

MODUL PBO

“GUI (GRAPHICAL USER INTERFACE)”



Disusun Oleh
NOVI DYAH PUSPITASARI

MODUL PBO

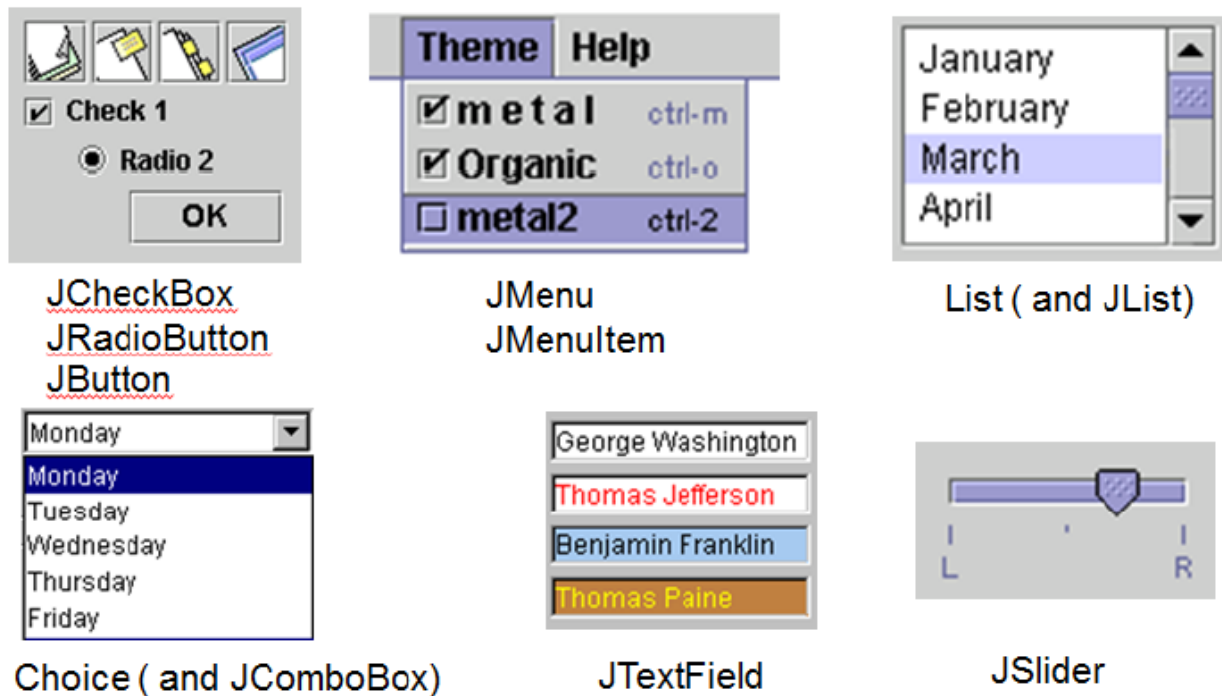
“GUI (GRAPHICAL USER INTERFACE)”

A. TUJUAN

- Memahami dan menerapkan pemrograman java GUI dengan menggunakan Netbeans

B. DASAR TEORI

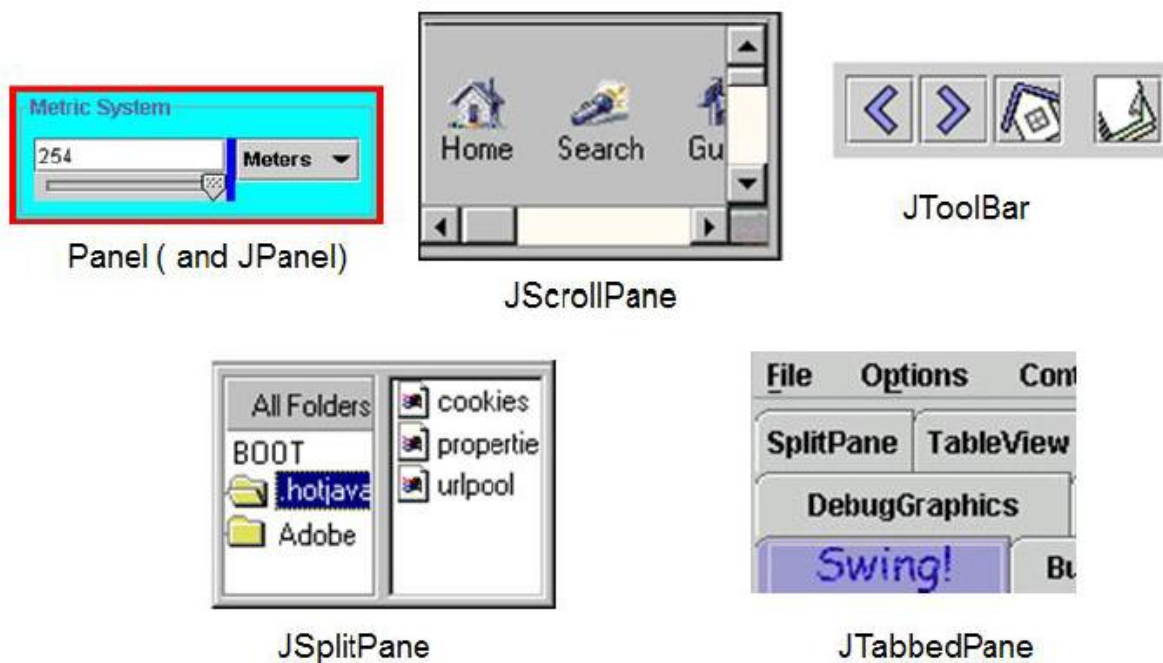
Komponen Dasar Swing



Gambar 1. Contoh JComponent

Secara umum terdapat 5 bagian swing yang akan sering digunakan yaitu:

1. **Top-level Container**, merupakan container dasar dimana komponen lainnya akan diletakkan. Contoh container seperti ini, yaitu Frame, Dialog, dan Applet yang diimplementasikan berupa class JFrame, JDialog, dan JApplet.
2. **Intermediate Container**, merupakan container perantara dimana komponen lainnya akan diletakkan, salah satu contohnya pane yang diimplementasikan berupa class JPanel.



Gambar 2. Contoh Container

3. **Atomic Component**, merupakan komponen yang memiliki fungsi yang spesifik, dimana umumnya user langsung berinteraksi dengan komponen jenis ini. Contoh JButton, JLabel, JTextField, dan JTextArea
4. **Layout Manager**, berfungsi untuk mengatur bagaimana tata letak atau posisi komponen yang akan diletakkan, satu sama lain di dalam suatu container. Contoh BorderLayout, BoxLayout, FlowLayout, GridBagLayout, dan GridLayout.
5. **Event Handling**, untuk menangani event yang dilakukan oleh user seperti menekan tombol, memperbesar atau memperkecil ukuran frame dan lain-lain

Event

Yang dimaksud dengan **event** adalah:

- Ketika user melakukan aksi terhadap sebuah user interface (misalnya meng-klik mouse atau menekan sebuah tombol), maka tindakan ini akan memunculkan sebuah **event**.
- **Event** adalah OBJEK yang mendeskripsikan sebuah kejadian (peristiwa yang terjadi)
- **Event Source** adalah pembangkit sebuah event, misalnya mouse click pada sebuah button akan membangkitkan sebuah ActionEvent dgn button sbg Event Source-nya.

- **Event Handler** adalah sebuah method yang menerima sebuah objek event, menterjemahkan, dan kemudian memproses interaksi user.

Event Listener

- Swing handle event dengan sekumpulan interface yang disebut dengan **Event Listeners**.
- Setiap kategori event, terdapat sebuah interface listener yang bersesuaian.
- Listener tersebut harus diimplementasikan oleh class dari objek yang akan menerima event tersebut.
- Listener ini akan menetapkan method mana yang harus didefinisikan dalam sebuah class yang sesuai untuk menerima tipe event tersebut.
- Method-method ini akan dipanggil ketika event yang bersangkutan terjadi.

