PROJECTS ON

**Profession Consultancy Services**

DEVELOPED BY

**PREETHI**

**PROJECT TITLE**

**Profession Consultancy Services**

Student Id: S201113400126

Center Name: Malleshwaram

Name of the Coordinator: Mrs. Lopamudra Bera

Batch Code: S210050

Course Name: PGJQP

Date of Submission: 02 August 2020

**CERTIFICATE**

This is to certify that this report, titled **Profession Consultancy Services**Embodies the original work done by **Preethi**, 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](file:///C:\Users\SUBBU\Downloads\jahanvi%20S%20(3).docx#_Toc46448897)

[2)Objective](file:///C:\Users\SUBBU\Downloads\jahanvi%20S%20(3).docx#_Toc46448898)

[3)Purpose of project](file:///C:\Users\SUBBU\Downloads\jahanvi%20S%20(3).docx#_Toc46448899)

[4)Phases in sdlc](file:///C:\Users\SUBBU\Downloads\jahanvi%20S%20(3).docx#_Toc46448900)

5)Configuration

6) Project requirement

[7)Sql input/output](file:///C:\Users\SUBBU\Downloads\jahanvi%20S%20(3).docx#_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),

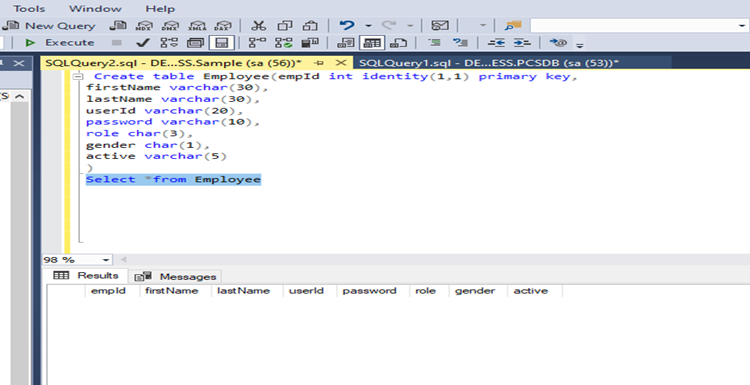
role char(3),

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),

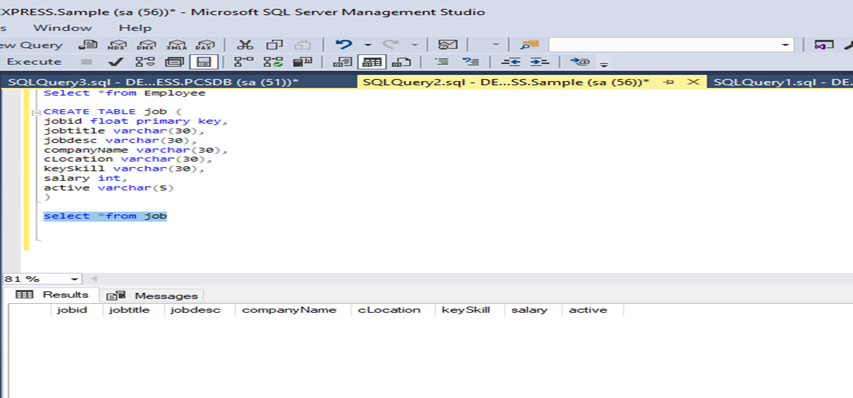
keySkill varchar(30),

salary int,

active varchar(5)

)

select \*from job

****

**For Skill:**

CREATE TABLE skill (

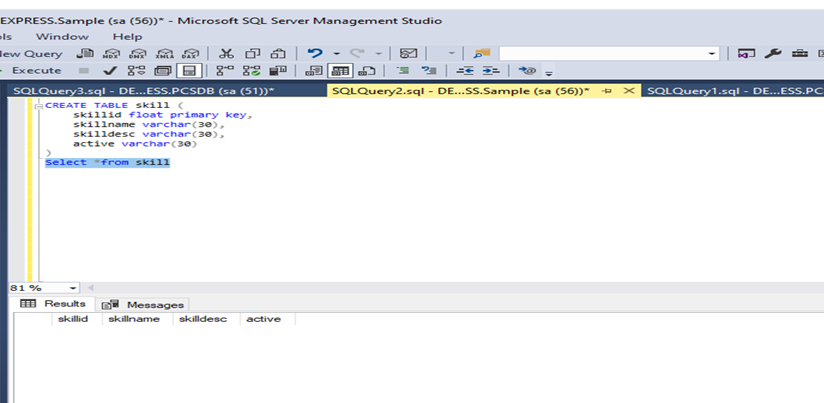
skillid float primary key,

skillname varchar(30),

skilldesc varchar(30),

active varchar(30)

)

