

# **COURSE: JAVA FULL STACK DEVELOPMENT**

**JAVA FSD PROJECT PHASE 2**

**PROJECT NAME: AIRWAY FLYAWAY**  
**SIMPLILEARN**

**Developed by,  
GANAVI M**

# Table Contents

Sl.NO	CONTENTS	Pg No
1	Project Objective	3
2	Problem Statement	3
3	Sprints Planning and task completion	3
4	Tools Used for development	4
5	GitHub repository	4
6	Program and Database Screenshots	5
7	Conclusion	24

## **1. PROJECT OBJECTIVE:**

As a Full Stack Developer, design and develop an airline booking portal named as AirWay FlyAway. Use the GitHub repository to manage the project artifacts.

## **2. PROBLEM STATEMENT:**

A search form in the homepage to allow entry of travel details, like the date of travel, source, destination, and the number of persons.

- ✓ Based on the travel details entered, it will show the available flights with their ticket prices.
- ✓ Once a person selects a flight to book, they will be taken to a register page where they must fill in their personal details. In the next page, they are shown the flight details of the flight that they are booking, and the payment is done via a dummy payment gateway.

For the above features to work, there will be an admin backend with the following features:

- ✓ An admin login page where the admin can change the password after login, if he wishes
- ✓ A master list of places for source and destination
- ✓ A master list of airlines
- ✓ A list of flights where each flight has a source, destination, airline, and ticket price

## **3. SPRINTS PLANNING AND TASK COMPLETION:**

The project is planned to be completed in 3 sprint. Tasks assumed to be completed in the sprint are:

### **Sprint 1:**

1. Creating the flow of the application
2. Initializing git repository to track changes as development progresses.
3. Writing the Java program to fulfil the requirements of the project.
4. Design Classes and Methods
5. Exceptions, Hibernation, Maven
6. Code Data

**Sprint 2:**

1. List Entities
2. Creating tables
3. Adding Details of flights, passengers
4. Add LiveServer

**Sprint 3:**

1. Server side coding
2. Testing the Java program with different kinds of User input
3. Pushing code to GitHub.
4. Creating this specification document highlighting application capabilities, appearance
5. Use String Formatting to display the output in a standard manner.

**4. TOOLS USED FOR DEVELOPMENT:**

1. Eclipse IDE
2. MySQL Relation Database Management System
3. Apache Tom-Cat 8.5 server
4. Java Servlets
5. Java Server Pages (JSP)
6. HTML
7. Apache Maven
8. JDBC Java Database Connectivity

**5. PROJECT GIT REPOSITORY:**

Link to the GitHub Repository -

<https://github.com/GanaviM/Airways-flyAway-Phase2-project>

## 6. SCREENSHOTS OF PROGRAM OUTPUT:

### DATABASE:

```
MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| airline   |
| class1   |
| flyaway   |
| information_schema |
| mysql     |
| performance_schema |
| sys       |
+-----+
7 rows in set (0.00 sec)

mysql> use airline;
Database changed
mysql> show tables;
+-----+
| Tables_in_airline |
+-----+
| flight      |
| flight_passenger |
| login       |
| passenger   |
+-----+
4 rows in set (0.00 sec)

mysql> select * from flight;
+-----+
| Tables_in_airline |
+-----+
| flight      |
| flight_passenger |
| login       |
| passenger   |
+-----+
4 rows in set (0.00 sec)

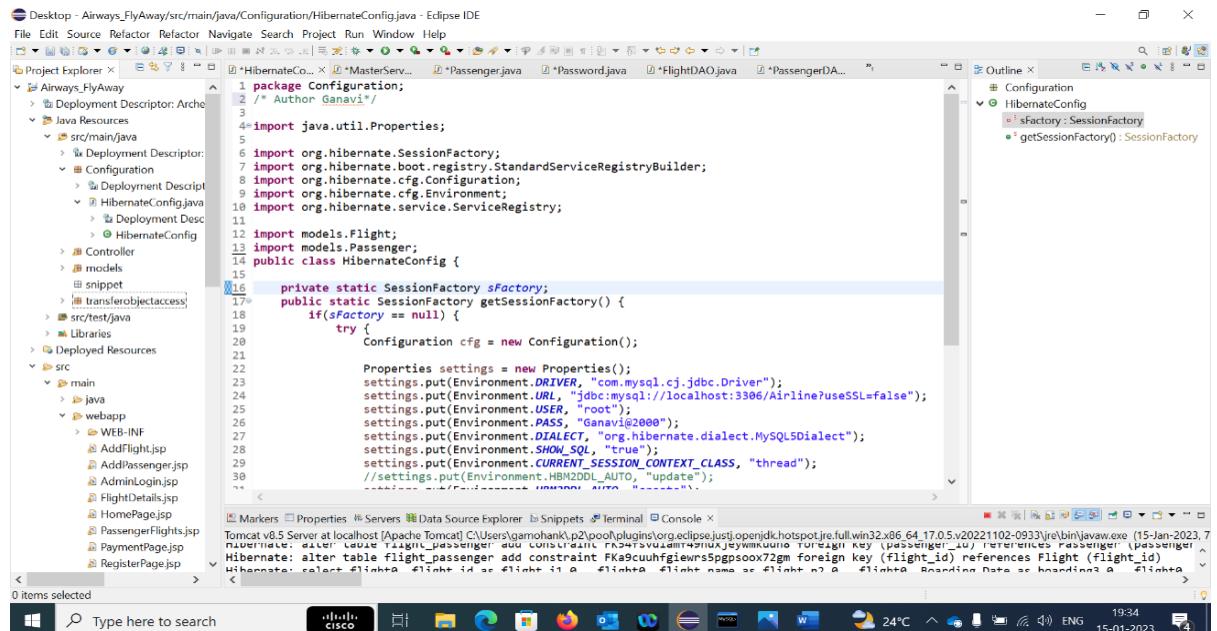
mysql> select * from flight;
+-----+
| flight_id | flight_name | Boarding_Date | flight_number | Source | Ticket_price | Destination |
+-----+
| 1 | AirIndia | 2023-01-16 | 1239 | Bengaluru | 2846 | Delhi |
| 2 | Indigo | 2023-01-16 | 2341 | Mumbai | 4532 | Kolkata |
| 3 | Vistara | 2023-01-17 | 7654 | Mysuru | 3425 | Chennai |
| 4 | Go First | 2023-01-17 | 2543 | Trivandrum | 2134 | Bengaluru |
| 5 | AirIndia | 2023-01-17 | 1239 | Bengaluru | 3241 | Chennai |
| 6 | AirIndia | 2023-01-16 | 1239 | Bengaluru | 2846 | Delhi |
+-----+
6 rows in set (0.00 sec)

mysql> select * from passenger;
+-----+
| passenger_id | passenger_age | passenger_mob | passenger_email | passenger_fname | passenger_lname |
+-----+
| 1 | 45 | 9844858343 | aryar54@hotmail.com | arya | r |
| 2 | 38 | 6545678443 | surabhi21@gmail.com | Surabhi | kumar |
+-----+
2 rows in set (0.00 sec)

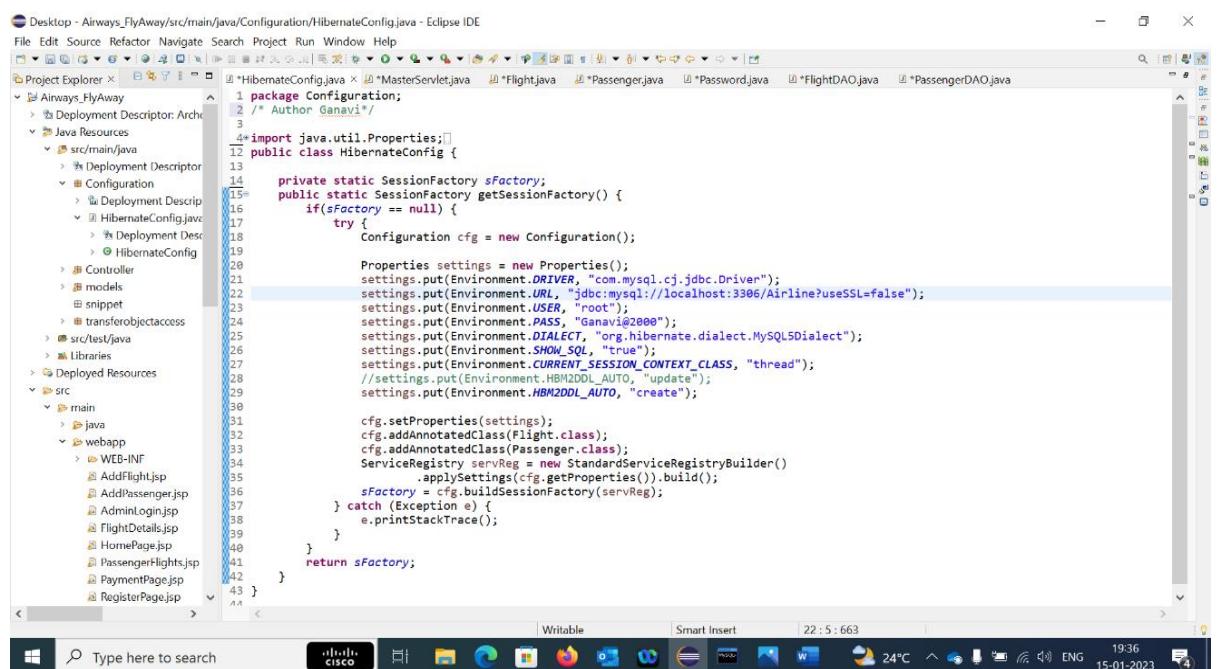
mysql> select * from flight_passenger;
+-----+
| flight_id | passenger_id |
+-----+
| 6 | 1 |
| 6 | 2 |
+-----+
2 rows in set (0.00 sec)
```

## CODING:

### 1. Hibernate Configuration

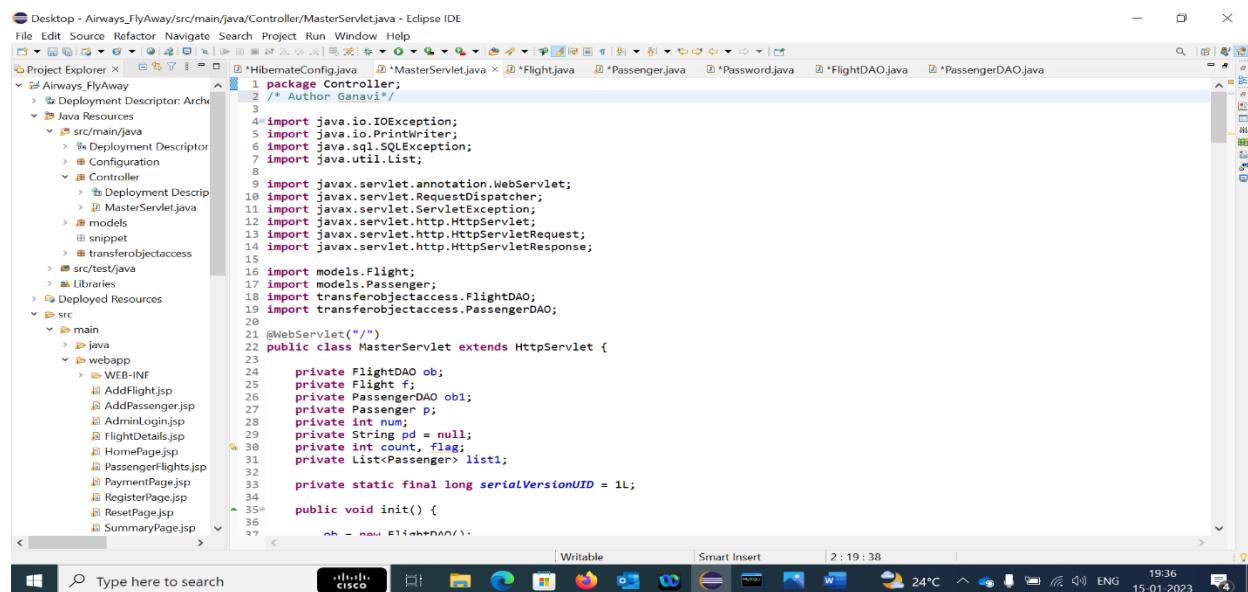


```
1 package Configuration;
2 /* Author Ganavi*/
3
4 import java.util.Properties;
5
6 import org.hibernate.SessionFactory;
7 import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
8 import org.hibernate.cfg.Configuration;
9 import org.hibernate.cfg.Environment;
10 import org.hibernate.service.ServiceRegistry;
11
12 import models.Flight;
13 import models.Passenger;
14 public class HibernateConfig {
15
16     private static SessionFactory sFactory;
17     public static SessionFactory getSessionFactory() {
18         if(sFactory == null) {
19             try {
20                 Configuration cfg = new Configuration();
21
22                 Properties settings = new Properties();
23                 settings.put(Environment.DRIVER, "com.mysql.cj.jdbc.Driver");
24                 settings.put(Environment.URL, "jdbc:mysql://localhost:3306/Airline?useSSL=false");
25                 settings.put(Environment.USER, "root");
26                 settings.put(Environment.PASS, "Ganavi@2000");
27                 settings.put(Environment.DIALECT, "org.hibernate.dialect.MySQL5Dialect");
28                 settings.put(Environment.SHOW_SQL, "true");
29                 //settings.put(Environment.HBM2DDL_AUTO, "update");
30             } catch (Exception e) {
31                 e.printStackTrace();
32             }
33         }
34         return sFactory;
35     }
36 }
37 }
```

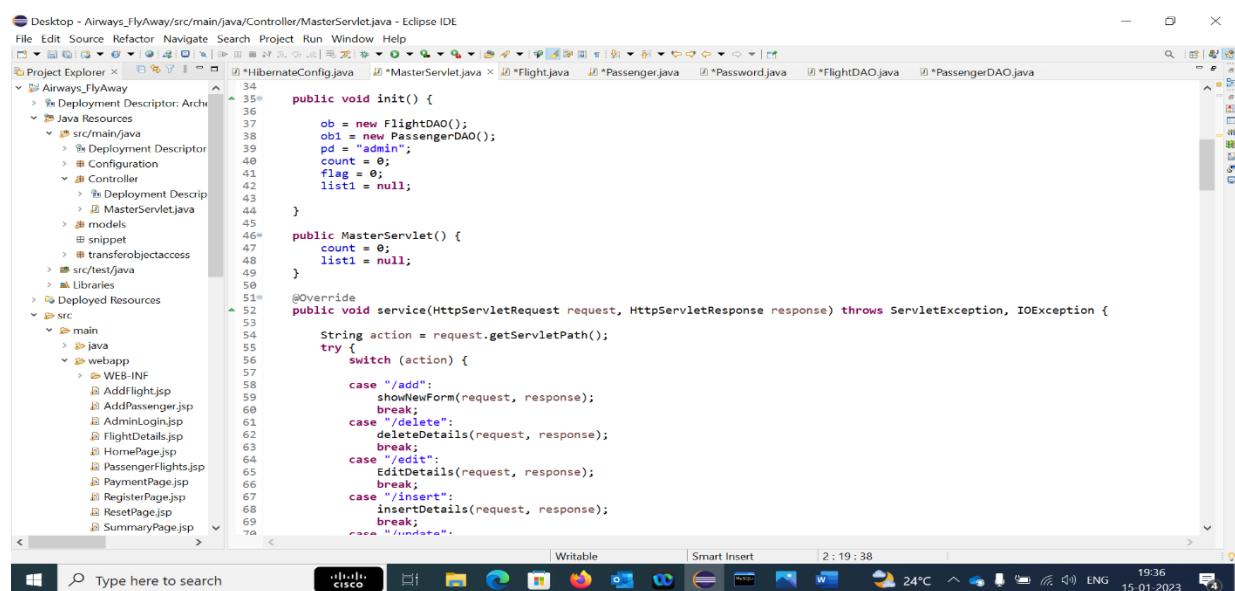


```
1 package Configuration;
2 /* Author Ganavi*/
3
4 import java.util.Properties;
5
6 import org.hibernate.SessionFactory;
7 import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
8 import org.hibernate.cfg.Configuration;
9 import org.hibernate.cfg.Environment;
10 import org.hibernate.service.ServiceRegistry;
11
12 import models.Flight;
13 import models.Passenger;
14 public class HibernateConfig {
15
16     private static SessionFactory sFactory;
17     public static SessionFactory getSessionFactory() {
18         if(sFactory == null) {
19             try {
20                 Configuration cfg = new Configuration();
21
22                 Properties settings = new Properties();
23                 settings.put(Environment.DRIVER, "com.mysql.cj.jdbc.Driver");
24                 settings.put(Environment.URL, "jdbc:mysql://localhost:3306/Airline?useSSL=false");
25                 settings.put(Environment.USER, "root");
26                 settings.put(Environment.PASS, "Ganavi@2000");
27                 settings.put(Environment.DIALECT, "org.hibernate.dialect.MySQL5Dialect");
28                 settings.put(Environment.SHOW_SQL, "true");
29                 //settings.put(Environment.HBM2DDL_AUTO, "update");
30                 settings.put(Environment.HBM2DDL_AUTO, "create");
31
32                 cfg.setProperties(settings);
33                 cfg.addAnnotatedClass(Flight.class);
34                 cfg.addAnnotatedClass(Passenger.class);
35                 ServiceRegistry servReg = new StandardServiceRegistryBuilder()
36                     .applySettings(cfg.getProperties()).build();
37                 sFactory = cfg.buildSessionFactory(servReg);
38             } catch (Exception e) {
39                 e.printStackTrace();
40             }
41         }
42         return sFactory;
43     }
44 }
```

## 2. Master Servlet



```
1 package Controller;
2 /* Author Ganav */
3
4 import java.io.IOException;
5 import java.io.PrintWriter;
6 import java.sql.SQLException;
7 import java.util.List;
8
9 import javax.servlet.annotation.WebServlet;
10 import javax.servlet.RequestDispatcher;
11 import javax.servlet.ServletException;
12 import javax.servlet.http.HttpServlet;
13 import javax.servlet.http.HttpServletRequest;
14 import javax.servlet.http.HttpServletResponse;
15
16 import models.Flight;
17 import models.Passenger;
18 import transferobjectaccess.FlightDAO;
19 import transferobjectaccess.PassengerDAO;
20
21 @WebServlet("/")
22 public class MasterServlet extends HttpServlet {
23
24     private FlightDAO ob;
25     private Flight f;
26     private PassengerDAO ob1;
27     private Passenger p;
28     private int num;
29     private String pd = null;
30     private int count, flag;
31     private List<Passenger> list1;
32
33     private static final long serialVersionUID = 1L;
34
35     public void init() {
36         ob = new FlightDAO();
37         ob1 = new PassengerDAO();
38         pd = "admin";
39         count = 0;
40         flag = 0;
41         list1 = null;
42     }
43
44     public MasterServlet() {
45         count = 0;
46         list1 = null;
47     }
48
49     @Override
50     public void service(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
51
52         String action = request.getServletPath();
53         try {
54             switch (action) {
55                 case "/add":
56                     showForm(request, response);
57                     break;
58                 case "/delete":
59                     deleteDetails(request, response);
60                     break;
61                 case "/edit":
62                     EditDetails(request, response);
63                     break;
64                 case "/insert":
65                     insertDetails(request, response);
66                     break;
67                 case "/update":
68                     updateDetails(request, response);
69                     break;
70             }
71         }
72     }
73 }
```



```
34
35     public void init() {
36
37         ob = new FlightDAO();
38         ob1 = new PassengerDAO();
39         pd = "admin";
40         count = 0;
41         flag = 0;
42         list1 = null;
43     }
44
45     public MasterServlet() {
46         count = 0;
47         list1 = null;
48     }
49
50     @Override
51     public void service(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
52
53         String action = request.getServletPath();
54         try {
55             switch (action) {
56                 case "/add":
57                     showForm(request, response);
58                     break;
59                 case "/delete":
60                     deleteDetails(request, response);
61                     break;
62                 case "/edit":
63                     EditDetails(request, response);
64                     break;
65                 case "/insert":
66                     insertDetails(request, response);
67                     break;
68                 case "/update":
69                     updateDetails(request, response);
70             }
71         }
72     }
73 }
```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the `Airways_FlyAway` project structure. It includes Java Resources (src/main/java), Configuration, Controller (Deployment Descriptor, MasterServlet.java), models, snippet, transferobjectaccess, src/test/java, and Libraries.
- Java Editor:** The current file is `MasterServlet.java`. The code implements a `Servlet` with methods for handling various requests like `/edit`, `/insert`, `/update`, `/reset`, `/register`, `/storage`, `/find`, `/login`, `/passenger`, `/insertPassenger`, `/deletePassenger`, and `/booking`.
- Bottom Bar:** Includes tabs for Writable, Smart Insert, and a status bar showing 2 : 19 : 38, 19:36, 15-01-2023, and ENG.

