



# Komponen GUI Lanjutan

Praktikum Pemrograman II - 05



## JComboBox

JComboBox adalah salah satu komponen antarmuka grafis (GUI) di Java Swing yang digunakan untuk menampilkan daftar item yang dapat dipilih oleh pengguna. JComboBox adalah kombinasi antara kotak teks (textbox) dan daftar drop-down. Pengguna dapat memilih salah satu item dari daftar atau mengetikkan nilai secara manual, tergantung pada konfigurasi.

### Latihan 1

Buat kelas ComboBoxExample kemudian ketikkan kode berikut.

```
import javax.swing.*;
import java.awt.event.*;

public class ComboBoxExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JComboBox Example");
        String[] options = { "Option 1", "Option 2", "Option 3", "Option 4" };

        // Membuat JComboBox dengan opsi
        JComboBox<String> comboBox = new JComboBox<>(options);

        // Tambahkan pendengar aksi
        comboBox.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                // Mengambil item yang dipilih
                String selected = (String) comboBox.getSelectedItem();
                System.out.println("Selected: " + selected);
            }
        });

        // Atur layout dan tambahkan ke frame
        frame.setLayout(null);
        comboBox.setBounds(50, 50, 150, 20);
        frame.add(comboBox);

        // Konfigurasi frame
        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

## ComboBoxModel

ComboBoxModel adalah antarmuka di Java Swing yang digunakan oleh JComboBox untuk menangani data yang ditampilkan dalam bentuk daftar item. ComboBoxModel adalah perpanjangan dari antarmuka ListModel, yang digunakan untuk menangani model data di komponen-komponen berbasis

daftar (list-based components), seperti JList. ComboBoxModel memungkinkan kita untuk mengelola elemen-elemen yang ditampilkan dalam JComboBox, termasuk pemilihan elemen.

## Latihan 2

Buat kelas ComboBoxModelExample kemudian ketikkan kode berikut.

```
import javax.swing.*;
import java.awt.event.*;

public class ComboBoxModelExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("ComboBoxModel Example");

        // Membuat model dengan opsi
        DefaultComboBoxModel<String> comboBoxModel = new
DefaultComboBoxModel<>();
        comboBoxModel.addElement("Item 1");
        comboBoxModel.addElement("Item 2");
        comboBoxModel.addElement("Item 3");
        comboBoxModel.addElement("Item 4");

        // Membuat JComboBox dengan model
        JComboBox<String> comboBox = new JComboBox<>(comboBoxModel);

        // Tambahkan listener untuk JComboBox
        comboBox.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                // Mendapatkan item yang dipilih
                String selectedItem = (String) comboBox.getSelectedItem();
                System.out.println("Selected: " + selectedItem);
            }
        });

        // Mengatur layout dan menambahkan JComboBox ke JFrame
        frame.setLayout(null);
        comboBox.setBounds(50, 50, 150, 20);
        frame.add(comboBox);

        // Konfigurasi JFrame
        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

## JList

JList adalah salah satu komponen GUI (Graphical User Interface) di Java Swing yang digunakan untuk menampilkan daftar elemen. JList memungkinkan pengguna untuk memilih satu atau lebih elemen dari

daftar yang disajikan secara vertikal atau horizontal. Komponen ini mendukung daftar statis (tetap) atau dinamis, yang berarti elemen-elemen dalam daftar dapat ditambah, dihapus, atau dimodifikasi secara real-time.

### Latihan 3

Buat kelas JListExample kemudian ketikkan kode berikut.

```
import javax.swing.*;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;

public class JListExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JList Example");

        // Data untuk JList
        String[] items = { "Item 1", "Item 2", "Item 3", "Item 4", "Item 5" };

        // Membuat JList dengan data
        JList<String> list = new JList<>(items);

        // Mengatur mode seleksi (dalam hal ini SINGLE_SELECTION)
        list.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);

        // Menambahkan listener untuk menangani seleksi item
        list.addListSelectionListener(new ListSelectionListener() {
            public void valueChanged(ListSelectionEvent e) {
                if (!e.getValueIsAdjusting()) { // Mencegah pemicu ganda
                    String selectedItem = list.getSelectedValue();
                    System.out.println("Selected: " + selectedItem);
                }
            }
        });

        // Menambahkan JScrollPane untuk JList jika item terlalu banyak
        JScrollPane scrollPane = new JScrollPane(list);

        // Mengatur layout dan menambahkan komponen ke frame
        frame.setLayout(null);
        scrollPane.setBounds(50, 50, 150, 100);
        frame.add(scrollPane);

        // Konfigurasi frame
        frame.setSize(300, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

## Latihan 4

Buat kelas dengan nama `DynamicJListExample` kemudian ketikkan kode berikut.

```
import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class DynamicJListExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Dynamic JList Example");

        // Membuat DefaultListModel
        DefaultListModel<String> listModel = new DefaultListModel<>();
        listModel.addElement("Dynamic Item 1");
        listModel.addElement("Dynamic Item 2");
        listModel.addElement("Dynamic Item 3");

        // Membuat JList dengan model
        JList<String> list = new JList<>(listModel);
        list.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);

        // JScrollPane untuk JList
        JScrollPane scrollPane = new JScrollPane(list);

        // Tombol untuk menambahkan item baru
        JButton addButton = new JButton("Add Item");
        addButton.setBounds(50, 170, 100, 30);
        addButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                listModel.addElement("New Item");
            }
        });

        // Tombol untuk menghapus item yang dipilih
        JButton removeButton = new JButton("Remove Item");
        removeButton.setBounds(160, 170, 120, 30);
        removeButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                int selectedIndex = list.getSelectedIndex();
                if (selectedIndex != -1) {
                    listModel.remove(selectedIndex);
                }
            }
        });

        // Mengatur layout dan menambahkan komponen ke frame
        frame.setLayout(null);
        scrollPane.setBounds(50, 50, 150, 100);
        frame.add(scrollPane);
        frame.add(addButton);
        frame.add(removeButton);
    }
}
```

```

        // Konfigurasi frame
        frame.setSize(350, 250);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}

```

## Menampilkan HTML

Di Java Swing, komponen JLabel mendukung penggunaan HTML untuk memformat teks secara lebih kaya. Dengan menggunakan markup HTML, Anda bisa menambahkan gaya pada teks yang ditampilkan, seperti **teks tebal**, *miring*, warna, ukuran font, daftar, tautan, dan elemen-elemen HTML dasar lainnya. Ini sangat berguna jika Anda ingin menampilkan teks yang lebih menarik tanpa harus menggunakan cara manual untuk memformat setiap bagian teks.

### Latihan 5

Buat kelas JLabelHTMLExample kemudian ketikkan kode berikut.

```

import javax.swing.*;

public class JLabelHTMLExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JLabel HTML Example");

        // Membuat JLabel dengan teks HTML
        JLabel label = new JLabel("<html><b>Bold Text</b>, <i>Italic Text</i>,"
        and <u>Underlined Text</u></html>");

        // Menambahkan JLabel ke JFrame
        frame.add(label);

        // Konfigurasi frame
        frame.setSize(300, 100);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}

```

## JTable

JTable adalah komponen dalam Java Swing yang digunakan untuk menampilkan dan mengelola data dalam bentuk tabel. Tabel ini terdiri dari kolom dan baris, di mana setiap baris mewakili suatu entitas (misalnya, sebuah objek atau data) dan setiap kolom mewakili atribut atau properti dari entitas tersebut. JTable sangat fleksibel dan dapat disesuaikan untuk memenuhi berbagai kebutuhan seperti pengeditan, pengurutan, atau penggambaran data yang lebih kompleks.

Seperti JComboBox dan JList, JTable juga memiliki model yang direpresentasikan dengan antarmuka TableModel. Antarmuka TableModel mengatur bagaimana data tabel dikelola dan diakses oleh JTable. Implementasi bawaan dari model ini adalah DefaultTableModel, yang memungkinkan Anda menambahkan, menghapus, dan memodifikasi data secara langsung.

## Latihan 6

Buat kelas dengan nama JTableExample kemudian ketikkan kode berikut.

```
import javax.swing.*;
import javax.swing.table.DefaultTableModel;

public class JTableExample {
    public static void main(String[] args) {
        // Membuat frame
        JFrame frame = new JFrame("JTable Example");

        // Membuat tabel dengan data awal dan header kolom
        String[] columnNames = { "ID", "Name", "Age" };
        Object[][] data = {
            { 1, "John", 25 },
            { 2, "Anna", 30 },
            { 3, "Mike", 35 },
        };

        // Membuat model tabel
        DefaultTableModel model = new DefaultTableModel(data, columnNames);

        // Membuat JTable dengan model
        JTable table = new JTable(model);

        // Membuat JScrollPane untuk menampung JTable
        JScrollPane scrollPane = new JScrollPane(table);

        // Menambahkan JScrollPane ke frame
        frame.add(scrollPane);

        // Konfigurasi frame
        frame.setSize(400, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

## Latihan 7

Berikut ini adalah contoh tabel yang dapat diedit. Buat kelas bernama JTableEditableExample kemudian ketikkan kode berikut.

```

import javax.swing.*;
import javax.swing.table.DefaultTableModel;

public class JTableEditableExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JTable Editable Example");

        String[] columnNames = { "ID", "Name", "Age" };
        Object[][] data = {
            { 1, "John", 25 },
            { 2, "Anna", 30 },
            { 3, "Mike", 35 },
        };

        // Membuat DefaultTableModel dengan data dan header
        DefaultTableModel model = new DefaultTableModel(data, columnNames) {
            @Override
            public boolean isCellEditable(int row, int column) {
                // Hanya kolom 'Name' yang dapat diedit
                return column == 1;
            }
        };

        JTable table = new JTable(model);
        JScrollPane scrollPane = new JScrollPane(table);
        frame.add(scrollPane);

        frame.setSize(400, 200);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}

```

## Latihan 8

Berikut ini adalah contoh penambahan dan penghapusan baris pada tabel. Buatlah kelas dengan nama `JTableAddRemoveRowExample` kemudian ketikkan kode berikut.

```

import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class JTableAddRemoveRowExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JTable Add/Remove Row Example");

        String[] columnNames = { "ID", "Name", "Age" };
        DefaultTableModel model = new DefaultTableModel(columnNames, 0); // Model
        dengan 0 baris awal

        JTable table = new JTable(model);
    }
}

```



```

// Tombol untuk menambahkan baris
JButton addButton = new JButton("Add Row");
addButton.setBounds(50, 220, 100, 30);
addButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        // Data baru untuk ditambahkan
        Object[] newRow = { model.getRowCount() + 1, "New Name", 20 };
        model.addRow(newRow);
    }
});

// Tombol untuk menghapus baris yang dipilih
JButton removeButton = new JButton("Remove Row");
removeButton.setBounds(160, 220, 120, 30);
removeButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        int selectedRow = table.getSelectedRow();
        if (selectedRow != -1) {
            model.removeRow(selectedRow);
        }
    }
});

// JScrollPane untuk JTable
JScrollPane scrollPane = new JScrollPane(table);
scrollPane.setBounds(20, 20, 300, 180);

// Menambahkan komponen ke JFrame
frame.setLayout(null);
frame.add(scrollPane);
frame.add(addButton);
frame.add(removeButton);

frame.setSize(400, 300);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setVisible(true);
}
}

```

## Latihan 9

Berikut ini adalah contoh tabel yang dapat diurutkan. Buatlah kelas dengan nama `JTableSortExample` kemudian ketikkan kode berikut.

```

import javax.swing.*.*;
import javax.swing.table.DefaultTableModel;

public class JTableSortExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JTable Sorting Example");
    }
}

```

```

String[] columnNames = { "ID", "Name", "Age" };
Object[][] data = {
    { 1, "John", 25 },
    { 3, "Mike", 35 },
    { 2, "Anna", 30 },
};

DefaultTableModel model = new DefaultTableModel(data, columnNames);
JTable table = new JTable(model);

// Mengaktifkan pengurutan otomatis
table.setAutoCreateRowSorter(true);

JScrollPane scrollPane = new JScrollPane(table);
frame.add(scrollPane);

frame.setSize(400, 200);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setVisible(true);
}
}

```