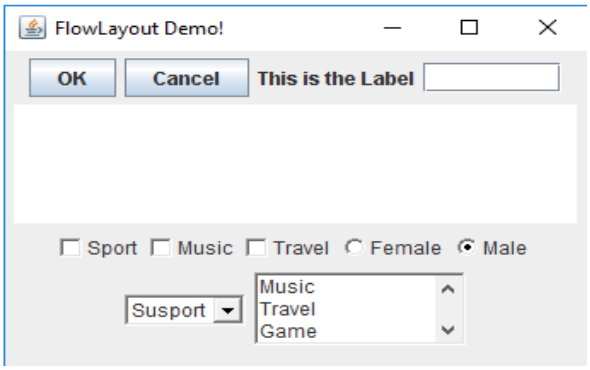
LAB 1 – LAYOUT MANAGERS

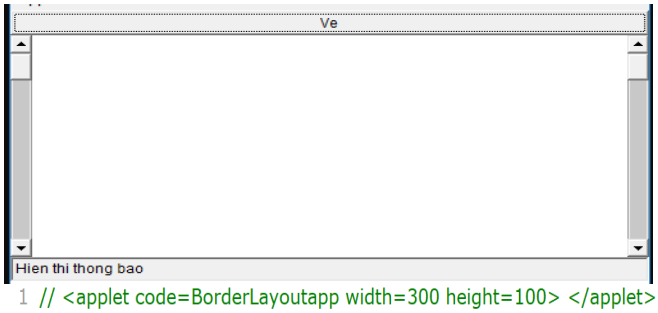
- Introduction to Layout Manager
- FlowLayout Manager
- BorderLayout Manager
- CardLayout Manager
- GridLayout Manager
- GridBagLayout Manager

1. FlowLayout Manager: The following program demonstrates the use of FlowLayoutDemo class



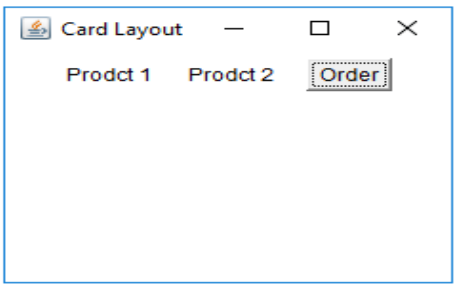

```
1  import java.awt.*;
2  import javax.swing.*;
3  class  FlowLayoutDemo extends JFrame
4  {
5     public static void main(String args[]){
6        FlowLayoutDemo  f= new FlowLayoutDemo();
7        f.setTitle("FlowLayout Demo!");
8        f.setBounds(200,500,700,700);
9        f.setLayout(new FlowLayout());
10       f.add(new JButton("OK"));
11       f.add(new JButton("Cancel"));
12
13       f.add(new JLabel("This is the Label"));
14       f.add(new JTextField(7));
15       f.add(new JTextArea(5,30));
16
17       f.add( new Checkbox("Sport"));
18       f.add( new Checkbox("Music"));
19       f.add( new Checkbox("Travel"));
20       CheckboxGroup cg=new CheckboxGroup();
21       f.add(new Checkbox ("Female", cg, false));
22       f.add(new Checkbox ("Male", cg, true));
23
24       Choice ch=new Choice();
25       ch.addItem("Susport");
26       ch.addItem("Music");
27       ch.addItem("Travel");
28       f.add(ch);
29       List list=new List(3,false);
30       list.add("Music");
31       list.add("Travel");
32       list.add("Game");
33       list.add("Telen");
34       f.add(list);
35       f.setVisible(true);
36    }
37 }
```

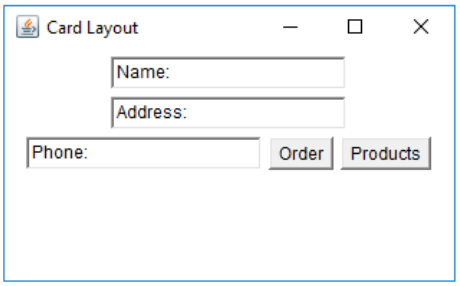
2. BorderLayout Manager: The following program demonstrates the BorderLayout class

