جامعة عفت

EFFAT UNIVERSITY

CS 2132 - Object Oriented Programming

Final Project Report - Spring 2022

Students:

Mehreen Junaid S20106424

Danah Milyani S19105577

Mageda Alasabi S19105541

Instructor: **Fidaa Abed**

Date Last Edited: May 19, 2022

Contents

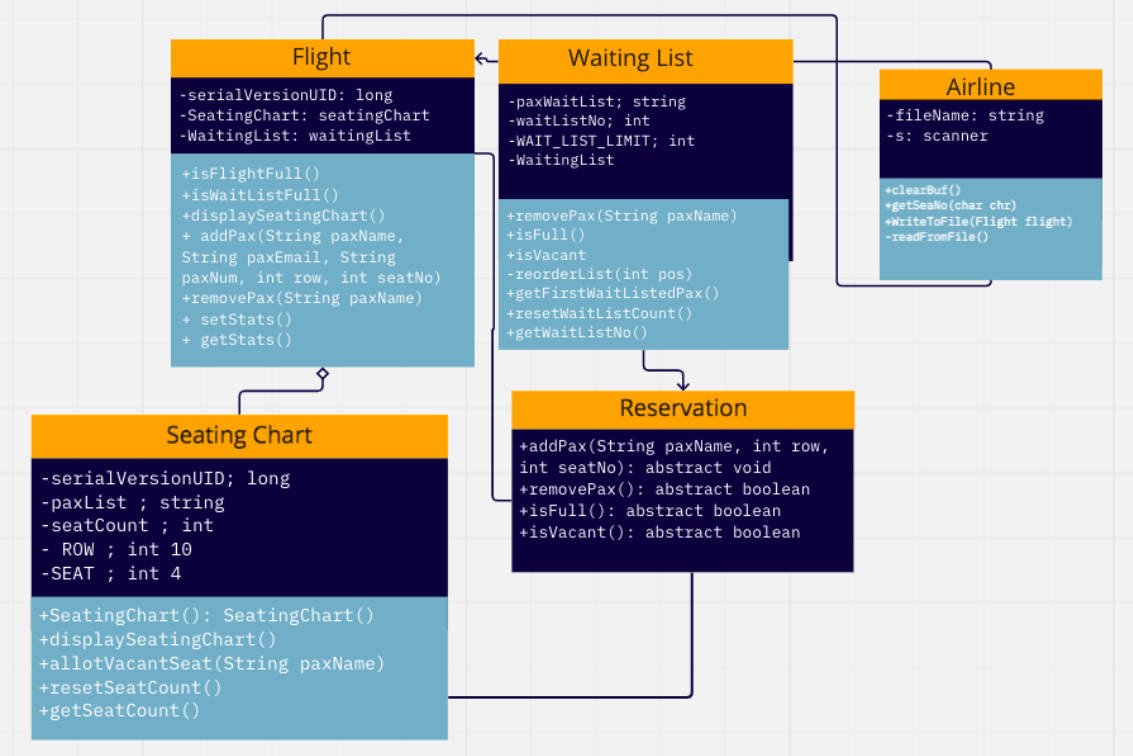
1 Objectives

- To learn how to build and evolve middle-scale programs using object-oriented programming, and work in teams learning from each other.

2 Learning Outcomes

- CLO2: Implement classes (base and derived) using java programming language and use function overloading appropriately. (PLO: B)
- CLO4: Build complete java programs from textual description using exceptions and java collections and iterators when needed. (PLO: A)
- CLO5: Work in teams to in the design and implementation process. (PLO: E)
- CLO6: Document programming projects and present programming projects. (PLO: C)

3 UML DIAGRAMS



4 CLASSES METHODS

4.1 Flight Class

Initializes the attributes seating chart and waiting list, add default constructors seatingChart and waitingList. Methods include checking if the flight is full and returns seating chart, checks if waiting list is full and returns wait list. Displays seating chart and then adds passenger onto flight, also removes passenger when asked (from waiting list and seating chart). Set and get static counter variables of SeatingChart and WaitingList.

4.2 WaitingList Class

Attributes include paxWaitList, waitListNo, and WAIT-LIST-LIMIT. It contains overrides from flight class, such as: check lists in full and add or remove passengers. Methods include, reorders the waiting list after deletion of a passenger and returns the first waitlisted passenger.

4.3 SeatingChart Class

Implements reservation class , attributes include paxList; seatCount; transient int ROW = 10; transient int SEAT = 4; contains overrides from flight class, such as: check lists in full and add or remove passengers. Methods include alloting the recent vacated seat to the first wait-listed passenger, eeturn the seat count after reading from file and return the seatCount

4.4 Airline Class

The Attributes are fileName and s which is a Scanner type. Methods include, getting the seat numbers, and writing to the file and reading from the file (Exception was used in both methods).

4.5 Reservation Class

This is an interface which is an abstract that is used to group the methods with empty bodies.To achieve security, having this class hides certain details and only show the important details of an object (interface). Since, java does not support "multiple inheritance", by having an interface, the class can implement multiple interfaces. The reservation interface has four abstract classes that are addPax(String paxName, int row, int seatNo), removePax(String paxName), isFull(), and isVacant().

5 USER MANUAL

The user is first shown 3 options as they run the program which are to add a passenger, remove a passenger, or exit the program.

```
AIRLINE RESERVATION SYSTEM
The services we provide you with inlcude:
1. Add a passenger to the system
2. Remove a passenger from the system
3. Exit the system
Enter your choice :
```

To add a passenger the user has to enter passenger details such as name, email, and phone number. Once the details are added, they choose a the seat by entering row and column number.

```
AIRLINE RESERVATION SYSTEM
The services we provide you with inlcude:
1. Add a passenger to the system
2. Remove a passenger from the system
3. Exit the system
Enter your choice :
1
Enter passenger's name , email and phone number :
Mageda, ma@gmail.com, 0555555555
-----------------------------------------
A B C D

_ _ _ _ 1
_ _ _ _ 2
_ _ _ _ 3
X _ _ _ 4
_ _ _ _ 5
_ X _ _ 6
_ _ _ _ 7
_ _ _ _ 8
_ _ _ X 9
_ _ _ _ 10
X - Occupied    _ - Vacant

-----------------------------------------
Enter the seat number [Row number followed by seat number] :
1A
```

To remove a passenger, the user has to enter the passenger details, if there is a match, the passenger reservation is cancelled. However, if there is no match, they get a message saying "no such passenger exists".

```
AIRLINE RESERVATION SYSTEM
The services we provide you with inlcude:
1. Add a passenger to the system
2. Remove a passenger from the system
3. Exit the system
Enter your choice :
2
Enter passenger's information :
Mageda, ma@gmail.com, 0555555555
Passenger removed from seating chart
```

When the user chooses to exit the program,they get the flight status which is the number of booked seats and number of people in the waiting list.

```
AIRLINE RESERVATION SYSTEM
The services we provide you with inlcude:
1. Add a passenger to the system
2. Remove a passenger from the system
3. Exit the system
Enter your choice :
3
FLIGHT STATUS
Total seats taken : 3
Total number of passengers in waiting list : 1
```

If the flight is full the user will get a message saying it's full and their name will be added to the waiting list.

```
AIRLINE RESERVATION SYSTEM
The services we provide you with inlcude:
1. Add a passenger to the system
2. Remove a passenger from the system
3. Exit the system
Enter your choice :
1
Enter passenger's name , email and phone number :
mehreen m@gmail.com 1233
Flight is full. Passenger added to the wait list
Do you want to continue[Y/N] :
```

6 MEMBERS CONTRIBUTION

6.1 Danah Milyani

- Flight and Airline classes.
- UML diagrams for flight, seatingclass, and waitinglist class
- classes methods description for those classes (in report)
- future recommendation (in report)

6.2 Mehreen Junaid

- WaitingList and SeatingList classes
- UML diagram for reservation class
- Classes methods description for the reservation class(in report)
- User manual (in report)

6.3 Mageda Alasabi

- The UML for Airline class,
- Reservations class
- classes methods description for Airline class (in report)
- Required video of our project

