Southern Luzon State University
College of Engineering
Computer Engineering Department

CPE13 Object Oriented Programming

Activity 7: Layout Managers and Panels

Name: REIMARC G. CORPUZ Date: NOV. 2, 2022

Section: BSCPE 3GF Score:_______________

1.1 Introduction

The Java Swing toolkit has two kind of components. Containers and children. The containers group children into suitable layouts. To create layouts, we use layout managers. Layout managers are one of the most difficult parts of modern GUI programming. Many beginning programmers have too much respect for layout managers. Mainly because they are usually poorly documented.

Flow Layout Manager is the simplest layout manager in the Java Swing toolkit. It is mainly used in combination with other layout managers. When calculating its children size, a flow layout lets each component assume its natural (preferred) size.

The **Grid Layout manager** lays out components in a rectangular grid. The container is divided into equally sized rectangles. One component is placed in each rectangle.

A **Border Layout manager** is a very handy layout manager. It divides the space into five regions. North, West, South, East and Centre. Each region can have only one component. If we need to put more components into a region, we can simply put a panel there with a manager of our choice. The components in N, W, S, E regions get their preferred size. The component in the center takes up the whole space left.

It does not look good, if child components are too close to each other. We must put some space among them. Each component in Swing toolkit can have borders around it's edges. To create a border, we either create a new instance of an EmptyBorder class or we use a BorderFactory.

1.2 Objective

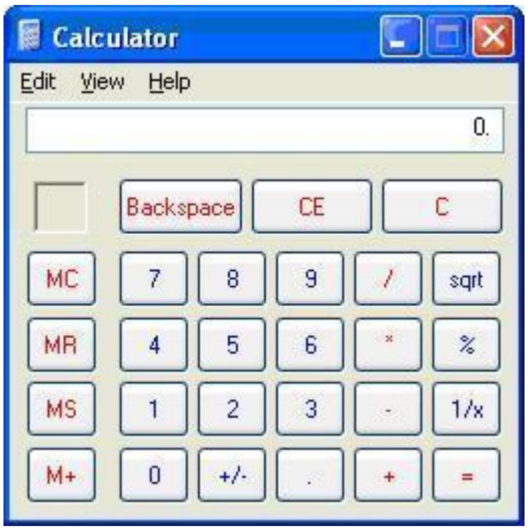
- To use Java programming language to create a program that uses Layout Managers and Panel.
- To conceptualize the process and manipulate the program
- To distinguish different parts of GUI Creation particularly the creation of frames and panel with its corresponding layout managers.

Sample Program:

```
import java.awt.*; // for dimensions
import javax.swing.*;// for component

public class CompositeLayout { //class

       public static void main (String[] args){//main method

             JFrame frame = new JFrame("Composite Layout"); //creation of frame with the name of
Composite Layout
             frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);//terminate when closed
             frame.setSize(350,900);//size or dimension horizontal and vertical
             frame.setLayout(new BorderLayout());//layout manager used in whole frame

             JPanel northPanel = new JPanel(new FlowLayout());// panel used in north region, flow
layout manager
             northPanel.add(new JLabel("BINGO!"));//label component in the north panel
             frame.add(northPanel, BorderLayout.NORTH);//add north panel to the frame
JPanel centerPanel = new JPanel(new GridLayout(16,5,5,5)); //panel for center region with
grid layout manager

             String[] buttons = {
                   "B", "I", "N", "G", "O",
                   "1", "16", "31", "46", "61",
                   "2", "17", "32", "47", "62",
                   "3", "18", "33", "48", "63",
                   "4", "19", "34", "49", "64",
                   "5", "20", "35", "50", "65",
                   "6", "21", "36", "51", "66",
                   "7", "22", "37", "52", "67",
                   "8", "23", "38", "53", "68",
                   "9", "24", "39", "54", "69",
                   "10", "25", "40", "55", "70",
                   "11", "26", "41", "56", "71",
                   "12", "27", "42", "57", "72",
                   "13", "28", "43", "58", "73",
                   "14", "29", "44", "59", "74",
                   "15", "30", "45", "60", "75"}; //string array for buttons of 75 numbers
and 5 letters

              for (int i = 0; i < buttons.length; i++) { //for loop in adding buttons to
center panel

                      centerPanel.add(new JButton(buttons[i])); //addding of buttons to the
center panel
              }

          frame.add(centerPanel, BorderLayout.CENTER); //adding of center panel to the
frame

          frame.setVisible(true); //set the visibility of frame as true

      }
```

1.3 Problem

Write a program of Layout of a Calculator using Different Layout Managers, Frame, Panel and Components.



