



PANDUAN PEMROGRAMAN EMBARCADERO DELPHI

```
elif operation == "MIRROR_Y":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = True  
    mirror_mod.use_z = False  
elif operation == "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True  
  
#selection at the end - add back the deselected mirror modifier object  
mirror_ob.select = 1  
modifier_ob.select = 1  
bpy.context.scene.objects.active = modifier_ob  
print("Selected" + str(modifier_ob)) # modifier ob is the active ob  
    mirror_ob.select = 0  
    #name = bpy.context.selected_objects[0]  
    #obj_data = object_data_modifiers[0].name
```

HELDIANSYAH

**PANDUAN PEMROGRAMAN
EMBARCADERO DELPHI**

Undang-Undang No. 28 Tahun 2014 Tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Perlindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap :

- i. penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- iv. penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp 100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp 500.000.000,00 (lima ratus juta rupiah).
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak Rp 1.000.000.000,00 (satu miliar rupiah).
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp 4.000.000.000,00 (empat miliar rupiah).

PANDUAN PEMROGRAMAN EMBARCADERO DELPHI

Heldiansyah



Poliban Press

PANDUAN PEMROGRAMAN EMBARCADERO DELPHI

Penulis :
Heldiansyah

ISBN Elektronis :
978-623-7694-44-1

Editor dan Penyunting :
Adi Pratomo

Desain Sampul dan Tata letak :
Rahma Indera; Eko Sabar Prihatin

Penerbit :
POLIBAN PRESS
Anggota APPTI (Asosiasi Penerbit Perguruan Tinggi Indonesia)
no.004.098.1.06.2019
Cetakan Pertama, 2020

Hak cipta dilindungi undang-undang
Dilarang memperbanyak karya tulis ini dalam bentuk
dan dengan cara apapun tanpa ijin tertulis dari penerbit

Redaksi :
Politeknik Negeri Banjarmasin, Jl. Brigjen H. Hasan Basry,
Pangeran, Komp. Kampus ULM, Banjarmasin Utara
Telp : (0511)3305052
Email : press@poliban.ac.id

Diterbitkan pertama kali oleh :
Poliban Press, Banjarmasin, Oktober 2020

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan karunianya sehingga buku pemrograman Embarcadero Delphi tahun 2020 telah dapat diselesaikan. Buku ini merupakan pengantar bagi mahasiswa Diploma III Manajemen Informatika.

Terimakasih disampaikan kepada Joni Riadi S.ST., M.T. selaku Direktur Politeknik Negeri Banjarmasin dan Nurmahaludin, S.T., M.T. selaku Ketua Pusat Penelitian dan Pengabdian Masyarakat beserta sekretaris dan staf. Terimakasih juga disampaikan kepada Faris Ade Irawan, Reza Fauzan, Eko Sabar Prihatin dan Rahma Indera yang telah berkontribusi dalam editing serta seluruh tim Poliban Press dan semua pihak yang telah ikut membantu dalam penyelesaian buku ini.

Kami menyadari masih terdapat kekurangan dalam buku ini untuk itu kritik dan saran terhadap penyempurnaan buku ini sangat diharapkan. Semoga buku ini dapat memberi manfaat bagi semua pihak.

Banjarmasin, September 2020

Poliban Press

PRAKATA

Segala puji bagi Allah SWT atas Rahmat dan Karunia yang diberikan sehingga buku ajar panduan pemrograman Embarcadero Delphi dapat diselesaikan. Sholawat dan Salam kepada junjungan Nabi Besar Muhammad Shallahu Alaihi Wassalam yang menjadi teladan utama bagi kita semua.

Buku ajar ini merupakan materi bahan ajar mata kuliah Pemrograman Visual yang disampaikan kepada mahasiswa Program Studi Manajemen Informatika Jurusan Administrasi Bisnis Politeknik Negeri Banjarmasin. Buku ini menjadi harapan memberikan manfaat kepada mahasiswa agar mudah memahami dan menerapkan secara langsung materi-materi perkuliahan Pemrograman Visual.

Penyusun berterima kasih kepada semua pihak yang telah membantu dan mendukung dalam penyusunan buku ajar ini.

Banjarmasin, September 2020
Penyusun,

HELDIANSYAH, S.Kom,
M.Kom

DAFTAR ISI

KATA PENGANTAR.....	v
PRAKATA.....	vi
DAFTAR ISI.....	vii
BAB 1.....	1
IDE DELPHI	1
1.1 Bagian-Bagian IDE Delphi 2010	1
1.1.1 Main Menu	2
1.1.2 Project Manager	3
1.1.3 Form Designer.....	4
1.1.4 Form Structure.....	4
1.1.5 Code Editor.....	5
1.1.6 Object Inspector	6
1.1.7 Tool Pallette.....	7
1.2 Manajemen Berkas Project Program.....	8
1.2.1 Membuat Project Baru.....	9
1.2.2 Menyimpan Project	9
1.2.3 Meng-compile Project	11
1.3 Desain Antarmuka.....	11
1.4 Membuat Program Hello World.....	14
1.5 Membuat Program Pesan.....	16
1.6 Membuat Program Merubah Warna Form	18
1.7 Latihan.....	22
BAB 2.....	24
VARIABEL, TIPE DATA DAN OPERATOR.....	24
2.1 Variabel	24
2.2 Konstanta.....	25

2.3 Deklarasi Lokal	25
2.4 Deklarasi Global.....	25
2.5 Tipe Data	26
2.6 Konversi Tipe Data.....	27
2.7 Operator.....	27
2.7.1 Operator Penugasan.....	27
2.7.2 Operator Aritmatika.....	28
2.7.3 Operator Teks	28
2.7.4 Operator Boolean.....	28
2.7.5 Operator Perbandingan	29
2.8 Membuat Program Nama Lengkap.....	30
2.9 Membuat Program Kalkulator Sederhana	33
2.10 Latihan.....	37
BAB 3.....	39
PENCABANGAN.....	39
3.1 Argumen IF	39
3.2 Argumen CASE.....	42
3.3 Membuat Program Nilai Mahasiswa	42
3.4 Membuat Program Biaya Peserta Kursus Komputer.....	46
3.5 Membuat Program Biaya Durasi Kursus Komputer.....	49
3.6 Membuat Program Biaya Paket Kursus Komputer	52
3.7 Latihan.....	55
BAB 4.....	58
PERULANGAN.....	58
4.1 Argumen FOR	58
4.2 Argumen WHILE	59
4.2 Argumen REPEAT	59
4.3 Membuat Program Logika Perulangan (1).....	60
4.4 Membuat Program Bilangan Genap Ganjil	64

4.5 Latihan.....	69
BAB 5.....	70
PROCEDURE DAN FUNCTION	70
5.1 Procedure.....	70
5.2 Function.....	71
5.3 Membuat Program Hitung Kembalian Penjualan.....	72
5.4 Membuat Program Hitung Luas Bangun Segitiga.....	77
5.5 Latihan.....	81
GLOSARIUM	83
DAFTAR PUSTAKA.....	85
BIOGRAFI PENULIS.....	86

BAB 1

IDE DELPHI

Capaian Pembelajaran:

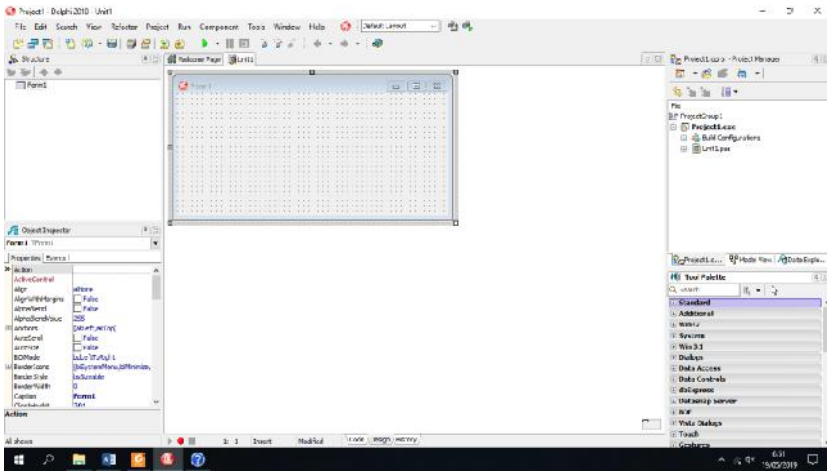
1. Mampu mengenali bagian-bagian IDE Delphi 2010
2. Mampu memahami kegunaan bagian-bagian IDE Delphi 2010
3. Mampu memahami manajemen berkas proyek Delphi 2010
4. Mampu membuat program sederhana

Delphi 2010 bahasa pemrograman pengembangan dari bahasa object pascal yang dikembangkan oleh perusahaan bernama Embarcadero dimana sebelumnya telah mengakuisisi perusahaan Borland yang merupakan perusahaan pengembang Delphi sejak versi awal. Delphi 2010 termasuk dalam pemrograman visual yang menyediakan lingkungan pengembangan terintegrasi dikenal dengan IDE (Integrated Development Environment). IDE Delphi ini disebut terintegrasi dimana dalam mengembangkan program aplikasi seperti mendesain antarmuka (form designing), menulis kode listing program (coding), mendeteksi error (debugging), membuat file exe (compiling) dan lain-lain dapat dilakukan semua disini. Selain itu IDE Delphi dilengkapi kemampuan intuitif dengan fitur IntelliSense yang dapat menampilkan pilihan property, method, procedure, function, dan variable milik suatu objek. (Embarcadero, 2009).

1.1 Bagian-Bagian IDE Delphi 2010

IDE Delphi 2010 terdiri dari beberapa bagian yang memiliki kegunaannya masing-masing diantaranya Main Menu, Project Manager, Form Designer, Form Structure, Code Editor, Object Inspector, dan Tool Pallete. Bagian-bagian IDE ini membantu memudahkan mendesain form secara visual, mengatur properti suatu objek, menulis kode listing program, dan manajemen proyek aplikasi

yang dikembangkan. Tampilan IDE Delphi 2010 dapat dilihat pada gambar 1.

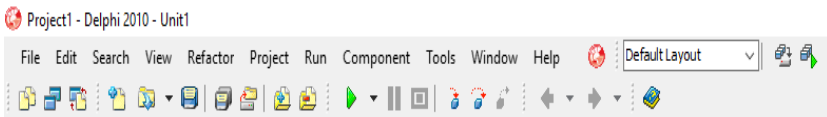


Gambar 1.1 IDE Delphi 2010

Bagian-bagian penting pada IDE Delphi 2010 akan dibahas sebagai berikut:

1.1.1 Main Menu

Main menu terdiri dari menubar dan toolbar yang berisi perintah-perintah kepada komputer untuk melakukan suatu tindakan seperti membuka, menutup, mengatur, dan menyimpan projek, meng-compile kode listing program dan lain sebagainya.



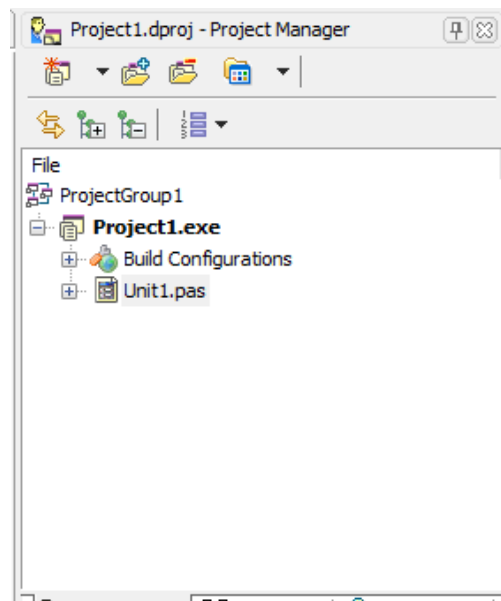
Gambar 1.2 Main Menu

Menubar terdiri dari horizontal menu bar yang bila di-klik akan menampilkan submenu-submenu. Toolbar berupa tombol-tombol dengan icon yang merupakan shortcut dari perintah-perintah yang ada

di menu bar. Toolbar yang ada merupakan jalan pintas untuk memudahkan menggunakan perintah-perintah yang sering digunakan.

1.1.2 Project Manager

Project Manager merupakan bagian untuk menampilkan dan mengatur struktur proyek program aplikasi yang dikembangkan. Pada bagian IDE ini akan tampil unit yang merupakan representasi dari form-form apa saja yang digunakan sebagai bagian dari proyek program aplikasi.

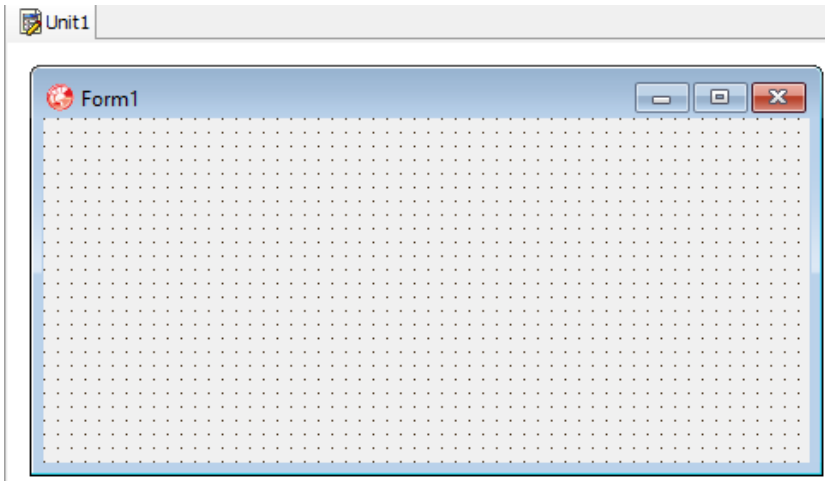


Gambar 1.3 Project Manager

Pada Project Manager ini juga menampilkan nama file exe yang akan dihasilkan dari proses kompilasi program yang berhasil tanpa pesan kesalahan. Nama file exe biasanya diambil nama proyek itu sendiri

1.1.3 Form Designer

Form Designer berfungsi mendesain form secara visual. Bagaimana mengatur tampilan Form dikerjakan pada bagian IDE ini. Suatu Form merupakan representasi dari jendela antarmuka yang akan berinteraksi langsung dengan pengguna program aplikasi.



