Railway Crossing Status

Source code:
`DashboardServlet`

```
package aadi;

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("/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 = "rock";
    private static final String DB_PASSWORD = "@aadi123";

    // 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 {
            // 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();

            // Retrieve favorite crossings from session
            List<Integer> favoriteCrossings = (List<Integer>)
request.getSession().getAttribute("favoriteCrossings");
            if (favoriteCrossings == null) {
                favoriteCrossings = new ArrayList<>();
                request.getSession().setAttribute("favoriteCrossings",
favoriteCrossings);
            }

            // 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("<table>");

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

            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("<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>");

                if (favoriteCrossings.contains(id)) {
                    out.println("<td><a href=\"DashboardServlet?removeFavorite=" +
id + "\">Remove Favorite</a></td>");
                } else {
                    out.println("<td><a href=\"DashboardServlet?markFavorite=" +
id + "\">Mark Favorite</a></td>");
                }

                out.println("</tr>");
            }

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

            // Favorite railway crossings
            out.println("<h2>Favorite Railway Crossings</h2>");
            if (favoriteCrossings.isEmpty()) {
                out.println("<p>No favorite crossings found.</p>");
            } else {
                out.println("<table>");

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

                for (int crossingId : favoriteCrossings) {
                    String sqlFavorite = "SELECT * FROM railway_crossings WHERE id
= ?";
                    PreparedStatement statementFavorite =
conn.prepareStatement(sqlFavorite);
                    statementFavorite.setInt(1, crossingId);
                    ResultSet resultSetFavorite =
statementFavorite.executeQuery();

                    while (resultSetFavorite.next()) {
                        int id = resultSetFavorite.getInt("id");
                        String name = resultSetFavorite.getString("name");
                        String address = resultSetFavorite.getString("address");
                        String landmark = resultSetFavorite.getString("landmark");
                        String schedules =
resultSetFavorite.getString("schedules");
                        String person = resultSetFavorite.getString("person");
                        String status = resultSetFavorite.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=\"DashboardServlet?removeFavorite=" + id + "\">Remove Favorite</a></td>");
                        out.println("</tr>");
                    }

                    statementFavorite.close();
                }

                out.println("</table>");
            }

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

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

    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
        String markFavoriteId = request.getParameter("markFavorite");
        String removeFavoriteId = request.getParameter("removeFavorite");

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

        if (favoriteCrossings == null) {
            favoriteCrossings = new ArrayList<>();
            request.getSession().setAttribute("favoriteCrossings",
favoriteCrossings);
        }

        if (markFavoriteId != null && !markFavoriteId.isEmpty()) {
            int crossingId = Integer.parseInt(markFavoriteId);
            if (!favoriteCrossings.contains(crossingId)) {
                favoriteCrossings.add(crossingId);
            }
        } else if (removeFavoriteId != null && !removeFavoriteId.isEmpty()) {
            int crossingId = Integer.parseInt(removeFavoriteId);
            favoriteCrossings.remove((Integer) crossingId);
        }

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

DeleteRailwayCrossingServlet

```
package aadi;
```

```
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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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());
}
}
}
```

FavoriteCrossingsServlet

```
package aadi;

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("/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 = "rock";
    private static final String DB_PASSWORD = "@aadi123";

    // 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();

        // Retrieve favorite crossings from session
```

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

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

        if (favoriteCrossings == null || favoriteCrossings.isEmpty()) {
            out.println("<p>No favorite crossings found.</p>");
        } else {
            try {
                // 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();

                out.println("<table>");

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

                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("<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("</tr>");
                }

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

                statement.close();
            } catch (SQLException e) {
```

```
                e.printStackTrace();
                out.println("Database error: " + e.getMessage());
            }
        }

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

GetRailwayDetailsServlet

```
package aadi;

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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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
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>Railway Crossings</title>");
out.println("</head>");
out.println("<body>");
out.println("<h1>Railway Crossings</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></tr>");
```

```
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");
String favourite = resultSet.getString("favourite");
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>" + favourite + "</td>");

out.println("<td><a href='update_railway.jsp?id=" + id +
"'>Update</a> | <a href='DeleteRailwayCrossingServlet?id=" + id +
"'>Delete</a></td>");
out.println("</tr>");
}
out.println("</table>");
out.println("</body>");
out.println("</html>");
statement.close();
conn.close();
} catch (SQLException e) {
e.printStackTrace();
out.println("Database error: " + e.getMessage());
}
}
}
```

GovernmentLoginServlet

```
package aadi;


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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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("add_railway.html");
} 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());
}
}
}
```

GovernmentRegisterServlet

```
package aadi;



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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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());
}
}
}
```

LoginServlet

```
package aadi;



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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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());
}
}
}
```

RailwayCrossingServlet

```
package aadi;



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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
```

```
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");
String favourite = request.getParameter("favourite");

System.out.println(favourite);
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,favourite) " +
"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);
statement.setString(7, favourite);
// 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());
}
}
}
```

RegisterServlet

```
package aadi;


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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
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());
}
}
}
```

`UpdateRailwayCrossingServlet`

```
package aadi;
```

```
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 = "rock";
private static final String DB_PASSWORD = "@aadi123";
// 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();
}
}
}
}
}
```

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>

<label for="favourite">favourite:</label>
<select id="favourite" name="favourite" required>
<option value="Yes">Yes</option>
<option value="No">No</option>
</select>
<br><br>
```

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

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>
</head>
<body>
<h2>Government User Login</h2>
    <form action="GovernmentLoginServlet" method="post">
        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required><br><br>

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br><br>

        <input type="submit" value="Login">
    </form>
</body>
</html>
```

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>
</head>
<body>
<h1>Govt. User Registration</h1>

    <h2>Government User Registration</h2>
    <form action="GovernmentRegisterServlet" method="post">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required><br><br>

        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required><br><br>

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required><br><br>

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

    </form>
    <p>  <a href="government_login.jsp">sign in</a></p>
```

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

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>Select from below option:</h3>

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

Login.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f5f5f5;
        }

        .container {
            max-width: 400px;
            padding: 20px;
            background-color: #fff;
            border-radius: 5px;
            text-align: center;
        }

        h1 {
            color: #333;
        }

        label {
            display: block;
            margin-bottom: 5px;
            text-align: left;
        }

        input[type="text"],
```

```
        input[type="email"],
        input[type="password"] {
            width: 100%;
            padding: 10px;
            margin-bottom: 15px;
            border: 1px solid #ccc;
            border-radius: 4px;
            box-sizing: border-box;
        }

        input[type="submit"] {
            width: 100%;
            padding: 10px;
            background-color: green;
            border: none;
            color: #fff;
            font-size: 16px;
            border-radius: 10px;
            cursor: pointer;
        }

        p {
            margin-top: 10px;
        }

        a {
            color: blue;
        }
    </style>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<div class="container">
        <h1>Railway Crossing</h1>
        <h4>User Register</h4>

<form method="post" action="login">
<label for="email">Enter Email:</label>
<input type="email" id="email" name="email" required><br><br>
<label for="password">Enter Password:</label>
<input type="password" id="password" name="password"
required><br><br>
<input type="submit" value="Login">
</form>
<br>
<p>Don't have an account? <a href="register.jsp">Create NewAccount</a></p>
 </div>
</body>
</html>
```

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>
.submitted-value {
color: green;
}
</style>
</head>
<title>Register</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f5f5f5;
        }

        .container {
            max-width: 400px;
            padding: 20px;
            background-color: #fff;
            border-radius: 5px;
            text-align: center;
        }

        h1 {
            color: #333;
        }

        label {
            display: block;
            margin-bottom: 5px;
            text-align: left;
        }

        input[type="text"],
        input[type="email"],
        input[type="password"] {
            width: 100%;
            padding: 10px;
            margin-bottom: 15px;
            border: 1px solid #ccc;
            border-radius: 4px;
            box-sizing: border-box;
        }

        input[type="submit"] {
            width: 100%;
            padding: 10px;
            background-color: green;
            border: none;
            color: #fff;
            font-size: 16px;
            border-radius: 10px;
            cursor: pointer;
        }

        p {
            margin-top: 10px;
        }
```

```
        a {
            color: blue;
        }
    </style>
<body>
<div class="container">
        <h1>Railway Crossing</h1>
        <h4>User Register</h4>

<form method="post" action="register">
<label for="name">Enter Name:</label>
<input type="text" id="name" name="name" required><br><br>
<label for="email">Enter Email:</label>
<input type="email" id="email" name="email" required><br><br>
<label for="password">Enter Password:</label>
<input type="password" id="password" name="password"
required><br><br>
<input type="submit" value="Register">
</form>
<br>
<p>Already have an account? <a href="login.jsp">Login</a></p>
 </div>
</body>
</html>
```

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>
<h3>Update Railway Crossing</h3>
<%
// 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>
```

Sql queries

```
CREATE DATABASE railway;
use railway;
CREATE TABLE users (
  id INT PRIMARY KEY AUTO_INCREMENT,
  name VARCHAR(100) NOT NULL,
  email VARCHAR(100) NOT NULL,
  password VARCHAR(100) NOT NULL
);
INSERT INTO users (name, email, password)
VALUES
  ('John Doe', 'john@example.com', 'password123'),
  ('Jane Smith', 'jane@example.com', 'password456'),
  ('aadi', 'aadi@example.com', 'password789');
  select*from railway_crossings;
CREATE TABLE railway_crossings (
  id INT PRIMARY KEY AUTO_INCREMENT,
  name VARCHAR(100) NOT NULL,
  address VARCHAR(200) NOT NULL,
  landmark VARCHAR(100) NOT NULL,
  schedules VARCHAR(100) NOT NULL,
  person VARCHAR(100) NOT NULL,
  status ENUM('Open', 'Closed') NOT NULL
);
INSERT INTO railway_crossings (name, address, landmark, schedules, person,
status)
VALUES
  ('Ishmeet Chowk Crossing', 'Ishmeet Singh Near Party People', 'Ishmeet Singh Near
Party People', '10:33 am', 'David', 'Open'),
  ('Midha Chowk Crossing', 'Midha Crossing', 'Midha Crossing', '10:40 am', 'Robin',
'Open'),
  ('ABC Crossing', 'ABCDEF', 'ABCDEF', '11:40 am', 'John', 'Open');
```

```
CREATE TABLE user_favourite (
  id INT PRIMARY KEY AUTO_INCREMENT,
  user_id INT NOT NULL,
  crossing_id INT NOT NULL,
  FOREIGN KEY (user_id) REFERENCES users(id),
  FOREIGN KEY (crossing_id) REFERENCES railway_crossings(id)
);
INSERT INTO user_favourite (user_id, crossing_id)
VALUES
  (1, 1), -- John Doe favorited Ishmeet Chowk Crossing
  (2, 2); -- Jane Smith favorited Midha Chowk Crossing
CREATE TABLE admin (
  id INT PRIMARY KEY AUTO_INCREMENT,
  email VARCHAR(100) NOT NULL,
  password VARCHAR(100) NOT NULL
);
INSERT INTO admin (email, password)
VALUES
  ('admin@example.com', 'admin123'),('aditya@exmple.com','aditya123');
  SELECT * FROM admin;
```