RailwayCrossingAlgorithm

GitHub Url:

https://github.com/Ankithachainpur/Mphasis-Simplilearn-project.git

1. **index.jsp**:Thisisthehomepageoftheapplication.Itprovideslinksforuserlogin and admin login.
2. **login.jsp**: This JSP file contains the login form for users. It collects the username

and password from the user.

1. **Login.java**(servlet):Thisservletisresponsibleforhandlingtheuserloginprocess.It retrieves the username and password from the request parameters, performs a database query to validate the credentials, and redirects the user to the "userhome.jsp" page if the login is successful. Otherwise, it displays an error message.
2. **Registration.jsp**:ThisJSPfilecontainstheregistrationformforusers.Itcollectsthe username, password, and email from the user.
3. **Register.java** (servlet): This servlet handles the user registration process. It retrieves theusername,password,andemailfromthe request parameters, createsa **Member** object, and uses a **RegisterDao** to insert the user into the database.
4. **RegisterDao.java**: This class is responsible for database operations related to user registration.Ithasamethod **insert**thattakesa**Member**objectasinputandinserts the user into the database.
5. **Member.java**:Thisclassrepresentsauserandcontainstheirusername,password, and email.
6. **Userhome.jsp**:Thisfilerepresentstheuser'shomepage.Itdisplaysalistofrailway crossing information retrieved from a MySQL database. It also provides links to navigate to other pages like search and favorite crossings.
7. **Search.jsp**: This file displays a simple form where users can enter a keyword to searchforrailwaycrossings.Itsubmitstheformdatatothe **searchResult.jsp** page.
8. **SearchResult.jsp**: This file processes the search query entered by the user and performsadatabasequerytoretrievematchingrailwaycrossings.Itdisplaysthe search results in a table format.
9. **Favorite.jsp**: Thisfile islikely intendedtodisplaytheuser'sfavorite railwaycrossings. However,the codeprovided issimilartothe **SearchResult.jsp**file,soit appearstobe a duplication or a mistake.
10. AddToFavorite.java (Servlet): This servlet is responsible for adding an item to the user's favorites. It establishes adatabase connection, prepares anSQLstatement to insertthefavoriteitem,executesthestatement,andthenredirectstheuserbackto the user home page.
11. RemoveFromFavorite.java (Servlet): It seems that there was an error in copying the code,asthisfileis identicaltoAddToFavorite.java.However,basedonthe name,it is likely intended to remove an item from the user's favorites. You would need to provide the correct code for this functionality.
12. Adminlogin.jsp: This JSP (JavaServer Pages) file is a view that displays search results from the adminhome table. It sets up a data source and performs a query to search forrecordsthatmatchagivenkeywordintheNameorAddresscolumns.Thesearch results are then displayed in an HTML table.
13. AdminLogin.java (Servlet): This servlet handles the login functionality for the admin user. It receives the username and password parameters, establishes a database connection,andexecutesaquerytocheckiftheprovidedcredentialsarevalid.Ifthe login is successful, it forwards the request to the adminhome.jsp page. Otherwise, it displays an error message indicating a failed login attempt.
14. **adminhome.jsp**:ThisJSPfiledisplayssearchresultsforrailwaycrossings.ItusesSQL queries to retrieve data from a MySQL database and displays it in a tabular format.
15. **Addrail.jsp**:ThisJSPfilecontainsaformforaddinganewrailwaycrossing.Itincludes input fields for the name, address, landmark, train schedule, person in charge, and status of the crossing.
16. **RailCross.java**: This Java class defines a **RailCross** object that represents a railway crossing.Ithasprivatefieldsforthecrossing'sattributesandcorrespondinggetters and setters.
17. **RailCrossing.java**:Thisservletreceivesformdata from**Addrail.jsp**andinsertsanew railway crossing into the MySQL database using the **RailCross** object.
18. **Update.jsp**:ThisJSPfiledisplaysaformforupdatinganexistingrailwaycrossing.Itis similar to **Addrail.jsp** but pre-fills the form with the existing crossing's data.
19. **UpdateProcess.jsp**: This JSP file handles the update process by updating the correspondingrecordintheMySQLdatabasebasedontheformdatareceivedfrom **Update.jsp**.
20. **Delete.jsp**:ThisJSPfilehandlesthedeletionofarailwaycrossing recordfromthe MySQL database.