



# Pemrograman Visual

MATISSE BUILDER 2  
(ACTION)

# Materi Pertemuan 3

- Penggunaan Matisse Builder
- Action Listener Komponen Swing

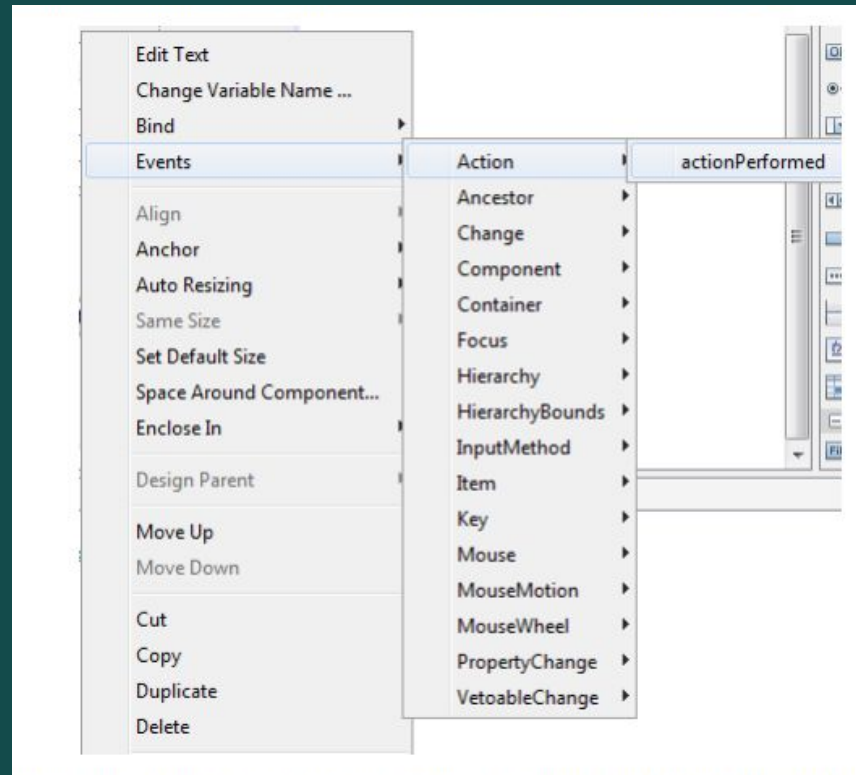
# EVENT LISTENER

- Untuk mendapatkan kesan “responsif” terhadap input user, aplikasi gui yang dibangun menggunakan bahasa java dilengkapi oleh event handler. Event handler didapat dari interface listener. Interface listener mempunyai method yang harus dituliskan kembali sebagai method yang harus dijalankan saat mendapat input user.
- Dengan menggunakan matisse builder, penambahan event handler menjadi lebih mudah. Tambahkan event dengan meng-klik kanan komponen yang akan ditambahkan event, lalu pilih event apa yang diinginkan (biasanya tipe action dengan method actionPerformed()).

# EVENT LISTENER

- Atau bisa juga dengan meng-klik 2x komponen yang ingin ditambahkan event. Untuk tombol (JButton), event yang terbentuk adalah action → actionPerformed(). Akan muncul bagian source code pada gui seperti berikut. Logika pemrograman dapat dituliskan di method tersebut.