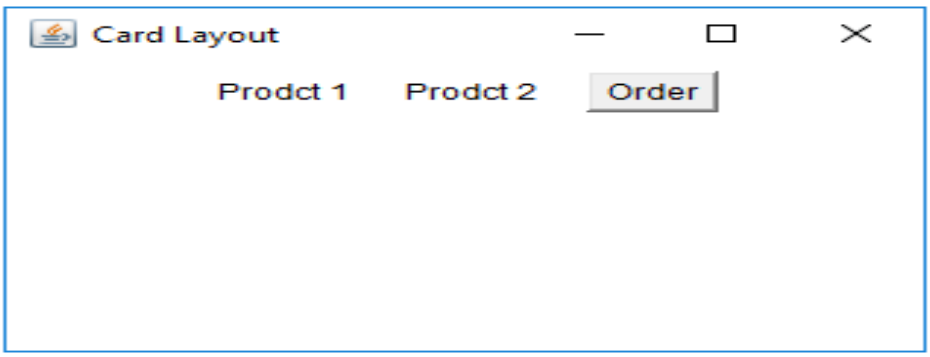



```
1  // <applet code=BorderLayoutapp width=300 height=100> </applet>
2  import java.awt.*;
3  import java.applet.*;
4  import javax.swing.*;
5  public class BorderLayoutDemo extends Applet {
6    public void init() {
7      JTextField msg = new JTextField("Hien thi thong bao");
8      msg.setEditable(false);
9      JButton nutVe = new JButton("Ve");
10     Canvas vungVe = new Canvas();
11     vungVe.setSize(150, 150);
12     vungVe.setBackground( Color.white);
13 Scrollbar sb1=new Scrollbar(Scrollbar.VERTICAL,0,10,30,100);
14 Scrollbar sb2=new Scrollbar(Scrollbar.VERTICAL,0,10,30,100);
15       setLayout(new BorderLayout());
16       add(nutVe, BorderLayout.NORTH);
17       add(msg, BorderLayout.SOUTH);
18       add(vungVe,BorderLayout.CENTER);
19       add(sb1, BorderLayout.WEST);
20          add(sb2, BorderLayout.EAST);
21     }
22 }
```

3. CardLayout Manager: The following program demonstrates the use of CardLayout with two panels. The 1st panel contain products and an Order button. When you click on the Order button, the 2st panel appears with name, address and phone textfields, Order and Product button. When you click on Product button, the 1st panel appears showing the products.

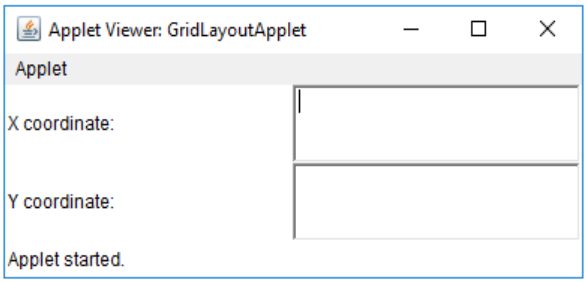






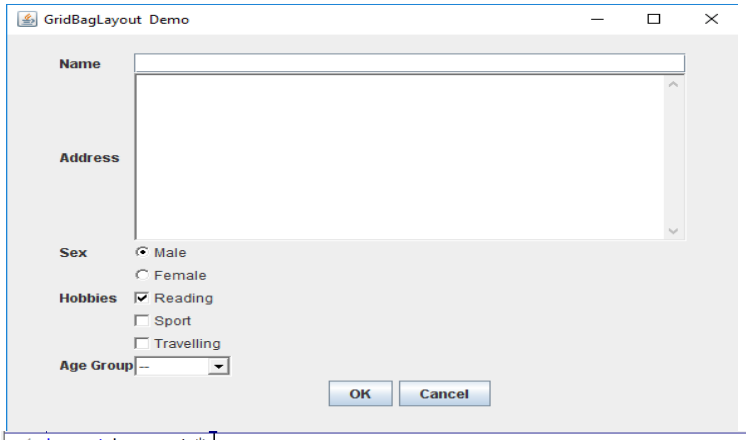
```
1   import  java.awt.*;
2   import  java.awt.event.*;
3   import javax.swing.*;
4   class CardLayoutDemo extends Frame implements ActionListener {
5   JButton order = new JButton("Order");
6   JButton products = new JButton("Products");
7   CardLayoutDemo(String title)  {
8      super(title);
9      setBounds(200 ,  200,  200,  200);
10     setLayout(new CardLayout());
11     JPanel p=new JPanel();
12     p.add(new JLabel("Product 1"));
13     p.add(new JLabel("Product 2"));
14     p.add(order);
15     order.addActionListener(this);
16     products.addActionListener(this);
17     add("products",p);
18     JPanel p1 = new JPanel();
19     p1.add(new JTextField("Name:  ",20));
20     p1.add(new JTextField("Address:  ",20));
21     p1.add(new JTextField("Phone:  ",20));
22     p1.add(new JButton("Order"));
23     p1.add(products);
24     add("order",p1) ;
25   addWindowListener (new  WindowAdapter( ) {
26   public  void  windowClosing ( WindowEvent  we) {
27         System.exit(0);
28            }
29         });
30         show();
31       }
32  public void actionPerformed(ActionEvent e)  {
33    CardLayout cl=(CardLayout) getLayout();
34    if (e.getSource()==order)
35       cl.show(this,"order");
36    if (e.getSource()==products) {
37       cl.show(this,"products");
38             }
39       }
40  public static void main (String args [] ) {
41    new CardLayoutTest("Card Layout");
42       }
43  }
```

