Laporan Praktikum

Praktik Pemrograman

**OLEH: Helga Arya Prayoga (24051130022)**

# MODUL 9

**TOPIK:**

EVENT HANDLING/ACTION PROGRAM



Table of Contents

[***Week #9 1***](#_gjdgxs)

[**A. Penjelasan Tugas Praktikum 3**](#_30j0zll)

[**B. Langkah-langkah dan Screenshot 3**](#_1fob9te)

[**C. Kendala yang Dialami 12**](#_3znysh7)

[**D. Kesimpulan 12**](#_2et92p0)

## Penjelasan Tugas Praktikum

* + - 1. Program ActionListener pada mouse
      2. Program ActionListener pada keyboard
      3. Penjelasan delegation event model dan macam – macam delegation event model
      4. Penjalasan event listener dan macam – macam event listener serta method – methodnya
      5. Program kalkulator sederhana
      6. Program percobaan event handling pada java
      7. Program kalkulator

## Langkah-langkah dan Screenshot

1. Program ActionListener pada mouse

Kode Program:

|  |
| --- |
| import java.awt.\*;  import java.awt.event.\*;  import javax.swing.\*;  public class ActionListenerMouse {  public static void main(String[] args) {  JFrame frame = new JFrame();  final JLabel lb = new JLabel("");  final String[] items = {" Roti ", " Madu "};  final JComboBox comboBox = new JComboBox(items);  JPanel pn = new JPanel();  frame.add(pn, BorderLayout.WEST);  pn.add(comboBox, BorderLayout.CENTER);  pn.add(lb, BorderLayout.NORTH);  comboBox.addActionListener(new ActionListener() {  public void actionPerformed (ActionEvent ae) {  String mn = (String) comboBox.getSelectedItem();  lb.setText(mn);  }  });  frame.setSize(200,200);  frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  frame.setVisible(true);  }  } |

*Screenshot:*

|  |
| --- |
|  |

1. Program ActionListener pada keyboard

Kode Program:

|  |
| --- |
| import javax.swing.\*;  import java.awt.\*;  import java.awt.event.\*;  class ActionListenerKeyboard extends JFrame {  JButton m = new JButton("TEKAN KEY 1,2,3,4");  JLabel tulisan = new JLabel ("UKURAN FRAME AKAN BERUBAH");    ActionListenerKeyboard() {  setTitle("INI EVENT PADA BUTTON");  setSize(400,200);  setLocation(200,100);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  }  void komponen() {  getContentPane().setLayout(new FlowLayout());  getContentPane().add(m);  getContentPane().add(tulisan);  setVisible(true);  }  void reaksi() {  m.addKeyListener(new KeyAdapter() {  public void keyPressed (KeyEvent e) {  if (e.getKeyCode() == e.VK\_1) {  setSize(300,500);  }  if(e.getKeyCode() == e.VK\_2) {  setSize(200,100);  }  if(e.getKeyCode() == e.VK\_3) {  setSize(100,10);  }  if(e.getKeyCode() == e.VK\_4) {  setSize(900,200);  }  }  });  }  public static void main (String[] args) {  ActionListenerKeyboard eb = new ActionListenerKeyboard();  eb.komponen();  eb.reaksi();  }  } |

*Screenshot:*

|  |
| --- |
|  |

1. Delegation event model menguraikan bagaimana program Anda dapat merespon interaksi dari user. Terdapat 3 macam delegation event model, yaitu :
2. Event Source

Objek yang memicu event. Contohnya adalah tombol (JButton), yang bisa memicu event ketika diklik oleh pengguna.

1. Event Listener (Pendengar Event)

Objek yang mendengarkan event dari source. Listener ini akan menangani aksi yang dihasilkan dari event tersebut.

1. Event Object (Objek Event)

Objek yang membawa informasi tentang event yang terjadi, seperti jenis event, sumber event, dan data tambahan lainnya. Contoh: ActionEvent, MouseEvent.

1. Event listeners adalah class yang mengimplementasikan interfaces <Type>Listener. Terdapat 3 macam event listeners pada java, yaitu :
2. Action Listener

Digunakan untuk menangani event seperti reaksi perubahan pada mouse atau keyboard

Method :

* void actionPerformed(ActionEvent e)

1. Mouse Listener

Digunakan untuk menangani event seperti reaksi pada pergerakan mouse

Method :

* void mouseClicked(MouseEvent e
* void mousePressed(MouseEvent e)
* void mouseReleased(MouseEvent e)
* void mouseEntered(MouseEvent e)
* void mouseExited(MouseEvent e)

1. Mouse Motion Listener

Digunakan untuk menangani event seperti drag and drop pada mouse

Method :

* void mouseDragged(MouseEvent e)
* void mouseMoved(MouseEvent e)

1. Program kalkulator sederhana

Kode Program:

|  |
| --- |
| import javax.swing.\*;  import java.awt.event.\*;  class KalkulatorSederhana extends JFrame{  JLabel angkaPertama = new JLabel ("Masukkan Angka Pertama");  JTextField textFieldAngkaPertama = new JTextField();  JLabel angkaKedua = new JLabel ("Masukkan Angka Kedua");  JTextField textFieldAngkaKedua = new JTextField();  JLabel hasil = new JLabel ("Hasil");  JTextField textFieldHasil = new JTextField();  JButton btnTambah = new JButton("+");  JButton btnKurang = new JButton("-");  JButton btnKali = new JButton("\*");  JButton btnBagi = new JButton("/");  KalkulatorSederhana() {  setTitle("Kalkulator Sederhana");  setSize(250,400);  setLocation(200,100);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  }  void komponen() {  getContentPane().setLayout(null);  angkaPertama.setBounds(40, 20, 200, 20);  textFieldAngkaPertama.setBounds(20, 45, 200, 25);  angkaKedua.setBounds(50, 80, 200, 20);  textFieldAngkaKedua.setBounds(20, 105, 200, 25);  hasil.setBounds(100, 140, 200, 20);  textFieldHasil.setBounds(20, 165, 200, 25);  textFieldHasil.setEditable(false);  btnTambah.setBounds(20, 200, 50, 40);  btnKurang.setBounds(95, 200, 50, 40);  btnKali.setBounds(170, 200, 50, 40);  btnBagi.setBounds(95, 265, 50, 40);  getContentPane().add(angkaPertama);  getContentPane().add(textFieldAngkaPertama);  getContentPane().add(angkaKedua);  getContentPane().add(textFieldAngkaKedua);  getContentPane().add(hasil);  getContentPane().add(textFieldHasil);  getContentPane().add(btnTambah);  getContentPane().add(btnKurang);  getContentPane().add(btnKali);  getContentPane().add(btnBagi);  setVisible(true);  }  void reaksi() {  textFieldAngkaPertama.addKeyListener(new KeyAdapter() {  @Override  public void keyTyped(KeyEvent e) {  char c = e.getKeyChar();  if (!Character.isDigit(c)) {  e.consume();  }  }  });  textFieldAngkaKedua.addKeyListener(new KeyAdapter() {  @Override  public void keyTyped(KeyEvent e) {  char c = e.getKeyChar();  if (!Character.isDigit(c)) {  e.consume();  }  }  });  btnTambah.addActionListener(new ActionListener() {  public void actionPerformed (ActionEvent ae) {  try {  double angkaPertama = Double.parseDouble(textFieldAngkaPertama.getText());  double angkaKedua = Double.parseDouble(textFieldAngkaKedua.getText());  double hasil = angkaPertama + angkaKedua;  textFieldHasil.setText(String.valueOf(hasil));  } catch (NumberFormatException ex) {  JOptionPane.showMessageDialog(null, "Masukkan angka yang valid!");  }  }  });  btnKurang.addActionListener(new ActionListener() {  public void actionPerformed (ActionEvent ae) {  try {  double angkaPertama = Double.parseDouble(textFieldAngkaPertama.getText());  double angkaKedua = Double.parseDouble(textFieldAngkaKedua.getText());  double hasil = angkaPertama - angkaKedua;  textFieldHasil.setText(String.valueOf(hasil));  } catch (NumberFormatException ex) {  JOptionPane.showMessageDialog(null, "Masukkan angka yang valid!");  }  }  });  btnKali.addActionListener(new ActionListener() {  public void actionPerformed (ActionEvent ae) {  try {  double angkaPertama = Double.parseDouble(textFieldAngkaPertama.getText());  double angkaKedua = Double.parseDouble(textFieldAngkaKedua.getText());  double hasil = angkaPertama \* angkaKedua;  textFieldHasil.setText(String.valueOf(hasil));  } catch (NumberFormatException ex) {  JOptionPane.showMessageDialog(null, "Masukkan angka yang valid!");  }  }  });  btnBagi.addActionListener(new ActionListener() {  public void actionPerformed (ActionEvent ae) {  try {  double angkaPertama = Double.parseDouble(textFieldAngkaPertama.getText());  double angkaKedua = Double.parseDouble(textFieldAngkaKedua.getText());  double hasil = angkaPertama / angkaKedua;  textFieldHasil.setText(String.valueOf(hasil));  } catch (NumberFormatException ex) {  JOptionPane.showMessageDialog(null, "Masukkan angka yang valid!");  }  }  });  }  public static void main (String[] args) {  KalkulatorSederhana eb = new KalkulatorSederhana();  eb.komponen();  eb.reaksi();  }  } |

