

Komponen dalam Delphi

Kegunaan, serta Pemanfaatan Properti, Event & Metodenya

Komponen merupakan objek di Delphi dalam bentuk elemen-elemen yang diimplementasikan dengan paket simbol/icon yang terdapat pada *Window Component Palette*. Komponen-komponen ini (termasuk form) dapat dimanipulasi sesuai kebutuhan. Setiap komponen memiliki properti, event, dan metode yang terdapat di dalamnya. Properti tiap komponen ada yang tunggal dan ada yang majemuk, properti yang majemuk (terdiri dari sub properti) ditandai dengan tanda plus (+) di depan item properti pada *Window Object Inspector* untuk menampilkannya klik ganda pada item properti yang bersangkutan.

Secara umum komponen dalam Delphi terbagi dua yaitu komponen visual dan komponen non visual.

→ Komponen Visual

Komponen ini memiliki sifat WYSIWYG (What You See Is What You Get) yaitu tampak sama pada saat didesain dan pada saat aplikasi dijalankan.

Jumlah komponen tergantung pada seberapa banyak icon yang terdapat di *Window Component Palette* pada halaman (kelompoknya) masing-masing. Karena sangat banyaknya komponen maka saat ini kita mempelajari hanya secara garis besarnya saja, hanya komponen-komponen yang sering digunakan dalam pembuatan aplikasi, yang diantaranya :

) Komponen untuk menampung teks, meliputi komponen Label, Edit, MaskEdit, ListBox, ComboBox, Memo.

- ◆ Komponen Label ◇  (Standard)
Label berguna untuk menjelaskan objek lain dalam sebuah aplikasi. Biasanya sebuah Label diletakkan berdekatan dengan objek yang diterangkan seperti Edit, Memo, atau sekelompok Check Box dan komponen lain sejenisnya.

- ◆ Komponen Edit ◇  (Standard)
Edit biasanya digunakan untuk memasukkan data (sebagai media masukan) pada suatu formulir transaksi, yang nantinya bisa diproses oleh aplikasi (disimpan ke database, atau yang lainnya).

- ◆ Komponen MaskEdit ◇  (Additional)
Komponen ini merupakan bentuk lain dari komponen Edit, dengan memiliki properti khusus yang digunakan untuk membuat suatu

masukan yang sesuai dengan format yang dikehendaki pada properti EditMask-nya.

- ◆ Komponen ListBox  (Standard)
Komponen ini berupa sebuah daftar pilihan. Setiap pilihan di dalamnya dapat ditambah atau dihilangkan secara dinamis sehingga berguna untuk menampilkan data yang selalu berubah.

- ◆ Komponen ComboBox  (Standard)
Komponen ini mirip dengan komponen ListBox, fungsinya juga untuk menampilkan pilihan. Namun ComboBox membutuhkan ruang yang lebih kecil dan hanya menunjukkan sebuah pilihan setiap saat dan jika diklik tanda panah ke bawah pada objeknya akan ditampilkan daftar (itemnya) yang dimilikinya.

- ◆ Komponen Memo  (Standard)
Komponen ini hampir sama fungsinya dengan komponen Edit, namun Memo bisa menampung lebih banyak karakter sebagai suatu masukan.

- } Komponen String-Grid.  (Additional)
Komponen ini berbentuk sekumpulan kisi-kisi (seperti tabel) string. Komponen ini dapat digunakan untuk membuat simulasi sebuah *spreadsheets* atau tombol-tombol tekan pada telepon/calculator. Tiap kisi dapat diakses dengan memakai properti Cells yang berupa array dua dimensi bertipe string. Properti cells ini didefinisikan nilainya saat eksekusi.

- } Komponen Button, Check-Box, RadioButton  (Standard)
Ketiga komponen komponen ini sering digunakan dalam aplikasi-aplikasi Windows. Komponen-komponen tersebut menampilkan informasi secara visual.

