

AEROSYNC

AIRCRAFT DATABASE MANAGEMENT SYSTEM

CREATED BY-

RAUNAK BALIYAN (2022UCM2324)

SAHIL GUPTA (2022UCM2382)

SHREYASH SHARMA (2022UCM2385)

CONTENTS

CHAPTER 1 INTRODUCTION

- 1.1 Introduction of the Problem Area**
- 1.2 Problem Statement/Purpose of the Project**
- 1.3 Objective of the Project**
- 1.4 Find and organize the information required(Requirement Analysis)**

CHAPTER 2 HARDWARE REQUIREMENT

- 2.1 Hardware Tools**

- 2.2 Software Tools**

CHAPTER 3 DESIGN

- 3.1 ER Model**

- 3.2 ER to Schema {Divide the information into tables}**

- 3.3 Functional Dependencies {Based on the requirement that matches with the sample data}**

- 3.4 Specify the primary key**

- 3.5 Set-up the table relationship**

CHAPTER 4 IMPLEMENTATION

- 4.1 Xampp Table**

- 4.2 Application with code**

- 4.3 Snapshots of running of application**

CHAPTER 5 REFERENCES



Aircraft Database Management System

CHAPTER 1: INTRODUCTION

1.1 Introduction:

This is an International Airport Database project which consists of developing of entities and their respective attributes .This approach also includes writing and applying queries as required . After forming initial Entities with their attributes then we proceed with developing ER diagram for given entities in further phases. Scheduling of flights, managing different service provided airplanes with respect to its terminals, Bookings of flights, details of passengers are the main aspects that are covered under this project . This database can handle different requests and provides an output based on above entities. The results from this database is used by airport authorities for better decision making.

1.2 Problem Statement/Purpose of the Project :

The Aircraft Database Management System (ADMS) project aims to revolutionize the aviation industry by addressing critical challenges in data management, safety, and operational efficiency. Currently, disparate systems and manual processes hinder the centralized storage and effective utilization of aircraft information. The ADMS seeks to centralize this data, providing a

comprehensive solution for tracking maintenance, ensuring regulatory compliance, and optimizing operational processes. By offering real-time insights through analytical tools, the system empowers decision-makers with the information needed for strategic planning, predictive maintenance, and resource optimization. The ADMS project strives to enhance overall safety, streamline operations, and contribute to a more efficient and secure aviation environment.

1.3 Objective of the Project:

The main objective of the project is to learn and implement a real-time application on database for airport database management system. The project, concentrates on bookings, tracking the flights . This Database will be a great solution for many functionalities in Airport.

The Aircraft Database Management System (ADMS) project seeks to revolutionize the aviation industry through a set of comprehensive objectives. First and foremost, the project aims to establish a centralized database, harmonizing diverse aircraft data, including specifications, maintenance records, and regulatory compliance

information. This centralization is designed to eliminate data silos, ensuring accurate and efficient data management.

Safety and compliance improvement form another critical objective, with the ADMS implementing real-time monitoring of maintenance schedules and safety protocols. By doing so, the project endeavors to minimize errors, oversights, and enhance overall regulatory adherence, fostering a safer aviation environment. Concurrently, the project strives to optimize operational efficiency by streamlining resource utilization, flight schedules, and crew assignments. The incorporation of decision support mechanisms, driven by data analytics, further empowers decision-makers with actionable insights for strategic planning and predictive maintenance.

1.4 Find and organize the information required(Requirement Analysis) :

AeroSync aims to develop a user-friendly Aircraft Database Management System, focusing on fundamental features to enhance data organization within the aviation sector. The system will efficiently store and manage essential aircraft information, records, and flight details.

AeroSync's user interface will be designed for simplicity, allowing users to navigate seamlessly and access pertinent information. It will feature basic

functionalities such as viewing and updating aircraft details, logging maintenance activities, and generating simple reports. The system will prioritize accessibility and ease of use, ensuring that users, regardless of their role, can interact with the platform effortlessly.

CHAPTER 2: HARDWARE REQUIREMENT

2.1 Hardware Tools

Hardware Interface:

- Windows Operating System
 - Minimum 1 GB of RAM
 - Minimum 100 GB of HDD

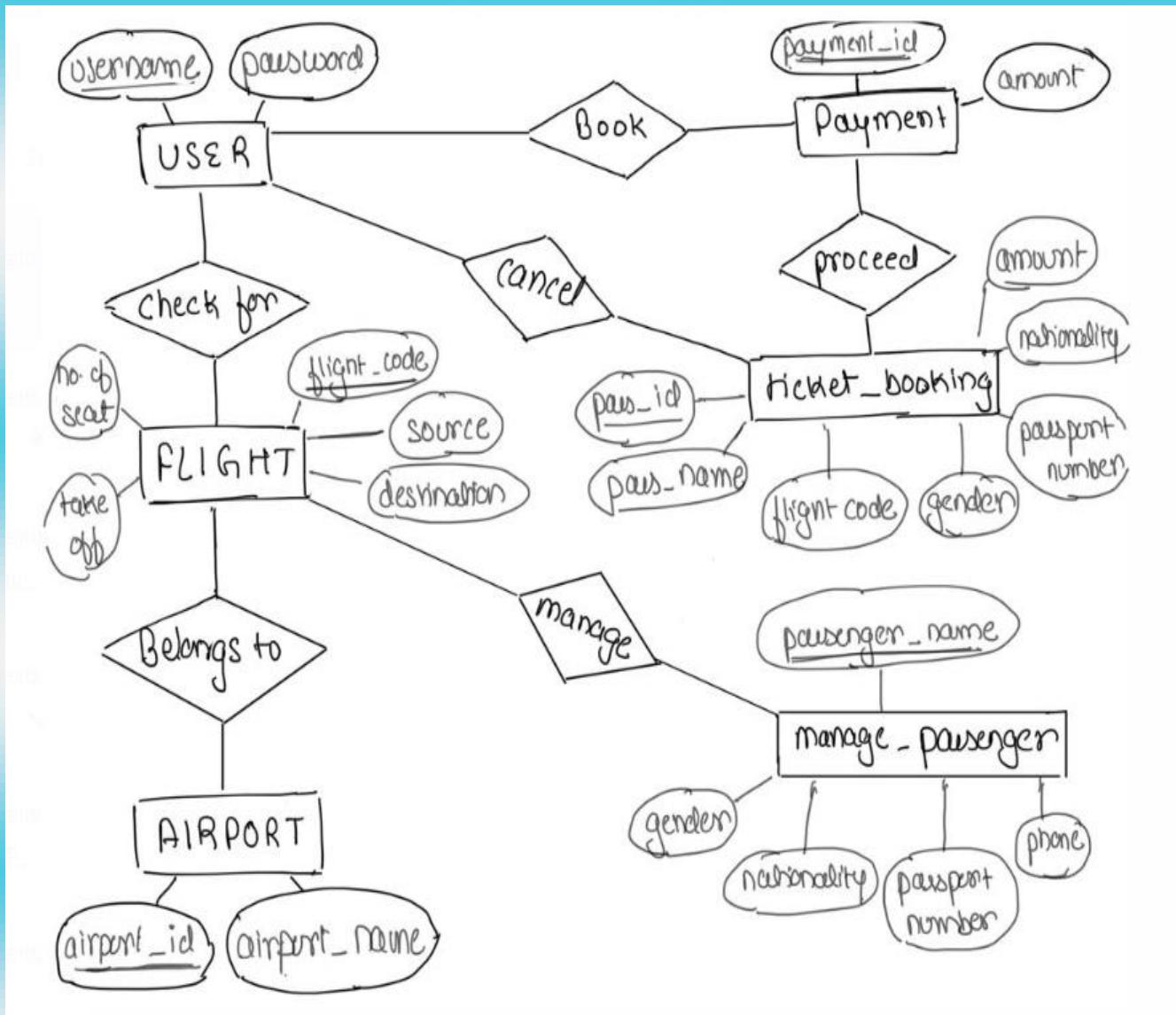
2.2 Software Tools

Software Interface:

- Operating System: Windows (Vista/7 or above)
- Apache NetBeans IDE 18
- WAMP server v2.5
 - Interfacing using PHP
 - Database Management using MySQL
 - MySQL connector
- ErWin : Ca ErWin Data Modulator for Entity Relational Diagram.

CHAPTER 3: DESIGN

3.1 ER Model



3.2 ER to Schema {Divide the information into tables}

USER (username, password)

FLIGHT (flight-code, source, destination, no.of seat, take off)

AIRPORT (airport-id, airport-name)

PAYMENT (payment-id, amount)

TICKET_BOOKING (pans-id, pans-name, amount, flight code, nationality
, gender, passport number)

MANAGE_PASSENGER (pans-name, gender, nationality, passport, phone
number)

IMPLEMENTATION

4.1 Xampp Tables

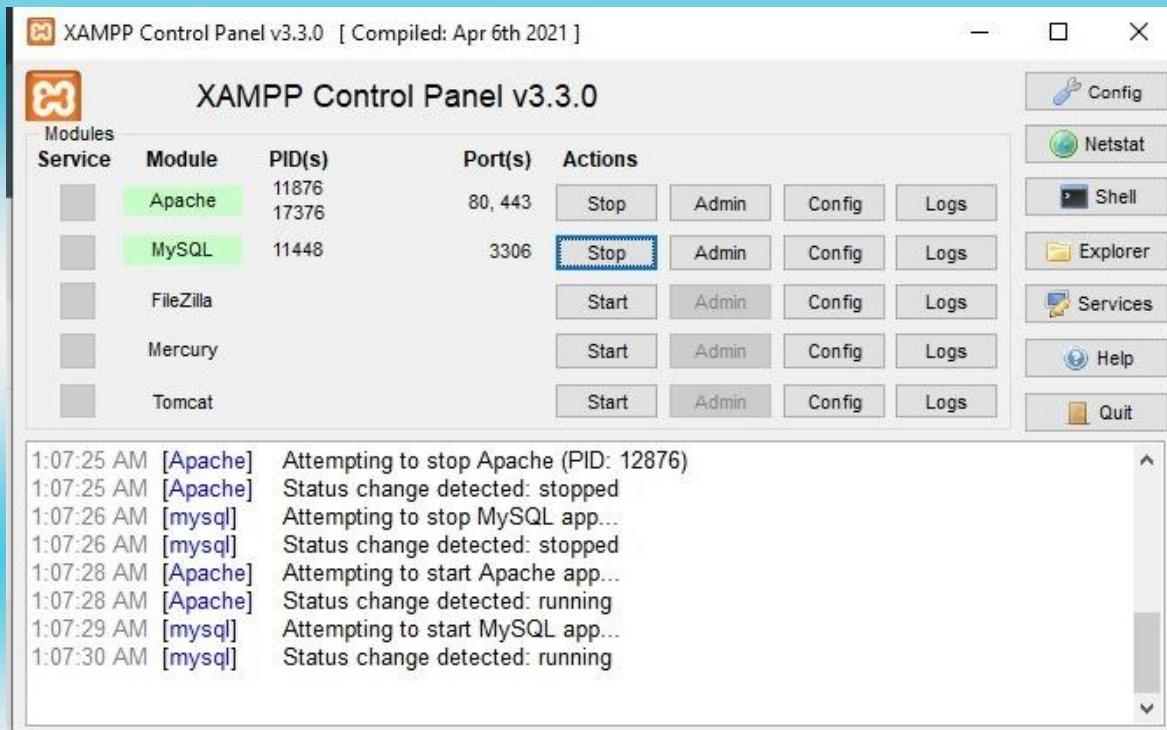


Table	Action	Rows	Type	Collation	Size	Overhead
airport	★ Browse Structure Search Insert Empty Drop	15	InnoDB	utf8mb4_general_ci	16.0 KiB	-
login	★ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
manageflight	★ Browse Structure Search Insert Empty Drop	16	InnoDB	utf8mb4_general_ci	16.0 KiB	-
managepassenger	★ Browse Structure Search Insert Empty Drop	17	InnoDB	utf8mb4_general_ci	16.0 KiB	-
payment	★ Browse Structure Search Insert Empty Drop	15	InnoDB	utf8mb4_general_ci	16.0 KiB	-
ticketbooking	★ Browse Structure Search Insert Empty Drop	14	InnoDB	utf8mb4_general_ci	32.0 KiB	-
6 tables	Sum	78	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B

Recent Favorites

Show all | Number of rows: 25 | Filter rows: Search this table

New
ams
New
airport
login
manageflight
managepassenger
payment
ticketbooking
cmc05
d1
information_schema
mysql
performance_schema
phpmyadmin
test

Extra options

airport_id	airport_name
ABC123	Sky Harbor Int.
DEF456	Heathrow Airport
GHI789	Changi Airport
JKL012	LAX International
MNO345	Dubai Int.
PQR678	Tokyo Haneda
STU901	O'Hare Int.
VWX234	Beijing Capital
YZA567	Frankfurt Airport
BCD890	Denver Int.
EFG123	JFK Int.
HIJ456	Sydney Airport
KLM789	Incheon Int.
NOP012	Hartsfield-Jackson
QRS345	DFW Int.

Show all | Number of rows: 25 | Filter rows: Search this table

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 flightcode	varchar(30)	utf8mb4_general_ci		No	None		More	
<input type="checkbox"/>	2 source	varchar(30)	utf8mb4_general_ci		No	None		More	
<input type="checkbox"/>	3 destination	varchar(30)	utf8mb4_general_ci		No	None		More	
<input type="checkbox"/>	4 takeoff	varchar(30)	utf8mb4_general_ci		No	None		More	
<input type="checkbox"/>	5 noofseat	varchar(30)	utf8mb4_general_ci		No	None		More	

Check all With selected: Browse Primary Index Fulltext

Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after noofseat

Indexes

No index defined!

phpMyAdmin

Recent Favorites

New
ams
New
airport
login
manageflight
managepassenger
payment
ticketbooking
cmcs05
d1
information_schema
mysql
performance_schema
phpmyadmin
test

Server: 127.0.0.1 » Database: ams » Table: manageflight

Browse Structure SQL Search Insert Export Import

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table

Extra options

flightcode	source	destination	takeoff	noofseat
FI-101	akjsjdh	andka	2023-11-09 12:30:00	220
101	Delhi	Australia	2023-11-10 15:45:00	120
FL123	New York	Los Angeles	2023-11-11 10:00:00	190
FL456	Los Angeles	Chicago	2023-11-09 12:30:00	180
FL789	Chicago	Miami	2023-11-10 15:45:00	220
FL987	Miami	Seattle	2023-11-11 10:00:00	190
FL654	Seattle	San Francisco	2023-11-12 14:30:00	210
FL321	San Francisco	Boston	2023-11-13 18:45:00	170
FL543	Boston	Denver	2023-11-14 08:30:00	230
FL876	Denver	Houston	2023-11-15 11:15:00	200
FL234	Houston	Las Vegas	2023-11-16 13:45:00	180
FL567	Las Vegas	Orlando	2023-11-17 16:00:00	210
FL890	Orlando	Dallas	2023-11-18 09:30:00	190
FL432	Dallas	Phoenix	2023-11-19 12:00:00	220
FL765	Phoenix	Atlanta	2023-11-20 14:15:00	200
FL098	Atlanta	New York	2023-11-21 17:30:00	180

- Drag to reorder.
- Click to mark/unmark.
- Double-click to copy column name.

localhost/phpmyadmin/index.php

phpMyAdmin

Server: 127.0.0.1 » Database: ams » Table: ticketbooking

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Trigger

Showing rows 0 - 13 (14 total, Query took 0.0004 seconds.)

SELECT * FROM `ticketbooking`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

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

Extra options

	id	name	flightcode	gender	passportnumber	amount	nationality
<input type="checkbox"/> Edit Copy Delete	1	Alice Johnson	FL123	Female	AB123456	250	USA
<input type="checkbox"/> Edit Copy Delete	10	Jack Wang	FL567	Male	ST567890	280	Brazil
<input type="checkbox"/> Edit Copy Delete	11	Katie Rodriguez	FL890	Female	UV123456	310	Argentina
<input type="checkbox"/> Edit Copy Delete	12	Leo Nguyen	FL432	Male	WX789012	260	France
<input type="checkbox"/> Edit Copy Delete	13	Mia Kim	FL765	Female	YZ901234	290	South Korea
<input type="checkbox"/> Edit Copy Delete	14	Nathan Patel	FL098	Male	AB345678	320	Vietnam
<input type="checkbox"/> Edit Copy Delete	15	Olivia Hernandez	FL111	Female	CD567890	270	India
<input type="checkbox"/> Edit Copy Delete	2	Bob Smith	FL456	Male	CD789012	300	Canada
<input type="checkbox"/> Edit Copy Delete	3	Charlie Brown	FL789	Male	EF345678	200	UK
<input type="checkbox"/> Edit Copy Delete	4	David Davis	FL987	Male	GH901234	350	Australia
<input type="checkbox"/> Edit Copy Delete	5	Eva Martinez	FL654	Female	IJ567890	400	Germany
<input checked="" type="checkbox"/> Console Copy Delete	6	Frank White	FL321	Male	KL123456	280	Japan

phpMyAdmin

Server: 127.0.0.1 » Database: ams » Table: payment

Browse Structure SQL Search Insert Export Import

Extra options

New payment_id amount

PAY001	100
PAY002	150
PAY003	200
PAY004	75
PAY005	120
PAY006	90
PAY007	180
PAY008	130
PAY009	160
PAY010	110
PAY011	190
PAY012	85
PAY013	140
PAY014	170
PAY015	200

