EAD 2 - Course Works

By Lasitha Gunawardena

CW 1 (1 Hour)

Objectives

1. Learning how to create a maven based java project.
2. Learning how to use eclipse IDE.
3. Learning how to manage dependencies based on POM.
4. Learning MVC design patterns.

Task 01

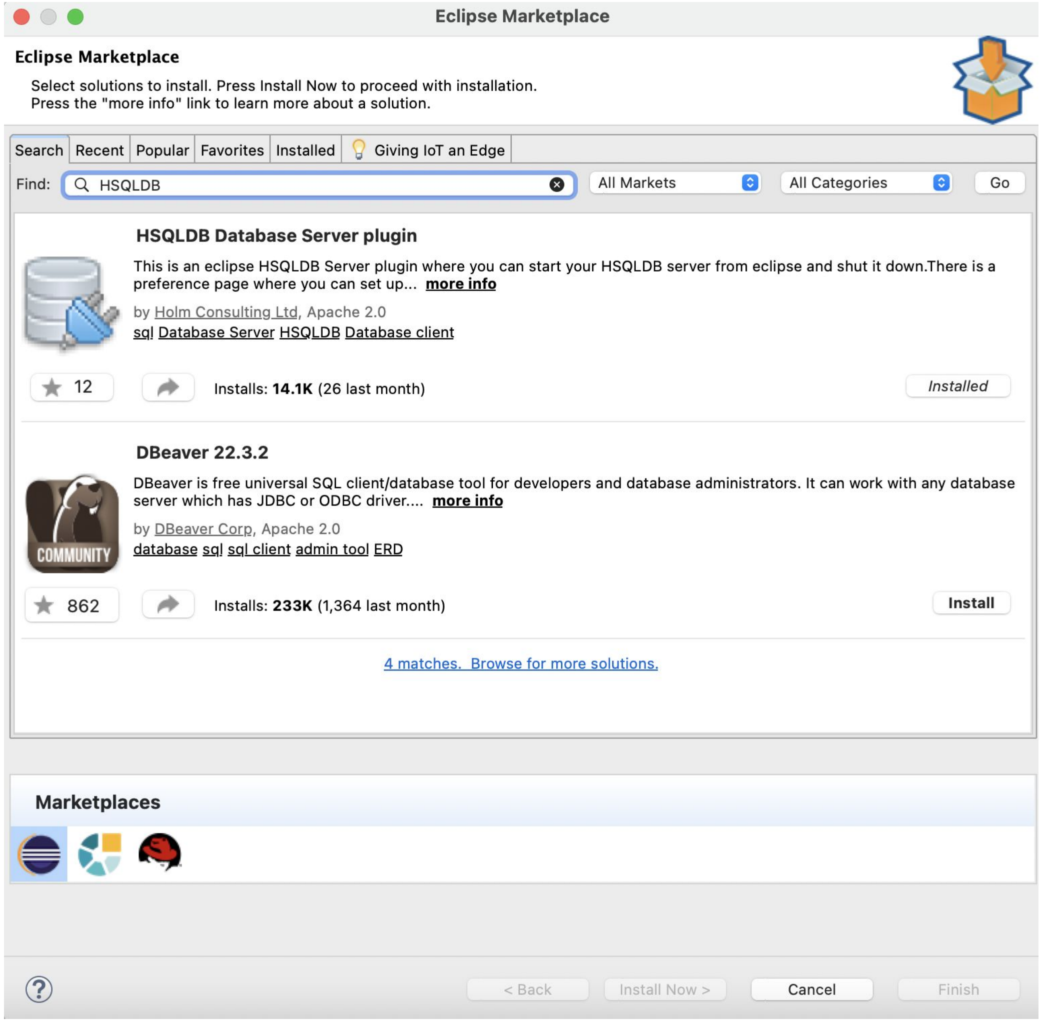
1. Create a maven project from eclipse IDE.Use OpenJDK 17 or higher as the java runtime.
   a. Group id : lk.nibm.ead2
   b. Artifact id: cw1-hrms
2. Add JavaFx and HSQLDB as dependencies for the POM.

```
<<dependency>
         <groupId>org.openjfx</groupId>
         <artifactId>javafx-controls</artifactId>
         <version>0.0.8</version>
</dependency>
<dependency>
         <groupId>org.openjfx</groupId>
         <artifactId>javafx-web</artifactId>
         <version>0.0.8</version>
</dependency>
<dependency>
         <groupId>org.openjfx</groupId>
         <artifactId>javafx-fxml</artifactId>
         <version>0.0.8</version>
</dependency>
<dependency>
    <groupId>org.hsqldb</groupId>
    <artifactId>hsqldb</artifactId>
    <version>2.2.4</version>
</dependency>
```

Task 1

3. Configure HSQLDB Plugin

1. Go to **Eclipse** -> **Help** -> **Eclipse Marketplace**
2. In the search field type HSQLDB and start search
3. Select HSQLDB Database server plugin and install it
4. Once the installation is completed navigate to **Window** -> **Show View** -> **Other**
5. Filter by HSQLDB and open HSQLDB server console
6. In the **HSQLDB server console** click on the Kebab menu in the toolbar then go to preferences
7. In preference add **file store name** (ex: db) . Set the **database name** as **HRMS**. **Username** should be **sa** and keep **password empty**.Click on **Apply and Close**
8. Finally start the server.
9. Go to the console tab and you can see database server is started with the database HRMS.
10. Again navigate to **HSQLDB server console - > Kebab menu -> Show database client**
11. In the HSQLDB database manager enter the sql script given and create the table.



Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: HSQLDB

All Markets

All Categories

HSQLDB Database Server plugin

This is an eclipse HSQLDB Server plugin where you can start your HSQLDB server from eclipse and shut it down.There is a preference page where you can set up... **more info**

by Holm Consulting Ltd, Apache 2.0

sql Database Server HSQLDB Database client

★ 12

Installs: **14.1K** (26 last month)

Installed

DBeaver 22.3.2

DBeaver is free universal SQL client/database tool for developers and database administrators. It can work with any database server which has JDBC or ODBC driver.... **more info**

by DBeaver Corp, Apache 2.0

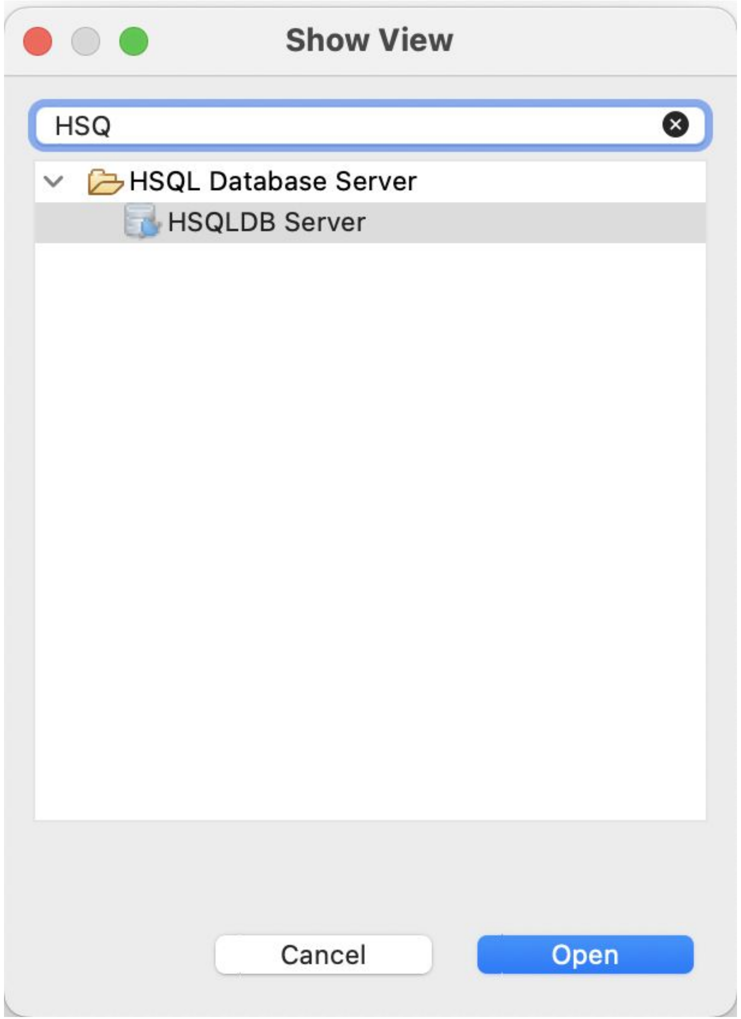
database sql sql client admin tool ERD

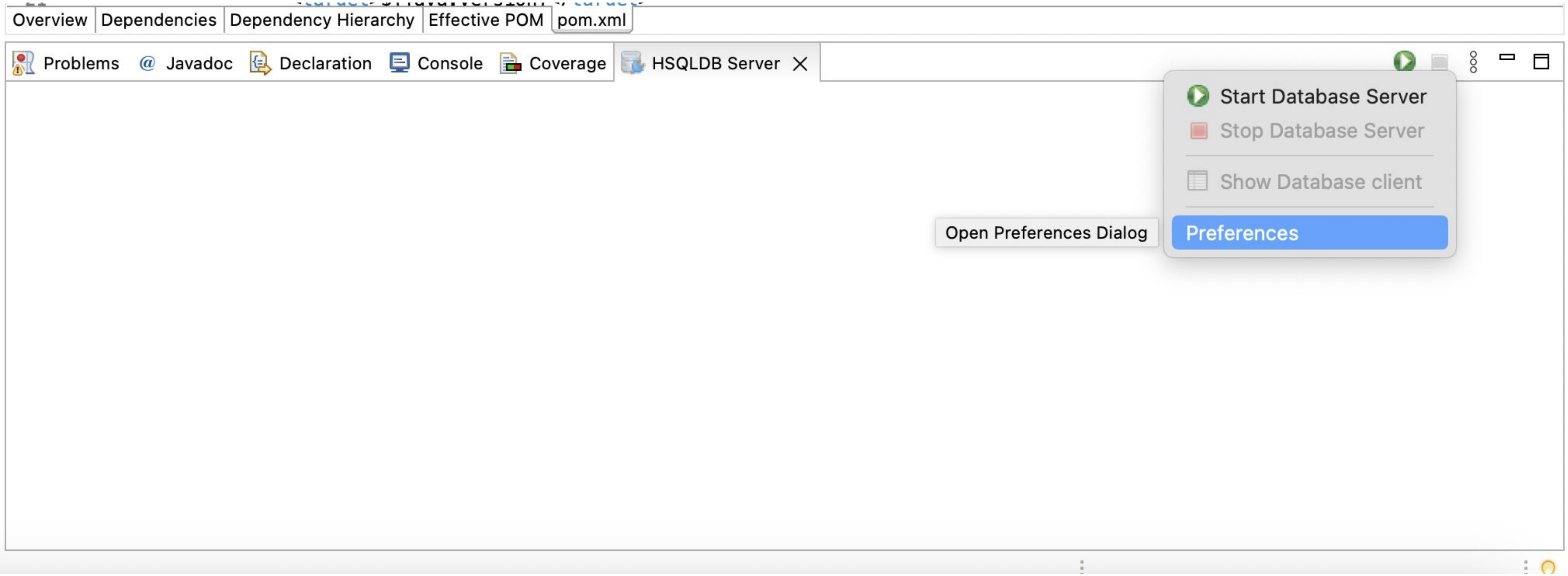
★ 862

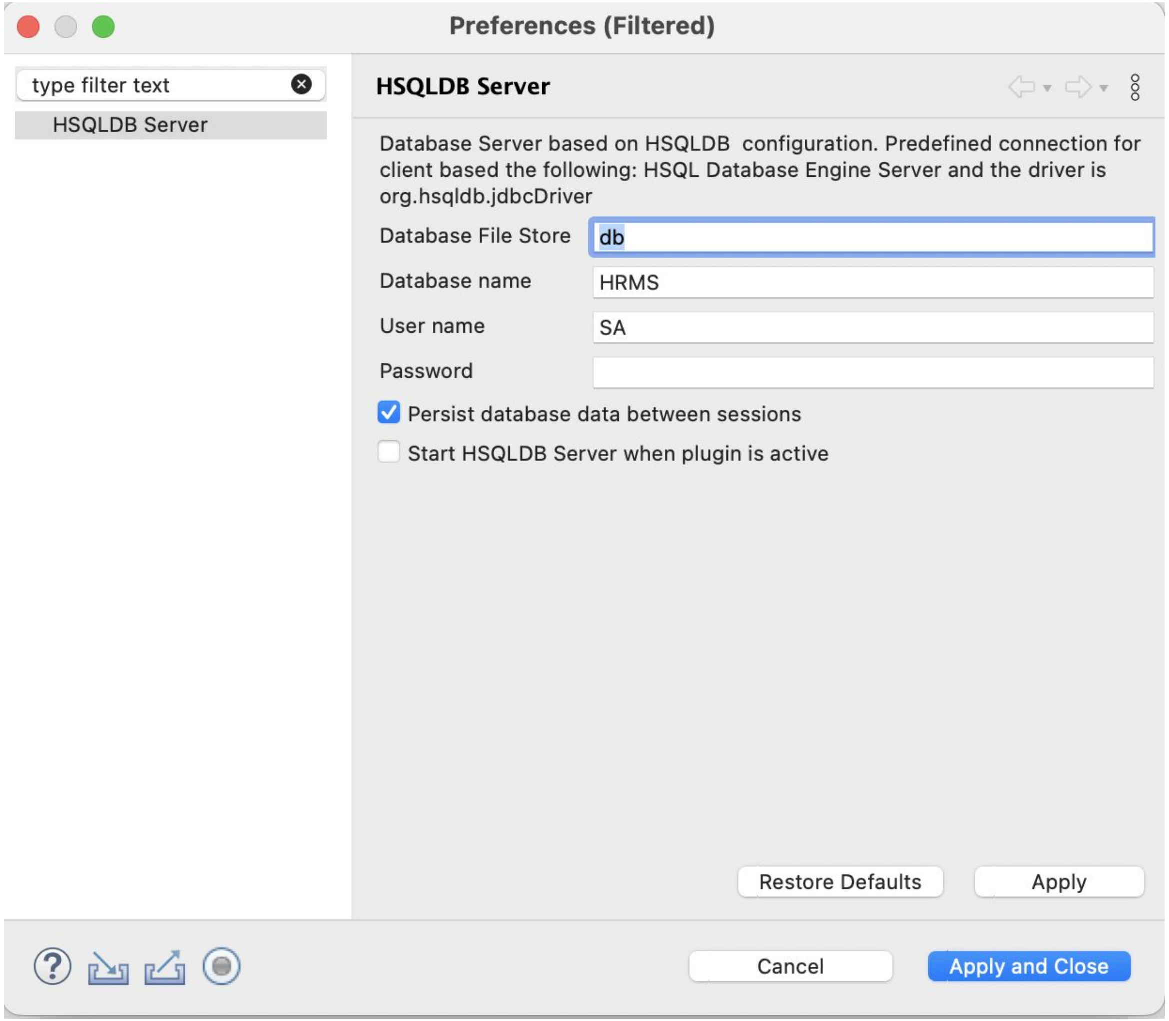
Installs: **233K** (1,364 last month)

4 matches. Browse for more solutions.

Marketplaces


Show View
HSQ
HSQL Database Server
HSQLDB Server
Cancel
Open


Overview
Dependencies
Dependency Hierarchy
Effective POM
pom.xml
Problems
Javadoc
Declaration
Console
Coverage
HSQLDB Server
Start Database Server
Stop Database Server
Show Database client
Open Preferences Dialog
Preferences



HSQLDB Server

Database Server based on HSQLDB configuration. Predefined connection for client based the following: HSQL Database Engine Server and the driver is org.hsqldb.jdbcDriver

Database File Store: db

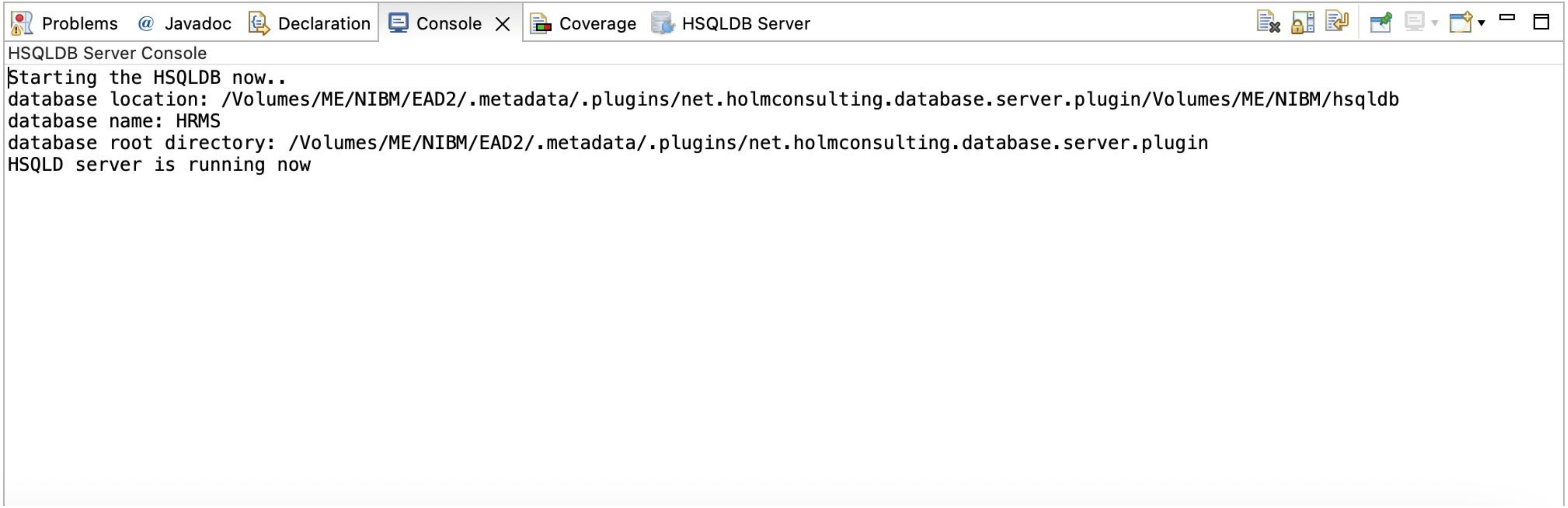
Database name: HRMS

User name: SA

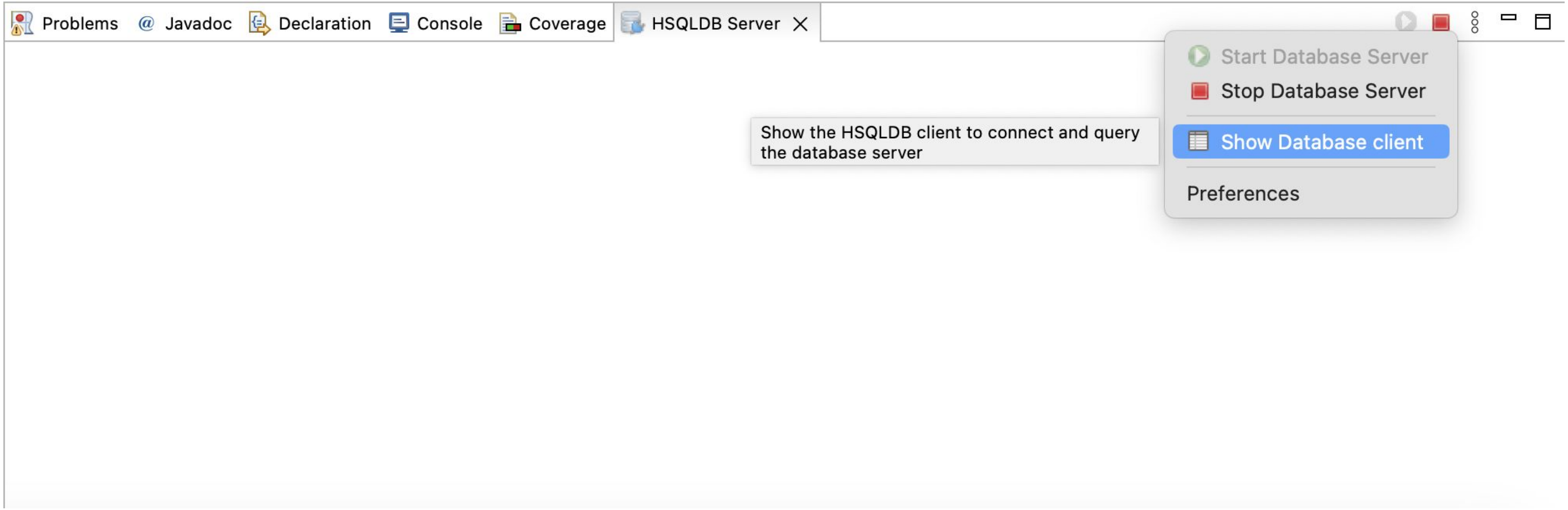
Password:

☑ Persist database data between sessions

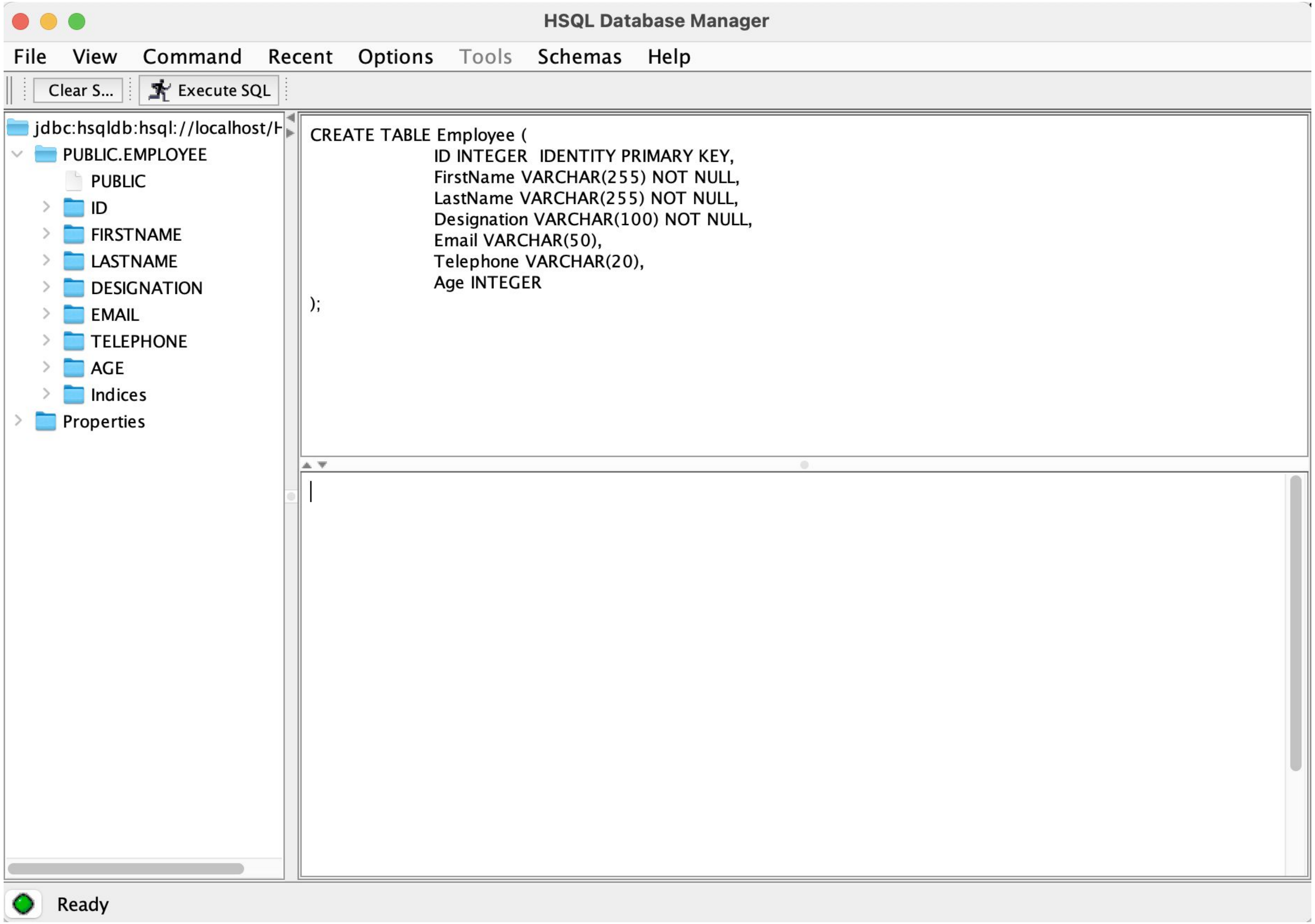
☐ Start HSQLDB Server when plugin is active



```
Starting the HSQLDB now..
database location: /Volumes/ME/NIBM/EAD2/.metadata/.plugins/net.holmconsulting.database.server.plugin/Volumes/ME/NIBM/hsqldb
database name: HRMS
database root directory: /Volumes/ME/NIBM/EAD2/.metadata/.plugins/net.holmconsulting.database.server.plugin
HSQLD server is running now
```



Show the HSQLDB client to connect and query the database server



```
CREATE TABLE Employee (
                ID INTEGER  IDENTITY PRIMARY KEY,
                FirstName VARCHAR(255) NOT NULL,
                LastName VARCHAR(255) NOT NULL,
                Designation VARCHAR(100) NOT NULL,
                Email VARCHAR(50),
                Telephone VARCHAR(20),
                Age INTEGER
);
```

Task 2

1. User the classes provided in this guide line for Task 2.
2. You need to follow up proper package structure mentioned in the class.
3. Some classes have missing implementations.You need to complete them
4. Add view FXML file to **main/java/resources/fxml/Registration.fxml**

Database

```java
package lk.nibm.ead2.hrms.db;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class Database {

        private static final String HSQLDB_CONNECTION_STRING = "jdbc:hsqldb:hsql://localhost/HRMS";
        private static final String HSQLDB_USERNAME = "SA";
        private static final String HSQLDB_PASSWORD = "";

        private static Database database;
        private Connection connection;

        private Database() throws SQLException {
                this.connection = connect();
                System.out.println("HSQLDB database connection established...");
        }

        public static Database getDatabase() throws SQLException {
                if(database == null) {
                        database = new Database();
                }

                return database;
        }
        private Connection connect() throws SQLException {
                DriverManager.registerDriver(new org.hsqldb.jdbcDriver());
                Connection conn = DriverManager.getConnection(HSQLDB_CONNECTION_STRING,HSQLDB_USERNAME,HSQLDB_PASSWORD);
                return conn;
        }

        public PreparedStatement getPreparedStatement(String sql) throws SQLException {
                return this.connection.prepareStatement(sql);
        }

        public void close() throws SQLException {
                this.connection.close();
        }
}
```

Model

```
package lk.nibm.ead2.hrms.model;

public class Employee {
      private Integer id;
      private String firstname;
      private String lastname;
      private String designation;
      private String email;
      private String telephone;
      private Integer age;

        // Create getters/setters by using eclipse
        // Alt + Shift + S , Then press R
        // Alt + Shift + S -> Generate toString() with all attributes
}
```

Service

```
package lk.nibm.ead2.hrms.service;

import java.sql.PreparedStatement;
import java.sql.SQLException;

import lk.nibm.ead2.hrms.db.Database;
import lk.nibm.ead2.hrms.model.Employee;

public class EmployeeService {

       public boolean insert(Employee employee) throws SQLException {
              String sql = "INSERT INTO public.employee(FirstName,LastName,Designation,Email,Telephone,Age)
VALUES(?,?,?,?,?,?)";
              Database db = Database.getDatabase();
              PreparedStatement ps = db.getPreparedStatement(sql);
              ps.setString(1, employee.getFirstname());
              // Implement adding rest of parameters
              ps.close();
              return status > 0;
       }
}
```

Controller

```java
package lk.nibm.ead2.hrms.controller;

import javafx.beans.binding.Bindings;
import javafx.beans.property.SimpleIntegerProperty;
import javafx.beans.property.SimpleStringProperty;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;
import javafx.util.converter.NumberStringConverter;
import lk.nibm.ead2.hrms.model.Employee;
import lk.nibm.ead2.hrms.service.EmployeeService;

public class EmployeeController {
        @FXML
        private TextField firstName;
        @FXML
        private TextField lastName;
        @FXML
        private TextField designation;
        @FXML
        private TextField email;
        @FXML
        private TextField telephone;
        @FXML
        private TextField age;
        @FXML
        private Label message;
        @FXML
        private Button submitButton;

        private SimpleStringProperty firstNameProperty = new SimpleStringProperty();
        private SimpleStringProperty lastNameProperty = new SimpleStringProperty();
        private SimpleStringProperty designationProperty = new SimpleStringProperty();
        private SimpleStringProperty emailProperty = new SimpleStringProperty();
        private SimpleStringProperty telephoneProperty = new SimpleStringProperty();
        private SimpleIntegerProperty ageProperty = new SimpleIntegerProperty();
        private SimpleStringProperty messageProperty = new SimpleStringProperty();

        private EmployeeService service = new EmployeeService();

        @FXML
        private void initialize() {
                this.firstName.textProperty().bindBidirectional(firstNameProperty);
                this.lastName.textProperty().bindBidirectional(lastNameProperty);
                this.designation.textProperty().bindBidirectional(designationProperty);
                this.email.textProperty().bindBidirectional(emailProperty);
                this.telephone.textProperty().bindBidirectional(telephoneProperty);
                this.message.textProperty().bindBidirectional(messageProperty);
                Bindings.bindBidirectional(age.textProperty(), ageProperty,new NumberStringConverter());
        }

        @FXML
        public void handleSubmitButtonAction(ActionEvent event) {
                Employee emp = new Employee();
                emp.setFirstname(firstNameProperty.get());
                emp.setLastname(lastNameProperty.get());
                emp.setDesignation(designationProperty.get());
                emp.setEmail(emailProperty.get());
                emp.setTelephone(telephoneProperty.get());
                emp.setAge(ageProperty.get());

                System.out.println(emp);
                try {
                        if(service.insert(emp)) {
                                messageProperty.setValue("Record inserted succesfuly");
                        }
                } catch (Exception e) {
                        e.printStackTrace();
                        messageProperty.setValue(e.getMessage());
                }
        }

        @FXML
        public void handleClearButtonAction(ActionEvent event) {
                //Implement clear function
        }
}
```

