



PBO LANJUT USING WYSIWYG EDITOR FOR BUILDING GUI

Niko Ibrahim, MIT
Universitas Kristen Maranatha

Prerequisites

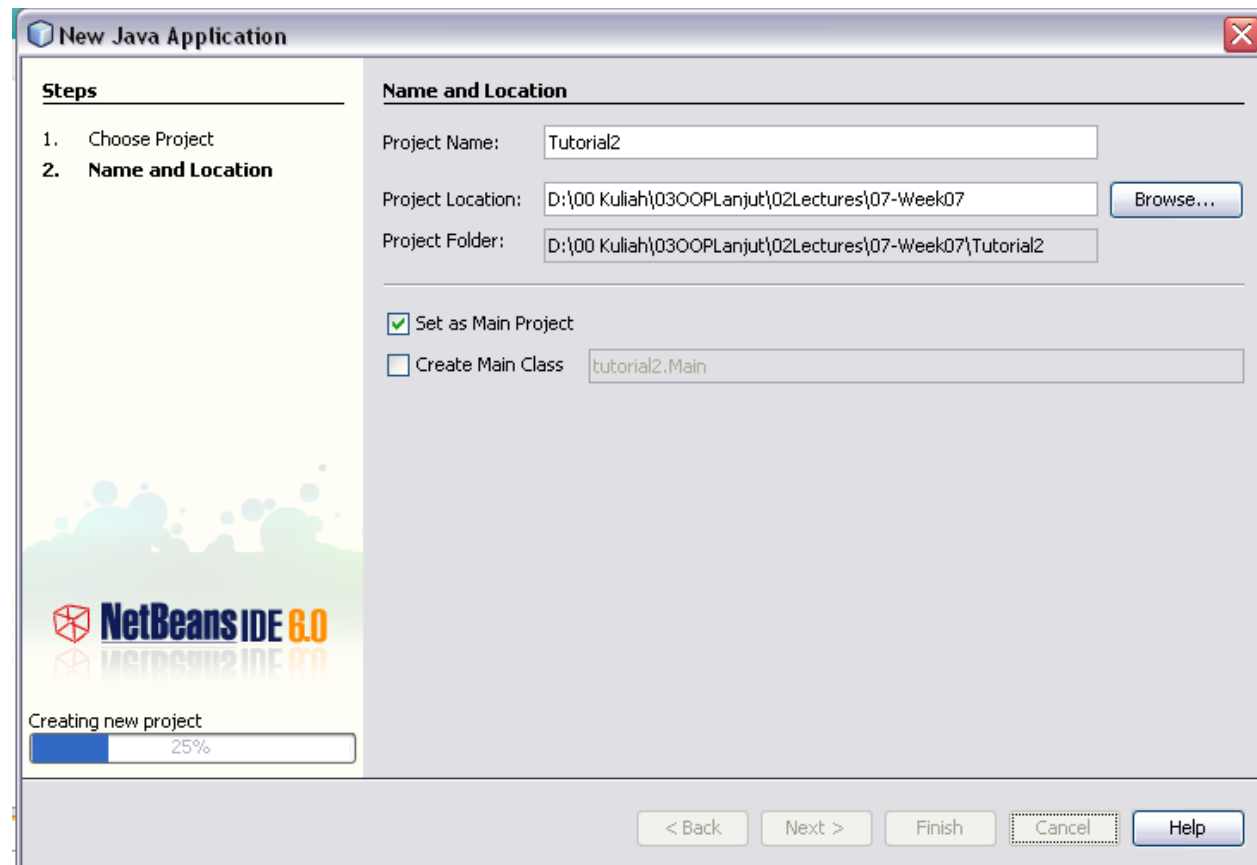
- Install NetBeans IDE 6
- Anda diasumsikan sudah mengenal:
 - Use the GUI Builder interface
 - Create a GUI Container
 - Add Components:
 - JFrame
 - JPanel
 - JButton
 - JLabel
 - JTextField
 - JComboBox
 - JList
 - JRadioButton & ButtonGroup
 - Resize Components
 - Align Components
 - Adjust Component Anchoring
 - Set Component Auto-Resizing Behavior
 - Edit Component Properties

Tutorial Content

- Pada tutorial ini, kita akan mencoba mengimplementasikan:
 - JMenu
 - JMenuItem & setMnemonics
 - JRadioButtonMenuItem
 - JTable
 - JDialog
 - JOptionPane

Step 1: Create a new project

- Buatlah sebuah project Java Application baru, dengan nama: **TutorialGUI**
- Buang tanda check pada checkbox “Create Main Class”
- Klik Next



Step2: Membuat JFrame Form

- Buatlah sebuah JFrame Form dengan meng-klik kanan project “TutorialGUI”
- Beri nama: **OrderSystem**
- Tuliskan nama package: **si.maranatha**
- Klik finish

New JFrame Form

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: SIAFTI

Project: Tutorial2

Location: Source Packages

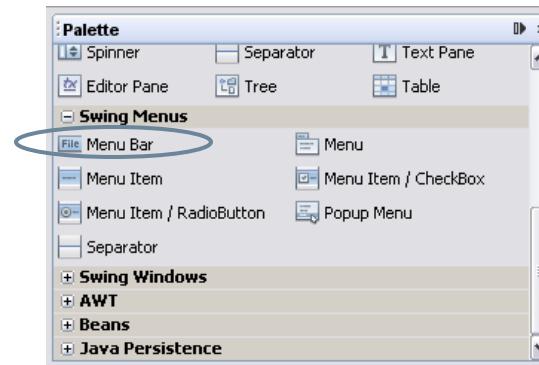
Package: si.maranatha

Created File: \\00 Kuliah\03OOP\lanjut\02Lectures\07-Week07\Tutorial2\src\si\maranatha\SIAFTI.java

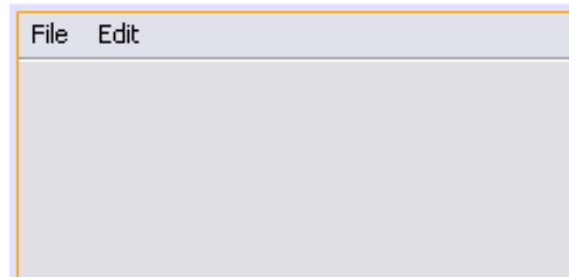
< Back Next > Finish Cancel Help

Step 3: Membuat menu

- NetBeans menyediakan berbagai komponen menu yang dapat dilihat di bagian Palette, Swing Menu

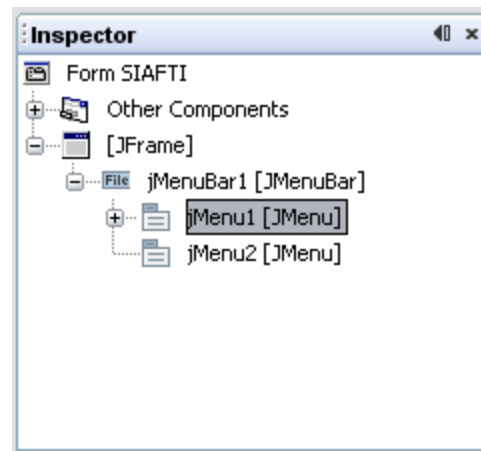


- Klik & drag komponen Menu Bar ke dalam JFrame yang telah dibuat
- Pada saat Menu Bar di-drop ke dalam JFrame, maka secara otomatis menu bar tersebut akan di-attach di bagian atas JFrame.

