1. Create a GUI application called MyComponents. Set the frame’s layout to

FlowLayout. It will initially look like the following:



CODE:

import java.awt.\*;

import java.awt.event.\*;

public class Main {

    public static void main(String[] args) {

        Frame frame = new Frame("My Components");

        frame.setLayout(new FlowLayout());

        Label label1 = new Label("Enter First number");

        Label label2 = new Label("Enter Second number");

        TextField t1 = new TextField(10);

        TextField t2 = new TextField(10);

        Button b1 = new Button("Button 1");

        Button b2 = new Button("Button 2");

        frame.add(label1);

        frame.add(t1);

        frame.add(label2);

        frame.add(t2);

        frame.add(b1);

        frame.add(b2);

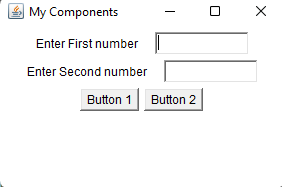
        frame.setSize(300, 200);

        frame.setVisible(true);

    }

}

OUTPUT:



2. Create the following GUI (you will provide functionality later): The GUI consists of

three Lists: two that contain the numbers 0–9, and one that contains three

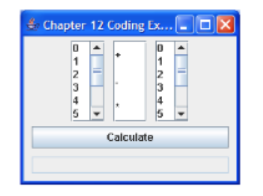
operations (+, - and \*). The GUI should also contain a Button with the label

"Calculate" and a TextField. The window shown is separated into two Panels — the

top one contains the three ScrollPanes in a GridLayout, and the bottom one contains

the "Calculate" button and the TextField in a BorderLayout. You may space and size

all the components as you like.



CODE:

import java.awt.\*;

import java.awt.event.\*;

public class calculator extends Frame {

    List l1,l2,l3;

    Button b1;

    TextField t1;

    public calculator(){

        setLayout(new GridLayout(3,1));

        l1 = new List();

        l2 = new List();

        l3 = new List();

       for (int i = 0; i<10 ; i++  ){

           l1.add(String.valueOf(i));

       }

       l2.add("+");

       l2.add("-");

       l2.add("\*");

        for (int i = 0; i<10 ; i++  ){

            l3.add(String.valueOf(i));

        }

        add(l1);

        add(l2);

        add(l3);

        setLayout(new GridLayout(2,1));

        b1 = new Button("Calculate");

        b1.setPreferredSize(new Dimension(250,50));

        t1 = new TextField();

        t1.setPreferredSize(new Dimension(250,50));

        add(b1);

        add(t1);

        setSize(250, 250);

        setLayout(new FlowLayout());

        setTitle("Calculator Components");

        addWindowListener(new WindowAdapter() {

            public void windowClosing(WindowEvent we) {

                dispose();

            }

        });

        setVisible(true);

    }

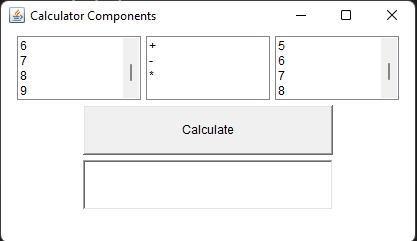
    public static void main(String[] args){

        new calculator();

    }

}

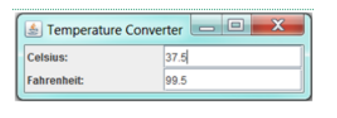
OUTPUT:



3. Write a GUI program called TemperatureConverter to convert temperature values

between Celsius and Fahrenheit. User can enter either the Celsius or the Fahrenheit

value, in floating-point number.



CODE:

import java.awt.\*;

import java.awt.event.\*;

public class temperature {

    public static void main(String[] args) {

        Frame frame = new Frame("Temperature Converter");

        // Set the layout to GridLayout with 2 rows and 2 columns

        frame.setLayout(new GridLayout(2, 2, 10, 10));

        Label celsiusLabel = new Label("Celsius: ");

        Label fahrenheitLabel = new Label("Fahrenheit: ");

        TextField celsiusField = new TextField(10);

        TextField fahrenheitField = new TextField(10);

        // Allow floating-point input

        celsiusField.setText("0.0");

        fahrenheitField.setText("32.0");

        frame.add(celsiusLabel);

        frame.add(celsiusField);

        frame.add(fahrenheitLabel);

        frame.add(fahrenheitField);

        frame.setSize(500, 100);

        // Add a window listener to handle closing events

        frame.addWindowListener(new WindowAdapter() {

            public void windowClosing(WindowEvent windowEvent) {

                System.exit(0);

            }

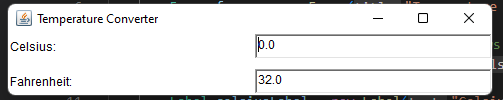
        });

        frame.setVisible(true);

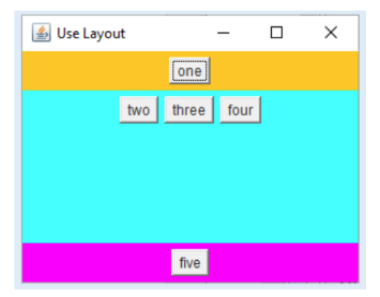
    }

}

OUTPUT:



4. Use suitable layout to display the following GUI



Code:

import java.awt.\*;

import java.awt.event.\*;

public class button\_layout {

    public static void main(String[] args) {

        Frame frame = new Frame("buttons layout");

        Panel panel1 = new Panel();

        Panel panel2 = new Panel();

        Panel panel3 = new Panel();

