**PRATIKUM PEMROGRAMAN**

**PERTEMUAN KE - 4**



Disusun Oleh :

**Rizki Pratama (223040017)**

**Kelas B**

**Teknik Informatika**

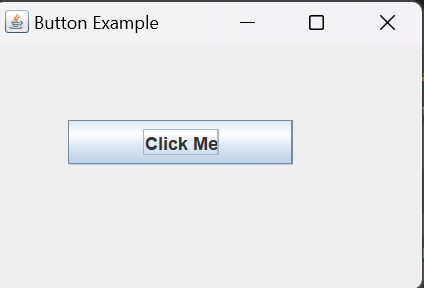
**Fakultas Teknik**

**Universitas Pasundan**

1. ButtonExample



Output



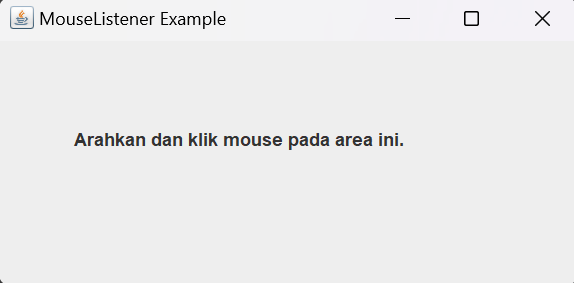
Penjelasan

1. **Pernyataan Impor**:
   * import java.awt.event.\*; dan import javax.swing.\*; digunakan untuk mengimpor kelas-kelas yang diperlukan untuk menangani event dan membuat antarmuka pengguna grafis (GUI).
2. **Deklarasi Kelas**:
   * public class ButtonExample { mendeklarasikan kelas publik bernama ButtonExample.
3. **Metode Main**:
   * public static void main(String[] args) { adalah metode utama di mana eksekusi program dimulai.
4. **Membuat JFrame**:
   * JFrame frame = new JFrame("Button Example"); membuat jendela baru dengan judul "Button Example".
5. **Membuat JButton**:
   * JButton button = new JButton("Click Me"); membuat tombol dengan teks "Click Me".
6. **Menambahkan ActionListener**:
   * button.addActionListener(new ActionListener(){ ... }); menambahkan pendengar aksi ke tombol untuk menangani event klik.
   * public void actionPerformed(ActionEvent e) { System.out.println("Button clicked"); } mendefinisikan aksi yang dilakukan ketika tombol diklik, yaitu mencetak "Button clicked" ke konsol.
7. **Menetapkan Posisi dan Ukuran Tombol**:
   * button.setBounds(50, 50, 150, 30); menetapkan posisi (x=50, y=50) dan ukuran (lebar=150, tinggi=30) tombol.
8. **Menambahkan Tombol ke Frame**:
   * frame.add(button); menambahkan tombol ke frame.
9. **Menetapkan Properti Frame**:
   * frame.setSize(300, 200); menetapkan ukuran frame menjadi 300x200 piksel.
   * frame.setLayout(null); menetapkan pengelola tata letak menjadi null, memungkinkan posisi absolut.
   * frame.setVisible(true); membuat frame terlihat.
   * frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); memastikan aplikasi keluar ketika frame ditutup.

