Name – Prerna Sunil Jadhav   **SAP ID - 60004220127**

Experiment No – 14

AIM: Designing Graphical User Interfaces in Java using AWT and Event handling (CO6)

THEORY:
Java AWT (Abstract Window Toolkit) is an API to develop Graphical User Interface (GUI) or windows-based applications in Java. Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavy weight i.e. its components are using the resources of underlying operating system (OS). The java.awt package provides classes for AWT API such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List etc. The AWT tutorial will help the user to understand Java GUI programming in simple and easy steps.

Code (i): Write java program to create a registration form using AWT..

```java
J Code1_Registration.java 1 ×

Exp14 > J Code1_Registration.java > Code1_Registration > Cod
1    package Exp14;
2    import java.awt.*;
3    import javax.swing.*;
4    import java.awt.event.*;
5    class Code1_Registration extends JFrame implements ActionListener {
6        Container c;
7        JLabel lFName, lLName, lPassword, lEmail;
8        JTextField txtFName, txtLName;
9        JTextField txtEmail;
10       JPasswordField txtPassword;
11       JButton btnSubmit, btnClear, btnExit;
12       String strFName, strLName, strPassword, strEmail;
13       Code1_Registration() {
14           c = getContentPane();
15           c.setLayout(new FlowLayout());
16           lFName = new JLabel(text: "First Name: ");
17           lLName = new JLabel(text: "Last Name: ");
18           lPassword = new JLabel(text: "Password: ");
19           lEmail = new JLabel(text: "Email: ");
20           txtFName = new JTextField(columns: 10);
21           txtLName = new JTextField(columns: 10);
22           txtEmail = new JTextField(columns: 10);
23           txtPassword = new JPasswordField(columns: 10);
24           txtPassword.setEchoChar(c: '*');
25           btnSubmit = new JButton(text: "Submit");
26           btnClear = new JButton(text: "Clear");
27           btnExit = new JButton(text: "Exit");
28           c.add(lFName);
29           c.add(txtFName);
30           c.add(lLName);
31           c.add(txtLName);
```
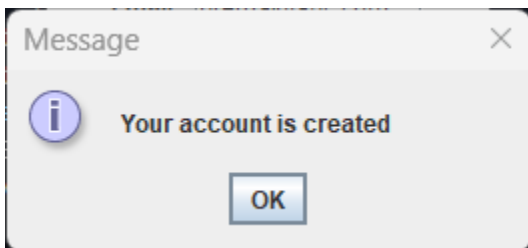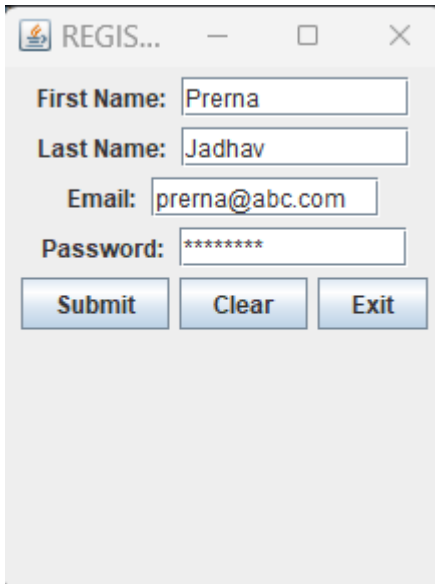
Exp14 > **J** Code1_Registration.java > **⏣** Code1_Registration > **⬡** actionPerformed(Act

```java
32              c.add(lEmail);
33              c.add(txtEmail);
34              c.add(lPassword);
35              c.add(txtPassword);
36              c.add(btnSubmit);
37              c.add(btnClear);
38              c.add(btnExit);
39              btnSubmit.addActionListener(this);
40              btnClear.addActionListener(this);
41              btnExit.addActionListener(this);
42          }
43          public void actionPerformed(ActionEvent ae) {
44              if (ae.getSource() == btnSubmit) {
45                  strLName = txtLName.getText();
46                  strFName = txtFName.getText();
47                  strPassword = txtPassword.getText();
48                  strEmail = txtEmail.getText();
49                  if (strFName.equals(anObject: "") || strLName.equals(anObject: "") ||
50                          strPassword.equals(anObject: "") ||
51                          strEmail.equals(anObject: "")) {
52                      JOptionPane.showMessageDialog(c, message: "Input field is empty!!");
53                      txtFName.requestFocus();
54                  } else {
55                      JOptionPane.showMessageDialog(c, message: "Your account is created ");
56                      txtFName.setText(t: "");
57                      txtPassword.setText(t: "");
58                      System.exit(status: 0);
59                  }
60              } else if (ae.getSource() == btnClear) {
61                  txtFName.setText(t: "");
62                  txtPassword.setText(t: "");
63                  txtLName.setText(t: "");
64                  txtEmail.setText(t: "");
65                  txtFName.requestFocus();
66              } else {
67
68                  System.exit(status: 0);
69              }
70          }
            Run | Debug
71 ∨       public static void main(String z[]) {
72              //Prerna Sunil Jadhav - 60004221027
73              Code1_Registration frm = new Code1_Registration();
74              frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
75              frm.setBounds(x: 200, y: 200, width: 400, height: 300);
76              frm.setVisible(b: true);
77              frm.setTitle(title: "REGISTER");
78          }
79      }
```

