

**TUGAS PRAKTIKUM  
PEMOGRAMAN II**



DISUSUN OLEH :

**Pramudya Dimas**

**223040005**

***TEKNIK INFORMATIKA  
UNIVERSITAS  
PASUNDAN BANDUNG  
2024***

1. Buatlah sebuah kelas bernama ButtonExample dengan kode berikut.  
Code :

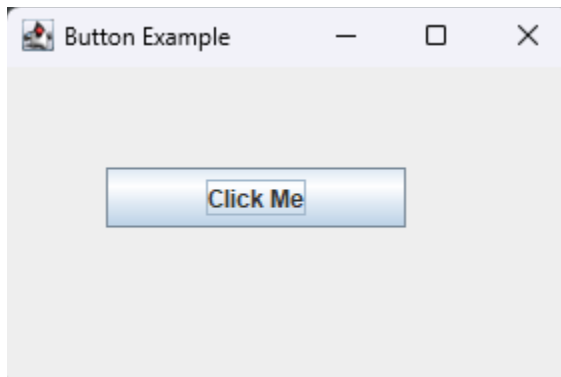
Code :

```

1 package Pertemuan4;
2
3 > import javax.swing.*; ...
4
5
6 public class ButtonExample {
7
8     public static void main(String[] args) {
9         JFrame frame = new JFrame("Button Example");
10        JButton button = new JButton("Click Me");
11
12        // Menambahkan ActionListener ke JButton
13        button.addActionListener(new ActionListener()
14        {
15            public void actionPerformed(ActionEvent e)
16            {
17                System.err.println("Button Clicked!");
18            }
19        });
20
21        button.setBounds(50, 50, 150, 30);
22        frame.add(button);
23        frame.setSize(300, 200);
24        frame.setLayout(null);
25        frame.setVisible(true);
26        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
27    }
28 }

```

Hasil :



Penjelasan :

Kode di atas membuat antarmuka grafis dengan Java Swing yang menampilkan tombol "Click Me". Ketika tombol diklik, pesan "Button clicked!" ditampilkan di konsol. Program ini menggunakan JFrame sebagai window utama dan JButton sebagai tombol, serta menambahkan *ActionListener* untuk menangani klik tombol. Ukuran dan posisi tombol diatur secara manual, dan window akan ditutup saat pengguna menutupnya.

2. Buat kelas bernama `MouseListenerExample` dengan kode sebagai berikut.

Code :

```
You, 5 days ago | 1 author (You)
package Pertemuan4;

import java.awt.event.*;
import javax.swing.*;

You, 5 days ago | You, 5 days ago | 1 author (You) | 1 author (You)
Run main | Debug main | Run | Debug
You, 5 days ago * Pertemuan 4

public class MouseListenerExample {

    public static void main(String[] args) {
        // Membuat JFrame
        JFrame frame = new JFrame(title:"MouseListener Example");

        // Membuat Label untuk menampilkan pesan
        JLabel label = new JLabel(text:"Arahkan dan klik mouse pada area ini.");
        label.setBounds(x:50, y:50, width:300, height:30);

        // Menambahkan MouseListener ke label
        You, 5 days ago | 1 author (You)
        label.addMouseListener(new MouseListener() {
            // Di Jalankan ketika mouse di klik (klik kiri, kanan atau tengah)
            public void mouseClicked(MouseEvent e) {
                label.setText("Mouse Clicked at: (" + e.getX() + ", " + e.getY() + ")");
            }

            // Di Jalankan ketika mouse ditekan (tanpa dilepaskan)
            public void mousePressed(MouseEvent e) {
                label.setText("Mouse Clicked at: (" + e.getX() + ", " + e.getY() + ")");
            }

            // Di Jalankan ketika mouse dilepaskan setelah ditekan
            public void mouseReleased(MouseEvent e) {
                label.setText("Mouse Clicked at: (" + e.getX() + ", " + e.getY() + ")");
            }

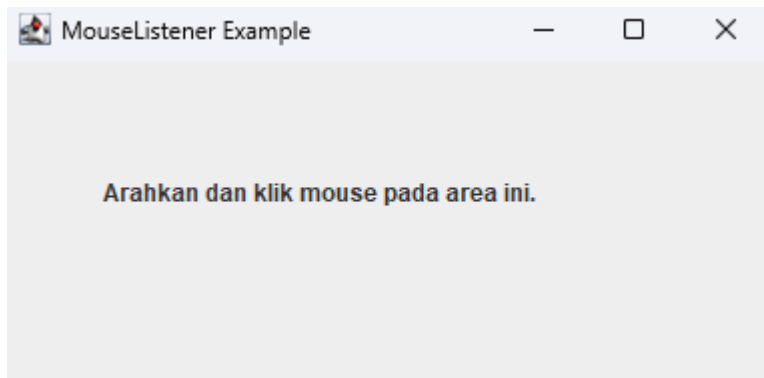
            // Di Jalankan ketika mouse masuk ke area komponen
            public void mouseEntered(MouseEvent e) {
                label.setText(text:"Mouse Entered the area.");
            }

            public void mouseExited(MouseEvent e) {
                label.setText(text:"Mouse Exited the area.");
            }
        });
    }
}
```

```
// Menambahkan Komponen ke frame
frame.add(label);
frame.add(textField);

// Setting frame
frame.setSize(width:400, height:200);
frame.setLayout(manager:null);
frame.setVisible(b:true);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

Hasil :



Penjelasan :

Kode di atas membuat program Java yang menggunakan *MouseListener* untuk menangani event mouse. Ketika mouse berinteraksi dengan label, event seperti klik, tekan, rilis, masuk, dan keluar dari area label akan memicu perubahan teks pada label. Program ini menampilkan posisi koordinat mouse atau pesan sesuai dengan event yang terjadi. *JFrame* digunakan sebagai window, dan *JLabel* menampilkan teks di dalamnya.

3. Buat kelas bernama KeyListenerExample dengan kode sebagai berikut.

Code :

```
You, 5 days ago | 1 author (You)
package Pertemuan4;

import java.awt.event.*;
import javax.swing.*;

You, 5 days ago | You, 5 days ago | 1 author (You) | 1 author (You)
public class KeyListenerExample {

    Run main | Debug main | Run | Debug
    public static void main(String[] args){
        // Membuat frame
        JFrame frame = new JFrame(title:"KeyListener Example");

        // Membuat label untuk menampilkan pesan
        JLabel label = new JLabel(text:"Tekan tombol pada keyboard");

        // Membuat text field untuk fokus keyboard
        JTextField textField = new JTextField();
        textField.setBounds(x:50, y:100, width:200, height:30);

        // Menambahkan KeyListener ke text field
        You, 5 days ago | 1 author (You)
        textField.addKeyListener(new KeyListener() {
            // Dijalankan ketika tombol ditekan
            @Override
            public void keyPressed(KeyEvent e) {
                label.setText("Key Pressed: " + KeyEvent.getKeyText(e.getKeyCode()));
            }

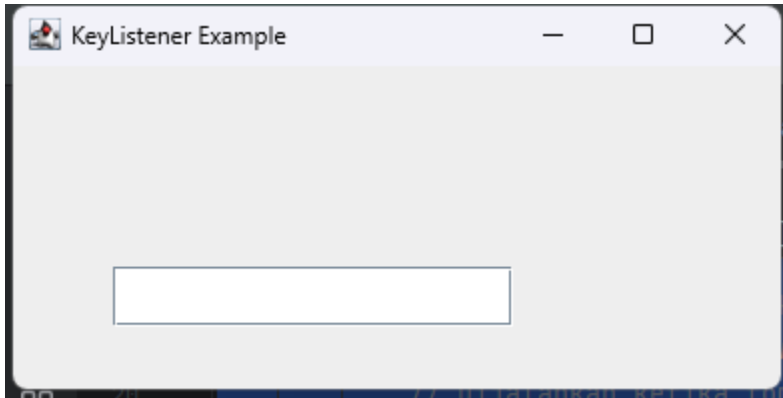
            // Dijalankan ketika tombol dilepaskan
            @Override
            public void keyReleased(KeyEvent e) {
                label.setText("Key Released: " + KeyEvent.getKeyText(e.getKeyCode()));
            }

