

Modul Praktek Laboratorium Komputer

Borland Delphi



Akademi Manajemen Informatika & Informatika

BINA SARANA INFORMATIKA

Jakarta

2010

Bab I

Mengenal Delphi

1.1. Pengertian Delphi

Delphi adalah suatu bahasa pemrograman (*development language*) yang digunakan untuk merancang suatu aplikasi program.

a. Kegunaan Delphi

1. untuk membuat aplikasi windows
2. Untuk merancang aplikasi program berbasis grafis
3. Untuk membuat program berbasis jaringan (client/server)
4. Untuk merancang program .Net (berbasis internet)

b. Keunggulan Delphi

1. IDE (*Integrated Development Environment*) atau lingkungan pengembangan aplikasi sendiri adalah satu dari beberapa keunggulan delphi, didalamnya terdapat menu - menu yang memudahkan kita untuk membuat suatu proyek program.
2. Proses Kompilasi cepat, pada saat aplikasi yang kita buat dijalankan pada Delphi, maka secara otomatis akan dibaca sebagai sebuah program, tanpa dijalankan terpisah.
3. Mudah digunakan, source kode delphi yang merupakan turunan dari pascal, sehingga tidak diperlukan suatu penyesuaian lagi.
4. Bersifat multi purpose, artinya bahasa pemrograman Delphi dapat digunakan untuk mengembangkan berbagai keperluan pengembangan aplikasi.

c. Sejarah Borland Delphi

1. Delphi versi 1 (berjalan pada windows 3.1 atau windows 16 bit)
2. Delphi versi 2 (Berjalan pada windows 95 atau delphi 32 bit)
3. Delphi versi 3 (berjalan pada windows 95 keatas dengan tambahan fitur internet atau web)
4. Perkembangan selanjutnya diikuti dengan Delphi versi 4, 5 dan 6.
5. Versi terkini dari delphi adalah versi 7 dengan tambahan fitur .net dengan tambahan file XML

1.2. OOP (Object Oriented Programming)

OOP adalah metode pemrograman dengan membantu sebuah aplikasi yang mendekati keadaan dunia yang sesungguhnya. Hal itu bisa dilakukan dengan cara mendesain object untuk menyelesaikan masalah.

a. Tiga unsur OOP

1. Encapsulation atau pemodelan

Encapsulation adalah konsep penggabungan data dengan operator. Dalam konsep pemodelan data dan operasi menjadi satu kesatuan yang disebut object. *Encapsulation* juga disebut dengan penyembunyian informasi (*information hiding*)

Contoh = ketika kita menyalakan sebuah TV kita tidak tahu apa yang terjadi dengan proses dan percakapan antar alat yang berhubungan dengan TV tersebut sehingga mampu menampilkan sebuah gambar.

Atau = ketika melakukan klik pada sebuah object dalam suatu aplikasi program kita tidak tahu apa yang dilakukan program sehingga ditampilkan hasil yang sesuai.

Catatan = dari dua contoh kasus tersebut terdapat kesamaan proses mengenai *information hiding* yang tidak diketahui oleh user sampai hasil ditampilkan.

b. Inheritance atau penurunan

Inheritance adalah sebuah object yang dapat diturunkan menjadi object yang baru dengan tidak menghilangkan sifat asli dari object tersebut.

Contoh = TV merupakan salah satu media elektronik yang digunakan untuk menampilkan gambar dengan tujuan untuk memberikan informasi kepada konsumen. Secara umum TV mempunyai cara kerja yang sama dengan media elektronik yang lain dalam proses penyampaian informasi, tetapi mempunyai sifat yang unik yang dapat membedakan dengan media elektronik yang ada.

Atau = Dalam aplikasi program kita sering menggunakan command button, untuk beberapa perintah yang berbeda.

c. Polymorphism atau Polimorfisme

Polymorphism merupakan penggunaan berbagai macam object yang berbeda tetapi secara fungsi bergantung pada satu object sebagai induk, dengan cara pelaksanaan yang berbeda - beda.

Contoh = TV dan radio adalah media elektronik yang mempunyai sistem yang sama tentang bagaimana menyebarkan suatu informasi, tetapi cara kerja dari masing - masing sistem pasti berbeda.

Atau

Object Simpan dan Update adalah icon yang berasal dari induk yang sama yaitu , command button tetapi cara kerja tersebut berbeda - beda.

1.3. Delphi dan OOP (Object Oriented Programming)

Secara default ketika kita merancang suatu aplikasi program, mau tidak mau dan tanpa kita sadari bahwa kita telah menerapkan OOP, walaupun secara teori kita kurang memahami OOP dalam arti yang sebenarnya.

Contoh sederhananya adalah ketika kita merancang suatu form (Tform1) baru, sadar atau tidak sebenarnya form yang kita aktifkan merupakan turunan dari Tform sebagai induknya atau ketika kita mengaktifkan button pada form merupakan turunan dari tbutton.

Atau

Contoh dalam bahasa program adalah sebagai berikut = ketika merancang suatu label di form secara otomatis delphi akan menuliskan label tersebut dalam jendela code editor tentang turunan dari label tersebut.

Type

Tform = class(tform)

Label1 = tlabel

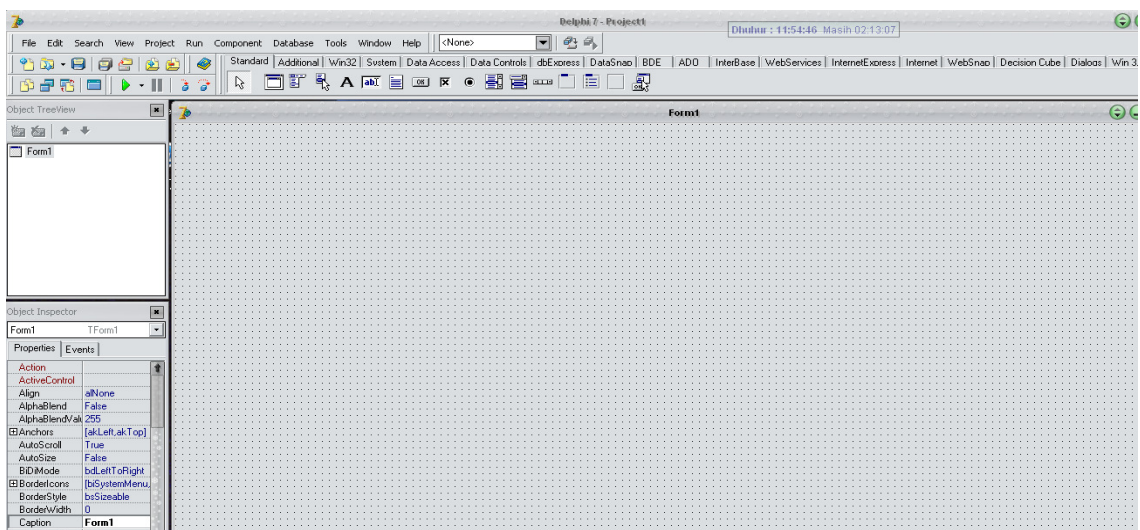
End;

1.4. IDE DELPHI

a. langkah - langkah mengaktifkan Delphi

- a. Klik start
- b. pilih program Files
- c. pilih Borland Delphi
- d. pilih dan klik Delphi 7

b. Jendela Utama Delphi

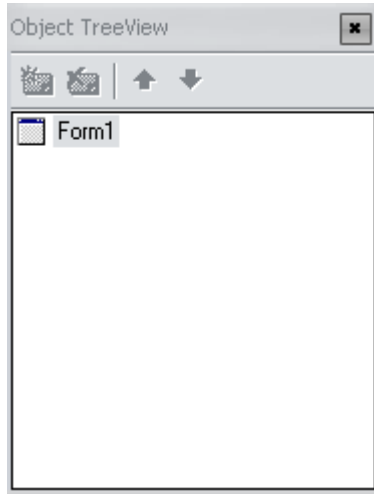


Gambar 1.1 Jendela Utama Delphi

c. Bagian - bagian dari Jendela Delphi

1. Object Tree View

Merupakan sebuah diagram pohon yang menggambarkan hubungan logis menghubungkan semua komponen yang terdapat dalam suatu proyek program. Komponen tersebut meliputi form, modul atau frame. Fungsinya digunakan untuk menampilkan seluruh daftar komponen program dalam sebuah aplikasi program sesuai dengan penempatannya.



Gambar 1.2 Jendela Object Tree View

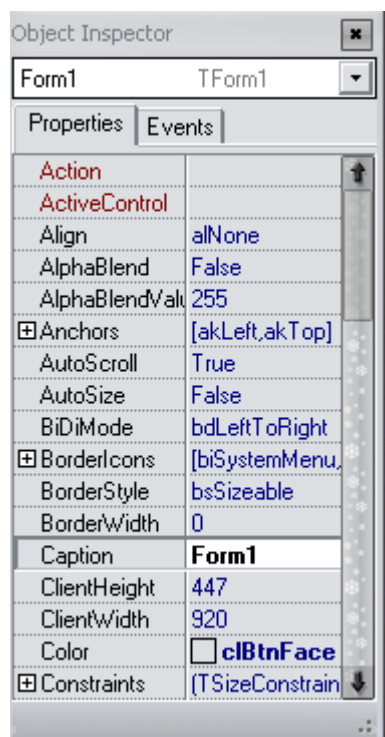
2. Object Inspector

Merupakan jendela yang digunakan untuk mengatur tampilan komponen pada form, misalnya bagaimana mengubah tulisan button pada command button menjadi Simpan, atau menghapus tulisan pada label dan mengganti nama menjadi Nama Mahasiswa atau memberikan perintah tertentu pada sebuah komponen sehingga ada interaksi ketika program dijalankan.

Secara Umum Object Inspector terbagi menjadi 2, yaitu =

a. Properties

Digunakan untuk mengatur tampilan pada sebuah komponen baik itu meliputi penggantian nama, warna, jenis huruf, border dan lain-lain.



Gambar 1.3 Jendela Inspector (properties)

b. Events

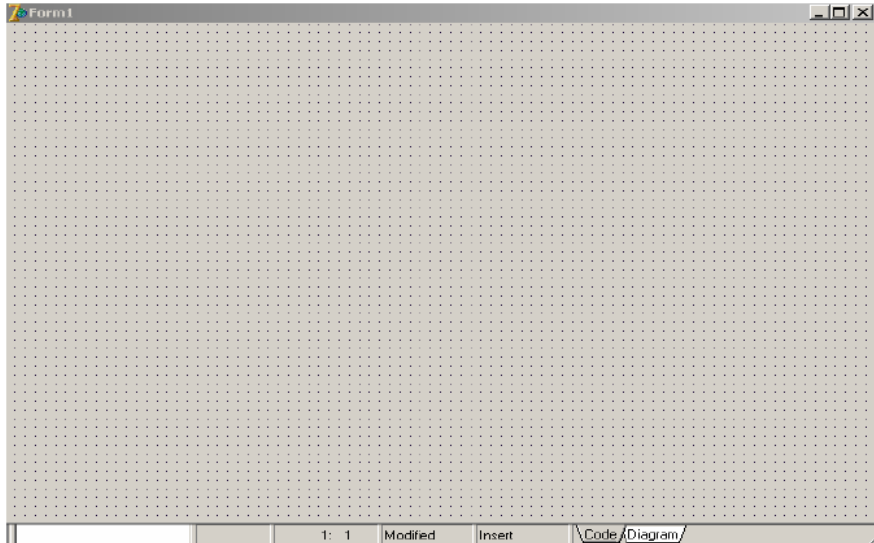
Merupakan jendela properties yang digunakan untuk memberikan fungsi yang lebih detail dari fungsi sebenarnya. Misalnya ketika tombol Simpan di klik maka program akan menjalankan perintah penyimpanan data. Dari kalimat tersebut ada event klik untuk mengeksekusi sebuah tombol simpan. Perintah event klik tersebut dapat diberikan melalui jendela events.



Gambar 1.4. Jendela Events pada Object Inspector

Form Designer

Merupakan tempat yang digunakan untuk merancang semua aplikasi program yang diambil dari komponen pallete.



Gambar 1.5. Jendela Form Designer

4. Component Palette

Merupakan kumpulan icon yang digunakan untuk merancang suatu aplikasi pada untuk membentuk sebuah aplikasi user interface. Dalam komponen pallete semua icon dikelompokkan dalam berbagai komponen sesuai dengan fungsi dan kegunaannya.