****

**For EMPSkill:**

Create table EMPSkill(

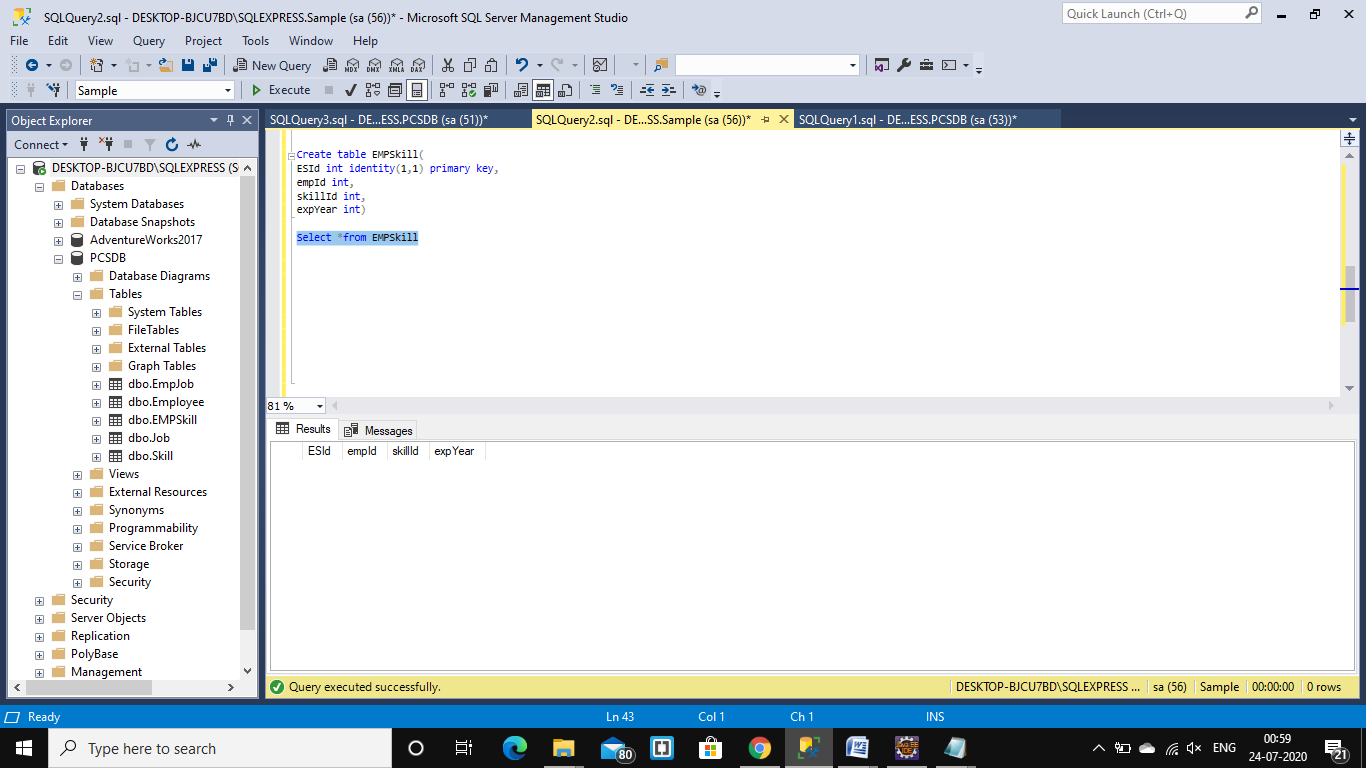
ESId int identity(1,1) primary key,

empId int,

skillId int,

expYear int)