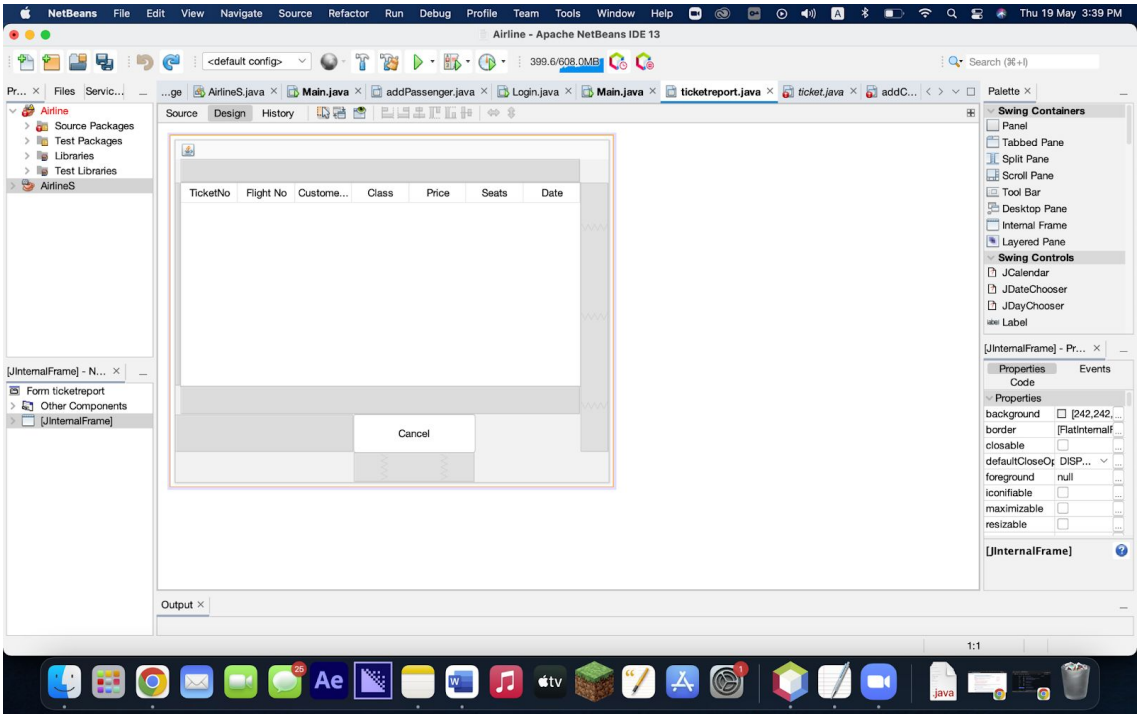
7 SHORT VIDEO

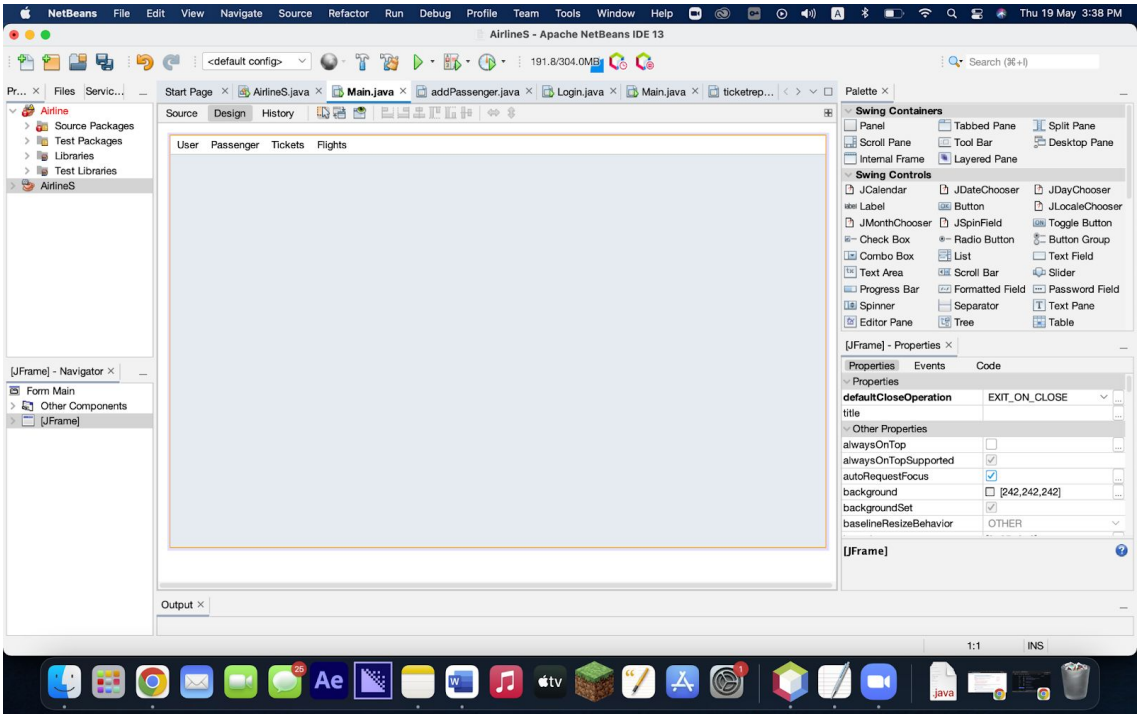
https://youtu.be/Y6tfvinZxDs

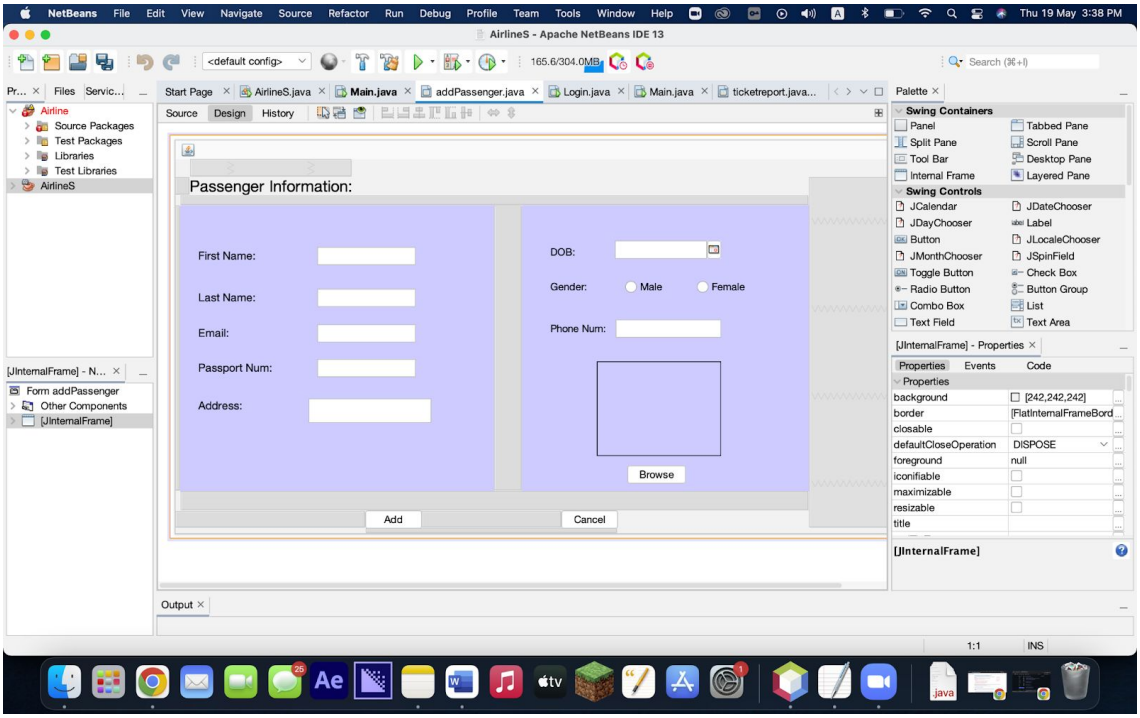
8 FUTURE RECOMMENDATIONS AND CONCLUSION

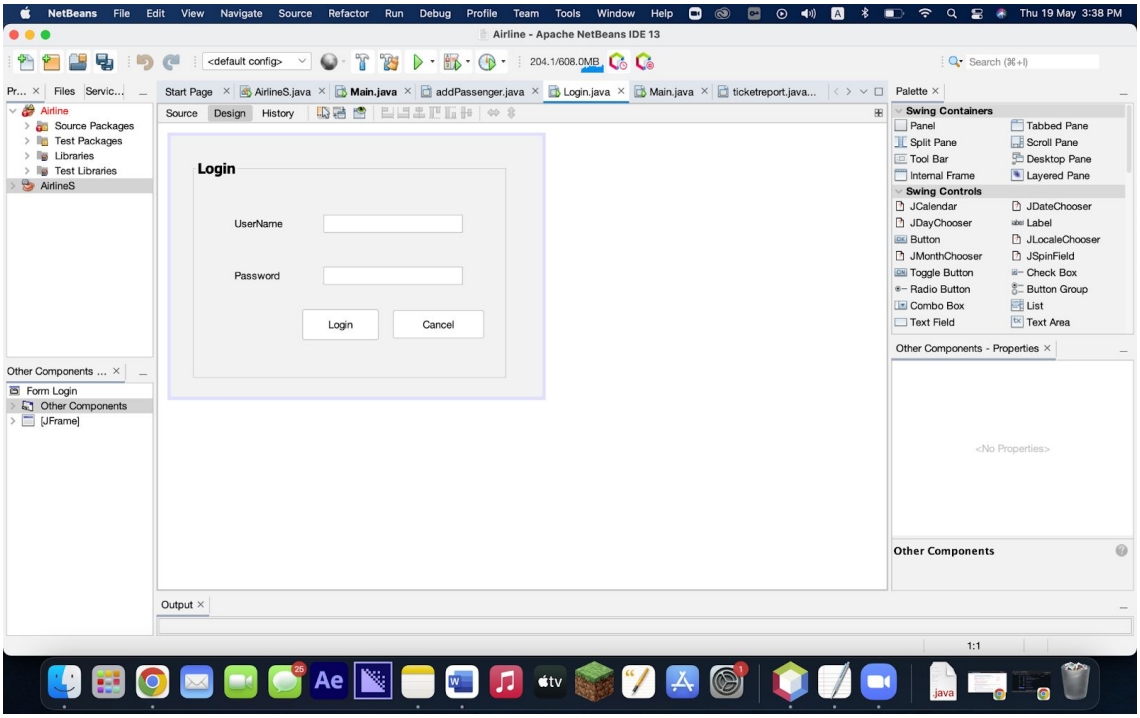
Unfortunately due to time constraints the GUI section was not able to be completed however we have