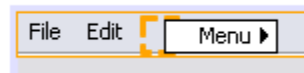


Step 4: Menambah JMenu

- Secara default, NetBeans memberikan 2 buah JMenu yaitu:
 - jMenu1 (File)
 - jMenu2 (Edit)
- Kita dapat melihat komponen-komponen Swing yang telah dibuat, melalui Inspector:

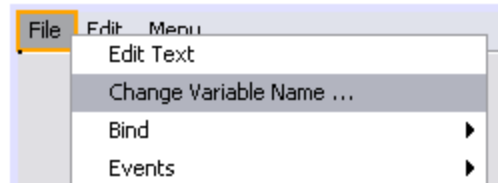


- Tambahkan sebuah JMenu baru ke dalam menu bar tersebut dengan cara men-drag&drop komponen Menu dari palette

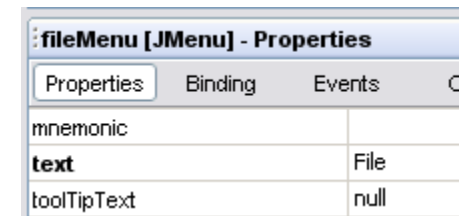


Step 5: Mengedit nama variabel & text JMenu

- Gantilah nama variabel dan text dari setiap JMenu tersebut dengan cara meng-klik kanan masing-masing menu, lalu memilih “Change variable name”



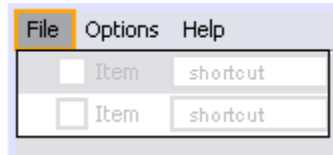
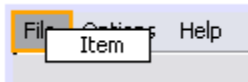
- Gantilah sbb:
 - jMenu1 (file) menjadi fileMenu
 - jMenu2 (Edit) menjadi optionMenu
 - jMenu3 (Menu) menjadi helpMenu
- Kemudian ganti tampilan teks untuk semua menu tersebut dengan cara mengubah di bagian Properties, text.
 - jMenu1 → File
 - jMenu2 → Options
 - jMenu 3 → Help



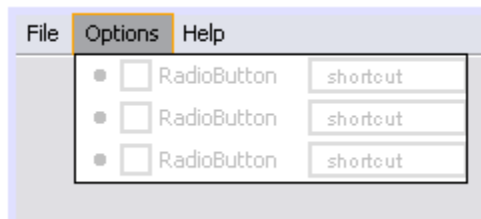
fileMenu [JMenu] - Properties			
Properties	Binding	Events	C
mnemonic			
text		File	
toolTipText		null	

Step 6: Menambahkan JMenuItem & JRadioButtonMenuItem

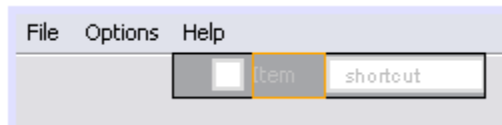
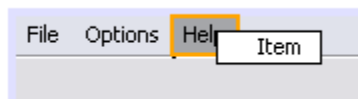
- Kita dapat menambahkan JMenuItem ke dalam JMenu dengan cara men-drag&drop komponen Menu Item dari pallette.
- Tambahkan 2 buah Menu Item kepada Menu “File”



- Tambahkan 3 buah Menu Item / RadioButton pada menu “Options”

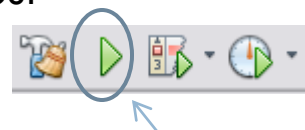
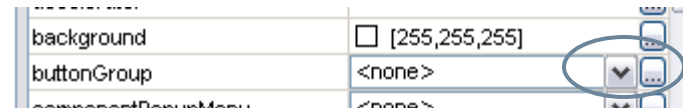


- Tambahkan 1 buah Menu Item pada menu “Help”



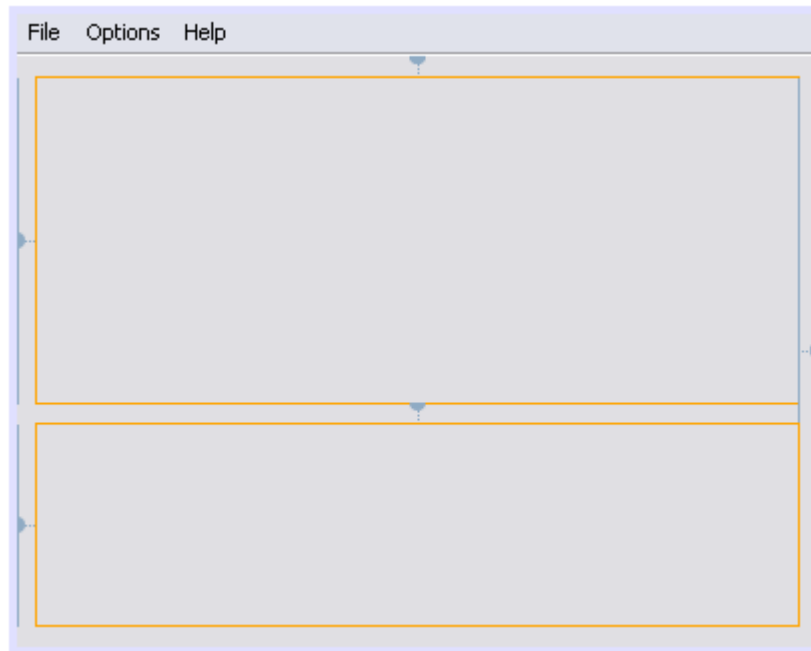
Step 7: Mengubah variable name dan teks & First Run

- Ubahlah semua variable name & teks untuk setiap menu item yang telah dibuat sbb:
 - Menu File
 - jMenuItem1 → var name: newItem, teks: New
 - jMenuItem2 → var name: quitItem, teks: Quit
 - Menu Options
 - jRadioButtonMenuItem1 → var name: hijauKuningItem, teks: “Menu Hijau, Panel Kuning”
 - jRadioButtonMenuItem2 → var name: biruPinkItem, teks: “Menu Biru, Panel Pink”
 - jRadioButtonMenuItem3 → var name: pinkHijauItem, teks: “Menu Pink, Panel Hijau”
 - Tambahkan (drag) sebuah ButtonGroup dari palette ke dalam JFrame
 - Untuk ketiga JRadioButtonMenuItem, pada bagian properties, kategori buttonGroup, klik drop down list, kemudian pilihlah “buttonGroup1”
 - Ubahlah properties **selected** sedemikian rupa sehingga hanya radio button yang pertama yang memiliki atribut selected.
 - Menu Help
 - jMenuItem3 → var name: aboutItem, teks: About
- Run project dengan menekan tombol Run



