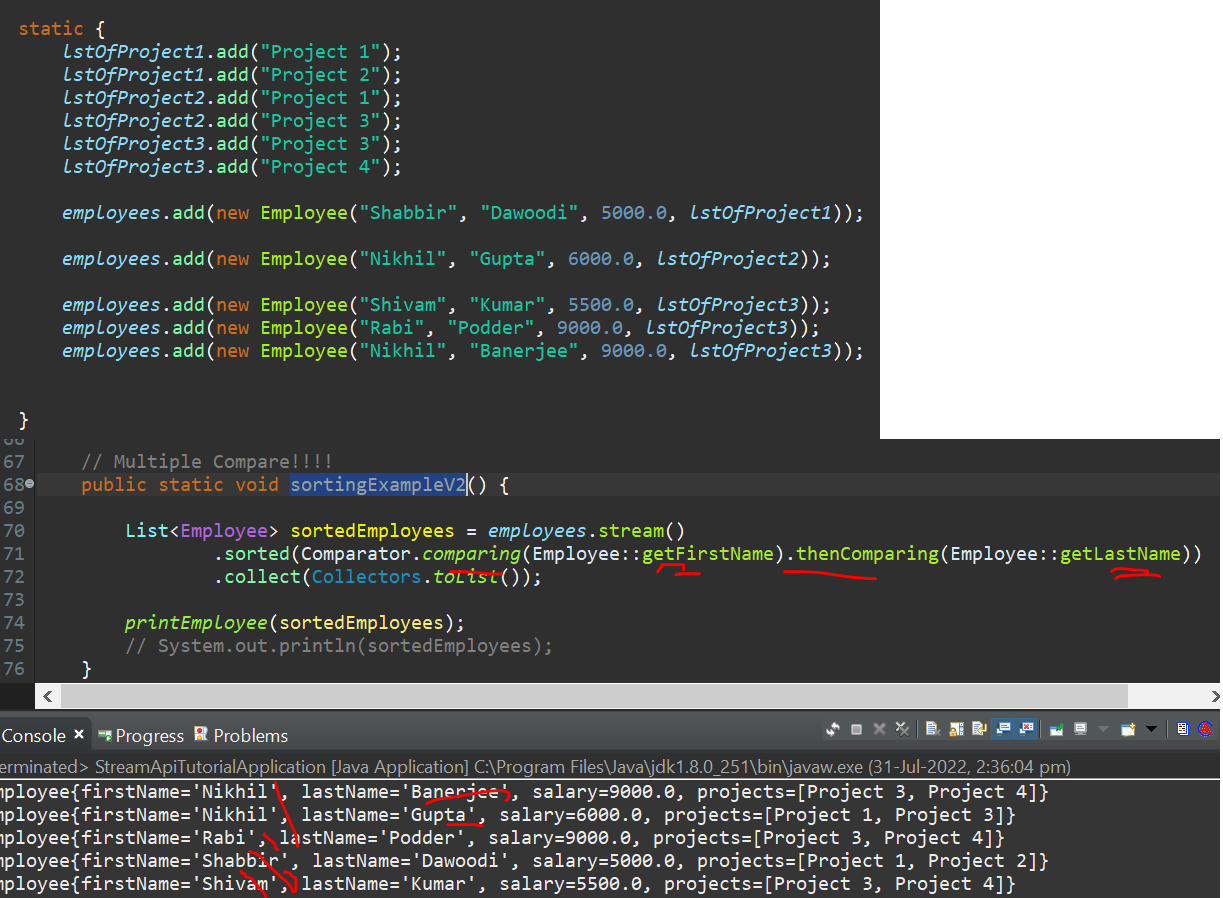
(Nikhil Gupta and Nikhil Banerjee are not sorted properly)

(as Last name sorting has not considered)



