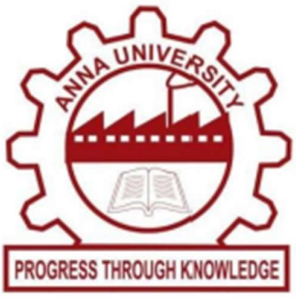
MADRAS INSTITUTE OF TECHNOLOGY

ANNA UNIVERSITY

CHENNAI – 600 044.



DEPARTMENT OF INFORMATION TECHNOLOGY

PROJECT REPORT

WEB TECHNOLOGIES

IT5501

Project Title: Car Rental Management System

Team Members:

Gayathri K 2022506032

Swetha sri 2022506027

Introduction

The **Car Rental Management System** is a comprehensive desktop-based application built using **Java Swing** for the user interface and **MySQL** as the backend database. The system is specifically designed for **administrative use** to efficiently manage car registrations, customer records, rental processes, return tracking, and user authentication. The system provides an intuitive user interface with essential features like **Add, Edit, Delete, and Search** for smooth system operations.

This system aims to provide an **all-in-one platform** for car rental businesses to manage their operations efficiently, track customer transactions, and maintain a detailed database of cars and rentals.

Objectives

- **Efficient Car Management**: Maintain a complete record of cars, their availability, and registration details.
- **Rental and Return Tracking**: Manage rental and return operations while ensuring accurate data logging.
- **Secure User Authentication**: Implement admin login to ensure only authorized personnel can access the system.
- **Data Integrity**: Ensure that all data related to customers, cars, rentals, and returns are stored securely in the MySQL database.
- **User-Friendly Interface**: Provide an interactive and simple interface for administrators using **Java Swing** for smooth navigation and operation.

Tech Stack

- **Frontend**:
    - **Java Swing**: For building the desktop user interface.
    - **NetBeans IDE**: Development environment to design the system and write Java code.
- **Backend**:
    - **MySQL**: A relational database used for managing the system's data storage and retrieval.
- **Additional Tools**:
    - **JDBC (Java Database Connectivity)**: Used for establishing a connection between Java and MySQL.
    - **MySQL Workbench** : For local MySQL server hosting and testing the database.

Features

1. **Admin Login**
    - Secure login system for administrators to manage cars, customers, rentals, and returns.
    - **Role-based access** ensures that only authorized personnel can access system features.
2. **Car Registration**
    - **Add, Edit, Delete, and View** car details such as car ID, model, brand, status, and availability.
    - Helps keep track of all registered cars in the system.
3. **Customer Management**
    - Store customer details such as name, contact information, and driver's license information.

    - **Add, Edit, Delete, and View** customer records to maintain an up-to-date customer database.
4. **Rental Management**
    - Record rental transactions, including customer name, car rented, rental date, and return date.
    - Calculate and store rental charges based on rental duration.
5. **Return Management**
    - Process vehicle returns, record return dates, and calculate any pending charges.
    - Track late returns and calculate additional fees accordingly.
6. **Dashboard**
    - Display an overview of active rentals, registered customers, and available cars.
7. **Data Validation and Alerts**
    - Provides alerts for **incorrect inputs** or **empty fields** during registration and data entry.
8. **Error Handling**
    - Handles SQL exceptions and displays user-friendly error messages for smoother operation.

System Design

Database Design

The MySQL database is structured using **five main tables** to ensure proper data management and relationships between various entities.

1. **Car Registration Table**
    - Stores details about all cars available for rent.
    - **Fields**: Car ID, Model, Brand, Registration No, Status (Available/Not Available), and Rent Price.

2. **Customer Table**

    - Stores information about customers registered in the system.
    - **Fields**: Customer ID, Name, Address, Contact No, Driver's License No, and Email.
3. **Rental Table**

    - Tracks all rental transactions between customers and cars.
    - **Fields**: Rental ID, Customer ID (Foreign Key), Car ID (Foreign Key), Rental Date, Return Date, Total Cost, and Status (Rented/Returned).
4. **Return Car Table**

    - Manages car return records and pending payments.
    - **Fields**: Return ID, Rental ID (Foreign Key), Actual Return Date, Late Fee, and Total Cost.
5. **Admin Login Table**

    - Stores admin login details to restrict access.
    - **Fields**: Admin ID, Username, and Password (hashed for security).

Implementation Details

Frontend (User Interface)

- **Language**: Java Swing
- **IDE**: NetBeans
- **Modules and Functions**:
    - **Login Page**: Admin authentication system.
    - **Dashboard**: Home screen that displays key metrics (cars available, customers, and rentals).
    - **Car Registration Form**: Interface to add, update, delete, and view car details.
    - **Customer Management Form**: Interface to manage customer information.

- **Rental Management Form**: Interface to process rentals, assign cars to customers, and track rental history.
- **Return Management Form**: Interface for processing car returns and calculating charges.

Backend (Database and Logic)

- **Language**: Java (Backend Logic), MySQL (Database)
- **Database**: MySQL (Using **JDBC** to connect the Java Swing application with the database)
- **Server**: MySQL Database Server (hosted on XAMPP)
- **CRUD Operations**:
  - **Create**: Add new cars, customers, rentals, and return entries.
  - **Read**: View and search car, customer, rental, and return data.
  - **Update**: Update existing records like car details, customer details, and rental information.
  - **Delete**: Remove cars, customers, and rental data as required.

Screenshots

1. Database tables

a. **Carregistration**
b. **Rental**
c. **Rentcar**
d. **Login**
e. **customer**

Query 1 | SQL File 3*

Limit to 1000 rows

```
43 • SHOW TABLES;
44
45 • DESCRIBE carregistration;
46 • DESCRIBE customer;
47 • DESCRIBE rental;
48 • DESCRIBE login;
49 • DESCRIBE returncar;
50 • select *from customer;
51 • select *from carregistration;
52 • select *from login;
53 • select *from returncar;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
car_no	varchar(255)	YES		NULL	
make	varchar(255)	YES		NULL	
model	varchar(255)	YES		NULL	
available	varchar(255)	YES		NULL	

Result Grid | Form Editor

Query 1 | SQL File 3*

Limit to 1000 rows

```
43 • SHOW TABLES;
44
45 • DESCRIBE carregistration;
46 • DESCRIBE customer;
47 • DESCRIBE rental;
48 • DESCRIBE login;
49 • DESCRIBE returncar;
50 • select *from customer;
51 • select *from carregistration;
52 • select *from login;
53 • select *from returncar;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
cust_id	varchar(255)	YES		NULL	
name	varchar(255)	YES		NULL	
address	tinytext	YES		NULL	
mobile	int	YES		NULL	

Result Grid | Form Editor

Result 1 | Result 2 | Result 3 | Result 4 | Result 5 | Read Only

Query 1 | SQL File 3*

Limit to 1000 rows

```
43  SHOW TABLES;
44
45  DESCRIBE carregistration;
46  DESCRIBE customer;
47  DESCRIBE rental;
48  DESCRIBE login;
49  DESCRIBE returncar;
50  select *from customer;
51  select *from carregistration;
52  select *from login;
53  select *from returncar;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
car_id	varchar(255)	YES		NULL	
cust_id	varchar(255)	YES		NULL	
fee	int	YES		NULL	
date	varchar(255)	YES		NULL	
due	varchar(255)	YES		NULL	

Result Grid | Form Editor

Result 1 | Result 2 | Result 3 | Result 4 | Result 5 | Read Only

Query 1 | SQL File 3*

Limit to 1000 rows

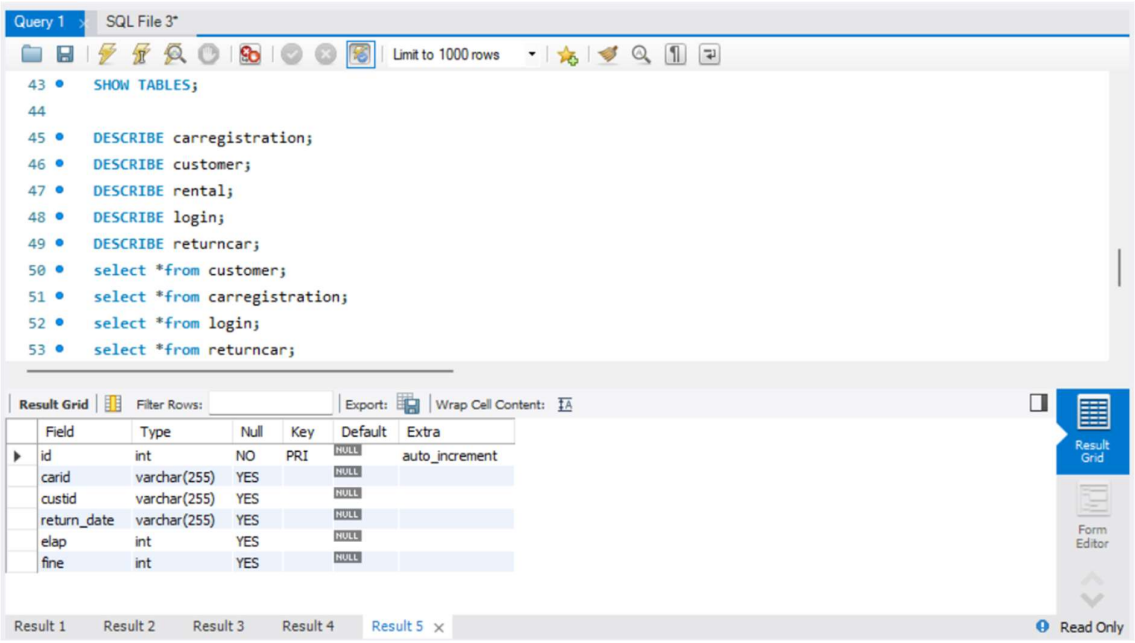
```
43  SHOW TABLES;
44
45  DESCRIBE carregistration;
46  DESCRIBE customer;
47  DESCRIBE rental;
48  DESCRIBE login;
49  DESCRIBE returncar;
50  select *from customer;
51  select *from carregistration;
52  select *from login;
53  select *from returncar;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
login_id	int	NO	PRI	NULL	auto_increment
username	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	