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/Controller/MasterServlet.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Project Explorer:** Shows the project structure for "Airways\_FlyAway".
- Code Editor:** Displays the Java code for `MasterServlet.java`. The code handles booking requests, retrieves all details from a database, and inserts passenger details into the database.
- Bottom Bar:** Includes tabs for Writable, Smart Insert, and a status bar showing 2 : 19 : 38, 19:36, ENG, and a date/time stamp of 15-01-2023.
- System Tray:** Shows icons for battery, signal strength, temperature, and network connection.

```
99         case "booking":
100             BookingDetails(request, response);
101             break;
102         default:
103             showAllDetails(response, request);
104             break;
105         }
106     } catch (Exception e) {
107         e.printStackTrace();
108     }
109 }
110 }
111 }
112
113 private void BookingDetails(HttpServletRequest request, HttpServletResponse response)
114     throws ServletException, IOException {
115
116     List<Passenger> list = ob1.getAllDetails();
117     list1 = list;
118     request.setAttribute("list", list);
119     RequestDispatcher rd = request.getRequestDispatcher("RegisterPage.jsp");
120     rd.forward(request, response);
121 }
122
123 private void passengerInsertDetails(HttpServletRequest request, HttpServletResponse response) throws IOException {
124
125     if (count > num)
126         response.sendRedirect("HomePage.jsp");
127     else {
128         String fname = request.getParameter("fname");
129         String lname = request.getParameter("lname");
130         long contact = Long.parseLong(request.getParameter("contact"));
131         int age = Integer.parseInt(request.getParameter("age"));
132         String email = request.getParameter("email");
133         Passenger p = new Passenger(fname, lname, age, contact, email);
134         insertDetails(p);
135     }
136 }
```



The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/Controller/MasterServlet.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Project Explorer:** Shows the project structure:
  - src/main/java:
    - Deployment Descriptor
    - Java Resources
      - Configuration
      - Controller
        - Deployment Descriptor
        - MasterServlet.java
      - models
      - snippet
      - transferobjectaccess
    - src/test/java
    - Libraries
  - Deployed Resources
  - src
    - main
      - java
      - webapp
        - WEB-INF
        - AddFlight.jsp
        - AddPassenger.jsp
        - AdminLogin.jsp
        - FlightDetails.jsp
        - HomePage.jsp
        - PassengerFlights.jsp
        - PaymentPage.jsp
        - RegisterPage.jsp
        - ResetPage.jsp
        - SummaryPage.jsp
- Code Editor:** Displays the Java code for `MasterServlet.java`. The code handles various HTTP requests for passenger management, flight details, and payment.
- Bottom Bar:** Shows the current file as `MasterServlet.java`, along with other tabs like `Flight.java`, `Passenger.java`, etc. It also displays the current time as 2:19:38 and system status like 24°C, ENG, and 15-01-2023.
- Search Bar:** Type here to search

Desktop - Airways\_FlyAway/src/main/java/Controller/MasterServlet.java - Eclipse IDE

```
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer X HibernateConfig.java MasterServlet.java Flight.java Passenger.java Password.java FlightDAO.java PassengerDAO.java
160     private void updateDetails(HttpServletRequest request, HttpServletResponse response)
161         throws ServletException, IOException {
162
163     int fid = Integer.parseInt(request.getParameter("fid"));
164     int fnumber = Integer.parseInt(request.getParameter("fnumber"));
165     String fname = request.getParameter("fname");
166     String forigin = request.getParameter("forigin");
167     String ftarget = request.getParameter("ftarget");
168     String date = request.getParameter("fdate");
169     float fprice = Float.parseFloat(request.getParameter("fprice"));
170     Flight fl = new Flight(fid, fnumber, fname, forigin, ftarget, date, fprice);
171     ob.updateFlight(fl);
172     response.sendRedirect("view");
173
174 }
175
176 private void deleteDetails(HttpServletRequest req, HttpServletResponse resp) throws IOException {
177
178     int id = Integer.parseInt(req.getParameter("fid"));
179     Flight f = ob.getFlightById(id);
180     ob.deleteFlight(f);
181     resp.sendRedirect("view");
182 }
183
184 private void showAllDetails(HttpServletRequest response, HttpServletResponse request)
185     throws ServletException, IOException {
186
187     List<Flight> list = ob.getAllDetails();
188     System.out.println(list);
189     request.setAttribute("list", list);
190     RequestDispatcher rd = request.getRequestDispatcher("FlightDetails.jsp");
191     rd.forward(request, response);
192
193 }
194
195 }
```

Desktop - Airways\_FlyAway/src/main/java/Controller/MasterServlet.java - Eclipse IDE

```
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer X HibernateConfig.java MasterServlet.java Flight.java Passenger.java Password.java FlightDAO.java PassengerDAO.java
197     private void showNewForm(HttpServletRequest request, HttpServletResponse response)
198         throws ServletException, IOException {
199
200     RequestDispatcher rd = request.getRequestDispatcher("AddFlight.jsp");
201     rd.forward(request, response);
202 }
203
204
205 private void insertDetails(HttpServletRequest request, HttpServletResponse response) throws IOException {
206
207     int fnumber = Integer.parseInt(request.getParameter("fnumber"));
208     String fname = request.getParameter("fname");
209     String forigin = request.getParameter("forigin");
210     String ftarget = request.getParameter("ftarget");
211     String date = request.getParameter("fdate");
212     float fprice = Float.parseFloat(request.getParameter("fprice"));
213     Flight fl = new Flight(fnumber, fname, forigin, ftarget, date, fprice);
214     ob.insertFlightInDB(fl);
215     response.sendRedirect("view");
216 }
217
218
219 private void passengerFlightDetails(HttpServletRequest request, HttpServletResponse response)
220     throws ServletException, IOException {
221
222     String origin = request.getParameter("origin");
223     String target = request.getParameter("target");
224     String date = request.getParameter("date");
225     num = Integer.parseInt(request.getParameter("qty"));
226
227     List<Flight> list = ob.getAllDetailsByOriginDate(origin, date, target);
228     // System.out.println(list);
229     // request.setAttribute("num", num);
230     request.setAttribute("Selectedlist", list);
231     RequestDispatcher rd = request.getRequestDispatcher("PassengerFlights.jsp");
232     rd.forward(request, response);
233 }
```

Desktop - Airways\_FlyAway/src/main/java/Controller/MasterServlet.java - Eclipse IDE

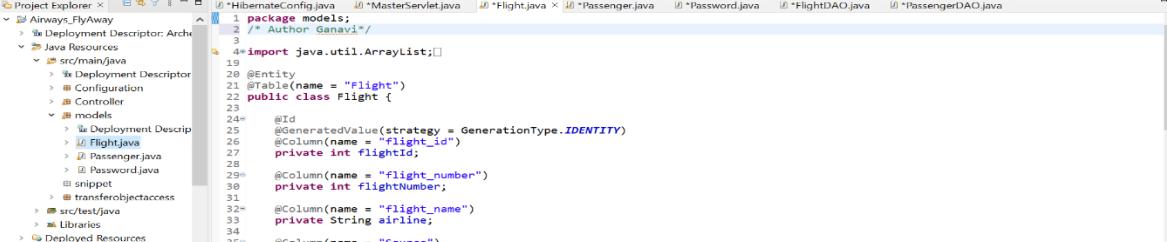
```
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer X HibernateConfig.java MasterServlet.java Flight.java Passenger.java Password.java FlightDAO.java PassengerDAO.java
234
235
236
237     private void getFlightDetailsById(HttpServletRequest request, HttpServletResponse response)
238         throws ServletException, IOException {
239
240     int id = Integer.parseInt(request.getParameter("fid"));
241     FlightAO x = new FlightAO();
242     f = x.getFlightById(id);
243     // request.setAttribute("num", num);
244     RequestDispatcher rd = request.getRequestDispatcher("booking");
245     rd.forward(request, response);
246
247
248 private void register(HttpServletRequest request, HttpServletResponse response)
249     throws ServletException, IOException {
250
251     if (count != 0) {
252         ob.relation(f, list1);
253         request.setAttribute("n", num);
254         request.setAttribute("f", f);
255         // request.setAttribute("p", p);
256         RequestDispatcher rd = request.getRequestDispatcher("SummaryPage.jsp");
257         rd.forward(request, response);
258     } else
259         response.sendRedirect("HomePage.jsp");
260
261
262 private void storage(HttpServletRequest request, HttpServletResponse response) throws IOException {
263
264     ob1.insertPassengerInDB(p);
265     response.sendRedirect("HomePage.jsp");
266
267 }
```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the `Airways_FlyAway` project structure. It includes a Deployment Descriptor, Java Resources (src/main/java), and Deployed Resources.
- Java Code Editor:** Displays the `MasterServlet.java` file. The code contains methods for changing a password and logging in users, both of which validate credentials against a database.
- Bottom Bar:** Shows the Writable status, Smart Insert mode, current time (2:19:38), and date (15-01-2023). It also includes system icons for battery, network, and volume.

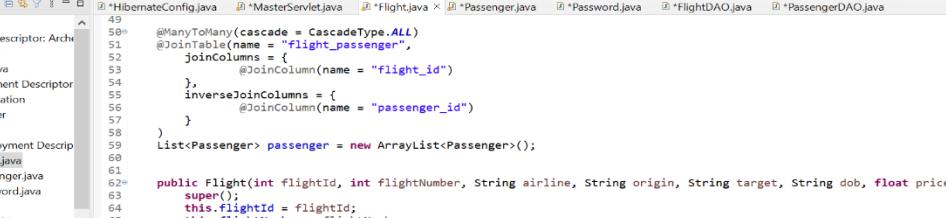
```
268     private void changePassword(HttpServletRequest request, HttpServletResponse response)
269         throws IOException, ServletException {
270
271     String newpwd = request.getParameter("newpwd");
272     String confpwd = request.getParameter("confpwd");
273
274     if (newpwd.compareTo(confpwd) == 0) {
275         pd = newpwd;
276         response.sendRedirect("AdminLogin.jsp");
277     } else {
278         RequestDispatcher rd = request.getRequestDispatcher("ResetPage.jsp");
279         PrintWriter out = response.getWriter();
280         rd.include(request, response);
281         out.println("<center> <span style='color:red'> Invalid Credentials!!! </span></center>");
282     }
283 }
284
285
286
287
288 private void login(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
289
290     String email = request.getParameter("email");
291     String pwd = request.getParameter("pwd");
292
293     RequestDispatcher rd = null;
294
295     if (email.equalsIgnoreCase("admin@test.com") && pwd.equals(pd)) {
296         rd = request.getRequestDispatcher("view");
297         rd.forward(request, response);
298     } else {
299         RequestDispatcher rd = request.getRequestDispatcher("AdminLogin.jsp");
300         PrintWriter out = response.getWriter();
301         rd.include(request, response);
302         out.println("<center> <span style='color:red'> Invalid Credentials!!! </span></center>");
303     }
304 }
```