Show all Number of rows: 25 Filter rows: Search this table

Recent Favorites

- New
- ams
- New
- airport
- login
- manageflight
- managepassenger
- payment
- ticketbooking
- cmcs05
- d1
- information_schema
- mysql
- performance_schema
- phpmyadmin
- test

phpMyAdmin

Server: 127.0.0.1 » Database: ams » Table: manageflight

Browse Structure SQL Search Insert Export Import

Extra options

New flightcode source destination takeoff noofseat

FI-101	akjsjdh	andka	kjdn	22
101	Delhi	Australia	12:00	120
FL123	New York	Los Angeles	2023-11-08 08:00:00	200
FL456	Los Angeles	Chicago	2023-11-09 12:30:00	180
FL789	Chicago	Miami	2023-11-10 15:45:00	220
FL987	Miami	Seattle	2023-11-11 10:00:00	190
FL654	Seattle	San Francisco	2023-11-12 14:30:00	210
FL321	San Francisco	Boston	2023-11-13 18:45:00	170
FL543	Boston	Denver	2023-11-14 08:30:00	230
FL876	Denver	Houston	2023-11-15 11:15:00	200
FL234	Houston	Las Vegas	2023-11-16 13:45:00	180
FL567	Las Vegas	Orlando	2023-11-17 16:00:00	210
FL890	Orlando	Dallas	2023-11-18 09:30:00	190
FL432	Dallas	Phoenix	2023-11-19 12:00:00	220
FL765	Phoenix	Atlanta	2023-11-20 14:15:00	200
FL098	Atlanta	New York	2023-11-21 17:30:00	180

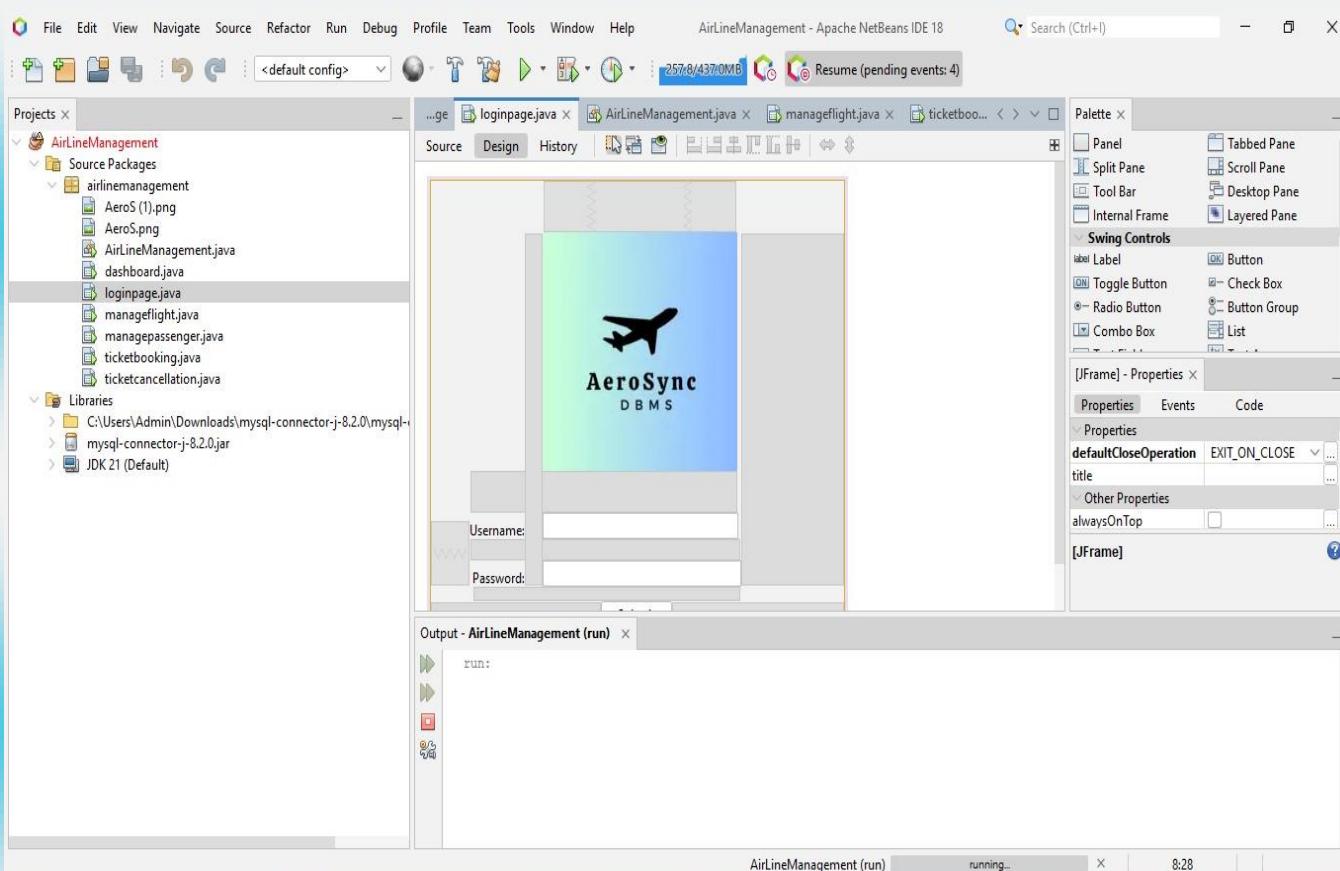
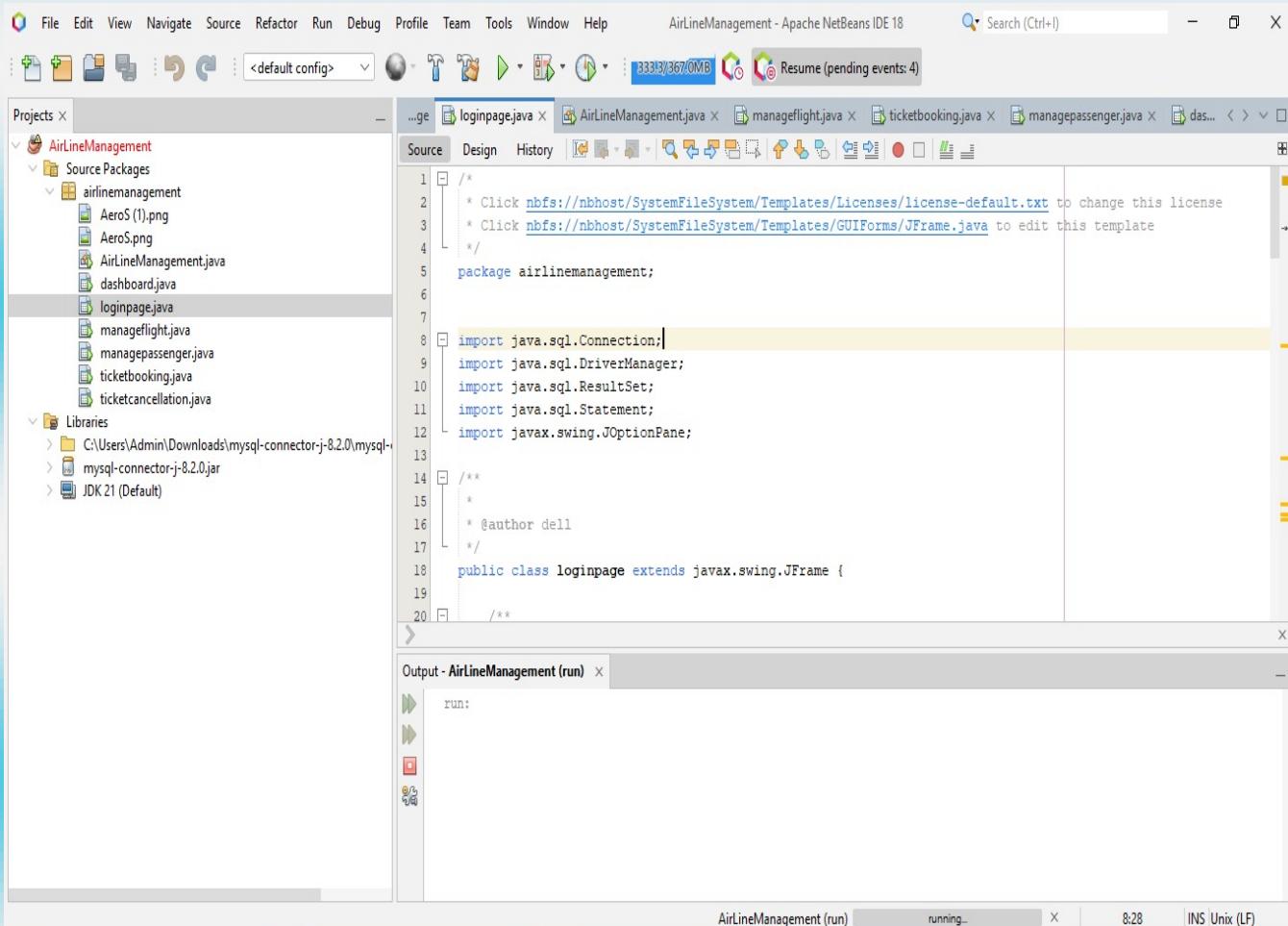
Show all Number of rows: 25 Filter rows: Search this table

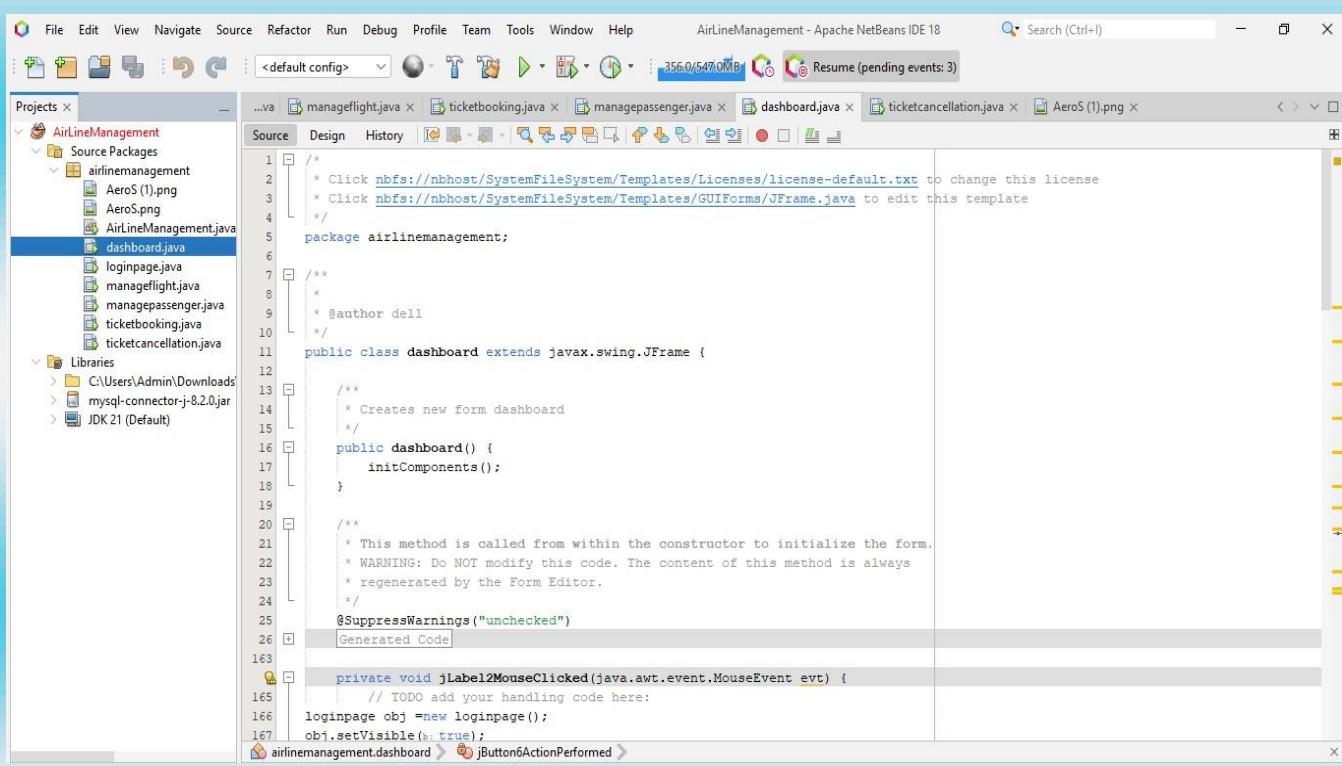
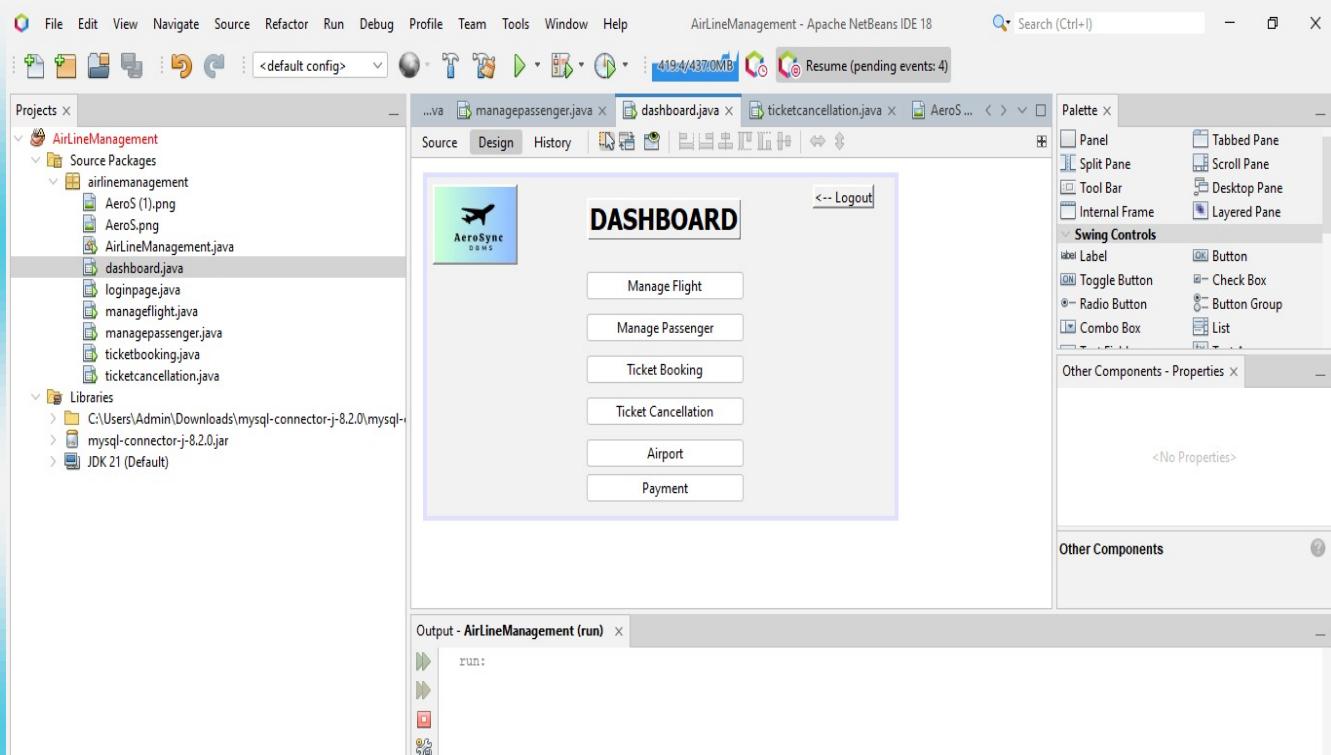
Query results operations