Select \*from EMPSkill

****

**For EmpJob:**

Create table EmpJob(

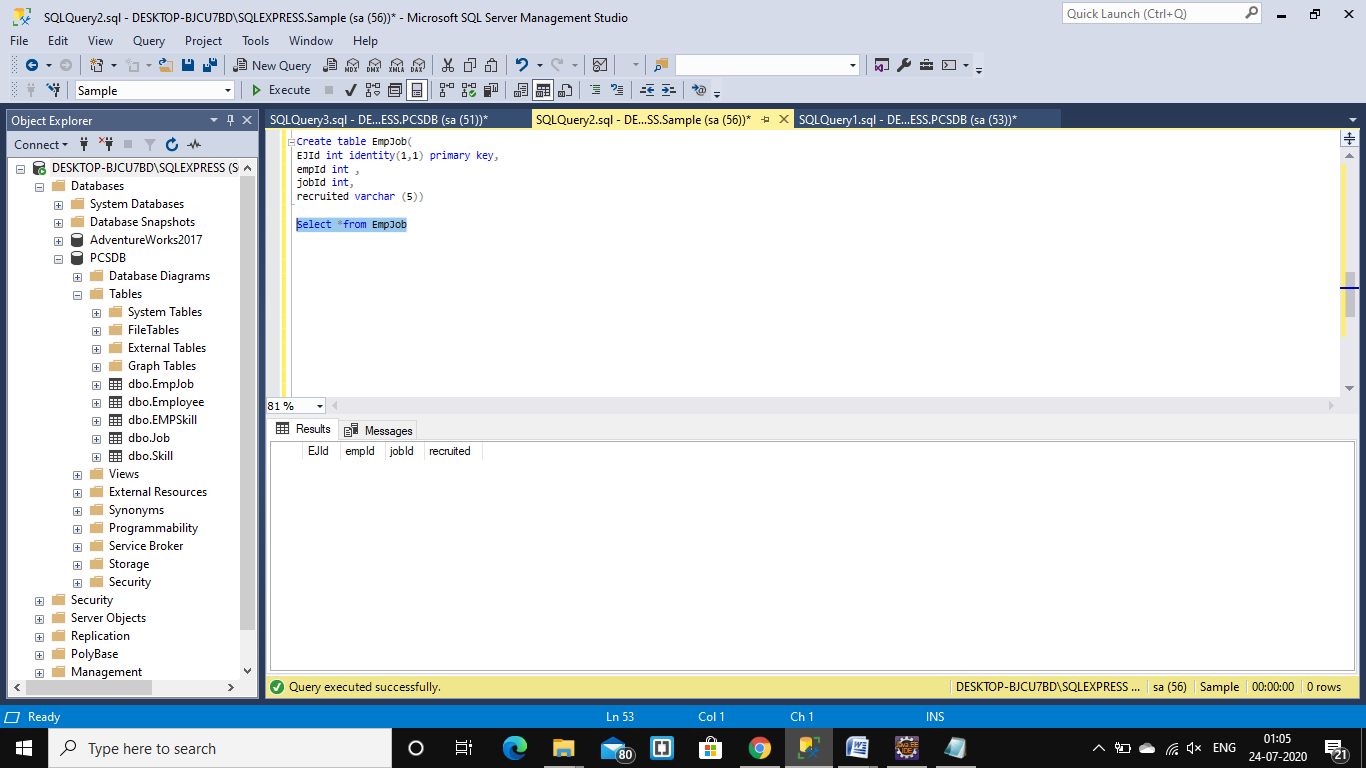
EJId int identity(1,1) primary key,

empId int ,

jobId int,

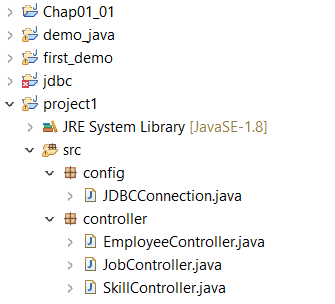
recruited varchar (5))

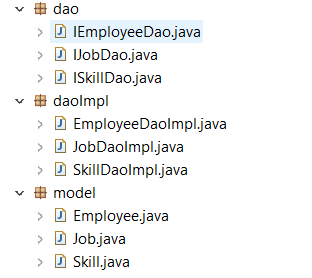
Select \*from EmpJob

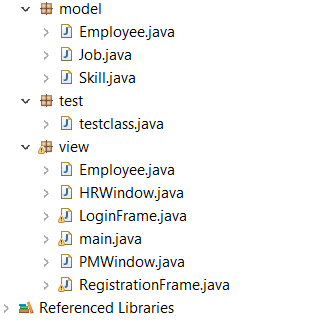
****

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

**1.ALL PACKAGES AND CLASSES**







2.Package code and output

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

**package** config;

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

**public** **class** JDBCConnection {

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

String url="jdbc:sqlserver:DESKTOP-CJ512VJ\\INSTANCE01:1433;databaseName=PCSDB;user=sa1;password=preethu";

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

Connection conn=DriverManager.*getConnection*(url);

**if**(conn!=**null**)

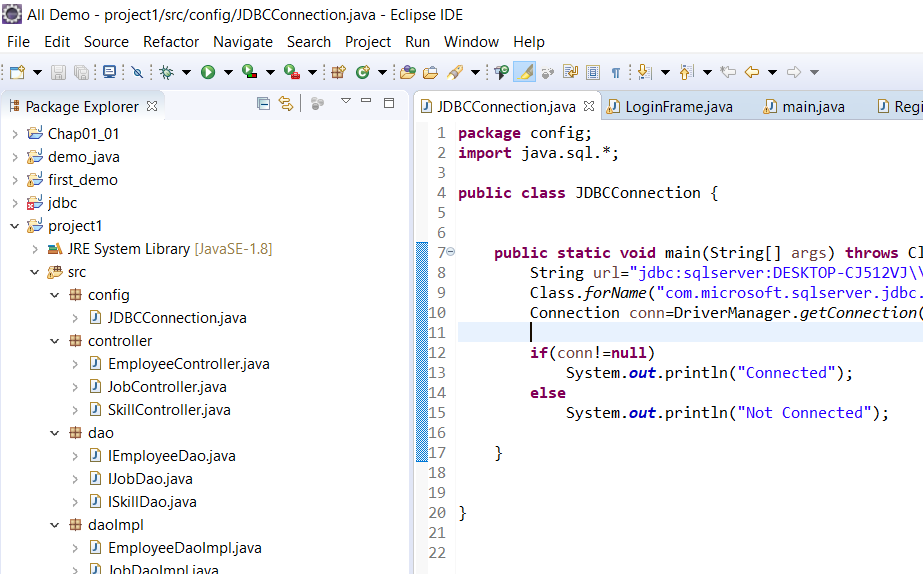
System.***out***.println("Connected");

**else**

System.***out***.println("Not Connected");

