

MODUL I

Instalasi Netbeans dan Setting Path

1. TUJUAN

- *Praktikan dapat melakukan instalasi dan setting Java Development Kit.*
- *Praktikan dapat menggunakan Jcreator sebagai editor pemrograman*
- *Praktikan dapat menjalankan (eksekusi) program Java sederhana yang ditulis dengan editor JCreator.*

2. TEORI DASAR

Untuk menulis program JAVA dapat digunakan berbagai editor, bahkan editor paling sederhana (seperti Notepad pun dapat digunakan. Namun pada umumnya, agar penulisan program lebih terarah, beberapa editor telah disediakan untuk membantu para programmer. Beberapa editor atau IDE (*Integrated Development Environment*) yang telah ada untuk Java, antara lain:

- a. NetBeans (*open source- Common Development and Distribution License (CDDL)*)
- b. NetBeans yang disponsori Sun Microsystems, yang dilengkapi dengan GUI Editor.
- c. Eclipse JDT (*open source- Eclipse Public License*)
- d. Eclipse dibuat dari kerja sama antara perusahaan-perusahaan anggota 'Eclipse Foundation' (beserta individu-individu lain). Banyak nama besar yang ikut dalam 'Eclipse Foundation', termasuk IBM, BEA, Intel, Nokia, Borland. Eclipse bersaing langsung dengan Netbeans IDE. Plugin tambahan pada Eclipse jauh lebih banyak dan bervariasi dibandingkan IDE lainnya.
- e. IntelliJ IDEA (commercial, free 30-day trial)
- f. Oracle JDeveloper (free)
- g. Xinox JCreator (ada versi berbayar maupun free)
- h. JCreator ditulis dalam C/C++ sehingga lebih cepat (dan menggunakan memori lebih sedikit) dari kebanyakan IDE lain.

Instalasi

Sebelum memulai pemrograman Java, kita harus menginstal Java Development Kit (JDK) yang disediakan secara gratis oleh *Sun Microsystem*. Java Development Kit tersedia dan dapat didownload pada situs *Java Sun Microsystem*, yaitu: <http://java.sun.com>. Java 1.6 untuk berbagai platform tersedia pada situs tersebut.

Untuk menginstal software Java dapat dilakukan dengan menjalankan file yang mengekstrak sendiri. Terdapat dua pilihan cara, yaitu:

- ☐ Menjalankan installer dari perintah MS DOS, dan
- ☐ Melakukan *double-click* ikon `jdk-6u11-windows-i586-p.exe`

Installer yang mengekstrak file JDK akan membuat direktori yang diperlukan, dan sekaligus menyalin file-file ke dalam direktori tersebut. Installshield Wizard memberikan tampilan bahwa Java siap diinstalasi.

Setting Variabel Path

Pada saat program MS DOS dieksekusi, secara otomatis mencari direktori yang ada untuk file target. Jika program atau file batch tidak tersedia dalam direktori yang ada, maka akan mencari semua drive dan direktori lainnya dalam *variable path system*.

Dalam pengembangan Java, tidak diperlukan variable path jika semua *software* yang diperlukan terdapat dalam direktori yang ada. Namun akan lebih mudah apabila dilakukan *setting variable path* agar sistem secara otomatis mencari file yang diperlukan. Dalam hal ini, path adalah drive dan direktori dimana java.exe, javac.exe, dan aplikasi pengembangan lain ditempatkan.

Apabila tidak dipilih direktori default, maka path adalah direktori yang dipilih pada saat instalasi. Perintah path sistem DOS biasanya terdapat dalam file autoexec.bat, yang dieksekusi selama sistem *startup*. Alternatif lain adalah, path dapat dimasukkan dari baris perintah MS DOS dengan sintaks sebagai berikut:

PATH=[drive:][path];[drive:][path]...

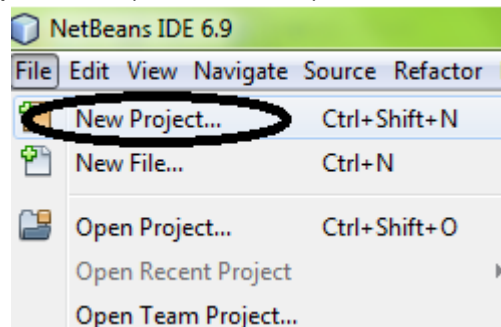
Apabila digunakan sistem operasi windows dengan JDK versi 1.5 atau yang lebih baru, maka setting variabel path tidak diperlukan lagi. Untuk pelaksanaan praktikum Pemrograman Berorientasi Objek nanti, digunakan JDK versi 1,6 sehingga proses instalasi tidak akan dilakukan. Namun apabila digunakan versi lama, maka diperlukan langkah-langkah *setting* sebagai berikut:

1. Klik kanan pada "My Computer" lalu klik properties. Kemudian klik "Advanced" lalu klik "Environment Variables" (**Gambar 1.1**).
2. Pada System Variables, pilih "path" lalu klik edit (**Gambar 1.2**).
3. Pada "variable value" tambahkan path dimana direktori bin nya Java diletakkan. Lalu klik OK (**Gambar 1.3**).

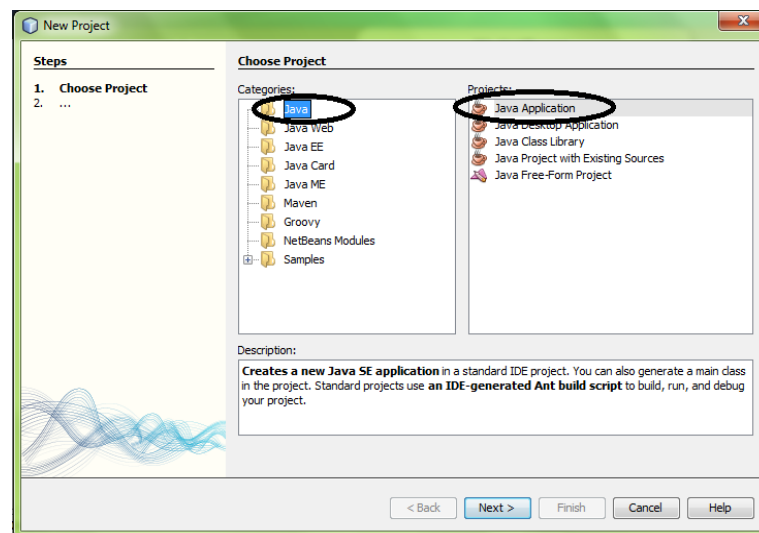


Gambar 1.1 Setting JDK pada Windows (langkah-1).

2. Untuk memulai membuat project baru, pilih file – New Project... pada bagian menu (**Gambar 1.5**).
3. Kemudian pada saat bagian choose project Category pilih java, dan pada bagian Projects pilih Java Application (**Gambar 1.6**).



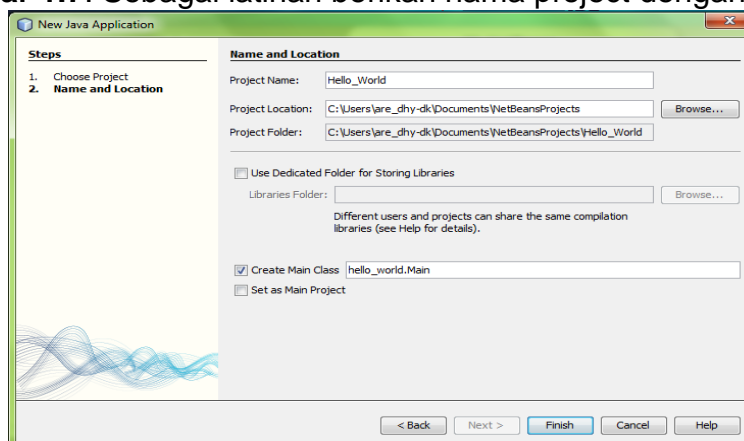
Gambar 1.5 Membuat project baru.



Gambar 1.6 Memilih kategori project.

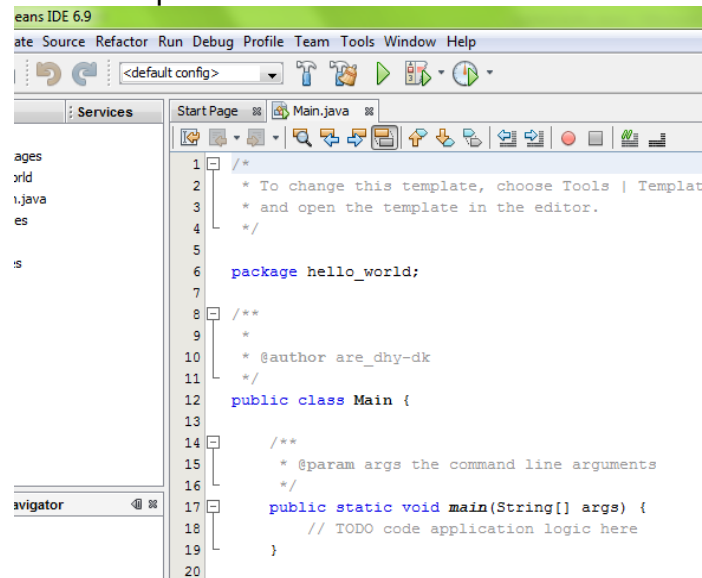
Selanjutnya klik next untuk masuk ke langkah selanjutnya.

4. Selanjutnya masukkan project name dan project location seperti terlihat pada **Gambar 1.7**. Sebagai latihan berikan nama project dengan hello_world.



Gambar 1.7 Memberi nama pada project.

Pada bagian create main class, dapat diberikan nama yang berbeda dari Project Name nya. Tidak perlu menambahkan .Main atau .Java, NetBeans akan secara otomatis memberikan ekstensi-nya. Setelah selesai, klik finish. Project hello_world yang dibuat seperti terlihat pada **Gambar 1.8**.



Gambar 1.8 Tampilan awal project hello_world.

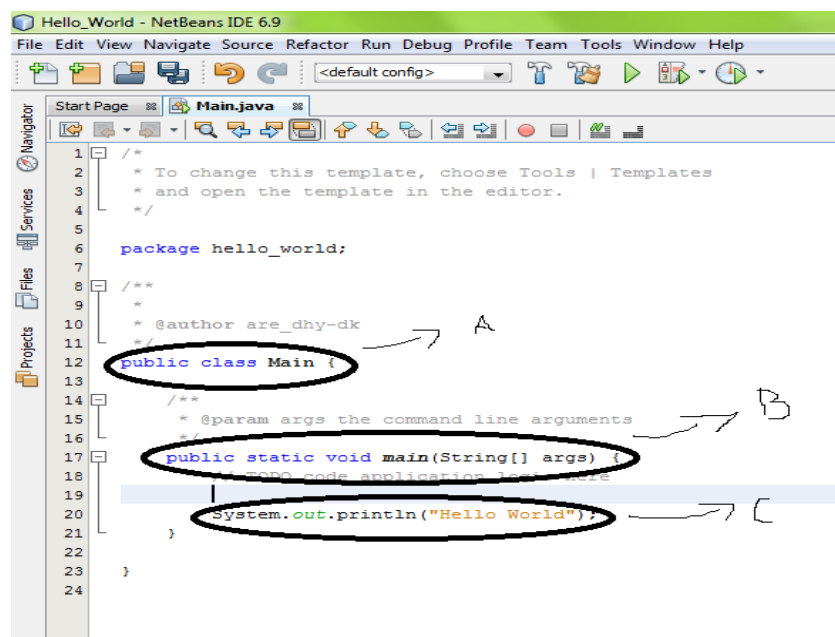
Dengan demikian sebuah project baru telah berhasil dibuat. Untuk pembuatan class baru, akan dijelaskan lebih lanjut pada praktikum berikutnya yang membahas khusus tentang pembuatan class.

Program sederhana

Pada project *hello_world* yang dibuat tambahkan perintah sebagai berikut:

```
System.out.println("Hello World!");
```

pada baris ke-19. Sehingga program yang terbentuk seperti pada **Gambar 1.9**.



Gambar 1.9 Program utama hello_world.

keterangan:

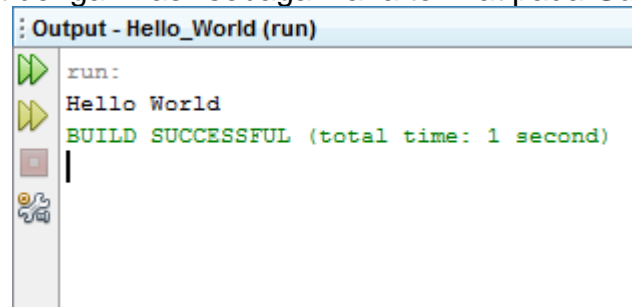
- a. Syntax utama untuk program yang ditulis dengan JAVA adalah:

```
[modifier][class] nama_class
{
    ...
}
```

Ada beberapa modifier pada JAVA, yaitu public, private dan protected. Modifier public diperlukan agar program dapat dikompilasi dengan baik.

- b. Bagian ini merupakan bagian utama yang ditempatkan pada bagian awal pada program JAVA. static menunjukkan tipe method. void menunjukkan bahwa method tidak mengembalikan nilai atau objek. main merupakan nama method utama program JAVA. String merupakan tipe argumen yang akan diterima sebagai parameter dari command JAVA. args merupakan array argumen yang dapat ditambahkan pada saat menggunakan command JAVA untuk menjalankan program JAVA.
- c. Perintah untuk menampilkan “Hello World” pada layar monitor.

Selanjutnya, program dapat dieksekusi dengan klik tombol .Hasilnya dapat dilihat pada window output dengan hasil sebagaimana terlihat pada **Gambar 1.10**.



Gambar 1.10 Hasil eksekusi program hello_world.

Penambahan komentar

Untuk membantu mengingat arti (tujuan) penulisan serangkaian kode yang telah ditulis, biasanya kita memberikan beberapa komentar pada program yang kita buat. Pada JAVA, untuk membuat komentar dapat dilakukan dengan cara:

- Komentar dengan menggunakan tanda //. Tanda ini digunakan untuk memberikan komentar dalam satu baris. Tanda // diletakkan pada awal baris.
- Komentar dengan menggunakan tanda /* dan */. Tanda ini digunakan untuk memberikan komentar yang ditulis dalam beberapa kalimat. Tanda /* diletakkan pada awal kalimat, dan tanda */ ditulis diakhir kalimat.

```
1 package hello_world;
2 /*
3  @author are_dhy-dk
4  Ini Adalah Program Pertama Ku
5  */
6
7 public class Main {
8
9     // Ini adalah Method Main
10    public static void main(String[] args) {
11
12        // Ini Adalah Syntax Output
13        System.out.println("Hello World");
14    }
15 }
16
```

Gambar 1.11 Penambahan komentar pada program

3. LATIHAN

Lakukan percobaan instalasi dan setting path mengikuti materi pada teori dasar.

MODUL II

PENGUNAAN KOMPONEN DASAR

1. TUJUAN

- Mahasiswa memahami cara penggunaan NetBeans 5.5 dan mampu membuat project sederhana dengan menggunakan Class JTextField, JButton dan JLabel.
- Menggunakan Class JTextField untuk meminta masukan dari user
- Menggunakan Class JButton untuk melakukan suatu pemrosesan
- Menggunakan Class JLabel untuk menampilkan keterangan

2. TEORI DASAR

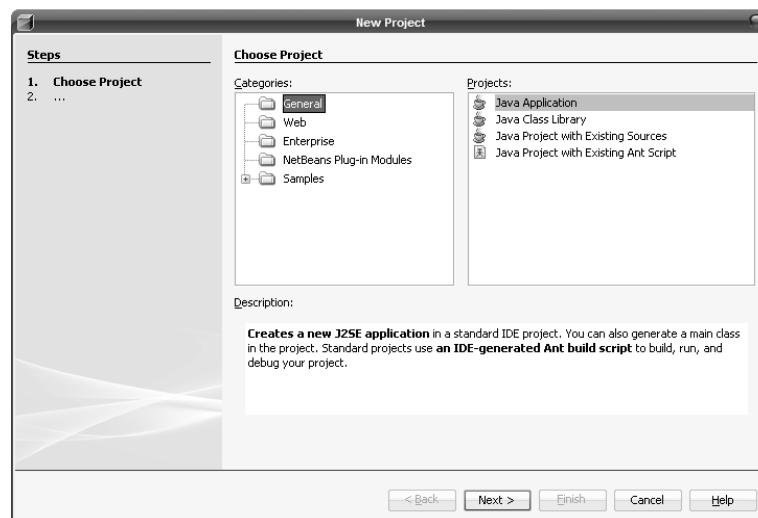
Netbeans sebagai IDE ditujukan untuk memudahkan pemrograman Java. Dalam NetBeans pemrograman dilakukan berbasiskan visual dan *event driven*. Untuk membuat dialog atau *user interface* tidak diperlukan untuk membuat teks program baris per baris secara manual melainkan cukup memilih dengan melakukan klik pada *component pallette* dan teks program akan di *generate* secara otomatis.

