PROJECTS ON

**Profession Consultancy Services**

DEVELOPED BY

**Pavithra M**

**PROJECT TITLE**

**Profession Consultancy Services**

Batch Code: S201113400127

Center Name: Malleshwaram centre

Name of the Coordinator: Mrs. Lopamudra Bera

Batch Code: S20111

Course Name: PGJQP

Date of Submission:03 August 2020

**CERTIFICATE**

This is to certify that this report, titled **Profession Consultancy Services**Embodies the original work done Pavithra M , in the partial fulfillment of her course requirement at NIIT.

**Coordinator: Mrs. .Lopamudra Bera**

**ACKNOWLEGEMENT**

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them. I am highly indebted to (NIIT) for their guidance and constant supervision as well as for providing necessary information regarding the project & for their support in completing the project.

I would like to express my gratitude towards my Faculty (Mrs. Lopamudra Bera) for her kind co-operation and encouragement, which help me in completion of this project. I would like to express my special gratitude and thanks to industry persons for giving me such attention and time.

**TABLE OF CONTENTS**

[1)Aim](#_Toc46448897)

[2)Objective](#_Toc46448898)

[3)Purpose of project](#_Toc46448899)

[4)Phases in sdlc](#_Toc46448900)

5)Configuration

6) Project requirement

[7)Sql input/output](#_Toc46448902)

8) Eclipse input/output

9)References

**1. AIM**

The main Aim of my project is to access different kinds of profiles in my login page, which includes various Roles such as HR, Employee and PM.This saves a lot of time of the user and also further provides safe and easy access to the authorized personnel trough modified login page.

**2. OBJECTIVES**

The main Aim of my project is to access different kinds of profiles in my login page, which includes various Roles such as HR, Employee and PM.This saves a lot of time of the user and further provides safe and easy access to the authorized personnel through modified login page.

Usually, the whole process is very time consuming and it involves lot of costing during job seekers and Project Manager. It is difficult to update records and takes much time to maintain the information. Thus, this project helps in solving all the issues related to time cost.

**3. Purpose of the Project**

A Login form is used to enter authentication credentials to access a restricted page or form. The login form contains a field for the username and another for the password. When the login form is submitted its underlying code checks that the credentials are authentic, giving the user can access the restricted page. If a user is not able to provide authentic credentials, they will not be able to proceed past the login form. A login form is a record form whose insert, update and delete properties have been disabled.

In my project Login Form has three profiles i.e. HR, PM and Employee and each one can access their data on their own.

**4. Phases in Project Execution (SDLC)**

* Requirement gathering and analysis
* Design
* Implementation or coding
* Testing
* Deployment
* Maintenance

### 1) Requirement Gathering and Analysis

During this phase, all the relevant information is collected from the customer to develop a product as per their expectation. Any ambiguities must be resolved in this phase only.

### 2) Design

In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

### 3) Implementation or Coding

Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.

### 4) Testing

Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.

### 5) Deployment

Once the product is tested, it is deployed in the production environment or first [UAT (User Acceptance testing)](https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/) is done depending on the customer expectation.

### 6) Maintenance

After the deployment of a product on the production environment, maintenance of the product i.e. if any issue comes up and needs to be fixed or any enhancement is to be done is taken care by the developers.

**5. CONFIGURATION**

* Hardware: 4 GB RAM, I5 Processor, 500 GB Hard disk
* Operating System:Windows 10
* Software: Microsoft SQL Server Management Studio, Eclipse IDE, Notepad, Google Chrome

**6. Project requirement**

**1. Open SQL Server Management Studio**

* Create database named PCSDB then within this database :
* Create following 5 tables:

-- Employee(empId[identity(1,1), primary key], firstName, lastName, userId, password, gender, role, active)

-- Skill(skillId[identity(1,1), primary key], skillName, skillDesc, active)

-- Job(jobId[identity(1,1), primary key], jobTitle, jobDescription, companyName, Location, keySkill,salary,active)

-- EMPSkill(ESId[identity(1,1), primary key], empId, skillId, expYear)

-- EmpJob(EJId[identity(1,1), primary key],empId,jobId,recruited)

**2. Create project in eclipse**

a. Create following packages:

model

dao

daoImpl

controller

view

test

config

b. Add following classes and interfaces as given:

i. within config package

-- JDBCConnection.java class

ii. within model package -- add classes

-- Employee.java

-- Skill.java

-- Job.java

iii. within dao package -- add interfaces

-- IEmployeeDao.java

-- ISkillDao.java

-- IJobDao.java

iv. within daoImpl package -- add classes

-- EmployeeDaoImpl.java

-- SkillDaoImpl.java

-- JobDaoImpl.java

v. within controller package -- add classes

-- EmployeeController.java

-- SkillController.java

-- JobController.java

vi. within view package -- add frame classes

-- LoginFrame.java

-- RegistrationFrame.java

-- EmpWindow.java (following are the services by Employee)

-- View Profile

-- Update Profile

-- Update Skill

-- Apply Job

-- Logout

-- PMWindow.java (following are the services by Project Manager)

-- View Skill-wise Employee List

-- View All Skills

-- Add Job

-- View All Job

-- View Skill-wise Job

-- View Employee list who applied for Job

-- Deactivate Job

-- Logout

-- HRWindow.java (following are the services by HR)

-- Active Employee

-- Deactivate Employee

-- View All Employee

-- View Selective Employee

-- Add Skill

-- View All Skills

-- Deactivate Skill

-- Logout

vii. within test package create test classes

**7. Microsoft SQL Server Management Studio INPUT & OUTPUT**

* **Requirement Analysis:--**
* Create database named PCSDB
* Create following 5 tables within this database:

-- Employee(empId[identity(1,1), primary key], firstName, lastName, userId, password, gender, role, active)

-- Skill(skillId[identity(1,1), primary key], skillName, skillDesc, active)

-- Job(jobId[identity(1,1), primary key], jobTitle, jobDescription, companyName, Location, keySkill, salary, active)

-- EMPSkill(ESId[identity(1,1), primary key], empId, skillId, expYear)

-- EmpJob(EJId[identity(1,1), primary key],empId, jobId, recruited)

* **Code for creating Database :**
* Create database PSCDB

**Code for Creating Tables and Insert records:**

* **For Employee:**

Create table Employee(empId int identity(1,1) primary key,

firstName varchar(30),

lastName varchar(30),

userId varchar(20),

password varchar(10),

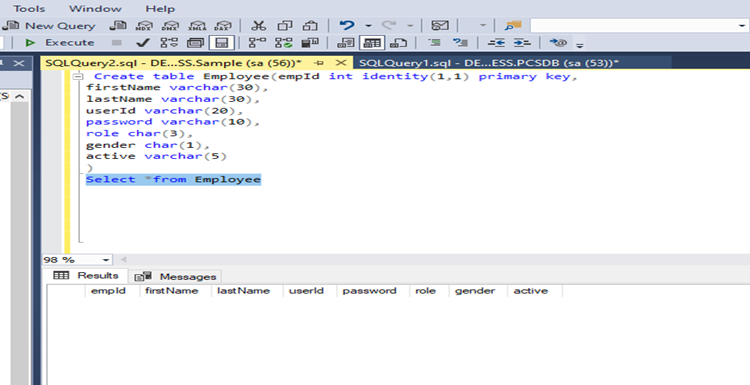
role char(3),

gender char(1),

active varchar(5)

)

Select \*from Employee

****

* **For Job:**

CREATE TABLE job (

jobid float primary key,

jobtitle varchar(30),

jobdesc varchar(30),

companyName varchar(30),

cLocation varchar(30),

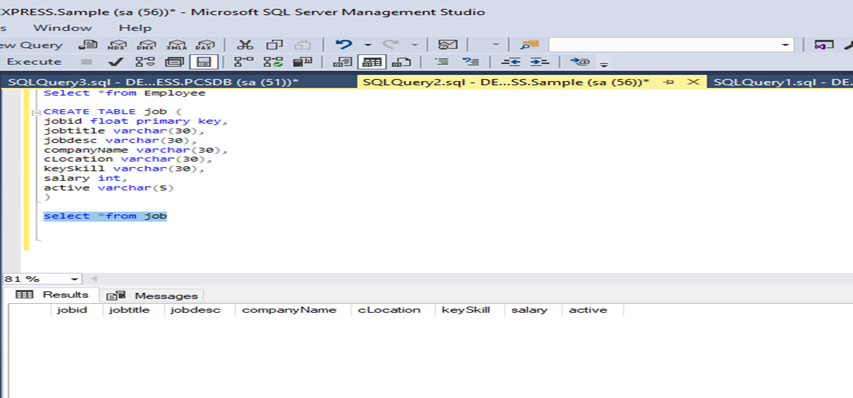
keySkill varchar(30),

salary int,

active varchar(5)

)

select \*from job

****

**For Skill:**

CREATE TABLE skill (

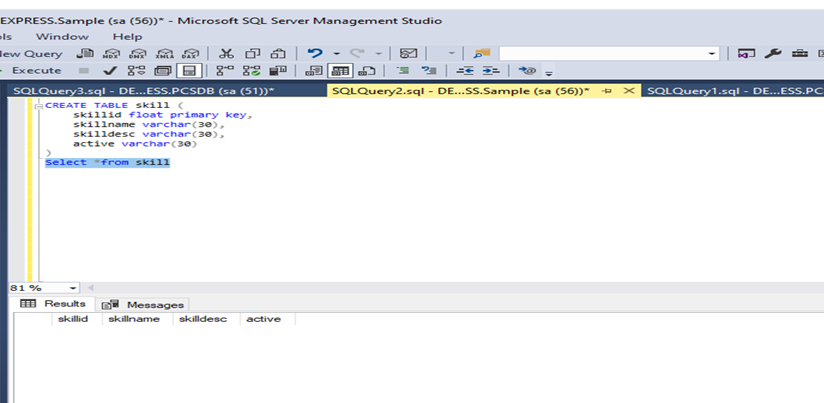
skillid float primary key,

skillname varchar(30),

skilldesc varchar(30),

active varchar(30)

)

****

**For EMPSkill:**

Create table EMPSkill(

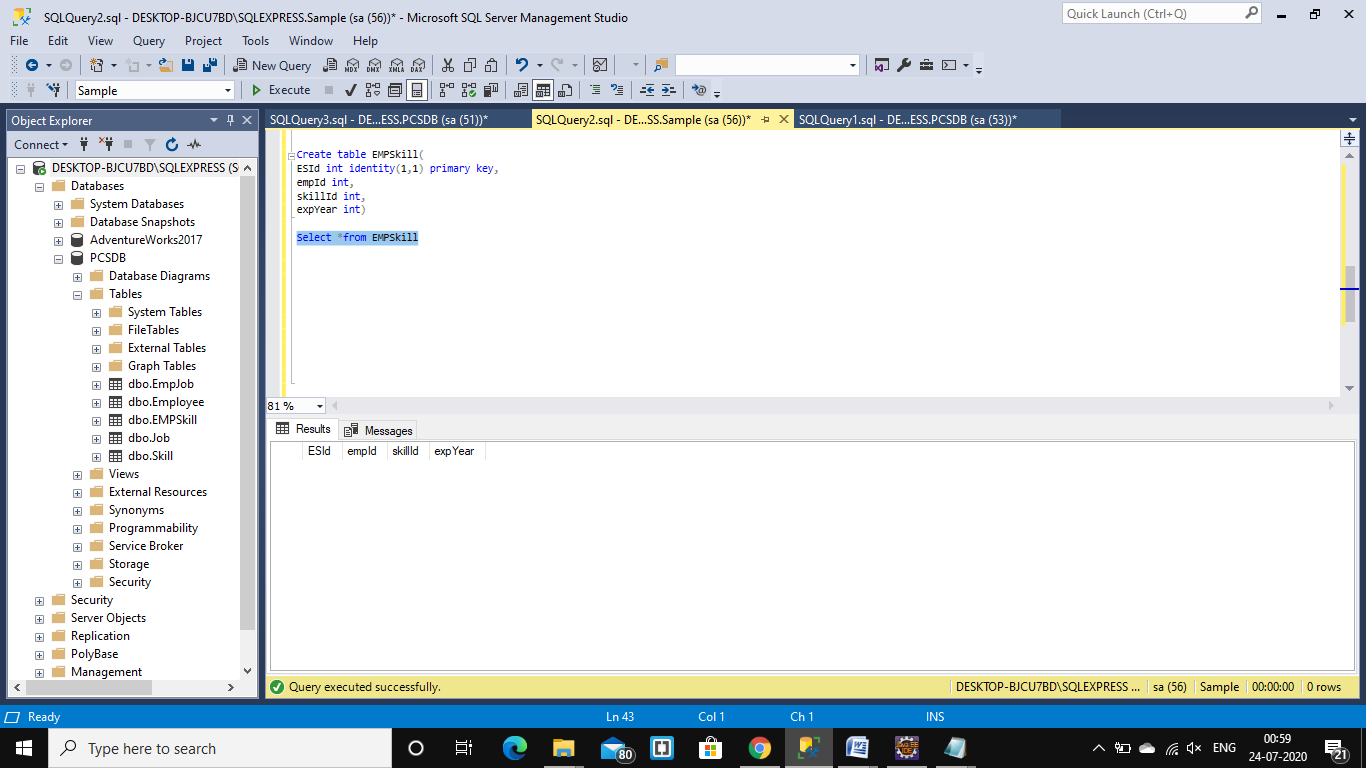
ESId int identity(1,1) primary key,

empId int,

skillId int,

expYear int)

