

BAB III

BEKERJA DENGAN KONTROL JCOMBOBOX, JLISTBOX

A. CAPAIAN PEMBELAJARAN

1. Mahasiswa dapat mengerti dan menggunakan kontrol `jComboBox`, `jListBox`
2. Mahasiswa dapat menganalisa kapan penggunaan kontrol `jComboBox`, `jListBox`
3. Mahasiswa dapat membuat program menggunakan kontrol `jComboBox`, `jListBox`

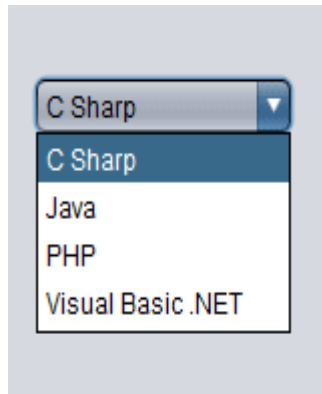
B. MATERI

1. Kontrol `jComboBox`

`jComboBox` adalah daftar item yang dapat dipilih oleh pengguna melalui drop-down. Kontrol ini dapat ditemukan di pallet NetBeans, di bawah *Swing Control*:

`jComboBox` menggunakan "List model" sendiri yang memungkinkan langsung penambahan dan penghapusan item dari daftar drop-down. Kontrol ini

juga menyediakan sendiri kemampuan scroll tanpa tambahan panel *scroll*



Gambar 3. 1 Control Jcombo Box

a. Properties yang sering digunakan pada JComboBox

Model	Menetapkan item dalam bagian daftar drop-down
ItemCount	Jumlah item dalam kotak kombo.
Font	Penamaan Font, style, ukuran comboBox
background	Latar Belakang warna pada comboBox
Foreground	Warna pada comboBox
Editable	Menentukan Lebar kolom dari comboBox

maximumRowCount	Menentukan Tinggi dari comboBox
selectedIndex	Jumlah baris dari comboBox
selectedItem	Variabel Boolean yang menunjukkan apakah teks Wrap bernilai <i>True</i> , default <i>False</i> .

b. Method pada JComboBox

setEditable	Menetapkan apakah item yang dipilih dapat diedit.
getItemCount	Mendapatkan jumlah item dalam comboBox.
setMaximumRowCount	Mengatur jumlah item untuk ditampilkan di kotak drop-down.
SetFont	Men set Penamaan Font, style, ukuran JComboBox.
setBackground	Men set Latar Belakang warna pada Text Area
setForeground	Men Warna Huruf pada label
getSelectedItem	Mengambil item yang dipilih

setSelectedIndex Memilih item tertentu dalam
jComboBox

c. Mengisi kan Item jComboBox

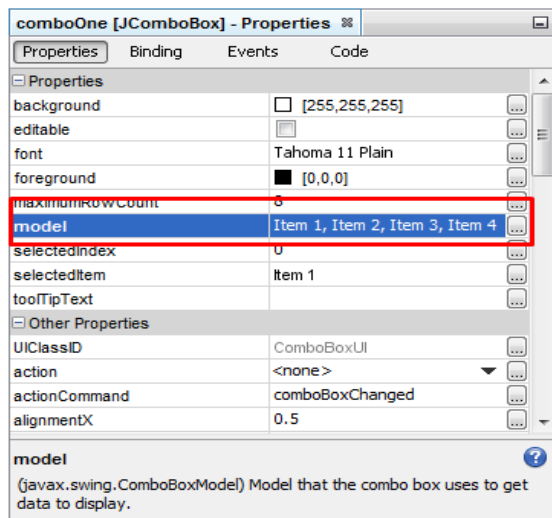
Untuk mengisi daftar item pada jComboBox
bisa dilakukan 2 cara

- 1) Melalui jendela properties **Model**
- 2) Melalui **code program**

1) **Melalui jendela properties Model**

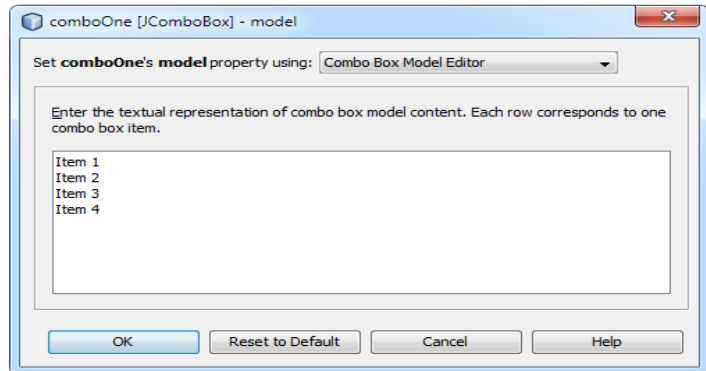
Langkah untuk pengisian melalui properties
model yaitu

- a) Klik kanan comboBox pilih menu
properties. Pada jendela properti di
sebelah kanan NetBeans. Temukan
properti model:



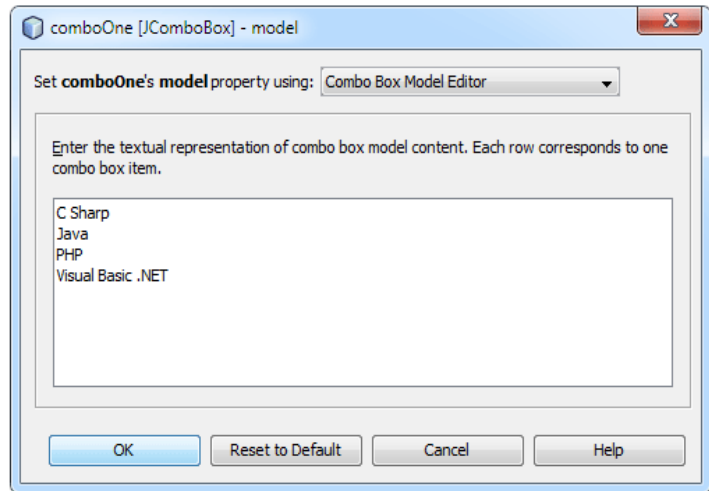
Gambar 3. 2 Properties Model JComboBox

- b) Klik tombol kecil di sebelah kanan baris model, yang memiliki tiga titik di dalamnya. Muncul kotak dialog berikut:



Gambar 3. 3 ComboBox Model Editor

- c) Blok semua item di area teks putih, lalu hapus dengan tombol Del. Ganti dan isikan dengan item berikut: C Sharp, Java, PHP, Visual Basic .NET.
- d) Kotak dialog kemudian akan terlihat seperti ini:

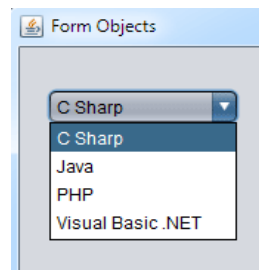


Gambar 3. 4 Pengisian Item ComboBox Model Editor

- e) Klik OK setelah melakukan perubahan. jComboBox kita sekarang akan terisi dengan item kita yang ditulisi sebelumnya.

Jalankan program kita dan ujilah: (Cukup klik OK ketika diminta untuk memilih Kelas Utama.)

Jika tidak ada masalah maka hasil run nampak seperti ini :

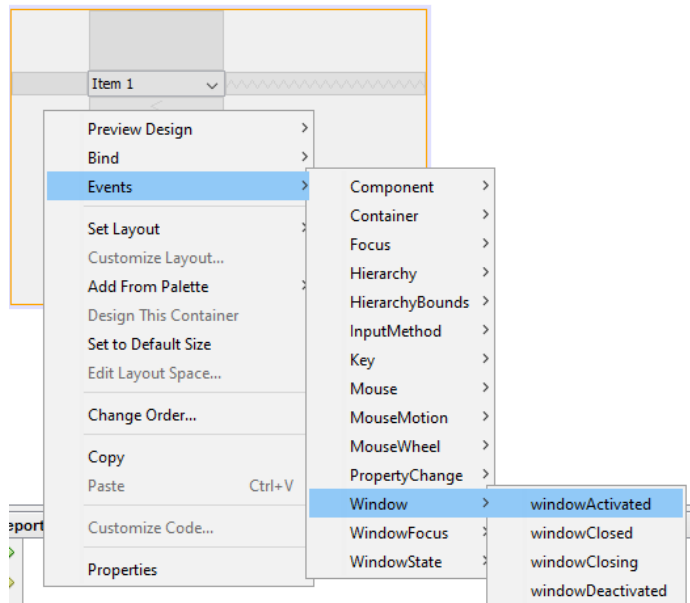


Gambar 3. 5 Hasil Run ComboBox

2) Melalui kode program

Langkah untuk pengisian melalui kode program yaitu

- a) Setelah JComboBox ditambahkan kedalam frame
- b) Klik kanan pada frame pilih menu event > Windows>>windowActivate



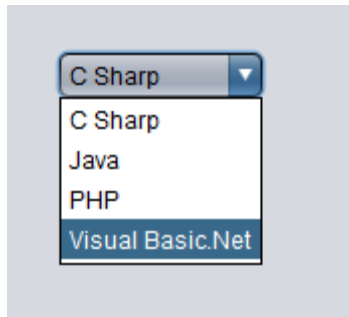
Gambar 3. 6 Langkah Kode Program ComboBox

- c) Selanjutnya akan dibawa ke editor code frame activate ketikan code seperti gambar dibawah ini :

```
private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    jComboBox1.addItem("C Sharp");
    jComboBox1.addItem("Java ");
    jComboBox1.addItem("PHP ");
    jComboBox1.addItem("Visual Basic.Net");
}

/**
```

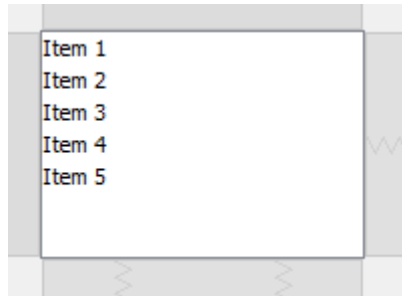
- d) Run program dengan shift+F6
- e) Jika berjalan baik hasil nya seperti dibawah ini :



Gambar 3. 7 Hasil Run Kode Program Combo Box

2. Kontrol *jList*

jCheckBox adalah kontrol yang berguna untuk memilih item dari daftar pilihan. Tapi, bagaimana jika kita memiliki daftar 100 item? Apakah kita akan membuat 100 *jCheckBox*? Komponen yang dapat memecahkan masalah ini adalah menggunakan kontrol *jList*. *jList* yang dapat menampilkan daftar item sebanyak item yang kita inginkan, dimana pengguna dapat memilih satu atau lebih item.



Gambar 3. 8 Kontrol JListBox

jList mirip seperti kontrol JComboBox untuk pengisian itemnya, perbedaan dari kontrol ini adalah pada comboBox daftar pilihan item tidak akan muncul terlebih dahulu, akan ditampilkan ketika panah di bagian kanan kita klik, pada jList daftar pilihan item akan muncul langsung.

a. Properties yang sering digunakan pada jList

Model	Menetapkan item dalam bagian daftar drop-down list
ItemCount	Jumlah item dalam jList .
Font	Penamaan Font, style, ukuran jList
background	Latar Belakang warna pada jList
Foreground	Warna huruf pada jList
Editable	Menentukan Lebar kolom dari jList

maximumRowCount	Menentukan Tinggi dari <code>jList</code>
selectedIndex	Jumlah baris dari <code>jList</code>
selectedItem	Variabel Boolean yang menunjukkan apakah teks Wrap bernilai <i>True</i> , default <i>False</i> .

b. Method pada `jList`

setEditable	Menetapkan apakah item yang dipilih dapat diedit.
getItemCount	Mendapatkan jumlah item dalam <code>jList</code> ..
setMaximumRowCount	Mengatur jumlah item untuk ditampilkan di kotak drop-down.
SetFont	Men set Penamaan Font, style, ukuran <code>jList</code>
setBackground	Men set Latar Belakang warna pada <code>jList</code>
setForeground	Men Warna Huruf pada label
getSelectedItem	Mengambil item yang dipilih
setSelectedIndex	Memilih item tertentu dalam <code>jList</code>

c. Mengisi kan Item JComboBox

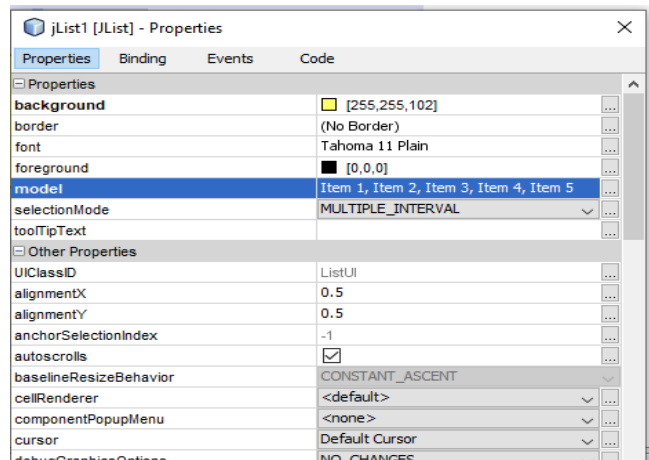
Untuk mengisi daftar item pada JComboBox bisa dilakukan 2 cara

- 1) Melalui jendela properties **Model**
- 2) Melalui **code program**

1) Melalui jendela properties Model

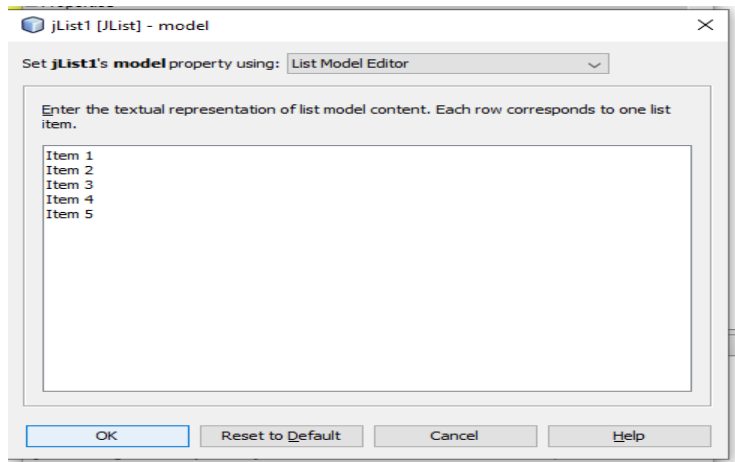
Langkah untuk pengisian melalui propoerties model yaitu

- a) Klik kanan jList pilih menu properties.
Pada jendela properti di sebelah kanan.
Temukan properti model:



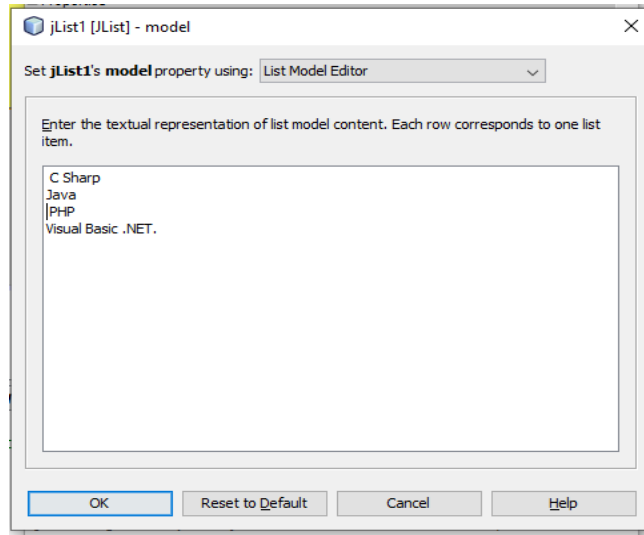
Gambar 3. 9 Jendela Properties JList

- b) Klik tombol kecil di sebelah kanan baris model, yang memiliki tiga titik di dalamnya.
Muncul kotak dialog berikut:



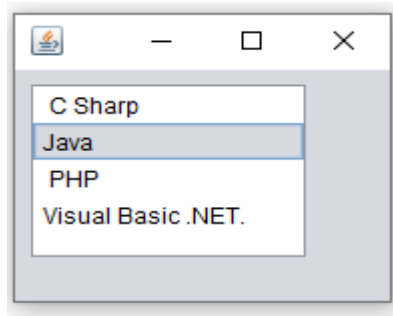
Gambar 3. 10 Item JList Model

- c) Blok semua item di area teks putih, lalu hapus dengan tombol Del. Ganti dan isikan dengan item berikut: C Sharp, Java, PHP, Visual Basic .NET.
- d) Kotak dialog kemudian akan terlihat seperti ini:



Gambar 3. 11 JList Model Editor

- e) Klik OK setelah melakukan perubahan. jList kita sekarang akan terisi dengan item kita yang ditulis sebelumnya. Jalankan program dari menu Run>Run Files atau tombol Shift+F6. Jika tidak ada masalah tampilan hasil nampak seperti gambar dibawah ini

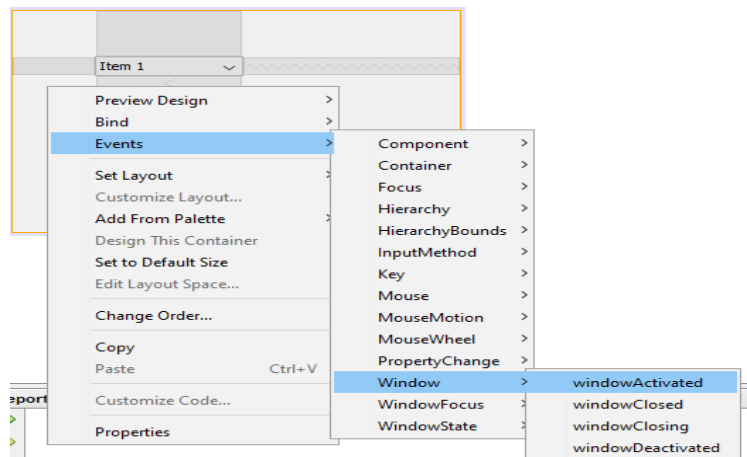


Gambar 3. 12 Hasil Run Jlist

2) Melalui kode program

Langkah untuk pengisian melalui kode program yaitu

- a) Sisipkan dengan drag drop dari menu Pallete- List kedalam frame
- b) Selanjutnya Klik kanan pada frame pilih menu event > Windows>windowActivate untuk *event handling*



Gambar 3. 13 Langkah Event Jlist kode Program

- c) Selanjutnya akan dibawa ke editor code frame activate ketikan code seperti tampilan dibawah ini :

```

import javax.swing.DefaultListModel;

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:

    DefaultListModel listModel1 = new DefaultListModel();

    listModel1.addElement("C Sharp");

    listModel1.addElement("Java");

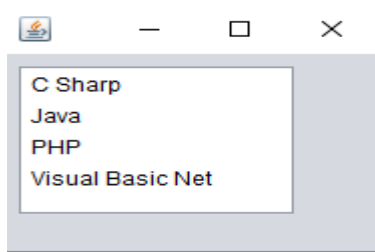
    listModel1.addElement("PHP");

    listModel1.addElement("Visual Basic Net");

    jList1.setModel(listModel1);
}
}

```

- d) Run program dengan shift+F6
- e) Jika berjalan baik hasil nya seperti dibawah ini :

