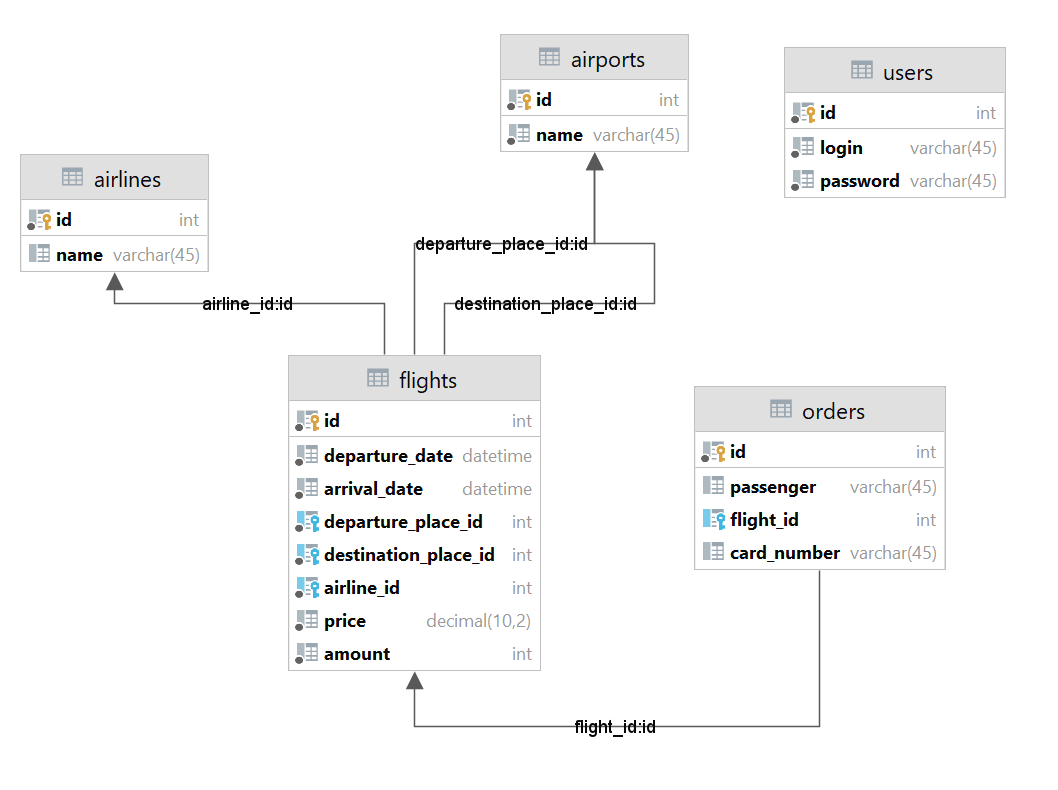
SQL Server side.

Database diagram:



View:

CREATE

ALGORITHM = UNDEFINED

DEFINER = `root`@`localhost`

SQL SECURITY DEFINER

VIEW `flytickets`.`miniflights` AS

SELECT

`flytickets`.`flights`.`price` AS `price`,

`dep`.`name` AS `source`,

`dest`.`name` AS `destination`,

`flytickets`.`airlines`.`name` AS `airline`

FROM

(((`flytickets`.`flights`

LEFT JOIN `flytickets`.`airports` `dep` ON ((`dep`.`id` = `flytickets`.`flights`.`departure\_place\_id`)))

LEFT JOIN `flytickets`.`airports` `dest` ON ((`dest`.`id` = `flytickets`.`flights`.`destination\_place\_id`)))

LEFT JOIN `flytickets`.`airlines` ON ((`flytickets`.`airlines`.`id` = `flytickets`.`flights`.`airline\_id`)))

Stored procedures:

CREATE DEFINER=`root`@`localhost` PROCEDURE `add\_order`(IN pFlightId int, IN pCard varchar(50), IN pPassenger varchar(50))

BEGIN

insert flytickets.orders(flight\_id,card\_number, passenger)

values(pFlightId, pCard, pPassenger) ;

END

CREATE DEFINER=`root`@`localhost` PROCEDURE `get\_flights`(IN p\_departure varchar(50), IN p\_destination varchar(50), p\_amount int,p\_startDate datetime, p\_endDate datetime)

select

flights.departure\_date,

flights.arrival\_date,

flights.price,

flights.amount-COALESCE (orders.am,0) as amount,

flights.id,

dep.name as departure,

dest.name as destination,

airlines.name as airline

from flights

left join airports as dep on dep.id=flights.departure\_place\_id

left join airports as dest on dest.id=flights.destination\_place\_id

left join airlines on airlines.id=flights.airline\_id

left join (SELECT

sum(1) as am,

flight\_id as id

FROM flytickets.orders

group by flight\_id) as orders

on flights.id=orders.id

where flights.amount-COALESCE (orders.am,0)>=p\_amount

and dep.name=p\_departure

and dest.name=p\_destination

and departure\_date between p\_startDate and p\_endDate

order by departure\_date

Project structure:

Изображение выглядит как текст

Автоматически созданное описание

JSP and HTML pages:

Index.html:

<html>  
<head>  
 <meta charset="UTF-8">  
 <title>FlyAway</title>  
</head>  
<body>  
<a href="login.html">Click here for login to admin tools pannel</a>  
<h1 align=center>Flight ticket booking application</h1>  
<center>  
 <form method=post action="getflightinformation">  
 <h2>To get information about available tickets specify all fields and click "Search tickets" </h2>  
 <table width="50%" border="0">  
 <tr valign="top">  
 <td width="40%">Departure</td>  
 <td><input type="text" name="departure" id="departure" maxlength=50></td>  
 </tr>  
  
 <tr valign="top">  
 <td width="40%">Destination</td>  
 <td><input type="text" name="destination" id="destination" maxlength=50></td>  
 </tr>  
  
 <tr valign="top">  
 <td width="40%">Date of flight</td>  
 <td><input type="date" name="dateofflight" id="dateofflight" maxlength=50></td>  
 </tr>  
  
 <tr valign="top">  
 <td width="40%">Number of person</td>  
 <td><input type="number" name="number" id="number" maxlength=50></td>  
 </tr>  
  
 <tr valign="top">  
 <td colspan=2 width="100%">  
 <button>Search tickets</button>  
 </td>  
 </tr>  
  
 </table>  
 </form>  
</center>  
</body>  
</html>

login.html:

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>FlyAway</title>  
</head>  
<body>  
<div align="left">  
 <h2> Login to admin page</h2>  
 <form name=Login method="post" action="login">  
 <table width="50%" border="0">  
 <tr valign="top">  
 <td width="40%">Enter UserName</td>  
 <td><input type="text" name="username" id="username" maxlength=30></td>  
 </tr>  
 <tr valign="top">  
 <td width="40%">Enter Password</td>  
 <td><input type="password" name="password" id="password" maxlength=30></td>  
 </tr>  
 <tr valign="top">  
 <td colspan=2 width="100%">  
 <button>Submit</button>  
 </td>  
  
 </tr>  
 </table>  
 </form>  
</div>  
</body>  
</html>

searchResult.jsp:

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>  
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>  
<!DOCTYPE html>  
<html>  
<head>  
 <title>FlyAway</title>  
</head>  
<body>  
<h1> List of compatible flights</h1>  
  
<c:forEach var="res" items="${sRes}" >  
 <form method=post action="bookFlight.jsp">  
 Departure from  
 ${res.departure} at  
 ${res.departureDate} ------> arrival to  
 ${res.destination} at  
 ${res.arrivalDate} operated by  
 ${res.airline} availaible  
 ${res.amount} tickets  
  