Select \*from EMPSkill

****

**For EmpJob:**

Create table EmpJob(

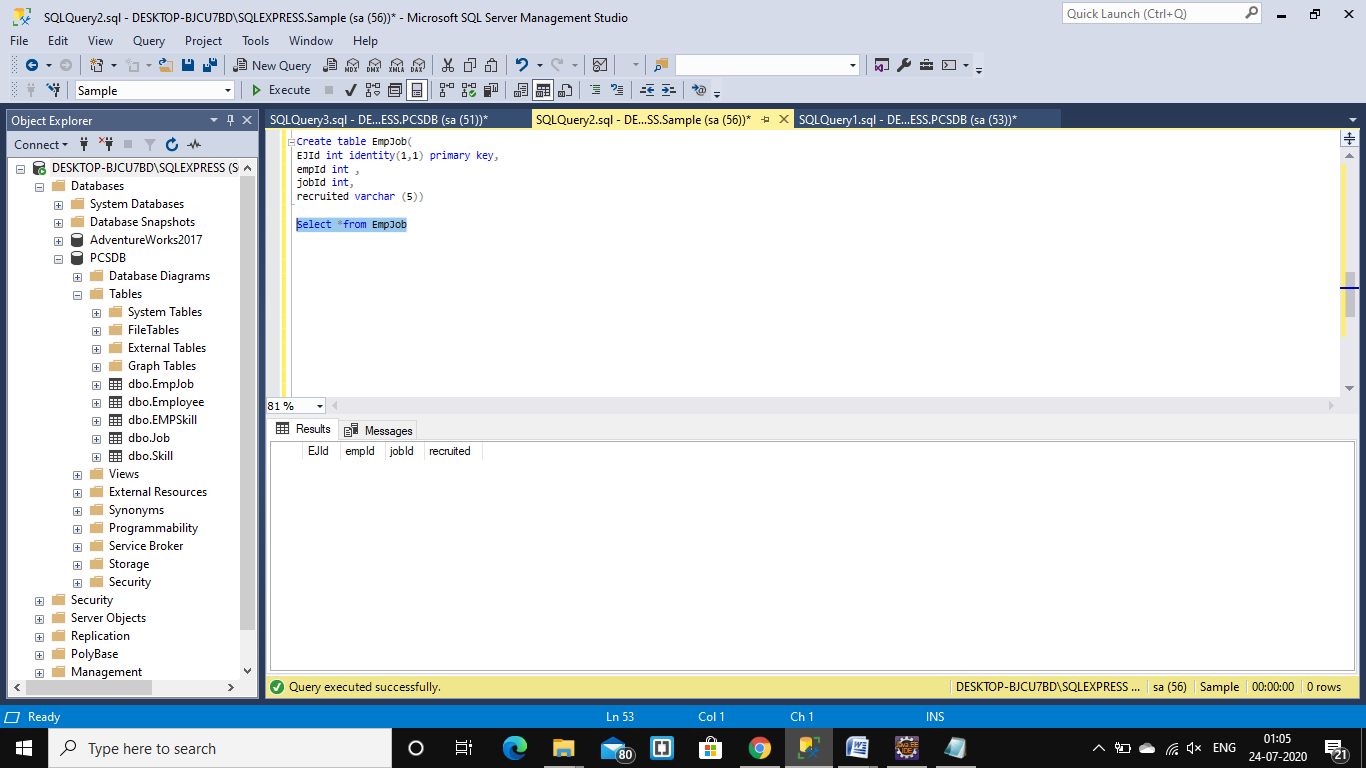
EJId int identity(1,1) primary key,

empId int ,

jobId int,

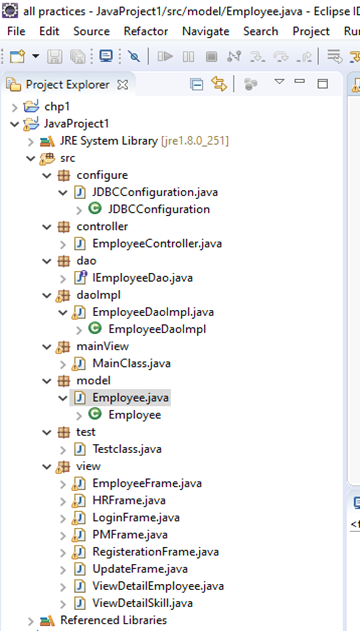
recruited varchar (5))

Select \*from EmpJob

****

**8. ECLIPSE IDE JAVA INPUT & OUTPUT**

1. All Packages and Classes



1. Package code and output

[\\JDBCConfiguration.java](file:///\\JDBCConfiguration.java)

**package** configure;

**import** java.sql.\*;

**public** **class** JDBCConfiguration {

**public** **static** Connection getDBConnection() **throws** ClassNotFoundException, SQLException{

String url="jdbc:sqlserver://DESKTOP-SEOJ2RF\\SQLEXPRESS:1434;database=PCSDB";

String username="sa";

String password="niit@123";

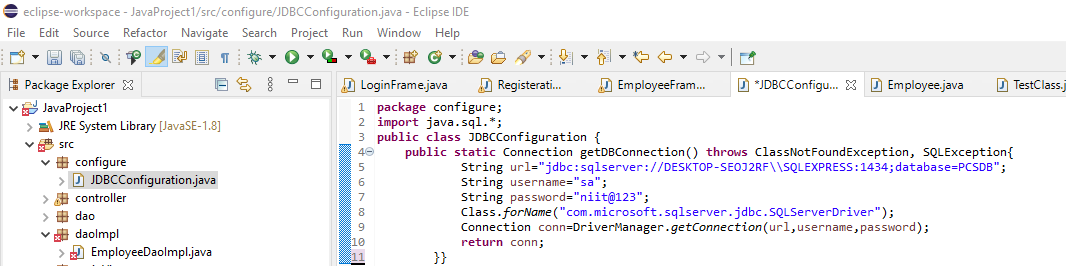
Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection conn=DriverManager.*getConnection*(url,username,password);

**return** conn;

}

}

****

**\\LoginFrame.java**

**package** view;

**import** javax.swing.\*;

**import** java.awt.event.\*;

**import** java.awt.\*;

**public** **class** LoginFrame **extends** JFrame{

JLabel lTitle, lUser,lPassword,lMessage,lRole;

JTextField tUser;

JPasswordField tPassword;

JComboBox c1;

JButton bLogin,bRegistration;

**public** LoginFrame() {

lTitle=**new** JLabel("Login Here");

lTitle.setBounds(180, 20, 150, 25);

lUser=**new** JLabel("User Id");

lUser.setBounds(50, 80, 150, 25);

tUser=**new** JTextField();

tUser.setBounds(200, 80, 150, 25);

lPassword=**new** JLabel("Password");

lPassword.setBounds(50, 160, 150, 25);

tPassword=**new** JPasswordField();

tPassword.setBounds(200, 160, 150, 25);

tPassword.setEchoChar('\*');

String[] values= {"NA","EMP","HR","PM"};

lRole=**new** JLabel("Role");

lRole.setBounds(50,200,150,25);

c1=**new** JComboBox (values);

c1.setBounds(200,200,150,25);

bLogin=**new** JButton("Login");

bLogin.setBounds(120, 250,150, 25);

bLogin.addActionListener((ActionListener) **this**);

Ì

bRegistration=**new** JButton("Registration");

bRegistration.setBounds(300, 250,150, 25);

bRegistration.addActionListener((ActionListener) **this**);

lMessage=**new** JLabel();

lMessage.setBounds(120, 320, 320, 25);

add(lTitle);

add(lUser);

add(lPassword);

add(tUser);

add(tPassword);

add(bLogin);

add(lMessage);

add(lRole);

add(c1);

add(bRegistration);

setSize(500,400);