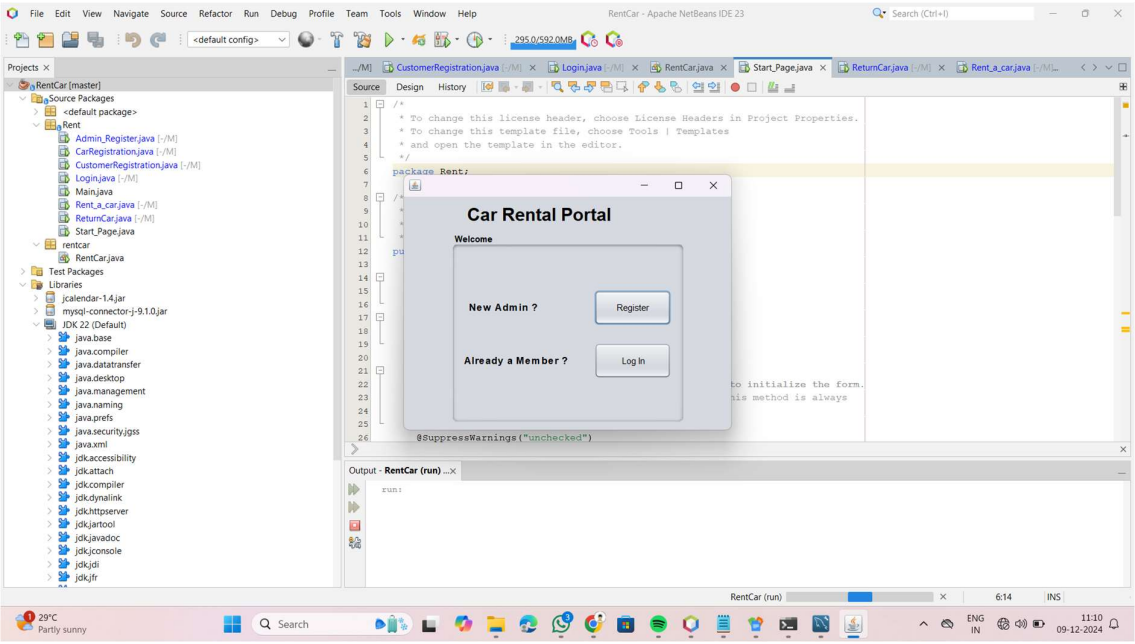
Result Grid | Form Editor

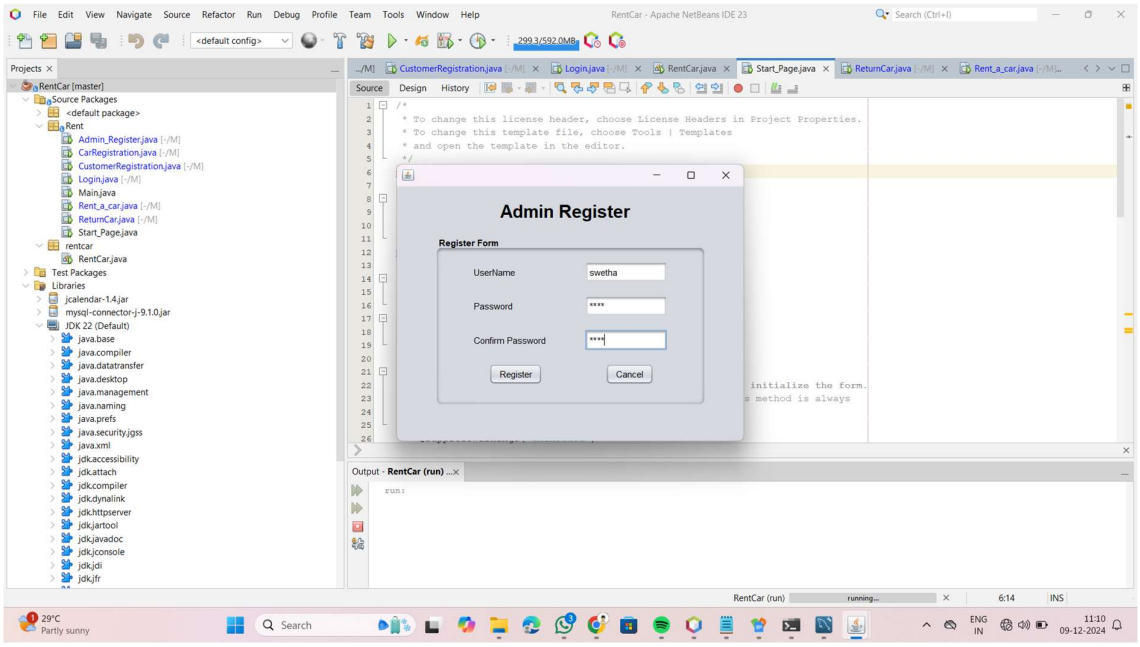
Result 1 | Result 2 | Result 3 | Result 4 | Result 5 | Read Only

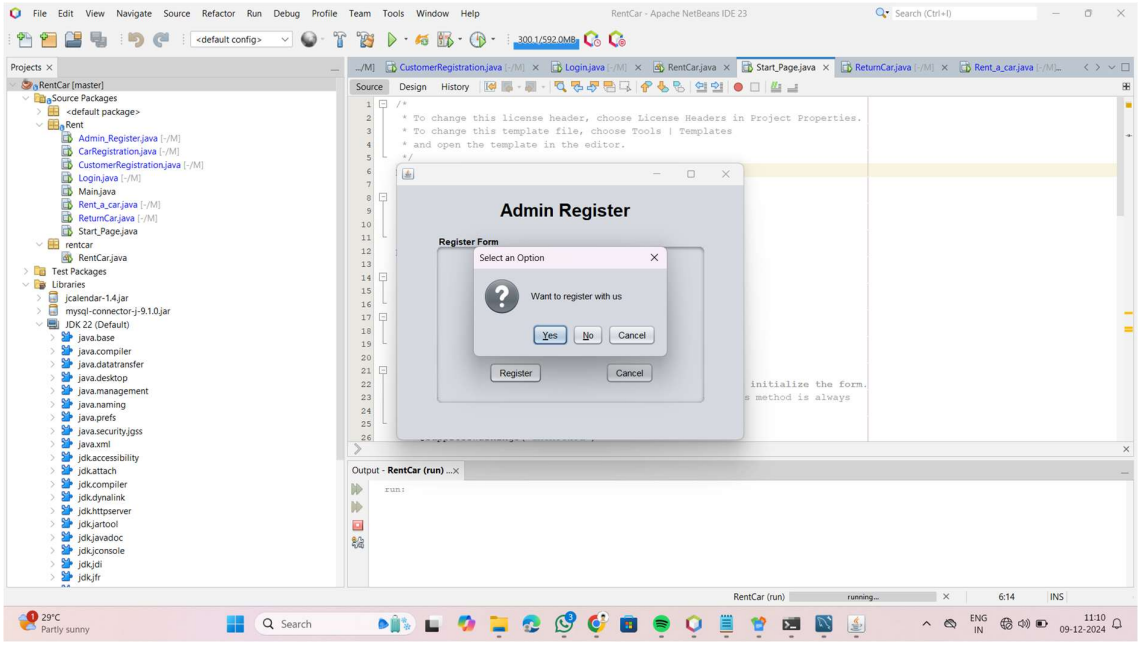


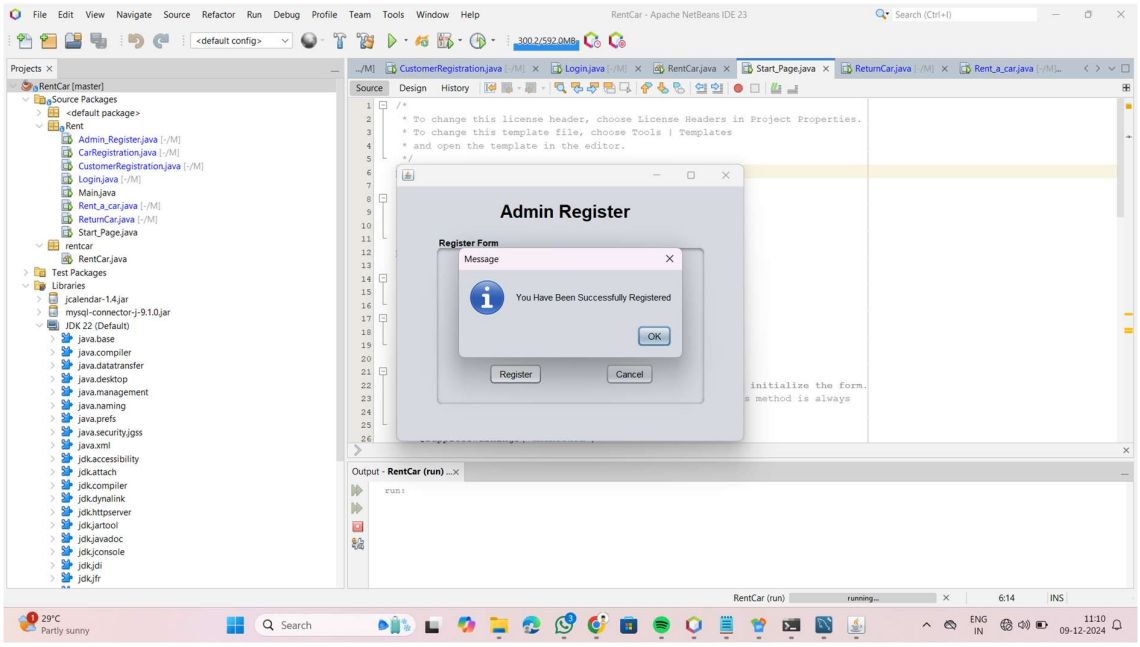
2. Register/Login Page

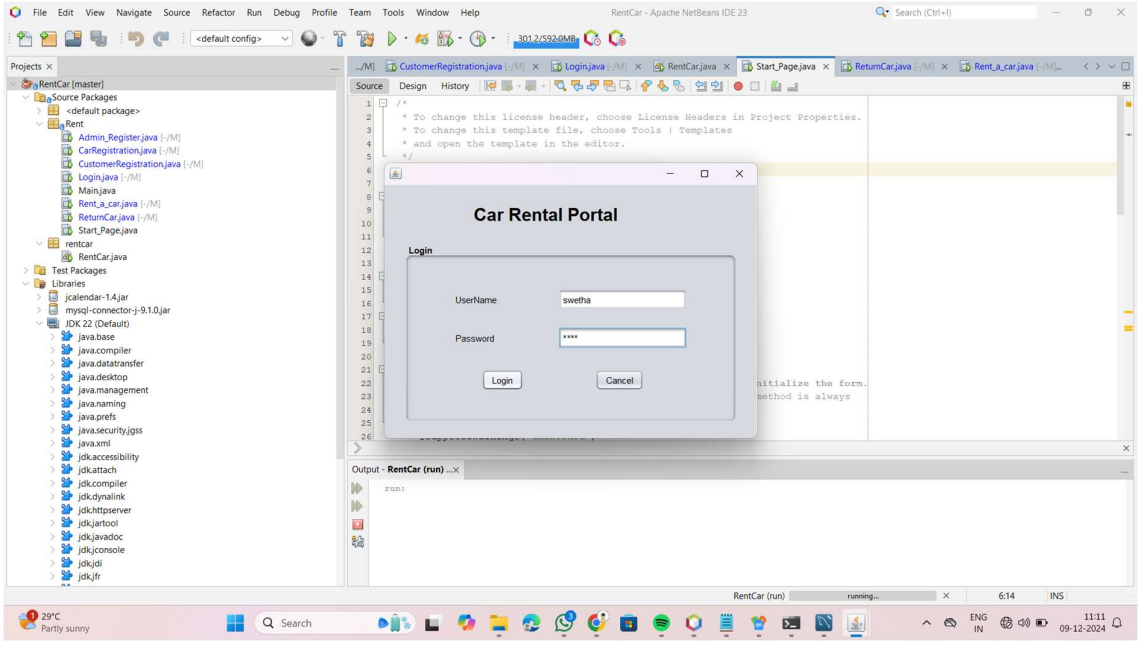
- Simple login page where admins enter their username and password.
- **Input validation** ensures the fields are not left blank.

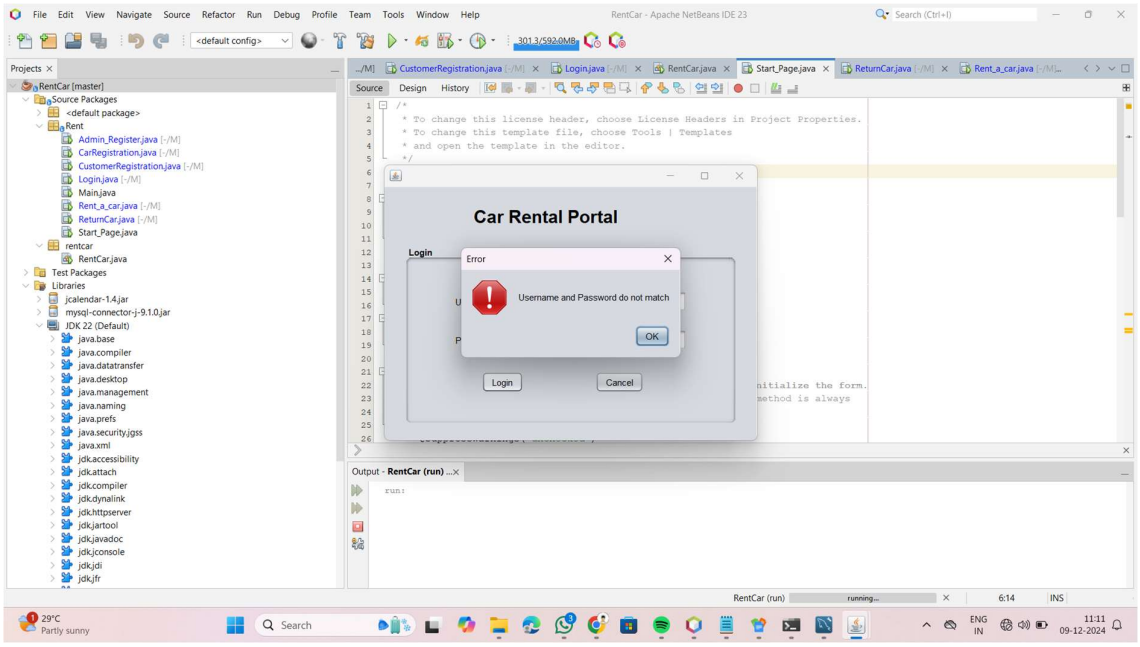



Admin Register
Register Form
UserName
swetha
Password
Confirm Password
Register
Cancel


Admin Register
Register Form
Select an Option
Want to register with us
Yes
No
Cancel
Register
Cancel

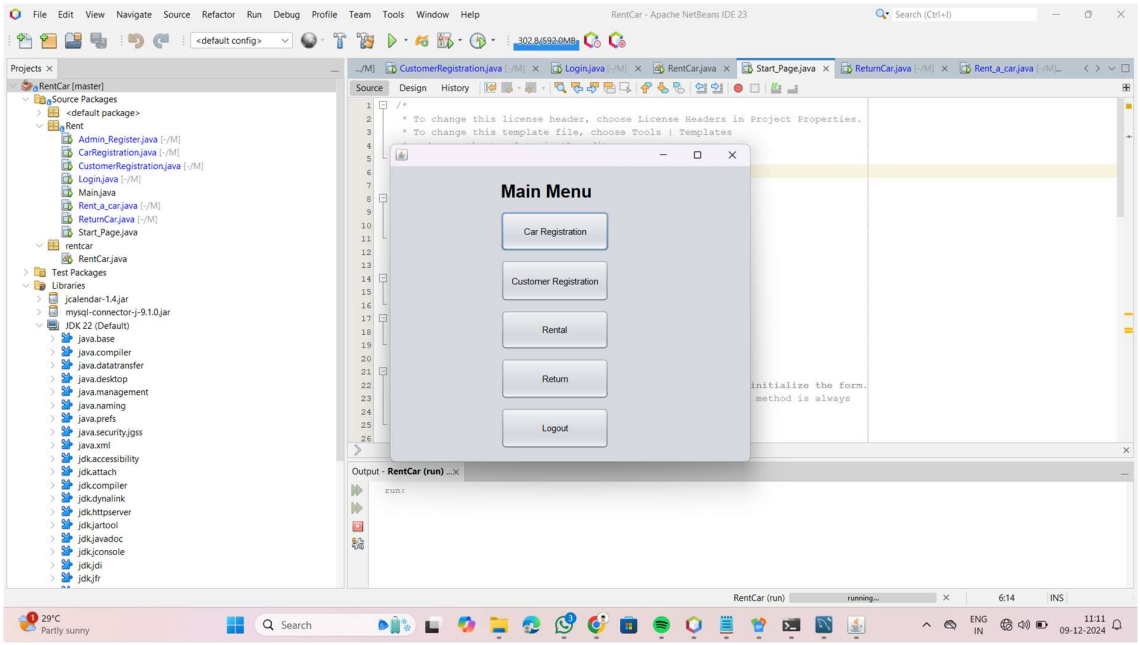

Admin Register
Register Form
Message
You Have Been Successfully Registered
OK
Register
Cancel


