

LAPORAN
PRAKTIKUM PEMOGRAMAN II
PERTEMUAN 4

Dosen Pengampu :
WANDA GUSDYA PURNAMA, ST., MT.

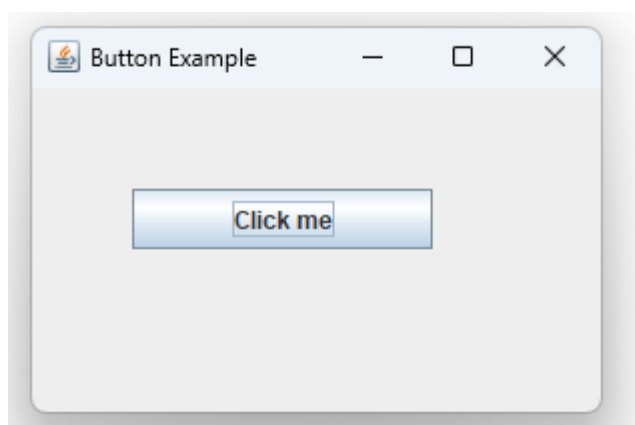


Disusun oleh :
Nama : Muhammad Rafly Alfarizi
Npm : 223040043
Kelas : A

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN
BANDUNG
2024

- Latihan 1

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

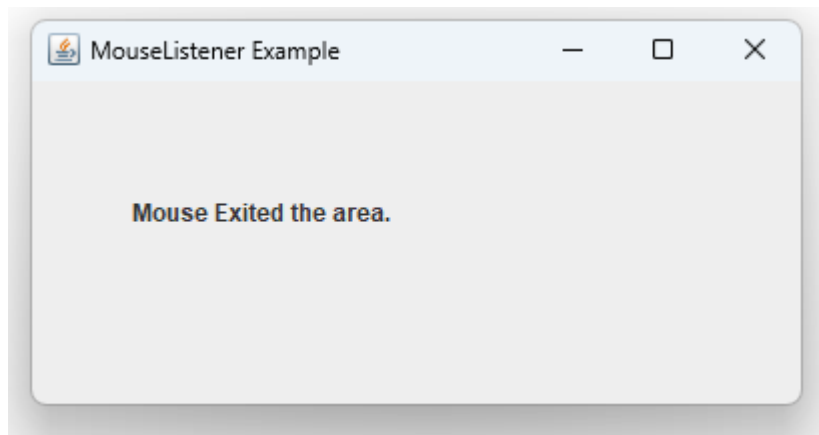


- JFrame: Membuat jendela utama untuk menampilkan komponen GUI. Ini adalah wadah tempat semua komponen, seperti tombol, ditempatkan. Pada kode ini, jendela diberi judul "Button Example".

- `JButton`: Membuat tombol interaktif dengan teks "Click me". Tombol ini dapat menerima aksi dari pengguna, seperti klik.
- `addActionListener()`: Menambahkan event listener ke tombol, yang akan merespons saat tombol diklik. Dalam hal ini, aksi yang dijalankan adalah mencetak teks "Button Clicked" ke konsol melalui metode `actionPerformed()`.