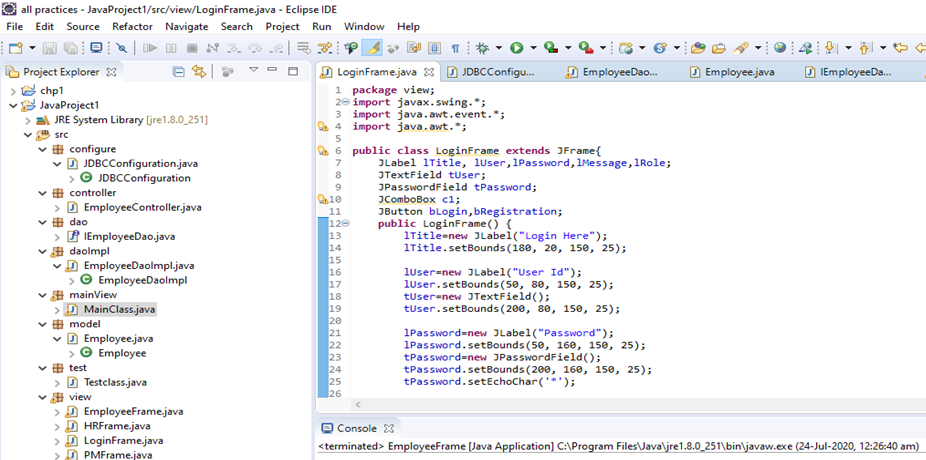
setLayout(**null**);

setVisible(**true**);

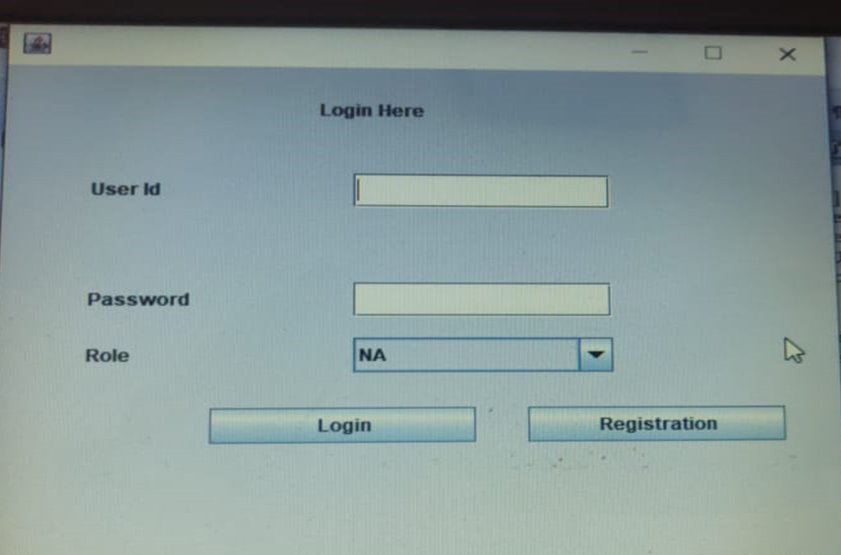
setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

}

****

**Ù**

****

**\\main.java**

**package mainView;**

**import java.sql.Connection;**

**import java.sql.DriverManager;**

**import java.sql.SQLException;**

**import view.LoginFrame;**

**public class MainClass {**

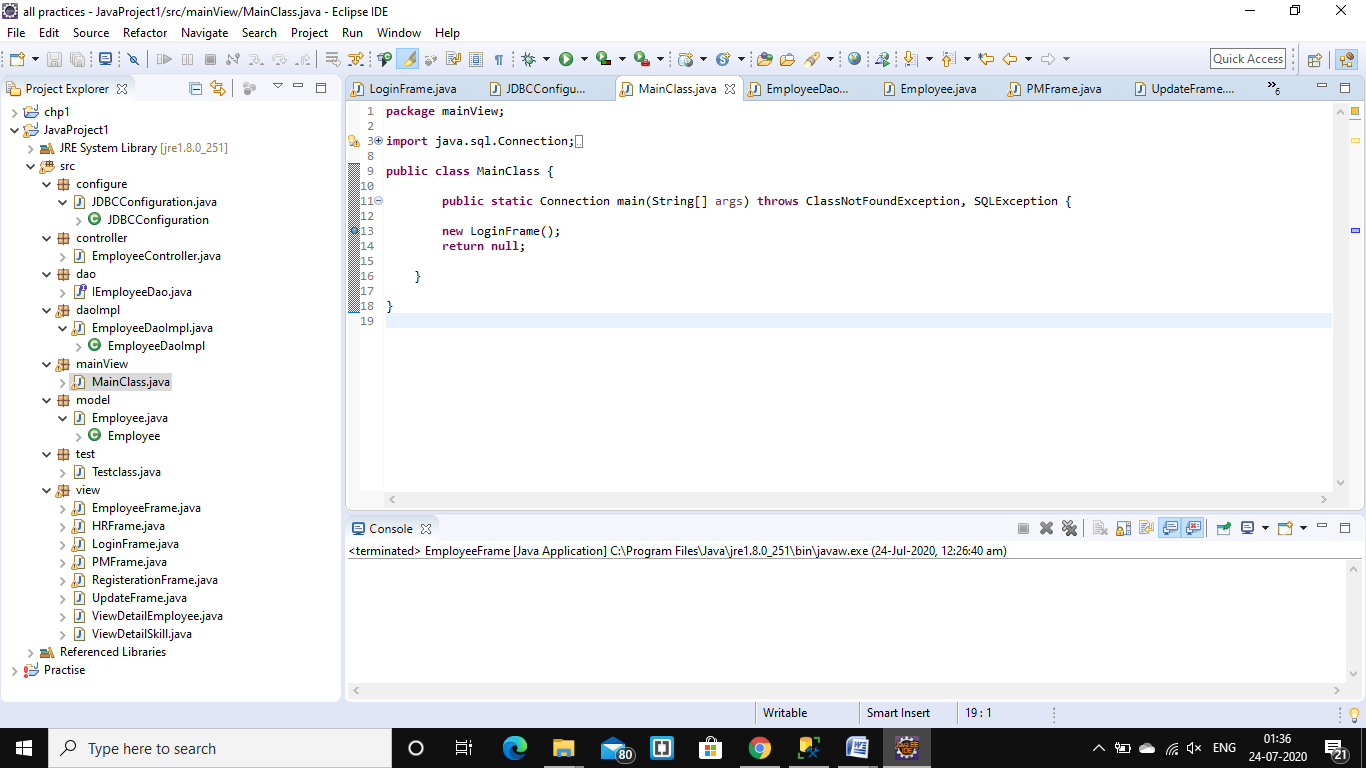
**public static Connection main(String[] args) throws ClassNotFoundException, SQLException {**

