

## Seri Interaksi User

# Command, List dan Interaksi User

**Chocolove Mic**

[chocolove\\_mic@yahoo.co.uk](mailto:chocolove_mic@yahoo.co.uk)

<http://www.mygiserver.com/~chocolove2003>

### **Lisensi Dokumen:**

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di **IlmuKomputer.Com** dapat digunakan, dimodifikasi dan disebarluaskan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari **IlmuKomputer.Com**.

Pada tulisan ini akan dibahas tentang penggunaan class Command dan tip-trik penggunaan class ini dalam sebuah aplikasi.

## Menambahkan Command

Objek yang dibentuk dari class Command dapat ditambahkan pada banyak objek, misalnya objek yang dibuat dari class Form, class TextBox, class Canvas dan lain-lain. Berikut ini objek-objek yang dibentuk dari class Command akan ditambahkan ke dalam objek yang dibentuk dari class Form.

### **CommandMidlet01.java**

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class CommandMidlet01 extends MIDlet implements CommandListener {

    private Command cmdKeluar;
    private Command cmdAlert;
    private Display display;
    private Form form;
    private Alert alert;

    public CommandMidlet01() {
        form = new Form("Aplikasi 01");
        cmdKeluar = new Command("Keluar", Command.EXIT, 1);
        cmdAlert = new Command("Alert", Command.SCREEN, 2);

        form.addCommand(cmdAlert);
        form.addCommand(cmdKeluar);
        form.setCommandListener(this);
    }

    public void startApp() {
        display = Display.getDisplay(this);
    }
}
```

```
        display.setCurrent(form);
    }

    public void pauseApp() {
}

    public void destroyApp(boolean unconditional) {
}

    public void commandAction(Command c, Displayable d) {
        if (d == form) {
            if (c == cmdKeluar) {
                destroyApp(false);
                notifyDestroyed();
            }
            else {
                alert = new Alert("Alert..!!!!");
                alert.setString("Tombol Alert ditekan... ");
                display.setCurrent(alert);
            }
        }
    }
}
```

Sumber : -

Berikut adalah keluaran dari contoh aplikasi di atas.

Di bawah ini adalah tampilan pertama ketika aplikasi dijalankan.



Sedangkan apabila tombol untuk Alert ditekan maka bisa dilihat tampilan seperti berikut di samping kanan ini.



## Command dan List

Tahap selanjutnya akan dibuat aplikasi yang didalamnya terdapat perintah (command) untuk keluar aplikasi dan untuk memanggil menu yang merupakan objek dari List. Pada menu terdapat item-item yang akan memberikan aksi berupa alert yang merupakan objek yang dibentuk dari class Alert. Selain itu juga terdapat aksi yang mengembalikan ke tampilan awal aplikasi.

Berikut adalah kode dari aplikasi yang telah dideskripsikan di atas.

```
CommandMidlet02.java
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class CommandMidlet02 extends MIDlet implements CommandListener {

    private Display display;
    private Command cmdMenu;
    private Command cmdKeluar;
    private TextBox textBox;
    private List mainMenu;

    public CommandMidlet02() {
        textBox = new TextBox("Aplikasi 02", "Ini Command Midlet 02", 256, 0);
    }

    protected void startApp() {
        display.setCurrent(textBox);
    }

    protected void pauseApp() {
    }

    protected void destroyApp(boolean unconditional) {
    }

    public void commandAction(Command c, Displayable d) {
        if (d == textBox) {
            if (c == cmdMenu) {
                mainMenu = new List("Main Menu", List.IMPLICIT);
                mainMenu.add("Exit");
                mainMenu.add("About");
                mainMenu.add("Help");
                mainMenu.setCommandListener(this);
                display.setCurrent(mainMenu);
            }
            else if (c == cmdKeluar) {
                destroyApp(false);
                notifyDestroyed();
            }
        }
    }

    public void commandSelected(Command c) {
        if (c == mainMenu.getCommand("Exit")) {
            destroyApp(true);
        }
        else if (c == mainMenu.getCommand("About")) {
            alert = new Alert("About..!!!!");
            alert.setString("Aplikasi 02");
            alert.setTimeout(Alert.FOREVER);
            display.setCurrent(alert);
        }
        else if (c == mainMenu.getCommand("Help")) {
            alert = new Alert("Help..!!!!");
            alert.setString("Aplikasi 02");
            alert.setTimeout(Alert.FOREVER);
            display.setCurrent(alert);
        }
    }

    public void commandDeselected(Command c) {
    }
}
```

```
cmdMenu = new Command("Menu", Command.SCREEN, 1);
cmdKeluar = new Command("Keluar", Command.SCREEN, 1);

textBox.addCommand(cmdMenu);
textBox.addCommand(cmdKeluar);
textBox.setCommandListener(this);
}

public void startApp() {
    display = Display.getDisplay(this);
    display.setCurrent(textBox);
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) {
}

public void commandAction (Command c, Displayable d) {
    if (d == textBox) {
        if (c == cmdKeluar) {
            destroyApp(false);
            notifyDestroyed();
        }
        else if (c == cmdMenu) {
            mainMenu = new List("Menu Aplikasi 02", List.IMPLICIT);
            mainMenu.append("Link 1", null);
            mainMenu.append("Link 2", null);
            mainMenu.append("Kembali", null);

            mainMenu.setCommandListener(this);
            display.setCurrent(mainMenu);
        }
    }
    else if (d == mainMenu) {
        if (mainMenu.isSelected(0)) {
            // Ketik operasi yang diinginkan
            // ketika tombol untuk Link1 ditekan
        }
        else if (mainMenu.isSelected(1)) {
            // Ketik operasi yang diinginkan
            // ketika tombol untuk Link2 ditekan
        }
        else if (mainMenu.isSelected(2)) {
            display.setCurrent(textBox);
        }
    }
}
}
```

Sumber : -

Berikut adalah tampilan pertama dari aplikasi dan ketika tombol untuk Menu ditekan.