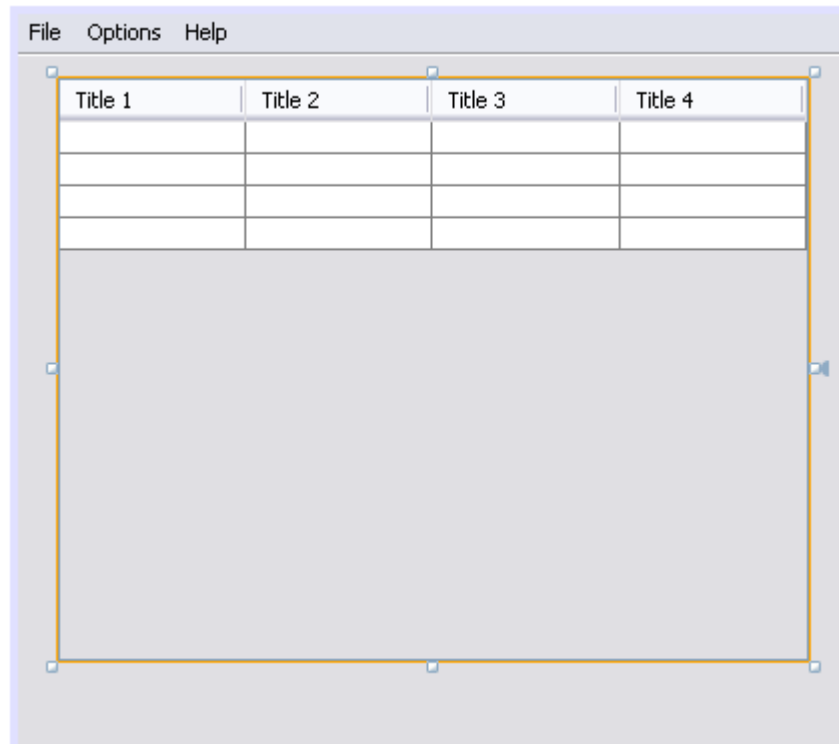
Step 8: Menambah Panel

- Buatlah 2 buah Panel di dalam JFrame sbb:



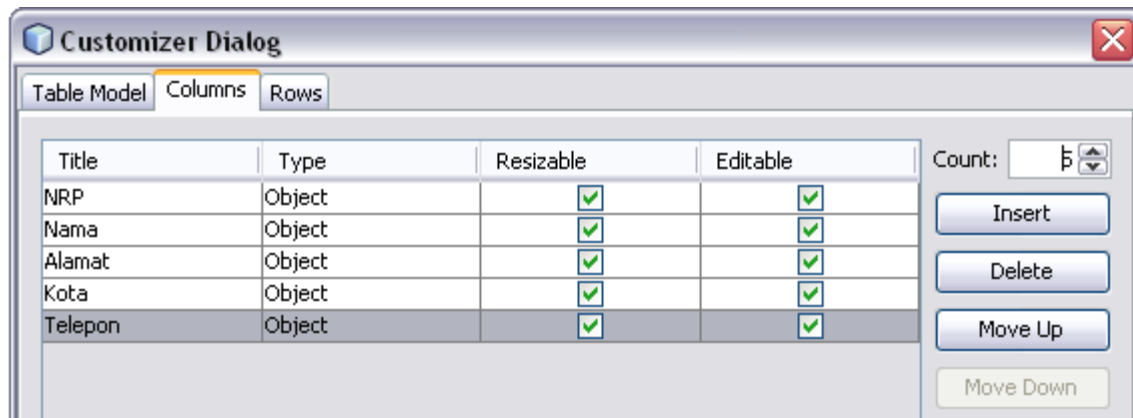
Step 9: Menambah jTable

- Drag&Drop komponen jTable ke dalam panel atas:

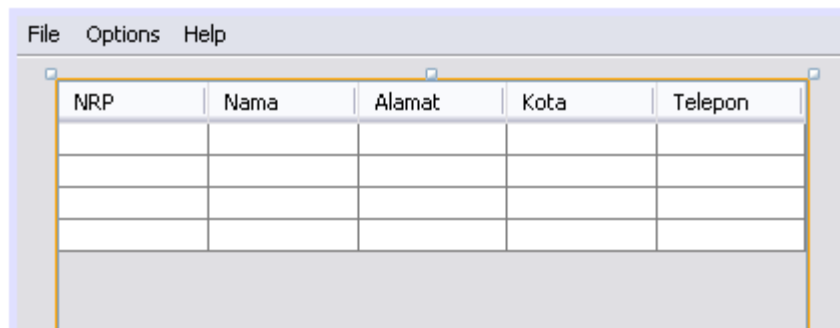


Step 10: Customizing JTable

- Klik kanan JTable tersebut, kemudian pilih “Table Contents...”
- Akan muncul Customizer Dialog untuk tabel tersebut
- Pilihlah tab Columns, kemudian ubahlah sbb:

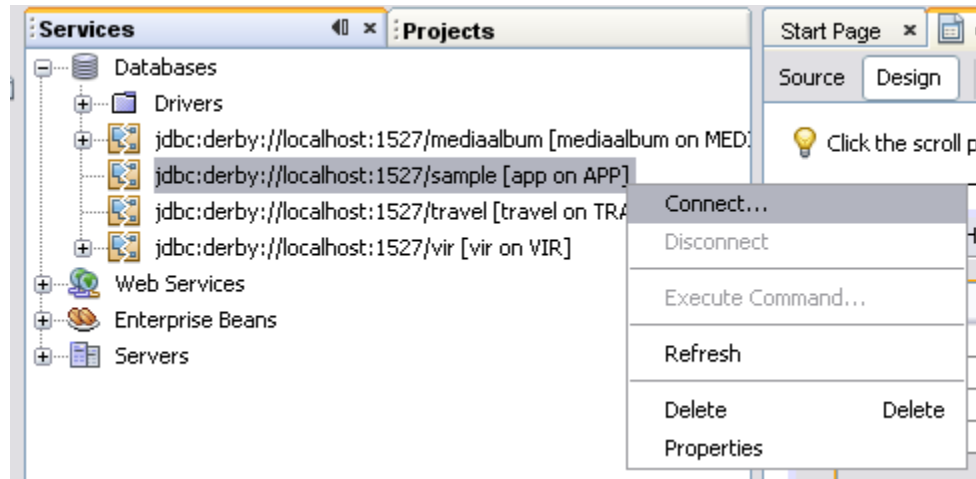


- Klik tombol Close
- Hasil:



Step 1 1: Melakukan Koneksi Database

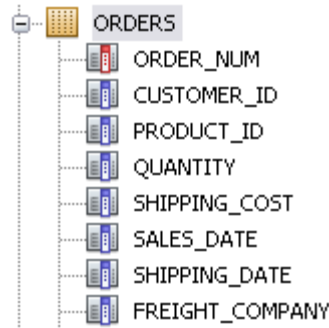
- NetBeans dilengkapi dengan feature RDBMS yaitu JavaDB
- Kita akan melakukan koneksi ke database yang telah ada, yaitu “sample”
- Pilihlah services, kemudian klik kanan database “sample”
- Klik connect



