Java Coding Challenge

Order Management System

Introduction

As part of the coding challenge, **I was mapped to develop an Order Management System** in Java.

This system allows users and admins to manage products, place and cancel orders, view order history, and interact with a backend SQL database using JDBC. The application was designed using Object-Oriented Programming principles and includes modular and maintainable code architecture.

During the implementation:

- I created the required SQL schema to store user, product, and order information.
- Developed entity classes (like Product) to represent data models.
- Implemented custom exceptions such as UserNotFoundException and OrderNotFoundException to handle specific edge cases.
- Designed and implemented a custom interface named IOrderManagementRepository, which defines core operations like creating orders, canceling orders, and viewing order history.

Functionalities Implemented

The Order Management System is designed to provide different functionalities for Users and Admins, ensuring a seamless experience for both roles.

User Functionalities

Once a user logs into the system, and can do these fucntions :

- **Create Product Order**: Allows users to place an order by selecting a product and specifying quantity and payment mode.
- **Cancel Order**: Enables users to cancel an existing order. On successful cancellation, stock is restored and payment is refunded.
- **View Order History**: Displays the user’s past orders, including order ID, product details, and status.
- **Update Profile**: Lets users update personal details like address or password.
- **Delete Account**: Permanently removes the user and their associated orders from the database.
- **Logout**: Safely logs the user out of the session.

Admin Functionalities

Admins have advanced controls to manage users, products, and orders.

- **View All Orders**: Displays the entire order database across all users.
- **View All Electronic Products**: Lists all products categorized as electronics with brand and warranty details.
- **View All Clothing Products**: Lists all clothing products with size and color.
- **View Orders by User**: Fetches and shows order details filtered by a specific user ID.
- **View All Users**: Displays all registered users along with their profile info.
- **Create Product**: Admins can add new products (either electronic or clothing) to the inventory.
- **Update Product**: Enables modification of product details like price, quantity, or category-specific attributes.
- **View All Products**: Lists the complete product catalog.
- **Logout**: Ends the admin session.

SQL Database Design

The backend of the Order Management System is powered by a MySQL relational database named orderManagement. The schema includes several interrelated tables to support users, products, and orders while maintaining normalization and referential integrity.

user Table

This table stores user credentials and roles.

Column	Data Type	Description
user_id	INT	Primary Key (Auto-incremented ID)
username	VARCHAR(50)	Unique username for login
password	VARCHAR(100)	Password (hashed/stored)
role	VARCHAR(10)	Role: Either 'Admin' or 'User'
address	VARCHAR(255)	Address of the user

A CHECK constraint ensures valid roles, and an address column is added for order delivery purposes.

product Table

This table stores all products, regardless of category.

Column	Data Type	Description
product_id	INT	Primary Key
product_name	VARCHAR(100)	Name of the product

Column	Data Type	Description
description	TEXT	Description of the product
price	DOUBLE	Price per unit
quantity_in_stock	INT	Number of available units
type	VARCHAR(20)	Product type: 'Electronics' or 'Clothing'

The type column distinguishes product categories, which are detailed in separate child tables.

electronics Table

Stores additional attributes specific to electronic items.

Column	Data Type	Description
product_id	INT	Foreign key to product table
brand	VARCHAR(50)	Brand name of the electronic item
warranty_period	INT	Warranty period in months

Uses a foreign key reference to product_id and ON DELETE CASCADE to ensure consistency.

clothing Table

Stores attributes specific to clothing items.

Column	Data Type	Description
product_id	INT	Foreign key to product table
size	VARCHAR(10)	Clothing size (e.g., M, L, XL)
color	VARCHAR(30)	Color of the clothing item

Similar to the electronics table, this uses ON DELETE CASCADE to maintain integrity.

orders Table

Stores order information placed by users.

Column	Data Type	Description
order_id	INT	Primary Key
user_id	INT	Foreign key from user table

Column	Data Type	Description
product_id	INT	Foreign key from product table
quantity	INT	Quantity of product ordered
total_amt	DOUBLE	Total bill amount
payment_mode	VARCHAR(20)	Allowed: 'Cash', 'Card', 'UPI'
status	VARCHAR(20)	Default: 'Placed'. Allowed: 'Cancelled', 'Delivered'
order_date	TIMESTAMP	Automatically set at the time of order placement

This table connects users and products and tracks the order lifecycle using a status field.

Project Structure

The project follows a modular and organized structure, adhering to best practices of separation of concerns. Each component is placed under relevant packages for better maintainability and scalability.

1. entity/

- Contains POJO classes representing tables in the database.
- Each class includes attributes, constructors, getters, setters, and toString() methods.

2. exception/

- Includes user-defined exceptions like:
  - UserNotFoundException: Thrown when operations involve a non-existent user.
  - OrderNotFoundException: Thrown if an invalid order ID is referenced.

3. dao/

- Declares the IOrderManagementRepository interface, defining all the core functionalities like placing orders, updating products, viewing users, etc and IAdmin interface defining all the admin actions.
- Contains the implementation class OrderProcessor handles all database operations using JDBC and User actions.
- Contains the implementation class AdminProcessor handles admin actions

5. util/

- Includes the DBConnUtil class which manages connection to the MySQL database.

6. main/

- MainModule.java contains the entry point of the program.
- Displays login/registration and role-based menus.

Key Components

1. entity Package

This package includes all the core domain classes (entities) that directly represent the real-world objects in the system such as users, products, and orders.

1. User Class

Represents a user in the system. A user can either be a regular **User** or an **Admin**.

Attributes:

- userId – Unique identifier for each user.
- username – Login name.
- password – Login password.
- role – Can be "admin" or "user".
- address – User's residential address.

Example:

Used during login, registration, profile update, and order placement to associate actions with specific users.

2. Product Class (Parent Class)

Represents a generic product and serves as a **base class** for specialized product types like Electronics and Clothing.

Attributes:

- productId – Unique product ID.
- productName – Name of the product.
- description – Product description.
- price – Cost of the product.
- quantityInStock – Available stock count.
- type – Product category ("clothing" or "electronics").

Note:

This class is **inherited** by Clothing and Electronics to enable polymorphism.

3. Electronics Class (Subclass of Product)

Represents an electronic product with extra details specific to electronics.

Additional Attributes:

- brand – Brand of the product (e.g., Samsung, Sony).

- warrantyPeriod – Warranty in months.

Example:

Used when admins create or update electronic products.

4. Clothing Class (Subclass of Product)

Represents clothing products with attributes that define style.

Additional Attributes:

- size – Size of the clothing (e.g., S, M, L, XL).
- color – Color of the item.

Example:

Used to manage and view clothing-specific product details.

5. Order Class

Represents a product order made by a user.

Attributes:

- orderId – Unique identifier for each order.
- userId – The user who placed the order.
- productId – Product being ordered.
- quantity – Quantity ordered.
- totalAmt – Total price for the quantity ordered.
- paymentMode – Payment method (e.g., UPI, Card, COD).
- status – Status of the order ("Pending", "Cancelled", etc.).
- orderDate – Timestamp when the order was placed.

