Railway crossing status

1. home.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>Welcome to the Railway Crossing Application</h1>
    <h3>Please select a section:</h3>

    <ul>
        <li><a href="government_users.jsp">Government Section</a></li>
        <li><a href="register.jsp">User Section</a></li>
    </ul>
</body>
</html>
```

2. government_users.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>

<style>
.container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
}

form {
  display: flex;
  flex-direction: column;
  align-items: flex-start; /* Align items to the left */
}

.input-group {
  display: flex;
  flex-direction: column;
  margin-bottom: 10px; /* Added margin-bottom to create space between
input groups */
}

.label {
  margin-bottom: 5px; /* Added margin-bottom to create space between
labels and inputs */
}

.text-input {
  padding: 5px;
}

input[type="submit"] {
  background-color: green;
  color: white;
  border-radius: 10px;
  padding: 10px 20px;
  border: none;
  cursor: pointer;
  margin-top: 10px;
```

```
}

p {
  margin-top: 10px;
}
</style>

</head>
<body>
<div class="container">
  <h1>Railway Crossing</h1>
  <h2>Admin Register</h2>
  <form action="GovernmentRegisterServlet" method="post">
    <div class="input-group">
      <label for="name" class="label">Name:</label>
      <input type="text" id="name" name="name" class="text-input"
required>
    </div>

    <div class="input-group">
      <label for="email" class="label">Email:</label>
      <input type="email" id="email" name="email" class="text-input"
required>
    </div>

    <div class="input-group">
      <label for="password" class="label">Password:</label>
      <input type="password" id="password" name="password"
class="text-input" required>
    </div>

    <input type="submit" value="Register">
  </form>

  <p>Already have an account? <a href="government_login.jsp">Sign
in</a></p>
</div>
```

```
</body>
</html>
```

3. government_login.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<style>
.container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
}

form {
  display: flex;
  flex-direction: column;
  align-items: flex-start; /* Align items to the left */
}

.input-group {
  display: flex;
  flex-direction: column;
  margin-bottom: 10px; /* Added margin-bottom to create space between
input groups */
}

.label {
  margin-bottom: 5px; /* Added margin-bottom to create space between labels and inputs */
}

.text-input {
  padding: 5px;
}

input[type="submit"] {
  background-color: green;
  color: white;
  border-radius: 10px;
  padding: 10px 20px;
  border: none;
  cursor: pointer;
  margin-top: 10px;
}

p {
  margin-top: 10px;
}
</style>

</head>

<body>
<div class="container">
<h1>Railway Crossing</h1>
<h2>Admin Login</h2>
    <form action="GovernmentLoginServlet" method="post">
    <div class="input-group">
        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required><br><br>
        </div>

        <div class="input-group">
        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br><br>
        </div>
        <input type="submit" value="Login">
    </form>
    <p>Don't have Account?<a href="government_users.jsp">Create New Account</a></p>
    </div>
</body>
</html>
```

4. add_railway.html:

```
<!DOCTYPE html>
<html>
<head>
    <title>Add Railway Crossing</title>

</head>
<body>
    <h1>Add Railway Crossing</h1>
    <form action="RailwayCrossingServlet" method="POST">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required>
        <br><br>

        <label for="address">Address:</label>
        <input type="text" id="address" name="address" required>
        <br><br>

        <label for="landmark">Landmark:</label>
        <input type="text" id="landmark" name="landmark" required>
        <br><br>

        <label for="schedules">Train Schedules:</label>
        <textarea id="schedules" name="schedules" required></textarea>
        <br><br>

        <label for="person">Person in Charge:</label>
        <input type="text" id="person" name="person" required>
        <br><br>

        <label for="status">Status:</label>
        <select id="status" name="status" required>
            <option value="Open">Open</option>
            <option value="Closed">Closed</option>
        </select>
        <br><br>