*Screenshot:*

|  |
| --- |
|  |

1. Program percobaan event handling

Kode Program:

|  |
| --- |
| import javax.swing.\*;  import java.awt.\*;  import java.awt.event.\*;  public class CobaAksiReaksi extends JFrame {  JTextArea nama = new JTextArea(10,10);  JButton bt = new JButton("Copy");  JTextArea txnama = new JTextArea(10,10);    CobaAksiReaksi() {  super("Coba Event Handling");  setLocation(200,300);  setSize(400,300);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  setVisible(true);  }  void tampilan() {  getContentPane().add(nama);  nama.append("Nama kamu siapa");  getContentPane().add(bt);  getContentPane().add(txnama);  txnama.setBackground(Color.LIGHT\_GRAY);  getContentPane().setLayout(new FlowLayout());  setVisible(true);  }  void aksi\_reaksi () {  bt.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent e) {  txnama.append(nama.getSelectedText());  }  });  }  public static void main(String[] args) {  CobaAksiReaksi f = new CobaAksiReaksi();  f.tampilan();  f.aksi\_reaksi();  }  } |

*Screenshot:*

|  |
| --- |
|  |

1. Program kalkulator

Kode Program:

|  |
| --- |
| import java.awt.Color;  import javax.swing.\*;  class Kalkulator extends JFrame{  JTextField textField = new JTextField();  JButton btn1 = new JButton("1");  JButton btn2 = new JButton("2");  JButton btn3 = new JButton("3");  JButton btnTambah = new JButton("+");  JButton btn4 = new JButton("4");  JButton btn5 = new JButton("5");  JButton btn6 = new JButton("6");  JButton btnKurang = new JButton("-");  JButton btn7 = new JButton("7");  JButton btn8 = new JButton("8");  JButton btn9 = new JButton("9");  JButton btnKali = new JButton("\*");  JButton btn0 = new JButton("0");  JButton btnC = new JButton("C");  JButton btnSamaDengan = new JButton("=");  JButton btnBagi = new JButton("/");  String operator = "";  double num1 = 0;  double num2 = 0;  Kalkulator() {  setTitle("Kalkulator");  setSize(285,310);  setLocation(200,100);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  }  void komponen() {  getContentPane().setLayout(null);  textField.setBounds(20, 10, 231, 40);    btn1.setBounds(20, 60, 50, 40);  btn1.setBackground(Color.LIGHT\_GRAY);  btn1.setForeground(Color.WHITE);  btn2.setBounds(80, 60, 50, 40);  btn2.setBackground(Color.BLACK);  btn2.setForeground(Color.WHITE);  btn3.setBounds(140, 60, 50, 40);  btn3.setBackground(Color.DARK\_GRAY);  btn3.setForeground(Color.WHITE);  btnTambah.setBounds(200, 60, 50, 40);  btnTambah.setBackground(Color.WHITE);  btn4.setBounds(20, 110, 50, 40);  btn4.setBackground(Color.LIGHT\_GRAY);  btn4.setForeground(Color.WHITE);  btn5.setBounds(80, 110, 50, 40);  btn5.setBackground(Color.BLACK);  btn5.setForeground(Color.WHITE);  btn6.setBounds(140, 110, 50, 40);  btn6.setBackground(Color.DARK\_GRAY);  btn6.setForeground(Color.WHITE);  btnKurang.setBounds(200, 110, 50, 40);  btnKurang.setBackground(Color.WHITE);  btn7.setBounds(20, 160, 50, 40);  btn7.setBackground(Color.LIGHT\_GRAY);  btn7.setForeground(Color.WHITE);  btn8.setBounds(80, 160, 50, 40);  btn8.setBackground(Color.BLACK);  btn8.setForeground(Color.WHITE);  btn9.setBounds(140, 160, 50, 40);  btn9.setBackground(Color.DARK\_GRAY);  