

# FlyAway Proj

## Project Source Code

### Version History:

Author:	Balanarenthiran
Purpose:	Project Source Code and GITHUB
Date:	2 <sup>nd</sup> Jan, 2022
Version:	1.0

## Table of Content:

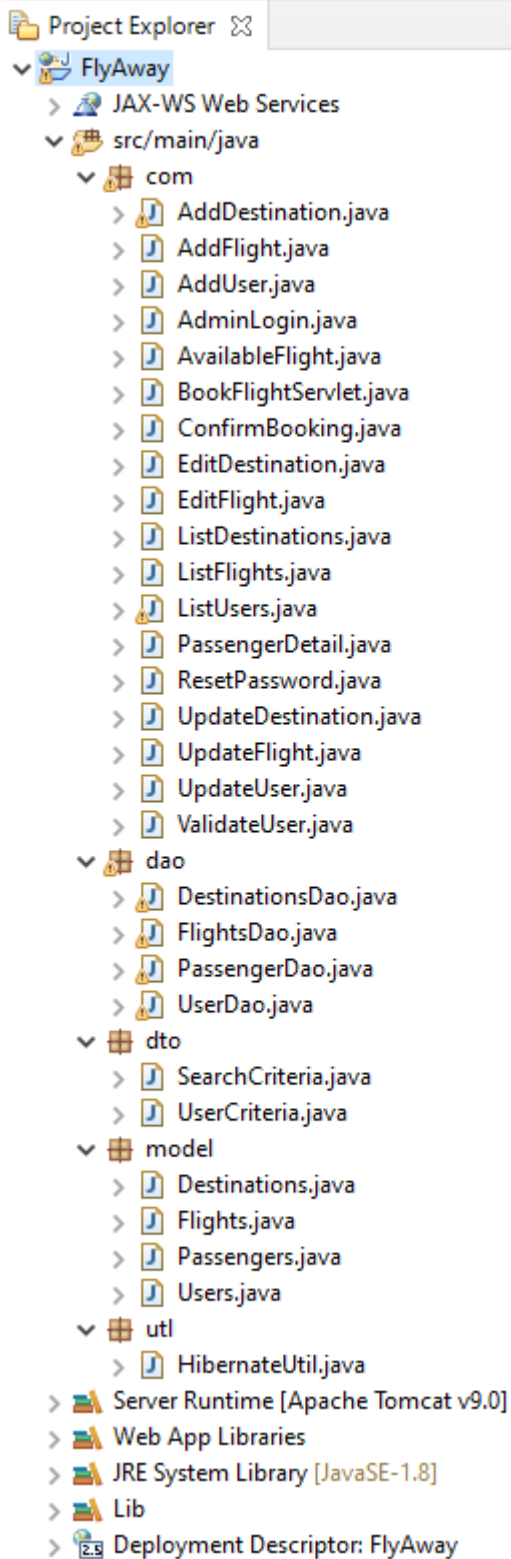
Project GITHUB Link: .....	3
Project Code: .....	4

## Project GITHUB Link:

Repository Name:	Course3-FlyAwayProj
Github Link:	<a href="https://github.com/bala-tw/Course3-FlyAwayProj">https://github.com/bala-tw/Course3-FlyAwayProj</a>

## Project Code:

### Folder Structure

- 
- The screenshot shows the Project Explorer of an IDE. The project is named 'FlyAway'. It contains several folders and files:
- Project Explorer (icon)
  - FlyAway
    - JAX-WS Web Services (icon)
    - src/main/java (icon)
      - com (icon)
        - AddDestination.java (icon)
        - AddFlight.java (icon)
        - AddUser.java (icon)
        - AdminLogin.java (icon)
        - AvailableFlight.java (icon)
        - BookFlightServlet.java (icon)
        - ConfirmBooking.java (icon)
        - EditDestination.java (icon)
        - EditFlight.java (icon)
        - ListDestinations.java (icon)
        - ListFlights.java (icon)
        - ListUsers.java (icon)
        - PassengerDetail.java (icon)
        - ResetPassword.java (icon)
        - UpdateDestination.java (icon)
        - UpdateFlight.java (icon)
        - UpdateUser.java (icon)
        - ValidateUser.java (icon)
      - dao (icon)
        - DestinationsDao.java (icon)
        - FlightsDao.java (icon)
        - PassengerDao.java (icon)
        - UserDao.java (icon)
      - dto (icon)
        - SearchCriteria.java (icon)
        - UserCriteria.java (icon)
      - model (icon)
        - Destinations.java (icon)
        - Flights.java (icon)
        - Passengers.java (icon)
        - Users.java (icon)
      - util (icon)
        - HibernateUtil.java (icon)
    - Server Runtime [Apache Tomcat v9.0] (icon)
    - Web App Libraries (icon)
    - JRE System Library [JavaSE-1.8] (icon)
    - Lib (icon)
    - Deployment Descriptor: FlyAway (icon)

> build  
▼ src  
  ▼ main  
    > java  
    ▼ webapp  
      > META-INF  
      ▼ WEB-INF  
        > lib  
          X web.xml  
          add-destination.html  
          add-flight.html  
          add-user.html  
          adminpage.jsp  
          availableflight.html  
          bookingdetails.jsp  
          destination-form.jsp  
          destinationlist.jsp  
          flight-form.jsp  
          flightinfo.html  
          flightlist.jsp  
          index.html  
          invalidlogin.html  
          login.html  
          passengerdetail.jsp  
          payment.html  
          PNR.jsp  
          search-flight.jsp  
          user-form.jsp  
          user-list.jsp

index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Fly Away</title>
</head>
</br>
</br>
</br>
<body>
  <h1>Fly Away</h1>
  <h1>Booking Portal</h1>
    <table>
      <tr></tr>
      <tr></tr>
      <tr>
        <td>
          <a href="BookFlightServlet">Book Flight</a>
```

```

        </td>
    </tr>
</table>
</form>
<table>
<tr></tr>
<tr></tr>
<tr>
    <td>
        <a href="AdminLogin">Admin Login</a>
    </td>
</tr>
</table>
</body>
</html>

```

### BookFlightServlet

```

package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class BookFlightServlet
 */
public class BookFlightServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public BookFlightServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        response.sendRedirect("flightinfo.html");
    }
}

```

flightinfo.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
    <link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-
ui.css">
    <script src="https://code.jquery.com/jquery-1.12.4.js"></script>
    <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
    <script> $( function() {$( "#datepicker" ).datepicker(); } ); </script>

</head>
<body>
    <h1>Fly Away</h1>
    <h2>Booking Portal</h2>
    <h3>Flight Search</h3>
    <form action="AvailableFlight" method="post">

        <table border="1" cellpadding="5">
            <tr>
                <td>Enter Date of Travel:</td>
                <td><input type="date" class="form-control" id="datepicker"
name="udate"></td>
            </tr>
            <tr>
                <td>Traveling From:</td>
                <td>
                    <select name="usource">
                        <option value="INDIA">India</option>
                        <option value="USA">USA</option>
                        <option value="UK">UK</option>
                    </select>
                </td>
            </tr>
            <tr>
                <td>Destination:</td>
                <td>
                    <select name="udestination">
                        <option value="UK">UK</option>
                        <option value="USA">USA</option>
                        <option value="INDIA">India</option>
                    </select>
                </td>
            </tr>
        </table>
    </form>
</body>
```

```

        <tr>
        <td>Number of Persons:</td>
        <td>
            <select name="upersons">
                <option value="1">1</option>
                <option value="2">2</option>
                <option value="3">3</option>
                <option value="4">4</option>
                <option value="5">5</option>
                <option value="6">6</option>
            </select>
        </td>
        </tr>

        <tr>
            <td></td>
            <td><input type="submit" value="Next"></td>
        </tr>
    </table>
</form>
</body>
</html>

```

#### AvailableFlight

```

package com;

import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import model.Flights;
import dao.FlightsDao;

import dto.SearchCriteria;

/**
 * Servlet implementation class AvailableFlight
 */
public class AvailableFlight extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private FlightsDao flightsDao;
    public void init() {