**new LoginFrame();**

**return null;**

**}**

**}**

****

\\RegistrationFrame.java

**package view;**

**import javax.swing.\*;**

**import java.awt.\*;**

**import java.awt.event.\*;**

**class Myform extends JFrame{**

**JLabel label1,label2,label3,label4,label5,label6,label7;**

**JTextField t1,t2,t3,t4,t5,t6;**

**JRadioButton Male,Female;**

**JButton submit;**

**JLabel msg;**

**Myform(){**

**setTitle("Registration Form");**

**setSize(700,500);**

**setLocationRelativeTo(null);**

**setDefaultCloseOperation(EXIT\_ON\_CLOSE);**

**Container c=getContentPane();**

**c.setLayout(null);**

**label1=new JLabel("First Name:");**

**label1.setBounds(20,50,100,20);**

**c.add(label1);**

**t1=new JTextField();**

**t1.setBounds(130,50,100,20);**

**c.add(t1);**

**label2=new JLabel("Last Name:");**

**label2.setBounds(20,100,100,20);**

**c.add(label2);**

**t2=new JTextField();**

**t2.setBounds(130,100,100,20);**

**c.add(t2);**

**label7=new JLabel("Gender");**

**label7.setBounds(20,150,100,20);**

**c.add(label7);**

**Male=new JRadioButton("Male");**

**Female=new JRadioButton("Female");**

**Male.setBounds(130,150,80,20);**

**Female.setBounds(220,150,80,20);**

**c.add(Male);**

**c.add(Female);**

**label3=new JLabel("Email Id:");**

**label3.setBounds(20,200,100,20);**

**c.add(label3);**

**t3=new JTextField();**

**t3.setBounds(130,200,100,20);**

**c.add(t3);**

**label4=new JLabel("User ID:");**

**label4.setBounds(20,250,100,20);**

**c.add(label4);**

**t4=new JTextField();**

**t4.setBounds(130,250,100,20);**

**c.add(t4);**

**label5=new JLabel("Password:");**

**label5.setBounds(20,300,100,20);**

**c.add(label5);**

**t5=new JTextField();**

**t5.setBounds(130,300,100,20);**

**c.add(t5);**

**label6=new JLabel("Mobile no.:");**

**label6.setBounds(20,350,100,20);**

**c.add(label6);**

**t6=new JTextField();**

**t6.setBounds(130,350,100,20);**

**c.add(t6);**

**submit=new JButton("Submit");**

**submit.setBounds(150,400,80,20);**

**c.add(submit);**

**setVisible(true);**

**}**

**}**

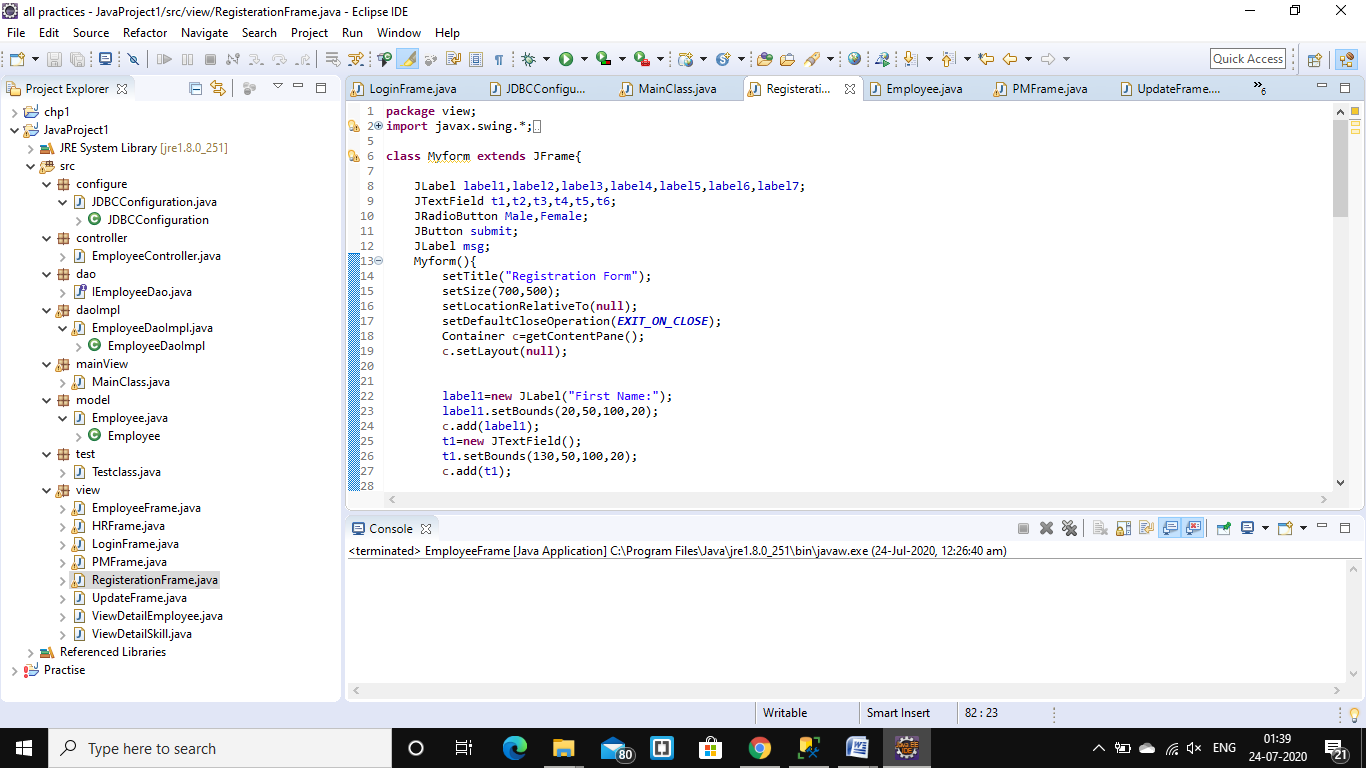
**public class RegisterationFrame {**

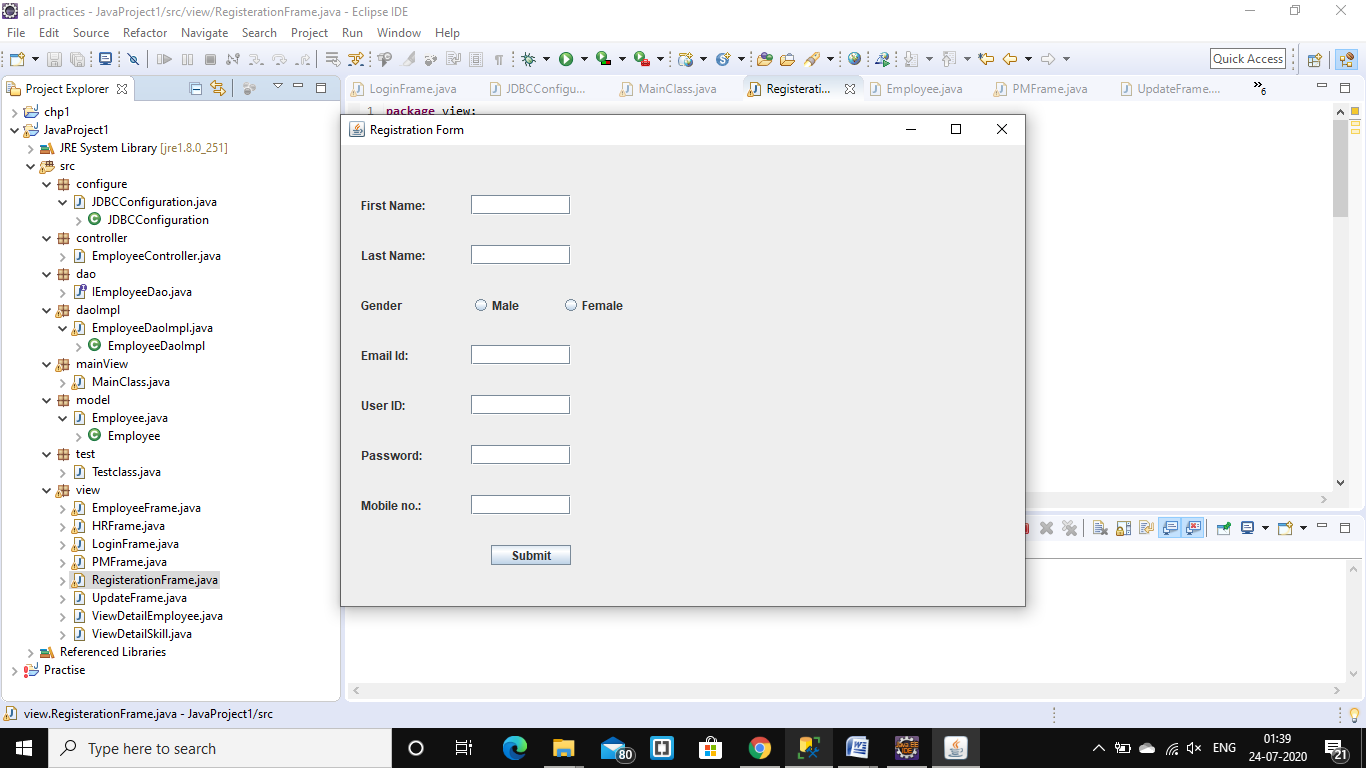
**public static void main(String args[]) {**

