

Praktikum : SWING

Setoran	:	Source code poin 6.
Batas Waktu Penyetoran	:	Selasa 18 November 2020, pukul 17.00
Tempat Penyetoran	:	ecourse

Perhatian:

Gunakan editor seperti notepad++ tidak (**belum**) diperkenankan menggunakan IDE misalnya Netbeans.

1. Menggunakan *Top Level Container*

a. Menampilkan *frame* dengan *extends* kelas `JFrame`.

```
1 //Filename: JFrameDemoExtends.java
2 import javax.swing.JFrame;
3 public class JFrameDemoExtends extends JFrame {
4     public JFrameDemoExtends(String title){
5         super(title);
6     }
7 }
```

```
1 //JFrameDemoExtendsTest.java
2 import javax.swing.JFrame;
3 public class JFrameDemoExtendsTest extends JFrame {
4     public static void main(String[] args) {
5         JFrameDemoExtends myJFrame = new JFrameDemoExtends("Demo
extends JFrame");
6         //Coba kalian hilangkan dua baris berikut ini
7         myJFrame.setSize(500,300);
8         myJFrame.setVisible(true);
9     }
10 }
```

Penting!

Ketika akan menampilkan *Frame* ke layar, **jangan lupa** memanggil *method* `setSize()` untuk mengatur ukuran *container* dan *method* `setVisible(true)` dari sebuah *top level container*.

Perhatikan ketika anda menutup aplikasi, program masih berjalan. Agar program diterminasi ketika aplikasi ditutup, tambahkan kode program berikut ini setelah baris

8 pada program `JFrameDemoExtendsTest.java`

```
myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

b. Menampilkan *frame* tanpa *extends*

```
1 //Filename: JFrameDemo.java
2 import javax.swing.JFrame;
3
4 class JFrameDemo {
5     public static void main(String args[]) {
6         JFrame frame = new JFrame("JFrame Demo");
7
8         frame.setSize(200, 100);
9         frame.setVisible(true);
10        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
11    }
12 }
```

Silakan membuka *URL* ini untuk mendapatkan informasi tambahan:

<https://docs.oracle.com/javase/tutorial/uiswing/components/frame.html>

- Menambahkan *component* *JLabel* ke dalam *container*. Kalian bisa menggunakan cara 1a atau 1b untuk membuat *container*, pada contoh berikut ini digunakan cara 1b.

a. Menambahkan *JLabel*

```
1 //Filename: AppAddJLabel.java
2 import javax.swing.JFrame;
3 import javax.swing.JLabel;
4 public class AppAddJLabel {
5     private JFrame myFrame;
6     private JLabel myLabel;
7
8     public AppAddJLabel () {
9         myFrame = new JFrame ("Add JLabel");
10        myLabel = new JLabel("Isi label");
11
12        myFrame.setSize(200, 100);
13        myFrame.setVisible(true);
14        myFrame.setLocationRelativeTo(null);
15        myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
16    }
17    public void addComponent() {
18        //menambahkan label ke dalam container
19        myFrame.add(myLabel);
20    }
21    public static void main(String[] args) {
22        new AppAddJLabel().addComponent();
23    }
24 }
```

Catatan:

Baris 14: agar posisi *JFrame* berada ditengah layar.

Baris 22: instansisasi anonymus object (object tanpa nama).

Perhatikan ketika program dieksekusi ada yang aneh pada saat aplikasi ditampilkan. Pindahkan kode program yang ada di baris 14 ke baris 11.

Recompile dan eksekusi program tersebut. Perhatikan bedanya.

b. Menambahkan JButton

```
1 //Filename: AppAddJButton.java
2 import javax.swing.JFrame;
3 import javax.swing.JLabel;
4 import javax.swing.JButton;
5 public class AppAddJButton{
6     private JFrame myFrame;
7     private JLabel myLabel;
8     private JButton myButton;
9
10    public AppAddJButton(){
11        myFrame = new JFrame ("Add JLabel");
12        myLabel = new JLabel("Isi label");
13        myButton = new JButton("Click me");
14
15        myFrame.setLocationRelativeTo(null);
16        myFrame.setSize(200, 100);
17        myFrame.setVisible(true);
18        myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
19
20    }
21    public void addComponent(){
22        //menambahkan label dan button ke dalam container
23        myFrame.add(myLabel);
24        myFrame.add(myButton);
25    }
26    public static void main(String[] args) {
27        new AppAddJButton().addComponent();
28    }
29 }
```

Perhatikan tampilan yang muncul, *label* tidak terlihat pada *frame*. Hal ini terjadi karena belum ditentukan *layout manager* pada *frame* sehingga komponen yang terakhir di-add yang terlihat.

c. Menambahkan JTextField

```
1 //Filename: AppAddJTextField.java
2 import javax.swing.JFrame;
3 import javax.swing.JLabel;
4 import javax.swing.JButton;
5 import javax.swing.JTextField;
6 public class AppAddJTextField{
7     private JFrame myFrame;
8     private JLabel myLabel;
9     private JButton myButton;
10    private JTextField myTextField;
11
12    public AppAddJTextField(){
13        myFrame = new JFrame ("Add JLabel");
14        myLabel = new JLabel("Isi label");
15        myButton = new JButton("Click me");
16        myTextField = new JTextField();
17
18        myFrame.setLocationRelativeTo(null);
19        myFrame.setSize(200,100);
20    }
21 }
```

```

19         myFrame.setVisible(true);
20         myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
21     }
22     public void addComponent(){
23         //menambahkan label dan button ke dalam container
24         myFrame.add(myLabel);
25         myFrame.add(myButton);
26         myFrame.add(myTextField);
27     }
28     public static void main(String[] args) {
29         new AppAddJTextField().addComponent();
30     }
31 }
32

```

Sama seperti program pada point 2b, *label* dan *button* tidak terlihat pada *frame*.

Supaya komponen tidak saling menimpa, perlu mengatur letak komponen pada *frame* menggunakan *Layout Manager*.

3. *Layout Manager*: mengatur *layout*(tata letak *component* di dalam *container*).

a. *FlowLayout*

```

1 //Filename: UseFlowLayout.java
2 import java.awt.FlowLayout;
3 import javax.swing.JFrame;
4 import javax.swing.JButton;
5
6 public class UseFlowLayout {
7     private JFrame myFrame;
8     public UseFlowLayout() {
9         myFrame = new JFrame("FlowLayout Demo");
10
11         myFrame.setVisible(true);
12         myFrame.setLocationRelativeTo(null);
13         myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14     }
15     public void addComponent(){
16         myFrame.add(new JButton("A"));
17         myFrame.add(new JButton("B"));
18         myFrame.add(new JButton("C"));
19         myFrame.add(new JButton("D"));
20         myFrame.add(new JButton("E"));
21         myFrame.add(new JButton("F"));
22     }
23     public void setLayout(){
24         myFrame.setLayout(new FlowLayout());
25     }
26     public void setPack(){
27         myFrame.pack();
28     }
29 }

```

```
29 }
```

```
1 //Filename: DemoFlowLayout.java
2 class DemoFlowLayout
3 {
4     public static void main(String[] args)
5     {
6         UseFlowLayout uf = new UseFlowLayout();
7         uf.setLayout();
8         uf.addComponent();
9         uf.setPack();
10    }
11 }
```

Catatan:

Method `setSize()` dapat diganti dengan method `pack()`: ukuran *container* akan otomatis disesuaikan agar seluruh komponen di dalamnya bisa ditampilkan.

Coba ganti kode program pada baris 9 dengan kode program pada baris 8 (demikian sebaliknya) pada file `DemoFlowLayout`, recompile dan eksekusi program tersebut. Perhatikan bedanya.

b. BorderLayout

```
1 //Filename: UseBorderLayout.java
2 import java.awt.BorderLayout;
3 import javax.swing.JFrame;
4 import javax.swing.JButton;
5
6 public class UseBorderLayout {
7     private JFrame myFrame;
8     public UseBorderLayout() {
9         myFrame = new JFrame("BorderLayout Demo");
10
11         myFrame.setVisible(true);
12         myFrame.setLocationRelativeTo(null);
13         myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14     }
15     public void addComponent() {
16         myFrame.add(new JButton("North"), BorderLayout.NORTH);
17         myFrame.add(new JButton("South"), BorderLayout.SOUTH);
18         myFrame.add(new JButton("East"), BorderLayout.EAST);
19         myFrame.add(new JButton("West"), BorderLayout.WEST);
20         myFrame.add(new JButton("Center"), BorderLayout.CENTER);
21     }
22     public void setLayout() {
23         myFrame.setLayout(new BorderLayout());
24     }
25     public void setPack() {
26         myFrame.pack();
27     }
28 }
```

```

1 //Filename: DemoBorderLayout.java
2 class DemoBorderLayout
3 {
4     public static void main(String[] args)
5     {
6         UseBorderLayout ub = new UseBorderLayout();
7         ub.setLayout();
8         ub.addComponent();
9         ub.setPack();
10    }
11 }

```