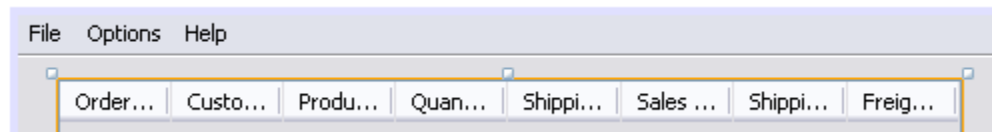
- Isilah:
 - Username: app
 - Password: app
- Klik OK

Step 12: Data Binding Tabel ke jTable

- Dengan menggunakan NetBeans, kita dapat dengan mudah melakukan binding sebuah tabel ke dalam jTable.
- Pada database “sample”, klik Tables, di situ terlihat banyak tabel seperti “CUSTOMER”, “DISCOUNT CODE”, dll.
- Klik tanda ‘+’ pada tabel ORDERS untuk melihat kolom-kolom yang ada.



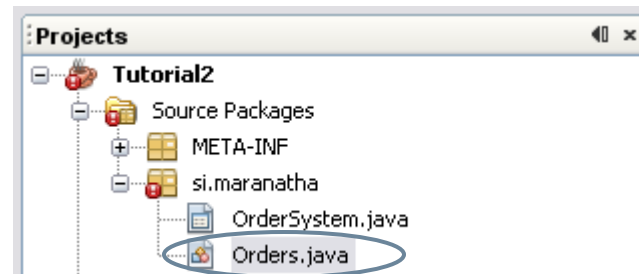
- Untuk melakukan binding, klik tabel ORDERS kemudian drag&drop ke dalam jTable yang telah dibuat sebelumnya.
- Klik kanan jTable tersebut, pilih “Set default size”
- Hasil:



> RUN !

Step 13: Intro to Java Persistence API

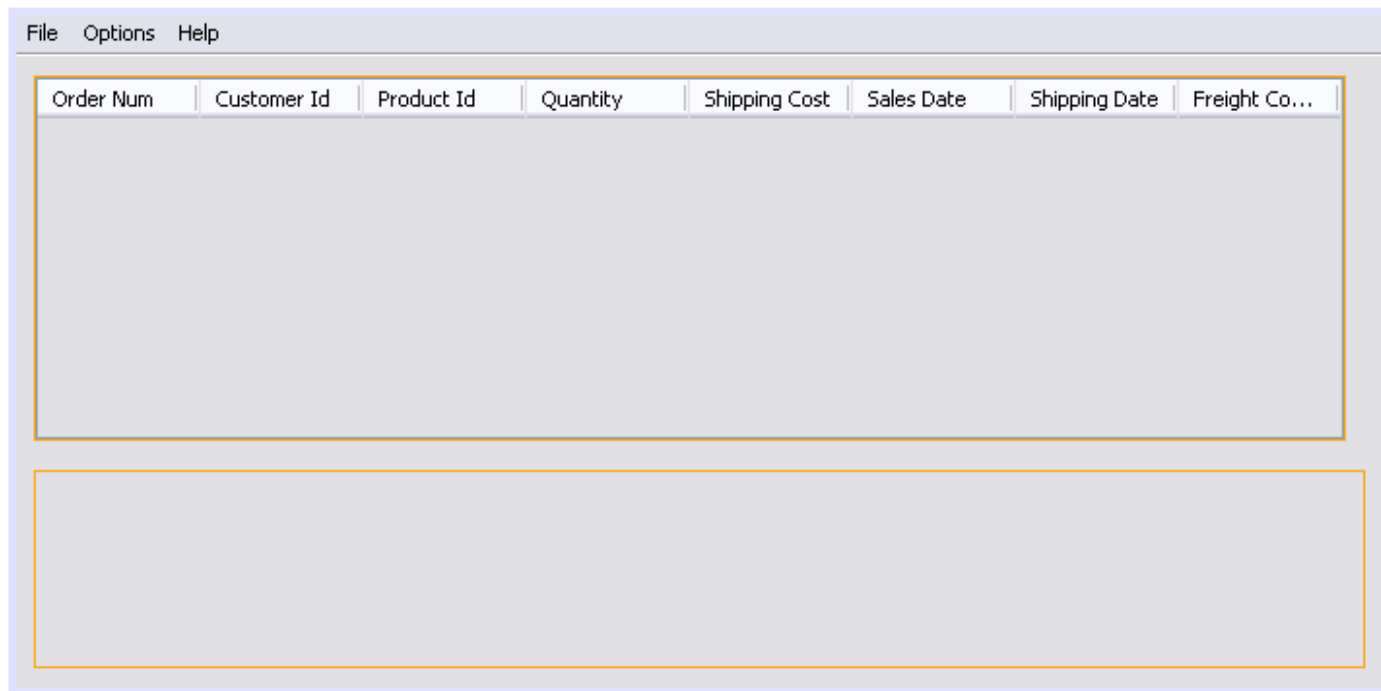
- Pada saat kita melakukan binding sebuah tabel, NetBeans secara otomatis membuat sebuah file yang bernama Orders.java



- Orders.java ini merupakan Entity Class yang mewakili tabel ORDERS dari database "sample".
- Entity Class ini merupakan bagian dari JPA (Java Persistence API) yang akan dibahas di lain waktu 😊.
- Untuk saat ini, silahkan klik file tersebut untuk melihat isi source code-nya.