- Latihan 2

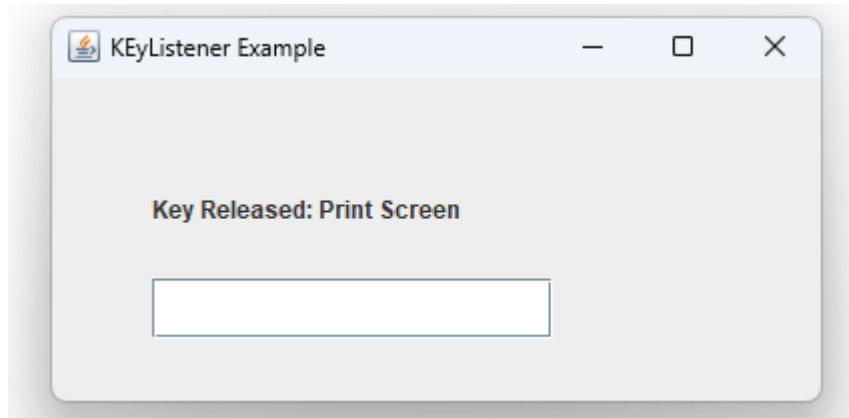
```
1  import javax.swing.*;
2  import java.awt.event.*;
3
4  public class MouseListenerExample {
5      public static void main (String[] args) {
6          // Membuat frame
7          JFrame frame = new JFrame("MouseListener Example");
8
9          // Membuat label untuk menampilkan pesan
10         JLabel label = new JLabel("Arahkan dan klik mouse pada area ini. ");
11         label.setBounds(50,50,300,30);
12
13         // Menambahkan MouseListener ke label
14         label.addMouseListener(new MouseListener() {
15             // Dijalankan Ketika mouse diklik (klik kiri, kanan atau tengah)
16             public void mouseClicked(MouseEvent e) {
17                 label.setText("Mouse Clicked at: (" + e.getX() + ", " + e.getY() + ")");
18             }
19
20             // Dijalankan ketika mouse ditekan (tanpa dilepaskan)
21             public void mousePressed(MouseEvent e) {
22                 label.setText("Mouse Pressed at: (" + e.getX() + ", " + e.getY() + ")");
23             }
24
25             // Dijalankan ketika mouse dileaskan setelah ditekan
26             public void mouseReleased(MouseEvent e) {
27                 label.setText("Mouse Released at: (" + e.getX() + ", " + e.getY() + ")");
28             }
29
30             // Dijalankan ketika mouse masuk area komponen
31             public void mouseEntered(MouseEvent e) {
32                 label.setText("Mouse Entered the area. ");
33             }
34
35             // Dijalankan ketika mouse keluar dari area komponen
36             public void mouseExited(MouseEvent e) {
37                 label.setText("Mouse Exited the area. ");
38             }
39         });
40
41         // menambahkan label ke frame
42         frame.add(label);
43
44         //setting frame
45         frame.setSize(400,200);
46         frame.setLayout(null);
47         frame.setVisible(true);
48         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
49     }
50 }
```



- JFrame: Membuat jendela utama dengan judul "MouseListener Example".
- JLabel: Label teks yang menampilkan pesan dan akan berubah ketika mouse berinteraksi dengan area label tersebut.
- addMouseListener(): Menambahkan MouseListener yang merespons berbagai aksi mouse:
- mouseClicked(): Dijalankan saat mouse diklik, menampilkan koordinat klik.
- mousePressed(): Menampilkan posisi ketika mouse ditekan.
- mouseReleased(): Menampilkan posisi ketika mouse dilepaskan setelah ditekan.
- mouseEntered(): Menampilkan pesan ketika mouse masuk ke area label.
- mouseExited(): Menampilkan pesan saat mouse keluar dari area label.

- Latihan 3

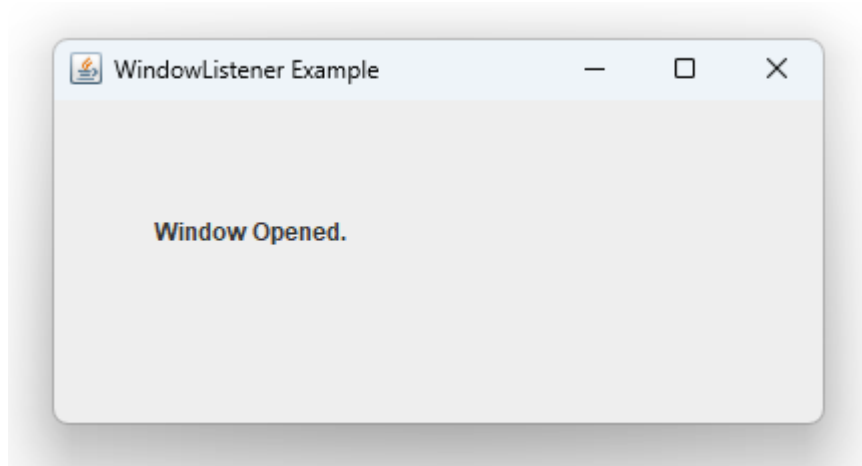
```
1 import javax.swing.*;
2 import java.awt.event.*;
3
4 public class KeyListenerExample {
5     public static void main (String[] args) {
6         //membuat frame
7         JFrame frame = new JFrame("KKeyListener Example");
8
9         //membuat label untuk menampilkan pesan
10        JLabel label = new JLabel("Tekan Tombol pada keyboard. ");
11        label.setBounds(50,50,300,30);
12
13        //membuat text field untuk okus keyboard
14        JTextField textField = new JTextField();
15        textField.setBounds(50,100,200,30);
16
17        // menambahkan keylistener ke txt field
18        textField.addKeyListener(new KeyListener(){
19            //dijalankan ketika tombol ditekan
20            public void keyPressed(KeyEvent e){
21                label.setText("Key pressed: " + KeyEvent.getKeyText(e.getKeyCode()));
22            }
23
24            //dijalankan ketika tombol dilepaskan
25            public void keyReleased(KeyEvent e){
26                label.setText("Key Released: " + KeyEvent.getKeyText(e.getKeyCode()));
27            }
28
29            //dijalankan ketika tombol ditekan dan diepaskan (sama dengan mengetik karalter)
30            public void keyTyped(KeyEvent e){
31                label.setText("Key Typed: " + e.getKeyChar());
32            }
33        });
34
35        // menambahkan komponen ke frame
36        frame.add(label);
37        frame.add(textField);
38
39        //setting frame
40        frame.setSize(400,200);
41        frame.setLayout(null);
42        frame.setVisible(true);
43        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
44    }
45 }
```