 <input type="hidden" name="flightID" value=${res.id}>  
 <input type="hidden" name="amount" value=${res.amount}>  
 <input type="hidden" name="departure" value=${res.departure}>  
 <input type="hidden" name="destination" value=${res.destination}>  
 <input type="hidden" name="departureDate" value=${res.departureDate}>  
 <input type="hidden" name="arrivalDate" value=${res.arrivalDate}>  
 <input type="hidden" name="airline" value=${res.airline}>  
 <input type="hidden" name="needed" value=${paramValues.number[0]}>  
  
 <button type="submit">Book</button>  
 </form> <br>  
</c:forEach>  
  
</body>  
</html>

error.html:

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>FlyAway</title>  
</head>  
<body>  
<h1>  
 <left>  
 <span style='color:red'> Invalid credentials!!</span>  
 </left>  
</h1>  
</body>  
</html>

bookFlight.jsp:

*<%--  
 Created by IntelliJ IDEA.  
 User: Александр  
 Date: 6.4.22  
 Time: 17:57  
 To change this template use File | Settings | File Templates.  
--%>*<%@ page contentType="text/html;charset=UTF-8" language="java" %>  
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>  
<html>  
<head>  
 <title>FlyAway</title>  
</head>  
<body>  
<h1> Booking page</h1>  
<h2> Ticket(s) from  
 ${paramValues.departure[0]} at  
 ${paramValues.departureDate[0]} ------> arrival to  
 ${paramValues.destination[0]} at  
 ${paramValues.arrivalDate[0]} operated by  
 ${paramValues.airline[0]}  
</h2>  
<form method=post action="paypage">  
 <c:forEach var="i" begin="1" end="${paramValues.needed[0]}" step="1" >  
 <h3>Enter Passenger${i} name </h3>  
 <input type="text" name="passenger" id="passenger"> <br>  
 </c:forEach>  
 <h3>Enter card number </h3>  
 <input type="text" name="cardnumber" id="cardnumber"> <br><br>  
 <input type="hidden" name="flightid" value=${paramValues.flightID[0]}>  
 <button type="submit">Book flight !</button>  
</form>  
</body>  
</html>

adminDashboard.jsp:

*<%--  
 Created by IntelliJ IDEA.  
 User: Александр  
 Date: 8.4.22  
 Time: 14:57  
 To change this template use File | Settings | File Templates.  
--%>*<%@ page contentType="text/html;charset=UTF-8" language="java" %>  
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>  
  
<html>  
<head>  
 <title>FlyAway</title>  
</head>  
<body>  
<a href="logout">Click here to logout from admin page</a><br><br>  
  
<div align="left">  
 <h2> You can change your password</h2>  
 <form name=Login method="post" action="changepassword">  
 <table width="50%" border="0">  
 <tr valign="top">  
 <td width="40%">Enter old password</td>  
 <td><input type="password" name="oldpassword" id="oldpassword" maxlength=30></td>  
 </tr>  
 <tr valign="top">  
 <td width="40%">Enter new password</td>  
 <td><input type="password" name="newpassword" id="password" maxlength=30></td>  
 </tr>  
 <tr valign="top">  
 <td colspan=2 width="100%">  
 <button>Change password</button>  
 </td>  
 </tr>  
 </table>  
 </form>  
</div>  
  
<br>  
  
<h3> List of airlines: </h3>  
<c:forEach var="res" items="${airlines}" >  
 ${res.name} <br>  
</c:forEach>  
<br>  
  
<h3> List of places: </h3>  
<c:forEach var="res" items="${airports}" >  
 ${res.name}<br>  
</c:forEach>  
  
<h3> List of flights: </h3>  
<table width="50%" border="0">  
 <tr valign="top">  
 <td width="40%"><b>Source:</b></td>  
 <td width="40%"><b>Destination:</b></td>  
 <td width="40%"><b>Airline:</b></td>  
 <td width="40%"><b>Price:</b></td>  
 </tr>  
 <c:forEach var="res" items="${miniflights}" >  
 <tr valign="left">  
 <td width="40%">${res.source}</td>  
 <td width="40%">${res.destination}</td>  
 <td width="40%">${res.airline}</td>  
 <td width="40%">${res.price}</td>  
 </tr>  
 </c:forEach>  
</table>  
  