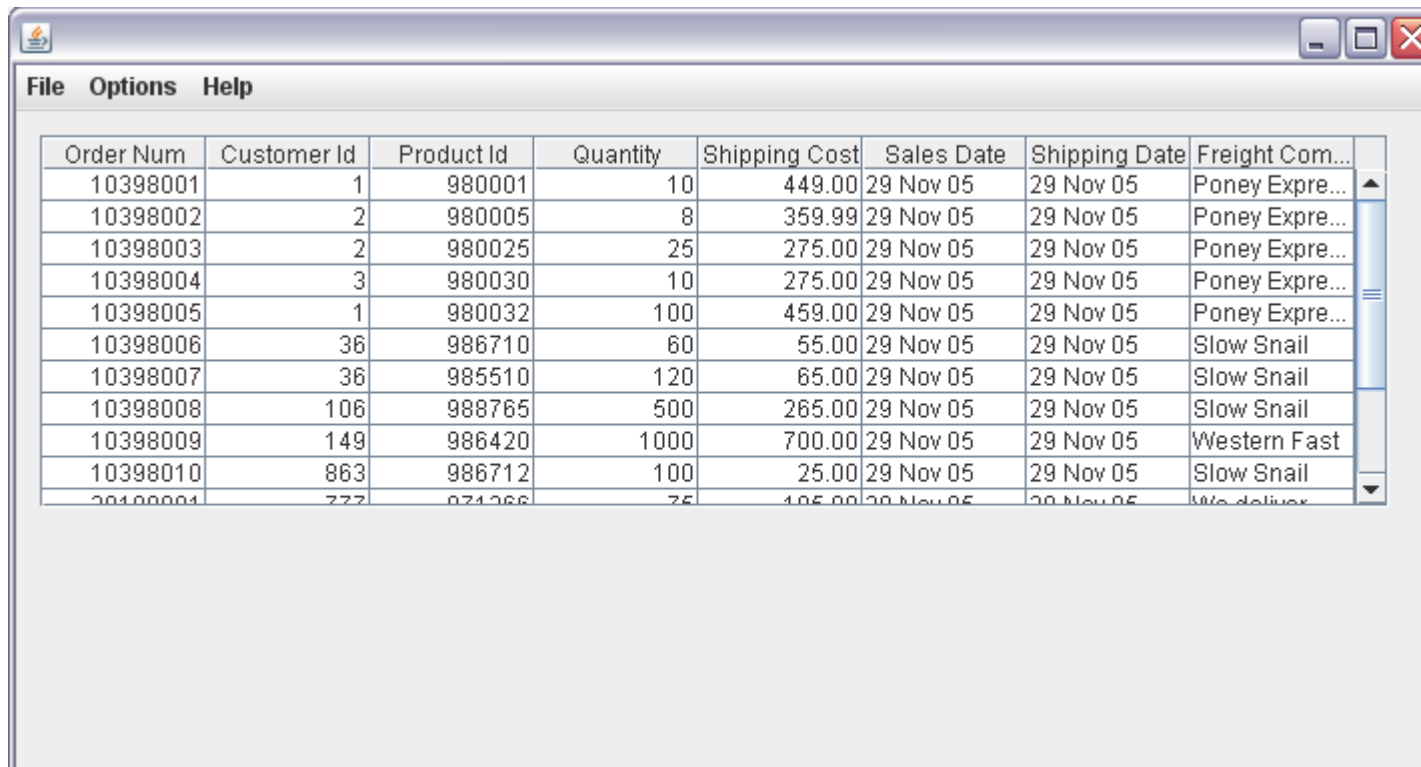
Step 14: Mengubah ukuran

- Ubahlah ukuran JFrame, JPanel & JTable dengan cara menggeser sisi-sisinya sedemikian rupa sehingga menjadi lebih panjang sbb:



Step 15: Resize dan Ubah posisi kolom

- Run Program:



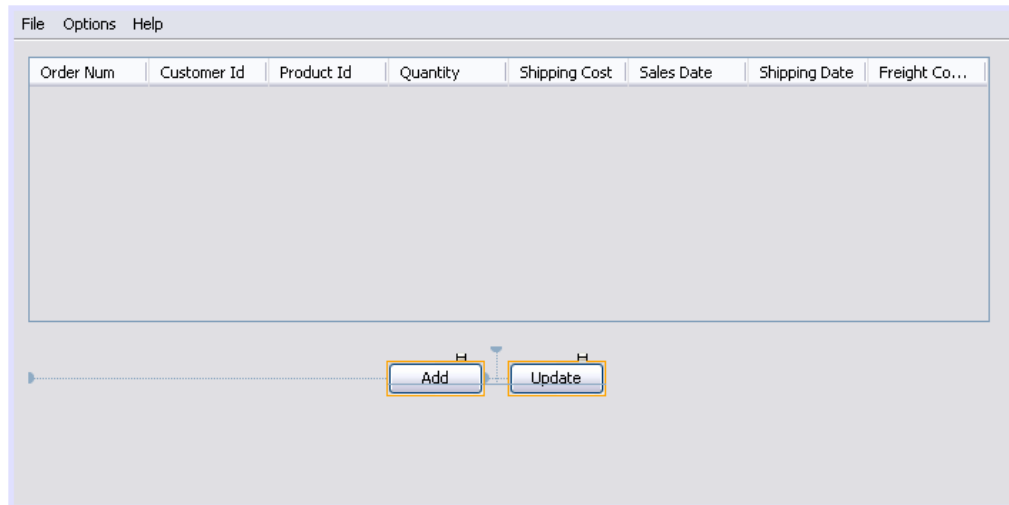
The screenshot shows a window titled "File Options Help" containing a data grid. The grid has 8 columns and 10 rows of data. The columns are: Order Num, Customer Id, Product Id, Quantity, Shipping Cost, Sales Date, Shipping Date, and Freight Com... The data is as follows:

Order Num	Customer Id	Product Id	Quantity	Shipping Cost	Sales Date	Shipping Date	Freight Com...
10398001	1	980001	10	449.00	29 Nov 05	29 Nov 05	Poney Expre...
10398002	2	980005	8	359.99	29 Nov 05	29 Nov 05	Poney Expre...
10398003	2	980025	25	275.00	29 Nov 05	29 Nov 05	Poney Expre...
10398004	3	980030	10	275.00	29 Nov 05	29 Nov 05	Poney Expre...
10398005	1	980032	100	459.00	29 Nov 05	29 Nov 05	Poney Expre...
10398006	36	986710	60	55.00	29 Nov 05	29 Nov 05	Slow Snail
10398007	36	985510	120	65.00	29 Nov 05	29 Nov 05	Slow Snail
10398008	106	988765	500	265.00	29 Nov 05	29 Nov 05	Slow Snail
10398009	149	986420	1000	700.00	29 Nov 05	29 Nov 05	Western Fast
10398010	863	986712	100	25.00	29 Nov 05	29 Nov 05	Slow Snail

- Cobalah untuk me-resize kolom.
- Cobalah untuk menggeser posisi kolom.

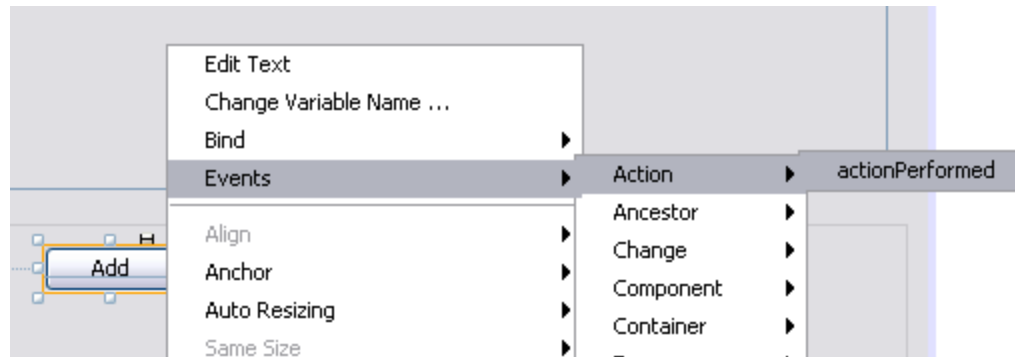
Step 16: Menambahkan JButton

- Tambahkan 2 buah JButton ke dalam panel bawah.
 - jButton1 → var name: addButton, teks: Add
 - jButton2 → var name: updateButton, teks: Update

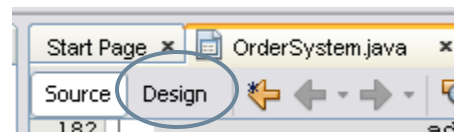


Step 17: Event-Handling

- NetBeans menyediakan fasilitas lengkap untuk melakukan event-handling.
- Untuk melakukan event-handling, pertama-tama kita harus memberikan “listener” kepada komponen yang diinginkan.
- Klik kanan tombol addButton, kemudian pilih “Events” > “Action”



- klik “actionPerformed”
- NetBeans akan melakukan switch dari “Design View” menjadi “Source Code View” dari file OrderSystem.java
- Untuk kembali ke “Design View”, klik saja tombol “Design” di atas.