        <input type="submit" value="Add Railway Crossing">
    </form>
</body>
</html>
```

5. update_railway.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>Update Railway Crossing</h1>
    <h1>Update Railway Crossing</h1>

    <%
    // Retrieve the railway crossing details from the request parameters
    int id = Integer.parseInt(request.getParameter("id"));
    String name = request.getParameter("name");
    String address = request.getParameter("address");
    String landmark = request.getParameter("landmark");
    String schedules = request.getParameter("schedules");
    String person = request.getParameter("person");
    String status = request.getParameter("status");
    %>

    <form action="UpdateRailwayCrossingServlet" method="post">
        <input type="hidden" name="id" value="<%= id %>">
        <label for="name">Name:</label>
        <input type="text" name="name" id="name" value="<%= name %>">
        <br>
        <label for="address">Address:</label>
        <input type="text" name="address" id="address" value="<%= address
%>">
        <br>
        <label for="landmark">Landmark:</label>
        <input type="text" name="landmark" id="landmark" value="<%=
landmark %>">
        <br>
        <label for="schedules">Schedules:</label>
        <input type="text" name="schedules" id="schedules" value="<%=
schedules %>">
        <br>
        <label for="person">Person:</label>
        <input type="text" name="person" id="person" value="<%= person
%>">
        <br>
        <label for="status">Status:</label>
        <input type="text" name="status" id="status" value="<%= status %>">
        <br>
        <input type="submit" value="Update">
    </form>
</body>
</html>
```

6. login.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<style>
.container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
}

form {
  display: flex;
  flex-direction: column;
  align-items: flex-start; /* Align items to the left */
}

.input-group {
  display: flex;
  flex-direction: column;
  margin-bottom: 10px; /* Added margin-bottom to create space between
input groups */
}

.label {
  margin-bottom: 5px; /* Added margin-bottom to create space between
labels and inputs */
}
```

```
.text-input {
  padding: 5px;
}

input[type="submit"] {
  background-color: green;
  color: white;
  border-radius: 10px;
  padding: 10px 20px;
  border: none;
  cursor: pointer;
  margin-top: 10px;
}

p {
  margin-top: 10px;
}
</style>

</head>
<body>
<div class="container">
<h1>Railway Crossing</h1>
<h2>User Login</h2>
    <form method="post" action="login">
    <div class="input-group">
        <label for="email">Enter Email:</label>
        <input type="email" id="email" name="email" required><br><br>
        </div>

        <div class="input-group">
        <label for="password">Enter Password:</label>
        <input type="password" id="password" name="password"
required><br><br>
        </div>
        <input type="submit" value="Login">
    </form>
    <br>
    <p>Don't have an account?  <a href="register.jsp">Create New
Account</a></p>
    </div>
</body>
</html>
```

7. register.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<style>
.container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  height: 100vh;
}

form {
  display: flex;
  flex-direction: column;
  align-items: flex-start; /* Align items to the left */
}

.input-group {
  display: flex;
  flex-direction: column;
  margin-bottom: 10px; /* Added margin-bottom to create space between
input groups */
}

.label {
  margin-bottom: 5px; /* Added margin-bottom to create space between
labels and inputs */
}

.text-input {
  padding: 5px;
}

input[type="submit"] {
  background-color: green;
  color: white;
  border-radius: 10px;
  padding: 10px 20px;
  border: none;
  cursor: pointer;
  margin-top: 10px;
}

p {
  margin-top: 10px;
}
</style>
</head>
<body>
<div class="container">
<h1>Railway Crossing</h1>
<h2>User Register</h2>
    <form method="post" action="register">
    <div class="input-group">
        <label for="name">Enter Name:</label>
        <input type="text" id="name" name="name" required><br><br>
        </div>
    <div class="input-group">
    <label for="email">Enter Email:</label>
    <input type="email" id="email" name="email" required><br><br>
    </div>

    <div class="input-group">
    <label for="password">Enter Password:</label>
    <input type="password" id="password" name="password"
required><br><br>
    </div>
    <input type="submit" value="Register">
  </form>
  <br>
  <p>Already have an account? <a href="login.jsp">Login</a></p>
  </div>
</body>
</html>
```

8. search_crossing.jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Search Crossing</title>
  <style>
    form {
      margin-bottom: 20px;
    }
    table {
      border-collapse: collapse;
      width: 100%;
    }
```

```
        th, td {
            padding: 8px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
        td:not(:last-child) {
            padding-right: 10px;
        }
    </style>
    <script>
        function searchCrossing() {
            var keyword = document.getElementById("keyword").value;
            window.location.href = "SearchRailwayCrossingServlet?keyword="
+ encodeURIComponent(keyword);
        }
    </script>
</head>
<body>
    <h1>Search Crossing</h1>

    <form>
        <input type="text" id="keyword" placeholder="Enter a keyword">
        <button type="button" onclick="searchCrossing()">Search</button>
    </form>