1.4 Questions

1 What is the use of Layout Managers?

Components are organized in a certain way using layout managers. We can easily control the size and positioning of the components in GUI forms because of the Java LayoutManagers. All classes of layout managers implement the interface known as LayoutManager. Although components can give size and alignment indications, the layout manager of a container has the final say in how big and where to place the components.

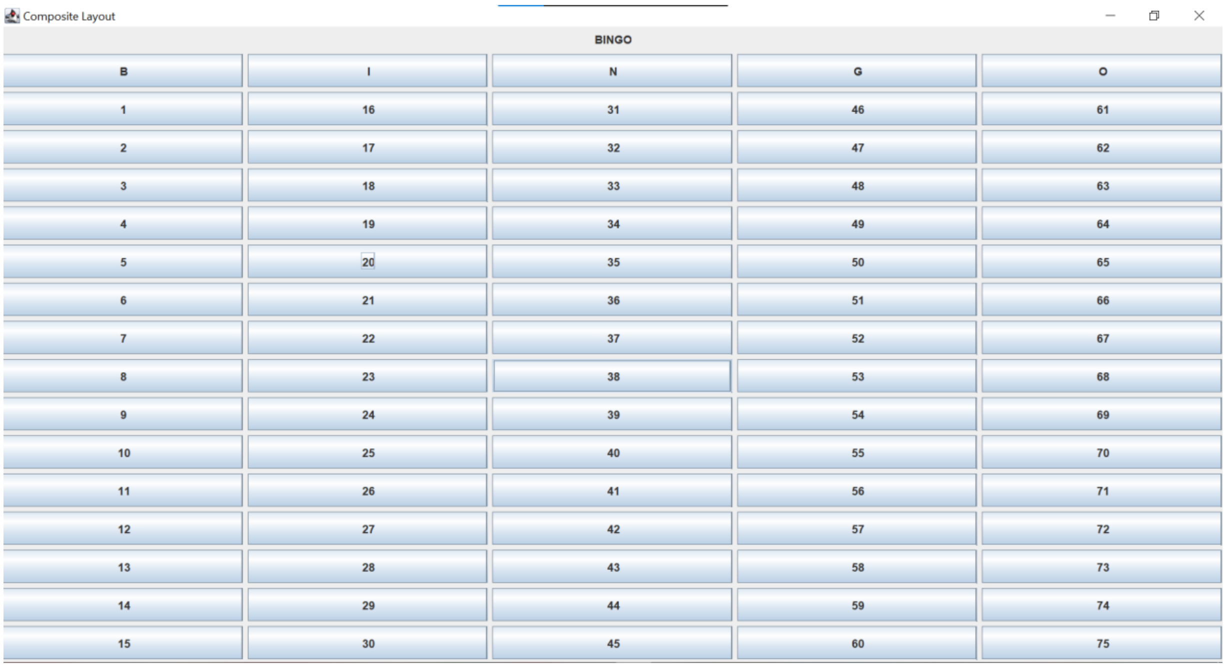
2. What is the purpose of panels in creating GUI?

A container is a panel. As each panel may have a distinct layout, it enables you to make use of the abilities of various layout managers. It also enables you to create extensive interfaces and combine components. It offers you the foundation for choosing how to deploy the panel and the ability to design reusable components and separate control. With a panel, you may add it as necessary to a frame or another component.

1.5 Conclusion

After creating the code of the given sample display which is the buttons of the calculator, I concluded that it is simpler to group components and arrange them in a Frame by utilizing Panels. Each Panel may have a different pattern. Therefore, you might place a Panel in the middle of your outer Frame and give it a BorderLayout. Following that, the Panel can have, for example, a GridBagLayout for its components. Because I want to place everything exactly where I want it on my swing components, it seems that anytime I try to write a program, I always end up utilizing the setLayout(null); Java command. Pack() "Causes this Window to be scaled to meet the chosen size and layouts of its subcomponents" when I use a layout, but I have to attempt to determine the boundaries myself when I don't.