- **JFrame**: Membuat jendela utama dengan judul "KeyListener Example".
- **JLabel**: Label yang menampilkan pesan saat tombol pada keyboard ditekan, dilepaskan, atau diketik.
- **TextField**: Kotak teks untuk menangkap fokus keyboard, memungkinkan pengguna mengetik atau menekan tombol.
- **addKeyListener()**: Menambahkan KeyListener ke TextField untuk merespons interaksi keyboard:
- **keyPressed()**: Dijalankan saat tombol ditekan, menampilkan nama tombol yang ditekan.
- **keyReleased()**: Dijalankan saat tombol dilepaskan, menampilkan nama tombol yang dilepaskan.
- **keyTyped()**: Dijalankan saat tombol ditekan dan dilepaskan (setara dengan mengetik karakter), menampilkan karakter yang diketik.

- Latihan 4

```
1 import javax.swing.*;
2 import java.awt.event.*;
3
4 public class WindowListenerExample {
5     public static void main (String[] args ){
6         //membuat frame
7         JFrame frame = new JFrame("WindowListener Example");
8
9         //membuat label untuk menampilkan pesan
10        JLabel label = new JLabel("Lakukan operasi pada jendela");
11        label.setBounds(50,50,300,30);
12
13        //menambahkan windowlistener ke frame
14        frame.addWindowListener(new WindowListener() {
15            //dijalankan ketika jendela dibuka
16            public void windowOpened(WindowEvent e){
17                label.setText("Window Opened. ");
18            }
19
20            //dijalankan ketika jendela sedang dalam proses penutupan
21            public void windowClosing(WindowEvent e){
22                label.setText("Window Closing. ");
23                //bisa menambahkan system.exit(0); jika ingin menghentikan program
24            }
25
26            //dijalankan ketika jendela benar benar ditutup
27            public void windowClosed(WindowEvent e){
28                label.setText("Window Closed. ");
29            }
30
31            //dijalankan ketika jendela diminimalkan
32            public void windowIconified(WindowEvent e){
33                label.setText("Window Minimized. ");
34            }
35
36            //dijalankan ketika jendela dipulihkan dari minimal
37            public void windowDeiconified(WindowEvent e){
38                label.setText("Window Restored. ");
39            }
40
41            //dijalankan ketika jendela menjadi aktif (berfokus)
42            public void windowActivated(WindowEvent e){
43                label.setText("Window Activated. ");
44            }
45
46            //dijalankan ketika jendela dimaksimalkan
47            public void windowDeactivated(WindowEvent e){
48                label.setText("Window Deactivated. ");
49            }
50        });
51
52        // menambahkan label ke frame
53        frame.add(label);
54
55        //setting frame
56        frame.setSize(400,200);
57        frame.setLayout(null);
58        frame.setVisible(true);
59        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
60    }
61 }
```

- JFrame: Membuat jendela utama dengan judul "WindowListener Example".
- JLabel: Label yang menampilkan pesan berdasarkan operasi yang dilakukan pada jendela (misalnya, dibuka, ditutup, diminimalkan).
- addWindowListener(): Menambahkan WindowListener ke jendela, untuk merespons berbagai aksi jendela:
- windowOpened(): Dijalankan saat jendela pertama kali dibuka.
- windowClosing(): Dijalankan saat jendela sedang dalam proses ditutup.
- windowClosed(): Dijalankan setelah jendela ditutup sepenuhnya.
- windowIconified(): Dijalankan saat jendela diminimalkan.
- windowDeiconified(): Dijalankan saat jendela dipulihkan dari minimisasi.
- windowActivated(): Dijalankan saat jendela menjadi aktif dan berfokus.
- windowDeactivated(): Dijalankan saat jendela kehilangan fokus atau tidak aktif.