### 3. Flight



The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/models/Flight.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbars:** Standard, Java, Database, XML, CSS, JavaScript, CSS3, HTML, JSP, JSTL, JBoss Seam, JBoss Seam 2, JBoss Seam 3, JBoss Seam 4, JBoss Seam 5, JBoss Seam 6, JBoss Seam 7, JBoss Seam 8, JBoss Seam 9, JBoss Seam 10, JBoss Seam 11, JBoss Seam 12, JBoss Seam 13, JBoss Seam 14, JBoss Seam 15, JBoss Seam 16, JBoss Seam 17, JBoss Seam 18, JBoss Seam 19, JBoss Seam 20, JBoss Seam 21, JBoss Seam 22, JBoss Seam 23, JBoss Seam 24, JBoss Seam 25, JBoss Seam 26, JBoss Seam 27, JBoss Seam 28, JBoss Seam 29, JBoss Seam 30, JBoss Seam 31, JBoss Seam 32, JBoss Seam 33, JBoss Seam 34, JBoss Seam 35, JBoss Seam 36, JBoss Seam 37, JBoss Seam 38, JBoss Seam 39, JBoss Seam 40, JBoss Seam 41, JBoss Seam 42, JBoss Seam 43, JBoss Seam 44, JBoss Seam 45, JBoss Seam 46, JBoss Seam 47, JBoss Seam 48, JBoss Seam 49, JBoss Seam 50, JBoss Seam 51.
- Project Explorer:** Shows the project structure for 'Airways\_FlyAway' with packages like 'src/main/java' containing 'models', 'Deployment Descriptor', 'Configuration', 'Controller', and 'Deployment Descriptor'. Other packages include 'src/main/resources', 'src/test/java', 'src/test/resources', 'src/main/webapp', and 'src/main/resources/META-INF'. Classes like 'Flight.java', 'Passenger.java', 'Password.java', 'FlightDAO.java', and 'PassengerDAO.java' are listed under 'models'.
- Code Editor:** The main editor window displays the 'Flight.java' code. The code defines a class 'Flight' with various annotations and fields. It includes annotations for @Table, @Id, @GeneratedValue, @Column, and @ManyToMany. Fields include flightId, flightNumber, flightName, source, origin, destination, target, boardingDate, dob, ticketPrice, and a many-to-many relationship with 'Passenger'.
- Bottom Status Bar:** Shows the current file as 'Flight.java', the status as 'Writable', the date and time as '2 : 19 : 34', and the system temperature as '24°C'.



The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/models/Flight.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Project Explorer:** Shows the project structure:
  - src/main/java
    - Deployment Descriptor
    - Java Resources
      - Deployment Descriptor
      - Flight.java
      - Passenger.java
      - Password.java
    - models
      - Flight.java
      - Passenger.java
      - Password.java
    - snippet
    - transferobjectaccess
  - src/test/java
  - libraries
  - Deployed Resources
  - src
    - main
      - java
      - webapp
        - WEB-INF
        - AddFlight.jsp
        - AddPassenger.jsp
        - AdminLogin.jsp
        - FlightDetails.jsp
        - HomePage.jsp
        - PassengerFlights.jsp
        - PaymentPage.jsp
        - RegisterPage.jsp
- Code Editor:** Displays the `Flight.java` file content.
- Bottom Bar:** Shows the status bar with "Writable", "Smart Insert", "2 : 19 : 34", and system icons like battery level, temperature (24°C), and network.
- Taskbar:** Shows various application icons including Microsoft Word, Mozilla Firefox, and Cisco Logon.
- Search Bar:** Type here to search

Desktop - Airways\_FlyAway/src/main/java/models/Flight.java - Eclipse IDE

```

public int getFlightId() {
    return flightId;
}

public void setFlightId(int flightId) {
    this.flightId = flightId;
}

public int getFlightNumber() {
    return flightNumber;
}

public void setFlightNumber(int flightNumber) {
    this.flightNumber = flightNumber;
}

public String getAirline() {
    return airline;
}

public void setAirline(String airline) {
    this.airline = airline;
}

public String getOrigin() {
    return origin;
}

public void setOrigin(String origin) {
    this.origin = origin;
}

public String getTarget() {
    return target;
}

}

```

Desktop - Airways\_FlyAway/src/main/java/models/Flight.java - Eclipse IDE

```

public String getUrgency() {
    return urgency;
}

public void setUrgency(String urgency) {
    this.urgency = urgency;
}

public String getTarget() {
    return target;
}

public void setTarget(String target) {
    this.target = target;
}

public String getDob() {
    return dob;
}

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

public float getPrice() {
    return price;
}

public void setPrice(float price) {
    this.price = price;
}

public List<Passenger> getPassenger() {
    return passenger;
}

public void setPassenger(List<Passenger> passenger) {
    this.passenger = passenger;
}

}

```

## 4. Passenger

Desktop - Airways\_FlyAway/src/main/java/models/Passenger.java - Eclipse IDE

```

package models;
/*
 * Author: Ganavati
 */
import java.util.ArrayList;

@Entity
@Table(name = "Passenger")
public class Passenger {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name="passenger_id")
    private int pid;
    @Column(name="passenger_fname")
    private String fname;
    @Column(name="passenger_lname")
    private String lname;
    @Column(name="passenger_age")
    private int age;
    @Column(name="passenger_mob")
    private long contact;
    @Column(name="passenger_email")
    private String email;
    @ManyToMany(mappedBy = "passenger")
    private List<Flight> flight = new ArrayList<Flight>();

    public Passenger() {
    }

    public Passenger(int pid, String fname, String lname, int age, long contact, String email) {
        this.pid = pid;
        this.fname = fname;
        this.lname = lname;
        this.age = age;
        this.contact = contact;
        this.email = email;
    }
}

```

Desktop - Airways\_FlyAway/src/main/java/models/Passenger.java - Eclipse IDE

```

public Passenger(int pid, String fname, String lname, int age, long contact, String email) {
    super();
    this.pid = pid;
    this.fname = fname;
    this.lname = lname;
    this.age = age;
    this.contact = contact;
    this.email = email;
}

public Passenger(String fname, String lname, int age, long contact, String email) {
    super();
    this.fname = fname;
    this.lname = lname;
    this.age = age;
    this.contact = contact;
    this.email = email;
}

public int getId() {
    return pid;
}

public void setId(int pid) {
    this.pid = pid;
}

public String getFname() {
    return fname;
}

public void setFname(String fname) {
    this.fname = fname;
}

```

Desktop - Airways\_FlyAway/src/main/java/models/Passenger.java - Eclipse IDE

```

public String getName() {
    return lname;
}

public void setLname(String lname) {
    this.lname = lname;
}

public int getAge() {
    return age;
}

public void setAge(int age) {
    this.age = age;
}

public long getContact() {
    return contact;
}

public void setContact(long contact) {
    this.contact = contact;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public List<Flight> getFlight() {
    return flight;
}

```

## 5. Password :

Desktop - Airways\_FlyAway/src/main/java/models/Password.java - Eclipse IDE

```

package models;
/* Author: Ganavi */

public class Password {
    private static String pwd;

    public Password() {
        pwd = "admin";
    }

    public static String getPwd() {
        return pwd;
    }

    public static void setPwd(String pwd) {
        Password.pwd = pwd;
    }
}

```

## 6. FlightDao:

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Displays the project structure for "Airways\_FlyAway".
- Code Editor:** Shows the Java code for `FlightDAO.java`. The code implements a DAO for flights, using Hibernate to interact with a database. It includes methods for inserting flights and retrieving flight details.
- Taskbar:** At the bottom, there are several browser tabs open, including Microsoft Edge, Mozilla Firefox, Google Chrome, and Internet Explorer, along with other system icons like Task View, File Explorer, and Task Manager.

The screenshot shows the Eclipse IDE interface with the title bar "Desktop - Airways\_FlyAway/src/main/java/transferobjectaccess/FlightDAO.java - Eclipse IDE". The left sidebar displays the "Project Explorer" with the package structure: "src/main/java" containing "Flight.java", "Passenger.java", "Password.java", and "FlightDAO.java"; "src/test/java"; and "src/webapp" containing various JSP files like "AddFlight.jsp", "AddPassenger.jsp", "AdminLogin.jsp", "FlightDetails.jsp", "HomePage.jsp", "PassengerFlight.jsp", "PaymentPage.jsp", and "RegisterPage.jsp". The main editor area shows the Java code for "FlightDAO.java".

```
43     } catch (Exception e) {
44         e.printStackTrace();
45     } finally {
46         dbSession.close();
47     }
48     return list;
49 }
50
51 }
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79

public Flight getFlightById(int flightId) {
    Transaction transaction = null;
    Session dbSession = null;
    Flight f = null;
    try {
        dbSession = HibernateConfig.getSessionFactory().openSession();
        transaction = dbSession.beginTransaction();
        f = dbSession.get(Flight.class, flightId);
    } catch (Exception e) {
        if(transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    } finally {
        dbSession.close();
    }
    return f;
}

public Flight updateFlight(Flight f) {
    Transaction transaction = null;
    Session dbSession = null;
    try {
        dbSession = HibernateConfig.getSessionFactory().openSession();
        transaction = dbSession.beginTransaction();
        dbSession.update(f);
        transaction.commit();
    } catch (Exception e) {
        if(transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    } finally {
        dbSession.close();
    }
    return f;
}
```

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/transferobjectaccess/FlightDAO.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Project Explorer View:** Shows the project structure:
  - Airways\_FlyAway
  - Deployment Descriptor: Arch
  - Java Resources
    - src/main/java
      - Deployment Descriptor
      - Configuration
      - Controller
      - models
        - Deployment Descrip
        - FlightJava
        - PassengerJava
        - PasswordJava
      - snippet
      - transferobjectaccess
    - src/test/java
    - Libraries
  - Deployed Resources
  - src
    - main
      - java
      - webapp
        - WEB-INF
          - AddFlight.jsp
          - AddPassenger.jsp
          - AdminLogin.jsp
          - FlightDetails.jsp
          - HomePage.jsp
          - PassengerFlights.jsp
          - PaymentPage.jsp
          - RegisterPage.jsp
- Code Editor:** Displays the content of FlightDAO.java. The code handles database transactions for adding and deleting flights using Hibernate.
- Bottom Status Bar:** Writable, Smart Insert, 2: 19:48, 24°C, ENG, 15-01-2023, 19:38

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** Desktop - Airways\_FlyAway/src/main/java/transferobjectaccess/FlightDAO.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbar:** Standard Eclipse toolbar with icons for file operations, search, and project navigation.
- Project Explorer:** Shows the project structure:
  - src/main/java
    - HibernateConfig.java
    - MasterServlet.java
    - Flight.java
    - Passenger.java
    - Password.java
    - FlightDAO.java
    - PassengerDAO.java
  - Java Resources
  - src/main/java
    - Deployment Descriptor
    - Configuration
    - Controller
    - models
      - Deployment Descrip
      - Flight.java
      - Passenger.java
      - Password.java
    - snippet
    - transferobjectaccess
  - src/test/java
  - Libraries
  - Deployed Resources
  - src
    - main
      - java
      - webapp
        - WEB-INF
          - AddFlight.jsp
          - AddPassenger.jsp
          - AdminLogin.jsp
          - FlightDetails.jsp
          - HomePage.jsp
          - PassengerFlights.jsp
          - PaymentPage.jsp
          - RegisterPage.jsp
- Code Editor:** Displays the content of FlightDAO.java. The code handles database transactions for adding passengers to a flight.
- Bottom Status Bar:** Shows the current time as 2:19:48 and various system status icons.