4. GridLayout Manager: The following program demonstrates the GridLayout class



```java
// <applet code=GridLayoutApplet width=300 height=100> </applet>
import java.awt.*;
import java.applet.*;
public class GridLayoutDemo extends Applet {
  public void init() {
    //Cread a list of colors
    Label xLabel = new Label("X coordinate: ");
    Label yLabel = new Label("Y coordinate: ");
    TextField xInput = new TextField(5);
    TextField yInput = new TextField(5);
    setLayout(new GridLayout(2,2));
    add(xLabel); add(xInput);
    add(yLabel); add(yInput);
    }
}
```

5. GridBagLayout Manager: The following program demonstrates the GridBagLayout class that can be used to position each UI component as required. Note the use of fill and anchor properties in GridBagConstranints class




```java
import java.awt.*;
import javax.swing.*;
class GridBagLayoutDemo extends JFrame {
   JLabel lb1=new JLabel("Name");
   JTextField tf1=new JTextField(50);
   JLabel lb2=new JLabel("Address");
   JTextArea ta1=new JTextArea();
   JLabel lb3=new JLabel("Sex");
   CheckboxGroup cb=new CheckboxGroup();
   Checkbox cb1=new Checkbox("Male",cb,true);
   Checkbox cb2=new Checkbox("Female",cb,false);
   JLabel lb4=new JLabel("Hobbies");
   Checkbox cb3=new Checkbox("Reading",true);
   Checkbox cb4=new Checkbox("Sport",false);
   Checkbox cb5=new Checkbox("Travelling",false);
   JLabel lb5=new JLabel("Age Group");
   Choice ch1=new Choice();
   JButton bt1=new JButton("OK");
   JButton bt2=new JButton("Cancel");
```

```
20    GridBagLayout   gb;
21    GridBagConstraints   gbc;
22    public GridBagLayoutDemo(String  title){
23       super(title);
24       gb=new  GridBagLayout();
25       setLayout(gb);
26       ch1.addItem("--");
27       ch1.addItem("0-10");
28       ch1.addItem("10-20");
29       ch1.addItem("20-30");
30       ch1.addItem("30-40");
31       ch1.addItem("above 40");
32       gbc=new  GridBagConstraints();
33       addComponent(lb1, 0, 0, 1, 1, GridBagConstraints.HORIZONTAL);
34       addComponent(tf1, 0, 1, 1, 1, GridBagConstraints.HORIZONTAL);
35       addComponent(lb2, 1, 0, 1, 1, GridBagConstraints.HORIZONTAL);
36       addComponent(ta1, 1, 1, 1, 1, GridBagConstraints.HORIZONTAL);
37       addComponent(lb3, 2, 0, 1, 1, GridBagConstraints.HORIZONTAL);
38       addComponent(cb1, 2, 1, 1, 1, GridBagConstraints.HORIZONTAL);
58     gbc.gridheight=nrow;
59     gbc.fill=fill;
60     gb.setConstraints(c,gbc);
61     add(c);
62    }
63    public static void main(String  args[])
64    {
65      new  GridBagLayoutDemo("GridBagLayout  Demo");
66    }}
```