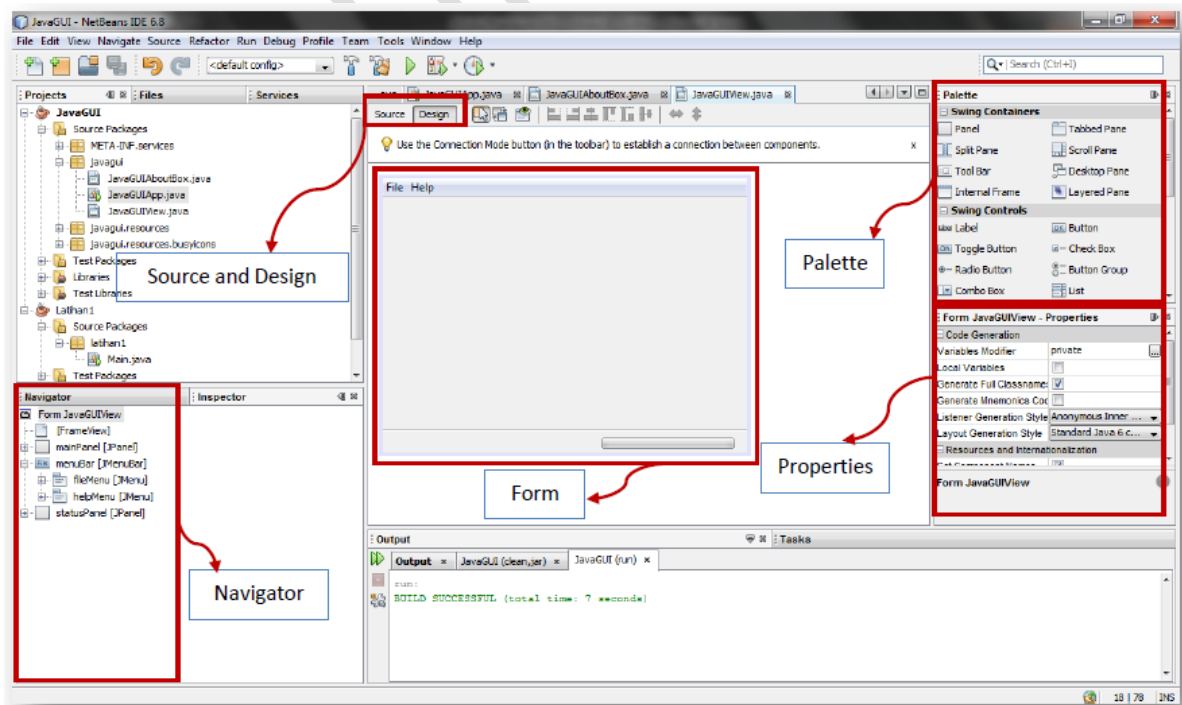
Kategori Event, Interface dan Methodnya

| Kategori | Nama Interface | Method |
|-----------------|-----------------------|--|
| Action | ActionListener | actionPerformed(ActionEvent) |
| Item | ItemListener | itemStateChanged(ItemEvent) |
| Mouse | MouseListener | mousePressed(MouseEvent) mouseReleased (MouseEvent) mouseEntered (MouseEvent) mouseExited (MouseEvent) mouseClicked (MouseEvent) |
| Mouse Motion | MouseMotionListener | mouseDragged (MouseEvent) mouseMoved (MouseEvent) |
| Key | KeyListener | keyPressed(KeyEvent) keyReleased (KeyEvent) keyTyped (KeyEvent) |
| Focus | FocusListener | focusGained(FocusEvent) focusLost (FocusEvent) |

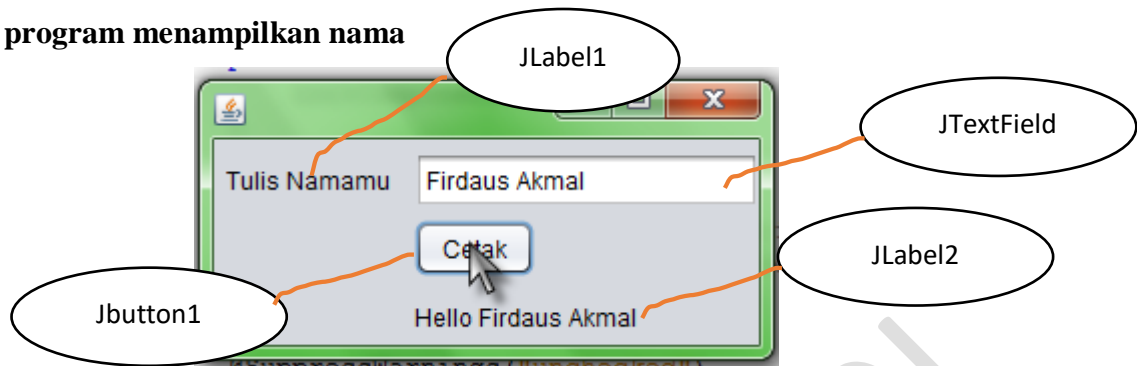
| Kategori | Nama Interface | Method |
|------------|--------------------|--|
| Adjustment | AdjustmentListener | adjustmentValueChanged(AdjustmentEvent) |
| Component | ComponentListener | componentMoved(ComponentEvent) componentHidden(ComponentEvent) componentResized (ComponentEvent) componentShown (ComponentEvent) |
| Window | WindowListener | windowClosing(WindowEvent) windowOpened (WindowEvent) windowIconified (WindowEvent) windowDeiconified (WindowEvent) windowClosed (WindowEvent) windowActivated (WindowEvent) windowDeactivated (WindowEvent) |
| Container | ContainerListener | componentAdded(ContainerEvent) componentRemoved (ContainerEvent) |
| Text | TextListener | textValueChanged(TextEvent) |

Java Desktop Application

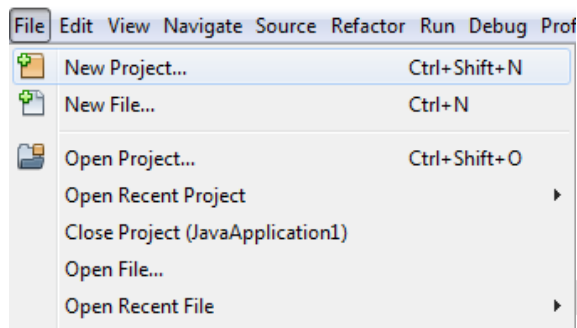
Java Desktop Application atau bisa juga disebut sebagai pemrograman berbasis GUI merupakan pemrograman yang menggunakan library Swing ataupun AWT. Pada pemrograman berbasis GUI, kita bisa membuat suatu aplikasi yang berbasis pada form yang sering kita jumpai saat menjalankan suatu aplikasi, software dan lainnya.



