import javax.swing.\*; import java.awt.\*; import java.awt.event.\*; import java.sql.\*;

public class LoginApp extends JFrame { private JTextField txtUsername; private JPasswordField txtPassword; private JButton btnLogin, btnClear, btnRegister; private Connection conn;

public LoginApp() { setTitle("Login Form"); setSize(400, 250); setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); setLocationRelativeTo(null);

// Panel

JPanel panel = new JPanel(new GridLayout(4, 2, 10, 10));

JLabel lblUsername = new JLabel("Username:"); JLabel lblPassword = new JLabel("Password:");

txtUsername = new JTextField(); txtPassword = new JPasswordField(); btnLogin = new JButton("Login"); btnClear = new JButton("Clear"); btnRegister = new JButton("Register");

panel.add(lblUsername); panel.add(txtUsername); panel.add(lblPassword); panel.add(txtPassword); panel.add(btnLogin); panel.add(btnClear); panel.add(btnRegister); add(panel);

// Connect DB connectDB();

// Login button btnLogin.addActionListener(e -> loginUser());

// Clear button btnClear.addActionListener(e -> { txtUsername.setText(""); txtPassword.setText(""); });

// Register button

btnRegister.addActionListener(e -> { new RegisterForm().setVisible(true); }); }

private void connectDB() { try {

conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/yourDB", "yourUser", System.out.println("Database connected.");

} catch (SQLException ex) {

JOptionPane.showMessageDialog(this, "DB Connection Failed: " + ex.getMessage());

} }

private void loginUser() { String username = txtUsername.getText();

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

try { String sql = "SELECT role FROM users WHERE username=? AND password=?"; PreparedStatement pst = conn.prepareStatement(sql); pst.setString(1, username); pst.setString(2, password); ResultSet rs = pst.executeQuery();

if (rs.next()) { String role = rs.getString("role");

JOptionPane.showMessageDialog(this, "Login successful! Role: " + role);

// Open new JFrame with welcome message new WelcomeFrame(username).setVisible(true); this.dispose(); } else {

JOptionPane.showMessageDialog(this, "Invalid username or password.");

}

} catch (SQLException ex) { ex.printStackTrace(); }

}

// Main

public static void main(String[] args) { SwingUtilities.invokeLater(() -> new LoginApp().setVisible(true));

}

}

// Separate JFrame for Welcome Message class WelcomeFrame extends JFrame { public WelcomeFrame(String username) { setTitle("Welcome"); setSize(300, 150); setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE); setLocationRelativeTo(null);

JLabel lblWelcome = new JLabel("Welcome, " + username + "!", SwingConstants.CENTER); add(lblWelcome); }

class RegisterForm extends JFrame { private JTextField txtUsername; private JPasswordField txtPassword; private JComboBox<String> cmbRole; private JButton btnRegister; private Connection conn;

public RegisterForm() { setTitle("Register New User"); setSize(400, 250); setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE); setLocationRelativeTo(null);

JPanel panel = new JPanel(new GridLayout(4, 2, 10, 10));

JLabel lblUsername = new JLabel("Username:");

JLabel lblPassword = new JLabel("Password:"); JLabel lblRole = new JLabel("Role:");

txtUsername = new JTextField(); txtPassword = new JPasswordField(); cmbRole = new JComboBox<>(new String[]{"admin", "student"}); btnRegister = new JButton("Register");

panel.add(lblUsername); panel.add(txtUsername); panel.add(lblPassword); panel.add(txtPassword); panel.add(lblRole); panel.add(cmbRole); panel.add(new JLabel()); panel.add(btnRegister); add(panel);

try { conn = DriverManager.getConnection("jdbc:postgresql://localhost:5432/yourDB", "yourUser", } catch (SQLException e) {

JOptionPane.showMessageDialog(this, "DB connection failed: " + e.getMessage()); }

btnRegister.addActionListener(e -> registerUser()); }

private void registerUser() { String username = txtUsername.getText();

String password = String.valueOf(txtPassword.getPassword()); String role = (String) cmbRole.getSelectedItem();

try { String sql = "INSERT INTO users (username, password, role) VALUES (?, ?, ?)"; PreparedStatement pst = conn.prepareStatement(sql); pst.setString(1, username); pst.setString(2, password); pst.setString(3, role); pst.executeUpdate();

JOptionPane.showMessageDialog(this, "User registered successfully!"); this.dispose(); } catch (SQLException ex) {

JOptionPane.showMessageDialog(this, "Error: " + ex.getMessage());

}

}

}