}

}



**\\LoginFrame.java**

**package** view;

**import** view.RegistrationFrame;

**import** view.Employee;

**import** view.HRWindow;

**import** config.JDBCConnection;

**import** java.awt.FlowLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.util.logging.Level;

**import** java.util.logging.Logger;

**import** javax.swing.JButton;

**import** javax.swing.JComboBox;

**import** javax.swing.JDialog;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JPasswordField;

**import** javax.swing.JTextField;

**public** **class** LoginFrame {

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

log obj=**new** log();

}

}

**class** log **extends** JFrame **implements** ActionListener

{

JLabel lTitle, lUser,lPassword,lMessage,lrole;

JTextField userid,passwords;

JPasswordField tPassword;

JButton bLogin,bregister;

JComboBox roleCombo;

**static** JDialog *dialog*;

/\*\*

\*

\*/

**public** log()

{

String[] role= {"HRA","EMP","PME"};

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

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

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

lUser.setBounds(50, 80, 180, 25);

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

lPassword.setBounds(50, 160, 180, 25);

userid=**new** JTextField();

userid.setBounds(200, 80, 180, 25);

passwords=**new** JPasswordField();

passwords.setBounds(200, 160, 180, 25);

((JPasswordField) passwords).setEchoChar('\*');

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

lrole.setBounds(50, 200, 180, 25);

roleCombo=**new** JComboBox(role);

roleCombo.setBounds(200,200,90,20);

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

bLogin.setBounds(100, 250,180, 25);

bLogin.addActionListener(**this**);

bregister=**new** JButton("Register");

bregister.setBounds(250, 250,180, 25);

lMessage=**new** JLabel();

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

*dialog*=**new** JDialog(**this**,"Login",**true**);

add(lTitle);

add(lUser);

add(lPassword);

add(lrole);

add(roleCombo);

add(userid);

add(passwords);

add(bLogin);

add(bregister);

add(lMessage);

*dialog*.add(lMessage);

*dialog*.setSize(350, 200);

setSize(500,500);

setLayout(**null**);

setVisible(**true**);

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

bregister.addActionListener(**new** ActionListener()

{

**public** **void** actionPerformed(ActionEvent e)

{

**new** RegistrationFrame();

}

});

}

@Override

**public** **void** actionPerformed(ActionEvent e)

{

**if**(e.getSource()==bLogin) {

**try** {

//PreparedStatement ps;

ResultSet rs;

String query = "SELECT \* FROM Employees WHERE Userid =? AND Pass =? AND Role=? And Active='TRUE'";

String url="jdbc:sqlserver:DESKTOP-CJ512VJ\\INSTANCE01:1433;databaseName=PCSDB;user=sa1;password=preethu";

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

Connection conn=DriverManager.*getConnection*(url);

PreparedStatement ps1=conn.prepareStatement(query);

ps1.setString(1, userid.getText());

ps1.setString(2, passwords.getText());

ps1.setString(3, roleCombo.getSelectedItem().toString());

rs = ps1.executeQuery();

**if**(rs.next())

{

**if**(roleCombo.getSelectedItem().equals("HRA"))

{

HRWindow hra = **new** HRWindow();

**this**.dispose();

}

**if**(roleCombo.getSelectedItem().equals("PME"))

{

PMWindow pme = **new** PMWindow();

**this**.dispose();

}

**if**(roleCombo.getSelectedItem().equals("EMP"))

{

Employee emp = **new** Employee();

**this**.dispose();

}

}

**else**

{

JOptionPane.*showMessageDialog*(**null**, "Incorrect Username Or Password", "Login Failed", 2);

}

}

**catch** (SQLException ex) {

Logger.*getLogger*(LoginFrame.**class**.getName()).log(Level.***SEVERE***, **null**, ex);

} **catch** (ClassNotFoundException e1) {

// **TODO** Auto-generated catch block

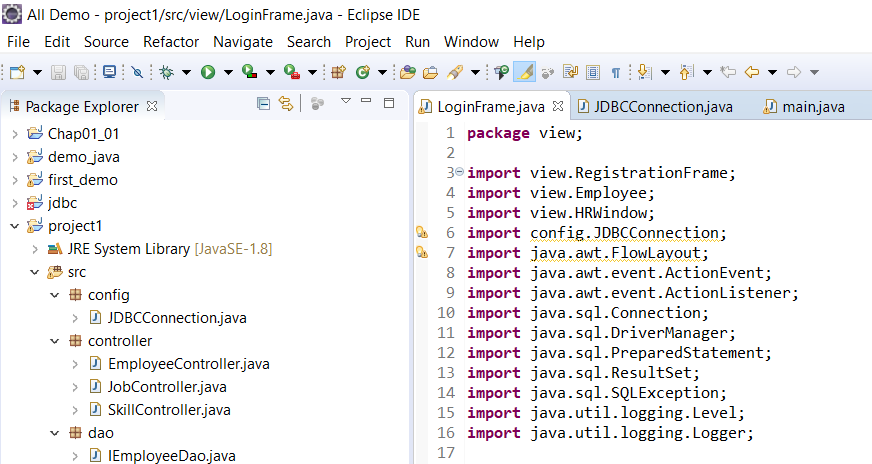
e1.printStackTrace();