## 7. PassengerDao

```
public void insertPassengerInDb(Passenger p) {
    Transaction transaction = null;
    Session dbSession = null;
    try {
        dbSession = HibernateConfig.getSessionFactory().openSession();
        transaction = dbSession.beginTransaction();
        dbSession.save(p);
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    } finally {
        dbSession.close();
    }
}

@SuppressWarnings("unchecked")
public List<Passenger> getAllDetails() {
    Session dbSession = null;
    List<Passenger> list = null;
    try {
        dbSession = HibernateConfig.getSessionFactory().openSession();
        list = dbSession.createQuery("from Passenger").list();
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        dbSession.close();
    }
    return list;
}
```

```

    public Passenger getPassengerById(int id) {
        Transaction transaction = null;
        Session dbSession = null;
        Passenger p = null;
        try {
            dbSession = HibernateConfig.getSessionFactory().openSession();
            transaction = dbSession.beginTransaction();
            p = dbSession.get(Passenger.class, id);
        } catch (Exception e) {
            if(transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        } finally {
            dbSession.close();
        }
        return p;
    }

    public Passenger deletePassenger(Passenger p) {
        Transaction transaction = null;
        Session dbSession = null;
        try {
            dbSession = HibernateConfig.getSessionFactory().openSession();
            transaction = dbSession.beginTransaction();
            dbSession.delete(p);
            transaction.commit();
        } catch (Exception e) {
            if(transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        } finally {
            dbSession.close();
        }
        return p;
    }
}

```

```

    public Passenger getPassengerById(int id) {
        Transaction transaction = null;
        Session dbSession = null;
        Passenger p = null;
        try {
            dbSession = HibernateConfig.getSessionFactory().openSession();
            transaction = dbSession.beginTransaction();
            p = dbSession.get(Passenger.class, id);
        } catch (Exception e) {
            if(transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        } finally {
            dbSession.close();
        }
        return p;
    }

    public Passenger deletePassenger(Passenger p) {
        Transaction transaction = null;
        Session dbSession = null;
        try {
            dbSession = HibernateConfig.getSessionFactory().openSession();
            transaction = dbSession.beginTransaction();
            dbSession.delete(p);
            transaction.commit();
        } catch (Exception e) {
            if(transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        } finally {
            dbSession.close();
        }
        return p;
    }
}

```

## 8. Web.XML

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" version="2.5">
    <!--Sun Microsystems, Inc.-->
    <!--DTD Web Application 2.3//EN (doctype with catalog)-->
    <!--Sun Microsystems, Inc.-->
    <!--DTD Web Application 2.3//EN-->
    <!--http://java.sun.com/dtd/web_app_2_3.dtd-->
    <!--Sun Microsystems, Inc.-->
    <!--Archetype Created Web Application-->
    <servlet>
        <servlet-name>MasterServlet</servlet-name>
        <description>MasterServlet</description>
        <servlet-class>Controller.MasterServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>MasterServlet</servlet-name>
        <url-pattern>/MasterServlet</url-pattern>
    </servlet-mapping>
    <welcome-file-list>
        <welcome-file>HomePage.jsp</welcome-file>
    </welcome-file-list>
</web-app>

```

## 9. Add Flight

Desktop - Airways\_FlyAway/src/main/webapp/AddFlight.jsp - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer ▾ AddFlight.jsp ▾ *HomePage.jsp ▾ *PassengerF... ▾ *PaymentPag... ▾ *RegisterPag... ▾ *ResetPage.jsp ▾ *SummaryPage.jsp ▾ *web.xml ▾ Airways_FlyA...
src
  WEB-INF
    web.xml
    AddFlight.jsp
    AddPassenger.jsp
    AdminLogin.jsp
    FlightDetails.jsp
    HomePage.jsp
    PassengerFlights.jsp
    PaymentPage.jsp
    RegisterPage.jsp
    ResetPage.jsp
    SummaryPage.jsp
  pom.xml
project2 (in JAVAFSD-Project02)
  Deployment Descriptor: Arch
  Java Resources
    src/main/java
      Deployment Descriptor
      Configuration
      Controller

```

```

3 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
4   pageEncoding="ISO-8859-1" isELIgnored=false%>
5 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
6 <!DOCTYPE html>
7 <html>
8   <head>
9     <meta charset="ISO-8859-1">
10    <title>FLYAWAY.COM</title>
11   </head>
12   <body>
13     <div align="center">
14       <c:if test="${f != null}">
15         <form action="update" method="post">
16           <c:if>
17             <c:if test="${f == null}">
18               <form action="insert" method="post">
19                 <table border="1" cellpadding="5">
20                   <c:if test="${f != null}">
21                     <tr>
22                       <td><input type="hidden" name="fid" value=">" />
23                     <tr>
24                       <th>Flight Number :</th>
25                       <td><input type="number" name="fnumber" size="45" value=">" required /></td>
26                     <tr>
27                       <th>Airline :</th>
28                       <td><input type="text" name="fphone" size="45" value=">" required /></td>
29                     <tr>
30                       <th>Origin :</th>
31                       <td><input type="text" name="forigin" size="45" value=">" required /></td>
32                     <tr>
33                       <th>Destination :</th>
34                       <td><input type="text" name="ftarget" size="45" value=">" required /></td>
35                     <tr>
36                       <th>Flight Date :</th>
37                       <td><input type="date" name="fdate" size="45" value=">" required /></td>
38                     <tr>
39                       <th>Ticket Price :</th>
40                       <td><input type="number" name="fprice" size="45" value=">" required /></td>
41                     <tr>
42                       <td colspan="2" align="center"><input type="submit" value="Save" /></td>
43                     </table>
44                   </form>
45                 </div>
46               </c:if>
47             </c:if>
48           </form>
49         </c:if>
50       </div>
51     </body>
52   </html>
53 <%>
54 <%>
55 <%>
56 <%>
57 <%>
58 <%>
59 <%>
60 <%>
61 <%>
62 <%>
63 <%>
64 <%>
65 <%>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console Tomcat v8.5 Server at localhost [Apache Tomcat C:\Users\qamohank\p2\pool\plugins\org.eclipse.jst.openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\re\bin\javaw.exe] (15-Jan-2023) INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Writable Smart Insert 1: 19:18 24°C ENG 19:46 15-01-2023

Desktop - Airways\_FlyAway/src/main/webapp/AddFlight.jsp - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer ▾ AddFlight.jsp ▾ *HomePage.jsp ▾ *PassengerF... ▾ *PaymentPag... ▾ *RegisterPag... ▾ *ResetPage.jsp ▾ *SummaryPage.jsp ▾ *web.xml ▾ Airways_FlyA...
src
  WEB-INF
    web.xml
    AddFlight.jsp
    AddPassenger.jsp
    AdminLogin.jsp
    FlightDetails.jsp
    HomePage.jsp
    PassengerFlights.jsp
    PaymentPage.jsp
    RegisterPage.jsp
    ResetPage.jsp
    SummaryPage.jsp
  pom.xml
project2 (in JAVAFSD-Project02)
  Deployment Descriptor: Arch
  Java Resources
    src/main/java
      Deployment Descriptor
      Configuration
      Controller

```

```

28   <th>Flight Number :</th>
29   <td><input type="number" name="fnumber" size="45" value=">" required /></td>
30   <tr>
31     <th>Airline :</th>
32     <td><input type="text" name="fphone" size="45" value=">" required /></td>
33   <tr>
34     <th>Origin :</th>
35     <td><input type="text" name="forigin" size="45" value=">" required /></td>
36   <tr>
37     <th>Destination :</th>
38     <td><input type="text" name="ftarget" size="45" value=">" required /></td>
39   <tr>
40     <th>Flight Date :</th>
41     <td><input type="date" name="fdate" size="45" value=">" required /></td>
42   <tr>
43     <th>Ticket Price :</th>
44     <td><input type="number" name="fprice" size="45" value=">" required /></td>
45   <tr>
46     <td colspan="2" align="center"><input type="submit" value="Save" /></td>
47   </table>
48 </form>
49 </div>
50 </c:if>
51 </c:if>
52 </c:if>
53 </c:if>
54 </c:if>
55 </c:if>
56 </c:if>
57 </c:if>
58 </c:if>
59 </c:if>
60 </c:if>
61 </c:if>
62 </c:if>
63 </c:if>
64 </c:if>
65 </c:if>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console Tomcat v8.5 Server at localhost [Apache Tomcat C:\Users\qamohank\p2\pool\plugins\org.eclipse.jst.openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\re\bin\javaw.exe] (15-Jan-2023) INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Writable Smart Insert 1: 19:18 24°C ENG 19:46 15-01-2023

Desktop - Airways\_FlyAway/src/main/webapp/AddFlight.jsp - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer ▾ AddFlight.jsp ▾ *HomePage.jsp ▾ *PassengerF... ▾ *PaymentPag... ▾ *RegisterPag... ▾ *ResetPage.jsp ▾ *SummaryPage.jsp ▾ *web.xml ▾ Airways_FlyA...
src
  WEB-INF
    web.xml
    AddFlight.jsp
    AddPassenger.jsp
    AdminLogin.jsp
    FlightDetails.jsp
    HomePage.jsp
    PassengerFlights.jsp
    PaymentPage.jsp
    RegisterPage.jsp
    ResetPage.jsp
    SummaryPage.jsp
  pom.xml
project2 (in JAVAFSD-Project02)
  Deployment Descriptor: Arch
  Java Resources
    src/main/java
      Deployment Descriptor
      Configuration
      Controller

