

Name : Chinmay Makrand Pimpalgaonkar

Student ID : WDGET2024026

Email ID: chinmaymp007@gmail.com

JDBC Airline Application

Github URL : <https://github.com/chinmay-2002/AirlineApplicationJDBC.git>

Passenger.java

```
package com.data;

public class Passenger {
    private long adhar;
    private String name;
    private String dob;
    private String sex;
    private String address;
    private int flightid;

    public Passenger(long adhar, String name, String dob, String sex, String address, int
flightid) {
        super();
        this.adhar = adhar;
        this.name = name;
        this.dob = dob;
        this.sex = sex;
        this.address = address;
        this.flightid = flightid;
    }

    public long getAdhar() {
        return adhar;
    }

    public void setAdhar(long adhar) {
        this.adhar = adhar;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getDob() {
        return dob;
    }

    public void setDob(String dob) {
        this.dob = dob;
    }

    public String getSex() {
        return sex;
    }

    public void setSex(String sex) {
        this.sex = sex;
    }

    public String getAddress() {
```

```

        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public int getFlightid() {
        return flightid;
    }

    public void setFlightid(int flightid) {
        this.flightid = flightid;
    }
}

```

Flight.java

```

package com.data;

public class Flight {
    private int flightid;
    private String name;
    private int seats;

    public Flight(int flightid, String name, int seats) {
        super();
        this.flightid = flightid;
        this.name = name;
        this.seats = seats;
    }

    public int getFlightid() {
        return flightid;
    }

    public void setFlightid(int flightid) {
        this.flightid = flightid;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getSeats() {
        return seats;
    }

    public void setSeats(int seats) {
        this.seats = seats;
    }
}

```

AirlineManager.java

```

import java.rmi.server.Operation;
import java.util.NoSuchElementException;
import java.util.Scanner;

import com.data.Passenger;
import com.operation.DatabaseManager;
import com.operation.Operations;

```

```

public class AirlineManager {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Operations op = new Operations();
        Scanner sc = new Scanner(System.in);
        DatabaseManager dm = new DatabaseManager();
        int choice = 0;
        System.out.println("XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX");
        System.out.println("-----SHRI SHRI CHINMAY JI AIRLINE SERVICES-----");
        System.out.println("XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX");
        while (true) {
            System.out.println("1. Insert Passenger Details");
            System.out.println("2. Insert Flight Details");
            System.out.println("3. Book Ticket");
            System.out.println("4. Cancel Ticket");
            System.out.println("5. Display Vacancy Seat Details");
            System.out.println("6. Update Passenger Information");
            System.out.println("7. Update Flight Details");
            System.out.println("8. Exit");
            System.out.print("Enter your choice: ");

            choice = sc.nextInt();
            sc.nextLine();

            switch (choice) {
                case 1:
                    op.insertPassengerDetails(dm);
                    break;
                case 2:
                    op.insertFlightDetails(dm);
                    break;
                case 3:
                    op.bookTicket(dm);
                    break;
                case 4:
                    op.cancelTicket(dm);
                    break;
                case 5:
                    op.displayVacant(dm);
                    break;
                case 6:
                    op.updatePassengerDetails(dm);
                    break;
                case 7:
                    op.updateFlightDetails(dm);
                    break;
                case 8:
                    op.closeConnection(dm);
                    System.out.println("Exiting... Bye Bye");
                    System.out.println("I hope You loved SHRI SHRI CHINMAY JI's AIRLINE service!");

            System.out.println("XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX");
            System.exit(0);
            break;
            default:
                System.out.println("Invalid choice. Please try again.");
            }
        }
    }
}

```

Operations.java

```

package com.operation;

import java.sql.DriverManager;
import java.util.Scanner;

import com.data.Flight;