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    //tuliskan logika pemrograman di blok ini  
}
```



# METHOD KOMPONEN SWING

- Beberapa komponen swing memiliki method yang bisa digunakan untuk dipakai pada logika pemrograman yang terdapat pada event handler. Beberapa yang paling sering digunakan adalah sebagai berikut:

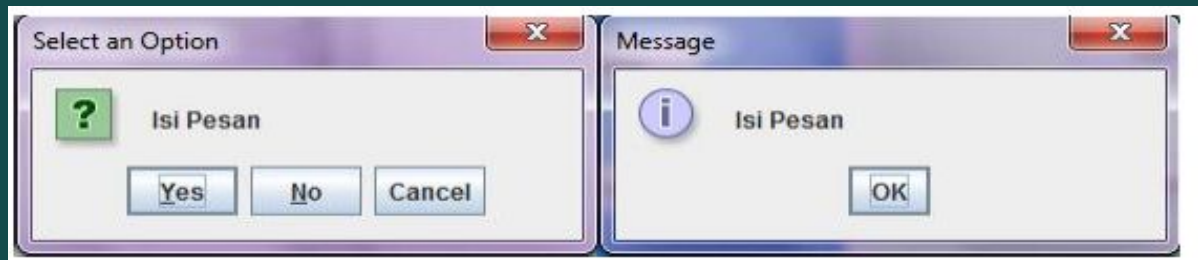
Method	Komponen	Fungsi
setText("X")	JButton, JTextfield, JLabel	Menge-set tulisan yang muncul pada masing-masing komponen
getText()	JButton, JTextfield, JLabel	Mengambil nilai String yang terdapat pada masing-masing komponen
setToolTipText("X")	JButton, JTextfield, JLabel	Memberikan tooltip pada komponen
setEnabled(false)	JButton, JTextfield, JLabel, JComboBox, JRadioButton, JCheckBox	Meng-enable suatu komponen atau tidak (dapat di-klik atau tidak)
setSelected(true)	JRadioButton, JCheckBox	Membuat masing-masing komponen terpilih atau tidak
isSelected()	JRadioButton, JCheckBox	Menge-cek apakah suatu komponen sedang terpilih atau tidak
setSelectedIndex(4)	JComboBox	Membuat indeks pada angka tertentu sebagai komponen terpilih
getSelectedIndex()	JComboBox	Mengambil indeks terpilih dari komponen. Indeks dimulai dari 0
getSelectedItem()	JComboBox	Mengambil objek terpilih dari komponen. Dapat langsung ditampilkan jika yang terpilih adalah objek String.

insertItemAt(objek, indek)	JComboBox	Memasukkan item pilihan berupa sebuah objek bertipe Object pada index ke indek
setValueAt(Objek, baris, kolom)	JTable	Menge-set table dengan nilai Objek yang bertipe data Object pada baris dan kolom tertentu
getSelectedRow()	JTable	Mengambil indeks baris tabel terpilih. Indeks dimulai dari 0
getSelectedColumn()	JTable	Mengambil indeks kolom tabel terpilih. Indeks dimulai dari 0



# PENGGUNAAN DIALOG (JOPTIONPANE)

- Dialog di modul ini merujuk pada sebuah komponen yang muncul sebagai informasi atau peringatan bahkan meminta input user setelah melakukan sesuatu



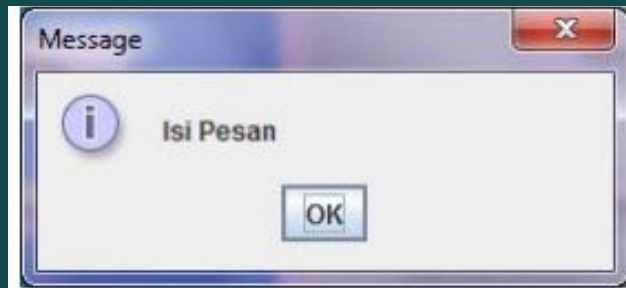
Pada komponen swing, komponen ini bernama JOptionPane. Terdapat 4 buah option pane: a) `showConfirmDialog` : Untuk memberi konfirmasi yes/no/cancel b) `showInputDialog`: Untuk menampilkan pop-up sebagai input data c) `showMessageDialog`: Untuk menampilkan status/pesan dengan 1 tombol “OK”. d) `showOptionDialog`: Gabungan dari ketiga komponen di atas. Ada kasi konfirmasi, meminta input dan menampilkan status/pesan

Pada pertemuan ini hanya dibahas 2 tipe option pane, `showConfirmDialog` dan `showMessageDialog`. Setiap option pane dapat dibuat objeknya dengan menggunakan jumlah parameter yang berbeda-beda. Hal ini akan berpengaruh pada tampilan option pane yang dihasilkan.

# showMessageDialog

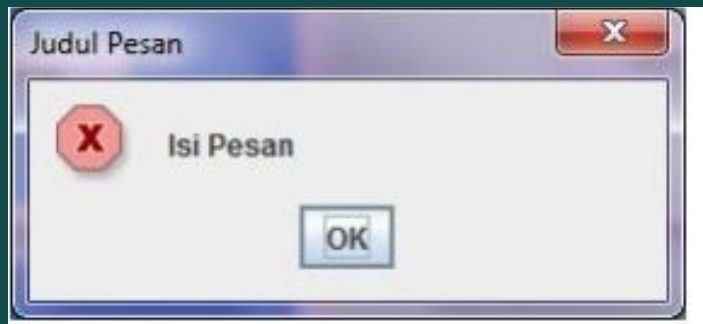
```
JOptionPane.showMessageDialog(null, "Isi Pesan");
```

Hasil:







Contoh Kode showMessageDialog dengan 4 parameter:

Hasil:



Lambang/symbol/icon yang dihasilkan (silang merah atau huruf “i” dalam lingkaran), tergantung pada parameter terakhir dari contoh kode di atas (untuk kode 4 parameter). Parameter ini disebut “option type”. Jenis-jenis dari option type adalah sebagai berikut:

Icon	Code	IDE Value
No icon	JOptionPane.PLAIN_MESSAGE	-1
	JOptionPane.ERROR_MESSAGE	0
	JOptionPane.INFORMATION_MESSAGE	1
	JOptionPane.WARNING_MESSAGE	2
	JOptionPane.QUESTION_MESSAGE	3

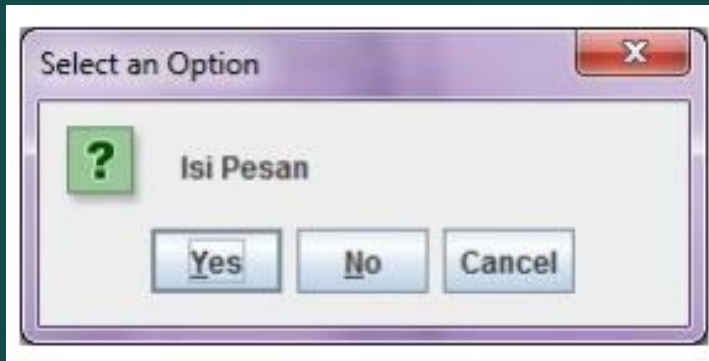


# showConfirmDialog

Contoh Kode showMessageDialog dengan 2 parameter:

```
JOptionPane.showConfirmDialog(null, "Isi Pesan");
```

Hasil:



Untuk mengambil nilai dari setiap tombol yang ditekan, gunakan kode sebagai berikut

```
//meminta input yang ditekan sekaligus deklarasi option pane
int hasil = JOptionPane.showConfirmDialog(null, "Isi Pesan");

//membandingkan hasil yang diterima dengan konstanta yang dimiliki option pane
if(hasil == JOptionPane.YES_OPTION){
    System.out.println("Yes!");
}else if(hasil == JOptionPane.NO_OPTION){
    System.out.println("No!");
}else if(hasil == JOptionPane.CANCEL_OPTION){
    System.out.println("Cancel!");
}
```

Pembanding pesan terdiri dari:

- YES\_OPTION: Jika memilih tombol “YES”
- OK\_OPTION: Jika memilih tombol “OK”
- NO\_OPTION: Jika memilih tombol “NO”
- CLOSED\_OPTION: Jika memilih tombol “CLOSED”
- CANCEL\_OPTION: Jika memilih tombol “CANCEL”

## CONTOH :

Nim

Nama

Tampilkan

Nim

Nama

jLabel5

jLabel6

```
Code

public void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    Nim = txtnim.getText();
    Nama = txtnama.getText();
    int jawab = JOptionPane.showConfirmDialog(this, "Apakah data ingin di tampilkan?");
    switch (jawab) {
        case JOptionPane.YES_OPTION:
            lblnim.setText(Nim);
            lblnama.setText(Nama);
            break;
        case JOptionPane.NO_OPTION:
            JOptionPane.showMessageDialog(this, "Data tidak ingin ditampilkan");
            break;
        case JOptionPane.CANCEL_OPTION:
            JOptionPane.showMessageDialog(this, "Kamu membatalkan utk ditampilkan");
    }
}
```

# LATIHA N

The screenshot shows a Java Swing window titled "Mine Nakahara Corps". Inside the window, there is a tabbed interface with two tabs: "Pilihan" (selected) and "Kosongkan". The main title of the application is "Aplikasi Perhitungan Biaya Kamar". The form contains the following fields and controls:

- Nama Penyewa:** A text input field.
- Alamat Asal:** A text input field.
- Periode:** Two radio buttons labeled "Tahunan" and "Bulanan".
- Cicilan:** A dropdown menu with the text "--Pilihan--" and a downward arrow.
- Fasilitas:** Three checkboxes labeled "Dispenser", "Kipas Angin", and "Televisi".
- Konfirmasi:** A large, light gray button with the text "Konfirmasi" in bold black font.

Buatlah aplikasi berdasarkan GUI di atas dan ketika tombol “Hitung Biaya” ditekan, maka muncul pesan konfirmasi sebagai berikut:

1. Jika Yes :

Nama Penyewa :

Alamat Asal :

Periode:

Cicilan :

Fasilitas :

2. Jika No :

“Konfirmasi tidak  
ditampilkan”

3. Jika Cancel:

“Konfirmasi dibatalkan”