provided a future recommendation section in hopes this could also be allowed and counted as a bonus (if the doctor wishes to grant it). Using APACHE NETBEANS 13 a java gui for ‘ Airline Reservation System ‘ project could be generated , with Java OOP , JSwing/Form and MYSQL Database. Users can register and create accounts for the reservation system, they would be able to input details in their passenger information section (including name , email , passport number, picture etc), they can book tickets and see the flights that they would be able to book.

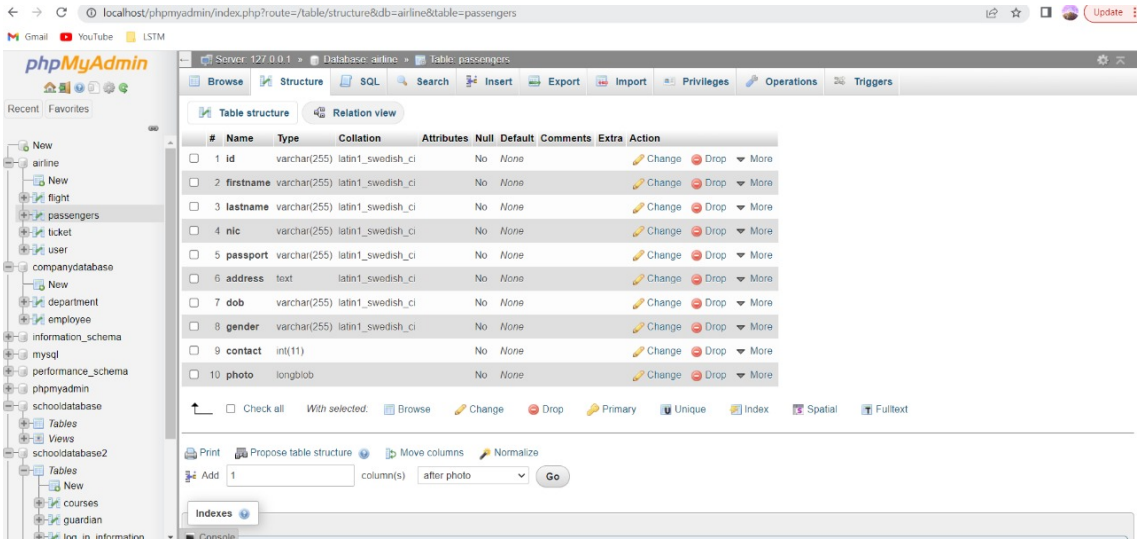
Below is an example of what it would look like (these screenshots are from our work not outside sources)

