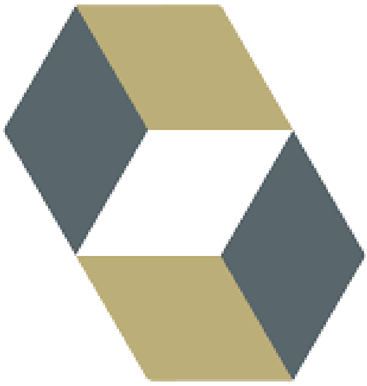
JAVA FRAMEWORKS

[PROJECT]



RAILWAY RESERVATION SYSTEM

SUBMITTED BY

8210 Prasad Belote

Mahatma Education Society's

Pillai College of Arts, Commerce & Science

(Autonomous)
Affiliated to University of Mumbai

NAAC Accredited 'A' grade (3 cycles) Best
College Award by University of Mumbai
ISO 9001:2015 Certified

CERTIFICATE

This is to certify that Mr. ***Prasad Uttam Belote*** *of* ***T.Y B.Sc. C.S. Semester V*** *has completed the project work in the Subject of* ***JAVA FRAMEWORKS*** *during the academic year 2022-23 under the guidance of Prof.* ***Manasi Khedekar Mam*** *being the partial requirement for the fulfillment of the curriculum of* ***Degree of Bachelor of Science in Computer Science, University of Mumbai****.*

Place:

Date:

Name & Signature of faculty

Name & Signature of external

Name & Signature of Co-ordinator

Abstract

India is the 7th largest country in terms of geographical size.This means there is a need for efficient means for long distance transportation. The long distance road network is very poorly developed in most parts of India. Bulk of long distance traffic is carried bythe Indian Railway as a result Indian railways. Therefore forms a backbone of public transport in India. For the reservation of the ticket a person go to ticket counter of the railway reservation office and expend its valuable time in standing queue. Now to save that time we have a facility of Online Reservation now we can book cancel or search other train information just by click on computer. This project contains introduction to the railways reservation system. It is the computerized system of reserving the seats of train seats in advance. Online reservation has made the process for the reservation of seats very much easier than ever before.

This project is basically concern with online reservation of Railway tickets to Passengers. In this project , user need to signup and create their account then after that user login to their account through their name and password .Then user can book their ticket according to their choice. After user can book another ticket or edit the previous ticket and can also delete the ticket.

Introduction

The “Railway reservation system” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. The application is reduced has much as possible to avoid errors while entering the data. It also provides error messages while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user friendly. Railway reservation system, has described, can lead to error free, and secure, reliable and fast managing system. It can assist the user to concentrate on the other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

The aim of the proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work. This system is basically concern with online reservation of Railway tickets to Passengers. In this Project we try to show how areservation is done. Store and retrieve information about various transactions related to rail travel. Keep track of all its passengers and thus scheduled their journey accordingly.Maintains records of passengers travelling in the different trains on different dates reaching different destinations.

Software Specification

Development Environments:

Hardware Intel core 3 duo T6400 2.00 GHz with 4GB RAM, 1 TB hard disk space and other Standard accessories.

Environment and Applications:

- Microsoft Windows 10.
- Eclipse IDE
- MYSQL
- Hibernate Framework
- Google Chrome / Firefox

Operating environment:

Hardware configuration:

The minimum configuration for hardware is given below:

- Intel® Pentium® or higher processor.
- 65 MB RAM or higher

Software configuration:

- Microsoft® Windows® XP or later versions
- A standard web browser.
- Hibernate Framework

Project overview

Modules:

User : User is the main character of the entired proposed system which can perform following tasks

- Signup
- Login(with validation)
- Book Ticket
- Update Ticket
- Delete Ticket

Bookticket:

- Tickets can be book by passenger .
- It have source , destination, date and class.
- It can be update , edit and delete by passeneger.

Database:

Mysql : Database is named with RRS_hibernate which contain 3 tables .

- User : User data like name , surname, email and password is situated in user table .
- Bookticket : Bookticket having users data of source , destination, return and class
- Station : Station having Station name , and fare

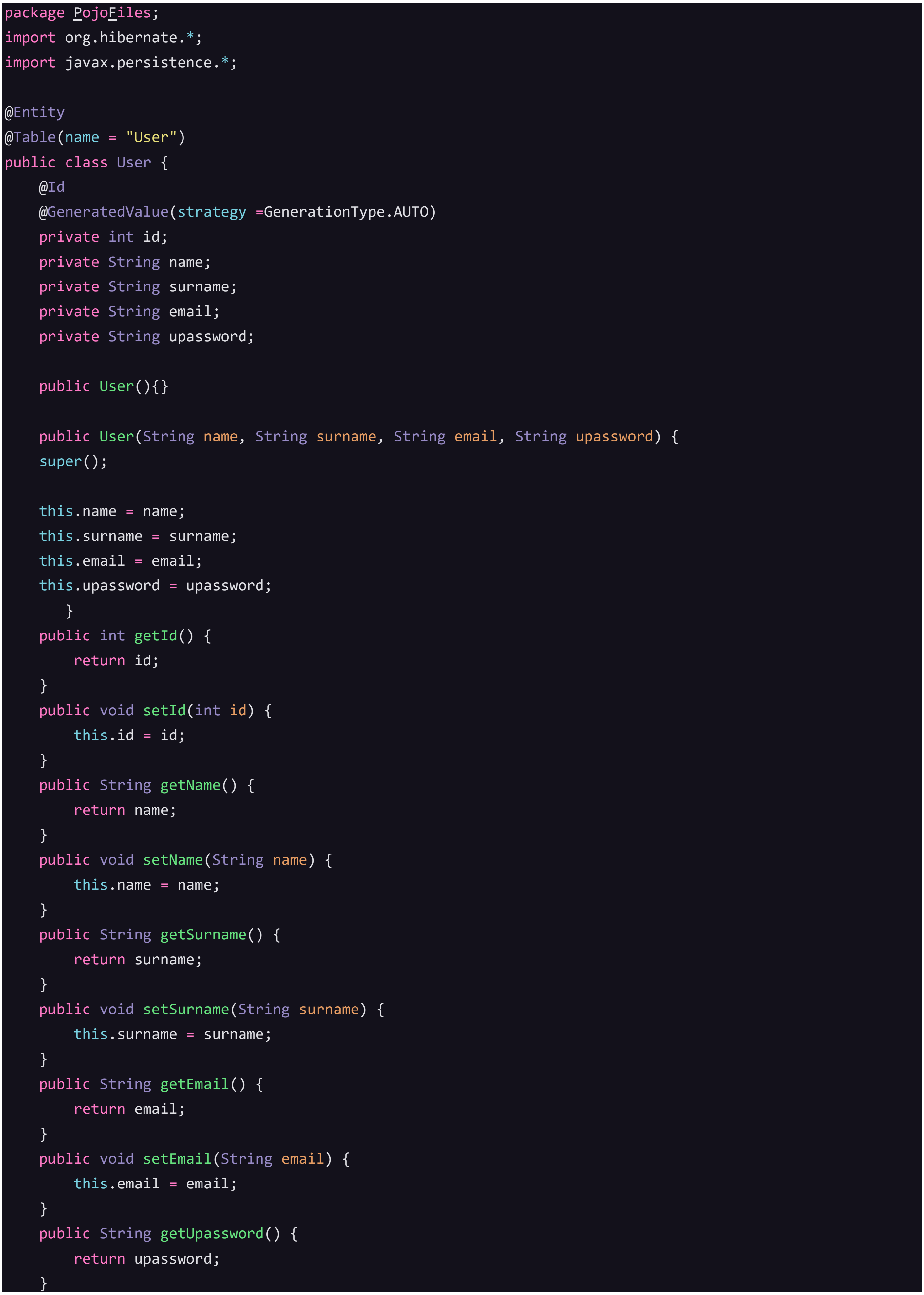
pom.xml

```xml
<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>Railway-mangment-Prasad</groupId>
  <artifactId>Railway-mangment-Prasad</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
   <build>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
        <configuration>
          <release>17</release>
        </configuration>
      </plugin>
      <plugin>
        <artifactId>maven-war-plugin</artifactId>
        <version>3.2.3</version>
      </plugin>
    </plugins>

  </build>

  <dependencies>
  <dependency>
    <groupId>org.json</groupId>
    <artifactId>json</artifactId>
    <version>20180130</version>
</dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>

     <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>javax.servlet-api</artifactId>
        <version>3.1.0</version>
        <scope>provided</scope>
    </dependency>
    <dependency>
        <groupId>javax.servlet.jsp</groupId>
        <artifactId>javax.servlet.jsp-api</artifactId>
        <version>2.3.1</version>
        <scope>provided</scope>
    </dependency>
        <!-- https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.5.4.Final</version>
</dependency>
<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.29</version>
</dependency>
<!-- https://mvnrepository.com/artifact/com.mashape.unirest/unirest-java -->
<dependency>
    <groupId>com.mashape.unirest</groupId>
    <artifactId>unirest-java</artifactId>
    <version>1.4.9</version>
</dependency>

  </dependencies>
</project>
```

hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
<session-factory>
<property name="connection.driver_class">com.mysql.jdbc.Driver</property>
<property name="connection.url">jdbc:mysql://localhost:3306/RSS-hib</property>
<property name="connection.username">root</property>
<property name="connection.password"></property>
<property name="show_sql">true</property>
<property name="format_sql">true</property>
<property name="dialect">org.hibernate.dialect.MySQL5Dialect</property>
<property name="hbm2ddl.auto">update</property>

<mapping class="PojoFiles.User"/>
<mapping class="PojoFiles.Ticket"/>


</session-factory>
</hibernate-configuration>
```

User.java



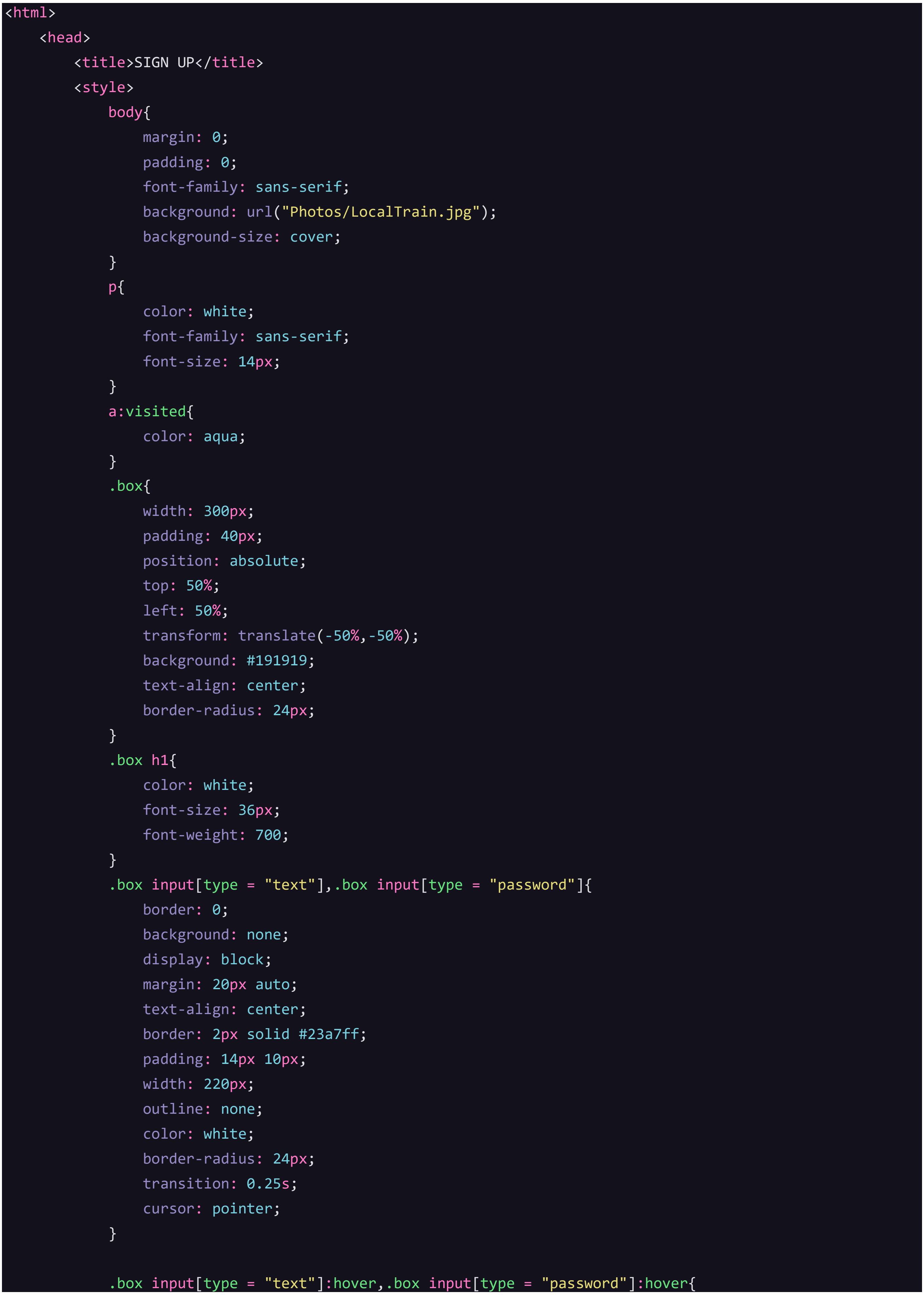
```java
package PojoFiles;
import org.hibernate.*;
import javax.persistence.*;

@Entity
@Table(name = "User")
public class User {
    @Id
    @GeneratedValue(strategy =GenerationType.AUTO)
    private int id;
    private String name;
    private String surname;
    private String email;
    private String upassword;

    public User(){}

    public User(String name, String surname, String email, String upassword) {
    super();

    this.name = name;
    this.surname = surname;
    this.email = email;
    this.upassword = upassword;
       }
    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 getSurname() {
        return surname;
    }
    public void setSurname(String surname) {
        this.surname = surname;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getUpassword() {
        return upassword;
    }
    public void setUpassword(String upassword) {
        this.upassword = upassword;
    }
   }
```

Ticket.java

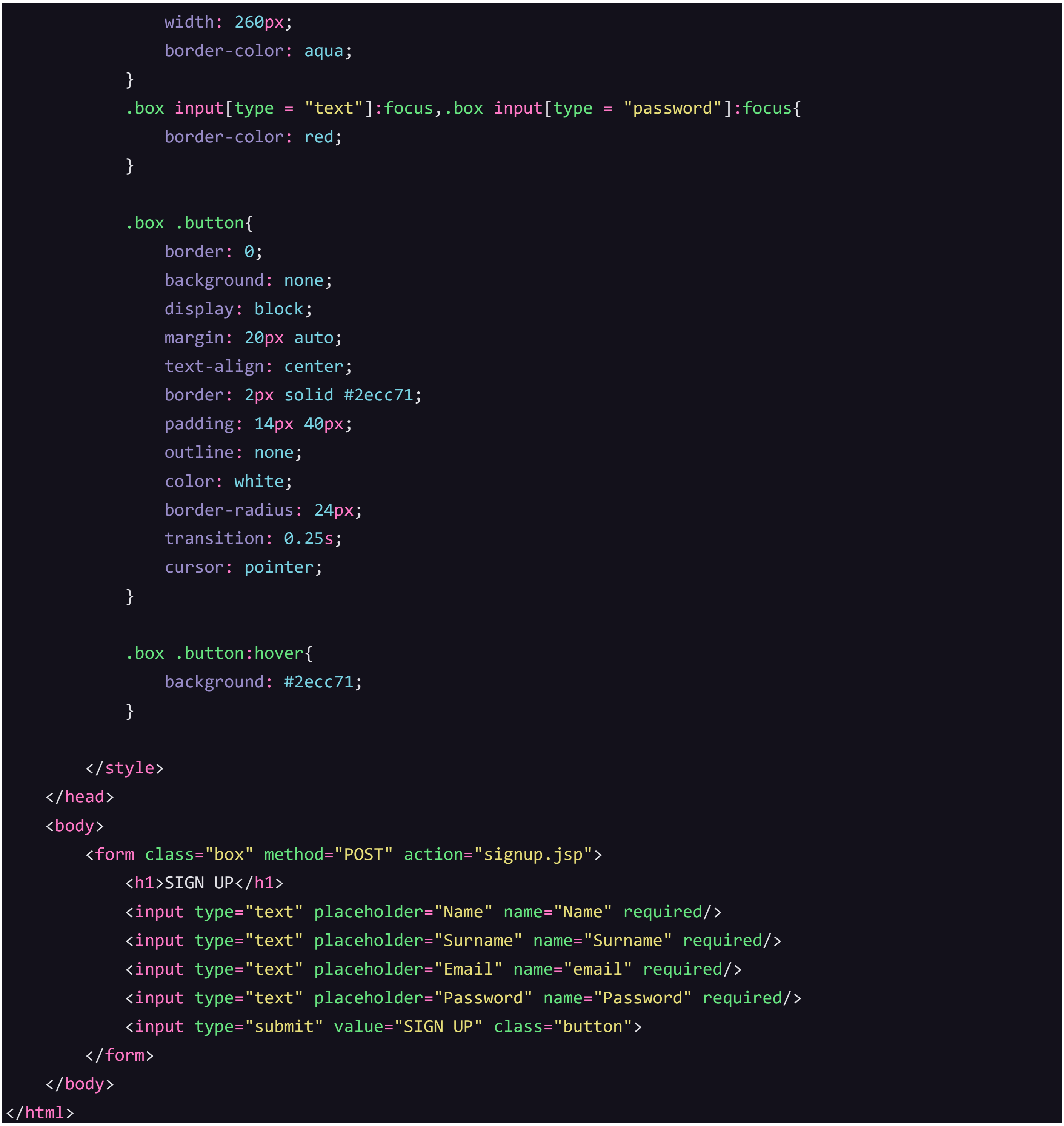
```
package PojoFiles;
import javax.persistence.JoinColumn;
import javax.persistence.*;

@Entity
@Table(name = "bookedTicket")
public class Ticket {
    @Id
        @GeneratedValue(strategy = GenerationType.AUTO)
        private int id;

        @ManyToOne
    @JoinColumn(name = "user_id")
    private User user;

        private String source;
        private String destination;
        private String classs;
        private String Date;
        @Column(name = "_return")
        private String Return;

        public Ticket() {
        super();
        // TODO Auto-generated constructor stub
    }

    public Ticket(User user, String source, String destination, String classs, String date, String
return1) {
        super();
        this.user = user;
        this.source = source;
        this.destination = destination;
        this.classs = classs;
        Date = date;
        Return = return1;
    }

    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public User getUser() {
        return user;
```

```java
    }
    public void setUser(User user) {
        this.user = user;
    }
    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 String getClasss() {
        return classs;
    }
    public void setClasss(String classs) {
        this.classs = classs;
    }
    public String getDate() {
        return Date;
    }
    public void setDate(String date) {
        Date = date;
    }
    public String getReturn() {
        return Return;
    }
    public void setReturn(String return1) {
        Return = return1;
    }

}
```

Signup.html



```
<html>
    <head>
        <title>SIGN UP</title>
        <style>
            body{
                margin: 0;
                padding: 0;
                font-family: sans-serif;
                background: url("Photos/LocalTrain.jpg");
                background-size: cover;
            }
            p{
                color: white;
                font-family: sans-serif;
                font-size: 14px;
            }
            a:visited{
                color: aqua;
            }
            .box{
                width: 300px;
                padding: 40px;
                position: absolute;
                top: 50%;
                left: 50%;
                transform: translate(-50%,-50%);
                background: #191919;
                text-align: center;
                border-radius: 24px;
            }
            .box h1{
                color: white;
                font-size: 36px;
                font-weight: 700;
            }
            .box input[type = "text"],.box input[type = "password"]{
                border: 0;
                background: none;
                display: block;
                margin: 20px auto;
                text-align: center;
                border: 2px solid #23a7ff;
                padding: 14px 10px;
                width: 220px;
                outline: none;
                color: white;
                border-radius: 24px;
                transition: 0.25s;
                cursor: pointer;
            }

            .box input[type = "text"]:hover,.box input[type = "password"]:hover{
```

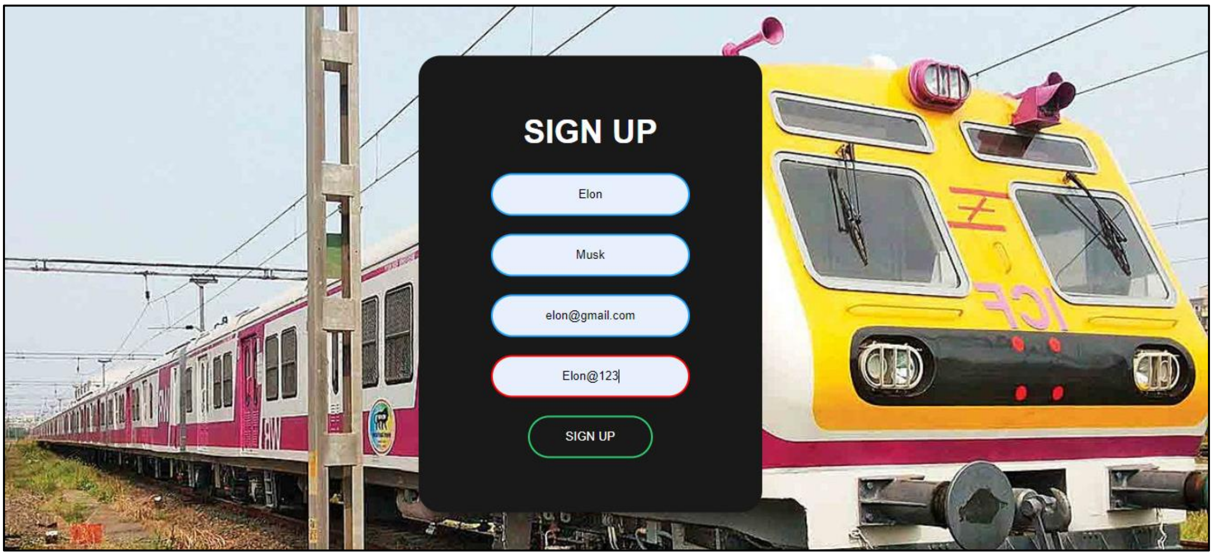


```
                width: 260px;
                border-color: aqua;
            }
            .box input[type = "text"]:focus,.box input[type = "password"]:focus{
                border-color: red;
            }

            .box .button{
                border: 0;
                background: none;
                display: block;
                margin: 20px auto;
                text-align: center;
                border: 2px solid #2ecc71;
                padding: 14px 40px;
                outline: none;
                color: white;
                border-radius: 24px;
                transition: 0.25s;
                cursor: pointer;
            }

            .box .button:hover{
                background: #2ecc71;
            }

        </style>
    </head>
    <body>
        <form class="box" method="POST" action="signup.jsp">
            <h1>SIGN UP</h1>
            <input type="text" placeholder="Name" name="Name" required/>
            <input type="text" placeholder="Surname" name="Surname" required/>
            <input type="text" placeholder="Email" name="email" required/>
            <input type="text" placeholder="Password" name="Password" required/>
            <input type="submit" value="SIGN UP" class="button">
        </form>
    </body>
</html>
```

Signup.jsp

```
<%--
    Document   : login
    Created on : 14 Aug, 2022, 12:47:20 PM
    Author     : dell
--%>
<%@page
    import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%
String name = request.getParameter("Name");
String sirname = request.getParameter("Surname");
String email = request.getParameter("email");
String pass = request.getParameter("Password");

try {
    SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
    Session session2 = factory.openSession();
    Transaction t = session2.beginTransaction();
    User u1 = new User(name, email, email, pass);
    session2.save(u1);
    t.commit();
    session2.close();

    out.print("Signup Successfull success!!");
    response.sendRedirect("login.html");