```

```

36   <th>Flight Number :</th>
37   <td><input type="number" name="fnumber" size="45" value=">" required /></td>
38   <tr>
39     <th>Origin :</th>
40     <td><input type="text" name="forigin" size="45" value=">" required /></td>
41   <tr>
42     <th>Destination :</th>
43     <td><input type="text" name="ftarget" size="45" value=">" required /></td>
44   <tr>
45     <th>Flight Date :</th>
46     <td><input type="date" name="fdate" size="45" value=">" required /></td>
47   <tr>
48     <th>Ticket Price :</th>
49     <td><input type="number" name="fprice" size="45" value=">" required /></td>
50   <tr>
51     <td colspan="2" align="center"><input type="submit" value="Save" /></td>
52   </table>
53 </form>
54 </div>
55 </c:if>
56 </c:if>
57 </c:if>
58 </c:if>
59 </c:if>
60 </c:if>
61 </c:if>
62 </c:if>
63 </c:if>
64 </c:if>
65 </c:if>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console Tomcat v8.5 Server at localhost [Apache Tomcat C:\Users\qamohank\p2\pool\plugins\org.eclipse.jst.openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\re\bin\javaw.exe] (15-Jan-2023) INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Writable Smart Insert 1: 19:18 24°C ENG 19:46 15-01-2023

### **10. Add Passengers:**

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure for "Airways\_FlyAway/src/main/webapp/AddPassenger.jsp".
- Code Editor:** Displays the JSP code for "AddPassenger.jsp". The code includes JSTL tags for form handling and validation.
- Terminal:** Shows the output of the Tomcat server startup command: "Tomcat v8.5 Server at localhost [Apache Tomcat] C:\Users\guru\Downloads\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32-x64\_17.0.5.v20221102-0933\releas\java.exe (15-Jan-2023) INFO: Reloading Context with name [/Airways\_FlyAway] is completed".
- Bottom Bar:** Includes the Windows taskbar with icons for File Explorer, OneDrive, Task View, Start, and a search bar.

The screenshot shows the Eclipse IDE interface with the following details:

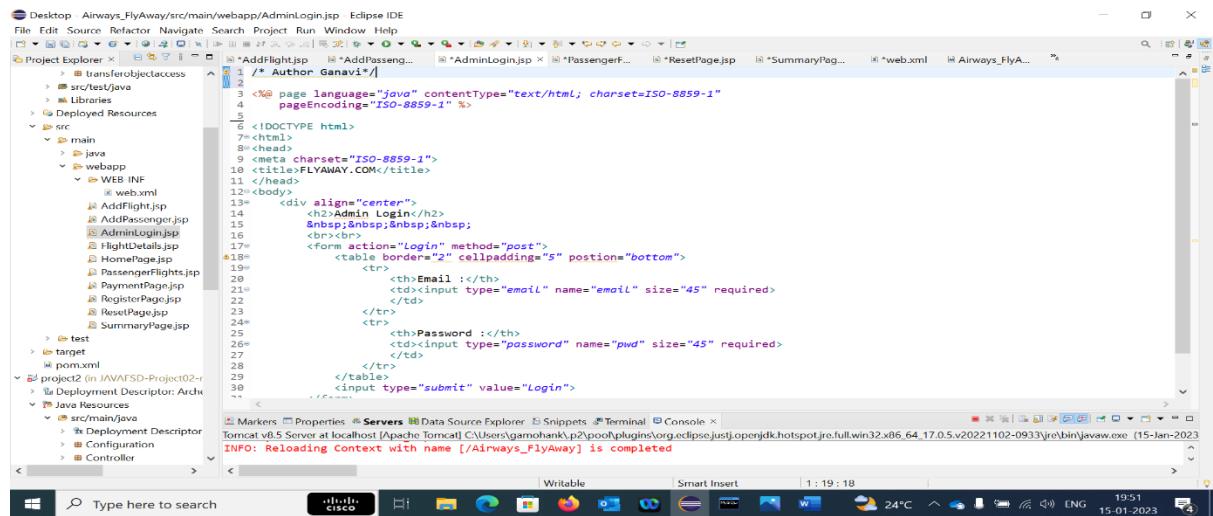
- Project Explorer:** Displays the project structure. The `AddPassenger.jsp` file is selected in the `src/main/webapp` folder.
- Java Editor:** Shows the code for `AddPassenger.jsp`. The code defines a form with fields for First Name, Last Name, Contact, Age, Email, and a Save button. It uses JSP EL expressions (`<%>`) to bind values to input fields.
- Bottom Status Bar:** Shows the current time as 1:19:18, the date as 15-Jan-2023, and the system temperature as 24°C.

```
<th>First Name :</th>
<td><input type="text" name="fname" size="45"
           value=<%>out.value='${p.fname}' /><!-- required --></td>
</tr>
<tr>
    <th>Last Name :</th>
    <td><input type="text" name="lname" size="45"
           value=<%>out.value='${p.lname}' /><!-- required --></td>
</tr>
<tr>
    <th>Contact :</th>
    <td><input type="number" name="contact" size="45"
           value=<%>out.value='${p.contact}' /><!-- required --></td>
</tr>
<tr>
    <th>Age :</th>
    <td><input type="text" name="age" size="45"
           value=<%>out.value='${p.age}' /><!-- required --></td>
</tr>
<tr>
    <th>Email :</th>
    <td><input type="email" name="email" size="45"
           value=<%>out.value='${p.email}' /><!-- required --></td>
</tr>
<tr>
    <td colspan="2" align="center"><input type="submit"
           value="Save" /></td>
</tr>
</table>
</form>
```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Displays the project structure for "Airways\_FlyAway". It includes a main package with Java and web resources, and a WEB-INF folder containing web.xml, AddFlight.jsp, AddPassenger.jsp, AdminLogin.jsp, FlightDetails.jsp, HomePage.jsp, PassengerFlights.jsp, PaymentPage.jsp, RegisterPage.jsp, ResetPage.jsp, and SummaryPage.jsp.
- Code Editor:** The current file is "AddPassenger.jsp". The code is a JSP page with form fields for lname, contact, age, email, and a submit button. Lines 57 and 58 show the closing of the body and html tags.
- Terminal:** Shows the output of a Tomcat 8.5.0 server running at localhost. The message "INFO: Reloading Context with name [/Airways\_FlyAway] is completed" is displayed.

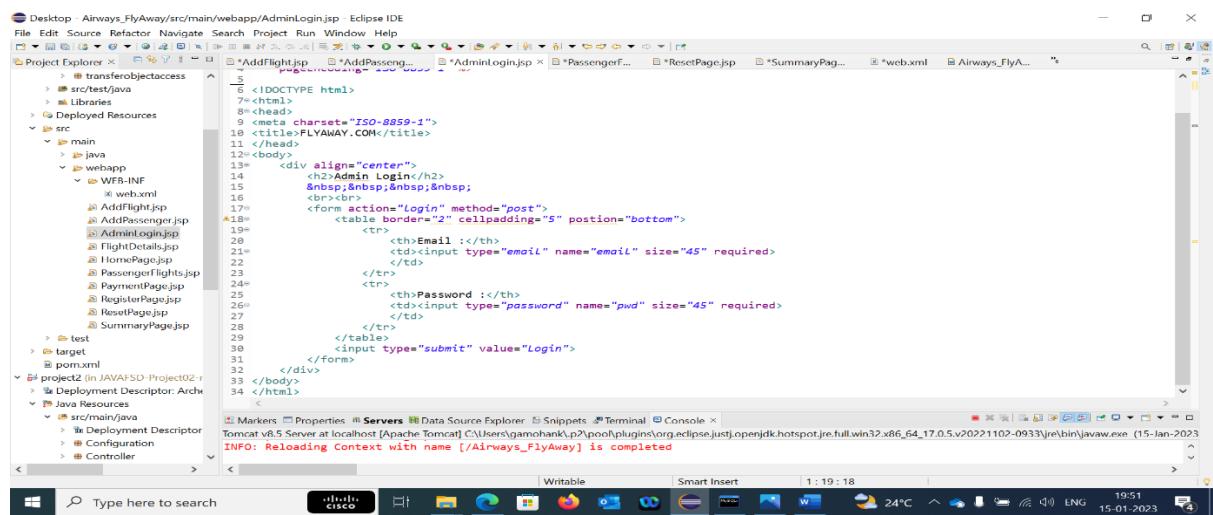
## 11. Admin login



```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body>
<div align="center">
<h2>Admin Login</h2>
<br><br>
<form action="login" method="post">
<table border="2" cellpadding="5" position="bottom">
<tr>
<td>Email :</td>
<td><input type="email" name="email" size="45" required></td>
</tr>
<tr>
<td>Password :</td>
<td><input type="password" name="pwd" size="45" required></td>
</tr>
<tr>
<td colspan="2" style="text-align: center;"><input type="submit" value="Login"></td>
</tr>
</table>
</form>
</div>

```

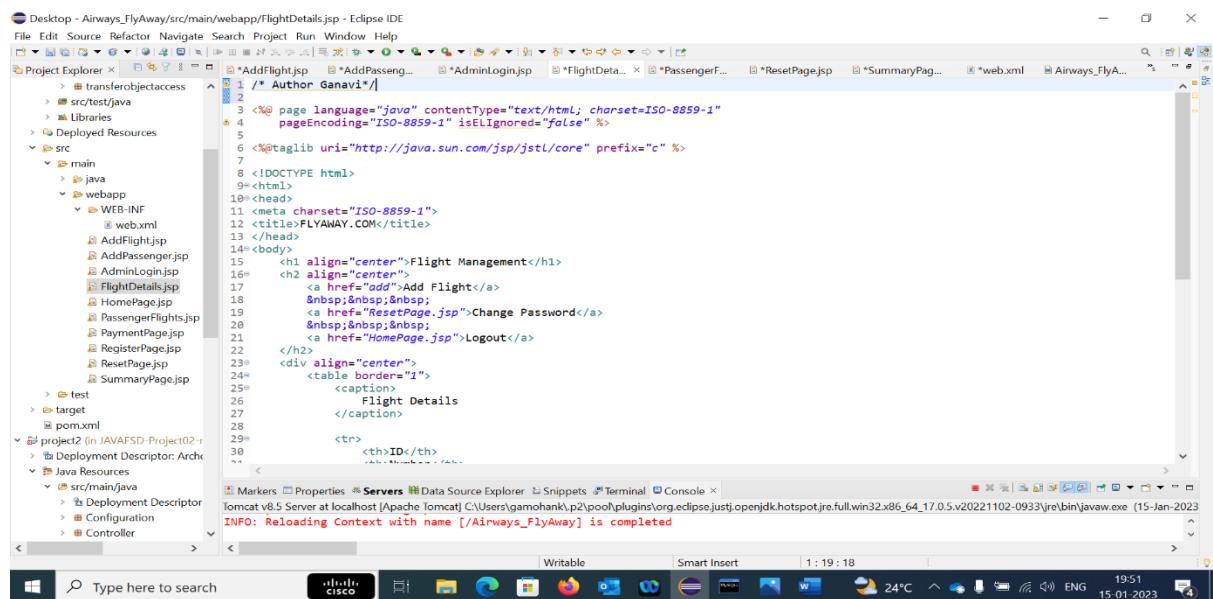


