

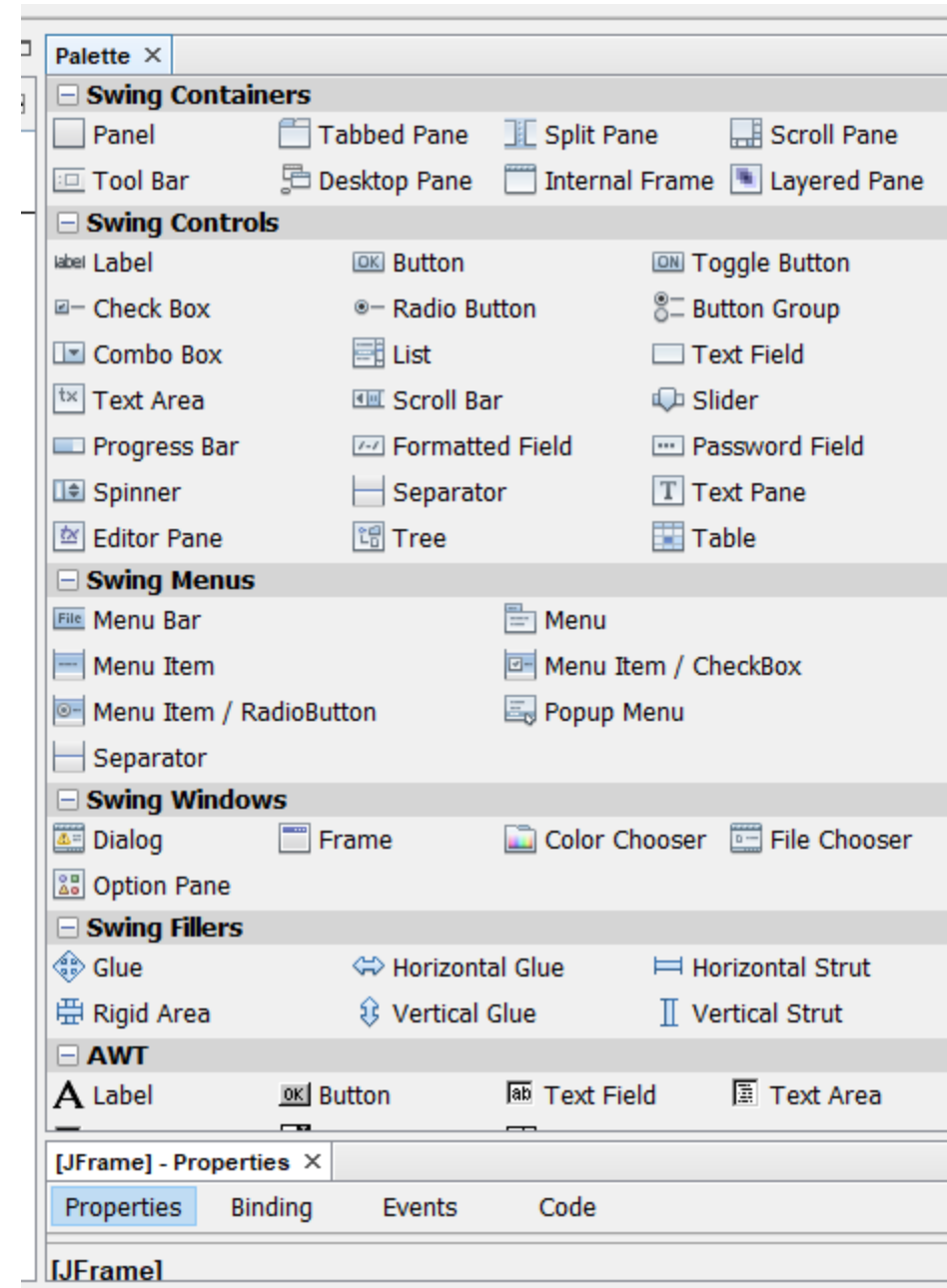
# TEKNIK DESAIN ANTARMUKA, PENGOLAHAN INPUT USER DAN MENGOLAH INPUT USER PADA APLIKASI DESKTOP



# Pengantar

- Berbicara GUI Builder Java kita akan menemukan dua komponen penting yang dijadikan sebagai landasan pemrograman GUI, yang pertama adalah komponen **Abstract Windowing Toolkit (AWT)** dan **javax swing**.
- Pustaka AWT adalah pustaka yang digunakan lebih awal jika dibandingkan dengan javax swing.
- javax swing adalah pustaka untuk memperbaiki kelemahan-kelemahan dan memperkaya pustaka komponen AWT.

# Swing dan AWT



# Swing Containers

- Swing Containers, berfungsi menempatkan komponen-komponen sebagai background.
  - Panel, Merupakan panel yang dapat digunakan sebagai papan dari suatu bidang atau layer lain.
  - Split Pane, Merupakan 2 tombol yang terbagi menjadi 2 sisi yaitu kiri dan kanan.
  - Tool Bar, Merupakan bar yang dapat di masukan tool di dalamnya, biasa nya di gunakan di atas layer.
  - Internal Frame, Merupakan bingkai layar baru yang berada di dalam bingkai utama.
  - Tabbed Pane, Merupakan panel yang memiliki tab yang bisa beralih dari tab lainnya.
  - Scrol Pane, Merupakan panel yang dapat di scrol baik horizontal maupun vertikal.
  - Desktop Pane, Merupakan container yang digunakan untuk membuat multi dokumen interface atau dekstop virtual.
  - Layer Pane, Sebuah container yang memungkinkan setiap komponen didalamnya saling overlap jika diperlukan

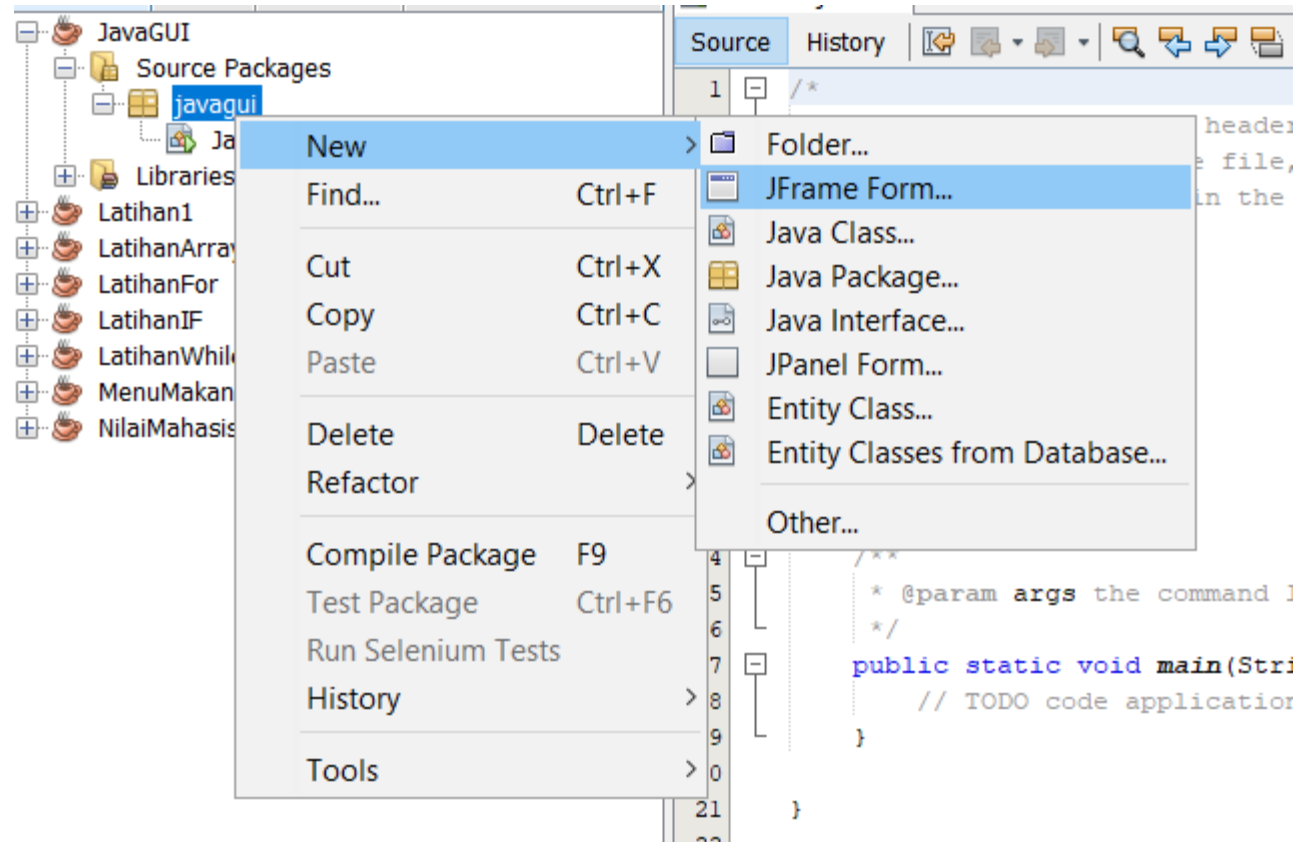
# Swing Control

- Swing Control yaitu menempatkan komponen-komponen yang berfungsi untuk pengecekan swing.
  - Label, Sebuah kolom atau area yang di gunakan untuk menampilkan tulisan.
  - Combo Box, Yang di gunakan untuk drop down list.
  - Text Field, Komponen yang digunakan sebagai input.
  - Button, Sebuah simbol sederhana.
  - Format Field, Mengedit sebuah nilai didalam nya yang memiliki angka tersebut.
  - Separator, Digunakan untuk keperluan umum seperti komponen dan memberikan garis pemisah.
  - Editor Pane, Adalah bidang yang disebut text area yang berfungsi untuk menulis apapun di dalamnya.
  - dll

# Swing Menus

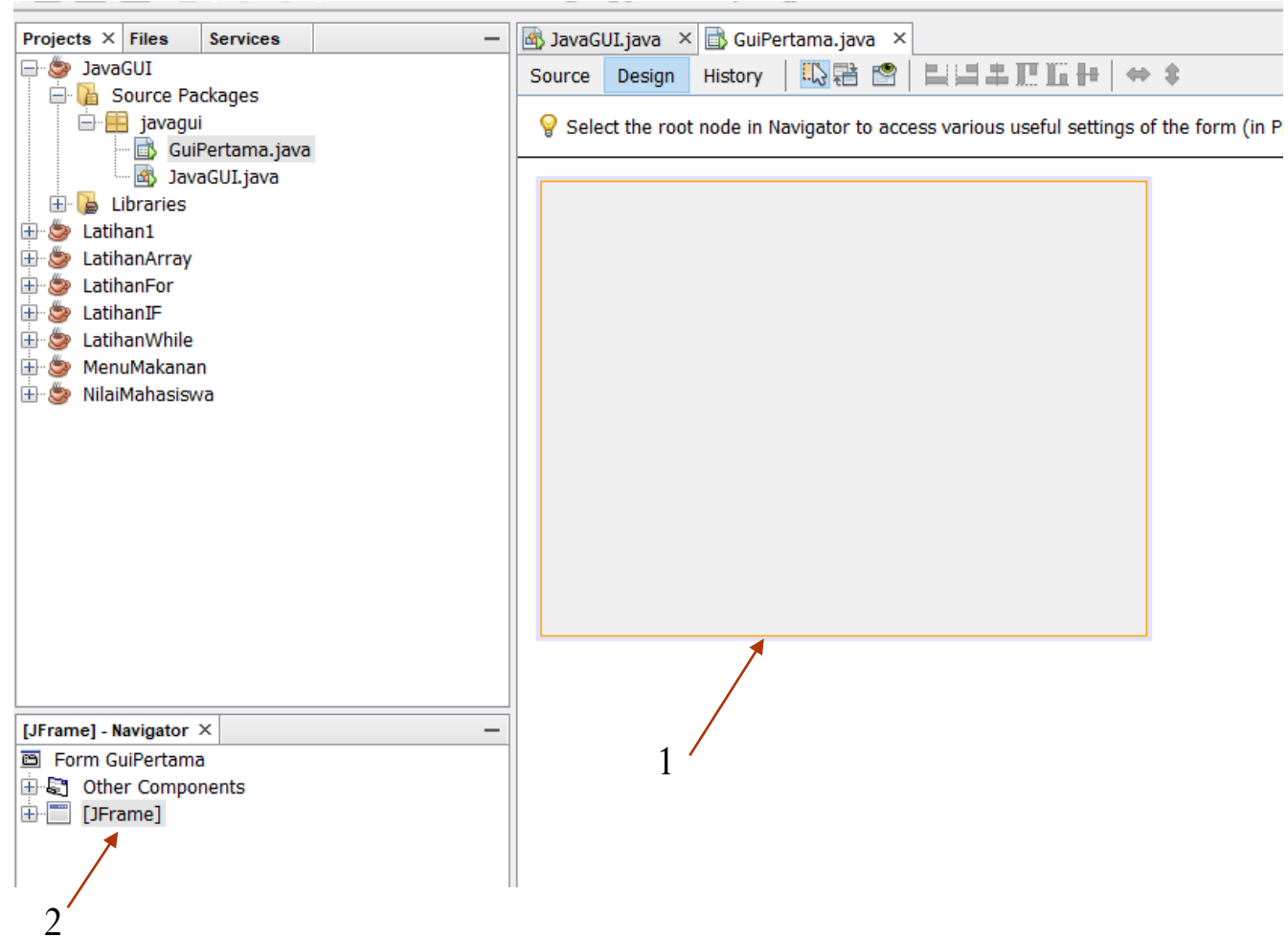
- Swing Menus memiliki komponen-komponen yang dapat digunakan untuk keperluan menu atau navigasi.
  - Menu Bar, komponen untuk menampilkan menu dari item.
  - Menu Item, Sebuah item atau menu tunggal dalam menu.
  - Radio Button, Sebuah item menu tunggal atau kontak centang di dalam menu.
  - Popup Menu, Layer munculkan kecil yang ada beberapa pilihan
  - Menu, Sebuah menu tunggal bisa digunakan sebagai menu.
  - dll

# Praktikum



# Praktikum

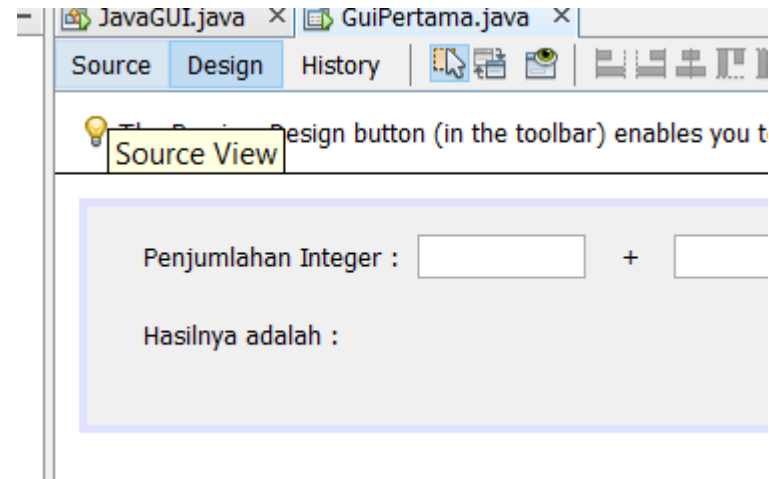
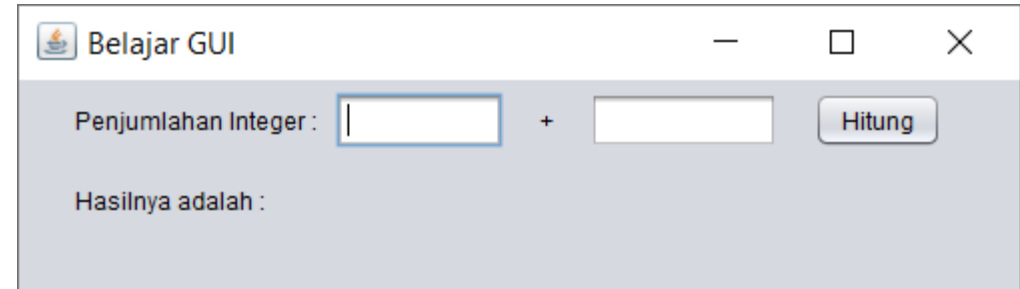
1. Tempat meletakkan palette
2. Untuk melihat susunan hirarki gui, bisa dilihat di window inspector (Netbeans 6.9.x ke bawah) atau di window navigator (Netbeans 7.0 ke atas).





# Bekerja dengan JLabel, JTextField dan JButton

- Buat desain seperti berikut
- Klik tombol Source untuk membuka jendela yang menampilkan kode sumber atau bisa klik dua kali di JButton



# Praktikum

- Tambahkan coding dibawah

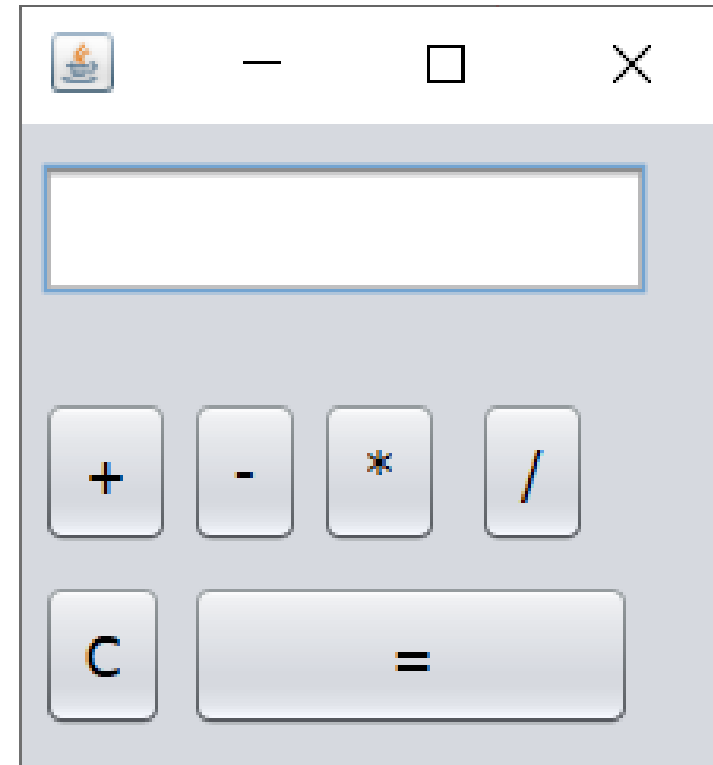
```
private String str = "Hasilnya adalah : ";  
private int a, b, hasil;  
/**
```

- Tambahkan coding berikut didalam method jButtonon

```
private void btnProsesActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int a = Integer.parseInt(txtNilai1.getText().trim());  
    int b = Integer.parseInt(txtNilai2.getText().trim());  
  
    hasil = a + b;  
  
    lblHasil.setText(str + hasil);  
}
```

# Latihan

- Buat program kalkulator sederhana
- `txtNilai.requestFocus();` digunakan agar cursor menuju TextField

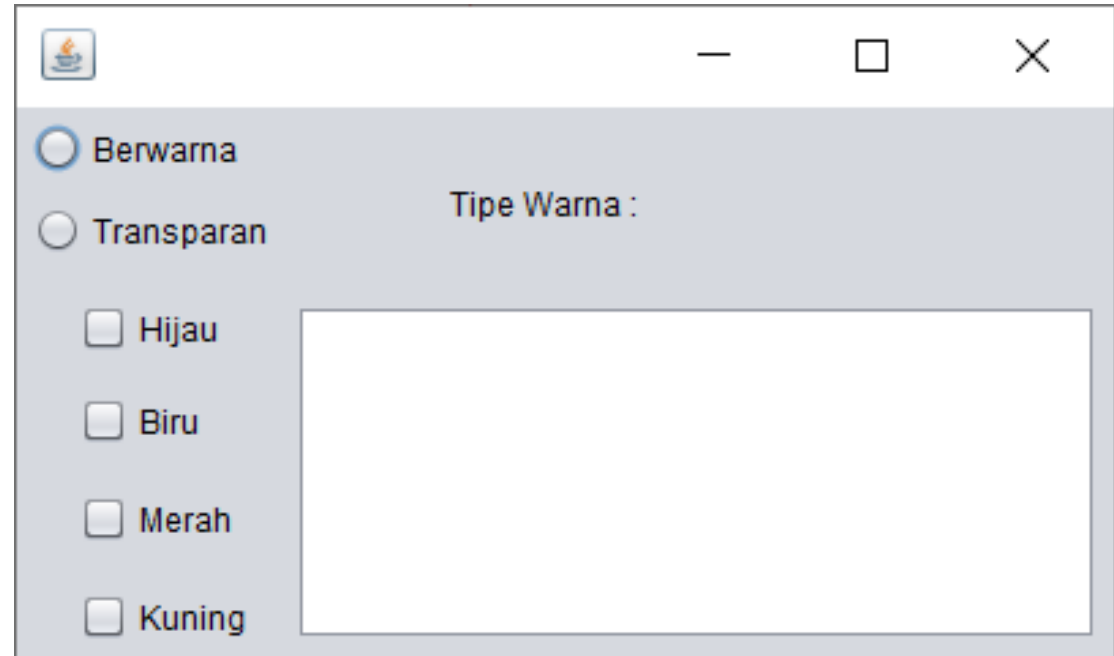
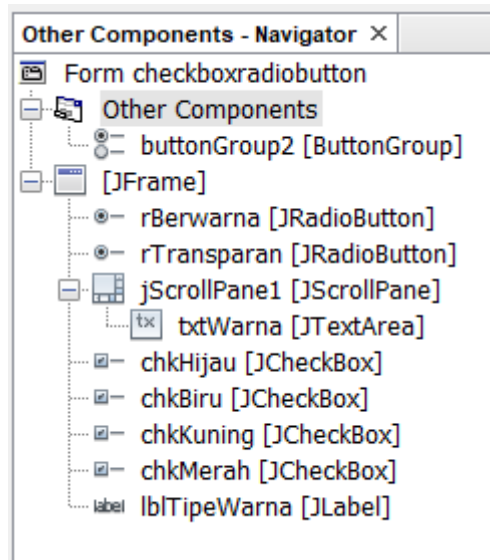


# JCheckBox dan JRadioButton

- JCheckBox dan JRadioButton hanya bisa mempunyai dua buah kemungkinan nilai, benar atau salah.
- Kedua komponen ini digunakan untuk merepresentasikan data yang berupa pilihan.
- JCheckBox digunakan jika pilihanya berupa multiple selection, sedangkan JRadioButton digunakan jika pilihanya berupa single selection.

# Praktikum

- Buat tampilan seperti disamping
- Palette yang diperlukan



# Praktikum

- Klik dua kali di JRadioButton

```
private void rBerwarnaActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    if(rBerwarna.isSelected()){  
        lblTipeWarna.setText("Tipe warna : " +  
            rBerwarna.getText());  
    }  
}
```

```
private void rTransparanActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    if(rTransparan.isSelected()){  
        lblTipeWarna.setText("Tipe warna : " +  
            rTransparan.getText());  
    }  
}
```

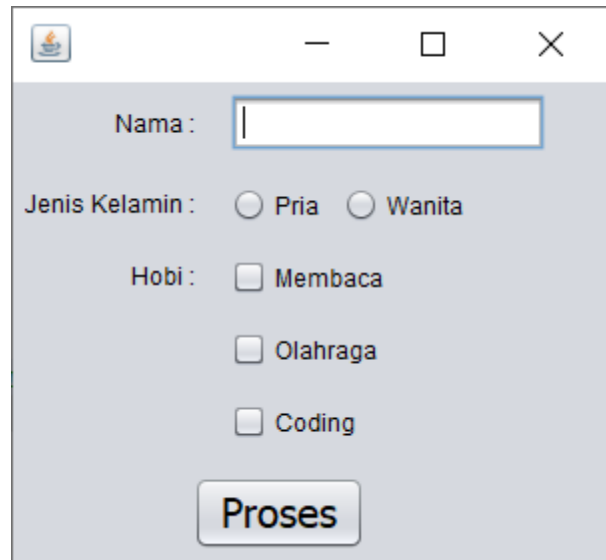
# Praktikum

- Buat method baru

```
private void tampilkanWarna() {  
    | StringBuffer warna = new StringBuffer();  
    if(chkBiru.isSelected()){  
        warna.append(chkBiru.getText() + " ");  
    }  
    if(chkHijau.isSelected()){  
        warna.append(chkHijau.getText() + " ");  
    }  
    if(chkKuning.isSelected()){  
        warna.append(chkKuning.getText() + " ");  
    }  
    if(chkMerah.isSelected()){  
        warna.append(chkMerah.getText() + " ");  
    }  
    txtWarna.setText(warna.toString());  
}
```

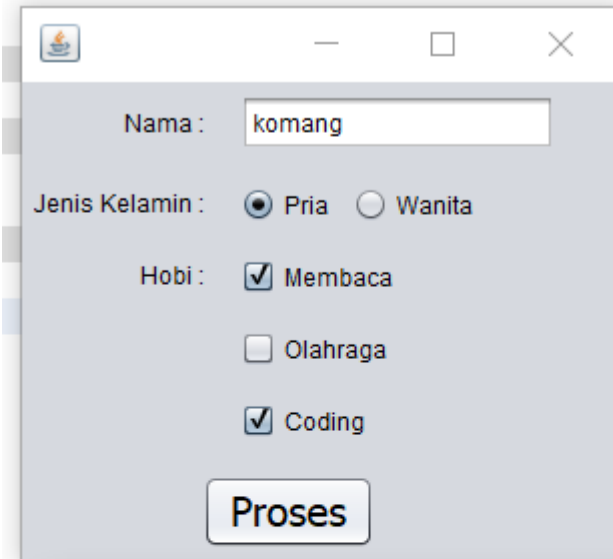
- Panggil method tersebut di setiap JCheckBox

# Latihan



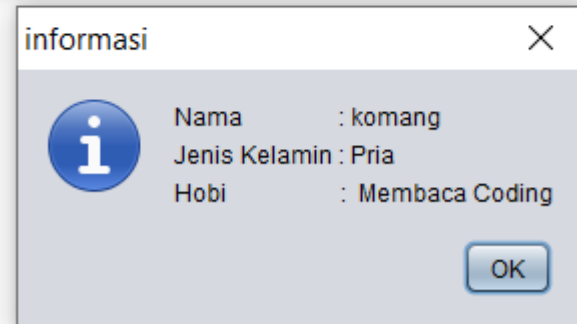
A Java Swing window titled "Proses" with a standard title bar (minimize, maximize, close). The window contains the following elements:

- A label "Nama :" followed by a text input field.
- A label "Jenis Kelamin :" followed by two radio buttons labeled "Pria" and "Wanita".
- A label "Hobi :" followed by three checkboxes labeled "Membaca", "Olahraga", and "Coding".
- A "Proses" button at the bottom.



The "Proses" window is shown with the following data entered:

- Nama : komang
- Jenis Kelamin : ☒ Pria ☐ Wanita
- Hobi : ☒ Membaca ☐ Olahraga ☒ Coding
- The "Proses" button is visible at the bottom.



An "informasi" (Information) dialog box with a blue information icon. It displays the data from the "Proses" form:

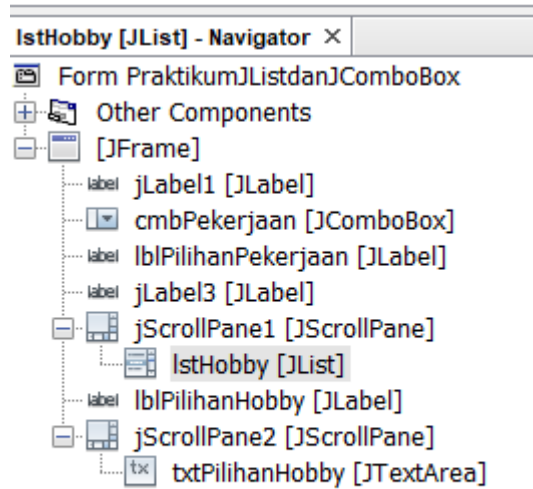
- Nama : komang
- Jenis Kelamin : Pria
- Hobi : Membaca Coding
- An "OK" button is at the bottom right.



# JList dan JComboBox

- JComboBox memerlukan tempat yang minimalis dibandingkan dengan JRadioButton
- JList memungkinkan multiple selection dengan menekan tombol : SHIFT + Left Click atau CTRL + Left Click. Kemampuan ini membantu user jika harus melakukan multiple selection.

# Praktikum



The screenshot shows the 'IstHobby' application window. It has a title bar with a standard icon and window controls. The main content area is divided into two sections. The top section is titled 'Pekerjaan' and contains a 'Dosen' dropdown menu and a label 'Pekerjaan anda :'. The bottom section is titled 'Hobi :' and contains a list box with the items 'Membaca', 'Olahraga', and 'Coding', and a label 'Hobi anda :'. To the right of the 'Hobi anda :' label is a large, empty text area with a horizontal scrollbar at the bottom.

# Praktikum

- Coding dari aplikasi yang dibuat

```
private void cmbPekerjaanActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    lblPilihanPekerjaan.setText(  
        "Pekerjaan anda : " +  
        cmbPekerjaan.getSelectedItem());  
}  
  
private void lstHobbyValueChanged(javax.swing.event.ListSelectionEvent evt) {  
    // TODO add your handling code here:  
    Object[] selectedItems = lstHobby.getSelectedValues();  
    if(selectedItems == null || selectedItems.length == 0)  
        txtPilihanHobby.setText("");  
    else{  
        StringBuffer strValues = new StringBuffer();  
        for(Object item : selectedItems){  
            strValues.append(item.toString() + ", ");  
        }  
        txtPilihanHobby.setText(strValues.substring(0, strValues.length()-2));  
    }  
}
```

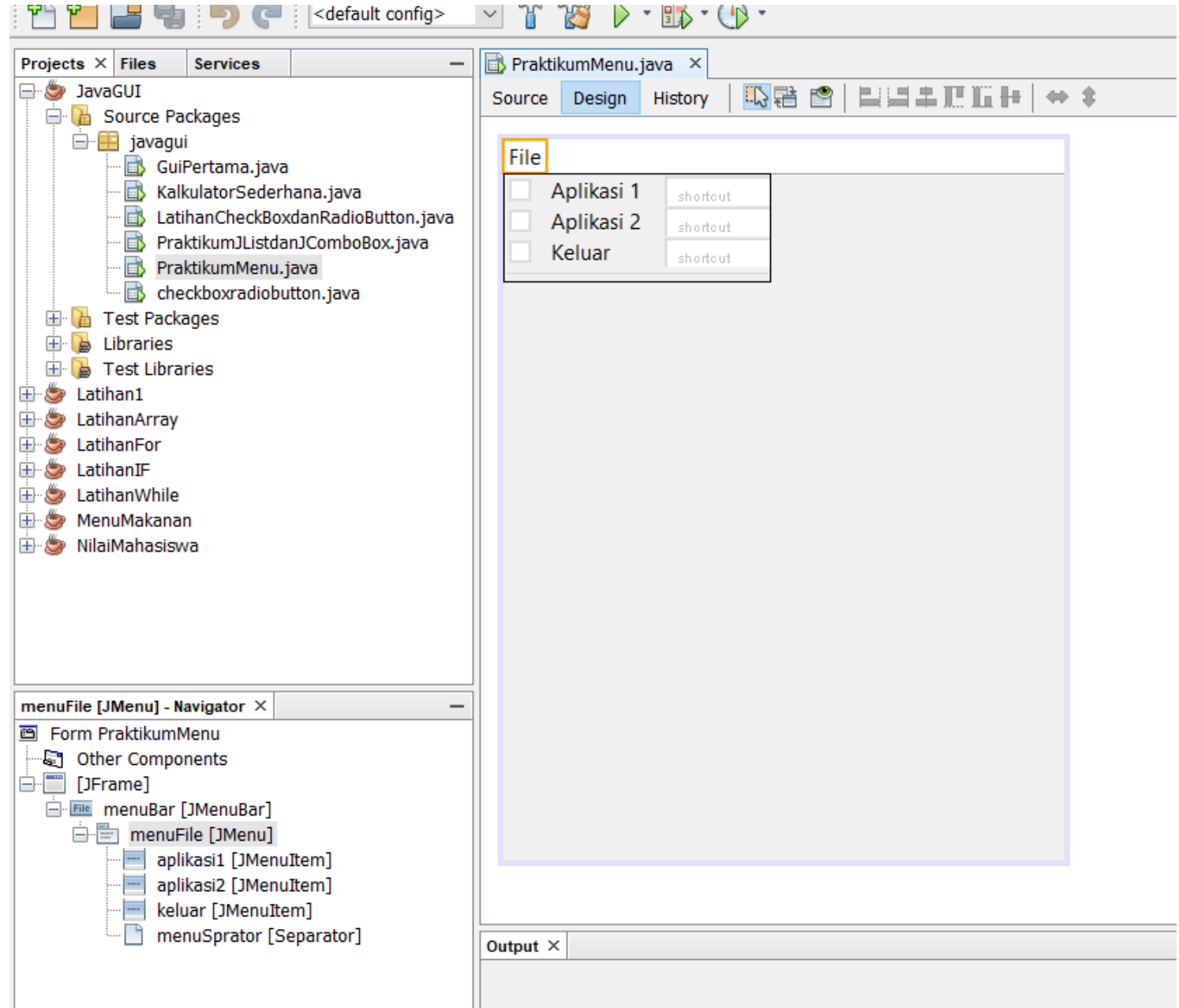
# Menu, Popup Menu dan Toolbar

- Menu, Popup menu dan Toolbar digunakan untuk melakukan navigasi dalam aplikasi. dengan ketiga komponen itu navigasi dalam aplikasi menjadi lebih fleksibel dan mudah digunakan oleh user.
- Menu dan Toolbar pada umumnya diletakkan di bagian atas dari aplikasi agar mudah ditemukan oleh user.
- Sedangkan Popup Menu bisa muncul di mana saja sesuai dengan konteks aplikasi.

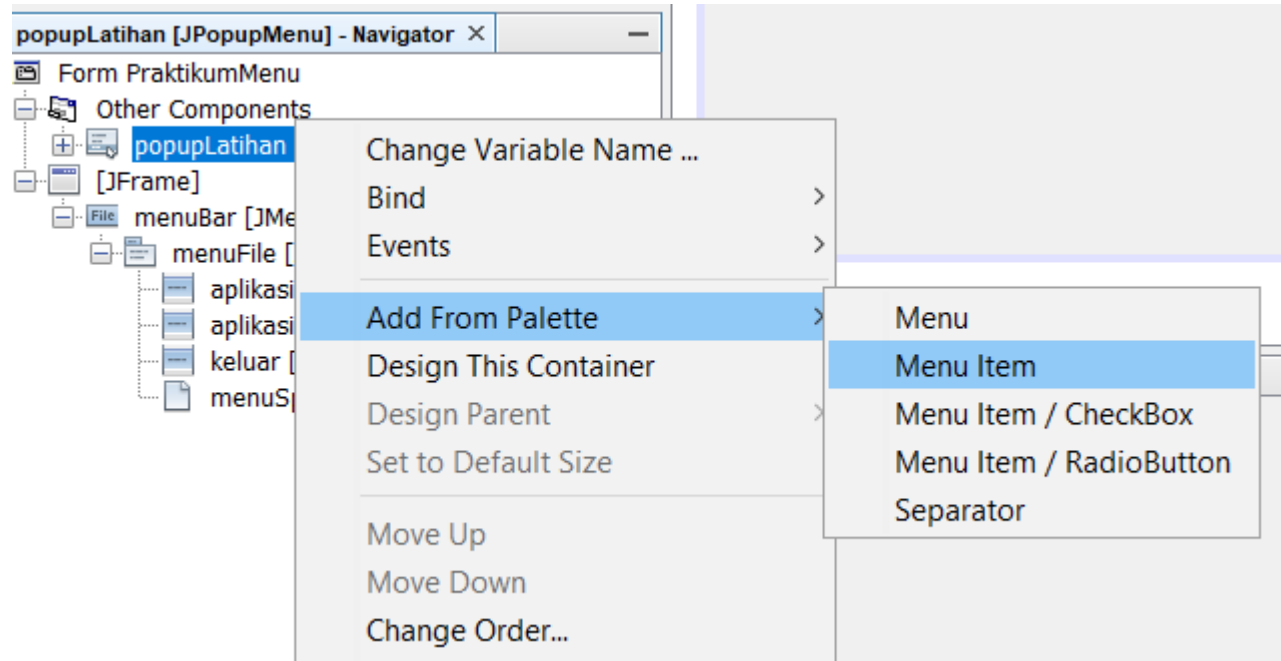
# Praktikum

- Membuat menu
- Klik 2 kali di aplikasi 1 dan kettikan coding dibawah

```
private void aplikasi1ActionPerformed(java.  
    // TODO add your handling code here:  
    GuiPertama gui = new GuiPertama();  
    gui.setVisible(true);  
}
```

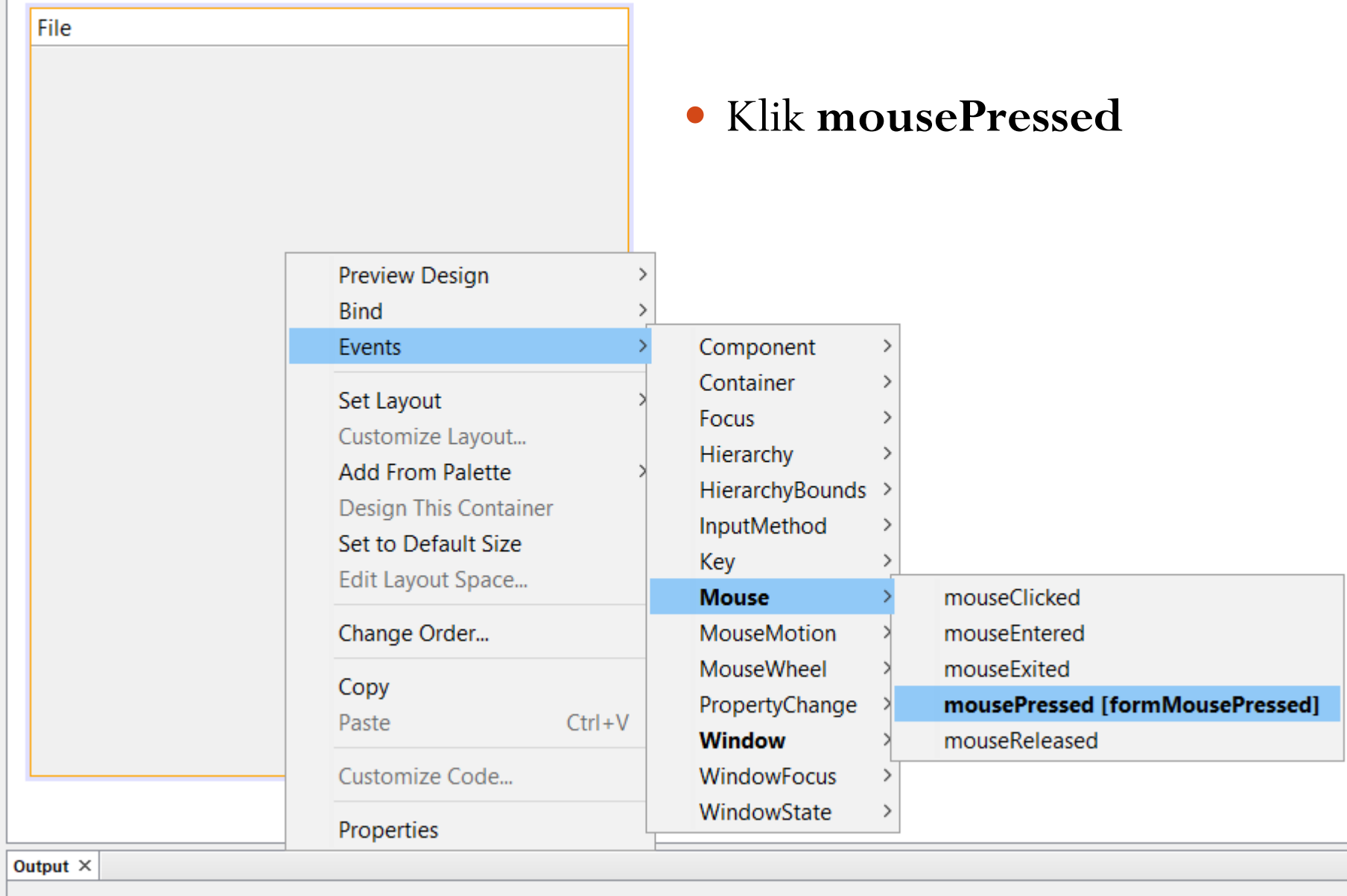


# Praktikum popup menu



# Praktikum

- Klik **mousePressed**



- Buat method baru dengan coding berikut

```
private void showpopupMenu(MouseEvent e) {  
    popupLatihan.show(this, e.getX(), e.getY());  
}
```

---

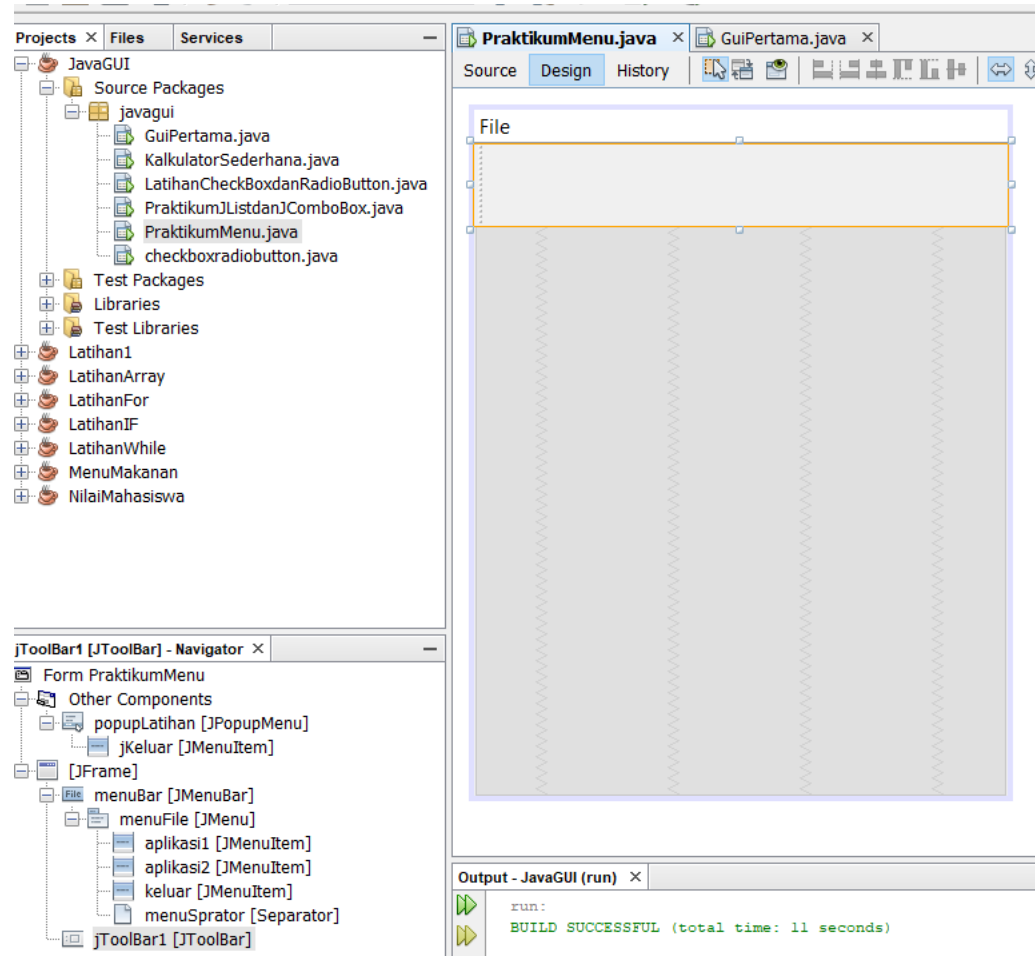
- Dan ketikkan didalam **mousePressed**

```
private void formMousePressed(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
    showpopupMenu(evt);  
}
```

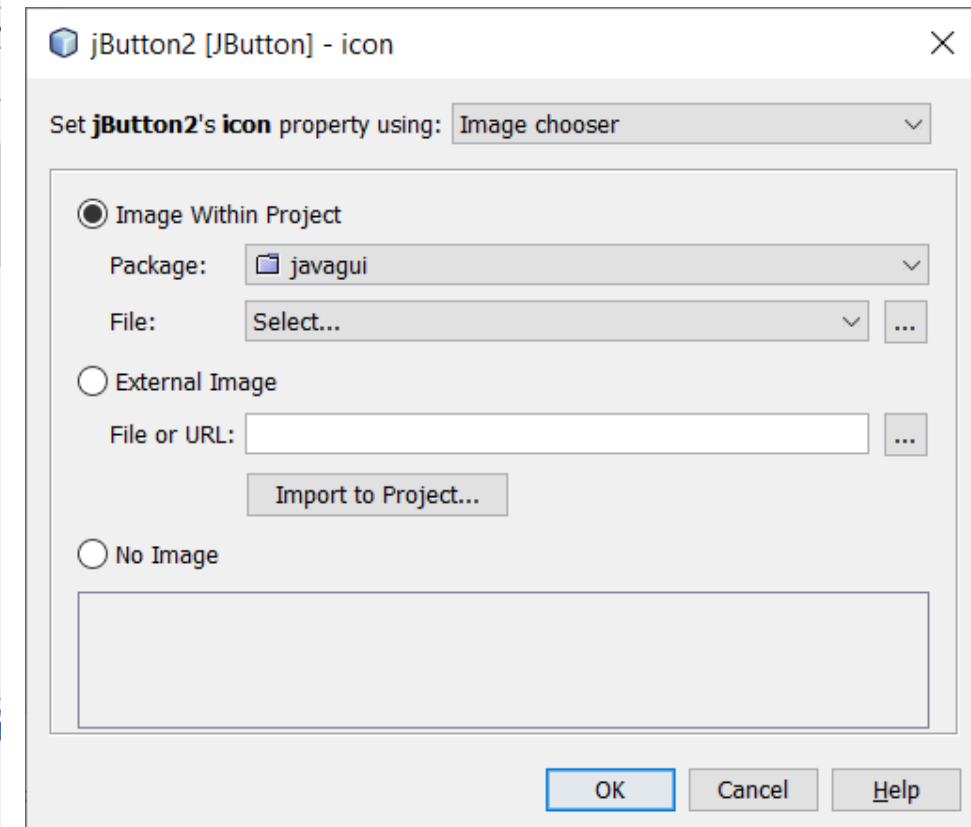
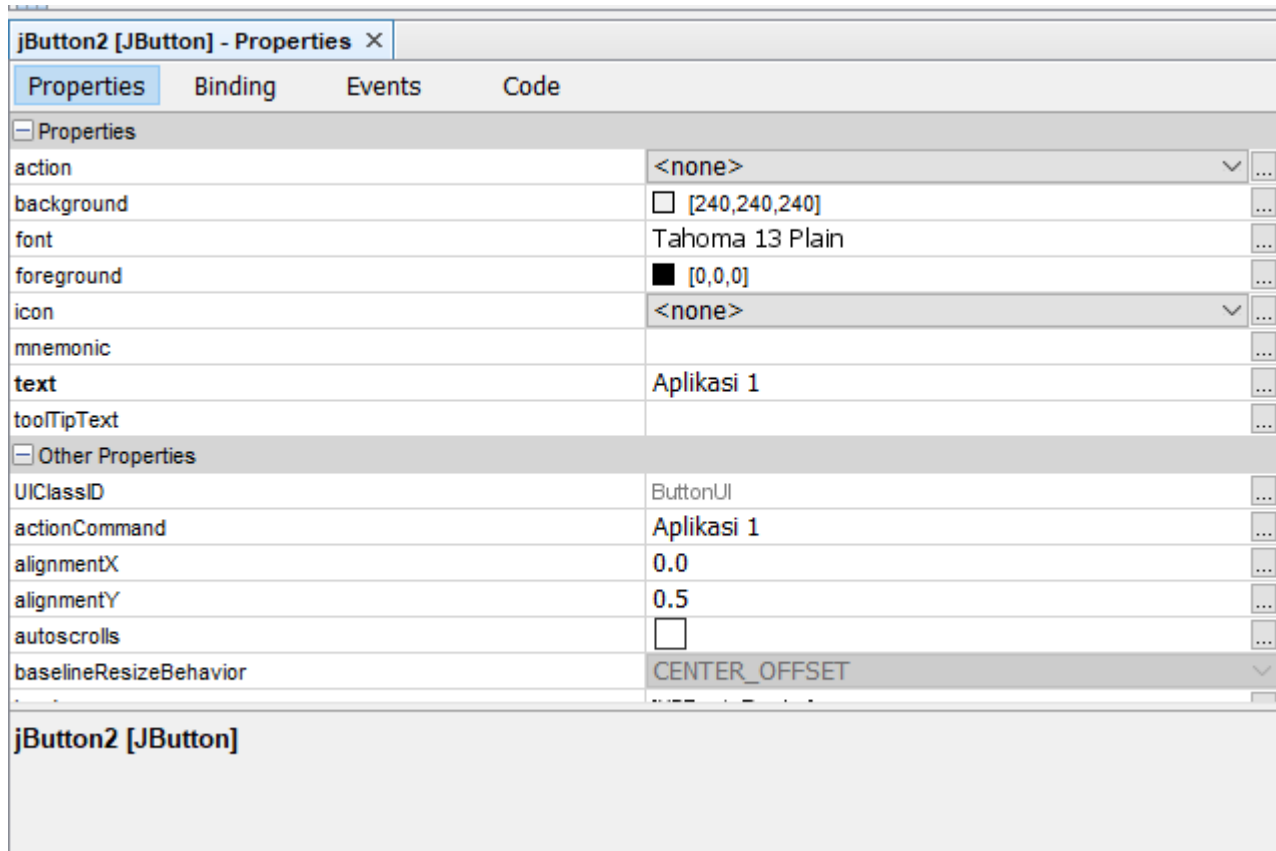


# Toolbar

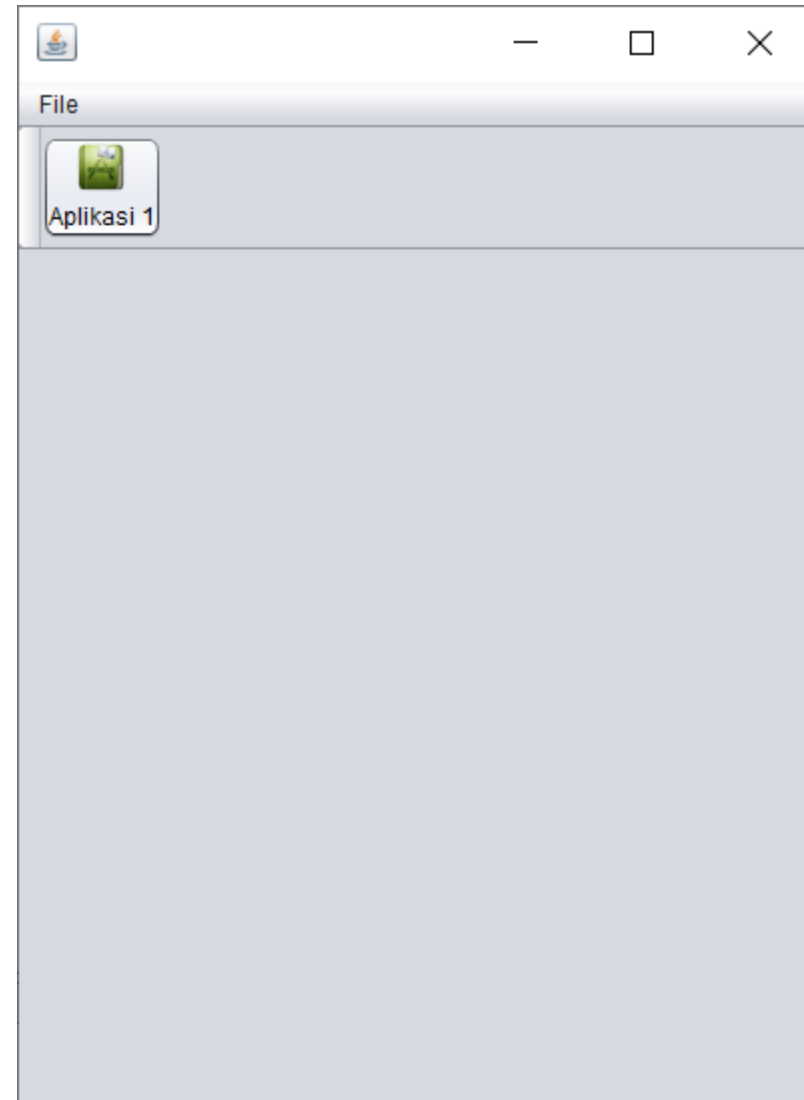
- Tambahkan toolbar di projek menu yang dibuat sebelumnya



- Tambahkan button dan isi icon



- Tampilan setelah ditambah icon



# Tugas

1. Deskripsikan hasil praktikum yang dilakukan ke dalam laporan!