Komponen lain yang mirip komponen button, yaitu BitBtn (Button Grafis)

-  (Additional) dan SpeedButton  (Additional)

- } Komponen yang mengelompokkan komponen lain
Fungsinya sebagai pengelola komponen-komponen lain, diantaranya :

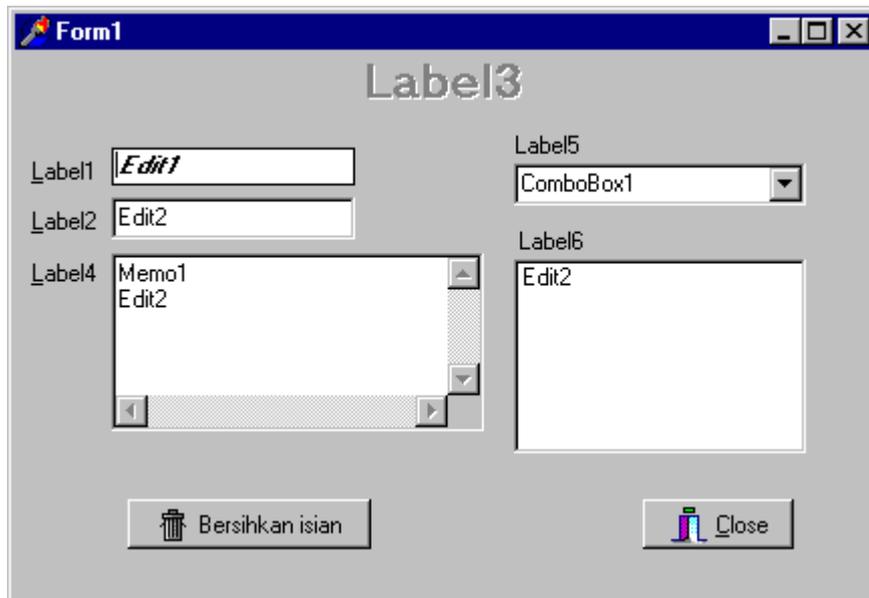
- ◆ Komponen GroupBox  (Standard)
Untuk mengelompokkan komponen CheckBox.

- ◆ Komponen RadioGroup  (Standard)
Untuk mengelompokkan komponen RadioButton.
 - ◆ Komponen Panel  (Standard)
Untuk mengelompokkan sembaran komponen visual.
 - ◆ Komponen TabControl  (Win32)
Untuk megelompokkan beberapa komponen seperti icon atau simbol.
 - ◆ Komponen PageControl  (Win32)
Hampir sama dengan TabControl namun pada saat desain dapat dilihat dan diatur/ditentukan langsung halaman yang dikehendaki
- } Komponen Scroll-Bar  (Standard) dan komponen Scroll-Box.  (Addiotional).
- } Komponen grafis seperti komponen Image, Shape, dan Bevel.
- Komponen Image  (Additional)
 - Komponen Shape  (Additional)
 - Komponen Bevel  (Additional)

Contoh penggunaan komponen-komponen di atas :

1. Pemanfaatan komponen label, edit, dan lainnya.

a) Susun beberapa komponen agar tampak seperti tampilan di bawah ini :



b) Kemudian atur properti dan event untuk masing-masing komponen yang telah diletakkan di form dengan menuliskan beberapa baris statement pada *Window Code Editor*, diantaranya adalah sebagai berikut :

Program di atas diharapkan :

⊗ Ketika Label1 diklik atau ditekan tombol kombinasi ALT+L maka focus/cursor edit berada pada komponen Edit1, demikian juga untuk label yang lain memfokuskan cursor edit ke komponen yang bersangkutan.

- ♣ Untuk menangkap penekanan kombinasi tombol ALT+L pada properti **FocusControl** komponen Label1 diisi **Edit1**.
- ♣ Untuk menangkap klik mouse pada event **OnClick** komponen Label1 diisi baris statemen sbb:

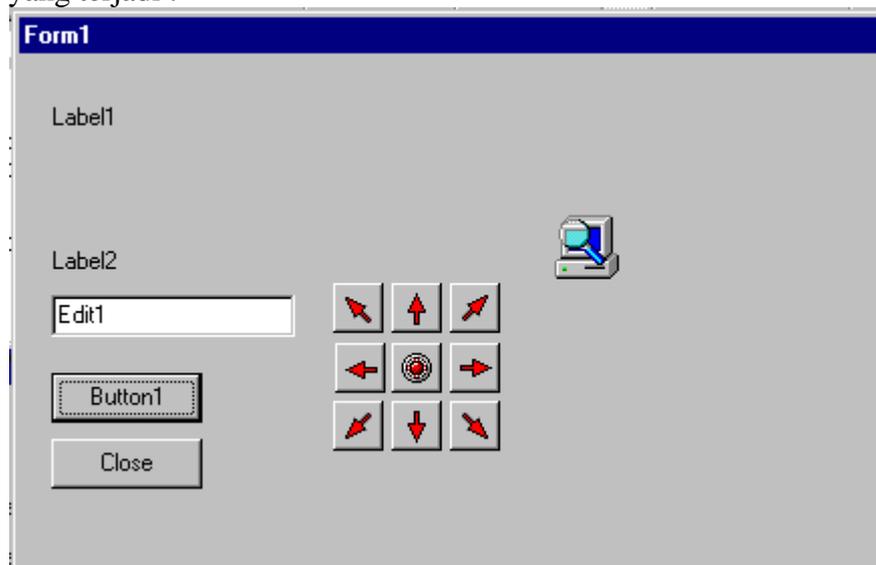
```
procedure TForm1.Label1Click(Sender: TObject);
begin
  (Sender as TLabel).FocusControl.SetFocus; //yg ditulis baris ini saja
end;
```

Coba terapkan untuk komponen label yang lainnya.

- ⊗ Ketika fokus/kursor edit keluar dari Label2 atau Edit2 maka memo1, ComboBox, dan ListBox itemnya bertambah. Pada event **OnExit** komponen **Edit2** tulis baris statemen berikut :

```
procedure TForm1.Edit2Exit(Sender: TObject);
begin
  Memo1.Lines.Add(Edit2.Text);
  Listbox1.Items.Add(Edit2.Text);
  Combobox1.Items.Add(Edit2.Text);
end;
```
- ⊗ Ketika **BitBtn1** (Bersihkan isian) diklik maka isian akan bersih/kosong kembali, tulis beberapa baris statemen berikut pada event **OnClick** komponen **BitBtn1**.
- ⊗ Ketika **BitBtn2** (Close) diklik maka program aplikasi selesai (keluar). Gunakan komponen BitBtn isi properti Kind dengan **bkClose**;
- ⊗ Lengkapi properti dan event pada komponen-komponen lainnya.
- ⊗ Jalankan program dan amati hasilnya.

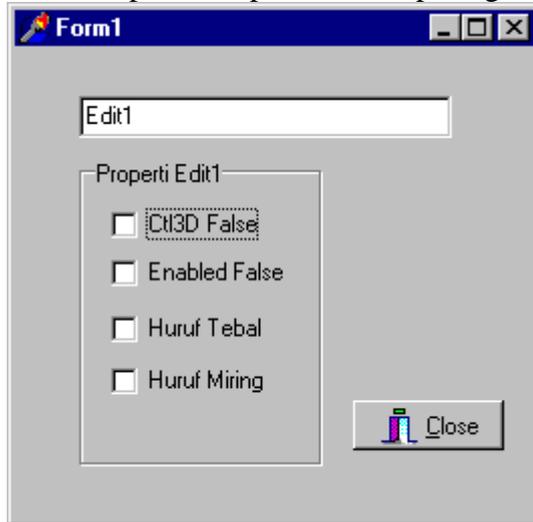
2. Buatlah aplikasi animasi seperti di bawah ini, pikirkan dan amati apa yang terjadi :



3. Untuk lebih memahami penggunaan komponen beserta properti dan event-nya silahkan berkreasi.

1. Contoh aplikasi lainnya :

Buatlah aplikasi seperti terlihat pada gambar di bawah ini :



Mengenai Program di atas :

- Program di atas memanfaatkan/memanipulasi properti milik komponen Edit1, dengan memberikan tanda check pada checkbox maka properti Edit1 dapat berubah.
- Cuplikan listing programnya :
- Untuk **CheckBox1.OnClick** (event OnClick pada checkbox) :

```
procedure TForm1.CheckBox1Click(Sender: TObject);
begin
  if CheckBox1.Checked=True then
    Edit1.Ctl3D:=False
  else
    Edit1.Ctl3D:=True;
end;
```