            // Dijalankan ketika tombol ditekan dan dilepaskan (sama dengan mengetik karakter)
            @Override
            public void keyTyped(KeyEvent e) {
                label.setText("Key Typed: " + e.getKeyChar());
            }
        });
        You, 5 days ago * Pertemuan 4

        // Menambahkan komponen ke frame
        frame.add(label);
        frame.add(textField);

        // Setting frame
        frame.setSize(width:400, height:200);
        frame.setLayout(manager:null);
        frame.setVisible(b:true);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

Hasil :



Penjelasan :

Kode di atas membuat program Java yang menggunakan *KeyListener* untuk menangani event keyboard. Saat pengguna menekan, melepaskan, atau mengetik karakter di text field, pesan yang menunjukkan tombol yang ditekan atau karakter yang diketik akan ditampilkan di label. JFrame digunakan sebagai window utama, JLabel untuk menampilkan pesan, dan JTextField untuk menangkap input keyboard.

4. Buat kelas bernama WindowListenerExample dengan kode sebagai berikut.

Code :

```
You, 5 days ago | 1 author (You)
package Pertemuan4;

import java.awt.event.*;
import javax.swing.*;

You, 5 days ago | You, 5 days ago | 1 author (You) | 1 author (You)
Run main | Debug main | Run | Debug
public class WindowListenerExample {
    public static void main(String[] args) {
        JFrame f = new JFrame(title:"Window Listener Example");

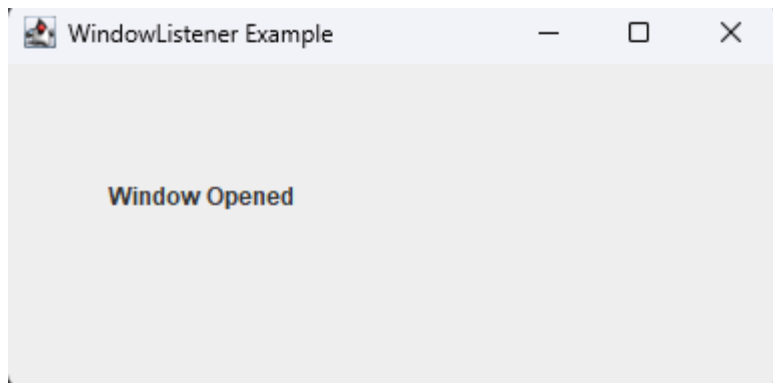
        JLabel label = new JLabel(text:"lakukan operasi pada jendela");
        label.setBounds(x:50,y:50,width:300, height:30);

        You, 5 days ago | 1 author (You)
        f.addWindowListener(new WindowListener() {

            public void windowOpened(WindowEvent e) {
                label.setText(text:"Window Opened");
            }
            public void windowClosing(WindowEvent e) {
                label.setText(text:"Window Closing");
            }
            public void windowClosed(WindowEvent e) {
                System.out.println(x:"Window Closed");
            }
            public void windowIconified(WindowEvent e) {
                label.setText(text:"Window Minimized");
            }
            public void windowDeiconified(WindowEvent e) {
                label.setText(text:"Window Restored");
            }
            public void windowActivated(WindowEvent e) {
                label.setText(text:"Window Activated");
            }
            public void windowDeactivated(WindowEvent e) {
                label.setText(text:"Window Deactivated");
            }
        });

        f.add(label);
        f.setSize(width:400, height:400);
        f.setLayout(manager:null);
        f.setVisible(b:true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

Hasil :



Penjelasan :

Kode di atas membuat program Java menggunakan *WindowListener* untuk menangani event pada jendela. Saat jendela dibuka, ditutup, diminimalkan, dipulihkan, diaktifkan, atau dinonaktifkan, pesan yang sesuai akan ditampilkan di label. *JFrame* digunakan sebagai window utama, dan *WindowListener* menangani event seperti "Window Opened", "Window Closing", dan lainnya.