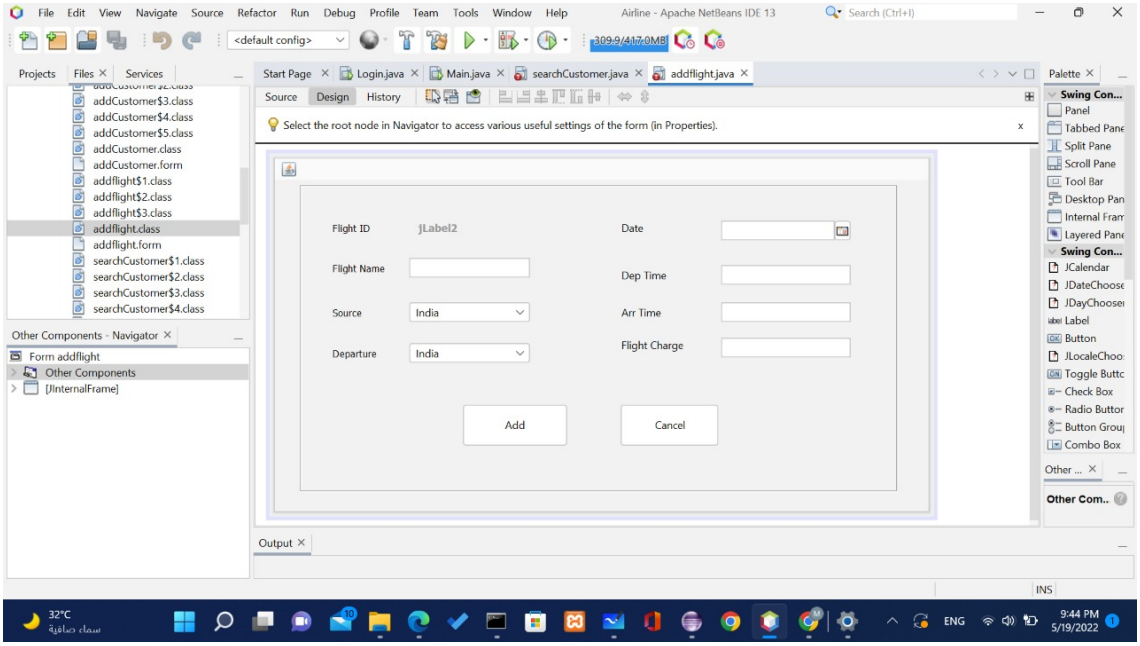

Airline - Apache NetBeans IDE 13
TicketNo
Flight No
Custome...
Class
Price
Seats
Date
Cancel


AirlineS - Apache NetBeans IDE 13
User
Passenger
Tickets
Flights


AirlineS - Apache NetBeans IDE 13
Passenger Information:
First Name:
Last Name:
Email:
Passport Num:
Address:
DOB:
Gender:
Male
Female
Phone Num:
Browse
Add
Cancel


Airline - Apache NetBeans IDE 13
Login
UserName
Password
Login
Cancel


phpMyAdmin
Server: 127.0.0.1 » Database: airline » Table: passengers
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 id varchar(255) latin1_swedish_ci No None Change Drop More
2 firstname varchar(255) latin1_swedish_ci No None Change Drop More
3 lastname varchar(255) latin1_swedish_ci No None Change Drop More
4 nic varchar(255) latin1_swedish_ci No None Change Drop More
5 passport varchar(255) latin1_swedish_ci No None Change Drop More
6 address text latin1_swedish_ci No None Change Drop More
7 dob varchar(255) latin1_swedish_ci No None Change Drop More
8 gender varchar(255) latin1_swedish_ci No None Change Drop More
9 contact int(11) No None Change Drop More
10 photo longblob No None Change Drop More
Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext
Print Propose table structure Move columns Normalize
Add 1 column(s) after photo Go
Indexes


Airline - Apache NetBeans IDE 13
Start Page Login.java Main.java searchCustomer.java addflight.java
Source Design History
Select the root node in Navigator to access various useful settings of the form (in Properties).
Flight ID jLabel2
Flight Name
Source India
Departure India
Date
Dep Time
Arr Time
Flight Charge
Add
Cancel
Output