```

```

import com.data.Passenger;

public class Operations {

    public void insertPassengerDetails(DatabaseManager dm) {

        Scanner sc = new Scanner(System.in);
        System.out.println("-----");
        System.out.println("Enter Passenger Adhar ID :");
        long adhar = sc.nextLong();

        System.out.println("Enter Passenger Name :");
        sc.nextLine();
        String name = sc.nextLine();

        System.out.println("Enter Passenger Gender :");

        String gender = sc.nextLine();
        sc.nextLine();

        System.out.println("Enter Passenger Date of Birth in dd-mm-yyyy :");

        String dob = sc.nextLine();
        sc.nextLine();
        System.out.println("Enter Passenger Address :");

        String address = sc.nextLine();

        Passenger ps = new Passenger(adhar, name, dob, gender, address, 0);
        dm.insertPassenger(ps);

    }

    public void insertFlightDetails(DatabaseManager dm) {
        Scanner sc = new Scanner(System.in);
        System.out.println("-----");
        System.out.println("Enter Flight ID :");
        int id = sc.nextInt();

        System.out.println("Enter Flight Name :");
        String name = sc.next();

        System.out.println("Enter Available Seats :");
        int seats = sc.nextInt();

        Flight f = new Flight(id, name, seats);
        dm.insertFlight(f);
    }

    public void bookTicket(DatabaseManager dm) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter Passenger Adhar :");
        long adhar = sc.nextLong();

        System.out.println("Enter Flight ID :");
        int flightId = sc.nextInt();

        dm.bookTicket(adhar, flightId);
    }

    public void cancelTicket(DatabaseManager dm) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Passenger Adhar to cancel Current Flight :");
        long adhar = sc.nextLong();
        dm.cancelTicket(adhar);
    }

    public void displayVacant(DatabaseManager dm) {
        dm.displayVacantSeatsandFlights();
        System.out.println("-----");
    }
}

```

```

public void updatePassengerDetails(DatabaseManager dm) {
    Scanner scanner = new Scanner(System.in);

    System.out.println("Enter Adhar number of the passenger: ");
    long adhar = scanner.nextLong();
    scanner.nextLine();
    System.out.println("Select the column to update:");
    System.out.println("1. Name");
    System.out.println("2. Sex");
    System.out.println("3. Date of Birth");
    System.out.println("4. Address");
    int choice = scanner.nextInt();
    scanner.nextLine();
    String columnName = "";
    switch (choice) {
        case 1:
            columnName = "name";
            break;
        case 2:
            columnName = "sex";
            break;
        case 3:
            columnName = "dob";
            break;
        case 4:
            columnName = "address";
            break;
        default:
            System.out.println("Invalid choice.");
            return;
    }

    System.out.println("Enter the new value for " + columnName + ": ");
    String newValue = scanner.nextLine();

    dm.updatePassenger(adhar, columnName, newValue);
}

public void updateFlightDetails(DatabaseManager dm) {
    Scanner scanner = new Scanner(System.in);
    System.out.println("Enter the flight ID: ");
    int flightId = scanner.nextInt();
    scanner.nextLine();

    System.out.println("Select the column to update:");
    System.out.println("1. Flight Name");
    System.out.println("2. Seats");
    int choice = scanner.nextInt();
    scanner.nextLine();

    String columnName = "";
    switch (choice) {
        case 1:
            columnName = "flightname";
            break;
        case 2:
            columnName = "seats";
            break;
        default:
            System.out.println("Invalid choice.");
            return;
    }

    System.out.println("Enter the new value for " + columnName + ": ");
    String newValue = scanner.nextLine();

    dm.updateFlight(flightId, columnName, newValue);
    System.out.println("-----");
}

public void closeConnection(DatabaseManager dm) {
    dm.close();
}

