# Relational Databases with MySQL Week 6 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 to the repository. Additionally, push an .sql file with all your queries and your Java project code to the same 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:**

This week you will be working together as a **team** to create a full CRUD application.

Your console CRUD application will need to use a database to store all the application data.

As a team, decide what you want your project to do. Get instructor approval early in the week before beginning development.

You need to have at least 3 entities.

Users should be able to interact via the console (i.e. Scanner(System.in)))

Everyone must individually submit their own assignment documents and the full code for the entire project. Inside the code, use comments to make it clear which code you specifically wrote.

Although git provides collaboration functionality, you are not required to use it to collaborate back and forth with your teammates. You can use any method you decide on as a team.

Everyone will be graded on their individual contributions.

**Project Name:**

Lego Store

**Project Team Members:**

Charlee Thao, Adam Fite, & Andrew Cham

**My Contribution to the Project:**

Database:

When creating the Lego Store Database, we each created a table. I created the sales table.

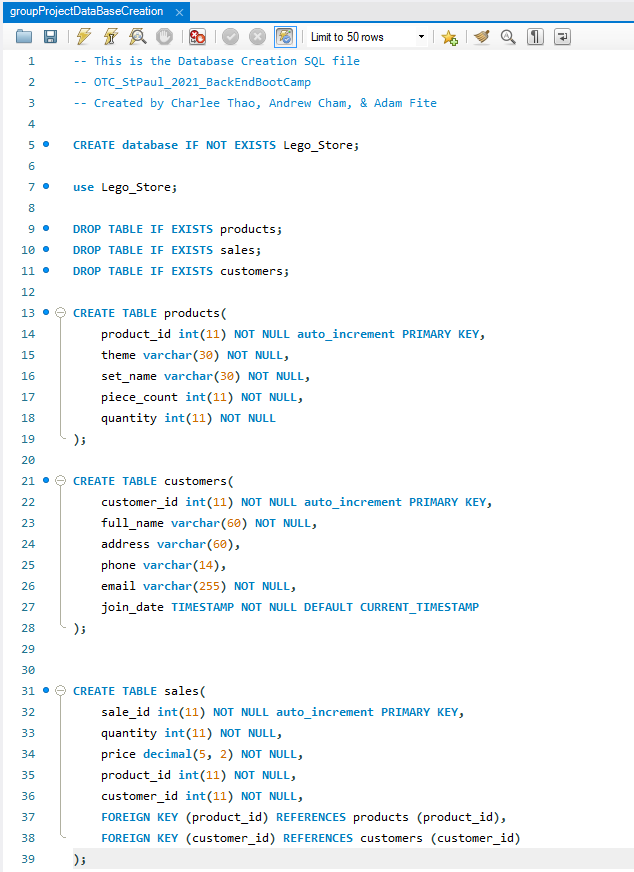
Java Application, Dao, and Entity:

I wrote the code for the CustomersMenu, CustomersDao, and Customers.java.

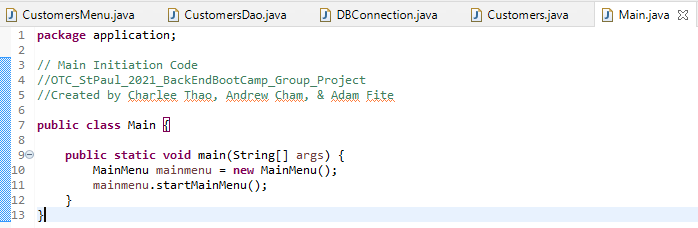
We all created the DBConnection, Main, and MainMenu.

**Screenshots of Code:**

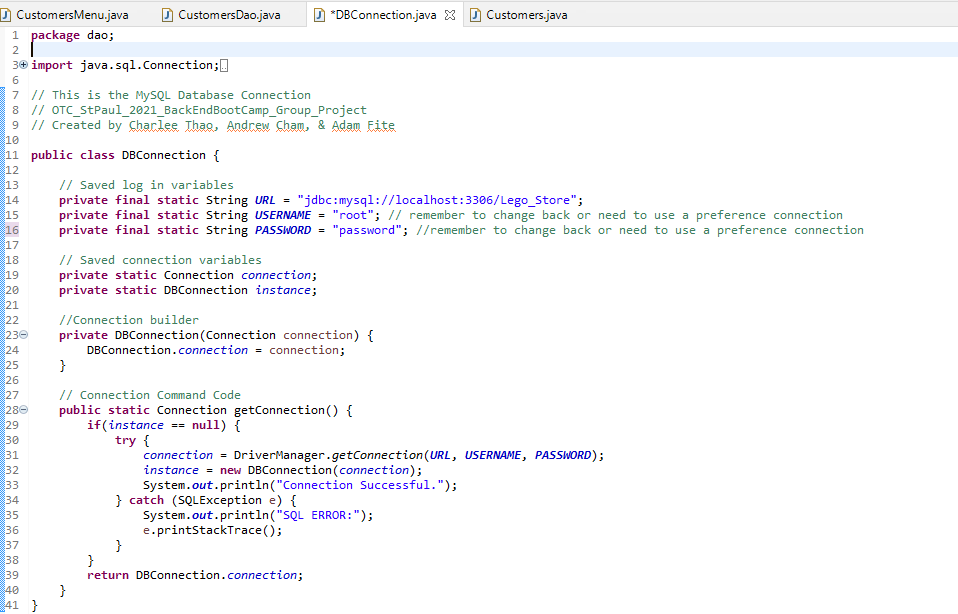
MySQL Database:



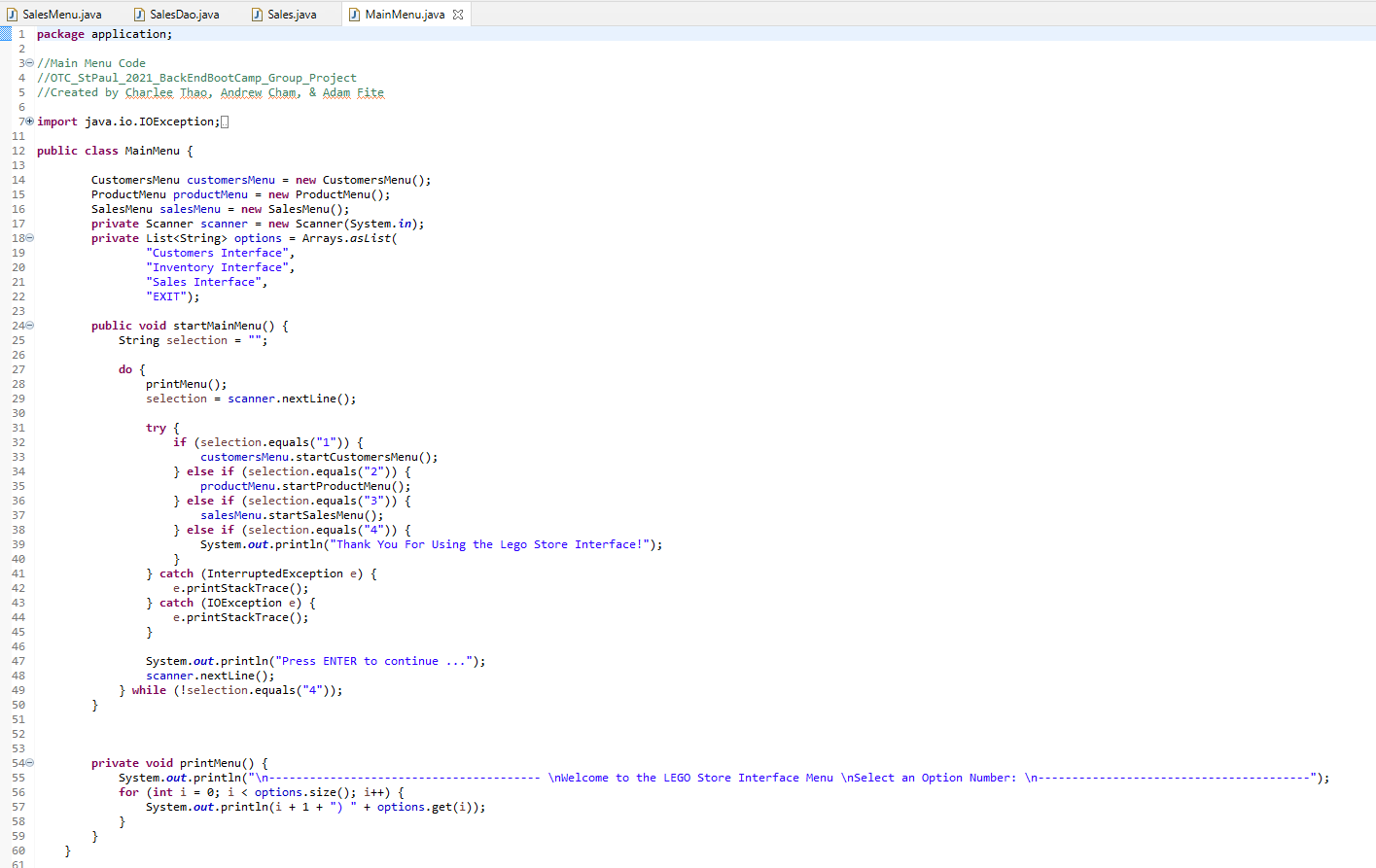
Main Application Code:



DBConnection Code:

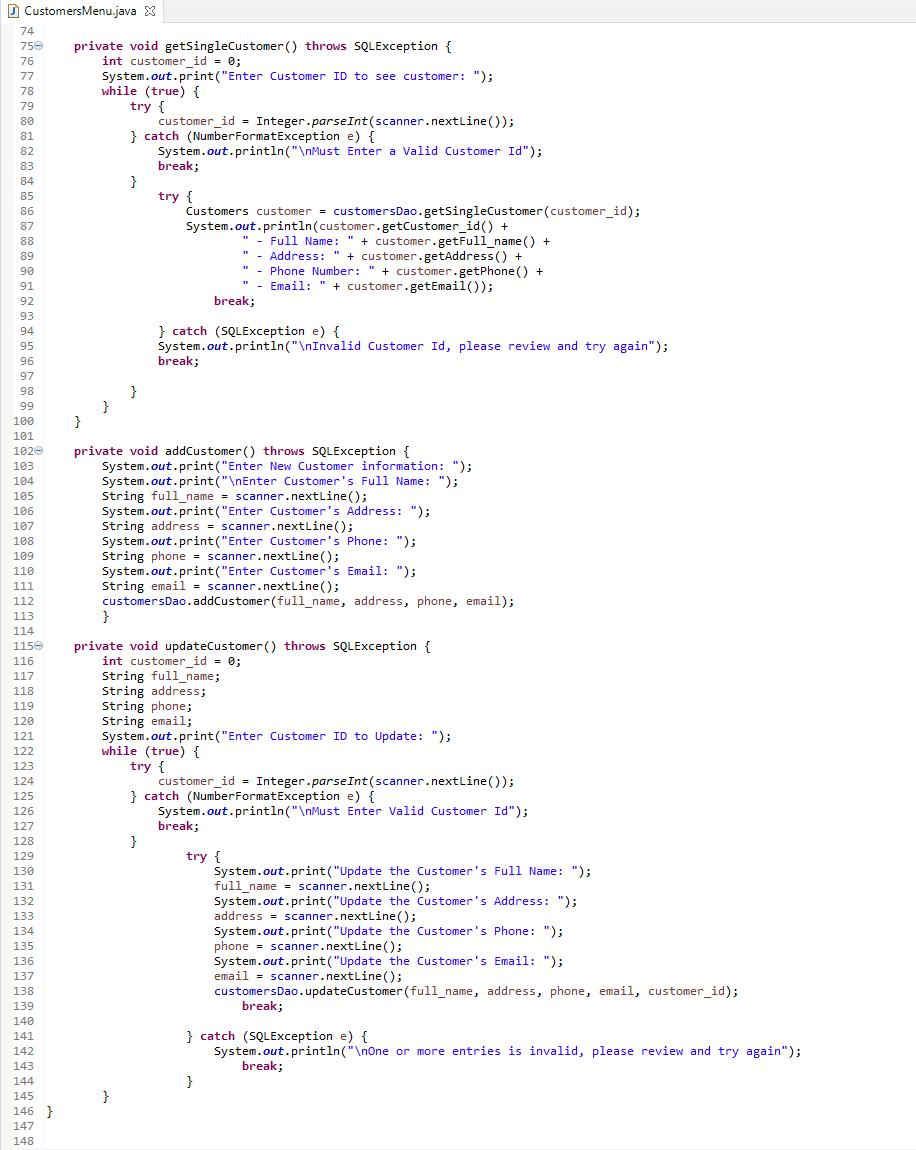
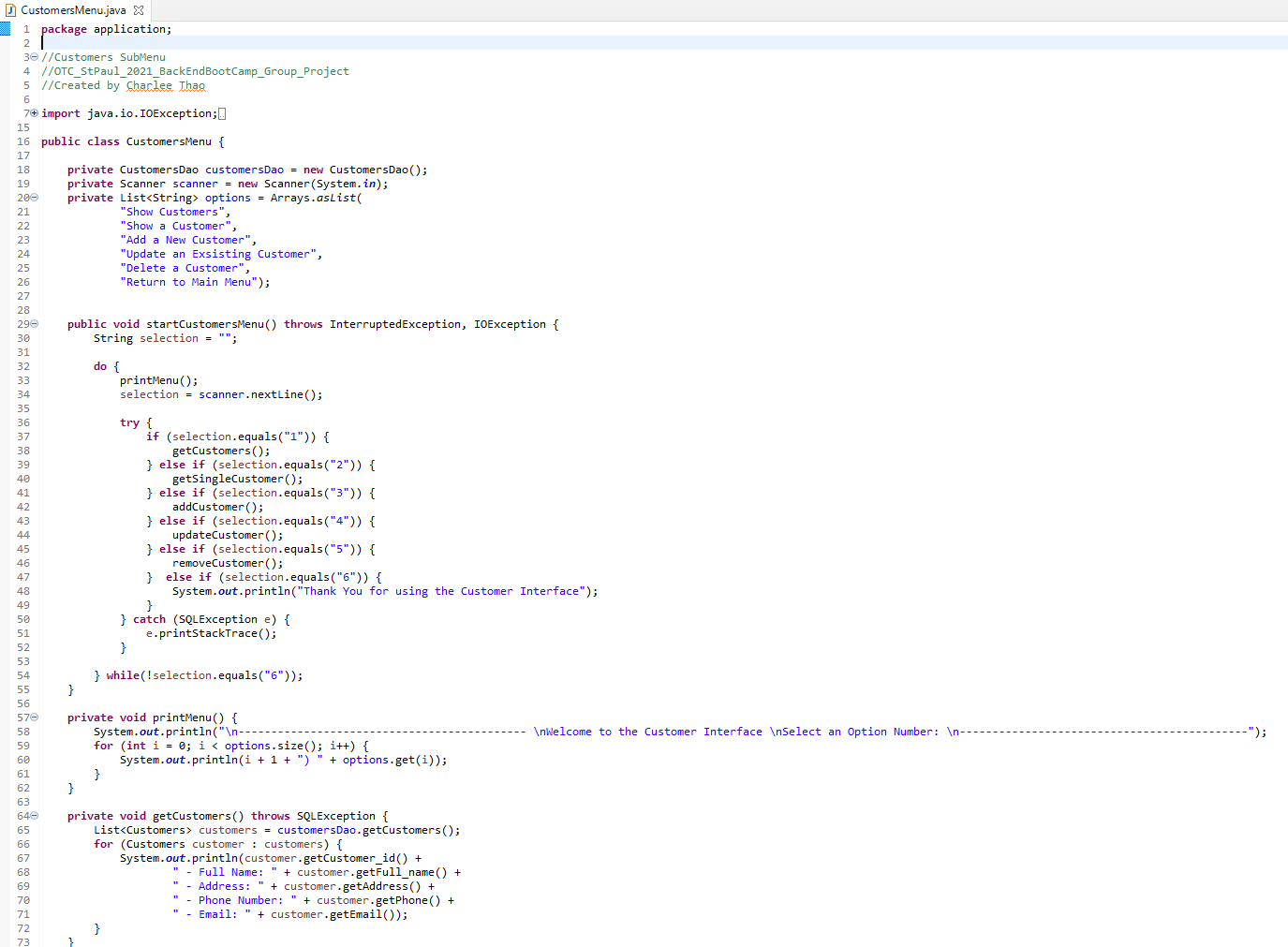