Suatu *user interface* mempunyai bentuk dasar yang sederhana yaitu tempat untuk mengisikan sesuatu, yang dapat diwakili oleh Class JTextField. Tulisan sebagai penjelas informasi yang dapat diwakili oleh Class JLabel dan suatu tombol untuk memberi kesempatan user melakukan pemrosesan yang dapat diwakili oleh Class JButton.

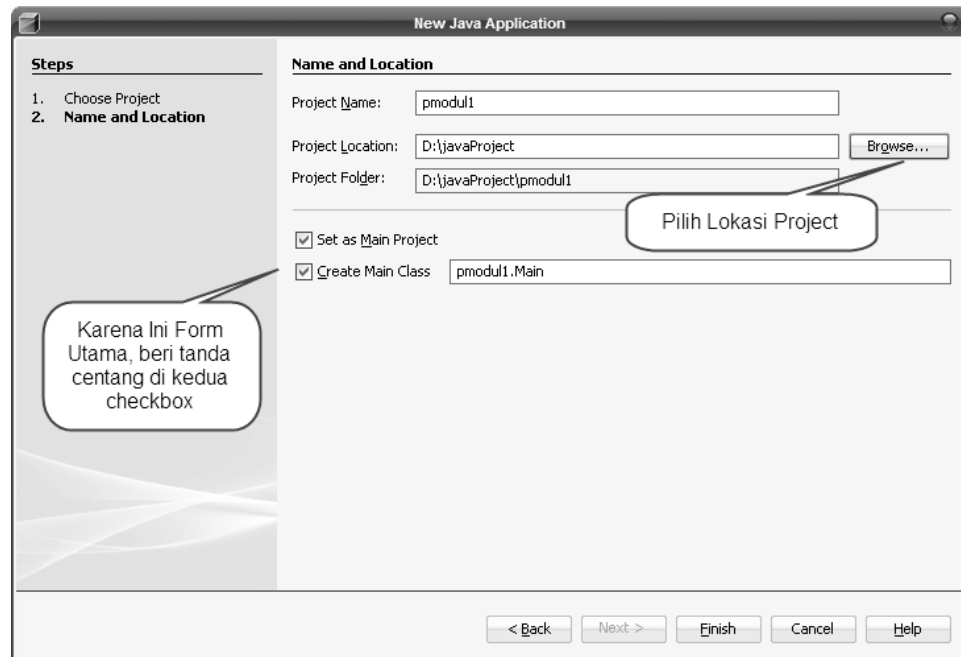
Cara Menggunakan Komponen

1. Jalankan NetBeans 5.5
2. Buat Project Baru dengan memilih File | New Project

Akan memunculkan form berikut, Pilih Next



3. Akan memunculkan form berikut :



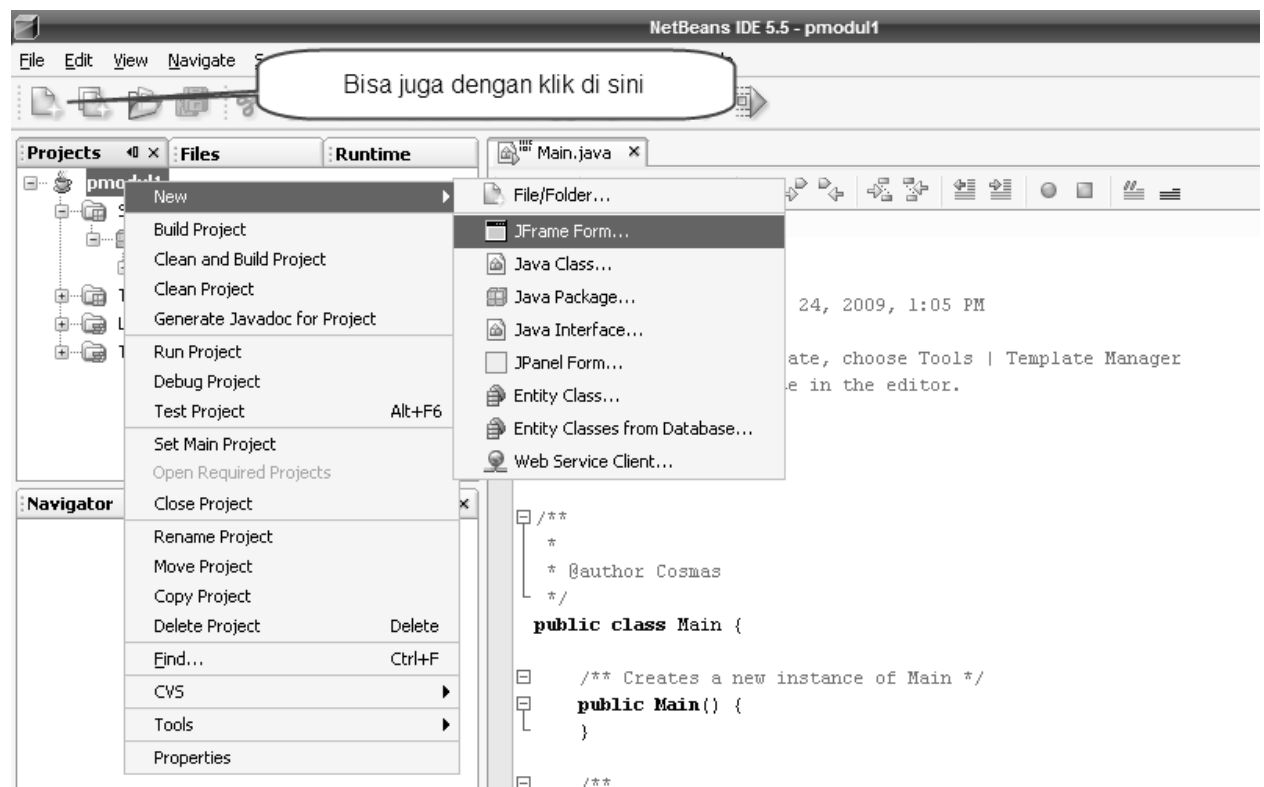
4. Pada Project Name beri nama : **pmodul1**

5. Pada Project Location pilih lokasi tempat penyimpanan Data (silakan simpan di drive masing-masing, misalnya di folder **H: \javaProject** (jika belum ada, buat lebih dahulu))

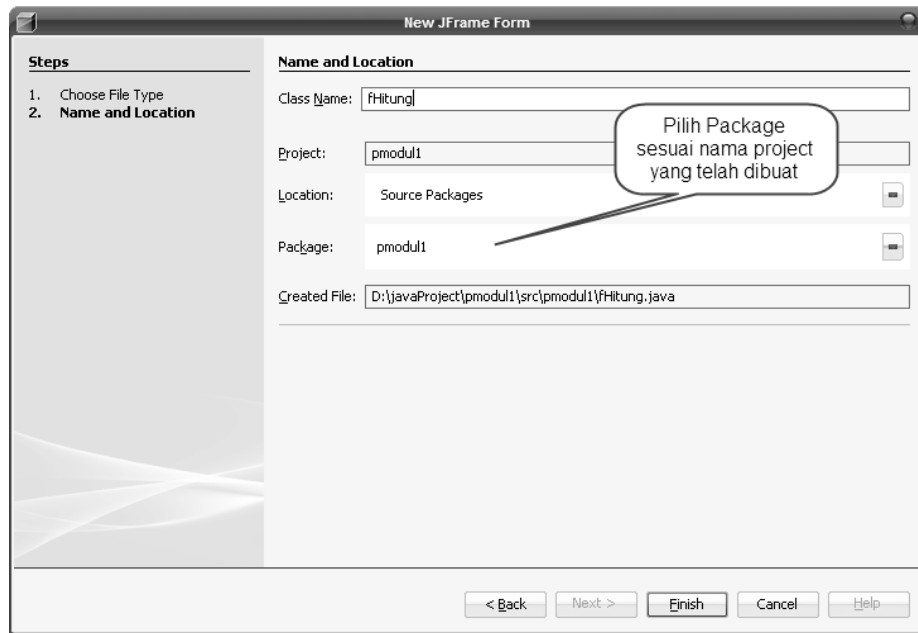
6. Klik Finish

7. Akan terbentuk file bernama : main.java yang merupakan program utama

8. buat form baru dengan melakukan Klik kanan pada : **project pmodul1 | new | JFrame Form...**



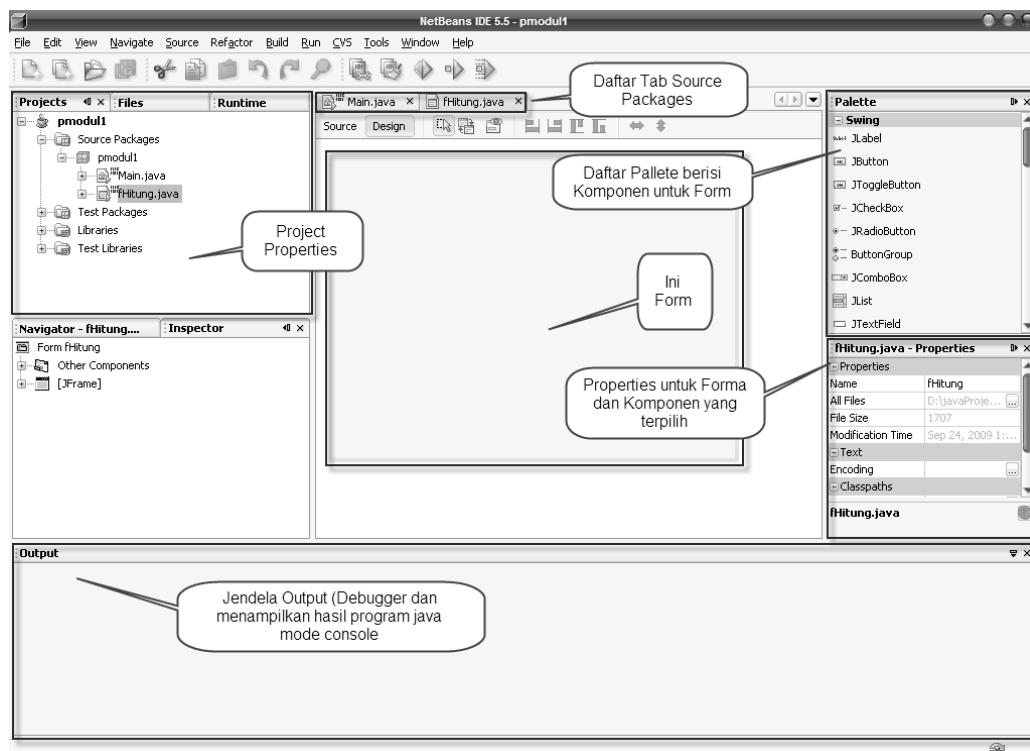
9. Akan memunculkan :



10. Pada Class Name beri nama : **fhitung**

11. Klik Finish

12. Akan masuk ke lingkungan kerja Netbeans dengan form sebagai berikut :



13. Rubahlah Title Form (lihat di properties) dengan tulisan : Form Hitung



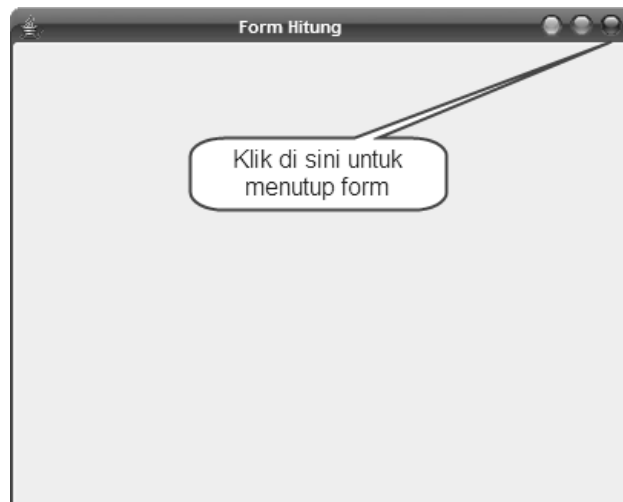
14. Kemudian Klik **main.java** di Tab Source, tambahkan pada bagian **main** statemen berikut ini yang berfungsi untuk menjadikan form fHitung akan dijalankan pada saat project di-Run

```
/**
 * @param args the command line arguments
 */
public static void main(String[] args) {
    // TODO code application logic here
    new fHitung().setVisible(true);
}
```

15. Kemudian jalankan project dengan cara klik toolbar Run atau Tekan F6



16. Maka Form fHitung akan dijalankan :

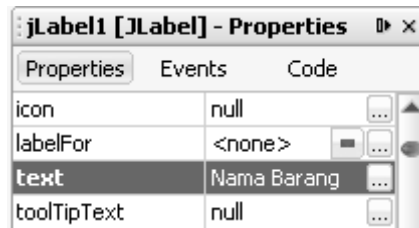


17. Untuk penambahan **Komponen Label** dalam form :

- Klik Class JLabel pada Swing Pallette
- Klik form pada posisi yang diinginkan untuk meletakkan Class JLabel tersebut sehingga terbentuk tampilan sbb :



- Rubah tulisan jLabel1 menjadi **Nama Barang**, untuk merubah tulisan jLabel1 dapat digunakan salah satu dari keempat cara berikut :
 - Klik Ganda pada tulisan jLabel1, atau
 - Klik Kanan pada komponen jLabel1 di form atau di komponen Inspector, pilih **Edit Text**, atau
 - Rubah Properties **Text** dari komponen jLabel1

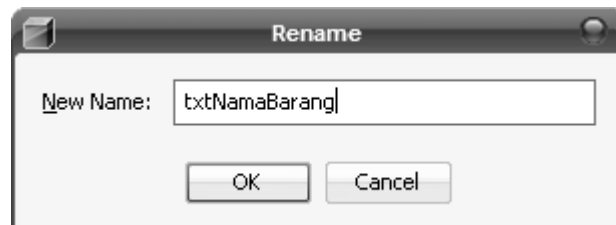


18. Untuk Penambahan **Komponen TextField** dalam form :

- a. Klik Class JTextField pada Swing Pallette
- b. Klik pada form untuk meletakkan Class JTextField tersebut sehingga terbentuk tampilan sbb:



- c. Kosongkan tulisan di dalam textfield dengan cara sama seperti penulisan text pada JLabel1 di atas.
- d. Rubah nama Variabel jTextField1 menjadi **txtNamaBarang**, ini dapat dilakukan dengan salah satu dari ketiga cara berikut:
 - i. Klik Kanan pada komponen jTextField1 di form, atau Komponen jTextField1 di komponen Inspector, kemudian pilih **Change Variable Name...** sehingga memunculkan :



- ii. atau bisa juga dengan merubah komponen **Variable Name** di Panel Code :



19. Untuk Komponen yang lain, lakukan dengan cara yang sama, sehingga terbentuk desain form seperti berikut :

20. Keterangan :

1	Class Text	: JLabel : Nama Barang
2	Class Text	: JLabel : Harga
3	Class Text	: JLabel : QTY
4	Class Text	: JLabel : Total
5	Class Variable Name Text Font HorizontalSize	: JLabel : lblTotal : TOTAL : : Verdana 14 Bold : Default
6	Class Variable Name	: JTextField : txtNamaBarang
7	Class Variable Name HorizontalAlignment	: JTextField : txtHarga : RIGHT
8	Class Variable Name HorizontalAlignment	: JTextField : txtQty : RIGHT
9	Class Variable Name HorizontalAlignment Enabled	: JTextField : txtTotal : RIGHT : false (tidak dicentang)
10	Class Variable Name Text Mnemonic	: JButton : btnHitung : Hitung : H
11	Class Variable Name Text Mnemonic	: JButton : btnKeluar : Keluar : K

Skenario :

Setelah diisi nama barang (misal: BAJU), harga (misal : 10000) dan Qty beli (misal : 25), kemudian ditekan tombol hitung akan memunculkan Jumlah (Harga x Qty) (misal : 250000) di txtTotal, sedangkan di lblTotal akan memunculkan tulisan : **TOTAL : Rp.**

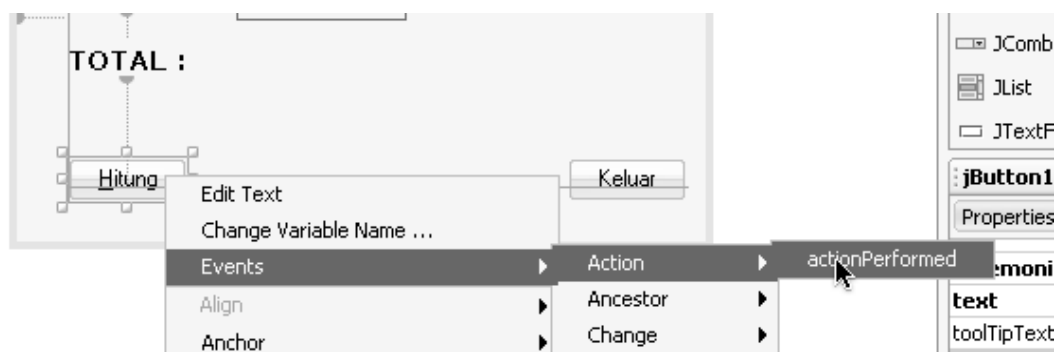
250,000. Kemudian apabila ditekan tombol Keluar maka akan memunculkan kotak Pesan dengan tulisan : **Terima Kasih telah membeli BAJU**, setelah diklik OK maka form hitung akan ditutup.

21. Untuk Melakukan Format penulisan angka dengan pemisah ribuan (**250,000**) dibutuhkan library dari java.text maka untuk menambahkan library tersebut, bukalah **Source dari fHitung.java**, kemudian ketikkan **import.java.*** di bawah package pmodul1 seperti di bawah ini :

```
package pmodul1;
import java.text.*;
```

22. Kemudian kembali ke tampilan Desain Form.

23. Agar Saat ditekan tombol hitung terjadi proses seperti skenario maka untuk menuliskan kode program, lakukan klik kanan pada tombol hitung sehingga memunculkan menu sbb :



24. Kemudian ketikkan kode program berikut ini :

```
private void btnHitungActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    float jumlah = Float.parseFloat(txtHarga.getText()) *
Float.parseFloat(txtQty.getText());
    txtTotal.setText(Float.toString(jumlah));
    DecimalFormat angka = new DecimalFormat("###,###");
    lblTotal.setText("TOTAL : Rp. "+angka.format(jumlah));
}
}
```

25. Agar pada saat keluar bisa memunculkan suatu kotak pesan, maka di bawah package pmodul1 berikan perintah :

```
import javax.swing.JOptionPane;
```

26. Lakukan Langkah no. 23 untuk mengisi kode pada tombol Keluar, kemudian ketikkan kode berikut ini :

```
private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
```

// TODO add your handling code here:

```
JOptionPane.showMessageDialog(this,
    "Terima Kasih Sudah Membeli "+txtNamaBarang.getText());
System.exit(0);
}
```

27. Catatan : Apabila sekedar untuk menutup form maka cukup menggunakan perintah :

System.exit(0)

28. Jalankan Program dan isikan nama barang, harga dan Qty, kemudian Klik Hitung (atau tekan CTRL+H), akan menghasilkan tampilan sbb :

The screenshot shows a Java Swing window titled "Form Hitung". It contains four text input fields with labels: "Nama Barang" (BAJU BEKAS), "Harga" (12000), "QTY" (10), and "Total" (120000.0). Below these fields, the text "TOTAL : Rp. 120,000" is displayed. At the bottom, there are two buttons: "Hitung" and "Keluar".

29. Kemudian Klik Tombol Keluar (atau tekan CTRL+K), akan menghasilkan tampilan sbb :

The screenshot shows the same "Form Hitung" window as before, but with a "Message" dialog box overlaid in the center. The dialog box has an information icon (i) and contains the text "Terima Kasih Sudah Membeli BAJU BEKAS". There is an "OK" button at the bottom of the dialog box. The "Form Hitung" window's buttons and input fields are still visible behind the dialog.

3. LATIHAN

1. Buat form dengan tampilan sbb :



Asal

Tujuan

Angka Awal

Hasil Kwadrat

Pindah

Kwadratkan

buat kode programnya dengan skenario sbb :

- Saat diklik **Pindah** maka isi tulisan yang terdapat di textfield Asal akan dipindah ke textfield tujuan
- Saat diklik **Kwadratkan** maka angka yang terdapat di textfield angka awal akan dikwadratkan dan hasil kwadrat tersebut akan ditampilkan di textfield Hasil kwadrat.

MODUL III

KOMPONEN PEMILIHAN 1

1. TUJUAN

- Mahasiswa mampu memahami cara penggunaan JRadioButton dan JCheckBox
- Menggunakan Class JRadioButton dan JCheckBox untuk melakukan pemilihan

2. TEORI DASAR

CheckBox adalah komponen yang biasa digunakan untuk memilih pilihan dari dua kemungkinan, biasanya berupa jawaban Ya atau Tidak.

RadioButton merupakan suatu kelompok pilihan, pilihan bisa dua atau lebih akan tetapi dalam satu kelompok tersebut hanya diperkenankan untuk memilih satu pilihan saja.

Dalam Swing, CheckBox bisa dibuat menggunakan Class JCheckBox, sedangkan untuk radio button dibutuhkan dua Class yaitu JRadioButton untuk pembuatan radiobutton dan ButtonGroup untuk mengelompokkan beberapa radiobutton ke dalam satu grup.

Contoh menggunakan komponen pemilihan

1. Jalankan Netbeans 5.5
2. Buat project Baru dengan nama **pmodul2**
3. Tambahkan Form (JFrame Form) dengan nama **fwarung**
4. Kemudian agar Form fwarung dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```
public static void main(String[] args) {
    // TODO code application logic here
    new fwarung().setVisible(true);
}
```

5. Kembali ke desain fwarung, tambahkan komponen sebagai berikut :

The screenshot shows a Java Swing window titled "MENU". It contains a table-like structure for a menu with two columns: "MENU" and "Quantity". The menu items are: Soto : Rp. 5000, Sate : Rp. 10000, Es Jeruk : Rp. 2000, and Es Teh : Rp. 1500. Each item has a checkbox and a corresponding quantity input field. Below the menu, there is a section for "Cara Bayar" (Payment Method) with three radio buttons: Cash, BCA Card, and Mandiri Card. To the right of the payment method, there is a "Hitung" (Calculate) button and a display showing "Rp. 0". At the bottom, there is a "Kosongkan" (Clear) button and a label "Cara Bayar".

Keterangan :

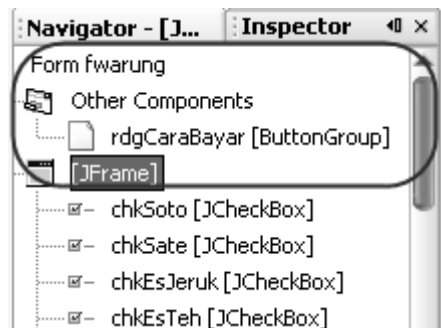
1	Class Text	: JLabel : MENU
2	Class Text	: JLabel : Quantity
3	Class Text Variable Name Mnemonic	: JCheckBox : Soto : Rp. 5000 : chkSoto : S
3	Class Text Variable Name Mnemonic	: JCheckBox : Sate : Rp. 10000 : chkSate : a
5	Class Text Variable Name Mnemonic	: JCheckBox : Es Jeruk : Rp. 2000 : chkEsJeruk : E
6	Class Text Variable Name Mnemonic	: JCheckBox : Es Teh : Rp. 1500 : chkEsTeh : T
7	Class Variable Name Editable Text	: JTextField : txtSoto : false : 0
8	Class Variable Name Editable Text	: JTextField : txtSate : false : 0
9	Class Variable Name Editable Text	: JTextField : txtEsJeruk : false : 0
10	Class Variable Name Editable Text	: JTextField : txtEsTeh : false : 0
11	Class	: JPanel
12	Class Text	: JLabel : Cara Bayar
13	Class Variable Name Text actionCommand selected	: JRadioButton : rdoCash : Cash : Uang Cash : true
14	Class Variable Name Text actionCommand selected	: JRadioButton : rdoBCA : BCA Card : Kartu Debit BCA : false
15	Class Variable Name Text actionCommand selected	: JRadioButton : rdoMandiri : Mandiri Card : Kartu Kredit Mandiri : false
16	Class Variable Name	: JLabel : lblCaraBayar

	Text	: Cara Bayar
17	Class Variable Name Text Mnemonic	: JButton : btnHitung : Hitung : H
18	Class Variable Name Text	: JLabel : lblTotal : Rp. 0
19	Class Variable Name Text Mnemonic	: JButton : btnKosong : Kosongkan : K
20	Class Variable Name	: ButtonGroup : rdgCaraBayar

6. catatan :

Untuk Pembuatan radio button langkahnya adalah :

- Tambahkan komponen no. 11 (JPanel)
 - Tambahkan Komponen no: 12, 13, 14 dan 15 dengan meletakkannya ke dalam komponen no 11 tadi
 - Tambahkan Komponen no. 20 (ButtonGroup) untuk menggabungkan ketiga radio button ke dalam satu grup
7. Untuk melakukan perubahan nama variabel komponen ButtonGroup dilakukan melalui **komponen Inspector**, pilih Klik kanan | Change Variable Name...



8. Kemudian buka source untuk **fwarung.java**, dan tambahkan kode berikut ini :

```
public fwarung() {
    initComponents();
    rdgCaraBayar.add(rdoCash);
    rdgCaraBayar.add(rdoBCA);
    rdgCaraBayar.add(rdoMandiri);
}
```

9. Skenario :

Pada Program Warung ini, apabila menu Soto diberi tanda Check (centang) maka txtSoto menjadi siap untuk diisi Quantity Pembeliannya, tetapi jika tanda Check dihilangkan maka txtSoto kembali menjadi kosong dan tidak bisa ditulisi/diedit. Hal yang sama juga berlaku bagi menu-menu yang lain. Untuk Cara Pembayaran bisa dipilih salah satu dari tiga pilihan yang ada. Apabila semua sudah diisi sesuai

kebutuhan, dan ditekan tombol hitung maka akan pada posisi lblTotal akan memunculkan Nilai Total dari Menu yang dipesan (masing-masing menu dihitung : harga x Quantity), dan di lblCaraBayar akan memunculkan tulisan : jika dipilih Cash akan tampil : Uang Cash, jika dipilih BCA Card akan tampil : Kartu Debit BCA, jika dipilih Mandiri Card akan tampil : Kartu Kredit Mandiri. Bila tombol Kosongkan diklik, maka semua isian dan tampilan akan kembali semula.

11. Sebagai awal dan contoh : Klik kanan pada chkSoto | Event | Action | ActionPerformed kemudian ketikkan kode berikut :

```
private void chkSotoActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    if (chkSoto.isSelected()) {
        txtSoto.setEditable(true);
        txtSoto.setText("1");
    } else {
        txtSoto.setEditable(false);
        txtSoto.setText("0");
    }
}
```

12. Lakukan hal yang sama untuk menu Sate, Es Jeruk dan Es Teh

13. Kemudian Pada btnHitung ketikkan kode program berikut ini :

```
private void btnHitungActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    boolean IDataOke = true;

    // cek Ada atau tidak :
    // menu dipesan tetapi tidak menuliskan Quantity
    // jika ada, maka variabel IDataOke menjadi false

    if (chkSoto.isSelected() && txtSoto.getText() == " ")
        IDataOke = false;
    if (chkSate.isSelected() && txtSate.getText() == " ")
        IDataOke = false;
    if (chkEsJeruk.isSelected() && txtEsJeruk.getText() == " ")
        IDataOke = false;
    if (chkEsTeh.isSelected() && txtEsTeh.getText() == " ")
        IDataOke = false;

    // jika variabel IDataOke == false, berarti masih ada kesalahan
    // sehingga perlu dimunculkan pesan kesalahan

    if (!IDataOke) {
        JOptionPane.showMessageDialog(this,
            "Data Dilengkapi dahulu");
    } else {
        // untuk memudahkan, buat variabel baru untuk menampung data
        int nSoto = Integer.parseInt(txtSoto.getText());
        int nSate = Integer.parseInt(txtSate.getText());
        int nEsJeruk = Integer.parseInt(txtEsJeruk.getText());
        int nEsTeh = Integer.parseInt(txtEsTeh.getText());
    }
}
```

```
// hitung total
int nTotal = (5000 * nSoto)+
(10000 * nSate)+(2000 * nEsJeruk)+(1500 * nEsTeh);
lblTotal.setText("Rp. "+nTotal);

// bagian ini untuk menampilkan cara Bayar
if (rdoCash.isSelected())
    lblCaraBayar.setText(rdoCash.getActionCommand());
if (rdoBCA.isSelected())
    lblCaraBayar.setText(rdoBCA.getActionCommand());
if (rdoMandiri.isSelected())
    lblCaraBayar.setText(rdoMandiri.getActionCommand());
}
}
```

14. Untuk btnKosongkan Ketikkan kode program berikut ini :

```
private void btnKosongActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    chkSoto.setSelected(false);
    chkSate.setSelected(false);
    chkEsJeruk.setSelected(false);
    chkEsTeh.setSelected(false);
    txtSoto.setText("0");
    txtSate.setText("0");
    txtEsJeruk.setText("0");
    txtEsTeh.setText("0");
    txtSoto.setEditable(false);
    txtSate.setEditable(false);
    txtEsJeruk.setEditable(false);
    txtEsTeh.setEditable(false);
    rdoCash.setSelected(true);
    lblTotal.setText("Rp. 0");
    lblCaraBayar.setText("Cara Bayar");
}
```

15. Apabila program dijalankan, kemudian dilakukan pemilihan menu serta cara bayar dan ditekan tombol hitung maka akan menghasilkan tampilan sbb :

MENU	Quantity
<input checked="" type="checkbox"/> Soto : Rp. 5000	2
<input checked="" type="checkbox"/> Sate : Rp. 10000	1
<input type="checkbox"/> Es Jeruk : Rp. 2000	0
<input type="checkbox"/> Es Teh : Rp. 1500	0

Cara Bayar

☐ Cash
☐ BCA Card
☒ Mandiri Card

Rp. 20000

Kartu Kredit Mandiri

Hitung Kosongkan

16. Kemudian saat ditekan tombol kosongkan maka tampilan menjadi sbb :

3. LATIHAN

Buat desain form sbb:

Skenario :

- Nama pegawai dan gaji pokok diisi biasa,
- Jika Seks dipilih Wanita, maka IblPasangan bertuliskan : Nama Suami, jika dipilih Pria maka IblPasangan bertuliskan : Nama Istri
- Jika Status Menikah diberi tanda check, maka txtJumAnak bisa diisi jumlah anaknya, tetapi jika tanda check dihilangkan maka txtJumAnak menjadi berisi angka 0 dan tak bisa dirubah/diedit
- Saat tombol Hitung Gaji Total diklik, maka akan dilakukan perhitungan gaji total dengan rumus :
 - o Untuk jumlah anak lebih dari atau sama dengan 3
 - o $Gaji\ Total = Gaji\ Pokok + (Gaji\ Pokok \times 10\% \times 3)$

- Untuk jumlah anak kurang dari 3
- $\text{Gaji Total} = \text{Gaji Pokok} + ((\text{gaji Pokok} \times 10\% \times \text{Jumlah Anak}))$
- Kemudian hasil perhitungan gaji total tersebut akan ditampilkan di textfield gaji total

MODUL IV

KOMPONEN PEMILIHAN 2

1. TUJUAN

- Mahasiswa mampu memahami cara penggunaan JList dan JComboBox
- Menggunakan Class JList dan JComboBox untuk melakukan pemilihan

2. TEORI DASAR

List dan ComboBox keduanya mempunyai fungsi yang hampir sama, yaitu memberi kesempatan bagi user untuk memilih dari suatu kumpulan daftar. List akan menampilkan daftar pilihan lebih dari 1 baris yang diletakkan dalam suatu kotak, sedangkan ComboBox pada dasarnya adalah suatu gabungan antara List dan TextBox (sehingga disebut Combo). Pilihan dalam comboBox hanya terlihat 1 baris, sedangkan daftar pilihan lainnya baru akan muncul bila dilakukan klik, biasanya di bagian kanan berbentuk segitiga terbalik, dari comboBox tersebut. Meskipun bisa juga dilakukan penambahan pengisian / dapat disunting, namun secara default biasanya comboBox bersifat non-editable. Perbedaan lainnya adalah, dalam List bisa diatur agar dapat dilakukan *Multiple Choice* (memilih lebih dari 1 pilihan dari daftar yang ada)

Dalam Swing, untuk membuat List dapat digunakan Class JList, sedangkan untuk ComboBox dapat digunakan Class JComboBox.

Membuat komponen pemilihan

1. Jalankan NetBeans 5.5
2. Buat project baru bernama **pmodul3**
3. Tambahkan Form (JFrame Form) dengan nama **fDaftar**
4. Kemudian agar Form fDaftar dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```
public static void main(String[] args) {
    // TODO code application logic here
    new fDaftar().setVisible(true);
}
```