```

```
}
```

DatabaseManager.java

```
package com.operation;

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

import com.data.Flight;
import com.data.Passenger;

public class DatabaseManager {
    private Connection connection;

    public DatabaseManager() {
        try {
            connection = DriverManager.getConnection("jdbc:mysql://localhost:3307/chinmayairline",
"root", "");
            System.out.println("Connected to database!");
        } catch (SQLException e) {
            System.out.println("Exception occurs in Connection!");
            e.printStackTrace();
        }
    }

    public void insertPassenger(Passenger passenger) {
        try {
            PreparedStatement statement = connection.prepareStatement("INSERT INTO passengers VALUES
(?, ?, ?, ?, ?, ?)");
            statement.setLong(1, passenger.getAdhar());
            statement.setString(2, passenger.getName());
            statement.setString(3, passenger.getSex());
            statement.setString(4, passenger.getDob());
            statement.setString(5, passenger.getAddress());
            statement.setInt(6, passenger.getFlightid());
            statement.executeUpdate();
            System.out.println("Passenger Added Successfully!");
            System.out.println("-----");
            System.out.println();
            statement.close();
        } catch (SQLException e) {
            System.out.println("Exception occurs in Passenger Insertion!");
            e.printStackTrace();
        }
    }

    public void insertFlight(Flight flight) {
        try {
            PreparedStatement statement = connection.prepareStatement("INSERT INTO flights values (?,
?, ?)");
            statement.setInt(1, flight.getFlightid());
            statement.setString(2, flight.getName());
            statement.setInt(3, flight.getSeats());
            statement.executeUpdate();
            System.out.println("Flight Added Successfully!");
            System.out.println("-----");
            statement.close();
        } catch (SQLException e) {
            System.out.println("Exception occurs in Flights Insertion!");
            e.printStackTrace();
        }
    }

    public void bookTicket(long passengerId, int flightId) {
        try {
            String checkSeatsQuery = "SELECT seats FROM flights WHERE id = ?";
            PreparedStatement preparedStatement = connection.prepareStatement(checkSeatsQuery);
            preparedStatement.setInt(1, flightId);
            ResultSet resultSet = preparedStatement.executeQuery();
```

```

if (resultSet.next()) {
    int availableSeats = resultSet.getInt("seats");
    if (availableSeats > 0) {
        // Book the ticket
        String bookTicketQuery = "UPDATE passengers set flightid = ? where adhar = ?";
        preparedStatement = connection.prepareStatement(bookTicketQuery);
        preparedStatement.setInt(1, flightId);
        preparedStatement.setLong(2, passengerId);
        int rowsAffected = preparedStatement.executeUpdate();
        if (rowsAffected > 0) {
            // Update available seats
            String updateSeatsQuery = "UPDATE flights SET seats = ? WHERE id = ?";
            preparedStatement = connection.prepareStatement(updateSeatsQuery);
            preparedStatement.setInt(1, availableSeats - 1);
            preparedStatement.setInt(2, flightId);
            int updatedRows = preparedStatement.executeUpdate();
            if (updatedRows > 0) {
                System.out.println("Opration Done!");
            }
            System.out.println("Ticket Booked Perfectly!");
            System.out.println("-----");
        }
    } else {
        System.out.println("Sorry, no available seats for this flight.");
    }
} else {
    System.out.println("Invalid flight ID.");
}
} catch (Exception e) {
    e.printStackTrace();
}
}

public void cancelTickett(long passengerId) {
    PreparedStatement preparedStatement = null;
    try {
        // Get the flight ID of the passenger
        String getFlightIdQuery = "SELECT flightid FROM passengers WHERE adhar = ?";
        preparedStatement = connection.prepareStatement(getFlightIdQuery);
        preparedStatement.setLong(1, passengerId);
        int flightId = 0;
        try (ResultSet resultSet = preparedStatement.executeQuery()) {
            if (resultSet.next()) {
                flightId = resultSet.getInt("flightid");
            }
        }

        // Cancel the passenger's ticket
        String cancelTicketQuery = "UPDATE passengers SET flightid = 0 WHERE adhar = ?";
        preparedStatement = connection.prepareStatement(cancelTicketQuery);
        preparedStatement.setLong(1, passengerId);
        int rowsAffectedPassenger = preparedStatement.executeUpdate();

        // Increment available seats for the flight
        String incrementSeatsQuery = "UPDATE flights SET seats = seats + 1 WHERE id = ?";
        preparedStatement = connection.prepareStatement(incrementSeatsQuery);
        preparedStatement.setInt(1, flightId);
        int rowsAffectedFlight = preparedStatement.executeUpdate();

        // If both updates were successful, set success flag to true
        if (rowsAffectedPassenger > 0 && rowsAffectedFlight > 0) {
            System.out.println("Ticket Canceled Successfully!!!");
            System.out.println("-----");
        }
    } catch (SQLException e) {
        e.printStackTrace();
        System.out.println("Got exception bro 1");
    }
}

public void displayVacantSeatsandFlights() {
    PreparedStatement preparedStatement = null;
    ResultSet resultSet = null;

