**Week 9 Assignment - ASSIGNMENT: CAR, PARKING PERMIT, PARKING TRANSACTION, PARKING OFFICE CLASSES**

For

ICT-4305 Object-Oriented Method & Program I

Luther Chikumba

University of Denver University College

November 11, 2022

Instructor: Dr. Sherri Maciosek

Table of Contents

[Write up Summary 3](#_Toc119068427)

[Class diagrams 5](#_Toc119068428)

[Screenshots 7](#_Toc119068429)

[Reference: 9](#_Toc119068430)

# Write up Summary

For the week's assignment, I had two things that I did find difficult. I would have liked to spend time adjusting my code and ensuring it worked as expected, but unfortunately, I did not have enough time. Most of my time was spent working on the group assignment; little did I know that my classmates were not behaving as they should be. Now you might be wondering why I did not do it. The reason was because of how much time I had left; I do not think I would have been able to meet the deadline to submit a functioning code if I had refactored my code there and then. I recently discovered that some parts of my code are disconnected; they do not function together as one system. I learned that when a customer registers and gets a new permit, the permit needs to be attached to a customer; you cannot have a permit without a customer. Also, I realized that a single customer could have multiple permits for each vehicle, but a permit cannot be registered to multiple vehicles. That had to be handled as well. Those are some things that I found challenging, given the time I had left. Now because of that disconnect, I could not complete one of the assignment requests, which was to Add a method to the Parking Office to return the collection of permit ids for a specific customer. Specific conditions had to be met before getting the correct value for that method. The customer had to be registered and have a vehicle. Also, customers can have one or more vehicles, translating into that a customer can have multiple permits if they have several vehicles. So the disconnect I mentioned earlier is the one that made it difficult to get the customer's permit ids because that required me to refactor my code.

From this week's assignment, I was primarily focused on the group assignment. I did not get enough time to seek help and get that fixed. What I wish I had known before was that disconnect in my classes. That knowledge would have helped me plan where and how I should have tackled the problem. It would not have been easy, but at least it would have given me a head start on what I needed to focus on.

About the implementation decisions and the reasoning behind adding equals () and hashcodes () was because they are essential. Those methods provide the capability for comparing objects. Object class is the parent class for all Java objects; hence all objects inherit the default implementation of these two methods. The equals method will return the true if both the objects are the same, and the hashcode() method returns the same hash value when called on two objects. Furthermore, if the objects are unequal, it returns different hash values, hence allowing comparing two different objects even if they have the same address or different addresses.

# Class diagrams

|  |
| --- |
| Money |
| cents: long |
| getDollars(): double toString(): String |

|  |
| --- |
| ParkingCharge |
| permitId: String lotId: String incurred: Instant\*\* amount: Money |
| toString(): String |

|  |
| --- |
| ParkingOffice |
| name: String address: String customers: List<Customer> cars: List<Car> lots: List<ParkingLot> charges: List<ParkingCharge> |
| register (String name,             String address,             String phone): Customer register (Customer c,              String license,              CarType t): Car  getCustomer(String name): Customer\*\*\* addCharge(ParkingCharge): Money  addCarToParkingLot():List<ParkingLot>  isCarRegistered(): boolean  getCustomerIds():Collection<String>  getPermitIds(): Collection<String>  customer (getPermitIds(Customer)): Collection<String> |

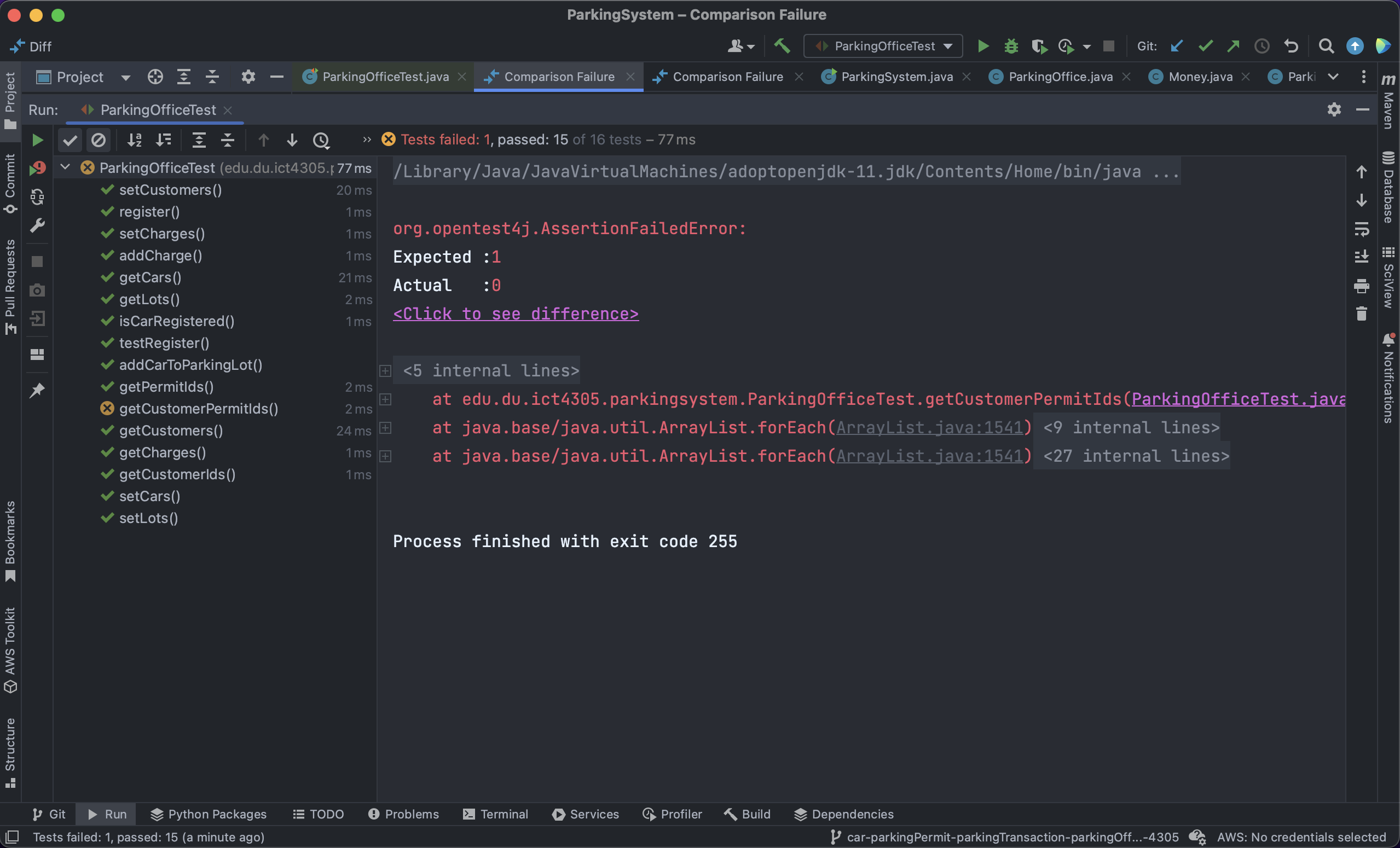
# Screenshots

Text

Description automatically generated

Text

Description automatically generated



# Reference:

“Equals() and Hashcode() in Java - Javatpoint.” n.d. Www.javatpoint.com. Accessed November 11, 2022. <https://www.javatpoint.com/equals-and-hashcode-in-java#:~:text=%E2%86%92%20%E2%86%90%20prev->

‌