

# **Pemrograman Visual Delphi 7**

## **D3 TI FMIPA UNS**

Sakroni

### **Kontrak perkuliahan**

- Matakuliah : Pemrograman visual (Delphi 7)
- SKS : 3 SKS ( 1 sks teori, 2 sks praktikum)
- Semester : II TI D (MI B)
- Pertemuan : 16 x (15 Februari – 30 Juni 2010)
- Uji Kompetensi Dasar terbagi dalam 4 blok dapat berupa test dan non test
  - Test : Ujian tertulis, tes lisan, tes kinerja, presentasi, praktikum, tugas & penilaian jenis lain yg sesuai dgn karakteristik KD.
  - Non Test : Partisipasi, skala sikap.
- Ujian Kompetensi tiap-tiap KD tidak terjadwal
- Mahasiswa yang tidak lulus dalam uji KD dilakukan remidi maksimal 2 kali
- Skor penilaian KD dengan skala 100 dengan batas penguasaan minimal ketuntasan adalah 60
- Teori : Selasa jam ke-7 (13.25-14.15) ruang A3.08
- Praktikum : Kelompok A setiap hari Selasa jam ke 8-10  
: Kelompok B setiap hari Rabu jam ke 4-6
- Korti : Widayato

- **Pembagian Kelompok praktikum :**
  - Kelompok A : M3209070 -
  - Kelompok B : M3209048-M3209069
  
- **Strategi Perkuliahan :**
  1. Ceramah
  2. Praktikum
  3. Tugas-tugas
  4. Seminar tugas akhir semester
  
- **Kriteria Penilaian :**
  1. KD 1 : 20 %
  2. KD 2 : 20 %
  3. KD 3 : 20 %
  4. KD 4 : 40 % (Tugas akhir smt)
  
- Nilai Akhir adalah nilai rerata  
(KD1 + KD2+KD3+KD4)/4
  
- Tingkat kehadiran di kelas minimal 75 %

## **Tujuan Instruksional :**

- Mahasiswa mampu memberikan konsep dasar pembuatan program dengan visual delphi
- Pada akhir perkuliahan mahasiswa diharapkan mampu membuat suatu program Aplikasi dengan menggunakan bahasa pemrograman visual Delphi

# Bahasa pemrograman ?

Teknik komando / instruksi standar untuk memerintah komputer

## Bahasa Pemrograman

1. Under dos
2. Under Windows

Under Dos	Under Windows / Visual
Pascal	Visual Delphi
Basic	Visual Basic
Foxpro	Visual Foxpro
Clipper	Visual dBase
dBase	Visual C
C	Dll
Fortran	
Assembler	
Cobol	
dll	

# Kenapa mesti Delphi ?

- Tool-nya lengkap dan terintegrasi
- Interface-nya user friendly (*more friendly than VB ?*) ☺
- Aplikasi hasil kompilasi-nya teruji cepat (VB < Delphi < C Builder/Visual)
- Kode pemrogramannya lebih ringan dibanding peng-kode-an pada C, sebab Delphi menggunakan kode object pascal (Pascal < C)
- Tersedia *Online Help* yang cerdas
- Fasilitas untuk aplikasi database-nya lengkap dan mudah digunakan
- Dapat terhubung ke beragam jenis database
- Mendukung pengembangan aplikasi modern (Jaringan/*Distributed*, Internet/WEB,...)
- Tersedia juga untuk sistem yang berbeda (*non Microsoft oriented*), seperti di Linux,...
- Desain
- Berbasis Windows
- ...

