

Zid'Avwa Al Bari'i

244107020083 / TI – 2I / 26

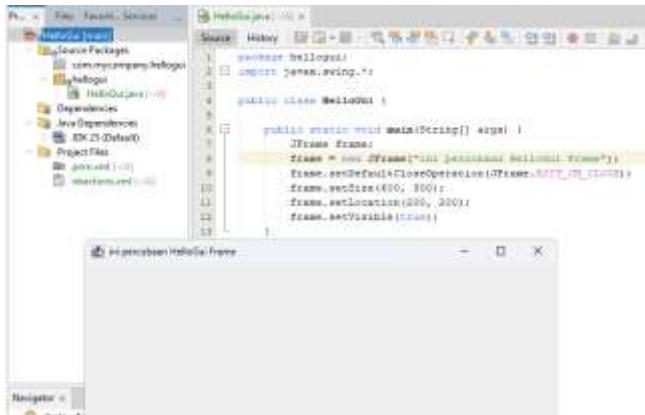
All the code here is on GitHub

<https://github.com/ZidAvwa/CollegeStudy/tree/main/3rdSemester/13.%20Jobsheet13>

Jobsheet 13

Experiment 1

Code



Experiment 2

Code

```

1 package sellogui;
2 import java.awt.event.ActionEvent;
3 import java.awt.event.ActionListener;
4 import javax.swing.*;
5
6 public class MyInputForm extends JFrame {
7
8     private static final int FRAME_WIDTH = 400;
9     private static final int FRAME_HEIGHT = 200;
10    private JLabel jLabel1;
11    private JLabel jLabel2;
12    private JLabel jLabel3;
13    private JTextField jTextField1;
14    private JTextField jTextField2;
15    private JButton button1;
16    private JPanel panel1;
17
18    public MyInputForm() {
19        createTextField();
20        createButton();
21        createPanel();
22        setSize(FRAME_WIDTH, FRAME_HEIGHT);
23    }
24
25    private void createTextField() {
26        jLabel1 = new JLabel("Value A: ");
27        jLabel2 = new JLabel("Value C: ");
28        jLabel3 = new JLabel("Result: ");
29
30        final int FIELD_WIDTH = 10;
31        jTextField1 = new JTextField(FIELD_WIDTH);
32        jTextField1.setText("0");
33        jTextField2 = new JTextField(FIELD_WIDTH);
34        jTextField2.setText("0");
35
36    }
37
38    private void createButton() {
39        button1 = new JButton("Calculate");
40        class AddInterestListener implements ActionListener {
41            @Override
42            public void actionPerformed(ActionEvent event) {
43                int a = Integer.valueOf(jTextField1.getText());
44                int b = Integer.valueOf(jTextField2.getText());
45                int c = a + b;
46                jLabel3.setText("Result: " + c);
47            }
48        }
49        button1.addActionListener(new AddInterestListener());
50    }
51
52    private void createPanel() {
53        panel1 = new JPanel();
54        panel1.setLayout(null);
55        panel1.add(jLabel1);
56        panel1.add(jTextField1);
57        panel1.add(jLabel2);
58        panel1.add(jTextField2);
59        panel1.add(button1);
60        panel1.add(jLabel3);
61        panel1.add(addButton);
62    }
63
64 }

```

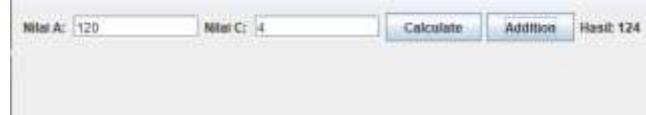
Question

Modify the program code by adding a new JButton to perform the addition calculation function, so that when the button is clicked (event click) it will display the addition results of the values A and B

```

62 addButton = new JButton("Addition");
63 class SumListener implements ActionListener {
64     @Override
65     public void actionPerformed(ActionEvent event) {
66         int a = Integer.valueOf(jTextField1.getText());
67         int b = Integer.valueOf(jTextField2.getText());
68         int c = a + b;
69         jLabel3.setText("Result: " + c);
70     }
71 }
72 addButton.addActionListener(new SumListener());
73
74
75 private void createPanel() {
76     panel1 = new JPanel();
77     panel1.add(jLabel1);
78     panel1.add(jTextField1);
79     panel1.add(jLabel2);
80     panel1.add(jTextField2);
81     panel1.add(button1);
82     panel1.add(jLabel3);
83     panel1.add(addButton);
84 }