Example:

Used to store, retrieve, and track orders placed by users.

Relationships:

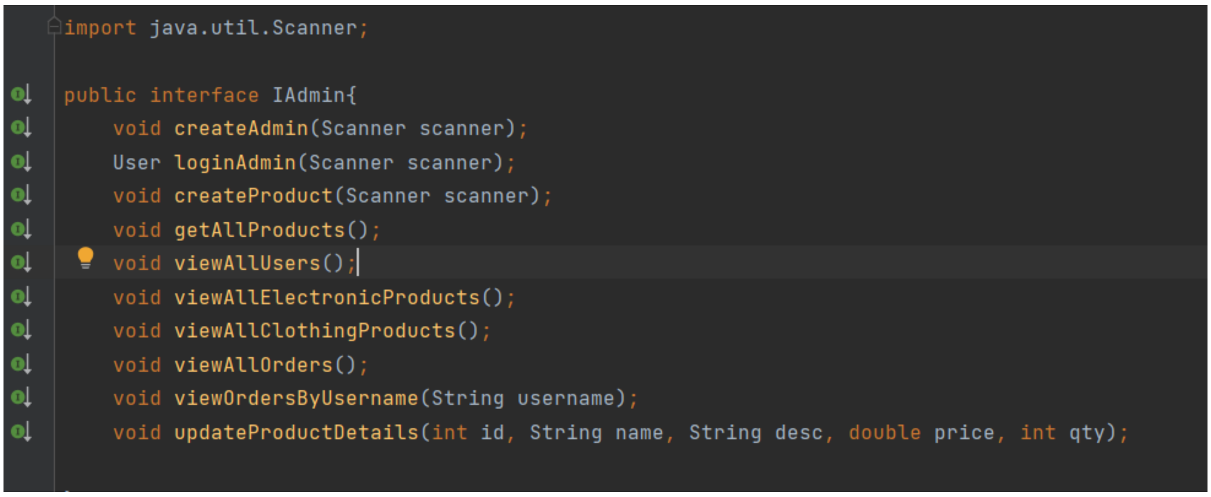
- **User ↔ Order**: A user can place multiple orders.
- **Product ↔ Electronics / Clothing**: Inheritance relationship for specialized products.
- **Product ↔ Order**: An order contains one product.

2. dao Package

The dao package is responsible for handling all admin-related data access operations. It includes:

- IAdmin Interface: Defines all operations an Admin can perform.
- IOrderManagementRepository – For user-related and order management operations
- AdminProcessor Class: Implements the functionalities defined in the IAdmin interface using JDBC.
- OrderProcessor Class : Implements the functionalities defined in IOrderManagementRepository the interface using JDBC.

IAdmin Interface and AdminProcessor Class :



```java
import java.util.Scanner;

public interface IAdmin{
    void createAdmin(Scanner scanner);
    User loginAdmin(Scanner scanner);
    void createProduct(Scanner scanner);
    void getAllProducts();
    void viewAllUsers();
    void viewAllElectronicProducts();
    void viewAllClothingProducts();
    void viewAllOrders();
    void viewOrdersByUsername(String username);
    void updateProductDetails(int id, String name, String desc, double price, int qty);
```

Method	Description
createAdmin(Scanner scanner)	Prompts the user for admin credentials and creates a new Admin entry in the user table.
loginAdmin(Scanner scanner)	Validates Admin login using username and password against the database. Returns a User object if successful.
createProduct(Scanner scanner)	Allows admin to enter details of a product and inserts it into the product table. Also adds extended attributes into either electronics or clothing table.
getAllProducts()	Displays all products including both generic and category-specific details (like brand, warranty for electronics and size, color for clothing).
viewAllUsers()	Lists all users with role 'user', showing their ID, username, role, and address.

Method	Description
viewAllElectronicProducts()	Shows only products of type Electronics by joining product and electronics tables.
viewAllClothingProducts()	Shows only products of type Clothing by joining product and clothing tables.
viewAllOrders()	Retrieves and displays all orders placed in the system (implementation not shown in current code).
viewOrdersByUsername(String username)	Shows orders placed by a specific user based on the username (implementation not shown in current code).
updateProductDetails(int id, String name, String desc, double price, int qty)	Updates product details for the given product ID. (implementation not shown in current code).

IOrderManagementRepository Interface and OrderProcessor Class :

Method	Description
createUser(Scanner scanner)	Takes input from the user and inserts a new user into the database.
loginUser(Scanner scanner)	Authenticates user credentials (username/password) and returns a User object if valid.
createOrder(User user)	Allows a user to place an order by selecting a product, quantity, and address.
cancelOrder(int orderId, int userId)	Cancels an existing order by verifying order and user IDs. Throws exceptions if not valid.
getOrderByUser(User user)	Displays all orders made by the specific user.
updateProfile(User user)	Lets the user update their profile information such as username, password, and address.
deleteAccount(User user)	Deletes the user account and their orders from the system.

3. exception Package

This package contains custom exception classes designed to handle specific runtime issues in the order management system.

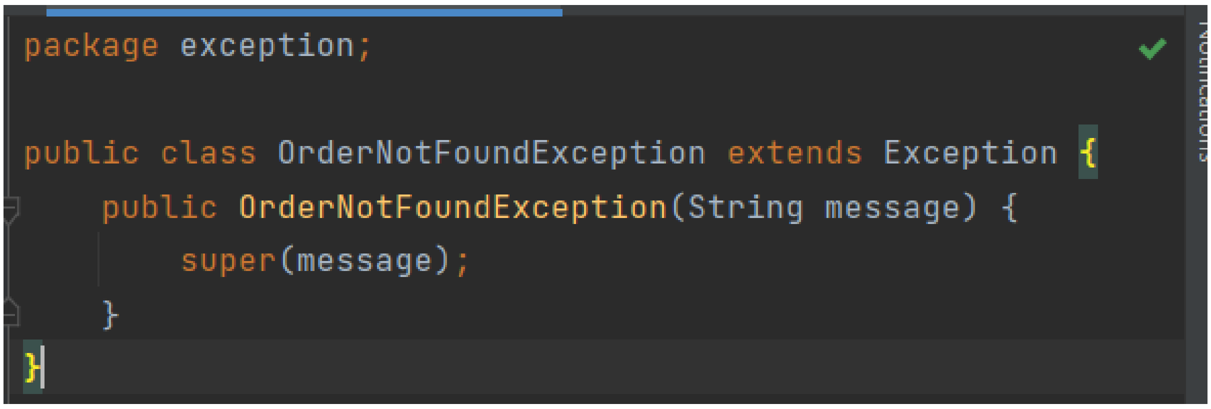
1. OrderNotFoundException

2. UserNotFoundException

1. OrderNotFoundException

OrderNotFoundException is a **custom checked exception**.It is used to signal that a specific **order could not be found** in the database.This might occur when:

- A user tries to cancel a non-existing order.
- An order ID entered by the user doesn't match any records in the orders table.