2. MouseListener



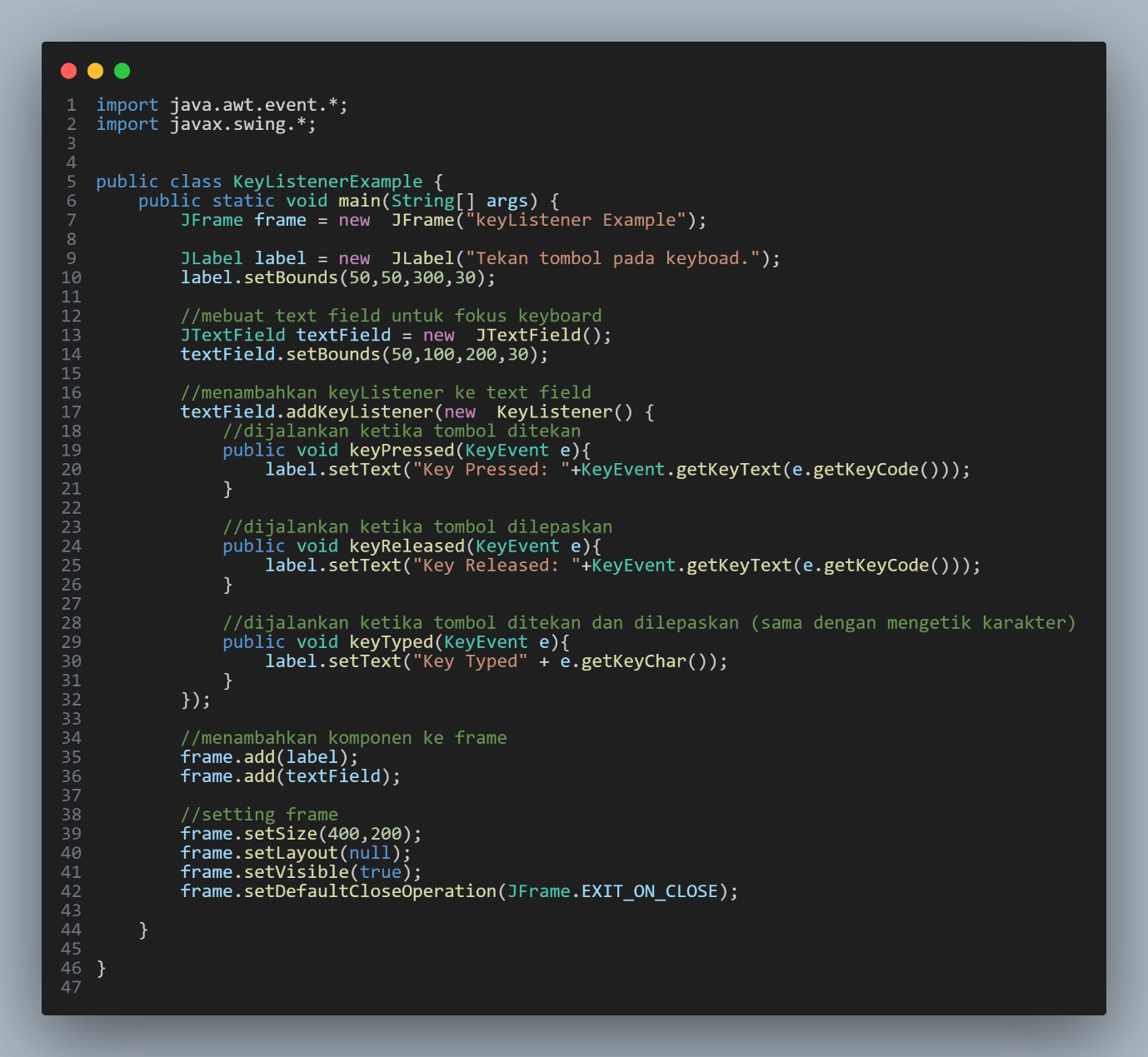
Output



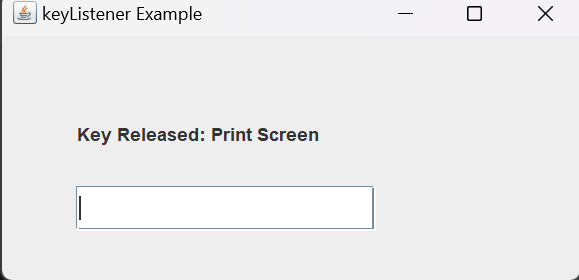
Penjelasan

1. **Import Statement**:
   * import java.awt.event.MouseEvent; dan import java.awt.event.MouseListener; digunakan untuk menangani event mouse.
   * import javax.swing.\*; digunakan untuk membuat komponen GUI seperti JFrame dan JLabel.
2. **Deklarasi Kelas**:
   * public class MouseListenerExample { mendeklarasikan kelas publik bernama MouseListenerExample.
3. **Metode Main**:
   * public static void main(String[] args) { adalah metode utama di mana program mulai berjalan.
4. **Membuat JFrame**:
   * JFrame frame = new JFrame("MouseListener Example"); membuat jendela dengan judul "MouseListener Example".
5. **Membuat JLabel**:
   * JLabel label = new JLabel("Arahkan dan klik mouse pada area ini."); membuat label dengan teks petunjuk.
6. **Menambahkan MouseListener**:
   * label.addMouseListener(new MouseListener() { ... }); menambahkan pendengar event mouse ke label.
   * public void mouseClicked(MouseEvent e) { ... } dijalankan ketika mouse diklik.
   * public void mousePressed(MouseEvent e) { ... } dijalankan ketika mouse ditekan.
   * public void mouseReleased(MouseEvent e) { ... } dijalankan ketika mouse dilepaskan.
   * public void mouseEntered(MouseEvent e) { ... } dijalankan ketika mouse masuk area komponen.
   * public void mouseExited(MouseEvent e) { ... } dijalankan ketika mouse keluar dari area komponen.
7. **Menetapkan Posisi dan Ukuran Label**:
   * label.setBounds(50, 50, 300, 30); menetapkan posisi dan ukuran label.
8. **Menambahkan Label ke Frame**:
   * frame.add(label); menambahkan label ke dalam frame.
9. **Menetapkan Properti Frame**:
   * frame.setSize(400, 200); menetapkan ukuran frame menjadi 400x200 piksel.
   * frame.setLayout(null); menetapkan layout manager ke null untuk posisi absolut.
   * frame.setVisible(true); membuat frame terlihat.
   * frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); memastikan aplikasi keluar saat frame ditutup.