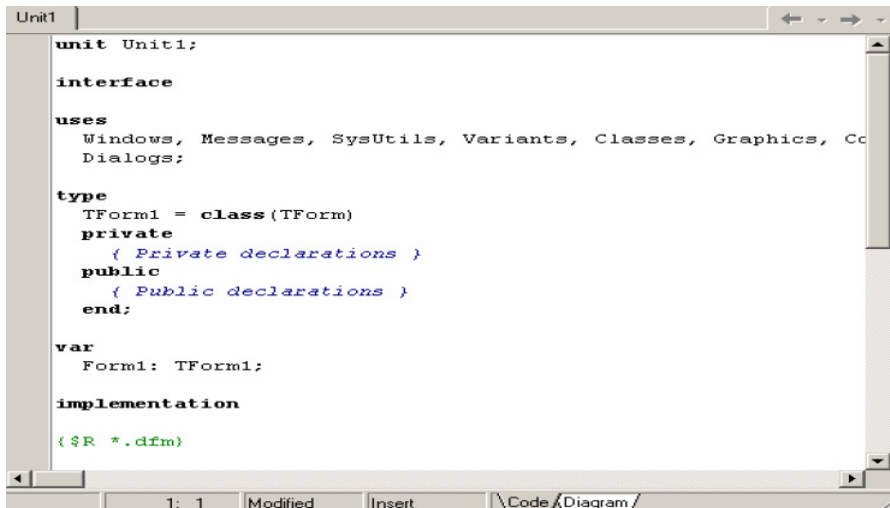


Gambar 1.6. Jendela Komponen Pallete

No	Icon	Name	Fungsi
1		Pointer	Mengembalikan fungsi mouse ke defaultnya
2		Frame	Membentuk suatu frame terhadap obyek yang ada didalamnya
3		Main menu	Membuat menu Utama
4		Popup Menus	
5		label	Hanya untuk menampilkan Teks
6		Edit	Untuk menampilkan dan input data (1 baris)
7		Memo	Sama seperti edit tetapi mempunyai kapasitas lebih besar (lebih dari 1 baris)
8		Button	Digunakan untuk melakukan eksekusi terhadap suatu proses
9		Checkbox	Digunakan untuk menentukan pilihan lebih dari satu
10		Radio Button	Digunakan untuk menentukan pilhan, tetapi hanya satu pilhan yang bisa digunakan
11		List Box	Menmpilkan pilihan dalam bentuk list
12		Combo Box	Menampilkan pilihan dalam bentuk popup
13		Scroll Bar	Merupakan icon yang berupa baris status
14		Group Box	Digunakan untuk mengelompokkan suatu icon
15		Radio Group	Digunakan untuk mengelompokkan pilihan

5. Code Editor

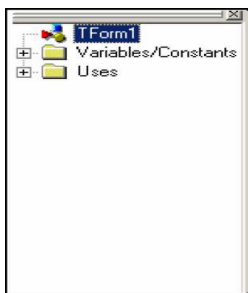
Bagian dari delphi yang digunakan untuk menuliskan kode program. Pada bagian code editor terdapat 3 bagian utama yaitu = bagian *paling kiri* yang berisi berupa *angka* menunjukkan baris dan kolom. Keterangan *modified* menunjukkan bahwa telah terjadi modifikasi terhadap baris program. Dan paling kanan menunjukkan status keyboard tentang tombol *insert* atau *over write*.



Gambar 1.7 Jendela Code Editor

6. Code Explorer

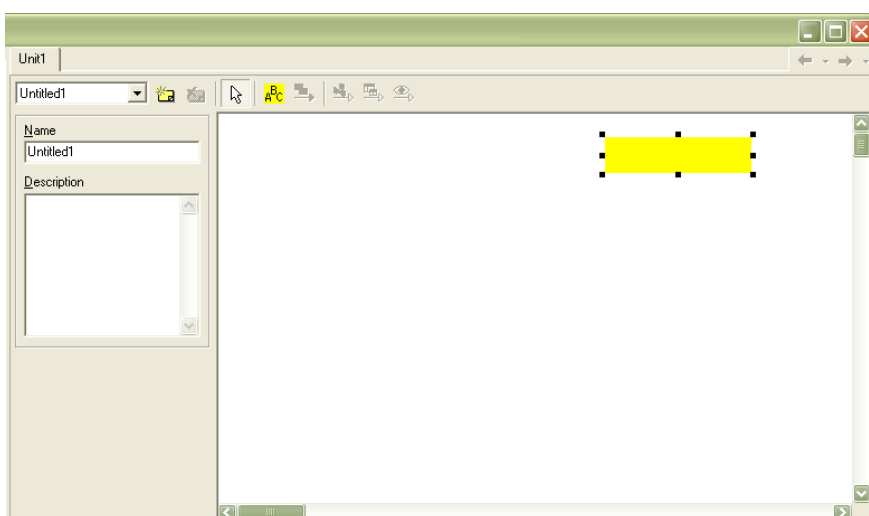
Jendela yang digunakan untuk menampilkan seluruh *variabel*, *type*, dan *routine* yang didefinisikan pada sebuah unit.



Gambar 1.8. Jendela Code Explorer

7. Code Diagram

Merupakan fasilitas pada delphi yang digunakan untuk mendesain sebuah diagram atas komponen - komponen yang digunakan dalam suatu rancangan aplikasi.



d. Proyek Delphi

1. File Proyek

File ini disimpan dengan ber-ekstensi **.dpr**. File ini berisi informasi mengenai seluruh proyek program

2. File Unit

File ini merupakan kumpulan dari barisan kode program yang terdapat di jendela

code editor, baik itu yang dituliskan oleh programmer maupun oleh system. Extension file ini adalah . pas File Unit dibagi menjadi 2

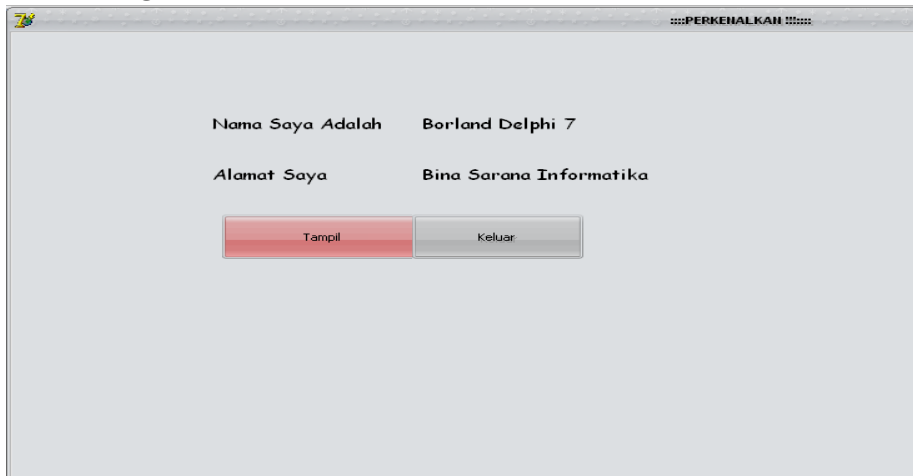
- a. Bagian Interface
Barisan ini dimulai dari kata Interface (setelah nama unit), berisi seluruh deklarasi variabel, tipe data object maupun deklarasi tambahan.
- b. Bagian Implementation
Dimulai dari kata kunci *implementation* dan diakhiri dengan kata *end*. Fungsi digunakan untuk menuliskan kode program sebagai bagain dari interaksi antar komponen ataupun dengan user.

3. File Form

Berisi tentang seluruh informasi yang ada kaitannya dengan form yang dibuat, meliputi tinggi, lebar, pososi form atau tentang komponen didalamnya. Penggunaan file ini tidak dianjurkan karena untu pengaturan sudah disediakan *object inspector* sebagai media pengaturan semua komponen.

1.5. Aplikasi Perkenalkan(LATIHAN 1)

a. Hasil Program



Gambar 1.10 Gambar Form Perkenalkan!!!

b. Desain Form



c. Desian Properties

Object	Name	Caption
Label 1	Label 1	Nama Saya Adalah
Label 2	Label 2	Alamat Saya
Label 3	Lnama	-
Label 4	Lalamat	-
Button1	Btampil	Tampil
Button2	Bkeluar	Keluar
Form 1	Form1	:::Perkenalan!!!:::

d. Listing program

1. Ketika form dalam keadaan aktif maka form akan menampilkan nama dan alamat pembuat program

```

procedure TForm1.btampilClick(Sender: TObject);
begin
label3.Caption:='Borland Delphi 7';
label4.Caption:='Bina Sarana Informatika';
end;

```

2. Untuk keluar dari program user mengklik Button keluar

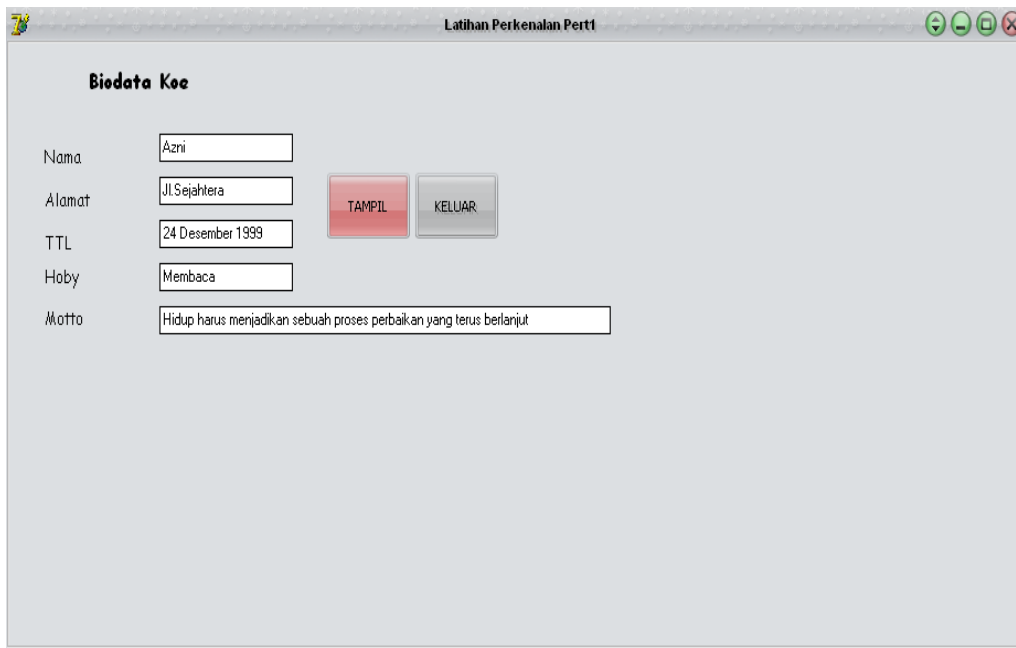
```

procedure TForm1.bkeluarClick(Sender: TObject);
begin
close;
end;

```

1.6. Latihan Membuat Program Biodata

a. Form Setelah dijalankan



Gambar 1.21 Gambar Form Latihan Perkenalan Pert1

b. Ketentuan yang diinginkan...

1. Jika diklik Button Tampil maka muncul biodata anda seperti gambar diatas
2. Jika diklik Button Keluar maka akan keluar dari aplikasi program

Bab II

Operator 1, Deklarasi, Tipe Data dan Mengenal Object I

2.1 Deklarasi

Dalam setiap penulisan bahasa pemrograman deklarasi sangat digunakan apabila dalam penulisan program dibutuhkan indentifier atau tanda pengenal. Identifier pada umumnya di buat oleh programmer yang digunakan untuk mewakili nilai dari suatu object. Identifier yang dikenal dalam Delphi adalah label, konstanta, tipe, fungsi, procedure maupun variabel.

2.1.1. Deklarasi Konstanta

Deklarasi konstanta adalah tanda pengenal dalam Delphi yang mempunyai nilai yang sudah tetap. Definisi konstanta diawali dengan kata baku **Const** diikuti dengan kumpulan indentifier yang diberi sebuah nilai.

Contoh

```
procedure TForm2.etertulisChange(Sender: TObject); const
  nil1:=30000';
begin end;
```

2.1.2. Deklarasi Variabel

Deklarasi variabel adalah tanda pengenal dalam Delphi yang mempunyai nilai yang mana nilai tersebut akan terus berubah selama proses berjalan. Definisi variabel diawali dengan kata baku **Var** diikuti dengan kumpulan indentifier yang diikuti dengan tipe data yang dibutuhkan.