</body>  
</html>

Pom.xml:

*<?*xml version="1.0" encoding="UTF-8"*?>*<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>com.example</groupId>  
 <artifactId>final01</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <name>final01</name>  
 <packaging>war</packaging>  
  
 <properties>  
 <maven.compiler.target>1.8</maven.compiler.target>  
 <maven.compiler.source>1.8</maven.compiler.source>  
 <junit.version>5.7.1</junit.version>  
 </properties>  
  
 <dependencies>  
 <dependency>  
 <groupId>mysql</groupId>  
 <artifactId>mysql-connector-java</artifactId>  
 <version>8.0.27</version>  
 </dependency>  
 <dependency>  
 <groupId>javax.servlet</groupId>  
 <artifactId>javax.servlet-api</artifactId>  
 <version>4.0.1</version>  
 <scope>provided</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-api</artifactId>  
 <version>${junit.version}</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.junit.jupiter</groupId>  
 <artifactId>junit-jupiter-engine</artifactId>  
 <version>${junit.version}</version>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>javax.servlet.jsp.jstl</groupId>  
 <artifactId>jstl-api</artifactId>  
 <version>1.2</version>  
 </dependency>  
 <dependency>  
 <groupId>taglibs</groupId>  
 <artifactId>standard</artifactId>  
 <version>1.1.2</version>  
 </dependency>  
 <dependency>  
 <groupId>org.projectlombok</groupId>  
 <artifactId>lombok</artifactId>  
 <version>RELEASE</version>  
 <scope>compile</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-war-plugin</artifactId>  
 <version>3.3.1</version>  
 </plugin>  
 </plugins>  
 </build>  
</project>

Web.xml:

*<?*xml version="1.0" encoding="UTF-8"*?>*<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_4\_0.xsd"  
 version="4.0">  
 <display-name>LoginLogout</display-name>  
  
 <welcome-file-list>  
 <welcome-file>index.html</welcome-file>  
 </welcome-file-list>  
  
 <servlet>  
 <servlet-name>GetFlightInformation</servlet-name>  
 <servlet-class> com.example.final01.GetFlightInformation</servlet-class>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>GetFlightInformation</servlet-name>  
 <url-pattern>/getflightinformation</url-pattern>  
 </servlet-mapping>  
 <servlet>  
 <servlet-name>LogoutServlet</servlet-name>  
 <servlet-class> com.example.final01.LogoutServlet</servlet-class>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>LogoutServlet</servlet-name>  
 <url-pattern>/logout</url-pattern>  
 </servlet-mapping>  
 <servlet>  
 <servlet-name>LoginServlet</servlet-name>  
 <servlet-class> com.example.final01.LoginServlet</servlet-class>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>LoginServlet</servlet-name>  
 <url-pattern>/login</url-pattern>  
 </servlet-mapping>  
  
 <servlet>  
 <servlet-name>ChangePassword</servlet-name>  
 <servlet-class>com.example.final01.ChangePassword</servlet-class>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>ChangePassword</servlet-name>  
 <url-pattern>/changepassword</url-pattern>  
 </servlet-mapping>  
  
 <servlet>  
 <servlet-name>PayPage</servlet-name>  
 <servlet-class> com.example.final01.PayPage</servlet-class>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>PayPage</servlet-name>  
 <url-pattern>/paypage</url-pattern>  
 </servlet-mapping>  
</web-app>

Servlets.