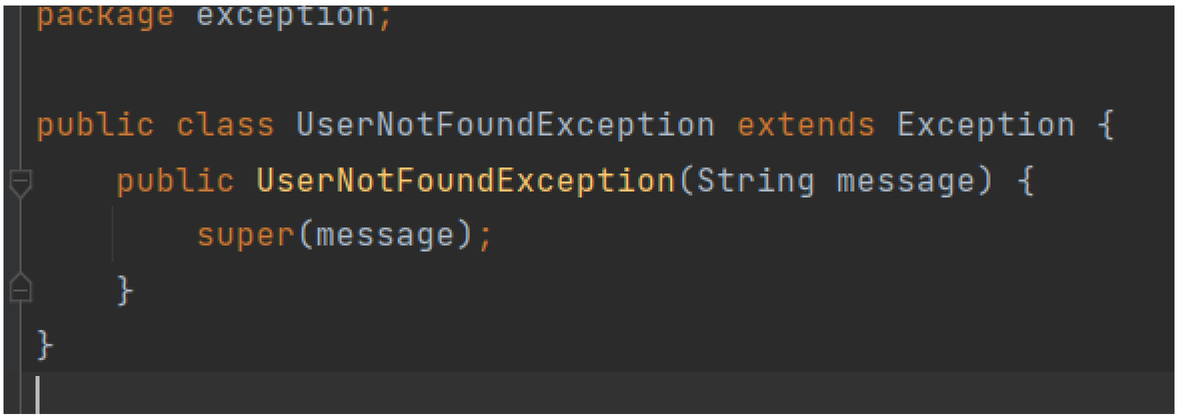
```
package exception;

public class OrderNotFoundException extends Exception {
    public OrderNotFoundException(String message) {
        super(message);
    }
}
```

2. UserNotFoundException

UserNotFoundException is a custom checked exception. It is used to handle cases where a user is not found in the database.

This might occur when:

- Logging in with incorrect credentials.
- Trying to fetch or cancel an order for a non-existent user.
- Deleting or updating a profile of a user that no longer exists.



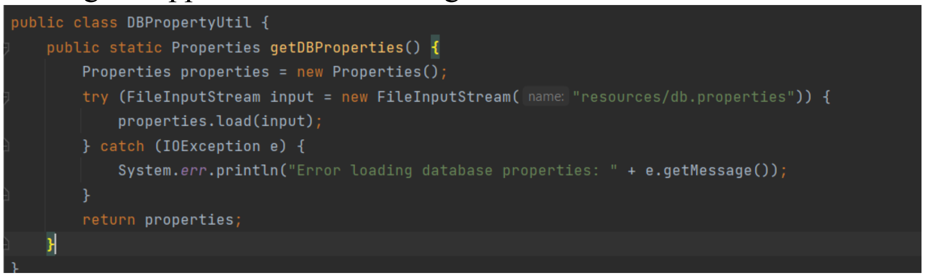
```
package exception;

public class UserNotFoundException extends Exception {
    public UserNotFoundException(String message) {
        super(message);
    }
}
```

4. util Package

This package contains utility classes used for managing database configuration and establishing database connections in a modular, reusable way.

DBPropertyUtil Class

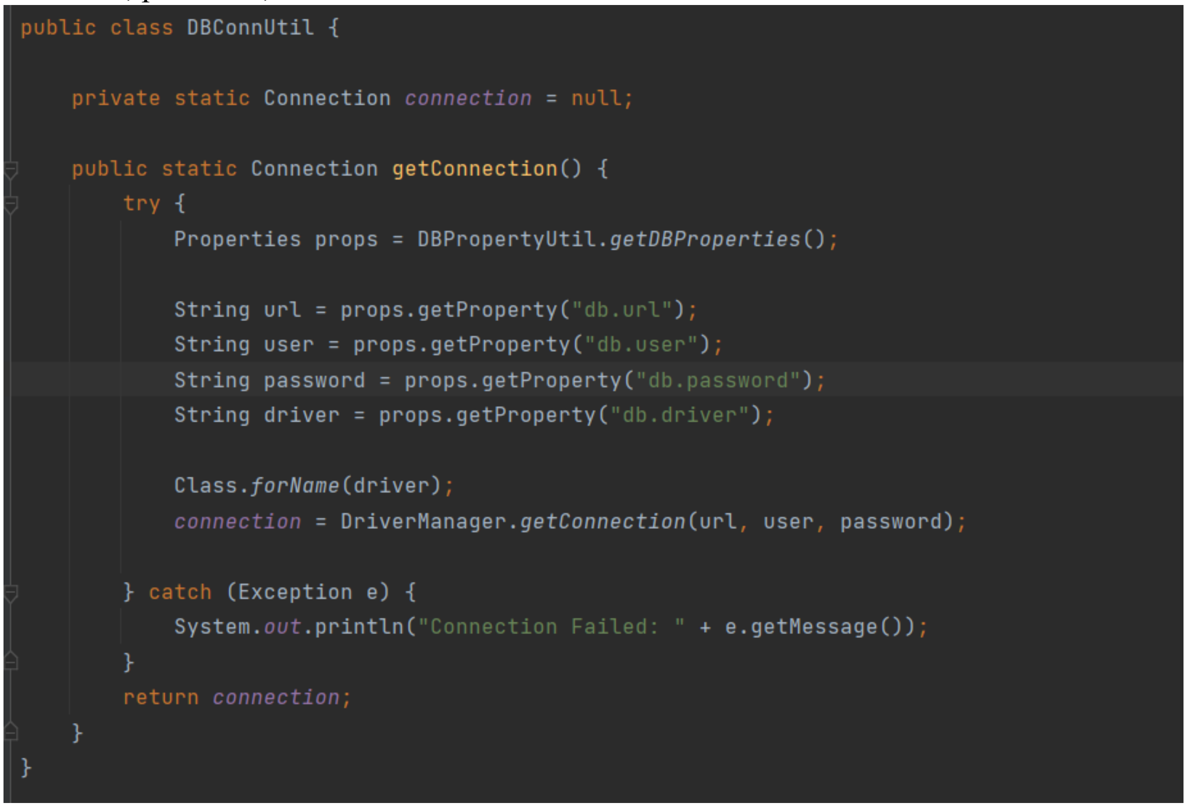
- This utility class is responsible for reading the database configuration from an external db.properties file.
- It loads properties like db.url, db.user, db.password, and db.driver which are used to set up the connection.
- By using a properties file, the database settings are kept separate from the source code, making the application more configurable and secure.



```java
public class DBPropertyUtil {
    public static Properties getDBProperties() {
        Properties properties = new Properties();
        try (FileInputStream input = new FileInputStream( name: "resources/db.properties")) {
            properties.load(input);
        } catch (IOException e) {
            System.err.println("Error loading database properties: " + e.getMessage());
        }
        return properties;
    }
}
```

