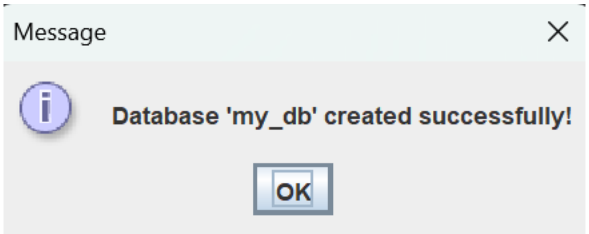
Lab 7



Database 'my_db' created successfully!

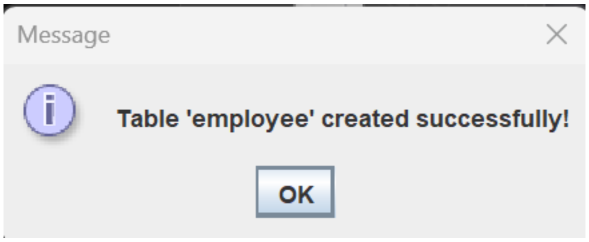
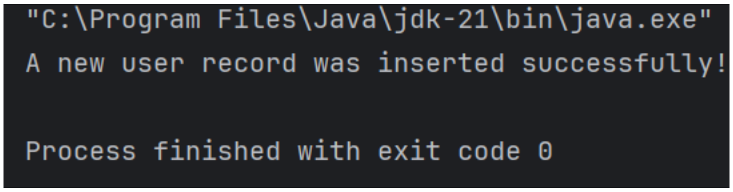
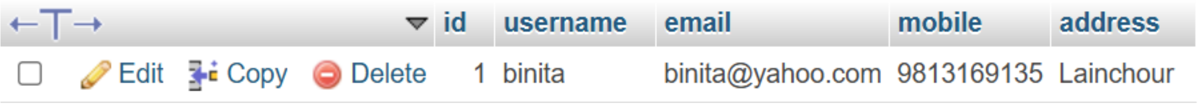


Table 'employee' created successfully!

Lab 8



```
"C:\Program Files\Java\jdk-21\bin\java.exe"
A new user record was inserted successfully!

Process finished with exit code 0
```



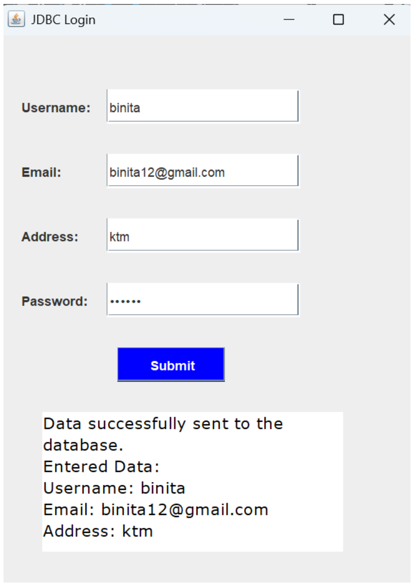
←T→	id	username	email	mobile	address
□ Edit Copy Delete	1	binita	binita@yahoo.com	9813169135	Lainchour