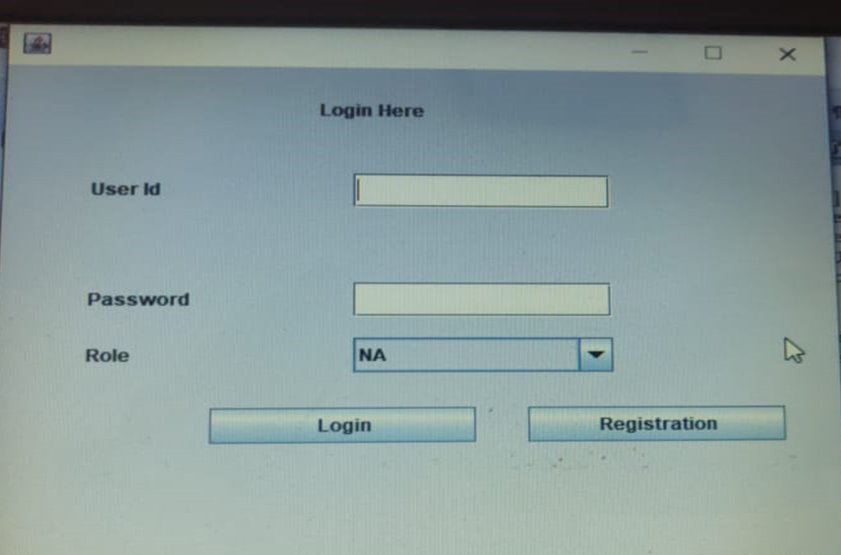
}

}

}

}





**\\main.java**

**package** view;

**public** **class** main{

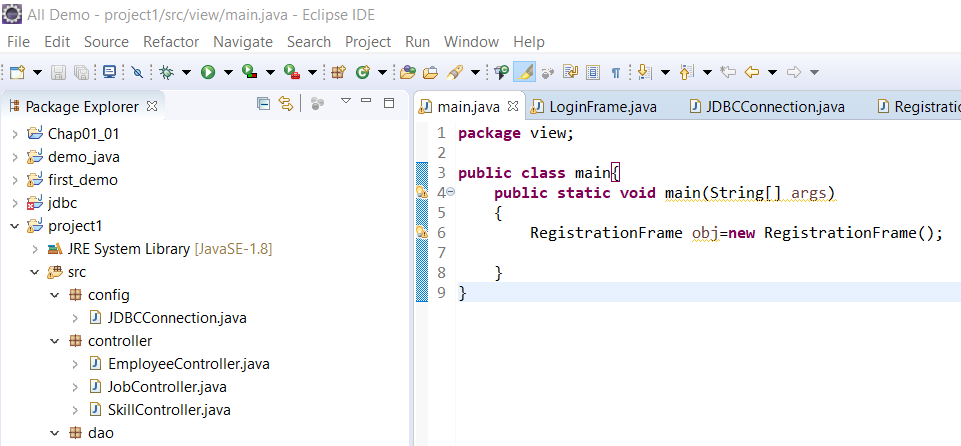
**public** **static** **void** main(String[] args)

{

RegistrationFrame obj=**new** RegistrationFrame();

}

}



**\\RegistrationFrame.java**

package view;

import javax.swing.\*;

import java.awt.event.ActionListener;

import java.awt.event.\*;

import java.awt.\*;

import java.sql.\*;

public class RegistrationFrame implements ActionListener {

JFrame frame;

String[] gender={"M","F"};

String[] role= {"HRA","EMP","PME"};

JLabel EmailId=new JLabel("Email Id");

JLabel FirstName=new JLabel("First Name");

JLabel LastName=new JLabel("Last Name");

JLabel Gender=new JLabel("Gender");

JLabel UserId=new JLabel("User Id");

JLabel Password=new JLabel("Password");

JLabel Role=new JLabel("Role");

JLabel Mobileno=new JLabel("Mobile No");

JTextField Emailid=new JTextField();

JTextField firstname=new JTextField();

JComboBox genderCombo=new JComboBox(gender);

JTextField lastname=new JTextField();

JTextField mobileno=new JTextField();

JTextField userid=new JTextField();

JPasswordField passwords=new JPasswordField();

JComboBox roleCombo=new JComboBox(role);

JButton registerButton=new JButton("REGISTER");

JButton resetButton=new JButton("RESET");

public RegistrationFrame()

{

createWindow();

setLocationAndSize();

addComponentsToFrame();

actionEvent();

}

public void createWindow()

{

frame=new JFrame();

frame.setTitle("Registration Form");

frame.setBounds(40,40,380,600);

frame.getContentPane().setBackground(Color.white);

frame.getContentPane().setLayout(null);

frame.setVisible(true);

//frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setResizable(false);

}

public void setLocationAndSize()

{

EmailId.setBounds(20,20,60,70);

FirstName.setBounds(20,70,100,70);

LastName.setBounds(20,120,100,70);

UserId.setBounds(20,170,100,70);

Password.setBounds(20,220,140,70);

Role.setBounds(20,270,100,70);

Gender.setBounds(20,320,100,70);

Mobileno.setBounds(20,370,100,70);