```

1 //Filename: DemoBorderLayout.java
2 class DemoBorderLayout
3 {
4     public static void main(String[] args)
5     {
6         UseBorderLayout ub = new UseBorderLayout();
7         ub.setLayout();
8         ub.addComponent();
9         ub.setPack();
10    }
11 }

```

c. GridLayout

```

1 //Filename: UseGridLayout.java
2 import java.awt.GridLayout;
3 import javax.swing.JFrame;
4 import javax.swing.JButton;
5
6 public class UseGridLayout{
7     private JFrame myFrame;
8     public UseGridLayout() {
9         myFrame = new JFrame("GridLayout Demo");
10
11         myFrame.setVisible(true);
12         myFrame.setLocationRelativeTo(null);
13         myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14     }
15     public void addComponent(){
16         for (int x=1; x<13; x++)
17             myFrame.add(new JButton("" + x));
18     }
19     public void setLayout(){
20         myFrame.setLayout(new GridLayout(3,4));
21     }
22     public void setPack(){
23         myFrame.pack();
24     }
25 }

```

```

1 //Filename: DemoGridLayout.java
2 class DemoGridLayout
3 {

```

4	public static void main(String[] args)
5	{
6	UseGridLayout ug = new UseGridLayout();
7	ug.setLayout();
8	ug.addComponent();
9	ug.setPack();
10	}
11	}

d. GridBagLayout

1	//Filename: UseGridBagLayout.java
2	import java.awt.GridBagLayout;
3	import java.awt.GridBagConstraints;
4	import javax.swing.JFrame;
5	import javax.swing.JButton;
6	
7	class UseGridBagLayout {
8	private JFrame myFrame;
9	private GridBagConstraints gb;
10	public UseGridBagLayout() {
11	myFrame = new JFrame("GridBagLayout Demo");
12	
13	gb = new GridBagConstraints();
14	
15	myFrame.setLocationRelativeTo(null);
16	myFrame.setVisible(true);
17	myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
18	}
19	
20	public void addComponent(){
21	gb.gridwidth = 1;
22	gb.gridheight = 1;
23	gb.gridx = 0;
24	gb.gridy = 0;
25	gb.fill = GridBagConstraints.BOTH;
26	myFrame.add(new JButton("First"), gb);
27	
28	gb.gridx = 1;
29	myFrame.add(new JButton("Second"), gb);
30	
31	gb.gridx = 2;
32	myFrame.add(new JButton("Third"), gb);
33	
34	gb.gridx = 0;
35	gb.gridy++;
36	gb.gridwidth = 3;
37	myFrame.add(new JButton("Fourth"), gb);
38	
39	gb.gridy++;
40	gb.gridwidth = 1;
41	myFrame.add(new JButton("Fifth"), gb);
42	
43	gb.gridx = 1;
44	gb.gridwidth = 2;
45	gb.gridheight = 2;

```

46         myFrame.add(new JButton("Sixth"), gb);
47
48         gb.gridx = 0;
49         gb.gridy++;
50         gb.gridwidth = 1;
51         gb.gridheight = 1;
52         myFrame.add(new JButton("Seventh"), gb);
53     }
54     public void setLayout() {
55         myFrame.setLayout(new GridBagLayout());
56     }
57     public void setPack() {
58         myFrame.pack();
59     }
60 }
61

```

```

1 //Filename: DemoGridBLayout.java
2 class DemoGridBLayout
3 {
4     public static void main(String[] args)
5     {
6         UseGridBagLayout ugb = new UseGridBagLayout();
7         ugb.setLayout();
8         ugb.addComponent();
9         ugb.setPack();
10    }
11 }

```

Catatan:

- *Default Layout* pada JFrame adalah BorderLayout, itulah sebabnya hanya satu *component* yang terlihat pada point 2.
- **gridx, gridy**: menentukan posisi komponen dengan menggunakan posisi *row* dan *column*, dimulai dari sudut kiri atas.
- **gridwidth, gridheight**: menentukan jumlah *column* (untuk *gridwidth*) atau *row* (untuk *gridheight*) yang digunakan oleh sebuah komponen. Contoh: lihat tombol Fourth dan Sixth.
- Silakan baca lebih lanjut mengenai GridBagLayout di <http://java.del.ac.id/tutorial/uising/layout/gridbag.html>.

4. Menggunakan *container* yang lain (bukan top level *container*) untuk mengelompokkan komponen.

Untuk mengelompokkan komponen-komponen di dalam *container* dapat menggunakan JPanel.

