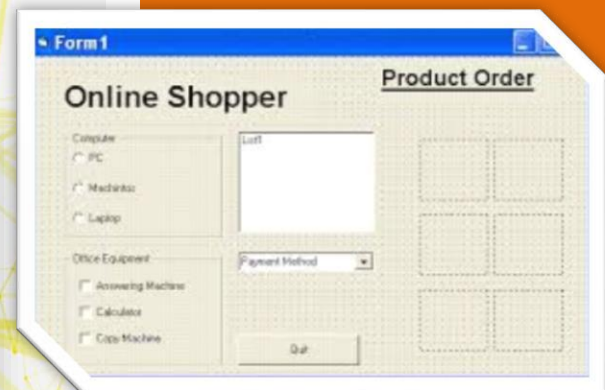




Modul Praktikum

# PEMROGRAMAN VISUAL



AGUS SUJARWADI

NIM :

NAMA:

# MODUL PRAKTIKUM PEMROGRAMAN VISUAL

**AGUS SUJARWADI**

FAKULTAS TEKNOLOGI INFORMASI & ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2017

## **KATA PENGANTAR**

Modul Praktikum Pemrograman Visual merupakan salah satu pegangan bagi mahasiswa yang mengambil matakuliah Praktikum pemrograman visual. Modul ini dapat dipakai sebagai acuan bagi mahasiswa dalam Praktikum Pemrograman Visual. Modul ini dibuat untuk Praktikum satu semester yang terdiri atas 14 kali pertemuan. Setiap kali pertemuan selalu didampingi dengan dosen/asisten untuk menambah penjelasan dan pemahaman sekaligus sebagai fasilitator dalam memahami modul ini.

Semoga Tuhan YME memberi hidayahNya sehingga modul ini dapat memberi manfaat bagi mahasiswa yang mengambil matakuliah Praktikum Pemrograman Visual. Tentu modul ini jauh dari kesempurnaan sehingga sangat membutuhkan masukan–masukan dari semua pihak dan silahkan menghubungi penulis melalui [agus.sujarwadi@staff.uty.ac.id](mailto:agus.sujarwadi@staff.uty.ac.id)

# DAFTAR ISI

COVER .....	1
KATA PENGANTAR.....	2
DAFTAR ISI.....	3
PENDAHULUAN .....	4
BAB 1. TIPE DATA PADA DELPHI .....	6
BAB 2. VARIABEL,KONSTANTA & OPERATOR .....	11
BAB 3. KOMPONEN LAIN .....	15
BAB 4. FILE TEKS & PESAN .....	24
BAB 5. OLE.....	28
BAB 6. GRAFIK .....	32
BAB 7. TOOL BAR & STATUS BAR.....	39
BAB 8. TREEVIEW & LISTVIEW .....	44
BAB 9. DATABASE DESKTOP.....	50
BAB 10. DATABASE ACCESS & PENCARIAN DATA .....	55
BAB 11. NAVIGASI RECORD .....	60
BAB 12. MEMFORMAT & MEMVALIDASI DATA .....	65
BAB 13. DATA MODUL.....	67
BAB 14. REPORT DENGAN QREPORT .....	69
DAFTAR PUSTAKA.....	70

## PENDAHULUAN

### *i. Deskripsi Materi*

Materi dalam modul ini menerangkan beberapa perangkat dan cara penggunaannya serta fungsi dalam pemrograman khususnya Bahasa Pemrograman Delphi. Selain perangkat, juga dibahas tentang tipe data dan istilah-istilah yang secara standard digunakan dalam Delphi. Dalam bagian lain juga dibahas tentang trik pemrograman, terakhir, modul ini membahas tentang pembuatan aplikasi sederhana dengan melibatkan DBMS berbasis desktop.

Modul yang disusun ini memiliki keterkaitan dengan modul pada mata kuliah lain sebagai alat bantu yang memberi kemudahan bagi mahasiswa baik untuk menyelesaikan tugas sehari-hari maupun mempersiapkan mahasiswa dalam penyusunan tugas akhir.

Setelah mempelajari modul ini, mahasiswa mampu merancang dan membangun sebuah aplikasi sederhana dengan bantuan bahasa pemrograman Delphi.