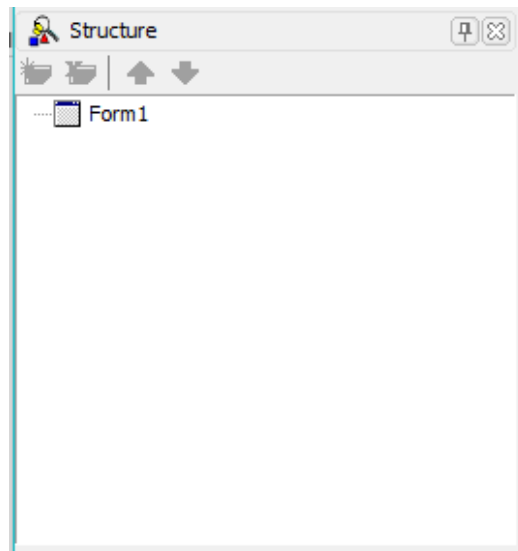
Gambar 1.4 Form Designer

Pada Form dapat diletakan komponen-komponen yang akan digunakan untuk keperluan antarmuka maupun fungsionalitas dari form itu sendiri. Form Designer sangat dimaksimalkan untuk komponen-komponen visual karena sangat mudah mengatur letak posisi dan ukuran dari komponen tersebut, bahkan untuk komponen yang dapat diletakan komponen (komponen didalam komponen) maka hal tersebut dapat sangat mudah dilakukan dengan Form Designer ini. Bagian IDE ini menjadi inti pemrograman visual pada IDE Delphi 2010.

1.1.4 Form Structure

Form Structure merupakan bagian IDE Delphi 2010 yang menampilkan struktur dari suatu Form yang sedang didesain. Form structure saling berhubungan dan sebagai pelengkap Form designer

dimana bila suatu form diletakan suatu komponen maka komponen-komponen itu akan muncul pula dalam Form structure secara berjenjang.



Gambar 1.5 Form Structure

Bila desain antarmuka suatu form menjadi kompleks dan rumit seperti penggunaan komponen dalam komponen maka akan lebih mudah menelusuri tingkatan letak kepemilikan dengan Form structure ini.

1.1.5 Code Editor

Code Editor merupakan bagian IDE yang tidak tampak secara langsung melainkan seakan-akan berada dibelakang Form. Code Editor berhubungan secara langsung dengan Form designer. Code Editor dapat ditampilkan dengan beberapa cara yaitu dengan menekan tombol F12, mengklik dua kali pada Form dan komponen, atau melalui Object Inspector.


```
Unit1
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Form
  Dialogs;
type
10  TForm1 = class(TForm)
      procedure FormCreate(Sender: TObject);
      private
        { Private declarations }
      public
        { Public declarations }
      end;
var
20  Form1: TForm1;
implementation
  {$R *.dfm}
27  procedure TForm1.FormCreate(Sender: TObject);
      begin
      end;
```

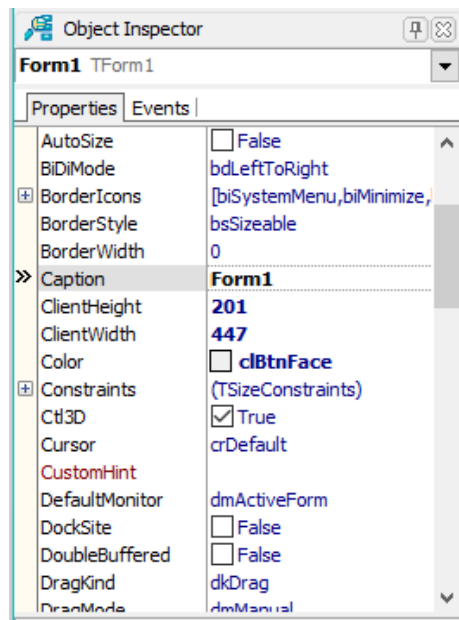
Gambar 1.6 Code Editor

Code Editor digunakan untuk menulis kode-kode listing program perintah yang diinginkan terhadap suatu objek sesuai dengan event yang ditentukan. Kode-kode listing program yang dapat ditulis antara lain membuat Procedure dan Function, mendeklarasikan variabel, mengambil dan memberi nilai suatu properti objek.

1.1.6 Object Inspector

Object Inspector merupakan bagian pengaturan semua komponen termasuk Form. Pengaturan yang bisa dilakukan pada bagian ini meliputi Properties dan Events. Properties berhubungan dengan hal-hal yang dimiliki suatu objek, misalkan objek Form memiliki properti Caption untuk mengisi judul Form, atau properti Name untuk memberi nama Form tersebut. Properti penting tiap-tiap objek akan berbeda satu sama lain dan itu dapat dilihat dari bagian IDE ini. Dari Object Inspector dapat dilakukan perubahan nilai suatu property dimana

perubahan tersebut dapat terlihat langsung atau harus di-compile terlebih dulu.



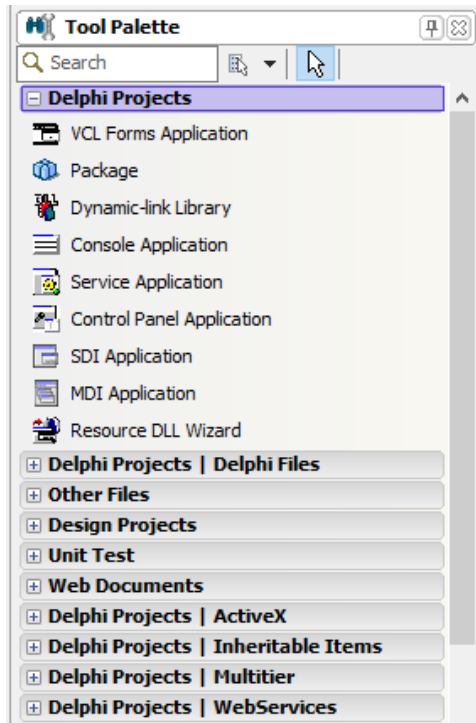
Gambar 1.7 Object Inspector

Selain Properties, setiap objek juga memiliki Events. Events ini merupakan kejadian kapan procedure yang akan aktif. Events pada Object Inspector terhubung langsung dengan Code Editor, misal pada objek Form Event onClick langsung mengaktifkan Code Editor menuliskan dan mengcompile kode-kode listing program. Setelah berhasil melalui semua error maka ketika Form di-klik maka kode-kode listing program pada procedure yang dibuat tadi akan dijalankan.

1.1.7 Tool Pallete

Tool Pallete adalah kumpulan komponen visual dan non-visual. Komponen visual mempunyai fungsi tertentu biasanya digunakan untuk merancang antarmuka, sedangkan komponen non-visual juga mempunyai fungsi-fungsi tertentu namun tidak mempunyai tampilan

antarmuka. Komponen-komponen pada tool pallette ini yang diletakan pada Form sesuai dengan fungsi yang dimilikinya.



Gambar 1.8 Tool Pallette

Komponen-komponen yang disediakan IDE Delphi 2010 sangat lengkap sehingga sangat membantu fokus memaksimalkan fungsi-fungsi yang sudah ada tanpa harus membuat kode-kode listing program dari awal.

1.2 Manajemen Berkas Project Program

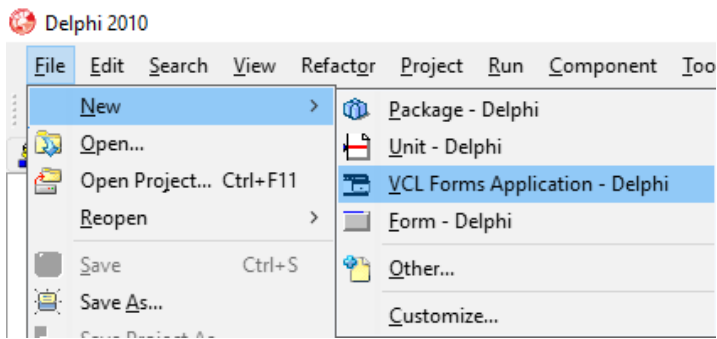
Object Pascal menyimpan kode listing program perintah-perintah yang ditulis dalam file berekstensi .pas. Delphi 2010 merupakan pengembangan

Object Pascal tetap mempertahankan file kode listing program ekstensi .pas, namun Delphi 2010 juga menyediakan Form dan komponen yang

memiliki banyak property beserta nilai property untuk disimpan kedalam project program aplikasi.

1.2.1 Membuat Project Baru


Project ini mewakili program aplikasi dimana dalam satu Project menyimpan listing program dan pengaturan untuk Project itu sendiri dan juga listing program dan pengaturan tiap-tiap Form yang terdapat dalam Project tersebut. Membuat project baru dengan memilih menu File >New > VCL Form Application-Delphi.

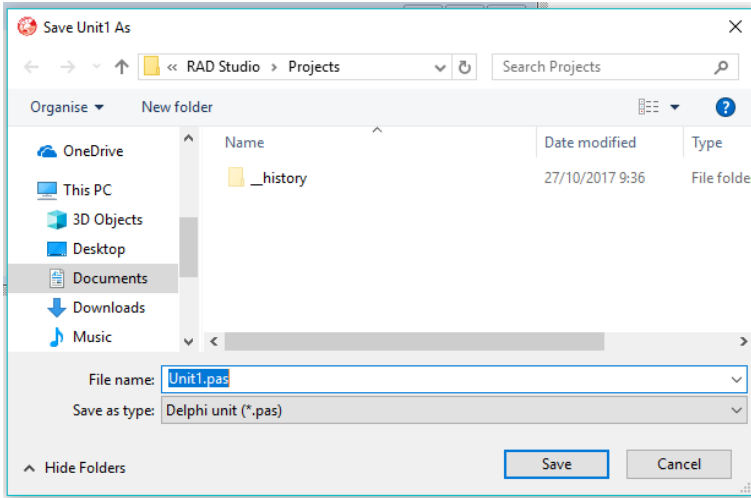


Gambar 1.9. Membuat Project baru

IDE Delphi akan membuat project baru dengan nama default Project1 dan sebuah unit baru dengan nama default Unit1. Didalam Unit1 ini sudah disediakan sebuah form kosong dengan nama default Form1 yang siap di modifikasi.

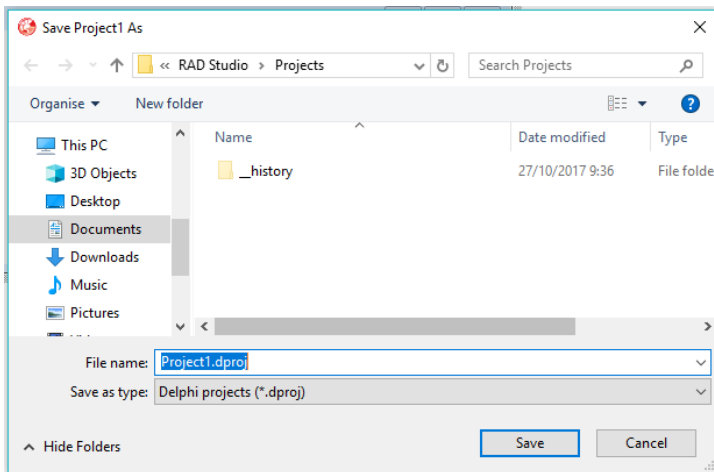
1.2.2 Menyimpan Project

Penyimpanan project dilakukan agar project yang telah dibuat, dapat dibuka kembali pada saat terakhir dimodifikasi tanpa harus mulai dari awal lagi. Penyimpanan project yang dilakukan dengan cara klik tombol Save All  dalam dua tahap. Tahap pertama menyimpan dan memberi nama Unit-Unit yang ada dalam project.



Gambar 1.10. Menyimpan dan Memberi Nama Unit Baru


Unit yang umum disimpan adalah Unit yang memiliki Form, sehingga dapat dikatakan tahap ini juga menyimpan Form-Form yang dibuat. Unit disimpan dalam format file ekstensi .pas, selain itu akan dibuat file-file pendukung file Unit dengan ekstensi .dfm dan .dcu. Pada tahap kedua baru menyimpan dan memberi nama project.



Gambar 1.11. Menyimpan dan Memberi Nama Project Baru

Project disimpan dalam format file ekstensi .dproj, selain itu akan dibuat file-file pendukung file project dengan ekstensi .dpr, .dproj.local, .res. Project yang sudah disimpan dapat dibuka kembali dengan memilih menu File > Open Project atau tekan tombol keyboard Ctrl+F11 lalu memilih File project pada lokasi penyimpanan. Project yang akan disalin atau dipindahkan ke komputer lain maka file-file pendukung Project tetap harus disertakan.