```

```

try {
    String query = "SELECT id, flightname, seats FROM flights";
    PreparedStatement = connection.prepareStatement(query);
    resultSet = preparedStatement.executeQuery();

    // Print table header
    System.out.println("-----");
    System.out.printf("%-10s %-20s %-10s\n", "Flight ID", "Flight Name", "Vacant Seats");
    System.out.println("-----");
    // Print table rows
    while (resultSet.next()) {
        int flightId = resultSet.getInt("id");
        String flightName = resultSet.getString("flightname");
        int vacantSeats = resultSet.getInt("seats");
        System.out.printf("%-10d %-20s %-10d\n", flightId, flightName, vacantSeats);
    }
    System.out.println("-----");
} catch (SQLException e) {
    e.printStackTrace();
}

}

public void updatePassenger(long adhar, String columnName, String newValue) {
    PreparedStatement preparedStatement = null;
    ResultSet resultSet = null;
    try {
        String updateQuery = "UPDATE passengers SET " + columnName + " = ? WHERE adhar = ?";
        preparedStatement = connection.prepareStatement(updateQuery);
        preparedStatement.setString(1, newValue);
        preparedStatement.setLong(2, adhar);

        int rowsAffected = preparedStatement.executeUpdate();

        if (rowsAffected > 0) {
            System.out.println("Passenger Details Updated Successfully!");
            System.out.println("-----");
        } else {
            System.out.println("No rows were updated. Please check the provided Adhar number.");
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

public void updateFlight(int flightId, String columnName, String newValue) {
    PreparedStatement preparedStatement = null;
    ResultSet resultSet = null;
    try {
        String updateQuery = "UPDATE flights SET " + columnName + " = ? WHERE id = ?";
        preparedStatement = connection.prepareStatement(updateQuery);
        preparedStatement.setString(1, newValue);
        preparedStatement.setInt(2, flightId);

        int rowsAffected = preparedStatement.executeUpdate();

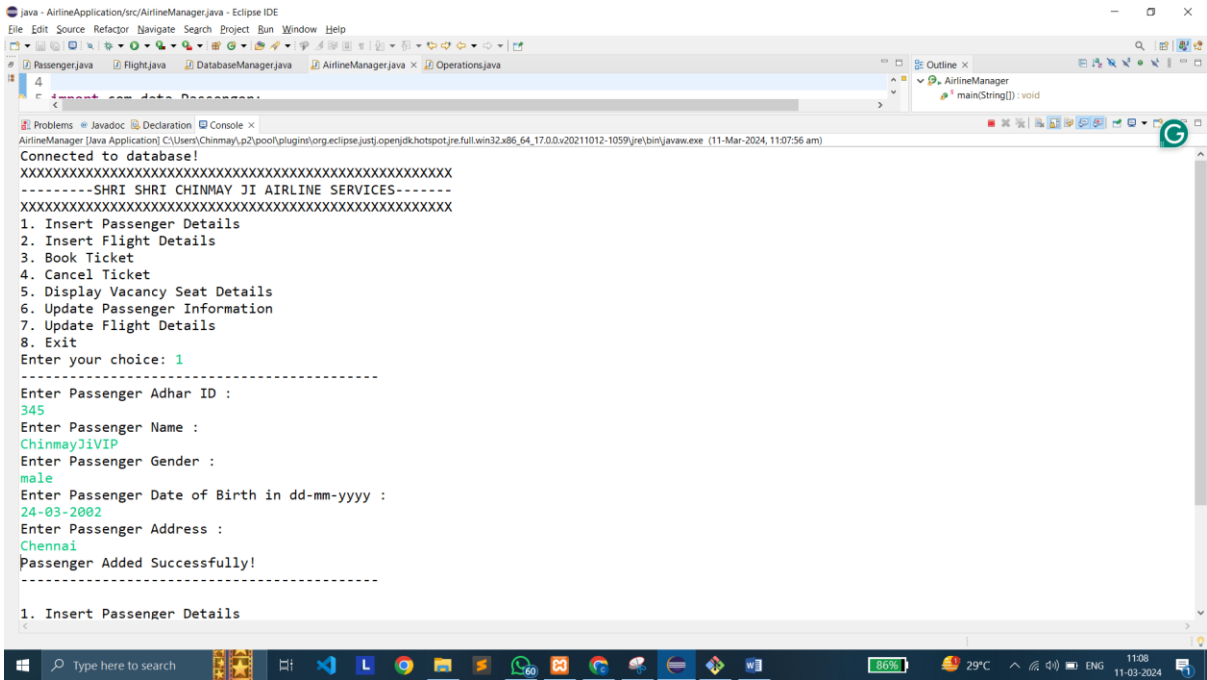
        if (rowsAffected > 0) {
            System.out.println("Flights details Updated Successfully!");
        } else {
            System.out.println("No rows were updated. Please check the provided flight ID.");
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

public void close() {
    try {
        connection.close();
    }
    catch (Exception e) {
        System.out.println("Failed to close connection!");
    }
}