    <table id="resultTable" style="display:none;">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Address</th>
            <th>Landmark</th>
            <th>Schedules</th>
            <th>Person</th>
            <th>Status</th>
            <th>Action</th>
```

```
  </tr>
</table>

<script>
  function showTable(results) {
    var table = document.getElementById("resultTable");
    table.style.display = "table";

    // Clear existing rows
    while (table.rows.length > 1) {
      table.deleteRow(table.rows.length - 1);
    }

    // Add search results to the table
    for (var i = 0; i < results.length; i++) {
      var crossing = results[i];
      var row = table.insertRow();

      var idCell = row.insertCell();
      idCell.textContent = crossing.id;

      var nameCell = row.insertCell();
      nameCell.textContent = crossing.name;

      var addressCell = row.insertCell();
      addressCell.textContent = crossing.address;

      var landmarkCell = row.insertCell();
      landmarkCell.textContent = crossing.landmark;

      var schedulesCell = row.insertCell();
      schedulesCell.textContent = crossing.schedules;

      var personCell = row.insertCell();
      personCell.textContent = crossing.person;

      var statusCell = row.insertCell();
```

```
                statusCell.textContent = crossing.status;

                var actionCell = row.insertCell();
                actionCell.innerHTML = "<a href='update_railway.jsp?id=" +
crossing.id + "'>Update</a> | <a href='DeleteRailwayCrossingServlet?id="
+ crossing.id + "'>Delete</a>";
            }
        }
    </script>
</body>
</html>
```

9. GovernmentRegisterServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.annotation.WebServlet;
@WebServlet(value = "/GovernmentRegisterServlet")

public class GovernmentRegisterServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";
}

    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        String name = request.getParameter("name");
        String email = request.getParameter("email");
        String password = request.getParameter("password");

        try {
          Class.forName("com.mysql.jdbc.Driver");
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            String sql = "INSERT INTO admin (name, email, password)
VALUES (?, ?, ?)";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, name);
            statement.setString(2, email);
            statement.setString(3, password);

            int rowsInserted = statement.executeUpdate();
            statement.close();
            conn.close();

            if (rowsInserted > 0) {
                response.sendRedirect("government_login.jsp");
            } else {
                response.getWriter().println("Registration failed. Please try
again.");
            }
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

10.   GovernmentLoginServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

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 javax.servlet.annotation.WebServlet;
@WebServlet(value = "/GovernmentLoginServlet")

public class GovernmentLoginServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";



    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        String email = request.getParameter("email");
        String password = request.getParameter("password");

        try {
          Class.forName("com.mysql.jdbc.Driver");
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            String sql = "SELECT * FROM admin WHERE email = ? AND
password = ?";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, email);
            statement.setString(2, password);

            ResultSet resultSet = statement.executeQuery();

            if (resultSet.next()) {
                HttpSession session = request.getSession();
                session.setAttribute("email", email);
                response.sendRedirect("GetRailwayDetailsServlet");
            } else {
                response.getWriter().println("Invalid email or password. Please
try again.");
            }

            resultSet.close();
            statement.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

11. UpdateRailwayCrossingServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

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

    // JDBC database URL, username, and password
    private static final String DB_URL = "jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    // JDBC driver class name
    private static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";

    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        int id = Integer.parseInt(request.getParameter("id"));
        String name = request.getParameter("name");
        String address = request.getParameter("address");
        String landmark = request.getParameter("landmark");
        String schedules = request.getParameter("schedules");
        String person = request.getParameter("person");
        String status = request.getParameter("status");

        Connection conn = null;
        PreparedStatement statement = null;
        try {
            // Register the JDBC driver
            Class.forName(JDBC_DRIVER);

            // Create a connection to the database
            conn = DriverManager.getConnection(DB_URL, DB_USERNAME,
DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "UPDATE railway_crossings SET name=?, address=?,
landmark=?, schedules=?, person=?, status=? WHERE id=?";
            statement = conn.prepareStatement(sql);
            statement.setString(1, name);
            statement.setString(2, address);
            statement.setString(3, landmark);
            statement.setString(4, schedules);
            statement.setString(5, person);
            statement.setString(6, status);
            statement.setInt(7, id);

            // Execute the statement
            int rowsUpdated = statement.executeUpdate();

            if (rowsUpdated > 0) {
                // Railway crossing updated successfully
                response.sendRedirect("GetRailwayDetailsServlet");
            } else {
                // Failed to update railway crossing
                response.getWriter().println("Failed to update railway crossing. Please
try again.");
            }
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
            response.getWriter().println("JDBC driver not found.");
        } catch (SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        } finally {
            // Close the resources
            if (statement != null) {
                try {
                    statement.close();
                } catch (SQLException e) {
                    e.printStackTrace();
                }
            }
            if (conn != null) {
                try {
                    conn.close();
                } catch (SQLException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}
```

12. GetRailwayDetailsServlet.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

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

```
@WebServlet("/GetRailwayDetailsServlet")
public class GetRailwayDetailsServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        try {
            // Create a connection to the database
            Class.forName("com.mysql.jdbc.Driver");
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "SELECT * FROM railway_crossings";
            PreparedStatement statement = conn.prepareStatement(sql);



            // Execute the query
            ResultSet resultSet = statement.executeQuery();

            // Generate HTML output
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Admin Home Page</title>");
        out.println("<style>");
        out.println(".button-container {");
        out.println("    display: flex;");
        out.println("    gap: 10px;");
        out.println("}");
        out.println(".logout-button {");
        out.println("    margin-left: auto;");
        out.println("}");
        out.println("table {");
        out.println("    border-collapse: collapse;");
        out.println("    width: 100%;");
        out.println("}");
        out.println("th, td {");
        out.println("    padding: 8px;");
        out.println("    text-align: left;");
        out.println("}");
        out.println("th {");
        out.println("    background-color: #f2f2f2;");
        out.println("}");
        out.println("td:not(:last-child) {");
        out.println("    padding-right: 20px;");
        out.println("}");
        out.println("</style>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Admin Home Page</h1>");

        // Get the total number of railway crossings
        int totalRailwayCrossings = 0;
        PreparedStatement countStatement =
conn.prepareStatement("SELECT COUNT(*) FROM railway_crossings");
        ResultSet countResultSet = countStatement.executeQuery();
        if (countResultSet.next()) {
            totalRailwayCrossings = countResultSet.getInt(1);
        }
        out.println("<p>Railway Crossings: [" + totalRailwayCrossings +
"]</p>");
```

```
        out.println("<div class=\"button-container\">");
        out.println("    <form action=\"home.jsp\" method=\"get\">");
        out.println("        <button type=\"submit\">Home</button>");
        out.println("    </form>");
        out.println("    <form action=\"add_railway.html\" method=\"get\">");
        out.println("        <button type=\"submit\">Add Railway Crossing</button>");
        out.println("    </form>");
        out.println("    <form action=\"search_crossing.jsp\" method=\"get\">");
        out.println("        <button type=\"submit\">Search Crossing</button>");
        out.println("    </form>");
        out.println("    <form action=\"government_login.jsp\" method=\"get\" class=\"logout-button\">");
        out.println("        <button type=\"submit\">Logout</button>");
        out.println("    </form>");
        out.println("</div>");

        out.println("<table>");

out.println("<tr><th>ID</th><th>Name</th><th>Address</th><th>Landmark</th><th>Schedules</th><th>Person</th><th>Status</th><th>Action</th</tr>");

        // Check if there are any railway crossings
        if (!resultSet.next()) {
            out.println("<tr><td colspan=\"7\">No railway crossings found</td></tr>");
        } else {
            do {
                int id = resultSet.getInt("id");
                String name = resultSet.getString("name");
                String address = resultSet.getString("address");
                String landmark = resultSet.getString("landmark");
                String schedules = resultSet.getString("schedules");
```

```
                    String person = resultSet.getString("person");
                    String status = resultSet.getString("status");

                    out.println("<tr>");
                    out.println("<td>" + id + "</td>");
                    out.println("<td>" + name + "</td>");
                    out.println("<td>" + address + "</td>");
                    out.println("<td>" + landmark + "</td>");
                    out.println("<td>" + schedules + "</td>");
                    out.println("<td>" + person + "</td>");
                    out.println("<td>" + status + "</td>");
                    out.println("<td><a href='update_railway.jsp?id=" + id +
"'>Update</a> | <a href='DeleteRailwayCrossingServlet?id=" + id +
"'>Delete</a></td>");
                    out.println("</tr>");
                } while (resultSet.next());
            }

            out.println("</table>");
            out.println("</body>");
            out.println("</html>");

            // Close the database connections
            countStatement.close();
            statement.close();
            conn.close();
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            out.println("Database error: " + e.getMessage());
        }
    }
}
```

13. DeleteRailwayCrossingServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

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

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        int id = Integer.parseInt(request.getParameter("id"));

        try {
            // Register the JDBC driver
            Class.forName("com.mysql.jdbc.Driver");

            // Create a connection to the database
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "DELETE FROM railway_crossings WHERE id = ?";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setInt(1, id);

            // Execute the statement
            int rowsDeleted = statement.executeUpdate();
            statement.close();
            conn.close();

            if (rowsDeleted > 0) {
                // Railway crossing deleted successfully
                response.sendRedirect("GetRailwayDetailsServlet");
            } else {
                // Failed to delete railway crossing
                response.getWriter().println("Failed to delete railway crossing.
Please try again.");
            }
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

14. LoginServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

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 javax.servlet.annotation.WebServlet;
@WebServlet(value = "/login")

public class LoginServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        String email = request.getParameter("email");
        String password = request.getParameter("password");

        try {
            // Load the MySQL JDBC driver
            Class.forName("com.mysql.jdbc.Driver");

            // Create a connection to the database
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "SELECT * FROM users WHERE email = ? AND
password = ?";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, email);
            statement.setString(2, password);

            // Execute the statement
            ResultSet resultSet = statement.executeQuery();

            if (resultSet.next()) {
                // Login successful, create a session for the user
                HttpSession session = request.getSession();
                session.setAttribute("email", email);
                response.sendRedirect("DashboardServlet");
            } else {
                // Login failed, display an error message
                response.getWriter().println("Invalid email or password. Please
try again.");
            }

            resultSet.close();
            statement.close();
            conn.close();
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
            response.getWriter().println("MySQL JDBC driver not found.");
        } catch (SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

15. RegisterServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.annotation.WebServlet;
@WebServlet(value = "/register")

public class RegisterServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        String name = request.getParameter("name");
        String email = request.getParameter("email");
        String password = request.getParameter("password");

        try {
            // Load the MySQL JDBC driver
            Class.forName("com.mysql.jdbc.Driver");

            // Create a connection to the database
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "INSERT INTO users (name, email, password)
VALUES (?, ?, ?)";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, name);
            statement.setString(2, email);
            statement.setString(3, password);
            // Execute the statement
            int rowsInserted = statement.executeUpdate();
            statement.close();
            conn.close();

            if (rowsInserted > 0) {
                // Registration successful, redirect to login page
                response.sendRedirect("login.jsp");
            } else {
                // Registration failed, display an error message
                response.getWriter().println("Registration failed. Please try
again.");
            }
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
            response.getWriter().println("MySQL JDBC driver not found.");
        } catch (SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

16. RailwayCrossingServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

@WebServlet(value = "/RailwayCrossingServlet")
public class RailwayCrossingServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        String name = request.getParameter("name");
        String address = request.getParameter("address");
        String landmark = request.getParameter("landmark");
        String schedules = request.getParameter("schedules");
        String person = request.getParameter("person");
        String status = request.getParameter("status");

        try {
            // Register JDBC driver
            Class.forName("com.mysql.jdbc.Driver");

            // Create a connection to the database
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "INSERT INTO railway_crossings (name, address,
landmark, schedules, person, status) " +
                    "VALUES (?, ?, ?, ?, ?, ?)";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, name);
            statement.setString(2, address);
            statement.setString(3, landmark);
            statement.setString(4, schedules);
            statement.setString(5, person);
            statement.setString(6, status);

            // Execute the statement
            int rowsInserted = statement.executeUpdate();
            statement.close();
            conn.close();