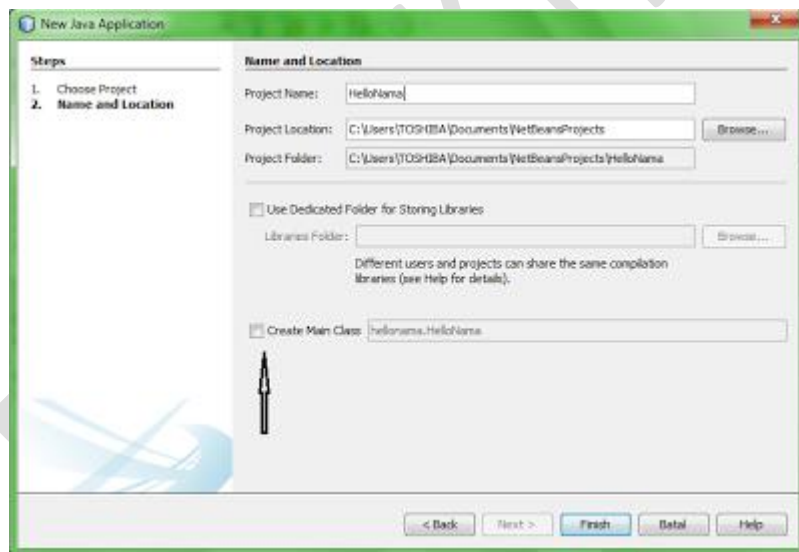
Membuat program menampilkan nama



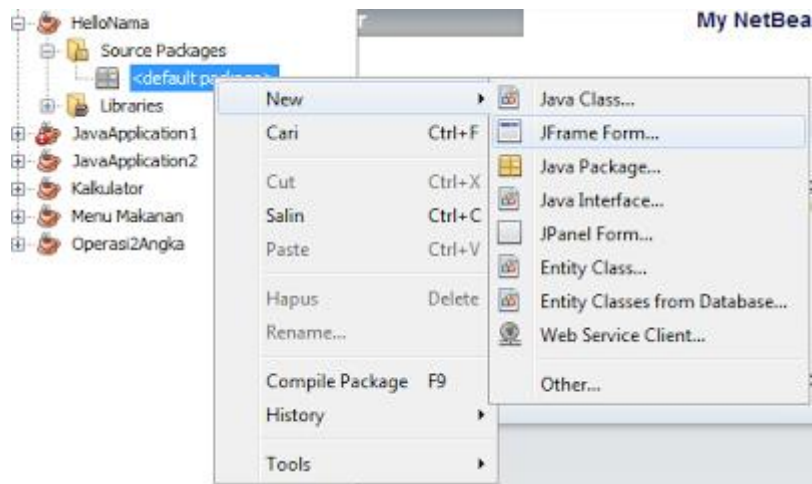
1. Masuk Netbeans dan buat project baru



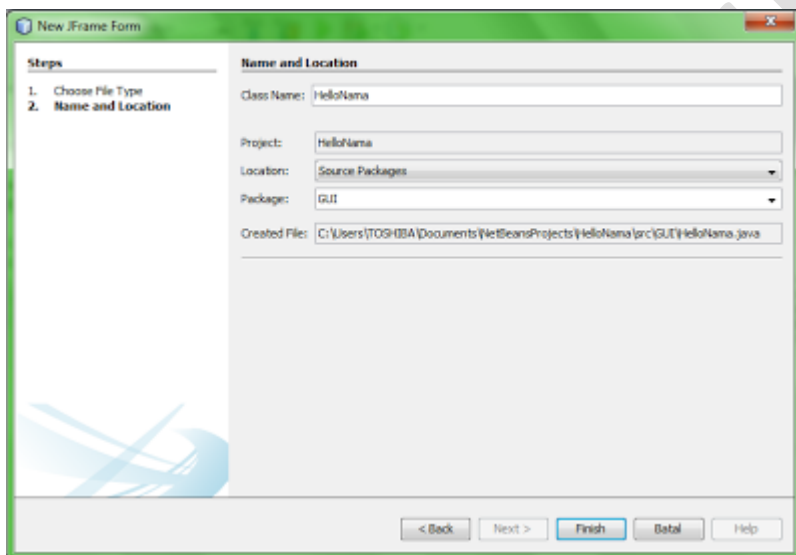
2. Pilih kategori Java dan project Java Application
3. Kemudian beri nama HelloNama. dan hapus centang pada Create Main Class



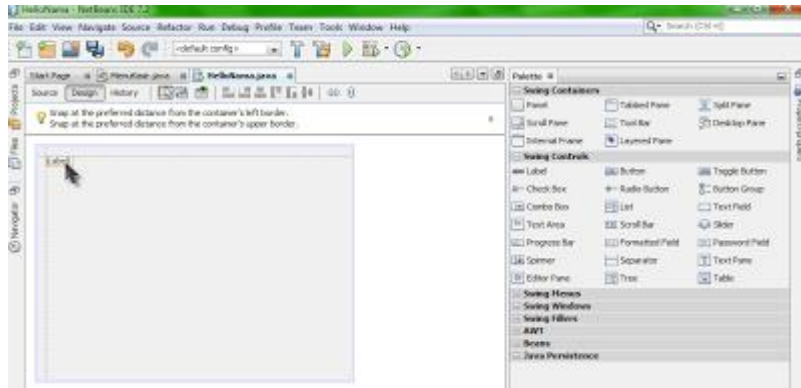
4. Pada default package klik kanan → new → JFrame From.



