Food Ordering System – Project Report

Nelson Dang Quang Pham (6317924)

DATA STRUCTURES AND OBJECT ORIENTED PROGRAMMING

Presented to: Yi Wang

Due Date: May 11, 2025

# Table of Contents

1. Project Description

2. Program Features

3. Screenshots & Output Examples

4. Challenges Faced

5. Learning Outcomes

# 1. Project Description

The Food Ordering System is a Java-based desktop application that allows users to place food orders either for pickup or delivery. The system supports different types of users: Customers, Admins, and Drivers. Each user type has specific responsibilities and permissions, such as placing orders, processing or canceling them, and managing deliveries.  
  
The project structure is object-oriented, with classes like User, Customer, Admin, Driver, Order, DeliveryOrder, PickupOrder, and Item. It uses file I/O to read menu data and log completed orders. The purpose of this system is to simulate the backend functionality of a real-world food ordering service.

# 2. Program Features

- Customer Functionalities:  
 - View the menu.  
 - Search for specific items.  
 - Place orders (pickup or delivery).  
 - Track the status of an order.

- Admin Functionalities:  
 - Process incoming orders.  
 - Cancel orders.  
 - View all completed and open orders.

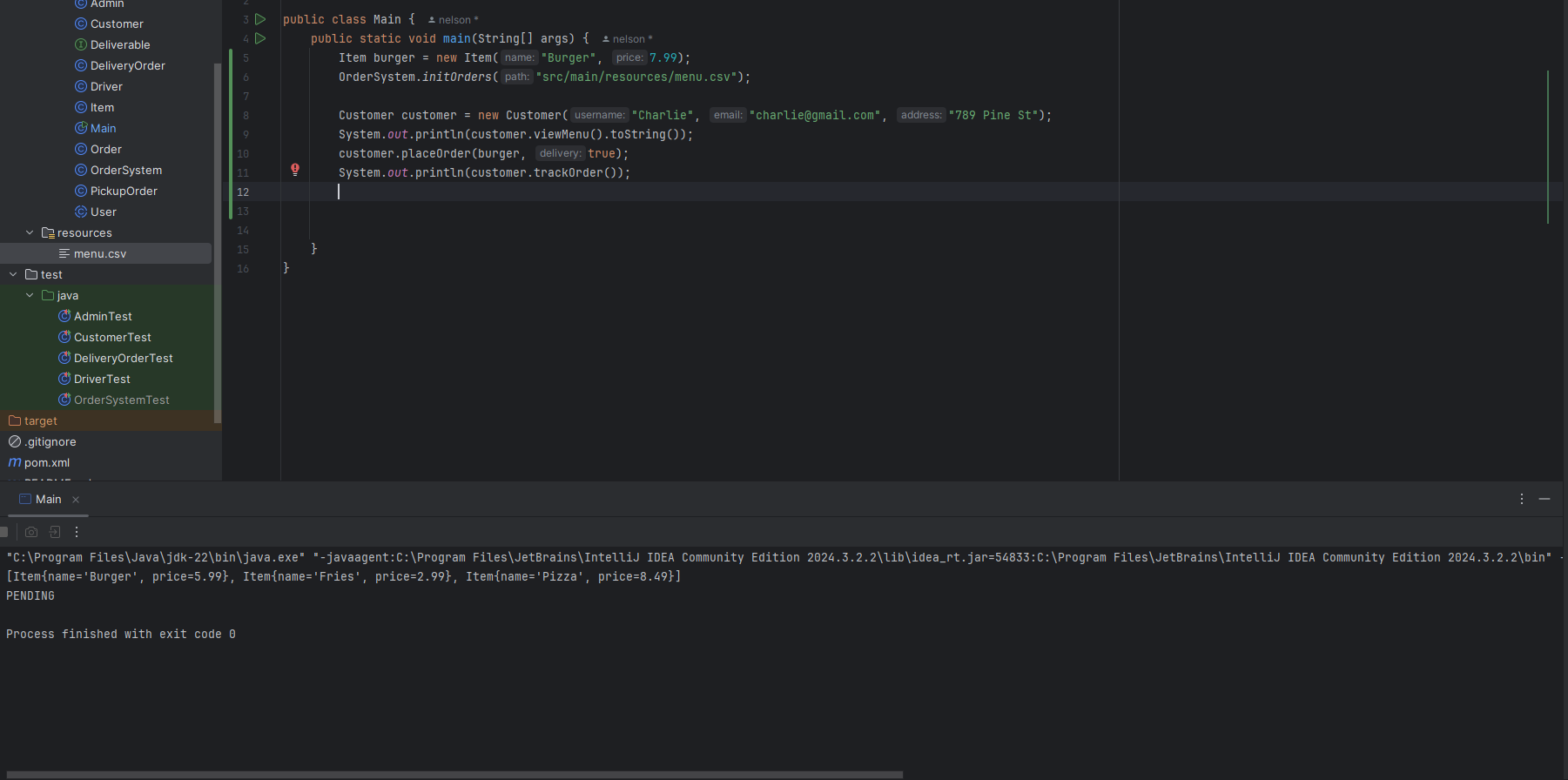
- Driver Functionalities:  
 - View available delivery orders.  
 - Accept and update order status.

- Order Types:  
 - PickupOrder: Includes basic order details.  
 - DeliveryOrder: Includes distance-based fee, tip calculation, and estimated time.

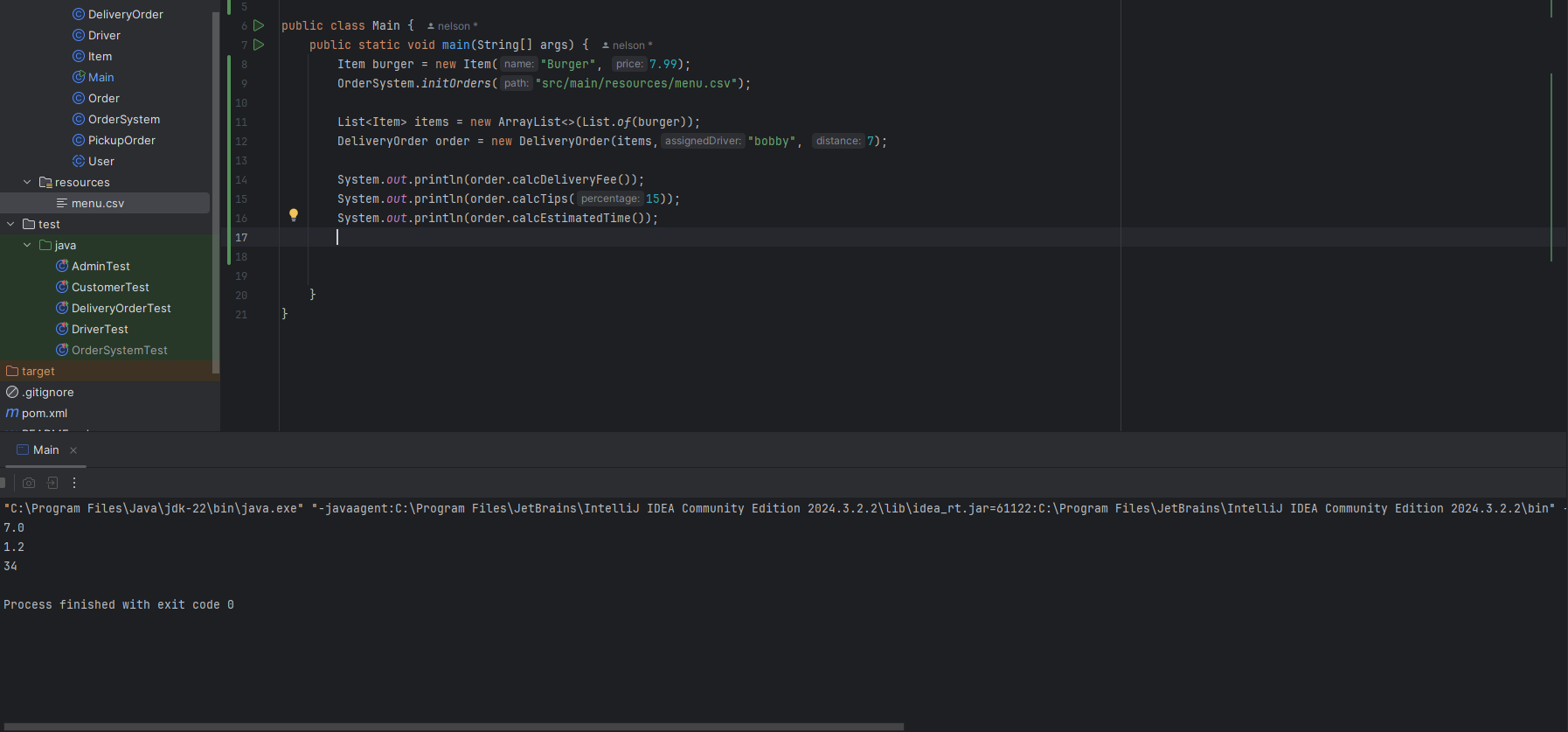
- Persistence:  
 - Read menu items from a file.  
 - Write completed order summaries to a file.

# 3. Screenshots & Output Examples

Customer placing an order:



Calculating tips and delivery fees:



# 4. Challenges Faced

- Object Equality Issues: Comparing two order objects in tests was problematic due to differing order numbers (auto-incremented). This was addressed by focusing on business logic validation.  
- Testing File I/O: Required careful file management and cleanup.  
- Project Design: It was difficult to come up with the design of the project and put it into reality.

# 5. Learning Outcomes

Through building this project, I learned:  
- How to structure an object-oriented Java application.  
- How to work with file I/O in Java.  
- How to use inheritance and interfaces effectively.  
- How to write unit tests using JUnit.  
- How to manage and push a project using Git and GitHub.