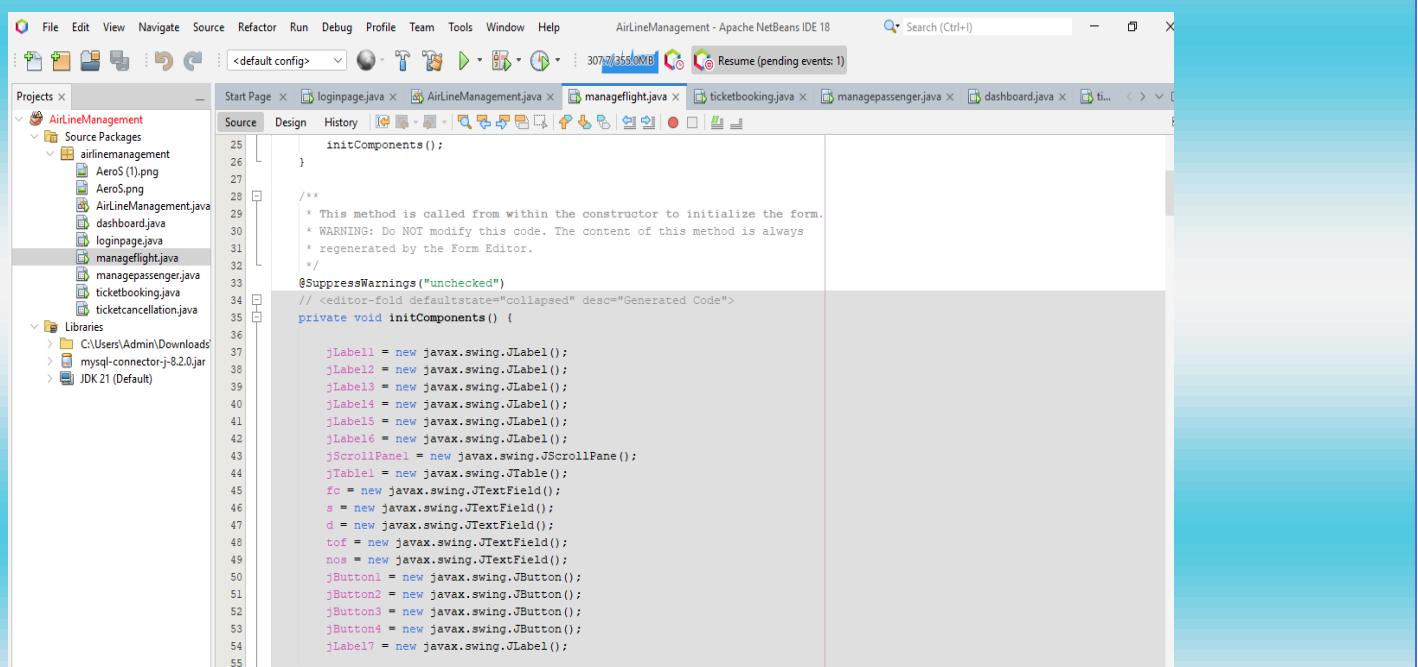
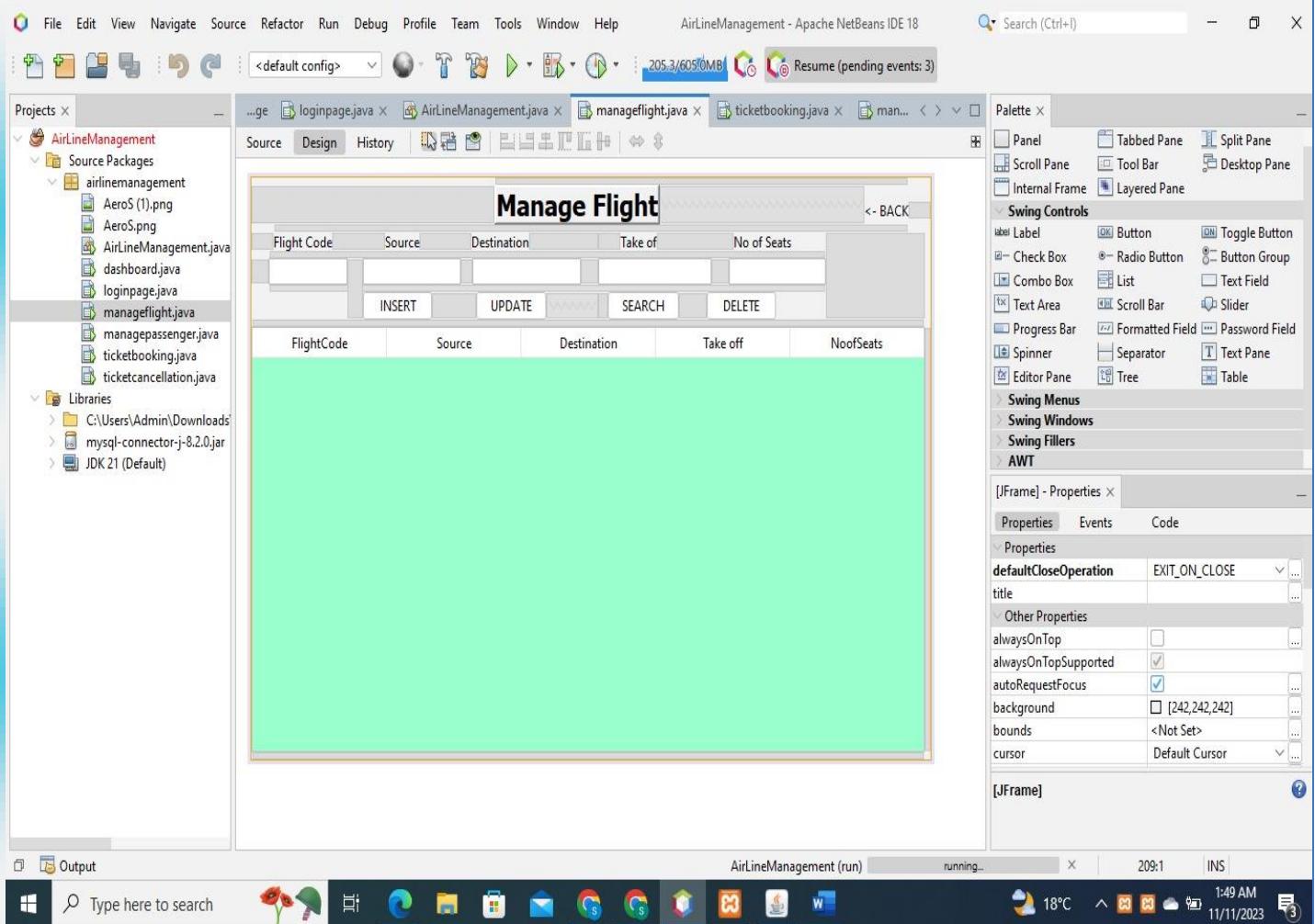
Recent Favorites

- New
- ams
- New
- airport
- login
- manageflight
- managepassenger
- payment
- ticketbooking
- cmcs05
- d1
- information_schema
- mysql
- performance_schema
- phpmyadmin
- test

4.2 Application with code







File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help AirLineManagement - Apache NetBeans IDE 18 Search (Ctrl+F) 2023/361.0MB Resume (pending events: 1)

Projects X Start Page X loginpage.java X AirLineManagement.java X manageflight.java X ticketbooking.java X managepassenger.java X dashboard.java X ti... X

AirLineManagement

Source Packages

airlinemanagement

AeroS (1).png
AeroS.png
AirLineManagement.java
dashboard.java
loginpage.java
manageflight.java
managepassenger.java
ticketbooking.java
ticketcancellation.java

Libraries

C:\Users\Admin\Downloads
mysql-connector-j-8.2.0.jar
JDK 21 (Default)

Source Design History

```
82     }
83   );
84   jScrollPane.setViewportView(view: jTable1);
85
86   jButton1.setText(text: "INSERT");
87   jButton1.addActionListener(new java.awt.event.ActionListener() {
88     public void actionPerformed(java.awt.event.ActionEvent evt) {
89       jButton1ActionPerformed(evt);
90     }
91   });
92
93   jButton2.setText(text: "UPDATE");
94   jButton2.addActionListener(new java.awt.event.ActionListener() {
95     public void actionPerformed(java.awt.event.ActionEvent evt) {
96       jButton2ActionPerformed(evt);
97     }
98   });
99
100  jButton3.setText(text: "SEARCH");
101  jButton3.addActionListener(new java.awt.event.ActionListener() {
102    public void actionPerformed(java.awt.event.ActionEvent evt) {
103      jButton3ActionPerformed(evt);
104    }
105  });
106
107  jButton4.setText(text: "DELETE");
108  jButton4.addActionListener(new java.awt.event.ActionListener() {
109    public void actionPerformed(java.awt.event.ActionEvent evt) {
110      jButton4ActionPerformed(evt);
111    }
112  });

```

Output

Type here to search 1:1 INS Unix (LF) 11/11/2023 1:45 PM

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help AirLineManagement - Apache NetBeans IDE 18 Search (Ctrl+F) 2947/361.0MB Resume (pending events: 1)

Projects X Start Page X loginpage.java X AirLineManagement.java X manageflight.java X ticketbooking.java X managepassenger.java X dashboard.java X ti... X

