

WEEK 11

AIM: To write a Java program that works as a simple calculator.

THEORY: GridLayout is one of the Layout managers. A layout manager automatically arranges your controls within a window by using some type of algorithm. Grid Layout lays out components in a two-dimensional grid. When you instantiate a GridLayout, you define the number of rows and columns.

ALGORITHM:

STEP1: START

STEP2: Create an applet using extends Applet class.

STEP3: Create buttons using Buttons() and text field using TextField() classes.

STEP4: Add all the buttons in the required order to a panel.

STEP5: Use GridLayout and place the buttons in this layout by using add().

STEP6: Finally add the TextField() and Panel to the window.

STEP7: Implement the actionPerformed() method.

STEP8: END

SOURCE CODE:

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="Calc" height=300 width=300>
</applet>
*/
public class Calc extends Applet implements ActionListener
{
    TextField tf;
    double arg=0;
    String op="=";
    boolean start=true;
    public void init()
    {
        setLayout(new BorderLayout());
        tf=new TextField("0");
        add(tf,BorderLayout.NORTH);
        Panel p=new Panel();
        p.setLayout(new GridLayout(4,4));
        String buttons="123/456*789-0.+=";
```

```

for(int i=0;i<buttons.length();i++)
{
    Button b=new Button(buttons.substring(i,i+1));
    p.add(b);
    b.addActionListener(this);
}
add(p);
}

public void actionPerformed(ActionEvent ae)
{
    String s=ae.getActionCommand();
    if('0'<=s.charAt(0)&& s.charAt(0)<='9'||s.equals("."))
    {
        if(start)
            tf.setText(s);
        else
            tf.setText(tf.getText()+s);
        start=false;
    }
    else
    {
        calcu(Double.parseDouble(tf.getText()));
        op=s;
        start=true;
    }
}

public void calcu(double n)
{
    if(op.equals("+"))
        arg+=n;
    else
        if(op.equals("-"))
            arg-=n;
        else
            if(op.equals("*"))
                arg*=n;

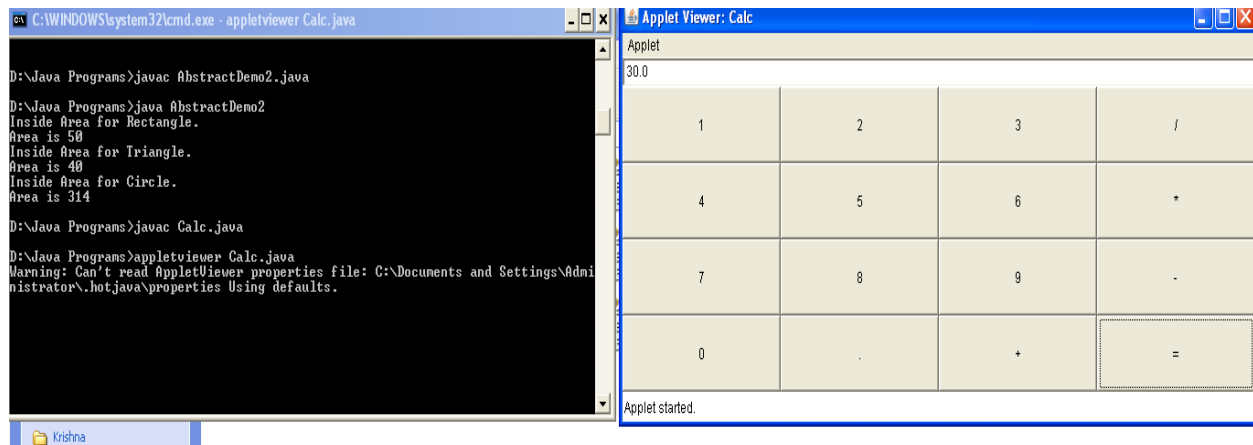
```

```

else
if(op.equals("/"))
{
try{
    arg/=n;
    }
catch(ArithmeticException e)
{
    tf.setText("Arithmetic Exception");
    }
}
else
if(op.equals("="))
arg=n;
tf.setText(""+arg);
}}

```

OUTPUT:



VIVA VOICE:

1. What is the difference between TextField and TextArea?

The object of a TextField class is a text component that allows the editing of a single line text. It inherits TextComponent class. On the other hand, the object of a TextArea class is a multi line region that displays text. It allows the editing of multi-line text. It inherits TextComponent class.

2. What is a Button and a Label?

The button class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. The object of Label class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly.

3. What is the use of Panel?

The Panel is a simplest container class. It provides space in which an application can attach any other component. It inherits the Container class. It doesn't have title bar.

3. What is the purpose of getActionCommand()?

getActionCommand() Returns the command string associated with this action. This string allows a "modal" component to specify one of several commands, depending on its state. For example, a single button might toggle between "show details" and "hide details".

5. What is the use of GridLayout?

The GridLayout is used to arrange the components in rectangular grid. One component is displayed in each rectangle.