**new Myform();**

**}**

**}**

****

****

[**\\Emplyee.java**](file:///\\Emplyee.java)

**package** model;

**public** **class** Employee {

**private** **int** empId;

**private** String firstName;

**private** String lastName;

**private** String userId;

**private** String password;

**private** String role;

**private** **char** gender;

**private** String active;

**public** String getActive() {

**return** active;

}

**public** **void** setActive(String active) {

**this**.active = active;

}

**public** Employee() {

}

**public** Employee(String firstName, String lastName, String userId, String password, String role, **char** gender) {

**super**();

**this**.firstName = firstName;

**this**.lastName = lastName;

**this**.userId = userId;

**this**.password = password;

**this**.role = role;

**this**.gender = gender;

}

**public** **int** getEmpId() {

**return** empId;

}

**public** **void** setEmpId(**int** empId) {

**this**.empId = empId;

}

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getUserId() {

**return** userId;

}

**public** **void** setUserId(String userId) {

**this**.userId = userId;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** String getRole() {

**return** role;

}

**public** **void** setRole(String role) {

**this**.role = role;

}

**public** **char** getGender() {

**return** gender;

}

**public** **void** setGender(**char** gender) {

**this**.gender = gender;

}

@Override

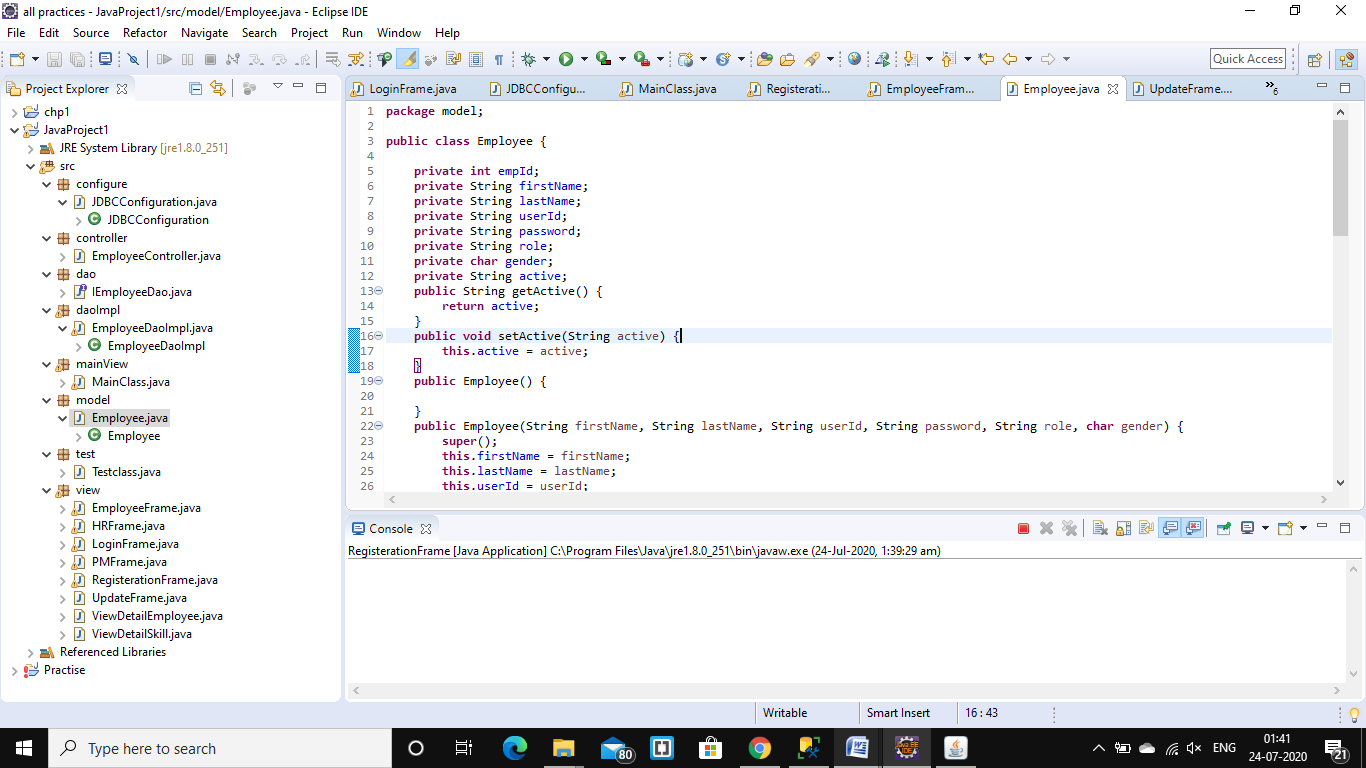
**public** String toString() {

**return** "Employee [empId=" + empId + ", firstName=" + firstName + ", lastName=" + lastName + ", userId=" + userId

+ ", password=" + password + ", role=" + role + ", gender=" + gender + ", active=" + active + "]";

}

}

****

[**\\EmployeeFrame.java**](file:///\\EmployeeFrame.java)

**package** view;

**import** java.awt.Container;

**import** javax.swing.\*;

**class** Myframe **extends** JFrame{

JButton ViewProfile,UpdateProfile, UpdateSkill, ApplyJob, Logout;

Myframe(){

setTitle("Employee Frame");

setSize(700,500);

setLocationRelativeTo(**null**);

setDefaultCloseOperation(***EXIT\_ON\_CLOSE***);

Container c=getContentPane();

c.setLayout(**null**);

ViewProfile=**new** JButton("View Profile");

ViewProfile.setBounds(20,50,100,20);

c.add(ViewProfile);

UpdateProfile=**new** JButton("Update Profile");

UpdateProfile.setBounds(20,100,100,20);

c.add(UpdateProfile);

UpdateSkill=**new** JButton("Update Skill");

UpdateSkill.setBounds(20,150,100,20);

c.add(UpdateSkill);

ApplyJob=**new** JButton("Apply Job");

ApplyJob.setBounds(20,200,100,20);

c.add(ApplyJob);

Logout=**new** JButton("Logout");

