

File Edit View

```
import java.sql.*;
import java.util.Scanner;

public class AirlineTrafficManagementSystem {
    // Database credentials and URL
    private static final String URL = "jdbc:mysql://localhost:3306/AirlineDB5";
    private static final String USER = "root"; // Replace with your MySQL username
    private static final String PASSWORD = "kovarthini@2005"; // Replace with your MySQL password

    // Establish database connection
    private Connection connect() throws SQLException {
        return DriverManager.getConnection(URL, USER, PASSWORD);
    }

    // Add a new flight
    public void addFlight(String flightNumber, String origin, String destination, int availableSeats) {
        String query = "INSERT INTO flights4 (flight_number, origin, destination, available_seats) VALUES (?, ?, ?, ?)";
        try (Connection conn = connect(); PreparedStatement stmt = conn.prepareStatement(query)) {
            stmt.setString(1, flightNumber);
            stmt.setString(2, origin);
            stmt.setString(3, destination);
            stmt.setInt(4, availableSeats);
            stmt.executeUpdate();
            System.out.println("Flight added successfully.");
        } catch (SQLException e) {
            System.err.println("Error adding flight: " + e.getMessage());
        }
    }

    // View all flights
    public void viewFlights() {
        String query = "SELECT * FROM flights4";
        try (Connection conn = connect(); PreparedStatement stmt = conn.prepareStatement(query)) {
            ResultSet rs = stmt.executeQuery();
            while (rs.next()) {
                System.out.println("Flight Number: " + rs.getString("flight_number"));
                System.out.println("Origin: " + rs.getString("origin"));
                System.out.println("Destination: " + rs.getString("destination"));
                System.out.println("Available Seats: " + rs.getInt("available_seats"));
                System.out.println("-----");
            }
        }
    }
}
```

Ln 35, Col 32 7,152 characters

100%

Windows (CRLF)

UTF-8



Search

ENG
IN01:18 PM
19-11-2024

```
        System.out.println("Flight Number: " + rs.getString("flight_number"));
        System.out.println("Origin: " + rs.getString("origin"));
        System.out.println("Destination: " + rs.getString("destination"));
        System.out.println("Available Seats: " + rs.getInt("available_seats"));
        System.out.println("-----");
    }
} catch (SQLException e) {
    System.err.println("Error viewing flights: " + e.getMessage());
}
}

// Book a ticket for a passenger
public void bookTicket(String passengerName, String flightNumber) {
    String selectQuery = "SELECT available_seats FROM flights4 WHERE flight_number = ?";
    String updateQuery = "UPDATE flights4 SET available_seats = available_seats - 1 WHERE flight_number = ?";
    String insertQuery = "INSERT INTO bookings4 (passenger_name, flight_number) VALUES (?, ?)";

    try (Connection conn = connect();
        PreparedStatement selectStmt = conn.prepareStatement(selectQuery);
        PreparedStatement updateStmt = conn.prepareStatement(updateQuery);
        PreparedStatement insertStmt = conn.prepareStatement(insertQuery)) {

        // Check available seats
        selectStmt.setString(1, flightNumber);
        ResultSet rs = selectStmt.executeQuery();
        if (rs.next()) {
            int availableSeats = rs.getInt("available_seats");
            if (availableSeats > 0) {
                // Update seat count
                updateStmt.setString(1, flightNumber);
                updateStmt.executeUpdate();

                // Insert booking
                insertStmt.setString(1, passengerName);
                insertStmt.setString(2, flightNumber);
                insertStmt.executeUpdate();

                System.out.println("Booking successful!");
            } else {
                System.out.println("No available seats on this flight.");
            }
        }
    }
}
```





File Edit View

```
int choice;
while (true) {
    if (scanner.hasNextInt()) {
        choice = scanner.nextInt();
        scanner.nextLine(); // Consume newline
        break;
    } else {
        System.out.println("Invalid input. Please enter a valid number.");
        scanner.nextLine(); // Clear invalid input
    }
}

switch (choice) {
    case 1:
        System.out.print("Enter flight number: ");
        String flightNumber = scanner.nextLine();
        System.out.print("Enter origin: ");
        String origin = scanner.nextLine();
        System.out.print("Enter destination: ");
        String destination = scanner.nextLine();
        System.out.print("Enter available seats: ");
        int seats = scanner.nextInt();
        system.addFlight(flightNumber, origin, destination, seats);
        break;

    case 2:
        system.viewFlights();
        break;

    case 3:
        System.out.print("Enter passenger name: ");
        String passengerName = scanner.nextLine();
        System.out.print("Enter flight number: ");
        String bookFlightNumber = scanner.nextLine();
        system.bookTicket(passengerName, bookFlightNumber);
        break;

    case 4:
        system.viewBookings();
        break;
}
```



Search



```
switch (choice) {  
    case 1:  
        System.out.print("Enter flight number: ");  
        String flightNumber = scanner.nextLine();  
        System.out.print("Enter origin: ");  
        String origin = scanner.nextLine();  
        System.out.print("Enter destination: ");  
        String destination = scanner.nextLine();  
        System.out.print("Enter available seats: ");  
        int seats = scanner.nextInt();  
        system.addFlight(flightNumber, origin, destination, seats);  
        break;  
  
    case 2:  
        system.viewFlights();  
        break;  
  
    case 3:  
        System.out.print("Enter passenger name: ");  
        String passengerName = scanner.nextLine();  
        System.out.print("Enter flight number: ");  
        String bookFlightNumber = scanner.nextLine();  
        system.bookTicket(passengerName, bookFlightNumber);  
        break;  
  
    case 4:  
        system.viewBookings();  
        break;  
  
    case 5:  
        System.out.println("Exiting... Goodbye!");  
        scanner.close();  
        return;  
  
    default:  
        System.out.println("Invalid choice. Please try again.");  
}  
}  
}
```



Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

PS C:\java> javac AirlineTrafficManagementSystem.java

PS C:\java> java -cp ".;C:\Users\Megala\Downloads\mysql-connector-j-9.1.0\mysql-connector-j-9.1.0\mysql-connector-j-9.1.0.jar" AirlineTrafficManagementSystem

--- Airline Traffic Management System ---

1. Add Flight

2. View Flights

3. Book Ticket

4. View Bookings

5. Exit

Enter your choice: 1

Enter flight number: DD101

Enter origin: US

Enter destination: UK

Enter available seats: 10

Flight added successfully.

--- Airline Traffic Management System ---

1. Add Flight
2. View Flights
3. Book Ticket
4. View Bookings
5. Exit

Enter your choice: 2

Flight Number: AA101

Origin: india

Destination: paris

Available Seats: 4

Flight Number: AZ202

Origin: london

Destination: newyork

Available Seats: 5

Flight Number: BA202

Origin: london

Destination: new york

Available Seats: 5

Flight Number: CA101

Origin: russia

Destination: britain

Available Seats: 2

Flight Number: DD101

Origin: US

Destination: UK

Available Seats: 10

Flight Number: FF103

Origin: India

Destination: Paris

Available Seats: 14

--- Airline Traffic Management System ---

1. Add Flight
2. View Flights
3. Book Ticket
4. View Bookings
5. Exit

Enter your choice: 3

Enter passenger name: Giri

Enter flight number: DD101

Booking successful!

--- Airline Traffic Management System ---

1. Add Flight
2. View Flights
3. Book Ticket
4. View Bookings
5. Exit

Enter your choice: 4

Booking ID: 1

Passenger Name: abc

Flight Number: BA202

Booking ID: 2

Passenger Name: kova

Flight Number: AA101

Booking ID: 3

Passenger Name: keerthi

Flight Number: CA101

Booking ID: 4

Passenger Name: def

Flight Number: BA202

Booking ID: 5

Passenger Name: def

Flight Number: BA202

--- Airline Traffic Management System ---

1. Add Flight
2. View Flights
3. Book Ticket
4. View Bookings
5. Exit

Enter your choice: 5

Exiting... Goodbye!

PS C:\java> |