Contoh

```
procedure TForm2.EpraktekKeyPress(Sender: TObject; var Key: Char); var
  praktek,nil2,nil1 ".real;
begin
if (key = #13) then
begin
  nil1 := strtointofloat(etertulis.Text);
```

2. Untuk keluar dari program user mengklik buton close

```
procedure TForm1.TcloseClick(Sender: TObject); begin close; end; end.
praktek:= strtointofloat(epraktek.Text); nil2:=
0.4* praktek; ehpraktek.Text := f
loattostr(nil2); form2.ActiveControl :=
cmi; emurni.Text := f loattostr(nil1 + ni
12); if nil1 > 60 then
  egrade.Text := 'Lulus'
else
  egrade.Text := 'Gagal'
end; end;
```

2.2 Tipe Data

Secara sederhana tipe data dapat didefinisikan dengan istilah tempat untuk menentukan pemberian nilai terhadap suatu variabel sesuai atau tidak dengan nilai yang diberikan oleh user. Dalam versi lain tipe data juga diartikan sebagai batasan terhadap fungsi tanda pengenal terhadap semua nilai yang diterima. logika yang dapat kita berikan adalah ketika kita

menempatkan tanda pengenal **harga** hanya mengenal angka, maka ketika kita memberikan nilai berupa string maka secara otomatis data tersebut akan ditolak karena nilai tersebut tidak dikenali oleh **tipe data** yang diberikan.

2.2.1 Tipe Data Numeric Integer

Tipe data integer merupakan tipe data bilangan bulat yang hanya mengenal bilangan decimal. Dimana tipe data **Integer** tidak mengenal pecahan

Bentuk Umum

```
Var  
  Nil: integer;  
Begin  
  Nil:=5000;
```

2.2.2 Tipe Data Real

Tipe data numeric real adalah tipe data dari suatu tanda pengenal selain mengenal bilangan bulat utuh tipe data ini juga mengenal nilai angka yang mengenal pecahan.

2.5. Mengenal Operator

Delphi mengenal banyak operator, sama seperti bahasa pemrograman yang lain, operator menjadi satu hal penting yang harus ada untuk perancangan program.

2.5.1. Operator Penugasan (assignment Operator)

Symbol operator digunakan untuk melakukan suatu proses atas suatu nilai dengan memberikan nilai baru pada suatu variabel

Lambang operator Penugasan " := "

Bentuk Umum penulisan

```
Var := perintah;
```

Contoh =

```
A:="B";  
Label 1.caption := "Irnawan"  
C:=A + B;
```

2.5.2. Operator Aritmatika

Contoh

B := 15 + 2;	hasil B = 17
B := 15 div 2	B = 7
B := 15/2	B = 7,5
B := 15 * 2	B = 30
B := 15 mod 2	B = 1

Operator aritmatika berfungsi untuk melakukan suatu proses aritmatika yang meliputi perkalian, pembagian, penjumlahan, pengurangan maupun pengurangan terhadap suatu nilai variabel yang tersimpan dalam suatu object, dengan memberikan nilai baru.

Bentuk Umum

```
Var  
  Nihreal;  
Begin  
  Nill:=20,5;
```

2.2.3 Tipe Data String

Tipe data string merupakan salah satu jenis tipe data selain mengenal angka disini tipe data dapat juga mengenala data berupa huruf maupun tanda baca.

Bentuk umum

```
Var  
  Nama: string;  
Begin  
  Nama:='Anton';
```




2.2.4 Tipe Data Char

Secara fungsi tipe data char sama dengan tipe data string tetapi dari segi kapasitas ruang diperoleh tipe data char jauh lebih sedikit karena hanya mengenal 1 karakter.

2.3. Dasar umum merancang Program aplikasi berbasis visual

- Merancang tampilan program (user interface) hal ini meliputi = Form dan toolbox
- Desain properties. Hal ini digunakan untuk merubah tampilan icon yang asli toolbox agar sesuai dengan tampilan yang tampilan yang diinginkan.
- Jendela Code Editor , digunakan sebagai media komunikasi antar object pada form dengan system yang ada.

2.4. Mengenal Komponen label, edit & Button

	Label	Hanya untuk menampilkan Teks
	Edit	Untuk menampilkan dan input data (1 baris)
	Button	Digunakan untuk melakukan eksekusi terhadap suatu proses

2.5.3. Operator String

Digunakan untuk menggabungkan dua teks/string atau lebih. Symbol

yang digunakan = + Contoh = Bentuk I mum

A := teks1 + teks2

Contoh

A: = "Aku"; B

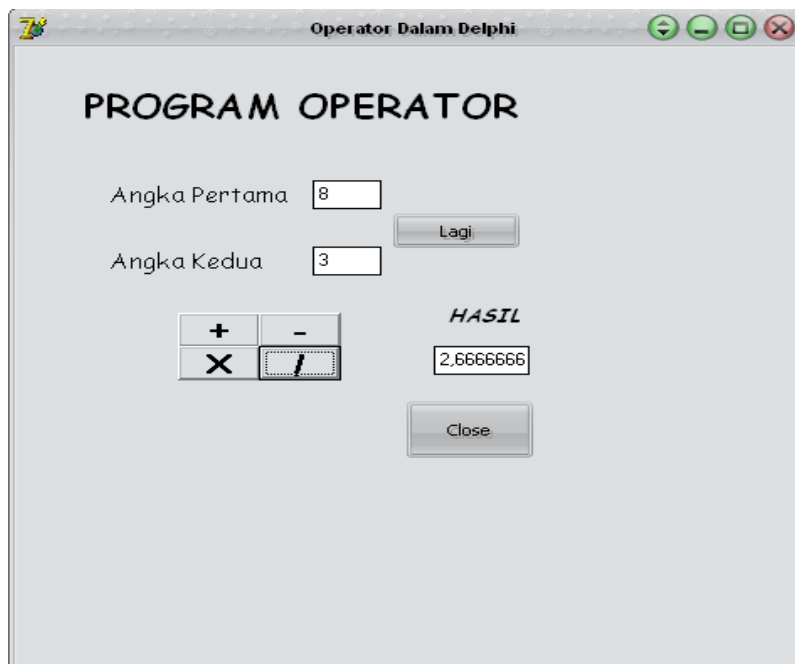
:= "Bisa"; C

:=A + B;

Hasil C = Aku Bisa

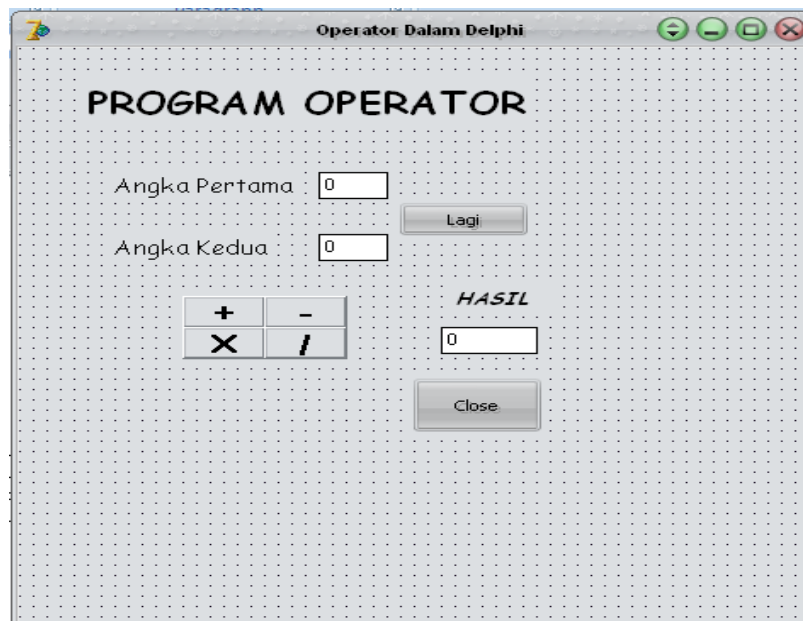
2.5.4. Merancang Aplikasi Dengan Delphi

a. Hasil ketika form dijalankan



Gambar 2.1. Form Operator dalam Delphi

b. Desain Form



Gambar 2.2 Design Form Operator Dalam Delphi

c. Desain Properties

Object	Name	Caption
Label 1	Label 1	PROGRAM OPERATOR
Label 2	Label 2	Angka Pertama
Label 3	Label3	Angka Kedua
Label 4	Label4	Hasil

Object	Name	Text
Edit1	esatu	-
Edit2	edua	-
Edit3	hasil	-

Object	Name	Caption
Button 1	btambah	Tambah
Button2	bkali	Kali
Button3	bbagi	Bagi
Button4	bkurang	Kurang
Button5	blagi	Lagi
Button6	bclose	Close

d. Listing Program

Catatan =

Dalam bahasa pemrograman Delphi semua data yang diinput melalui edit box dideklarasikan dengan tipe data string, sehingga ketika kita akan melakukan proses perkalian atau pengurangan kita tidak bias mengalikan secara langsung edit box yang dimaksud tetap kita perlu merubah menjadi nilai dengan tipe data menjadi nilai dengan dengan tipe data yang dapat dikalikan.

1. Program Tambah

Digunakan untuk menampilkan hasil penjumlahan antara Angka Pertama dengan Angka Kedua, cara double klik pada button tambah dan tuliskan kode berikut ini =

procedure TForm2.btambahClick(Sender: TObject);

```
var
nil1, nil2 : real;
tambah: real; begin
nil1 := strtofloat(esatu.text);
nil2 := strtofloat(edua.text);
tambah := nil1 + nil2;
hasil.text := floattostr(tambah);
end;
```

2. Program Kurang

Digunakan untuk menampilkan hasil pengurangan antara Angka Pertama dengan Angka Kedua, cara double klik pada button kurang dan tuliskan kode berikut ini =

procedure TForm4.bkurangClick(Sender: TObject);

```
var
nil1, nil2: real;
kurang: real;
begin
nil1 := strtofloat(esatu.Text);
nil2 := strtofloat(edua.Text);
kurang := nil1 - nil2;
hasil.Text := floattostr(kurang);
end;
```

3. Program Kali

Digunakan untuk menampilkan hasil Perkalian antara Angka Pertama dengan Angka Kedua, cara double klik pada button kali dan tuliskan kode berikut ini =

```
procedure TForm4.bkaliClick(Sender: TObject);  
var  
    nil1,nil2:real;  
    kali:real;  
begin  
    nil1:=strtofloat(esatu.Text);  
    nil2:=strtofloat(edua.Text);  
    kali:=nil1*nil2;  
    hasil.Text:=floattostr(kali);  
end;
```

4. Program Bagi

Digunakan untuk menampilkan hasil penjumlahan antara Angka Pertama dengan Angka Kedua, cara double klik pada button Bagi dan tuliskan kode berikut ini =

```
procedure TForm2.bbagiClick(Sender: TObject);  
var  
    nil1,nil2:real;  
    bagi:real;  
begin  
    nil1:=strtofloat(esatu.Text);  
    nil2:=strtofloat(edua.Text);  
    bagi:=nil1/nil2;  
    hasil.Text:=floattostr(bagi);  
end;
```

Catatan

Var

```
Nil1,nil2 : real;  
Tambah : real;
```

Var digunakan untuk mendeklarasikan suatu nama varibael yag dibuat oleh programmer yang berfungsi untuk menyimpan nilai atau data selama proses program berjalan.

Nil1,nil2 dan **tambah** adalah nama variabel baru yang dibentuk oleh progmmmer

Real adalah nama dari sekeian banyak tipe data yang dikenal dalam bahasa peograman Delphi. Cakupan tipe data real adalah membaca bilangan angka dalam bentuk decimal.

Strtofloat adalah fungsi yang digunakan merubah nilai dari tipe data string menjadi tipe data numeric.

Floattostr adalah fungsi yang digunakan merubah nilai pecahan menjadi data string.

5. Program untuk keluar

```
procedure TForm2.bcloseClick(Sender: TObject);  
begin  
    messageDlg('Ingin Menutup form'(mterror, [mbok],0);  
    close;  
end;
```

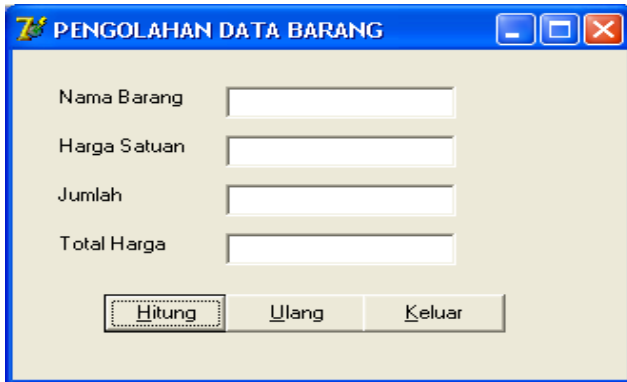
Catatan

MessageDlg adalah penggalan program yang digunakan untuk menampilkan kotak pesan.

Close digunakan untuk menutup form.

2.5.5. Latihan Program Pembayaran

a. Form setelah dijalankan



Gambar 2.3 Gambar Toko

b. Ketentuan yang diinginkan pemilik toko

1. Nama barang, harga barang dan jumlah merupakan media input bagi user mengenai detail nama barang, harga barang serta jumlah yang dibeli.
2. Bayar digunakan untuk menampilkan perkalian dari jumlah yang dibeli dengan harga barang, dengan cara mengklik hitung.
3. Untuk menutup form user tinggal mengklik exit, dengan menampilkan pesan untuk user.

Catatan :

1. Untuk Menambah Form Baru pilih dan klik icon **New Form**
2. Untuk Mengaktifkan Form Yang Pernah Dibuat adalah pilih dan klik **View**



Form

pada jendela View Form pilih form yang ingin diaktifkan dan klik **OK**

3. Untuk Menjalankan Form yang diinginkan dari beberapa form yang telah dibuat adalah dengan cara pilih **Project option**. Pada combo **Project Option**, pilih dan klik form yang diinginkan dengan mengaktifkan Combo Main form. Setelah selesai klik **OK**.

4. Untuk menyimpan semua project pilih dan klik **Save All**.



BAB III

Operator String

3.1. Menggabungkan String

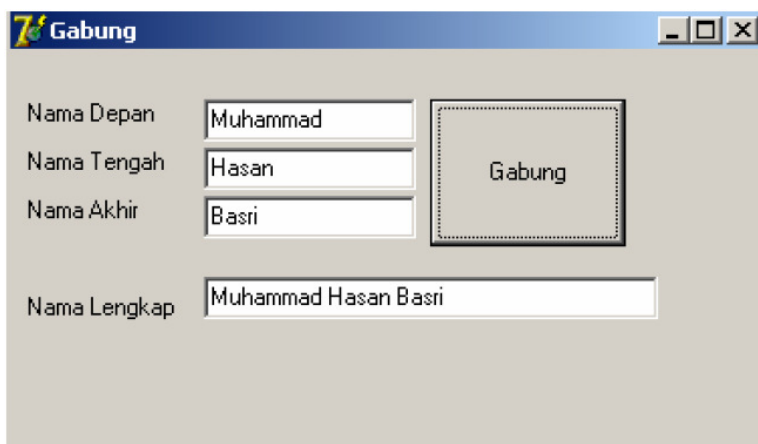
menggabungkan string adalah proses merangkai dua karakter atau lebih menjadi satu kalimat atau kata yang baru. Operator string yang dikenal dalam Bahasa Delphi adalah operator dengan symbol '+'.

a. Model Penulisan

```
var
    nama: string[20];
    saya : string[15];
    namasaya : string[50];
Begin
    Nama := 'Nama Saya adalah = '; Saya :
    = 'Azni';
    Namasaya := 'nama+' '+saya;
    Ehasil.text: = namasaya;
End.
```

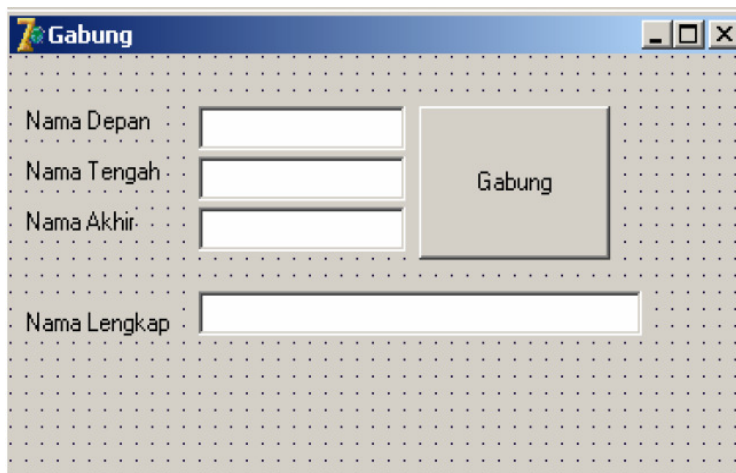
b. Contoh Program

Hasil yang Diperoleh



Gambar 3.1 Form Penggabungan String

DesainForm



Gambar 3.2 Desain Form Penggabungan String

Desain Properties

Object	Caption/Text	Name
Label 1	Nama Depan	Label 1
Label 2	Nama Tengah	Label2
Label 3	Nama Akhir	Label3
Label 4	Nama Lengkap	Label4
Edit1	-	Edepan
Edit2	-	Etengah
Edit3	-	Eakhir
Edit4	-	Egabung
Button 1	Gabung	bgabung

Listing Program

procedure TForm13.bgabungClick(Sender: TObject);

```
var
depan : string[20]; tengah : string[20];
akhir : string[20]; gabung : string[80];
begin
depan := edepan.Text;
tengah := etengah.Text; akhir :=
eakhir.Text;
gabung := depan+' '+tengah+' '+akhir;
elengkap.Text := gabung;
end;
```

3.2. Menghapus Teks

Prosedur standart ini digunakan untuk menghapus atau mengurangi sebagian atau seluruh karakter terhadap string atau teks.

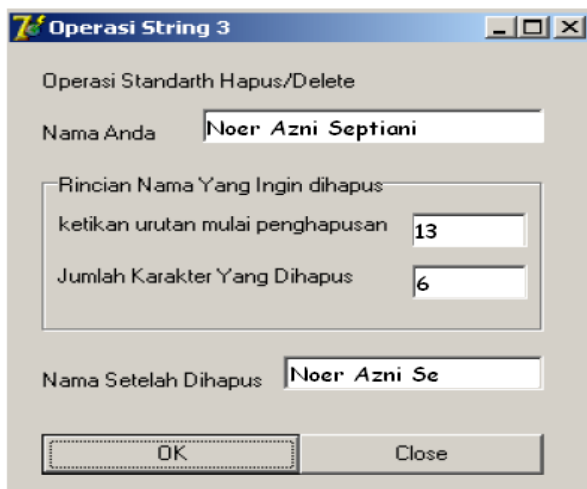
Bentuk umum penulisan

Delete(teks, index, jumlah)

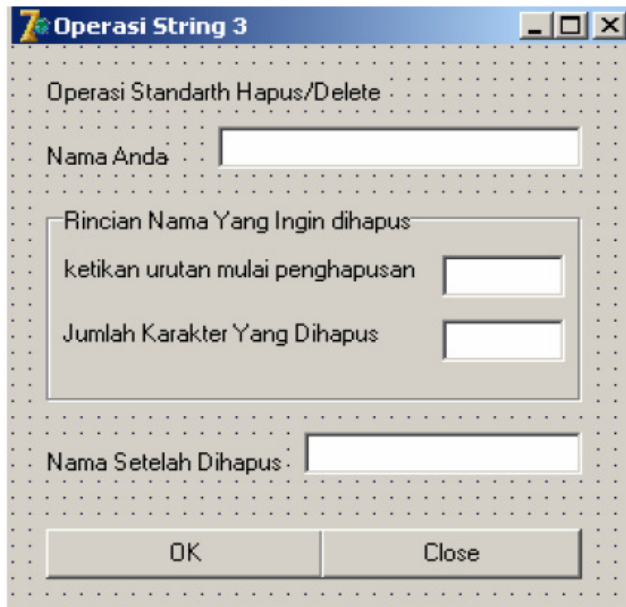
Delete adalah prosedur standart yang digunakan untuk menghapus teks. *Teks* merupakan kalimat atau string yang akan dihapau, *Index* menunjukan posisi awal yang akan dihapus. *Jumlah* menunjukan jumlah karakter yang akan dihapus.

Contoh

a. Hasil Setelah Form Dijalankan



b.Design Form



c. Desain Properties

Object	Caption/Text	Name
Label 1	Nama Anda	Label 1
Label 2	Ketikkan urutan mulai penghapusan	Label2
Label 3	Jumlah Karakter yang dihapus	Label3
Label 4	Nama Sesudah dihapus	Label4
Edit1	-	Enama
Edit2	-	Eurut
Edit3	-	Ejumlah
Edit4	-	Ehasil
Button 1	OK	bOK
Button2	Close	Bclose
Group Box 1	Rincian Nama Yang Ingin Dihapus	Group box 1

d. Listing Program

Program untuk eksekusi perintah penghapusan teks.

procedure TForm9.bokClick(Sender: TObject);

var

karakter : string; pos ,jumlah:

integer;

```
begin
karakter:= enama.Text;
pos := strtoint(eurut.Text);
jumlah := strtoint(ejumlah.Text);
Delete(karakter,pos,jumlah);
ehasil.Text := karakter;
end;
```

Program untuk keluar dari form

```
procedure TForm9.bcloseClick(Sender: TObject);
begin
close;
end;
```

3.3. Mencari Panjang teks (length)

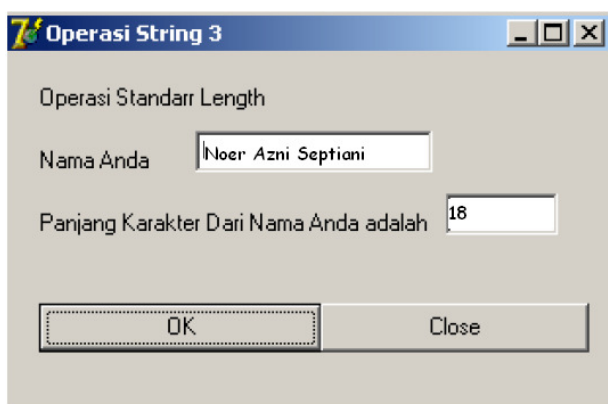
fungsi ini digunakan untuk menghitung panjang atau jumlah karakter dari suatu teks atau kalimat. Dalam menghitung jumlah karakter dalam suatu teks spasi akan dibaca sebagai satu karakter. Sebagai catatan hasil yang diperoleh dari perhitungan adalah bilangan bulat positif.

Bentuk Umum

Length(teks)

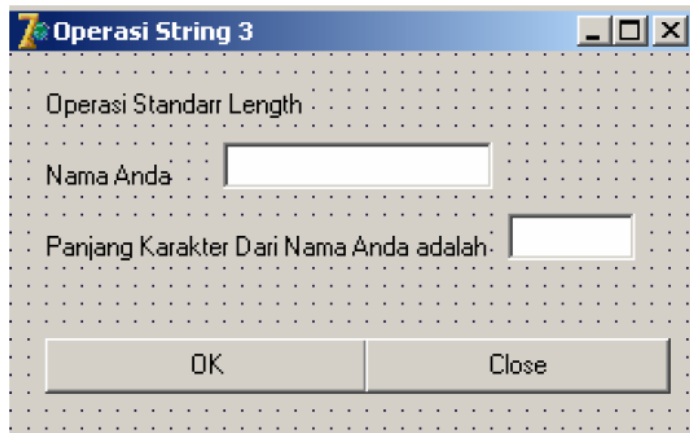
Contoh

a. Hasil setelah form dijalankan



Gambar 3.5 Hasil Form Panjang String

b. Design Form



Gambar 3.6 Design Form Panjang String

d. Desain properties

Object	Caption/Text	Name
Label 1	Operasi Standart Length	Label 1
Label 2	Nama Anda	Label2
Label 3	Panjang Karakter dari Nama Anda	Label3
Edit1	-	Enama
Edit2	-	Ehasil
Button 1	OK	bOK
Button2	Close	Bclose

e. Listing Program

Program untuk meneksekusi pencarian panjang karakter

```
procedure TForm8.bokClick(Sender: TObject);
```

```
var
```

```
nama: string; begin
```

```
nama := enama.Text;
```

```
ehasil.Text := inttostr(length(nama));
```

```
end;
```

Program Untuk menutup Form

```
procedure TForm8.bcloseClick(Sender: TObject);
```

```
begin
```

```
Close;
```

```
End;
```

3.4. Mencari Karakter Pada Teks (POS)

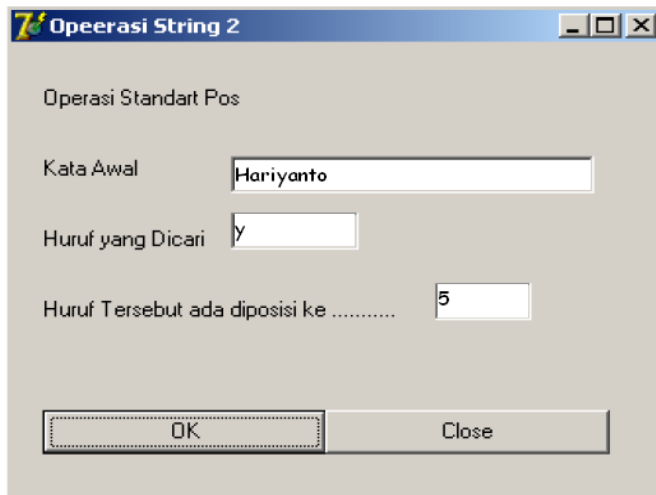
Fungsi ini digunakan untuk mencari letak atau posisi dari suatu karakter dari suatu nilai string. Nilai yang dihasilkan adalah berupa byte.

Bentuk umum

Pos(sustr, string);

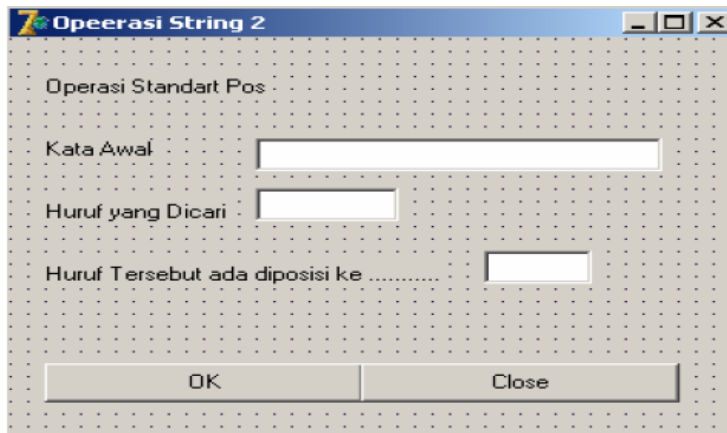
Contoh

a. Hasil setelah dijalankan



Gambar 3.7 Hasil Form Pencarian Huruf

b. Desain Form



c.Design Properties

Object	Caption/Text	Name
Label 1	Operasi Standart Pos	Label1
Label 2	Kata Awal	Label2
Label 3	Huruf yang Dicari	Label3
Label4	Huruf tersebut ada diposisi ke	
Edit1	-	eawal
Edit2	-	ecar
Edit3	-	ehasil
Button1	OK	bOK
Button2	Close	Bclose

Program untuk eksekusi perintah pencarian

procedure TForm7.bokClick(Sender: TObject);

```
var
karakter : string;
kar : string;
hasil : integer;
begin
karakter:= eawal.Text;
kar:= ecari.Text;
hasil := pos(kar,karakter);
ehasil.Text := inttostr(hasil);
end;
```

Program Untuk menutup Form

procedure TForm7.bcloseClick(Sender: TObject);

```
begin
close;
end;
end.
```

3.5. Mengkopi String (Copy)

Fungsi ini digunakan untuk mencetak ulang string atau karakter sebanyak huruf yang dipilih dengan posisi karakter awal yang sudah ditentukan.

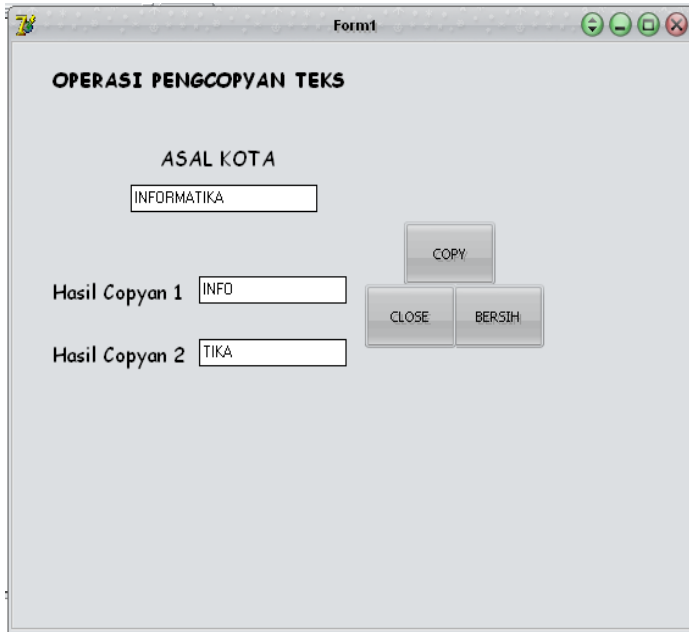
Bentuk Umum

```
Copy(teks, index, jumlah);
```

Penulisan awal diawali dengan perintah copy, dengan diikuti teks yang ditunjukkan teks. Awal pengcopian ditunjukkan dengan index, dan jumlah karakter yang akan di tulis ulang ditunjukkan deengan perintah jumlah.

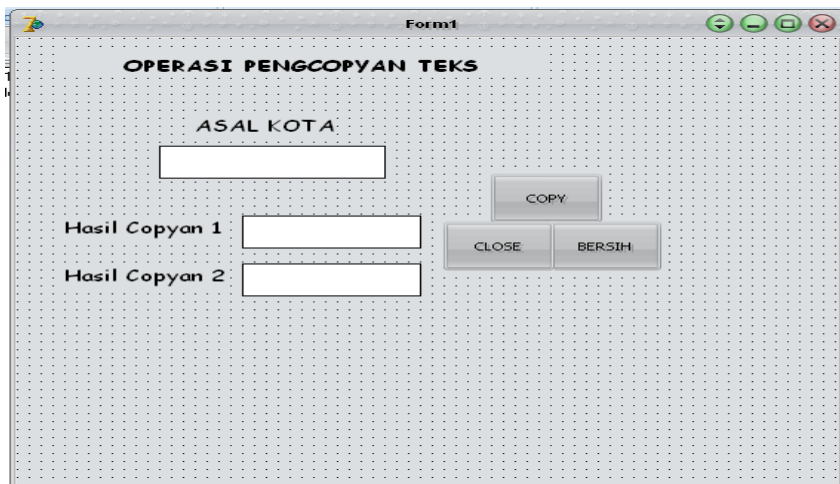
Contoh

a. Hasil setelah form dijalankan



Gambar 3.9 Hasil Form Copy Teks

b.Design Form



Gambar 3.10 Design Form Copy Teks

c.Design Properties

Object	Caption/Text	Name
Label 1	Operasi Copy Teks	Label1
Label 2	Kata Awal	Label2
Label 3	Hasil Copy 1	Label3
Label4	Hasil Copy 2	Label4
Edit1	-	eawal
Edit2	-	Ehasil1
Edit3	-	Ehasil2
Button1	OK	bOK
Button2	Close	Bclose

Program untuk mengeksekusi program

```
procedure TForm6.bokClick(Sender: TObject);
```

```
var
```

```
karakter:string; begin
```

```
karakter := eawal.Text;
```

```
ehasill.Text := copy(karakter,1,4);
```



```

ehasil2.Text := copy(karakter,8,4)
end;

```

Program untuk menutup Form

```

procedure TForm6.bcloseClick(Sender: TObject);
begin
close;
end;

```

3.6. Konversi Karakter ke Ascii (Chr)

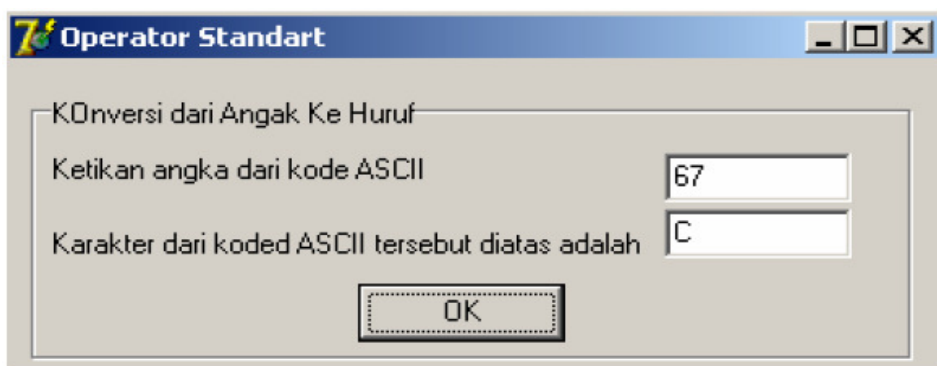
Fungsi ini digunakan untuk mengkonversi Kode Ascii menjadi nilai karakter atau huruf.

Bentuk Umum

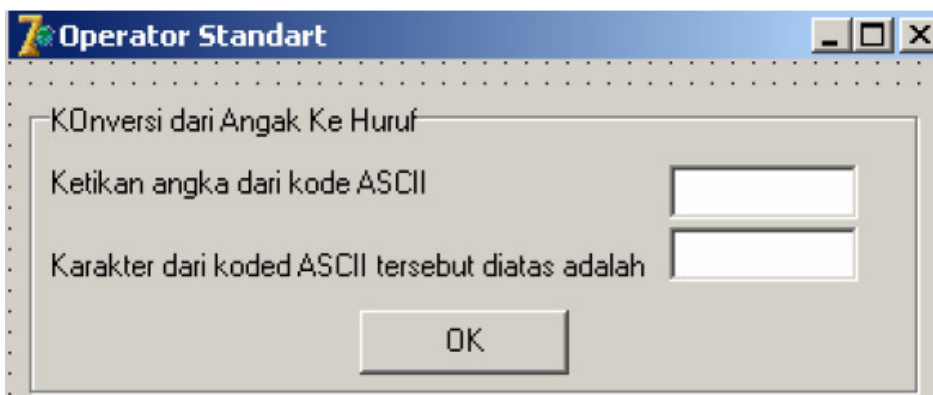
Chr (karakter)

Contoh

a. Hasil setelah dijalankan



b.Design Form



c. Desain properties

Object	Caption/Text	Name
Label 1	Ketikan angka dari kode ASCII	Label 1
Label 2	Karakter dari kode Ascii tersebut adalah	Label2
Edit1	-	eanangka
Edit2	-	ehuruf
Button 1	OK	bOK
Groupbox1	Konversi dari Angka ke huruf	Groupbox1

d. Listing Program

```
procedure TForm1.bokClick(Sender: TObject);  
var  
angka : integer;  
begin  
angka:= strtoint(eangka.Text);  
ehuruf Text := chr(angka);  
end;
```

BAB IV

Percabangan dan Mengenal Object II (combo box, radio button)

4.1. Operator Percabangan

percabangan adalah merupakan operator yang digunakan untuk menentukan pilihan terhadap beberapa pilihan yang ada.

Dalam bahasa pemrograman Delphi mengenal dua operator percabangan **4.1.1.**

Percabangan If

merupakan operator percabangan yang digunakan untuk menentukan pilihan atas beberapa kondisi yang merupakan syarat terhadap pilihan yang sudah ditentukan.

Ada dua model percabangan if

a. Percabangan untuk kondisi pilihan tunggal

Merupakan operator percabangan yang digunakan untuk menentukan sebuah pilihan dengan kondisi tunggal

Bentuk Umum

If Syarat **then** hasil;

Contoh

If Nilai > 80 **then** keterangan = 'Lulus';

b. Percabangan untuk kondisi majemuk

Merupakan operator percabangan yang digunakan untuk menentukan pilihan dengan kondisi yang harus dipeuhi lebih dari satu.

Bentuk Umum

If Syarat 1 **then**

Hasil1

Else

If syarat2 **then**

Hasil2

Else

.....

.....

end;

Contoh

If nilai > 80 **then**

```

Grade = "A" Else
if nilai > 70 then
  grad= "B" else
if nilai > 60 then grade= "C"
else
grade = "E";

```

4.1.2. Percabangan case

case of adalah merupakan metode lain dari sebuah percabangan, berfungsi sama seperti fungsi if yaitu untuk melakukan seleksi atas beberapa pilihan dengan kondisi sebagai syarat yang harus terpenuhi. Secara fungsi scee dan if tidak ada perbedaan tetapi untuk penulisan fungsi case lebih mudah diterapkn untuk pilihan atau kondisi lebih dari satu.

Bentuk Umum fungsi Sace OF

```

Case <variabel> of
  <pilihan ke 1> ; hasil 1;
  <pilihanke2>; hasil2;
  <pilihanke3>hasil3;
  .....
  <pilihan ke n>; hasiln;

```

end;

atau fungsi case of bisa juga diberikan **else** untuk pilihan terakhir.

Bentuk Umum

```

Case <variabel> of
  <pilihan ke 1>;hasil1;
  <pilihan ke 2>;hasil2;
  <pilihanke3>;hasil3;
  .....
  else
  hasiln;

```

end;





Contoh

```

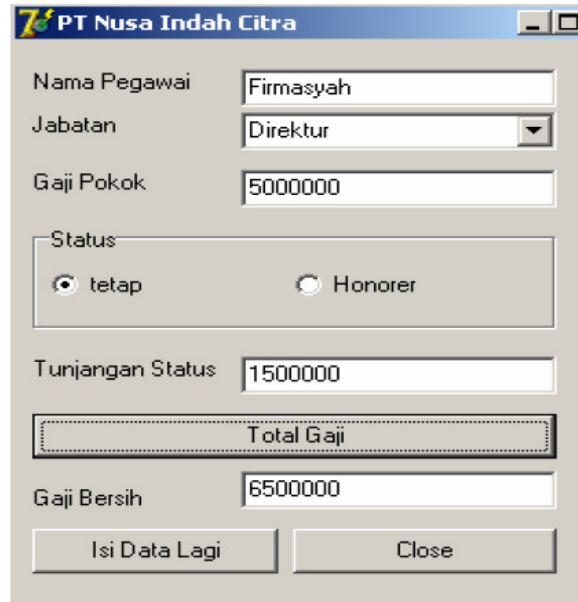
Case Bilangan of
1:showmessage('Angka 1');
2:showmessage('Angka 2');
3:showmessage('Angka 3');
end;

```

4.2. Mengenal Object II (combo box, list box dan radio button)

1		Checkbox	Digunakan untuk menentukan pilihan lebih dari satu
2		Radio Button	Digunakan untuk menentukan pilhan, tetapi hanya satu pilhan yang bisa digunakan
3		List Box	Menmpilkan pilihan dalam bentuk list
4		Combo Box	Menampilkan pilihan dalam bentuk popup

a. Hasil Yang Diperoleh



Gambar 4.1 Gambar Perhitungan Gaji

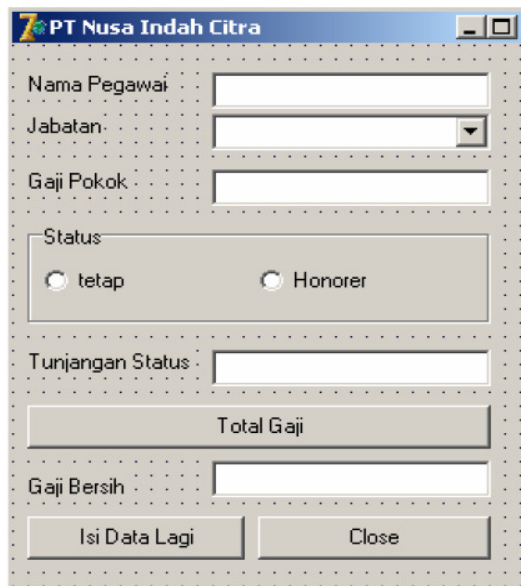
Ketentuan Perhitungan Gaji

1. Dalam Straktur organisasi perusahaan ketentuan gaji perusahaan terbagi menjadi 3 straktur utama

Jabatan	Gaji Pokok
Direktur	5000000
Manager	3000000
Karyawan	1000000

2. Besarnya tunjangan ditentukan oleh status kep

b. Desian Form



Gambar 4.2. Desain Form Gaji

c. Desain Properties

Object	Name	Caption
Label 1	Label 1	Nama Pegawai
Label 2	Label2	Jabatan
Label 3	Label3	Gaji Pokok
Label 4	Label4	Tunjangan Status
Label5	Label5	Gaji Bersih
GroupBox	GroupBox1	Status
RadioButton1	Rtetap	Tetap
RadioButton2	Rhonor	Honor
Object	Name	Text
Edit1	enama	-
Edit2	epokok	-
Edit3	etunjangan	-
Edit4	ebersih	-
Combo Box	Cjabatan	-

Object	Name	Caption
Button1	Btotal	Total Gaji
Button2	Blagi	Isi Data Lagi
Button3	Bclose	Close

d. Listing Program

> Memberikan pilihan pada combo box sesuai dengan pilihan pada jabatan

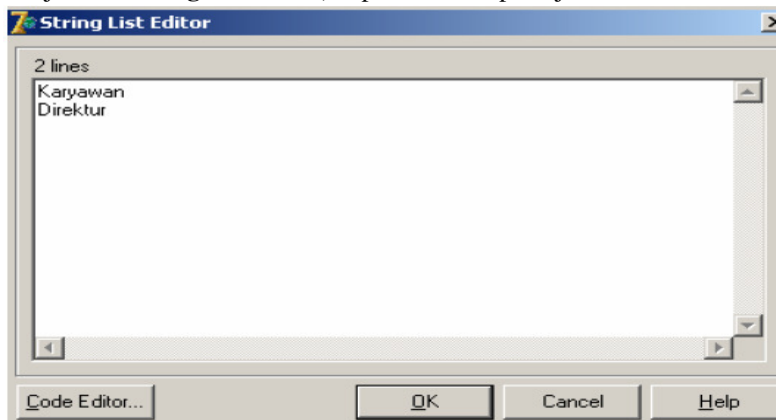
Dengan Menggunakan Object Properties

Langkah - langkahnya

1. aktifkan combo box yang akan dipilih
2. pada properties pilih dan klik item, lalu klik command (...)



3. Pada jendela **String list Editor**, Seperti terlihat pada jendela berikut



4. Ketikkan kata sebagai kata pilihan pada jendela seperti contoh tersebut diatas.
5. Setelah selesai klik OK

Melalui jendela code editor

Double click pada form, sebarang tempat dan ketikkan program berikut ini.

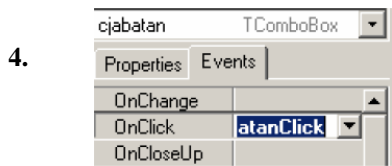
```

procedure TForm1.FormCreate(Sender: TObject);
begin
  cjabatan.Items.Add('Direktur'); cjabatan.Items. Add('
  Manager'); cjabatan.Items.Add(' Karyawan');
end;

```

> **Program untuk mencari gaji pokok**

1. Aktifkan combo box untuk jabatan
2. Pada pada jendela properties pilih **event**,
3. pilih dan aktifkan **onclick**, Seperti yang terlihat seperti gambar berikut ini.



5. Double click pada combo **onclick**.
6. Ketikan program berikut ini

```
procedure TForm3.cjabatanClick(Sender: TObject); begin
  if cjabatan.Text = 'Direktur' then
    epokok.Text := '5000000' else
  if cjabatan.Text = 'Manager' then
    epokok.Text := '2000000' else
  if cjabatan.Text = 'Karyawan' then
    epokok.Text := '1000000' else
    epokok.Text := '0'
  end;
```

> **Program menghitung Tunjangan Karyawan Tetap**

Double click pada option tetap dan ketikan program berikut ini

```
procedure TForm3.etetapClick(Sender: TObject);
begin
  if etetap.Checked = true then a := strtol(loat(epokok.Text)); hasil
    :=a*0.3;
    etunjangan.Text := f loattostr(hasil)
  end;
```

> **Program menghitung Tunjangan Karyawan Honorer**

Double click pada option honorer dan ketikan program berikut ini

```
procedure TForm3.rhonorerClick(Sender: TObject);
begin
  if rhonorer.Checked = true then a := strtol(loat(epokok.Text));
    hasil :=a*0.1;
    etunjangan.Text := f loattostr(hasil)
  end;
```

> **Program menghitung Gaji Bersih**

Double click pada command **total gaji**, dan ketikan program berikut ini:

```
procedure TForm3.btotalClick(Sender: TObject);
begin
  a := strtol(loat(epokok.Text)); b := strtol(loat(etunjangan.Text)); c
    := a+b;
    ebersih.Text := f loattostr(c);
  end;
```

> **Program bersih**

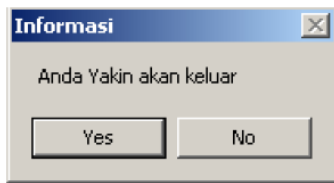
Double click pada **Isi Data Lagi**, dan ketikan program berikut ini

```

procedure TForm3.blagiClick(Sender: TObject);
begin
cjabatan.Text := 'Jabatan'; enama.Text := ''; ebersih.Text :=
'0'; etunjangan.Text := '0'; epokok.Text := '0';
etetap.Checked := false; rhonorer.Checked := false;
end;

```

> Program Close dengan pesan, seperti tampilan sebagai berikut.



Gambar 4.3 Form Pesan

```

procedure TForm3.bcloseClick(Sender: TObject);
begin
if (application.MessageBoxC Anda Yakin akan keluar', 'Informasi' (MB_YESNO)=
IDYES)then
close;
end;

```

Program Biaya Rumah Sakit

Buatlah program untuk mencari biaya ujian untuk calon siswa pada suatu perguruan tinggi. Tampilan yang diinginkan adalah sebagai berikut



Gambar 4.4 Form Penilaian Siswa

Ketentuan Soal!!!

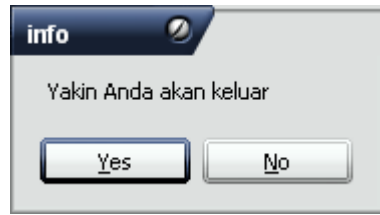
❖ Gunakan fungsi IF

Kode_Dokter	Nama_Dokter	Spesialis	Biaya Periksa
DRJ01	Raffa	Jantung	2000000
DRA02	Sheilla Anisa	Anak	150000
DRM03	Putri	Mata	1000000

❖ Total bayar di dapat dari harga satuan * jumlah Beli

❖ Kembali muncul otomatis ketika bayar di input, yang didapat dari bayar-total bayar

Jika di klik Keluar Muncul pesan seperti di bawah ini



BAB V

Perulangan dan Megenal Input Box

5.1. Perulangan

Merupakan control program yang digunakan untuk suatu proses yang akan berjalan terus menerus. Kondisi perulangan merupakan proses berjalannya program secara terus menerus dan akan berhenti ketika proses mendapatkan kondisi yang sudah ditentukan.

Di dalam Delphi mengenal 3 jenis perulangan

a. Perulangan For - To - Do

Perulangan dengan statement for adalah perulangan yang digunakan untuk melakukan suatu proses dalam sebuah blok program. Proses perulangan For - To -Do dimulai dengan nilai terkecil ke besar.

Bentuk Umum

For variabel := nilai awal to nilaiakhir statement

Catatan : semua variabel yang berhubungan dengan perulangan harus mempunyai tipe data sama.

Contoh:

```
procedure TForm4.Button1Click(Sender: TObject);
```

```
var  
  i: integer; begin  
for i := 1 to 5 do edit1.SelText := inttostr(i);  
end;  
end.
```

Bila Program tersebut dijalankan maka nilai i akan ditampilkan pada edit1 dengan hasil sebagai berikut = **12345**

b. Perulangan For - DownTo-Do

Perulangan For-Dowbto-Do adalah perulangan yang menghitung suatu proses dengan nilai awal besar dan nilai akhirnya lebih kecil, maka variabel sebagai control program yang diperoleh adalah dari besar ke kecil.

Bentuk Umum

For Variabel := nilai-awal Downto Nilai-akhir Do Pernyataan

Contoh

```
procedure TForm4.Button2Click(Sender: TObject);
```

```
var  
  i: integer;  
  begin  
for i := 5 Downto 1 do  
edit2.SelText := inttostr(i); end;
```

Bila program tersebut dijalankan maka hasil yang diperoleh adalah sebagai berikut;

54321

c. Perulangan While Do

Perulangan **While Do** adalah statement perulangan akan terus melakukan suatu proses selama kondisi/syarat yang ditentukan bernilai benar.

Bentuk Umum

Statement While - Variabel Syarat - Do Pernyataan Contoh

```
procedure TForm4.Button3Click(Sender: TObject); var
i : integer;
begin
  i := 0;
  while i < 5 do
  begin
    edit3.SelText := inttostr(i); i := i + 1;
  end;
end;
```

Bila program tersebut dijalankan maka hasil yang akan diperoleh adalah sebagai berikut:

101234

d. Perulangan Repeat Until

Perulangan **repeat until** digunakan untuk mengulang suatu kondisi sampai (until) kondisi bernilai salah.

Bentuk Umum

Repeat - Statement/penyataan — Until — ungkapan logika

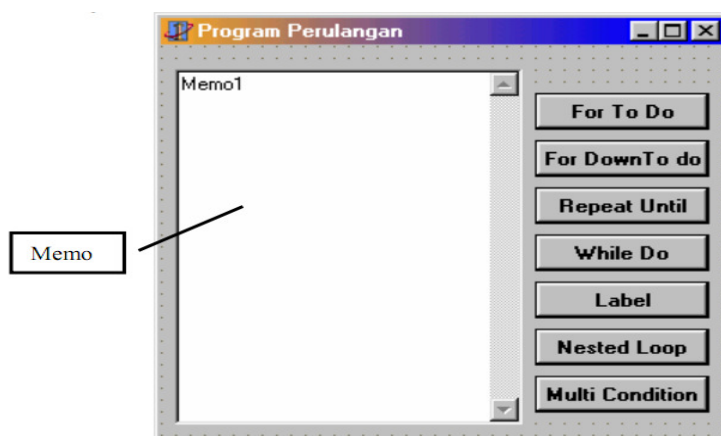
Latihan 1..

Perulangan adalah suatu cara untuk mengulang satu atau sekumpulan perintah sampai mencapai kondisi tertentu.

Dalam Delphi terdapat beberapa perulangan yaitu :

1. Perulangan menggunakan For To Do
2. Perulangan menggunakan For DownTo Do
3. Perulangan menggunakan Repeat Until
4. Perulangan menggunakan While Do
5. Perulangan menggunakan Label
6. Perulangan tersarang (perulangan dalam perulangan)
7. Perulangan dengan banyak kondisi

Untuk lebih jelasnya, buatlah form seperti berikut dengan menggunakan 1 buah memo dan 7 buah BitBtn. Memo digunakan sebagai tempat untuk melihat hasil perulangannya.



Catatan :

Komponen Memo () umumnya digunakan untuk pemasukan data string yang bisa menampung data dalam bentuk beberapa paragraf (banyak baris). Komponen ini dapat kita samakan dengan Notepad. Isi dari komponen ini ada di properti Lines.

Method-method yang banyak dipakai yang dimiliki oleh memo adalah :

– Clear, gunanya untuk mengosongkan memo.

Contoh :

```
Memo1.Clear;
```

– Lines.Add, gunanya untuk memasukan/menambahkan baris di Memo.

Inputan untuk method add harus berupa string. Jika yang ingin ditampilkan berupa angka, maka harus dikonversikan dulu.

Contoh :

```
Memo1.Lines.Add('Text Yang Ditambahkan');
```

```
A:=100;
```

```
Memo1.Lines.Add('Nilai A adalah : '+inttostr(A));
```

– Lines.Delete, gunanya untuk menghapus suatu baris.

Contoh :

```
Memo1.Lines.Delete(0); // Menghapus baris ke-1
```

1. For To Do

Perulangan ini berjalan dengan menggunakan suatu variabel counter yang akan bertambah secara otomatis ketika perintah yang diulang telah selesai dikerjakan.

Bentuk umum dari perulangan ini adalah :

```
For counter := nilai awal To Nilai akhir Do
```

```
Begin
```

```
Perintah;
```

```
Perintah;
```

```
End;
```

Perulangan ini akan berulang selama nilai variabel counter masih lebih kecil atau sama dengan nilai akhir.

Contoh Program yang ditulis dalam event OnClick dari tombol For To Do

```
procedure TForm1.BitBtn1Click(Sender: TObject);
```

```
var
```

```
  i:integer;
```

```
  c:char;
```

```
begin
```

```
  Memo1.Clear;
```

```
  for i:=1 to 10 do
```

```
    Memo1.Lines.Add(IntToStr(i));
```

```
  for c:='A' to 'Z' do
```

```
    Memo1.Lines.Add(c);
```

```
end;
```

2. For DownTo Do

Perulangan ini sama seperti perulangan dengan menggunakan For To Do tetapi arah isi counternya berkurang (mundur/turun).

Bentuk umum dari perulangan ini adalah :

```
For counter := nilaiawal DownTo Nilaiakhir Do
```

```
Begin
```

```
Perintah;
```

```
Perintah;
```

```
End;
```

Perulangan ini akan berulang selama nilai variabel counter masih lebih besar atau sama dengan nilai akhir. Nilai awal suatu counter harus lebih besar atau sama dengan daripada nilai akhirnya.

Contoh Program yang ditulis dalam event OnClick dari tombol For To Do

```
procedure TForm1.BitBtn2Click(Sender: TObject);
```

```
var
```

```
  i:integer;
```

```
  c:char;
```

```
begin
```

```
  Memo1.Clear;
```

```
  for i:=10 downto 1 do
```

```
    Memo1.Lines.Add(IntToStr(i));
```

```
  for c:='Z' downto 'A' do
```

```
    Memo1.Lines.Add(c);
```

```
end;
```

3. Repeat Until

Perulangan ini bekerja sampai kondisi yang diinginkan tercapai. Perulangan ini bisa digunakan sebagai perulangan yang menggunakan banyak kondisi keluar (multi condition loop).

Bentuk umum :

```
Repeat  
  Perintah;  
  Perintah;  
Until Kondisi;
```

```
procedure TForm1.BitBtn3Click(Sender: TObject);  
var  
  i:real;  
begin  
  Memo1.Clear;  
  i:=0;  
  repeat  
    i:=i+0.5;  
    Memo1.Lines.Add(FloatToStr(i));  
  until i=10;  
end;
```

Perulangan ini berulang sampai kondisi yang ada setelah until mempunyai nilai true. Jadi repeat until berjalan selama kondisi salah sampai kondisi benar.

Pengecekan kondisi berada di bagian bawah yaitu setelah until sehingga semua program yang memakai perulangan ini akan melakukan minimal 1 kali proses perulangan.

4. While Do

Perulangan ini mirip dengan perulangan Repeat Until tetapi pengecekannya berada di awal sebelum melakukan proses yang berulang. Perulangan ini berjalan selama kondisi pengecekan bernilai true. Jika kondisi telah mempunyai nilai false maka perulangan tidak dilakukan lagi.

Bentuk umum dari While Do adalah seperti berikut :

While kondisi do

```
Begin  
  Perintah;  
  Perintah;
```

End;

Contoh program while do yang ditulis pada event onclick dari tombol While Do.

```
procedure TForm1.BitBtn4Click(Sender: TObject);  
var  
  i:real;  
begin  
  Memo1.Clear;  
  i:=0;  
  while i<10 do  
    begin  
      i:=i+0.5;  
      Memo1.Lines.Add(FloatToStr(i));  
    end;  
end;
```

5. Label

Label adalah suatu perintah untuk memandai suatu baris program. Untuk pindah ke posisi yang ditandai oleh label, maka digunakan perintah Goto.

Untuk lebih jelasnya lihatlah program berikut yang ditulis dalam event OnClick dari tombol Label.

```
procedure TForm1.BitBtn5Click(Sender: TObject);  
label  
  A;  
var  
  i:integer;  
begin  
  Memo1.Clear;  
  i:=0;  
  A:  
    i:=i+1;  
    Memo1.Lines.Add(IntToStr(i));  
    if i<10 then  
      Goto A;
```

End;

6. Perulangan Tersarang (perulangan dalam perulangan)

Perulangan tersarang (nested loop) adalah suatu perulangan dimana di dalam perulangan tersebut ada perulangan lain. Perulangan yang di ada di bagian terdalam, maka akan dikerjakan dulu sampai selesai baru mengerjakan perulangan yang ada diluarnya.

Nested loop bisa merupakan kombinasi antara For To Do, For DownTo Do, Repeat Until, While

Do.

Contoh Program

```
procedure TForm1.BitBtn6Click(Sender: TObject);
var
  i,j:integer;
begin
  Memo1.Clear;
  for i:=1 to 5 do
  begin
    j:=0;
    while j<5 do
    begin
      j:=j+1;
      Memo1.Lines.Add(IntToStr(i)+' X '+IntToStr(j)
        +' = '+IntToStr(i*j));
    end;
    Memo1.Lines.Add('-----');
  end;
end;
```

7. Perulangan dengan banyak kondisi

Perulangan dengan banyak kondisi adalah suatu perulangan dimana berjalannya suatu perulangan tidak hanya berdasarkan suatu kondisi saja tetapi bisa lebih dari satu kondisi. Operator yang bisa digunakan biasanya seperti And, Or, Not dan lain-lain.

Contoh Program :

```
procedure TForm1.BitBtn7Click(Sender: TObject);
var
  i:integer;
  total:integer;
begin
  Memo1.Clear;
  total:=0;
  i:=0;
  repeat
    i:=i+1;
    Total:=Total+i;
    Memo1.Lines.Add('Isi I = '+IntToStr(i));
    Memo1.Lines.Add('Isi Total = '+IntToStr(Total));
  until (total>150)or(i>25);

end;
```

5.4. Study Kasus Perulangan

a. Hasil Setelah Dijalankan

The screenshot shows a Windows-style application window titled "Penjualan". It features several sections for data entry:

- Daftar Penjualan:** Includes a text box for "Total Item yang Dibeli" containing the number "2" and an "OK" button.
- Tanggal:** A date field showing "Hari ini 06-09-2005".
- No Faktur:** An empty text box for invoice number.
- Daftar Barang Yang Dibeli:** A list box containing "Gatsby" and "Biore Body Form".
- Daftar Harga Barang:** A list box containing "4500" and "5000".
- Total Penjualan:** A text box at the bottom right showing the calculated total of "9500".
- Buttons:** "ISI DATA LAGI" and "CLOSE" buttons are located at the bottom of the window.

Gambar 5.1 Form Penjualan Ketentuan Dan Alur Logika

Total item digunakan sebagai pembatas berapa

kali, pelanggan akan membeli barang.

1. Ketika jumlah item diberikan nilai maka ketika kita klik OK akan ditampilkan kotak pesan (input Box) untuk menginput nama barang dan harga barang yang dibeli. Begitu seterusnya sampai kondisi dari total item terpenuhi.
2. Total Penjualan diperoleh dari penjumlahan seluruh harga barang yang dibeli.
3. isi data lagi diberikan untuk mengulang pembelian.
4. Close untuk menutup form.
5. Untuk tanggal ditampilkan secara otomatis.

b. Desain Form

Gambar 5.2 Desain Form Penjualan

```
procedure TForm4.Button4Click(Sender: TObject); var  
  i : integer;  
begin i := 1;  
  repeat i := i + 2;  
    edit4.SelText := inttostr(i); until i = 15;  
end;
```

Bila program tersebut dijalankan maka akan diperoleh hasil sebagai berikut:

13579111315

5.2. Mengenal Input Box

Input Box adalah sebuah kotak pesan bagi user. Selain itu juga bisa digunakan untuk menginput data.

Bentuk umum

Var

Identifier: typedata;

begin

Identifier := inputbox('string_title','string_subtitle','');

end;

Contoh

Var

Inputnama : string;

Begin

Inputnama := inputbox(TInfo,'Ketikan Nama',''); End;

c. Desain Properties

Object	Caption	Name
Label 1	Total Item yang dibeli	Label 1
Label 2	Tanggal	Label2
Label 3	NO Faktur	Label3
Label 4	Total Penjualan	Label4
GroupBox1	Data Penjualan	Groupbox1
GroupBox2	Daftar Nama Barang Yang Debeli	Groupbox2
Groupbox3	Daftar Harga Barang	Groupbox3
Command 1	OK	Cmdok
Command2	Isi Data Lagi	Cmdlagi
Command3	Close	cmdclose

Object	Text	Name
Edit1	-	eitem
Edit2	-	enofaktur
Edit3	-	etgl
Edit4	-	Ettotal
ListBox1	-	lnama
Listbox2	-	lharga

d. Listing Program

Program untuk menampilkan tanggal secara otomatis ketika form dijalankan

```
procedure TForm5.FormCreate(Sender: TObject); begin
DateSeparator := '-'; ShortDateFormat := 'mm/dd/yyyy';
etanggal.Text := 'Hari ini' + DateToStr(Date);
end;
```

Program command OK

```
procedure TForm5.bokClick(Sender: TObject); var
inputnama : string;
inputharga : string; i :
integer; j : integer; harga :
real; begin
harga := 0; j := 0;
j:= strtoint(eitem.Text);
if j <= 0 then
begin
showmessage('Data Tidak boleh lebih kecil dari Nol');
exit;
end else
begin
for i := 1 to j do begin
inputnama := inputbox('INput','Ketikan Nama Barang,');
inputharga := inputbox('INput','Ketikan Harga Barang,');
Inama.Items.Add(inputnama);
Iharga.Items.Add(inputharga);
harga := harga + strtoint(inputharga);
end;
end;
etotal.Text := floattostr(harga); end;
```

Program Untuk Isi Data Lagi

```
procedure TForm5.clagiClick(Sender: TObject); begin  
eitem.Text := ""; enofaktur.Text := " ;  
Inama.Clear; lharga.Clear; etotal.Text := " ;  
end;
```

Program Untuk Menutup Form

```
Close;
```

5.3 Aplikasi Konversi Suhu

a. hasil yang Diinginkan

Mencari KONversi Suhu dari Celcius ke farenheitt	
Menentukan Batasan Awal (dalam Celcius)	
Batas Awal	5
Batas Akhir	20
Penambahan	5
OK	
Daftar Angka Dalam KONversi Suhu Yang diperoleh	
Hasil Dalam Celcius	Hasil Dalam Farenheit
5	41
10	50
15	59
20	68
Close	

Gambar 4.4 Hasil Form Penjualan

b. Ketentuan Program

1. Batas Awal dan Batas akhir diberikan untuk mengetahui nilai awal dan akhir derajat celcius yang ingin cetak.
2. Penambahan diberikan untuk memberikan penambahan dari nilai awal untuk mencapai nilai akhir.
3. Close adalah untuk keluar dari program
4. Rumus Mencari Fahrenheit adalah $= 1.8 * \text{Celcius} + 32$

BAB VI

Prosedure

6.1. Prosedur

Prosedur adalah suatu program terpisah dan berdiri dalam suatu blok program dan berfungsi sebagai sebuah sub program (program bagian). Penulisan prosedur diawali dengan kata **Procedure** pada bagian deklarasi program dan cukup menuliskan nama prosedur yang dibuat pada bagian **Implementasi**.

Alasan penggunaan prosedur

1. Digunakan untuk penggalan program yang akan digunakan secara berulang-ulang dalam suatu proses program.
2. Digunakan untuk memecah - memecah program menjadi sebuah modul program, sehingga listing program menjadi lebih sederhana.

Syarat penulisan nama procedure

1. harus diawali dengan karakter.
2. untuk nama prosedur dengan menggunakan dua kata atau lebih penulisannya tidak boleh menggunakan spasi, harus digabung atau dihubungkan dengan underscore (_)
3. tidak mengenal tanda baca.

Contoh

a. Pendeklarasian Prosedur

Penulisan pendeklarasian sebuah prosedur dalam Delphi ditempatkan setelah kata **public** agar dapat dikenali oleh semua object yang ada dalam proses program tersebut. Penulisannya diawali dengan kata **Procedure** diikuti nama procedure.

Public

Procedure bersih;

b. Penulisan prosedur

Penulisan program prosedur terdapat dalam bagian **implementation**,

Procedure TForm1.Bersih;

Begin

 Ename.text := ";

 Ealamat.text := ";

 Ekota := ";

End;

c. Pemanggilan Prosedur

Pemanggilan prosedur terdapat dalam bagian implementation, pada umumnya pemanggilan prosedur terdapat dalam suatu ruang lingkup object yang mempunyai sebuah event. Cara pemanggilannya cukup hanya dituliskan nama prosedurnya saja.

Procedure TForm1.button1click(sender: TObject); Begin

 Bersih;

End;

atau ketika program prosedur digunakan pada object yang lain untuk memanggil program yang sama kita hanya cukup menuliskan nama prosedurnya tanpa dibutuhkan pendeklarasian prosedur baru.

```

Procedure TForm1.enamakeypress(sender: TObject; var key.char);
Begin
    If key = #13 then
        Bersih;
    End;

```

BAB VII

FUNGSI

7.1 Pengertian Fungsi

Pengertian secara umum mengenai fungsi adalah merupakan sebuah penggalan program yang terpisah dari program utama dan berfungsi sebagai sebuah program bagian dari program utama. Penulisan fungsi diawali dengan kata cadangan *function* dan dideklarasikan dalam bagian deklarasi fungsi. Dan penulisan program fungsi ditempatkan pada program utama. Satu hal yang perlu diperhatikan dalam penulisan fungsi adalah harus diikuti dengan tipe datanya.

Bentuk Umum penulisan Fungsi

identifier(daftar-parameter): type; Contoh

Function hitung(var a,b : real) :real;

Penulisan blok fungsi diawali dengan kata cadangan *begin* dan diakhiri dengan *end*;

7.2. Fungsi Tunggal

fungsi tunggal merupakan suatu fungsi dimana proses pemanggilan dirinya sendiri tanpa melalui fungsi yang lain atau fungsi yang tidak terdapat dalam fungsi yang lain.

Contoh

a. Hasil yang diinginkan

The screenshot shows a Windows application window with the title bar 'FORM BALOK DENGAN FUNGSI'. The main content area has a dotted grid background. At the top center, the text 'Luas Balok' is displayed. Below this, there are four input fields arranged vertically, each with a label to its left: 'Panjang', 'Lebar', 'Tinggi', and 'Luas'. To the right of these input fields, there are two buttons stacked vertically, labeled 'HITUNG' and 'HAPUS'. The window also features standard Windows window controls (minimize, maximize, close) in the top right corner.

c. Listing Program Dengan Fungsi

```
function hitung(var p,l,t:integer):integer;  
begin  
hitung:=p*l*t;  
end;
```

```
procedure TForm1.BhitungClick(Sender: TObject);  
var  
pjl,lbr,tgi,luas:integer;  
begin  
pjl:=strtoint(epajang.Text);  
lbr:=strtoint(elebar.Text);  
tgi:=strtoint(etinggi.Text);  
luas:=hitung(pjl,lbr,tgi);  
eluas.Text:=inttostr(l);  
end;
```

```
procedure TForm1.BhapusClick(Sender: TObject);  
begin  
epanjang.Text :='0';  
elebar.Text :='0';  
etinggi.Text :='0';  
eluas.Text :='0';  
epanjang.SetFocus;  
end;
```

Didalam program utama tidak ditempatkan rumus untuk menghitung penjumlahan tetapi ketika dilakukan eksekusi terhadap program dan kita melakukan input Panjang, lebar dan tinggi dan kita klik Hitung maka akan ditampilkan hasil yang diinginkan. Hasil itu diperoleh dari pengiriman nilai dari teks ke nilai pada variabel deprogram utama (*pjl := strtoint(epanjang.Text); lbr := strtoint(elebar.Text); tgi := strtoint(etinggi.Text);*). Selanjutnya nilai pada variabel deprogram utama dikirim ke fungsi hitung untuk mendapat hasil penjumlahan (*luas:= hitung(pjl,lbr,tgi);*).

7.3. Fungsi Untuk Memanggil Fungsi yang Lain

Merupakan pembuatan program fungsi yang digunakan untuk memanggil program fungsi yang lain didalam satu listing program.

a. Hasil Yang di Inginkan

Rincian Anggota	
Nama Peminjam	Agung S
Jumlah Uang Pinjam	2000000
Tingkat Bunga (dalam %)	2
Lama Pinjam	12
Besarnya Uang yang Harus Dikembalikan	2.536.483,5891
Cicilan Uang Setiap Bulan	211.373,6324

Total Bersih

Close

b. Design Form



c. Desain Properties

Object	Caption/Text	Name
Label 1	Nama Peminjam	Label 1
Label 2	Jumlah Uang Pinjam	Label2
Label 3	Tingkat Bunga (dalarn %)	Label3
Label 4	Lama Pinjam	Label4
Label5	Besarnya Uang Yang Harus Dikembalikan	Label5
Label6	Cicilan Uang Setiap Bulan	Label6
GroupBox1	Rincian Nama Anggota	Groupbox1
Edit1	-	Epeminjam
Edit2	-	Epinjam
Edit3	-	Ebunga
Edit4	-	Elama
Edit5	-	Ekembalian
Edit6	-	Ecicil
Button 1	Total	Btotal
Button2	Bersih	B bersih
Button3	Close	bclose

d. Listing Program

> Fungsi Perhitungan Bunga

```
function pangkat(x,y : real):real; begin
    pangkat := exp(ln(x)*y); end;
function bunga(e,f ,g : real):real; begin
    bunga := e*pangkat((1+f/100),g); end;
```

> Program Total

```
procedure TForm15.bttotalClick(Sender: TObject); var
    a,b,c,d,e : real; begin
    a := strtolofloat(ejumlah.Text);
    b := strtolofloat(ebunga.Text);
    c := strtolofloat(elama.Text);
    d := bunga(a,b,c);
```

```
ekembalian.Text := floattostr(d);
ekembalian.Text := formatfloat('#.###(0',d); e := d/c;
ecicil.Text := floattostr(e);
ecicil.Text := formatfloat('#.###(0',e);
end;
```

> **Program Bersih**

```
procedure TForm15.bbbersihClick(Sender: TObject); begin
ejumlah.Text := "";
elama.Text := ""; epeminjam.Text := ""; ebunga.Text :=
"; ekembalian.Text := "";
ecicil.Text := "";
end;
```

> **Program Menutup Form**

```
procedure TForm15.bcloseClick(Sender: TObject);
begin
close;
end;
```

BAB VIII ARRAY

8.1. Pengertian Array

Array (larik) merupakan tipe data terstruktur dimana didalamnya terdiri dari komponen - komponen yang mempunyai tipe data yang sama. Didalam suatu array jumlah komponen banyaknya adalah tetap. Didalam suatu larik atau array setiap kompoenen ditunjukkan oleh suatu index yang unik. Index dari setiap komponen array menunjukkan urutan data atau identitas yang mewakili data yang ada didalamnya. Logika sederhananya array itu bisa disamakan dengan dua orang dengan nama yang sama didalam suatu komunitas, untuk membedakan antara nama yang satu atau dengan nama yang lain maka diberikan initial tambahan untuk setiap nama.

8.2. Deklarasi Array

Didalam penulisan bahasa pemograman setiap penggunaan array harus dideklarsikan terlebih dahulu. Pendeklarasian array diawali dengan nama variabel array diikuti dengan indeks array yang dituliskan didalam tanda "[]" , diikuti dengan kata cadangan of dan tipe data yang dibutuhkan.

Bentuk Umum Penulisan

Tanda_pengenal: array [..tipe index ..] of tipe data;

Contoh :

```
Var
A : array[1..4] of integer; B :
array[1..5] of string; C:
array[1..10] of real;
```

Keterangan :

A,B,C merupakan tanda pengenalan/ nama variabel dari array;
1.4 : merupakan tipe indek dari array, yang menunjukkan banyaknya data yang mampu disimpan.
Integer : menunjukkan bahwa data yang diinput berupa bilangan bulat.

8.3. Alokasi Penggunaan Array

a. Array Static (*Static Array*)

array static adalah model pendeklarasian array dimana tipe data yang digunakan mempunyai nilai yang tetap. Nilai yang digunakan untuk menentukan jangkauan pada umumnya bernilai integer. Array Static juga bisa disebut Array dengan deklarasi tipe indeks subrange integer.

Bentuk Umum

```
array[indexType1,indexType2] of baseType
```

Keterangan = index type menunjukkan tipe data ordinal yang menunjukkan batasan atau elemen maksimal terhadap seberapa besar variabel tersebut menyimpan komponen.

Contoh

```
Var arrayku : array[1..5] of char Atau
```

juga

type

```
jangkauan = 1..5;
var
```

nilai: array [jangkauan] of integer;

b. Array Dinamis (*Dynamic arrays*)

Larik atau array dinamis merupakan array yang tidak mempunyai suatu jangkauan atau ukuran yang tetap. Tetapi ketika program dijalankan maka memori untuk suatu array dinamis direalokasikan ketika kita menugaskan suatu nilai kepada array. Dynamic-Array jenis ditandai oleh konstruksi (menyangkut) format

Bentuk Umum

array of baseType

Contoh

var nilai: array of Real;

Dari deklarasi tersebut nilai yang merupakan deklarasi array belum memperoleh nilai yang tetap, tetapi hanya diberikan batasan sebagai tipe data real. Untuk mendeklarasikan array tersebut kita harus menempatkan array didalam suatu memori, caranya adalah dengan memanfaatkan fungsi dari perintah *sellength*.

Sellength(m\si,20)

Dari penggalan program tersebut nilai untuk array nilai tersebut mempunyai range sebanyak atau cakupan 20 untuk tipe data real, dengan indeex dimulai dari 0 sampai dengan 20.

8.4. Studi Kasus program dengan Array

a. Hasil ketika form Dijalankan



Gambar 8.1 Form Konversi Tanggal

b. Desain Properties

Object	Caption/Text	Name
Label 1	Tanggal	Label 1
Label 2	Bulan	Label2
Label 3	Tahun	Label3
Label 4	Hari Pada Tanggal tersebut Adalah	Label4
GroupBox1	Entry Tanggal	Groupbox1
Edit1	-	Etgl
Edit2	-	Ebln
Edit3	-	Etahun
Edit4	-	Ehari
Command 1	OK	bok

c. Listing Program

Program untuk Command OK

```

procedure TForm12.bhariClick(Sender: TObject); type
  x = string[7]; const
faktorbln : array[1..12] of byte = (0,3,3,6,1,4,6,2,5,0,3,5); hari : array[0..8]
of
x=('Minggu','Senin','Selasa','Rabu','Kamis','Jum','at','Sabtu'); var
  hr : string;
  nama : string[255];
  J1,j2,j3,j4 : integer;
  tanggal,bulan,tahun : integer; begin

```



```
tanggal := strtoint(etgl.Text);
bulan := strtoint(ebln.Text);
tahun := strtoint(ethn.Text);
if tahun > 1900 then tahun := tahun - 1900;
j1 := trunc(tahun * 365.25);
j2 := j1 + faktorblnfbulan];
if (tahun/4 = int(tahun/4)) and (bulan < 3) then j2 := j2-1;
j3 := j2 + tanggal;
j4 := trunc(frac(j3/7) * 10);
hr := Hari[j4];
ehari.Text := hr;
end;
```