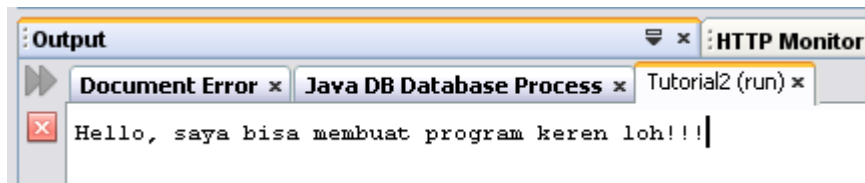


Step 18: Mengimplementasikan Event-Handling

- Pada saat kita klik “actionPerformed”, maka secara otomatis NetBeans akan membuat sebuah method yang bernama xxxActionPerformed, dimana xxx merupakan nama variabel dari komponennya.
- Berikut adalah method actionPerformed untuk AddButton

```
192  
193 private void AddButtonActionPerformed(java.awt.event.ActionEvent evt) {  
194     // TODO add your handling code here:  
195 }  
196
```

- Untuk menguji event-handling, cobalah ketikkan:
 System.out.println("Hello, saya bisa membuat program keren loh!!!");
- Run program, kemudian klik tombol Add
- Hasil println dapat dilihat di bagian output, pada tab “Tutorial2(run)”

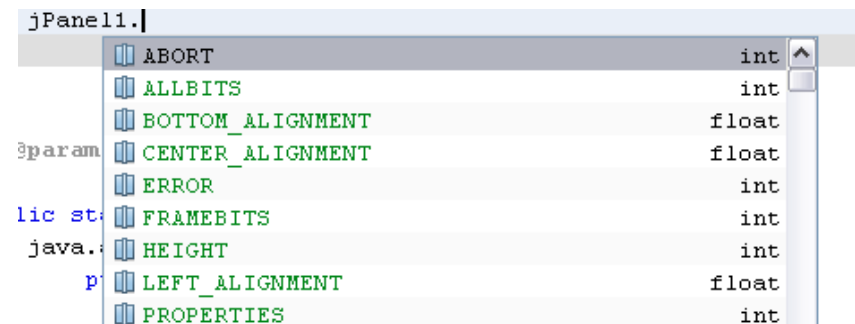


Step 19: NetBeans Autocomplete

- NetBeans memiliki feature autocomplete pada saat mengetik kode program.
- Sekarang kita coba memberikan “event-listener” kepada `JRatioButtonItem`.
- Klik kanan `hijauKuningItem`, kemudian pilih `Events>Action>actionPerformed`

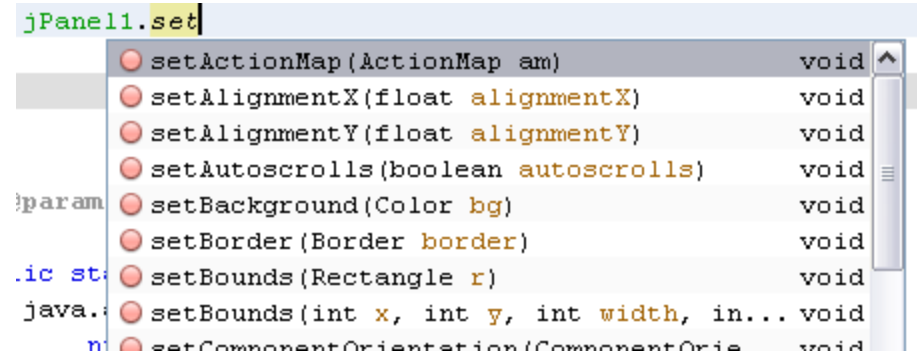
```
private void hijauKuningItemActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
}
```

- Pada method tersebut, cobalah ketik: `jPanel1`
- Kemudian lanjutkan dengan mengetik tanda titik: `jPanel1.`
- NetBeans akan menampilkan berbagai atribut/method yang dapat digunakan oleh `jPanel1`.



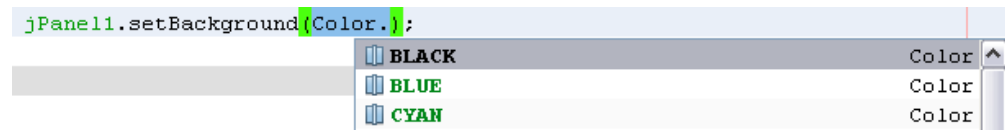
Step 20: Ctrl + Space

- Setelah mengetik tanda titik “.” kita dapat memaksa NetBeans untuk menampilkan auto-complete dengan menekan tombol “Ctrl+Space”
- Cobalah mengetikan: `jPanel1.set` (lalu berhenti)
- Kemudian tekan tombol “Ctrl+space”
- Secara otomatis, NetBeans akan menampilkan auto-complete untuk method yang dimulai dengan kata “set”:



```
jPanel1.set
  ● setActionMap(ActionMap am) void
  ● setAlignmentX(float alignmentX) void
  ● setAlignmentY(float alignmentY) void
  ● setAutoscrolls(boolean autoscrolls) void
param ● setBackground(Color bg) void
  ● setBorder(Border border) void
.ic st ● setBounds(Rectangle r) void
java. ● setBounds(int x, int y, int width, in... void
n ● setComponentOrientation(ComponentOri void
```

- NetBeans juga menampilkan dokumentasi “javaDoc” untuk setiap atribut/method yang kita sorot.
- Carilah dan klik method “setBackground”
- Lalu langsung ketikan: `Color.`
- Pilih: `yellow`



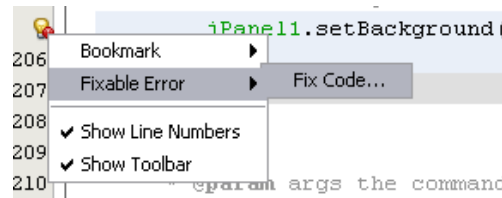
```
jPanel1.setBackground(Color.);
  ■ BLACK Color
  ■ BLUE Color
  ■ CYAN Color
```

Step 21: Automatic Fixing Error

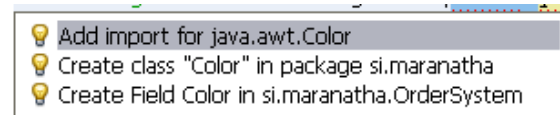
- ❑ NetBeans juga dilengkapi dengan feature “Automatic fixing error” 😊
- ❑ Perhatikanlah bahwa di dalam source code tersebut, kata `Color.yellow` diberi garis warna merah.
- ❑ NetBeans juga menampilkan icon lampu disebelah kiri, yang berarti ada eror di dalam kode kita.

```
204 | // TODO add your handling code here:  
    | jPanel1.setBackground(Color.yellow);
```

- ❑ Klik kanan icon lampu tersebut, lalu pilih Fixable Error, klik Fix Code...