btn9.setForeground(Color.WHITE);  btnKali.setBounds(200, 160, 50, 40);  btnKali.setBackground(Color.WHITE);  btn0.setBounds(20, 210, 50, 40);  btn0.setBackground(Color.WHITE);  btnC.setBounds(80, 210, 50, 40);  btnC.setBackground(Color.WHITE);  btnSamaDengan.setBounds(140, 210, 50, 40);  btnSamaDengan.setBackground(Color.WHITE);  btnBagi.setBounds(200, 210, 50, 40);  btnBagi.setBackground(Color.WHITE);  getContentPane().add(textField);  getContentPane().add(btn1);  getContentPane().add(btn2);  getContentPane().add(btn3);  getContentPane().add(btnTambah);  getContentPane().add(btn4);  getContentPane().add(btn5);  getContentPane().add(btn6);  getContentPane().add(btnKurang);  getContentPane().add(btn7);  getContentPane().add(btn8);  getContentPane().add(btn9);  getContentPane().add(btnKali);  getContentPane().add(btn0);  getContentPane().add(btnC);  getContentPane().add(btnSamaDengan);  getContentPane().add(btnBagi);  setVisible(true);  }  void reaksi() {  btn1.addActionListener(e -> textField.setText(textField.getText() + "1"));  btn2.addActionListener(e -> textField.setText(textField.getText() + "2"));  btn3.addActionListener(e -> textField.setText(textField.getText() + "3"));  btn4.addActionListener(e -> textField.setText(textField.getText() + "4"));  btn5.addActionListener(e -> textField.setText(textField.getText() + "5"));  btn6.addActionListener(e -> textField.setText(textField.getText() + "6"));  btn7.addActionListener(e -> textField.setText(textField.getText() + "7"));  btn8.addActionListener(e -> textField.setText(textField.getText() + "8"));  btn9.addActionListener(e -> textField.setText(textField.getText() + "9"));  btn0.addActionListener(e -> textField.setText(textField.getText() + "0"));  btnTambah.addActionListener(e -> setOperator("+"));  btnKurang.addActionListener(e -> setOperator("-"));  btnKali.addActionListener(e -> setOperator("\*"));  btnBagi.addActionListener(e -> setOperator("/"));  btnC.addActionListener(e -> textField.setText("")); // Menghapus isi text field  btnSamaDengan.addActionListener(e -> calculateResult());  }  private void setOperator(String op) {  operator = op;  num1 = Double.parseDouble(textField.getText());  textField.setText("");  }  private void calculateResult() {  num2 = Double.parseDouble(textField.getText());  double result = 0;  switch (operator) {  case "+":  result = num1 + num2;  break;  case "-":  result = num1 - num2;  break;  case "\*":  result = num1 \* num2;  break;  case "/":  result = num1 / num2;  break;  }  textField.setText(String.valueOf(result));  }  public static void main(String[] args) {  Kalkulator kk = new Kalkulator();  kk.komponen();  kk.reaksi();  }  } |

*Screenshot:*

|  |
| --- |
|  |
|  |

## Kendala yang Dialami

Tidak ada kendala dalam membuat program

## Kesimpulan

Delegation event model menguraikan bagaimana program Anda dapat merespon interaksi dari user. Terdapat 3 macam delegation event model, yaitu Event Source, Event Listener, dan Event Object. Event listeners adalah class yang mengimplementasikan interfaces <Type>Listener. Terdapat 3 macam event listeners pada java, yaitu Action Listener, Mouse Listener, Mouse Motion Listener.