```



```

        flightsDao = new FlightsDao();
    }
    /**
     * @see HttpServlet#HttpServlet()
     */
    public AvailableFlight() {
        super();
        // TODO Auto-generated constructor stub
    }

    protected void service(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        SearchCriteria sc = new SearchCriteria();
        HttpSession session = request.getSession();

        try {
            Date date = new Date();
            SimpleDateFormat formatter = new
SimpleDateFormat("ddMMyyyyhhmmss");
            sc.setPnr(formatter.format(date));
            String dateStr = request.getParameter("udate");
            SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");
            sc.setUdate(sdf.parse(dateStr));
            sc.setUsource(request.getParameter("usource"));
            sc.setUdestination(request.getParameter("udestination"));

            sc.setUpersons(Integer.parseInt(request.getParameter("upersons")));
            session.setAttribute("searchcriteria", sc);

            List < Flights > listFlights =
flightsDao.getFlights(sc.getUsource(), sc.getUdestination(), sc.getUdate());
            request.setAttribute("listFlights", listFlights);
            RequestDispatcher dispatcher = request.getRequestDispatcher("search-
flight.jsp");
            dispatcher.forward(request, response);
        }
        catch(Exception ex) {
            throw new ServletException(ex);
        }
    }
}

```

search-flight.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Flyaway</title>
</head>

```

```

<body>
<center>
<h1>Fly Away</h1>
<h2>Available Flights</h2>
</center>
<div align="center">
<form action="PassengerDetail" method="post">
<table border="1" cellpadding="5">
<caption><h2>List of available flights</h2></caption>
<tr>
<th>Flight</th>
<th>Source</th>
<th>Destination</th>
<th>Date</th>
<th>Ticket Price</th>
<th>Select Flight</th>
</tr>
<c:forEach var="flights" items="${listFlights}">
<tr>
<td><c:out value="${flights.flight}" /></td>
<td><c:out value="${flights.source}" /></td>
<td><c:out value="${flights.destination}" /></td>
<td><c:out value="${flights.date}" /></td>
<td><c:out value="${flights.ticketprice}" /></td>
<td>
<a href="PassengerDetail?id=<c:out value='${flights.id}' />">Select</a>
</td>
</tr>
</c:forEach>
</table>
</form>
</div>
</body>
</html>

```

### PassengerDetail

```
package com;
```

```
import java.io.IOException;
import java.util.List;
```

```
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
```

```
import dto.SearchCriteria;
import model.Flights;
import model.Passengers;
```

```

import dao.PassengerDao;
import dao.FlightsDao;

/**
 * Servlet implementation class PassengerDetail
 */
@WebServlet("/PassengerDetail")
public class PassengerDetail extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private PassengerDao passengerDao;
    private FlightsDao flightsDao;

    public void init() {
        passengerDao = new PassengerDao();
        flightsDao = new FlightsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public PassengerDetail() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        //SearchCriteria sc = new SearchCriteria();
        HttpSession session = request.getSession();

        try {
            SearchCriteria sc =(SearchCriteria)
session.getAttribute("searchcriteria");
            int nop = sc.getUpersons();
            int padded = sc.getPadded();
            if (padded==0) {
                sc.setFid(Integer.parseInt(request.getParameter("id")));
                padded++;
                sc.setPadded(padded);
                response.sendRedirect("passengerdetail.jsp");
            }
            else {
                String name = request.getParameter("name");
                String email = request.getParameter("email");
                String country = request.getParameter("country");
                int fid = sc.getFid();

```

```

        String pnr = sc.getPnr();
        String sid = session.getId();
        Passengers newPassengers = new Passengers(name, email, country, fid,
sid, pnr);

        passengerDao.saveUser(newPassengers);
        padded++;
        sc.setPadded(padded);
            if (padded <= nop) {
                response.sendRedirect("passengerdetail.jsp");

            } else {

                List < Flights > listFlights =
flightsDao.getSelectedFlight(fid);
                request.setAttribute("listFlights", listFlights);

                List < Passengers > listPassengers =
passengerDao.getPassengerDetails(pnr);
                request.setAttribute("listPassengers", listPassengers);

sc.setUticketprice(flightsDao.getFlgihtFare(fid).getTicketprice());

                sc.setTotalfare(sc.getUpersons()*sc.getUticketprice());

                RequestDispatcher dispatcher =
request.getRequestDispatcher("bookingdetails.jsp");
                dispatcher.forward(request, response);
            }
    }
    catch(Exception ex) {
        throw new ServletException(ex);

    }
}

```

passengerdetail.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Fly Away</title>
</head>
<body>
<center>
<h1>Fly Away</h1>
<h2>Add Passenger Details</h2>
```

```

</center>
<div align="center">
  <form action="PassengerDetail" method="post">
    <table border="1" cellpadding="5">
      <tr>
        <th>Passenger Name: </th>
        <td>
          <input type="text" name="name" size="45"
            value="<c:out value='${passengers.name}' />"
          />
        </td>
      </tr>
      <tr>
        <th>Passenger Email: </th>
        <td>
          <input type="text" name="email" size="45"
            value="<c:out value='${passengers.email}' />"
          />
        </td>
      </tr>
      <tr>
        <th>Country: </th>
        <td>
          <input type="text" name="country" size="15"
            value="<c:out value='${passengers.country}' />"
          />
        </td>
      </tr>
      <tr>
        <td colspan="2" align="center">
          <input type="submit" value="Add" />
        </td>
      </tr>
    </table>
  </form>
</div>
</body>
</html>

```

bookingdetails.jsp

```

<%@page import="javax.servlet.http.HttpSession" %>
<%@page import="dto.SearchCriteria" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
  pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<% SearchCriteria sc =(SearchCriteria) session.getAttribute("searchcriteria"); %>
<html>
<head>
  <title>Fly Away</title>
</head>
<body>

```

```

<center>
<h1>Fly Away</h1>
<h2>Booking Details</h2>
</center>
<div align="center">
<form action="payment.html" method="post">
<table border="1" cellpadding="5">
<caption><h2>Your flight details</h2></caption>
<caption><h3>PNR # <% out.print(sc.getPnr()); %> </h3></caption>
<tr>
<th>Flight</th>
<th>Source</th>
<th>Destination</th>
<th>Date</th>
<th>Ticket Price</th>
</tr>
<c:forEach var="flights" items="${listFlights}">
<tr>
<td><c:out value="${flights.flight}" /></td>
<td><c:out value="${flights.source}" /></td>
<td><c:out value="${flights.destination}" /></td>
<td><c:out value="${flights.date}" /></td>
<td><c:out value="${flights.ticketprice}" /></td>
</tr>
</c:forEach>
</table>
<table border="1" cellpadding="5">
<caption><h2>Passengers details</h2></caption>
<tr>
<th>Passenger Name</th>
<th>Email</th>
<th>Country</th>
</tr>
<c:forEach var="passengers" items="${listPassengers}">
<tr>
<td><c:out value="${passengers.name}" /></td>
<td><c:out value="${passengers.email}" /></td>
<td><c:out value="${passengers.country}" /></td>
</tr>
</c:forEach>
</table>
<table border="1" cellpadding="5">
<caption><h2>Fare Details</h2></caption>
<tr><td>Flight Fare Per Passenger</td>
<td>
<%
out.print(sc.getUticketprice());
%>
</td>
</tr>

```

```

        <tr>
            <td>No of Passengers</td>
            <td>
                <%
                //SearchCriteria sc =(SearchCriteria) session.getAttribute("searchcriteria");
                //out.print(request.getParameter("upersons"));
                out.print(sc.getUpersons());
                int x = sc.getUpersons();
                %>
            </td>
        </tr>
        <tr>
            <td>Total Fare</td>
            <td>
                <%
                out.print(sc.getTotalfare());
                %>
            </td>
        </tr>
        <tr>
            <td>Paid Amount</td>
            <td>0</td>
        </tr>
        <tr>
            <td>Balance</td>
            <td>
                <%
                out.print(sc.getTotalfare());
                %>
            </td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" value="Payment"></td>
        </tr>
    </table>
</form>
</div>
</body>
</html>

```

payment.html

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>

</head>
<body>
    <h1>Fly Away</h1>

```

```

<h2>Booking Portal</h2>
<h3>Payment Detail</h3>
<form action="ConfirmBooking" method="post">
<table border="1" cellpadding="5">
  <tr>
    <td>Credit/Debit Card No:</td>
    <td> <input type="text" name="ccno"></td>
  </tr>
  <tr>
    <td>Name on the Card:</td>
    <td><input type="text" name="cname"></td>
  </tr>
  <tr>
    <td>Card Exipry:</td>
    <td> <select name="emonth">
      <option value="Jan">Jan</option>
      <option value="Feb">Feb</option>
      <option value="Mar">Mar</option>
      <option value="Apr">Apr</option>
      <option value="May">May</option>
      <option value="Jun">Jun</option>
      <option value="Jul">Jul</option>
      <option value="Aug">Aug</option>
      <option value="Sep">Sep</option>
      <option value="Oct">Oct</option>
      <option value="Nov">Nov</option>
      <option value="Dec">Dec</option>
    </select>
    <select name="eyear">
      <option value="2021">2021</option>
      <option value="2022">2022</option>
      <option value="2023">2023</option>
      <option value="2024">2024</option>
      <option value="2025">2025</option>
    </select>
  </td>
  </tr>
  <tr><td></td>
    <td><input type="submit" name="click" value="Pay"></td>
  </tr>
</table>
</form>
</body>
</html>

```

```
ConfirmBooking
```

```
package com;
```

```
import java.io.IOException;
import java.util.List;
```



```

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dao.FlightsDao;
import dao.PassengerDao;
import dto.SearchCriteria;
import model.Flights;
import model.Passengers;

/**
 * Servlet implementation class ConfirmBooking
 */
@WebServlet("/ConfirmBooking")
public class ConfirmBooking extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private PassengerDao passengerDao;
    private FlightsDao flightsDao;

    public void init() {
        passengerDao = new PassengerDao();
        flightsDao = new FlightsDao();
    }
    /**
     * @see HttpServlet#HttpServlet()
     */
    public ConfirmBooking() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        // TODO Auto-generated method stub
        HttpSession session = request.getSession();

        try {
            SearchCriteria sc =(SearchCriteria)
session.getAttribute("searchcriteria");

            int fid = sc.getFid();
            String pnr = sc.getPnr();

```

```

        List < Flights > listFlights =
flightsDao.getSelectedFlight(fid);
        request.setAttribute("listFlights", listFlights);

        List < Passengers > listPassengers =
passengerDao.getPassengerDetails(pnr);
        request.setAttribute("listPassengers", listPassengers);

sc.setUticketprice(flightsDao.getFlightFare(fid).getTicketprice());

        sc.setTotalfare(sc.getUpersons()*sc.getUticketprice());

        RequestDispatcher dispatcher =
request.getRequestDispatcher("PNR.jsp");
        dispatcher.forward(request, response);
    } catch (Exception ex) {
        throw new ServletException(ex);
    }
}

```

#### PNR.jsp

```

<%@page import="javax.servlet.http.HttpSession" %>
<%@page import="dto.SearchCriteria" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<% SearchCriteria sc =(SearchCriteria) session.getAttribute("searchcriteria"); %>
<html>
<head>
    <title>Fly Away</title>
</head>
<body>
    <center>
        <h1>Fly Away</h1>
        <h2>Booking Details</h2>
    </center>
    <div align="center">
        <form action="payment.html" method="post">
            <table border="1" cellpadding="5">
                <caption><h2>Your Booking is Confirmed</h2></caption>
                <caption><h3>PNR # <% out.print(sc.getPnr()); %> </h3></caption>
                <tr>
                    <th>Flight</th>
                    <th>Source</th>
                    <th>Destination</th>
                    <th>Date</th>
                    <th>Ticket Price</th>
                </tr>
                <c:forEach var="flights" items="${listFlights}">

```

```

        <tr>
            <td><c:out value="{{flights.flight}}"/></td>
            <td><c:out value="{{flights.source}}"/></td>
            <td><c:out value="{{flights.destination}}"/></td>
            <td><c:out value="{{flights.date}}"/></td>
            <td><c:out value="{{flights.ticketprice}}"/></td>
        </tr>
    </c:forEach>
</table>
<table border="1" cellpadding="5">
    <caption><h2>Passengers details</h2></caption>
    <tr>
        <th>Passenger Name</th>
        <th>Email</th>
        <th>Country</th>
    </tr>
    <c:forEach var="passengers" items="{{listPassengers}}">
        <tr>
            <td><c:out value="{{passengers.name}}"/></td>
            <td><c:out value="{{passengers.email}}"/></td>
            <td><c:out value="{{passengers.country}}"/></td>
        </tr>
    </c:forEach>
</table>
<table border="1" cellpadding="5">
    <caption><h2>Fare Details</h2></caption>
    <tr><td>Flight Fare Per Passenger</td>
        <td>
            <td>
                <%
                out.print(sc.getUticketprice());
                %>
            </td>
        </td>
    </tr>
    <tr>
        <td>No of Passengers</td>
        <td>
            <%
            //SearchCriteria sc =(SearchCriteria) session.getAttribute("searchcriteria");
            //out.print(request.getParameter("upersons"));
            out.print(sc.getUpersons());
            int x = sc.getUpersons();
            %>
        </td>
    </tr>
    <tr>
        <td>Total Fare</td>
        <td>
            <td>
                <%
                out.print(sc.getTotalfare());
                %>
            </td>
        </td>
    </tr>
</table>

```

```

        </td>
    </tr>
    <tr>
        <td>Paid Amount</td>
        <td>
            <%
                out.print(sc.getTotalfare());
            %>
        </td>
    </tr>
    <tr>
        <td>Balance</td>
        <td>0</td>
    </tr>
</table>
</form>
</div>
</body>
</html>

```

#### AdminLogin

```

package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dto.UserCriteria;
/**
 * Servlet implementation class AdminLogin
 */
//@WebServlet("/AdminLogin")
public class AdminLogin extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AdminLogin() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request,
     HttpServletResponse response)
     */

```

```

        protected void service(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
            HttpSession session = request.getSession();
            UserCriteria uc = new UserCriteria();
            session.setAttribute("usercriteria", uc);
            response.sendRedirect("login.html");
        }
    }
}

```

#### login.html

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>

</head>
<body>
    <h1>Fly Away</h1>
    <h2>Admin Login</h2>
    <form action="ValidateUser" method="post">
        <table border="1" cellpadding="5">
            <tr>
                <td>Login ID:</td>
                <td><input type="text" placeholder="Enter Username" name="username"
required></td>
            </tr>
            <tr>
                <td>Password:</td>
                <td><input type="password" placeholder="Enter Password"
name="userpassword" required></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Login"></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

#### ValidateUser

```

package com;

import java.io.IOException;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

```

```

import dao.UserDao;

/**
 * Servlet implementation class ValidateUser
 */
@WebServlet("/ValidateUser")
public class ValidateUser extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private UserDao userDao;

    public void init() {
        userDao = new UserDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ValidateUser() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        //HttpSession session = request.getSession();
        try {
            System.out.println("validate user servlet");
            String un=request.getParameter("username");
            String up=request.getParameter("userpassword");
            boolean result = userDao.getSelectedUser(un, up);
            System.out.println(result);
            if (result) {
                response.sendRedirect("adminpage.jsp");
            } else {
                response.sendRedirect("invalidlogin.html");
            }
        } catch (Exception e) {
            throw new ServletException(e);
        }
        //e.printStackTrace();
    }
}

```

invalidlogin.html

```

<!DOCTYPE html>
<html>
<head>

```

```

<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
    <h1>Fly Away</h1>
    <h2>Admin Login</h2>
    <form action="ValidateUser" method="post">
        <table border="1" cellpadding="5">
            <tr>
                <td></td>
                <td><p style="color:red">Invalid User Name or Passowrd</p></td>
            <tr>
                <td>Login ID:</td>
                <td><input type="text" placeholder="Enter Username" name="username"
required></td>
            </tr>
            <tr>
                <td>Password:</td>
                <td><input type="password" placeholder="Enter Password"
name="userpassword" required></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Login"></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