```



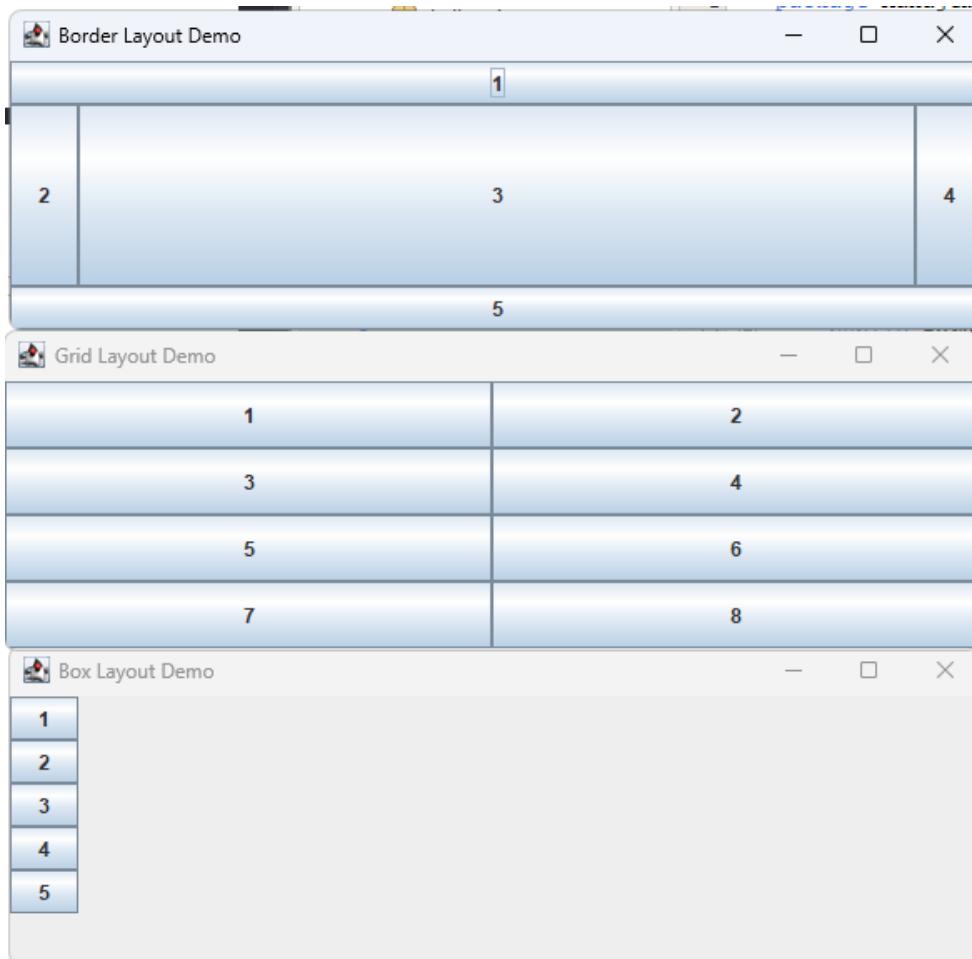
The screenshot shows the Java code for the addition button. The code defines a JButton named addButton and a new class SumListener that implements the ActionListener interface. The actionPerformed method calculates the sum of the values from the two text fields and updates the result label. The code then adds the addButton to the panel1.

1. Declared private JButton addButton; at the top.
2. Inside createButton(): Added code to create the "Addition" button and a new listener (SumListener) that performs $\text{int } c = a + b;$.
3. Inside createPanel(): Added panel.add(addButton); so the button actually shows up on the screen.