### *ii. Prasyarat*

- Mahasiswa mampu mengoperasikan seperangkat komputer.
- Mahasiswa tidak asing dengan istilah-istilah komputer seperti mouse, klik ganda, klik kanan, enter, drag and drop.
- Mahasiswa disarankan telah mengambil mata kuliah Aplikasi Komputer dan Algoritma dan Struktur Data.

### *iii. Petunjuk Pemakaian Modul*

Modul ini dapat digunakan mahasiswa dengan pertimbangan sebagai berikut :

- Mahasiswa telah memiliki modul dan telah membaca modul sebelum mata kuliah dimulai.
- Mahasiswa mempelajari serta mengidentifikasi isi modul yang diuraikan lebih rinci oleh dosen pengampu.
- Mahasiswa dan dosen mendiskusikan materi untuk mencari penyelesaian terhadap kasus tertentu.
- Mahasiswa menyimpulkan isi materi yang didiskusikan
- Mahasiswa menjawab soal latihan yang diberikan
- Pemberian pengayaan materi bagi mahasiswa yang telah memahami dan menyelesaikan soal latihan.
- Memberikan tinjauan ulang terhadap materi sekaligus mengidentifikasi kesulitan-kesulitan mahasiswa dalam memahami materi.

#### *iv. Standar Kompetensi*

1. Mahasiswa mampu menggunakan perangkat dalam Delphi
2. Mahasiswa mampu membuat dan menyelesaikan tugas-tugas pembuatan program berbasis Delphi
3. Mahasiswa mampu merancang dan membangun sebuah aplikasi sederhana berbasis Delphi

## BAB 1. TIPE DATA PADA DELPHI

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran : Mengetahui bentuk-bentuk tipe data yang digunakan dalam delphi

Indikator : - Mampu menggunakan berbagai bentuk tipe data yang ada

### Dasar Teori

Apa saja tipe-tipe dasar dari data yang bisa diolah dalam pemrograman Delphi? Kita perlu tahu hal tersebut sebab pengolahan data untuk tiap tipe yang berbeda akan berbeda pula. Misalnya untuk angka 1024, angka 1024 yang tertera pada nomor undian tidak akan digunakan untuk operasi matematika, hal tersebut membuat tipe data angka 1024 pada nomor undian berbeda dengan tipe data pada angka 1024 pada perhitungan jumlah barang. Pada perhitungan jumlah barang angka 1024 akan digunakan dengan operasi matematika, untuk menghitung laba misalnya. Untuk lebih jelasnya akan dibahas beberapa tipe data yang sering digunakan dalam pengolahan data dan konversi tipe data satu ke tipe data yang lain:

### I. JENIS TIPE DATA

#### a. Bilangan Integer

Tabel 1. Tipe Data Bilangan Integer

<b>Tipe Data</b>	<b>Ukuran Tempat</b>	<b>Rentang Nilai</b>
Byte	1 byte	0 s/d +255
Shortint	1 byte	-28 s/d +127
integer	2 bytes	-32768 s/d 32767
Word	2 bytes	0 s/d 65535
Longint	4 bytes	2147483648 s/d 2147483647

Contoh bilangan integer adalah: 34 6458 -90 0 1112 Penggolongan tipe data integer tersebut dimaksudkan untuk membatasi alokasi memori yang dibutuhkan misalkan untuk suatu perhitungan dari suatu variabel bilangan diperkirakan nilai maksimumnya 32767 kita cukup mendeklarasikan variabel bilangan sebagai integer (2 byte), daripada sebagai longint(4 byte).

### b. Bilangan Real

Bilangan real atau nyata merupakan jenis bilangan pecahan, dapat dituliskan secara biasa atau model scientific . Contoh bilangan real: 34.265 -3.55 0.0 35.997E+11, dimana E merupakan simbol perpangkatan 10. Jadi 452.13 mempunyai nilai sama dengan 4.5213e2. Penggolongan tipe data bilangan real dapat dilihat pada tabel 2.

Tabel 2. Bilangan Real

<b>Type Data</b>	<b>Ukuran Tempat</b>	<b>Rentang Nilai</b>
Real	6 bytes	2.9 x 10 <sup>-39</sup> s/d 1.7 x 10 <sup>38</sup>
single	4 bytes	1.5 x 10 <sup>45</sup> s/d 3.4 x 10 <sup>38</sup>
double	8 bytes	5.0 x 10 <sup>-324</sup> s/d 1.7 x 10 <sup>308</sup>
extended	10 bytes	3.4 x 10 <sup>-4932</sup> s/d 1.1 x 10 <sup>4932</sup>
comp	8 bytes	-9.2x 10 <sup>18</sup> s/d 9.2x 10 <sup>18</sup>

### c. Char

tipe data ini menyimpan karakter yang diketikkan dari keyboard, memiliki 266 macam yang terdapat dalam tabel ASCII (American Standard Code for Information Interchange). Contoh: 'a' 'B' '+', dsb. Yang perlu diingat bahwa dalam menuliskannya harus dengan memakai tanda kutip tunggal. Jenis data ini memerlukan alokasi memori sebesar 1(satu) byte untuk masing-masing data.

### d. Tipe Data Boolean

merupakan tipe data logika, yang berisi dua kemungkinan nilai: TRUE (benar) atau FALSE (salah). Turbo Pascal for Windows memiliki tiga macam jenis ini yaitu:



Boolean, WordBool, dan LongBool. Tipe boolean memakai memori paling kecil, sedangkan WordBool dan LongBool dipakai untuk menulis program yang sesuai dengan lingkungan Windows.

Tabel 2. Tipe Data Boolean

<b>Tipe Data</b>	<b>Ukuran Tempat</b>
Boolean	1 byte
WordBool	2 byte
Longbool	3 byte

Sebagai bilangan ordinal boolean true mempunyai nilai 1(satu), sedangkan false nilainya adalah 0(nol).

## **2.Tipe Data Terstruktur**

tipe ini terdiri atas : array, record, set, dan file. String adalah tipe data jenis array, tetapi karena string memiliki kekhasan tersendiri sebagai array dari karakter maka penulis perlu memberikan penjelasan tersendiri. Sedangkan untuk array, record, dan file perlu dijelaskan dalam bab yang lain karena agak banyak hal-hal yang perlu dibahas.

## **3-Tipe Data Pointer**

Pointer merupakan variabel khusus yang berisi suatu address (alamat) di lokasi lain didalam memory. Suatu variabel yang points(menunjuk) ke sesuatu sehingga disebut pointer. Ada dua macam pointer:

-typed(tertentu): merupakan pointer yang menunjuk pada tipe data tertentu pada variable.

-generic(umum): merupakan pointer yang tidak menunjuk pada tipe data tertentu pada variable.

Contoh deklarasi pointer:

```
var p : ^integer;
```

## 6. Tipe Data Enumerated

Tipe data ini adalah berupa nilai urutan dari data yang telah dibuat.

Penulisan pada deklarasi variable:

```
[variable] : ( [data0],[data1],[data2] );
```

Masing-masing data akan diberikan nilai integer secara urut. Urutan dimulai dengan nilai urut 0.

Jika ingin melihat nilai integer dari setiap data maka dengan perintah:

```
ord( [data1] )
```

## 7. Tipe data Sub Range

Tipe data ini memungkinkan kita membuat range sendiri. Dengan perintah:

```
[variable] : [constant start]..[constant end];
```

contoh:

```
a : 5..10;
```

Jika kita mendeklarasikan 5 adalah a maka proses compile akan terus berjalan jika kita mendeklarasikan 15 adalah a maka proses compile tidak akan berjalan atau berhenti, sehingga terdapat error pada proses compile sehingga tidak dapat di run.

