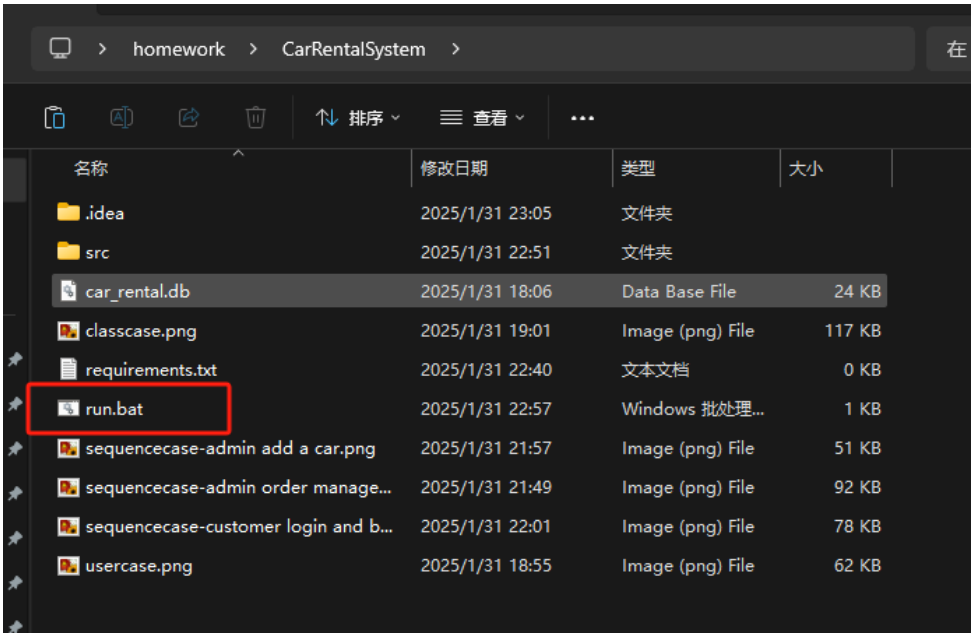
Car Rental System

1. Getting start

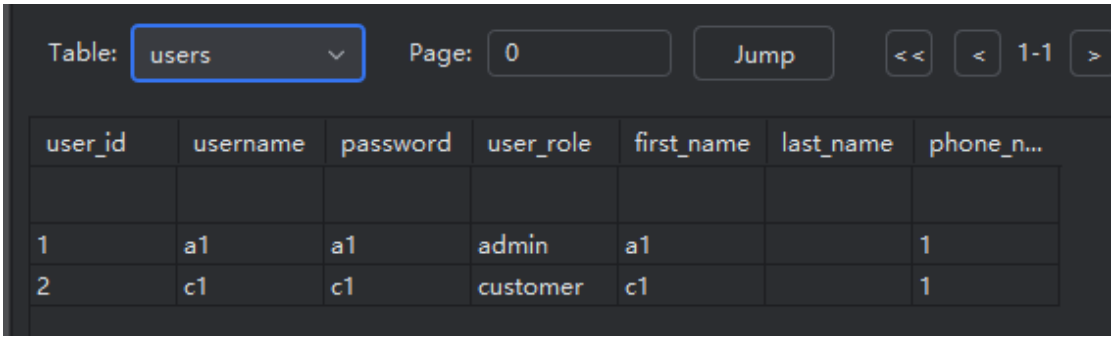
1. JUST CKICK THE FILE “run.bat”, and start to use