1.2.3 Meng-compile Project

Proses compile project dapat dilakukan dengan beberapa cara yaitu dengan memilih menu Run > Run atau mengklik tombol Run  atau tekan tombol keyboard F9. Bila proses compile berhasil tanpa mendapati error satupun maka ada satu file lagi yang akan dibuat otomatis oleh IDE Delphi 2010 yaitu file dengan ekstensi .exe, dimana ekstensi .exe ini merupakan berkas eksekusi dari program aplikasi yang dapat dijalankan di Sistem Operasi Windows secara mandiri terlepas ketergantungan dari IDE Delphi 2010.

1.3 Desain Antarmuka

Pada pemrograman visual menggunakan Delphi 2010 memiliki dasar yang umum dilakukan dalam mengembangkan program aplikasi dimulai dengan mendesain antarmuka. Desain antarmuka dilakukan dengan merancang form, tata letak objek komponen, merubah properties dan menulis kode-kode program.

Form merupakan object yang akan berinteraksi dengan pengguna program. Form memiliki banyak Property dan Event. Property merupakan atribut mendefinisikan suatu sifat tertentu yang menempel pada Form. Event merupakan kejadian kapan suatu kode-kode program yang ditulis akan dijalankan oleh Form tersebut. Property milik form yang populer dimodifikasi antara lain:

Properties	Kegunaan
Caption	Untuk merubah judul form
BorderIcons	Untuk menentukan tombol dasar apa yang tampil pada pojok kanan atas form.
BorderStyle	Untuk menentukan model Form bisa di resize atau tidak.
Color	Untuk menentukan warna latar
Cursor	Untuk menentukan bentuk icon kursor mouse selama berada di area form.
FormStyle	Untuk menentukan model posisi form ketika bekerja dengan form-form lain ataupun aplikasi windows lainnya.
Height	Untuk menentukan tinggi
Name	Untuk merubah nama form sebagai identitas akses
Position	Untuk menentukan posisi awal ketika form pertama kali dibuat dan dijalankan.
Visible	Untuk menyembunyikan form
Width	Untuk menentukan lebar
WindowState	Untuk menentukan kondisi luas tampilan form pertama kali dibuat dan dijalankan apakah sesuai saat desain, minimize, atau maximize.

Event milik form yang populer digunakan antara lain:

Event	Keterangan
OnCreate	Menjalankan kode program ketika form pertama kali dibuat pada memori komputer.
OnActivate	Menjalankan kode program ketika form dalam keadaan aktif, biasanya ditandai dengan warna form yang cerah.
OnClick	Menjalankan kode program ketika bagian latar form di-klik.
OnClose	Menjalankan kode program ketika form ditutup.

OnDbClick	Menjalankan kode program ketika bagian latar form di-klik ganda.
OnDeactivate	Menjalankan kode program ketika form dalam keadaan inaktif, biasanya ditandai dengan warna form yang pudar.
OnShow	Menjalankan kode program ketika form dipanggil dengan perintah Show.

Komponen standar yang populer digunakan antara lain:

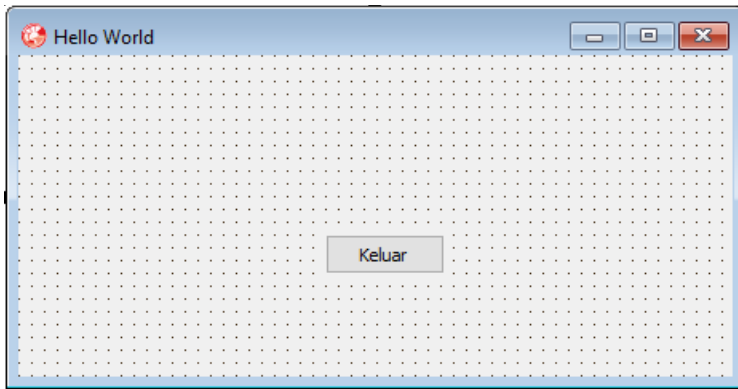
Komponen	Keterangan
TLabel	Komponen label untuk membuat tulisan statik.
TEdit	Komponen textbox untuk input dan output teks.
TButton	Komponen tombol untuk menjalankan perintah dengan cara klik.
TCheckBox	Komponen checkbox untuk memberikan pilihan lebih dari satu.
TRadioButton	Komponen tombol option untuk memberikan pilihan salah satu.
TListBox	Komponen list untuk menampilkan text yang memiliki baris-baris daftar.
TRadioGroup	Komponen pengelompokan tombol option untuk memberikan pilihan salah satu. Digunakan bila ada beberapa kelompok RadioButton untuk keperluan yang berbeda.
TPanel	Komponen bentuk bangun persegi yang dapat menampung komponen lain untuk keperluan tata letak lebih rapi. Dapat digunakan untuk membentuk tampilan garis lurus horizontal dan vertical.

Property dan event ini dimiliki oleh Form dan setiap objek termasuk komponen-komponen yang ada pada IDE Delphi dan dapat berbeda-beda setiap objek. Property dapat dilihat dan dimodifikasi pada Object Inspector atau melalui kode program yang melalui fitur code intellisense yaitu mengetik nama objek dan karakter titik.

1.4 Membuat Program Hello World

Program Hello World merupakan program sederhana untuk mengilustrasikan proses compile pada IDE Delphi 2010 dengan tingkat keberhasilan tinggi dan tingkat error yang rendah. Untuk membuat Project baru dengan memilih menu File >New > VCL Form Application-Delphi.

Tampilan Form yang masih kosong, tambahkan sebuah komponen TButton dari Tool Pallette pada page Standard. Atur tata letak seperti gambar 1.12.



Gambar 1.12. Desain Form Hello World

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Hello World
Button1	Caption	&Keluar
Button1	Name	BtKeluar

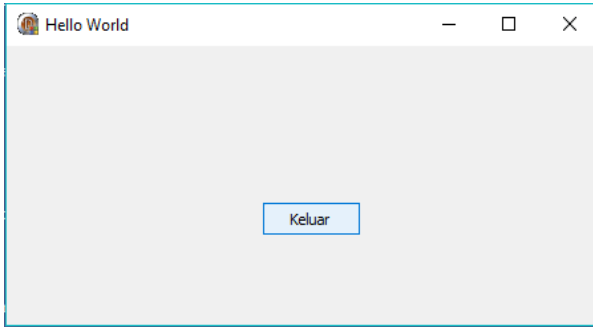
Listing program:

Object	Event	Code
BtKeluar	OnClick	procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;

Listing program lengkap:

```
unit Unit1;  
  
interface  
  
uses  
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
    Forms,  
    Dialogs, StdCtrls;  
  
type  
    TForm1 = class(TForm)  
        BtKeluar: TButton;  
        procedure BtKeluarClick(Sender: TObject);  
    private  
        { Private declarations }  
    public  
        { Public declarations }  
    end;  
  
var  
    Form1: TForm1;  
  
implementation  
  
    {$R *.dfm}  
  
    procedure TForm1.BtKeluarClick(Sender: TObject);  
    begin  
        Close;  
    end;  
  
end.
```

Run project Hello World maka akan menampilkan Form dengan judul Hello World pada bagian kiri atas dan sebuah tombol keluar yang bila di-klik akan menutup program.



Gambar 1.13 Run Form Hello World

Form Hello World ini mempunyai fitur dasar yang dimiliki Form yang disediakan IDE Delphi yaitu tombol minimize, maximize, dan close pada bagian kanan atas. Selain itu Form ini juga mempunyai fitur resize dan movable.

1.5 Membuat Program Pesan

Membuat program pesan menampilkan kotak pesan menggunakan method yang telah disediakan oleh IDE Delphi 2010. Buat Project baru dengan memilih menu File >New > VCL Form Application-Delphi. Atur tata letak komponen seperti gambar 1.14.



Gambar 1.14 Form Pesan Selamat Datang

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Selamat Datang
Button1	Caption	&KOTAK PESAN
Button1	Name	BtPesan
Button2	Caption	&KELUAR
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
BtPesan	OnClick	procedure TForm1. BtPesan Click(Sender: TObject); begin showmessage('Selamat Datang'); end;
BtKeluar	OnClick	procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;

Listing program lengkap:

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    BtKeluar: TButton;
```

```

Button1: TButton;
procedure Button1Click(Sender: TObject);
procedure BtKeluarClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  showmessage('Selamat Datang');
end;

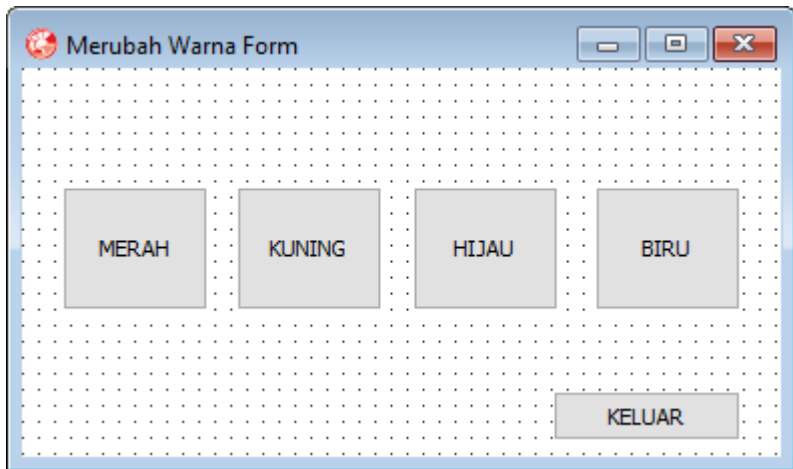
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;

end.

```

1.6 Membuat Program Merubah Warna Form

Membuat program untuk mengilustrasikan perubahan property color milik form pada saat Design-time yaitu saat belum di klik tombol Run menggunakan Object Inspector dan saat Run-time yaitu saat sesudah di klik tombol Run menggunakan kode program. Buat Project baru dengan memilih menu File >New > VCL Form Application-Delphi. Atur tata letak komponen seperti gambar 1.15.



Gambar 1.15 Form Merubah Warna

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Merubah Warna Form
Form1	Color	clWhite
Button1	Caption	&MERAH
Button1	Name	BtMerah
Button2	Caption	&KUNING
Button2	Name	BtKuning
Button3	Caption	&HIJAU
Button3	Name	BtHijau
Button4	Caption	&BIRU
Button4	Name	BtBiru
Button5	Caption	&KELUAR
Button5	Name	BtKeluar

Listing program:

Object	Event	Code
Form1	OnCreate	<pre> procedure TForm1.FormCreate(Sender: TObject); begin Color := clBlack; end; </pre>
BtMerah	OnClick	<pre> procedure TForm1.BtMerahClick(Sender: TObject); begin Color := clRed; end; </pre>
BtKuning	OnClick	<pre> procedure TForm1.BtKuningClick(Sender: TObject); begin Color := clYellow; end; </pre>
BtHijau	OnClick	<pre> procedure TForm1.BtHijauClick(Sender: TObject); begin Color := clGreen; end; </pre>
BtBiru	OnClick	<pre> procedure TForm1.BtBiruClick(Sender: TObject); begin Color := clBlue; end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    BtMerah: TButton;
    BtKeluar: TButton;
    BtKuning: TButton;
    BtHijau: TButton;
    BtBiru: TButton;
    procedure BtMerahClick(Sender: TObject);
    procedure BtKeluarClick(Sender: TObject);
    procedure BtKuningClick(Sender: TObject);
    procedure BtHijauClick(Sender: TObject);
    procedure BtBiruClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.FormCreate(Sender: TObject);
begin
  Color := clBlack;
end;
```



```
procedure TForm1.BtMerahClick(Sender: TObject);
begin
  Color := clRed;
end;

procedure TForm1.BtKuningClick(Sender: TObject);
begin
  Color := clYellow;
end;

procedure TForm1.BtHijauClick(Sender: TObject);
begin
  Color := clGreen;
end;

procedure TForm1.BtBiruClick(Sender: TObject);
begin
  Color := clBlue;
end;

procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;

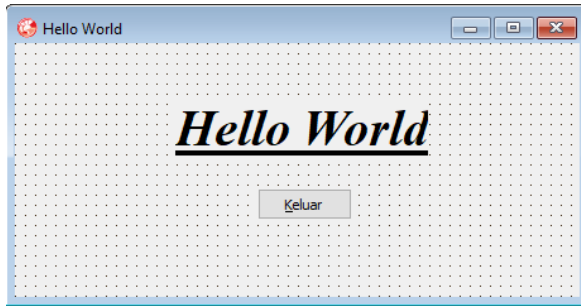
end.
```

1.7 Latihan

Latihan membuat program untuk mengilustrasikan cara kerja IDE Delphi 2010.

1. Latihan 1

Modifikasi program Hello World dengan menambahkan sebuah label dengan judul Hello World. Tentukan sendiri property yang harus diatur agar tampil seperti contoh pada gambar 1.16.



Gambar 1.16 Modifikasi Form Hello World

Ketentuan program:

Font yang digunakan pada label yaitu Times New Roman dengan ukuran 20. Style yang digunakan *italic* dan *underline*.

2. Latihan 2

Modifikasi program pesan dengan menambahkan sebuah input edit dan label yang diatur agar tampil seperti contoh pada gambar 1.17.



Gambar 1.17 Modifikasi Form Pesan

Ketentuan program:

Tombol “KOTAK PESAN” di-klik menampilkan “Selamat Datang” kepada nama yang diinputkan.