        //panel1

        panel1.setBackground(Color.orange);

        panel1.setPreferredSize(new Dimension(800,100));

        panel1.setLayout(new FlowLayout(FlowLayout.CENTER));

        Button button\_one = new Button("one");

        button\_one.setPreferredSize(new Dimension(70,50));

        panel1.add(button\_one);

        //panel2

        panel2.setBackground(Color.pink);

        Button button\_two = new Button("two");

        Button button\_three = new Button("three");

        Button button\_four = new Button("four");

        button\_two.setPreferredSize(new Dimension(70,50));

        button\_three.setPreferredSize(new Dimension(70,50));

        button\_four.setPreferredSize(new Dimension(70,50));

        panel2.setLayout(new FlowLayout(FlowLayout.CENTER));

        panel2.add(button\_two);

        panel2.add(button\_three);

        panel2.add(button\_four);

        //panel3

        panel3.setBackground(Color.blue);

        panel3.setPreferredSize(new Dimension(800,100));

        Button button\_five = new Button("five");

        button\_five.setPreferredSize(new Dimension(70,50));

        panel3.setLayout(new FlowLayout(FlowLayout.CENTER));

        panel3.add(button\_five);

        frame.setLayout(new BorderLayout());

        frame.add(panel1, BorderLayout.NORTH);

        frame.add(panel2,BorderLayout.CENTER);

        frame.add(panel3,BorderLayout.SOUTH);

        frame.setSize(800,500);

        frame.setVisible(true);

        frame.addWindowListener(new WindowAdapter() {

            public void windowClosing(WindowEvent e){

                frame.dispose();

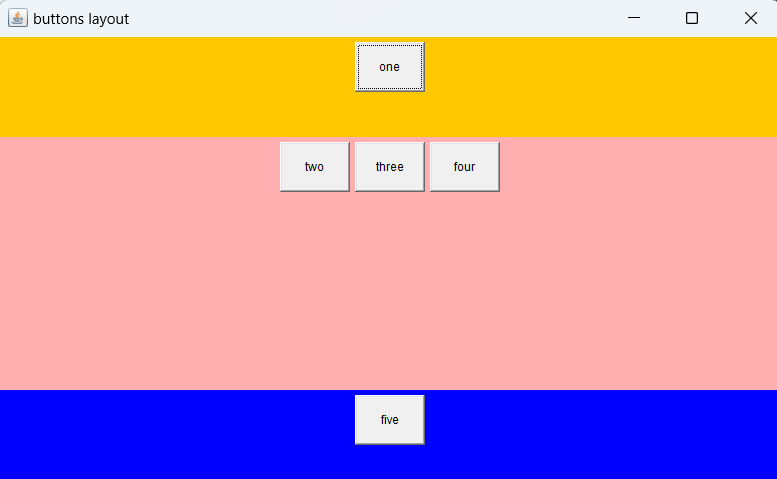
            }

        });

    }

}

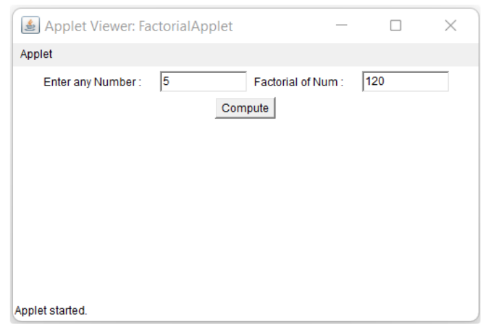
OUTPUT:



5. Develop an applet in Java that receives an integer in one text field, and computes its

factorial Value and returns it in another text field, when the button named

“Compute” is clicked.



Code:

import java.awt.\*;

import java.awt.event.\*;

public class factorial {

    public static void main(String[] args) {

        // Creating the frame

        Frame frame = new Frame("Factorial");

        // Creating the panel to hold the buttons and text fields

        Panel panel = new Panel();

        panel.setLayout(new FlowLayout(FlowLayout.CENTER));

        frame.add(panel);

        // Creating the first label and text field

        Label numberLabel = new Label("Enter any number: ");

        TextField numberInput = new TextField();

        numberInput.setPreferredSize(new Dimension(150, 30));

        // Creating the result label and area

        Label resultLabel = new Label("Factorial of number: ");

        TextField answerField = new TextField();

        answerField.setPreferredSize(new Dimension(150, 30));

        answerField.setEditable(false); // Make the answer field non-editable

        // Creating the compute button

        Button computeButton = new Button("Compute");

        computeButton.setPreferredSize(new Dimension(100, 50));

        // Adding components to the panel

        panel.add(numberLabel);

        panel.add(numberInput);

        panel.add(resultLabel);

        panel.add(answerField);

        panel.add(computeButton);

        // Adding action listener to the compute button

        computeButton.addActionListener(new ActionListener() {

            @Override

            public void actionPerformed(ActionEvent e) {

                try {

                    int number = Integer.parseInt(numberInput.getText());

                    long factorial = calculateFactorial(number);

                    answerField.setText(String.valueOf(factorial));

                } catch (NumberFormatException ex) {

                    answerField.setText("Invalid input");

                }

            }

        });

        frame.setSize(600, 500);

        frame.setVisible(true);

        frame.addWindowListener(new WindowAdapter() {

            public void windowClosing(WindowEvent e) {

                frame.dispose();

            }

        });

    }

    private static long calculateFactorial(int number) {

        if (number < 0) {

            throw new IllegalArgumentException("Number must be non-negative");

        }

        long result = 1;

        for (int i = 1; i <= number; i++) {

            result \*= i;

        }

        return result;

    }

}

OUTPUT:

