**Experiment No 6**

**60009210105**

**Amitesh Sawarkar**

**D 12**

**Aim: -** Write a GUI programming in JAVA using Swing components, Containers, JLabel, JButton, JCheckBox, JRadio Buttons, JTextField etc

**Theory: -**

**Java Swing**

Java Swing is a GUI Framework that contains a set of classes to provide more powerful and flexible GUI components than AWT. Swing provides the look and feel of modern Java GUI. Swing library is an official Java GUI tool kit released by Sun Microsystems. It is used to create graphical user interface with Java.Swing classes are defined in javax. swing package and its sub-packages.

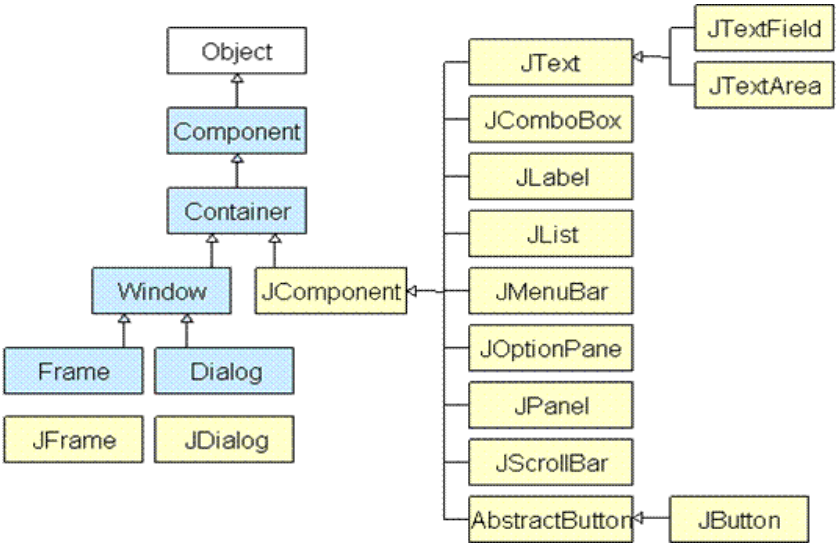
**Swing and JFC**

JFC is an abbreviation for Java Foundation classes which encompass a group of features for building Graphical User Interfaces(GUI) and adding rich graphical functionalities and interactivity to Java applications. Java Swing is a part of Java Foundation Classes (JFC).

Features of JFC

- Swing GUI components.
- Look and Feel support.
- Java 2D.

**AWT and Swing Hierarchy**


Object
Component
Container
Window
JComponent
Frame
Dialog
JFrame
JDialog
JText
JTextField
JTextArea
JComboBox
JLabel
JList
JMenuBar
JOptionPane
JPanel
JScrollBar
AbstractButton
JButton

**Introduction to Swing Classes**

JPanel: JPanel is Swing's version of AWT class Panel and uses the same default layout, FlowLayout. JPanel is descended directly from JComponent.

JFrame: JFrame is Swing's version of Frame and is descended directly from Frame class. The component which is added to the Frame, is referred as its Content.

JWindow: This is Swing's version of Window and has descended directly from Window class. Like Window it uses BorderLayout by default.

JLabel: JLabel has descended from JComponent, and is used to create text labels.

JButton: JButton class provides the functioning of push button. JButton allows an icon, string or both associated with a button.

JTextField: JTextField allow editing of a single line of text.

**Creating a JFrame**

There are two ways to create a JFrame Window.

**By instantiating JFrame class.**

```
import javax.swing.*;  //importing swing package
import javax.swing.*;  //importing swing package
import java.awt.*;     //importing awt package
public class First
{
        JFrame jf;
        public First() {
        jf = new JFrame("MyWindow");           //Creating a JFrame with name MyWindow
        JButton btn = new JButton("Say Hello");//Creating a Button named Say Hello
        jf.add(btn);                           //adding button to frame
        jf.setLayout(new FlowLayout());        //setting layout using FlowLayout object
        jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);     //setting close  operation.
                jf.setSize(400, 400);                  //setting size
                jf.setVisible(true);                   //setting frame visibility
        }
        public static void main(String[] args)
        {
                new First();
        }
}
```

**By extending JFrame class.**

```
import javax.swing.*; //importing swing package
import java.awt.*; //importing awt package
public class Second extends JFrame
```

```
{
        public Second()
        {
                setTitle("MyWindow"); //setting title of frame as  MyWindow
                JLabel lb = new JLabel("Welcome to My Second Window");//Creating a label named
Welcome to My Second Window
                add(lb);                    //adding label to frame.
                setLayout(new FlowLayout()); //setting layout using FlowLayout object.
                setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //setting close operation.
                setSize(400, 400);            //setting size
                setVisible(true);            //setting frame visibility
        }

        public static void main(String[] args)
        {
                new Second();
        }
}
```

**Points To Remember**

Import the javax.swing and java.awt package to use the classes and methods of Swing.

While creating a frame (either by instantiating or extending Frame class), following two attributes are must for visibility of the frame:

setSize(int width, int height);

setVisible(true);

Copy

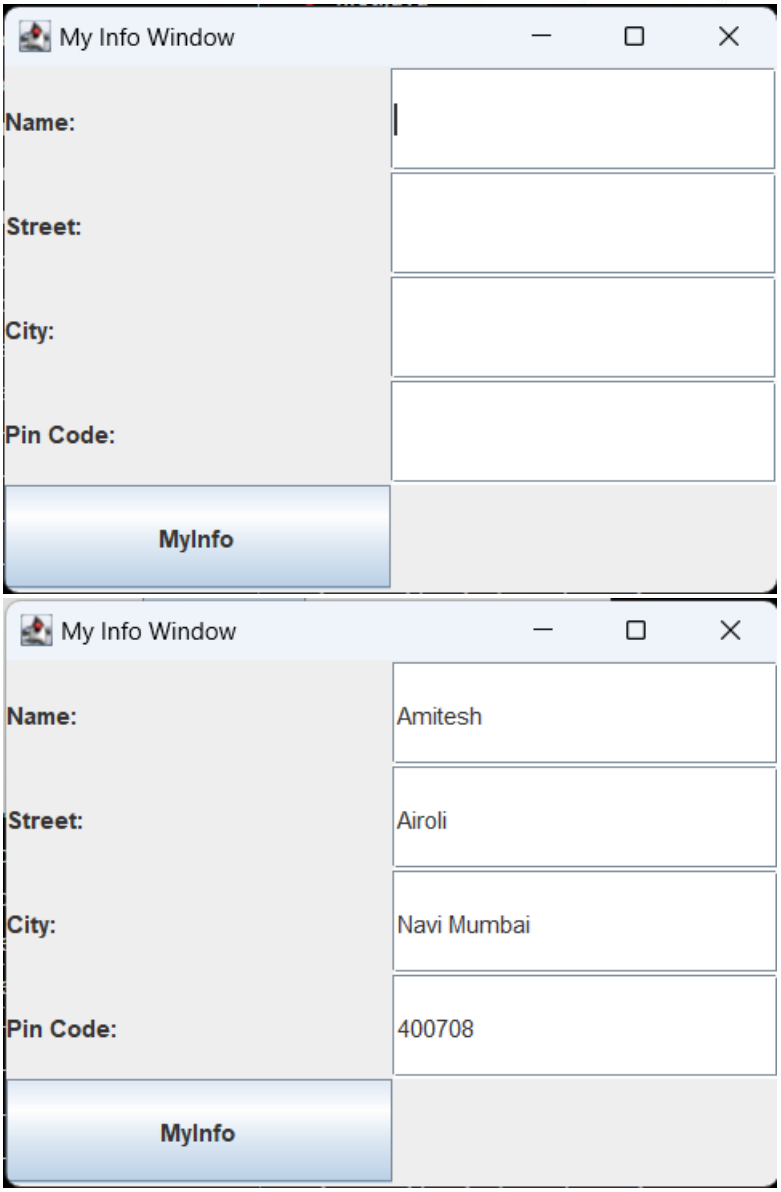
When you create objects of other components like Buttons, TextFields, etc. Then you need to add it to the frame by using the method - add(Component's Object);

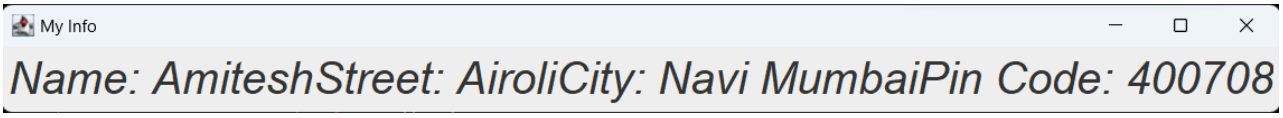
You can add the following method also for resizing the frame - setResizable(true);

## 4. Lab Assignments to complete in this session

i. Write a program to create a window with four text fields for the name, street, city and pin code with suitable labels. Also windows contain a button MyInfo. When the user types the name, his street, city and pincode and then clicks the button, the types details must appear in Arial Font with Size 32, Italics.

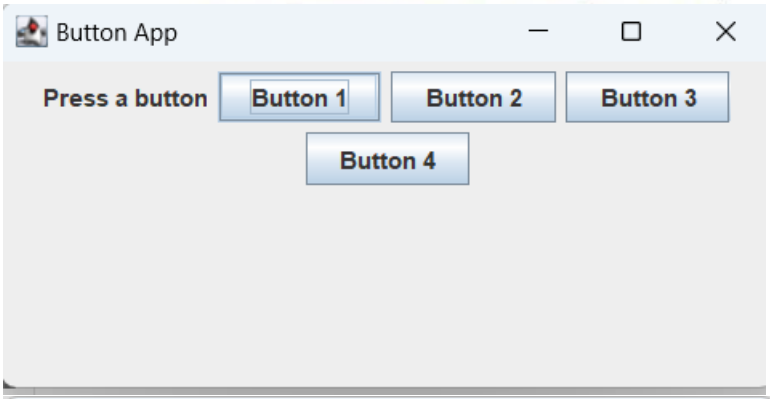
```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class first {
    public static void main(String[] args) {
        JFrame frame = new JFrame("My Info Window");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize(400, 300);
        frame.setLayout(new GridLayout(5, 2));

        JLabel nameLabel = new JLabel("Name:");
        JTextField nameField = new JTextField();
        JLabel streetLabel = new JLabel("Street:");
        JTextField streetField = new JTextField();
        JLabel cityLabel = new JLabel("City:");
        JTextField cityField = new JTextField();
        JLabel pincodeLabel = new JLabel("Pin Code:");
        JTextField pincodeField = new JTextField();

        JButton myInfoButton = new JButton("MyInfo");
        myInfoButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String name = nameField.getText();
                String street = streetField.getText();
                String city = cityField.getText();
                String pincode = pincodeField.getText();

                JLabel infoLabel = new JLabel("Name: " + name + "\nStreet: " + street + "\nCity: " + city + "\nPin Code: " + pincode);
                infoLabel.setFont(new Font("Arial", Font.ITALIC, 32));

                JFrame infoFrame = new JFrame("My Info");
                infoFrame.setLayout(new FlowLayout());
                infoFrame.add(infoLabel);
                infoFrame.pack();
                infoFrame.setVisible(true);
            }
        });

        frame.add(nameLabel);
        frame.add(nameField);
        frame.add(streetLabel);
        frame.add(streetField);
        frame.add(cityLabel);
        frame.add(cityField);
        frame.add(pincodeLabel);
        frame.add(pincodeField);
        frame.add(myInfoButton);

        frame.setVisible(true);
    }
}
```

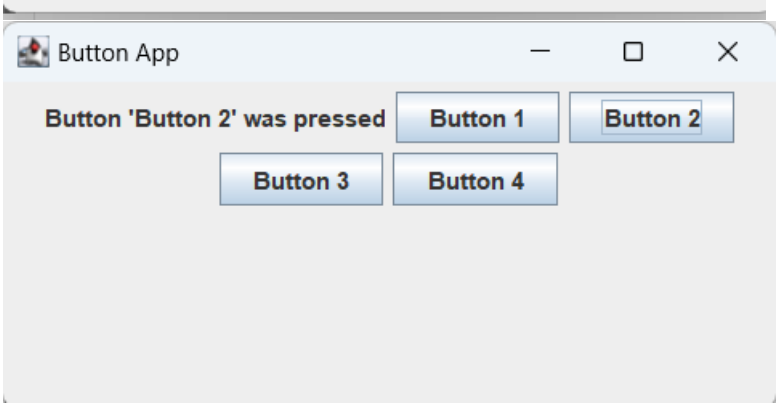



ii. WA applet with 4 swing buttons with suitable texts on them. When the user presses a button a message should appear in the label as to which button was pressed by the user

```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class second extends JFrame implements ActionListener {
    private JLabel label;

    public second() {
        super(title:"Button App");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(width:400, height:200);
        setLayout(new FlowLayout());

        label = new JLabel(text:"Press a button");
        add(label);

        JButton button1 = new JButton(text:"Button 1");
        JButton button2 = new JButton(text:"Button 2");
        JButton button3 = new JButton(text:"Button 3");
        JButton button4 = new JButton(text:"Button 4");

        button1.addActionListener(this);
        button2.addActionListener(this);
        button3.addActionListener(this);
        button4.addActionListener(this);

        add(button1);
        add(button2);
        add(button3);
        add(button4);
    }

    public void actionPerformed(ActionEvent e) {
        String buttonLabel = ((JButton) e.getSource()).getText();
        label.setText("Button '" + buttonLabel + "' was pressed");
    }

    Run | Debug
    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            second app = new second();
            app.setVisible(b:true);
        });
    }
}
```