} catch (Exception e) {
    out.print("Something went wrong!");
    e.printStackTrace();
}
%>
```

Output:



Elon

Musk

elon@gmail.com

Elon@123

login.html

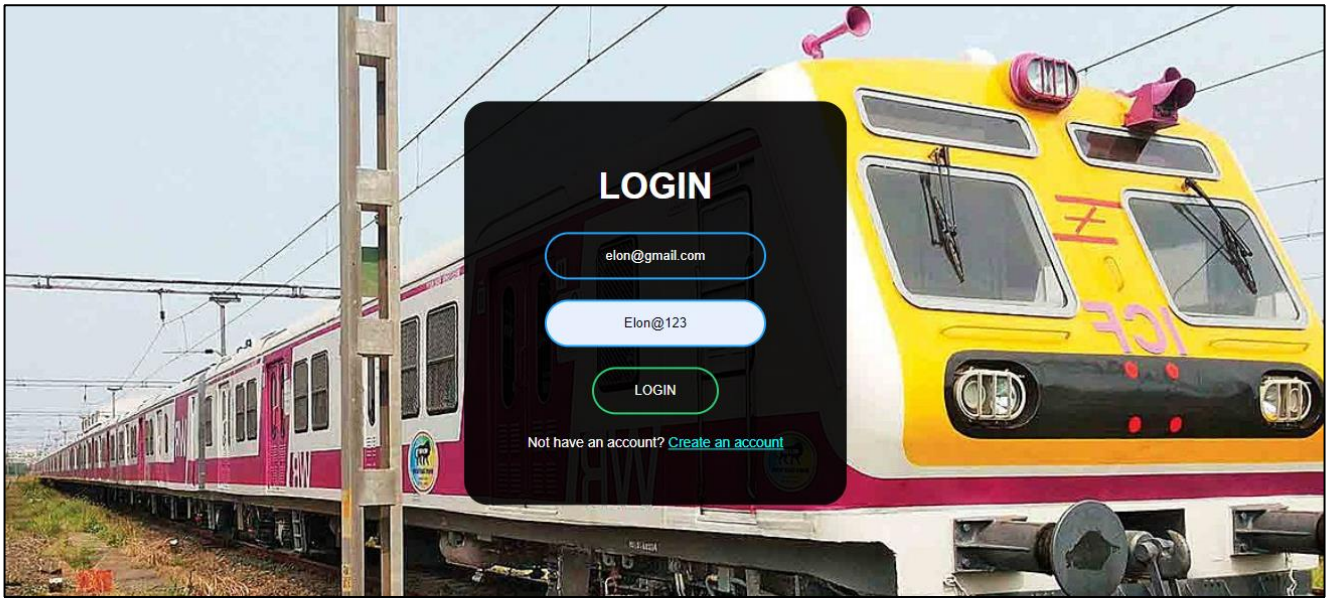
```
<html>
    <head>
        <title>LOGIN</title>
        <style>
            body{
                margin: 0;
                padding: 0;
                font-family: sans-serif;
                background: url("Photos/LocalTrain.jpg");
                background-size: cover;
            }
            p{
                color: white;
                font-family: sans-serif;
                font-size: 14px;
            }
            a:visited{
                color: aqua;
            }
            .box{
                width: 300px;
                padding: 40px;
                position: absolute;
                top: 50%;
                left: 50%;
                transform: translate(-50%,-50%);
/*                background:red;*/

            }
            .box h1{
                color: white;
                font-size: 36px;
                font-weight: 700;
            }
            .box input[type = "text"],.box input[type = "password"]{
                border: 0;
                background: none;
                display: block;
                margin: 20px auto;
                text-align: center;
                border: 2px solid #23a7ff;
                padding: 14px 10px;
                width: 220px;
                outline: none;
                color: white;
                border-radius: 24px;
                transition: 0.25s;
                cursor: pointer;
            }

            .box input[type = "text"]:hover,.box input[type = "password"]:hover{
                width: 260px;
                border-color: aqua;
            }
            .box input[type = "text"]:focus,.box input[type = "password"]:focus{
                border-color: red;
            }

            .box .button{
                border: 0;
                background: none;
                display: block;
                margin: 20px auto;
                text-align: center;
                border: 2px solid #2ecc71;
                padding: 14px 40px;
                outline: none;
                color: white;
                border-radius: 24px;
                transition: 0.25s;
                cursor: pointer;
            }

            .box .button:hover{
                background: #2ecc71;
            }

        </style>
    </head>
    <body>
        <form class="box" method="POST" action="login.jsp">

            <h1>LOGIN</h1>
            <input type="text" name="GmailID" placeholder="Email Address" required/>
            <input type="text" name="Password" placeholder="Password" required/>
            <input type="submit" value="LOGIN" class="button">
            <p>Not have an account? <a href="signup.html">Create an account</a></p>
        </form>
    </body>
</html>
```

login.jsp

```
<%--
    Document   : login
    Created on : 14 Aug, 2022, 12:47:20 PM
    Author     : dell
--%>
<%@page import="com.google.protobuf.Type"%>
<%@page
    import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*,java.util.*"%>
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%
String email = request.getParameter("GmailID");
```

```java
String pass = request.getParameter("Password");
try {

    SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
    Session session2 = factory.openSession();
    Query query = session2.createQuery("from User where email=:uemail and upassword=:upassword");
    query.setParameter("uemail", email);
    query.setParameter("upassword", pass);

    List<User> list = query.list();
    if (list.size() >= 1) {
    out.print("Logedin");
    response.sendRedirect("Bookticket.jsp");
    session.setAttribute("email", email);
    session.setAttribute("name",list.get(0).getName() );
    session.setAttribute("user_id", list.get(0).getId());
    } else {
    out.print("Invalid Creds");
    }