3. KeyListenerExample



Output



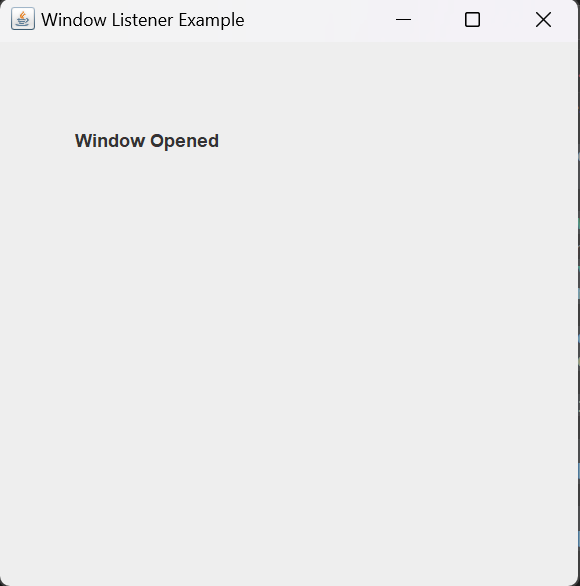
Penjelasan

1. **Pernyataan Impor**:
   * import java.awt.event.\*; dan import javax.swing.\*; digunakan untuk mengimpor kelas-kelas yang diperlukan untuk menangani event dan membuat antarmuka pengguna grafis (GUI).
2. **Deklarasi Kelas**:
   * public class KeyListenerExample { mendeklarasikan kelas publik bernama KeyListenerExample.
3. **Metode Main**:
   * public static void main(String[] args) { adalah metode utama di mana program mulai berjalan.
4. **Membuat JFrame**:
   * JFrame frame = new JFrame("keyListener Example"); membuat jendela dengan judul "keyListener Example".
5. **Membuat JLabel**:
   * JLabel label = new JLabel("Tekan tombol pada keyboard."); membuat label dengan teks petunjuk.
6. **Membuat JTextField**:
   * JTextField textField = new JTextField(); membuat bidang teks untuk fokus keyboard.
7. **Menambahkan KeyListener**:
   * textField.addKeyListener(new KeyListener() { ... }); menambahkan pendengar event keyboard ke bidang teks.
   * public void keyPressed(KeyEvent e) { ... } dijalankan ketika tombol ditekan.
   * public void keyReleased(KeyEvent e) { ... } dijalankan ketika tombol dilepaskan.
   * public void keyTyped(KeyEvent e) { ... } dijalankan ketika tombol ditekan dan dilepaskan.
8. **Menetapkan Posisi dan Ukuran Komponen**:
   * label.setBounds(50, 50, 300, 30); menetapkan posisi dan ukuran label.
   * textField.setBounds(50, 100, 200, 30); menetapkan posisi dan ukuran bidang teks.
9. **Menambahkan Komponen ke Frame**:
   * frame.add(label); menambahkan label ke dalam frame.
   * frame.add(textField); menambahkan bidang teks ke dalam frame.
10. **Menetapkan Properti Frame**:
    * frame.setSize(400, 200); menetapkan ukuran frame menjadi 400x200 piksel.
    * frame.setLayout(null); menetapkan layout manager ke null untuk posisi absolut.
    * frame.setVisible(true); membuat frame terlihat.
    * frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); memastikan aplikasi keluar saat frame ditutup.

4. WindowListenerExample



Output



Penjelasan

1. **Pernyataan Impor**:
   * import java.awt.event.\*; dan import javax.swing.\*; mengimpor kelas-kelas yang diperlukan untuk menangani event dan membuat antarmuka pengguna grafis (GUI).
2. **Deklarasi Kelas**:
   * public class WindowListenerExample { mendeklarasikan kelas publik bernama WindowListenerExample.
3. **Metode Main**:
   * public static void main(String[] args) { adalah metode utama di mana program mulai berjalan.
4. **Membuat JFrame**:
   * JFrame frame = new JFrame("Window Listener Example"); membuat jendela dengan judul "Window Listener Example".
5. **Membuat JLabel**:
   * JLabel label = new JLabel("lakukan operasi pada jendela"); membuat label dengan teks petunjuk.
6. **Menambahkan WindowListener**:
   * frame.addWindowListener(new WindowListener() { ... }); menambahkan pendengar event jendela ke frame.
   * public void windowOpened(WindowEvent e) { ... } dijalankan ketika jendela dibuka.
   * public void windowClosing(WindowEvent e) { ... } dijalankan ketika jendela dalam proses penutupan.
   * public void windowClosed(WindowEvent e) { ... } dijalankan ketika jendela benar-benar ditutup.
   * public void windowIconified(WindowEvent e) { ... } dijalankan ketika jendela diminimalkan.
   * public void windowDeiconified(WindowEvent e) { ... } dijalankan ketika jendela dipulihkan setelah diminimalkan.
   * public void windowActivated(WindowEvent e) { ... } dijalankan ketika jendela menjadi aktif (berfokus).
   * public void windowDeactivated(WindowEvent e) { ... } dijalankan ketika jendela kehilangan fokus.
7. **Menetapkan Posisi dan Ukuran Label**:
   * label.setBounds(50, 50, 300, 30); menetapkan posisi dan ukuran label.
8. **Menambahkan Label ke Frame**:
   * frame.add(label); menambahkan label ke dalam frame.
9. **Menetapkan Properti Frame**:
   * frame.setSize(400, 400); menetapkan ukuran frame menjadi 400x400 piksel.
   * frame.setLayout(null); menetapkan layout manager ke null untuk posisi absolut.
   * frame.setVisible(true); membuat frame terlihat.
   * frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); memastikan aplikasi keluar saat frame ditutup.