ChangePassword:

package com.example.final01;  
  
import com.example.final01.util.DBConnection;  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.PrintWriter;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.util.Properties;  
  
@WebServlet(name = "ChangePassword", value = "/ChangePassword")  
public class ChangePassword extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
  
 try {  
  
 HttpSession httpSession= request.getSession(false);  
 String login = (String) httpSession.getAttribute("username");  
 String oldpassword = request.getParameter("oldpassword");  
 String newpassword = request.getParameter("newpassword");  
  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
  
 PreparedStatement preparedStatementSelect = conn.getConnection().prepareStatement("select id from users where login = ? and password = ? ");  
  
 preparedStatementSelect.setString(1, login);  
 preparedStatementSelect.setString(2, oldpassword);  
  
 ResultSet rs = preparedStatementSelect.executeQuery();  
  
 if (rs.next()) {  
 PreparedStatement preparedStatementUpdate = conn.getConnection().prepareStatement("update users set password=? where id = ?");  
 preparedStatementUpdate.setString(1, newpassword);  
 preparedStatementUpdate.setInt(2,rs.getInt("id"));  
 preparedStatementUpdate.executeUpdate();  
  
 preparedStatementSelect.close();  
 preparedStatementUpdate.close();  
 conn.closeConnection();  
  
 PrintWriter out=response.getWriter();  
 out.println("<html><body><h1>");  
 out.println("Password changed Successful");  
 out.println("</h1></body></html>");  
  
 } else {  
 preparedStatementSelect.close();  
 conn.closeConnection();  
 RequestDispatcher requestDispatcher = request.getRequestDispatcher("error.html");  
 requestDispatcher.forward(request, response);  
 }  
  
  
  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
  
  
 }  
}

GetFlightInformation:

package com.example.final01;  
  
import com.example.final01.model.Flight;  
import com.example.final01.util.DBConnection;  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
import java.io.IOException;  
import java.io.InputStream;  
import java.sql.CallableStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.time.LocalDate;  
import java.time.LocalDateTime;  
import java.time.LocalTime;  
import java.util.ArrayList;  
import java.util.Properties;  
  
@WebServlet(name = "GetFlightInformation", value = "/GetFlightInformation")  
public class GetFlightInformation extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 try {  
  
 Integer numberOfPerson = Integer.*parseInt*(request.getParameter("number"));  
 String placeOfDeparture=request.getParameter("departure");  
 String placeOfDestination=request.getParameter("destination");  
  
 String dateOfFlight=request.getParameter("dateofflight");  
 LocalDate localDate=LocalDate.*parse*(dateOfFlight);  
 LocalDateTime startOfDay = LocalDateTime.*of*(localDate, LocalTime.*MIN*);  
 LocalDateTime endOfDay = LocalDateTime.*of*(localDate, LocalTime.*MAX*);  
  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
  
  
 CallableStatement stmtCallable = conn.getConnection().prepareCall("{call get\_flights(?,?,?,?,?)}");  
 stmtCallable.setString(1, placeOfDeparture);  
 stmtCallable.setString(2, placeOfDestination);  
 stmtCallable.setInt(3, numberOfPerson);  
 stmtCallable.setTimestamp(4,java.sql.Timestamp.*valueOf*(startOfDay));  
 stmtCallable.setTimestamp(5, java.sql.Timestamp.*valueOf*(endOfDay));  
  
 ResultSet rst=stmtCallable.executeQuery();  
  
 ArrayList<Flight> flightArrayList=Flight.*getFlightsList*(rst);  
  
 stmtCallable.close();  
 conn.closeConnection();  
  
 request.setAttribute("sRes",flightArrayList );  
  
  
 RequestDispatcher view =  
 request.getRequestDispatcher("searchResult.jsp");  
 view.forward(request, response);  
  
  
  
  
  
  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
 } catch (NumberFormatException e) {  
 e.printStackTrace();  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
 }  
}

LoginServlet:

package com.example.final01;  
  
import com.example.final01.model.Airline;  
import com.example.final01.model.Airport;  
import com.example.final01.model.MiniFlight;  
import com.example.final01.util.DBConnection;  
  
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 java.io.IOException;  
import java.io.InputStream;  
import java.sql.\*;  
import java.util.Properties;  
  
