# GUI Programming (1)

Teguh Sutanto, M.Kom.

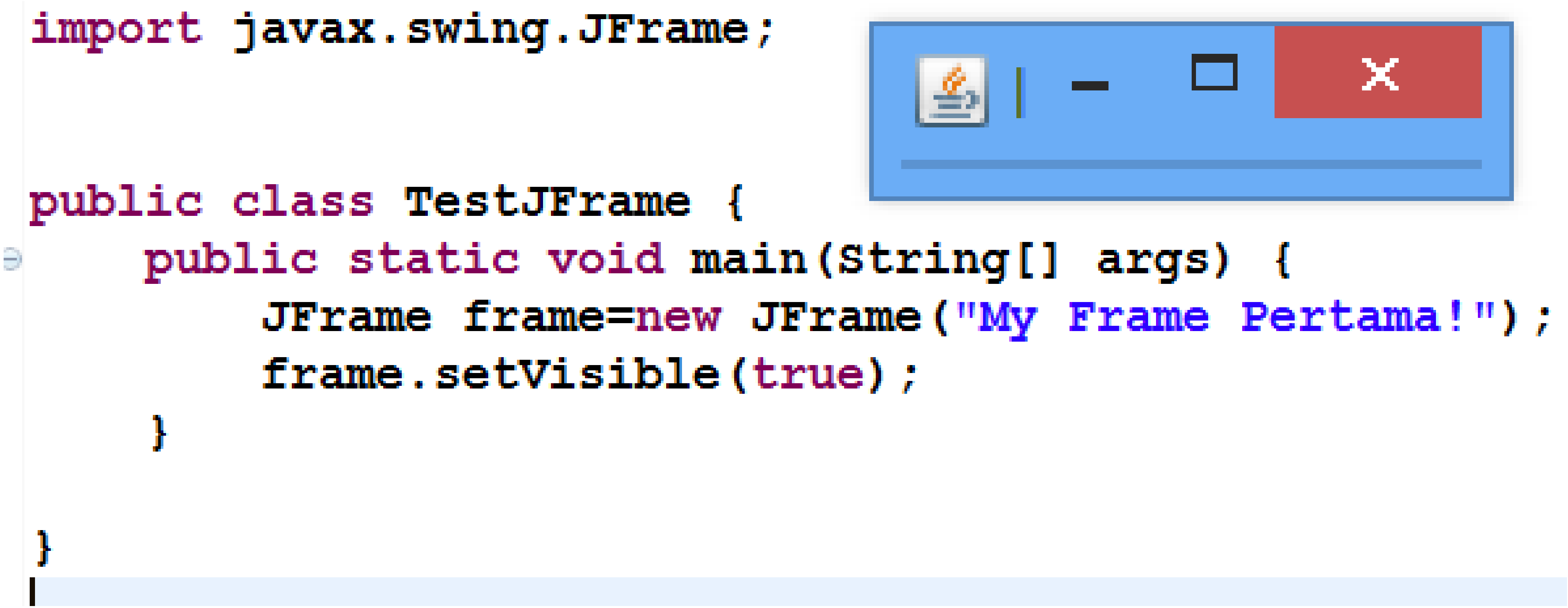
E-Mail: teguh@stikom.edu

Blog: http://blog.stikom.edu/teguh

# Tujuan

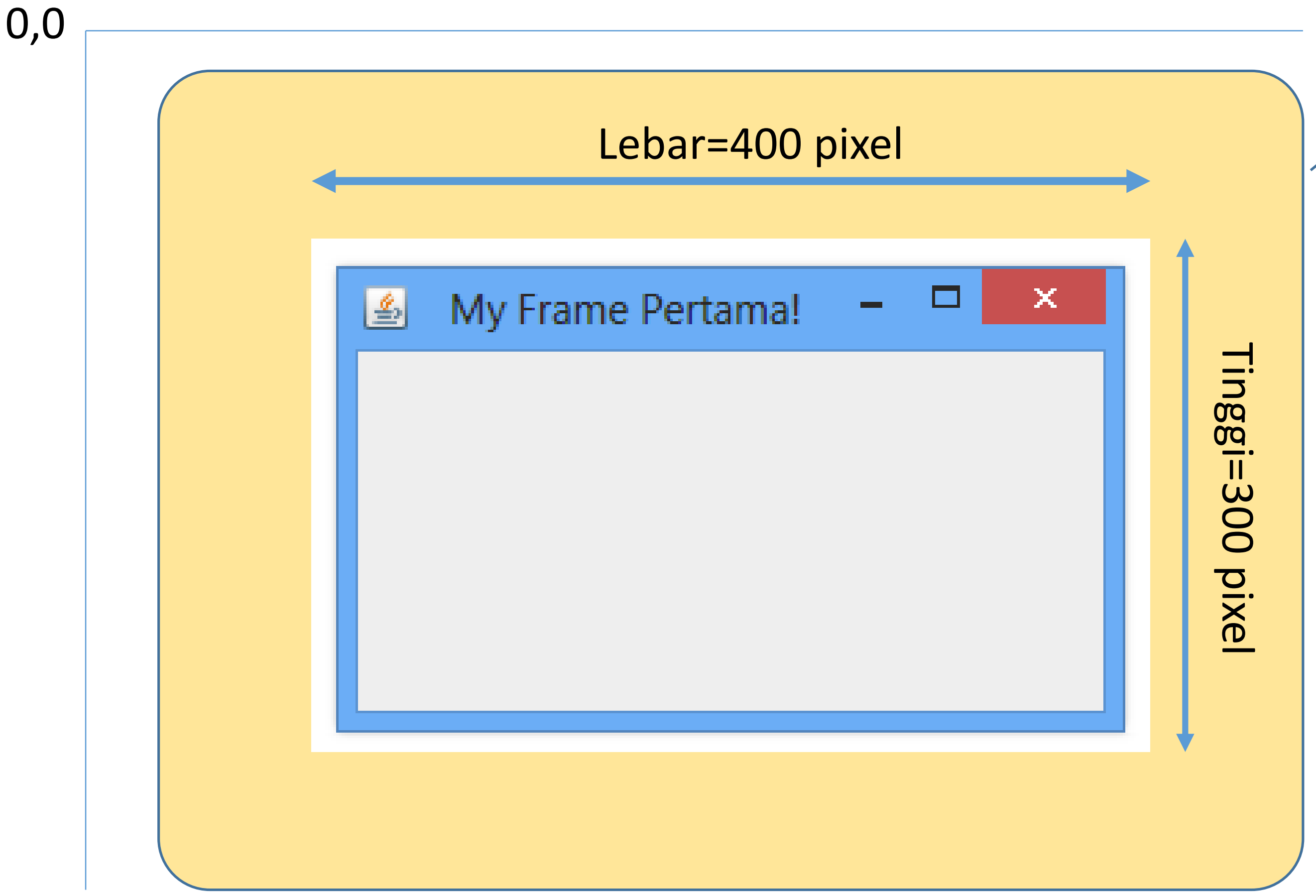
- Mahasiswa dapat membuat program GUI dengan Swing
- Mahasiwa dapat membuat program untuk menghandle penekanan Button
- Mahasiswa dapat mengatur layout

# Menggunakan JFrame

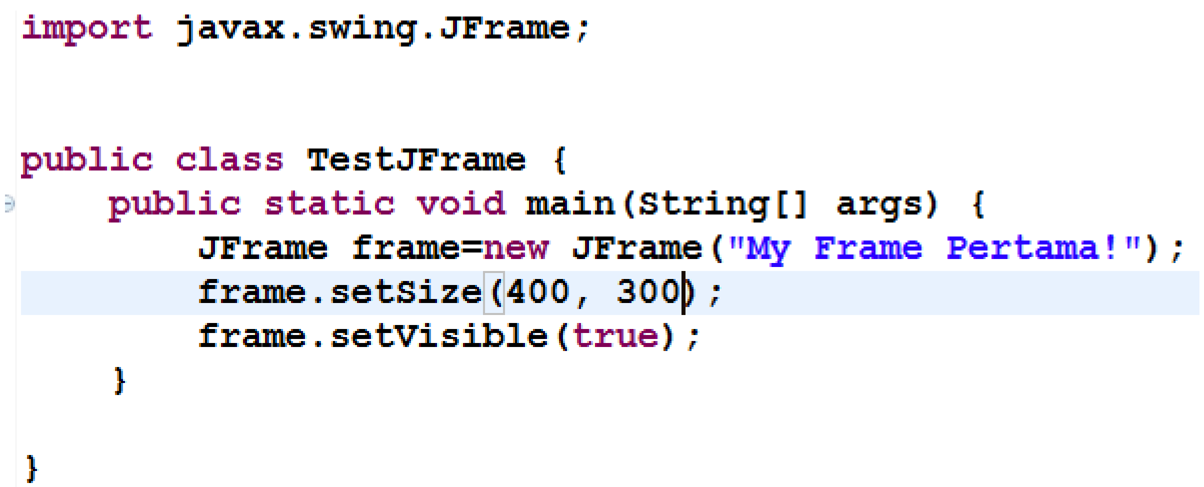


```
import javax.swing.JFrame;


public class TestJFrame {
    public static void main(String[] args) {
        JFrame frame=new JFrame("My Frame Pertama!");
        frame.setVisible(true);
    }

}
```

# Membuat Kustomisasi JFrame



Layar Monitor Komputer

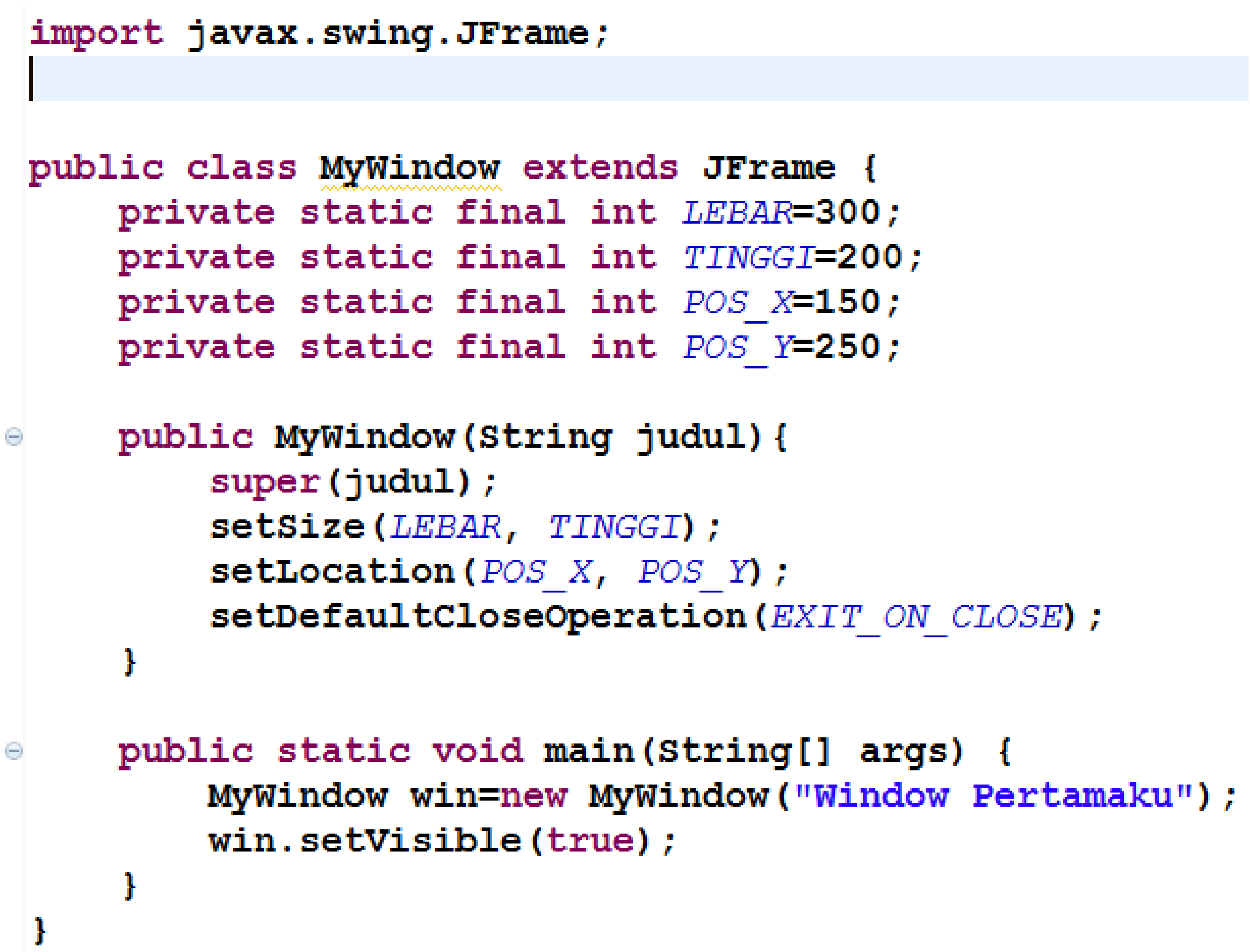


```
import javax.swing.JFrame;


public class TestJFrame {
    public static void main(String[] args) {
        JFrame frame=new JFrame("My Frame Pertama!");
        frame.setSize(400, 300);
        frame.setVisible(true);
    }

}
```

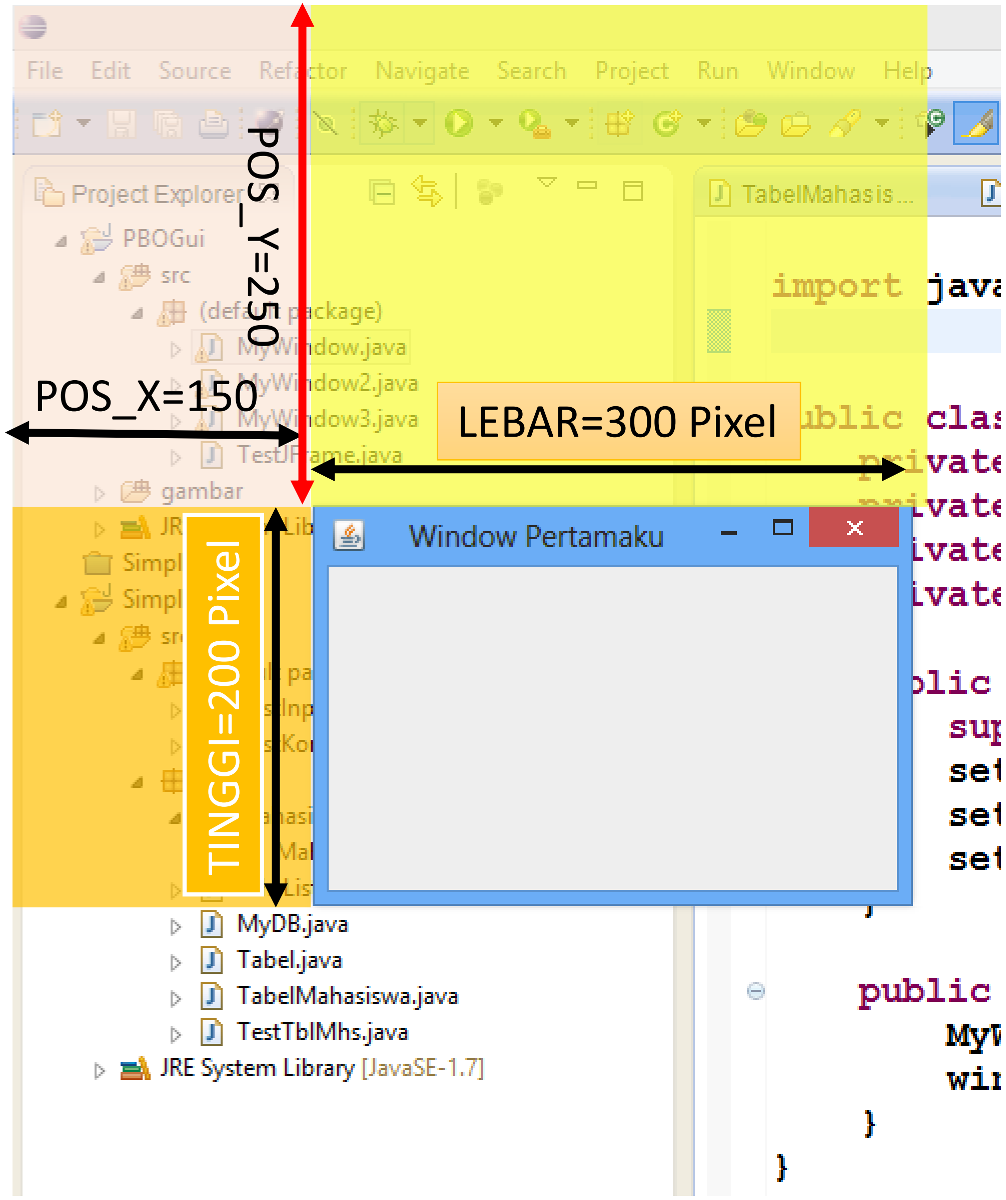
# Membuat Class Turunan JFrame

- Judul window = “Window Pertamaku”
- Program akan berhenti (exit) ketika Close box diclick
- Ukuran frame
  - Lebar = 300 pixel
  - Tinggi = 200 pixel
- Posisi
  - X=150
  - Y=250

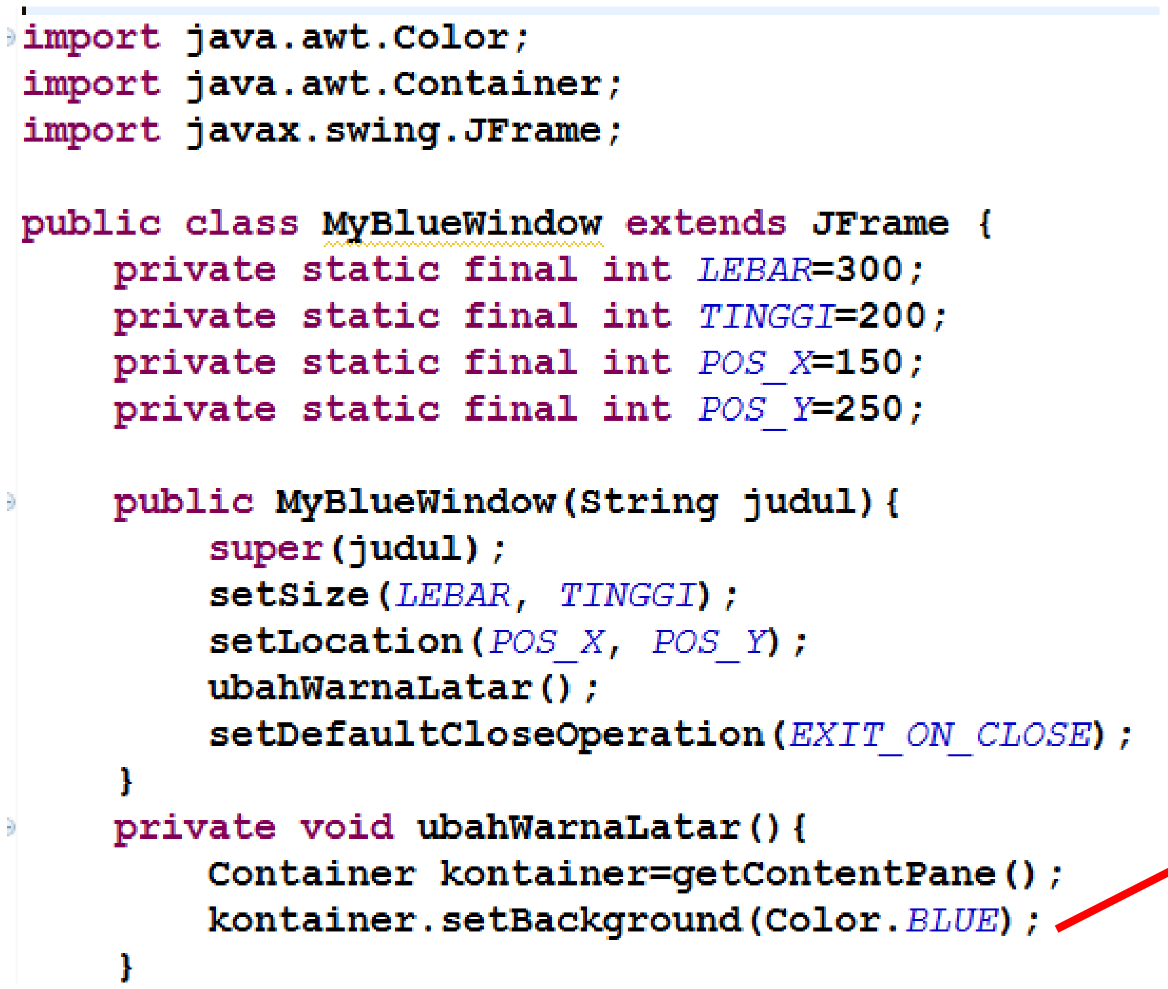
# Implementasi



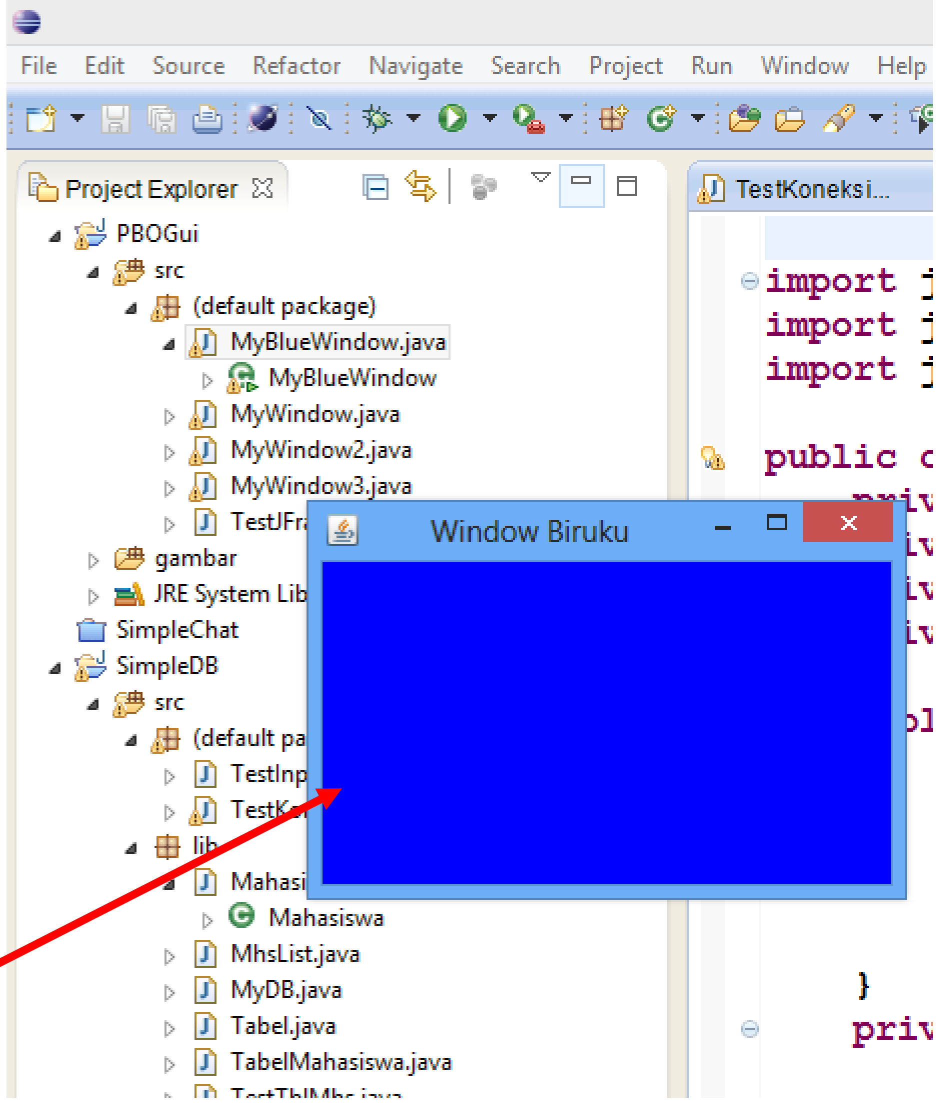
```
import javax.swing.JFrame;

public class MyWindow extends JFrame {
    private static final int LEBAR=300;
    private static final int TINGGI=200;
    private static final int POS_X=150;
    private static final int POS_Y=250;

    public MyWindow(String judul){
        super(judul);
        setSize(LEBAR, TINGGI);
        setLocation(POS_X, POS_Y);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }

    public static void main(String[] args) {
        MyWindow win=new MyWindow("Window Pertamaku");
        win.setVisible(true);
    }
}
```



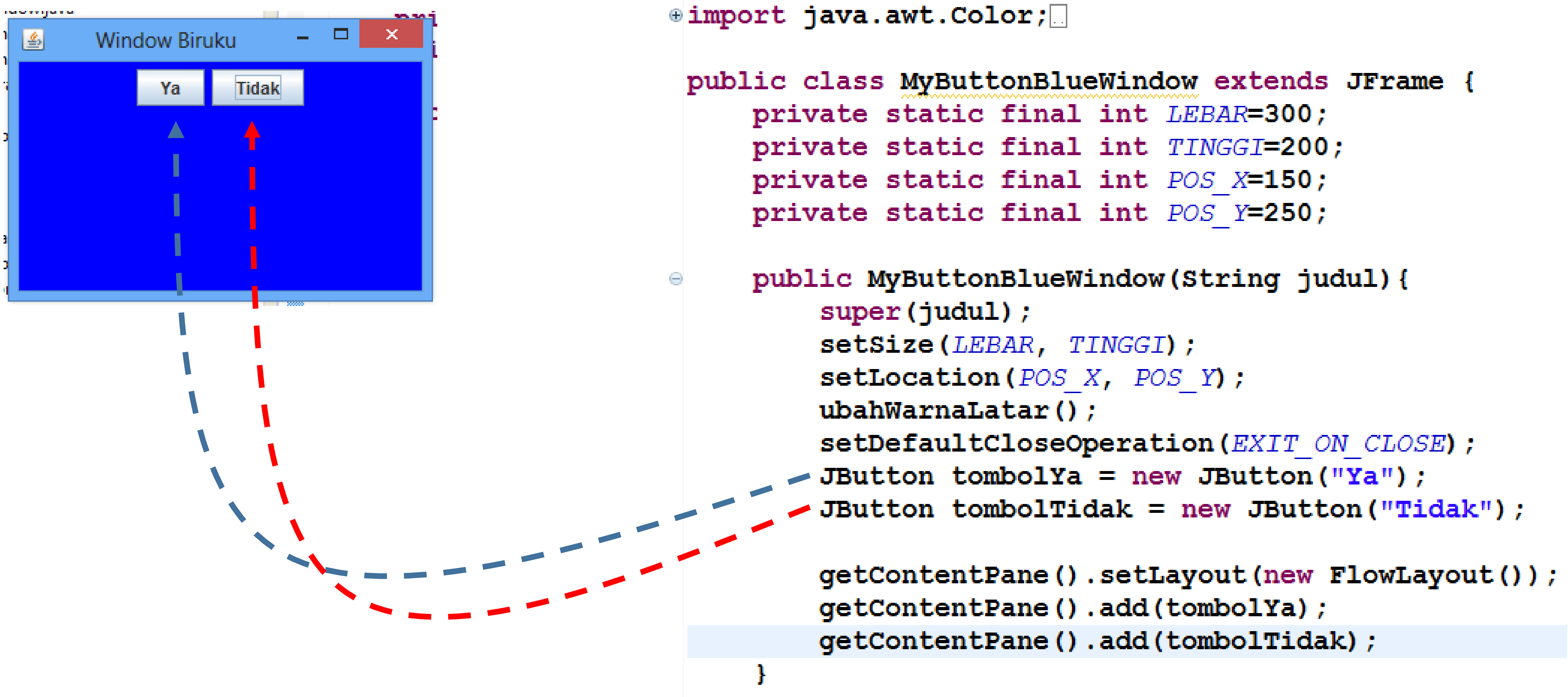
# MyBlueWindow



```
import java.awt.Color;
import java.awt.Container;
import javax.swing.JFrame;

public class MyBlueWindow extends JFrame {
    private static final int LEBAR=300;
    private static final int TINGGI=200;
    private static final int POS_X=150;
    private static final int POS_Y=250;

    public MyBlueWindow(String judul){
        super(judul);
        setSize(LEBAR, TINGGI);
        setLocation(POS_X, POS_Y);
        ubahWarnaLatar();
        setDefaultCloseOperation(EXIT_ON_CLOSE);
    }
    private void ubahWarnaLatar(){
        Container kontainer=getContentPane();
        kontainer.setBackground(Color.BLUE);
    }
```



# Menambahkan Button

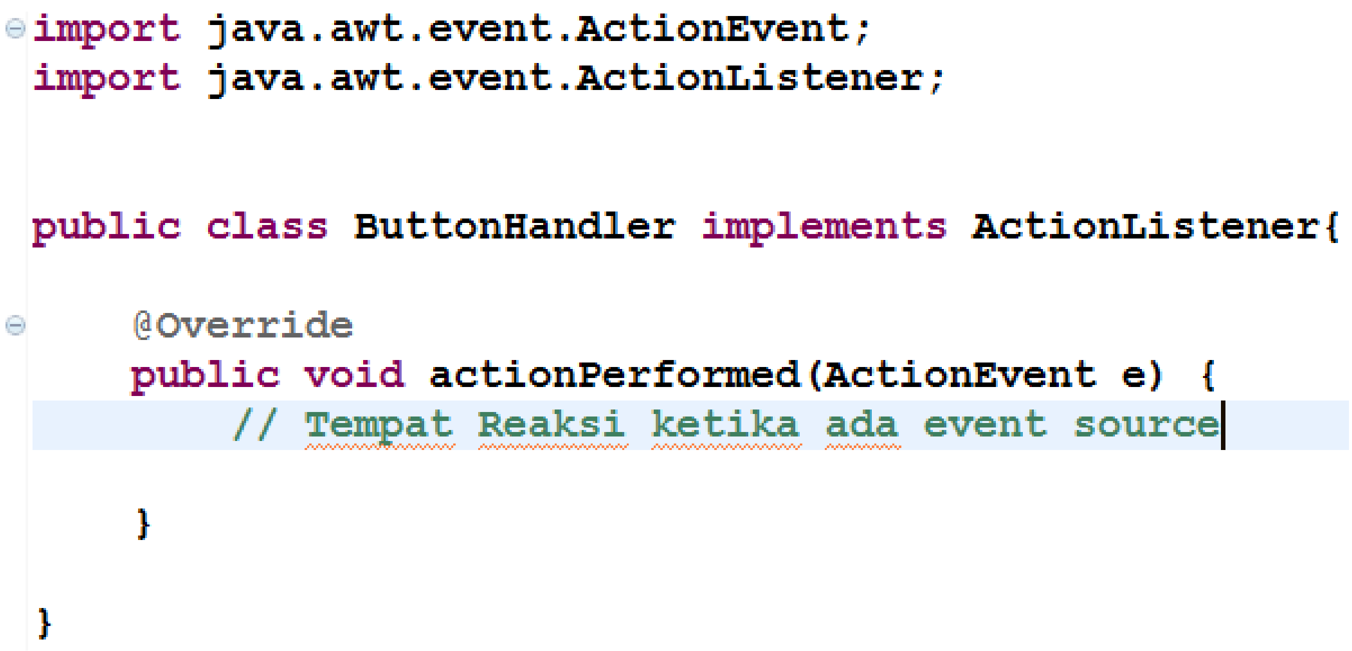


```
import java.awt.Color;

public class MyButtonBlueWindow extends JFrame {
    private static final int LEBAR=300;
    private static final int TINGGI=200;
    private static final int POS_X=150;
    private static final int POS_Y=250;

    public MyButtonBlueWindow(String judul){
        super(judul);
        setSize(LEBAR, TINGGI);
        setLocation(POS_X, POS_Y);
        ubahWarnaLatar();
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        JButton tombolYa = new JButton("Ya");
        JButton tombolTidak = new JButton("Tidak");

        getContentPane().setLayout(new FlowLayout());
        getContentPane().add(tombolYa);
        getContentPane().add(tombolTidak);
    }
```

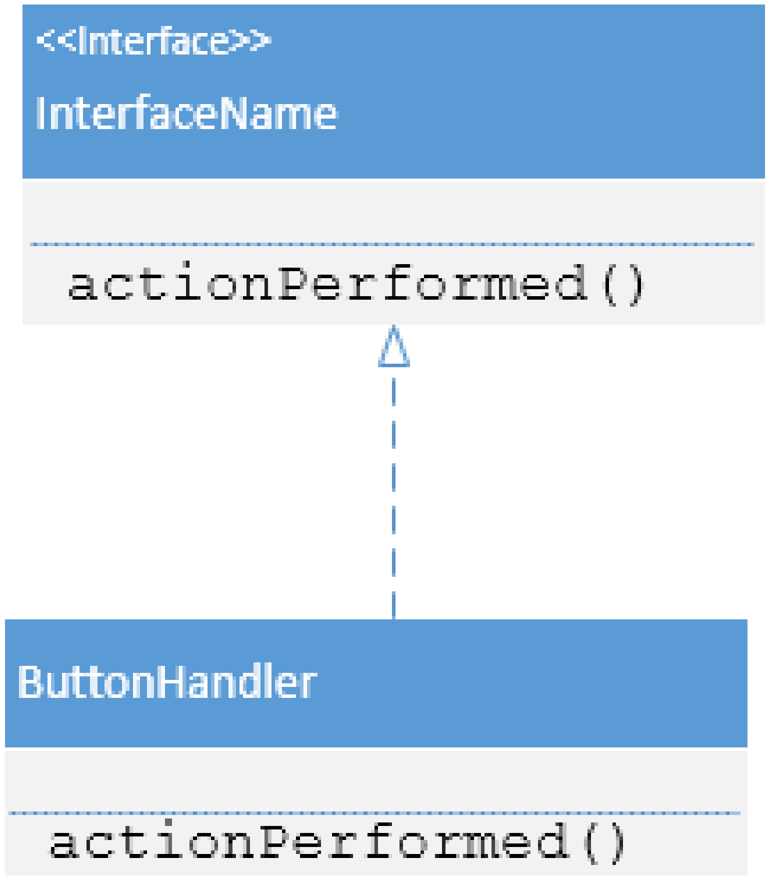
# Menambahkan Event Handling

- An action such as clicking a button is called an ***event***, and the mechanism to process the events *event handling*.
- Event handling :
  - event source objects→object GUI contoh button
  - event listener objects→object yang berisi method yang akan melakukan aksi ketika ada event source object

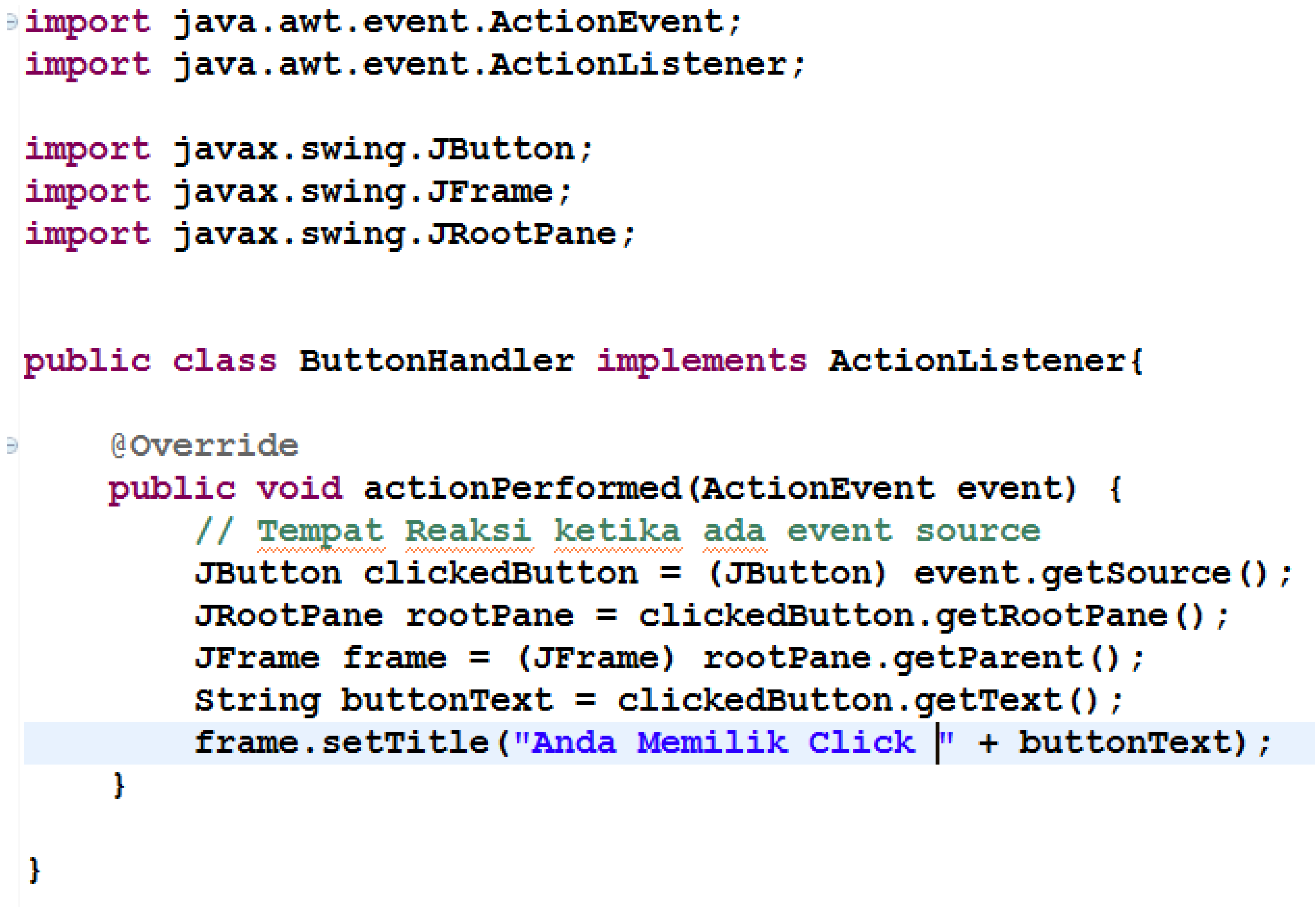
# Kerangka Object listener



```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;


public class ButtonHandler implements ActionListener{

    @Override
    public void actionPerformed(ActionEvent e) {
        // Tempat Reaksi ketika ada event source

    }

}
```

