Lab 9: To illustrate the concept of Layout Management

Instructor:Bhawana Poudel

Theory: Define layout management , its use and types with definition and syntax.

Program 1: Concept of Flow Layout

Q1. Write the program of flow layout and write the output as well.

```
public class FlowLayoutExample extends Frame{
    // constructor
    public FlowLayoutExample(String title)
    {
        /* It would create the Frame by calling
        * the constructor of Frame class.
        */
        super(title);

        //Setting up Flow Layout
        setLayout(new FlowLayout());

        //Creating a button and adding it to the frame
        Button b1 = new Button("Button:1");
        add(b1);

        /* Adding other components to the Frame
        */
        Button b2 = new Button("Button:2");
        add(b2);

        Button b3 = new Button("Button:3");
        add(b3);

        Button b4 = new Button("Button:4");
        add(b4);

        Button b5 = new Button("Button:5");
        add(b5);

        Button b6 = new Button("Button:6");
        add(b6);

        Button b7 = new Button("Button:7");
        add(b7);

        Button b8 = new Button("Button:8");
        add(b8);
    }
    public static void main(String[] args)
    {   FlowLayoutExample f =
            new FlowLayoutExample("Flow Layout Examples");
        f.setSize(400,150);
        f.setVisible(true);
    }
}
```

2. Write the program of Border layout and find the output of each.
Example 1:

```java
import java.awt.*;
public class BorderLayoutExample2 extends Frame {
    public BorderLayoutExample2(String title) {
        super(title);
    add("North", new Button("North"));
    add("South", new Button("South"));
    add("East", new Button("East"));
    add("West", new Button("West"));
    add("Center", new Button("Center"));
    }
    public static void main(String[] args) {
        BorderLayoutExample2 b = new BorderLayoutExample2("BorderLayoutExample");
    b.setSize(300, 200);
    b.setVisible(true);
    }
}
```

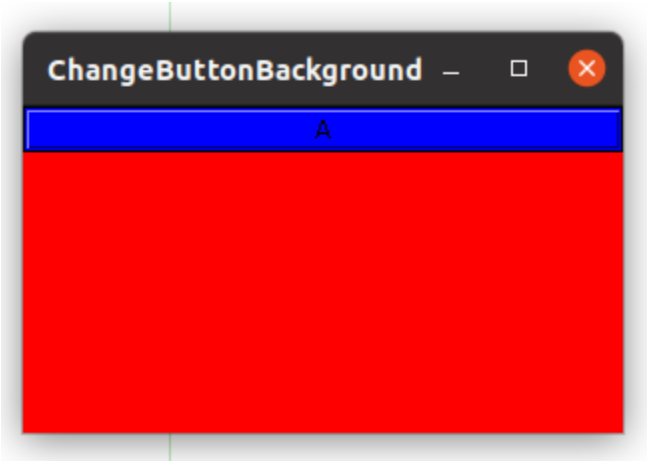
Example 2:

```java
import java.awt.*;
public class BorderPanelExample3 extends Frame {
    public BorderPanelExample3(String title) {
        super(title);
    addComponent("North", new Button("North"));
    addComponent("South", new Button("South"));
    addComponent("East", new Button("East"));
    addComponent("West", new Button("West"));
    addComponent("Center", new Button("Center"));
    }
    public void addComponent(String region, Component component) {
    Panel panel = new Panel();
    panel.add(component);
    add(region, panel); //one should add panel
    }
    public static void main(String[] args) {
        BorderPanelExample3 bpe =
            new BorderPanelExample3("Example3");
    bpe.setSize(200, 150);
    bpe.setVisible(true);
    }
}
```

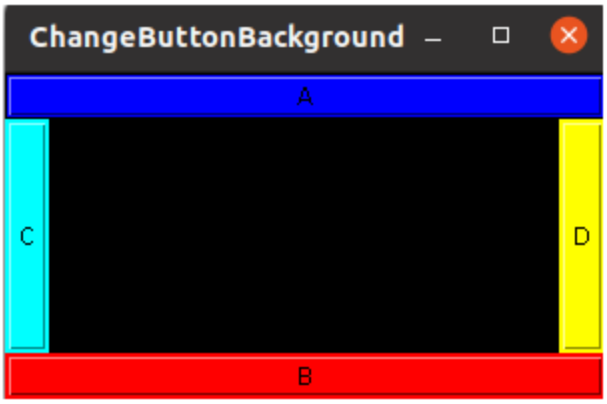
Example 3

```
import java.awt.BorderLayout;
import java.awt.Button;
import java.awt.Color;
import java.awt.Frame;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
class ChangeButtonBackground {
public static void main(String args[]) {
Frame frame = new Frame("ChangeButtonBackground");
Button button1 = new Button("A");
button1.setBounds(100,200,75,50);
Button button2 = new Button("B");
button1.setBackground(Color.blue);
frame.setBackground(Color.red);
frame.add(button1, BorderLayout.NORTH);
frame.pack();
frame.setVisible(true);
frame.setSize(300,200);
frame.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}
});
}
}
```

Output:



Task: Using the concept of border layout, WAP makes the output as shown below. Set the background
as black.



3. Write down the program of grid layout and find the output:
Program 3.1

```
import java.awt.*;
import javax.swing.*;
public class ButtonGrid4{
JFrame f;
ButtonGrid4(){
f=new JFrame();
JButton b1=new JButton("1");
JButton b2=new JButton("2");
JButton b3=new JButton("3");
JButton b4=new JButton("4");
JButton b5=new JButton("5");
JButton b6=new JButton("6");
JButton b7=new JButton("7");JButton b8=new JButton("8");
JButton b9=new JButton("9");
f.add(b1); f.add(b2); f.add(b3);
f.add(b4); f.add(b5); f.add(b6);
f.add(b7); f.add(b8); f.add(b9);
f.setLayout(new GridLayout(3,3));
f.setSize(300,300);
f.setVisible(true);
}
public static void main(String[] args) {
new ButtonGrid4();
}
}
```

Output: ?

Program 3.2

```java
import java.awt.*;
import javax.swing.*;
public class GridLayoutExample5
{
JFrame frameObj;
GridLayoutExample5()
{
frameObj = new JFrame();
JButton btn1 = new JButton("1");
JButton btn2 = new JButton("2");
JButton btn3 = new JButton("3");
JButton btn4 = new JButton("4");
JButton btn5 = new JButton("5");
JButton btn6 = new JButton("6");
JButton btn7 = new JButton("7");JButton btn8 = new JButton("8");
JButton btn9 = new JButton("9");
frameObj.add(btn1); frameObj.add(btn2); frameObj.add(btn3);
frameObj.add(btn4); frameObj.add(btn5); frameObj.add(btn6);
frameObj.add(btn7); frameObj.add(btn8); frameObj.add(btn9);
frameObj.setLayout(new GridLayout(3, 3, 20, 25));
frameObj.setSize(300, 300);
frameObj.setVisible(true);
}
public static void main(String argvs[])
{
new GridLayoutExample5();
}
}
```

Output: ?

4. Box Layout

Write down the program of box layout and find the output:

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

public class BoxLayoutExample6 extends Frame {
 Button buttons[];

 public BoxLayoutExample6 () {
   buttons = new Button [5];

   for (int i = 0;i<5;i++) {
      buttons[i] = new Button ("Button " + (i + 1));
      // adding the buttons so that it can be displayed
      add (buttons[i]);
    }
  // the buttons will be placed horizontally
setLayout (new BoxLayout (this, BoxLayout.Y AXIS));
setSize(400,400);
setVisible(true);
}
  // main method
public static void main(String args[]){
BoxLayoutExample6 b=new BoxLayoutExample6();
}
}
```

Output=?

5. Card Layout

Write a program of card layout and find output.

```java
import java.awt.*;
import java.awt.event.*;

import javax.swing.*;
public class CardLayoutClass{

public static void main(String[] args) {
    JFrame aWindow = new JFrame();
    aWindow.setBounds(200, 200, 200, 200);
    aWindow.setDefaultCloseOperation(JFrame.EXIT ON CLOSE);

    Container content = aWindow.getContentPane();
    content.add(new CardLayoutPanel());
    aWindow.setVisible(true);
}
}

class CardLayoutPanel extends JPanel implements ActionListener {
CardLayout card = new CardLayout(50, 50);

public CardLayoutPanel() {

    setLayout(card);
    JButton button;
    for (int i = 1; i <= 6; i++) {
    add(button = new JButton("Press " + i), "Card" + i);
    button.addActionListener(this);
    }
}

// Handle button events
public void actionPerformed(ActionEvent e) {
    card.next(this);
}
}
```