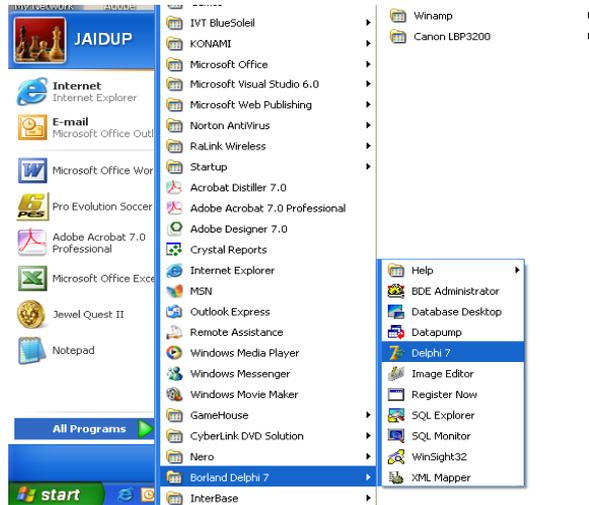
## Visual Delphi 7

- Pemrograman berorientasi obyek → Program yang semua komponennya merupakan suatu obyek

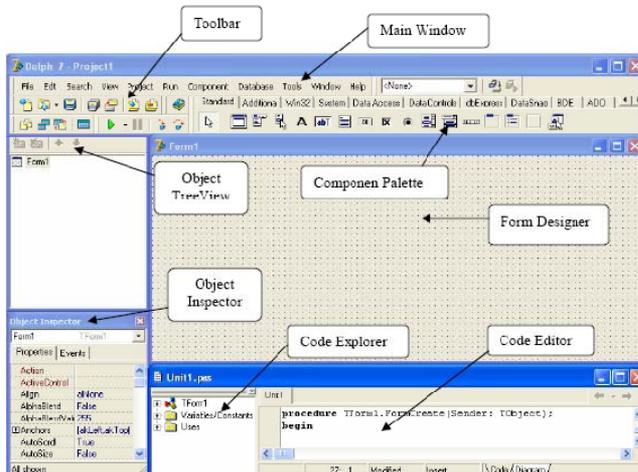
### Ciri-ciri Obyek ?

- Memiliki nama
- Property
- Method
- Procedure
- dll

# Menjalankan program Delphi



# Tampilan Delphi



# Komponen Delphi

- **Project** → sekumpulan form, unit dan beberapa hal lain dalam program aplikasi.
- **Form** → Suatu object yang dipakai sebagai tempat untuk bekerja program aplikasi
- **Unit** → Modul kode program
- **Program** → Program dibangun dari satu unit atau lebih.
- **Property** → digunakan untuk mendefinisikan atribut atau setting suatu object
- **Event** → Peristiwa atau kejadian yang diterima suatu object.
- **Method** → procedure atau perintah yang melekat pada suatu object

## Struktur Isi File Project (\*.dpr), Unit (\*.pas), dan Form (\*.dfm)

### 1. File Project

Kita dapat melihat isi file project, caranya klik menu PROJECT > VIEW SOURCE. Berikut ini adalah salah satu contoh bentuk isi file project:

```
program Project1; uses  
  Forms,  
  Unit1 in 'Unit1.pas' {Form1};  
{ $R *.RES }  
begin  
  Application.Initialize;  
  Application.CreateForm(TForm1, Form1);  
  Application.Run;  
end.
```

Nama Project

File Unit

Perintah Unit1 in 'Unit1.pas' {Form1}; bermakna bahwa dalam PROJECT1 terdapat satu buah form dengan nama formnya adalah FORM1. Dalam FORM1 memiliki file unit dengan nama UNIT1.PAS. Perintah yang sama akan muncul kembali apabila terdapat form dan unit yang lebih dari satu pada PROJECT1.

#### Tips:

Jangan coba-coba untuk utak-atik atau mengubah file ini, karena apabila Anda tidak memahaminya, bisa rusak projectnya.

## 2. Form

Berikut ini adalah contoh isi file\*.dfm

```
object Form1: TForm1
  Left = 192
  Top = 107
  Width = 544
  Height = 375
  Caption = 'Form1'
  Color = clBtnFace
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -11
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  OldCreateOrder = False
  PixelsPerInch = 96
  TextHeight = 13
```

```
object Button1: TButton
  Left = 120
  Top = 16
  Width = 75
  Height = 25
  Caption = 'Button1'
  TabOrder = 0
  OnClick = Button1Click
end
End
```

File tersebut menginformasikan tentang properti yang dimiliki oleh suatu file unit atau suatu form.