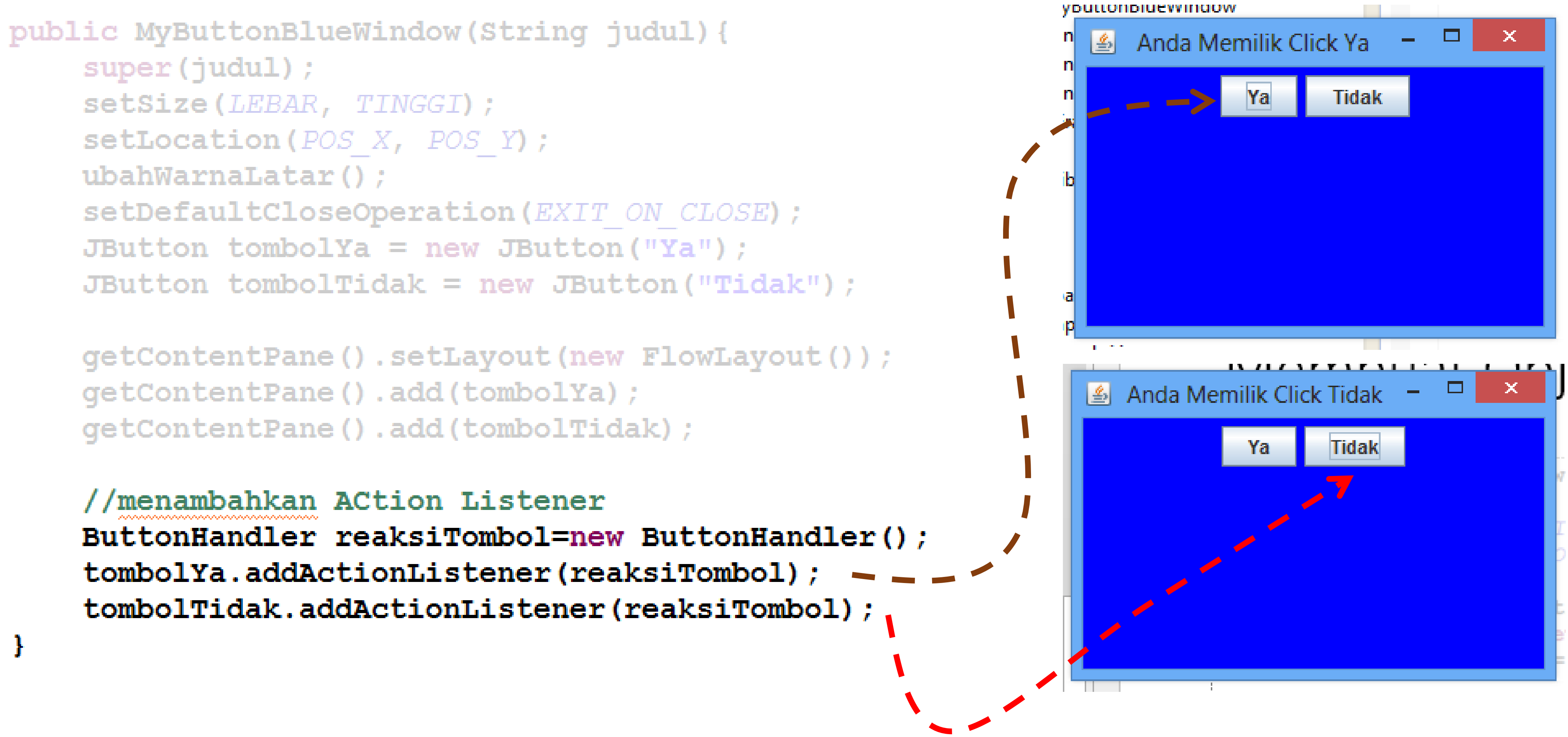


# Menambahkan Reaksi



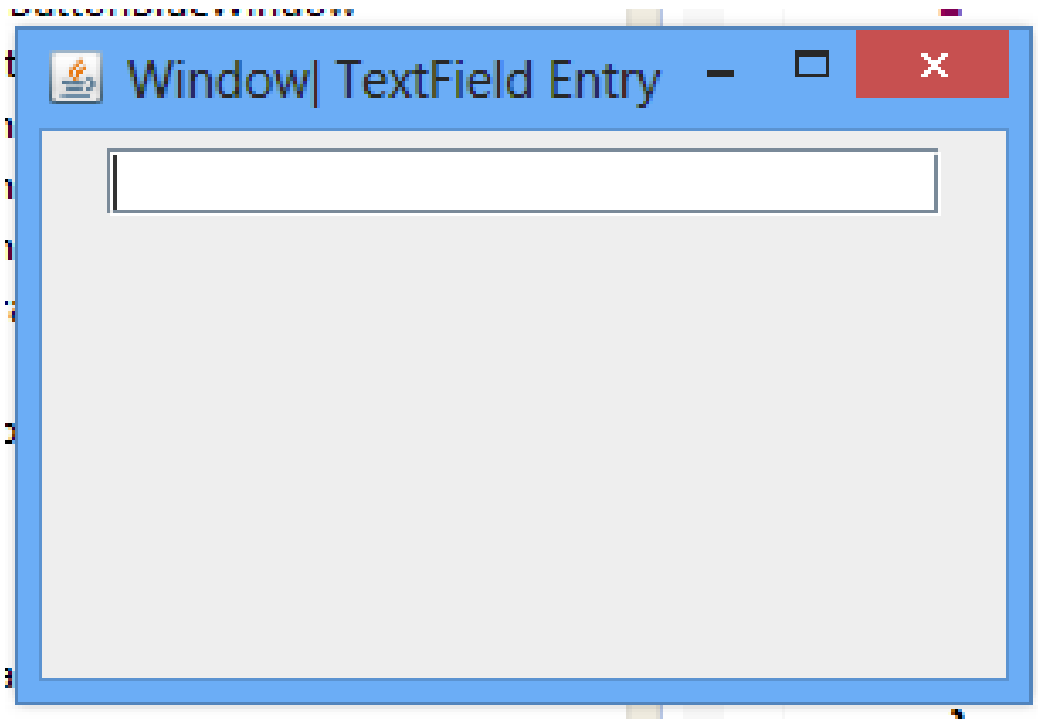
```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JRootPane;


public class ButtonHandler implements ActionListener{

    @Override
    public void actionPerformed(ActionEvent event) {
        // Tempat Reaksi ketika ada event source
        JButton clickedButton = (JButton) event.getSource();
        JRootPane rootPane = clickedButton.getRootPane();
        JFrame frame = (JFrame) rootPane.getParent();
        String buttonText = clickedButton.getText();
        frame.setTitle("Anda Memilik Click " + buttonText);
    }

}
```