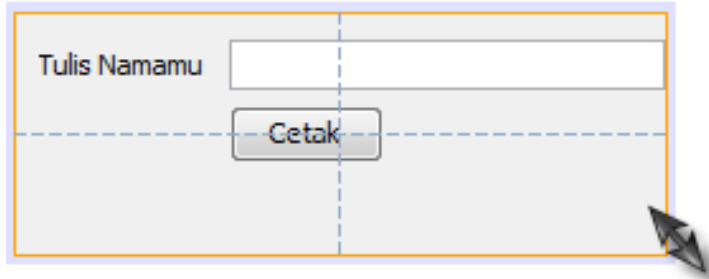
5. Kita beri nama **HelloNama** dan package nya GUI



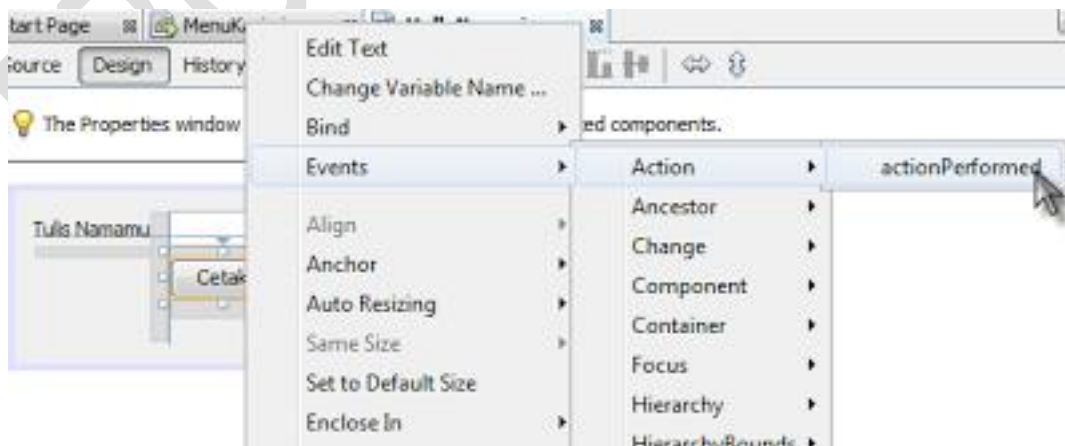
6. Tempatkan komponen GUI ke design dengan cara mengklik komponen kemudian klik pada tempat design. Komponen GUI kali ini adalah **Text Field** digunakan untuk menginput nama, kemudian **Label** digunakan untuk menampilkan sebuah kata atau kalimat, dan **Button** digunakan untuk menampilkan hasil dengan mengklik Button tersebut.



7. Ubah **text** dari komponen tersebut. **klik kanan** → **edit text**.
 - a. JTextField1 : 'dikosongkan'
 - b. JLabel1 : Tulis Namamu
 - c. JLabel2 : 'dikosongkan'
 - d. JButton1 : Cetak
8. Kemudian setiap komponen GUI tersebut kita ubah nama **variabelnya**. klik kanan pada komponen atau juga bisa pada **navigator** → **change variabel name**.
 - a. JTextField1 : jtinput
 - b. JLabel1 : jnama
 - c. JLabel2 : jhasil
 - d. JButton1 : jbcetak
9. Rapiakan frame dengan menggeser design



10. Buat even pada **jbcetak**. klik kanan pada jbcetak → Event → Action → ActionPerformed

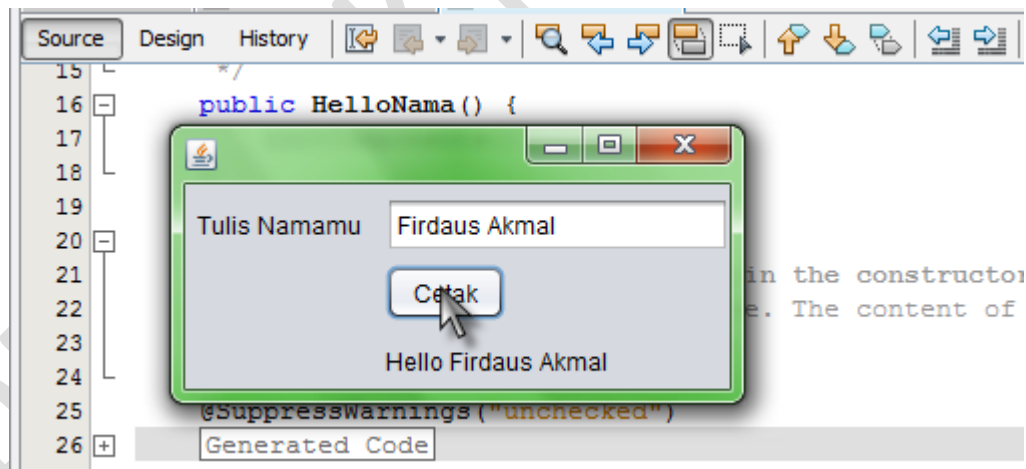


11. Buat code untuk *Event handling* di *jbcetak*

```
24  */
25  @SuppressWarnings("unchecked")
26  Generated Code
76
77  private void jtcetakActionPerformed(java.awt.event.ActionEvent evt) {
78
79      jthasil.setText("Hello " + jtinput.getText());
80  }
81
82  /**
83   * @param args the command line arguments
84   */
85  public static void main(String args[]) {
86      /* Set the Nimbus look and feel */
87      Look and feel setting code (optional)
108
109      /* Create and display the form */
```

Maksud dari *jhasil.setText()*; adalah komponen *jhasil* akan mengeluarkan kata atau kalimat dari dalam kurung berupa text. Kemudian maksud *jtinput.getText()* ialah komponen *jtinput* menerima data dari keyboard berupa text. Jadi, Maksud dari *jhasil.setText("Hello " + jtinput.getText());* ialah *jhasil* akan menghasilkan keluaran berupa *Hello* dan inputan dari *jtinput* yaitu nama tadi.