Car Rental Portal
Login
UserName
swetha
Password
Login
Cancel



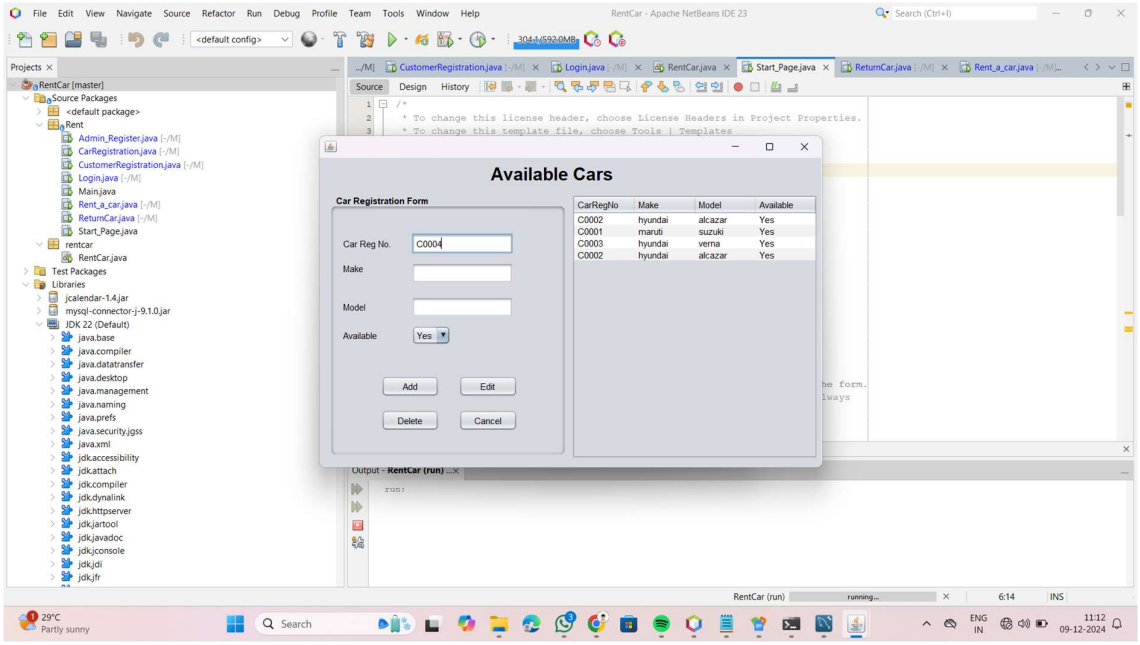
2. Dashboard

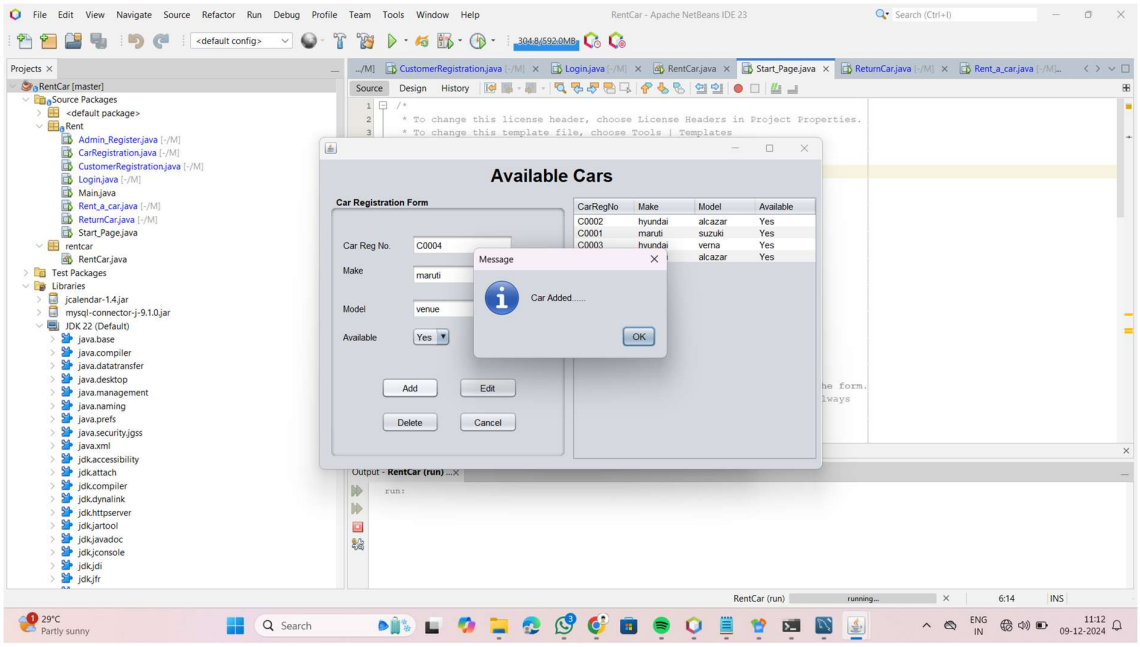
- Overview of Car registration, Customer registration, Rent a car, Return, Logout