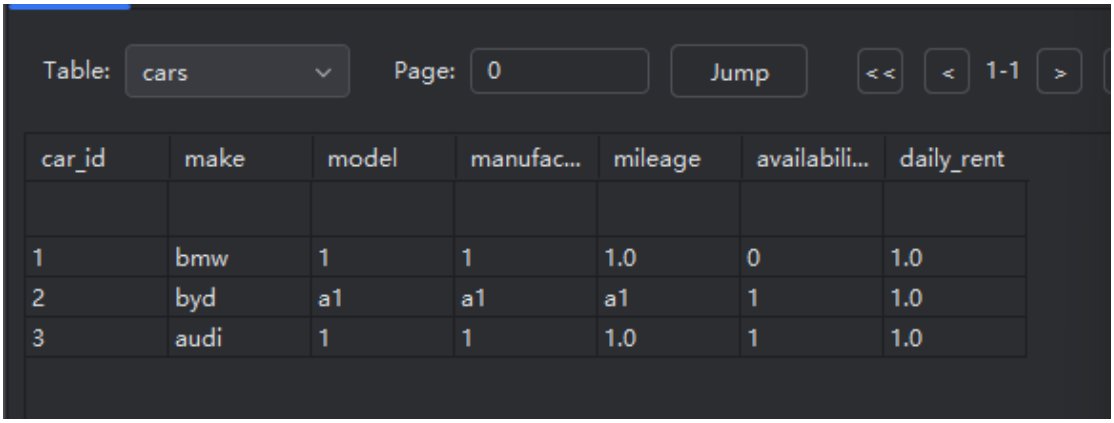


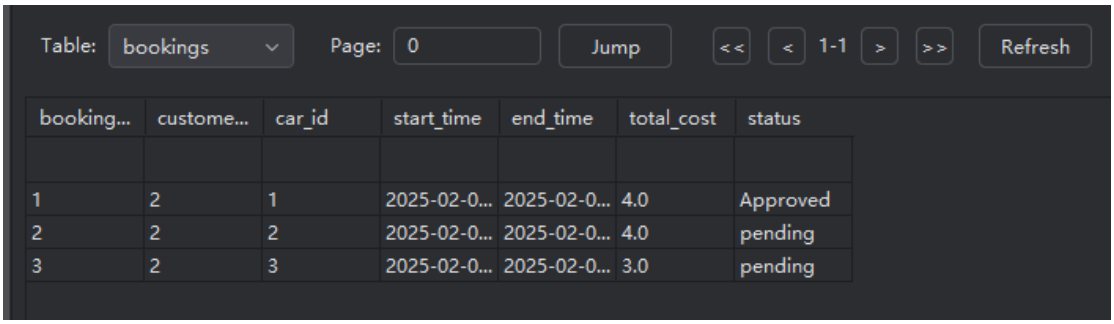
2. Test guide

After running the file “run.bat”

I prepared some accounts for texting, you can use them to login and experience add, remove cars, and approve or reject orderings







2. Overview

The Car Rental System is a CLI (Command line interface) application to manage car rentals for customers and administrators.

Design patterns

the project primarily employs the following design patterns:

1) Layered Architecture (Layered Architecture)

The project separates different functional modules into layers, including:

Database Layer (Database Layer): Responsible for interacting with the database and performing operations such as CRUD (Create, Read, Update, Delete).

Service Layer (Service Layer): Handles business logic processing and calls functions from the database layer.

User Interface Layer (User Interface Layer): Manages user interactions, displays menus, receives input, and shows results.

This layered architecture helps break down a complex system into independent parts, making each layer's responsibilities clear and facilitating development and maintenance.

2) Command Pattern (Command Pattern)

In the project's menu system, each option (such as "login", "register", and "browse available cars") can be considered a command. When a user selects a menu option, the system executes the corresponding operation.

Specifically:

Main Menu (Main Menu): Provides commands for login, registration, and exit.

Customer Menu (Customer Menu): Offers commands to browse vehicles, book vehicles, view booking details, and log out.

Admin Menu (Admin Menu): Provides commands to add vehicles, delete vehicles, update vehicle information, browse orders, manage orders, and log out.

The advantages of the command pattern include encapsulating requests as objects, which makes it easy to extend and modify functionality.

3) Data Access Object Pattern (DAO Pattern)

In the database operations of the project, methods such as add_user, add_car, and update_car_details are defined in the database_operations.py file. These methods encapsulate the specific operations of the database, allowing the service layer (such as user_service.py and car_service.py) to call these methods without needing to know the implementation details of the database.

This pattern's advantage is separating data access logic from business logic, improving code maintainability and testability.

4) Factory Pattern (Factory Pattern)

Although the project does not explicitly use the factory pattern, when registering a user, different user objects (Admin or Customer) are created based on the user's role. This can be seen as an application of the simple factory pattern.

Specifically:

```
if role_input == "1":
    user_role = "admin"
elif role_input == "2":
    user_role = "customer"
```

And in the user.py file, different user objects are created based on the role:

```
class Customer(User):
    pass


class Admin(User):
    pass
```

5) State Pattern (State Pattern)

In order management, the status of an order (such as "pending", "Approved", "Rejected") can be seen as an application of the state pattern. Changes in the order's status affect its behavior (such as whether it can be modified or canceled).