5. Buat Desain Form sbb :

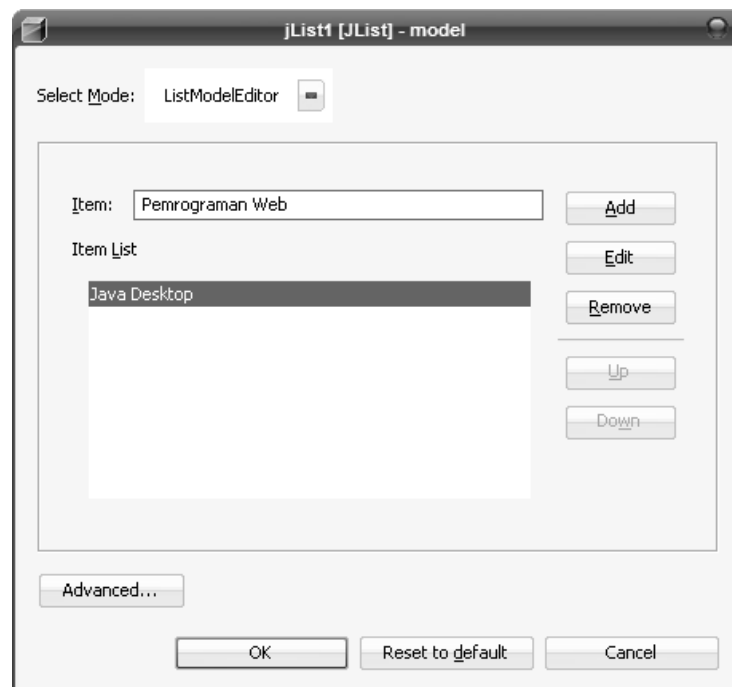


6. Keterangan :

1	Class Text	: JLabel : Daftar Mata Kuliah
2	Class Variable Name Model	: JList : IstKuliah : [...diisi..lihat petunjuk di bagian bawah keterangan..]
3	Class Variable Name Text	: JButton : btnCopy : >>
4	Class Text	: JLabel : Mata Kuliah Terpilih
5	Class Variable Name Model	: JList : IstPilih : [...Kosongkan..]
6	Class Text	: JLabel : Kuliah Yang Dipilih
7	Class Name editable	: JTextField : txtKuliah : false
8	Class Text	: JLabel : Banyak Mata Kuliah Terpilih
9	Class Name Editable Text	: JTextField : txtTerpilih : false : 0
10	Class Variable Name Text	: JButton : btnHapus : Hapus Terpilih
11	Class Text	: JLabel : Angkatan
12	Class Variable Name Model	: JComboBox : cboAngkatan : [...diisi : 2009, 2008, 2007, < 2007 ..]
13	Class Variable Name	: JButton : btnHitung

	Text	: Hitung Biaya
14	Class	: JTextArea
	Variable Name	: txtResume
	Editable	: false

7. Untuk Mengisikan Model List data pada JList dan comboBox, bisa dilakukan dengan memilih properties, kemudian memilih Model. Berikut ini contoh tampilan pengisian model untuk IstKuliah :



8. Skenario :

Setiap kali daftar mata kuliah diklik (dipilih) maka nama mata kuliah tersebut akan muncul di txtKuliah. Jika kemudian tombol Copy (>>) diklik, maka mata kuliah yang dipilih tadi akan dicopykan ke daftar mata kuliah terpilih, dan otomatis banyak mata kuliah terpilih (txtTerpilih) akan bertambah 1. Untuk menghapus mata kuliah terpilih dilakukan dengan cara: meng-klik mata kuliah yang akan dihapus, kemudian tekan tombol Hapus Terpilih, otomatis banyak mata kuliah terpilih akan berkurang 1. Untuk menghitung biaya, disederhanakan dengan rumus : Banyak Mata kuliah x Biaya per mata kuliah. Biaya per mata kuliah dihitung berdasar Angkatan dengan ketentuan : Angkatan 2009 → 100000, 2008 → 90000, 2007 → 75000, Kurang dari 2007 → 50000. Saat tombol hitung biaya diklik, selain melakukan perhitungan juga akan menampilkan resume di txtResume.

9. Dikarenakan untuk IstPilih, akan ditentukan sendiri List Model datanya maka dibutuhkan import untuk DefaultListModel, sebagai berikut :

```
package pmodul3;
import javax.swing.DefaultListModel;
```

10. Kemudian siapkan instance ListModel :

```
public class fDaftar extends javax.swing.JFrame {

    DefaultListModel pilih;

    /** Creates new form fDaftar */
    public fDaftar() {
        initComponents();
        //agar mudah di tambah dan hapus
        //lstPilih dibuatkan ListModel
        pilih = new DefaultListModel();
        lstPilih.setModel(pilih);
    }
}
```

10. Berikutnya, mengisikan kode pada lstKuliah pada events : ValueChanged, dengan cara Klik Kanan pada lstKuliah kemudian pilih: **Events | List Selection | ValueChanged**

11. Ketikkan kode berikut :

```
private void lstKuliahValueChanged(javax.swing.event.ListSelectionEvent evt) {
    // TODO add your handling code here:
    // setiap kali diklik, munculkan nama mata kuliah di txtKuliah
    int index = lstKuliah.getSelectedIndex();
    String kuliah = lstKuliah.getSelectedValue().toString();
    txtKuliah.setText(kuliah);
}
```

12. Berikut ini kode untuk tombol btnCopy :

```
private void btnCopyActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    // tambahkan tulisan di lstKuliah yang dipilih
    // ke dalam ListData Model pilih
    pilih.addElement(lstKuliah.getSelectedValue().toString());
    // setelah ditambah, hitung jumlah pilihan
    int size = pilih.getSize();
    txtTerpilih.setText(Integer.toString(size));
}
```

13. Berikut ini kode untuk tombol btnHapus

```
private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    int index = lstPilih.getSelectedIndex();
    //yang di-remove adalah listData model-nya
    pilih.remove(index);
    int size = pilih.getSize();
    txtTerpilih.setText(Integer.toString(size));
}
```

14. Kemudian kode untuk tombol btnHitung sbb :

```
private void btnHitungActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

    int nPilih = Integer.parseInt(txtTerpilih.getText());
    1 = cboAngkatan.getSelectedIndex();
    String isian = cboAngkatan.getSelectedItem().toString();

    // hitung Biaya
    int nPerMK = 0;
    switch (i) {
    2 case 0:nPerMK = 100000;break;
        case 1:nPerMK = 90000;break;
        case 2:nPerMK = 75000;break;
        default : nPerMK = 50000;
    }
    int nBiaya = nPerMK * nPilih;

    // Isikan Resume Ke textArea txtResume
    String sResume = "Anda Angkatan : "+isian+"\n"+
        "Banyak Mata Kuliah dipilih :"+
    3 Integer.toString(nPilih)+"\n"+
        "Biaya Per Mata Kuliah : "+
        Integer.toString(nPerMK)+"\n"+
        "TOTAL BIAYA : "+Integer.toString(nBiaya);
    txtResume.setText(sResume);
}
```

15. Catatan :

- a. Bagian 1: adalah pembuatan variabel untuk mengakses tiap object yang akan dipakai untuk perhitungan.
- b. Bagian 2: Proses penentuan biaya dengan mempertimbangkan nilai i, yaitu index dari cboAngkatan yang dipilih (index 0 berarti 2009, 1 berarti 2008, dst).
- c. Bagian 3: membuat resume dari semua data yang telah dipilih dan diisikan, kemudian menampilkan ke dalam JTextArea (txtResume) dengan perintah txtResume.append()

16. Saat dijalankan (tekan F6) maka akan menghasilkan tampilan sbb :

The screenshot shows a Java Swing window titled "Daftar Mata Kuliah". It contains two main panels. The left panel, titled "Daftar Mata Kuliah", has a list box containing five items: "Java Desktop", "Pemrograman Web", "Bahasa Inggris", "Logika dan Algoritma", and "Organisasi Komputer". Below the list box is a text field labeled "Kuliah yang dipilih" and a dropdown menu labeled "Angkatan" with "2009" selected. A "Hitung Biaya" button is at the bottom. The right panel, titled "Mata Kuliah Terpilih", has an empty list box. Below it is a text field labeled "Banyak Mata Kuliah Terpilih" with "0" entered, and a "Hapus Terpilih" button. A large empty text area is at the bottom right.

17. Dan berikut ini setelah dilakukan pemilihan dan penekanan tombol Hitung Biaya :

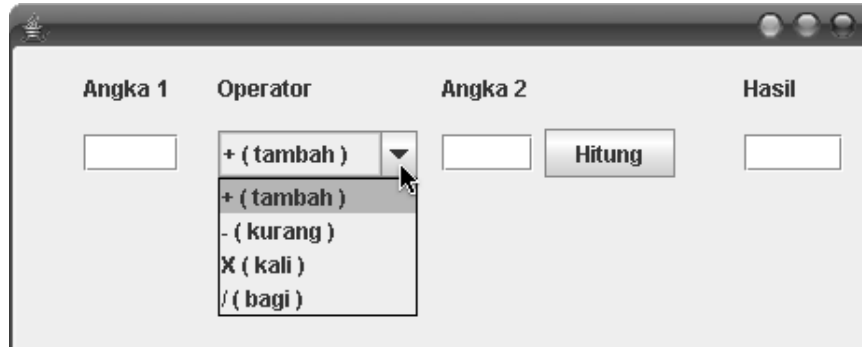
The screenshot shows the same application window after the "Hitung Biaya" button was clicked. The "Daftar Mata Kuliah" list box now has "Organisasi Komputer" selected. The "Kuliah yang dipilih" text field now contains "Organisasi Komputer". The "Angkatan" dropdown menu now shows "2008". The "Mata Kuliah Terpilih" list box now contains three items: "Bahasa Inggris", "Pemrograman Web", and "Organisasi Komputer". The "Banyak Mata Kuliah Terpilih" text field now shows "3". The "Hapus Terpilih" button is still present. The large text area at the bottom right now displays the following information: "Anda Angkatan : 2008", "Banyak Mata Kuliah dipilih : 3", "Biaya Per Mata Kuliah : 90000", and "TOTAL BIAYA : 270000".

3. LATIHAN

Buatlah aplikasi sederhana untuk melakukan pemilihan mata kuliah dengan menggunakan 2 list (mirip program di atas) dan 2 button yaitu button Pilih dan Button Hapus. Saat button Pilih diklik maka mata kuliah terpilih akan **dipindahkan (bukan dicopykan)** ke daftar mata kuliah terpilih, saat button hapus diklik maka terjadi sebaliknya.

E. TUGAS :

Buatlah program kalkulator dengan desain form sbb :



Saat diklik tombol hitung maka textbox hasil akan menampilkan hasil perhitungan sesuai operator yang dipilih

MODUL V

KOTAK DIALOG dan TABPANE

1. TUJUAN

- Mahasiswa mampu memahami pentingnya kotak dialog dalam pemrograman dan menggunakan TabPane untuk pengelompokan tampilan dalam pembuatan *user interface*
- Menggunakan TabPane untuk pengelompokan *user interface* dalam satu form
- Menggunakan kotak dialog dan berbagai variasinya untuk berbagai keperluan

2. TEORI DASAR

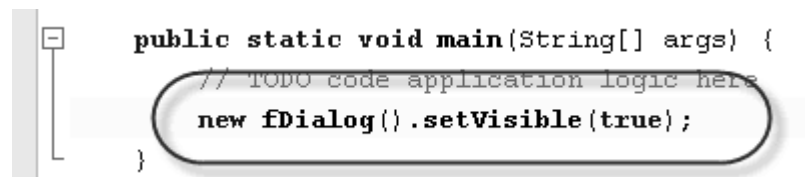
Kotak dialog cukup penting dalam aplikasi yang dikembangkan dalam dengan bahasa pemrograman java, sebab melalui kotak dialog ini aplikasi bisa berkomunikasi dengan pengguna. Kotak Dialog bersifat modal, artinya saat kotak dialog muncul maka akan semua jendela yang saat itu sedang terbuka akan menjadi *inactive* hingga nanti kotak dialog tersebut ditutup.

Pembuatan kotak dialog dalam swing mengacu pada Class `JOptionPane`, sehingga untuk mengaksesnya dibutuhkan statemen : `import javax.swing.JOptionPane`

Terdapat berbagai macam kotak dialog, akan tetapi yang akan dibahas pada praktikum ini ada 3 macam, yaitu Kotak dialog sederhana (hanya memunculkan tombol OK), kotak dialog konfirmasi (bisa memunculkan Yes, No dan Cancel) serta kotak dialog pilihan.