#### adminpage.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
    <h1>Fly Away</h1>
    <h2>Admin Page</h2>
    <div align="left">
        <table border="1" cellpadding="5">
            <tr>
                <td>Manage Users</td>
                <td><a href="ListUsers">View List</a></td>
                <td><a href="add-user.html">Add New</a></td>
            </tr>

```

```

        <tr>
            <td>Manage Flights</td>
            <td><a href="ListFlights">View List</a></td>
            <td><a href="add-flight.html">Add New</a></td>
        </tr>
        <tr>
            <td>Manage Source/Destinations</td>
            <td><a href="ListDestinations">View List</a></td>
            <td><a href="add-destination.html">Add New</a></td>
        </tr>
    </table>
</div>
</body>
</html>

```

## ListUsers

```

package com;

import java.io.IOException;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.UserDao;
import model.Users;

/**
 * Servlet implementation class ListUsers
 */
@WebServlet("/ListUsers")
public class ListUsers extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private UserDao userDao;

    public void init() {
        userDao = new UserDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ListUsers() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**

```



```

    * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
    */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        List < Users > listUser = userDao.getAllUser();
        request.setAttribute("listUser", listUser);
        RequestDispatcher dispatcher = request.getRequestDispatcher("user-list.jsp");
        dispatcher.forward(request, response);
    }
}

```

#### user-list.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
    <title>Flyaway</title>
</head>
<body>
    <center>
        <h1>Fly Away</h1>
        <h2>User Management</h2>
        <h3><a href="adminpage.jsp">go to main page</a></h3>
    </center>
    <div align="center">
        <table border="1" cellpadding="5">
            <caption><h2>List of Users</h2></caption>
            <tr>
                <th>ID</th>
                <th>Name</th>
                <th>Select</th>
            </tr>
            <c:forEach var="user" items="${listUser}">
                <tr>
                    <td><c:out value="${user.id}" /></td>
                    <td><c:out value="${user.name}" /></td>
                    <td>
                        <a href="ResetPassword?id=<c:out value='${user.id}' />">Reset Password</a>
                    </td>
                </tr>
            </c:forEach>
        </table>
    </div>
</body>
</html>

```

#### ResetPassword

```

package com;

import java.io.IOException;

```

```

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dto.UserCriteria;

/**
 * Servlet implementation class ResetPassword
 */
@WebServlet("/ResetPassword")
public class ResetPassword extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ResetPassword() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        HttpSession session = request.getSession();
        UserCriteria uc =(UserCriteria) session.getAttribute("usercriteria");
        uc.setUid(Integer.parseInt(request.getParameter("id")));
        session.setAttribute("usercriteria", uc);
        response.sendRedirect("user-form.jsp");
    }
}

```

user-form.jsp

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
    <h1>Fly Away</h1>
    <h2>Password Reset</h2>
    <form action="UpdateUser" method="post">
        <table border="1" cellpadding="5">
            <tr>
                <td>Type New Password:</td>

```

```

                <td><input type="password" placeholder="Enter Password"
name="upassword" required></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Reset Password"></td>
            </tr>
        </table>
    </form>
</body>
</html>

```

### UpdateUser

```

package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dao.UserDao;
import dto.UserCriteria;

/**
 * Servlet implementation class UpdateUser
 */
@WebServlet("/UpdateUser")
public class UpdateUser extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private UserDao userDao;
    public void init() {
        userDao = new UserDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public UpdateUser() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        HttpSession session = request.getSession();

```

```
        UserCriteria uc =(UserCriteria) session.getAttribute("usercriteria");
        String up = request.getParameter("upassword");
        userDao.updateUser(uc.getUid(), up);
        response.sendRedirect("adminpage.jsp");
    }
}
```

#### add-user.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
    <h1>Fly Away</h1>
    <h2>Add New User</h2>
    <form action="AddUser" method="post">
        <table border="1" cellpadding="5">
            <tr>
                <td>Login ID:</td>
                <td><input type="text" placeholder="Enter Username" name="username"
required></td>
            </tr>
            <tr>
                <td>Password:</td>
                <td><input type="password" placeholder="Enter Password"
name="nuserpassword" required></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Add"></td>
            </tr>
        </table>
    </form>
</body>
</html>
```

AddUser

```

package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.UserDao;
import model.Users;

/**
 * Servlet implementation class AddUser
 */
public class AddUser extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private UserDao userDao;

    public void init() {
        userDao = new UserDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AddUser() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

        String nun = request.getParameter("username");
        String nup = request.getParameter("userpassword");

        Users user = new Users(nun,nup);
        userDao.addUser(user);
        response.sendRedirect("adminpage.jsp");
    }

```

```
}
```

```
ListFlights
```

```

package com;

import java.io.IOException;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.FlightsDao;
import model.Flights;

/**
 * Servlet implementation class ListFlights
 */
@WebServlet("/ListFlights")
public class ListFlights extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private FlightsDao flightsDao;

    public void init() {
        flightsDao = new FlightsDao();
    }
    /**
     * @see HttpServlet#HttpServlet()
     */
    public ListFlights() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        List < Flights > listFlights = flightsDao.getAllFlights();
        request.setAttribute("listFlights", listFlights);
        RequestDispatcher dispatcher = request.getRequestDispatcher("flightlist.jsp");
        dispatcher.forward(request, response);
    }

```



```
}
```

```
flightlist.jsp
```

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
    <title>Flyaway</title>
</head>
<body>
    <center>
        <h1>Fly Away</h1>
        <h2>Manage Flights</h2>
        <h3><a href="adminpage.jsp">go to main page</a></h3>
    </center>
    <div align="center">
        <form action="EditFlight" method="post">
            <table border="1" cellpadding="5">
                <caption><h2>List of flights</h2></caption>
                <tr>
                    <th>Flight</th>
                    <th>Source</th>
                    <th>Destination</th>
                    <th>Date</th>
                    <th>Ticket Price</th>
                    <th>Select Flight</th>
                </tr>
                <c:forEach var="flights" items="${listFlights}">
                    <tr>
                        <td><c:out value="${flights.flight}" /></td>
                        <td><c:out value="${flights.source}" /></td>
                        <td><c:out value="${flights.destination}" /></td>
                        <td><c:out value="${flights.date}" /></td>
                        <td><c:out value="${flights.ticketprice}" /></td>
                        <td>
                            <a href="EditFlight?fid=<c:out value='${flights.id}' />">Edit Details</a>
                        </td>
                    </tr>
                </c:forEach>
            </table>
        </form>
    </div>
</body>
</html>
```

EditFlight

```

package com;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dao.FlightsDao;
import dto.UserCriteria;
import model.Flights;

/**
 * Servlet implementation class EditFlight
 */
@WebServlet("/EditFlight")
public class EditFlight extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private FlightsDao flightsDao;

    public void init() {
        flightsDao = new FlightsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public EditFlight() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        HttpSession session = request.getSession();
        UserCriteria uc =(UserCriteria) session.getAttribute("usercriteria");
        uc.setFid(Integer.parseInt(request.getParameter("fid")));
        Flights listFlights = flightsDao.getFlightFare(uc.getFid());
    }
}

```

```

        System.out.println(listFlights.getDestination());
        request.setAttribute("listFlights", listFlights);
        session.setAttribute("usercriteria", uc);
        RequestDispatcher dispatcher = request.getRequestDispatcher("flight-form.jsp");
        dispatcher.forward(request, response);
    }
}

```

#### flight-form.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Flyaway</title>
</head>
<body>
<center>
<h1>Fly Away</h1>
<h2>Flights Management</h2>
</center>
<div align="center">
<form action="UpdateFlight" method="post">
<table border="1" cellpadding="5">
<caption>
<h2>Edit Flight Details</h2>
</caption>
<input type="hidden" name="id" value="<c:out value='${listFlights.id}' />" />
<tr>
<th>Flight: </th>
<td>
<input type="text" name="flight" size="45" value="<c:out
value='${listFlights.flight}' />" />
</td>
</tr>
<tr>
<th>Flight Date: </th>
<td>
<input type="date" name="date" size="45" value="<c:out
value='${listFlights.date}' />" />
</td>
</tr>
<tr>
<th>Source: </th>
<td>
<input type="text" name="source" size="45" value="<c:out
value='${listFlights.source}' />" />
</td>
</tr>
<tr>