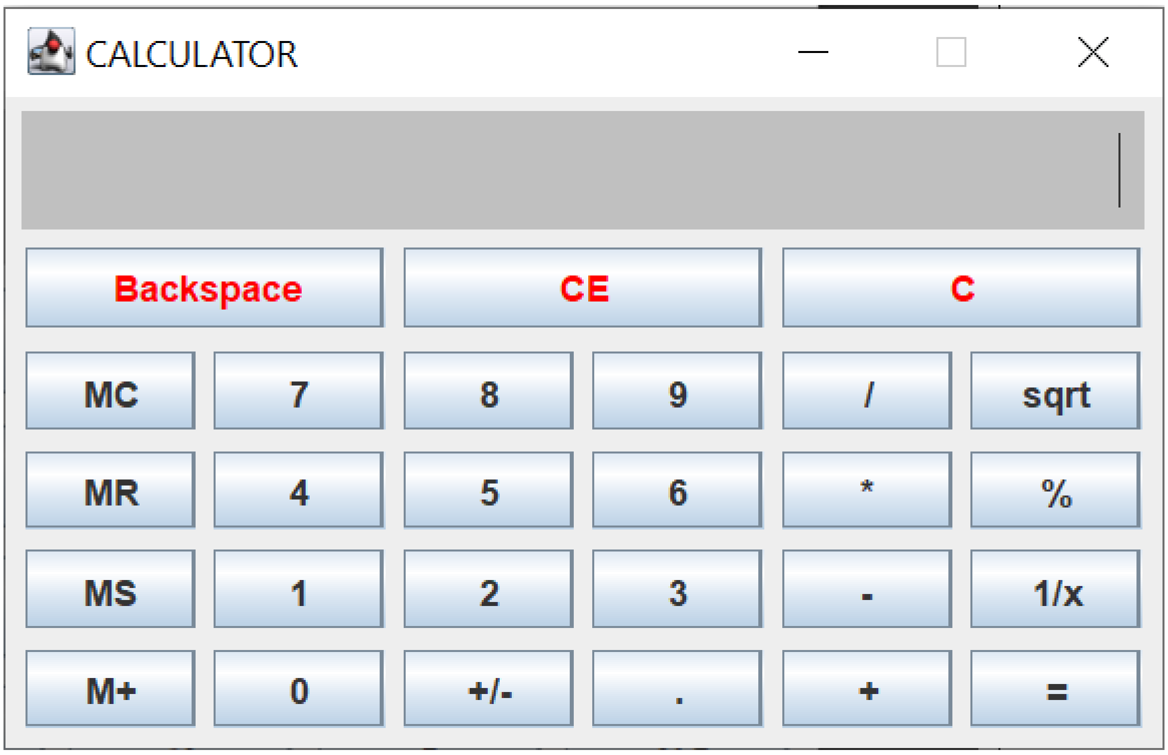
Output Screenshot of the Sample Program:

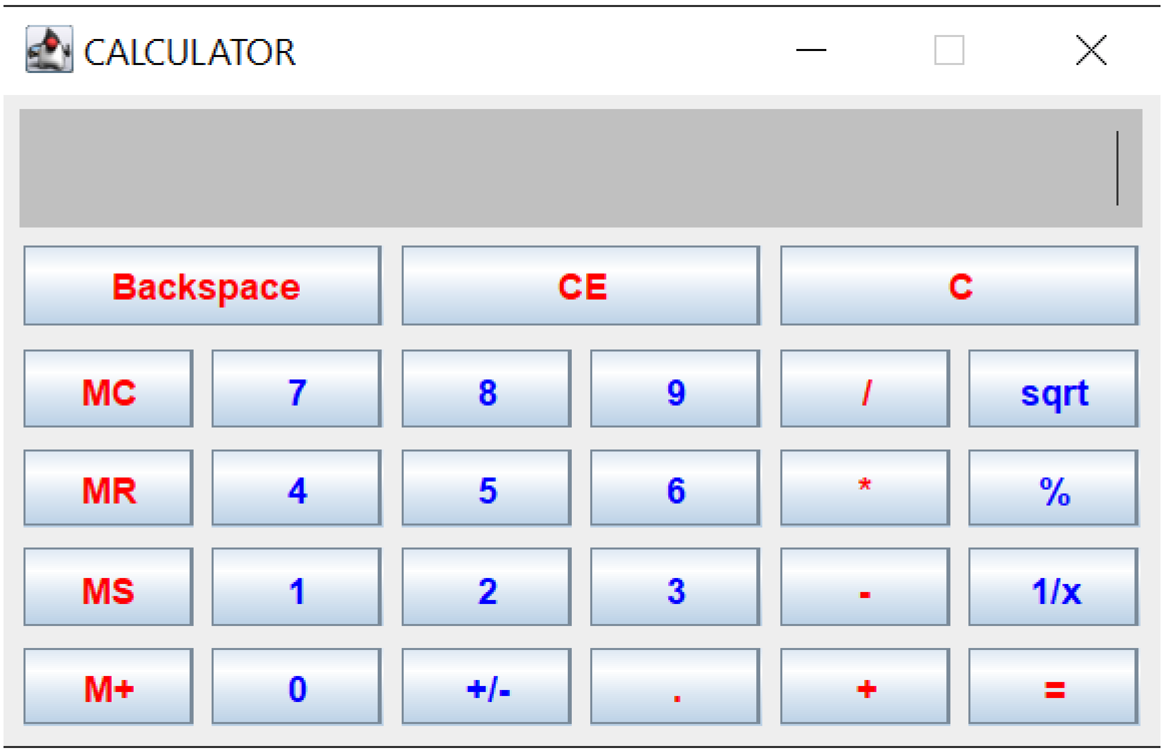


BINGO

B	I	N	G	O
1	16	31	46	61
2	17	32	47	62
3	18	33	48	63
4	19	34	49	64
5	20	35	50	65
6	21	36	51	66
7	22	37	52	67
8	23	38	53	68
9	24	39	54	69
10	25	40	55	70
11	26	41	56	71
12	27	42	57	72
13	28	43	58	73
14	29	44	59	74
15	30	45	60	75

Output Screenshot of the Given Sample Calculator:





Java Code1:

```
import java. awt.*;
import javax.swing.*;
public class CORPUZ_7 {

        public static void main (String[] args)
        {
                JFrame frame = new JFrame("CALCULATOR");

                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.setSize(400,254);
        frame.setLayout(new BorderLayout());

        JPanel northPanel = new JPanel(new FlowLayout());

        frame.add(northPanel, BorderLayout.NORTH);

        JTextField input_equation = new JTextField(24);
        input_equation.setFont(new Font("Serif", Font.PLAIN, 19));
        input_equation.setHorizontalAlignment(JTextField.RIGHT);
        input_equation.setBorder(BorderFactory.createEmptyBorder(7,7,7,7));
        input_equation.setBackground(Color.LIGHT_GRAY);
        frame.add(input_equation);
        northPanel.add(input_equation);

        JPanel centerPanel2 = new JPanel(new GridLayout(1,3,6,7));
        centerPanel2.setBorder(BorderFactory.createEmptyBorder(1,7,1,7));

        JButton Backspace = new JButton("Backspace");
        Backspace.setForeground(Color.RED);
        JButton CE = new JButton("CE");
        CE.setForeground(Color.RED);
        JButton C = new JButton("C");
        C.setForeground(Color.RED);



        frame.add(Backspace);
        frame.add(CE);
        frame.add(C);
        frame.add(centerPanel2, BorderLayout.CENTER);

        centerPanel2.add(Backspace);
        centerPanel2.add(CE);
        centerPanel2.add(C);

        JPanel centerPanel = new JPanel(new GridLayout(4,6,6,7));
        centerPanel.setBorder(BorderFactory.createEmptyBorder(7,7,7,7));

        String[] buttons =
            {
                "MC", "7", "8", "9", "/", "sqrt",
                "MR", "4", "5", "6", "*", "%",
                "MS", "1", "2", "3", "-", "1/x",
                "M+", "0", "+/-", ".", "+", "="
            };
        for (int i = 0; i < buttons.length; i++)
        {
            centerPanel.add(new JButton(buttons[i]));
        }

        frame.add(centerPanel, BorderLayout.SOUTH);
        frame.setVisible(true);
        //frame. setResizable(false);
        frame.setLayout(null);
    }
}
```