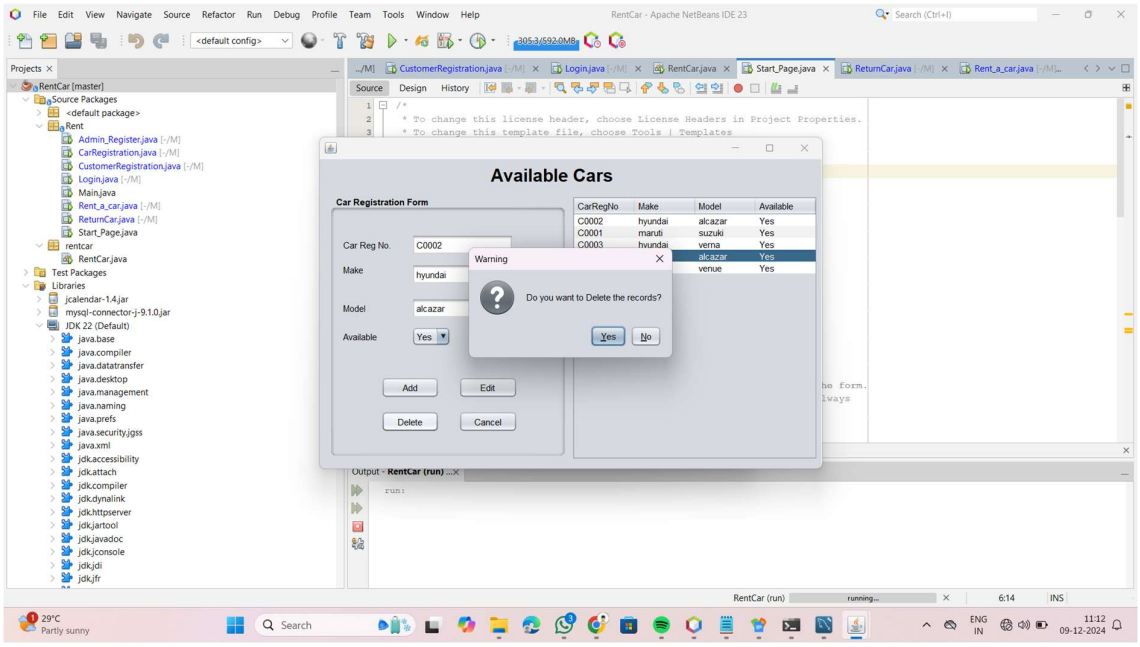


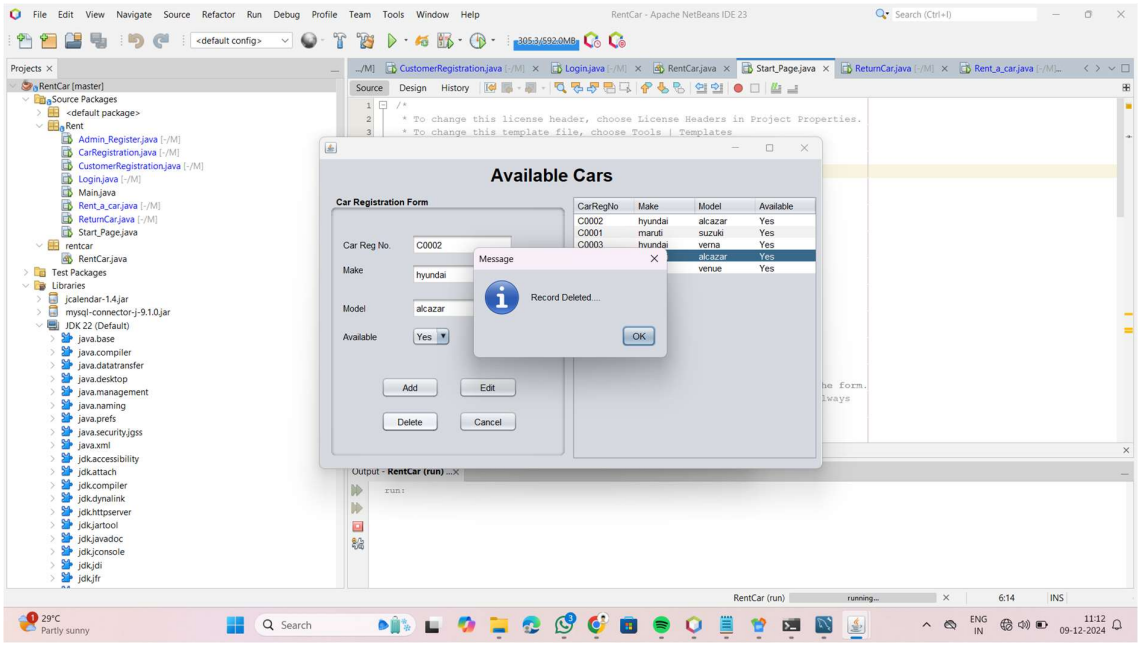
3. Car Registration

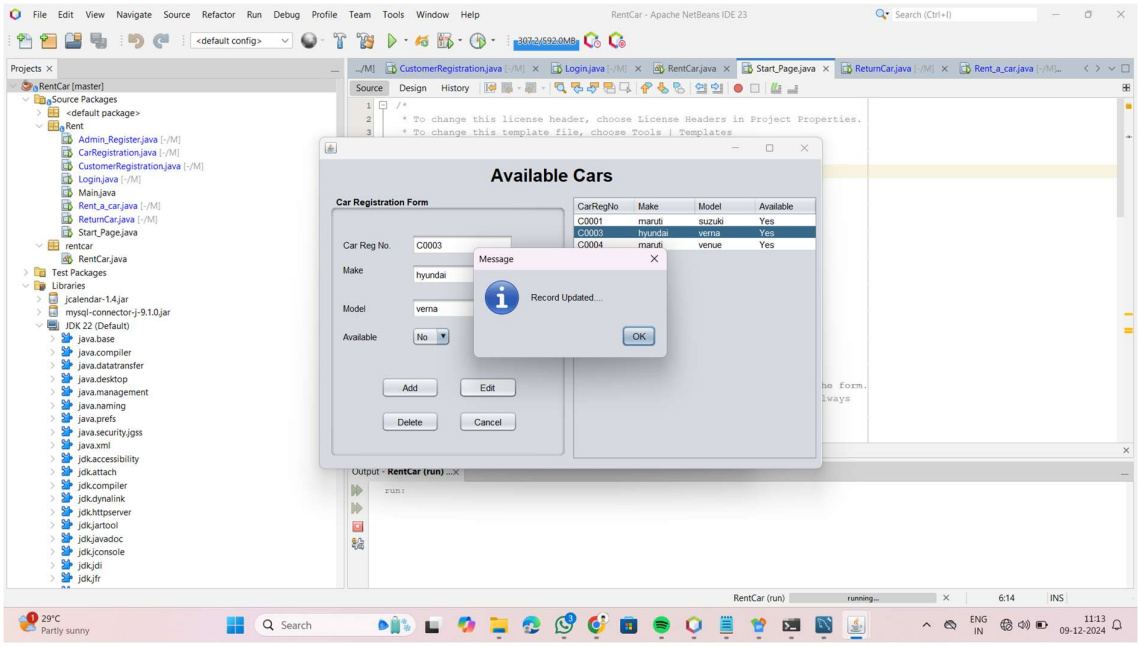
- **Add, Edit, Delete, and View** car details.
- Inputs include Car ID, Model, Brand, Rent Price, Status (Available/Not Available).

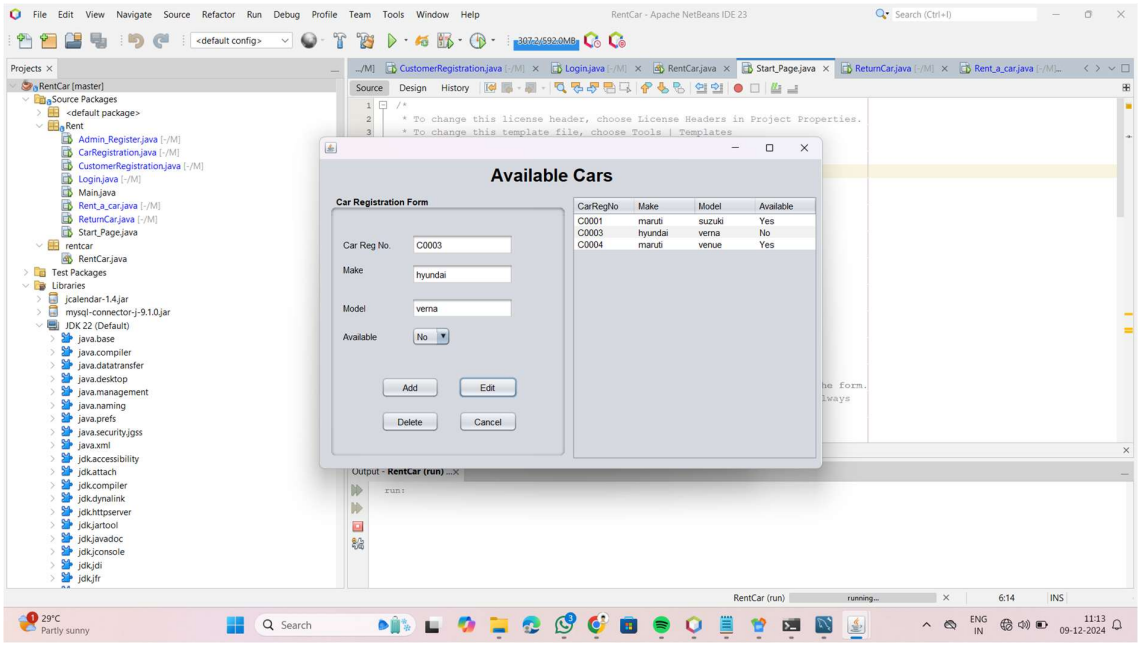





Available Cars
Car Registration Form
Car Reg No.
C0002
Make
hyundai
Model
alcazar
Available
Yes
Add
Edit
Delete
Cancel
CarRegNo
Make
Model
Available
C0002 hyundai alcazar Yes
C0001 maruti suzuki Yes
C0003 hyundai verna Yes
alcazar Yes
venue Yes
Warning
Do you want to Delete the records?
Yes
No

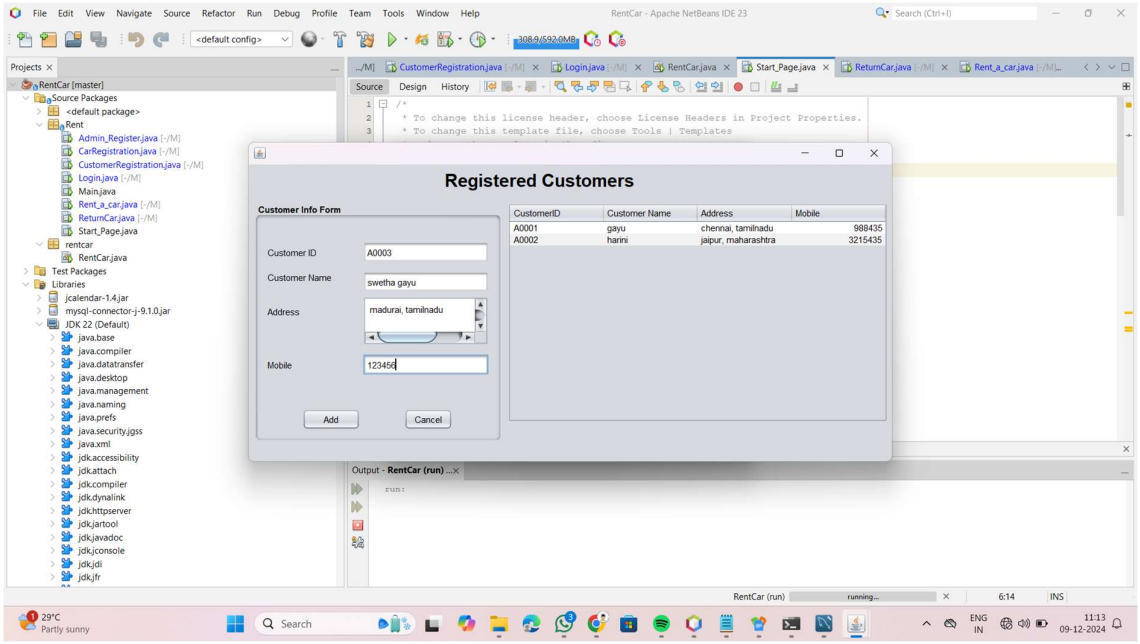

Available Cars
Car Registration Form
Car Reg No.
C0002
Make
hyundai
Model
alcazar
Available
Yes
Add
Edit
Delete
Cancel
CarRegNo
Make
Model
Available
C0002 hyundai alcazar Yes
C0001 maruti suzuki Yes
C0003 hyundai verna Yes
alcazar Yes
venue Yes
Message
Record Deleted....
OK