CODE

```
package JDBC_Program;

import java.sql.*;
class JDBCSwing {
    public JDBCSwing(String Username, String Address, String Email, String Password) {
        String url = "jdbc:mysql://localhost:3306/my_db";  // Database URL
        String user = "root";  // Database username
        String dbPassword = "";  // Database password
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection connection = DriverManager.getConnection(url, user, dbPassword);
            String checkSql = "SELECT * FROM employee WHERE username = ?";
            PreparedStatement psCheck = connection.prepareStatement(checkSql);
            psCheck.setString(1, Username);
            ResultSet rs = psCheck.executeQuery();

            if (rs.next()) {
                System.out.println("User already exists.");
            } else {
                // SQL query for inserting the user data into the database
                String sql = "INSERT INTO employee (username, email, address, password) VALUES
(?, ?, ?, ?)";
                PreparedStatement ps = connection.prepareStatement(sql);
                ps.setString(1, Username);
                ps.setString(2, Email);
                ps.setString(3, Address);
                ps.setString(4, Password);
                int rows = ps.executeUpdate();
                if (rows > 0) {
                    System.out.println("Data sent to database.");
                }
                ps.close();
            }
            rs.close();
            connection.close();

        } catch (ClassNotFoundException e) {
            System.out.println("JDBC Driver not found: " + e.getMessage());
        } catch (SQLException e) {
            System.out.println("Database error: " + e.getMessage());
        }
    }
}

class LoginForm extends JFrame {
    JLabel name = new JLabel("Username: ");
    JTextField nameField = new JTextField();
    JLabel email = new JLabel("Email: ");
    JTextField emailField = new JTextField();
    JLabel address = new JLabel("Address: ");
    JTextField addField = new JTextField();
    JLabel password = new JLabel("Password: ");
    JPasswordField passField = new JPasswordField();
    JButton btnSubmit = new JButton("Submit");
    JTextArea outputText = new JTextArea("Status: ");

    public LoginForm() {
        setLayout(null);
        setTitle("JDBC Login");
```

```
setSize(400, 550);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLocationRelativeTo(null);
name.setBounds(20, 50, 80, 32);
nameField.setBounds(100, 50, 180, 32);
email.setBounds(20, 110, 80, 32);
emailField.setBounds(100, 110, 180, 32);
address.setBounds(20, 170, 120, 32);
addField.setBounds(100, 170, 180, 32);
password.setBounds(20, 230, 120, 32);
passField.setBounds(100, 230, 180, 32);
btnSubmit.setBounds(110, 290, 100, 32);
btnSubmit.setBackground(Color.blue);
btnSubmit.setForeground(Color.WHITE);
btnSubmit.setFocusPainted(false);
outputText.setBounds(40, 350, 280, 130);
outputText.setEditable(false);
outputText.setLineWrap(true);
outputText.setWrapStyleWord(true);
outputText.setFont(new Font("Verdana", Font.PLAIN, 15));
add(name);
add(email);
add(address);
add(password);
add(btnSubmit);
add(outputText);
add(nameField);
add(emailField);
add(addField);
add(passField);
btnSubmit.addActionListener(new ActionListener() {
      @Override
      public void actionPerformed(ActionEvent e) {

        try {
          // Get values from text fields
          String Username = nameField.getText().trim();
          String Email = emailField.getText().trim();
          String Address = addField.getText().trim();
          String Password = new String(passField.getPassword()).trim();
          if (Username.isEmpty() || Email.isEmpty() || Address.isEmpty() ||
Password.isEmpty()) {
            outputText.setForeground(Color.RED);
            outputText.setText("Please ensure no field is empty.");
          } else if (!isValidEmail(Email)) {
            outputText.setForeground(Color.RED);
            outputText.setText("Invalid email format.");
          } else {
            new JDBCSwing(Username, Address, Email, Password);
            outputText.setForeground(Color.BLACK);
            outputText.setText("Data successfully sent to the database.\nEntered
Data:\nUsername: " + Username + "\nEmail: " + Email + "\nAddress: " + Address);
            System.out.println("Entered data:\nUsername: " + Username + "\nEmail: " +
Email + "\nAddress: " + Address);
          }

        } catch (Exception ex) {
          System.out.println("Error on submission: " + ex.getMessage());
        }
      }
    });
  }
  private boolean isValidEmail(String email) {
    String emailPattern = "^[a-zA-Z0-9_+&-]+(?:\\.[a-zA-Z0-9_+&-]+)*@(?:[a-zA-Z0-9-
]+\\.)+[a-zA-Z]{2,7}$";
    return Pattern.matches(emailPattern, email);
  }
  public static void main(String[] args) {
    LoginForm loginForm = new LoginForm();
    loginForm.setVisible(true);
  }
}
```

OUTPUT



Username: binita

Email: binita12@gmail.com

Address: ktm

Password: ••••••

```
Data successfully sent to the
database.
Entered Data:
Username: binita
Email: binita12@gmail.com
Address: ktm
```

CONCLUSION

This program demonstrates how to create a simple login page using Java Swing for the graphical user interface and JDBC to interact with a MySQL database for user authentication. It validates the user credentials entered on the Swing form by checking them against the table in the MySQL database. The program provides feedback to the user based on whether the login was successful or not.

CODE