# Membuat Object Listener



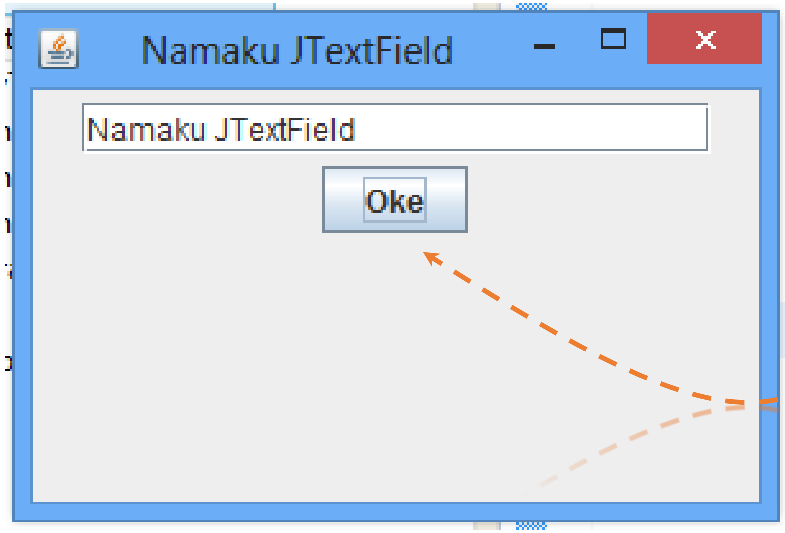
```
public MyButtonBlueWindow(String judul){
    super(judul);
    setSize(LEBAR, TINGGI);
    setLocation(POS_X, POS_Y);
    ubahWarnaLatar();
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    JButton tombolYa = new JButton("Ya");
    JButton tombolTidak = new JButton("Tidak");

    getContentPane().setLayout(new FlowLayout());
    getContentPane().add(tombolYa);
    getContentPane().add(tombolTidak);

    //menambahkan ACtion Listener
    ButtonHandler reaksiTombol=new ButtonHandler();
    tombolYa.addActionListener(reaksiTombol);
    tombolTidak.addActionListener(reaksiTombol);
}
```

# Menggunakan JTextField



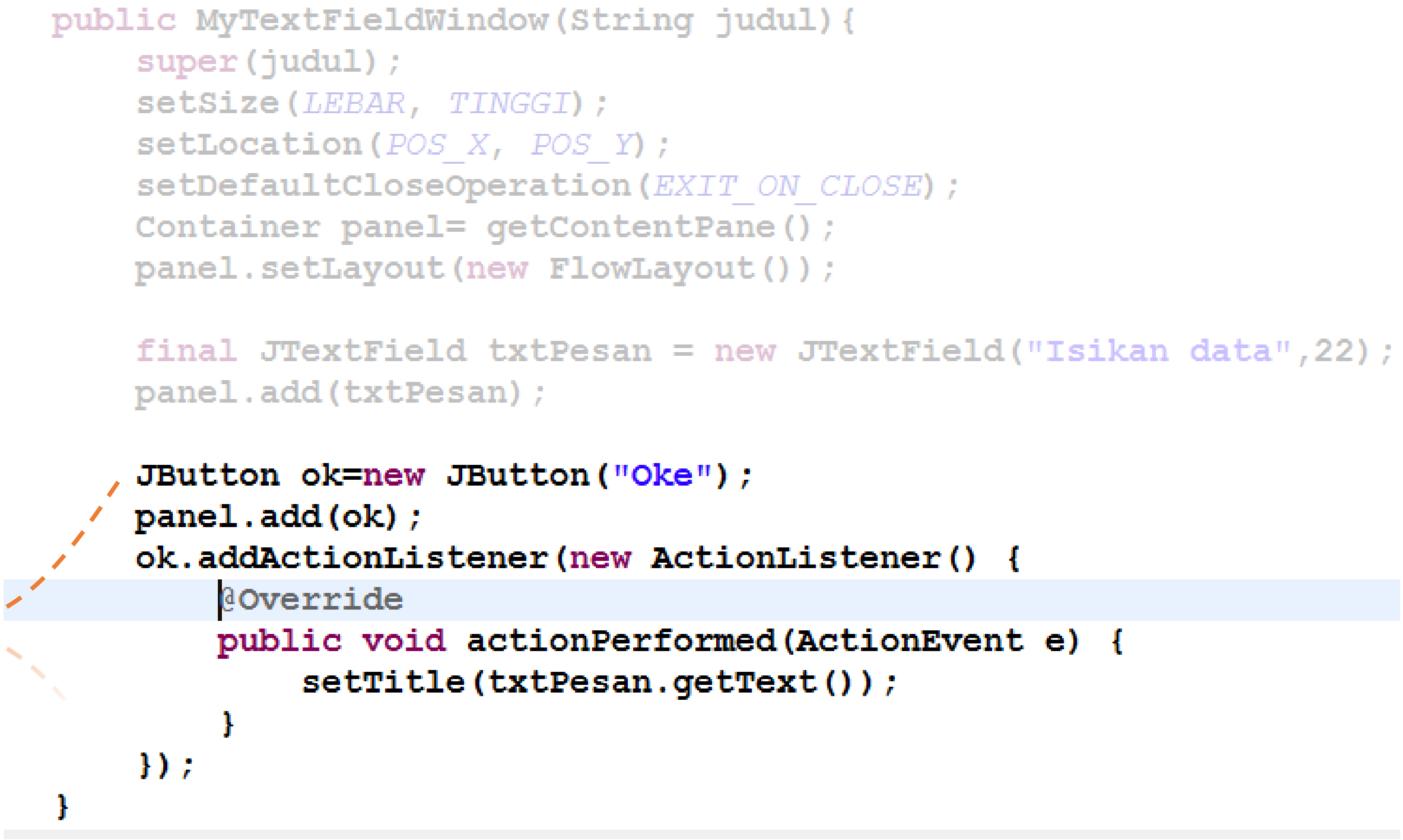
```
super(judul);
setSize(LEBAR, TINGGI);
setLocation(POS_X, POS_Y);
setDefaultCloseOperation(EXIT_ON_CLOSE);
Container panel= getContentPane();
panel.setLayout(new FlowLayout());

JTextField txtPesan = new JTextField(22);
panel.add(txtPesan);
```

# Menambahkan Button Oke



Namaku JTextField

Namaku JTextField



```
public MyTextFieldWindow(String judul){
    super(judul);
    setSize(LEBAR, TINGGI);
    setLocation(POS_X, POS_Y);
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    Container panel= getContentPane();
    panel.setLayout(new FlowLayout());

    final JTextField txtPesan = new JTextField("Isikan data",22);
    panel.add(txtPesan);

    JButton ok=new JButton("Oke");
    panel.add(ok);
    ok.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            setTitle(txtPesan.getText());
        }
    });
}
```

# Next

- Dengan cara yang sama seperti slide sebelumnya, maka tambahkan object Jlabel