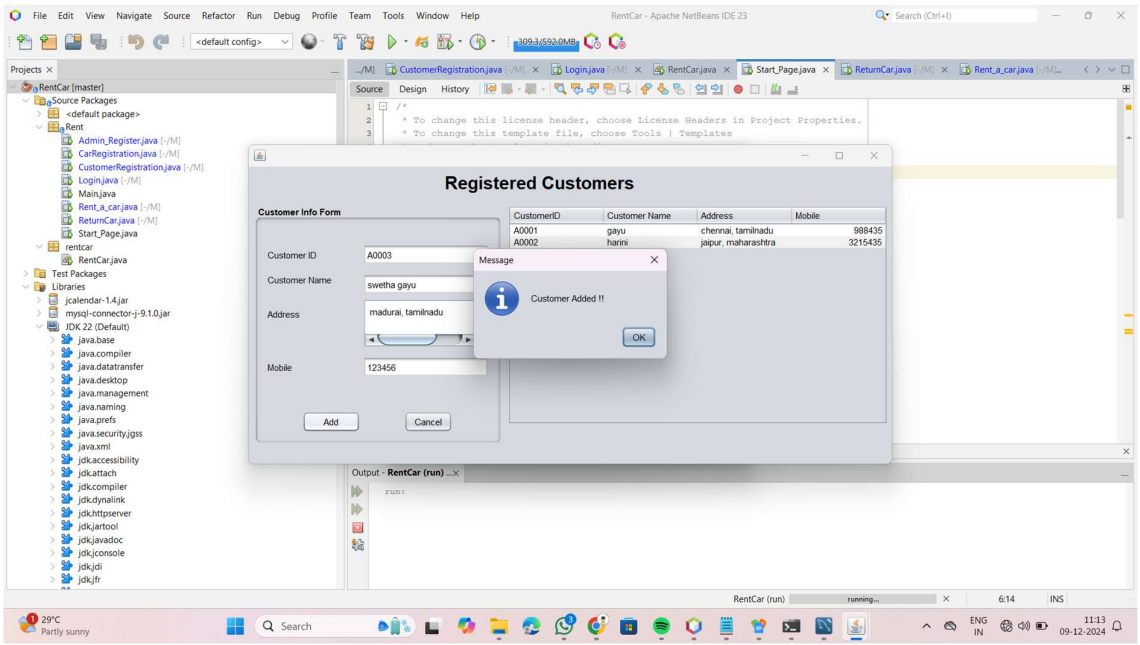

Available Cars
Car Registration Form
Car Reg No.
C0003
Make
hyundai
Model
verna
Available
No
Add
Edit
Delete
Cancel
CarRegNo
Make
Model
Available
C0001
maruti
suzuki
Yes
C0003
hyundai
verna
Yes
C0004
maruti
venue
Yes
Message
Record Updated....
OK

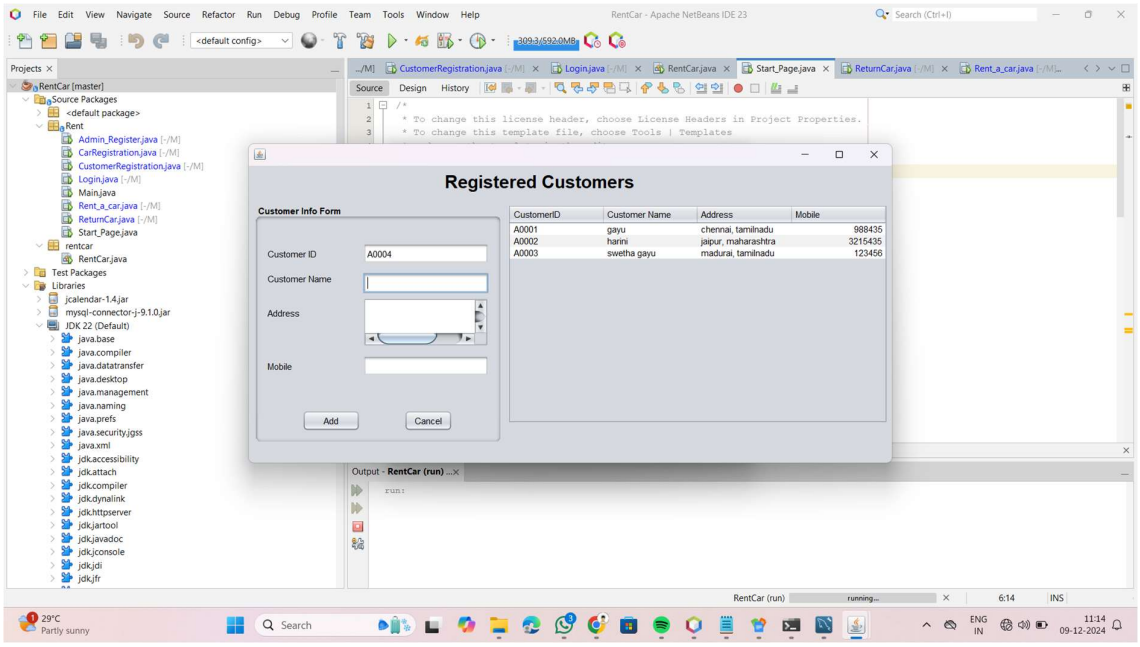

Available Cars
Car Registration Form
Car Reg No.
C0003
Make
hyundai
Model
verna
Available
No
Add
Edit
Delete
Cancel
CarRegNo
Make
Model
Available
C0001
maruti
suzuki
Yes
C0003
hyundai
verna
No
C0004
maruti
venue
Yes