} catch (Exception e) {
    e.printStackTrace();
    out.print(e.toString());
}
%>
```

Output:

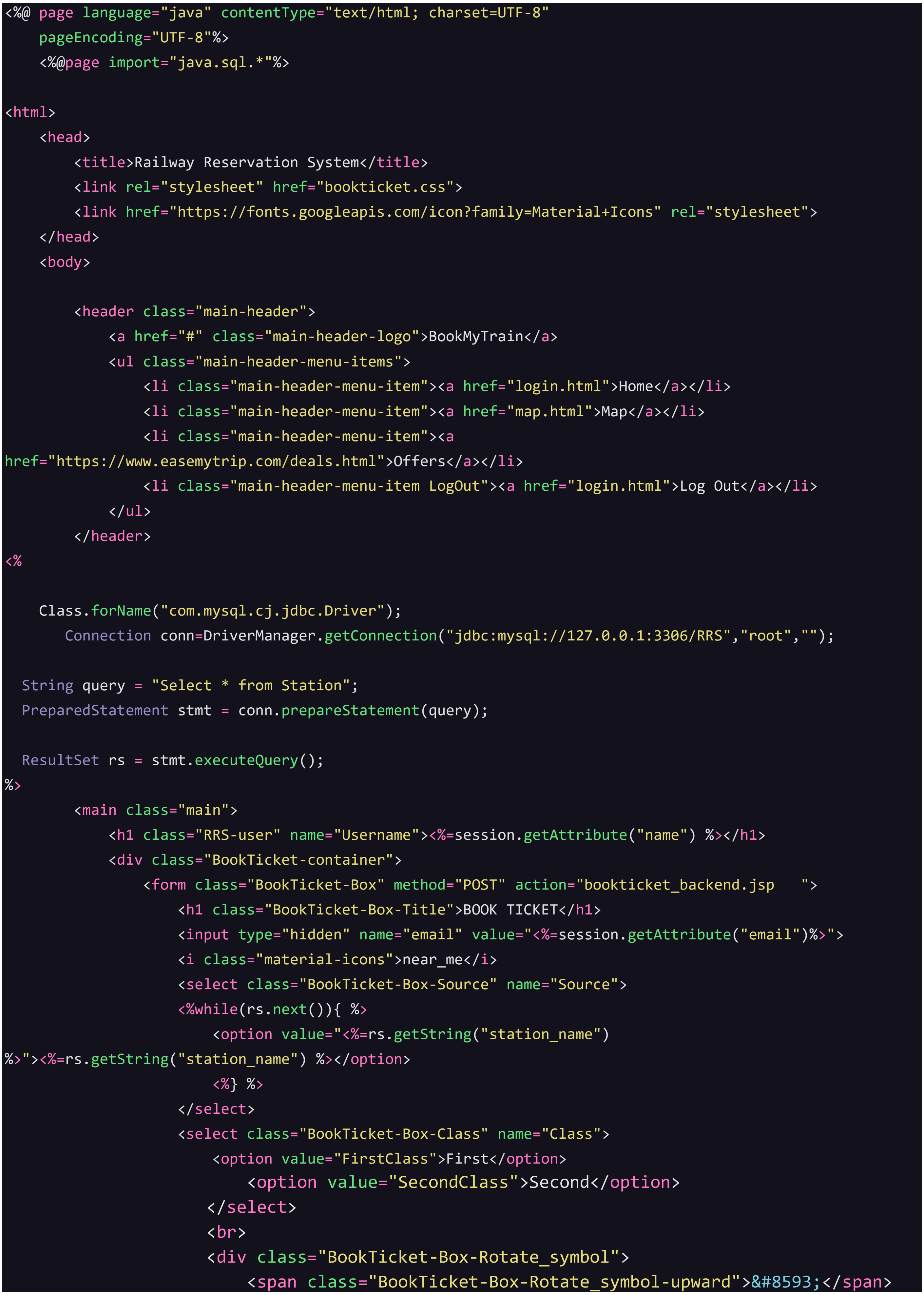


LOGIN

elon@gmail.com

Elon@123

Bookticket1.jsp

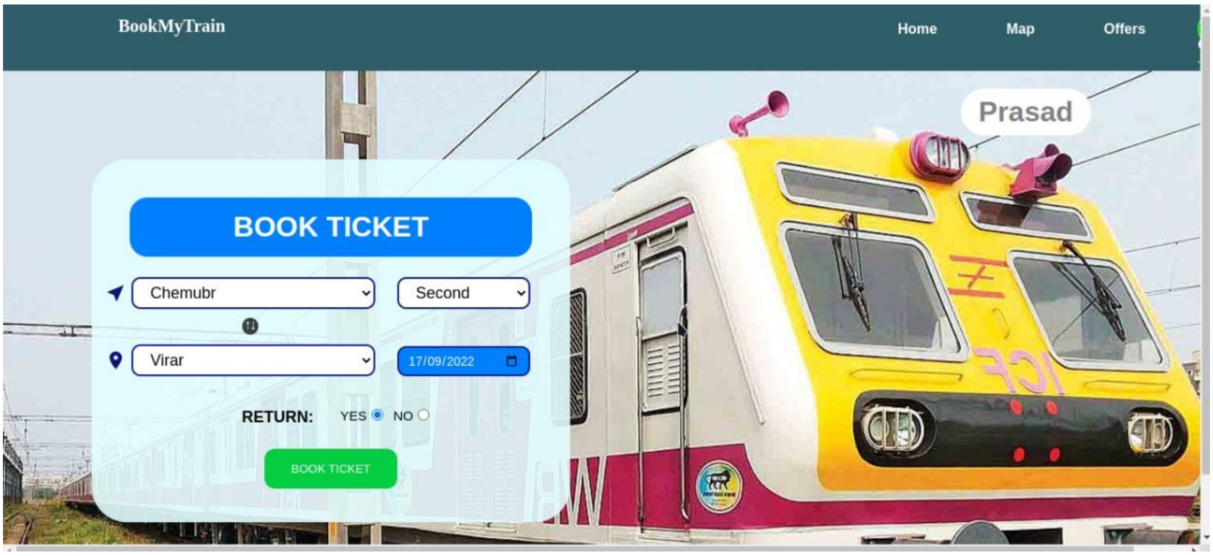


```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@page import="java.sql.*"%>

<html>
    <head>
        <title>Railway Reservation System</title>
        <link rel="stylesheet" href="bookticket.css">
        <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
    </head>
    <body>

        <header class="main-header">
            <a href="#" class="main-header-logo">BookMyTrain</a>
            <ul class="main-header-menu-items">
                <li class="main-header-menu-item"><a href="login.html">Home</a></li>
                <li class="main-header-menu-item"><a href="map.html">Map</a></li>
                <li class="main-header-menu-item"><a
href="https://www.easemytrip.com/deals.html">Offers</a></li>
                <li class="main-header-menu-item LogOut"><a href="login.html">Log Out</a></li>
            </ul>
        </header>
<%

    Class.forName("com.mysql.cj.jdbc.Driver");
       Connection conn=DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/RRS","root","");

  String query = "Select * from Station";
  PreparedStatement stmt = conn.prepareStatement(query);

  ResultSet rs = stmt.executeQuery();
%>
        <main class="main">
            <h1 class="RRS-user" name="Username"><%=session.getAttribute("name") %></h1>
            <div class="BookTicket-container">
                <form class="BookTicket-Box" method="POST" action="bookticket_backend.jsp   ">
                    <h1 class="BookTicket-Box-Title">BOOK TICKET</h1>
                    <input type="hidden" name="email" value="<%=session.getAttribute("email")%>">
                    <i class="material-icons">near_me</i>
                    <select class="BookTicket-Box-Source" name="Source">
                    <%while(rs.next()){ %>
                        <option value="<%=rs.getString("station_name")
%>"><%=rs.getString("station_name") %></option>
                        <%} %>
                    </select>
                    <select class="BookTicket-Box-Class" name="Class">
                        <option value="FirstClass">First</option>
                            <option value="SecondClass">Second</option>
                        </select>
                        <br>
                        <div class="BookTicket-Box-Rotate_symbol">
                            <span class="BookTicket-Box-Rotate_symbol-upward">&#8593;</span>
                        <span class="BookTicket-Box-Rotate_symbol-downward">&#8595;</span>
                    </div>
                    <br>
                    <br>
                    <i class="material-icons">place</i>
                    <select class="BookTicket-Box-Destination" name="Destination">
                      <% rs = stmt.executeQuery();
                      while(rs.next()){ %>
                        <option value="<%=rs.getString("station_name")
%>"><%=rs.getString("station_name") %></option>
                        <%} %>

                    </select>
                    <input type="date" class="BookTicket-Box-Date" name="Date" required><br>

                    <br><br>
                    <div class="BookTicket-Box-Return">
                        <label class="BookTicket-Box-Return-label">RETURN<b>:</b></label>
                        <span class="BookTicket-Box-Return-Radio-Yes">YES<input type="radio"
class="BookTicket-Box-Return-Radio-Yes-Radio" id="Yes" name="return" value="Yes"></span>
                        <span class="BookTicket-Box-Return-Radio-No">NO<input type="radio"
class="BookTicket-Box-Return-Radio-No-Radio" id="No" name="return" value="No" checked="checked"></span>
                    </div>

                    <input type="Submit" class="BookTicket-Box-BTN" value="BOOK TICKET">
                </form>
            </div>
        </main>

        <footer></footer>

    </body>
</html>
```

Bookticket_backend.jsp

```
<%@page import="java.sql.*"%>
<%@page
    import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*,java.util.*"%>
<%@page import="java.sql.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%
String email = request.getParameter("email");
int user_id = Integer.parseInt(session.getAttribute("user_id").toString());
String Source = request.getParameter("Source");
String Destination = request.getParameter("Destination");
String Class = request.getParameter("Class");
String Date = request.getParameter("Date");
String Return = request.getParameter("return");

try {
    SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
    Session session2 = factory.openSession();
    Transaction t = session2.beginTransaction();
    Ticket ticket = new Ticket((User) session2.load(User.class, user_id), Source, Destination, Class,
Date, Return);
    session2.save(ticket);
    t.commit();
    out.print("Ticket Booked Sucessfully");
    response.sendRedirect("ShowTicket.jsp");
} catch (Exception e) {
    out.print(e);
    e.printStackTrace();

