

Materi 6 Pemrograman Visual

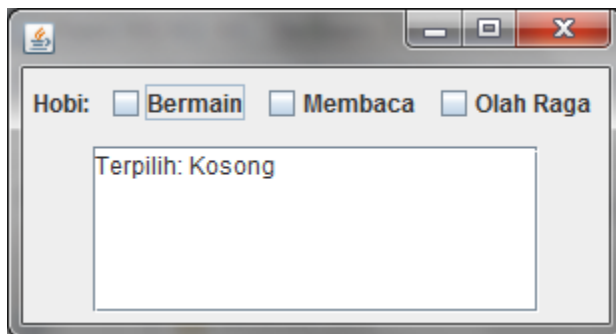
Pada materi ini akan dipelajari tentang pemberian aksi pada komponen swing (listener), terutama ItemListener.

Sesi Materi

Interface Listener memiliki berbagai macam tipe, salah satunya adalah ItemListener. Interface ItemListener dapat dikenai oleh beberapa komponen swing, terutama JCheckBox dan JRadioButton. Cara menggunakan ItemListener hampir sama dengan ActionListener. Langkah pertama, implements interface “ItemListener”, lalu override method “itemStateChanged(ItemEvent ie)”. Tambahkan listener kepada masing-masing komponen yang memerlukan dengan memanggil method “addItemListener” melalui objek komponen swing terkait.

Studi Kasus

Buatlah sebuah GUI seperti contoh di samping.



Setiap terdapat perubahan state, maka pada JTextArea akan ditampilkan berbagai check box yang terpilih. Jika tidak ada yang terpilih, maka akan ditampilkan “Terpilih: Kosong” di layar. GUI di samping memiliki 2 panel (JPanel) atas dan bawah. Panel atas berisikan komponen JLabel, dan 3 JCheckBox. Panel ini menempati bagian NORTH dari JFrame. Panel kedua yaitu panel bawah. Panel ini memiliki satu komponen JTextArea. Penempatan panel bawah pada JFrame yaitu pada posisi CENTER. Pada saat pertama kali dijalankan, JTextArea akan menampilkan “Terpilih: Kosong”.

Langkah pertama untuk membuat GUI, yaitu menentukan komponen swing yang digunakan serta jumlah objeknya.

1. JFrame
2. JPanel (2)
3. JLabel (1)
4. JCheckBox (3)

5. JTextArea (1) & JScrollPane (1)

Perhatikan bahwa JFrame dan JPanel menggunakan default layout masing-masing, sehingga tidak diperlukan pembuatan objek untuk masing-masing layout yang digunakan. Selain itu, JScrollPane dibutuhkan sebagai tempat meletakkan JTextArea. Hal ini seperti JTable dengan JScrollPane. Sehingga, dapat dituliskan kode sebagai berikut:

```
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class NewClass extends JFrame{
    JPanel panelAtas, panelBawah;
    JLabel labelHobi;
    JCheckBox cekMain, cekBaca, cekOlahRaga;
    JScrollPane panelScroll;
    JTextArea teks;

    public void aturGui(){

    }
}
```

Pada method aturGui(), bentuk objek masing-masing komponen, dan hubungkan komponen tersebut seperti konfigurasi sebelumnya (panel atas menyimpan label dan check box, sedangkan panel bagian bawah menyimpan text area).

```
panelAtas = new JPanel();
panelBawah = new JPanel();
labelHobi = new JLabel("Hobi: ");
cekMain = new JCheckBox("Bermain");
cekBaca = new JCheckBox("Membaca");
cekOlahRaga = new JCheckBox("Olah Raga");
panelScroll = new JScrollPane();
//membentuk objek JTextArea dengan ukuran panjang x = 25, panjang y = 5
teks = new JTextArea(5,25);

panelAtas.add(labelHobi);
panelAtas.add(cekMain);
panelAtas.add(cekBaca);
panelAtas.add(cekOlahRaga);

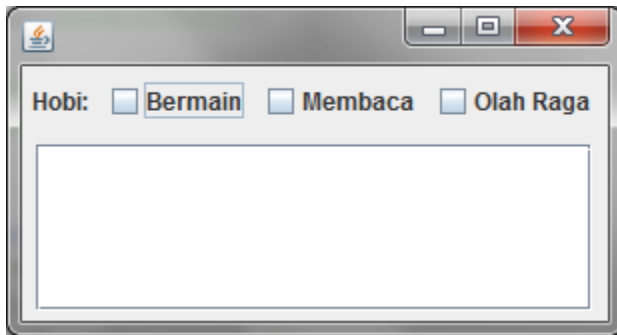
panelScroll.setViewportView(teks);
panelBawah.add(panelScroll);

this.add(panelAtas, BorderLayout.NORTH);
this.add(panelBawah, BorderLayout.SOUTH);
```