DBConnUtil Class

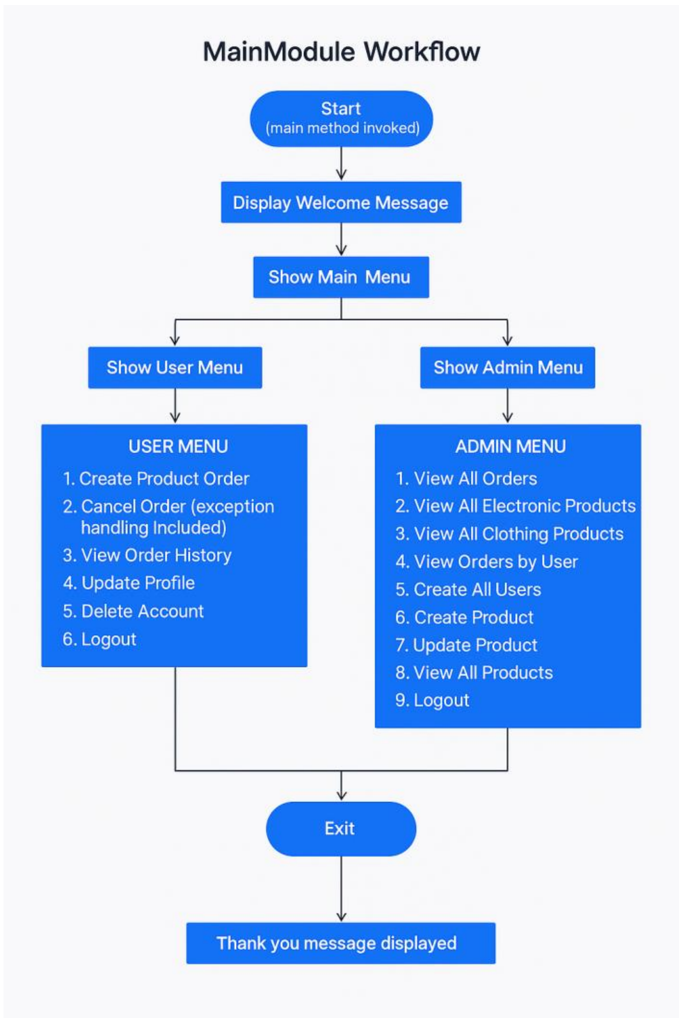
- This class is responsible for establishing a connection with the database.
- It uses the properties loaded by DBPropertyUtil to retrieve connection parameters.
- Uses DriverManager.getConnection() to open a connection using the retrieved URL, username, password, and driver.



```java
public class DBConnUtil {

    private static Connection connection = null;

    public static Connection getConnection() {
        try {
            Properties props = DBPropertyUtil.getDBProperties();

            String url = props.getProperty("db.url");
            String user = props.getProperty("db.user");
            String password = props.getProperty("db.password");
            String driver = props.getProperty("db.driver");

            Class.forName(driver);
            connection = DriverManager.getConnection(url, user, password);

        } catch (Exception e) {
            System.out.println("Connection Failed: " + e.getMessage());
        }
        return connection;
    }
}
```

5. main Package

MainModule serves as the entry point of the Order Management System. It presents a menu-driven console interface that allows users and administrators to interact with the system. The class handles user input and delegates operations to the respective service layers (OrderProcessor for users and AdminProcessor for admins).



Work Flow

main(String[] args)

- Displays a welcome message.
- Continuously shows the main menu to navigate as a User or Admin.
- Routes user input to the appropriate section (userActions, adminActions, or exits the program).

UserActions(Scanner scanner, OrderProcessor orderProcessor)

- Provides a sub-menu for user actions:
  - **Create Account**
  - **Login**

    - **Exit**
- On successful login, redirects the user to userMenu.

adminActions(Scanner scanner, AdminProcessor adminProcessor)

- Provides a sub-menu for admin actions:
    - **Create Account**
    - **Login**
    - **Exit**
- On successful login, redirects the admin to adminMenu.

userMenu(Scanner scanner, User user)

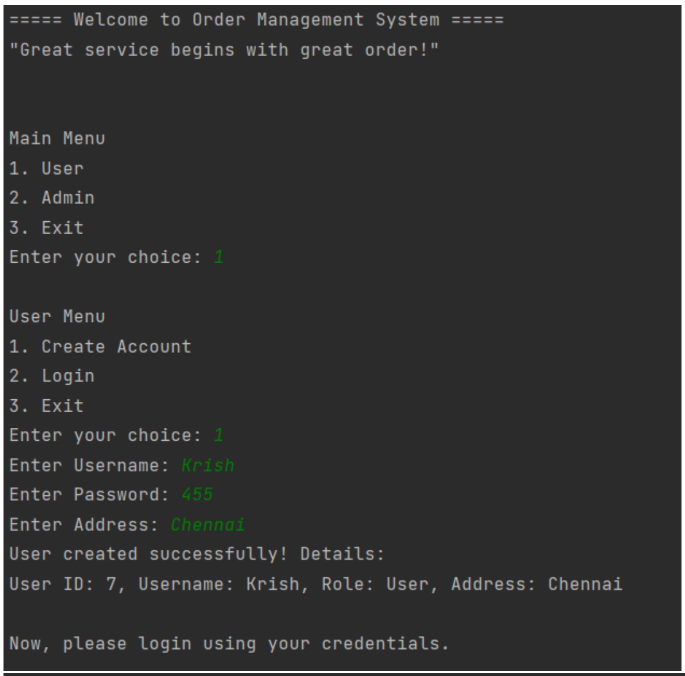
- Available after a user logs in.
- Options include:
    - **Create Order**
    - **Cancel Order**
    - **View Order History**
    - **Update Profile**
    - **Delete Account**
    - **Logout**
- Handles cancellation with exception management for OrderNotFoundException and UserNotFoundException.

adminMenu(Scanner scanner)

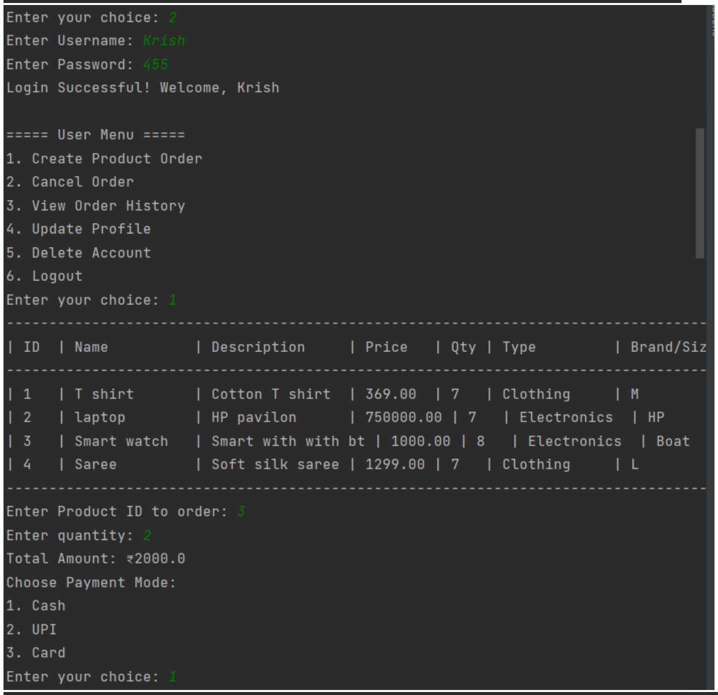
- Available after an admin logs in.
- Options include:
    - **View All Orders**
    - **View Products by Category (Electronic/Clothing)**
    - **View Orders by User**
    - **View All Users**
    - **Create/Update Product**
    - **View All Products**
    - **Logout**
- Allows inline updating of product fields. Accepts 'null' or blank input to skip updating specific fields.