## II. KONVERSI TIPE DATA

Untuk konversi dari type data satu ke yang lain anda dapat menggunakan perintah seperti yang berikut ini:

[tipe data yang akan diubah] to [tipe data yang diinginkan] ( [data yang typenya akan diubah] )

Misalnya:

- `inttostr` : mengubah type data integer ke string
- `strtobool` : mengubah type data string ke boolean

Perintah lain untuk konversi:

- `int([data])` : mengubah type data floating point ke integer
- `round([data])` : mengubah type data floating point ke integer

### Latihan

1. Buat sebuah form yang didalamnya terdapat 3 buah komponen edit, 3 buah komponen label dan 1 button proses
2. Ubah masing-masing property dari komponen dan sesuaikan, dimana form soal pertama digunakan untuk proses penjumlahan dan pengurangan
3. Ubah tipe data yang digunakan pada soal 2, dan Anda diberi kebebasan untuk menentukan tipe datanya, apa yang terjadi pada hasil proses, diskusikan

### Tugas

1. Buat algoritma dari soal latihan diatas
2. Kumpulkan dalam bentuk tercetak beserta screen shoot latihan yang dibuat diatas

## BAB 2. VARIABEL, KONSTANTA & OPERATOR

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Mengetahui tentang variable, konstanta dan operator yang digunakan dalam delphi

Indikator : Mampu menggunakan variable, konstanta dan operator dalam pemrograman

### Dasar Teori

#### □ Komentar

Dipakai untuk memberikan penjelasan atau keterangan di dalam baris program. Teks yang ditulis sebagai komentar tidak akan dikompilasi oleh kompiler pada saat program aplikasi dijalankan.

Untuk menuliskan sebuah komentar, dapat menggunakan salah satu tanda dari 3 bentuk tanda yang disediakan:

- ❖ Kurung kurawal  $\Rightarrow$  { Komentar Program }
- ❖ Kurung Bintang  $\Rightarrow$  (\* Komentar Program \*)
- ❖ Slash ganda  $\Rightarrow$  // Komentar Program

Komentar dengan tanda kurung kurawal dan kurung bintang menggunakan tanda pembuka dan tanda penutup, sehingga teks yang dianggap sebagai komentar adalah teks yang terletak diantara tanda pembuka dan tanda penutup.

Sedangkan tanda komentar slash ganda tidak menggunakan tanda penutup, sehingga semua teks yang terletak di belakang tanda slash ganda dianggap sebagai komentar.

#### □ Variable

Secara umum variabel berfungsi untuk menyimpan data. Pendeklarasian variabel susunannya:

**Var nama\_variabel : tipe\_variabel;**

Aturan yang harus di ikuti dalam menuliskan nama variabel:

- ❖ Panjang penulisannya maksimum 63 karakter. Jika lebih dari 63 maka compiler.
- ❖ Penulisannya hanya terdiri dari huruf, angka, garis bawah.
- ❖ Penulisannya tidak boleh diawali dengan angka. Jadi harus diawali dengan huruf atau garis bawah.
- ❖ Nama variabel tidak boleh merupakan sebuah kata kunci milik Delphi, seperti case, if, while, end, dan lain-lain.
- ❖ Nama variabel tidak boleh mengandung karakter yang di pakai untuk operator, seperti (, ), +, -, \*, /, <, >, :, ;, dan lain-lain.

□ Konstanta

Merupakan suatu nilai yang bersifat tetap. Jenis data untuk suatu nilai konstanta dapat berupa data angka (numeric), teks (string), true, false, dan nil. Anda dapat menggunakan suatu nama untuk mewakili konstanta.

□ Operator

Dalam melaksanakan pengolahan data, Delphi menyediakan berbagai operator dengan urutan atau derajat proses pelaksanaan yang berbeda untuk beberapa operator yang di libatkan pada suatu proses. Tabel derajat proses pengolahan data yang di miliki Delphi:

Urutan	Operator
1	@, not
2	*, /, div, mod, and, shl, shr, as
3	+, -, or, xor
4	=, <, >, <=, >=, <>, in, is

Apabila ada proses yang terdiri dari beberapa operator yang mempunyai urutan derajat yang sama, maka proses pengoperasian akan mendahulukan proses yang di tulis paling kiri.