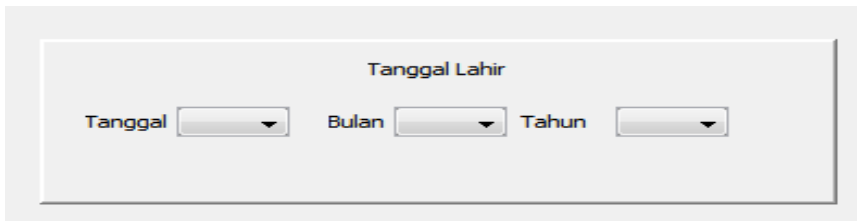


Gambar 3. 14 Hasil Run Jlist

C. LATIHAN

1. kontrol JComboBox

- a. Buat program untuk memilih tanggal, bulan, tahun menggunakan comboBox tampilan desain seperti dibawah ini :

The image shows a Java Swing window titled "Tanggal Lahir". Inside the window, there are three JComboBox components arranged horizontally. The first is labeled "Tanggal", the second "Bulan", and the third "Tahun". Each component has a small downward arrow on its right side, indicating it is a dropdown menu. The window has a standard Mac OS-style title bar with a red close button, a yellow maximize button, and a green minimize button.

Gambar 3. 15 Desain Latihan Input Tanggal Lahir ComboBox

- b. Isikan setiap Combo menggunakan code dengan perulangan control flow for atau do while yang diketahui

penjelasan :

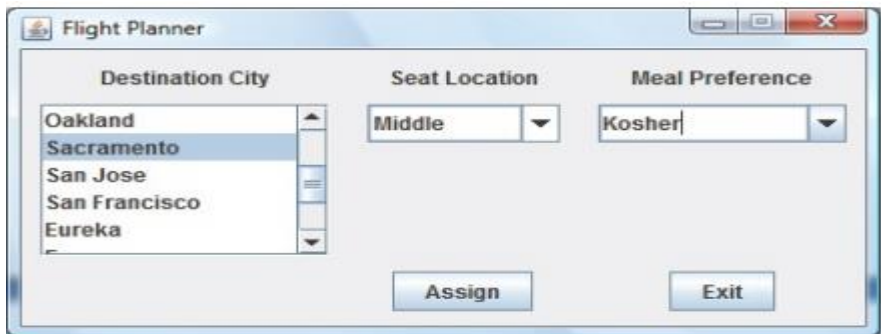
Keterangan :

- Tanggal
combo item terdiri angka mulai 1 s/d 31
- Bulan
combo item terdiri angka mulai 1 s/d 12
- Tahun Lahir
combo item terdiri angka mulai 2000 s/d 1900 (Descending)

2. Kontrol **jList** dan **jComboBox**

- a. Buat program untuk perencanaan penerbangan isian nya terdiri dari Destination City (Jlist) , Seat Location (jCombo), Meal Preference (jCombo

Tampilan Desain seperti gambar dibawah ini :



Gambar 3. 16 Desain JComboBox Jlist

- b. Isikan Item **Destination city** , **Seat Location** , **Meal Preference Destination city**

"Los Angeles", "Orange County", "Ontario", "Bakersfield",
"Oakland", "Sacramento", "San Jose" "San Francisco", "Eureka", "Eugene", "Portland", "Spokane". "Seattle"

Seat Location :

"Aisle", "Middle", "Window"

Meal Preference :

"Chicken", "Mystery Meat", "Kosher", "Vegetarian", "Fruit Plate", "No Preference"

Pengisian masing masing komponen boleh menggunakan Properties ataupun Code (Opsional) yang kalian ketahui.

D. REFERENSI

Philip Conrod & Lou Tylee. Learn Java™ GUI Applications A JFC Swing NetBeans Tutorial 8th Edition , Kidware Software, LLC, 2015

Danny Poo Derek Kiong Swarnalatha Ashok, Object-Oriented Second edition Programming and Java , Springer 2008

<https://docs.oracle.com/javase/7/docs/api/javax/swing/JFileChooser.html>, diakses pada tanggal 23 Juni 2022.

<https://www.javatpoint.com/java-bufferedreader-class>, diakses pada tanggal 23 Desember 2021.

https://www.homeandlearn.co.uk/java/java_check_boxes.html, diakses pada tanggal 23 Juni 2022.

Romi satrio Wahono, Java Gui, 2016