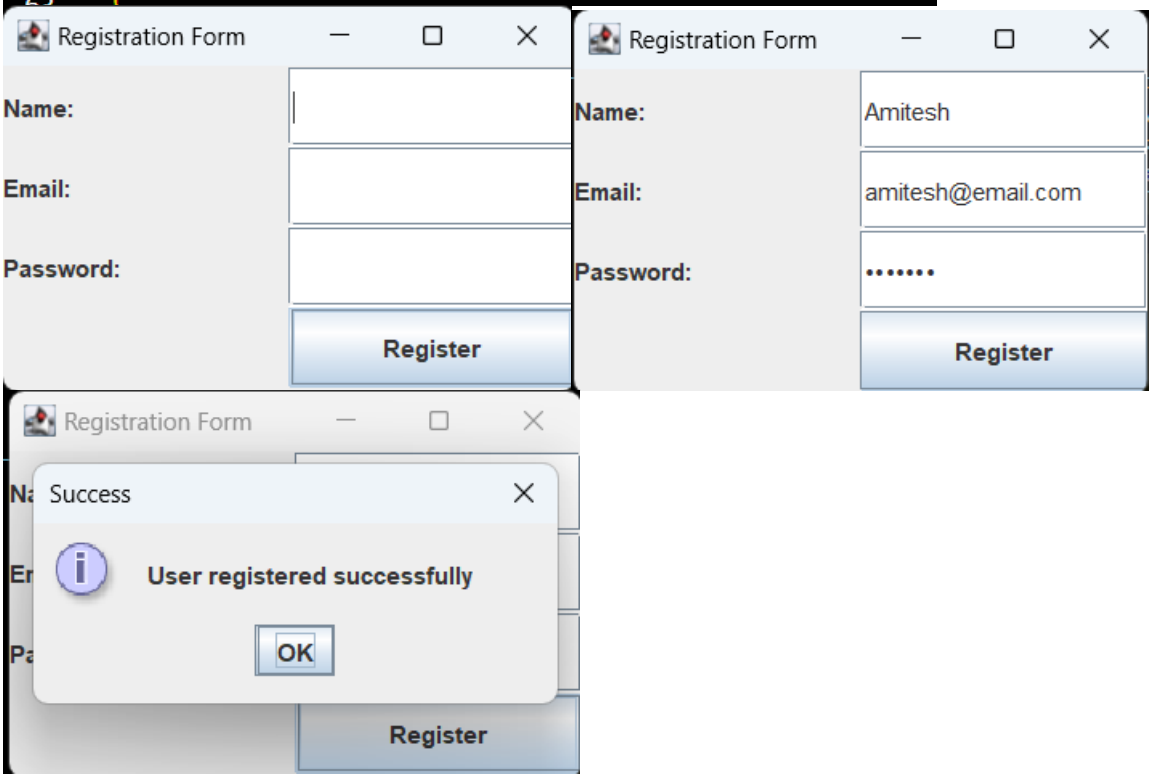

Button App
Press a button
Button 1
Button 2
Button 3
Button 4


Button App
Button 'Button 2' was pressed
Button 1
Button 2
Button 3
Button 4

iii. Write java program to create a registration form using Swing

```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class third extends JFrame {
    private JTextField nameField;
    private JTextField emailField;
    private JPasswordField passwordField;
    private JButton registerButton;

    public third() {
        setTitle(title:"Registration Form");
        setSize(width:300, height:200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new GridLayout(rows:4, cols:2));

        JLabel nameLabel = new JLabel(text:"Name:");
        nameField = new JTextField(columns:20);

        JLabel emailLabel = new JLabel(text:"Email:");
        emailField = new JTextField(columns:20);

        JLabel passwordLabel = new JLabel(text:"Password:");
        passwordField = new JPasswordField(columns:20);

        registerButton = new JButton(text:"Register");

        add(nameLabel);
        add(nameField);
        add(emailLabel);
        add(emailField);
        add(passwordLabel);
        add(passwordField);
        add(new JLabel());
        add(registerButton);

        registerButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String name = nameField.getText();
                String email = emailField.getText();
                char[] passwordChars = passwordField.getPassword();
                String password = new String(passwordChars);

                System.out.println("Name: " + name);
                System.out.println("Email: " + email);
                System.out.println("Password: " + password);

                JOptionPane.showMessageDialog(third.this, message:"User registered successfully", title:"Success", JOptionPane.INFORMATION_MESSAGE);

            }
        });
    }

```

```
Run | Debug
55      public static void main(String[] args) {
56          SwingUtilities.invokeLater(new Runnable() {
57              public void run() {
58                  third form = new third();
59                  form.setVisible(b:true);
60              }
61          });
62      }
```