Output :

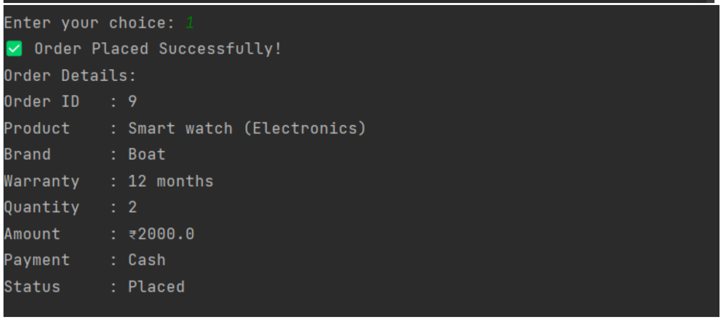
User



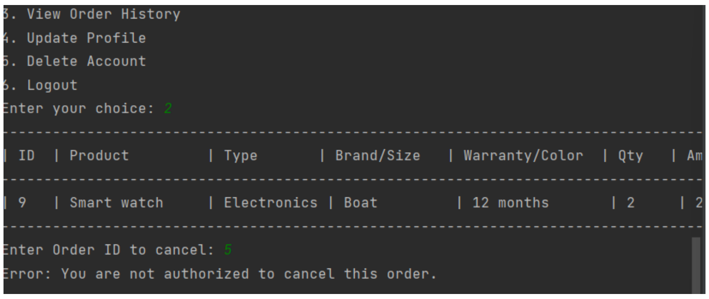
```
===== Welcome to Order Management System =====
"Great service begins with great order!"


Main Menu
1. User
2. Admin
3. Exit
Enter your choice: 1

User Menu
1. Create Account
2. Login
3. Exit
Enter your choice: 1
Enter Username: Krish
Enter Password: 455
Enter Address: Chennai
User created successfully! Details:
User ID: 7, Username: Krish, Role: User, Address: Chennai

Now, please login using your credentials.
```



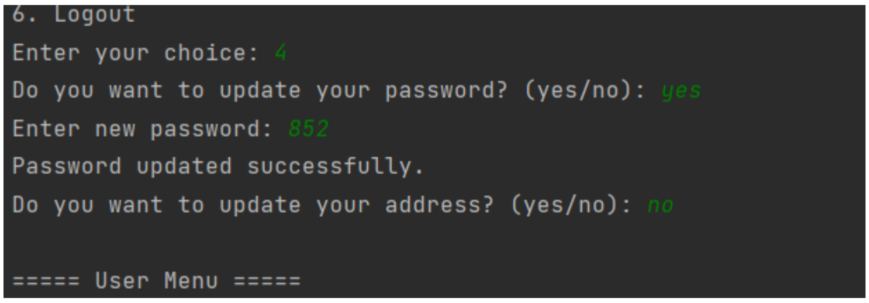
```
Enter your choice: 2
Enter Username: Krish
Enter Password: 455
Login Successful! Welcome, Krish

===== User Menu =====
1. Create Product Order
2. Cancel Order
3. View Order History
4. Update Profile
5. Delete Account
6. Logout
Enter your choice: 1
---------------------------------------------------------------------------------
| ID  | Name          | Description      | Price    | Qty | Type          | Brand/Siz
---------------------------------------------------------------------------------
| 1   | T shirt       | Cotton T shirt  | 369.00   | 7   | Clothing     | M
| 2   | laptop        | HP pavilon       | 750000.00 | 7   | Electronics  | HP
| 3   | Smart watch   | Smart with with bt | 1000.00 | 8   | Electronics  | Boat
| 4   | Saree         | Soft silk saree | 1299.00 | 7   | Clothing     | L
---------------------------------------------------------------------------------
Enter Product ID to order: 3
Enter quantity: 2
Total Amount: ₹2000.0
Choose Payment Mode:
1. Cash
2. UPI
3. Card
Enter your choice: 1
```



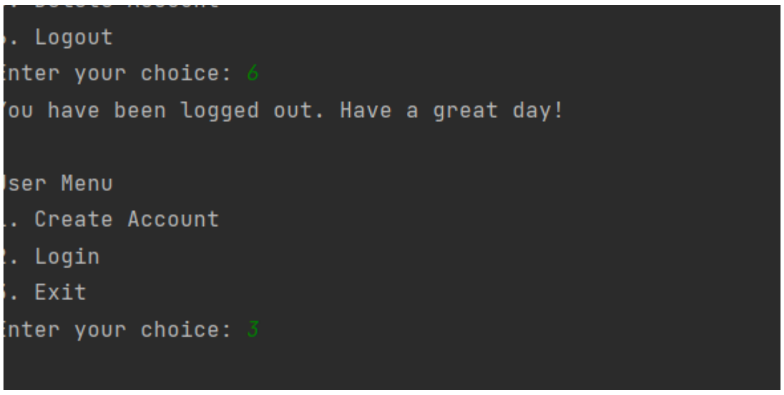
```
Enter your choice: 1
✅ Order Placed Successfully!
Order Details:
Order ID    : 9
Product     : Smart watch (Electronics)
Brand       : Boat
Warranty    : 12 months
Quantity    : 2
Amount      : ₹2000.0
Payment     : Cash
Status      : Placed
```



```
3. View Order History
4. Update Profile
5. Delete Account
6. Logout
Enter your choice: 2
-----------------------------------------------------------------------------------------
| ID  | Product        | Type        | Brand/Size    | Warranty/Color   | Qty    | Am
-----------------------------------------------------------------------------------------
| 9   | Smart watch    | Electronics | Boat          | 12 months        | 2      | 2
-----------------------------------------------------------------------------------------
Enter Order ID to cancel: 5
Error: You are not authorized to cancel this order.
```

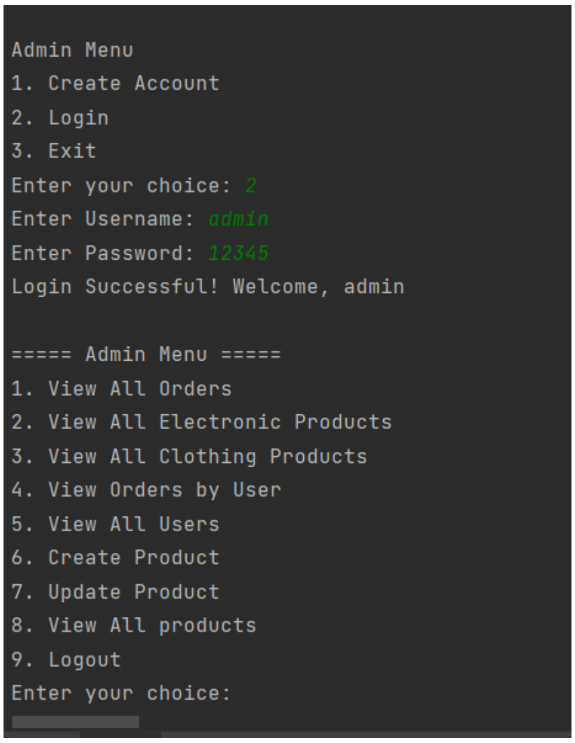


```
6. Logout
Enter your choice: 4
Do you want to update your password? (yes/no): yes
Enter new password: 852
Password updated successfully.
Do you want to update your address? (yes/no): no

===== User Menu =====
```

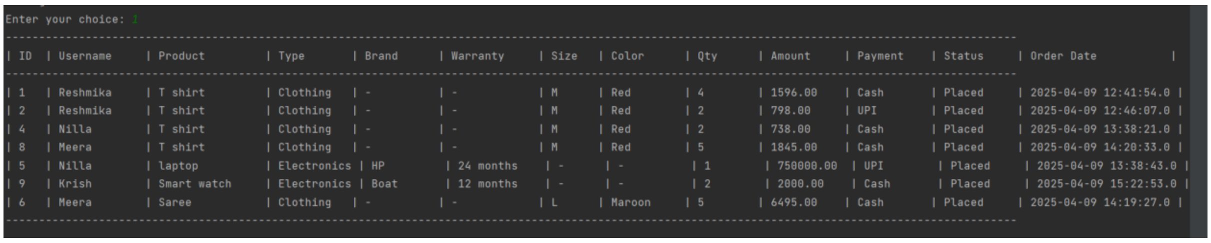


```
. Logout
nter your choice: 6
ou have been logged out. Have a great day!

ser Menu
. Create Account
. Login
. Exit
nter your choice: 3
```

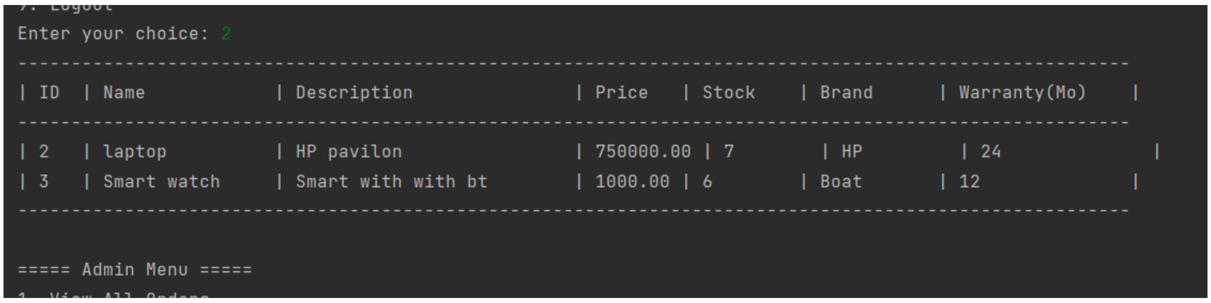
Admin



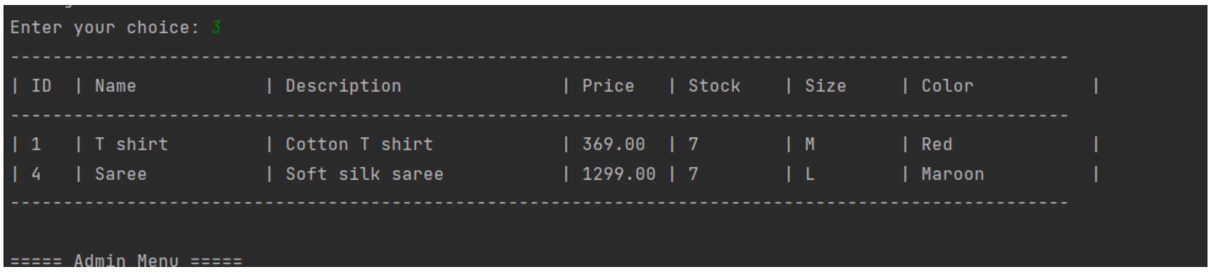
```
Admin Menu
1. Create Account
2. Login
3. Exit
Enter your choice: 2
Enter Username: admin
Enter Password: 12345
Login Successful! Welcome, admin

===== Admin Menu =====
1. View All Orders
2. View All Electronic Products
3. View All Clothing Products
4. View Orders by User
5. View All Users
6. Create Product
7. Update Product
8. View All products
9. Logout
Enter your choice:
```



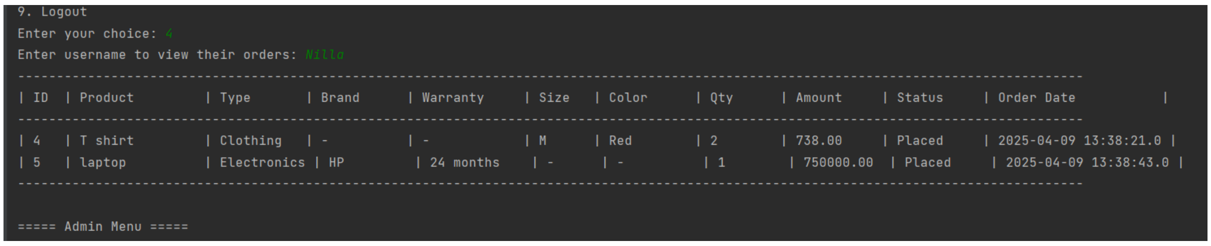
```
Enter your choice: 1
------------------------------------------------------------------------------------------------------------------------------------------------------
| ID  | Username     | Product         | Type        | Brand        | Warranty     | Size    | Color       | Qty       | Amount      | Payment     | Status      | Order Date            |
------------------------------------------------------------------------------------------------------------------------------------------------------
| 1   | Reshmika     | T shirt         | Clothing    | -            | -            | M       | Red         | 4         | 1596.00     | Cash        | Placed      | 2025-04-09 12:41:54.0 |
| 2   | Reshmika     | T shirt         | Clothing    | -            | -            | M       | Red         | 2         | 798.00      | UPI         | Placed      | 2025-04-09 12:46:07.0 |
| 4   | Nilla        | T shirt         | Clothing    | -            | -            | M       | Red         | 2         | 738.00      | Cash        | Placed      | 2025-04-09 13:38:21.0 |
| 8   | Meera        | T shirt         | Clothing    | -            | -            | M       | Red         | 5         | 1845.00     | Cash        | Placed      | 2025-04-09 14:20:33.0 |
| 5   | Nilla        | laptop          | Electronics | HP           | 24 months    | -       | -           | 1         | 750000.00   | UPI         | Placed      | 2025-04-09 13:38:43.0 |
| 9   | Krish        | Smart watch     | Electronics | Boat         | 12 months    | -       | -           | 2         | 2000.00     | Cash        | Placed      | 2025-04-09 15:22:53.0 |
| 6   | Meera        | Saree           | Clothing    | -            | -            | L       | Maroon      | 5         | 6495.00     | Cash        | Placed      | 2025-04-09 14:19:27.0 |
------------------------------------------------------------------------------------------------------------------------------------------------------
```