Untuk **CheckBox2.OnClick** :

```
procedure TForm1.CheckBox2Click(Sender: TObject);
begin
  if CheckBox2.Checked=True then
    Edit1.Enabled := False
  else
    Edit1.Enabled:=True;
end;
```

Untuk **CheckBox3.OnClick** :

```
procedure TForm1.CheckBox3Click(Sender: TObject);
begin
  if CheckBox3.State = cbChecked then
```

```

Edit1.Font.Style:=Edit1.Font.Style+[fsBold]
else
Edit1.Font.Style:=Edit1.Font.Style-[fsBold];
end;

```

Untuk **CheckBox4.OnClick** :

```

procedure TForm1.CheckBox4Click(Sender: TObject);
begin
if CheckBox4.State = cbChecked then
Edit1.Font.Style:=Edit1.Font.Style+[fsItalic]
else
Edit1.Font.Style:=Edit1.Font.Style-[fsItalic];
end;

```

2. Contoh aplikasi lainnya :

Buat desain pada form sesuai gambar di bawah ini, dan kemudian tulis kode programnya (disediakan) pada event tiap komponen masing-masing.



Gambaran program di atas :

- Program menerima masukan dari komponen **EditNamaMahasiswa**
- Jika tombol **Teknik** ditekan maka **ListboxTeknik** akan bertambah
- Jika tombol **Non Teknik** ditekan maka **ListboxNonTeknik** akan bertambah
- Jika **simbol panah kanan** ditekan maka isi **ListboxTeknik** yang dipilih akan pindah ke **ListboxNonTeknik**.
- Jika tombol panah kiri ditekan maka isi **ListboxNonTeknik** yang dipilih akan pindah ke **ListboxTeknik**.
- Tulislah program sesuai dengan event pada komponen masing-masing.
- Cuplikan listing programnya :

```

procedure TForm1.ButtonTeknikClick(Sender: TObject);
begin
  if EditNamaMahasiswa.Text<>" then
    begin
      ListBoxTeknik.Items.Add(EditNamaMahasiswa.Text);
      EditNamaMahasiswa.Clear;
    end
  Else
    begin
      ShowMessage('Nama Mahasiswa diisi dulu !');
      EditNamaMahasiswa.SetFocus;
    end;
end;

procedure TForm1.ButtonNonTeknikClick(Sender: TObject);
begin
  if EditNamaMahasiswa.Text<>" then
    begin
      ListBoxNonTeknik.Items.Add(EditNamaMahasiswa.Text);
      EditNamaMahasiswa.Clear;
    end
  Else
    begin
      ShowMessage('Nama Mahasiswa diisi dulu !');
      EditNamaMahasiswa.SetFocus;
    end;
end;

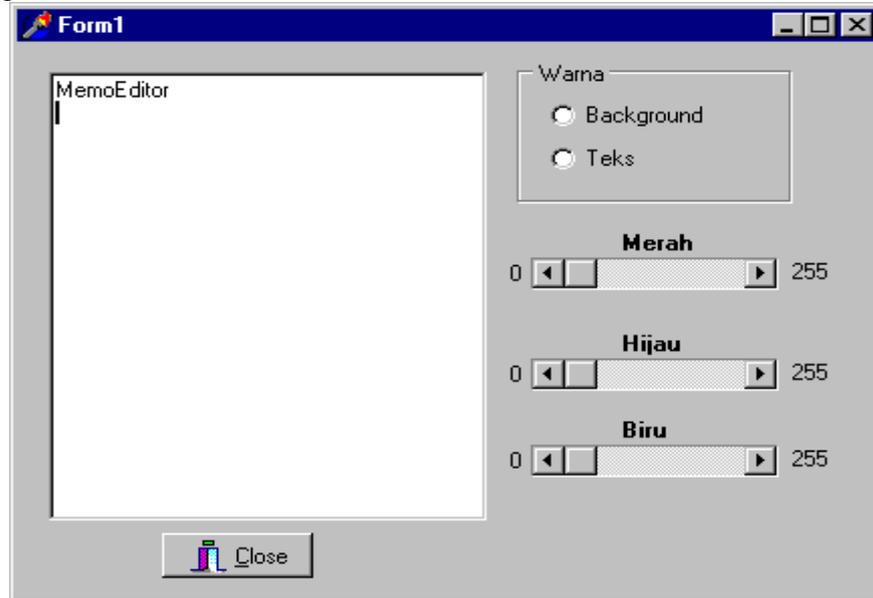
procedure TForm1.SpeedButtonPindahKananClick(Sender: TObject);
begin
  if ListBoxTeknik.ItemIndex >=0 then
    begin
      ListBoxNonTeknik.items.Add(ListBoxTeknik.Items[ListBoxTeknik.ItemIndex]);
      ListBoxTeknik.Items.Delete(ListBoxTeknik.ItemIndex);
    end
  else
    ShowMessage('Belum dipilih/Kosong tuh !');
end;

procedure TForm1.SpeedButtonPindahKiriClick(Sender: TObject);
begin
  if ListBoxNonTeknik.ItemIndex >=0 then
    begin
      ListBoxTeknik.items.Add(ListBoxNonTeknik.Items[ListBoxNonTeknik.ItemIndex]);
      ListBoxNonTeknik.Items.Delete(ListBoxNonTeknik.ItemIndex);
    end
  else
    ShowMessage('Belum dipilih/Kosong tuh !');
end;

```

3. Contoh program lainnya :

- Desainlah komponen-komponen yang sesuai pada form baru seperti gambar berikut :



- Buatlah programnya sesuai listing program berikut ini, sesuaikan dengan event pada tiap-tiap komponen yang bersangkutan.

- Listing programnya :

```
procedure TForm1.Warna;
var R, G, B: integer;
begin
  R:= ScrollBarMerah.Position;
  G:= ScrollBarHijau.Position;
  B:= ScrollBarBiru.Position;

  if RadioGroupWarna.ItemIndex=0 then
    MemoEditor.Color:=RGB(R,G,B);
  if RadioGroupWarna.ItemIndex=1 then
    MemoEditor.Font.Color:=RGB(R,G,B);
end; //

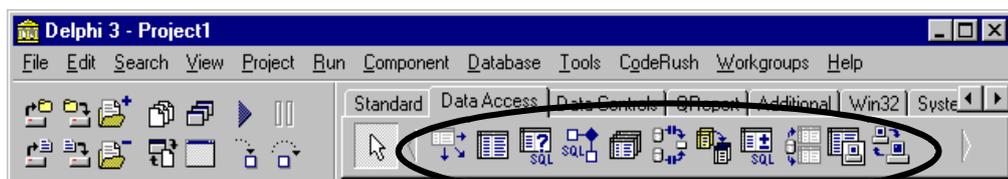
procedure TForm1.ScrollBarMerahChange(Sender: TObject);
begin
  Warna;
end;

procedure TForm1.ScrollBarHijauChange(Sender: TObject);
begin
  Warna;
end;

procedure TForm1.ScrollBarBiruChange(Sender: TObject);
begin
  Warna;
end;
```

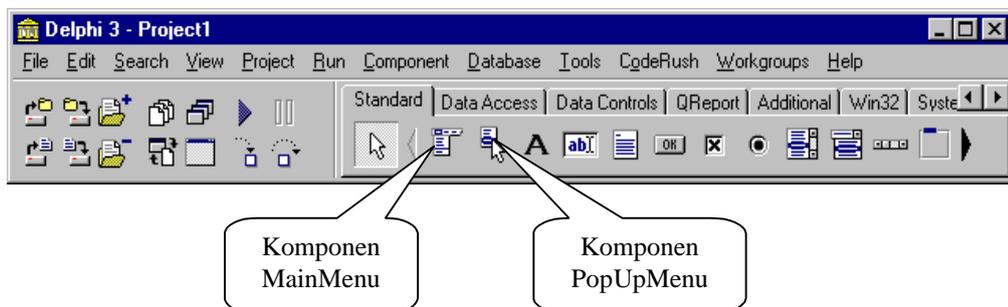
↳ Komponen Non Visual

Komponen ini hanya tampak saat desain saja sedangkan pada saat program dijalankan tidak tampak. Komponen-komponen ini tidak dapat berdiri sendiri tetapi harus dilekatkan pada sebuah form. Saat bekerja dengan komponen-komponen ini seringkali dibutuhkan penulisan kode program agar dapat menggunakan fasilitas setiap komponen. Beberapa komponen non visual memiliki fasilitas khusus dalam perancangannya, seperti pada komponen **MainMenu**, **PopUpMenu**, dan komponen **Query** serta komponen database lainnya.