            if (rowsInserted > 0) {
                // Railway crossing added successfully
                response.sendRedirect("GetRailwayDetailsServlet");
            } else {
                // Failed to add railway crossing
                response.getWriter().println("Failed to add railway crossing.
Please try again.");
            }
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            response.getWriter().println("Database error: " + e.getMessage());
        }
    }
}
```

17. DashboardServlet.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.annotation.WebServlet;

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

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    // JDBC driver and connection variables
    private static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    private Connection conn;

    @Override
    public void init() throws ServletException {
        super.init();

        try {
            // Register JDBC driver
            Class.forName(JDBC_DRIVER);

            // Open a connection to the database
            conn = DriverManager.getConnection(DB_URL, DB_USERNAME,
DB_PASSWORD);
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            throw new ServletException("Database connection error: " +
e.getMessage());
        }
    }
    @Override
    public void destroy() {
        super.destroy();

        try {
            // Close the database connection
            if (conn != null && !conn.isClosed()) {
                conn.close();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        try {
            // Retrieve favorite crossings from session
            List<Integer> favoriteCrossings = (List<Integer>)
request.getSession().getAttribute("favoriteCrossings");

            // Prepare the SQL statement
            String searchName = request.getParameter("searchName");
            String sql = "SELECT * FROM railway_crossings";
            if (searchName != null && !searchName.isEmpty()) {
                sql += " WHERE name LIKE '%" + searchName + "%'";
            }
            PreparedStatement statement = conn.prepareStatement(sql);

            // Execute the query
            ResultSet resultSet = statement.executeQuery();
            // Generate HTML output
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Railway Crossings</title>");

            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Railway Crossings</h1>");

            // Search form
            out.println("<h2>Search Railway Crossing</h2>");
            out.println("<form method=\"GET\" action=\"DashboardServlet\">");
            out.println("<input type=\"text\" name=\"searchName\"
placeholder=\"Enter name\">");
            out.println("<input type=\"submit\" value=\"Search\">");
            out.println("</form>");

            out.println("<h2>Added Railway Crossings</h2>");

            while (resultSet.next()) {
                int id = resultSet.getInt("id");
                String name = resultSet.getString("name");
                String address = resultSet.getString("address");
                String landmark = resultSet.getString("landmark");
                String schedules = resultSet.getString("schedules");
                String person = resultSet.getString("person");
                String status = resultSet.getString("status");

                out.println("<h3>Railway Crossing ID: " + id + "</h3>");
                out.println("<p><strong>Name:</strong> " + name + "</p>");
                out.println("<p><strong>Address:</strong> " + address + "</p>");
                out.println("<p><strong>Landmark:</strong> " + landmark +
"</p>");
                out.println("<p><strong>Schedules:</strong> " + schedules +
"</p>");
                out.println("<p><strong>Person:</strong> " + person + "</p>");
                out.println("<p><strong>Status:</strong> " + status + "</p>");
                if (favoriteCrossings != null && favoriteCrossings.contains(id)) {
                    out.println("<p><a
href=\"FavoriteCrossingsServlet?removeFavorite=" + id + "\">Remove
Favorite</a></p>");
                } else {
                    out.println("<p><a
href=\"FavoriteCrossingsServlet?markFavorite=" + id + "\">Mark
Favorite</a></p>");
                }
                out.println("<hr>"); // Add a horizontal line between each railway
crossing
            }

            // Option to see favorite crossings
            out.println("<h2>Favorite Railway Crossings</h2>");
            out.println("<form method=\"GET\" action=\"FavoriteListServlet\">");
            out.println("<input type=\"submit\" value=\"View Favorite
Crossings\">");
            out.println("</form>");

            out.println("</body>");
            out.println("</html>");

            statement.close();
        } catch (SQLException e) {
            e.printStackTrace();
            out.println("Database error: " + e.getMessage());
        }
    }
}
```

18. FavoriteCrossingsServlet.java:

```
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.annotation.WebServlet;

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

    // JDBC database URL, username, and password
    private static final String DB_URL = "jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";

    // JDBC driver and connection variables
    private static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    private Connection conn;

    @Override
    public void init() throws ServletException {
        super.init();

        try {
            // Register JDBC driver
            Class.forName(JDBC_DRIVER);

            // Open a connection to the database
            conn = DriverManager.getConnection(DB_URL, DB_USERNAME,
DB_PASSWORD);
```

```
        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            throw new ServletException("Database connection error: " +
e.getMessage());
        }
    }

    @Override
    public void destroy() {
        super.destroy();

        try {
            // Close the database connection
            if (conn != null && !conn.isClosed()) {
                conn.close();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        String markFavoriteParam = request.getParameter("markFavorite");
        String removeFavoriteParam = request.getParameter("removeFavorite");

        if (markFavoriteParam != null && !markFavoriteParam.isEmpty()) {
            int crossingId = Integer.parseInt(markFavoriteParam);
            markFavorite(crossingId, request);
        } else if (removeFavoriteParam != null && !removeFavoriteParam.isEmpty())
{
            int crossingId = Integer.parseInt(removeFavoriteParam);
            removeFavorite(crossingId, request);
        }

        response.sendRedirect(request.getContextPath() + "/DashboardServlet");
    }

    private void markFavorite(int crossingId, HttpServletRequest request) {
        List<Integer> favoriteCrossings = (List<Integer>)
request.getSession().getAttribute("favoriteCrossings");
        if (favoriteCrossings == null) {
            favoriteCrossings = new ArrayList<>();
        }
        favoriteCrossings.add(crossingId);
        request.getSession().setAttribute("favoriteCrossings", favoriteCrossings);
    }

    private void removeFavorite(int crossingId, HttpServletRequest request) {
        List<Integer> favoriteCrossings = (List<Integer>)
request.getSession().getAttribute("favoriteCrossings");
        if (favoriteCrossings != null) {
            favoriteCrossings.remove(Integer.valueOf(crossingId));
            request.getSession().setAttribute("favoriteCrossings", favoriteCrossings);
        }
    }
}
```