Logout.setBounds(20,250,100,20);

c.add(Logout);

setVisible(**true**);

}

}

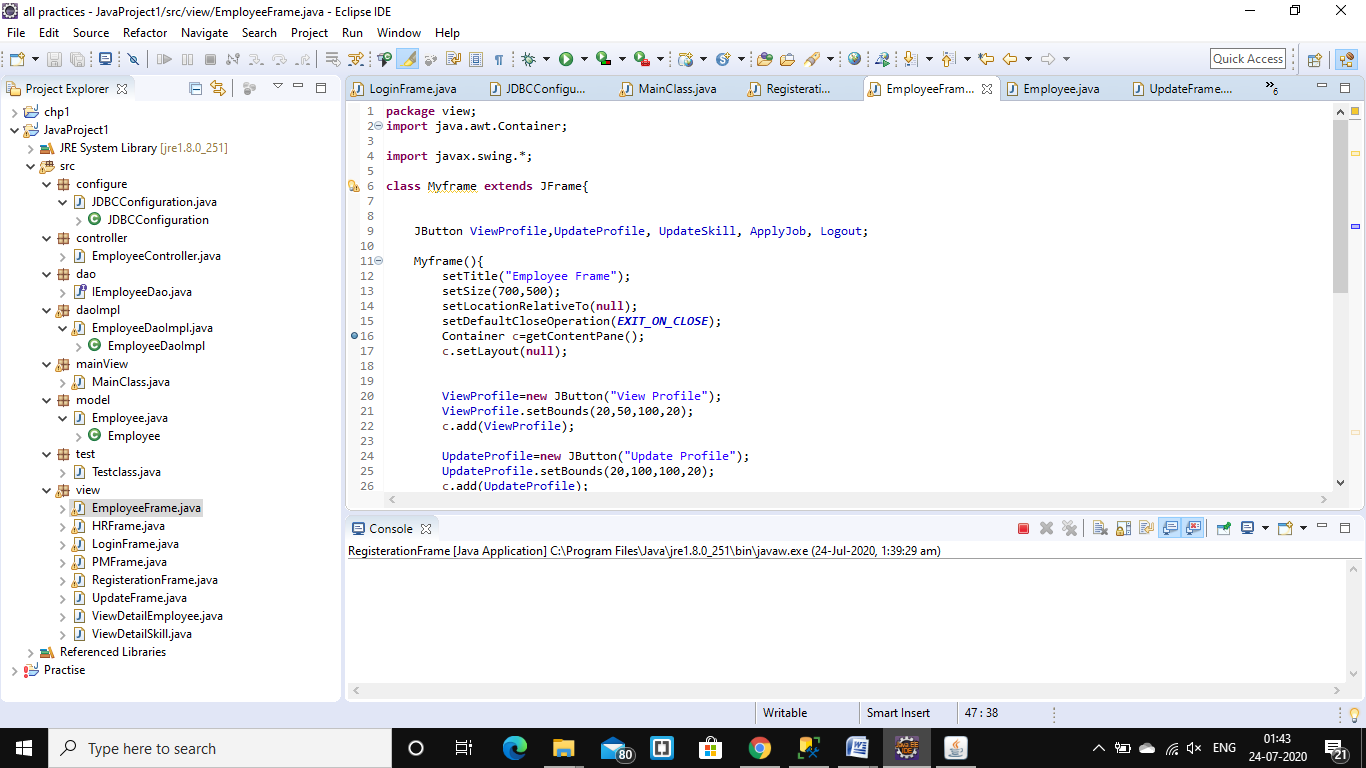
**public** **class** EmployeeFrame {

**public** **static** **void** main(String args[]) {

**new** Myframe();

}

}

****

[**\\HRFrame.java**](file:///\\HRFrame.java)

**package** view;

**import** javax.swing.\*;

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** form **extends** JFrame{

JButton ViewSkill\_wiseEmployeeList, ViewAllSkills, AddJob,ViewAllJob, ViewSkill\_wiseJob, ViewEmployeelist , DeactivateJob,Logout;

form(){

setTitle("HR Frame");

setSize(700,500);

setLocationRelativeTo(**null**);

setDefaultCloseOperation(***EXIT\_ON\_CLOSE***);

Container c=getContentPane();

c.setLayout(**null**);

ViewSkill\_wiseEmployeeList=**new** JButton("Skill\_wiseEmployee List");

ViewSkill\_wiseEmployeeList.setBounds(20,50,100,20);

c.add(ViewSkill\_wiseEmployeeList);

ViewAllSkills=**new** JButton("All Skills");

ViewAllSkills.setBounds(20,100,100,20);

c.add(ViewAllSkills);

AddJob=**new** JButton("Add Job");

AddJob.setBounds(20,150,100,20);

ViewAllJob=**new** JButton("All Job");

ViewAllJob.setBounds(20,200,100,20);

c.add(ViewAllJob);

ViewSkill\_wiseJob=**new** JButton("Skill\_wiseJob");

ViewSkill\_wiseJob.setBounds(20,250,100,20);

c.add(ViewSkill\_wiseJob);

ViewEmployeelist=**new** JButton("Employee list");

ViewEmployeelist.setBounds(20,300,100,20);

c.add(ViewEmployeelist);

DeactivateJob=**new** JButton("Deactivate Job");

DeactivateJob.setBounds(20,350,100,20);

c.add(DeactivateJob);

Logout=**new** JButton("Logout");

Logout.setBounds(20,400,100,20);

c.add(Logout);

setVisible(**true**);