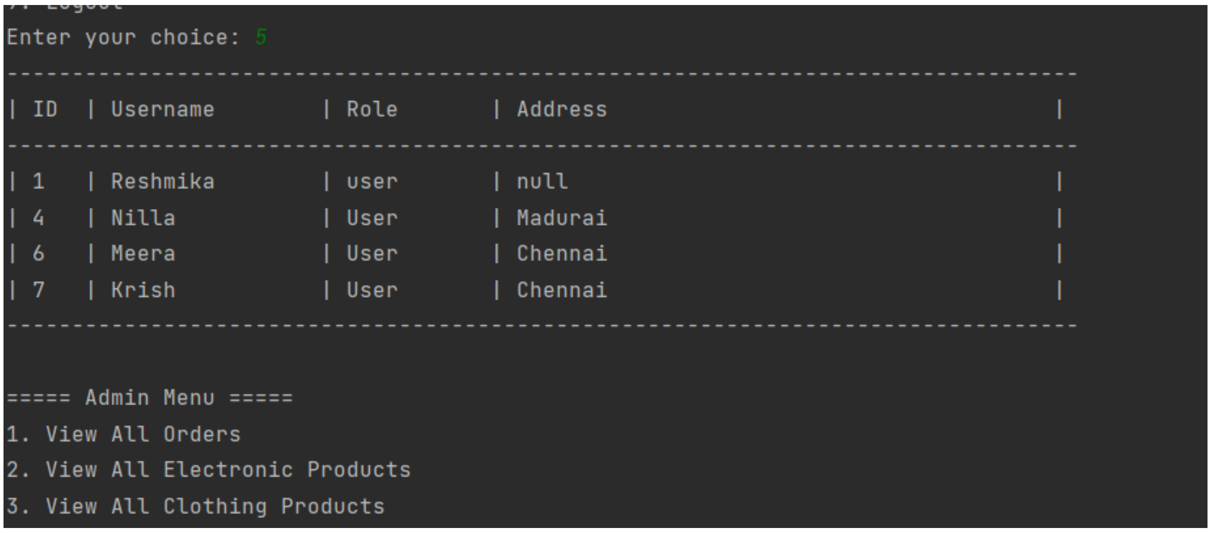
```
Enter your choice: 2
------------------------------------------------------------------------------------------------------------
| ID   | Name                 | Description                    | Price     | Stock      | Brand         | Warranty(Mo)     |
------------------------------------------------------------------------------------------------------------
| 2    | laptop               | HP pavilon                     | 750000.00 | 7          | HP            | 24               |
| 3    | Smart watch          | Smart with with bt             | 1000.00   | 6          | Boat          | 12               |
------------------------------------------------------------------------------------------------------------


===== Admin Menu =====
```



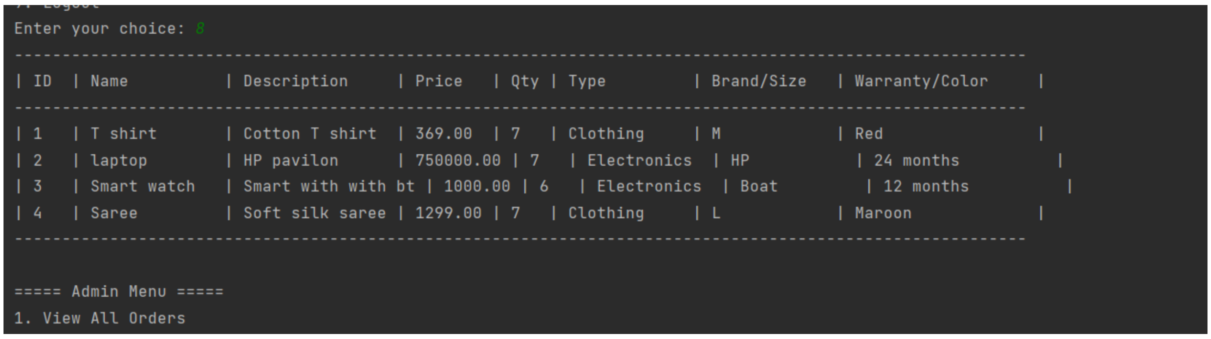
```
Enter your choice: 3
------------------------------------------------------------------------------------------------------
| ID   | Name                 | Description                    | Price     | Stock      | Size       | Color           |
------------------------------------------------------------------------------------------------------
| 1    | T shirt              | Cotton T shirt                 | 369.00    | 7          | M          | Red             |
| 4    | Saree                | Soft silk saree                | 1299.00   | 7          | L          | Maroon          |
------------------------------------------------------------------------------------------------------


===== Admin Menu =====
```



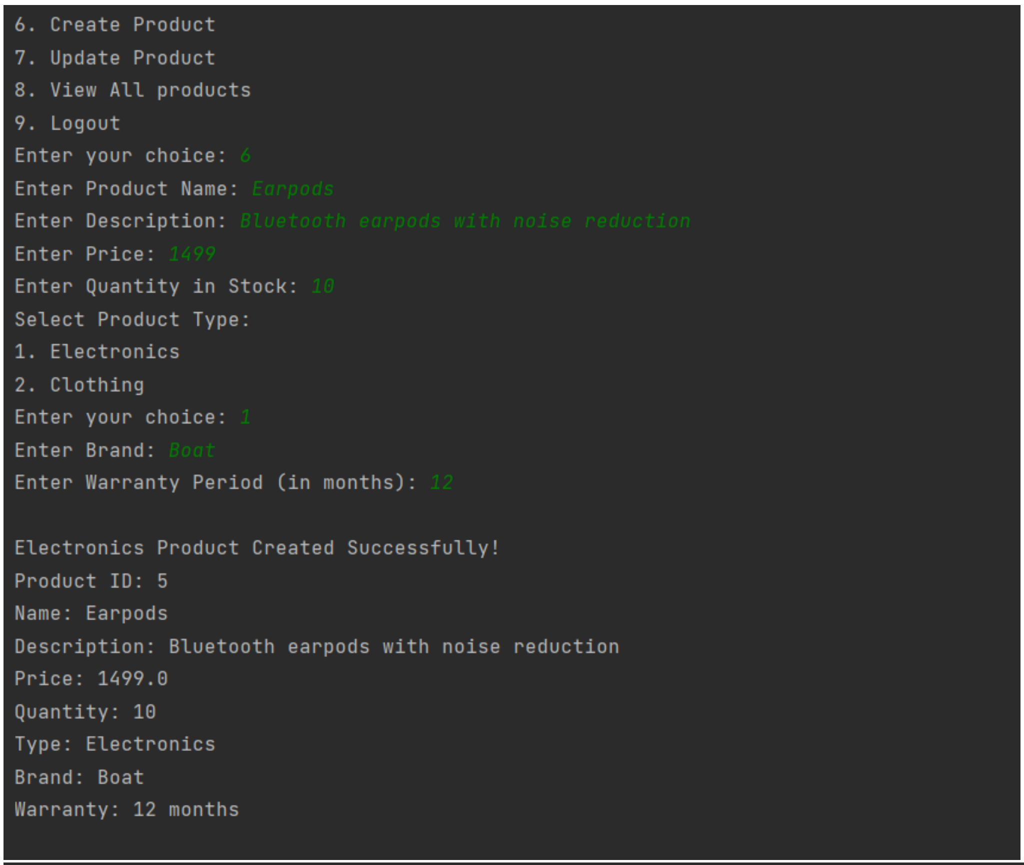
```
9. Logout
Enter your choice: 4
Enter username to view their orders: Nilla
----------------------------------------------------------------------------------------------------------------------------------
| ID  | Product          | Type        | Brand       | Warranty     | Size   | Color       | Qty       | Amount      | Status      | Order Date            |
----------------------------------------------------------------------------------------------------------------------------------
| 4   | T shirt          | Clothing    | -           | -            | M      | Red         | 2         | 738.00      | Placed      | 2025-04-09 13:38:21.0 |
| 5   | laptop           | Electronics | HP          | 24 months    | -      | -           | 1         | 750000.00   | Placed      | 2025-04-09 13:38:43.0 |
----------------------------------------------------------------------------------------------------------------------------------


===== Admin Menu =====
```