```

```
<th>Destination: </th>
<td>
  <input type="text" name="destination" size="15" value="<c:out
value='${listFlights.destination}' />" />
</td>
</tr>
<tr>
<th>Fare: </th>
<td>
  <input type="number" name="ticketprice" size="15" value="<c:out
value='${listFlights.ticketprice}' />" />
</td>
</tr>
<tr>
<td colspan="2" align="center">
  <input type="submit" value="Update" />
</td>
</tr>
</table>
</form>
</div>
</body>
</html>
```

UpdateFlight

```

package com;

import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.FlightsDao;
import model.Flights;

/**
 * Servlet implementation class UpdateFlight
 */
@WebServlet("/UpdateFlight")
public class UpdateFlight extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private FlightsDao flightsDao;

    public void init() {
        flightsDao = new FlightsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public UpdateFlight() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        try {
            int fid = Integer.parseInt(request.getParameter("id"));
            String flight = request.getParameter("flight");
            String source = request.getParameter("source");
            String destination = request.getParameter("destination");
            String dateStr = request.getParameter("date");

```

```

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");
Date udt =sdf.parse(dateStr);
int tp = Integer.parseInt(request.getParameter("ticketprice"));

Flights flightinfo = new Flights(fid, flight, source, destination, udt, tp);
flightsDao.updateFlight(flightinfo);
response.sendRedirect("adminpage.jsp");
} catch(Exception ex) {
throw new ServletException(ex);

}
}
}

```

#### add-flight.html

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>

<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-
ui.css">
<script src="https://code.jquery.com/jquery-1.12.4.js"></script>
<script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
<script> $( function() {$( "#datepicker" ).datepicker() }); </script>

</head>
<body>
<h1>Fly Away</h1>
<h2>Booking Portal</h2>
<h3>Flight Search</h3>
<form action="AddFlight" method="post">

<table border="1" cellpadding="5">
<tr>
<td>Flight Code:</td>
<td>
<input type="text" name="nflightcode">
</td>
</tr>
<tr>
<td>Flight Date:</td>
<td><input type="datetime-local" class="form-control" id="datepicker"
name="ndate"></td>
</tr>
<tr>

```



```
<td>Source:</td>
<td>
    <select name="nsource">
        <option value="INDIA">India</option>
        <option value="USA">USA</option>
        <option value="UK">UK</option>
    </select>
</td>
</tr>
<tr>
<td>Destination:</td>
<td>
    <select name="ndestination">
        <option value="UK">UK</option>
        <option value="USA">USA</option>
        <option value="INDIA">India</option>
    </select>
</td>
</tr>
<tr>
<td>Ticket Fare:</td>
<td>
    <input type="number" name="nticketprice">
</td>
</tr>
<tr>
<td></td>
<td><input type="submit" value="Add"></td>
</tr>
</table>
</form>
</body>
</html>
```

AddFlight

```

package com;

import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.FlightsDao;
import model.Flights;

/**
 * Servlet implementation class AddFlight
 */
@WebServlet("/AddFlight")
public class AddFlight extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private FlightsDao flightsDao;

    public void init() {
        flightsDao = new FlightsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AddFlight() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        try {
            String flight = request.getParameter("nflightcode");
            String source = request.getParameter("nsource");
            String destination = request.getParameter("ndestination");
            String dateStr = request.getParameter("ndate");
            SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");

```

```
        Date udt =sdf.parse(dateStr);
        int tp = Integer.parseInt(request.getParameter("nticketprice"));

        Flights fl = new Flights(flight, source, destination, udt, tp);
        flightsDao.addFlight(fl);
        response.sendRedirect("adminpage.jsp");
    } catch(Exception ex) {
        throw new ServletException(ex);
    }
}
```

ListDestinations

```

package com;

import java.io.IOException;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.DestinationsDao;
import model.Destinations;

/**
 * Servlet implementation class ListDestinations
 */
@WebServlet("/ListDestinations")
public class ListDestinations extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private DestinationsDao destinationsDao;

    public void init() {
        destinationsDao = new DestinationsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ListDestinations() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        List < Destinations > listDestinations = destinationsDao.getAllDestintions();
        request.setAttribute("listDestinations", listDestinations);
        RequestDispatcher dispatcher = request.getRequestDispatcher("destinationlist.jsp");
        dispatcher.forward(request, response);
    }
}

```

```
}
```

```
destinationlist.jsp
```

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<html>
<head>
<title>Flyaway</title>
</head>
<body>
<center>
<h1>Fly Away</h1>
<h2>Source/Destinations Management</h2>
<h3><a href="adminpage.jsp">go to main page</a></h3>
</center>
<div align="center">
<table border="1" cellpadding="5">
<caption><h2>List of Source/Destinations</h2></caption>
<tr>
<th>ID</th>
<th>Destination</th>
</tr>
<c:forEach var="destination" items="${listDestinations}">
<tr>
<td><c:out value="${destination.id}" /></td>
<td><c:out value="${destination.destination}" /></td>
<td>
<a href="EditDestination?did=<c:out value='${destination.id}' />">Edit</a>
</td>
</tr>
</c:forEach>
</table>
</div>
</body>
</html>
```

EditDestination

```

package com;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dao.DestinationsDao;
import dto.UserCriteria;
import model.Destinations;

/**
 * Servlet implementation class EditDestination
 */
@WebServlet("/EditDestination")
public class EditDestination extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private DestinationsDao destinationsDao;

    public void init() {
        destinationsDao = new DestinationsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public EditDestination() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        HttpSession session = request.getSession();
        UserCriteria uc =(UserCriteria) session.getAttribute("usercriteria");
        uc.setDid(Integer.parseInt(request.getParameter("did")));
    }

```



```
        Destinations listDestinations = destinationsDao.getDestination(uc.getDid());
        request.setAttribute("listDestinations", listDestinations);
        session.setAttribute("usercriteria", uc);
        RequestDispatcher dispatcher = request.getRequestDispatcher("destination-form.jsp");
        dispatcher.forward(request, response);
    }
}
```

#### destination-form.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
    <h1>Fly Away</h1>
    <h2>Update Source/Destination Info.</h2>
    <form action="UpdateDestination" method="post">
        <table border="1" cellpadding="5">
            <tr>
                <td>Source/Destination:</td>
                <td><input type="text" name="udestination" required></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Update"></td>
            </tr>
        </table>
    </form>
</body>
</html>
```

UpdateDestination

```

package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import dao.DestinationsDao;
import dto.UserCriteria;

/**
 * Servlet implementation class UpdateDestination
 */
@WebServlet("/UpdateDestination")
public class UpdateDestination extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private DestinationsDao destinationsDao;

    public void init() {
        destinationsDao = new DestinationsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public UpdateDestination() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        HttpSession session = request.getSession();
        UserCriteria uc =(UserCriteria) session.getAttribute("usercriteria");
        String ud = request.getParameter("udestination");
        destinationsDao.updateDestination(uc.getDid(), ud);
        response.sendRedirect("adminpage.jsp");
    }
}

```

add-destination.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Flyaway</title>
</head>
<body>
  <h1>Fly Away</h1>
  <h2>Add New Source/Destinaion</h2>
  <form action="AddDestination" method="post">
    <table border="1" cellpadding="5">
      <tr>
        <td>Source/Destination Name:</td>
        <td><input type="text" name="ndestination" required></td>
      </tr>
      <tr>
        <td></td>
        <td><input type="submit" value="Add"></td>
      </tr>
    </table>
  </form>
</body>
</html>
```