4. Customer Management

- Manage customer details, including Cust id, Name, Contact, and Address

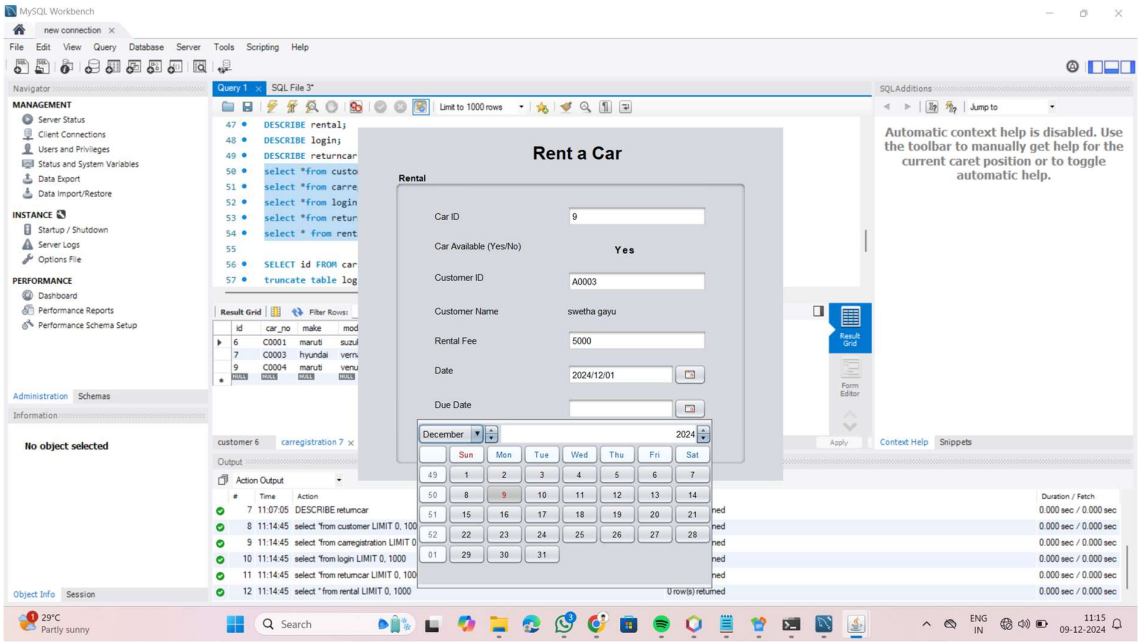


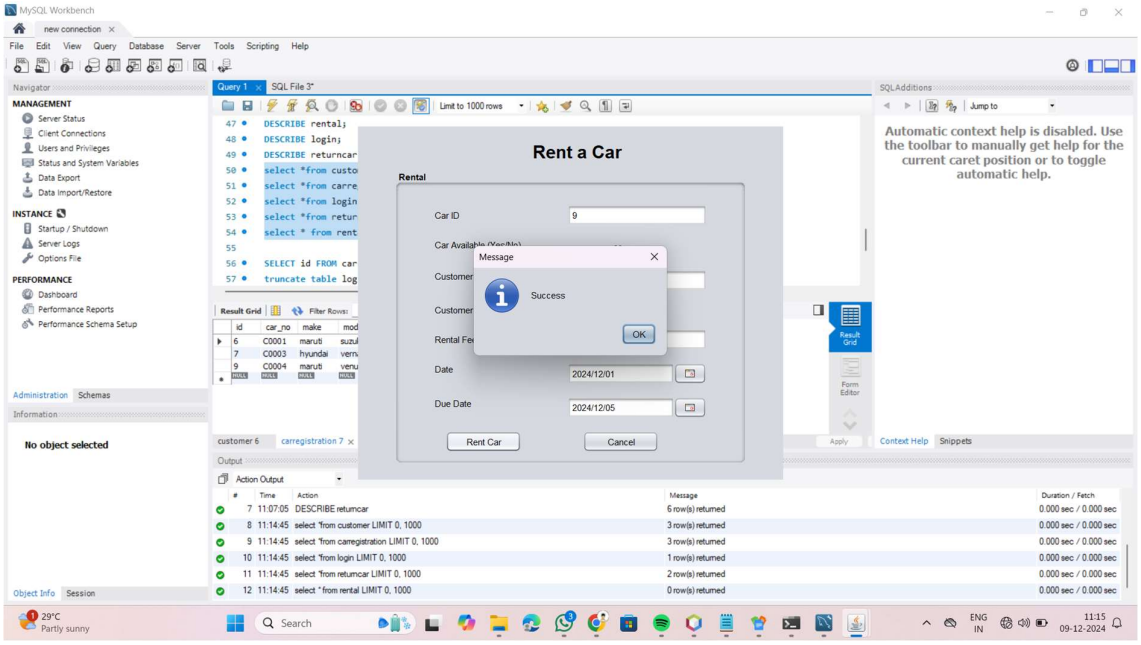




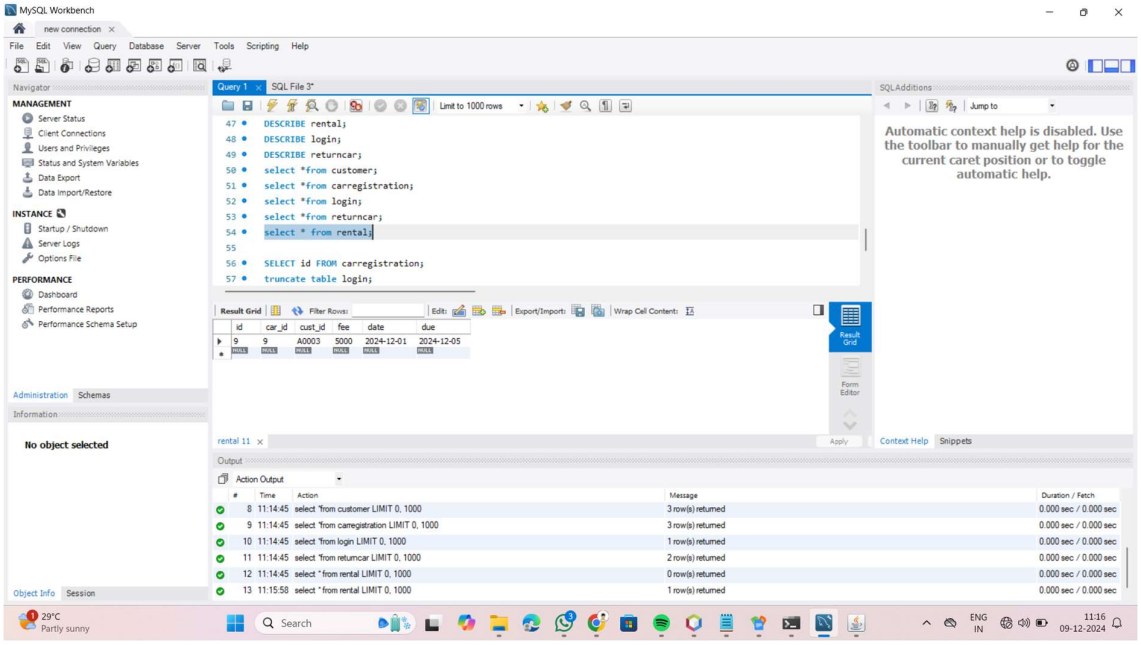
5. Rental Management

- Process a new rental, select the car, customer, rent price and rental from and to date.
- Calculate the rental amount dynamically based on the number of days.

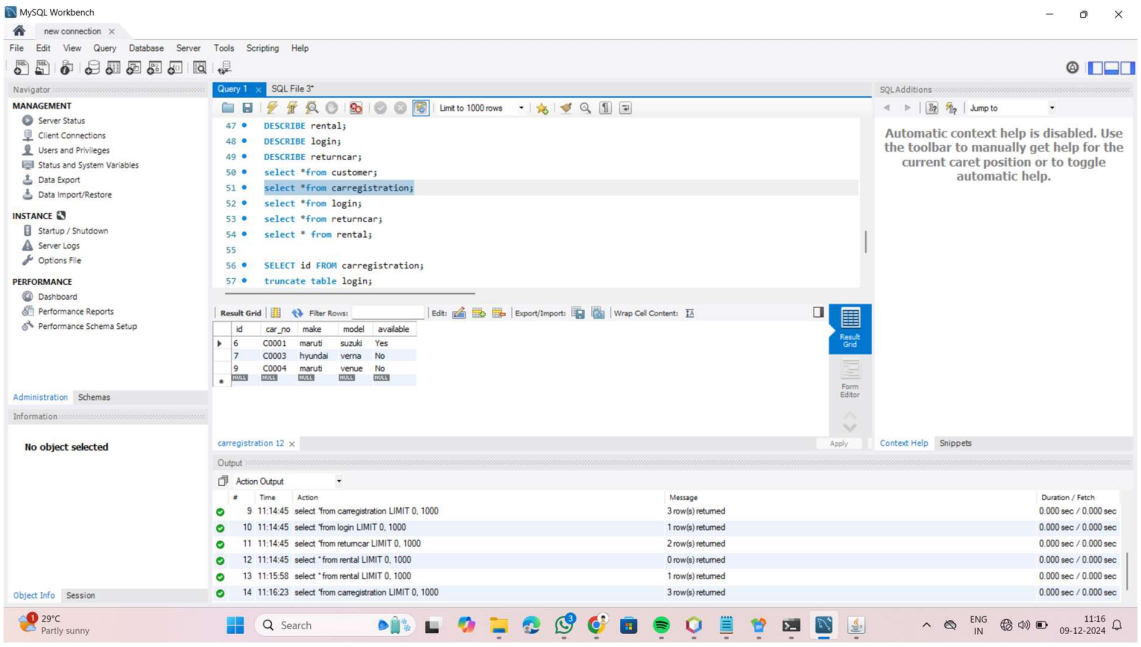




The rental table gets updated

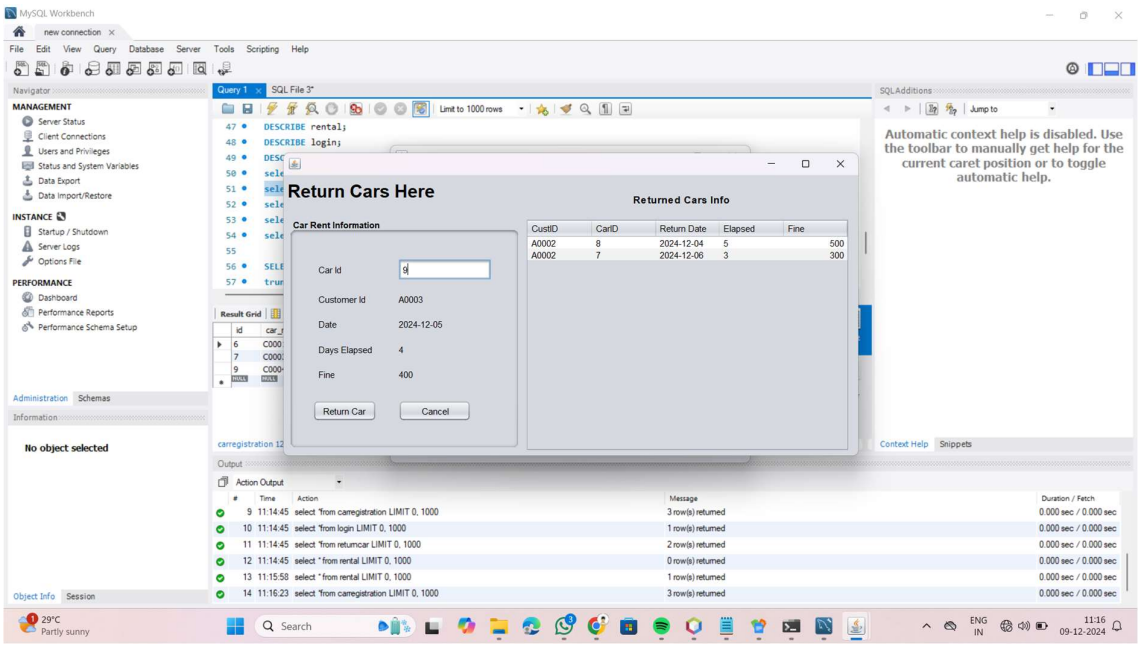


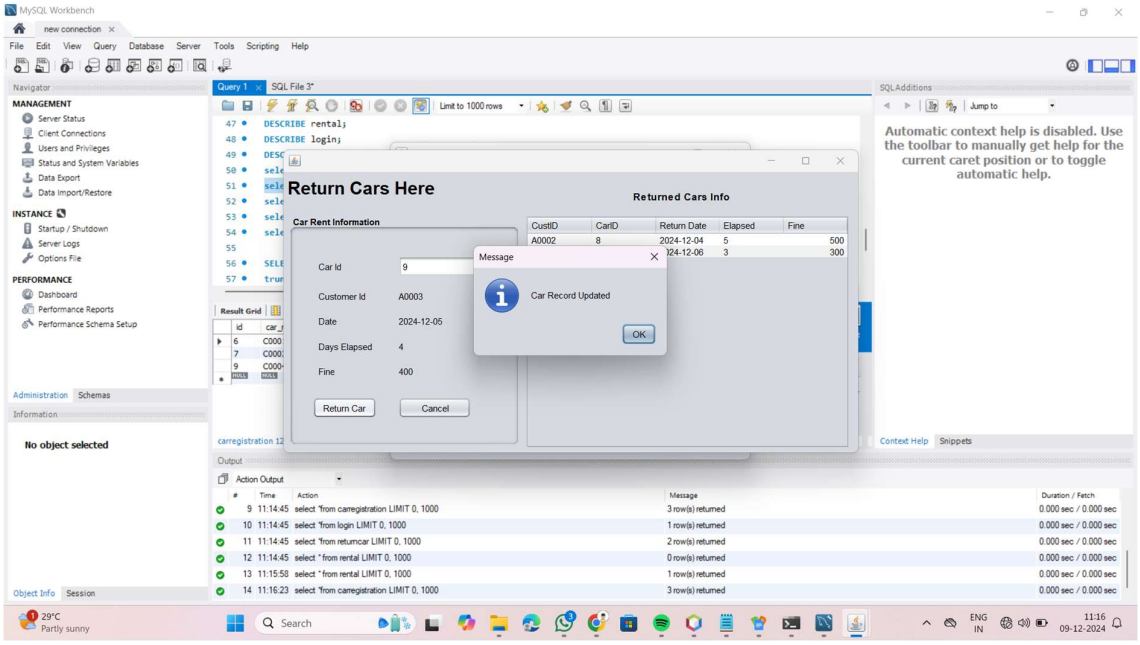
After renting a car, The respective car availability turns to “No”

