Bus Ticket Reservation System (Java + SQL)

```
// Java + SQL: Bus Ticket Reservation System (Simple CLI-based project)

// 1. Java Main Class (BusReservationSystem.java)
import java.sql.*;
import java.util.*;

public class BusReservationSystem {
    static final String DB_URL = "jdbc:mysql://localhost:3306/bus_system";
    static final String USER = "root";
    static final String PASS = "your_password"; // Replace with your MySQL password

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        try (Connection conn = DriverManager.getConnection(DB_URL, USER, PASS)) {
            while (true) {
                System.out.println("\n=== BUS TICKET RESERVATION SYSTEM ===");
                System.out.println("1. View Buses");
                System.out.println("2. Book Ticket");
                System.out.println("3. View Bookings");
                System.out.println("4. Exit");
                System.out.print("Enter your choice: ");
                int choice = scanner.nextInt();

                switch (choice) {
                    case 1:
                        viewBuses(conn);
                        break;
                    case 2:
                        bookTicket(conn, scanner);
                        break;
                    case 3:
                        viewBookings(conn);
                        break;
                    case 4:
                        System.out.println("Thank you for using the system.");
                        return;
                    default:
                        System.out.println("Invalid choice!");
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    static void viewBuses(Connection conn) throws SQLException {
        Statement stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT * FROM buses");
        System.out.println("\nBus List:");
        while (rs.next()) {
```

```
                    System.out.println("Bus ID: " + rs.getInt("bus_id") + ", Route: " +
rs.getString("route") + ", Seats Available: " + rs.getInt("seats_available"));
        }
    }

    static void bookTicket(Connection conn, Scanner scanner) throws SQLException {
        System.out.print("Enter Bus ID to book: ");
        int busId = scanner.nextInt();
        scanner.nextLine();
        System.out.print("Enter your name: ");
        String name = scanner.nextLine();

          PreparedStatement checkSeats = conn.prepareStatement("SELECT seats_available FROM buses
WHERE bus_id = ?");
        checkSeats.setInt(1, busId);
        ResultSet rs = checkSeats.executeQuery();

        if (rs.next() && rs.getInt("seats_available") > 0) {
              PreparedStatement book = conn.prepareStatement("INSERT INTO bookings (name, bus_id)
VALUES (?, ?)");
            book.setString(1, name);
            book.setInt(2, busId);
            book.executeUpdate();

                    PreparedStatement updateSeats = conn.prepareStatement("UPDATE buses SET
seats_available = seats_available - 1 WHERE bus_id = ?");
            updateSeats.setInt(1, busId);
            updateSeats.executeUpdate();

            System.out.println("Ticket booked successfully for " + name);
        } else {
            System.out.println("No seats available on selected bus.");
        }
    }

    static void viewBookings(Connection conn) throws SQLException {
        Statement stmt = conn.createStatement();
           ResultSet rs = stmt.executeQuery("SELECT bookings.booking_id, name, buses.route FROM
bookings JOIN buses ON bookings.bus_id = buses.bus_id");
        System.out.println("\nBooking List:");
        while (rs.next()) {
                  System.out.println("Booking ID: " + rs.getInt("booking_id") + ", Name: " +
rs.getString("name") + ", Route: " + rs.getString("route"));
        }
    }
}

/*
SQL SETUP:
Run these SQL queries in your MySQL database to create the required tables:

CREATE DATABASE bus_system;
USE bus_system;
```

```
CREATE TABLE buses (
    bus_id INT AUTO_INCREMENT PRIMARY KEY,
    route VARCHAR(100),
    seats_available INT
);

CREATE TABLE bookings (
    booking_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    bus_id INT,
    FOREIGN KEY (bus_id) REFERENCES buses(bus_id)
);

-- Insert example buses:
INSERT INTO buses (route, seats_available) VALUES
('Chennai to Madurai', 40),
('Coimbatore to Chennai', 35),
('Salem to Trichy', 30);
*/
```