19. FavoriteListServlet.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.List;

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

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

  // JDBC database URL, username, and password
  private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
  private static final String DB_USERNAME = "root";
  private static final String DB_PASSWORD = "Simplilearn";

  // JDBC driver and connection variables
  private static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
  private Connection conn;

  @Override
  public void init() throws ServletException {
    super.init();

    try {
      // Register JDBC driver
      Class.forName(JDBC_DRIVER);

      // Open a connection to the database
      conn = DriverManager.getConnection(DB_URL, DB_USERNAME,
DB_PASSWORD);
    } catch (ClassNotFoundException | SQLException e) {
      e.printStackTrace();
      throw new ServletException("Database connection error: " +
e.getMessage());
    }
  }

  @Override
  public void destroy() {
    super.destroy();
```

```
        try {
            // Close the database connection
            if (conn != null && !conn.isClosed()) {
                conn.close();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        List<Integer> favoriteCrossings = (List<Integer>)
request.getSession().getAttribute("favoriteCrossings");

        try {
            // Generate HTML output
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Favorite Crossings</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Favorite Crossings</h1>");

            if (favoriteCrossings != null && !favoriteCrossings.isEmpty()) {
                // Prepare the SQL statement
                String sql = "SELECT * FROM railway_crossings WHERE id IN
(";
                for (int i = 0; i < favoriteCrossings.size(); i++) {
                    sql += favoriteCrossings.get(i);
                    if (i < favoriteCrossings.size() - 1) {
```

```
            sql += ",";
          }
        }
        sql += ")";
        PreparedStatement statement = conn.prepareStatement(sql);

        // Execute the query
        ResultSet resultSet = statement.executeQuery();

        while (resultSet.next()) {
          int id = resultSet.getInt("id");
          String name = resultSet.getString("name");
          String address = resultSet.getString("address");
          String landmark = resultSet.getString("landmark");
          String schedules = resultSet.getString("schedules");
          String person = resultSet.getString("person");
          String status = resultSet.getString("status");

          out.println("<h2>Railway Crossing ID: " + id + "</h2>");
          out.println("<p><strong>Name:</strong> " + name + "</p>");
          out.println("<p><strong>Address:</strong> " + address +
"</p>");
          out.println("<p><strong>Landmark:</strong> " + landmark +
"</p>");
          out.println("<p><strong>Schedules:</strong> " + schedules +
"</p>");
          out.println("<p><strong>Person:</strong> " + person +
"</p>");
          out.println("<p><strong>Status:</strong> " + status + "</p>");
          out.println("<hr>");
        }

        statement.close();
      } else {
        out.println("<p>No favorite crossings found.</p>");
      }
```