View

```
<?import javafx.scene.layout.GridPane?>
<?import javafx.geometry.Insets?>
<?import javafx.scene.layout.ColumnConstraints?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.text.Font?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.control.PasswordField?>
<?import javafx.scene.control.Button?>

<GridPane xmlns:fx="http://javafx.com/fxml" alignment="center"
          hgap="10" vgap="10"
          fx:controller="lk.nibm.ead2.hrms.controller.EmployeeController">
    <padding><Insets top="40" right="40" bottom="40" left="40"/></padding>
    <columnConstraints>
        <ColumnConstraints minWidth="100" prefWidth="100"
                           maxWidth="Infinity" halignment="RIGHT">
        </ColumnConstraints>
        <ColumnConstraints minWidth="200" prefWidth="200"
                           maxWidth="Infinity" hgrow="ALWAYS">
        </ColumnConstraints>
    </columnConstraints>

    <!-- Add Header Label -->
    <Label text="Registration Form" GridPane.columnIndex="0"
           GridPane.rowIndex="0" GridPane.columnSpan="2"
           GridPane.rowSpan="1" GridPane.halignment="CENTER" >
        <font>
            <Font name="Arial" size="24" ></Font>
        </font>
        <GridPane.margin>
            <Insets top="20" right="0" bottom="20" left="0"></Insets>
        </GridPane.margin>
    </Label>

    <!-- Add First Name Label -->
    <Label text="First Name" GridPane.columnIndex="0" GridPane.rowIndex="1"/>
    <!-- Add First Name Text Field -->
    <TextField fx:id="firstName" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="1"/>
    <!-- Add Last Name Label -->
    <Label text="First Name" GridPane.columnIndex="0" GridPane.rowIndex="2"/>
    <!-- Add Last Name Text Field -->
    <TextField fx:id="lastName" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="2"/>
    <!-- Add Designation Label -->
    <Label text="Designation" GridPane.columnIndex="0" GridPane.rowIndex="3"/>
    <!-- Add Designation Text Field -->
    <TextField fx:id="designation" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="3"/>
    <!-- Add Email Label -->
    <Label text="Email" GridPane.columnIndex="0" GridPane.rowIndex="4"/>
    <!-- Add Email Text Field -->
    <TextField fx:id="email" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="4"/>
    <!-- Add Telephone Label -->
    <Label text="Telephone" GridPane.columnIndex="0" GridPane.rowIndex="5"/>
    <!-- Add Telephone Text Field -->
    <TextField fx:id="telephone" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="5"/>
    <!-- Add Age Label -->
    <Label text="Age" GridPane.columnIndex="0" GridPane.rowIndex="6"/>
    <!-- Add Age Text Field -->
    <TextField fx:id="age" prefHeight="40" GridPane.columnIndex="1" GridPane.rowIndex="6"/>


    <!-- Add Submit Button -->
    <Button fx:id="submitButton" text="Submit"
            prefWidth="100" prefHeight="40" defaultButton="true"
            GridPane.columnIndex="0" GridPane.rowIndex="7"
            GridPane.halignment="CENTER"
            onAction="#handleSubmitButtonAction">
        <GridPane.margin>
            <Insets top="20" right="0" bottom="20" left="0"></Insets>
        </GridPane.margin>
    </Button>
    <!-- Add Clear Button -->
    <Button fx:id="clearButton" text="Clear"
            prefWidth="100" prefHeight="40" defaultButton="true"
            GridPane.columnIndex="1" GridPane.rowIndex="7"
            GridPane.halignment="CENTER"
            onAction="#handleClearButtonAction">
        <GridPane.margin>
            <Insets top="20" right="0" bottom="20" left="0"></Insets>
        </GridPane.margin>
    </Button>

    <!-- Message Label -->
    <Label  fx:id="message"
                                                   GridPane.columnIndex="0"
                                                   GridPane.columnSpan="4"
                                                   GridPane.halignment="CENTER"
                                                   GridPane.rowIndex="8"/>

</GridPane>
```

Main

```java
package lk.nibm.ead2.hrms;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;
import lk.nibm.ead2.hrms.controller.EmployeeController;

public class Main extends Application {

          @Override
          public void start(Stage primaryStage) throws Exception {
                    //Create Controller
        EmployeeController controller = new EmployeeController();

        Parent root = FXMLLoader.load(getClass().getResource("/fxml/Registration.fxml"));

        //Show stage
        primaryStage.setTitle("HRMS");
        primaryStage.setScene(new Scene(root,400,400));
        primaryStage.show();
          }

          public static void main(String[] args) {
                    launch(args);
          }

}
```