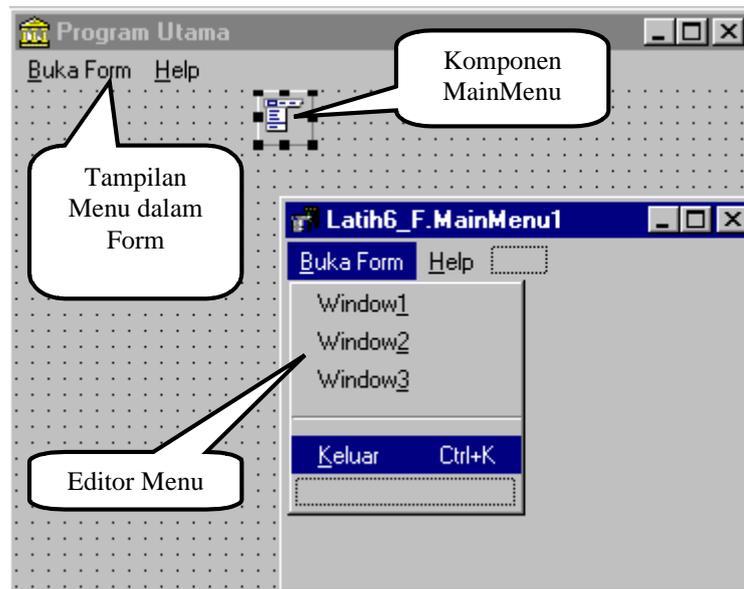
Komponen-komponen non visual, pada palette Data Acces

Untuk penggunaan komponen Query dan komponen data acces lainnya kita bahas di bagian pembuatan database dengan Delphi.



Komponen **MainMenu** merupakan komponen pokok untuk membuat menu standar Windows. Menu standar Windows terdiri dari menu bar, Menu Pulldown, dan Menu Pop-up. Menu Bar terletak di bagian atas Window yang apabila dipilih nantinya akan memunculkan menu pulldown. Menu pop-up merupakan jenis menu lainnya yang dapat muncul di bagian mana saja di window pada saat user menekan tombol kanan mouse (klik kanan) dengan syarat properti (\diamond PopUpMenu) dari form (atau komponen lainnya) telah dihubungkan dengan MenuPopUp yang bersangkutan.

Bentuk icon komponen **MainMenu** dan komponen **PopUpMenu** seperti terlihat pada gambar di atas. Jika komponen tersebut diletakkan di form dan jika diklik ganda maka akan muncul editor menu yang berfungsi untuk membuat item-item menu yang diinginkan.



Langkah-langkah membuat Menu Bar atau Menu Pop-up :

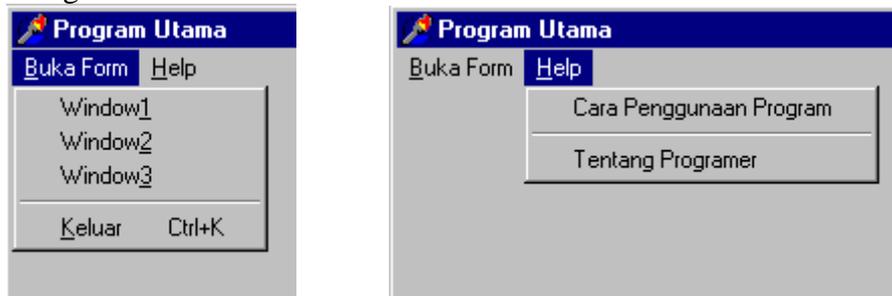
1. Letakkan komponen MainMenu atau PopUpMenu pada form.
2. Klik ganda komponen tersebut, akan muncul editor menu.
3. Pada editor menu pilih Properti Caption, ubahlah caption itemnya, lalu tekan ENTER.
4. Editor menu akan menuju ke sub menu (menu pulldown), isikan caption item seperti di atas, lalu tekan ENTER.
5. Demikian seterusnya hingga terbentuk menu yang dikehendaki.