```
            out.println("</body>");
            out.println("</html>");
        } catch (SQLException e) {
            e.printStackTrace();
            out.println("Database error: " + e.getMessage());
        }
    }
}
```

20. SearchRailwayCrossingServlet.java:

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

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

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

    // JDBC database URL, username, and password
    private static final String DB_URL =
"jdbc:mysql://localhost:3306/railway";
    private static final String DB_USERNAME = "root";
    private static final String DB_PASSWORD = "Simplilearn";
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String keyword = request.getParameter("keyword");

        try {
            // Create a connection to the database
            Class.forName("com.mysql.jdbc.Driver");
            Connection conn = DriverManager.getConnection(DB_URL,
DB_USERNAME, DB_PASSWORD);

            // Prepare the SQL statement
            String sql = "SELECT * FROM railway_crossings WHERE name
LIKE ?";
            PreparedStatement statement = conn.prepareStatement(sql);
            statement.setString(1, "%" + keyword + "%");

            // Execute the query
            ResultSet resultSet = statement.executeQuery();

            // Create a list to hold the search results
            List<RailwayCrossing> crossings = new ArrayList<>();

            // Iterate over the result set and add each crossing to the list
            while (resultSet.next()) {
                int id = resultSet.getInt("id");
                String name = resultSet.getString("name");
                String address = resultSet.getString("address");
                String landmark = resultSet.getString("landmark");
                String schedules = resultSet.getString("schedules");
                String person = resultSet.getString("person");
                String status = resultSet.getString("status");
                RailwayCrossing crossing = new RailwayCrossing(id, name,
address, landmark, schedules, person, status);
                crossings.add(crossing);
            }

            // Close the database connections
            statement.close();
            conn.close();

            // Display the search results
            out.println("<html>");
            out.println("<head><title>Search Results</title>");
            out.println("<style>");
            out.println("table {");
            out.println("    border-collapse: collapse;");
            out.println("    width: 100%;");
            out.println("}");
            out.println("th, td {");
            out.println("    padding: 8px;");
            out.println("    text-align: left;");
            out.println("}");
            out.println("th {");
            out.println("    background-color: #f2f2f2;");
            out.println("}");
            out.println("td:not(:last-child) {");
            out.println("    padding-right: 20px;");
            out.println("}");
            out.println("</style>");
            out.println("</style>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Search Results</h1>");
            out.println("<table>");

out.println("<tr><th>ID</th><th>Name</th><th>Address</th><th>Landmar
k</th><th>Schedules</th><th>Person</th><th>Status</th><th>Action</th
</tr>");
```

```
            // Iterate over the search results and display them in the table
            for (RailwayCrossing crossing : crossings) {
                    out.println("<tr>");
                out.println("<td>" + crossing.getId() + "</td>");
                out.println("<td>" + crossing.getName() + "</td>");
                out.println("<td>" + crossing.getAddress() + "</td>");
                out.println("<td>" + crossing.getLandmark() + "</td>");
                out.println("<td>" + crossing.getSchedules() + "</td>");
                out.println("<td>" + crossing.getPerson() + "</td>");
                out.println("<td>" + crossing.getStatus() + "</td>");
                out.println("<td>");
                out.println("<a href='update_railway.jsp?id=" + crossing.getId() +
"'>Update</a> | ");
                out.println("<a href='DeleteRailwayCrossingServlet?id=" +
crossing.getId() + "'>Delete</a>");
                out.println("</td>");
                out.println("</tr>");
            }

            out.println("</table>");
            out.println("</body>");
            out.println("</html>");


        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace();
            out.println("Database error: " + e.getMessage());
        }
    }
}
```

21. RailwayCrossing.java:

```
public class RailwayCrossing {
    private int id;
    private String name;
    private String address;
    private String landmark;
    private String schedules;
    private String person;
    private String status;

    public RailwayCrossing(int id, String name, String address, String landmark, String schedules, String person, String status) {
        this.id = id;
        this.name = name;
        this.address = address;
        this.landmark = landmark;
        this.schedules = schedules;
        this.person = person;
        this.status = status;
    }

    public int getId() {
        return id;
    }

    public String getName() {
        return name;
    }

    public String getAddress() {
        return address;
    }

    public String getLandmark() {
        return landmark;
    }

    public String getSchedules() {
        return schedules;
    }

    public String getPerson() {
        return person;
    }

    public String getStatus() {
        return status;
    }
}
```

MYSQL COMMANDS

```
create database railway;
use railway;
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    email VARCHAR(50) NOT NULL UNIQUE,
    password VARCHAR(50) NOT NULL
);
select * from users;
use railway;
CREATE TABLE admin (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(50) NOT NULL,
  email VARCHAR(50) NOT NULL,
  password VARCHAR(50) NOT NULL
);
select * from admin;
use railway;
CREATE TABLE railway_crossings (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    address VARCHAR(255) NOT NULL,
    landmark VARCHAR(255),
    schedules VARCHAR(255),
    person VARCHAR(255),
    status ENUM('open', 'closed') DEFAULT 'open'
);
select * from railway_crossings;
```