```


} }
Outputs

1. Insert Passenger Details



☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key: None

Extra options

	adhar	name	sex	dob	address	flightid
<input type="checkbox"/> Edit Copy Delete	54	Lingapalli	female	05-07-1970	Andhra	12
<input type="checkbox"/> Edit Copy Delete	88	chimaji	female	04-09-2002	Vellore	0
<input checked="" type="checkbox"/> Edit Copy Delete	345	ChinmayJiVIP	male	24-03-2002	Chennai	0
<input type="checkbox"/> Edit Copy Delete	9966	flybird	male	23-02-2021	chennai	0
<input type="checkbox"/> Edit Copy Delete	5542213	Chinmay	male	23-02-2021	Rakshas Bhuvan	0

☐ Check all | With selected: [Edit](#) [Copy](#) [Delete](#) [Export](#)

2. Insert Flight Details

The screenshot shows the Eclipse IDE with the `AirlineManager.java` file open. The console displays the following output:

```

1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket
4. Cancel Ticket
5. Display Vacancy Seat Details
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice: 2
-----
Enter Flight ID :
123
Enter Flight Name :
ChennaiAirlines
Enter Available Seats :
8
Flight Added Successfully!
-----
1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket
4. Cancel Ticket
5. Display Vacancy Seat Details
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice:

```

The screenshot shows the phpMyAdmin interface with the 'flights' table selected in the 'chinmayairline' database. The table contains the following data:

id	flightname	seats
12	IndianAirlines	99
13	Indigo	80
51	JamnagarAirService	55
67	Hongkong	2
98	TamilAirlines	78
123	ChennaiAirlines	8

3. Book Ticket

```

java - AirlineApplication/src/AirlineManager.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Passenger.java Flight.java DatabaseManager.java AirlineManager.java Operations.java
4
1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket
4. Cancel Ticket
5. Display Vacancy Seat Details
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice: 3
Enter Passenger Adhar :
345
Enter Flight ID :
123
Operation Done!
Ticket Booked Perfectly!
-----
1. Insert Passenger Details
2. Insert Flight Details