Specifically:

In the admin menu, administrators can approve or reject orders, which changes the order's status. Different behaviors are displayed based on the order's status.

6) Singleton Pattern (Singleton Pattern)

Although the project does not explicitly implement the singleton pattern, in database connections, it is typically used to ensure that there is only one database connection instance, avoiding the overhead of multiple database connections.

Conclusion

The design patterns used in the project include:

Layered Architecture (Layered Architecture): Clearly separates database layer, service layer, and user interface layer.

Command Pattern (Command Pattern): Encapsulates menu options as commands for easy extension and maintenance.

Data Access Object Pattern (DAO Pattern): Encapsulates database operations in independent classes to improve maintainability.

Factory Pattern (Factory Pattern): Creates different user objects based on user roles.

State Pattern (State Pattern): Manages changes in order status and their behaviors.

Singleton Pattern (Singleton Pattern): Ensures the uniqueness of database connections (although not explicitly implemented in the code, it is usually necessary).

These design patterns help improve the readability, maintainability, and extensibility of the code, making the project more robust and flexible.

MENU	OPTIONS	DESCRIPTION
Main menu		**Car Rental System**
1	**Login**	Log in with existing credentials. (Already registered users can sign in.)
2	**Register**	Create a new user account. (New users can sign up here.)
3	**Exit**	Exit the application. (Close the program.)
Customer menu		**Customer Menu**
1	**Browse Available Cars**	View a list of available cars with details. (See all cars that are currently available for rental.)
2	**Book a Car**	Select a car and specify the rental period. (Choose a car and set the start and end dates for your rental.)
3	**View Rental Details**	Check booked cars and their status. (View the status of your bookings, including pending, approved, or rejected.)
4	**Logout**	Return to the main menu. (Sign out from the current session and go back to the main menu.)
Admin menu		**Admin Menu**
1	**Add a Car**	Add a new car to the database. (Enter details of a new car to add it to the inventory.)
2	**Delete a Car**	Delete a car from the database. (Remove a car that is no longer available or needed.)
3	**Update Car Details**	Modify details of an existing car. (Change information such as make, model, year, mileage, availability, or daily rent.)
4	**Browse Orders**	View all orders or unapproved orders. (See a list of all orders or filter to see only pending or unapproved orders.)
5	**Manage Orders**	Approve or reject bookings. (Process pending orders by approving or rejecting them.)
6	**Logout**	Return to the main menu. (Sign out from the admin session and go back to the main menu.)

Usage

Main Menu:

Login: Log in with existing credentials.

Register: Create a new user account.

Exit: Exit the application.

Customer Menu:

Browse Available Cars: View a list of available cars with details.

Book a Car: Select a car and specify the rental period.

View Rental Details: Check booked cars and their status.

Logout: Return to the main menu.

Admin Menu:

Add a Car: Add a new car to the database.

Remove a Car: Delete a car from the database.

Update Car Details: Modify details of an existing car.

Browse Orders: View all orders or unapproved orders.

Manage Orders: Approve or reject bookings.