Tambahkan `public static void main(String args[])`, dan tampilkan objek dari `JFrame`.

```
public static void main(String[] args) {
    NewClass fr = new NewClass();
    fr.aturGui();
    fr.setDefaultCloseOperation(3);
    fr.pack();
    fr.setLocationRelativeTo(null);
    fr.setVisible(true);
}
```

Hasil GUI:



Tambahkan “Terpilih: Kosong” di objek `JTextArea` pada method `aturGui()`. Hal ini membuat GUI seperti pada requirement awal.

```
teks.setText("Terpilih: Kosong\n");
```

Setelah memastikan GUI telah tampil seperti yang seharusnya, tambahkan kode untuk meng-implements interface `ItemListener`, lalu override method `itemStateChanged`.

```
import java.awt.BorderLayout;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class NewClass extends JFrame implements ItemListener{

    JPanel panelAtas, panelBawah;
    JLabel labelHobi;
    JCheckBox cekMain, cekBaca, cekOlahRaga;
    JScrollPane panelScroll;
    JTextArea teks;

    public void aturGui() {
```

```

    /*atur kemunculan GUI*/
}

public static void main(String[] args) {
    NewClass fr = new NewClass();
    fr.aturGui();
    fr.setDefaultCloseOperation(3);
    fr.pack();
    fr.setLocationRelativeTo(null);
    fr.setVisible(true);
}

@Override
public void itemStateChanged(ItemEvent ie) {

}
}

```

Berikan kode untuk menambahkan listener kepada masing-masing komponen terkait (JCheckBox) dengan menggunakan method addItemListener.

```

public void aturGui() {
    panelAtas = new JPanel();
    panelBawah = new JPanel();
    labelHobi = new JLabel("Hobi: ");
    cekMain = new JCheckBox("Bermain");
    cekBaca = new JCheckBox("Membaca");
    cekOlahRaga = new JCheckBox("Olah Raga");
    panelScroll = new JScrollPane();
    teks = new JTextArea(5,25);

    cekMain.addItemListener(this);
    cekBaca.addItemListener(this);
    cekOlahRaga.addItemListener(this);

    panelAtas.add(labelHobi);
    panelAtas.add(cekMain);
    panelAtas.add(cekBaca);
    panelAtas.add(cekOlahRaga);

    panelScroll.setViewportViewView(teks);
    panelBawah.add(panelScroll);

    this.add(panelAtas, BorderLayout.NORTH);
    this.add(panelBawah, BorderLayout.SOUTH);

    teks.setText("Terpilih: Kosong\n");
}

```

Keyword this pada method addItemListener, memiliki arti yang sama dengan penggunaan method addActionListener. Setelahnya, dapat dibentuk aksi dari masing-masing JCheckBox.

```

@Override
public void itemStateChanged(ItemEvent ie) {
    JCheckBox cek = (JCheckBox) ie.getItem();
    StringBuilder sb = new StringBuilder();
}

```

```

        if(cek.equals(cekMain)){
            sb.append("Bermain");
        }else if(cek.equals(cekBaca)){
            sb.append("Membaca");
        }else if(cek.equals(cekOlahRaga)){
            sb.append("Olah Raga");
        }

        if(cek.isSelected()){
            sb.append(" Terpilih\n");
        }else{
            sb.append(" Tidak Terpilih\n");
        }
        teks.append(String.valueOf(sb));
    }
}

```

Kode Lengkap:

```

import java.awt.BorderLayout;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;

public class NewClass extends JFrame implements ItemListener{

    JPanel panelAtas, panelBawah;
    JLabel labelHobi;
    JCheckBox cekMain, cekBaca, cekOlahRaga;
    JScrollPane panelScroll;
    JTextArea teks;

    public void aturGui() {
        panelAtas = new JPanel();
        panelBawah = new JPanel();
        labelHobi = new JLabel("Hobi: ");
        cekMain = new JCheckBox("Bermain");
        cekBaca = new JCheckBox("Membaca");
        cekOlahRaga = new JCheckBox("Olah Raga");
        panelScroll = new JScrollPane();
        teks = new JTextArea(5,25);

        cekMain.addItemListener(this);
        cekBaca.addItemListener(this);
        cekOlahRaga.addItemListener(this);

        panelAtas.add(labelHobi);
        panelAtas.add(cekMain);
        panelAtas.add(cekBaca);
        panelAtas.add(cekOlahRaga);

        panelScroll.setViewportViewView(teks);
        panelBawah.add(panelScroll);
    }
}