12. Kemudian jalankan aplikasi → **Shift+F6**. Masukkan nama kalian pada text field kemudian klik button cetak dan lihat hasilnya.



C. LATIHAN

1. Latihan 1: aplikasi sederhana berbasis GUI

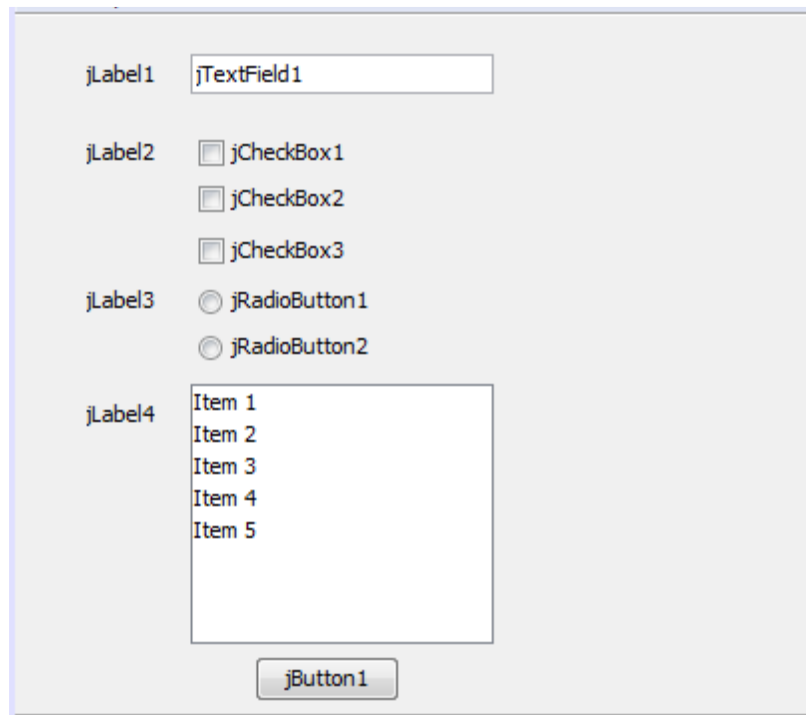
- a. Tambahkan beberapa komponen swing ke dalam **form** dengan cara **di-drag** ke dalam form.

Komponen-komponen swing pada java terletak di bagian **palette**. Komponen:

- | | |
|------------------------|--------------------------|
| 1) Label (4 buah) | 3) Check Box (3 buah) |
| 2) Text Field (1 buah) | 4) Radio Button (2 buah) |

5) List (1 buah)

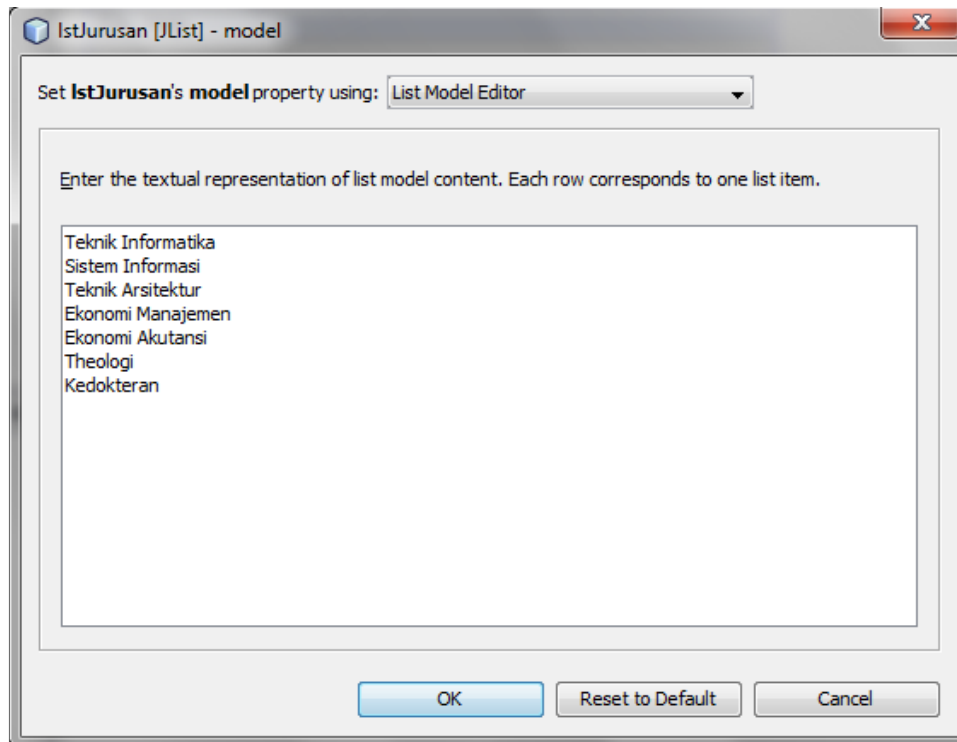
6) Button (1 buah)



b. Ubalah **property text** dan name dari masing-masing komponen form menjadi:

| Komponen | Name | Text |
|--------------|-------------|----------------|
| CheckBox1 | cbHob1 | Makan |
| CheckBox2 | cbHob2 | Shopping |
| CheckBox3 | cbHob3 | Tidur |
| Button 1 | btnSubmit | Submit! |
| TextBox1 | txtNama | (kosong) |
| RadioButton1 | rdLaki | Laki-Laki |
| RadioButton2 | rdPerempuan | Perempuan |
| ListBox1 | lstJurusan | (tidak diubah) |

c. Setelah itu, tekan tombol “...” pada properti **Model** milik **lstJurusan**, akan muncul sebuah jendela dan isilah dengan:



- d. Ubahlah juga semua property text pada masing-masing Label sehingga menjadi seperti gambar di bawah ini.

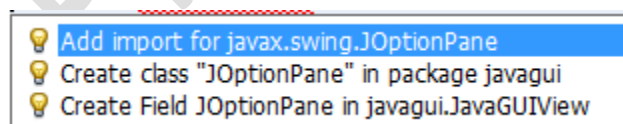
- e. Klik kanan pada btnSubmit kemudian pilih [Events]→ [Action] →[actionPerformed] maka akan masuk ke code editor. Kemudian ketikkan code di bawah ini:

```
private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {  
    String nama="";  
    String hobi="";  
    String jenisKelamin="";  
    String jurusan="";  
    String info="";  
  
    nama=txtNama.getText();  
    if(cbHob1.isSelected())  
        hobi += " Makan ";  
    if(cbHob2.isSelected())  
        hobi += " Shopping";  
    if(cbHob3.isSelected())  
        hobi += " Tidur ";  
  
    if(rdLaki.isSelected())  
        jenisKelamin="Laki-laki";  
    else  
        jenisKelamin="Perempuan";  
  
    jurusan=lstJurusan.getSelectedValue().toString();  
  
    info="Nama saya adalah " +nama +"\n";  
    info += "Hobby saya: " + hobi +"\n";  
    info += "Jenis Kelamin: " +jenisKelamin +"\n";  
    info += "Saya kuliah di UKDW dan mengambil jurusan " +jurusan +".|";  
    JOptionPane.showMessageDialog(null, info);  
}
```

Gambar di bawah untuk meng-import JOptionPane secara otomatis.

```
325 |         info="Nama Saya adalah " +nama;  
    |         JOptionPane.showMessageDialog(null, info);  
327 |     }
```

Klik icon lampu dan pilih “Add import for javax.swing.JOptionPane”

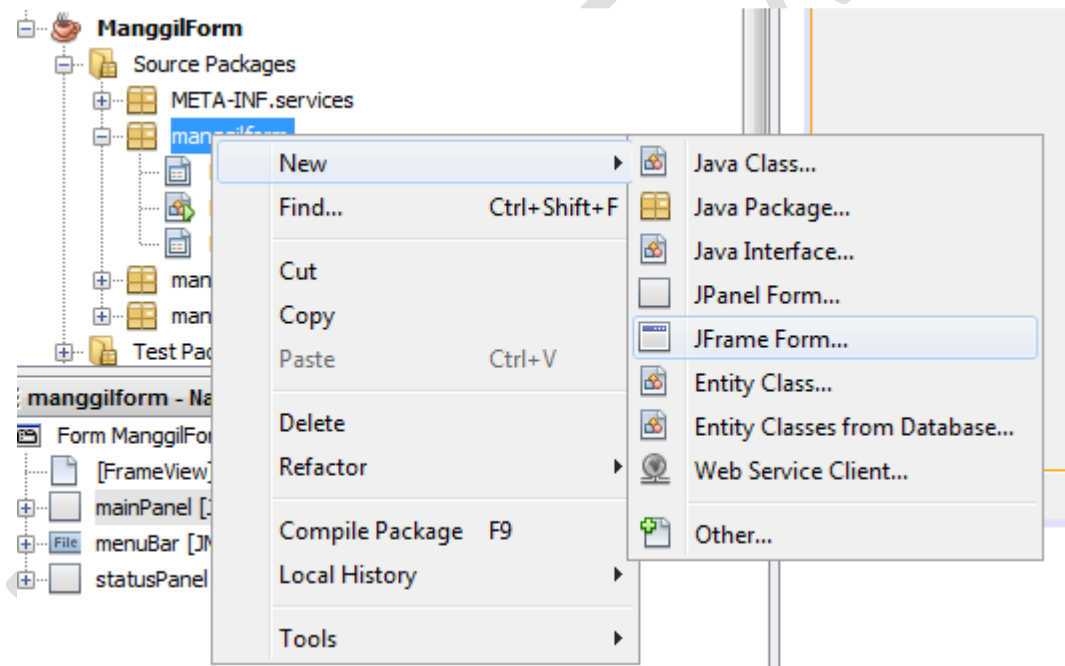


- f. Pada tahap ini telah berhasil membuat suatu aplikasi sederhana berbasis GUI. Untuk melihat hasilnya, klik **Build** dan **Run**. Coba klik radio button Laki-laki dan Perempuan. Lihat apa yang terjadi! Keanehan apa yang bisa kalian temukan?

- g. Untuk mengatasi keanehan tersebut, mari kembali membuka design pada form dan menambahkan satu komponen baru ke dalam form yaitu “**Button Group**”. Masuk ke **property buttonGroup** milik **rdLaki** dan **rdPerempuan** kemudian pilih **buttonGroup1**. **Build** dan **Run** kembali aplikasi Anda kemudian klik radio button Laki-laki dan Perempuan.

2. Latihan 2: pemanggilan form lain dari suatu form induk

- Buatlah sebuah **Project Baru** dan pilih **Java Desktop Application**
- Gantilah **Nama Project** menjadi **ManggilForm**
- Tambahkan sebuah **Button** pada form **ManggilForm**. Ubah property text Button tersebut menjadi **Panggil Form**.
- Tambahkan 1 form lagi sebagai form yang akan dipanggil.
Caranya: Klik kanan pada package **manggilform** kemudian pilih **[new]→ [JFrame Form]** seperti tampak pada gambar di bawah ini.



Ubahlah **Class Name** menjadi **FormAnak**. Kemudian tekan **finish**. Pembentukan form anak telah selesai.

- Tambahkan sebuah Label. Ubah property text Label tersebut dengan “Saya adalah Form Anak”.

A screenshot of a Java Swing window with a light gray background and a thin orange border. The text "Saya Form Anak" is displayed in the center of the window.

- f. Tambahkan code di bawah ini pada Button yang telah kita buat tadi di form ManggilForm.

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    FormAnak formAnak=new FormAnak();  
    formAnak.show();  
}
```

- g. Build dan Run aplikasi tersebut. Coba Tekan tombol Panggil Form.

3. Latihan 3: menggunakan fungsi kondisi (pemilihan)

penentuan lulus atau tidak.Syarat lulus jika nilai akhir di atas 75.

A screenshot of a Java Swing window titled "SELEKSI LULUS / TIDAK". The window has a light gray background and a blue border. It contains two rows of labels and text fields. The first row has the label "NILAI AKHIR" followed by a text field and an "OK" button. The second row has the label "KELULUSAN" followed by a text field.

- Buka Netbeans → File → New Project → Java → Java Application
- Nama project : **belajar lagi** → klik kanan pada folder project→ New JFrame Form
- Beri nama kelas : **LULUS** → Desain form dengan menggunakan toolbox

Terdiri dari

- jLabel1 → “SELEKSI LULUS/TIDAK”
- jLabel2 → “NILAI AKHIR”
- jLabel3 →”KELULUSAN”
- jButton1 → “OK”

- jTextField1 → untuk input nilai
 - jTextField2 → untuk output kelulusan
- d. masukkan program di bawah ini pada tombol OK:

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    double nilai;
    String hasil;
    nilai = Double.parseDouble(jTextField1.getText());
    if (nilai >= 60) {
        hasil = "LULUS";
        jTextField2.setText(hasil);
    }
    else {
        hasil = "TIDAK LULUS";
        jTextField2.setText(hasil);
    }
}
```

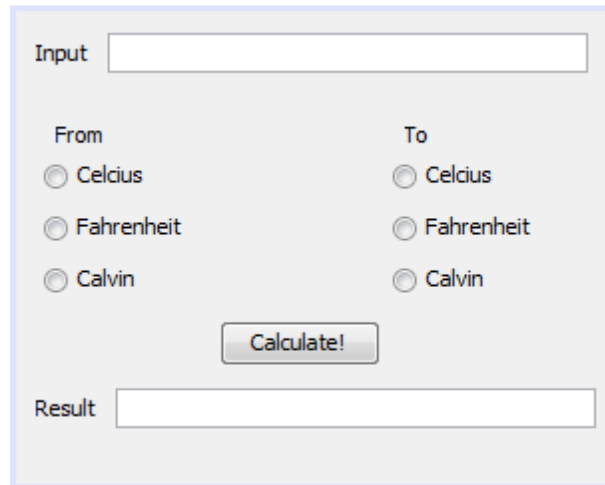
- e. Build dan Run program.

D. PRAKTIKUM

1. Buatlah program kalkulator sederhana seperti yang terdapat pada modul 10 dengan memanfaatkan bermacam-macam komponen – komponen SWING yang terdapat pada NetBeans! Ketentuan: jika nilai 1 dan 2 tidak diisi dan anda menekan salah satu tombol, maka akan keluar pemberitahuan bahwa nilai belum terisi, bagian **hasil** tidak bisa di isi manual. Contoh:



2. Buatlah sebuah Aplikasi yang bisa mengkonversi suatu temperatur dengan Java. Contoh bentuk output:



3. Buatlah form pendaftaran peserta didik baru di SMK Negeri 2 Trenggalek, dengan ketentuan:

Jumlah nilai NUN sebagai penentu masuk jurusan, misal kriteria jumlah NUN dari setiap jurusan: (Jika kalian ingin membuat standar sendiri, dipersilakan)

- KGSP : min 150
 - DPIB : min 220
 - TPTU : min 170
 - RPL : min 240
 - TB : min 120
 - AKL : min 200
4. Buatlah sebuah aplikasi pemesanan makanan di Kantin Sekolah dengan ketentuan, terdapat pilihan menu makanan, menu minuman, total harga yang harus di bayar. (Untuk pemilihan menu memanfaatkan checklist atau radio button).
5. Jangan lupa sertakan identitas kalian pada di setiap program yang kalian buat dengan menggunakan Label, misal copyright by Novi Dyah Puspitasari.
6. Buatlah laporan praktikum!

😊😊😊😊😊SELAMAT MENGERJAKAN😊😊😊😊😊