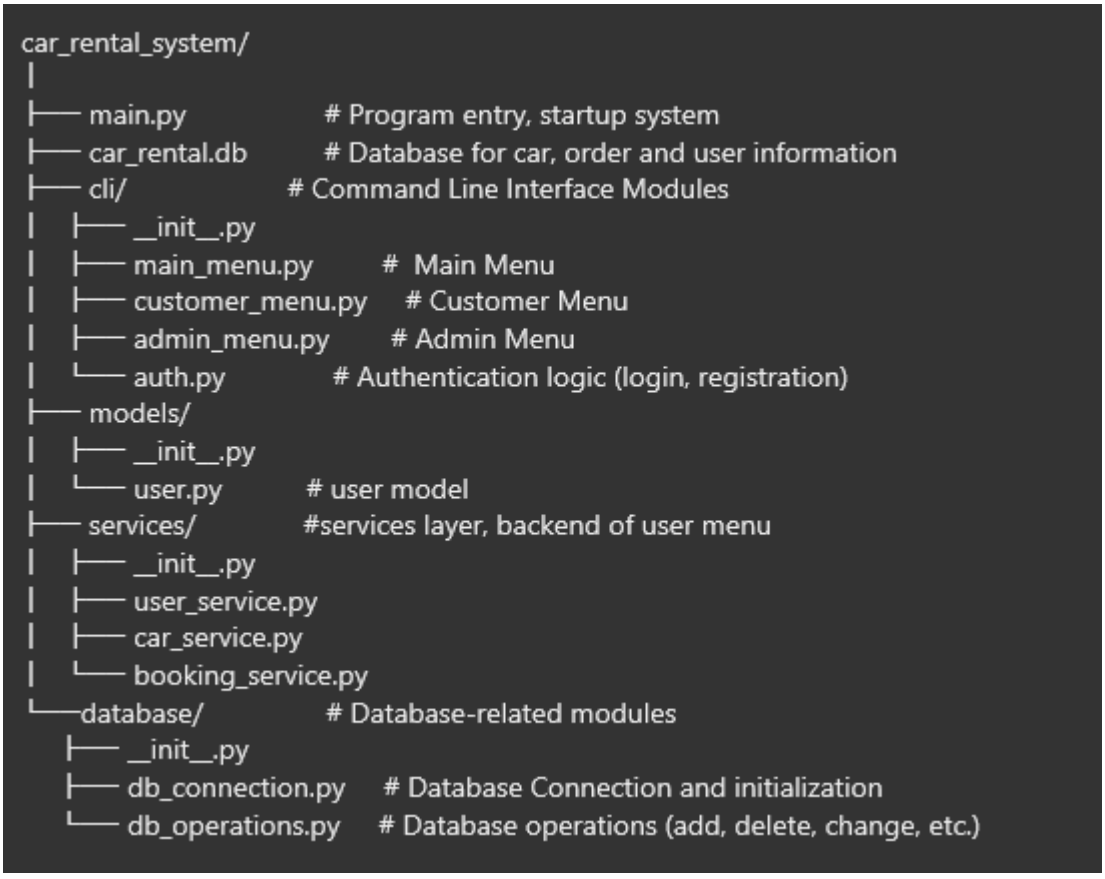
Logout: Return to the main menu.

3. Architecture

The system is organized into the following modules:

1. CLI (Command Line Interface): Handles user interactions through the command line.
2. Models: Defines data models for users, cars, and bookings.
3. Services: Implements business logic for user, car, and booking operations.
4. Database: Manages database connections and operations.

Project Structure



How to use?

1) Users can select an option by entering a number and pressing the Enter key in the command-line interface.

For example:

Enter 1 and press Enter to log in.

Enter 2 and press Enter to register.

Enter 3 and press Enter to exit the system.

```
=== Car Rental System ===
1. Login
2. Register
3. Exit
Enter your choice number(1-3): 2
```

2) For other operations, follow the system guidelines to enter text and press Enter to confirm.

```
Enter your choice number(1-3): 2
Enter your username: admin1
Enter your password: admin1
Enter phone number: 123
Enter your role (1 for admin, 2 for customer): 1
Registration successful! Welcome, admin1 (admin).
```

4. Features

Register:

Users choose to become administrators or customers by registering.

```
=== Car Rental System ===
1. Login
2. Register
3. Exit
Enter your choice number(1-3): 2
Enter your username: admin1
Enter your password: admin1
Enter phone number: 123
Enter your role (1 for admin, 2 for customer): 1
Registration successful! Welcome, admin1 (admin).
```

User Roles

Customer:

Customer main menu ↓

```
=== Customer Menu ===
1. Browse Available Cars
2. Book a Car
3. View Rental Details
4. Logout

Enter your choice(number 1-4): 4
```

1. Browse available cars.

Output of "1" option

```
Available Cars:
-------------------------------------------------------------------------------
ID   Make          Model         Year      Mileage   Price/Day Availability
-------------------------------------------------------------------------------
2    BMW           1             1         1.0       1.0       Available
-------------------------------------------------------------------------------
```

2. Book a car for a specific period.

Output of "2" option and using process

```
Available Cars:
-------------------------------------------------------------------------------
ID   Make          Model         Year      Mileage   Price/Day Availability
-------------------------------------------------------------------------------
2    BMW           1             1         1.0       1.0       Available
-------------------------------------------------------------------------------
Enter the car ID you want to book: 2
Enter start date (YYYY-MM-DD): 2025-02-01
Enter end date (YYYY-MM-DD): 2025-02-03
Booking successful! Your booking ID is 2.
```