OUTPUT:

THEORY:

In this program we have created a login form where used gets logged in if he enters the correct username and password. After OK button is pressed the username and password is checked, if it matches the actual credentials then used successfully logs in or else it's an unsuccessful login.

Components Used:
• JFrame
• JTextField
• JPasswordField
• JButton

Code (ii): On Applet: Take a Login and Password from the user and display it on the third Text Field which appears only on clicking OK button and clear both the Text Fields on clicking RESET button

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Academic Year: 2022-2023

J Code2_Login.java 1 ✕
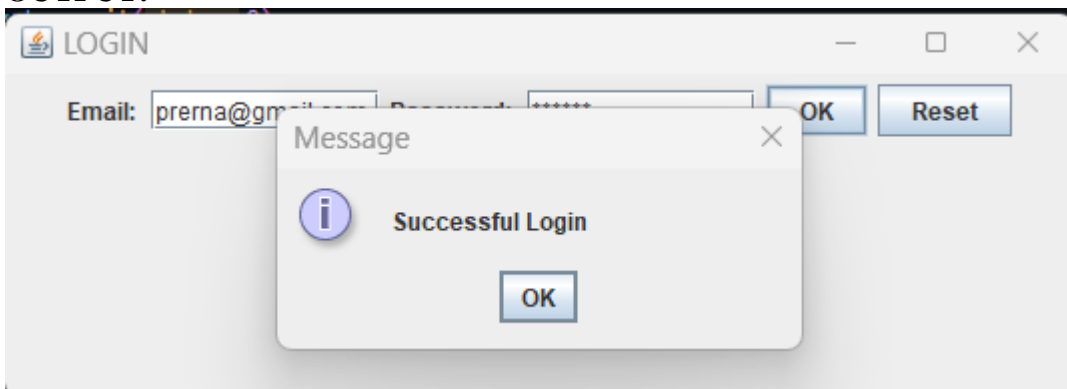
Exp14 > J Code2_Login.java > ⚛ Code2_Login > ⬡ actionPerfor

```java
package Exp14;

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

class Code2_Login extends JFrame implements ActionListener {
    Container c;
    JLabel lPassword, lEmail;
    JTextField txtEmail;
    JPasswordField txtPassword;
    JButton btnSubmit, btnClear;
    String strPassword, strEmail;

    Code2_Login() {
        c = getContentPane();
        c.setLayout(new FlowLayout());
        // labels
        lPassword = new JLabel(text: "Password: ");
        lEmail = new JLabel(text: "Email: ");
        // text fields
        txtEmail = new JTextField(columns: 10);
        txtPassword = new JPasswordField(columns: 10);
        txtPassword.setEchoChar(c: '*');
        // buttons
        btnSubmit = new JButton(text: "OK");
        btnClear = new JButton(text: "Reset");
        c.add(lEmail);
        c.add(txtEmail);
        c.add(lPassword);
        c.add(txtPassword);
        c.add(btnSubmit);
        c.add(btnClear);
        btnSubmit.addActionListener(this);
        btnClear.addActionListener(this);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == btnSubmit) {
            strPassword = txtPassword.getText();
            strEmail = txtEmail.getText();
            if (strPassword.equals(anObject: "Prerna") &&
                    strEmail.equals(anObject: "prerna@gmail.com")) {
                JOptionPane.showMessageDialog(c, message: "Successful Login");
                System.exit(status: 0);
            } else {
                JOptionPane.showMessageDialog(c, message: "Unsuccessful Login");
                txtEmail.setText(t: "");
                txtPassword.setText(t: "");
            }
```

```
51  ∨          } else if (ae.getSource() == btnClear) {
52                  txtPassword.setText(t: "");
53                  txtEmail.setText(t: "");
54              }
55          }
56

            Run | Debug
57  ∨      public static void main(String z[]) {
58              Code2_Login frm = new Code2_Login();
59              frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
60              frm.setBounds(x: 200, y: 200, width: 550, height: 200);
61              frm.setVisible(b: true);
62              frm.setTitle(title: "LOGIN");
63          }
64      }
```

OUTPUT:

CONCLUSION: Thus, we implemented Graphical User Interfaces in Java using AWT and Event handling