Experiment 3

Code

```
1 package ManajemenLayout;
2 import java.awt.BorderLayout;
3 import javax.swing.JButton;
4 import javax.swing.JFrame;
5 import javax.swing.JPanel;
6
7 public class Border extends JFrame {
8     private static final int FRAME_WIDTH = 600;
9     private static final int FRAME_HEIGHT = 200;
10    private JPanel panel;
11
12    public Border() {
13        panel = new JPanel();
14        panel.setLayout(new BorderLayout());
15
16        panel.add(new JButton("1"), BorderLayout.NORTH);
17        panel.add(new JButton("2"), BorderLayout.WEST);
18        panel.add(new JButton("3"), BorderLayout.CENTER);
19        panel.add(new JButton("4"), BorderLayout.EAST);
20        panel.add(new JButton("5"), BorderLayout.SOUTH);
21
22        add(panel);
23        setSize(FRAME_WIDTH, FRAME_HEIGHT);
24        setTitle("Border Layout Demo");
25    }
26}
27
28 package ManajemenLayout;
29
30 import java.awt.GridLayout;
31 import javax.swing.JButton;
32 import javax.swing.JFrame;
33 import javax.swing.JPanel;
34
35 public class Grid extends JFrame {
36     private static final int FRAME_WIDTH = 600;
37     private static final int FRAME_HEIGHT = 200;
38     private JPanel panel;
39
40     public Grid() {
41         panel = new JPanel();
42         panel.setLayout(new GridLayout(4, 2));
43         panel.add(new JButton("1"));
44         panel.add(new JButton("2"));
45         panel.add(new JButton("3"));
46         panel.add(new JButton("4"));
47         panel.add(new JButton("5"));
48         panel.add(new JButton("6"));
49         panel.add(new JButton("7"));
50         panel.add(new JButton("8"));
51
52         add(panel);
53         setSize(FRAME_WIDTH, FRAME_HEIGHT);
54         setTitle("Grid Layout Demo");
55     }
56}
57
58 package ManajemenLayout;
59
60 import javax.swing.Box;
61 import javax.swing.BoxLayout;
62 import javax.swing.JButton;
63 import javax.swing.JFrame;
64 import javax.swing.JPanel;
65
66 public class Box extends JFrame {
67     private static final int FRAME_WIDTH = 600;
68     private static final int FRAME_HEIGHT = 200;
69     private JPanel panel;
70
71     public Box() {
72         panel = new JPanel();
73         panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));
74
75         panel.add(new JButton("1"));
76         panel.add(new JButton("2"));
77         panel.add(new JButton("3"));
78         panel.add(new JButton("4"));
79         panel.add(new JButton("5"));
80
81         add(panel);
82         setSize(FRAME_WIDTH, FRAME_HEIGHT);
83         setTitle("Box Layout Demo");
84     }
85}
86
87 package ManajemenLayout;
88
89 import javax.swing.JFrame;
90
91 public class LayoutGUI {
92     public static void main(String debby[]) {
93         JFrame frame = new Border();
94         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
95         frame.setVisible(true);
96
97         JFrame frame2 = new Grid();
98         frame2.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
99         frame2.setVisible(true);
100
101         JFrame frame3 = new Box();
102         frame3.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
103         frame3.setVisible(true);
104     }
105 }
```