Lego Store Main Menu Code:

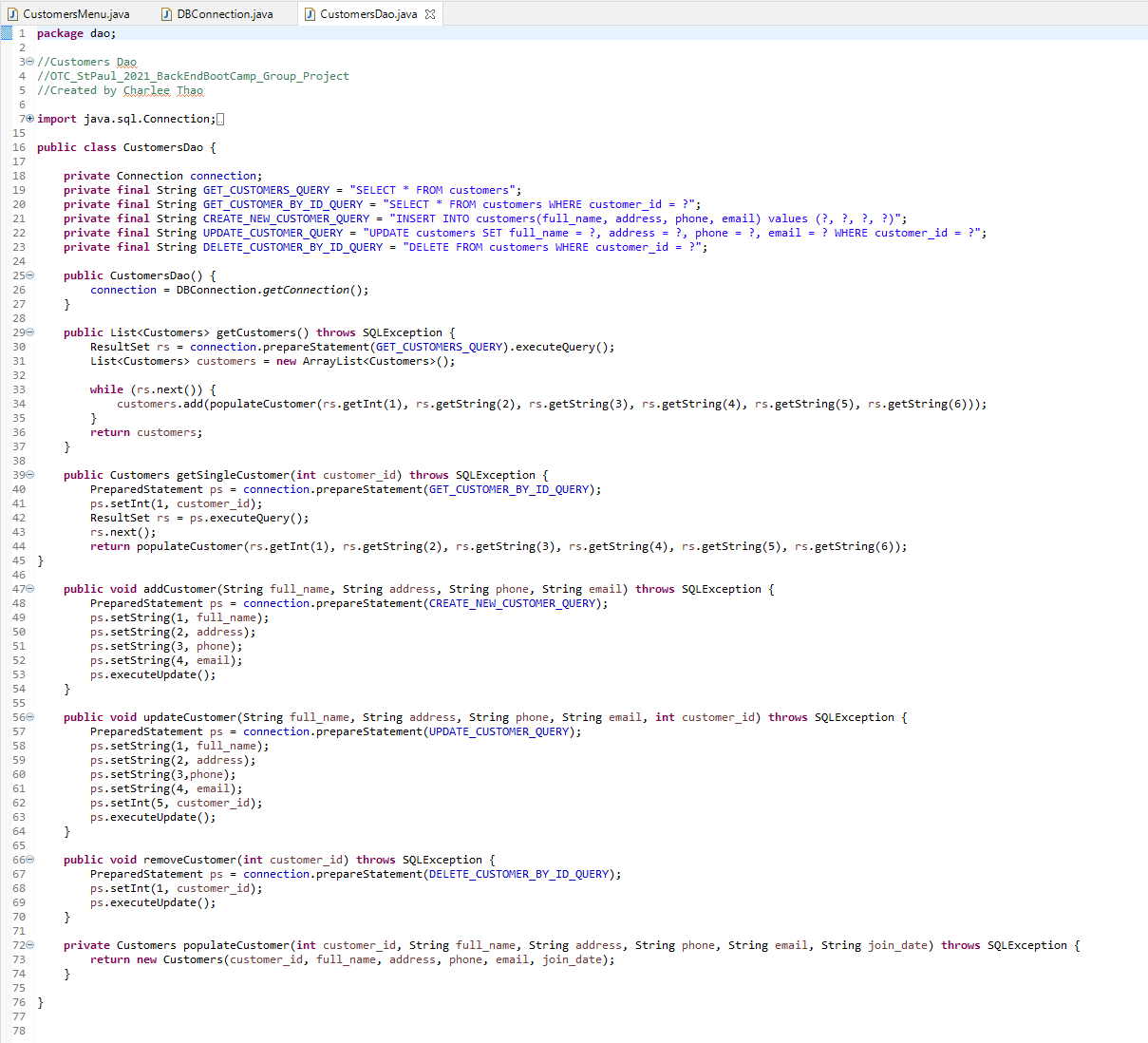


Customers Codes:

*Menu*



*Dao*

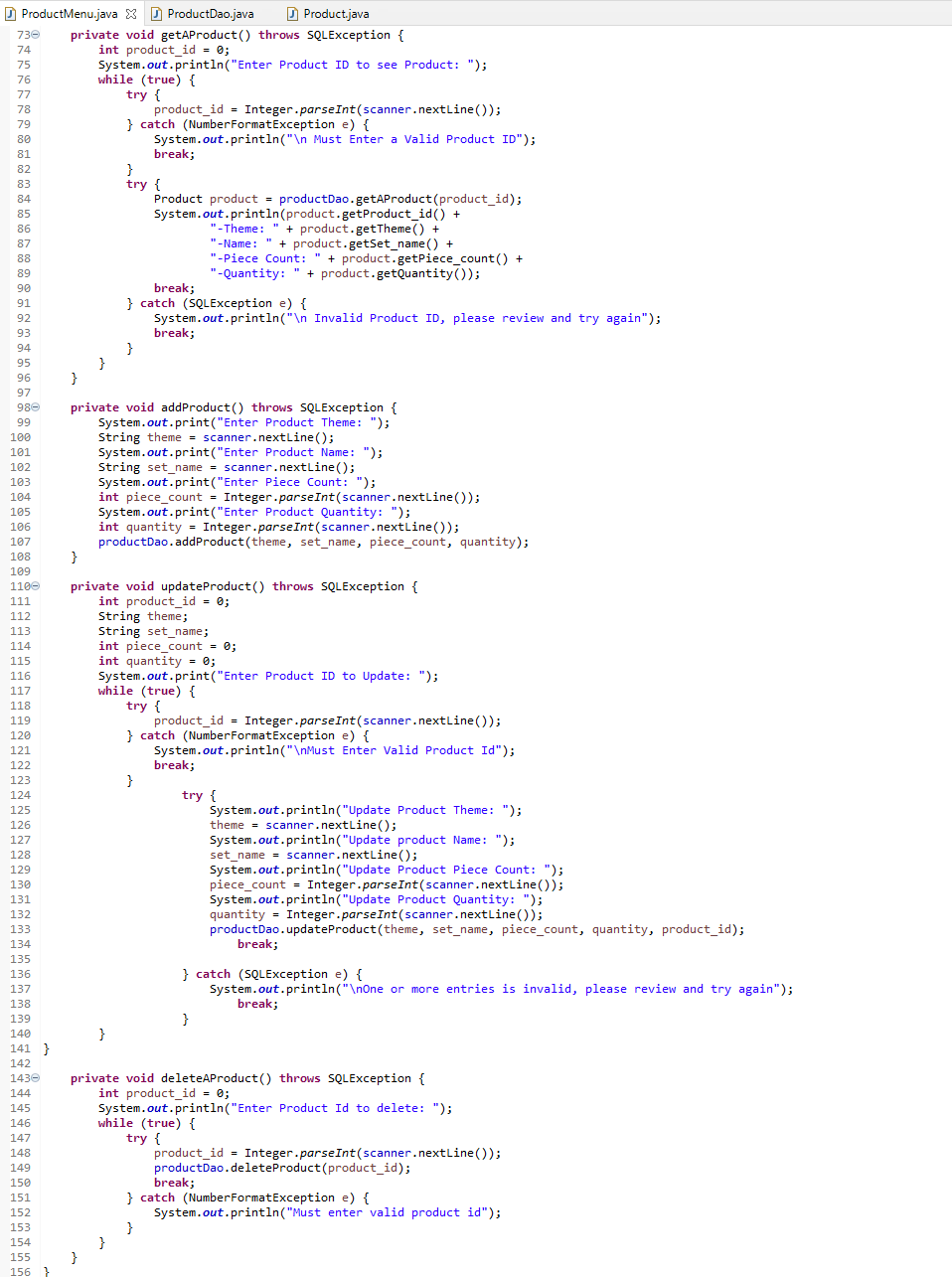
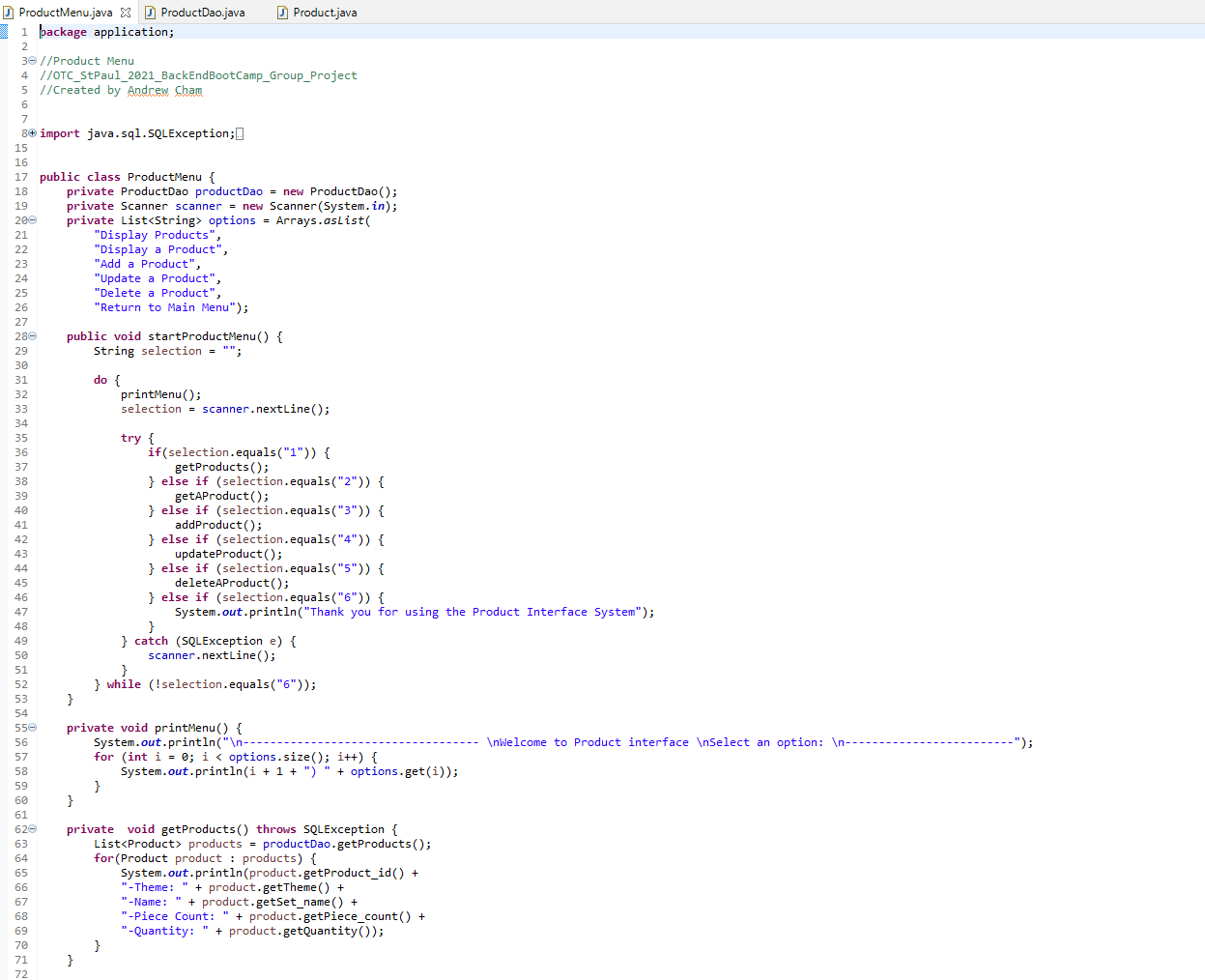