BAB 2

VARIABEL, TIPE DATA DAN OPERATOR

Capaian Pembelajaran:

1. Mampu memahami jenis-jenis variabel
2. Mampu mengetahui ruang lingkup variabel
3. Mampu mengetahui tipe data
4. Mampu menggunakan operator
5. Mampu membuat program penerapan variabel

Dalam menulis kode-kode perintah program diperlukan variabel. Variabel adalah pengenal penyimpanan nilai di memori komputer yang bersifat sementara. Variabel harus dideklarasikan agar dikenali oleh IDE Delphi sebagai bagian dari kode program. Variabel yang dideklarasikan harus memiliki nama sebagai pengenal. (Setiawan, 2004)

2.1 Variabel

Variabel menampung nilai yang dapat diubah-ubah sesuai dengan tipe data yang ditentukan. Deklarasi variabel pada IDE Delphi diawali kata kunci *var* diikuti nama variabel dan tipe datanya.

```
var <nama variabel> : <tipe data>;
```

contoh 1:

```
var hitung : integer;
```

contoh 2:

```
var hitung, hasil : integer;
```

contoh 3:

```
var hitung, hasil : integer;
```

```
nama, alamat : string;
```

2.2 Konstanta

Konstanta dapat dianggap sebagai variabel hanya saja menampung nilai yang tetap saat dideklarasikan dan tidak dapat diubah lagi. Deklarasi konstanta pada IDE Delphi diawali kata kunci *const* diikuti nama konstanta dan nilai yang ditetapkan.

```
const <nama konstanta> := <nilai konstansta>;
```

contoh 1:

```
const uang := 1000;
```

contoh 2:

```
const terbilang := 'seribu';
```

2.3 Deklarasi Lokal

Deklarasi lokal yaitu deklarasi suatu variabel atau konstanta dalam procedure atau function. Variabel atau konstanta yang dideklarasikan disebut sebagai variabel atau konstanta lokal, dimana hanya bisa dikenali dan diakses oleh procedure atau function tersebut. Deklarasi variabel atau konstanta lokal diletakan setelah nama procedure atau function sebelum kata kunci *begin*.

contoh 1:

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
const uang := 1000;
```

```
var Angka:integer;
```

```
huruf:string;
```

```
begin
```

```
...
```

```
end;
```

contoh 2:

```
function hitung:integer;
```

```
const uang := 1000;
```

```
var Angka:integer;
```

```
huruf:string;
```

```
begin
```

```
...
```

```
return ...
```

```
end;
```

2.4 Deklarasi Global

Deklarasi global yaitu deklarasi suatu variabel atau konstanta dalam dalam suatu unit program. Variabel atau konstanta yang dideklarasikan disebut sebagai variabel atau konstanta global, dimana bisa dikenali dan diakses oleh seluruh bagian dalam unit program. Deklarasi variabel atau kontanta global diletakan pada bagian *private*, *public*, dan *var* sebelum kata kunci *implementation*.

```
contoh:  
private  
  { Private declarations }  
  var_global1:string; // variabel global  
public  
  { Public declarations }  
  var_global2:string; // variabel global  
end;  
  
var  
  Form1: TForm1;  
  var_global3:string; // variabel global
```

Perbedaan mendasar yaitu variabel global pada bagian *private* dapat dikenali oleh seluruh bagian unit program itu sendiri, tetapi tidak dapat dikenali dan diakses oleh unit program lain. Sedangkan variabel global pada bagian *public* dan *var* dapat dikenali oleh seluruh bagian unit program itu sendiri, juga dapat dikenali dan diakses oleh unit program lain.

2.5 Tipe Data

Tipe data merupakan jenis data yang bisa ditampung. Tipe data yang disediakan IDE Delphi sangat banyak namun dapat disederhanakan menjadi beberapa tipe data yang paling sering dipakai dalam membuat kode-kode program. Tipe data populer untuk menampung teks yaitu tipe data string. Tipe data populer untuk menampung bilangan bulat yaitu tipe data integer. Tipe data populer untuk menampung bilangan pecahan yaitu tipe data real. Tipe data populer untuk menampung logika yaitu tipe data boolean.

Tipe	Tipe Data Populer	Nilai Tampung
Teks Karakter	string	“A-Z”, “a-z”, “0-9”
Bilangan bulat	integer	-2147483648 .. 2147483647
Bilangan pecahan	real	2.9 x 10 ⁻³⁹ .. 1.7 x 10 ³⁸
Logika	boolean	true, false

2.6 Konversi Tipe Data

Pada Delphi 2010 telah disediakan function-function yang berguna untuk konversi tipe data menjadi tipe data lainnya. Konversi tipe data diperlukan ketika suatu nilai dengan tipe data tertentu harus ditampung pada variabel atau property suatu yang object dengan tipe data berbeda. Beberapa function-function populer yang digunakan untuk konversi tipe data antara lain:

Function	Kegunaan
StrToInt	Konversikan tipe data string teks angka menjadi tipe data integer
IntToStr	Konversikan tipe data integer menjadi tipe data string
StrToFloat	Konversikan tipe data string berupa teks angka menjadi tipe data float
FloatToStr	Konversikan tipe data float menjadi tipe data string

2.7 Operator

Delphi 2010 mengenal banyak operator. Operator biasanya menangani suatu operasi pada operand. Operator yang disediakan oleh Delphi diantaranya:

2.7.1 Operator Penugasan

Operator penugasan merupakan operator untuk melakukan suatu proses pemberian nilai. Simbol operator penugasan pada Delphi 2010 menggunakan := (titik dua sama dengan). Contoh penggunaan operator penugasan proses pemberian nilai:

```
i := 5; // memberikan nilai numerik lima pada variabel i
j := 'Sukses'; // memberikan teks "sukses" pada variabel j
```

2.7.2 Operator Aritmatika

Operator aritmatika digunakan untuk operasi penghitungan. Jenis operator aritmatika populer yaitu operator binary. Operator aritmatika binary menangani suatu operasi aritmatika diantara dua operand bertipe bilangan bulat atau bilangan pecahan. Contoh X, Y sebagai operand.

Operator Aritmatika	Operasi	Contoh
*	Perkalian	X * Y
/	Pembagian	X / Y
+	penjumlahan	X + Y
-	pengurangan	X - Y

2.7.3 Operator Teks

Operator teks digunakan untuk operasi penggabungan teks (concatenate). Simbol operator teks menggunakan + dimana operand harus bertipe data string. Contoh penggunaan operator teks:

```
buah := 'Jambu' + 'Air'; // variabel buah bernilai teks "JambuAir"
```

2.7.4 Operator Boolean

Operator boolean digunakan untuk operasi logika boolean dengan operand bertipe data boolean dan hasil operasi juga bertipe data boolean. Simbol operator boolean populer antara lain AND, OR, dan NOT. Berikut contoh gerbang logika boolean:

OPERAND	OPERASI
---------	---------

X	Y	X AND Y	X OR Y	NOT X	NOT Y	NOT X AND Y	NOT X OR Y
TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE

2.7.5 Operator Perbandingan

Operator perbandingan merupakan operator untuk membandingkan dua buah nilai yang menghasilkan nilai logika boolean TRUE atau FALSE. Operator perbandingan pada Delphi 2010 antara lain:

Operator Perbandingan	Operasi
=	Sama dengan
>	Lebih dari
<	Kurang dari
>=	Lebih dari sama dengan
<=	Kurang dari sama dengan
<>	Tidak sama dengan

Contoh operasi menggunakan operator perbandingan:

Contoh Operasi	Hasil
5 = 5	TRUE
5 = 4	FALSE
5 > 4	TRUE
4 > 5	FALSE
5 < 4	FALSE
4 < 5	TRUE
5 >= 4	TRUE
4 >= 5	FALSE
5 <= 4	FALSE

4 <= 5	TRUE
5 <> 4	TRUE
5 <> 5	FALSE

2.8 Membuat Program Nama Lengkap

Membuat program dengan hasil nama lengkap berdasarkan input nama depan dan nama belakang. Buat Project baru dengan memilih menu File >New > VCL Form Application-Delphi. Atur tata letak komponen seperti gambar 2.1



Gambar 2.1 Form Nama Lengkap

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Nama Lengkap
Label1	Caption	Nama Depan
Label2	Caption	Nama Tengah
Label3	Caption	Nama Belakang
Label4	Caption	Nama Lengkap
Edit1	Caption	(dikosongkan)
Edit1	Name	EdNamaDepan
Edit2	Caption	(dikosongkan)

Edit2	Name	EdNamaBelakang
Edit3	Caption	(dikosongkan)
Edit3	Name	EdNamaLengkap
Edit3	ReadOnly	True
Edit3	Color	clInfoBk
Button1	Caption	&NAMA LENGKAP
Button1	Name	BtNamaLengkap
Button2	Caption	&Keluar
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
BtNamaLengkap	OnClick	<pre> procedure TForm1.BtNamaLengkapClick(Sender: TObject); var nmdepan, nmbelakang, nmlengkap : string; begin nmdepan := EdNamaDepan.Text; nmbelakang := EdNamaBelakang.Text; nmlengkap := nmdepan + ' ' + nmbelakang; EdNamaLengkap.Text := nmlengkap; end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

unit Unit1;

```

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    Label2: TLabel;
    EdNamaDepan: TEdit;
    BtNamaLengkap: TButton;
    BtKeluar: TButton;
    EdNamaBelakang: TEdit;
    Label3: TLabel;
    EdNamaLengkap: TEdit;
    Label4: TLabel;
    procedure BtNamaLengkapClick(Sender: TObject);
    procedure BtKeluarClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtNamaLengkapClick(Sender: TObject);
var nmdepan, nmbelakang, nmlengkap : string;
begin
  nmdepan := EdNamaDepan.Text;
  nmbelakang := EdNamaBelakang.Text;

  nmlengkap := nmdepan + ' ' + nmbelakang;
  EdNamaLengkap.Text := nmlengkap;
end;

```

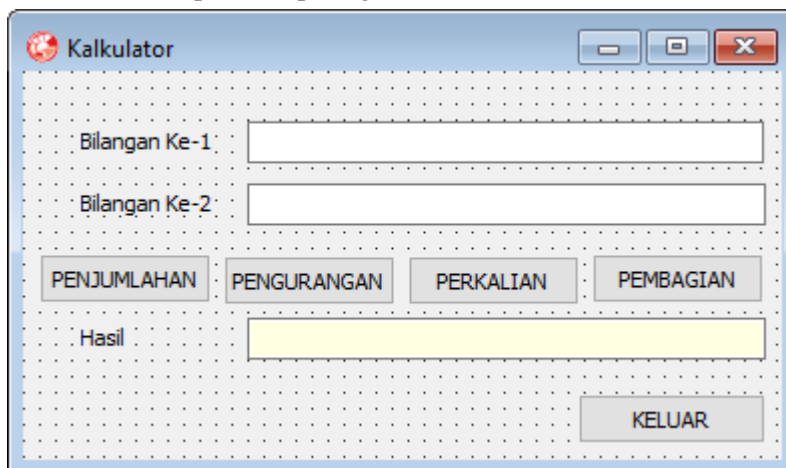
```

procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;
end.

```

2.9 Membuat Program Kalkulator Sederhana

Membuat program menghitung hasil penjumlahan, pengurangan, perkalian, dan pembagian dua buah bilangan bulat. Buat Project baru dengan memilih menu File >New > VCL Form Application-Delphi. Atur tata letak komponen seperti gambar 2.2



.Gambar 2.2 Form Kalkulator

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Kalkulator
Label1	Caption	Bilangan Ke-1

Label2	Caption	Bilangan Ke-2
Label3	Caption	Hasil
Edit1	Caption	0
Edit1	Name	EdBilangan1
Edit2	Caption	0
Edit2	Name	EdBilangan2
Edit3	Caption	0
Edit3	Name	EdHasil
Edit3	ReadOnly	True
Edit3	Color	clInfoBk
Button1	Caption	PEN&JUMLAHAN
Button1	Name	BtJumlah
Button2	Caption	PENG&URANGAN
Button2	Name	BtKurang
Button3	Caption	PER&KALIAN
Button3	Name	BtKali
Button4	Caption	PEM&BAGIAN
Button4	Name	BtBagi
Button5	Caption	&Keluar
Button5	Name	BtKeluar

Listing program:

Object	Event	Code
BtJumlah	OnClick	<pre> procedure TForm1.BtJumlahClick(Sender: TObject); var bilangan1, bilangan2 : integer; begin bilangan1 := StrToInt(EdBilangan1.Text); bilangan2 := StrToInt(EdBilangan2.Text); EdHasil.Text := IntToStr(bilangan1 + bilangan2); end; </pre>

BtKurang	OnClick	<pre> procedure TForm1.BtKurangClick(Sender: TObject); var bilangan1, bilangan2 : integer; begin bilangan1 := StrToInt(EdBilangan1.Text); bilangan2 := StrToInt(EdBilangan2.Text); EdHasil.Text := IntToStr(bilangan1 - bilangan2); end; </pre>
BtKali	OnClick	<pre> procedure TForm1.BtKaliClick(Sender: TObject); var bilangan1, bilangan2 : integer; begin bilangan1 := StrToInt(EdBilangan1.Text); bilangan2 := StrToInt(EdBilangan2.Text); EdHasil.Text := IntToStr(bilangan1 * bilangan2); end; </pre>
BtBagi	OnClick	<pre> procedure TForm1.BtBagiClick(Sender: TObject); var bilangan1, bilangan2 : integer; begin bilangan1 := StrToInt(EdBilangan1.Text); bilangan2 := StrToInt(EdBilangan2.Text); EdHasil.Text := FloatToStr(bilangan1 / bilangan2); end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    Label2: TLabel;
    EdBilangan1: TEdit;
    BtJumlah: TButton;
    BtKeluar: TButton;
    EdBilangan2: TEdit;
    Label3: TLabel;
    EdHasil: TEdit;
    Label4: TLabel;
    BtKurang: TButton;
    BtKali: TButton;
    BtBagi: TButton;
    procedure BtJumlahClick(Sender: TObject);
    procedure BtKeluarClick(Sender: TObject);
    procedure BtKurangClick(Sender: TObject);
    procedure BtKaliClick(Sender: TObject);
    procedure BtBagiClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtJumlahClick(Sender: TObject);
var bilangan1, bilangan2 : integer;
begin
  bilangan1 := StrToInt(EdBilangan1.Text);