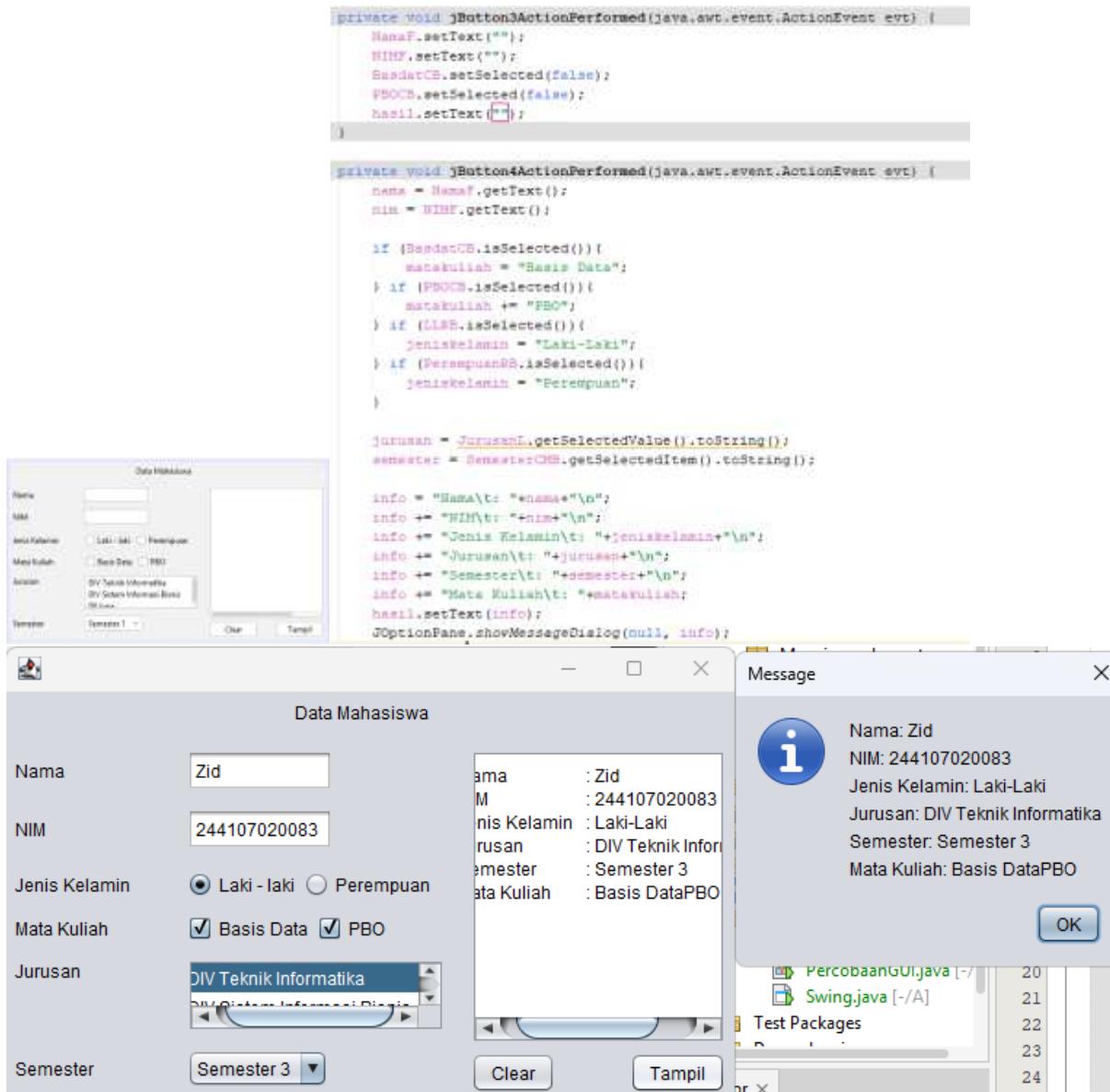


Question

- What is the difference between Grid Layout, Box Layout and Border Layout?
 - BorderLayout: Divides the window into 5 fixed areas (North, South, East, West, Center).
 - GridLayout: Arranges components in a grid of equal-sized cells.
 - BoxLayout: Stacks components in a single straight line (either vertical or horizontal).
- What is the function of each of the following codes?
 These lines create the window object for that specific layout, ensure the program actually stops running when you close the window, and make the window visible to the user.