AirLineManagement

Source Packages

airlinemanagement

AeroS (1).png
AeroS.png
AirLineManagement.java
dashboard.java
loginpage.java
manageflight.java
managepassenger.java
ticketbooking.java
ticketcancellation.java

Libraries

C:\Users\Admin\Downloads
mysql-connector-j-8.2.0.jar
JDK 21 (Default)

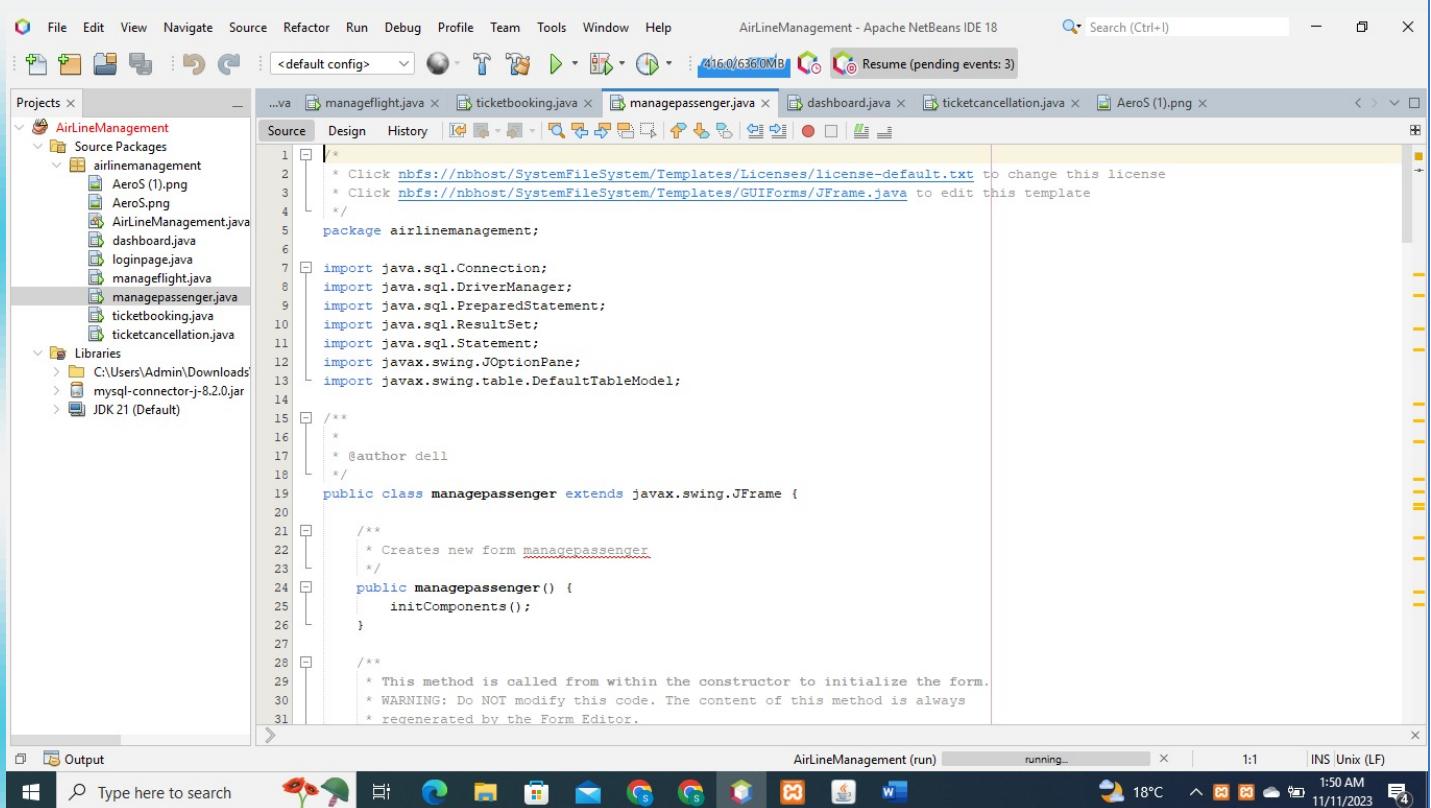
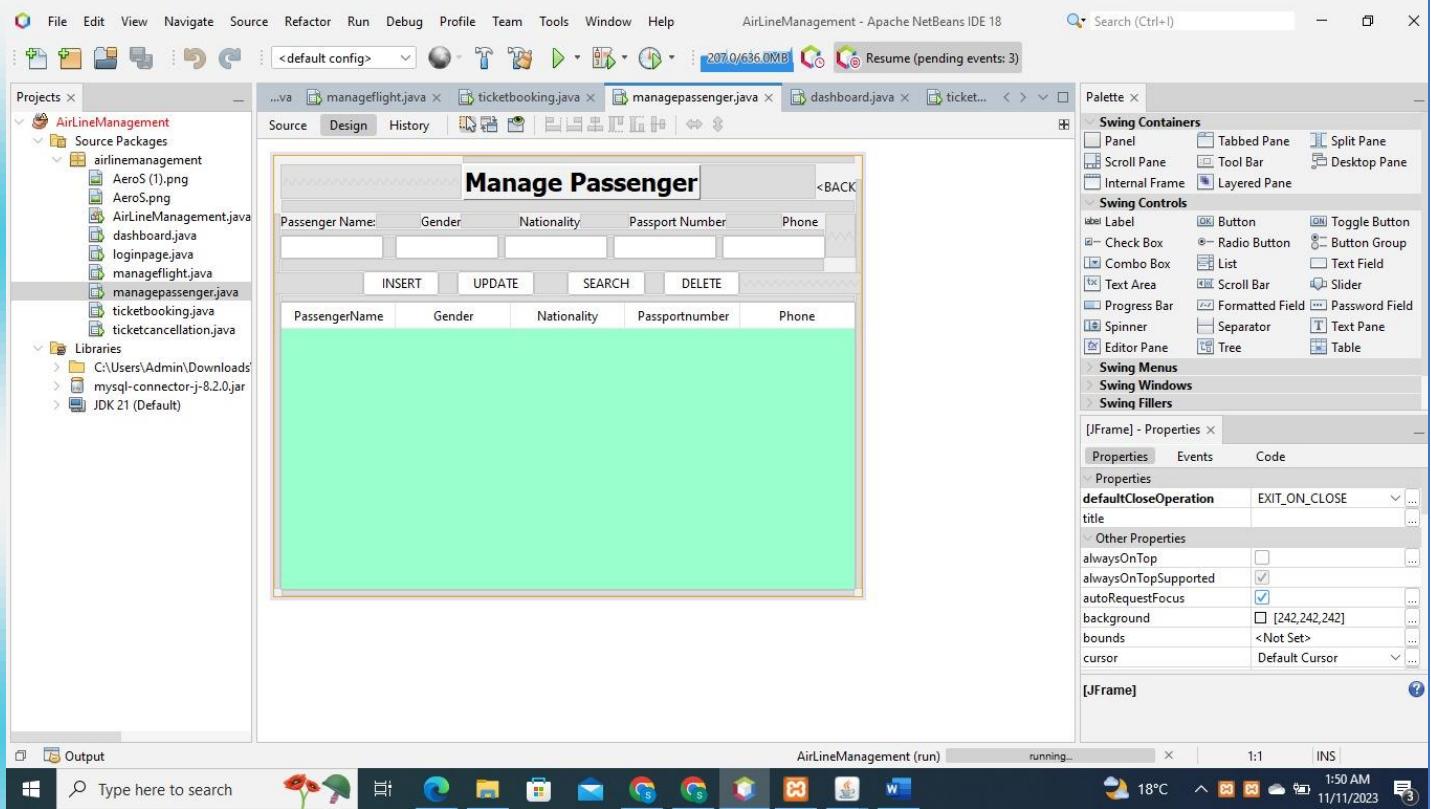
Source Design History

```
142         .addGroup(group: layout.createSequentialGroup()
143           .addComponent(component: jButton1)
144           .addGap(min: 46, pref: 46, max: 46)
145           .addComponent(component: jButton2)
146           .addPreferredGap(type: javax.swing.LayoutStyle.ComponentPlacement.RELATED, pref: javax.swing.GroupLayout.DEFAULT_SIZE)
147           .addGroup(group: layout.createSequentialGroup()
148             .addComponent(component: s)
149             .addPreferredGap(type: javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
150             .addComponent(component: d, min: javax.swing.GroupLayout.PREFERRED_SIZE, pref: 113, max: javax.swing.GroupLayout.DEFAULT_SIZE)
151             .addGap(min: 18, pref: 18, max: 18)))
152         .addGroup(group: layout.createParallelGroup(alignment: javax.swing.GroupLayout.Alignment.LEADING)
153           .addGroup(group: layout.createSequentialGroup()
154             .addGap(min: 23, pref: 23, max: 23)
155             .addComponent(component: jLabel15)
156             .addGap(min: 80, pref: 80, max: 80)
157             .addComponent(component: jLabel16))
158           .addGroup(group: layout.createSequentialGroup()
159             .addComponent(component: tof, min: javax.swing.GroupLayout.PREFERRED_SIZE, pref: 118, max: javax.swing.GroupLayout.DEFAULT_SIZE)
160             .addGap(min: 18, pref: 18, max: 18)
161             .addComponent(component: nos, min: javax.swing.GroupLayout.PREFERRED_SIZE, pref: 101, max: javax.swing.GroupLayout.DEFAULT_SIZE)
162             .addGroup(group: layout.createSequentialGroup()
163               .addGap(min: 10, pref: 10, max: 10)
164               .addComponent(component: jButton3)
165               .addGap(min: 29, pref: 29, max: 29)
166               .addComponent(component: jButton4)))
167             .addGap(min: 103, pref: 103, max: 103)
168             .addComponent(component: jScrollPane, alignment: javax.swing.GroupLayout.Alignment.TRAILING)
169             .addContainerGap())
170           .addGroup(group: layout.createSequentialGroup()
171             .addGap(min: 253, pref: 253, max: 253)
172           .addContainerGap())