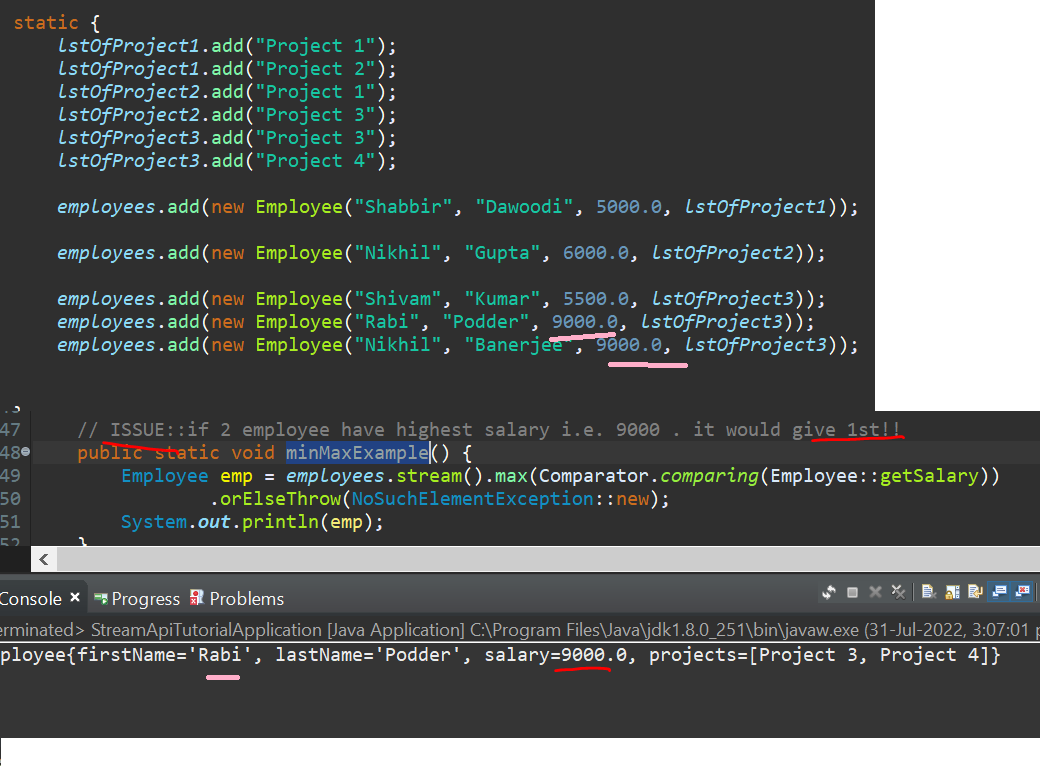
7A. Sorting using First name and second name and more

(Comparator.**comparing().thenComparing()** )

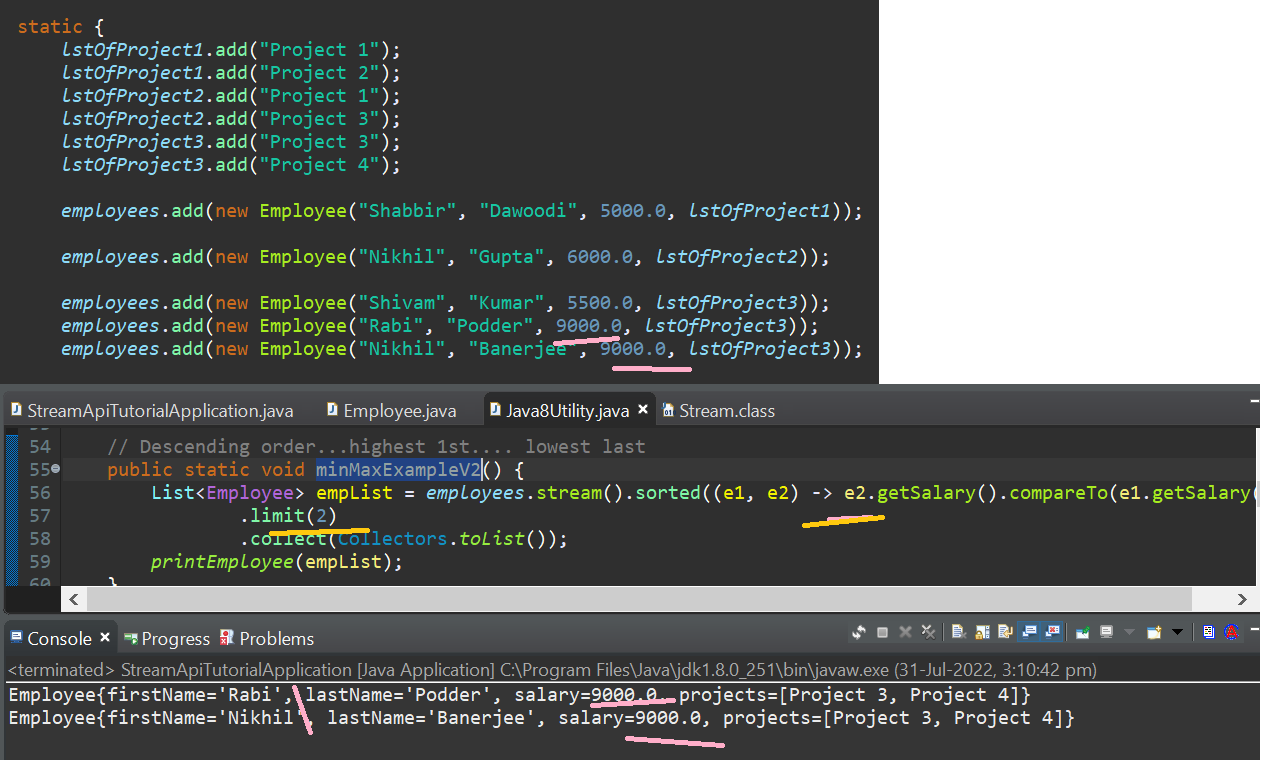


1. **Max()** 🡪 return single object …Highest salary only for 1 person

(if 2 person is getting 9000 …it will give 1st person in the list only )



8A. Fix for #8 using **sorted() + limit(2)**



1. Aggregate function like summation of all salary using “Sum” ,”max” ,”min”