## AddDestination

```
package com;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.DestinationsDao;
import model.Destinations;

/**
 * Servlet implementation class AddDestination
 */
@WebServlet("/AddDestination")
public class AddDestination extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private DestinationsDao destinationsDao;

    public void init() {
        destinationsDao = new DestinationsDao();
    }

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AddDestination() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        String nd = request.getParameter("ndestination");
        Destinations destination = new Destinations(nd);
        destinationsDao.addDestination(destination);
        response.sendRedirect("adminpage.jsp");
    }
}
```

dao.DestinationsDao

```

package dao;

import java.util.List;

import org.hibernate.Session;
import org.hibernate.Transaction;

import model.Destinations;
import utl.HibernateUtil;

public class DestinationsDao {

    /**
     * Get all Users
     * @return
     */
    @SuppressWarnings("unchecked")
    public List < Destinations > getAllDestintions() {

        Transaction transaction = null;
        List < Destinations > listOfDestinations = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            // start a transaction
            transaction = session.beginTransaction();
            // get an user object

            listOfDestinations = session.createQuery("from Destinations").getResultList();

            // commit transaction
            transaction.commit();
        } catch (Exception e) {
            if (transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        }
        return listOfDestinations;
    }
    /**
     * Get Destination By Destination ID
     * @param did
     * @return
     */
}

```

```
public Destinations getDestination(int did) {
```

```
    Transaction transaction = null;
```

```
    Destinations destinations = null;
```

```
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
```

```
        // start a transaction
```

```
        transaction = session.beginTransaction();
```

```
        // get an user object
```

```
        destinations = session.get(Destinations.class, did);
```

```
        // commit transaction
```

```
        transaction.commit();
```

```
    } catch (Exception e) {
```

```
        if (transaction != null) {
```

```
            transaction.rollback();
```

```
        }
```

```
        e.printStackTrace();
```

```
    }
```

```
    return destinations;
```

```
}
```

```
/**
```

```
 * Update Destination
```

```
 * @param did
```

```
 */
```

```
    public void updateDestination(int did, String ud) {
```

```
        Transaction transaction = null;
```

```
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
```

```
            Transaction txn = session.beginTransaction();
```

```
            // 1. fetch the object of specific employeeid - select query to be
```

executed

```
            Destinations destination = session.get(Destinations.class, did);
```

```
            destination.setDestination(ud);
```

```
            // 3. send the object back
```

```
            session.update(destination);
```

```
            txn.commit();
```

```
            //session.close();
```

```
        } catch (Exception e) {
```

```
            if (transaction != null) {
```

```
                transaction.rollback();
```

```
            }
```

```
            e.printStackTrace();
```

```
        }
```



```
    }  
    /**  
    * Add New Destination  
    * @param destination  
    */  
    public void addDestination(Destinations destination) {  
        Integer destinationId = null;  
        Session session = HibernateUtil.getSessionFactory().openSession();  
        Transaction txn = session.beginTransaction();  
        // save methods adds new row into database table  
        destinationId = (Integer) session.save(destination);  
        txn.commit();  
    }  
}  
dao.FlightsDao
```

```

package dao;

import java.util.Date;
import java.util.List;

import org.hibernate.Session;
import org.hibernate.Transaction;
import org.hibernate.query.Query;

import model.Flights;
import model.Users;
import utl.HibernateUtil;

public class FlightsDao {

    /**
     * Get all Flights
     * @return
     */
    @SuppressWarnings("unchecked")
    public List < Flights > getAllFlights() {

        Transaction transaction = null;
        List < Flights > listOfFlights = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            // start a transaction
            transaction = session.beginTransaction();
            // get an user object

            listOfFlights = session.createQuery("from Flights").getResultList();

            // commit transaction
            transaction.commit();
        } catch (Exception e) {
            if (transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        }
        return listOfFlights;
    }

    /**

```

```

* Get Available Flights
* @return
*/
@SuppressWarnings("unchecked")
public List < Flights > getFlights(String src, String des, Date dt) {

    Transaction transaction = null;
    List < Flights > listOfFlights = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();
        // get an user object
        //listOfFlights = session.createQuery("from Flights where
source=:source").getResultList();
        Query query = session.createQuery("from Flights where source=:source and
destination=:destination and date(date)=:date");
        query.setCacheable(true);
        query.setParameter("source", src);
        query.setParameter("destination", des);
        query.setParameter("date", dt);
        listOfFlights = query.list();
        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return listOfFlights;
}

/**
* Get Available Flights
* @return
*/
@SuppressWarnings("unchecked")
public List < Flights > getSelectedFlight(int fid) {

    Transaction transaction = null;
    List < Flights > listOfFlights = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();

```

```

        // get an user object
        //listOfFlights = session.createQuery("from Flights where
source=:source").getResultList();
        Query query = session.createQuery("from Flights where id=:fid");
        query.setCacheable(true);
        query.setParameter("fid", fid);
        listOfFlights = query.list();
        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return listOfFlights;
}
/**
 * Get Flight Fare By Flight ID
 * @param fid
 * @return
 */
public Flights getFlgihtFare(int fid) {

    Transaction transaction = null;
    Flights flights = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();
        // get an user object
        flights = session.get(Flights.class, fid);
        int ff = flights.getTicketprice();
        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return flights;
}
/**

```

```

* Update User
* @param user
*/
public void updateFlight(Flights flight) {
    Transaction transaction = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();
        // save the student object
        session.update(flight);
        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
}

/**
* Add New Flight
* @param flight
*/
    public void addFlight(Flights flight) {
        Integer flightId = null;
        Session session = HibernateUtil.getSessionFactory().openSession();
        Transaction txn = session.beginTransaction();
        // save methods adds new row into database table
        flightId = (Integer) session.save(flight);
        txn.commit();
    }
}

```

dao.PassengerDao

package dao;

import java.util.List;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.hibernate.query.Query;

import model.Passengers;

import utl.HibernateUtil;

/\*\*

```

* CRUD database operations
* @author Balanarenthiran
*
*/
public class PassengerDao {

    /**
     * Save Passengers
     * @param passengers
     */
    public void saveUser(Passengers passengers) {
        Transaction transaction = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            // start a transaction
            transaction = session.beginTransaction();
            // save the student object
            session.save(passengers);
            // commit transaction
            transaction.commit();
        } catch (Exception e) {
            if (transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        }
    }

    /**
     * Get Available Flights
     * @return
     */
    @SuppressWarnings("unchecked")
    public List < Passengers > getPassengerDetails(String pnr) {

        Transaction transaction = null;
        List < Passengers > listOfPassengers = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            // start a transaction
            transaction = session.beginTransaction();
            // get an user object
            //listOfFlights = session.createQuery("from Flights where
source=:source").getResultList();
            Query query = session.createQuery("from Passengers where pnr=:pnr");
            query.setCacheable(true);
            query.setParameter("pnr", pnr);
            listOfPassengers = query.list();
            // commit transaction
            transaction.commit();
        } catch (Exception e) {
            if (transaction != null) {

```

```

        transaction.rollback();
    }
    e.printStackTrace();
}
return listOfPassengers;
}
}

```

dao.UserDao

package dao;

import java.util.List;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.hibernate.query.Query;

import model.Users;

import utl.HibernateUtil;

public class UserDao {

/\*\*

\* Get Available Flights

\* @return

\*/

// @SuppressWarnings("unchecked")

public boolean getSelectedUser(String un, String up) {

Transaction transaction = null;

Users user=null;

boolean result=false;

System.out.println("I'm in user dao");

try (Session session = HibernateUtil.getSessionFactory().openSession()) {

// start a transaction

transaction = session.beginTransaction();

// get an user object

Query query=session.createQuery("from Users where name=:username and password=:password");

query.setParameter("username", un);

query.setParameter("password", up);

user=(Users)query.uniqueResult();

if(user!=null) {

System.out.println("username and password are valid");

result=true;

}else {

System.out.println("username and password are not valid");

result = false;