```

```
        this.add(panelAtas, BorderLayout.NORTH);
        this.add(panelBawah, BorderLayout.SOUTH);

        teks.setText("Terpilih: Kosong\n");
    }

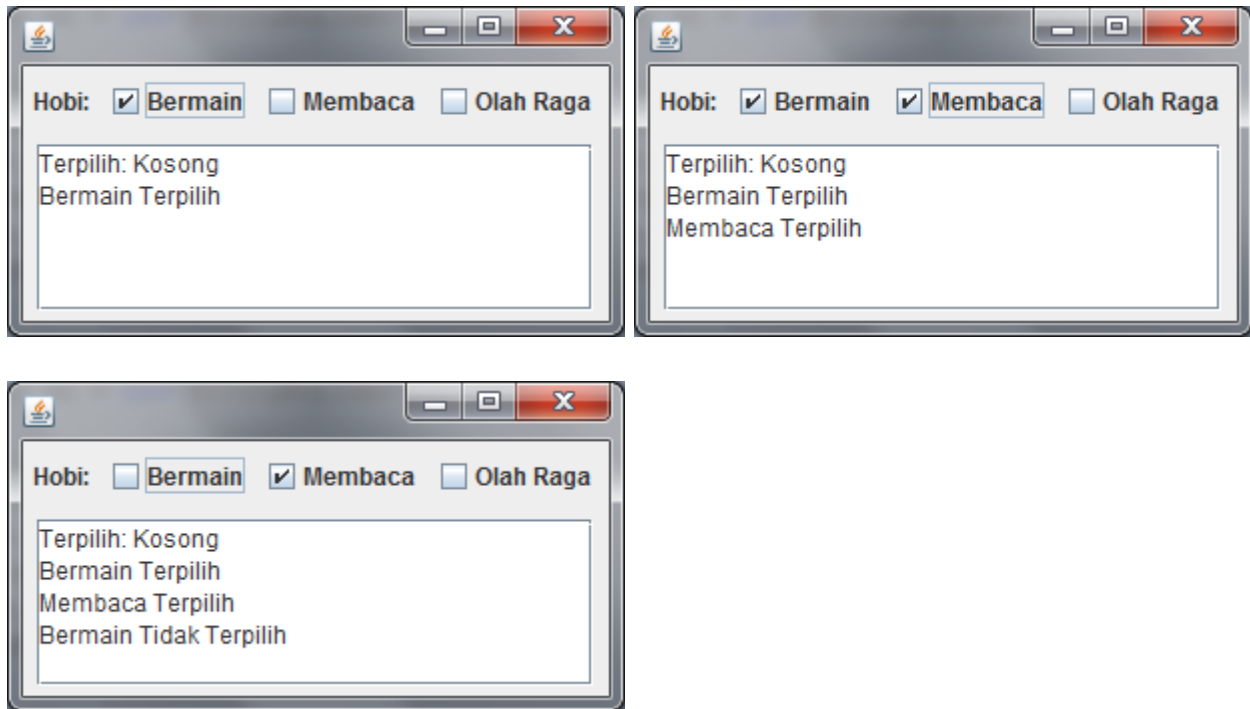
    public static void main(String[] args) {
        NewClass fr = new NewClass();
        fr.aturGui();
        fr.setDefaultCloseOperation(3);
        fr.pack();
        fr.setLocationRelativeTo(null);
        fr.setVisible(true);
    }

    @Override
    public void itemStateChanged(ItemEvent ie) {
        JCheckBox cek = (JCheckBox) ie.getItem();
        StringBuilder sb = new StringBuilder();

        if(cek.equals(cekMain)){
            sb.append("Bermain");
        }else if(cek.equals(cekBaca)){
            sb.append("Membaca");
        }else if(cek.equals(cekOlahRaga)){
            sb.append("Olah Raga");
        }

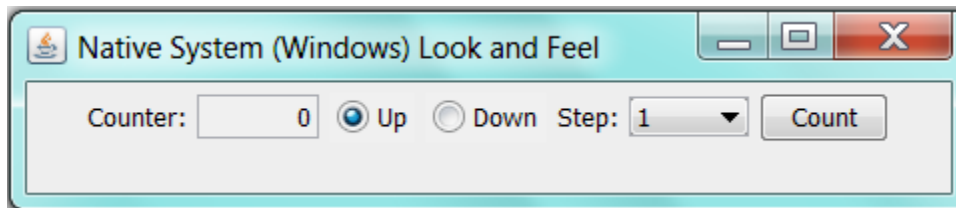
        if(cek.isSelected()){
            sb.append(" Terpilih\n");
        }else{
            sb.append(" Tidak Terpilih\n");
        }
        teks.append(String.valueOf(sb));
    }
}
```

Hasil Akhir:



Try-it Yourself

Duplikasi GUI berikut:



Ketika tombol “Count” ditekan, maka counter akan menampilkan angka sesuai radio button juga combo box terpilih. Gunakan ItemListener dan ActionListener sebagai interface yang di-implements.