```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body>
<div align="center">
<h2>Admin Login</h2>
<br><br>
<form action="login" method="post">
<table border="2" cellpadding="5" position="bottom">
<tr>
<td>Email :</td>
<td><input type="email" name="email" size="45" required></td>
</tr>
<tr>
<td>Password :</td>
<td><input type="password" name="pwd" size="45" required></td>
</tr>
<tr>
<td colspan="2" style="text-align: center;"><input type="submit" value="Login"></td>
</tr>
</table>
</form>
</div>

```

## 12. Flight Details



```

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body>
<h1 align="center">Flight Management</h1>
<h2 align="center"><a href="#">Add Flight</a><br/><a href="ResetPage.jsp">Change Password</a><br/><a href="HomePage.jsp">Logout</a></h2>
<h2 align="center"><table border="1"><caption>Flight Details</caption><thead><tr><th>ID</th></tr></thead></table></h2>

```

Desktop - Airways\_FlyAway/src/main/webapp/flightDetails.jsp - Eclipse IDE

```

<tr>
    <th>ID</th>
    <th>Number</th>
    <th>Air Name</th>
    <th>Origin</th>
    <th>Destination</th>
    <th>Price</th>
    <th>Booking Price</th>
    <th>Status</th>
</tr>
<c:forEach var="f" items="${list}">
    <tr>
        <td><c:out value="${f.flightId}" /></td>
        <td><c:out value="${f.flightNumber}" /></td>
        <td><c:out value="${f.airline}" /></td>
        <td><c:out value="${f.origin}" /></td>
        <td><c:out value="${f.target}" /></td>
        <td><c:out value="${f.dob}" /></td>
        <td><c:out value="${f.price}" /></td>
        <td>
            <a href="edit?fid=<c:out value='${f.flightId}' />">Edit</a>
            &nbsp;&nbsp;&nbsp;
            <a href="delete?fid=<c:out value='${f.flightId}' />">Delete</a>
        </td>
    </tr>
</c:forEach>
</table>
</body>
</html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

## 13. HomePage

Desktop - Airways\_FlyAway/src/main/webapp/HomePage.jsp - Eclipse IDE

```

1 /* Author Ganavi */
2 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
3     pageEncoding="ISO-8859-1"%>
4 <!DOCTYPE html>
5 <html>
6     <head>
7         <meta charset="ISO-8859-1">
8         <title>FLYAWAY.COM</title>
9     </head>
10    <body>
11        <div class="container" align="center">
12            <a href="AdminLogin.jsp">Admin</a> <br>
13            <br>
14            <form action="passenger" method="post">
15                <table border="2" cellpadding="5" position="bottom">
16                    <tr>
17                        <th>Origin :</th>
18                        <td><input type="text" name="origin" size="45" required>
19                        </td>
20                    </tr>
21                    <tr>
22                        <th>Destination :</th>
23                        <td><input type="text" name="target" size="45" required>
24                        </td>
25                    </tr>
26                    <tr>
27                        <th>Boarding Date :</th>
28                        <td><input type="date" name="date" size="45" required>
29                        </td>
30                    </tr>
31                </table>
32            </form>
33        </div>
34    </body>
35 </html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Desktop - Airways\_FlyAway/src/main/webapp/HomePage.jsp - Eclipse IDE

```

13        <a href="AdminLogin.jsp">Admin</a> <br>
14        <br>
15        <form action="passenger" method="post">
16            <table border="2" cellpadding="5" position="bottom">
17                <tr>
18                    <th>Origin :</th>
19                    <td><input type="text" name="origin" size="45" required>
20                    </td>
21                </tr>
22                <tr>
23                    <th>Destination :</th>
24                    <td><input type="text" name="target" size="45" required>
25                    </td>
26                </tr>
27                <tr>
28                    <th>Boarding Date :</th>
29                    <td><input type="date" name="date" size="45" required>
30                    </td>
31                </tr>
32                <tr>
33                    <th>Persons :</th>
34                    <td><input type="number" name="qty" size="45" required>
35                    </td>
36                </tr>
37            </table>
38            <input type="submit" value="Search">
39        </form>
40    </div>
41 </body>
42 </html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

## 14. PassengerDetails :

The screenshot shows the Eclipse IDE interface with the code editor open to the PassengerDetails.jsp file. The code is JSP (Java Server Pages) and displays flight details. The browser taskbar at the bottom shows the URL [http://localhost:8080/Airways\\_FlyAway/](http://localhost:8080/Airways_FlyAway/).

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1" isELIgnored="false"%>
3 <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4
5 <!DOCTYPE html>
6 <html charset="ISO-8859-1">
7   <head>
8     <meta charset="ISO-8859-1">
9     <title>FLYAWAY.COM</title>
10    </head>
11    <body>
12      <div align="center">
13        <h2>Selected Flight Details</h2>
14        &nbsp;&nbsp;&nbsp;
15        <table border="1">
16          <tr>
17            <th>ID</th>
18            <th>Number</th>
19            <th>Airline</th>
20            <th>Origin</th>
21            <th>Destination</th>
22            <th>Date</th>
23            <th>Booking Price</th>
24            <th>Status</th>
25          </tr>
26          <c:forEach var="f" items="${Selectedlist}">
27            <tr>
28              <td>${f.flightId}</td>
29              <td>${f.flightNumber}</td>
30              <td>${f.airline}</td>
31              <td>${f.origin}</td>
32              <td>${f.destination}</td>
33              <td>${f.date}</td>
34              <td>${f.price}</td>
35              <td>${f.status}</td>
36              <td>${f.dob}</td>
37              <td>${f.dob}</td>
38              <td>${f.dob}</td>
39              <td><a href="#">Book Now</a></td>
40            </tr>
41          </c:forEach>
42        </table>
43      </div>
44    </body>
45  </html>

```

This screenshot shows the same Eclipse IDE environment, but with several lines of code removed from the PassengerDetails.jsp file. The browser taskbar at the bottom shows the URL [http://localhost:8080/Airways\\_FlyAway/](http://localhost:8080/Airways_FlyAway/).

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1" isELIgnored="false"%>
3 <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4
5 <!DOCTYPE html>
6 <html charset="ISO-8859-1">
7   <head>
8     <meta charset="ISO-8859-1">
9     <title>FLYAWAY.COM</title>
10    </head>
11    <body>
12      <div align="center">
13        <h2>Selected Flight Details</h2>
14        &nbsp;&nbsp;&nbsp;
15        <table border="1">
16          <tr>
17            <th>ID</th>
18            <th>Number</th>
19            <th>Airline</th>
20            <th>Origin</th>
21            <th>Destination</th>
22            <th>Date</th>
23            <th>Booking Price</th>
24            <th>Status</th>
25          </tr>
26          <c:forEach var="f" items="${Selectedlist}">
27            <tr>
28              <td>${f.flightId}</td>
29              <td>${f.flightNumber}</td>
30              <td>${f.airline}</td>
31              <td>${f.origin}</td>
32              <td>${f.destination}</td>
33              <td>${f.date}</td>
34              <td>${f.price}</td>
35              <td>${f.status}</td>
36              <td>${f.dob}</td>
37              <td>${f.dob}</td>
38              <td>${f.dob}</td>
39              <td><a href="#">Book Now</a></td>
40            </tr>
41          </c:forEach>
42        </table>
43      </div>
44    </body>
45  </html>

```

## 15. Payment Page

The screenshot shows the Eclipse IDE interface with the code editor open to the PaymentPage.jsp file. The code is JSP and handles payment method selection. The browser taskbar at the bottom shows the URL [http://localhost:8080/Airways\\_FlyAway/](http://localhost:8080/Airways_FlyAway/).

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1" isELIgnored="false"%>
3 <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4
5 <!DOCTYPE html>
6 <html>
7   <head>
8     <meta charset="ISO-8859-1">
9     <title>FLYAWAY.COM</title>
10    </head>
11    <body>
12      <div align="center">
13        &nbsp;&nbsp;&nbsp;
14        <table border="2" cellpadding="5" position="bottom">
15          <tr>
16            <th>Payment Method :</th>
17            <td><a href="#">PayPal</a></td>
18            <td><a href="#">PayTM</a></td>
19            <td><a href="#">Debit/Credit Cards</a></td>
20          </tr>
21        </table>
22        <br><br>
23        <a href="HomePage.jsp">Back To Home</a>
24      </div>
25    </body>
26  </html>

```

## 16. Register page

The screenshot shows the Eclipse IDE interface with the project 'Airways\_FlyAway' open. The 'RegisterPage.jsp' file is selected in the Project Explorer. The code in the editor is as follows:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body align="center">
<h1 align="center">Passenger Details</h1>
<h2 align="center"><a href="AddPassenger.jsp">Add Passenger</a> &ampnbsp&ampnbsp&ampnbsp</h2>
<div align="center">
<table border="1">
<caption>Passenger Details</caption>
<tr>
<th>ID</th>
<th>First Name</th>
<th>Last Name</th>
<th>Contact</th>
<th>Age</th>
<th>Email</th>
</tr>
<c:forEach var="p" items="${list}">
<tr>
<c:out value="${p.pid}" /></td>
<c:out value="${p.fname}" /></td>
<c:out value="${p.lname}" /></td>
<c:out value="${p.contact}" /></td>
<c:out value="${p.age}" /></td>
<c:out value="${p.email}" /></td>
<td><a href="deletePassenger?id=<c:out value='${p.pid}' />">Delete</a></td>
</tr>
</c:forEach>
</table>
<a href="register">Confirm</a>
</div>
</body>
</html>
```