Emailid.setBounds(180,43,165,23);

firstname.setBounds(180,93,165,23);

lastname.setBounds(180,143,165,23);

userid.setBounds(180,193,165,23);

passwords.setBounds(180,243,165,23);

roleCombo.setBounds(180,293,165,23);

genderCombo.setBounds(180,343,165,23);

mobileno.setBounds(180,393,165,23);

registerButton.setBounds(70,450,100,35);

resetButton.setBounds(220,450,100,35);

}

public void addComponentsToFrame()

{

frame.add(EmailId);

frame.add(FirstName);

frame.add(LastName);

frame.add(UserId);

frame.add(Password);

frame.add(Role);

frame.add(Gender);

frame.add(Mobileno);

frame.add(Emailid);

frame.add(firstname);

frame.add(lastname);

frame.add(userid);

frame.add(passwords);

frame.add(roleCombo);

frame.add(genderCombo);

frame.add(mobileno);

frame.add(registerButton);

frame.add(resetButton);

}

public void actionEvent()

{

registerButton.addActionListener(this);

resetButton.addActionListener(this);

}

/\*\*

\*

\*/

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==registerButton)

{

try {

Connection conn=DriverManager.getConnection("jdbc:sqlserver:DESKTOP-CJ512VJ\\INSTANCE01;databaseName=pcsdb","sa1","preethu");

//Preapared Statement

PreparedStatement Pstatement=conn.prepareStatement("insert into Employees values(?,?,?,?,?,?,?,?,?)");

//Specifying the values of it's parameter

Pstatement.setString(1,Emailid.getText());

Pstatement.setString(2,firstname.getText());

Pstatement.setString(3,lastname.getText());

Pstatement.setString(4,userid.getText());

Pstatement.setString(5,passwords.getText());

Pstatement.setString(6,roleCombo.getSelectedItem().toString());

Pstatement.setString(7,genderCombo.getSelectedItem().toString());

Pstatement.setString(8,"FALSE");

Pstatement.setString(9,mobileno.getText());

Pstatement.executeUpdate();

Statement stmt = conn.createStatement();

String sql = "UPDATE Employees SET Active='TRUE' WHERE Role='HRA'";

stmt.executeUpdate(sql);

JOptionPane.showMessageDialog(null,"Data Registered Successfully");

}

catch (SQLException e1) {

e1.printStackTrace();

}

}

if(e.getSource()==resetButton)

{

//Clearing Fields

Emailid.setText("");

firstname.setText("");

lastname.setText("");

userid.setText("");

passwords.setText("");

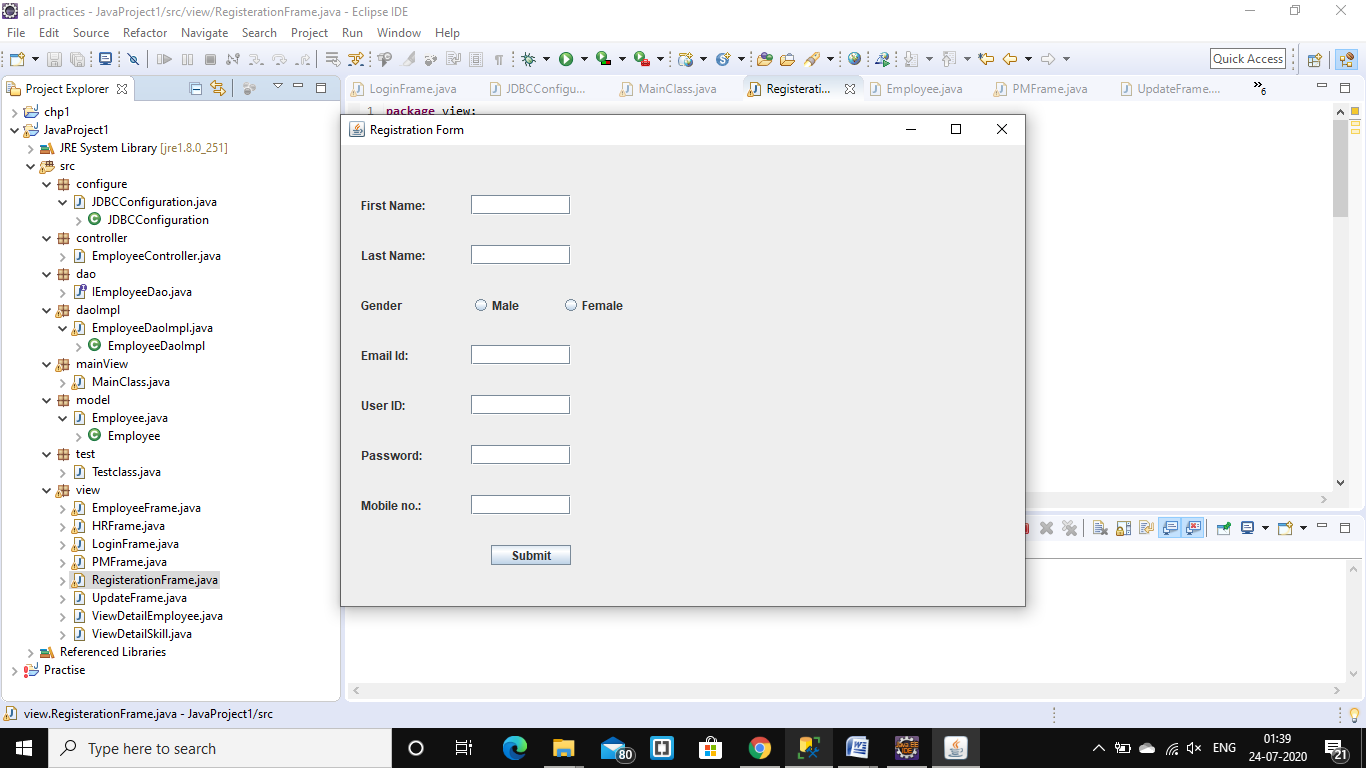
roleCombo.setSelectedItem("");

genderCombo.setSelectedItem("");

mobileno.setText("");

}

}



**\\Emplyee.java**

package view;

import java.awt.FlowLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

public class Employee implements ActionListener{

JFrame frame = new JFrame("Employee Window");

JButton Viewdata,Viewjob, Changepass,Logout;

public Employee(){

createwindow();

addtoframe();

actionEvent();

}

public void createwindow() {

Viewdata= new JButton("View Data");

Viewjob= new JButton("View Job");

Changepass= new JButton("Change Password");

Logout= new JButton("Log Out");

}

public void addtoframe() {

frame.add(Viewdata);

frame.add(Viewjob);

frame.add(Changepass);

frame.add(Logout);

frame.setLayout(new FlowLayout());

frame.setSize(300,300);

frame.setVisible(true);

}

public void actionEvent()

{

Viewdata.addActionListener(this);

Viewjob.addActionListener(this);

Changepass.addActionListener(this);

Logout.addActionListener(this);

}

public void actionPerformed(ActionEvent e) {

if(e.getSource()==Viewdata)

{

}

if(e.getSource()==Viewjob)

{

}

if(e.getSource()==Changepass)

{

}

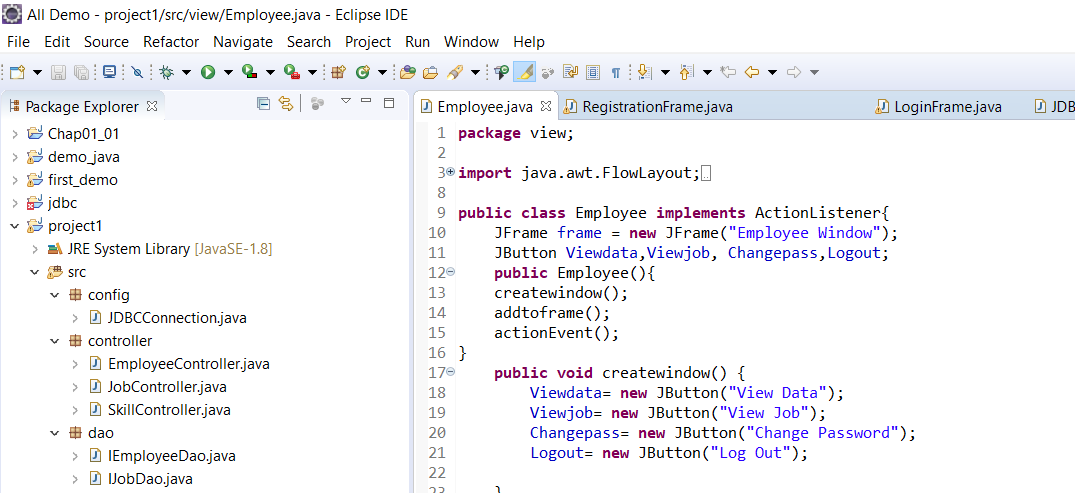
if(e.getSource()==Logout)

{

}

}

}



**\\HRWindow.java**

package view;

import java.awt.FlowLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

public class HRWindow implements ActionListener {

JFrame frame = new JFrame("HR Window");

JButton Activate,Deactivate, Changepass,Logout;

public HRWindow(){

createwindow();

addtoframe();

actionEvent();

}

public void createwindow() {

Activate= new JButton("Activate Employee");

Deactivate= new JButton("Deactivate Employee");

Changepass= new JButton("Change Password");

Logout= new JButton("Log Out");

}

public void addtoframe() {

frame.add(Activate);

frame.add(Deactivate);

frame.add(Changepass);

frame.add(Logout);

frame.setLayout(new FlowLayout());

frame.setSize(300,300);

frame.setVisible(true);

}

public void actionEvent()

{

Activate.addActionListener(this);

Deactivate.addActionListener(this);

Changepass.addActionListener(this);

Logout.addActionListener(this);

}

public void actionPerformed(ActionEvent e) {

if(e.getSource()==Activate)

{

}

if(e.getSource()==Deactivate)

{

}

if(e.getSource()==Changepass)

{

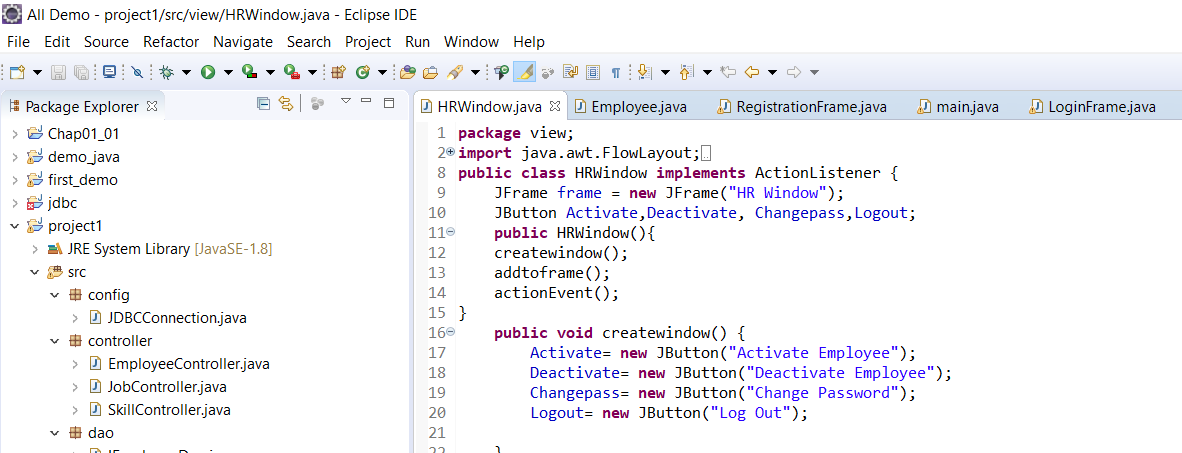
}

if(e.getSource()==Logout)

{

}

}



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

**package** view;

**import** java.awt.FlowLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**public** **class** PMWindow **implements** ActionListener {

JFrame frame = **new** JFrame("Project Manager Window");

JButton Viewdata,Addjob, Changepass,Logout;

**public** **void** ProjectManager(){

createwindow();

addtoframe();

actionEvent();

}

**public** **void** createwindow() {

Viewdata= **new** JButton("View Data");

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

Changepass= **new** JButton("Change Password");

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

}

**public** **void** addtoframe() {

frame.add(Viewdata);

frame.add(Addjob);

frame.add(Changepass);

frame.add(Logout);

frame.setLayout(**new** FlowLayout());

frame.setSize(300,300);

frame.setVisible(**true**);

}

**public** **void** actionEvent()

{

Viewdata.addActionListener(**this**);

Addjob.addActionListener(**this**);

Changepass.addActionListener(**this**);

Logout.addActionListener(**this**);

}

**public** **void** actionPerformed(ActionEvent e) {

**if**(e.getSource()==Viewdata)

{

}

**if**(e.getSource()==Addjob)

{

}

**if**(e.getSource()==Changepass)

{

}

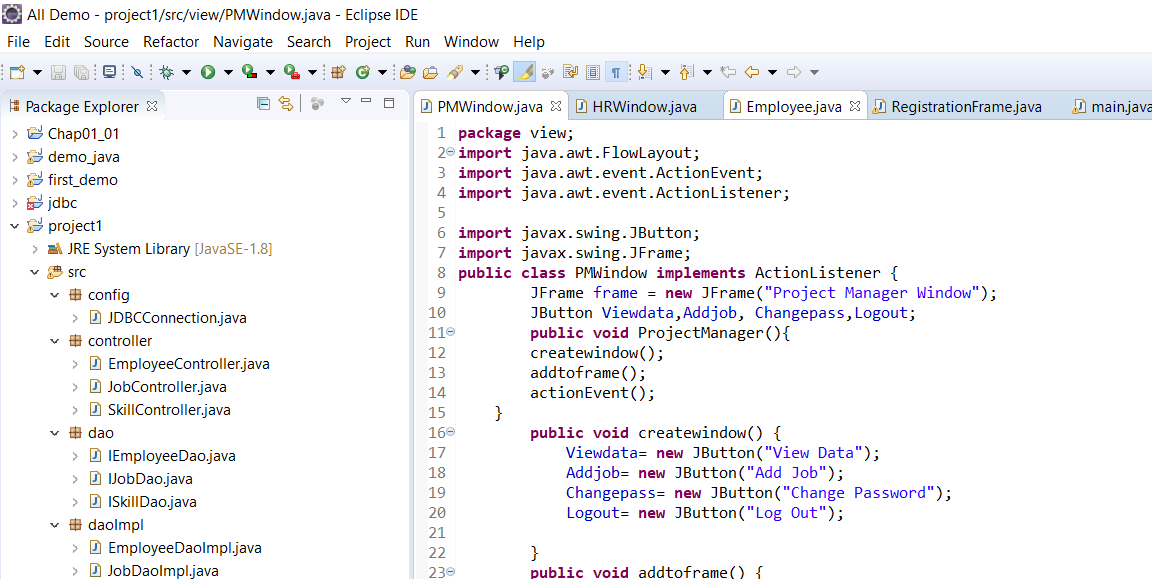
**if**(e.getSource()==Logout)

{

}

}

}



**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/>

**THANK YOU!!!**

‘