- ❑ Pilih “Add import for java.awt.Color”



- ❑ Perhatikan bahwa garis merah menghilang 😊. Scroll source code ke atas, lihatlah bahwa NetBeans telah menambahkan kode import:

```
7 package si.maranatha;  
8  
9 import java.awt.Color;  
10
```

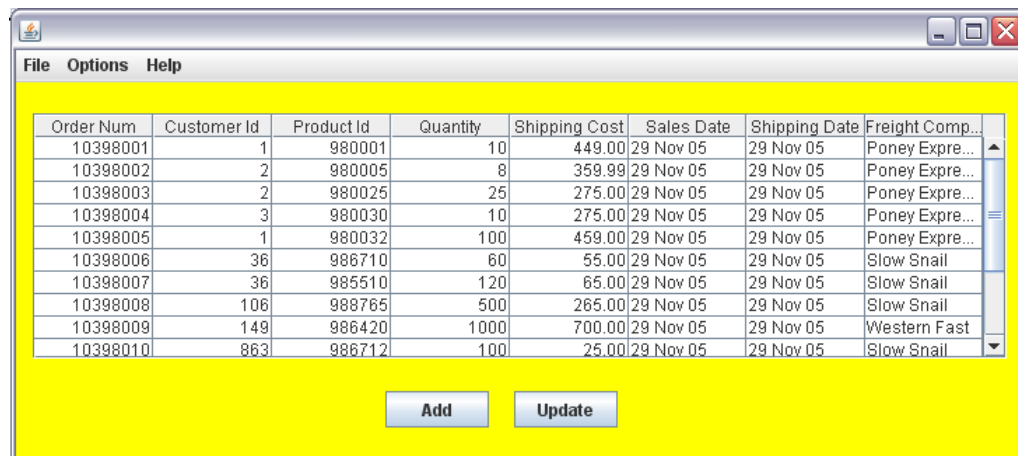
- ❑ Tips: Anda juga bisa langsung melakukan klik kiri icon tsb untuk langsung menampilkan pilihan Fix Code.

Step 22: modifikasi jPanel2 & JFrame

- Beri background warna kuning juga untuk jPanel2 & JFrame

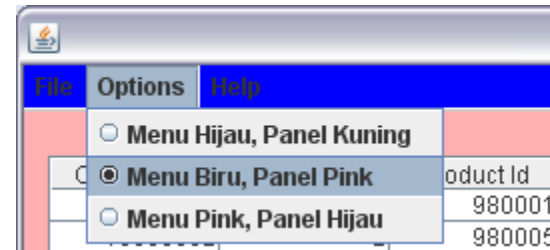
```
jPanel1.setBackground(Color.yellow);  
jPanel2.setBackground(Color.yellow);  
this.getContentPane().setBackground(Color.yellow);
```

- Run program!
- Klik radio item yang pertama.
- Note: Untuk memberi warna JFrame, Anda harus mengambil komponen ContentPane terlebih dahulu. ContentPane inilah yang kita beri warna background.



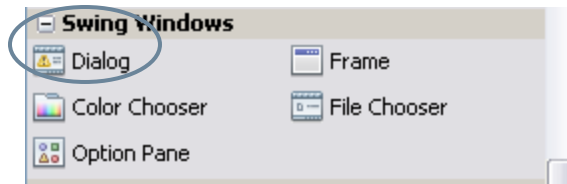
Step 23: Do It Yourself (DIY)

- Pada method tersebut, beri warna background “green” untuk jMenuBar1.
- Setelah itu, berikan pula action-listener untuk kedua menu item lainnya.
 - ▣ biruPink item
 - ▣ pinkHijau menu item
- Hasil:

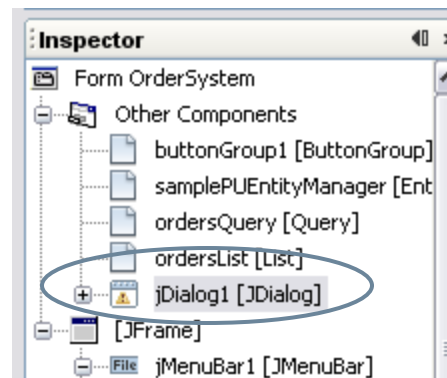


Step 24: Membuat JDialog

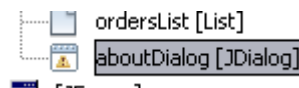
- Sekarang kita akan membuat sebuah JDialog.
- Drag&drop Dialog pada palette ke dalam form yang telah kita buat.



- Perhatikan di dalam Inspector, bahwa komponen JDialog telah ditambahkan ke dalam JFrame.



- Klik kanan jDialog1 tsb, kemudian pilih Change variable name, beri nama: aboutDialog



Step 25: Dekorasi JDialog

- aboutDialog tersebut masih kosong.
- Kita akan mendekorasinya dengan cara menambahkan sebuah gambar, label, dan tombol.
- Double klik aboutDialog tersebut.
- Akan ditampilkan form untuk dialog tersebut:



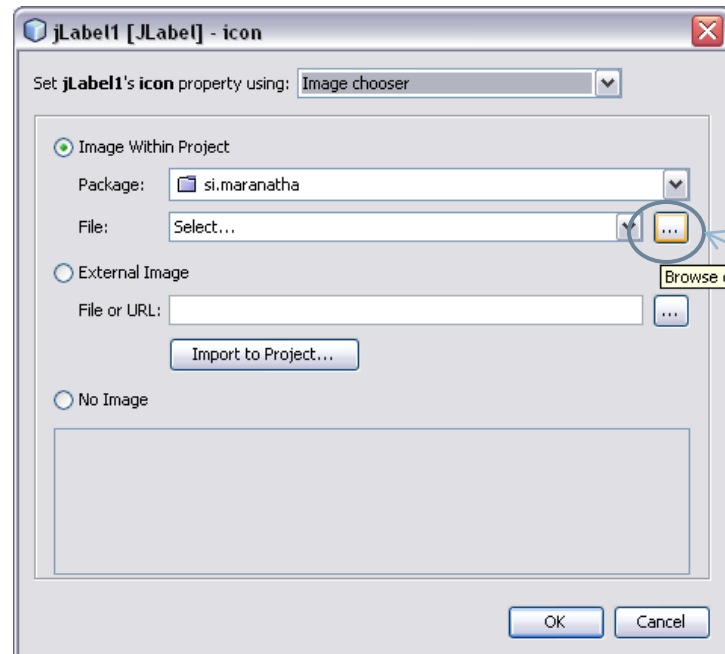
- Kita akan menambahkan sebuah gambar dan label pada form dialog tersebut.
- Sebelumnya, copy-kan gambar logo.png ke folder “\Tutorial2\src”
- Drag&drop sebuah label ke dalam form dialog.
- Lalu pada properties, klik tanda elipsis dari icon:



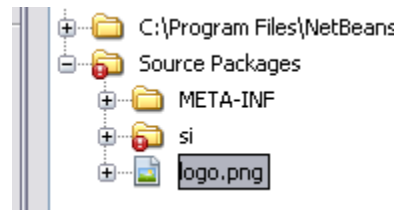
- Akan muncul dialog untuk mengelola jlabel tersebut.