```

Server: 127.0.0.1:3307 » Database: chinmayairline » Table: passengers

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	adhar	name	sex	dob	address	flightid
<input type="checkbox"/>	54	Lingapalli	female	05-07-1970	Andhra	12
<input type="checkbox"/>	88	chimaji	female	04-09-2002	Vellore	0
<input type="checkbox"/>	345	ChinmayJiVIP	male	24-03-2002	Chennai	123
<input type="checkbox"/>	9966	flybird	male	23-02-2021	chennai	0
<input type="checkbox"/>	5542213	Chinmay	male	23-02-2021	Rakshas Bhuvan	0

☐ Check all | With selected: ☐ Edit ☐ Copy ☐ Delete ☐ Export

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

4. Cancel Ticket

```

java - AirlineApplication/src/AirlineManager.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Passenger.java Flight.java DatabaseManager.java AirlineManager.java Operations.java
4
Ticket booked Perfectly:
-----
1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket
4. Cancel Ticket
5. Display Vacancy Seat Details
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice: 4
Enter Passenger Adhar to cancel Current Flight :
345
Ticket Canceled Successfully!!!
-----
1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket

```

Server: 127.0.0.1:3307 Database: chinmayairline Table: passengers

Number of rows: 25 Filter rows: Search this table Sort by key: None

	adhar	name	sex	dob	address	flightid
<input type="checkbox"/> Edit Copy Delete	54	Lingapalli	female	05-07-1970	Andhra	12
<input type="checkbox"/> Edit Copy Delete	88	chimaji	female	04-09-2002	Vellore	0
<input type="checkbox"/> Edit Copy Delete	345	ChinmayJiVIP	male	24-03-2002	Chennai	0
<input type="checkbox"/> Edit Copy Delete	9966	flybird	male	23-02-2021	chennai	0
<input type="checkbox"/> Edit Copy Delete	5542213	Chinmay	male	23-02-2021	Rakshas Bhuvan	0

Check all With selected: Edit Copy Delete Export

5. Display Vacancy Seat Details

```

java - AirlineApplication/src/AirlineManager.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Passenger.java Flight.java DatabaseManager.java AirlineManager.java Operations.java
4
Problems Javadoc Declaration Console
AirlineManager [Java Application] C:\Users\Chinmay\p2\pooft\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.0.v20211012-1059\jre\bin\javaw.exe (11-Mar-2024, 11:07:56 am)
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice: 5
-----
Flight ID  Flight Name      Vacant Seats
-----
12         IndianAirlines    99
13         Indigo           80
51         JamnagarAirService 55
67         Hongkong         2
98         TamilAirlines    78
123        ChennaiAirlines   8
-----
1. Insert Passenger Details
2. Insert Flight Details
3. Book Ticket

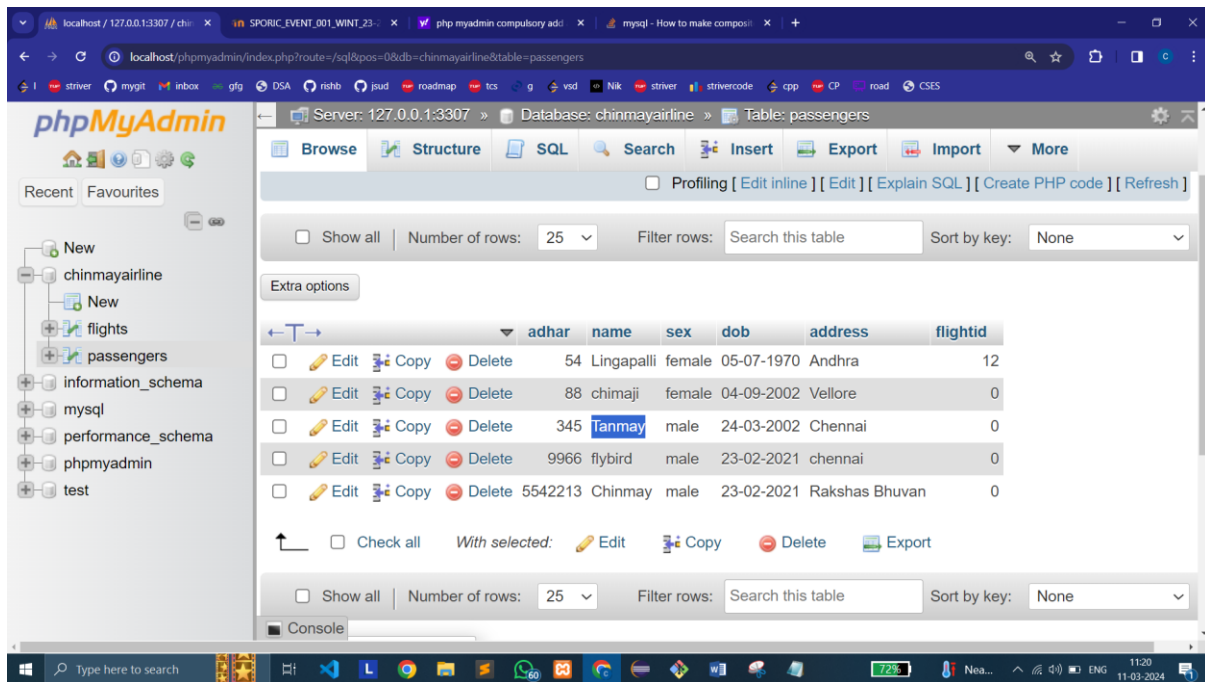
```

6. Update Passenger Information

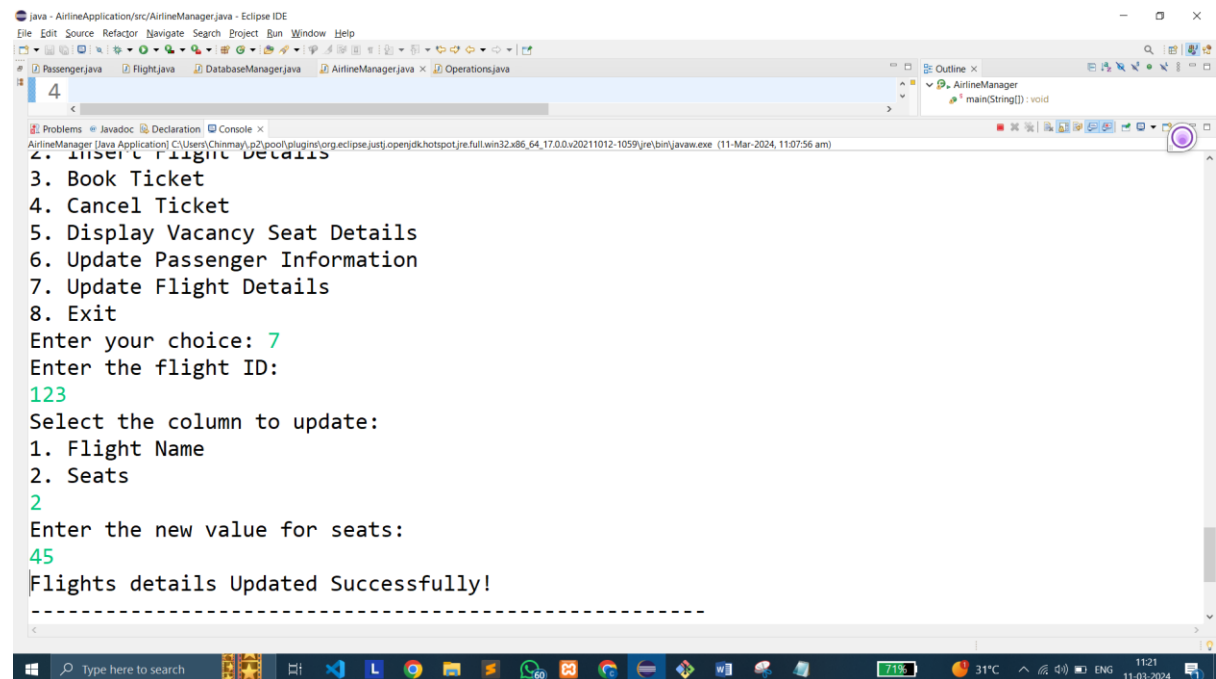
```

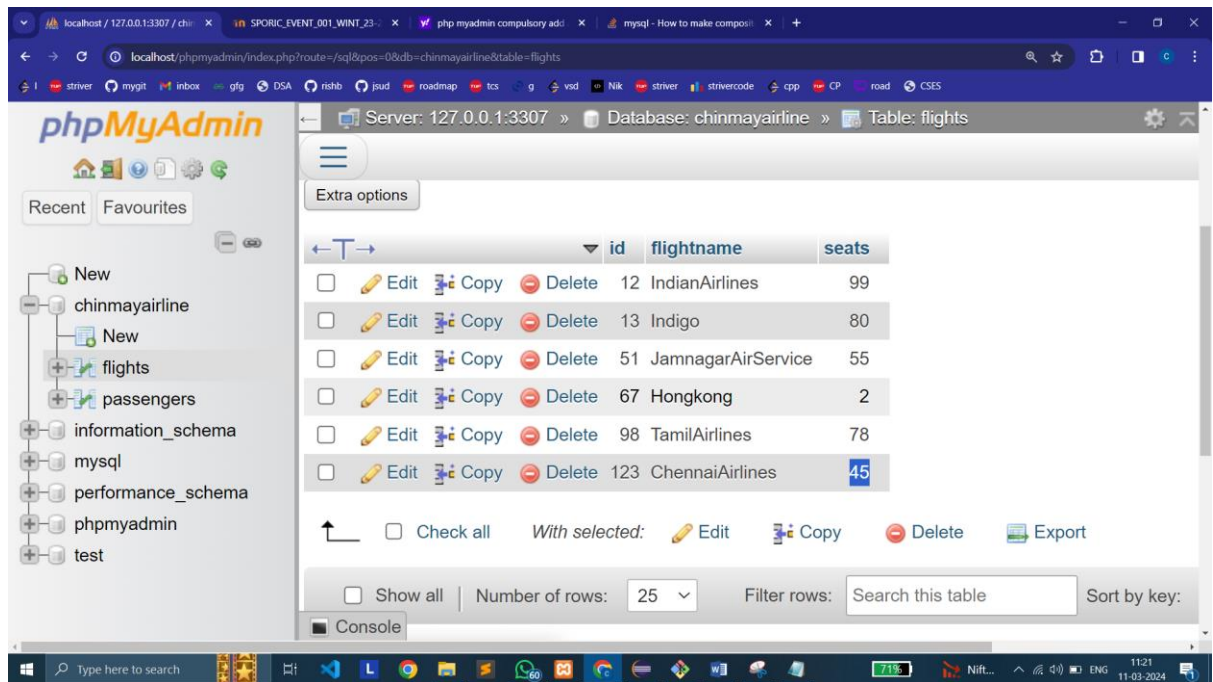
java - AirlineApplication/src/AirlineManager.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Passenger.java Flight.java DatabaseManager.java AirlineManager.java Operations.java
4
Problems Javadoc Declaration Console
AirlineManager [Java Application] C:\Users\Chinmay\p2\pooft\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.0.v20211012-1059\jre\bin\javaw.exe (11-Mar-2024, 11:07:56 am)
4. CANCEL TICKET
5. Display Vacancy Seat Details
6. Update Passenger Information
7. Update Flight Details
8. Exit
Enter your choice: 6
Enter Adhar number of the passenger:
345
Select the column to update:
1. Name
2. Sex
3. Date of Birth
4. Address
1
Enter the new value for name:
Tanmay
Passenger Details Updated Successfully!
-----

```



7. Update Flight Details





8. Exit