@WebServlet(name = "LoginServlet", value = "/LoginServlet")  
public class LoginServlet extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 doPost(request, response);  
  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 try {  
  
 String login = request.getParameter("username");  
 String password = request.getParameter("password");  
  
  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
  
 PreparedStatement preparedStatement = conn.getConnection().prepareStatement("select \* from users where login = ? and password = ? ");  
  
 preparedStatement.setString(1, login);  
 preparedStatement.setString(2, password);  
  
 ResultSet rs = preparedStatement.executeQuery();  
  
  
 Boolean success=rs.next();  
  
 if (success) {  
 HttpSession session = request.getSession();  
 String userName = request.getParameter("username");  
 session.setAttribute("username", userName);  
  
 Statement stmt = conn.getConnection().createStatement(ResultSet.*TYPE\_SCROLL\_INSENSITIVE*, ResultSet.*CONCUR\_READ\_ONLY*);  
 ResultSet rst = stmt.executeQuery("select name from airlines");  
 request.setAttribute("airlines",Airline.*getAirlinesList*(rst));  
 rst = stmt.executeQuery("select name from airports");  
 request.setAttribute("airports", Airport.*getAirportsList*(rst));  
 rst = stmt.executeQuery("select \* from miniflights");  
 request.setAttribute("miniflights", MiniFlight.*getMiniFlightsList*(rst));  
  
  
 RequestDispatcher requestDispatcher = request.getRequestDispatcher("adminDashboard.jsp");  
 requestDispatcher.forward(request, response);  
 } else {  
 RequestDispatcher requestDispatcher = request.getRequestDispatcher("error.html");  
 requestDispatcher.forward(request, response);  
 }  
  
  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
  
  
  
 }  
  
}

LogoutServlet:

package com.example.final01;  
  
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 java.io.IOException;  
import java.io.PrintWriter;  
  
@WebServlet(name = "LogoutServlet", value = "/LogoutServlet")  
public class LogoutServlet extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 PrintWriter out=response.getWriter();  
 HttpSession httpSession= request.getSession(false);  
  
  
 out.println("<html><body>");  
 out.println("<h2>Bye "+httpSession.getAttribute("username")+" </h2>");  
 out.println("</body></html>");  
  
  
 httpSession.invalidate();  
 }  
}

PayPage:

package com.example.final01;  
  
import com.example.final01.util.DBConnection;  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
import javax.servlet.annotation.\*;  
import java.io.IOException;  
import java.io.InputStream;  
import java.io.PrintWriter;  
import java.sql.CallableStatement;  
import java.sql.SQLException;  
import java.util.Properties;  
  
@WebServlet(name = "PayPage", value = "/PayPage")  
public class PayPage extends HttpServlet {  
 @Override  
 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
  
 }  
  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
  
 try {  
  
 InputStream in = getServletContext().getResourceAsStream("/WEB-INF/config.properties");  
 Properties props = new Properties();  
 props.load(in);  
  
 DBConnection conn = new DBConnection(props.getProperty("url"), props.getProperty("userid"), props.getProperty("password"));  
  
  
 CallableStatement stmtCallable = conn.getConnection().prepareCall("{call add\_order(?,?,?)}");  
  
 Integer flightID = Integer.*parseInt*(request.getParameter("flightid"));  
 String cardNumber=request.getParameter("cardnumber");  
 String[] passengers=request.getParameterValues("passenger");  
  
 for (String passenger : passengers) {  
 stmtCallable.setInt(1, flightID);  
 stmtCallable.setString(2, cardNumber);  
 stmtCallable.setString(3, passenger);  
 stmtCallable.executeUpdate();  
 }  
  
 stmtCallable.close();  
 conn.closeConnection();  
  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
 } catch (NumberFormatException e) {  
 e.printStackTrace();  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }  
  
 PrintWriter out=response.getWriter();  
 out.println("<html><body>");  
 out.println("<h1>Congratulations! Your ticket(s) paid successful<h1>");  
 out.println("</html></body>");  
 }  
}

Util classes.

