

```

package RailwayDetails;

import java.sql.*;
import java.util.*;

public class RailwayDetails {

    static String DB_URL = "jdbc:mysql://localhost:3306/RailwayDetail";
    static String USER = "root";
    static String PASSWORD = "root";

    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");

            Connection con = DriverManager.getConnection(DB_URL, USER, PASSWORD);
            Statement stmt = con.createStatement();

            // Create table
            String createTableQuery = "CREATE TABLE IF NOT EXISTS
RailwayTicketDetail("
+ "ticket_id VARCHAR(20) NOT NULL,"
+ "name VARCHAR(50) NOT NULL,"
+ "source VARCHAR(50) NOT NULL,"
+ "destination VARCHAR(50) NOT NULL,"
+ "class VARCHAR(20) NOT NULL,"
+ "age VARCHAR(5) NOT NULL,"
+ "gender VARCHAR(1) NOT NULL,"
+ "ph_number VARCHAR(10) NOT NULL,"
+ "fare VARCHAR(50) NOT NULL)";
            stmt.executeUpdate(createTableQuery);
            System.out.println("Table created successfully!");

            Scanner sc = new Scanner(System.in);
            int choice = -1;
            while (choice != 0) {
                System.out.println("\n----- RAILWAY BOOKING TICKET SYSTEM -----");
                System.out.println("1. Book a Ticket");
                System.out.println("2. Update Ticket Fare");
                System.out.println("3. Update Ticket Class");
                System.out.println("4. Update Ticket Destination");
                System.out.println("5. Update passenger Age");
                System.out.println("6. Update passenger ph_number");
                System.out.println("7. Delete a Ticket");
                System.out.println("8. Show All Tickets");
                System.out.println("9. Exit");
                System.out.print("Enter your choice: ");
                choice = sc.nextInt();
                sc.nextLine();

                switch (choice) {
                    case 1:
                        // Book a ticket
                        System.out.print("enter the number of passenger:");
                        int n=sc.nextInt();
                        sc.nextLine();
                        for(int i=0;i<n;i++)
                        {
                            System.out.print("Enter passenger id: ");

```

```

String ticket_id = sc.nextLine();
System.out.print("Enter passenger name: ");
String name = sc.nextLine();
System.out.print("Enter source station: ");
String source = sc.nextLine();
System.out.print("Enter destination station: ");
String destination = sc.nextLine();
System.out.print("Enter class (First/Second): ");
String ticketClass = sc.nextLine();
System.out.print("Enter passenger age: ");
String age = sc.nextLine();
System.out.print("Enter passenger gender: ");
String gender = sc.nextLine();
System.out.print("Enter passenger ph_number: ");
String ph_number = sc.nextLine();
System.out.print("Enter ticket fare: ");
String fare = sc.nextLine();
String insertQuery = "INSERT INTO RailwayTicketDetail(ticket_id,name,
source, destination,class,age,gender,ph_number,fare) VALUES
(?,?,?,?,?,?,?,?,?)";
PreparedStatement insertStmt = con.prepareStatement(insertQuery);
insertStmt.setString(1,ticket_id);
insertStmt.setString(2, name);
insertStmt.setString(3, source);
insertStmt.setString(4, destination);
insertStmt.setString(5, ticketClass);
insertStmt.setString(6, age);
insertStmt.setString(7, gender);
insertStmt.setString(8, ph_number);
insertStmt.setString(9, fare);
insertStmt.executeUpdate();
}
System.out.println("Ticket booked successfully!");
break;

case 2:
// Update ticket fare
System.out.print("Enter ticket ID to update fare: ");
String ticketId = sc.nextLine();
System.out.print("Enter new fare: ");
String newFare = sc.nextLine();
String updateQuery = "UPDATE RailwayTicketDetail SET fare= ? WHERE
ticket_id = ?";
PreparedStatement updateStmt = con.prepareStatement(updateQuery);
updateStmt.setString(1, newFare);
updateStmt.setString(2, ticketId);
int rowsUpdated = updateStmt.executeUpdate();
if (rowsUpdated > 0) {
System.out.println("Fare updated successfully!");
} else {
System.out.println("Ticket not found!");
}
break;
case 3:
// Update ticket class
System.out.print("Enter ticket ID to update Class: ");
String ticketId1 = sc.nextLine();
System.out.print("Enter new class: ");
String newClass = sc.nextLine();
String updateQuery1 = "UPDATE RailwayTicketDetail SET class = ? WHERE
ticket_id = ?";