Step 26: Menambahkan resources

- Akan muncul dialog sbb:



- Klik tanda elipsis pada "File:". Kemudian pilihlah gambar yang telah di-copykan tadi.

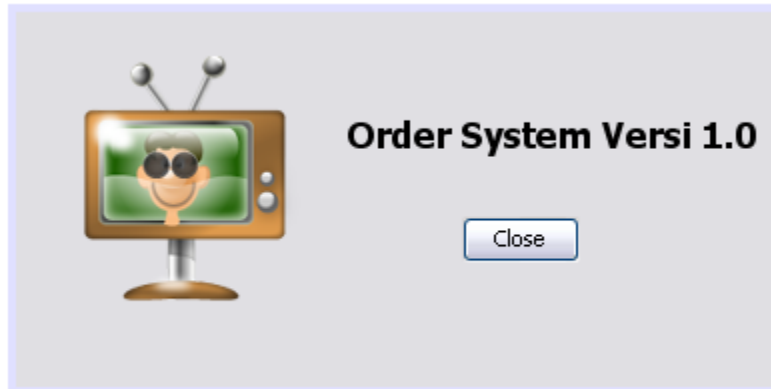


- Klik OK.

Step 27: Tambahkan Label & Button & Modifikasi properties dialog

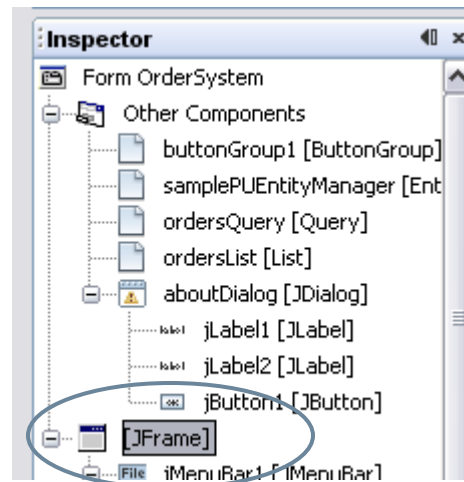
- Tambahkan sebuah label dengan tulisan: “Order System Versi 1.0”
- Anda dapat mengubah jenis, ukuran, dan warna font pada bagian properties.
- Kemudian tambahkan juga sebuah button.
- Berikanlah action untuk close button tersebut:
`aboutDialog.setVisible(false);`
- Pada bagian properties, modifikasi hal-hal berikut:
 - title: About
 - minimumSize: 400, 200
 - modal: true (beri tanda check) → agar bisa on-top (tidak bisa ke form sebelumnya)
 - resizable: false (uncheck)

- Hasil:



Step 28: Kembali ke form utama

- Untuk kembali ke “design-view” dari form utama, double-klik saja JFrame melalui inspector:

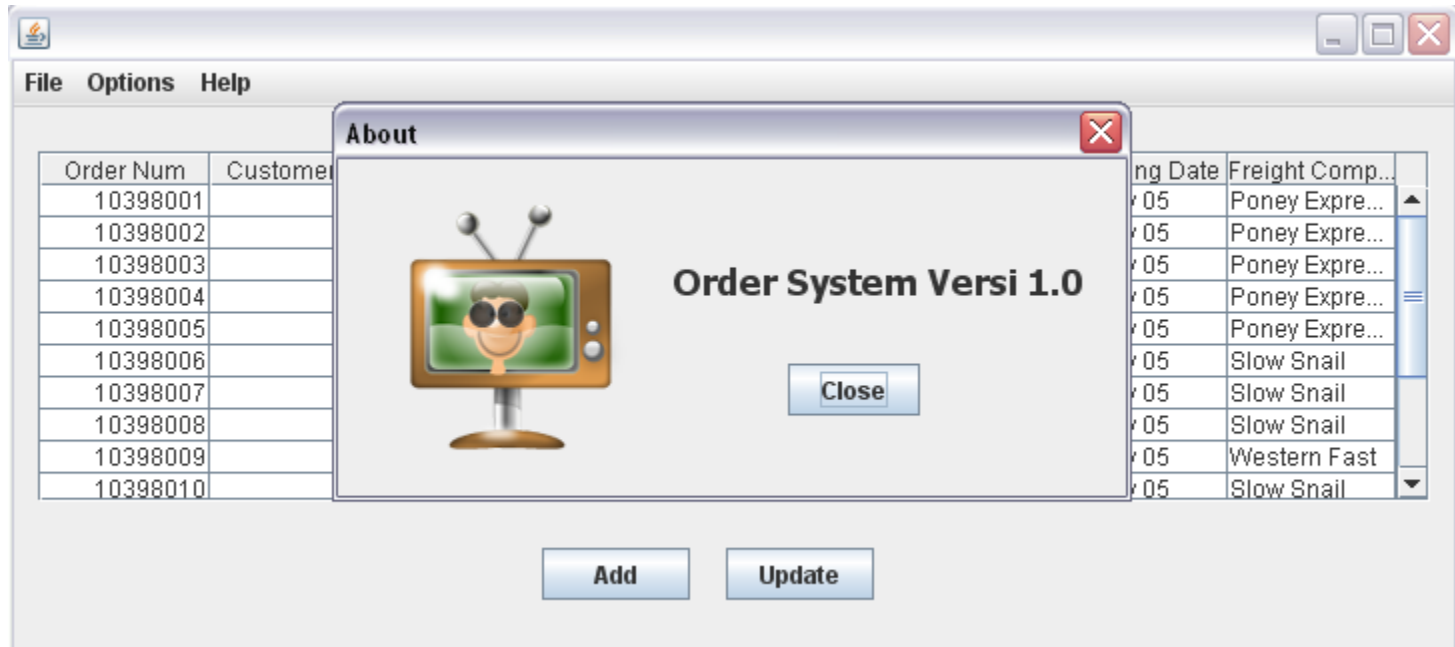


- Beri action untuk menu item About.
- Ketikkan kode berikut:

```
aboutDialog.setVisible(true);
```
- Berikan pula action untuk menu item Quit:

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

FINAL RESULT



Congratulations!
© Niko Ibrahim, MIT