□ Operator Assignment

Assignment atau operator fungsi di tuliskan dengan bentuk “:=” (titik dua sama dengan) dan berfungsi untuk memasukan suatu nilai data ke dalam sebuah variabel, dengan bentuk penulisan:

**Nama Variabel := Ekspresi ;**

□ Operator Aritmatika

Berikut ini adalah tabel operator aritmatika yang dapat di gunakan untuk mendukung operasi aritmatika:

Operator	Fungsi	Tipe yang Diproses	Tipe Hasil Proses
*	Perkalian	Integer, Real	Integer, Real
/	Pembagian real	Integer, Real	Integer, Real
+	Penjumlahan	Integer, Real	Integer, Real
-	Pengurangan	Integer, Real	Integer, Real
Div	Pembagian integer	Integer	Integer
Mod	Sisa hasil bagi	Integer	Integer

Sedangkan untuk operasi perpangkatan, Anda dapat menggunakan rumus:

$$a^b = \exp(b * \ln(a))$$

□ Operator Relasi

Berfungsi untuk membandingkan suatu nilai (ekspresi) dengan nilai (ekspresi) lain yang akan menghasilkan suatu nilai logika (boolan) yaitu True atau False. Kedua data nilai yang dibandingkan tersebut harus memiliki tipe data yang sama. Berikut ini adalah table operator relasi yang di sediakan Delphi:

Operator	Keterangan
=	Sama dengan
>	Lebih besar
<	Lebih kecil
>=	Lebih besar atau sama dengan
<=	Lebih kecil atau sama dengan

<>	Tidak sama dengan
----	-------------------

□ Operator Logika

Operator logika di bagi menjadi dua kelompok: operator bit dan operator Boolean. Operator bit berhubungan dengan pergeseran atau perbandingan pada level bit. Operator Boolean digunakan untuk menyatakan satu atau lebih data atau ekspresi logika yang akan menghasilkan nilai logika (Boolean) yang baru True atau False

**Operator Bit:**

Operator	Keterangan	Tipe Data	Tipe Hasil
And	Dan	Integer	Boolean
Or	Atau	Integer	Boolean
Not	Tidak	Integer	Boolean
Xor	Exclusive Or	Integer	Boolean
Shl	Geser ke kiri	Integer	Boolean
Shr	Geser ke kanan	Integer	Boolean

Operator Boolean selalu memberikan hasil True atau False, sedangkan operasi bit melakukan operasi bit per bit pada nilai tipe integer.

**Operator Boolean:**

Operator	Keterangan	Tipe Data	Tipe Hasil
And	Dan	Boolean	Boolean
Or	Atau	Boolean	Boolean
Not	Tidak	Boolean	Boolean
Xor	Exclusive Or	Boolean	Boolean

Operator logika **And** hanya akan menghasilkan nilai True jika lebih dari satu ekspresi yang menggunakan operator And bernilai True. Apabila ada satu ekspresi atau lebih bernilai False maka operator logika And akan menghasilkan nilai False.

### Latihan

1. Buat sebuah form yang dapat digunakan untuk penjumlahan dan pengurangan dua buah nilai
2. Tambahkan proses perkalian dan pembagian dari soal latihan nomor 1 diatas

### Tugas

Kumpulkan dalam bentuk tercetak beserta screen shoot latihan yang dibuat diatas



## BAB 3. KOMPONEN LAIN

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Mengetahui bentuk penggunaan beberapa komponen interface  
Indikator : - Mampu menggunakan beberapa komponen Delphi