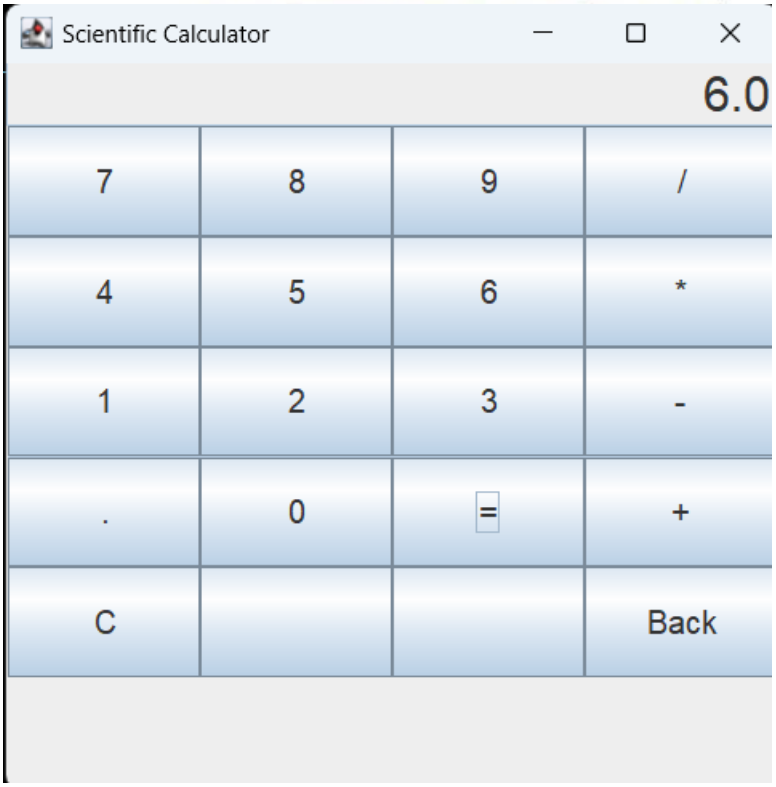



iv. Implement Scientific Calculator Using JAVA Swing

```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.lang.Math;

public class fourth extends JFrame implements ActionListener {
    private JTextField textField;
    private double firstOperand, secondOperand, result;
    private String operator;

    public fourth() {
        setTitle(title:"Scientific Calculator");
        setSize(width:400, height:400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        textField = new JTextField(columns:20);
        textField.setHorizontalAlignment(JTextField.RIGHT);
        textField.setFont(new Font(name:"Arial", Font.PLAIN, size:24));
        textField.setEditable(b:false);

        JPanel buttonPanel = new JPanel();
        buttonPanel.setLayout(new GridLayout(rows:6, cols:4));

        String[] buttonLabels = {
            "7", "8", "9", "/",
            "4", "5", "6", "*",
            "1", "2", "3", "-",
            ".", "0", "=", "+",
            "C", "", "", "Back"
        };

        for (String label : buttonLabels) {
            JButton button = new JButton(label);
            button.setFont(new Font(name:"Arial", Font.PLAIN, size:16));
            button.addActionListener(this);
            buttonPanel.add(button);
        }

        add(textField, BorderLayout.NORTH);
        add(buttonPanel, BorderLayout.CENTER);

        firstOperand = 0;
        secondOperand = 0;
        operator = "";
        result = 0;
    }
```

```
50      @Override
51      public void actionPerformed(ActionEvent e) {
52          String command = e.getActionCommand();
53
54          if ("0123456789.".contains(command)) {
55              textField.setText(textField.getText() + command);
56          } else if ("+-*/".contains(command)) {
57              firstOperand = Double.parseDouble(textField.getText());
58              operator = command;
59              textField.setText(t:"");
60          }
61          else if ("=".equals(command)) {
62              secondOperand = Double.parseDouble(textField.getText());
63              calculate();
64              operator = "";
65          } else if ("C".equals(command)) {
66              textField.setText(t:"");
67          } else if ("Back".equals(command)) {
68              String text = textField.getText();
69              if (!text.isEmpty()) {
70                  textField.setText(text.substring(beginIndex:0, text.length() - 1));
71              }
72          }
73      }
74
```

```
74
75      private void calculate() {
76          switch (operator) {
77              case "+":
78                  result = firstOperand + secondOperand;
79                  break;
80              case "-":
81                  result = firstOperand - secondOperand;
82                  break;
83              case "*":
84                  result = firstOperand * secondOperand;
85                  break;
86              case "/":
87                  if (secondOperand != 0) {
88                      result = firstOperand / secondOperand;
89                  } else {
90                      textField.setText(t:"Error");
91                      return;
92                  }
93                  break;
94          }
95          textField.setText(String.valueOf(result));
96      }
97
            Run | Debug
98      public static void main(String[] args) {
99          SwingUtilities.invokeLater(() -> {
100             fourth calculator = new fourth();
101             calculator.setVisible(b:true);
102         });
103     }
104 }
105
```


Scientific Calculator
6.0
7
8
9
/
4
5
6
*
1
2
3
-
.
0
=
+
C
Back