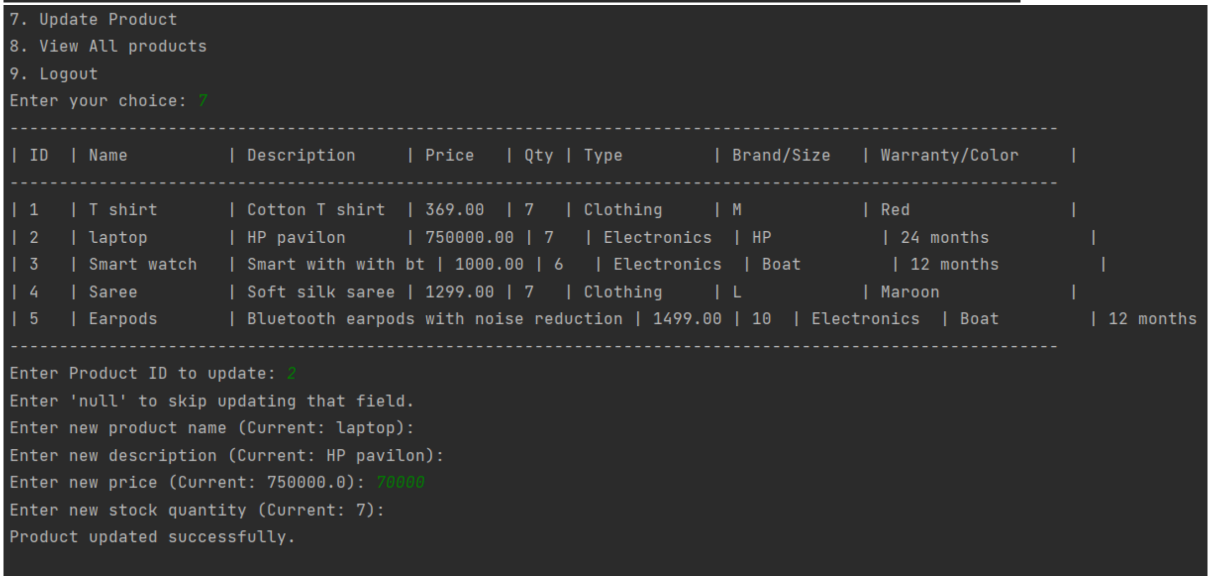
```
Enter your choice: 5
----------------------------------------------------------------------------------
| ID   | Username          | Role         | Address                                  |
----------------------------------------------------------------------------------
| 1    | Reshmika          | user         | null                                     |
| 4    | Nilla             | User         | Madurai                                  |
| 6    | Meera             | User         | Chennai                                  |
| 7    | Krish             | User         | Chennai                                  |
----------------------------------------------------------------------------------


===== Admin Menu =====
1. View All Orders
2. View All Electronic Products
3. View All Clothing Products
```



```
Enter your choice: 8
----------------------------------------------------------------------------------------------------------
| ID  | Name          | Description     | Price    | Qty | Type         | Brand/Size    | Warranty/Color     |
----------------------------------------------------------------------------------------------------------
| 1   | T shirt       | Cotton T shirt  | 369.00   | 7   | Clothing     | M             | Red                |
| 2   | laptop        | HP pavilon      | 750000.00 | 7   | Electronics  | HP            | 24 months          |
| 3   | Smart watch   | Smart with with bt | 1000.00 | 6   | Electronics  | Boat          | 12 months          |
| 4   | Saree         | Soft silk saree | 1299.00 | 7   | Clothing     | L             | Maroon             |
----------------------------------------------------------------------------------------------------------


===== Admin Menu =====
1. View All Orders
```



```
6. Create Product
7. Update Product
8. View All products
9. Logout
Enter your choice: 6
Enter Product Name: Earpods
Enter Description: Bluetooth earpods with noise reduction
Enter Price: 1499
Enter Quantity in Stock: 10
Select Product Type:
1. Electronics
2. Clothing
Enter your choice: 1
Enter Brand: Boat
Enter Warranty Period (in months): 12

Electronics Product Created Successfully!
Product ID: 5
Name: Earpods
Description: Bluetooth earpods with noise reduction
Price: 1499.0
Quantity: 10
Type: Electronics
Brand: Boat
Warranty: 12 months
```



```
7. Update Product
8. View All products
9. Logout
Enter your choice: 7
----------------------------------------------------------------------------------------------------------
| ID  | Name          | Description     | Price    | Qty | Type         | Brand/Size    | Warranty/Color     |
----------------------------------------------------------------------------------------------------------
| 1   | T shirt       | Cotton T shirt  | 369.00   | 7   | Clothing     | M             | Red                |
| 2   | laptop        | HP pavilon      | 750000.00 | 7   | Electronics  | HP            | 24 months          |
| 3   | Smart watch   | Smart with with bt | 1000.00 | 6   | Electronics  | Boat          | 12 months          |
| 4   | Saree         | Soft silk saree | 1299.00 | 7   | Clothing     | L             | Maroon             |
| 5   | Earpods       | Bluetooth earpods with noise reduction | 1499.00 | 10  | Electronics  | Boat          | 12 months
----------------------------------------------------------------------------------------------------------
Enter Product ID to update: 2
Enter 'null' to skip updating that field.
Enter new product name (Current: laptop):
Enter new description (Current: HP pavilon):
Enter new price (Current: 750000.0): 70000
Enter new stock quantity (Current: 7):
Product updated successfully.
```

Conclusion:

In this module, I have successfully implemented the core structure and functionality of the Order Management System. All the assigned tasks and functionalities were completed, including separate login flows for users and admins, detailed menu navigation, and interaction with service layers for handling products, users, and orders. Proper exception handling and user-friendly prompts were also integrated to enhance the overall user experience. This module serves as the backbone of the system, ensuring smooth control flow between the user interface and the processing logic.