```

//Filename: AppUseJPanel.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class AppUseJPanel
{

```



```

private JFrame myFrame;
private JTextField text;
private JButton[] button;

public AppUseJPanel(){
    myFrame = new JFrame("My calculator");

    myFrame.setLocationRelativeTo(null);
    myFrame.setVisible(true);
    myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

//setter
private void setText(String newText){
    text.setText(newText);
}
//getter
public String getText(){
    return text.getText();
}
public void setLayout(){
    myFrame.setLayout(new BorderLayout());
}
public void addComponent(){
    myFrame.setJMenuBar(makeMenu());
    myFrame.add(makePanelUp(), BorderLayout.NORTH);
    myFrame.add(makePanelBottom(), BorderLayout.CENTER);
}
public void setPack(){
    myFrame.pack();
}

public JMenuBar makeMenu(){
    JMenuBar mb = new JMenuBar();
    JMenu file = new JMenu("File");
    mb.add(file);
    JMenuItem reset = new JMenuItem("Reset");
    JMenuItem exit = new JMenuItem("Exit");
    file.add(reset);
    file.addSeparator();
    file.add(exit);

    JMenu view = new JMenu("View");
    JMenuItem standard = new JMenuItem("Standard");
    JMenuItem scientific = new JMenuItem("Scientific");
    view.add(standard);
    view.add(scientific);

    mb.add(view);
    return mb;
}
public JPanel makePanelUp(){
    JPanel panel = new JPanel();
    text = new JTextField(20);
    panel.add(text);
    return panel;
}

```

```

    }
    public JPanel makePanelBottom() {
        JPanel panel = new JPanel();
        panel.setLayout(new GridBagLayout());

        GridBagConstraints gb = new GridBagConstraints();
        panel.setLayout(new GridBagLayout());

        gb.gridwidth = 1;
        gb.gridheight = 1;
        gb.gridx = 0;
        gb.gridy = 0;
        gb.fill = GridBagConstraints.BOTH;
        panel.add(new JButton("First"), gb);

        gb.gridx = 1;
        panel.add(new JButton("Second"), gb);

        gb.gridx = 2;
        panel.add(new JButton("Third"), gb);

        gb.gridx = 0;
        gb.gridy++;
        gb.gridwidth = 3;
        panel.add(new JButton("Fourth"), gb);

        gb.gridy++;
        gb.gridwidth = 1;
        panel.add(new JButton("Fifth"), gb);

        gb.gridx = 1;
        gb.gridwidth = 2;
        gb.gridheight = 2;
        panel.add(new JButton("Sixth"), gb);

        return panel;
    }
}

```

```

//Filename: DemoJPanel.java
class DemoJPanel
{
    public static void main(String[] args)
    {
        AppUseJPanel uj = new AppUseJPanel();
        uj.setLayout();
        uj.addComponent();
        uj.setPack();
    }
}

```

5. Opsi lain dalam menentukan posisi komponen adalah menggunakan method `setBound()` seperti berikut ini:

```
//Filename: AppUseSetBound.java
import java.awt.BorderLayout;
import javax.swing.JFrame;
import javax.swing.JButton;
import javax.swing.JPanel;

public class AppUseSetBound {
    private JFrame myFrame;
    private JPanel myPanel;
    private JButton a;
    private JButton b;
    private JButton c;
    public AppUseSetBound() {
        myFrame = new JFrame("No Layout Demo");
        myPanel = new JPanel();
        a = new JButton("1");
        b = new JButton("2");
        c = new JButton("3");

        myFrame.setLocationRelativeTo(null);
        myFrame.setSize(240,220);
        myFrame.setVisible(true);
        myFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void addComponent() {
        myFrame.add(myPanel);
        myPanel.add(a);
        a.setBounds(70, 10, 59, 20);
        myPanel.add(b);
        b.setBounds(20, 90, 90, 23);
        myPanel.add(c);
        c.setBounds(20, 10, 50, 30);
    }

    public void setLayout() {
        //Jika menggunakan setbound harus setLayout null terlebih dahulu
        myPanel.setLayout(null);
    }

    public void setPack() {
        myFrame.pack();
    }
}
```

6. Latihan:

- a. Perbaiki kode program pada poin 2.c, sehingga seluruh komponen bisa ditampilkan.
- b. Buatlah program `AppUseSetBoundDriver.java` untuk poin 5.
- c. Silakan membuat tampilan berikut ini: Calculator

