

1. Flight Reservation System

Description

Build a console-based flight reservation system where users can search for flights, book tickets, and manage their bookings. The system should store flight and booking details in a database.

Features

- Add flights to the database (flight number, source, destination, departure time, seats available).
- Allow users to search for flights based on source, destination, and date.
- Book tickets by selecting a flight and providing passenger details.
- Cancel bookings and update seat availability.
- View all bookings or search for a specific booking.

OOP Concepts

- Create Flight, Passenger, and Booking classes.
- Use a FlightDAO class to manage flight-related database operations and a BookingDAO class for booking-related operations.

JDBC

Store flight and booking details in a database and retrieve them using SQL queries.

Unique Aspect

Simulate real-time seat availability and booking constraints.

2. Online Quiz Platform

Description

Develop a console-based online quiz platform where users can take quizzes on various topics. The system should store quiz questions, user scores, and performance history in a database.

Features

- Add quizzes and questions to the database (question text, options, correct answer).
- Allow users to select a quiz and answer questions.
- Calculate and display the user's score at the end of the quiz.
- Store user performance history (username, quiz taken, score).
- View leaderboard for a specific quiz.

OOP Concepts

- Create Quiz, Question, and User classes.
- Use a QuizDAO class to manage quiz-related database operations and a UserDAO class for user-related operations.

JDBC

Store quiz questions, user details, and performance history in a database.

Unique Aspect

Dynamic quiz generation and real-time score tracking.

3. Car Rental Management System

Description

Create a console application to manage a car rental service. Users can rent cars, return them, and view available cars. The system should store car and rental details in a database.

Features

- Add cars to the database (model, brand, year, rental price, availability).
- Allow users to rent a car by selecting from available cars and providing rental details (start date, end date).
- Calculate rental cost based on the number of days.
- Return a car and update its availability.
- View all cars or search for a specific car.

OOP Concepts

- Create Car, Rental, and Customer classes.
- Use a CarDAO class to manage car-related database operations and a RentalDAO class for rental-related operations.

JDBC

Store car and rental details in a database and retrieve them using SQL queries.

Unique Aspect

Simulate real-time car availability and rental cost calculations.