Contoh aplikasi : (lihat gambar di atas)

1. Buat project baru (New Application)
2. Tambahkan form baru lagi 3 buah sehingga ada empat form dalam project tersebut.
3. Beri nama masing-masing form (properti **Name**) berturut-turut :
 Form1 ◇ Latih6_F,
 Form2 ◇ Latih6A_F,
 Form3 ◇ Latih6B_F, dan
 Form4 ◇ Latih6C_F.
4. Ubahlah Caption masing-masing form (properti **Caption**) berturut-turut :
 Form1 ◇ Program Utama,
 Form2 ◇ Window I,
 Form3 ◇ Window II, dan
 Form4 ◇ Window III.
5. Simpan semua (**Save All**) :
 Form1 ◇ Latih6_U,
 Form2 ◇ Latih6A_U,
 Form3 ◇ Latih6B_U,
 Form4 ◇ Latih6C_U dan

Project1 ◇ Latih6_P.

6. Pada form1 (Latih6_F) (sebagai program utama) letakkan komponen **MainMenu**
7. Klik ganda komponen **MainMenu** tersebut dan buatlah item-item menu sebagai berikut :



8. Gunakan Shift+F12 untuk memilih Form yang ingin diaktifkan, Ctrl+F12 untuk memilih Unit yang ingin diaktifkan.
9. Aktifkan form1 (Latih6_F) dengan melakukan klik tunggal pada form tersebut,
10. Hubungkan (**Use Unit**) form1 (Latih6_F) dengan form-form lainnya, bertujuan agar form yang akan dipanggil nantinya dapat dikenal oleh form1.
11. Kemudian hubungkan menu pulldown-nya dengan form-form yang dimaksud dengan mengklik tunggal pada sub menu di form1 maka akan dibawa ke code editor dan isikan (diantara begin dan end -nya untuk masing-masing sub menu sbb:

♣ **Window1** :

```
procedure TLatih6_F.Window1Click(Sender: TObject);
begin
    Latih6A_F.ShowModal;
    Latih6A_F.free;
end;
```

♣ **Window2** :

```
procedure TLatih6_F.Window2Click(Sender: TObject);
begin
    Latih6B_F.ShowModal;
    Latih6B_F.free;
end;
```

♣ **Window3** :

```
procedure TLatih6_F.Window3Click(Sender: TObject);
begin
    Latih6C_F.ShowModal;
    Latih6C_F.Free;
end;
```

♣ **Keluar** :

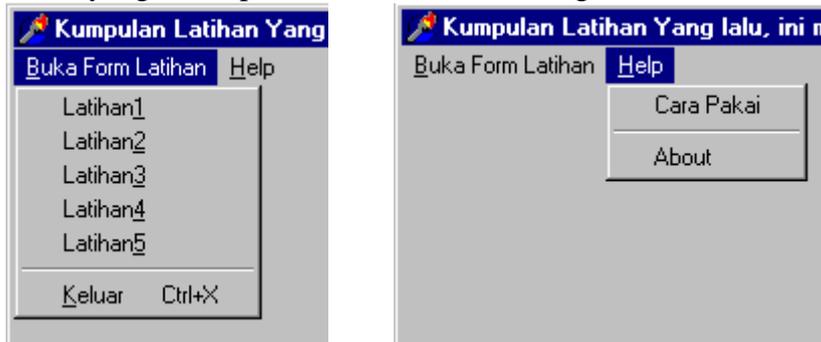
```
procedure TLatih6_F.Keluar1Click(Sender: TObject);
begin
    Close;
end;
```

end;

Latihan :

Buatlah project baru kemudian gabungkan unit-unit latihan yang lalu dengan dihubungkan oleh sub menu, sesuai kreatifitas Anda.

Menu yang diharapkan minimal adalah sebagai berikut :



- Latih1 ◇ akan memanggil form latih1
- Latih2 ◇ akan memanggil form latih2
- Latih3 ◇ akan memanggil form latih3
- Latih4 ◇ akan memanggil form latih4
- Latih5 ◇ akan memanggil form latih5
- Keluar ◇ akan keluar dari aplikasi tersebut.

Jangan lupa untuk menghubungkan form utama dengan form-form lainnya menggunakan fasilitas use unit (**F**ile**U**se **U**nit...)