The screenshot shows the Eclipse IDE interface with the project 'Airways\_FlyAway' open. The 'RegisterPage.jsp' file is selected in the Project Explorer. The code in the editor has been modified to include a 'confirm' button:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body align="center">
<h1 align="center">Passenger Details</h1>
<h2 align="center"><a href="AddPassenger.jsp">Add Passenger</a> &ampnbsp&ampnbsp&ampnbsp</h2>
<div align="center">
<table border="1">
<caption>Passenger Details</caption>
<tr>
<th>ID</th>
<th>First Name</th>
<th>Last Name</th>
<th>Contact</th>
<th>Age</th>
<th>Email</th>
</tr>
<c:forEach var="p" items="${list}">
<tr>
<c:out value="${p.pid}" /></td>
<c:out value="${p.fname}" /></td>
<c:out value="${p.lname}" /></td>
<c:out value="${p.contact}" /></td>
<c:out value="${p.age}" /></td>
<c:out value="${p.email}" /></td>
<td><a href="deletePassenger?id=<c:out value='${p.pid}' />">Delete</a></td>
</tr>
</c:forEach>
</table>
<a href="register">Confirm</a>
</div>
</body>
</html>
```

## 17. Reset Page:

The screenshot shows the Eclipse IDE interface with the project 'Airways\_FlyAway' open. The 'ResetPage.jsp' file is selected in the Project Explorer. The code in the editor is as follows:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>FLYAWAY.COM</title>
</head>
<body align="center">
<h2>Reset Password</h2>
<br><br>
<form action="reset" method="post">
<table border="2" cellpadding="5" position="bottom">
<tr>
<th>Enter new Password :</th>
<td><input type="password" name="newpwd" size="45" required></td>
</tr>
<tr>
<th>Confirm Password :</th>
<td><input type="password" name="confpwd" size="45" required></td>
</tr>
<tr>
<td colspan="2" style="text-align: center"><input type="submit" value="Save" /></td>
</tr>
</table>
</form>
</body>
</html>
```

Desktop - Airways\_FlyAway/src/main/webapp/ResetPage.jsp - Eclipse IDE

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1" isErrorPage="false"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="ISO-8859-1">
        <title>FLYAWAY.COM</title>
    </head>
    <body>
        <div align="center">
            <h2>Reset Password</h2>
            &nbsp;&nbsp;&nbsp;
            <br><br>
            <form action="reset" method="post">
                <table border="2" cellpadding="5" position="bottom">
                    <tr>
                        <th>Enter new Password :</th>
                        <td><input type="password" name="newpwd" size="45" required>
                        </td>
                    </tr>
                    <tr>
                        <th>Confirm Password :</th>
                        <td><input type="password" name="confpwd" size="45" required>
                        </td>
                    </tr>
                    <tr>
                        <td></td>
                        <td><input type="submit" value="Save">
                        </td>
                    </tr>
                </table>
            </form>
        </div>
    </body>
</html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

Tomcat v8.5 Server at localhost [Apache Tomcat] C:\Users\gammohank\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\jre\bin\javaw.exe (15-Jan-2023)

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Type here to search

## 18. Summary Page:

Desktop - Airways\_FlyAway/src/main/webapp/SummaryPage.jsp - Eclipse IDE

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1" isErrorPage="false"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="ISO-8859-1">
        <title>FLYAWAY.COM</title>
    </head>
    <body>
        <div align="center">
            <h2>Summary Details:</h2>
            &nbsp;&nbsp;&nbsp;
            <form action="PaymentPage.jsp" method="post">
                <table border="1">
                    <tr>
                        <th>Flight Number :</th>
                        <td><:out value="${f.flightNumber}" /></td>
                    </tr>
                    <tr>
                        <th>Flight Name :</th>
                        <td><:out value="${f.airline}" /></td>
                    </tr>
                    <tr>
                        <th>Flight From :</th>
                        <td><:out value="${f.origin}" /></td>
                    </tr>
                </table>
            </form>
        </div>
    </body>
</html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

Tomcat v8.5 Server at localhost [Apache Tomcat] C:\Users\gammohank\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\jre\bin\javaw.exe (15-Jan-2023)

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Type here to search

Desktop - Airways\_FlyAway/src/main/webapp/SummaryPage.jsp - Eclipse IDE

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1" isErrorPage="false"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="ISO-8859-1">
        <title>FLYAWAY.COM</title>
    </head>
    <body>
        <div align="center">
            <table border="1">
                <tr>
                    <th>Flight Number :</th>
                    <td><:out value="${f.flightNumber}" /></td>
                </tr>
                <tr>
                    <th>Flight Name :</th>
                    <td><:out value="${f.airline}" /></td>
                </tr>
                <tr>
                    <th>Flight From :</th>
                    <td><:out value="${f.origin}" /></td>
                </tr>
                <tr>
                    <th>Flight To :</th>
                    <td><:out value="${f.target}" /></td>
                </tr>
                <tr>
                    <th>Flight Boarding Date :</th>
                    <td><:out value="${f.dob}" /></td>
                </tr>
                <tr>
                    <th>Ticket Price :</th>
                    <td><:out value="${f.price * n}" /></td>
                </tr>
            </table>
            <input type="submit" value="Payment">
        </div>
    </body>
</html>

```

Markers Properties Servers Data Source Explorer Snippets Terminal Console

Tomcat v8.5 Server at localhost [Apache Tomcat] C:\Users\gammohank\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32.x86\_64\_17.0.5.v20221102-0933\jre\bin\javaw.exe (15-Jan-2023)

INFO: Reloading Context with name [/Airways\_FlyAway] is completed

Type here to search

## 19. POM.XML :

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Displays the project structure:
  - src/
    - main/
      - java
      - webapp
    - test
    - target
  - project2 (in JAVAFSD-Project02)
    - Deployment Descriptor Archetype
    - Java Source
      - src/main/java
      - Deployment Descriptor
      - Configuration
      - Controller
      - models
      - transformerfactoryaccess
    - src/test/java
    - libraries
  - Deployed Resources
  - Images
  - src/
    - main/
      - java
      - webapp
      - WEB-INF
    - web.xml

- pom.xml Content:**

```
<!-- This file was generated by the Maven Archetype. -->
<!-- To change its behavior please edit pom.xml directly. -->
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.cisco.</groupId>
  <artifactId>Airways_Flyaway</artifactId>
  <version>0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <name>Airways_Flyaway Maven Webapp</name>
  <!-- FIXME: Change it to the project's website -->
  <url>http://www.example.com</url>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

```
- Overview Bar:** Shows tabs for Overview, Dependencies, Dependency Hierarchy, Effective POM, and pom.xml.
- Bottom Status Bar:** Displays the message "INFO: Reloading Context with name [Airways\_FlyAway] is completed".
- System Tray:** Shows icons for battery level (24%), temperature (24°C), signal strength, and date/time (15-01-2023).

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Displays the project structure for "Airways\_FlyAway". It includes:
  - src/test/java
  - Libraries
  - Deployed Resources
  - src
    - main
      - java
      - webapp
    - test
    - target
    - pom.xml
  - project2 (in JAVAFSD-Project02-r)
    - Deployment Descriptor: Archetype
    - Java Resources
      - src/main/java
        - Deployment Descriptor
        - Configuration
        - Controller
        - models
        - transferobjectaccess
      - src/test/java
      - Libraries
    - Deployed Resources
    - Images
    - src
      - main
        - java
        - webapp
          - WEB-INF

- POM.xml Editor:** The current file is "pom.xml". The code is as follows:

```
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax-servlet-api</artifactId>
    <version>3.0-alpha-1</version>
</dependency>
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.13</version>
</dependency>
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.4.29.Final</version>
</dependency>
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-entitymanager</artifactId>
    <version>4.1.8.Final</version>
    <exclusions>
        <exclusion>
            <groupId>org.hibernate.javax.persistence</groupId>
            <artifactId>hibernate-jpa-2.0-api</artifactId>
        </exclusion>
    </exclusions>
</dependency>
<dependency>
    <groupId>javax.persistence</groupId>
    <artifactId>persistence-api</artifactId>
</dependency>
```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer View:** Displays the project structure for "Airways\_FlyAway". It includes Java source code under "src/main/java" (FlightData, HomePage.jsp, PassengerForm, PaymentPage, RegisterPage, ResetPage, SummaryPage), test code under "src/test/java", and configuration files like "pom.xml" and "web.xml".
- Code Editor View:** The main editor window displays the "FlightData.java" file, which contains JSP code and several dependency declarations using Maven's XML syntax.
- Bottom Status Bar:** Shows the message "INFO: Reloading Context with name [/Airways\_FlyAway] is completed".
- System Tray:** Shows icons for Task View, Start, Taskbar, and Network.

```
<dependency>
    <groupId>javax.persistence</groupId>
    <artifactId>persistence-api</artifactId>
    <version>1.0.2</version>
</dependency>
<dependency>
    <groupId>commons-codec</groupId>
    <artifactId>commons-codec</artifactId>
    <version>1.2</version>
</dependency>
<dependency>
    <groupId>org.glassfish.jaxb</groupId>
    <artifactId>jaxb-runtime</artifactId>
    <version>2.3.1</version>
</dependency>
<dependency>
    <groupId>javax.xml.bind</groupId>
    <artifactId>jaxb-api</artifactId>
    <version>2.3.1</version>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jstl</artifactId>
    <version>1.2</version>
</dependency>
<dependency>
    <groupId>javax.servlet.jsp</groupId>
    <artifactId>jsp-api</artifactId>
    <version>2.2</version>
    <scope>provided</scope>
</dependency>
```

```

<!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
<pluginManagement>
    <plugins>
        <plugin>
            <artifactId>maven-clean-plugin</artifactId>
            <version>3.1.0</version>
        </plugin>
        <plugin>
            <artifactId>maven-resources-plugin</artifactId>
            <version>3.0.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.8.0</version>
        </plugin>
        <plugin>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.22.1</version>
        </plugin>
        <plugin>
            <artifactId>maven-war-plugin</artifactId>

```

```

<!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
<pluginManagement>
    <plugins>
        <plugin>
            <artifactId>maven-resources-plugin</artifactId>
            <version>3.0.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.8.0</version>
        </plugin>
        <plugin>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.22.1</version>
        </plugin>
        <plugin>
            <artifactId>maven-war-plugin</artifactId>
            <version>3.2.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-install-plugin</artifactId>
            <version>2.5.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-deploy-plugin</artifactId>
            <version>2.8.2</version>
        </plugin>
    </plugins>
</pluginManagement>
<build>
    </build>
</project>

```

## 7. CONCLUSION:

The Airways FlyAway project was created in 3 sprints. Special attention was paid to implement the concept of servlet, hibernation. Best practices were followed as and when possible. Rigorous testing was done to ensure that there are no spontaneous exits, and all exceptions are handled. Some Exceptions are handled using Custom exception classes. The throws and throw keywords were used to handle exceptions.