}

}

```

        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return result;
}

/**
 * Get all Users
 * @return
 */
@SuppressWarnings("unchecked")
public List < Users > getAllUser() {

    Transaction transaction = null;
    List < Users > listOfUser = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();
        // get an user object

        listOfUser = session.createQuery("from Users").getResultList();

        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return listOfUser;
}

/**
 * Get User By ID
 * @param id
 * @return
 */
public Users getUser(int id) {

    Transaction transaction = null;
    Users user = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {
        // start a transaction
        transaction = session.beginTransaction();

```



```

        // get an user object
        user = session.get(Users.class, id);
        // commit transaction
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
    return user;
}

/**
 * Update User
 * @param user
 */
public void updateUser(int id, String up) {
    Transaction transaction = null;
    try (Session session = HibernateUtil.getSessionFactory().openSession()) {

        Transaction txn = session.beginTransaction();
        // 1. fetch the object of specific employeeId - select query to be
executed
        Users user = session.get(Users.class, id);
        System.out.println(user);
        System.out.println("inside update method");
        System.out.println(up);
        // 2. Modify employee exp - using setter method
        user.setPassword(up);
        System.out.println(user);
        // 3. send the object back
        session.update(user);
        txn.commit();
        //session.close();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
}

/**
 * Add New User
 * @param user
 */
public void addUser(Users user) {
    Integer userId = null;

```

```

        Session session = HibernateUtil.getSessionFactory().openSession();
        Transaction txn = session.beginTransaction();
        // save methods adds new row into database table
        userId = (Integer) session.save(user);
        txn.commit();
    }
}

```

#### dto.SearchCriteria

```

package dto;

import java.util.Date;

public class SearchCriteria {

    private Date udate;
    private String usource;
    private String udestination;
    private int upersons;
    private int fid;
    private int padded;
    private String pnr;
    private int uticketprice;
    private double totalfare;

    public SearchCriteria() {

    }

    public Date getUdate() {
        return udate;
    }

    public void setUdate(Date udate) {
        this.udate = udate;
    }

    public String getUsource() {
        return usource;
    }

    public void setUsource(String usource) {
        this.usource = usource;
    }

    public String getUdestination() {
        return udestination;
    }

    public void setUdestination(String udestination) {

```

```
        this.udestination = udestination;
    }

    public int getUpersons() {
        return upersons;
    }

    public void setUpersons(int upersons) {
        this.upersons = upersons;
    }

    public int getFid() {
        return fid;
    }

    public void setFid(int fid) {
        this.fid = fid;
    }

    public int getPadded() {
        return padded;
    }

    public void setPadded(int padded) {
        this.padded = padded;
    }

    public String getPnr() {
        return pnr;
    }

    public void setPnr(String pnr) {
        this.pnr = pnr;
    }

    public int getUticketprice() {
        return uticketprice;
    }

    public void setUticketprice(int uticketprice) {
        this.uticketprice = uticketprice;
    }

    public double getTotalfare() {
        return totalfare;
    }

    public void setTotalfare(double totalfare) {
        this.totalfare = totalfare;
    }
}
```

```

@Override
public String toString() {
    return "SearchCriteria [udate=" + udate + ", usource=" + usource + ",
udestination=" + udestination
        + ", upersons=" + upersons + ", fid=" + fid + ", padded=" +
padded + ", pnr=" + pnr + ", uticketprice="
        + uticketprice + ", totalfare=" + totalfare + "]";
}
}

```

#### dto.UserCriteria

```

package dto;

public class UserCriteria {
    private int uid;
    private int fid;
    private int did;

    public UserCriteria() {

    }

    public int getUid() {
        return uid;
    }

    public void setUid(int uid) {
        this.uid = uid;
    }

    public int getFid() {
        return fid;
    }

    public void setFid(int fid) {
        this.fid = fid;
    }

    public int getDid() {
        return did;
    }

    public void setDid(int did) {
        this.did = did;
    }
}

```

```

        @Override
        public String toString() {
            return "UserCriteria [uid=" + uid + ", fid=" + fid + ", did=" + did + "]";
        }
    }
}

```

model.destinations

```

package model;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

/**
 * Destinations.java
 * This is a model class represents a destinations entity
 * @author Balanarenthiran
 */

@Entity
@Table(name="destinations")
public class Destinations {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    @Column(name="id")
    protected int id;

    @Column(name="destination")
    protected String destination;

    public Destinations() {
    }

    public Destinations(String destination) {
        super();
        this.destination = destination;
    }

    public Destinations(int id, String destination) {
        super();
        this.id = id;
        this.destination = destination;
    }
}

```

```

    }

    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }

    public String getDestination() {
        return destination;
    }

    public void setDestination(String destination) {
        this.destination = destination;
    }

    @Override
    public String toString() {
        return "Destinations [id=" + id + ", destination=" + destination + "]";
    }
}

```

model.Flights

package model;

```

import java.util.Date;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

/**
 * User.java
 * This is a model class represents a Flights entity
 * @author Balanarenthiran
 *
 */

@Entity
@Table(name="flightdetails")
public class Flights {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    @Column(name="id")
    protected int id;
}

```

```
@Column(name="flight")
protected String flight;

@Column(name="source")
protected String source;

@Column(name="destination")
protected String destination;

@Column(name="date")
protected Date date;

@Column(name="ticketprice")
protected int ticketprice;

public Flights() {
}

public Flights(String flight, String source, String destination, Date date, int ticketprice) {
    super();
    System.out.println("inside flight update");
    this.flight = flight;
    this.source = source;
    this.destination = destination;
    this.date = date;
    this.ticketprice = ticketprice;
}

public Flights(int id, String flight, String source, String destination, Date date, int
ticketprice) {
    super();
    this.id = id;
    this.flight = flight;
    this.source = source;
    this.destination = destination;
    this.date = date;
    this.ticketprice = ticketprice;
}

public int getId() {
    return id;
}

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

public String getFlight() {
    return flight;
}
```

```

    public void setFlight(String flight) {
        this.flight = flight;
    }

    public String getSource() {
        return source;
    }

    public void setSource(String source) {
        this.source = source;
    }

    public String getDestination() {
        return destination;
    }

    public void setDestination(String destination) {
        this.destination = destination;
    }

    public Date getDate() {
        return date;
    }

    public void setDate(Date date) {
        this.date = date;
    }

    public int getTicketprice() {
        return ticketprice;
    }

    public void setTicketprice(int ticketprice) {
        this.ticketprice = ticketprice;
    }
}

```

model.Passengers

package model;

```

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

```

/\*\*

\* User.java

\* This is a model class represents a User entity



```
* @author Balanarenthiran  
*  
*/
```

```
@Entity  
@Table(name="passengerdetails")  
public class Passengers {
```

```
    @Id  
    @GeneratedValue(strategy=GenerationType.IDENTITY)  
    @Column(name="id")  
    protected int id;
```

```
    @Column(name="name")  
    protected String name;
```

```
    @Column(name="email")  
    protected String email;
```

```
    @Column(name="country")  
    protected String country;
```

```
    @Column(name="fid")  
    protected int fid;
```

```
    @Column(name="sid")  
    protected String sid;
```

```
    @Column(name="pnr")  
    protected String pnr;
```

```
    public Passengers() {  
    }
```

```
    public Passengers(String name, String email, String country, int fid, String sid, String pnr) {  
        super();  
        this.name = name;  
        this.email = email;  
        this.country = country;  
        this.fid = fid;  
        this.sid = sid;  
        this.pnr = pnr;  
    }
```

```
    public Passengers(int id, String name, String email, String country, int fid, String sid, String  
pnr) {  
        super();  
        this.id = id;  
        this.name = name;
```

```
this.email = email;
this.country = country;
this.fid = fid;
this.sid = sid;
this.pnr = pnr;
}

public int getId() {
    return id;
}
public void setId(int id) {
    this.id = id;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}
public String getCountry() {
    return country;
}
public void setCountry(String country) {
    this.country = country;
}
public int getFid() {
    return fid;
}
public void setFid(int fid) {
    this.fid = fid;
}
public String getSid() {
    return sid;
}
public void setSid(String sid) {
    this.sid = sid;
}
public String getPnr() {
    return pnr;
}
public void setPnr(String pnr) {
    this.pnr = pnr;
}
}
```