}
%>
```

Output:



Chemubr

Second

Virar

17/09/2022

RETURN: YES ● NO ○

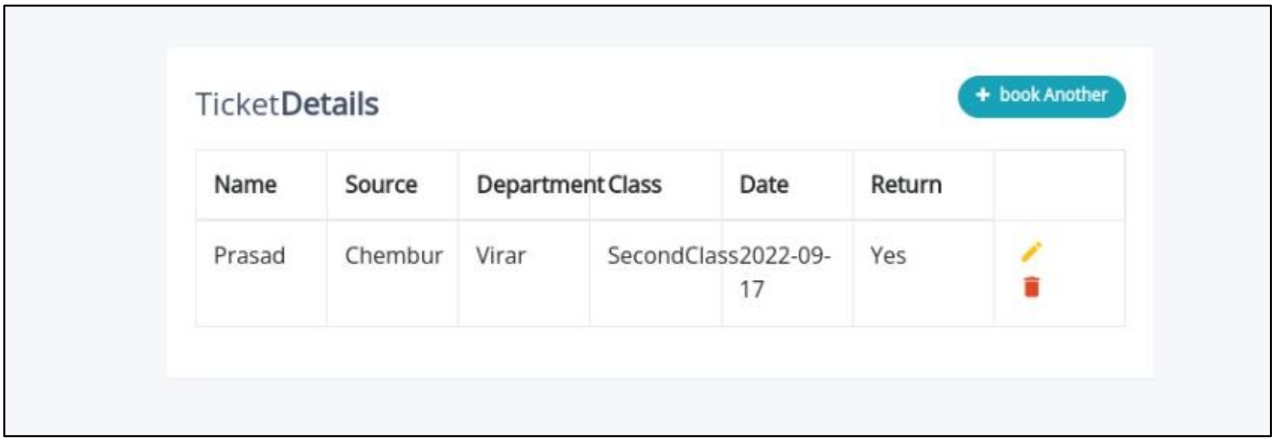
showticket.jsp

```
<%@page import="java.sql.*"%>
<%@page import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*,java.util.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%
try {

    SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
    Session session2 = factory.openSession();
    Transaction t = session2.beginTransaction();
    int user_id = Integer.parseInt(session.getAttribute("user_id").toString());
    Query query =session2.createQuery("from Ticket WHERE user_id = :uid");
    query.setParameter("uid",user_id);
    List<Ticket> list = query.list();
    t.commit();

%>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport"
    content="width=device-width, initial-scale=1, shrink-to-fit=no">
<title>Show Ticket</title>
<link rel="stylesheet"
    href="https://fonts.googleapis.com/css?family=Roboto|Varela+Round|Open+Sans">
<link rel="stylesheet"
    href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">
<link rel="stylesheet"
    href="https://fonts.googleapis.com/icon?family=Material+Icons">
<link rel="stylesheet"
    href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css">
<script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
<script
    src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"></script>
<script
    src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"></script>
<style>
body {
    color: #404E67;
    background: #F5F7FA;
    font-family: 'Open Sans', sans-serif;
}

.table-wrapper {
    width: 700px;
    margin: 30px auto;
    background: #fff;
    padding: 20px;
    box-shadow: 0 1px 1px rgba(0, 0, 0, .05);
}
```

```
.table-title {
    padding-bottom: 10px;
    margin: 0 0 10px;
}

.table-title h2 {
    margin: 6px 0 0;
    font-size: 22px;
}

.table-title .add-new {
    float: right;
    height: 30px;
    font-weight: bold;
    font-size: 12px;
    text-shadow: none;
    min-width: 100px;
    border-radius: 50px;
    line-height: 13px;
}

.table-title .add-new i {
    margin-right: 4px;
}

table.table {
    table-layout: fixed;
}

table.table tr th, table.table tr td {
    border-color: #e9e9e9;
}

table.table th i {
    font-size: 13px;
    margin: 0 5px;
    cursor: pointer;
}

table.table th:last-child {
    width: 100px;
}

table.table td a {
    cursor: pointer;
    display: inline-block;
    margin: 0 5px;
    min-width: 24px;
}

table.table td a.add {
    color: #27C46B;
}
```

```
table.table td a.edit {
    color: #FFC107;
}

table.table td a.delete {
    color: #E34724;
}

table.table td i {
    font-size: 19px;
}

table.table td a.add i {
    font-size: 24px;
    margin-right: -1px;
    position: relative;
    top: 3px;
}

table.table .form-control {
    height: 32px;
    line-height: 32px;
    box-shadow: none;
    border-radius: 2px;
}

table.table .form-control.error {
    border-color: #f50000;
}

table.table td .add {
    display: none;
}
</style>
</head>
<body>
    <div class="container-lg">
        <div class="table-responsive">
            <div class="table-wrapper">
                <div class="table-title">
                    <div class="row">
                        <div class="col-sm-8">
                            <h2>
                                Ticket<b>Details</b>
                            </h2>
                        </div>
                        <div class="col-sm-4">
                            <a type="button" href="Bookticket.jsp" class="btn btn-info add-new">
                                <i class="fa fa-plus" ></i> book Another
                            </a>
                        </div>
                    </div>
                </div>
                <table class="table table-bordered">
                    <thead>
                        <tr>
                            <th>Name</th>
                            <th>Source</th>
                            <th>Department</th>
                            <th>Class</th>
                            <th>Date</th>
                            <th>Return</th>


                        </tr>
                    </thead>
                    <tbody>
                    <% for (Ticket ticket : list) {%>
                        <tr>
                            <td><%=ticket.getUser().getName() %></td>
                            <td><%=ticket.getSource() %></td>
                            <td><%=ticket.getDestination() %></td>
                            <td><%=ticket.getClasss() %></td>
                            <td><%=ticket.getDate() %></td>
                            <td><%=ticket.getReturn() %></td>

                            <td><a class="add" title="Add" data-toggle="tooltip"><i
                                    class="material-icons">&#xE03B;</i></a> <a
href="editTicket.jsp?id=<%= ticket.getId() %>" class="edit"
                                title="Edit" data-toggle="tooltip"><i class="material-
icons">&#xE254;</i></a>
                                <a class="delete" href="delete_ticket.jsp?id=<%= ticket.getId() %>"
title="Delete" data-toggle="tooltip"><i
                                    class="material-icons">&#xE872;</i></a></td>
                        </tr>
                        <%}%>