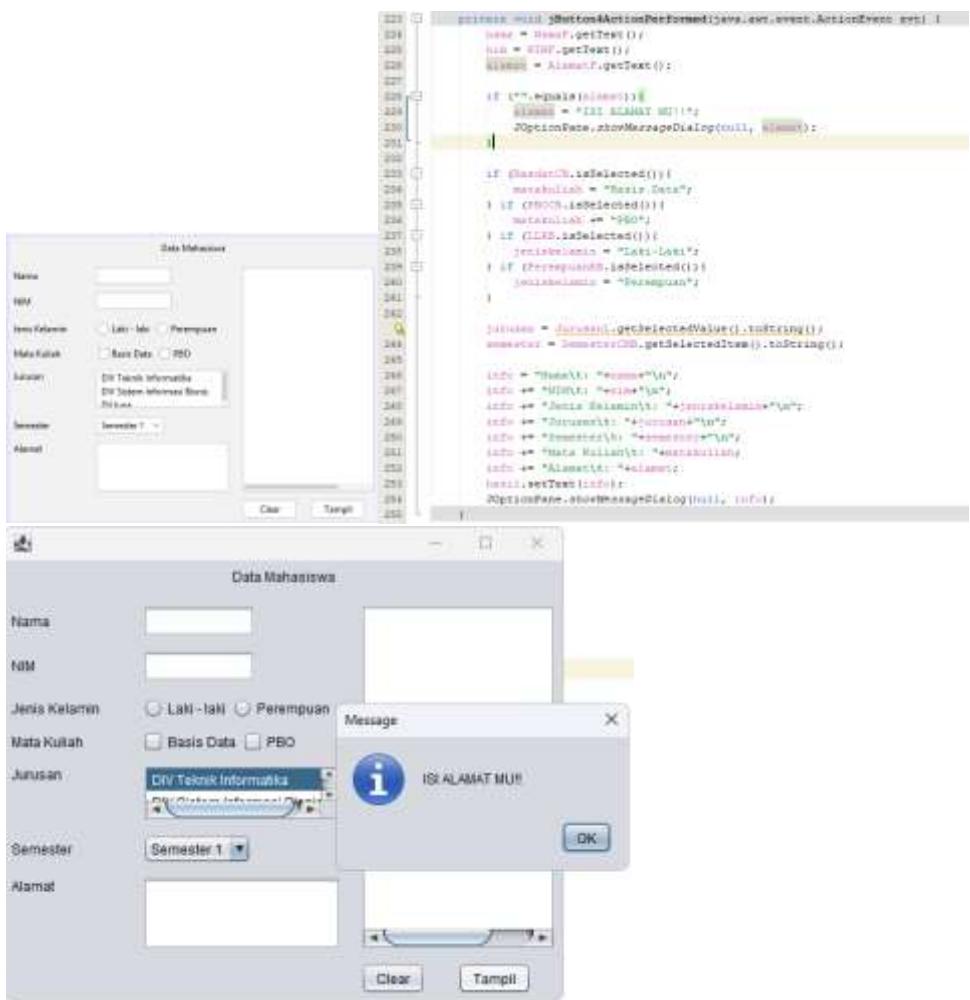
Experiment 4

Code



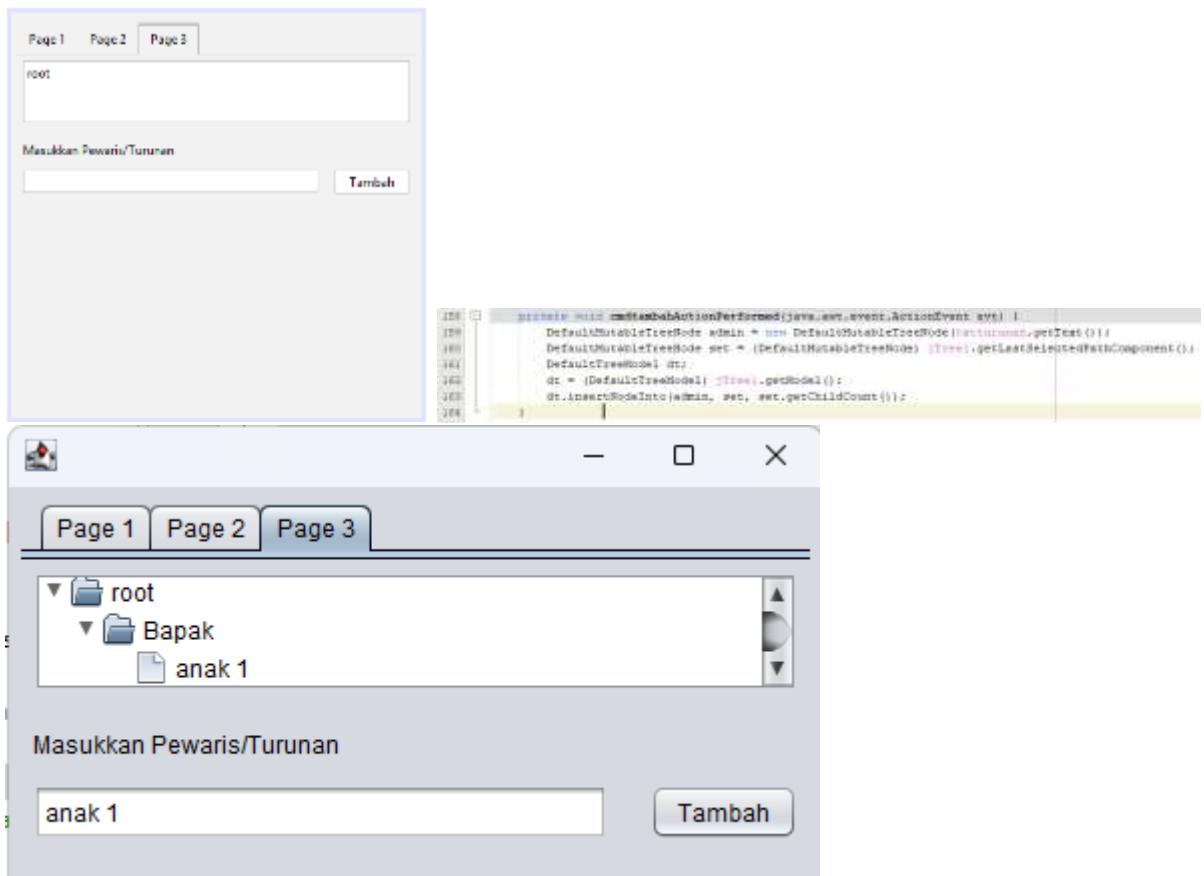
Questions

1. It ensures the application starts safely and correctly.
2. Because the radio buttons and checkbox have 2 state true and false
3. Just add another text field (also declare variables and add in the clear button) and if logic like this



