**1. Requirements**

A web browser (e.g., Google Chrome, Mozilla Firefox, Microsoft Edge).

A text editor (optional, for editing code) like Visual Studio Code, Sublime Text, or Notepad++.

**2. Folder Structure**

Make sure your files are organized as follows:

**scss**

**project-folder**

**index.**html        (Main HTML file)

**styles.css        (**CSS file for styling)

**script.js** (JavaScript file for interactivity)

**3. Steps to Open the Project**

**Option 1:** Open Directly

Navigate to the folder where your project files are saved.

Locate the index.html file.

Double-click on the index.html file.

Your default browser will open and display the project.

**Option 2:** Use a Text Editor with Live Server (Recommended)

Open the project folder in Visual Studio Code (or your preferred editor):

Right-click the folder and select Open with Code (for VS Code users).

Install the Live Server Extension (if not already installed):

Go to the Extensions view (Ctrl+Shift+X in VS Code), search for Live Server, and install it.

Start Live Server:

Right-click on index.html in VS Code and select Open with Live Server.

The project will open in your default browser, running on a local server (e.g., http://127.0.0.1:5500).

**Option 3:** Open with a Browser Using Drag and Drop

Open your preferred web browser.

Drag and drop the index.html file directly into the browser window.

**4. Steps to run JSP Code**:

**1. Set Up the Development Environment**:  
 Open Eclipse.  
 Configure Tomcat in Eclipse:  
 Go to Window > Preferences > Server > Runtime Environments.  
 Click Add, select Apache Tomcat, and specify the installation directory.  
  
**2. Create a New Dynamic Web Project:**  
 In Eclipse, go to File > New > Dynamic Web Project.  
 Enter a project name (e.g., `FleetManagementSystem`).  
 Set the target runtime to the configured Tomcat server.  
 Click Finish.  
  
**3. Add the JSP File**:  
 Locate the `WebContent` folder in your project structure.  
 Right-click on `WebContent` and select New > JSP File.  
 Name the file `fms.jsp` and click Finish.  
 Paste the provided JSP code into the `fms.jsp` file and save it.  
  
**4. Configure the `web.xml` File**:  
 Navigate to the `WEB-INF` folder inside `WebContent`.  
 If the `web.xml` file is missing:  
 Right-click `WEB-INF` and select New > File.  
 Name the file `web.xml` and click Finish.  
 Add the following content to define the welcome file:  
   
**5. Add MySQL JDBC Driver (Optional):** If the code requires database connectivity:  
 Download the MySQL JDBC Driver from the MySQL website.  
 Place the `.jar` file in the `WEB-INF/lib` folder of your project.  
  
**6. Deploy the Project**:  
 Right-click the project in Eclipse and select Run As > Run on Server.  
 Choose the configured Tomcat server and click Finish.  
  
**7. Access the Application**:  
 Open a web browser and navigate to `http://localhost:8080/YourProjectName`.  
 Use one of the predefined usernames and passwords (e.g., `admin1` and `password1`) to log in.

**8. Test the Application**:  
 **Login:** Enter valid credentials to access the dashboard.  
 Dashboard: Explore features like "Manage Vehicles," "Route Optimization," and "Real-Time Monitoring ”.  
 Logout: Click the logout button to return to the login page.  
  
  
  
**Predefined Credentials :**

**Username:** ‘admin1’ , ‘admin2’ , ’ user1’ , ’ user2’

**Password:** ‘password1’ , ‘password2’ , ‘userpass1’ , ‘userpass2’  
  
  
**Troubleshooting**  
**Error 404:** Ensure the `fms.jsp` file is in the `WebContent` folder and the `web.xml` file correctly references it.  
**Login Issues:** Verify that the username and password match one of the predefined pairs.  
Server Errors: Check the Tomcat console for detailed error messages and resolve any configuration issues.  
  
 **By following these steps, you should be able to successfully run and test the provided JSP code.**

1. **Set Up Apache Tomcat in Eclipse:** Open Eclipse. Go to Window > Preferences. Under Server > Runtime Environments, click Add External Servers. Choose Apache Tomcat and follow the steps to link your Tomcat folder.

1. **Create a New Project:** In Eclipse, go to File > New > Dynamic Web Project. Name the project (e.g., Fleet Management), and choose Tomcat as the server. Click Finish.

1. **Add the JSP Code to Your Project:** In your new project, go to Web Content folder (this is where your web pages go). Put your JSP code inside Web Content.

1. **Add MySQL JDBC Driver**: Download the MySQL JDBC Driver from the MySQL website. In Eclipse, go to your project’s WEB-INF/lib folder and add the .jar file you downloaded.

1. **Set Up Your MySQL Database**: Open MySQL and create a database (e.g., fleet management). Create a table called users with columns for username and password.

**Example:**

**SQL**

**CREATE DATABASE fleet management;**

**USE fleet management;**

**CREATE TABLE users (id INT AUTO\_INCREMENT PRIMARY KEY,**

**username VARCHAR (255),**

**password VARCHAR (255));**

**7. Run the Project on Tomcat**: Right-click your project in Eclipse and select Run As > Run on Server. Choose Tomcat as the server. Your browser will open, and you should see the login page.

**8. Test It**: Enter a username and password on the login page. The code will check if the login is correct by querying the database.

**5. Testing the Features**

**Login:**

Enter dummy credentials and click "Login" to access the dashboard.

**Registration:**

Fill out the registration form and observe the redirection to the login page.

**Dashboard:**

Interact with the cards and test the "Logout" button.

**6. Deployment (Optional)**

If you'd like to share the project or make it accessible online: