

**CITY UNIVERSITY**

Faculty of Science and Engineering

Department of Computer Science and Engineering

Fall 2021

**CSE 312 Database Management System (DBMS)**

**Lab Report No**: 04

**Topic name**: Login and Registration page with data connection

|  |  |
| --- | --- |
| ***Submitted To***  Khatuney Jannat Mafruza Mitu  Lecture, Dept. of CSE  City University | ***Submitted By***  Hasibul Islam  1935202062 |

|  |  |  |
| --- | --- | --- |
|  |  |  |

Date of Submissions: 2022-01-26

# Login & Registration page with data connection

Here, we are using `Java` as a language. By using that we create `GUI Form` in which we approach `Swing UI Designer`.

## What is `Java Swing`?

Swing is a part of the JFC (Java Foundation Classes). Building Graphical User Interface in Java requires the use of Swings. Swing Framework contains a large set of components which allow a high level of customization and provide rich functionalities, and is used to create window-based applications. Java swing components are lightweight, platform-independent, provide powerful components like tables, scroll panels, buttons, list, color chooser, etc.

## What is `Swing UI Designer`?

SWT/Swing Designer is **a Java GUI designer** that makes it very easy to create Java GUI applications without spending a lot of time writing code to display simple forms.

By using `Java Swing` we’ve created a ‘login form and a registration’ form given below:

### **A Data Connected ` Form`:**

We’ve created a ‘Login Form’ with few steps which given below:

1. Create a `Swing file` through right click from ‘src’ folder
2. Download ‘FX Kit’ for design tools and ‘Maven Kit’ for “GUI to Java Code” transform
3. Connect ‘JDBC’ .jar file with existing project and

**A Login Form Below:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.\*;

public class LoginForm extends JDialog {

private JTextField tfEmail;

private JPasswordField pfPassword;

private JButton btnOK;

private JButton btnCancel;

private JPanel loginPanel;

public LoginForm(JFrame parent) {

super(parent);

setTitle("Login");

setContentPane(loginPanel);

setMinimumSize(new Dimension(450, 474));

setModal(true);

setLocationRelativeTo(parent);

setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

btnOK.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

String email = tfEmail.getText();

String password = String.valueOf(pfPassword.getPassword());

user = getAuthenticatedUser(email, password);

if (user != null) {

dispose();

}

else {

JOptionPane.showMessageDialog(LoginForm.this,

"Email or Password Invalid",

"Try again",

JOptionPane.ERROR\_MESSAGE);

}

}

});

btnCancel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

dispose();

}

});

setVisible(true);

}

public User user;

private User getAuthenticatedUser(String email, String password) {

User user = null;

final String DB\_URL = "jdbc:mysql://localhost/MyStore?serverTimezone=UTC";

final String USERNAME = "root";

final String PASSWORD = "";

try{

Connection conn = DriverManager.getConnection(DB\_URL, USERNAME, PASSWORD);

// Connected to database successfully...

Statement stmt = conn.createStatement();

String sql = "SELECT \* FROM users WHERE email=? AND password=?";

PreparedStatement preparedStatement = conn.prepareStatement(sql);

preparedStatement.setString(1, email);

preparedStatement.setString(2, password);

ResultSet resultSet = preparedStatement.executeQuery();

if (resultSet.next()) {

user = new User();

user.name = resultSet.getString("name");

user.email = resultSet.getString("email");

user.phone = resultSet.getString("phone");

user.address = resultSet.getString("address");

user.password = resultSet.getString("password");

}

stmt.close();

conn.close();

}catch(Exception e){

e.printStackTrace();

}

return user;

}

public static void main(String[] args) {

LoginForm loginForm = new LoginForm(null);

User user = loginForm.user;

if (user != null) {

System.out.println("Successful Authentication of: " + user.name);

System.out.println(" Email: " + user.email);

System.out.println(" Phone: " + user.phone);

System.out.println(" Address: " + user.address);

}

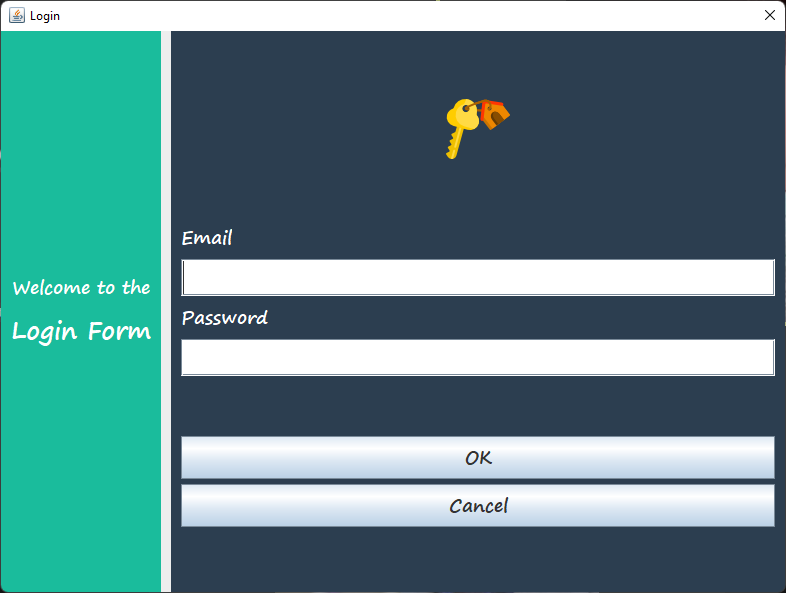
else {

System.out.println("Authentication canceled");

}

}

}



**Fig: 4.1, A Login Form**

**A Registration Form below:**

import javax.swing.\*;

import java.awt.\*;

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.Statement;

public class RegistrationForm extends JDialog {

private JTextField tfName;

private JTextField tfEmail;

private JTextField tfPhone;

private JTextField tfAddress;

private JPasswordField pfPassword;

private JPasswordField pfConfirmPassword;

private JButton btnRegister;

private JButton btnCancel;

private JPanel registerPanel;

public RegistrationForm(JFrame parent) {

super(parent);

setTitle("Create a new account");

setContentPane(registerPanel);

setMinimumSize(new Dimension(450, 474));

setModal(true);

setLocationRelativeTo(parent);

setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

btnRegister.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

registerUser();

}

});

btnCancel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

dispose();

}

});

setVisible(true);

}

private void registerUser() {

String name = tfName.getText();

String email = tfEmail.getText();

String phone = tfPhone.getText();

String address = tfAddress.getText();

String password = String.valueOf(pfPassword.getPassword());

String confirmPassword = String.valueOf(pfConfirmPassword.getPassword());

if (name.isEmpty() || email.isEmpty() || phone.isEmpty() || address.isEmpty() || password.isEmpty()) {

JOptionPane.showMessageDialog(this,

"Please enter all fields",

"Try again",

JOptionPane.ERROR\_MESSAGE);

return;

}

if (!password.equals(confirmPassword)) {

JOptionPane.showMessageDialog(this,

"Confirm Password does not match",

"Try again",

JOptionPane.ERROR\_MESSAGE);

return;

}

user = addUserToDatabase(name, email, phone, address, password);

if (user != null) {

dispose();

}

else {

JOptionPane.showMessageDialog(this,

"Failed to register new user",

"Try again",

JOptionPane.ERROR\_MESSAGE);

}

}

public User user;

private User addUserToDatabase(String name, String email, String phone, String address, String password) {

User user = null;

final String DB\_URL = "jdbc:mysql://localhost/MyStore?serverTimezone=UTC";

final String USERNAME = "root";

final String PASSWORD = "";

try{

Connection conn = DriverManager.getConnection(DB\_URL, USERNAME, PASSWORD);

// Connected to database successfully...

Statement stmt = conn.createStatement();

String sql = "INSERT INTO users (name, email, phone, address, password) " +

"VALUES (?, ?, ?, ?, ?)";

PreparedStatement preparedStatement = conn.prepareStatement(sql);

preparedStatement.setString(1, name);

preparedStatement.setString(2, email);

preparedStatement.setString(3, phone);

preparedStatement.setString(4, address);

preparedStatement.setString(5, password);

//Insert row into the table

int addedRows = preparedStatement.executeUpdate();

if (addedRows > 0) {

user = new User();

user.name = name;

user.email = email;

user.phone = phone;

user.address = address;

user.password = password;

}

stmt.close();

conn.close();

}catch(Exception e){

e.printStackTrace();

}

return user;

}

public static void main(String[] args) {

RegistrationForm myForm = new RegistrationForm(null);

User user = myForm.user;

if (user != null) {

System.out.println("Successful registration of: " + user.name);

}

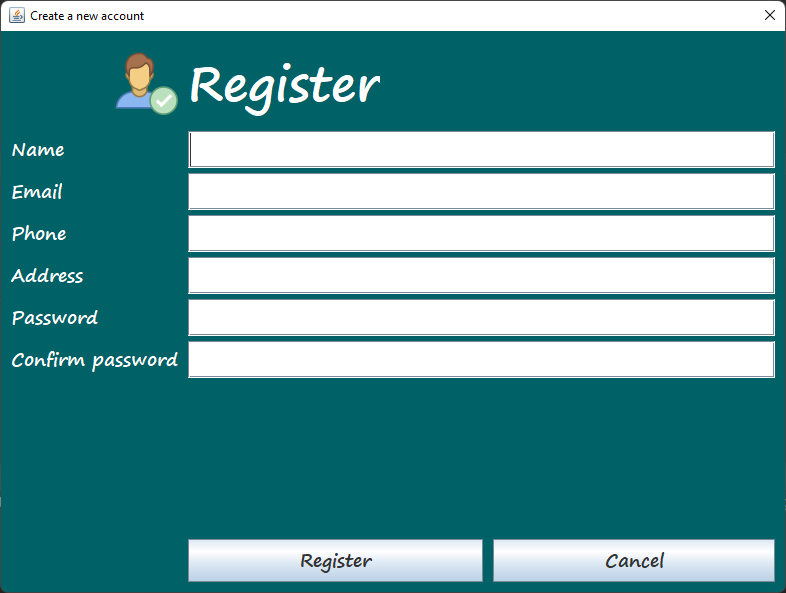
else {

System.out.println("Registration canceled");

}

}

}



**Fig: 4.2, Registration Form**