```

Output

Type here to search 1:1 INS Unix (LF) 11/11/2023 1:45 PM



AirLineManagement - Apache NetBeans IDE 18

Source ticketbooking.java

```
1  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
2  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
3  */
4  package airlinemangement;
5
6  import java.sql.Connection;
7  import java.sql.DriverManager;
8  import java.sql.PreparedStatement;
9  import java.sql.ResultSet;
10 import java.sql.Statement;
11 import javax.swing.JOptionPane;
12 import javax.swing.table.DefaultTableModel;
13
14 /**
15  *
16  * @author dell
17  */
18 public class ticketbooking extends javax.swing.JFrame {
19
20     /**
21      * Creates new form ticketbooking
22      */
23     public ticketbooking() {
24         initComponents();
25     }
26
27     /**
28      * This method is called from within the constructor to initialize the form.
29      * WARNING: Do NOT modify this code. The content of this method is always
30      * regenerated by the Form Editor.
31     */
32 }
```

Output

AirLineManagement (run) running... 1:1 INS Unix (LF)

18°C 1:51 AM 11/11/2023

AirLineManagement - Apache NetBeans IDE 18

Source ticketcancellation.java

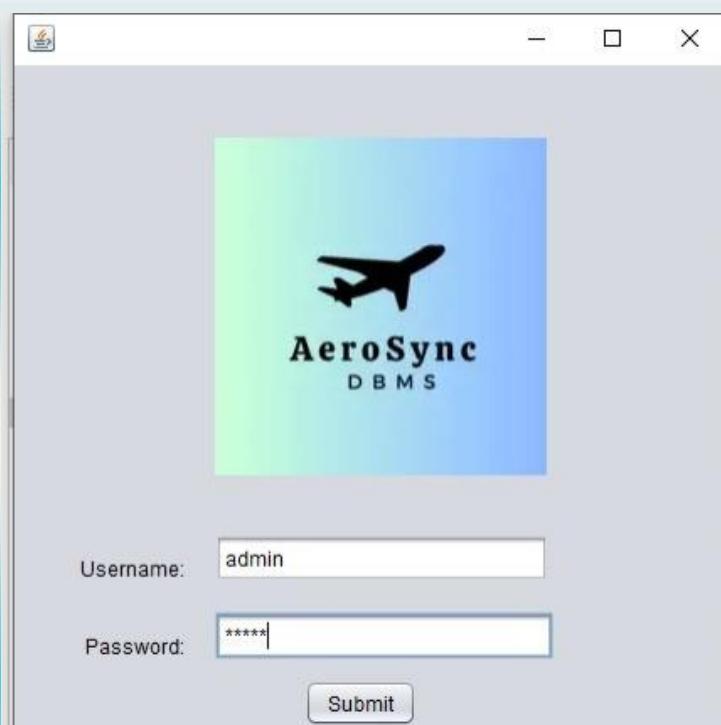
```
4  */
5  package airlinemangement;
6
7  import java.sql.Connection;
8  import java.sql.DriverManager;
9  import java.sql.PreparedStatement;
10 import java.sql.Statement;
11 import javax.swing.JOptionPanel;
12
13 /**
14  *
15  * @author dell
16  */
17 public class ticketcancellation extends javax.swing.JFrame {
18
19     /**
20      * Creates new form ticketcancellation
21      */
22     public ticketcancellation() {
23         initComponents();
24     }
25
26     /**
27      * This method is called from within the constructor to initialize the form.
28      * WARNING: Do NOT modify this code. The content of this method is always
29      * regenerated by the Form Editor.
30      */
31     @SuppressWarnings("unchecked")
32     // Generated Code
33
34     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
```

Output

AirLineManagement (run) running... 1:1 INS Unix (LF)

18°C 1:52 AM 11/11/2023

4.3 Snapshots of running of application





The screenshot shows a "Manage Flight" window. At the top, there are five input fields labeled "Flight Code", "Source", "Destination", "Take off", and "No of Seats", each with an associated text input box. Below these are four buttons: "INSERT", "UPDATE", "SEARCH" (which is highlighted in blue), and "DELETE". To the right of the search button is a link "[-> BACK](#)". At the bottom is a data grid table with columns: "FlightCode", "Source", "Destination", "Take off", and "NoofSeats". The table contains 20 rows of flight information.

FlightCode	Source	Destination	Take off	NoofSeats
FL-101	akjsjdh	andka	kjdn	22
101	Delhi	Australia	12:00	120
FL123	New York	Los Angeles	2023-11-08 08:00:00	200
FL456	Los Angeles	Chicago	2023-11-09 12:30:00	180
FL789	Chicago	Miami	2023-11-10 15:45:00	220
FL987	Miami	Seattle	2023-11-11 10:00:00	190
FL654	Seattle	San Francisco	2023-11-12 14:30:00	210
FL321	San Francisco	Boston	2023-11-13 18:45:00	170
FL543	Boston	Denver	2023-11-14 08:30:00	230
FL876	Denver	Houston	2023-11-15 11:15:00	200
FL234	Houston	Las Vegas	2023-11-16 13:45:00	180
FL567	Las Vegas	Orlando	2023-11-17 16:00:00	210
FL890	Orlando	Dallas	2023-11-18 09:30:00	190
FL432	Dallas	Phoenix	2023-11-19 12:00:00	220
FL765	Phoenix	Atlanta	2023-11-20 14:15:00	200
FL098	Atlanta	New York	2023-11-21 17:30:00	180

Manage Passenger <BACK

Passenger Name:	Gender	Nationality	Passport Number	Phone
<input type="text"/>				

Buttons: INSERT, UPDATE, SEARCH, DELETE

PassengerName	Gender	Nationality	Passportnumber	Phone
John Doe	Male	USA	AB123456	123-456-7890
Lucas Nguyen	Male	Vietnam	AB345678	678-901-2345
Isabella Patel	Female	India	CD567890	789-012-3456
Jane Smith	Female	Canada	CD789012	987-654-3210
Mike Johnson	Male	UK	EF345678	456-789-0123
Emily Davis	Female	Australia	GH901234	789-012-3456
David Brown	Male	Germany	IJ567890	012-345-6789
Sophia Lee	Female	Japan	KL123456	234-567-8901
Daniel Kim	Male	South Korea	MN789012	567-890-1234
Olivia Wang	Female	China	OP345678	890-123-4567
TAHA	MALE	PAKISTANI	PK9289Y1	12831231
Liam Martinez	Male	Mexico	QR901234	123-234-5678
Ava Garcia	Female	Brazil	ST567890	234-567-8901
Noah Hernandez	Male	Argentina	UV123456	345-678-9012
Emma Rodriguez	Female	France	WX789012	456-789-0123

Manage Passenger <BACK

Passenger Name:	Gender	Nationality	Passport Number	Phone
<input type="text" value="Johnny"/>	<input type="text" value="Male"/>	<input type="text" value="USA"/>	<input type="text" value="AB123767"/>	<input type="text" value="12258965385"/>

Buttons: INSERT, UPDATE, SEARCH, DELETE

PassengerName	Gender	Nationality	Passportnumber	Phone
John Doe	Male	USA	AB123456	123-456-7890
Lucas Nguyen	Male	Vietnam	AB345678	678-901-2345
Isabella Patel	Female	India	CD567890	789-012-3456
Jane Smith	Female	Canada	CD789012	987-654-3210
Mike Johnson	Male	UK	EF345678	456-789-0123
Emily Davis	Female	Australia	GH901234	789-012-3456
David Brown	Male	Germany	IJ567890	012-345-6789
Sophia Lee	Female	Japan	KL123456	234-567-8901
Daniel Kim	Male	South Korea	MN789012	567-890-1234
Olivia Wang	Female	China	OP345678	890-123-4567
TAHA	MALE	PAKISTANI	PK9289Y1	12831231
Liam Martinez	Male	Mexico	QR901234	123-234-5678
Ava Garcia	Female	Brazil	ST567890	234-567-8901
Noah Hernandez	Male	Argentina	UV123456	345-678-9012
Emma Rodriguez	Female	France	WX789012	456-789-0123

Message

Data inserted Successfully!