## Model.Users

```
package model;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

/**
 * User.java
 * This is a model class represents a Users entity
 * @author Balanarenthiran
 *
 */

@Entity
@Table(name="users")
public class Users {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    @Column(name="id")
    protected int id;

    @Column(name="name")
    protected String name;

    @Column(name="password")
    protected String password;

    public Users() {
    }

    public Users(String name, String password) {
        super();
        this.name = name;
        this.password = password;
    }

    public Users(int id, String name, String password) {
        super();
        this.id = id;
        this.name = name;
        this.password = password;
    }

    public int getId() {
        return id;
    }
}
```

```

    }
    public void setId(int id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    }

    @Override
    public String toString() {
        return "Users [id=" + id + ", name=" + name + ", password=" + password + "]\n";
    }
}

```

#### Utl.HibernateUtil

```

package utl;

import java.util.Properties;

import org.hibernate.SessionFactory;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;
import org.hibernate.cfg.Environment;
import org.hibernate.service.ServiceRegistry;

import model.Destinations;
import model.Flights;
import model.Passengers;
import model.Users;

/**
 * Java based configuration
 * @author Balanarenthiran
 *
 */
public class HibernateUtil {
    private static SessionFactory sessionFactory;

    public static SessionFactory getSessionFactory() {
        if (sessionFactory == null) {
            try {

```

```

Configuration configuration = new Configuration();

// Hibernate settings equivalent to hibernate.cfg.xml's properties
Properties settings = new Properties();
settings.put(Environment.DRIVER, "com.mysql.jdbc.Driver");
settings.put(Environment.URL, "jdbc:mysql://localhost:3306/demo?useSSL=false");
settings.put(Environment.USER, "root");
settings.put(Environment.PASS, "root");
settings.put(Environment.DIALECT, "org.hibernate.dialect.MySQL5Dialect");

settings.put(Environment.SHOW_SQL, "true");

settings.put(Environment.CURRENT_SESSION_CONTEXT_CLASS, "thread");

//settings.put(Environment.HBM2DDL_AUTO, "create-drop");
settings.put(Environment.HBM2DDL_AUTO, "update");

configuration.setProperties(settings);
configuration.addAnnotatedClass(Flights.class);
configuration.addAnnotatedClass(Passengers.class);
configuration.addAnnotatedClass(Users.class);
configuration.addAnnotatedClass(Destinations.class);

ServiceRegistry serviceRegistry = new StandardServiceRegistryBuilder()
    .applySettings(configuration.getProperties()).build();
System.out.println("Hibernate Java Config serviceRegistry created");
SessionFactory sessionFactory = configuration.buildSessionFactory(serviceRegistry);
return sessionFactory;

} catch (Exception e) {
    e.printStackTrace();
}
}
return sessionFactory;
}
}

```

#### web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID" version="2.5">
    <display-name>FlyAway</display-name>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>

    </welcome-file-list>

    <servlet>
        <description>BookFlightServlet</description>

```

```
<display-name>BookFlightServlet</display-name>
<servlet-name>BookFlightServlet</servlet-name>
<servlet-class>com.BookFlightServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>BookFlightServlet</servlet-name>
  <url-pattern>/BookFlightServlet</url-pattern>
</servlet-mapping>

<servlet>
  <description>AvailableFlight</description>
  <display-name>AvailableFlight</display-name>
  <servlet-name>AvailableFlight</servlet-name>
  <servlet-class>com.AvailableFlight</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AvailableFlight</servlet-name>
  <url-pattern>/AvailableFlight</url-pattern>
</servlet-mapping>

<servlet>
  <description>PassengerDetail</description>
  <display-name>PassengerDetail</display-name>
  <servlet-name>PassengerDetail</servlet-name>
  <servlet-class>com.PassengerDetail</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>PassengerDetail</servlet-name>
  <url-pattern>/PassengerDetail</url-pattern>
</servlet-mapping>

<servlet>
  <description>ConfirmBooking</description>
  <display-name>ConfirmBooking</display-name>
  <servlet-name>ConfirmBooking</servlet-name>
  <servlet-class>com.ConfirmBooking</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ConfirmBooking</servlet-name>
  <url-pattern>/ConfirmBooking</url-pattern>
</servlet-mapping>

<servlet>
  <description>AdminLogin</description>
  <display-name>AdminLogin</display-name>
  <servlet-name>AdminLogin</servlet-name>
  <servlet-class>com.AdminLogin</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AdminLogin</servlet-name>
```

```
<url-pattern>/AdminLogin</url-pattern>
</servlet-mapping>

<servlet>
  <description>ValidateUser</description>
  <display-name>ValidateUser</display-name>
  <servlet-name>ValidateUser</servlet-name>
  <servlet-class>com.ValidateUser</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ValidateUser</servlet-name>
  <url-pattern>/ValidateUser</url-pattern>
</servlet-mapping>

<servlet>
  <description>ResetPassword</description>
  <display-name>ResetPassword</display-name>
  <servlet-name>ResetPassword</servlet-name>
  <servlet-class>com.ResetPassword</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ResetPassword</servlet-name>
  <url-pattern>/ResetPassword</url-pattern>
</servlet-mapping>

<servlet>
  <description>UpdateUser</description>
  <display-name>UpdateUser</display-name>
  <servlet-name>UpdateUser</servlet-name>
  <servlet-class>com.UpdateUser</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>UpdateUser</servlet-name>
  <url-pattern>/UpdateUser</url-pattern>
</servlet-mapping>

<servlet>
  <description>AddUser</description>
  <display-name>AddUser</display-name>
  <servlet-name>AddUser</servlet-name>
  <servlet-class>com.AddUser</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AddUser</servlet-name>
  <url-pattern>/AddUser</url-pattern>
</servlet-mapping>

<servlet>
  <description>EditFlight</description>
  <display-name>EditFlight</display-name>
```

```
<servlet-name>EditFlight</servlet-name>
<servlet-class>com.EditFlight</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>EditFlight</servlet-name>
  <url-pattern>/EditFlight</url-pattern>
</servlet-mapping>

<servlet>
  <description>UpdateFlight</description>
  <display-name>UpdateFlight</display-name>
  <servlet-name>UpdateFlight</servlet-name>
  <servlet-class>com.UpdateFlight</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>UpdateFlight</servlet-name>
  <url-pattern>/UpdateFlight</url-pattern>
</servlet-mapping>

<servlet>
  <description>AddFlight</description>
  <display-name>AddFlight</display-name>
  <servlet-name>AddFlight</servlet-name>
  <servlet-class>com.AddFlight</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AddFlight</servlet-name>
  <url-pattern>/AddFlight</url-pattern>
</servlet-mapping>

<servlet>
  <description>ListUsers</description>
  <display-name>ListUsers</display-name>
  <servlet-name>ListUsers</servlet-name>
  <servlet-class>com.ListUsers</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ListUsers</servlet-name>
  <url-pattern>/ListUsers</url-pattern>
</servlet-mapping>

<servlet>
  <description>ListDestinations</description>
  <display-name>ListDestinations</display-name>
  <servlet-name>ListDestinations</servlet-name>
  <servlet-class>com.ListDestinations</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ListDestinations</servlet-name>
  <url-pattern>/ListDestinations</url-pattern>
```



```
</servlet-mapping>

<servlet>
  <description>UpdateDestination</description>
  <display-name>UpdateDestination</display-name>
  <servlet-name>UpdateDestination</servlet-name>
  <servlet-class>com.UpdateDestination</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>UpdateDestination</servlet-name>
  <url-pattern>/UpdateDestination</url-pattern>
</servlet-mapping>

<servlet>
  <description>AddDestination</description>
  <display-name>AddDestination</display-name>
  <servlet-name>AddDestination</servlet-name>
  <servlet-class>com.AddDestination</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>AddDestination</servlet-name>
  <url-pattern>/AddDestination</url-pattern>
</servlet-mapping>

</web-app>
```

Lib (dependencies)