                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>
</html>

<%
     } catch (Exception e) {
        e.printStackTrace();
    }
%>
```

Output :



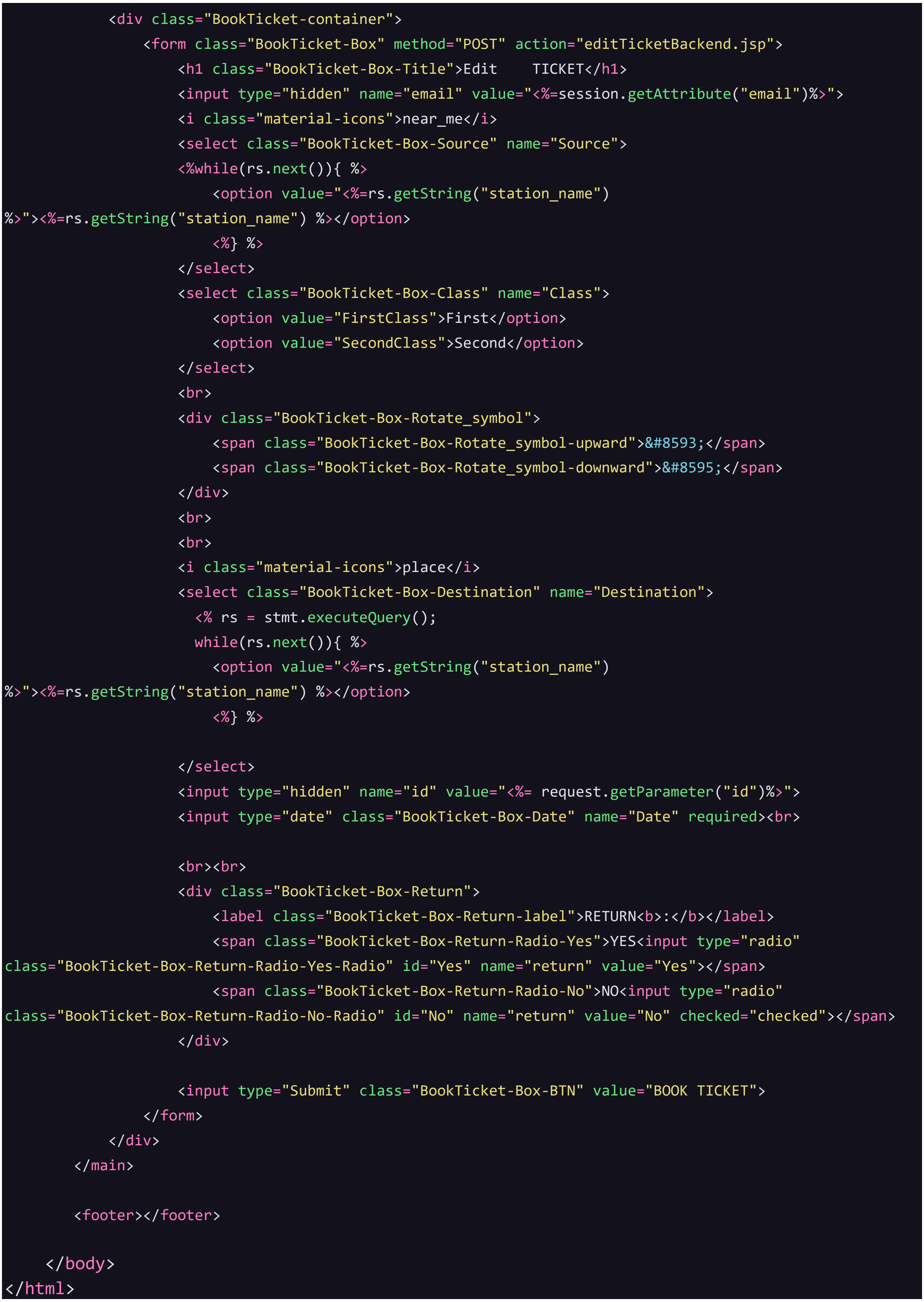
TicketDetails

+ book Another

Name	Source	Department	Class	Date	Return	
Prasad	Chembur	Virar	SecondClass	2022-09-17	Yes	✎ 🗑

Edit_ticket.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@page import="java.sql.*"%>

<html>
    <head>
        <title>Railway Reservation System</title>
        <link rel="stylesheet" href="bookticket.css">
        <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
    </head>
    <body>

        <header class="main-header">
            <a href="#" class="main-header-logo">BookMyTrain</a>
            <ul class="main-header-menu-items">
                <li class="main-header-menu-item"><a href="login.html">Home</a></li>
                <li class="main-header-menu-item"><a href="map.html">Map</a></li>
                <li class="main-header-menu-item"><a
href="https://www.easemytrip.com/deals.html">Offers</a></li>
                <li class="main-header-menu-item LogOut"><a href="login.html">Log Out</a></li>
            </ul>
        </header>
<%

    Class.forName("com.mysql.cj.jdbc.Driver");
       Connection conn=DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/RRS","root","");

  String query = "Select * from Station";
  PreparedStatement stmt = conn.prepareStatement(query);

  ResultSet rs = stmt.executeQuery();
%>
         <main class="main">
              <h1 class="RRS-user" name="Username"><%=session.getAttribute("name") %></h1>
```

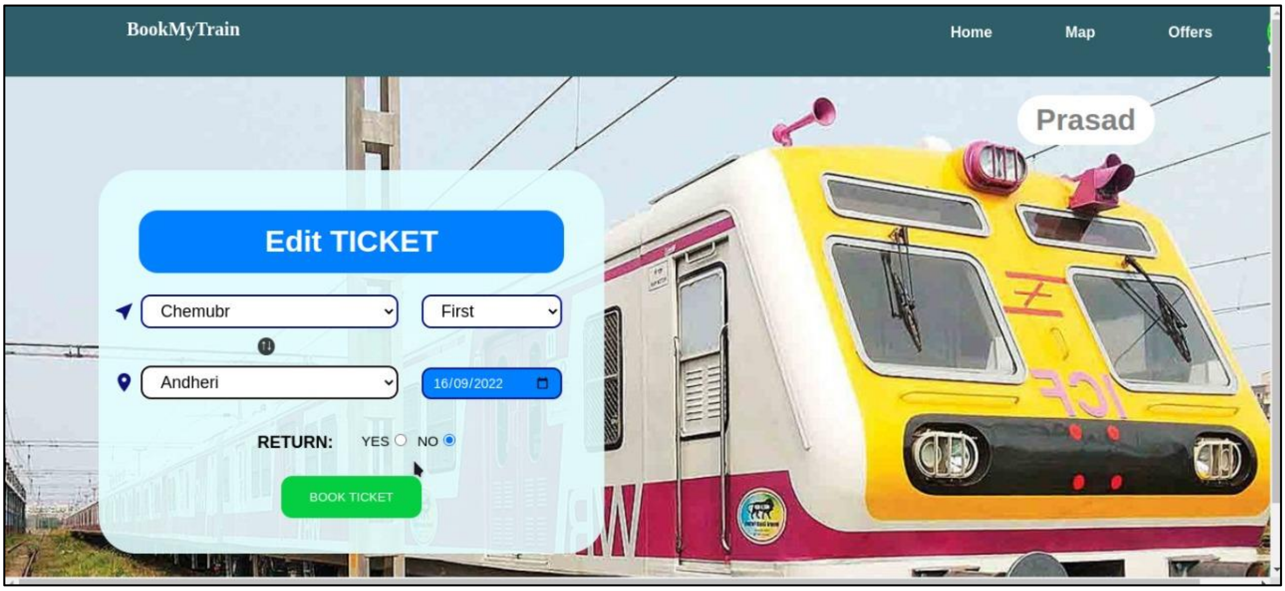


```
            <div class="BookTicket-container">
                <form class="BookTicket-Box" method="POST" action="editTicketBackend.jsp">
                    <h1 class="BookTicket-Box-Title">Edit    TICKET</h1>
                    <input type="hidden" name="email" value="<%=session.getAttribute("email")%>">
                    <i class="material-icons">near_me</i>
                    <select class="BookTicket-Box-Source" name="Source">
                    <%while(rs.next()){ %>
                        <option value="<%=rs.getString("station_name")
%>"><%=rs.getString("station_name") %></option>
                        <%} %>
                    </select>
                    <select class="BookTicket-Box-Class" name="Class">
                        <option value="FirstClass">First</option>
                        <option value="SecondClass">Second</option>
                    </select>
                    <br>
                    <div class="BookTicket-Box-Rotate_symbol">
                        <span class="BookTicket-Box-Rotate_symbol-upward">&#8593;</span>
                        <span class="BookTicket-Box-Rotate_symbol-downward">&#8595;</span>
                    </div>
                    <br>
                    <br>
                    <i class="material-icons">place</i>
                    <select class="BookTicket-Box-Destination" name="Destination">
                      <% rs = stmt.executeQuery();
                      while(rs.next()){ %>
                        <option value="<%=rs.getString("station_name")
%>"><%=rs.getString("station_name") %></option>
                        <%} %>

                    </select>
                    <input type="hidden" name="id" value="<%= request.getParameter("id")%>">
                    <input type="date" class="BookTicket-Box-Date" name="Date" required><br>

                    <br><br>
                    <div class="BookTicket-Box-Return">
                        <label class="BookTicket-Box-Return-label">RETURN<b>:</b></label>
                        <span class="BookTicket-Box-Return-Radio-Yes">YES<input type="radio"
class="BookTicket-Box-Return-Radio-Yes-Radio" id="Yes" name="return" value="Yes"></span>
                        <span class="BookTicket-Box-Return-Radio-No">NO<input type="radio"
class="BookTicket-Box-Return-Radio-No-Radio" id="No" name="return" value="No" checked="checked"></span>
                    </div>

                    <input type="Submit" class="BookTicket-Box-BTN" value="BOOK TICKET">
                </form>
            </div>
        </main>

        <footer></footer>

    </body>
</html>
```