```

```

    bilangan2 := StrToInt(EdBilangan2.Text);

    EdHasil.Text := IntToStr(bilangan1 + bilangan2);
end;

procedure TForm1.BtKurangClick(Sender: TObject);
var bilangan1, bilangan2 : integer;
begin
    bilangan1 := StrToInt(EdBilangan1.Text);
    bilangan2 := StrToInt(EdBilangan2.Text);

    EdHasil.Text := IntToStr(bilangan1 - bilangan2);
end;

procedure TForm1.BtKaliClick(Sender: TObject);
var bilangan1, bilangan2 : integer;
begin
    bilangan1 := StrToInt(EdBilangan1.Text);
    bilangan2 := StrToInt(EdBilangan2.Text);

    EdHasil.Text := IntToStr(bilangan1 * bilangan2);
end;

procedure TForm1.BtBagiClick(Sender: TObject);
var bilangan1, bilangan2 : integer;
begin
    bilangan1 := StrToInt(EdBilangan1.Text);
    bilangan2 := StrToInt(EdBilangan2.Text);

    EdHasil.Text := FloatToStr(bilangan1 / bilangan2);
end;

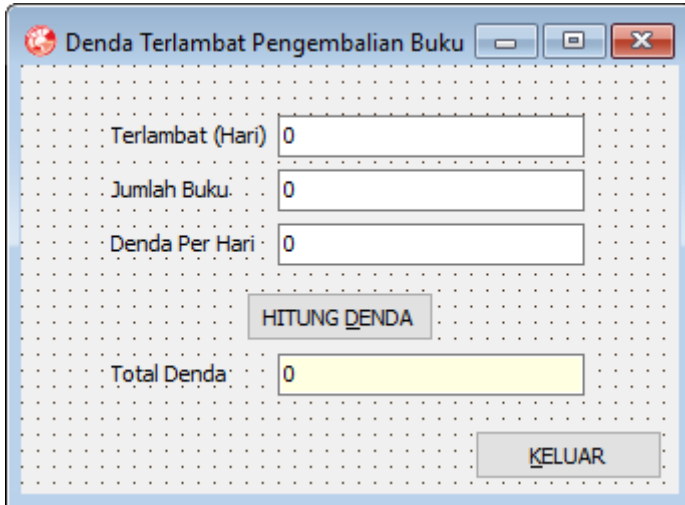
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
    Close;
end;

end.

```

2.10 Latihan

Buatlah project baru untuk menghitung denda keterlambatan pengembalian buku perpustakaan dengan tata letak Form seperti gambar 2.3.



The image shows a Windows application window titled "Denda Terlambat Pengembalian Buku". The window contains a form with the following elements:

- Input field for "Terlambat (Hari)" with the value "0".
- Input field for "Jumlah Buku" with the value "0".
- Input field for "Denda Per Hari" with the value "0".
- A button labeled "HITUNG DENDA" located below the input fields.
- An output field for "Total Denda" with the value "0", which is highlighted in yellow.
- A button labeled "KELUAR" located at the bottom right of the form.

Gambar 2.3 Form Hitung Denda Buku

Ketentuan program:

1. Terlambat (Hari): input dengan angka jumlah hari keterlambatan.
2. Jumlah Buku: input dengan angka jumlah buku yang terlambat dikembalikan
3. Denda Per Hari: input dengan angka biaya denda untuk setiap hari keterlambatan.
4. Hitung Denda: Proses menghitung total denda yang harus dibayar.

BAB 3

PENCABANGAN

Capaian Pembelajaran:

1. Mampu memahami logika pencabangan
2. Mampu menggunakan argumen logika pencabangan IF
3. Mampu menggunakan argumen logika pencabangan CASE
4. Mampu membuat program penerapan logika pencabangan

Pencabangan merupakan logika pemrograman untuk menjalankan suatu kode program hanya bila suatu kondisi yang ditentukan telah terpenuhi. Kondisi yang dimaksud memiliki dua nilai boolean yaitu TRUE atau FALSE. Pada Delphi 2010 terdapat dua macam argumen perintah yang dapat digunakan sebagai penerapan logika pencabangan yaitu argumen IF dan CASE.(Wahana, 2011)

3.1 Argumen IF

Argumen IF memiliki beberapa model penulisan, model pencabangan IF sederhana jika kondisi terpenuhi bernilai TRUE maka argumen akan dijalankan.

```
if <kondisi> then <argumen>;  
contoh 1:  
if 5 > 2 then showmessage('Benar'); //tampil jendela pesan tulisan  
"Benar"  
contoh 2:  
if 5 < 2 then showmessage('Benar'); //tidak tampil jendela pesan  
apapun.
```

Model pencabangan IF sederhana jika kondisi terpenuhi bernilai TRUE maka lebih dari satu argumen akan dijalankan.

```
if <kondisi> then  
begin  
  <argumen1>;  
  <argumen2>;  
end
```

```

    <argumen3>;
end;

contoh 1:
if 5 > 2 then
begin
    showmessage('Benar'); //tampil jendela pesan tulisan "Benar"
    showmessage('Lebih'); //tampil jendela pesan tulisan "Lebih"
    showmessage('Besar'); //tampil jendela pesan tulisan "Besar"
end;

contoh 2:
if 5 < 2 then
begin
    showmessage('Benar'); //tidak tampil jendela pesan apapun.
    showmessage('Lebih'); //tidak tampil jendela pesan apapun.
    showmessage('Besar'); //tidak tampil jendela pesan apapun.
end;

```

Model pencabangan IF jika kondisi terpenuhi bernilai TRUE maka argumen akan dijalankan dan bila kondisi bernilai FALSE maka akan menjalankan argumen yang lain.

```

if <kondisi> then
begin
    <argumen1>;
    <argumen2>;
end
else
begin
    <argumen3>
    <argumen4>
end

contoh 1:
if 5 > 2 then
begin
    showmessage('Benar'); //tampil jendela pesan tulisan "Benar"
    showmessage('Lebih'); //tampil jendela pesan tulisan "Lebih"
    showmessage('Besar'); //tampil jendela pesan tulisan "Besar"
end
else

```

```

showmessage('Salah'); //tidak tampil jendela pesan apapun
showmessage('Lebih'); //tidak tampil jendela pesan apapun
showmessage('Kecil'); //tidak tampil jendela pesan apapun
end

```

contoh 2:

```
if 5 < 2 then
```

```
begin
```

```
    showmessage('Benar'); //tidak tampil jendela pesan apapun
```

```
    showmessage('Lebih'); //tidak tampil jendela pesan apapun
```

```
    showmessage('Besar'); //tidak tampil jendela pesan apapun
```

```
end
```

```
else
```

```
    showmessage('Salah') //tampil jendela pesan tulisan "Salah"
```

```
    showmessage('Lebih') //tampil jendela pesan tulisan "Lebih"
```

```
    showmessage('Kecil') //tampil jendela pesan tulisan "Kecil"
```

```
end
```

Model pencabangan IF bersarang merupakan argumen IF didalam argumen IF lainnya.

```
if <kondisi> then
```

```
    if <kondisi> then
```

```
        <argumen1>
```

```
    else
```

```
        <argumen2>
```

```
else
```

```
    <argumen3>
```

```
end
```

contoh 1:

```
if 5 > 2 then
```

```
    if 4 > 2 then
```

```
        showmessage('Empat lebih dari dua') // tampil pesan
```

```
    else
```

```
        showmessage('Empat tidak lebih dari dua') // tidak tampil
```

```
pesan
```

```
else
```

```
    showmessage('Lima tidak lebih dari dua') // tidak tampil pesan
```

contoh 1:

```
if 5 < 2 then
```

```

if 4 < 2 then
    showmessage('Empat lebih dari dua') // tidak tampil pesan
else
    showmessage('Empat tidak lebih dari dua') // tidak tampil
pesan
else
    showmessage('Lima tidak lebih dari dua') // tampil pesan

```

3.2 Argumen CASE

Argumen CASE merupakan logika pencabangan yang menyederhanakan argumen IF bersarang. Argumen CASE lebih efisien dan mudah dalam mengelola logika IF bersarang.

```

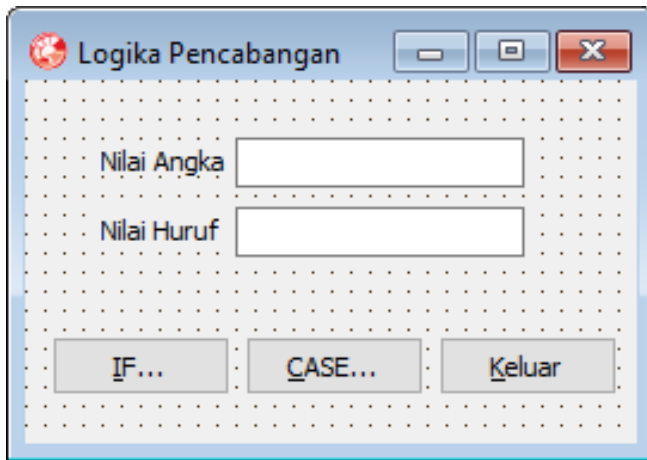
case <parameter>of
    <nilai parameter1> : <argumen1>;
    <nilai parameter2> : <argumen2>;
    <nilai parameter3> : <argumen3>;
else <argumen4>;
end;

contoh1:
index:=1;
case index of
    0 : ShowMessage('Pilihan pertama'); //tidak tampil pesan
    1 : ShowMessage('Pilihan kedua'); // tampil pesan
    2 : ShowMessage('Pilihan ketiga'); //tidak tampil pesan
else ShowMessage('Pilihan lainnya'); //tidak tampil pesan
end;

```

3.3 Menbuat Program Nilai Mahasiswa

Membuat program untuk mengilustrasikan cara kerja logika pencabangan menggunakan argumen IF dan CASE. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 3.1.



Gambar 3.1 Form Nilai Mahasiswa

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Logika Pencabangan
Label1	Caption	Nilai Angka
Label2	Caption	Nilai Huruf
Edit1	Text	(dikosongkan)
Edit1	Name	EdAngka
Edit2	Text	(dikosongkan)
Edit2	Name	EdHuruf
Button1	Caption	&IF...
Button1	Name	BtIF
Button2	Caption	&CASE...
Button2	Name	BtCase
Button3	Caption	&Keluar
Button3	Name	BtKeluar

Listing program:

Object	Event	Code
BtIF	OnClick	<pre> procedure TForm1. BtIFClick(Sender: TObject); var nAngka:integer; begin nAngka := StrToInt(EdAngka.Text); if nAngka >= 90 then EdHuruf.Text := 'A' else if nAngka >= 80 then EdHuruf.Text := 'B' else if nAngka >= 70 then EdHuruf.Text := 'C' else if nAngka >= 60 then EdHuruf.Text := 'D' else EdHuruf.Text := 'E' end; end; end; end; </pre>
BtCase	OnClick	<pre> procedure TForm1.BtCaseClick(Sender: TObject); var nAngka:integer; begin nAngka := StrToInt(EdAngka.Text); case nAngka of 90..100 : EdHuruf.Text := 'A'; 80..89 : EdHuruf.Text := 'B'; 70..79 : EdHuruf.Text := 'C'; 60..69 : EdHuruf.Text := 'D'; else EdHuruf.Text := 'E'; end; end; end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; end; </pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    BtKeluar: TButton;
    Label1: TLabel;
    EdAngka: TEdit;
    BtIF: TButton;
    Label2: TLabel;
    EdHuruf: TEdit;
    BtCase: TButton;
    procedure BtKeluarClick(Sender: TObject);
    procedure BtIFClick(Sender: TObject);
    procedure BtCaseClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtIFClick(Sender: TObject);
var nAngka:integer;
begin
  nAngka := StrToInt(EdAngka.Text);
  case nAngka of
    90..100 : EdHuruf.Text := 'A';
    80..89 : EdHuruf.Text := 'B';
    70..79 : EdHuruf.Text := 'C';
  end;
end;

```



```

    60..69 : EdHuruf.Text := 'D';
    else
    EdHuruf.Text := 'E';
    end;
end;

procedure TForm1.BtCaseClick(Sender: TObject);
var nAngka:integer;
begin
    nAngka := StrToInt(EdAngka.Text);
    case nAngka of
    90..100 : EdHuruf.Text := 'A';
    80..89 : EdHuruf.Text := 'B';
    70..79 : EdHuruf.Text := 'C';
    60..69 : EdHuruf.Text := 'D';
    else
    EdHuruf.Text := 'E';
    end;
end;