```

```

PreparedStatement updateStmt1 = con.prepareStatement(updateQuery1);
updateStmt1.setString(1, newClass);
updateStmt1.setString(2, ticketId1);
int rowsUpdated1 = updateStmt1.executeUpdate();
if (rowsUpdated1 > 0) {
    System.out.println("Class updated successfully!");
} else {
    System.out.println("Ticket not found!");
}
break;
case 4:
    // Update ticket Destination
    System.out.print("Enter ticket ID to update Destination: ");
    String ticketId2 = sc.nextLine();
    System.out.print("Enter new Destination: ");
    String newDestination = sc.nextLine();
    String updateQuery2 = "UPDATE RailwayTicketDetail SET destination= ?
    WHERE ticket_id = ?";
    PreparedStatement updateStmt2 = con.prepareStatement(updateQuery2);
    updateStmt2.setString(1, newDestination);
    updateStmt2.setString(2, ticketId2);
    int rowsUpdated2 = updateStmt2.executeUpdate();
    if (rowsUpdated2 > 0) {
        System.out.println("Class updated successfully!");
    } else {
        System.out.println("Ticket not found!");
    }
    break;
case 5:
    // Update passenger age
    System.out.print("Enter ticket ID to update Age: ");
    String ticketId3 = sc.nextLine();
    System.out.print("Enter new Age: ");
    String newAge = sc.nextLine();
    String updateQuery3= "UPDATE RailwayTicketDetail SET age= ? WHERE
    ticket_id = ?";
    PreparedStatement updateStmt3 = con.prepareStatement(updateQuery3);
    updateStmt3.setString(1, newAge);
    updateStmt3.setString(2, ticketId3);
    int rowsUpdated3 = updateStmt3.executeUpdate();
    if (rowsUpdated3 > 0) {
        System.out.println("Class updated successfully!");
    } else {
        System.out.println("Ticket not found!");
    }
    break;
case 6:
    // Update passenger ph_number
    System.out.print("Enter ticket ID to update ph_number: ");
    String ticketId4 = sc.nextLine();
    System.out.print("Enter new ph_number ");
    String newph_number = sc.nextLine();
    String updateQuery4 = "UPDATE RailwayTicketDetail SET ph_number= ?
    WHERE ticket_id = ?";
    PreparedStatement updateStmt4 = con.prepareStatement(updateQuery4);
    updateStmt4.setString(1, newph_number);
    updateStmt4.setString(2, ticketId4);
    int rowsUpdated4 = updateStmt4.executeUpdate();
    if (rowsUpdated4 > 0) {
        System.out.println("Class updated successfully!");
    } else {

```

```

System.out.println("Ticket not found!");
}
break;
case 7:
// Delete a ticket
System.out.print("Enter ticket ID to delete: ");
String deleteTicketId = sc.nextLine();
String deleteQuery = "DELETE FROM RailwayTicketDetail WHERE ticket_id
= ?";
PreparedStatement deleteStmt = con.prepareStatement(deleteQuery);
deleteStmt.setString(1, deleteTicketId);
int rowsDeleted = deleteStmt.executeUpdate();
if (rowsDeleted > 0) {
System.out.println("Ticket deleted successfully!");
} else {
System.out.println("Ticket not found!");
}
break;

case 8:
// Show all tickets
String selectQuery = "SELECT * FROM RailwayTicketDetail";
ResultSet resultSet = stmt.executeQuery(selectQuery);
if (!resultSet.next()) {
System.out.println("No tickets found!");
}
else
{
System.out.print(" ----- TICKET DETAILS ----- ");
System.out.println();
System.out.print("\n-----
-----\n");
String output = String.format(" %-20s%-20s%-20s%-20s%-20s%-20s%-
20s%s", "Ticket_ID",
"Name", "Source", "Destination", "Class", "age", "gender", "ph_number", "Far
e");
System.out.print(output);
System.out.print("\n-----
-----\n");
System.out.println();
while (resultSet.next())
{
String output1 = String.format(" %-15s%-20s%-20s%-20s%-20s%-
20s%-
20s%s", resultSet.getString(1), resultSet.getString(2), resultSet.getSt
ring(3), resultSet.getString(4), resultSet.getString(5), resultSet.getSt
ring(6), resultSet.getString(7), resultSet.getString(8), resultSet.getSt
ring(9));
System.out.print(output1);
System.out.println();
}
System.out.print("\n-----
-----\n");
}
break;

case 9:
//exit program