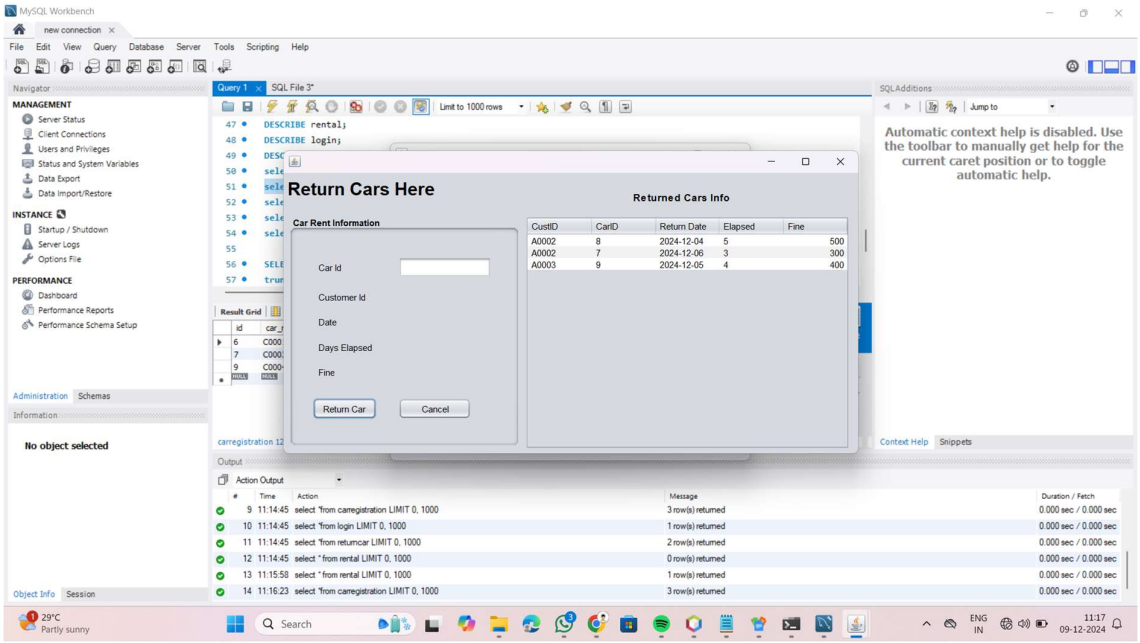


6. Return Car

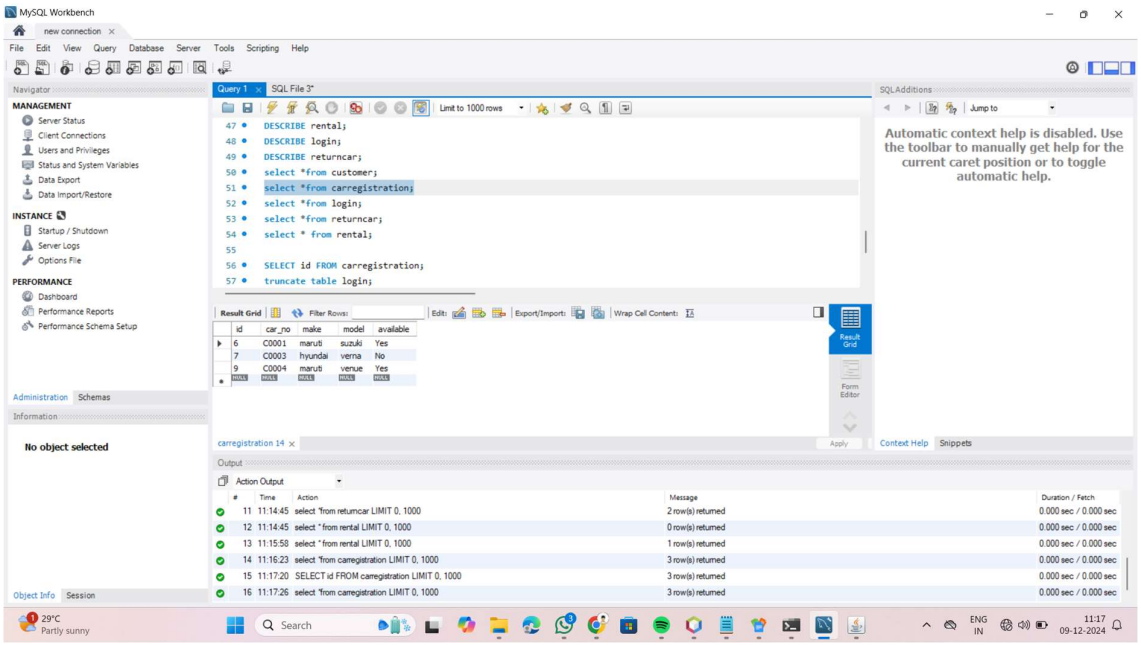
- Process car returns, calculate late fees, and update the rental record status.



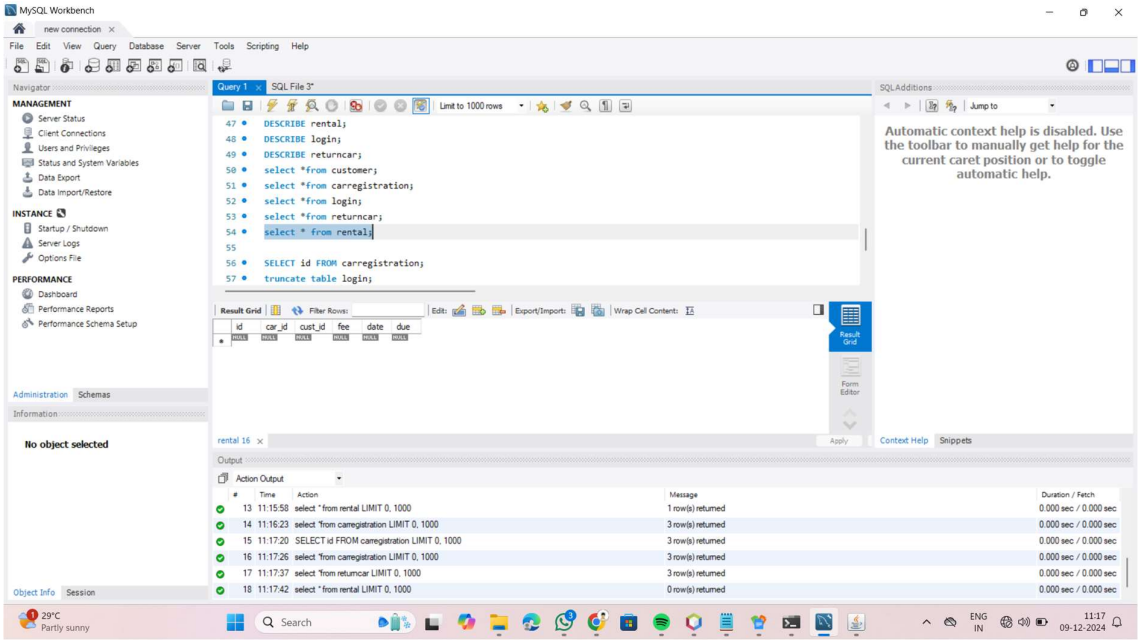

MySQL Workbench
Return Cars Here
Returned Cars Info
Car Rent Information
Car Id
9
Customer Id
A0003
Date
2024-12-05
Days Elapsed
4
Fine
400
Return Car
Cancel
Message
Car Record Updated
OK
CustID
CarID
Return Date
Elapsed
Fine
A0002
8
2024-12-04
5
500
A0002
7
2024-12-06
3
300


MySQL Workbench
Return Cars Here
Returned Cars Info
Car Rent Information
Car Id
Customer Id
Date
Days Elapsed
Fine
Return Car
Cancel
CustID
CarID
Return Date
Elapsed
Fine
A0002
8
2024-12-04
5
500
A0002
7
2024-12-06
3
300
A0003
9
2024-12-05
4
400

After returning a car, the availability changes to “Yes”



The rental records get deleted once returned



Conclusion

The **Car Rental Management System** streamlines the rental process by automating key activities like car registration, customer management, rentals, and returns. It ensures a **secure, user-friendly, and efficient system** for administrators. With **data validation, real-time updates, and error handling**, the system guarantees **data integrity** and **operational efficiency**. Future enhancements aim to expand the system's reach to customers, enhance the security of login processes, and introduce advanced analytics.

This project demonstrates the **integration of Java Swing and MySQL** to develop a full-fledged desktop application. It showcases the importance of **data handling, secure login systems, and modular UI design** for administrative control.