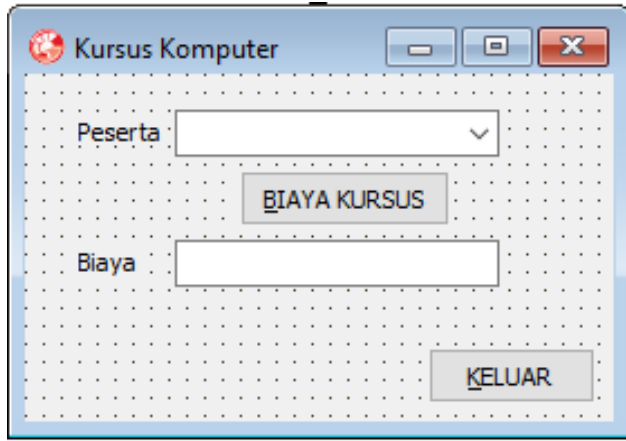
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
    Close;
end;

end.

```

3.4 Membuat Program Biaya Peserta Kursus Komputer

Membuat program untuk mengilustrasikan cara kerja logika percabangan. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 3.2.



Gambar 3.2 Form Biaya Peserta Kursus Komputer

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Kursus Komputer
Label1	Caption	Peserta
Label2	Caption	Biaya
ComboBox1	Text	(dikosongkan)
ComboBox1	Items...	PELAJAR MAHASISWA UMUM
ComboBox1	Name	CbPeserta
Edit1	Text	(dikosongkan)
Edit1	Name	EdBiaya
Button1	Caption	&BIAYA KURSUS
Button1	Name	BtBiaya
Button2	Caption	&KELUAR
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
BtBiaya	OnClick	<pre> procedure TForm1.BtBiayaClick(Sender: TObject); var IndeksPeserta : integer; begin IndeksPeserta := CbPeserta.ItemIndex; if IndeksPeserta = 0 then EdBiaya.Text := '100000'; if IndeksPeserta = 1 then EdBiaya.Text := '200000'; if IndeksPeserta = 2 then EdBiaya.Text := '300000'; end;</pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;</pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    CbPeserta: TComboBox;
    BtKeluar: TButton;
    BtBiaya: TButton;
    EdBiaya: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    procedure BtKeluarClick(Sender: TObject);
    procedure BtBiayaClick(Sender: TObject);
  private
```

```

    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtBiayaClick(Sender: TObject);
var IndeksPeserta : integer;
begin
    IndeksPeserta := CbPeserta.ItemIndex;
    if IndeksPeserta = 0 then
        EdBiaya.Text := '100000';
    if IndeksPeserta = 1 then
        EdBiaya.Text := '200000';
    if IndeksPeserta = 2 then
        EdBiaya.Text := '300000';
end;

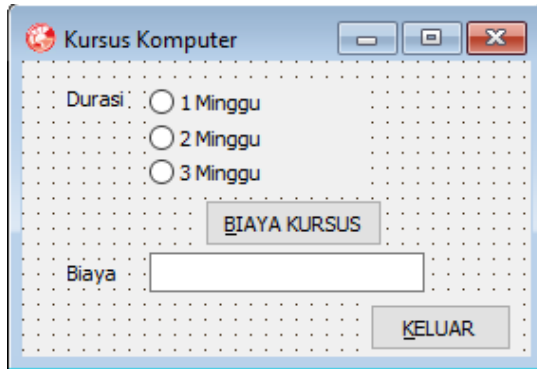
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
    Close;
end;

end.

```

3.5 Membuat Program Biaya Durasi Kursus Komputer

Membuat program untuk mengilustrasikan cara kerja logika percabangan. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 3.3.



Gambar 3.3 Form Biaya Durasi Kursus Komputer

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Kursus Komputer
Label2	Caption	Biaya
RadioGroup1	Caption	Durasi
RadioGroup1	Name	RgDurasi
RadioGroup1	Items...	1 Minggu 2 Minggu 3 Minggu
Edit1	Text	(dikosongkan)
Edit1	Name	EdBiaya
Button1	Caption	&BIAYA KURSUS
Button1	Name	BtBiaya
Button2	Caption	&KELUAR
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
BtBiaya	OnClick	<pre> procedure TForm1.BtBiayaClick(Sender: TObject); begin if RgDurasi.ItemIndex = 0 then EdBiaya.Text := '300000'; </pre>

		<pre> if RgDurasi.ItemIndex = 1 then EdBiaya.Text := '400000'; if RgDurasi.ItemIndex = 2 then EdBiaya.Text := '500000'; end;</pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;</pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    BtKeluar: TButton;
    BtBiaya: TButton;
    EdBiaya: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Rb1Minggu: TRadioButton;
    Rb2Minggu: TRadioButton;
    Rb3Minggu: TRadioButton;
    procedure BtBiayaClick(Sender: TObject);
    procedure BtKeluarClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
```

```

Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtBiayaClick(Sender: TObject);
begin
  if RgDurasi.ItemIndex = 0 then
    EdBiaya.Text := '300000';
  if RgDurasi.ItemIndex = 1 then
    EdBiaya.Text := '400000';
  if RgDurasi.ItemIndex = 2 then
    EdBiaya.Text := '500000';
end;

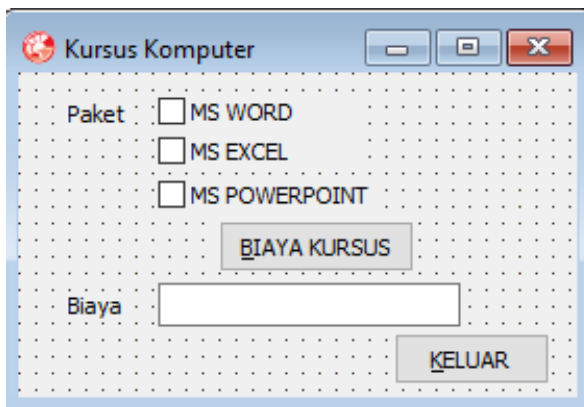
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;

end.

```

3.6 Membuat Program Biaya Paket Kursus Komputer

Membuat program untuk mengilustrasikan cara kerja logika percabangan. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 3.4.



Gambar 3.4 Form Biaya Paket Kursus Komputer

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Kursus Komputer
Label1	Caption	Paket
Label2	Caption	Biaya
Checkbox1	Text	MS WORD
Checkbox1	Name	CkWord
Checkbox2	Text	MS EXCEL
Checkbox2	Name	CkExcel
Checkbox3	Text	MS POWERPOINT
Checkbox3	Name	CkPowerPoint
Edit1	Text	(dikosongkan)
Edit1	Name	EdBiaya
Button1	Caption	&BIAYA KURSUS
Button1	Name	BtBiaya
Button2	Caption	&KELUAR
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
BtBiaya	OnClick	<pre> procedure TForm1.BtBiayaClick(Sender: TObject); var Biaya : real; begin Biaya := 0; if CkWord.Checked then Biaya := Biaya + 500000; if CkExcel.Checked then Biaya := Biaya + 600000; if CkPowerPoint.Checked then </pre>

		<pre> Biaya := Biaya + 700000; EdBiaya.Text := FloatToStr(Biaya); end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics,
    Controls, Forms,
    Dialogs, StdCtrls;

type
    TForm1 = class(TForm)
        BtKeluar: TButton;
        BtBiaya: TButton;
        EdBiaya: TEdit;
        Label1: TLabel;
        CkWord: TCheckBox;
        CkExcel: TCheckBox;
        CkPowerPoint: TCheckBox;
        Label2: TLabel;
        procedure BtBiayaClick(Sender: TObject);
        procedure BtKeluarClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form1: TForm1;

```

```
implementation
```

```
{ $R *.dfm }
```

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

```
var Biaya : real;
```

```
begin
```

```
  Biaya := 0;
```

```
  if CkWord.Checked then
```

```
    Biaya := Biaya + 500000;
```

```
  if CkExcel.Checked then
```

```
    Biaya := Biaya + 600000;
```

```
  if CkPowerPoint.Checked then
```

```
    Biaya := Biaya + 700000;
```

```
  EdBiaya.Text := FloatToStr(Biaya);
```

```
end;
```

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

```
begin
```

```
  Close;
```

```
end;
```

```
end.
```

3.7 Latihan

Buatlah Project baru untuk menghitung biaya kursus komputer dengan tata letak Form seperti gambar 3.5.

Gambar 3.5 Form Latihan Biaya Kursus Komputer

Ketentuan program:

5. Peserta merupakan pilihan yang terdiri dari PELAJAR, MAHASISWA, dan UMUM.
6. Biaya kursus komputer:

Peserta	Durasi	Paket	Biaya
PELAJAR	1 Minggu	MS WORD	100.000,-
PELAJAR	2 Minggu	MS WORD	200.000,-
PELAJAR	3 Minggu	MS WORD	300.000,-
PELAJAR	1 Minggu	MS EXCEL	100.000,-
PELAJAR	2 Minggu	MS EXCEL	200.000,-
PELAJAR	3 Minggu	MS EXCEL	300.000,-
PELAJAR	1 Minggu	MS POWERPOINT	100.000,-
PELAJAR	2 Minggu	MS POWERPOINT	200.000,-
PELAJAR	3 Minggu	MS POWERPOINT	300.000,-
MAHASISWA	1 Minggu	MS WORD	200.000,-

MAHASISWA	2 Minggu	MS WORD	300.000,-
MAHASISWA	3 Minggu	MS WORD	400.000,-
MAHASISWA	1 Minggu	MS EXCEL	200.000,-
MAHASISWA	2 Minggu	MS EXCEL	300.000,-
MAHASISWA	3 Minggu	MS EXCEL	400.000,-
MAHASISWA	1 Minggu	MS POWERPOINT	200.000,-
MAHASISWA	2 Minggu	MS POWERPOINT	300.000,-
MAHASISWA	3 Minggu	MS POWERPOINT	400.000,-
UMUM	1 Minggu	MS WORD	300.000,-
UMUM	2 Minggu	MS WORD	400.000,-
UMUM	3 Minggu	MS WORD	500.000,-
UMUM	1 Minggu	MS EXCEL	300.000,-
UMUM	2 Minggu	MS EXCEL	400.000,-
UMUM	3 Minggu	MS EXCEL	500.000,-
UMUM	1 Minggu	MS POWERPOINT	300.000,-
UMUM	2 Minggu	MS POWERPOINT	400.000,-
UMUM	3 Minggu	MS POWERPOINT	500.000,-

7. Jika mengambil dua paket kursus dapat diskon 20% dari total biaya kursus komputer.

BAB 4

PERULANGAN

Capaian Pembelajaran:

1. Mampu memahami logika perulangan
2. Mampu menggunakan argumen perulangan FOR
3. Mampu menggunakan argumen perulangan WHILE
4. Mampu menggunakan argumen perulangan REPEAT
5. Mampu membuat program penerapan logika perulangan

Perulangan merupakan logika pemrograman untuk menjalankan suatu kode program lebih dari satu kali, dimana proses berulang berdasarkan nilai awal inisialisasi dan nilai akhir kondisi berhenti. Pada Delphi 2010 terdapat tiga macam argumen perintah yang dapat digunakan sebagai penerapan logika perulangan yaitu argumen FOR, WHILE, dan REPEAT. (Wahana, 2011)

4.1 Argumen FOR

Argumen FOR akan melakukan perulangan dimana nilai inisialisasi awal dan nilai akhir kondisi berhenti langsung ditentukan menggunakan variabel bantu yang memiliki nilai batasan dalam range tertentu.

```
for <variabel bantu> := <nilai awal inisialisasi> to <nilai akhir  
berhenti> do  
  <argumen1>;
```

```
for <variabel bantu> := <nilai awal inisialisasi> to <nilai akhir  
berhenti> do  
  begin  
    <argumen1>;  
    <argumen2>;  
  end;
```

```
contoh 1:  
for i := 0 to 4 do
```

```
showmessage('Benar'); //tampil jendela pesan tulisan "Benar"  
5 kali
```

contoh 2:

```
for i := 0 to 4 do
```

```
begin
```

```
  showmessage('Benar'); //tampil jendela pesan tulisan "Benar"  
  5 kali
```

```
  showmessage('Berulang'); //tampil jendela pesan tulisan  
  "Berulang" 5 kali
```

```
end;
```

4.2 Argumen WHILE

Argumen WHILE akan melakukan perulangan dengan memeriksa nilai kondisi yang ditentukan diawal terpenuhi bernilai TRUE dan untuk berhenti perlu menentukan kondisi agar bernilai FALSE.

```
while <kondisi=True> do  
begin  
  <argumen1>;  
  <argumen2>;  
  <kendali kondisi=False>;  
end;
```

contoh 1:

```
i := 0;
```

```
while (i <= 4) do
```

```
begin
```

```
  showmessage('Benar'); //tampil jendela pesan tulisan "Benar"  
  5 kali
```

```
  i := i + 1; //penambahan nilai 1 pada variabel i sampai i > 4  
end;
```

4.2 Argumen REPEAT

Argumen REPEAT akan melakukan perulangan minimal sekali dan akan terus melakukan perulangan sampai memeriksa nilai kondisi yang ditentukan terpenuhi bernilai TRUE maka akan berhenti.

```
repeat
```

```
<argumen1>;  
<argumen2>;  
untill <kendali kondisi=True>;  
  
contoh 1:  
i := 0;  
repeat  
    showmessage('Benar'); //tampil jendela pesan tulisan "Benar"  
5 kali  
    i := i + 1; //penambahan nilai 1 pada variabel i sampai i > 4  
until (i = 5);
```

4.3 Membuat Program Logika Perulangan (1)

Membuat program untuk mengilustrasikan cara kerja logika perulangan menggunakan argumen FOR, WHILE dan REPEAT. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 3.1.