Membuat Dialog

1. Jalankan NetBeans 5.5
2. Buat project baru bernama **pmodul4**
3. Tambahkan Form (JFrame Form) dengan nama **fDialog**
4. Kemudian agar Form fDialog dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :



```
public static void main(String[] args) {
    // TODO code application logic here
    new fDialog().setVisible(true);
}
```

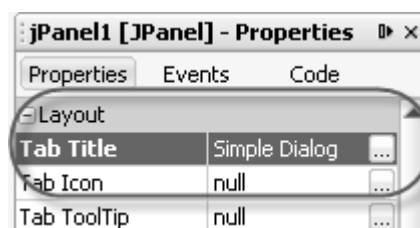
5. Buat Form dengan Komponen TabbedPane sebanyak 3 buah Tab, cara membuatnya :
6. Tambahkan Class **JTabbedPane** ke dalam form :



7. Kemudian di dalam TabbedPane tersebut tambahkan Class **Jpanel**



8. Kemudian, ganti judul Tab tersebut menjadi **Simple Dialog**, dengan merubah di properties:



9. Sehingga tampilan menjadi sbb:

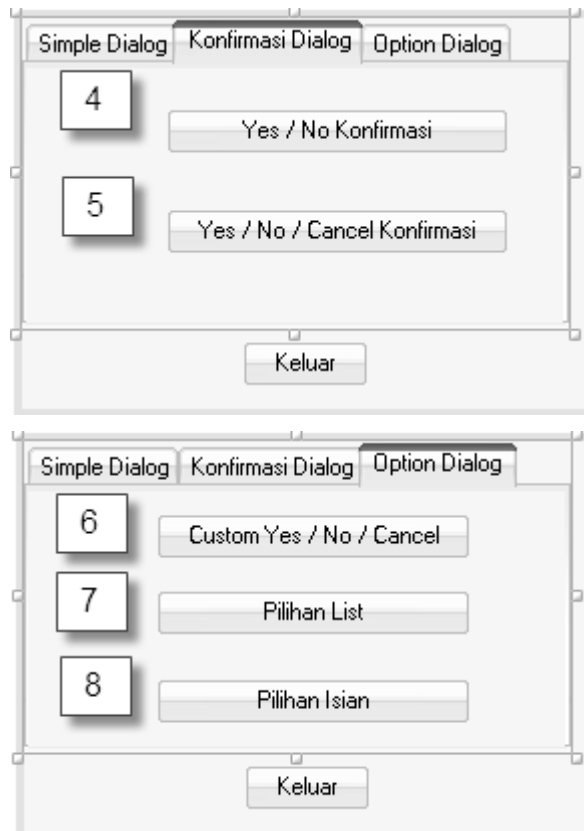


10. Tambahkan Class JPanel lagi ke dalam komponen JTabbedPane sehingga tampilan menjadi sbb :



11. Lengkapi hingga Tab berjumlah 4, kemudian tambahkan JButton sehingga tampilan untuk masing-masing Tab adalah sbb :





12. Keterangan :

Komponen 1 s/d 8 adalah JButton, variable name untuk tiap button adalah sbb :

1 → btnOK, 2 → btnError, 3 → btnKeluar, 4 → btnYN, 5 → btnYNC,
6 → btnCustom, 7 → btnPilihList, 8 → btnPilihIsi

13. Kemudian ketikkan kode program berikut untuk tiap-tiap button :

btnOK

```
private void btnOKActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JOptionPane.showMessageDialog(this,
        "Mudahnya Belajar Java");
}
```

btnError

```
private void btnErrorActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JOptionPane.showMessageDialog(this,
        "Belajar Java Harus Serius!!",
        "Peringatan",
        JOptionPane.ERROR_MESSAGE);
}
```

btnKeluar

```
private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    System.exit(0);
}
```

btnYN

```
private void btnYNActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    int jwb = JOptionPane.showConfirmDialog(this,
        "Belajar Java itu Mudah\nSetujukah Anda ?",
        "Konfirmasi",
        JOptionPane.YES_NO_OPTION);
    if (jwb == 0)
        JOptionPane.showMessageDialog(this,
            "Anda Memilih Yes");
    else
        JOptionPane.showMessageDialog(this,
            "NO adalah Pilihan Anda");
}
```

btnYNC

```
private void btnYNCActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    int jwb = JOptionPane.showConfirmDialog(this,
        "Akan Menghapus Data"+"nAnda Yakin ?",
        "Konfirmasi",
        JOptionPane.YES_NO_CANCEL_OPTION,
        JOptionPane.QUESTION_MESSAGE);
    switch (jwb){
        case 0: JOptionPane.showMessageDialog(this, "Jawab Yes");
            break;
        case 1: JOptionPane.showMessageDialog(this, "Jawab NO");
            break;
        case 2: JOptionPane.showMessageDialog(this, "Jawab Cancel");
            break;
        default :
            JOptionPane.showMessageDialog(this, "Koq Gak Jawab?");
    }
```

```

    }
}

```

btnCustom

```

private void btnCustomActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
Object[] options = {"Ya, Siip", "Tidak, Trims", "Krim Saja!"};
int jwb = JOptionPane.showOptionDialog(this,
    "Anda mau minum Kopi dengan Gula ?",
    "Pertanyaan",
    JOptionPane.YES_NO_CANCEL_OPTION,
    JOptionPane.QUESTION_MESSAGE,
    null,
    options,
    options[2]);

String respon;
switch (jwb){
    case 0 : respon = "Ini Kopi plus Gulanya";break;
    case 1 : respon = "OK, Kopi Saja ya";break;
    case 2 : respon = "Kopi dan Krim Siap";break;
    default : respon = "Koq gak Jawab Siy?";
}
JOptionPane.showMessageDialog(this,respon);
}

```

btnPilihList

```

private void btnPilihListActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
Object[] jawaban = {"Perut", "Kaki", "Hidung"};
String s = (String)JOptionPane.showInputDialog(this,
    "Lengkapi Kalimat Berikut:\n"+
    "\"Kepala,Pundak,Lutut, ...\"",
    "Custom Dialog",
    JOptionPane.PLAIN_MESSAGE,
    null,
    jawaban,
    "Perut");
}

```

```
//kalau s ada isinya,maka...
if ((s != null) && (s.length() > 0)) {
    JOptionPane.showMessageDialog(this,
        "Kepala,Pundak,Lutut, " + s + "!");
}
}
```

btnPilihIsi

```
private void btnPilihIsiActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
String s = (String)JOptionPane.showInputDialog(this,
    "Lengkapi Kalimat Berikut:\n"+
    "\"Kepala,Pundak,Lutut, ...\"",
    "Custom Dialog",
    JOptionPane.PLAIN_MESSAGE,
    null,
    null,
    "");

//kalau s ada isinya,maka...
if ((s != null) && (s.length() > 0)) {
    JOptionPane.showMessageDialog(this,
        "Kepala,Pundak,Lutut, " + s + "!");
}
}
```

14. Bila dijalankan maka akan muncul tampilan sbb :



15. Dan bila dipilih di tab Option Dialog, kemudian memilih Tombol Custom Yes/No/Cancel akan menghasilkan tampilan :



3. LATIHAN

Dengan memanfaatkan `showInputDialog` dan `showMessageDialog` buatlah aplikasi penjumlahan sederhana dengan tampilan sbb :



MODUL VII

PENGELOLAAN FILE TEXT

1. TUJUAN

- Mahasiswa mampu memahami teknik pengelolaan File Text
- Menggunakan Class JFileChooser dan JEditorPane untuk mengelola Dokumen berjenis Text

2. TEORI DASAR

Pemilih berkas menyediakan *user interface* berbasis grafis untuk mengelola sistem berkas (*file system*), dan kemudian user dapat memilih file atau direktori dari daftar atau memasukkan nama berkas atau direktori. Untuk menampilkan pemilih berkas dapat digunakan Class JFileChooser yang menampilkan dialog modal yang secara default memuat pemilih berkas.

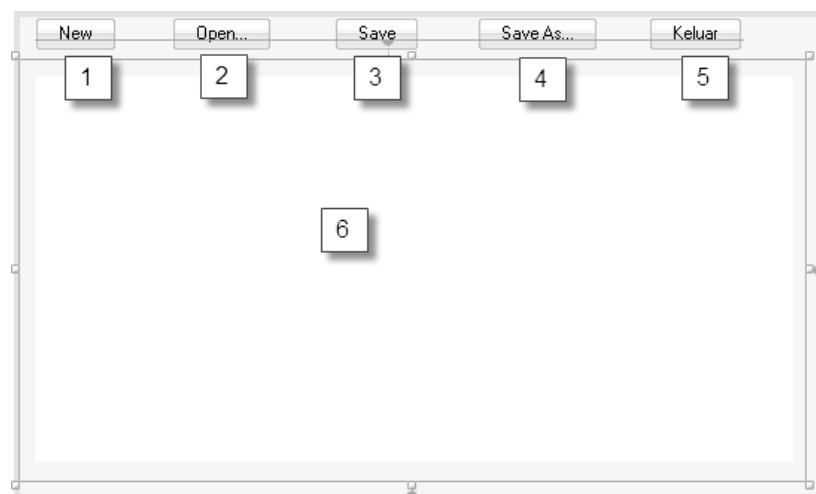
Membuat pengelolaan text

1. Jalankan NetBeans 5.5
2. Buat project baru bernama **pmodul5**
3. Tambahkan Form (JFrame Form) dengan nama **fNotes**
4. Kemudian agar Form fNotes dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```
public static void main(String[] args) {
    // TODO code application logic here

    new fNotes().setVisible(true);
}
```

5. Buatlah Desain Form sbb :



6. Keterangan :

1	Class Variable Name Text	: JButton : btnNew : New ...
2	Class Variable Name Text	: JButton : btnOpen : Open ...
3	Class Variable Name Text	: JButton : btnsave : Save ...
4	Class Variable Name Text	: JButton : btnSaveAs : Save As ...
5	Class Variable Name Text	: JButton : btnKeluar : Keluar
6	Class Variable Name Autoscroll	: JEditorPane : editor : true

7. Skenario :

Aplikasi ini bertindak seperti notepad sederhana, sekedar untuk membuka file, mengedit, dan menyimpannya. File yang bisa dibuka adalah yang memiliki extension : txt, java, csv.

Suatu file baru otomatis akan diberi judul : Untitled. Judul akan berubah menjadi nama file jika suatu file sudah disimpan atau sedang membuka file yang sudah tersimpan. Pada kondisi file baru, penekanan tombol Save (simpan saja) akan memicu proses Save As (simpan dengan nama), tetapi jika sudah pernah disimpan maka kedua tombol itu akan mempunyai fungsi yang berbeda. Saat penyimpanan, jika nama file sudah ada akan memunculkan pesan : “Apakah file yang sudah ada akan ditimpa? “, jika dijawab tidak maka penyimpanan batal.

8. Langkah yang dilakukan, pertama import class yang dibutuhkan :

```
package pmodul5;
import javax.swing.*;
import java.io.*;
import javax.swing.filechooser.FileFilter;
```

Pada bagian paling bawah dari source : fNotes.java, sesudah *variable declaration* buatlah method dan class baru sbb :

9. Method **msg**, untuk memudahkan dalam menampilkan pesan

```
private void msg(String pesan) {
    // untuk memunculkan pesan dengan tombol OK
    JOptionPane.showMessageDialog(this,pesan,"Informasi",
        JOptionPane.INFORMATION_MESSAGE);
}
```


10. Method **okToReplace**, untuk menampilkan pertanyaan File sudah ada, akan ditimpa?

```
private boolean okToReplace() {
    //Jika Pilihan mengembalikan 0 (Yes) berarti true
    return (JOptionPane.showConfirmDialog(this,
        "File "+file.getName()+" Sudah Ada\n"+
        "Akan Ditimpa ?",
        "Peringatan",JOptionPane.YES_NO_OPTION)==0) ;
}
```

11. Method **SimpanFile**, untuk melakukan penyimpanan File yang sedang dibuka

```
private void SimpanFile() {
    PrintWriter pw = null;
    try {
        pw = new PrintWriter(new BufferedWriter(new FileWriter(file)));
    } catch (IOException e) {
        msg("Gagal Simpan File : " + file.getName());
        return;
    }
    pw.print(editor.getText());
    pw.close();
    // Jika berhasil simpan, munculkan nama file
    // dan variabel fileBaru menjadi FALSE
    this.setTitle(file.getName()+" - NOTES" );
    fileBaru = false;
}
```

12. Class **jenisFile** , untuk melakukan filter file yang bisa dibuka

```
class jenisFile extends FileFilter {
    private String[] s;

    jenisFile(String[] sArg) {
        s = sArg;
    } // Tentukan file apa yang akan ditampilkan

    public boolean accept(File fArg) {
        if (fArg.isDirectory())
            return true;

        // Jika Extension dari file sesuai syarat, tampilkan
```

```

for (int i = 0; i < s.length; ++i)
    if (fArg.getName().toLowerCase().indexOf(s[i].toLowerCase()) > 0)
        return true; // selain kedua hal di atas, anggap false
    return false;
} public String getDescription() {
    String tmp = "";
    for (int i = 0; i < s.length; ++i)
        tmp += "*" + s[i] + " ";
    return tmp;
}
}

```

13. Untuk kelengkapan variabel dan seting lain ditunjukkan di bagian berikut ini (hati-hati, bagian berikut ini merupakan potongan program, untuk penentuan lokasi penulisan diberikan petunjuk tulisan yang dicetak tebal, jadi cari tulisan yang sama dengan yang bercetak tebal, kemudian tulis dibawahnya):

```

public class fNotes extends javax.swing.JFrame {
    // Siapkan variabel
    JFileChooser fc;
    File file;
    boolean fileBaru;
    final String[] EXT = { ".csv" };
    final String[] EXT2 = { ".txt", ".java" };
public fNotes() {
    initComponents();
    //Saat di awal aplikasi
    //Judul = Untitled
    this.setTitle("Untitled - NOTES");

    //Atur Fillter jenis file dengan Handle fc
    fc = new JFileChooser();
    fc.addChoosableFileFilter(new jenisFile(EXT));
    fc.addChoosableFileFilter(new jenisFile(EXT2
    fc.setAcceptAllFileFilterUsed(false);

```

```
//variabel fileBaru diinisialisai dengan TRUE
fileBaru = true;
}
```

Untuk Tiap tombol, kodenya ditunjukkan sebagai berikut :

14. Tombol : New...

```
private void btnNewActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    //Jika Klik Baru maka /editor dikosongkan, focus diarahkan ke editor
    //Judul diganti menjadi Untitled,fileBaru menjadi TRUE
    editor.setText("");
    editor.requestFocus();
    this.setTitle("Untitled - NOTES");
    fileBaru = true;
}
```

15. Tombol Save As...

```
private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    if (fc.showSaveDialog(this) == JFileChooser.APPROVE_OPTION){
        //Pilih yes
        file = fc.getSelectedFile();
        // Jika file belum ada, atau sudah ada
        // tetapi OK untuk ditimpa -> SimpanFile
        if (!file.exists() || okToReplace())
            SimpanFile();
    }
}
```

16. Tombol Save...

```
private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    if (fileBaru)
        //Jika fileBaru, Picu tombol Save As untuk event Klik;
        btnSaveAs.doClick();
    else
        // kalo bukan file baru, simpan langsung
        SimpanFile();
}
```

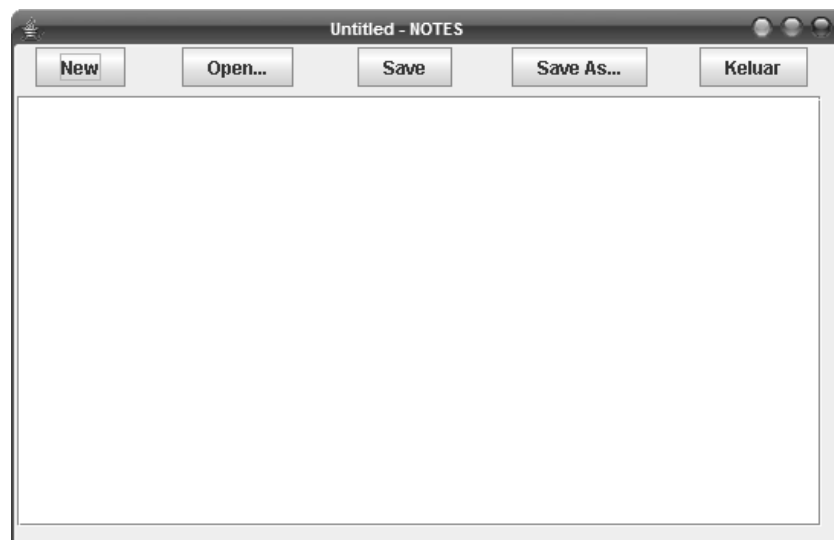
17. Tombol Open ...