Edit_ticket_backend.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
    <%@page import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*,java.util.*"%>
<%@page import="java.sql.*" %>
<%
    int ticketId = Integer.parseInt(request.getParameter("id"));
    String Source=request.getParameter("Source");
    String Destination=request.getParameter("Destination");
    String Class=request.getParameter("Class");
    String Date=request.getParameter("Date");
    String Return=request.getParameter("return");




    try{
    SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
        Session session2 = factory.openSession();
        Transaction t = session2.beginTransaction();
        Query query =session2.createQuery("Update Ticket t set t.source = :src ,t.destination =
:dst,t.classs = :cls,t.Return = :rtn,t.Date = :date  WHERE id = :tid");


       query.setParameter("src",Source);
       query.setParameter("dst",Destination);
       query.setParameter("cls",Class);
       query.setParameter("rtn",Return);
       query.setParameter("date",Date);
       query.setParameter("tid",ticketId);
       query.executeUpdate();
       t.commit();
       response.sendRedirect("ShowTicket.jsp");
    }

        catch(Exception e){
                out.print(e);
            }

    %>
```

Output :



Chemubr
First
Andheri
16/09/2022
RETURN: YES ○ NO ●

Deleteticket.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@page import="java.sql.*"%>
<%@page import="java.sql.*,org.hibernate.*,org.hibernate.cfg.*,PojoFiles.*,java.util.*"%>
<%
int Ticketid = Integer.parseInt(request.getParameter("id"));
SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
Session session2 = factory.openSession();
Transaction t  = session2.beginTransaction();
Query query = session2.createQuery("Delete FROM  Ticket WHERE id = :tid");
query.setParameter("tid",Ticketid);
query.executeUpdate();
t.commit();
response.sendRedirect("ShowTicket.jsp");
%>
```

Database :

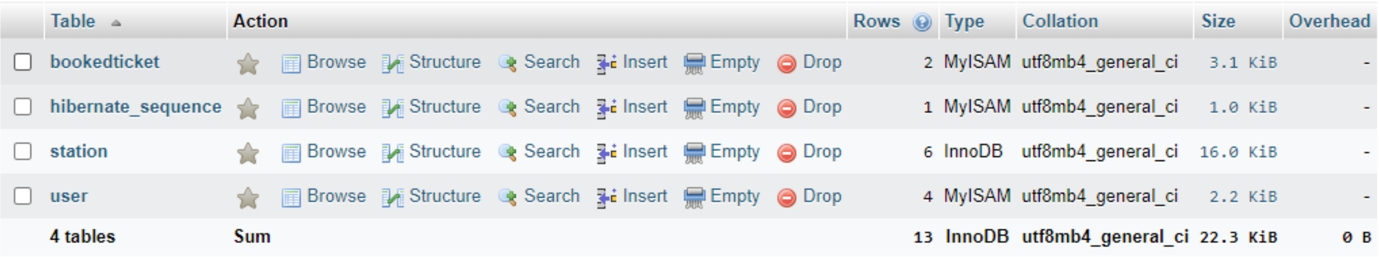
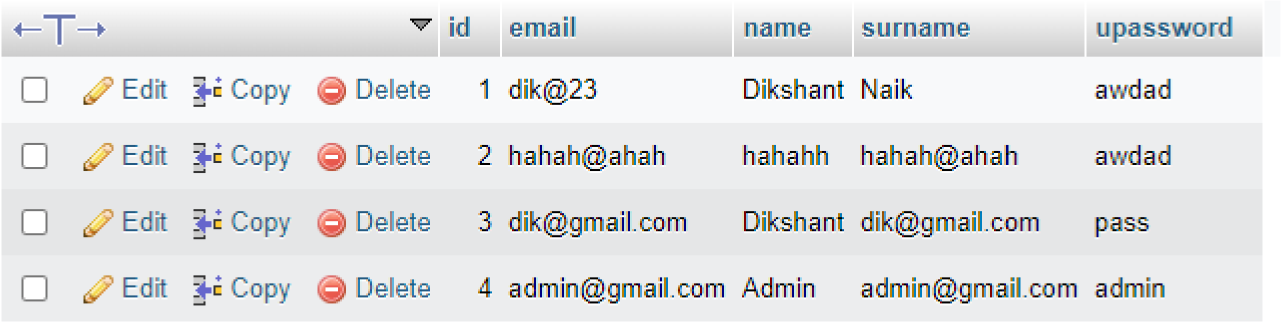
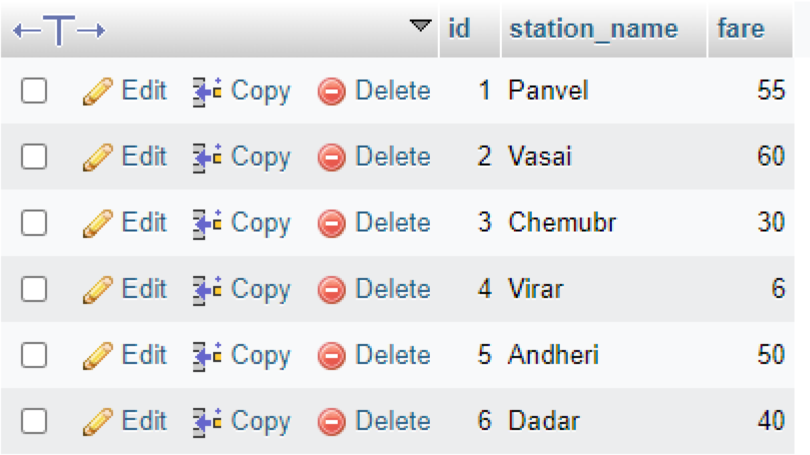


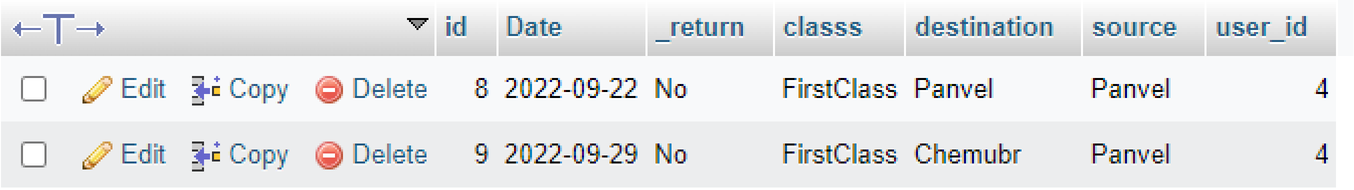
	Table	Action	Rows	Type	Collation	Size	Overhead
☐	bookedticket	★ Browse Structure Search Insert Empty Drop	2	MyISAM	utf8mb4_general_ci	3.1 KiB	-
☐	hibernate_sequence	★ Browse Structure Search Insert Empty Drop	1	MyISAM	utf8mb4_general_ci	1.0 KiB	-
☐	station	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_general_ci	16.0 KiB	-
☐	user	★ Browse Structure Search Insert Empty Drop	4	MyISAM	utf8mb4_general_ci	2.2 KiB	-
	4 tables	Sum	13	InnoDB	utf8mb4_general_ci	22.3 KiB	0 B



←T→				id	email	name	surname	upassword
☐	Edit	Copy	Delete	1	dik@23	Dikshant	Naik	awdad
☐	Edit	Copy	Delete	2	hahah@ahah	hahahh	hahah@ahah	awdad
☐	Edit	Copy	Delete	3	dik@gmail.com	Dikshant	dik@gmail.com	pass
☐	Edit	Copy	Delete	4	admin@gmail.com	Admin	admin@gmail.com	admin



←T→				id	station_name	fare
☐	Edit	Copy	Delete	1	Panvel	55
☐	Edit	Copy	Delete	2	Vasai	60
☐	Edit	Copy	Delete	3	Chemubr	30
☐	Edit	Copy	Delete	4	Virar	6
☐	Edit	Copy	Delete	5	Andheri	50
☐	Edit	Copy	Delete	6	Dadar	40



←T→				id	Date	_return	classs	destination	source	user_id
☐	Edit	Copy	Delete	8	2022-09-22	No	FirstClass	Panvel	Panvel	4
☐	Edit	Copy	Delete	9	2022-09-29	No	FirstClass	Chemubr	Panvel	4

THANK YOU!

Pillai