3. View rental details and booking status.

Output of "3" option

```
Enter your choice(number 1-4): 3

Your Rental Details:
-----------------------------------------------------------------------------------------------------------------------
Booking ID  Car ID    Make          Model         Year      Mileage   Start Date          End Date            Total Cost     Status
-----------------------------------------------------------------------------------------------------------------------
1           1         audi          1             1         1.0       2025-02-01 00:00:00   2025-02-03 00:00:00   3.0            Approved
2           2         BMW           1             1         1.0       2025-02-01 00:00:00   2025-02-03 00:00:00   3.0            pending
-----------------------------------------------------------------------------------------------------------------------
```

4. Logout.

Choose “4” → Back to the main menu

```
=== Car Rental System ===
1. Login
2. Register
3. Exit
Enter your choice number(1-3):
```

Administrator:

Administrator main menu ↓

```
=== Admin Menu ===
1. Add a Cars
2. Remove a Car
3. Update car details
4. Browse Orders
5. Manage Orders
6. Logout

Enter your choice(number1-6):
```

1. Add, remove, and update car details.

1) Add a car

the information needed to input in “Add car details”

```
Enter your choice(number1-6): 1
Enter car make: BYD
Enter car model: 1
Enter car manufacture year: 1
Enter car mileage: 1
Enter car daily rent: 1
Car added successfully!
```

2) Remove a car

Enter the car id you want to remove and press “Enter” button to remove

```
Enter car id to remove: 3
Car removed successfully!
```

3) Update car details

the information needed to input in “Update car details”

```
Enter car id to update: 2

Current Car Details:
Make: 2
Model: BMW
Manufacture Year: 1
Mileage: 1
Availability: 1.0
Daily Rent: 1
Enter new make (press enter to keep current): BMW
Enter new model (press enter to keep current): 2
Enter new manufacture year (press enter to keep current): 2
Enter new mileage (press enter to keep current): 2
Enter new availability (Please enter number 1 or 0 for True/False,press enter to keep current): 1
Enter new daily rent (press enter to keep current): 3
Car details updated successfully!
```

2. Browse all orders or Browse unapproved orders.

```
=== Browse Orders ===
1. View All Orders
2. View Unapproved Orders
Enter your choice (1 or 2):
```

Output of “1” option “View all orders”:

```
All Orders:
-------------------------------------------------------------------------------------------------------------------------------
Booking ID  Car ID    Make          Model         Year      Mileage   Start Date           End Date             Total Cost     Status
-------------------------------------------------------------------------------------------------------------------------------
1           1         audi          1             1         1.0       2025-02-01 00:00:00  2025-02-03 00:00:00  3.0            Approved
2           2         BMW           2             2         2.0       2025-02-01 00:00:00  2025-02-03 00:00:00  3.0            pending
-------------------------------------------------------------------------------------------------------------------------------
```

Output of “2” option “View unapproved orders”:

```
Unapproved Orders:
-------------------------------------------------------------------------------------------------------------------------------
Booking ID  Car ID    Make          Model         Year      Mileage   Start Date           End Date             Total Cost     Status
-------------------------------------------------------------------------------------------------------------------------------
2           2         BMW           2             2         2.0       2025-02-01 00:00:00  2025-02-03 00:00:00  3.0            pending
-------------------------------------------------------------------------------------------------------------------------------
```

3. Approve or reject booking.

Choose option “1” to approve this order or Choose option “2” to reject

```
Enter your choice(number1-6): 5

Unapproved Orders:
-------------------------------------------------------------------------------------------------------------------------------------------
Booking ID  Car ID    Make           Model          Year      Mileage   Start Date            End Date              Total Cost     Status
-------------------------------------------------------------------------------------------------------------------------------------------
2           2         BMW            2              2         2.0       2025-02-01 00:00:00   2025-02-03 00:00:00   3.0            pending
-------------------------------------------------------------------------------------------------------------------------------------------
Enter the booking ID you want to process: 2

1. Approve Booking
2. Reject Booking
Enter your choice (1 or 2): 1
Booking 2 approved successfully.
```

4. Logout.