Gambar 4.1. Form Logika Perulangan

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Logika Perulangan
Listbox	Name	ListTampil
Button1	Caption	&FOR
Button1	Name	BtFOR
Button2	Caption	&WHILE
Button2	Name	BtWHILE
Button3	Caption	&REPEAT
Button3	Name	BtREPEAT
Button4	Caption	&Keluar
Button4	Name	BtKeluar

Listing program:

Object	Event	Code
BtFOR	OnClick	<pre> procedure TForm1.BtFORClick(Sender: TObject); var i : integer; begin ListTampil.Items.Clear; //kosongkan listbox ListTampil.Items.Add('Perulangan FOR'); //judul perulangan listbox for i := 0 to 4 do ListTampil.Items.Add(IntToStr(i)); end; </pre>
BtWHILE	OnClick	<pre> procedure TForm1.BtWHILEClick(Sender: TObject); var i : integer; begin ListTampil.Items.Clear; //kosongkan listbox ListTampil.Items.Add('Perulangan WHILE'); //judul perulangan listbox i := 0; </pre>

		<pre> while (i <= 4) do begin ListTampil.Items.Add(IntToStr(i)); i := i + 1; // increment variabel i cara lain inc(i) end; end; </pre>
BtREPEAT	OnClick	<pre> procedure TForm1.BtREPEATClick(Sender: TObject); var i : integer; begin ListTampil.Items.Clear; //kosongkan listbox ListTampil.Items.Add('Perulangan REPEAT'); //judul perulangan listbox i := 0; repeat ListTampil.Items.Add(IntToStr(i)); i := i + 1; // increment variabel i cara lain inc(i) until i>4; end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type

```

```

TForm1 = class(TForm)
  BtFOR: TButton;
  BtWHILE: TButton;
  BtREPEAT: TButton;
  BtKeluar: TButton;
  ListTampil: TListBox;
  procedure BtFORClick(Sender: TObject);
  procedure BtWHILEClick(Sender: TObject);
  procedure BtREPEATClick(Sender: TObject);
  procedure BtKeluarClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BtFORClick(Sender: TObject);
var i : integer;
begin
  ListTampil.Items.Clear; //kosongkan listbox
  ListTampil.Items.Add('Perulangan FOR'); //judul perulangan
listbox

  for i := 0 to 4 do
    ListTampil.Items.Add(IntToStr(i));
end;

procedure TForm1.BtWHILEClick(Sender: TObject);
var i : integer;
begin
  ListTampil.Items.Clear; //kosongkan listbox
  ListTampil.Items.Add('Perulangan WHILE'); //judul perulangan
listbox

  i := 0;

```

```

while (i <= 4) do
begin
  ListTampil.Items.Add(IntToStr(i));
  i := i + 1; // increment variabel i cara lain inc(i)
end;
end;

procedure TForm1.BtREPEATClick(Sender: TObject);
var i : integer;
begin
  ListTampil.Items.Clear; //kosongkan listbox
  ListTampil.Items.Add('Perulangan REPEAT'); //judul perulangan
  listbox

  i := 0;
  repeat
    ListTampil.Items.Add(IntToStr(i));
    i := i + 1; // increment variabel i cara lain inc(i)
  until i>4;
end;

procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;

end.

```

4.4 Membuat Program Bilangan Genap Ganjil

Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 4.2.



Gambar 4.2. Form Bilangan Genap Ganjil

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Bilangan Genap Ganjil
Label1	Caption	Batas Bawah
Label2	Caption	Batas Atas
Edit1	Caption	0
Edit1	Name	EdBawah
Edit2	Caption	0
Edit2	Name	EdAtas
Listbox	Name	ListTampil
Button1	Caption	GE&NAP
Button1	Name	BtGenap
Button2	Caption	&GANJIL

Button2	Name	BtGanjil
Button4	Caption	&Keluar
Button4	Name	BtKeluar

Listing program:

Object	Event	Code
BtGenap	OnClick	<pre> procedure TForm1.BtGenapClick(Sender: TObject); var i, BatasBawah, BatasAtas : integer; begin BatasBawah := StrToInt(EdBawah.Text); BatasAtas := StrToInt(EdAtas.Text); ListTampil.Items.Clear; //kosongkan listbox ListTampil.Items.Add('Bilangan Genap'); //judul perulangan listbox for i := BatasBawah to BatasAtas do begin if i mod 2 = 0 then ListTampil.Items.Add(IntToStr(i)); end; end; end; </pre>
BtGanjil	OnClick	<pre> procedure TForm1.BtGanjilClick(Sender: TObject); var i, BatasBawah, BatasAtas : integer; begin BatasBawah := StrToInt(EdBawah.Text); BatasAtas := StrToInt(EdAtas.Text); ListTampil.Items.Clear; //kosongkan listbox ListTampil.Items.Add('Bilangan Ganjil'); //judul perulangan listbox for i := BatasBawah to BatasAtas do begin if i mod 2 <> 0 then ListTampil.Items.Add(IntToStr(i)); </pre>

		end; end;
BtKeluar	OnClick	procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    BtGanjil: TButton;
    BtGenap: TButton;
    BtKeluar: TButton;
    ListTampil: TListBox;
    Label1: TLabel;
    Label2: TLabel;
    EdBawah: TEdit;
    EdAtas: TEdit;
    procedure BtGanjilClick(Sender: TObject);
    procedure BtGenapClick(Sender: TObject);
    procedure BtKeluarClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

```

```

{$R *.dfm}

procedure TForm1.BtGenapClick(Sender: TObject);
var i, BatasBawah, BatasAtas : integer;
begin
  BatasBawah := StrToInt(EdBawah.Text);
  BatasAtas := StrToInt(EdAtas.Text);

  ListTampil.Items.Clear; //kosongkan listbox
  ListTampil.Items.Add('Bilangan Genap'); //judul perulangan
listbox

  for i := BatasBawah to BatasAtas do
  begin
    if i mod 2 = 0 then
      ListTampil.Items.Add(IntToStr(i));
    end;
  end;

procedure TForm1.BtGanjilClick(Sender: TObject);
var i, BatasBawah, BatasAtas : integer;
begin
  BatasBawah := StrToInt(EdBawah.Text);
  BatasAtas := StrToInt(EdAtas.Text);

  ListTampil.Items.Clear; //kosongkan listbox
  ListTampil.Items.Add('Bilangan Ganjil'); //judul perulangan
listbox

  for i := BatasBawah to BatasAtas do
  begin
    if i mod 2 <> 0 then
      ListTampil.Items.Add(IntToStr(i));
    end;
  end;

procedure TForm1.BtKeluarClick(Sender: TObject);
begin
  Close;
end;

```

end.

4.5 Latihan

Membuat program bilangan prima yaitu bilangan yang lebih besar dari 1 dan hanya dapat dibagi oleh 1 dan bilangan itu sendiri serta tidak dapat dibagi dengan bilangan lain. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 4.3.



Gambar 4.3. Form Bilangan Prima

Ketentuan program:

Batas bawah dan batas atas merupakan bilangan bulat untuk menentukan range nilai deret bilangan prima yang akan ditampilkan pada Listbox. Proses mencari bilangan prima setelah tombol “BILANGAN PRIMA” di-klik.

BAB 5

PROCEDURE DAN FUNCTION

Capaian Pembelajaran:

1. Mampu memahami procedure dan function
2. Mampu mengetahui perbedaan procedure dan function
3. Mampu menggunakan procedure dan function
4. Mampu membuat program aplikasi penerapan procedure dan function

Procedure dan Function merupakan sub program berupa beberapa blok perintah-perintah untuk mengerjakan operasi tertentu. Perbedaan utama antara Procedure dan Function yaitu nilai balik (*Return Value*). Procedure menjalankan blok perintah-perintah sesuai operasi yang ditentukan tanpa memiliki nilai balik dari operasi yang dikerjakan tersebut sedangkan Function menjalankan blok perintah-perintah sesuai operasi yang ditentukan dan memiliki nilai balik berdasarkan operasi yang dikerjakan. Procedure dan Function dapat ditulis tanpa parameter atau memiliki parameter tergantung keperluan operasi kode perintah yang dibuat. Parameter merupakan sebutan variabel yang digunakan pada Procedure dan Function dengan tujuan tertentu. Parameter ini dapat menambah fleksibilitas dalam penggunaan Procedure dan Function. (Wahana, 2009)

5.1 Procedure

Procedure digunakan dalam suatu program untuk mengurangi penulisan kode-kode perintah yang sama berulang kali. Procedure juga digunakan untuk menyederhanakan program utama menjadi sub-sub program agar lebih mudah dalam penanganan kode-kode perintah yang semakin banyak dan rumit. Penulisan Procedure tanpa parameter.

```
procedure <nama procedure>;
```

```

begin
  <argumen1>;
  <argumen2>;
end;

contoh:
procedure Jumlah;
begin
  Edit1.Text := 2+3;
end;

```

Penulisan Procedure dengan parameter.

```

procedure <nama procedure>(<parameter> : <tipe data>);
begin
  <argumen1>;
  <argumen2>;
end;

contoh:
procedure Jumlah (Bilangan1, Bilangan2:integer);
begin
  Edit1.Text := Bilangan1+ Bilangan2;
end;

```

5.2 Function

Function digunakan dalam suatu program dengan kegunaan yang sama dengan Procedure, perbedaan utamanya Function memiliki parameter nilai balik. Pada penulisan kode-kode perintah program, selain menjalankan suatu operasi tertentu juga diperlukan suatu nilai yang diharapkan dari hasil operasi kode-kode perintah program. Berdasarkan hal ini, maka function diperlukan untuk memenuhi kebutuhan penulisan kode program tersebut. Penulisan function tanpa parameter.

```

function <nama function>: <tipe data nilai balik>;
begin
  <argumen1>;
  <argumen2>;
  return <nilai balik>;
end;

```

```
contoh:
function Jumlah:integer;
var hitung:integer;
begin
  hitung :=2+3;
  return hitung;
end;
```

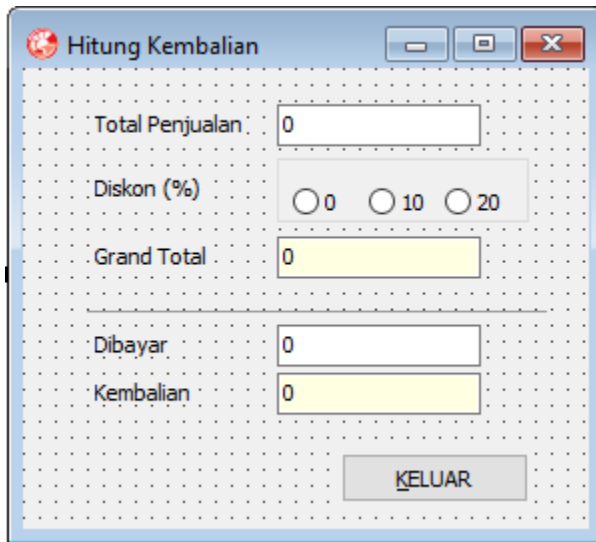
Penulisan Function dengan parameter.

```
function <nama function>(<parameter>:<tipe data parameter>):
<tipe data nilai balik>;
begin
  <argumen1>;
  <argumen2>;
  return <nilai balik>;
end;

contoh:
function Jumlah(bilangan1, bilangan2:integer):integer;
var hitung:integer;
begin
  hitung := bilangan1+ bilangan2;
  return hitung;
end;
```

5.3 Membuat Program Hitung Kembalikan Penjualan

Membuat program untuk menghitung kembalikan pada transaksi penjualan memanfaatkan penarapan procedure. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 5.1.



Gambar 5.1. Form Hitung Kembalikan Penjualan

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Hitung Kembalikan
Label1	Caption	Total Penjualan
Label2	Caption	Diskon (%)
Label3	Caption	Grand Total
Label4	Caption	Dibayar
Label5	Caption	Kembalian
Edit1	Caption	0
Edit1	Name	EdTotal
Edit2	Caption	0
Edit2	Name	EdGrandTotal
Edit2	ReadOnly	True
Edit2	Color	clInfoBk
Edit3	Caption	0
Edit3	Name	EdBayar
Edit4	Caption	0

Edit4	Name	EdKembalian
Edit4	ReadOnly	True
Edit4	Color	clInfoBk
RadioGroup1	Name	RgDiskon
RadioGroup1	Columns	3
RadioGroup1	Items...	0 10 20
Panel1	Caption	(Kosongkan)
Panel1	Height	1
Button1	Caption	&Keluar
Button1	Name	BtKeluar

Listing program:

Object	Event	Code
-	-	<pre>//Deklarasi procedure pada bagian public Unit public { Public declarations } procedure HitungKembalian; end; // Tekan tombol kombinasi Keyboard Ctrl+Shift+C procedure TForm1.HitungKembalian; var total, diskon, grandtotal, bayar, kembalian : real; begin total := StrToFloat(EdTotal.Text); if RgDiskon.ItemIndex = 0 then diskon := 0 * total; if RgDiskon.ItemIndex = 1 then diskon := 0.1 * total; if RgDiskon.ItemIndex = 2 then diskon := 0.2 * total; grandtotal := total - diskon; bayar := StrToFloat(EdBayar.Text); kembalian := bayar - grandtotal;</pre>

		<pre> EdGrandTotal.Text:= FloatToStr(grandtotal); EdKembalian.Text := FloatToStr(kembalian); end; </pre>
EdTotal	OnExit	<pre> procedure TForm1.EdTotalExit(Sender: TObject); begin HitungKembalian; end; </pre>
RgDiskon	OnClick	<pre> procedure TForm1.RgDiskonClick(Sender: TObject); begin HitungKembalian; end; </pre>
EdBayar	OnExit	<pre> procedure TForm1.EdBayarExit(Sender: TObject); begin HitungKembalian; end; </pre>
BtKeluar	OnClick	<pre> procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end; </pre>

Listing program lengkap:

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls;

type
  TForm1 = class(TForm)
    EdTotal: TEdit;

```

```

Label2: TLabel;
Label1: TLabel;
EdGrandTotal: TEdit;
Label3: TLabel;
EdKembalian: TEdit;
Label4: TLabel;
RgDiskon: TRadioGroup;
BtKeluar: TButton;
Panel1: TPanel;
EdBayar: TEdit;
Label5: TLabel;
procedure EdTotalExit(Sender: TObject);
procedure RgDiskonClick(Sender: TObject);
procedure EdBayarExit(Sender: TObject);
procedure BtKeluarClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
  procedure HitungKembalian;
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.HitungKembalian;
var total, diskon, grandtotal, bayar, kembalian : real;
begin
  total := StrToFloat(EdTotal.Text);
  if RgDiskon.ItemIndex = 0 then
    diskon := 0 * total;
  if RgDiskon.ItemIndex = 1 then
    diskon := 0.1 * total;
  if RgDiskon.ItemIndex = 2 then
    diskon := 0.2 * total;

  grandtotal := total - diskon;
  bayar := StrToFloat(EdBayar.Text);

```

```

kembalian := bayar - grandtotal;

EdGrandTotal.Text:= FloatToStr(grandtotal);
EdKembalian.Text := FloatToStr(kembalian);
end;

procedure TForm1.EdTotalExit(Sender: TObject);
begin
    HitungKembalian;
end;

procedure TForm1.RgDiskonClick(Sender: TObject);
begin
    HitungKembalian;
end;

procedure TForm1.EdBayarExit(Sender: TObject);
begin
    HitungKembalian;
end;

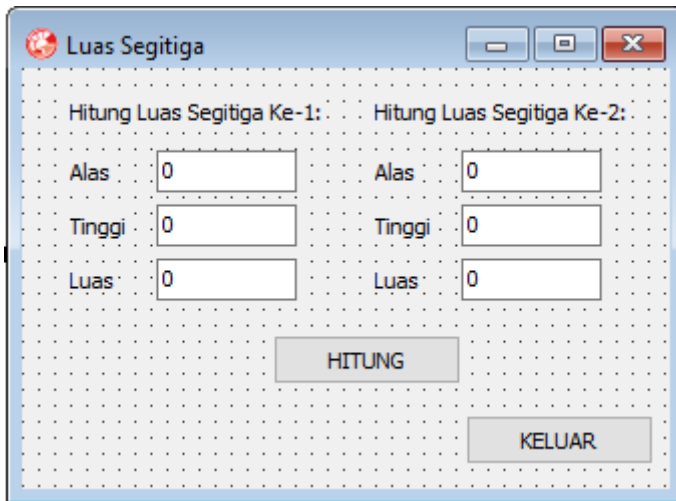
procedure TForm1.BtKeluarClick(Sender: TObject);
begin
    Close;
end;

end.

```

5.4 Membuat Program Hitung Luas Bangun Segitiga

Membuat program untuk menghitung luas dua buah bangun segitiga yang diketahui nilai alas dan tingginya memanfaatkan penarapan function. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 5.2.



Gambar 5.2. Form Hitung Luas Bangun Segitiga

Pengaturan Objek Properties:

Object	Property	Value
Form1	Caption	Luas Segitiga
Label1	Caption	Hitung Luas Segitiga Ke-1:
Label2	Caption	Alas
Label3	Caption	Tinggi
Label4	Caption	Luas
Label5	Caption	Hitung Luas Segitiga Ke-2:
Label6	Caption	Alas
Label7	Caption	Tinggi
Label8	Caption	Luas
Edit1	Caption	0
Edit1	Name	EdAlas1
Edit2	Caption	0
Edit2	Name	EdTinggi1
Edit3	Caption	0

Edit3	Name	EdLuas1
Edit4	Caption	0
Edit4	Name	EdAlas2
Edit5	Caption	0
Edit5	Name	EdTinggi2
Edit6	Caption	0
Edit6	Name	EdLuas2
Button1	Caption	&HITUNG
Button1	Name	BtHitung
Button2	Caption	&Keluar
Button2	Name	BtKeluar

Listing program:

Object	Event	Code
-	-	<pre>//Deklarasi function pada bagian public Unit public { Public declarations } function LuasSegitiga(Alas, Tinggi:integer):real; end; // Tekan tombol kombinasi Keyboard Ctrl+Shift+C function TForm7.LuasSegitiga(Alas, Tinggi: integer): real; begin LuasSegitiga := 0.5 * Alas * Tinggi; end;</pre>
BtHitung	OnClick	<pre>procedure TForm1.BtHitungClick(Sender: TObject); var Alas, Tinggi : integer; begin Alas := StrToInt(EdAlas1.Text); Tinggi := StrToInt(EdTinggi1.Text); EdLuas1.Text := FloatToStr(LuasSegitiga(Alas,Tinggi)); Alas := StrToInt(EdAlas2.Text);</pre>

		<pre>Tinggi := StrToInt(EdTinggi2.Text); EdLuas2.Text := FloatToStr(LuasSegitiga(Alas,Tinggi)); end;</pre>
BtKeluar	OnClick	<pre>procedure TForm1.BtKeluarClick(Sender: TObject); begin Close; end;</pre>

Listing program lengkap:

```
unit Unit1;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics,
    Controls, Forms,
    Dialogs, StdCtrls;

type
    TForm1 = class(TForm)
        EdAlas1: TEdit;
        EdTinggi1: TEdit;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        EdAlas2: TEdit;
        EdTinggi2: TEdit;
        Label4: TLabel;
        Label5: TLabel;
        Label6: TLabel;
        EdLuas1: TEdit;
        Label7: TLabel;
        EdLuas2: TEdit;
        Label8: TLabel;
        BtKeluar: TButton;
        BtHitung: TButton;
        procedure BtHitungClick(Sender: TObject);
        procedure BtKeluarClick(Sender: TObject);
    private
```

```

    { Private declarations }
public
    { Public declarations }
    function LuasSegitiga(Alas, Tinggi:integer):real;
end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

function TForm1.LuasSegitiga(Alas, Tinggi: integer): real;
begin
    LuasSegitiga := 0.5 * Alas * Tinggi;
end;

procedure TForm1.BtHitungClick(Sender: TObject);
var Alas, Tinggi : integer;
begin
    Alas := StrToInt(EdAlas1.Text);
    Tinggi := StrToInt(EdTinggi1.Text);
    EdLuas1.Text := FloatToStr(LuasSegitiga(Alas,Tinggi));

    Alas := StrToInt(EdAlas2.Text);
    Tinggi := StrToInt(EdTinggi2.Text);
    EdLuas2.Text := FloatToStr(LuasSegitiga(Alas,Tinggi));
end;

procedure TForm1.BtKeluarClick(Sender: TObject);
begin
    Close;
end;

end.

```

5.5 Latihan

Membuat program menghitung sisi miring dua buah bangun segitiga dengan rumus pythagoras menerapkan procedure atau function. Buat Project baru dengan memilih menu File >New >VCL Form Application-Delphi. Atur tata letak Form yang masih kosong seperti gambar 5.3.

Rumus Pythagoras	
Sisi Miring Segitiga Ke-1:	Sisi Miring Segitiga Ke-2:
Sisi Alas: <input type="text" value="0"/>	Sisi Alas: <input type="text" value="0"/>
Sisi Tegak: <input type="text" value="0"/>	Sisi Tegak: <input type="text" value="0"/>
Sisi Miring: <input type="text" value="0"/>	Sisi Miring: <input type="text" value="0"/>
<input type="button" value="HITUNG"/>	
<input type="button" value="KELUAR"/>	

Gambar 5.3. Form Hitung Sisi Miring Bangun Segitiga

Ketentuan program:

Nilai sisi alas dan sisi tegak masing-masing segitiga diketahui. Proses menghitung sisi miring pada kedua bangun segitiga setelah tombol “HITUNG” di-klik dan menampilkan hasil nilai sisi miring.

GLOSARIUM

- Antarmuka: Bagian yang berinteraksi dengan pengguna program.
- Argumen: Kode perintah program.
- begin: Kata kunci untuk memulai suatu kode perintah Delphi.
- Bilangan Bulat: Bilangan tanpa nilai desimal.
- Bilangan Pecahan: Bilangan desimal.
- Borland: Perusahaan pengembang awal Delphi.
- Coding: Kegiatan menulis kode program.
- Compiling: Proses membuat file executable.
- const: Kata kunci untuk mendeklarasikan konstanta pada Delphi.
- Debugging: Mendeteksi pesan kesalahan.
- Design-time: Saat project belum dijalankan.
- Embarcadero: Perusahaan pengembang Delphi saat ini.
- exe: Ekstensi file yang dapat dieksekusi pada sistem operasi Windows.
- F9: Salah satu tombol bagian atas keyboard.
- FALSE: Istilah umum untuk logika salah.
- Font: Jenis huruf.
- Font Style: Gaya penulisan huruf.
- Form Designing: Kegiatan desain Form.
- Form Kosong: Form yang baru dibuat dan belum memiliki komponen apapun.
- IDE: Lingkungan pengembangan terintegrasi.
- IF Bersarang: Argumen IF didalam Argumen IF lainnya.
- implementation: Bagian mendeklarasikan Form dan variabel publik.
- IntelliSense: Fitur membantu menampilkan property dan event yang dimiliki objek.
- Italic: Penulisan teks miring.
- Keyboard: Papan ketik komputer.
- Kompilasi: Proses meng-compile project.
- Kompleks: bertambah banyak.
- Kondisi: Keadaan yang bernilai TRUE atau FALSE.

Listing Program: Kode-kode program.

Logika Pemrograman: Alur kontrol proses pemrograman.

Logika Percabangan: Alur kontrol memilih suatu proses program.

Logika Perulangan: Alur kontrol proses mengulang perintah kode program.

Method: procedure atau function yang dimiliki oleh objek.

Nilai Inisialisasi: Penentuan nilai awal suatu variabel.

Numerik: Tipe angka yang dapat dilakukan proses aritmatika.

Object Pascal: Bahasa pemrograman Pascal.

Operand: Objek pada operasi menggunakan simbol operator.

Operator Teks: Simbol untuk pengolahan tipe data String.

Parameter: Variabel pada suatu procedure atau function.

Populer: Sering digunakan.

private: Bagian mendeklarasikan variabel publik dikenali unitnya sendiri.

public: Bagian mendeklarasikan variabel publik dikenali seluruh unit lain.

Run-time: Saat project sudah dijalankan.

Simbol Operator: Lambang yang digunakan menulis suatu operator.

Teks: ditangani oleh tipe data String.

Times New Roman: Jenis font populer yang disediakan sistem operasi.

TRUE: Istilah umum untuk logika benar.

Underline: Penulisan teks dengan garis bawah.

var: Kata kunci untuk mendeklarasikan variabel pada Delphi.

Variabel Bantu: Variabel dengan nama bebas selama diperbolehkan oleh IDE Delphi.

Visual: Terlihat mata.

Windows: Sistem operasi pengembangan Microsoft.

DAFTAR PUSTAKA

Embarcadero, Inc., Delphi 2010 Documentation.

Komputer, Wahana, Panduan Praktis Delphi 2009, ANDI, 2009

Setiawan, Yudha C, Panduan Object-Oriented Programming (OOP)

Dasar Pemrograman Delphi, ANDI, 2004

Komputer, Wahana, Shortcourse: Cepat Mahir Delphi 2011, ANDI,
2011

BIOGRAFI PENULIS

HELDIANSYAH. Lahir di Balikpapan, 22 Januari 1985. Menyelesaikan program S1 jurusan Sistem Informasi di STMIK Indonesia Banjarmasin Tahun 2008 dan S2 jurusan Teknologi Informasi STTS Surabaya tahun 2014. Saat ini bekerja sebagai staf pengajar pada Program Studi Manajemen Informati Jurusan Administrasi Bisnis Politeknik Negeri Banjarmasin. Waktu luang digunakan mengajar di beberapa perguruan tinggi swasta di Banjarmasin dan menjadi konsultan IT serta melayani pembuatan aplikasi database pada instansi pemerintah dan perusahaan swasta. Penulis juga mempunyai minat besar pada penulisan jurnal penelitian bidang Deep Learning, Big Data, dan Machine Learning.

PANDUAN PEMROGRAMAN EMBARCADERO DELPHI

HELDIANSYAH

Delphi 2010 bahasa pemrograman pengembangan dari bahasa object pascal yang dikembangkan oleh perusahaan bernama Embarcadero dimana sebelumnya telah mengakuisisi perusahaan Borland yang merupakan perusahaan pengembang Delphi sejak versi awal. Delphi 2010 termasuk dalam pemrograman visual yang menyediakan lingkungan pengembangan terintegrasi dikenal dengan IDE (Integrated Development Environment). IDE Delphi ini disebut terintegrasi dimana dalam mengembangkan program aplikasi seperti mendesain antarmuka (form designing), menulis kode listing program (coding), mendeteksi error (debugging), membuat file exe (compiling) dan lain-lain dapat dilakukan semua disini. Selain itu IDE Delphi dilengkapi kemampuan intuitif dengan fitur IntelliSense yang dapat menampilkan pilihan property, method, procedure, function, dan variable milik suatu objek. (Embarcadero, 2009).



Penerbit Poliban Press

Redaksi :

Politeknik Negeri Banjarmasin, Jl. Brigjen H. Hasan Basry,
Pangeran, Komp. Kampus ULM, Banjarmasin Utara

Telp : (0511)3305052

Email : press@poliban.ac.id

ISBN 978-623-7694-10-6