DBConnection:

package com.example.final01.util;  
  
  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;  
  
public class DBConnection {  
  
 private Connection connection;  
  
 public DBConnection(String dbURL, String user, String pwd) throws ClassNotFoundException, SQLException{  
  
 Class.*forName*("com.mysql.jdbc.Driver");  
 this.connection = DriverManager.*getConnection*(dbURL, user, pwd);  
 }  
  
 public Connection getConnection(){  
 return this.connection;  
 }  
  
 public void closeConnection() throws SQLException {  
 if (this.connection != null)  
 this.connection.close();  
 }  
}

POJO classes.

Airline:

package com.example.final01.model;  
  
import lombok.Getter;  
import lombok.Setter;  
  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.util.ArrayList;  
  
@Getter  
@Setter  
public class Airline {  
 String name;  
  
 public static ArrayList<Airline> getAirlinesList(ResultSet rst) throws SQLException {  
  
 ArrayList<Airline> airlines=new ArrayList<>();  
  
 while (rst.next()){  
 Airline airline=new Airline();  
 airline.name=rst.getString("name");  
 airlines.add(airline);  
  
 }  
 return airlines;  
 }  
  
}

Airport:

package com.example.final01.model;  
  
  
import lombok.Getter;  
import lombok.Setter;  
  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.util.ArrayList;  
  
@Getter  
@Setter  
public class Airport {  
 String name;  
  
 public static ArrayList<Airport> getAirportsList(ResultSet rst) throws SQLException {  
  
 ArrayList<Airport> airports=new ArrayList<>();  
  
 while (rst.next()){  
 Airport airport=new Airport();  
 airport.name=rst.getString("name");  
 airports.add(airport);  
  
 }  
 return airports;  
 }  
}

Flight:

package com.example.final01.model;  
  
import lombok.Getter;  
import lombok.Setter;  
  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Timestamp;  
import java.time.LocalDateTime;  
import java.util.ArrayList;  
import java.util.List;  
  
@Getter  
@Setter  
public class Flight {  
 private String departure;  
 private String destination;  
 private String airline;  
 private int amount;  
 private Timestamp departureDate;  
 private Timestamp arrivalDate;  
 private int id;  
  
 public static ArrayList<Flight>getFlightsList(ResultSet rst) throws SQLException {  
  
 ArrayList<Flight> flightArrayList=new ArrayList<>();  
  
 while (rst.next()){  
  
 Flight flight=new Flight();  
  
 flight.amount=rst.getInt("amount");  
 flight.id=rst.getInt("id");  
 flight.departure=rst.getString("departure");  
 flight.destination=rst.getString("destination");  
 flight.airline=rst.getString("airline");  
  
 flight.departureDate=rst.getTimestamp("departure\_date");  
 flight.arrivalDate=rst.getTimestamp("arrival\_date");  
  
 flightArrayList.add(flight);  
  
 }  
 return flightArrayList;  
 }  
  
}

MiniFlight:

package com.example.final01.model;  
  
import lombok.Getter;  
import lombok.Setter;  
  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.util.ArrayList;  
  
@Getter  
@Setter  
public class MiniFlight {  
 String source;  
 String destination;  
 String price;  
 String airline;  
  
 public static ArrayList<MiniFlight> getMiniFlightsList(ResultSet rst) throws SQLException {  
  
 ArrayList<MiniFlight> miniFlights=new ArrayList<>();  
  
 while (rst.next()){  
 MiniFlight miniFlight=new MiniFlight();  
 miniFlight.source=rst.getString("source");  
 miniFlight.destination=rst.getString("destination");  
 miniFlight.airline=rst.getString("airline");  
 miniFlight.price=rst.getString("price");  
  
 miniFlights.add(miniFlight);  
  
 }  
 return miniFlights;  
 }  
}