}

}

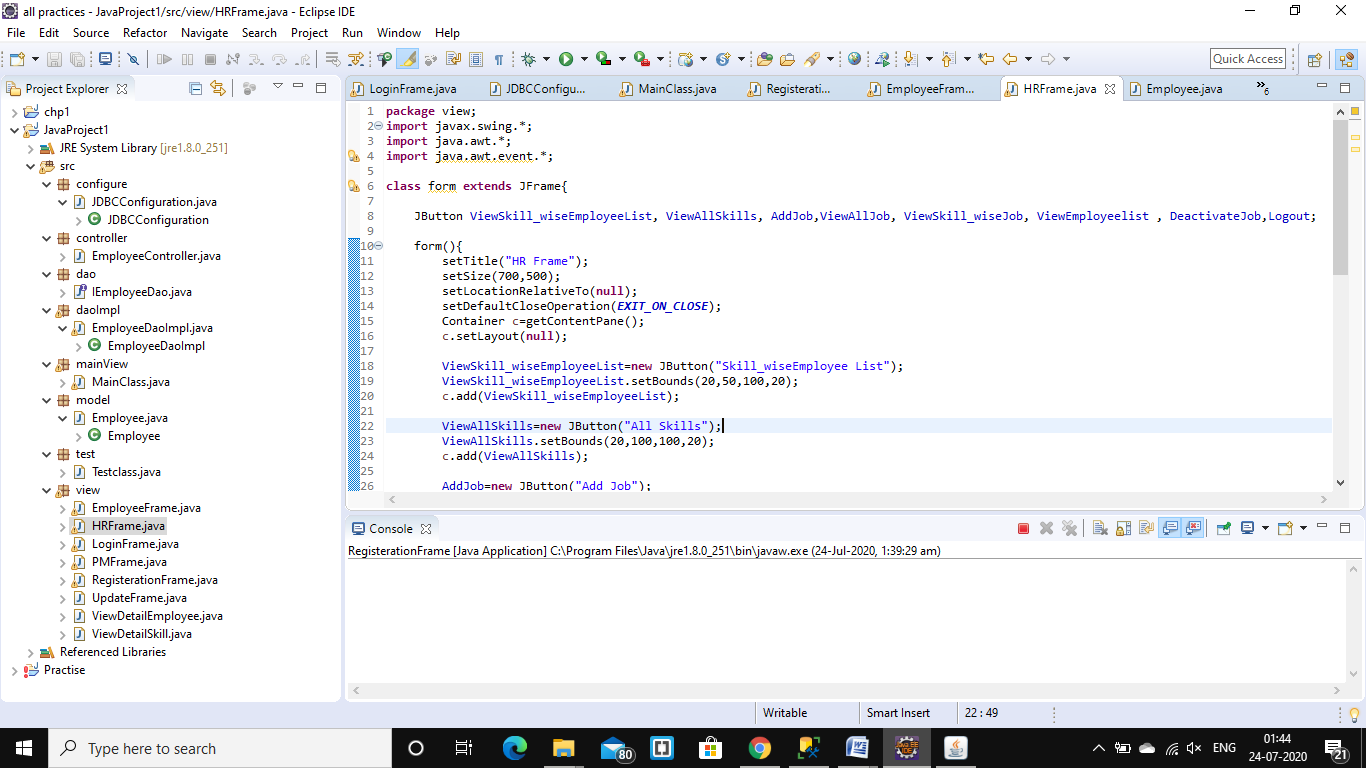
**public** **class** HRFrame {

**public** **static** **void** main(String[] args) {

**new** form();

}

}

****

[**\\PMFrame.java**](file:///\\PMFrame.java)

**package** view;

**import** javax.swing.\*;

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** forms **extends** JFrame{

JButton ActiveEmployee, DeactiveEmployee, ViewAllEmployee, ViewSelectiveEmployee, AddSkill, ViewAllSkills, DeactiveSkill,Logout;

forms(){

setTitle("PM Frame");

setSize(700,500);

setLocationRelativeTo(**null**);

setDefaultCloseOperation(***EXIT\_ON\_CLOSE***);

Container c=getContentPane();

c.setLayout(**null**);

ActiveEmployee=**new** JButton("ActiveEmployee");

ActiveEmployee.setBounds(20,50,100,20);

c.add(ActiveEmployee);

ViewAllSkills=**new** JButton("All Skills");

ViewAllSkills.setBounds(20,100,100,20);

c.add(ViewAllSkills);

DeactiveEmployee=**new** JButton("DeactiveEmployee");

DeactiveEmployee.setBounds(20,150,100,20);

c.add(DeactiveEmployee);

ViewAllEmployee=**new** JButton("All Employee");

ViewAllEmployee.setBounds(20,200,100,20);

c.add(ViewAllEmployee);

ViewSelectiveEmployee=**new** JButton("Selective Employee");

ViewSelectiveEmployee.setBounds(20,250,100,20);

c.add(ViewSelectiveEmployee);

AddSkill=**new** JButton("Add Skill");

AddSkill.setBounds(20,300,100,20);

c.add(AddSkill);

DeactiveSkill=**new** JButton("Deactive Skill");

DeactiveSkill.setBounds(20,350,100,20);

c.add(DeactiveSkill);

Logout=**new** JButton("Logout");

Logout.setBounds(20,400,100,20);

c.add(Logout);

setVisible(**true**);

}

}

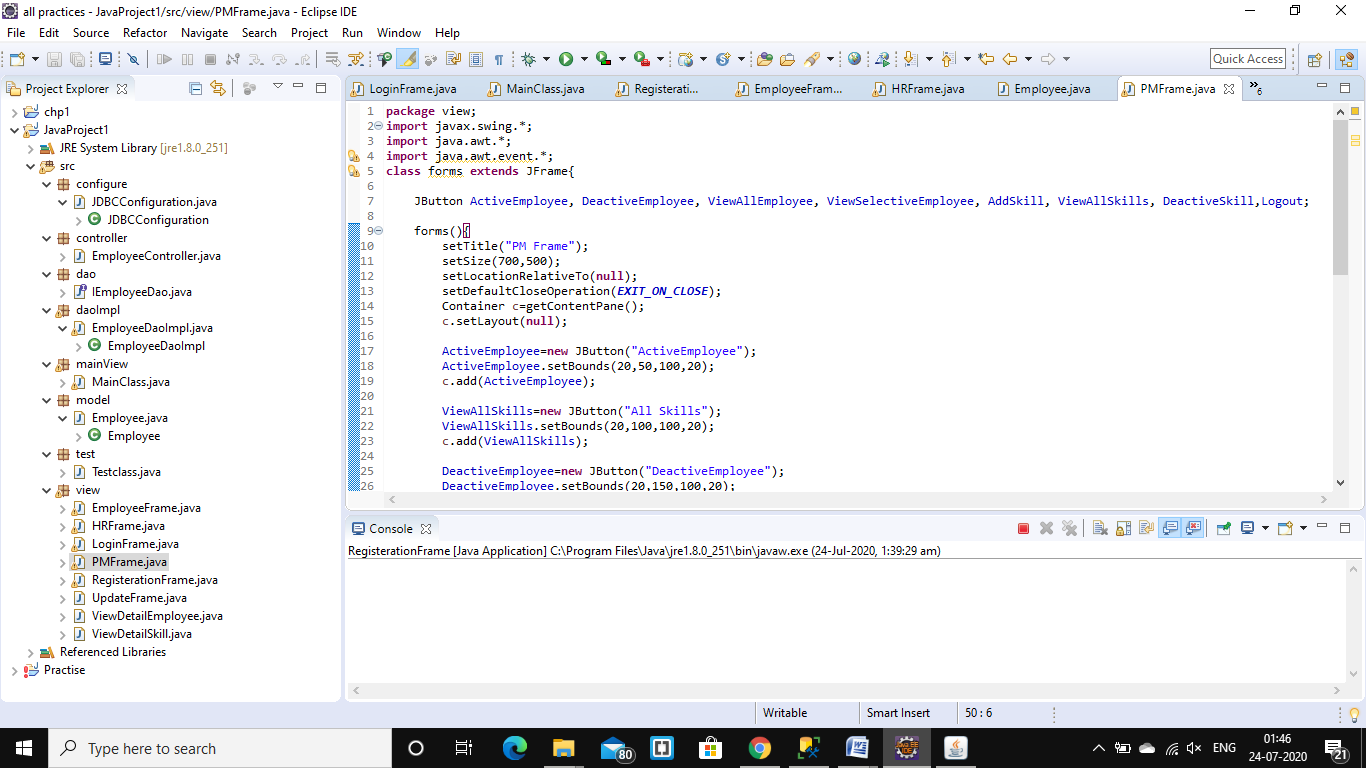
**public** **class** PMFrame {

**public** **static** **void** main(String[] args) {

**new** forms();

}

}

****

**9. References and Bibliography**

* <https://www.tutorialspoint.com/jdbc/jdbc-db-connections.htm>
* <https://www.w3schools.com/sql/sql_syntax.asp>
* <https://www.softwaretestinghelp.com/software-development-life-cycle-sdlc/>