## 3. File Unit

Berikut ini adalah salah satu contoh isi dari file unit:

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Classes,
  Graphics, Controls,
  Forms, Dialogs;
type
  TForm1 = class(TForm)
    Button1: TButton;
    procedure Button1Click(Sender:
      TObject);
  private
    { Private declarations }
  public
    { Public declarations }
end;
```

```
var
  Form1: TForm1;
implementation
{$R *.DFM}
```

<----- tempat menuliskan event-  
event ----->

```
end.
```

## Properti

Setiap obyek dalam Delphi (termasuk form) pasti memiliki properti. Properti dapat diubah-ubah sesuai keinginan. Untuk melihat properti dari suatu obyek, langkahnya:

- klik obyek yang dipilih
- properti dapat dilihat pada bagian OBJECT INSPECTOR (sebelah kiri)
- Untuk mengganti properti obyek cukup mengganti value yang ada di kolom sebelah kirinya.

## EVENT

Selain properti, dalam Object Inspector juga terdapat Event dari suatu obyek. Event digunakan untuk membuat procedure yang diaktifkan (trigered) lewat sebuah event.

Macam-macam event yang ada banyak sekali, sebagai contoh event adalah "ON CLICK" karena suatu proses akan terjadi ketika kita mengCLICK tombol.

- Untuk memberikan event suatu obyek, caranya:
- Klik obyek yang akan diberikan event
- Klik menu tab EVENT pada OBJECT INSPECTOR
- Pilih event yang sesuai
- Double click kotak kosong di sebelah kanan event yang dipilih

### Tips:

- Apabila Anda tidak melihat OBJECT INSPECTOR, atau tiba-tiba OBJECT INSPECTORNYA hilang, jangan panik. Untuk menampilkannya kembali caranya Anda tekan F11 atau klik menu VIEW > OBJECT INSPECTOR

## Membuat project baru

Aplikasi atau program yang kita buat dalam Delphi menggunakan istilah “project”. Untuk membuat project baru, langkahnya:

- Klik menu FILE
- Pilih NEW APPLICATION

## Menyimpan Project/Aplikasi

- Menyimpan kode aplikasi delphi berbeda dengan menyimpan kode program lain seperti kode program DOS misalnya
- Satu program DOS umumnya memiliki satu file saja, walaupun bisa juga lebih (misalnya overlay di pascal atau membuat library header di C), tetapi file tersebut sengaja dibuat oleh programmer-nya
- Sedangkan pada Delphi, satu program Delphi menghasilkan banyak file, dan sebagian besar file itu ada tanpa disengaja oleh programmernya
- Oleh karena itu menyimpan aplikasi Delphi perlu diatur dalam folder khusus untuk mengumpulkan file-file yang dihasilkan oleh satu aplikasi

- Satu aplikasi Delphi setidaknya akan menghasilkan file-file :
  - \*.dpr , yaitu : ...
  - \*.pas , yaitu : ...
  - \*.dfm , yaitu : ...
  - \*.cfg , yaitu : ...
  - \*.dof , yaitu : ...
  - \*.res , yaitu : ...
- Setelah di-compile akan bertambah 2 jenis file lagi yaitu :
  - \*.dcu , yaitu : ...
  - \*.exe , yaitu : ...
- Jika aplikasi di-Edit (ada yang diubah) kemudian disimpan lagi, akan bertambah file dengan ekstension :
  - \*.~pas , yaitu : ...
  - \*.~dfm , yaitu : ...

Dalam Delphi ada 3 buah file utama (**\*.dpr**, **\*.pas** dan **\*.dfm**).

1. **\*.dpr** adalah file proyek yang dibuat berisi program kecil untuk :
  - mendefinisikan Unit yang ada dalam file proyek
  - menginisialisasi data
  - membangun form
  - menjalankan aplikasi
2. **\*.pas** adalah file unit-unit (*pascal code file*), bisa terdiri satu atau banyak file dalam satu project. File inilah tempat kita menuliskan listing program.
3. **\*.dfm** adalah file definisi Form (*special pseudo code file*), bisa terdiri satu atau banyak file. Isi dari file ini adalah properti-properti dari obyek dalam suatu form. Jika formnya lebih dari satu, maka file ini juga akan lebih dari satu.

**Catatan:**

- Setiap Form (.dfm) harus memiliki sebuah Unit (.pas), tetapi Anda dapat memiliki Unit tanpa sebuah Form (hanya kode saja).

## Membuka File Project

- Untuk membuka file project caranya pilih menu OPEN PROJECT > pilih file project yang diinginkan.

### Tips:

- Ingat!! Jangan membuka file project dengan menu OPEN, karena dapat dimungkinkan Anda membuka file unit. Kalau hal ini terjadi ketika Anda masih bekerja dengan suatu project, maka file unit yang Anda buka tadi akan menjadi satu dengan project yang sedang Anda kerjakan.

## Meletakkan komponen dalam form



Ada beberapa cara menempatkan komponen ke form.

- Pilih komponen (klik komponen yang diinginkan) pada **Components Palette** kemudian klik pada form dimana komponen itu akan diletakkan.
- Klik ganda pada komponen yang diinginkan, maka komponen tersebut akan ditambahkan pada form
- Anda dapat menggunakan *Copy* dan *Paste* bila ingin membuat komponen yang sama yang sudah ada pada form. Caranya Shift-Klik kiri pada komponen yang ada di form, lalu pilih menu *Copy* (Ctrl-C) kemudian pilih menu *Paste* (Ctrl-V).

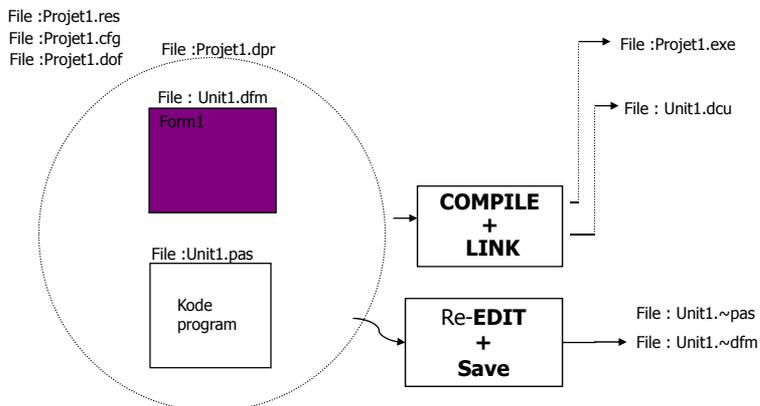
# Memahami komponen-komponen Delphi

- Delphi memiliki banyak sekali komponen yang digunakan untuk membangun interface aplikasi yang dibuat.
- Dari sekian banyak komponen tersebut maka FORM adalah komponen utama/induk, dari aplikasi sebab semua komponen lain umumnya diletakkan di atas Form
- Delphi meletakkan komponen-komponen-nya pada satu tempat yang disebut : Componen Palette :

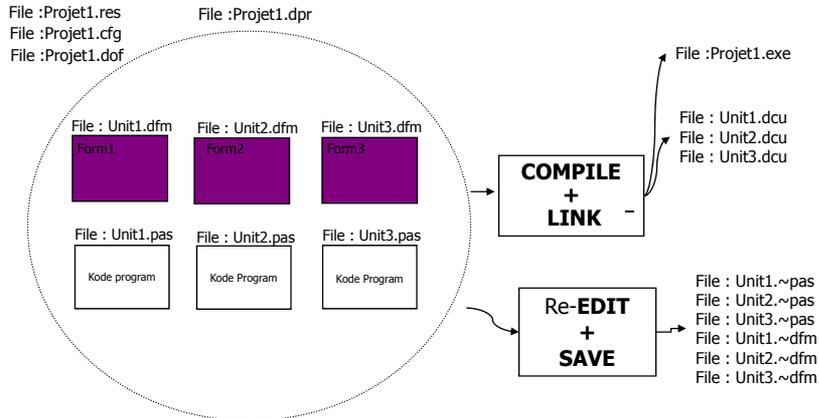


- Delphi mengelompokkan komponen-komponen tersebut dalam tab-tab sesuai dengan fungsi dan penggunaannya
- Jika anda terbiasa dengan pemrograman DOS yang mengenal perintah-perintah input dan output, maka komponen Delphi juga ada yang berperan sebagai input (untuk menerima data) dan ada juga yang berperan sebagai output (untuk menampung dan menampilkan data)

## File-file aplikasi Delphi 1 Form



## File-file aplikasi Delphi MultiForm



## Memahami pembuatan kode program

- Kode program yang di diketikkan umumnya berguna untuk memberikan reaksi atas event tertentu dari suatu komponen.

### Contoh :

Kode program di Event *OnClick* pada komponen *Button1* akan dilaksanakan hanya saat tombol *Button1* di-click.

- Untuk memilih event apa yang akan diberi kode program pada sebuah komponen dapat dipilih dari *Object Inspector*
- Untuk itu perlu dipahami bagaimana cara menentukan kode program tersebut.

### Perintah yang sering digunakan :

```
Application.terminate;
Form1.show;
Form1.showmodal;
Form1.close;
Inttostr()
Strtoint()
floattostr()
Strtofloaty()
Floattostr()
Showmessage (pesan);
MessageDlg (pesan, typepesan, typebutton, integer)
...
...
```

### Kode program yang melibatkan komponen

#### Bentuk umum :

```
<Nama_komponen.nama_property>
```

Contoh :

```
Label1.caption
```

#### Bentuk lain :

```
<Nama_komponen.nama_method>
```

Contoh :

```
Form.show
```

#### Memberikan nilai ke property komponen :

Contoh :

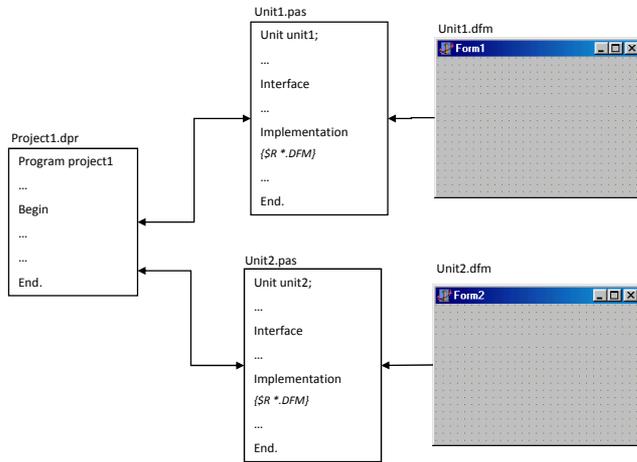
```
Label1.caption := 'Penjumlahan';
```

```
Label1.font.color := clblack;
```

- Struktur dan syntax dasar pemrograman Delphi sama dengan di pascal seperti :

Deklarasi var, type, Operator, kontrol if-then, if-then-else, case of, pengulangan for-do, while-do, repeat-until, procedure, function, begin-end, ... dan seterusnya

## Bagaimana kode program Delphi dijalankan



## Contoh kode program lengkap

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Classes,
  Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls;
type
  TForm1 = class(TForm)
    Button1: TButton;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Button2: TButton;
    Label5: TLabel;
    Label6: TLabel;
    RadioGroup1: TRadioGroup;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
uses unit2;
{$R *.DFM}

function jumlah(A,B:integer):integer;
var c:integer;
begin
  c:=A+B;
  jumlah :=c;
end;
    
```

```

procedure TForm1.Button1Click(Sender: TObject);
begin
  application.terminate;
end;

procedure TForm1.Button2Click(Sender: TObject);
var A,B :integer;
    C : real;
begin
  A:=strtoint(Edit1.text);
  B:=strtoint(Edit2.text);
  Case RadioGroup1.ItemIndex of
    0: C:=jumlah(A,B);
    1: C:=A-B;
    2: C:=A*B;
    3: C:=A/B;
  end;
  Label6.Caption:=floattostr(C);
end;
end.
    
```