Experiment 5

Code

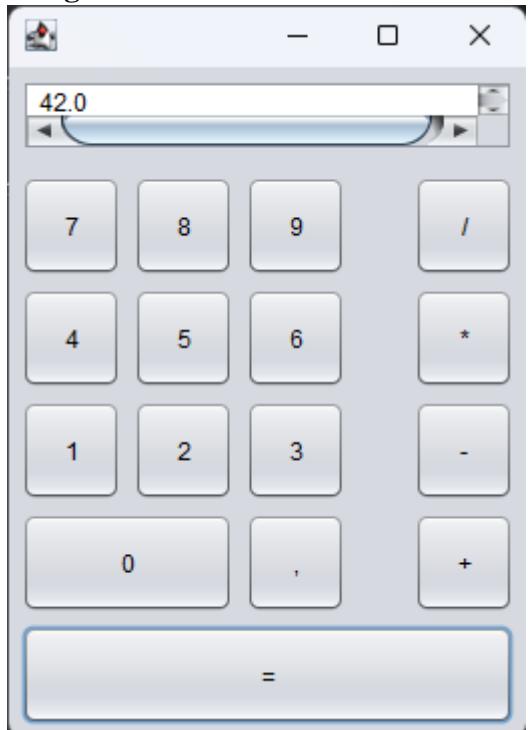


Questions

1. To make tabs
2. Just add the table like this

Title 1	Title 2	Title 3	Title 4

Assignment



the program is on github

<https://github.com/ZidAvwa/CollegeStudy/tree/main/3rdSemester/13.%20Jobsheet13/Assignment>