*Entity*

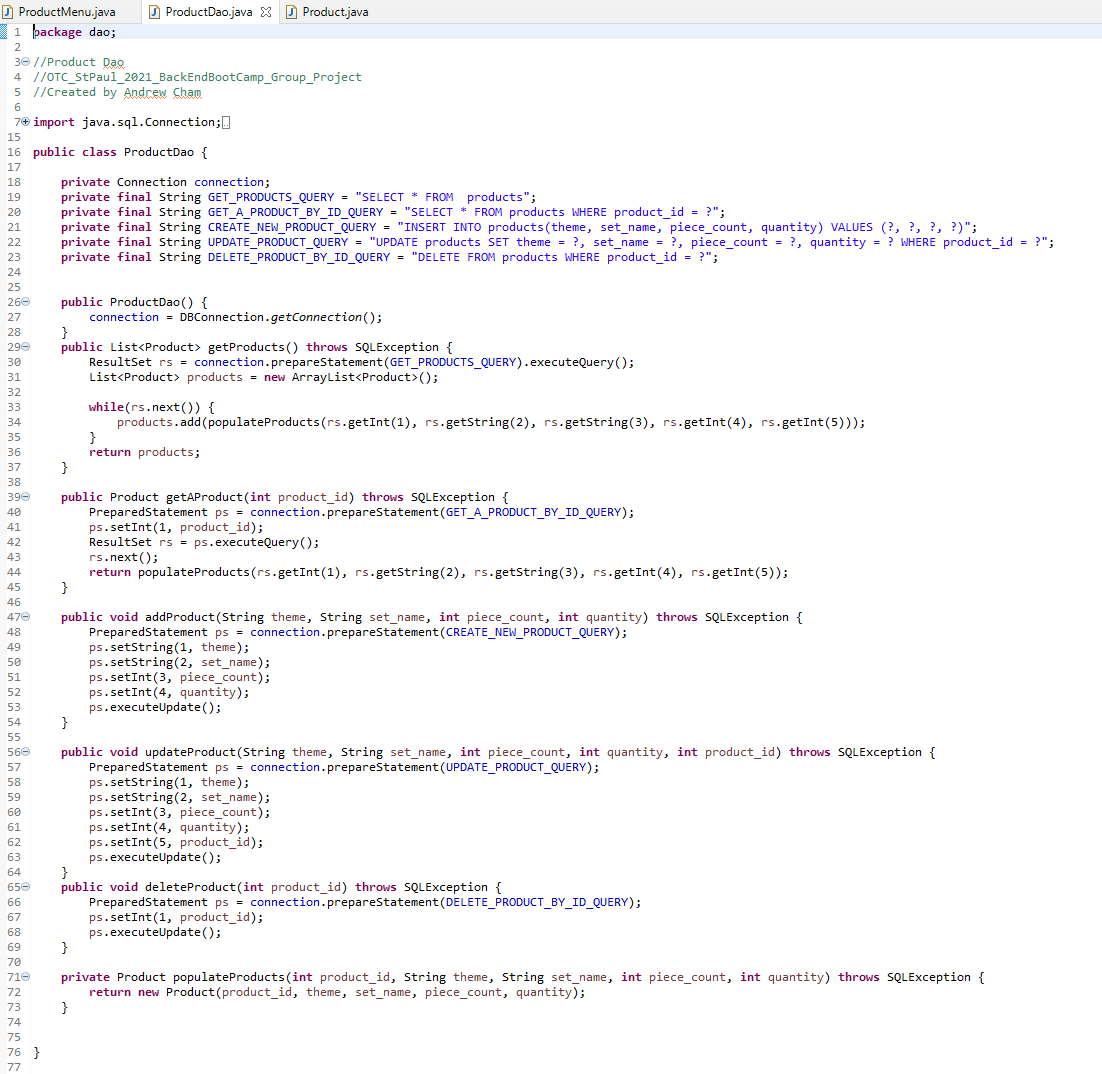


Products Codes:

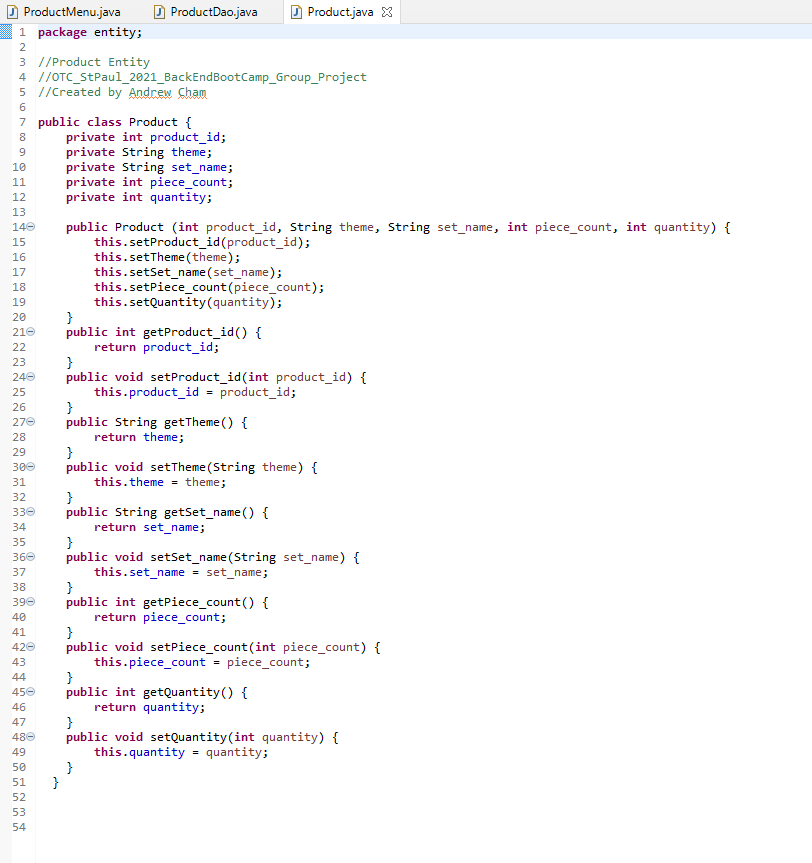
*Menu*



*Dao*

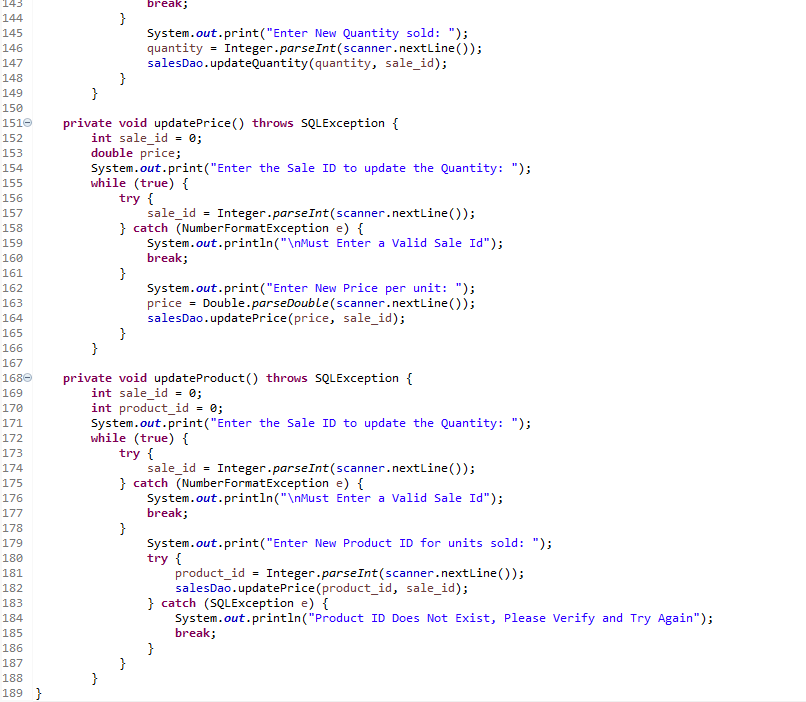
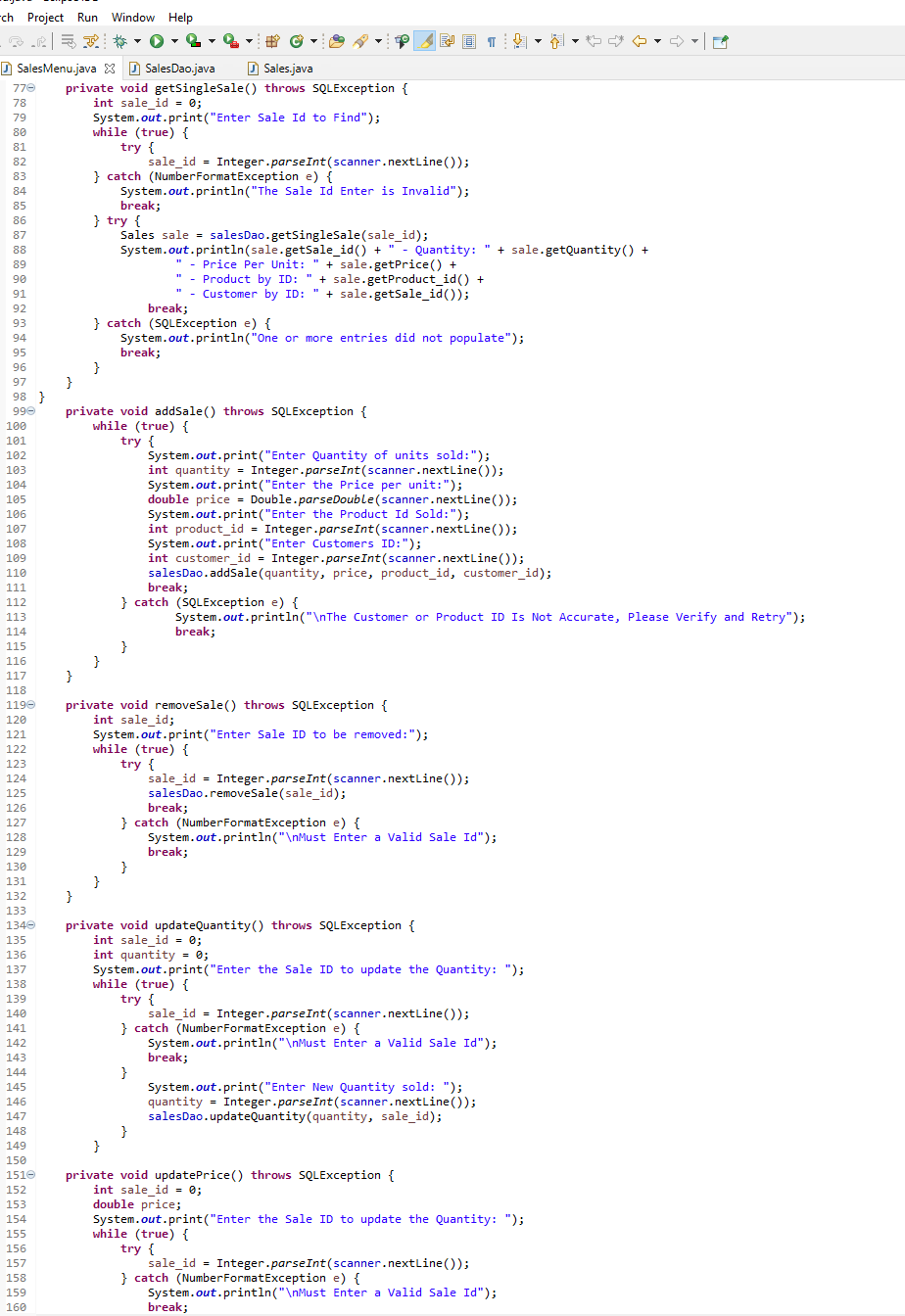
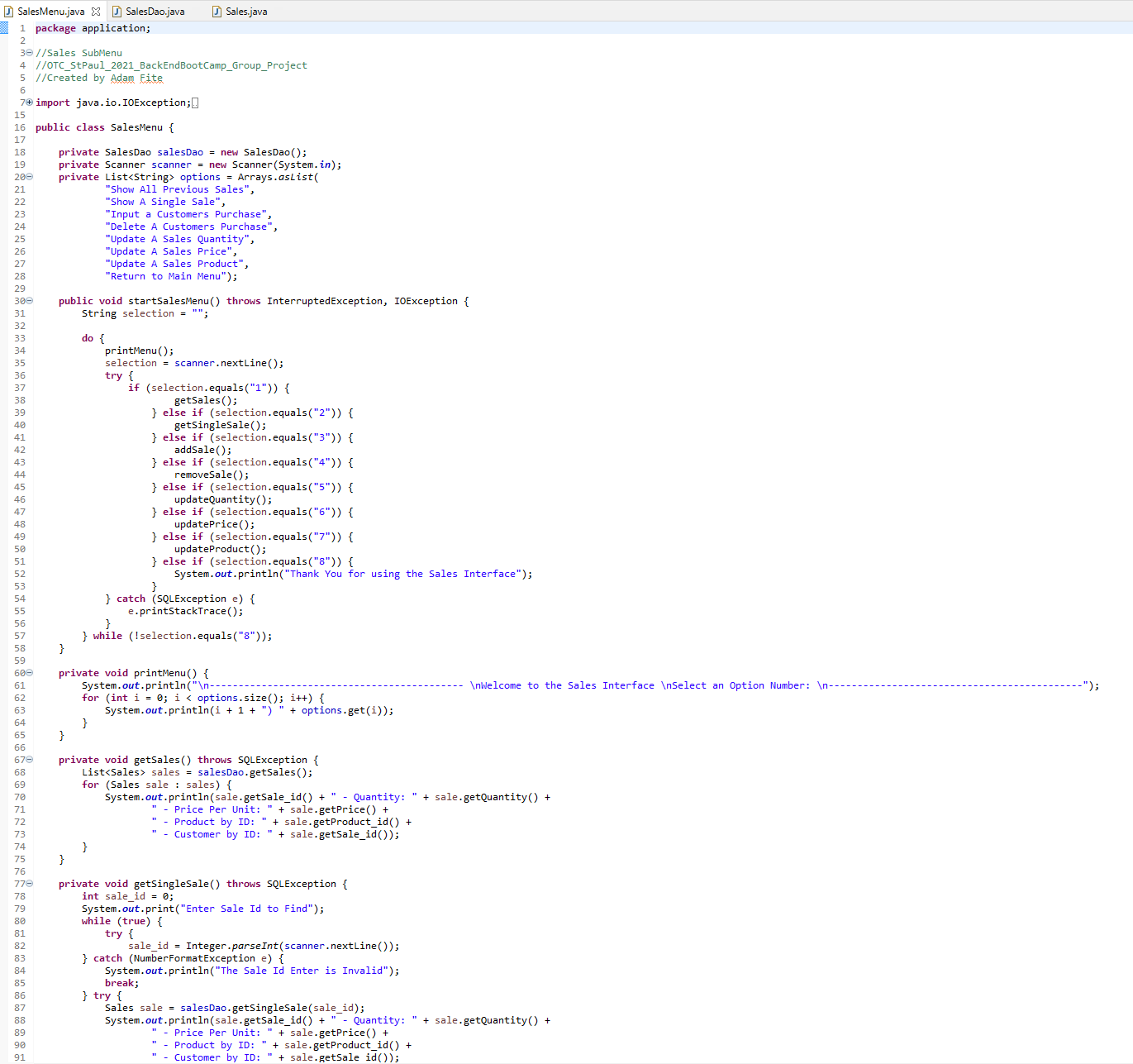


*Entity*



Sales Codes:

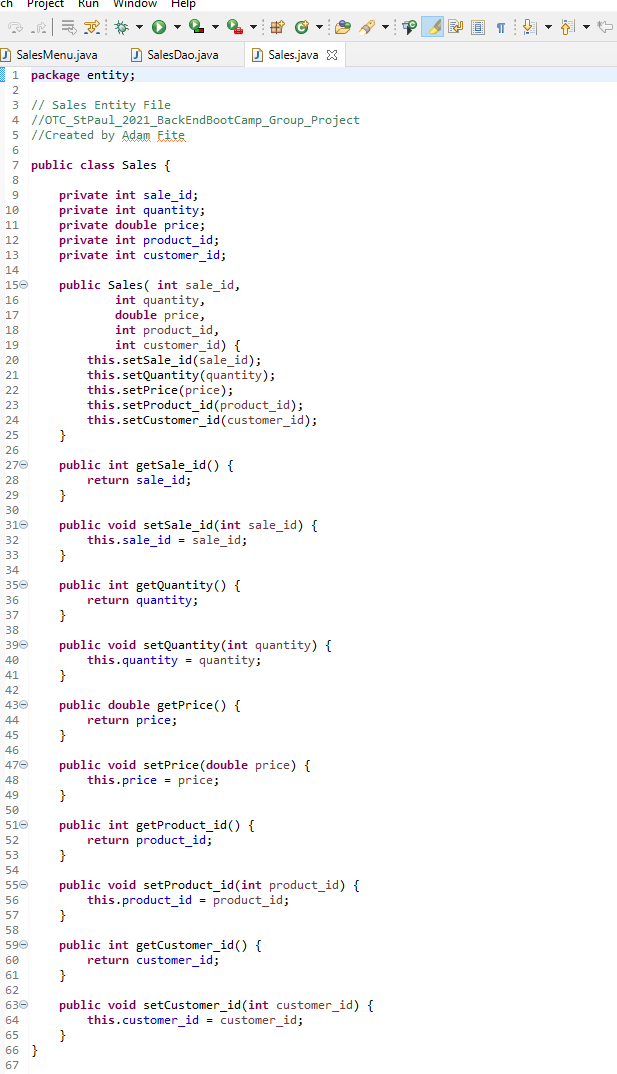
*Menu*



*Dao*

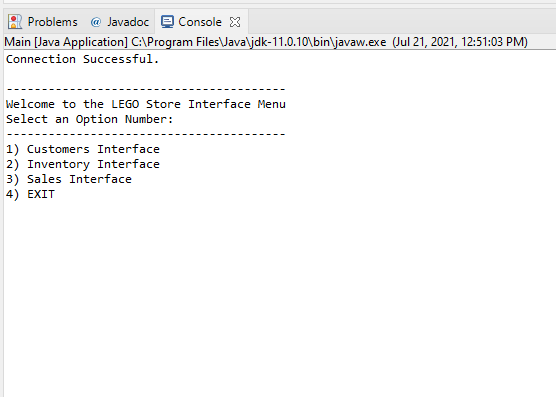


*Entity*



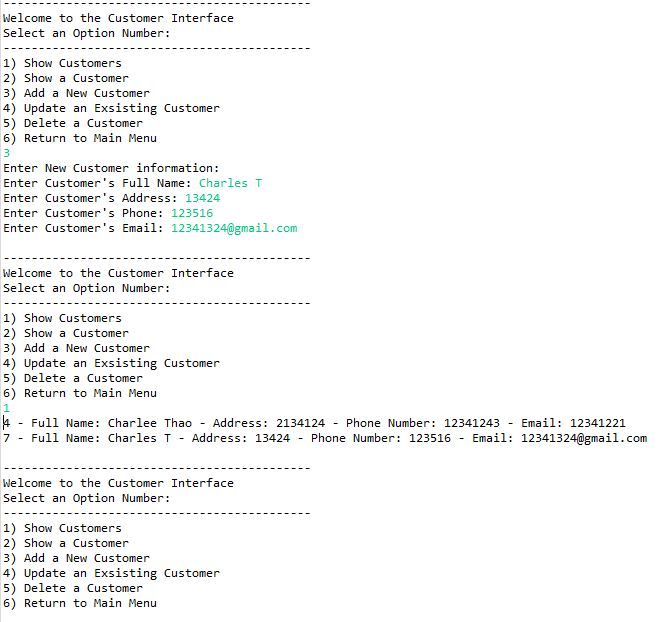
**Screenshots of Running Application:**

Main Menu:

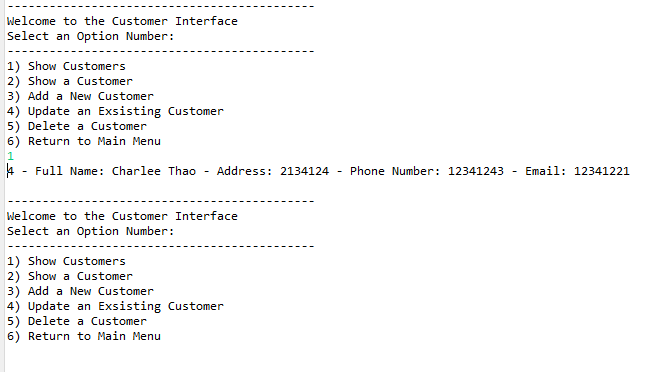


Customers CRUD:

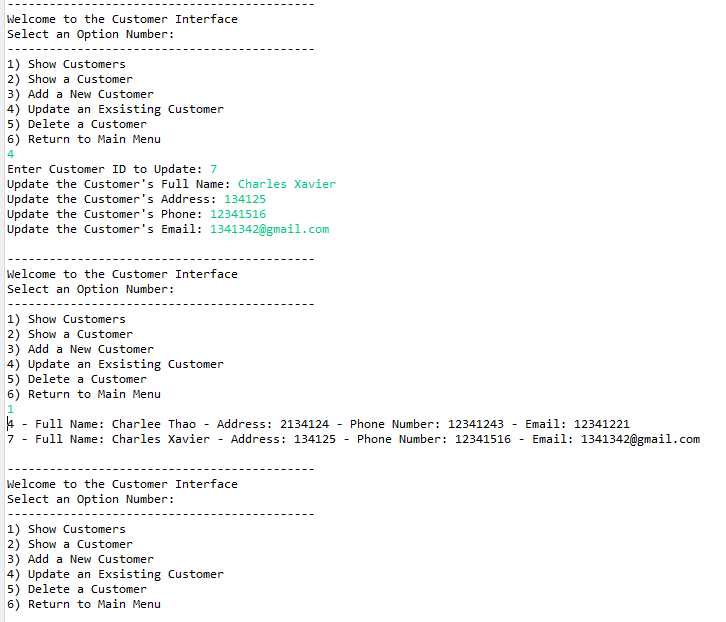
*Create*



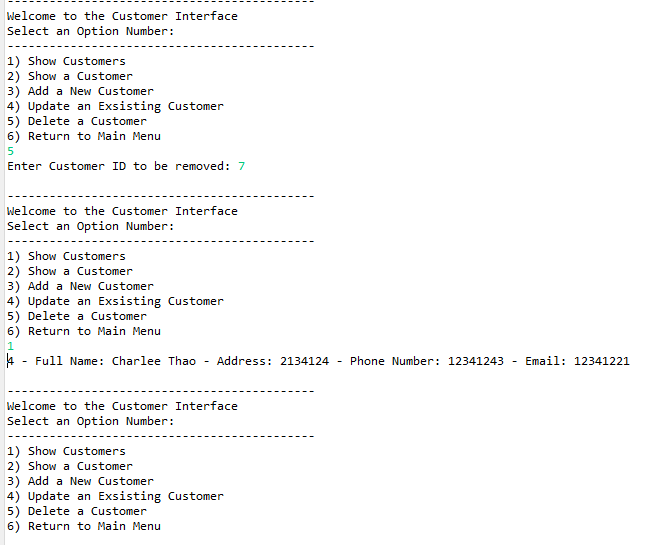
*Read*



*Update*

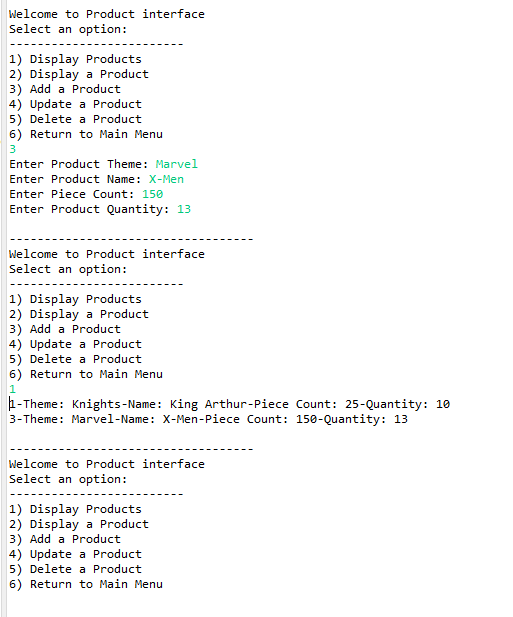


*Delete*

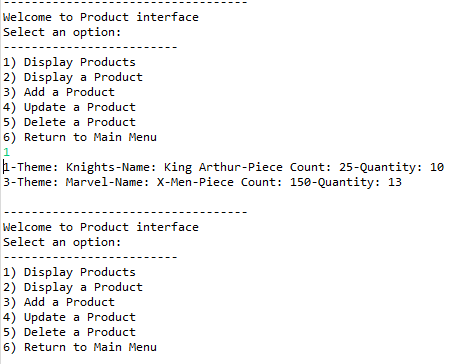


Products CRUD:

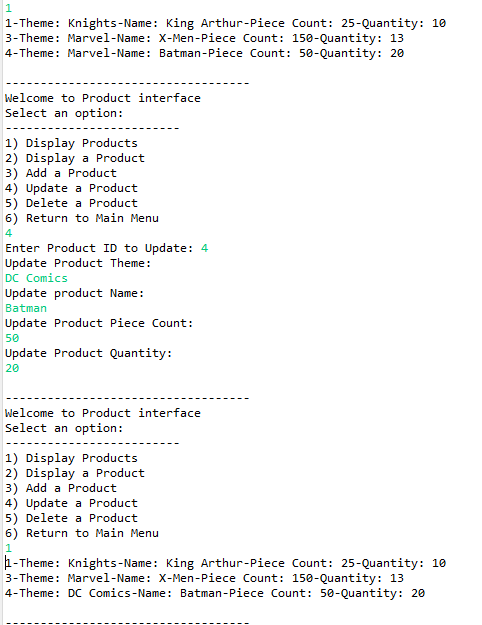
*Create*



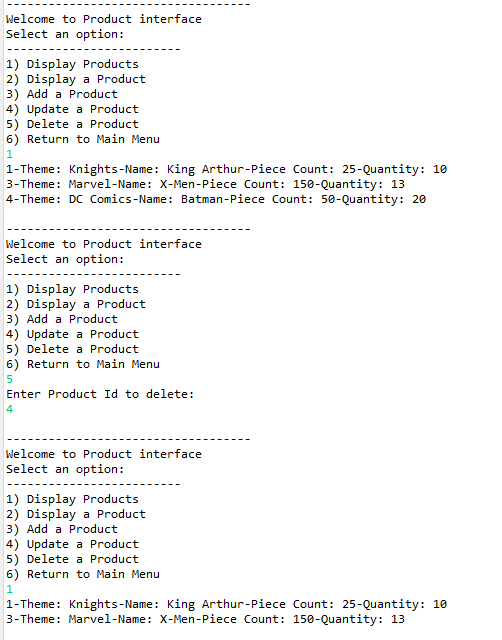
*Read*



*Update*

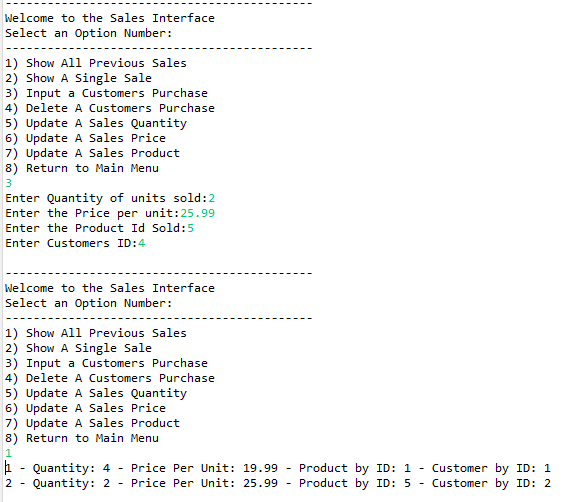


*Delete*

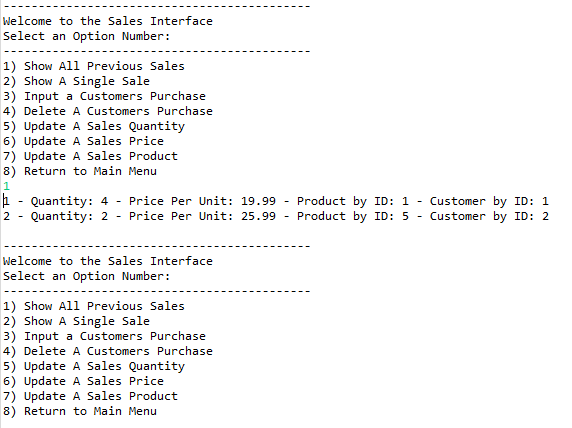


Sales CRUD:

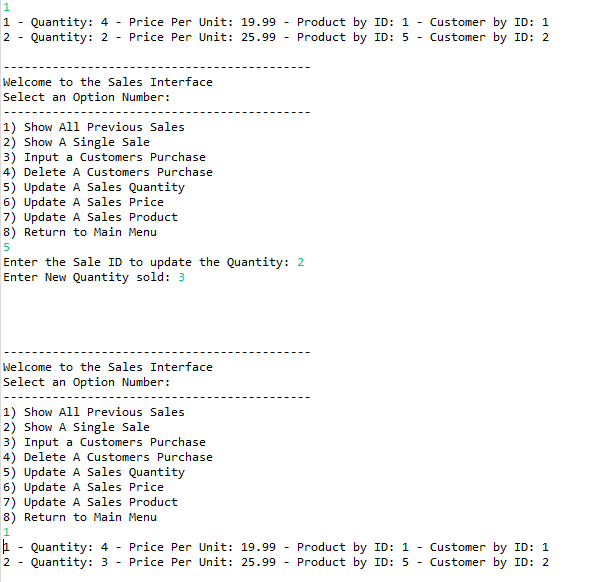
*Create*



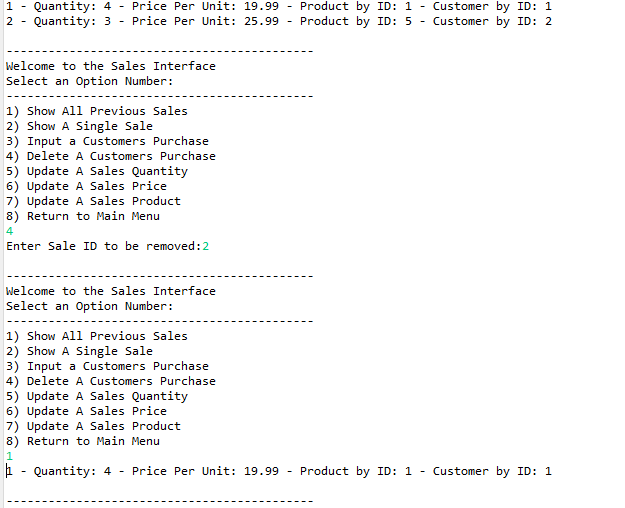
*Read*



*Update*



*Delete*



**URL to GitHub Repository:**

[**https://github.com/DippedNugget/OTC\_StPaul\_2021\_BackEndBootCamp\_GroupProject**](https://github.com/DippedNugget/OTC_StPaul_2021_BackEndBootCamp_GroupProject)