```
private void btnOpenActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    int hasil = fc.showOpenDialog(this);
    if (hasil == JFileChooser.APPROVE_OPTION) {
        //Pilih Yes
        file = fc.getSelectedFile();
        try {
            editor.read(new FileInputStream(file), null);
        } catch (IOException e) {
            msg("Gagal Buka File : "+file.getName() );
            return ;
        }
    }
    // Jika berhasil buka, munculkan nama file
    // dan variabel fileBaru menjadi FALSE
    this.setTitle(file.getName()+" - NOTES" );
    fileBaru = false;
}
```

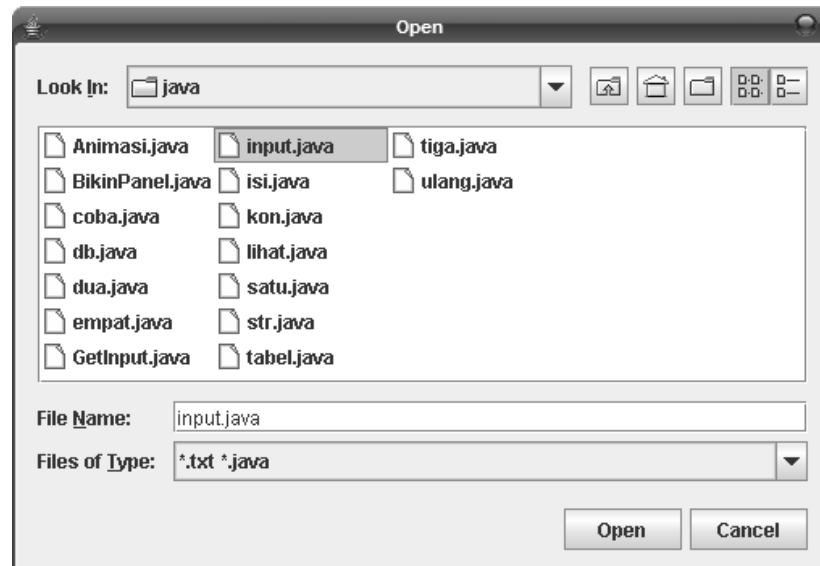
18. Tombol Keluar

```
private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    System.exit(0);
}
```

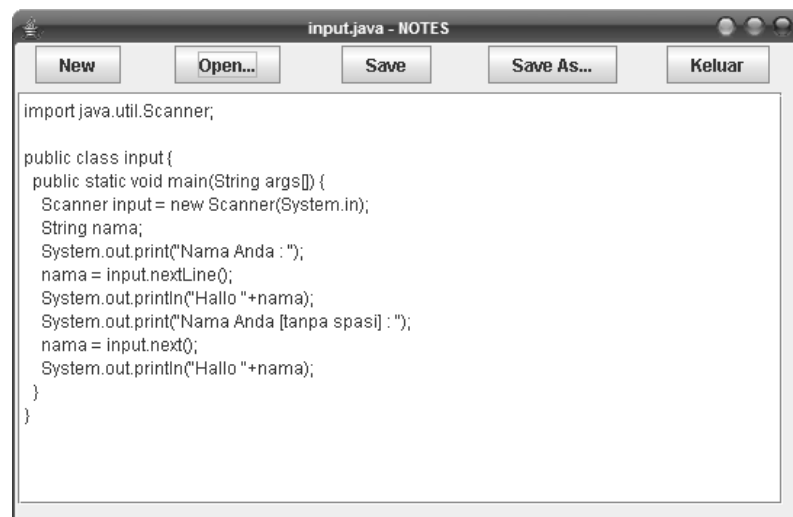
19. Setelah semua diketikkan dan aplikasi dijalankan (F6) maka akan memunculkan tampilan sbb :



20. Bila menekan tombol Open ... maka akan muncul tampilan :



21. Apabila sudah memilih salah satu file, kemudian menekan Open maka akan muncul tampilan :



3. LATIHAN

Modifikasilah program di atas agar bisa digunakan untuk membuka file : config.sys dan win.ini

MODUL VII

APLIKASI IMAGE VIEWER

1. TUJUAN

- Mahasiswa mampu memahami teknik pengaksesan dan menampilkan gambar
- Membuat Class sendiri untuk melakukan filter file
- Menggunakan Class JScrollPane dan JLabel untuk menampilkan gambar

2. TEORI DASAR

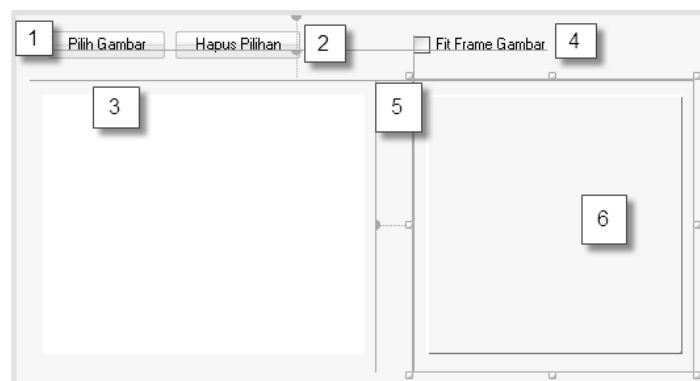
Gambar atau image sering digunakan dalam suatu aplikasi, baik bertujuan untuk memperjelas informasi atau juga sekedar untuk memperindah tampilan, akan tetapi beberapa aplikasi lain memang bertujuan sebagai pengelola image. Pada praktikum kali ini akan dibahas bagaimana menampilkan gambar ke dalam suatu form. Meskipun praktikum ini banyak menggunakan komponen SWING untuk membangunnya, namun untuk gambar masih dibutuhkan komponen AWT yaitu saat melakukan resize suatu ukuran gambar. Komponen SWING lebih disukai karena memiliki proses internal yang memudahkan seperti Media Tracker, dan juga dengan SWING akan mengurangi terjadinya *flicker* saat menampilkan gambar yang cukup besar.

Membuat aplikasi menampilkan gambar

1. Jalankan NetBeans 5.5
2. Buat project baru bernama **pmodul6**
3. Tambahkan Form (JFrame Form) dengan nama **fGambar**
4. Kemudian agar Form fGambar dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```
public static void main(String[] args) {
    // TODO code application logic here
    new fGambar().setVisible(true);
}
```

5. Buatlah Desain Form sbb :



6. Keterangan :

1	Class Variable Name Text	: JButton : btnPilih : Pilih Gambar ...
2	Class Variable Name Text	: JButton : btnHapus : Hapus Pilihan
3	Class Variable Name Model	: JList : lstDaftar : [... kosongkan ...]
4	Class Variable Name Text	: JCheckBox : chkFit : Fit Lebar Frame Gambar
5	Class Horizontal Size Vertical Size Horizontal Resizeable Vertical Resizeable	: JScrollPane : 200 : 200 : false : false
6	Class Variable Name Text	: JLabel : lblGambar : [... kosongkan...]

7. Skenario :

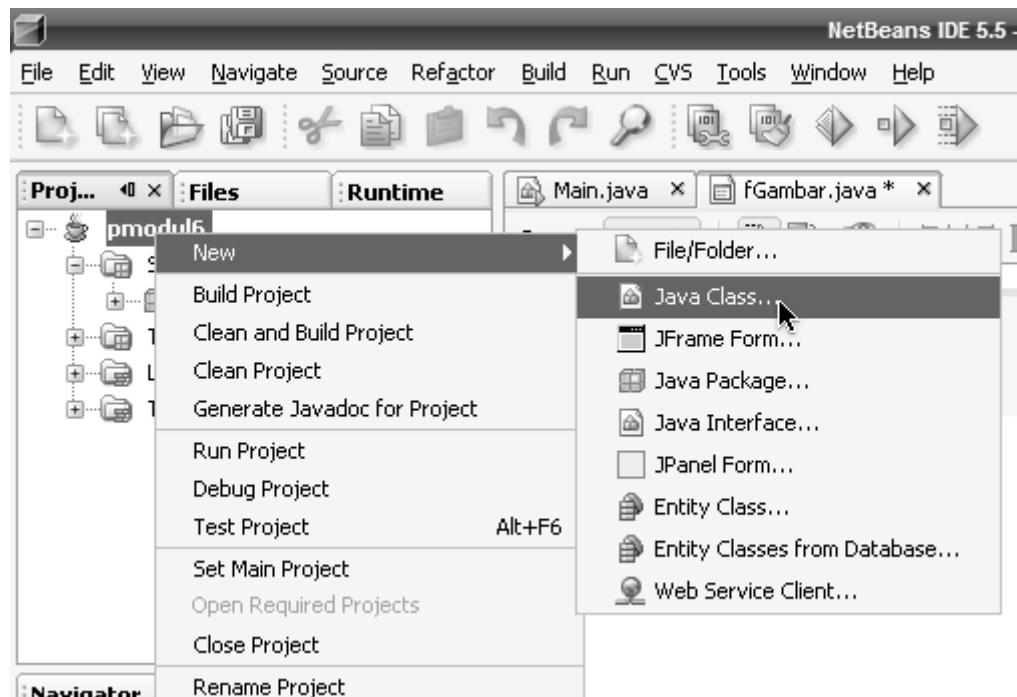
User dapat memilih gambar dengan menekan tombol pilih gambar, setiap gambar yang dipilih, path dan nama file gambarnya akan ditambahkan ke dalam list lstDaftar, untuk menghapus gambar yang ada di daftar tinggal di klik salah satu gambar dan tekan tombol Hapus Pilihan. Untuk menampilkan gambar dilakukan dengan meng-klik salah satu data gambar dari daftar, maka otomatis gambar akan ditampilkan menggunakan lblGambar. Ada dua kemungkinan yang terjadi : Jika status checkbox Fit Lebar Frame Gambar di centang maka Jika gambar terlalu lebar maka gambar akan dirubah ukuran lebarnya menjadi 200, sedangkan tingginya akan menyesuaikan., tetapi jika checkbox tersebut tidak dicentang maka gambar akan ditampilkan apa adanya dan otomatis di sekitar gambar akan muncul scrollbar untuk melihat bagian gambar yang tertutup.

8. Agar saat gambar yang terlalu besar bisa otomatis muncul scrollbar maka langkah pembuatan desain form di atas untuk komponen no 5 dan 6 adalah sbb :

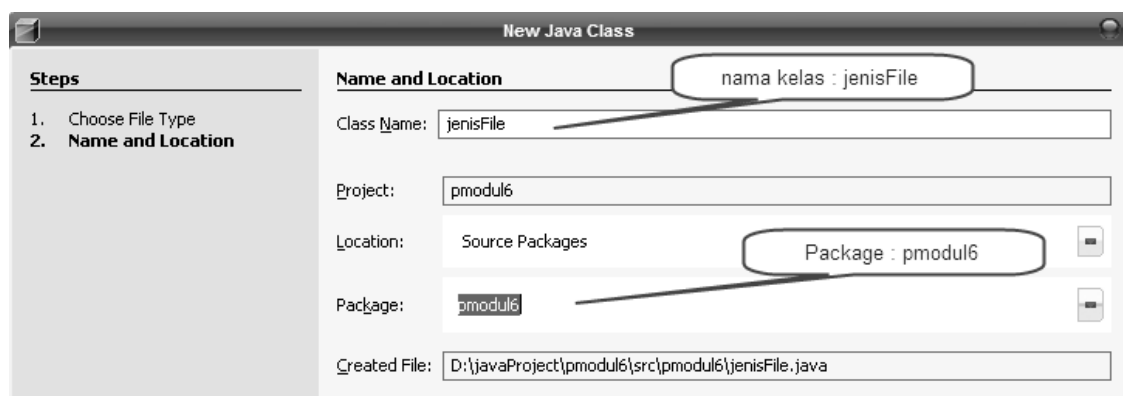
- Tambahkan Class JScrollPane ke dalam form, atur properties seperti keterangan di atas
- Tambahkan Class JLabel ke dalam JScrollPane yang ditambahkan tadi

9. Untuk membuat Class baru yang akan digunakan sebagai filter file langkahnya sbb :

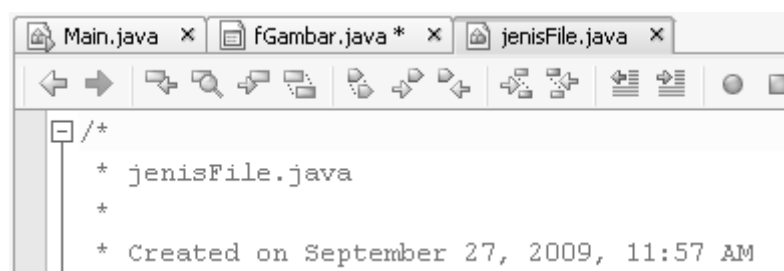
10. Klik Kanan pada project : **pmodul6**



11. Akan memunculkan tampilan sbb :



12. setelah diisi dan diklik Finish, maka akan terbentuk Class baru bernama jenisFile.java



13. Ketikkan kode program berikut ini (anda bisa lihat juga kode program ini di modul 5, tetapi dengan beberapa modifikasi disesuaikan dengan posisinya sebagai eksternal class), catatan : yang dicetak tebal sudah dibuat otomatis oleh Netbeans, sehingga ketikkan saja yang tidak tercetak tebal :

```

package pmodul6;
import javax.swing.filechooser.FileFilter;
import java.io.*;
public class jenisFile extends FileFilter {
  
```



```

private String[] s;

/** Creates a new instance of jenisFile */
public jenisFile(String[] sArg) {
    s = sArg;
}

// Tentukan file apa yang akan ditampilkan
public boolean accept(File fArg) {
    if (fArg.isDirectory())
        return true;

    // Jika Extension dari file sesuai syarat, tampilkan
    for (int i = 0; i < s.length; ++i)
        if (fArg.getName().toLowerCase().indexOf(s[i].toLowerCase()) > 0)
            return true;

    // selain kedua hal di atas, anggap false
    return false;
}

public String getDescription() {
    String tmp = "";
    for (int i = 0; i < s.length; ++i)
        tmp += "*" + s[i] + " ";
    return tmp;
}
}

```

14. Simpan Class tersebut di atas, kemudian kembali buka form fGambar, dan masuk ke source
15. Kode-kode program berikut ini dituliskan di class fGambar.java
16. Lakukan import class yang diperlukan :

```

package pmodul6;

import java.awt.Image;
import javax.swing.*.*;
import java.io.*;
import javax.swing.filechooser.FileFilter;

```

17. Pada bagian deklarasi awal :

```
public class fGambar extends javax.swing.JFrame {

    DefaultListModel gambar;
    // buat konstanta untuk filter file
    final String[] EXT= {".jpg", ".jpeg", ".gif", ".tiff", ".png"};

    /** Creates new form fGambar */
    public fGambar() {
        initComponents();
        gambar = new DefaultListModel();
        lstDaftar.setModel(gambar);
    }
}
```

18. Kode Program Pada btnPilih

```
private void btnPilihActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JFileChooser fc = new JFileChooser();
    //Atur Filter Menggunakan Class jenisFile
    //aturan mengikuti konstanta EXT
    fc.addChoosableFileFilter(new jenisFile(EXT));
    int hasil = fc.showOpenDialog(this);
    if (hasil == JFileChooser.APPROVE_OPTION) {
        //Pilih Yes
        File file = fc.getSelectedFile();
        // Ambil nama file beserta pathnya
        String nf = file.getAbsolutePath();
        // masukkan ke listData Model gambar
        gambar.addElement(nf);
    }
}
```

19. Kode Program pada btnHapus

```
private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int index = lstDaftar.getSelectedIndex();
    //yang di-remove adalah listData model-nya
    gambar.remove(index);
}
```

```

lblGambar.setIcon(null);
}

```

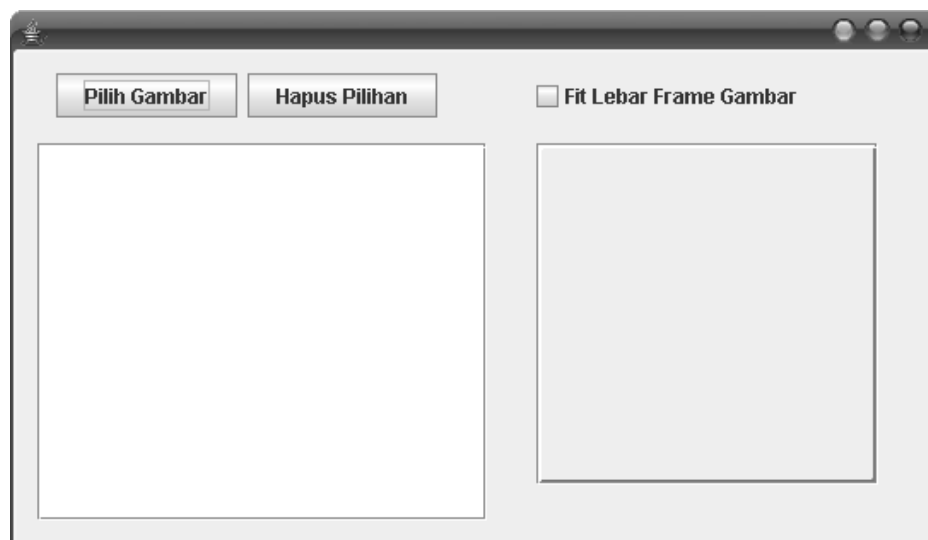
20. Kode program pada IstDaftar, perhatikan bahwa untuk IstDaftar ini yang digunakan adalah event saat list mendapatkan Klik mouse, sehingga untuk mengisi kodenya lakukan Klik Kanan pada komponen List IstDaftar, kemudian pilih Events | Mouse | mouseClicked

```

private void IstDaftarMouseClicked(java.awt.event.MouseEvent evt) {
// TODO add your handling code here:
//Ambil nama file gambarnya
String nf = IstDaftar.getSelectedValue().toString();
// Jadikan Icon
Imagelcon gbr = new Imagelcon(nf);
// Jika CheckBox Fit ke Frame gambar di Check
if (chkFit.isSelected())
// buat gambar mempunyai lebar 200,
// dengan tinggi menyesuaikan
// menggunakan algoritma default
gbr = new Imagelcon(gbr.getImage().getScaledInstance(200,
-1, Image.SCALE_DEFAULT));
//tampilkan gambar
lblGambar.setIcon(gbr);
}