Java Code2:

```
import java.awt.*;
import javax.swing.*;
public class CORPUZ_7 {

	public static void main (String[] args)
	{
		JFrame frame = new JFrame("CALCULATOR");

		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.setSize(400,254);
		frame.setLayout(new BorderLayout());

		JPanel northPanel = new JPanel(new FlowLayout());

		frame.add(northPanel, BorderLayout.NORTH);

		JTextField input_equation = new JTextField(24);
		input_equation.setFont(new Font("Serif", Font.PLAIN, 19));
		input_equation.setHorizontalAlignment(JTextField.RIGHT);
		input_equation.setBorder(BorderFactory.createEmptyBorder(7,7,7,7));
		input_equation.setBackground(Color.LIGHT_GRAY);
		frame.add(input_equation);
		northPanel.add(input_equation);

		JPanel centerPanel2 = new JPanel(new GridLayout(1,3,6,7));
		centerPanel2.setBorder(BorderFactory.createEmptyBorder(1,7,1,7));

		JButton Backspace = new JButton("Backspace");
		Backspace.setForeground(Color.RED);
		JButton CE = new JButton("CE");
		CE.setForeground(Color.RED);
		JButton C = new JButton("C");
		C.setForeground(Color.RED);



		frame.add(Backspace);
		frame.add(CE);
		frame.add(C);

		centerPanel2.add(Backspace);
		centerPanel2.add(CE);
		centerPanel2.add(C);

		JPanel centerPanel = new JPanel(new GridLayout(4,6,6,7));
		centerPanel.setBorder(BorderFactory.createEmptyBorder(7,7,7,7));

		/*String[] buttons =
			{
				"MC", "7", "8", "9", "/", "sqrt",
				"MR", "4", "5", "6", "*", "%",
				"MS", "1", "2", "3", "-", "1/x",
				"M+", "0", "+/-", ".", "+", "="
			};
```

```
for (int i = 0; i < buttons.length; i++)
{
        centerPanel.add(new JButton(buttons[i]));
}*/

JButton MC = new JButton("MC");
MC.setForeground(Color.RED);
JButton seven = new JButton("7");
seven.setForeground(Color.BLUE);
JButton eight = new JButton("8");
eight.setForeground(Color.BLUE);
JButton nine = new JButton("9");
nine.setForeground(Color.BLUE);
JButton slash = new JButton("/");
slash.setForeground(Color.RED);
JButton sqrt = new JButton("sqrt");
sqrt.setForeground(Color.BLUE);
JButton MR = new JButton("MR");
MR.setForeground(Color.RED);
JButton four = new JButton("4");
four.setForeground(Color.BLUE);
JButton five = new JButton("5");
five.setForeground(Color.BLUE);
JButton six = new JButton("6");
six.setForeground(Color.BLUE);
JButton asterisk = new JButton("*");
asterisk.setForeground(Color.RED);
JButton percent = new JButton("%");
percent.setForeground(Color.BLUE);
JButton MS = new JButton("MS");
MS.setForeground(Color.RED);
JButton one = new JButton("1");
one.setForeground(Color.BLUE);
JButton two = new JButton("2");
two.setForeground(Color.BLUE);
JButton three = new JButton("3");
three.setForeground(Color.BLUE);
JButton minus = new JButton("-");
minus.setForeground(Color.RED);
JButton oneoverx = new JButton("1/x");
oneoverx.setForeground(Color.BLUE);
JButton Mplus = new JButton("M+");
Mplus.setForeground(Color.RED);
JButton zero = new JButton("0");
zero.setForeground(Color.BLUE);
JButton plusorminus = new JButton("+/-");
plusorminus.setForeground(Color.BLUE);
JButton point = new JButton(".");
point.setForeground(Color.RED);
JButton plus = new JButton("+");
plus.setForeground(Color.RED);
JButton equal = new JButton("=");
equal.setForeground(Color.RED);

frame.add(MC);
frame.add(seven);
frame.add(eight);
frame.add(nine);
```

```
        frame.add(slash);
        frame.add(sqrt);
        frame.add(MR);
        frame.add(four);
        frame.add(five);
        frame.add(six);
        frame.add(asterisk);
        frame.add(percent);
        frame.add(MS);
        frame.add(one);
        frame.add(two);
        frame.add(three);
        frame.add(minus);
        frame.add(oneoverx);
        frame.add(Mplus);
        frame.add(zero);
        frame.add(plusorminus);
        frame.add(point);
        frame.add(plus);
        frame.add(equal);

        centerPanel.add(MC);
        centerPanel.add(seven);
        centerPanel.add(eight);
        centerPanel.add(nine);
        centerPanel.add(slash);
        centerPanel.add(sqrt);
        centerPanel.add(MR);
        centerPanel.add(four);
        centerPanel.add(five);
        centerPanel.add(six);
        centerPanel.add(asterisk);
        centerPanel.add(percent);
        centerPanel.add(MS);
        centerPanel.add(one);
        centerPanel.add(two);
        centerPanel.add(three);
        centerPanel.add(minus);
        centerPanel.add(oneoverx);
        centerPanel.add(Mplus);
        centerPanel.add(zero);
        centerPanel.add(plusorminus);
        centerPanel.add(point);
        centerPanel.add(plus);
        centerPanel.add(equal);

        frame.add(centerPanel2, BorderLayout.CENTER);
        frame.add(centerPanel, BorderLayout.SOUTH);
        frame.setVisible(true);
        //frame. setResizable(false);
        frame.setLayout(null);
    }
}
```