 OK

TICKET BOOKING

<BACK

Passengerid	Passenger name	Flight Code	Gender	Passportnumber	Amount	nationality
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="SAVE"/>			<input type="button" value="RESET"/>		<input type="button" value="SEARCH"/>	

Passenger...	Passenger...	Flight Code	Gender	Passport N...	Amout	Nationality
1	Alice Johns...	FL123	Female	AB123456	250	USA
10	Jack Wang	FL567	Male	ST567890	280	Brazil
11	Katie Rodri...	FL890	Female	UV123456	310	Argentina
12	Leo Nguyen	FL432	Male	WX789012	260	France
13	Mia Kim	FL765	Female	YZ901234	290	South Korea
14	Nathan Patel	FL098	Male	AB345678	320	Vietnam
15	Olivia Hern...	FL111	Female	CD567890	270	India
2	Bob Smith	FL456	Male	CD789012	300	Canada
3	Charlie Bro...	FL789	Male	EF345678	200	UK
4	David Davis	FL987	Male	GH901234	350	Australia
5	Eva Martinez	FL654	Female	IJ567890	400	Germany
6	Frank White	FL321	Male	KL123456	280	Japan
7	Grace Lee	FL543	Female	MN789012	320	South Korea
8	Henry Kim	FL876	Male	OP345678	270	China
9	Isabel Garc...	FL234	Female	QR901234	330	Mexico

TICKET BOOKING

<BACK

Passengerid	Passenger name	Flight Code	Gender	Passportnumber	Amount	nationality
<input type="text"/> 99	<input type="text"/> Isabel	<input type="text"/> FL888	<input type="text"/> Male	<input type="text"/> QR901234	<input type="text"/> 330	<input type="text"/> Mexico
<input type="button" value="SAVE"/>			<input type="button" value="RESET"/>		<input type="button" value="SEARCH"/>	

Message



Data inserted Succesfully!

Passenger...	Passenger...	out	Nationality
1	Alice Johns...	USA	USA
10	Jack Wang	Brazil	Brazil
11	Katie Rodri...	Argentina	Argentina
12	Leo Nguyen	France	France
13	Mia Kim	South Korea	South Korea
14	Nathan Patel	Vietnam	Vietnam
15	Olivia Hern...	India	India
2	Bob Smith	Canada	Canada
3	Charlie Bro...	UK	UK
4	David Davis	Australia	Australia
5	Eva Martinez	Germany	Germany
6	Frank White	Japan	Japan
7	Grace Lee	South Korea	South Korea
8	Henry Kim	China	China
9	Isabel Garc...	Mexico	Mexico

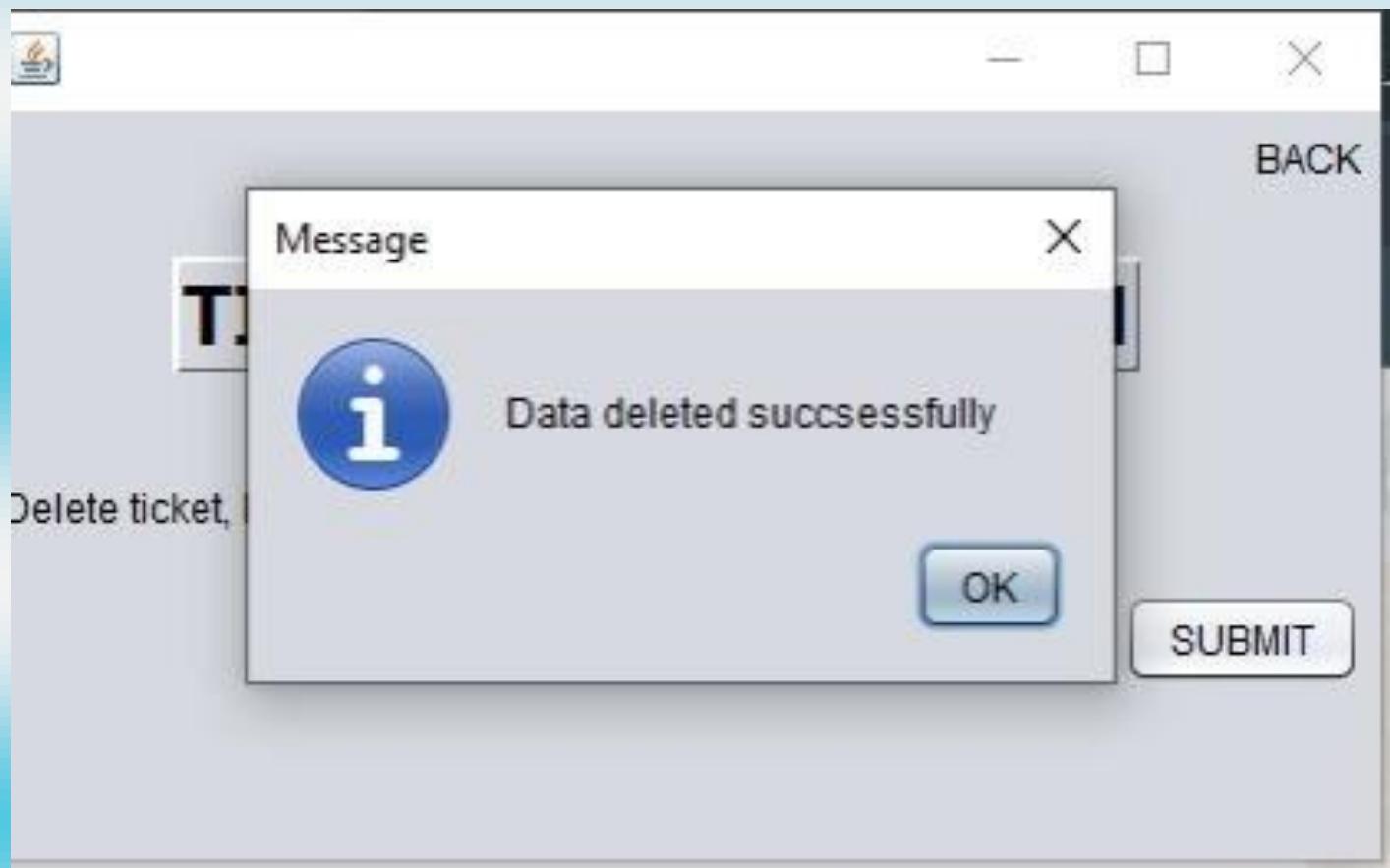
 - □ ×

BACK

TICKET CANCELLATION

Delete ticket, Enter Ticket ID:

SUBMIT



CHAPTER 5 REFERENCES

1. Java and Java Swing:

- Oracle Java Tutorials:

[<https://docs.oracle.com/javase/tutorial/>](https://docs.oracle.com/javase/tutorial/)

- GeeksforGeeks Java Swing Tutorial: [<https://www.geeksforgeeks.org/java-swing-set-1-set-2/>](https://www.geeksforgeeks.org/java-swing-set-1-set-2/)

2. Database Management System (DBMS):

- W3Schools SQL Tutorial:

[<https://www.w3schools.com/sql/>](https://www.w3schools.com/sql/)

- Tutorialspoint DBMS Tutorial:

[<https://www.tutorialspoint.com/dbms/>](https://www.tutorialspoint.com/dbms/)

3. *MySQL:

- MySQL Documentation:

[<https://dev.mysql.com/doc/>](https://dev.mysql.com/doc/)

- MySQL Tutorial by TutorialsPoint:

[<https://www.tutorialspoint.com/mysql/index.htm>](https://www.tutorialspoint.com/mysql/index.htm)