```

21. Bila sudah selesai, jalankan program (F6), sehingga akan memunculkan tampilan sbb :



22. Berikut ini tampilan saat gambar terlalu besar dengan kondisi Fit tidak diberi tanda check :

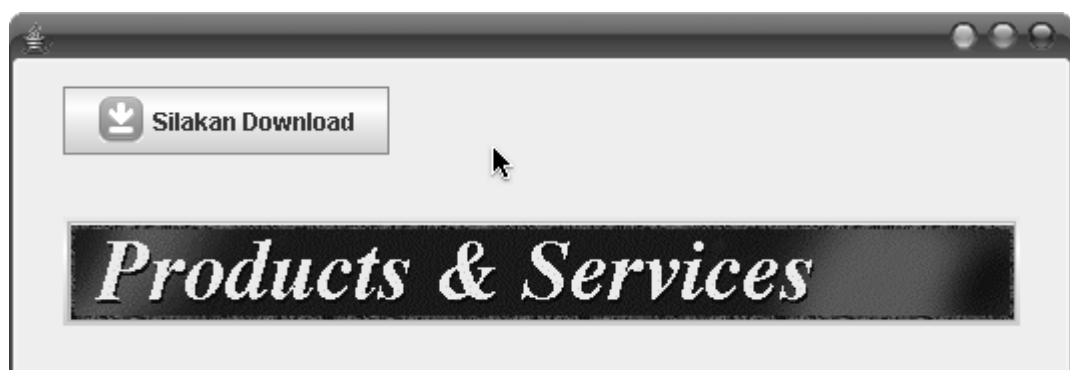


23. Berikut in tampilan gambar yang sama tetapi Kondisi Fit sudah diberi tanda check

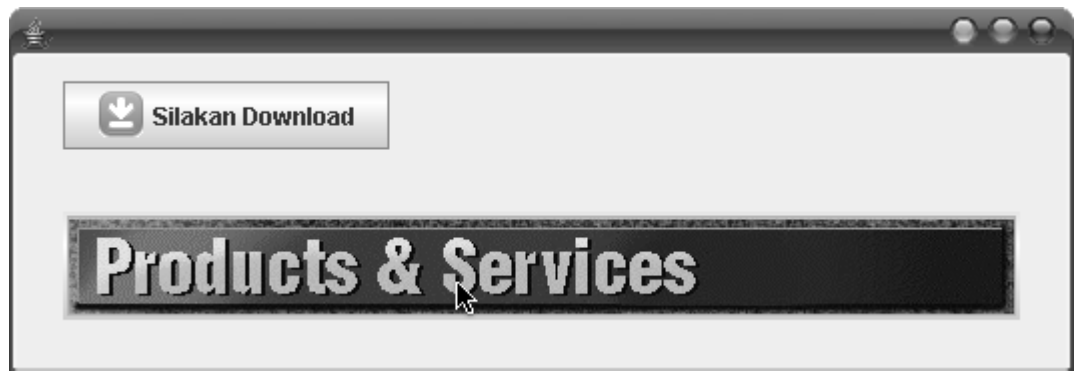


3. LATIHAN

Buatlah program sederhana yang berisi 2 buah Button, keduanya memiliki gambar (gambar bebas) :



Tetapi pada saat mouse menyentuh Button yang kedua (product & Service) maka tampilan menjadi :



MODUL VIII

MENU DAN *MULTI FORM*

1. TUJUAN

- Mahasiswa mampu memahami teknik pembuatan menu dan multi form
- Menjelaskan Modal Form dan Modeless Form
- Membuat Menu Pulldown menggunakan Class JMenu
- Menggunakan JDialog dalam pembuatan Form

2. TEORI DASAR

Menu dalam suatu aplikasi memegang peranan yang sangat penting. Menu adalah suatu media untuk melakukan pemilihan atau menjalankan suatu proses dalam aplikasi. Pilihan diberikan dari sebuah menu dapat dipilih oleh operator oleh sejumlah metode (disebut interface):

- menekan satu atau lebih tombol pada keyboard atau mouse
- menggunakan elektromekanis pointer, seperti pena cahaya
- menyentuh pada tampilan layar dengan jari
- berbicara dengan sistem pengenalan suara

Sebuah komputer menggunakan antarmuka pengguna grafis menyajikan menu dengan kombinasi teks dan simbol untuk mewakili pilihan, dengan mengklik pada salah satu simbol, para operator memilih instruksi yang mewakili simbol tersebut.

Penggunaan umum menu adalah untuk menyediakan akses mudah ke berbagai operasi seperti menyimpan atau membuka file, Keluar dari program, atau memanipulasi data. Kebanyakan program aplikasi menyediakan beberapa bentuk pull-down. Menu pull-down adalah jenis yang biasa digunakan dalam menu bar (biasanya di dekat bagian atas jendela atau layar), yang paling sering digunakan untuk melakukan tindakan. Menurut kebiasaan manusia pada umumnya, nama menu seharusnya berbentuk verbal, seperti "file", "edit" dan seterusnya.

Membuat menu

10. Jalankan NetBeans 5.5
11. Buat project baru bernama **pmodul7**
12. Tambahkan Form (JFrame Form) dengan nama **fMenuUtama**
13. Kemudian agar Form fMenuUtama dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```

public static void main(String[] args) {
    // TODO code application logic here

    new fMenuUtama().setVisible(true);
}

```

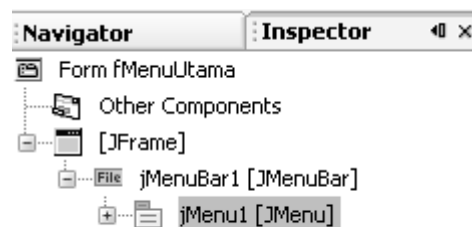
14. Buatlah Desain Form sbb :



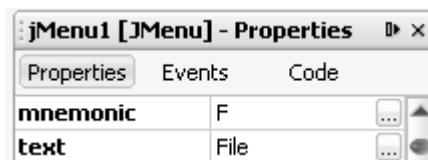
15. Keterangan :

1	Class Variable Name	: JMenuBar : JMenuBar1
2	Class Variable Name Text	: JMenu (sub dari JMenuBar, jadi otomatis akan muncul saat pilih JMenuBar) : JMenu1 : Menu

16. Pada Inspector akan terlihat sbb :



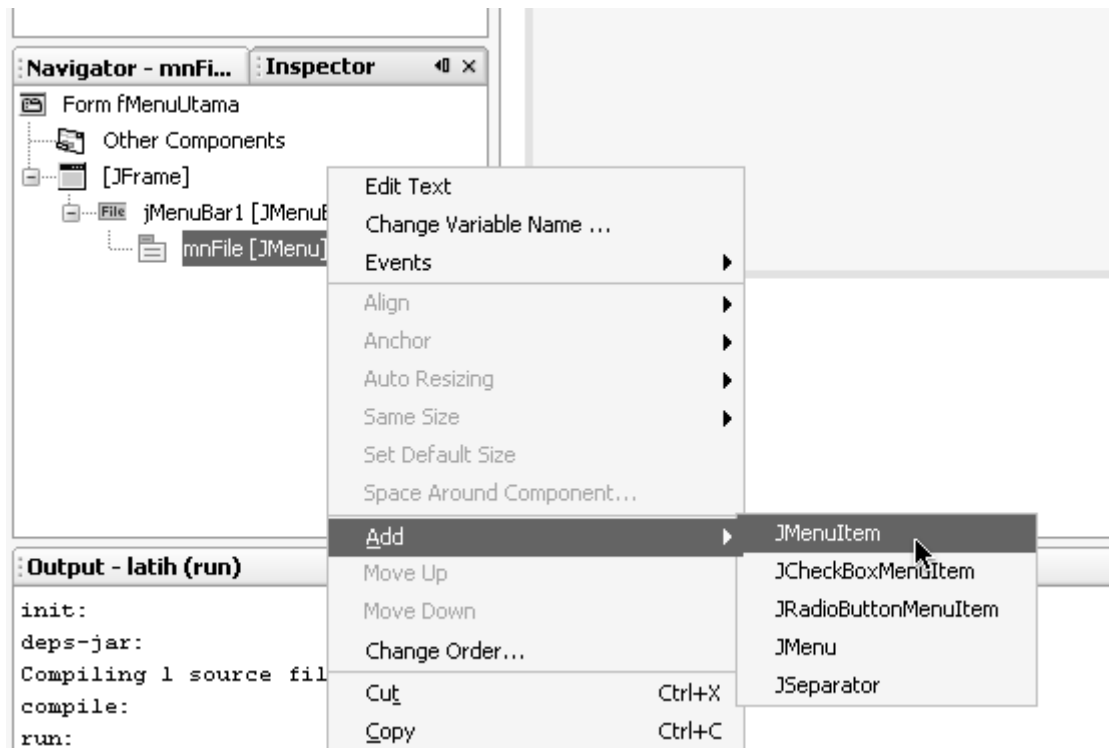
17. Pilih / Klik **jMenu1** kemudian pada properties rubahlah sebagai berikut :



18. Sedangkan Nama Variabelnya dirubah menjadi mnFile sbb :

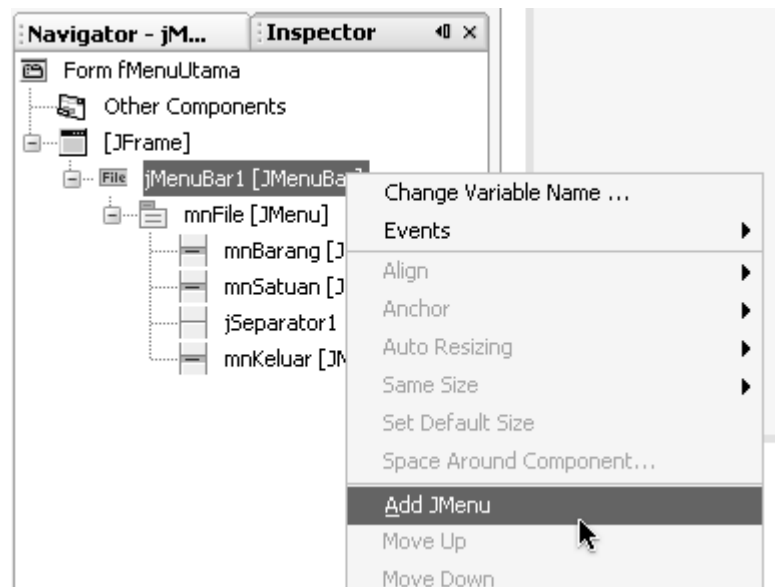


19. Kemudian tambahkan Menu Item pada mnFile sbb :



20. Tambahkan Menu Item (JMenuItem): **Barang (mnBarang)**, **Satuan (mnSatuan)**, **Separator** (dari JSeparator) dan **Keluar (mnKeluar)**

21. Kemudian tambahkan pada Menu (JMenu) pada **jMenuBar1** sbb :

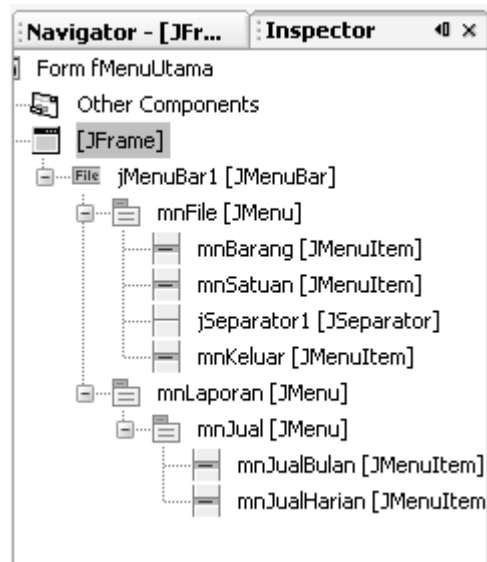


22. Tambahkan Menu Laporan (JMenu) beri nama **mnLaporan**

23. Tambahkan Menu baru (JMenu) di **mnLaporan**, yaitu **Penjualan (mnJual)**

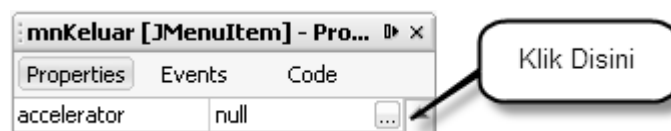
24. Tambahkan Menu Item Baru (JMenuItem) di **mnJual** yaitu : **Omset Bulanan (mnJualBulan)** dan **Omset Harian (mnJualHarian)**

25. Sehingga akhirnya akan terbentuk sbb :



26. Untuk menambahkan Accelerator Key (ShortCut) pada menu keluar, lakukan langkah berikut :

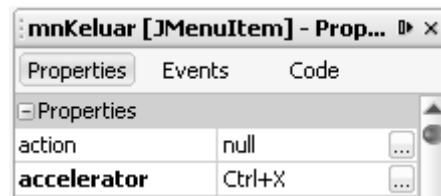
27. Pilih / Klik **mnKeluar** , kemudian pada properties pilih Accelerator



28. Sehingga muncul :



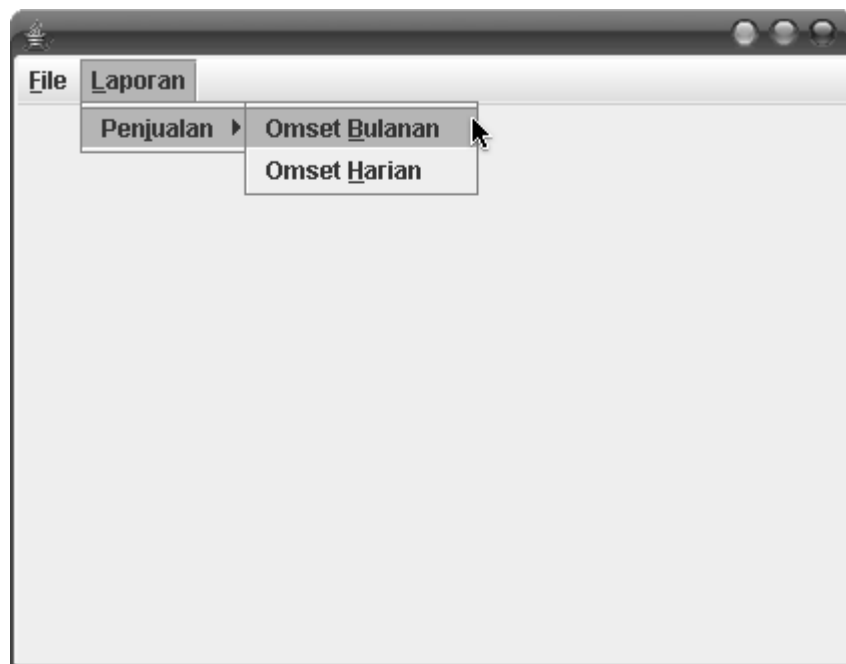
29. Klik OK, maka pada properties mnKeluar akan terbentuk :



30. Jalankan Form dengan menekan F6, maka akan muncul :

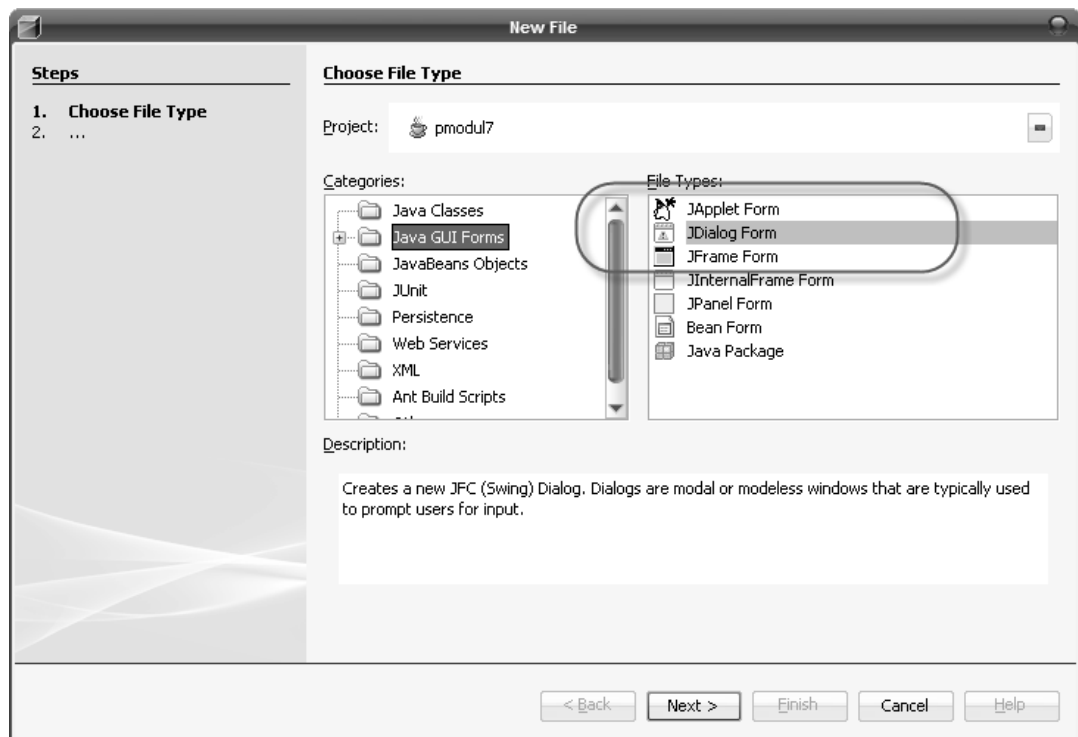


31. Sedangkan untuk menu Laporan :

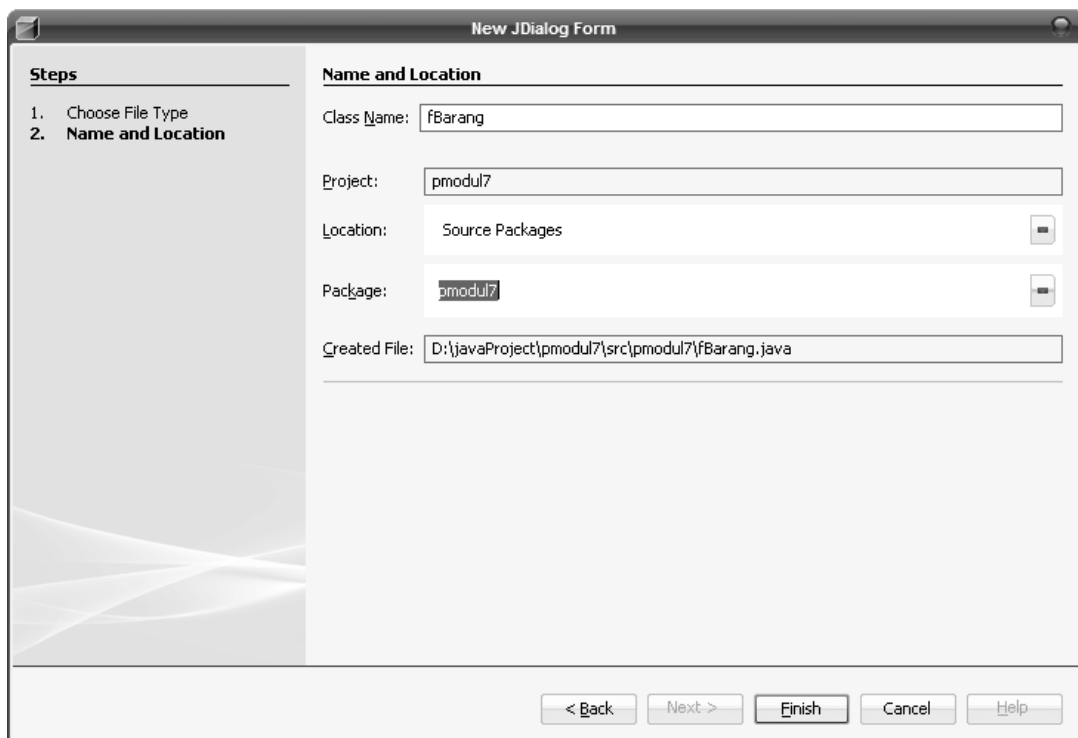


32. Berikutnya akan dibuat form barang, yang kemudian form tersebut akan dipanggil menggunakan menu File | Barang.

33. Buat File Baru, Pilih JDialog Form

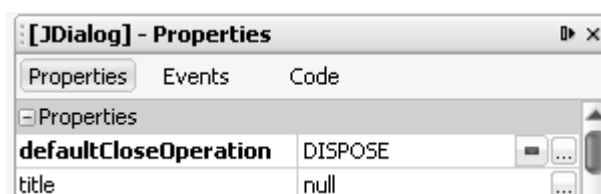


34. Klik Next

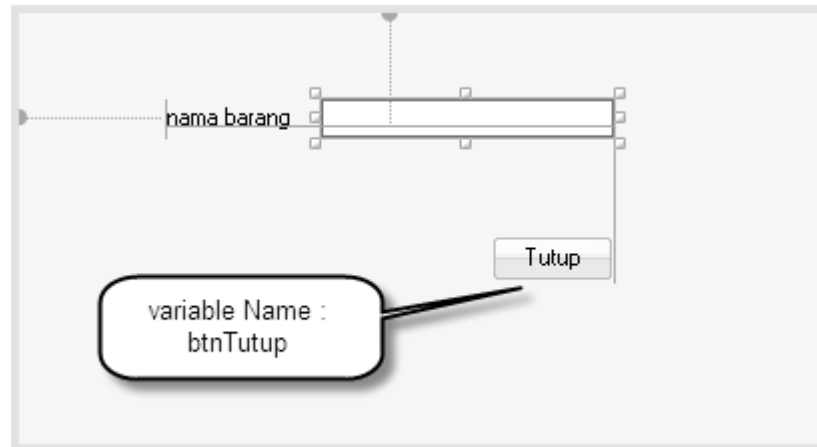


35. Class Name diisikan : **fBarang**, kemudian klik Finish

36. Akan Terbentuk Form baru, lihat dibagian Properties, jika **defaultCloseOperation** belum berisi **DISPOSE**, rubahlah seperti gambar berikut :



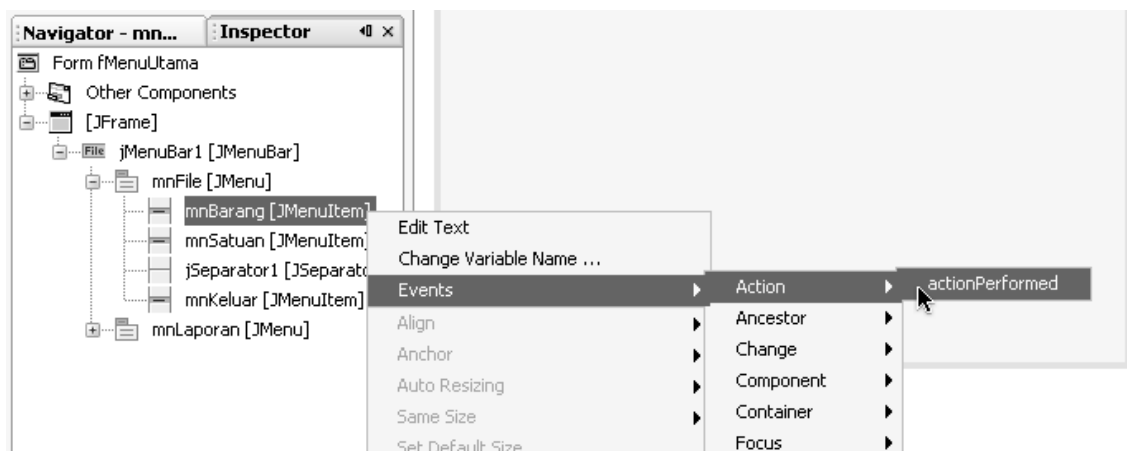
37. Buatlah Desain Form **fBarang** seperti berikut ini ;



38. Pada **btnTutup**, tambahkan action sbb :

