



#2



MODUL PRAKTIKUM

pemrograman visual



Borland®

function
interface
begin

Delphi™

Enterprise

©2002 Borland Software Corporation



Disusun oleh Tim Assisten Dosen
Mata Kuliah Pemrograman Visual

Program Studi Ilmu Komputer
Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam
Universitas Pendidikan Indonesia

DAFTAR ISI

DAFTAR ISI	1
1. TIPE DATA DALAM BORLAND DELPHI7	2
1.1. Integer	2
1.2. Real	2
1.3. Boolean	3
1.4. Char	3
1.5. String	3
2. VARIABLE, KONSTANTA DAN KOMENTAR DALAM BORLAND DELPHI7	3
2.1. Variable	3
2.2. Konstanta	4
2.3. Komentar	5
3. OPERASI DALAM BORLAND DELPHI7	5
3.1. Operasi Boolean pada Borland Delphi7	5
3.2. Operasi Aritmatika pada Borland Delphi7	6
4. PEMILIHAN DAN PENGULANGAN DALAM BORLAND DELPHI7	6
4.1. Pemilihan	6
4.2. Pengulangan	9
5. IMPLEMENTASI PROGRAM #2 DENGAN BORLAND DELPHI7	11
6. PROBLEM TO SOLVE	12
REFERENSI	13

1. TIPE DATA DALAM BORLAND DELPHI7

1.1. Integer

Integer adalah bilangan bulat. Terdapat beberapa tipe data yang merupakan bagian dari integer yang dibagi menjadi dua kelompok yaitu :

a. Tipe Generik

Tipe Data	Range	Format
Integer	-2,147,483,648 s.d. 2,147,483,647	Signed 32-bit
Cardinal	0 s.d. 4,294,967,295	Unsigned 32-bit

b. Tipe Fundamental

Tipe Data	Range	Format
Shortint	-128 s.d. 127	Signed 8-bit
SmallInt	-32,768 s.d. 32,767	Signed 16-bit
LongInt	-2,147,483,648 s.d. 2,147,483,647	Signed 32-bit
Int64	$-(2^{63})$ s.d. $(2^{63})-1$	Signed 65-bit
Byte	0 s.d. 255	Unsigned 8-bit
Word	0 s.d. 65,535	Unsigned 16-bit
LongWord	0 s.d. 4,294,967,295	Unsigned 32-bit

1.2. Real

Tipe data real merupakan himpunan bilangan asli. Seperti halnya tipe data integer, terdapat beberapa tipe data yang termasuk ke dalam tipe data real yaitu :

Tipe Data	Size (Byte)	Range	Significant Digits
Real	6	2.9×10^{-39} s.d. 1.7×10^{38}	11-12
Single	4	1.5×10^{-45} s.d. 3.4×10^{38}	7-8
Double	8	5.0×10^{-324} s.d. 1.7×10^{308}	15-16
Extended	10	3.4×10^{-4932} s.d. 1.1×10^{4932}	19-20
Comp	8	-9,223,372,036,854,775,808 s.d. 9,223,372,036,854,775,807	19-20
Currency	8	-9,223,372,036,854,77.5808 s.d. 9,223,372,036,854,77.5807	19-20

1.3. Boolean

Boolean merupakan tipe data yang hanya memiliki dua nilai yaitu **true** dan **false**. Dalam Delphi, tipe data ini disimpan dengan panjang 1 byte.

1.4. Char

Tipe data char merupakan tipe data yang berisi sebuah karakter.

1.5. String

Tipe data string merupakan kumpulan dari karakter-karakter (array of char). Pada tipe data ini, setiap karakter juga dapat diakses dengan cara **nama_variabel[no_urut_karakter]**.

a. Short String

Short string merupakan tipe data string yang memiliki panjang maksimum 255 karakter. Pendefinisian Short String dapat dilakukan dengan dua cara, yaitu dengan menuliskan ShortString atau string[jumlah_karakter].

b. Long String

Long string (atau cukup disebut string saja) merupakan tipe data string yang jumlah maksimum karakternya ditentukan oleh banyaknya memori yang tersedia.

2. VARIABLE, KONSTANTA DAN KOMENTAR DALAM BORLAND DELPHI7

2.1. Variable

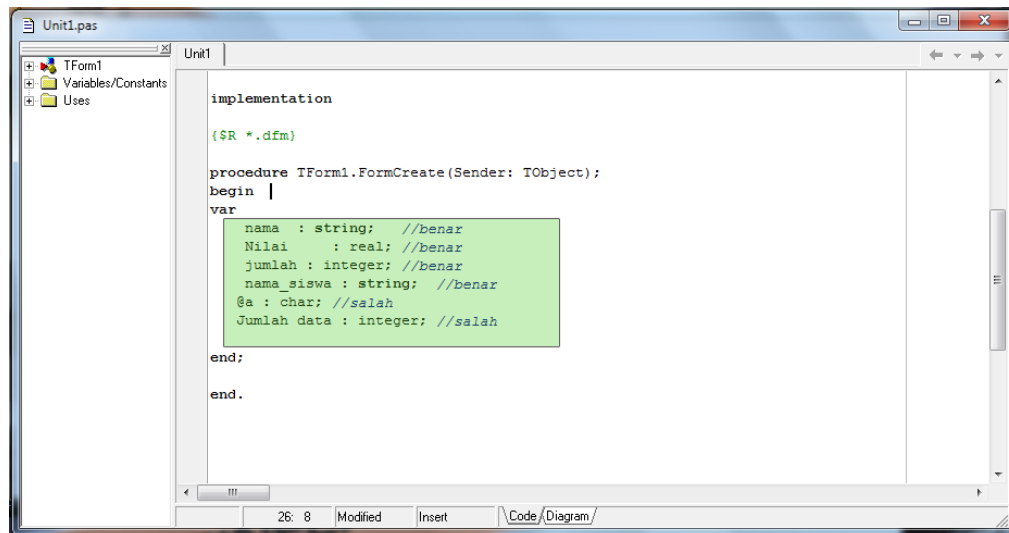
Variabel merupakan suatu identifier yang nilainya dapat berubah-ubah sesuai kebutuhan. Sebelum suatu variabel dapat digunakan, variabel tersebut harus didefinisikan nama dan tipe datanya terlebih dahulu.

Terdapat beberapa aturan dalam pendefinisian variabel:

- a. Nama variabel hanya boleh terdiri dari angka, abjad, dan karakter underscore (“_”)
- b. Jika nama variabel terdiri dari 2 kata, kedua kata tersebut harus dihubungkan dengan karakter underscore(“_”)

Dalam Delphi huruf kapital atau bukan, akan dianggap sama (tidak case sensitive).

Berikut adalah contoh pendefinisian variable:



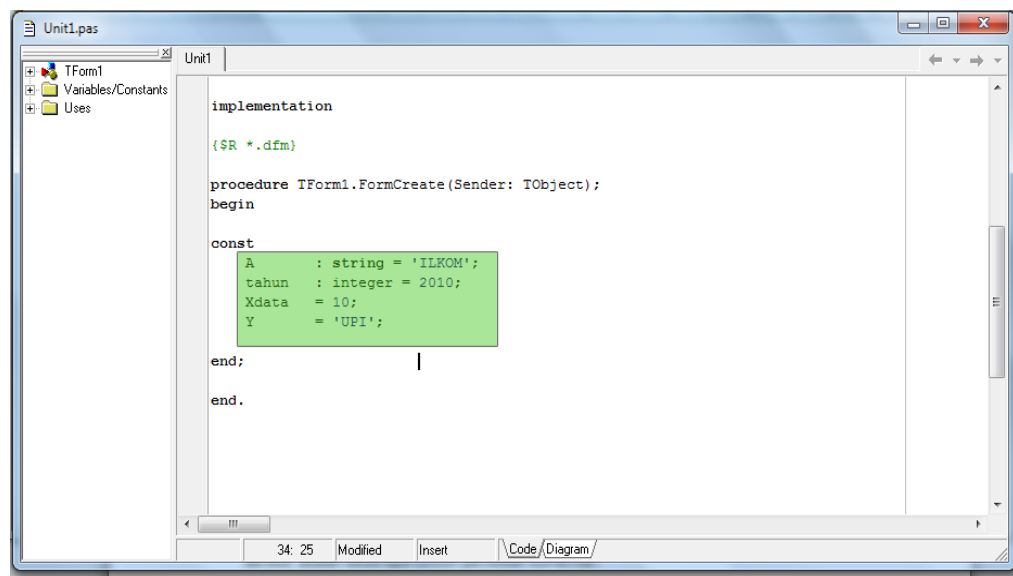
Gambar 2.1 Pendefinisian Variable

2.2. Konstanta

Konstanta adalah identifier yang nilainya bersifat tetap selama program berjalan. Pendefinisian konstanta dilakukan pada blok kode yang diawali dengan **const**. Blok pendefinisian konstanta ini dapat diletakkan pada bagian interface, implementation, atau di mana saja dalam kode program.

Jika konstanta diletakkan pada bagian interface, maka konstanta tersebut akan dikenal di unit lain. Jika diletakkan pada bagian implementation, maka konstanta hanya akan dikenali oleh unit tersebut. Dan jika diletakkan pada suatu prosedur atau fungsi, maka konstanta hanya akan dikenali pada prosedur atau fungsi tersebut.

Berikut adalah contoh pendefinisian konstanta:

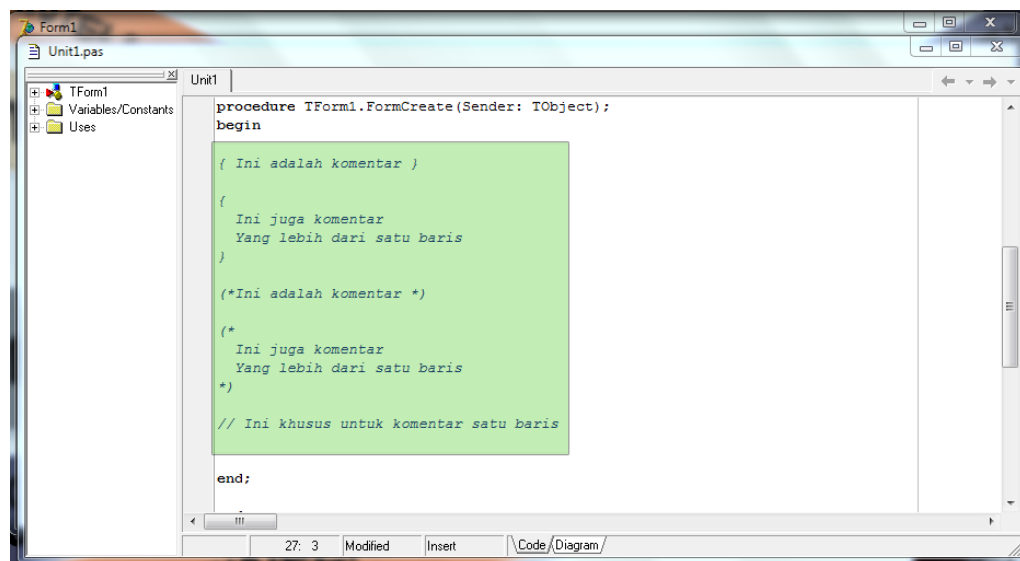


Gambar 2.2 Pendefinisian Konstanta

2.3. Komentar

Komentar adalah tulisan pada kode yang digunakan untuk kepentingan dokumentasi. Compiler tidak akan menerjemahkan komentar tersebut. Pada Delphi, penulisan komentar dibagi menjadi 2 jenis, yaitu komentar yang terdiri dari satu baris (ditulis dengan menggunakan garis miring ganda “//”) dan komentar yang lebih dari satu baris (ditulis dengan menggunakan kurung kurawal “{ }” atau kurung biasa yang disertai tanda bintang “(* *)”).

Berikut adalah beberapa contoh penulisan komentar:



Gambar 2.3 Penulisan Komentar

3. OPERASI DALAM BORLAND DELPHI7

3.1. Operasi Boolean pada Borland Delphi7

Operasi boolean memerlukan operand dengan tipe boolean (True atau False) dan menghasilkan nilai dengan tipe boolean pula.

Operator	Operasi	Contoh
not	Negasi	not(A>10)
and	Menghasilkan nilai True jika kedua operand bernilai True. Jika salah satu operand bernilai False maka akan menghasilkan nilai False	A and B
or	Menghasilkan nilai True jika salah satu atau kedua operand bernilai True. Dan akan menghasilkan nilai False jika kedua operand bernilai False	A or B

xor	Menghasilkan nilai True hanya jika salah satu operand bernilai True. Jika kedua operand bernilai True atau False, maka akan menghasilkan nilai False	A xor B
------------	--	----------------

3.2. Operasi Aritmatika pada Borland Delphi7

Operasi aritmatika digunakan untuk melakukan perhitungan aritmetik. Operand dan hasil dari operasi aritmatika bertipe integer atau real.

Operator	Operasi	Tipe Operand	Tipe Hasil
+	Penjumlahan	Integer, real	Integer, real
-	Pengurangan	Integer, real	Integer, real
*	Perkalian	Integer, real	Integer, real
/	Pembagian Real	Integer, real	Real
div	Pembagian Integer	Integer	Integer
mod	Sisa Pembagian	Integer	Integer

4. PEMILIHAN DAN PENGULANGAN DALAM BORLAND DELPHI7

4.1. Pemilihan

Pengulangan adalah mengontrol program sehingga perintah yang dijalankan mengikuti alur tertentu. Yang termasuk percabangan adalah perintah **if – then – else** dan perintah **case**.

a. If – then –else

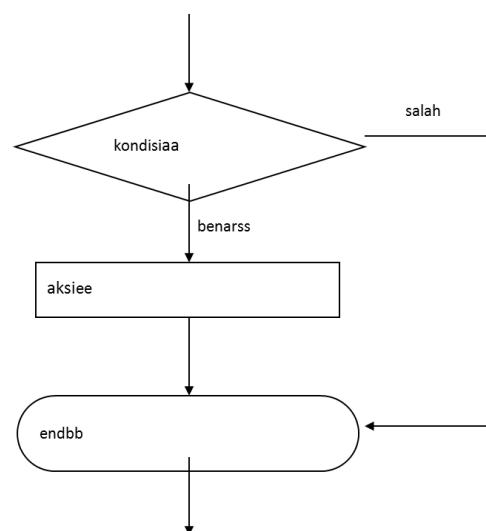
➤ Satu Kasus (IF-THEN)

Bentuk Umum:

IF <kondisi> THEN

<aksi>

Pernyataan di atas berarti aksi hanya akan dijalankan jika kondisi terpenuhi. Jika kondisi tidak terpenuhi, maka program tidak akan menjalankan operasi apapun. Berikut adalah bagan alir untuk pemilihan satu kasus



Contoh:

```

if x > 100 then
  x := x+1;
if nilai > 80 then
  begin
    indeks := 'A';
    status := 'Lulus';
  end;
end;

```

➤ Dua Kasus

Bentuk umum:

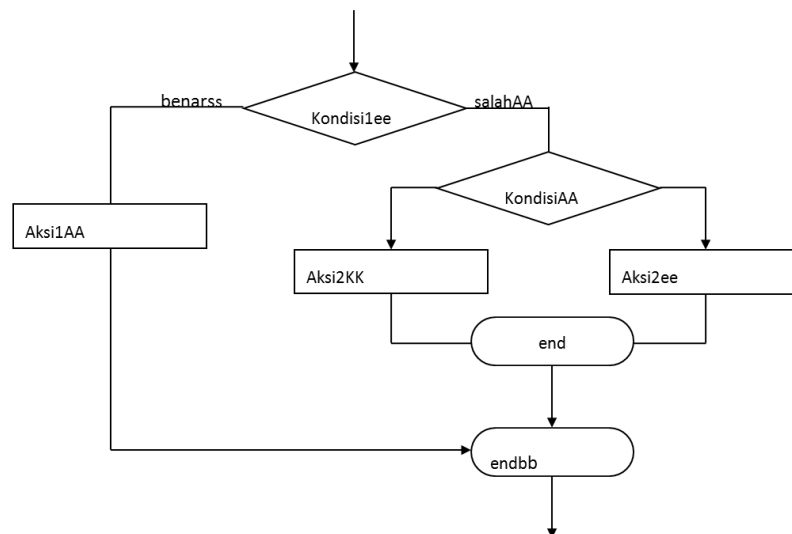
```

IF <kondisi> THEN
  <aksi1>
ELSE
  <aksi2>

```

Pernyataan di atas berarti Aksi1 akan dijalankan jika kondisi terpenuhi.

Dan jika kondisi tidak terpenuhi, maka program akan menjalankan Aksi2.



Contoh:

```

if x > 100 then
  x := x+1
else
  x := x - 1;

```

```

if matkul = 'provis' then
  begin
    a := 'pemrograman';
    b := 'visual';
  end

```



```

else
  begin
    a := 'desain';
    b := 'grafis';
  end;

```

➤ Tiga Kasus atau Lebih

Bentuk umum:

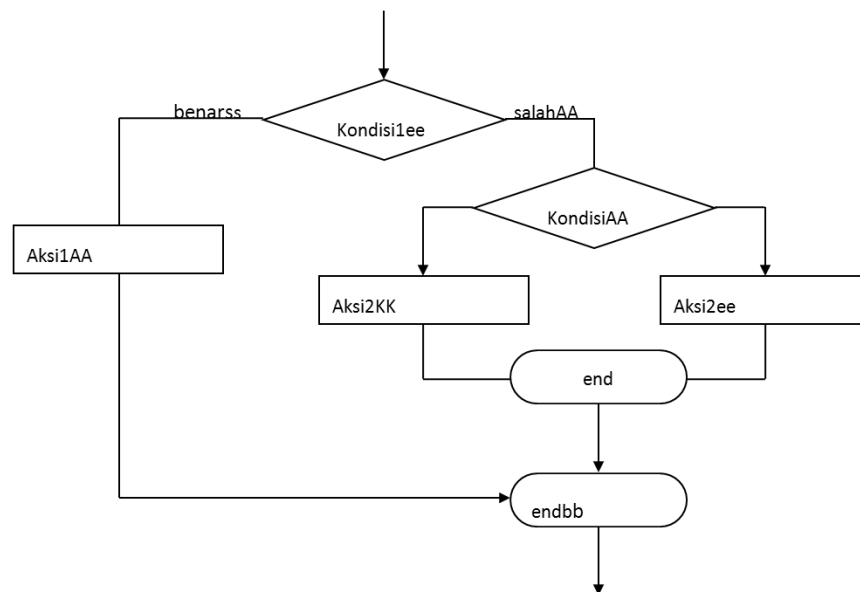
```

IF <kondisi1> THEN
<aksi1>
ELSE IF <kondisi2> THEN
<aksi2>
ELSE
<aksi3>

```

Sama dengan pemilihan dua kasus, hanya saja program akan melakukan pengecekan terhadap beberapa kondisi lagi sebelum menjalankan aksi sesuai dengan kondisi yang terpenuhi.

Berikut adalah bagan alir untuk pemilihan dua kasus:



Contoh:

```

If x > 0 then
  Ket := 'positif'
Else if x < 0 then
  Ket := 'negatif'
Else
  Ket := 'nol';

```

b. Case

Case merupakan pemilihan yang digunakan untuk menangani dua kasus atau lebih. Pemilihan case juga dapat ditulis dalam bentuk If bersarang.

Bentuk umum:

```
Case <ekspresi> of
  <nilai 1> : <aksi 1>;
  <nilai 2> : <aksi 2>;
  ...
  <nilai n> : <aksi n>;
Else
  <aksi>
End;
```

Contoh:

```
Case input of
  '1' : bulan := 'Januari';
  '2' : bulan := 'Februari';
  '3' : bulan := 'Maret';
Else
  bulan := 'Desember';
End;
```

4.2. Pengulangan

a. For

Konstruksi for digunakan untuk melakukan perulangan terhadap baris kode tertentu dengan jumlah yang telah ditentukan. Untuk mencacah berapa kali perulangan dilakukan, diperlukan suatu variabel pencacah yang bertipe integer. Variabel tersebut akan bertambah nilainya setiap kali perulangan dilakukan.

Bentuk umum:

- For menaik

```
FOR <nilai awal> TO <nilai akhir> DO
  <aksi>
```

- For menurun

```
FOR <nilai awal> DOWNTO <nilai akhir> DO
  <aksi>
```

Contoh:

```
FOR a:=1 to 10 do
  Jumlah:=jumlah+10;
```

Contoh di atas akan menambahkan nilai variabel jumlah dengan 10 sebanyak 10 kali.

b. While

Konstruksi while digunakan untuk melakukan perulangan terhadap baris kode tertentu selama suatu kondisi terpenuhi. Jika kondisi sudah tidak terpenuhi, maka program akan keluar dari perulangan.

Bentuk umum:

```
WHILE <kondisi> DO  
<aksi>
```

Contoh:

```
While (i < 30) do  
begin  
    jumlah := jumlah + 15;  
    i := i + 1;  
end;
```

Contoh di atas akan menambahkan nilai variabel jumlah dengan 15 selama nilai variabel $i < 30$.

c. Repeat

Konstruksi repeat digunakan untuk melakukan perulangan sampai suatu kondisi terpenuhi. Berbeda dengan while yang melakukan pengecekan sebelum pengulangan dilakukan, pengulangan jenis repeat melakukan pengecekan kondisi setelah perulangan dilakukan.

Bentuk umum:

```
REPEAT <aksi> UNTIL  
<kondisi>
```

Contoh:

```
Repeat  
begin  
    jumlah := jumlah + 15;  
    i := i + 1;  
end;  
Until (i = 30)
```

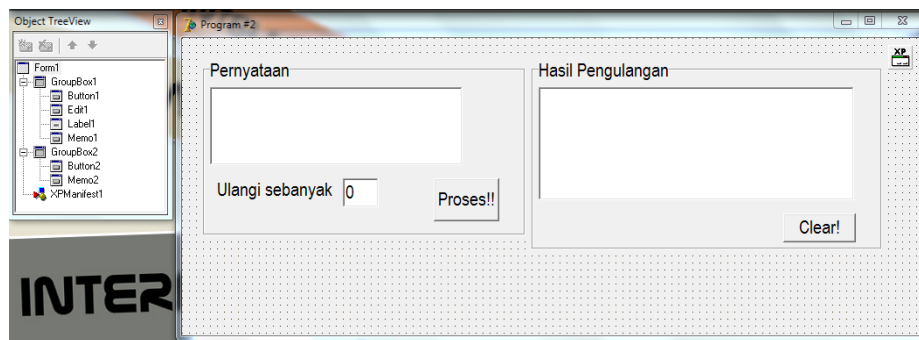
Contoh di atas akan menambahkan nilai variabel jumlah dengan 15 sampai variabel i bernilai 30

5. IMPLEMENTASI PROGRAM #2 DENGAN BORLAND DELPHI7

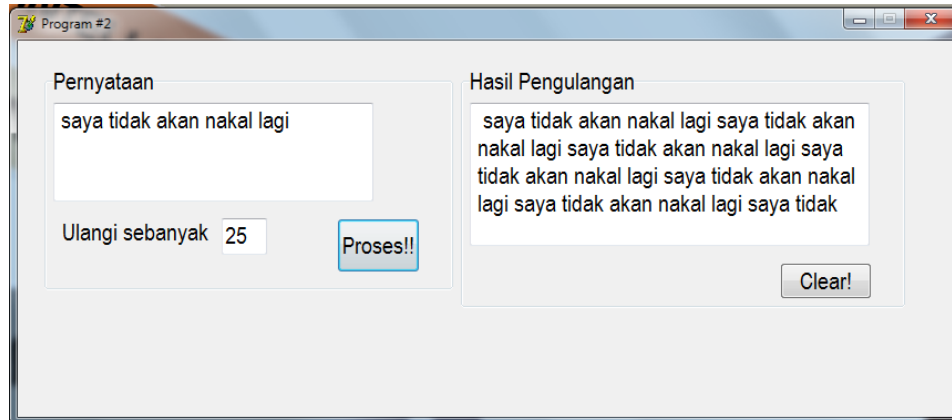
Sekarang kita akan mengimplementasikan konsep di atas kedalam sebuah program. Sekarang kita masih bermain di program yang sederhana dan tidak terlalu kompleks. Tapi untuk kali ini program tidak se-sederhana pada pertemuan kemarin. Sekarang kita mulai bermain dengan multi-form. OK... hehe...

OK let's gooo.... Eit, sebelum ke program yang multi-form, kita coba dulu program single-form yang melibatkan looping. Okey...

This is it!!! Pertama kita buat sebuah form sesuai dengan gambar di bawah ini



Buatlah sebuah form yang berisi komponen-komponen seperti pada gambar. Adapun properties dan action akan saya jelaskan pada saat praktikum.

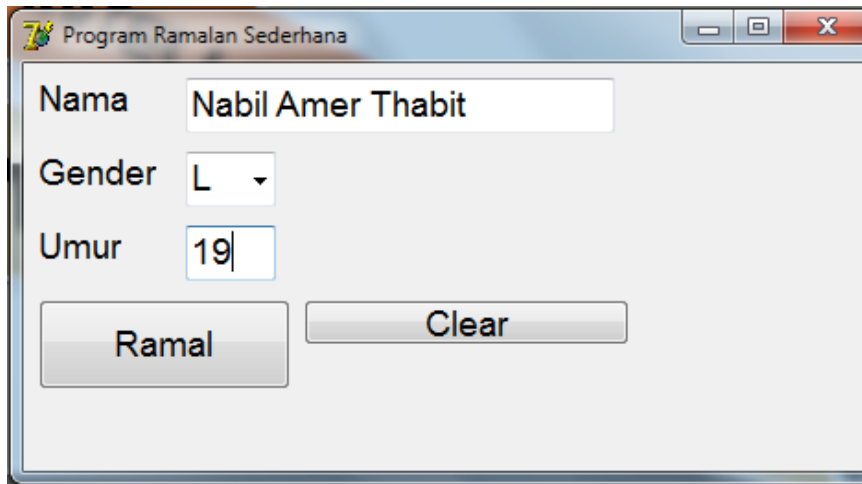


Gambar di atas menjelaskan saat program dijalankan. Saat program dijalankan, kita akan diberi sebuah kolom pengisian yang dikhususkan untuk kalimat. Setelah kolom pengisian terisi, kita masukan angka berapa banyak pengulangan yang akan dilakukan, conto 25. Maka, saat kita klik proses maka kalimat tersebut akan ditulis ulang sebanyak yang kita isikan. Lihat gambar di atas.

6. PROBLEM TO SOLVE

Pada sesi ini kita akan sedikit melakukan pendalaman terhadap materi di atas. Kita akan bermain dengan multi-form application. Adapun konten dari materi ini akan kita bahas langsung. Okey...

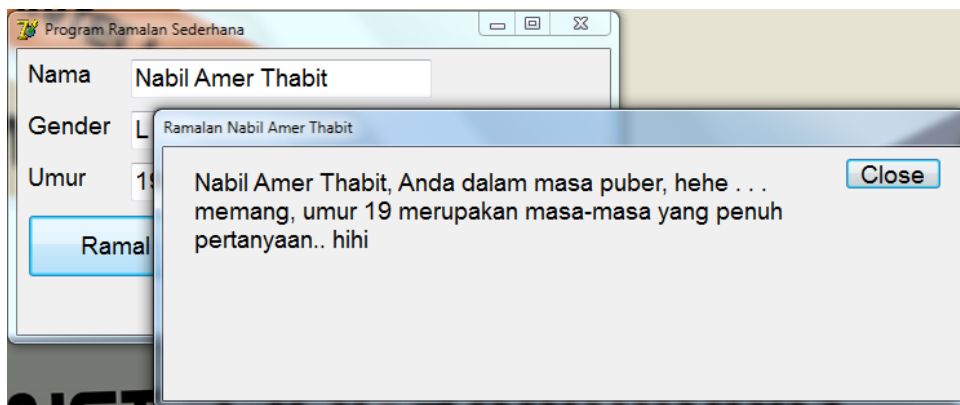
Berikut sedikit screen shot dari program yang akan kita buat ini.



Ilustrasi: Saat Program Dijalankan, Kita Isi Field yang Ada..

Saat tombol Ramal diklik, maka akan muncul form lain yang berisi ramalan sederhana, hehe.

Catatan: Saat Form2 muncul, maka Form1 berada dibelakang Form2.



Ilustrasi: Saat Program Diberi Aksi, Akan Muncul Form Lain Sesuai Gambar.

REFERENSI

<http://www.ilmukomputer.com>

“Tekhnik Pemrograman Delphi” by Kusnassriyanto dan Wawan S.

“Algoritma & Pemrograman” by Rinaldi Munir

Tim Asisten Dosen Pemrograman Visual

- Gilang Abdul Aziz
- Muhammad Yusuf
- Jaka Indria Pratama
- Rinaldi Guarsa
- Rendy
- Rini M. Haryanto
- Toni Haryanto
- Ruli Angriawan