```

```

System.out.println("Exiting the program.");
break;

default:
//invalid choice
System.out.println("Invalid choice! Please try again.");
}
}

con.close();
} catch (ClassNotFoundException e)
{
e.printStackTrace();
} catch (SQLException e) {
e.printStackTrace();
}
}
}
}

```

Before Insertion the Values

----- TICKET DETAILS -----								
Ticket_ID	Name	Source	Destination	Class	age	gender	ph_number	Fare
7	Dhoni	Ranchi	Chennai	First	42	M	944XXXX404	68575
1	Rahul	Bangalore	Lucknow	Second	31	M	908XXXX066	37569
5	Jemimah	Mumbai	Chennai	Second	22	F	960XXXX444	33560
33	Hardik	Gujarat	Delhi	First	29	M	994XXXX094	56780
77	Harmanpreet	Punjab	Mumbai	Second	34	F	883XXXX299	36750

After Insertion value of ID 18

----- TICKET DETAILS -----								
Ticket_ID	Name	Source	Destination	Class	age	gender	ph_number	Fare
7	Dhoni	Ranchi	Chennai	First	42	M	944XXXX404	68575
1	Rahul	Bangalore	Lucknow	First	31	M	908XXXX066	37569
5	Jemimah	Mumbai	Chennai	Second	22	F	960XXXX444	33560
33	Hardik	Gujarat	Delhi	First	29	M	994XXXX094	56780
77	Harmanpreet	Punjab	Mumbai	Second	34	F	883XXXX299	36750
18	Virat	Delhi	Bangalore	First	38	M	989XXXX744	36780

**After Updating the new class value
For ID 1**

----- TICKET DETAILS -----

Ticket_ID	Name	Source	Destination	Class	age	gender	ph_number	Fare
7	Dhoni	Ranchi	Chennai	First	42	M	944XXXX404	68575
1	Rahul	Bangalore	Lucknow	First	31	M	908XXXX066	37569
5	Jemimah	Mumbai	Chennai	Second	22	F	960XXXX444	33560
33	Hardik	Gujarat	Delhi	First	29	M	994XXXX094	56780
77	Harmanpreet	Punjab	Mumbai	Second	34	F	883XXXX299	36750

**After Updating the new Destination
for ID 18**

----- TICKET DETAILS -----

Ticket_ID	Name	Source	Destination	Class	age	gender	ph_number	Fare
7	Dhoni	Ranchi	Chennai	First	42	M	944XXXX404	68575
1	Rahul	Bangalore	Lucknow	First	31	M	908XXXX066	37569
5	Jemimah	Mumbai	Chennai	Second	22	F	960XXXX444	33560
33	Hardik	Gujarat	Delhi	First	29	M	994XXXX094	56780
77	Harmanpreet	Punjab	Mumbai	Second	34	F	883XXXX299	36750
18	Virat	Delhi	Hyderabad	First	38	M	989XXXX744	36780

After Deleting the value of ID 18

----- TICKET DETAILS -----

Ticket_ID	Name	Source	Destination	Class	age	gender	ph_number	Fare
7	Dhoni	Ranchi	Chennai	First	42	M	944XXXX404	68575
1	Rahul	Bangalore	Lucknow	First	31	M	908XXXX066	37569
5	Jemimah	Mumbai	Chennai	Second	22	F	960XXXX444	33560
33	Hardik	Gujarat	Delhi	First	29	M	994XXXX094	56780
77	Harmanpreet	Punjab	Mumbai	Second	34	F	883XXXX299	36750