```
private void btnTutupActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}
```

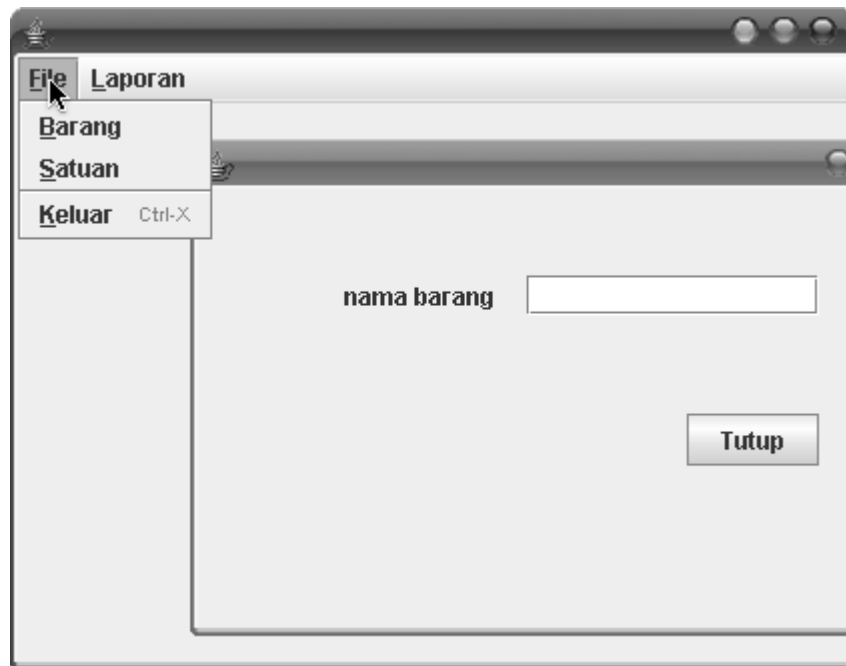
39. Kemudian untuk Menu Barang diisi Actionnya sbb :



40. Kemudian ketikkan sebagai berikut :

```
private void mnBarangActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new fBarang(this,false).setVisible(true);
}
```

41. Perintah di atas jika dijalankan akan mengakibatkan form barang dibuka secara modelless



42. Kemudian Rubahlah action untuk menu barang menjadi :

```
private void mnBarangActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    new fBarang(this,true).setVisible(true);
}
```

43. Maka hasilnya Form Barang akan dibuka secara modal



44. Untuk Menu Keluar, isikan action sbb :

```
private void mnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
```

```
// TODO add your handling code here:
```

```
    System.exit(0);  
}
```

3. LATIHAN

1. Tambahkan di menu Bar : **Help**
2. Kemudian di dalam **menu Help**, tambahkan menu item : **About**,
3. Buatlah Form baru (JDialog) bernama **fAbout** yang berisi, nama program dan copyrightnya
4. Hubungkan menu About dalam Help dengan form fAbout agar saat menu About di klik, form fAbout akan dibuka secara Modal

MODUL IX

APLIKASI SEDERHANA

1. TUJUAN

- Mahasiswa mampu membuat aplikasi sederhana berdasar materi yang telah dipelajari sebelumnya
- Membuat aplikasi sederhana dengan menu pulldown
- Melakukan validasi pengakses aplikasi berbasis user dan password

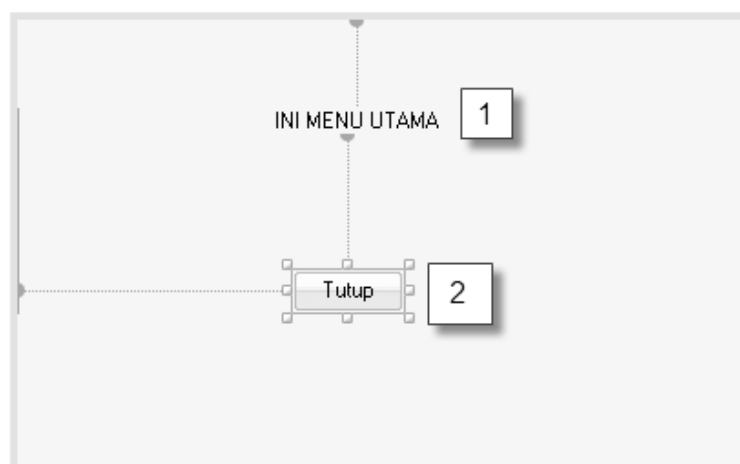
2. TEORI DASAR

Program Aplikasi dibuat dengan cara memudahkan user dalam menjalankan setiap form / fitur yang ada dalam aplikasi tersebut, tetapi sering terjadi bahwa tidak semua user diperbolehkan untuk membuka semua fitur yang ada dalam suatu aplikasi. Pengaturan tentang boleh tidaknya user untuk mengakses suatu fitur atau bagian dari aplikasi disebut dengan pengaturan hak akses user.

Pada praktikum kali ini akan dbuat suatu pengaturan hak akses yang paling dasar yaitu dengan memberikan nama user dan password standar untuk masuk ke dalam aplikasi.

Membuat aplikasi Sederhana

45. Jalankan NetBeans 5.5
46. Buat project baru bernama **pmodul8**
47. Tambahkan Form (JFrame Form) dengan nama **fUtama**
48. Buatlah Desain Form sbb :



Keterangan :

1	Class Variable Name Text	: JLabel : JLabel1 : INI MENU UTAMA
2	Class	: JButton

	Variable Name Text	: btnTutup : Tutup
--	-----------------------	-----------------------

6. Kemudian pada actionPerformed btnTutup ketikkan kode sbb :

```
private void btnTutupActionPerformed(java.awt.event.ActionEvent
evt) {

    // TODO add your handling code here:

    System.exit(0);

}
```

7. Tambahkan Lagi sebuah Form (JFrame Form) dengan nama **fPass**

8. Buatlah Desain Form sbb :

9. Keterangan :

1	Class Variable Name Text	: JLabel : JLabel1 : User
2	Class Variable Name Text	: JLabel : JLabel2 : Password
3	Class Variable Name Text	: JTextField : user :
4	Class Variable Name Text Echo Char	: JPasswordField : pass : : #
5	Class Variable Name Text	: JButton : btnKeluar : Keluar
6	Class Variable Name Text	: JButton : btnLogin : Login

10. Kemudian pada actionPerformed btnKeluar ketikkan kode sbb :

```
private void btnKeluarActionPerformed(java.awt.event.ActionEvent
evt) {

    // TODO add your handling code here:
```



```
System.exit(0);
```

```
}
```

11. Kemudian pada actionPerformed btnLogin ketikkan kode sbb :

```
private void btnLoginActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:

    String cUserBenar = "stmik";
    String cPassBenar = "akakom";
    String cUser = user.getText();
    String cPass = String.valueOf(pass.getPassword());

    if ((cUser.compareTo(cUserBenar)==0) &&
        (cPass.compareTo(cPassBenar)==0)) {

        new futura().setVisible(true);
        dispose();
    } else {
        user.setText("");
        pass.setText("");
    }
}
```

12. Kemudian agar Form fMenuUtama dijalankan pertama kali bila program di –Run maka pada program **Main.java** tambahkan kode berikut :

```
public static void main(String[] args) {
    // TODO code application logic here

    new fPass().setVisible(true);
}
```

13. Jalankan Form dengan menekan F6

14. Isikan user dan password sembarang, amati apa yang terjadi!

15. Isikan user : stmik dan password : akakom, amati apa yang terjadi!

3. LATIHAN

Buatlah suatu aplikasi geometri dengan menggunakan menu pulldown. Isi menu adalah :

- Menu: Hitung , yang berisi menu item: hitung luas Persegi Panjang dan Hitung Luas Segitiga
- Menu: Utility, yang berisi menu item : Keluar
- Saat di klik Keluar maka akan muncul tombol konfirmasi : Yakin akan keluar dengan pilihan Yes dan No, jika di pilih Yes, akan keluar dari aplkasi, Jika dipilih No, akankembali ke aplikasi.