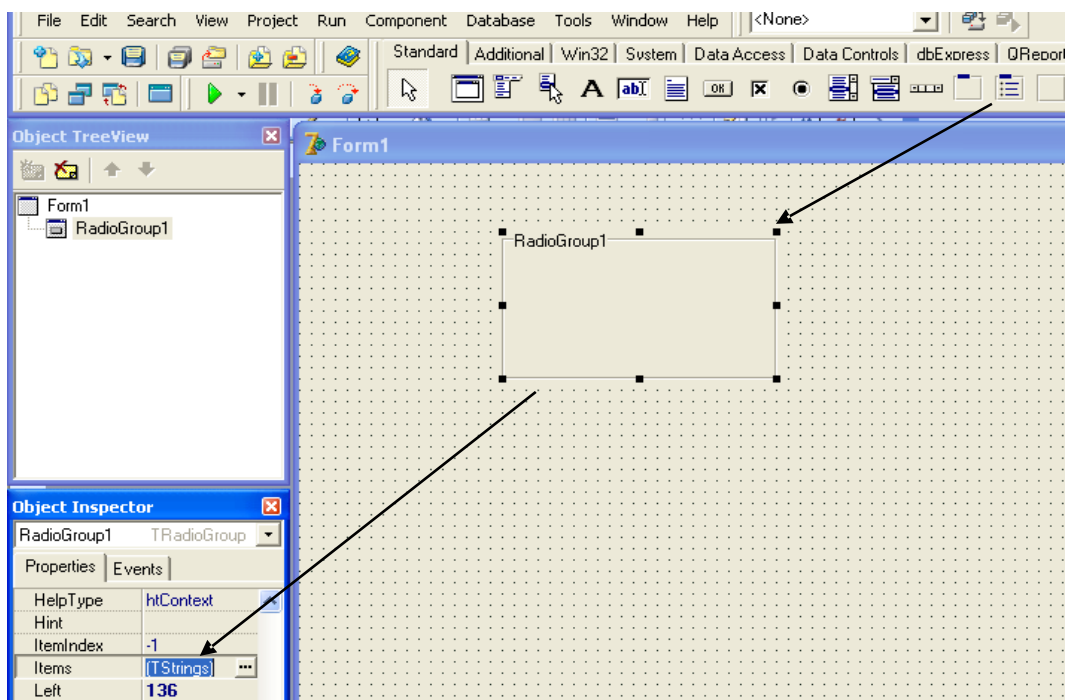
### Dasar Teori dan Praktek

i merupakan bahasa pemrograman berbasis visual (OOP), oleh karena itu untuk pengaturan interface biasanya melibatkan objek-objek yang terdapat dalam komponen palletnya Delphi. Beberapa komponen inputan di dalam Delphi, diantaranya: komponen Radiogroup, Checkbox, Mainmenu, dan memo.

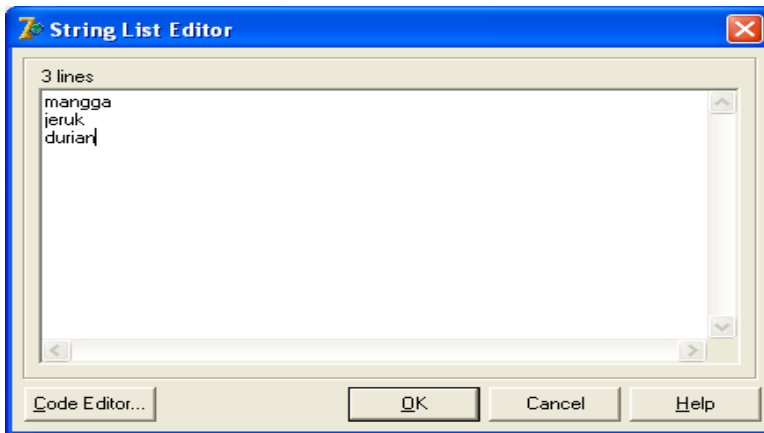
Untuk lebih jelasnya perhatikan gambar interface pada Delphi berikut serta lakukan sesuai perintah yang diberikan:

### 1.1 Komponen TRadioGroup

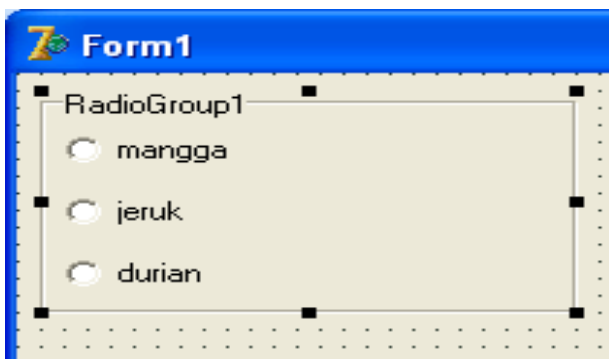
1. Pilih komponen TRadioGroup kemudian drag ke dalam TForm



2. Isi properties Items untuk komponen TradioGroup setelah klik 2 kali pada isian items tersebut.
3. Selanjutnya akan muncul window String List Editor.



4. Isi string secara berurutan sebanyak 3 baris yaitu : mangga , jeruk, dan durian.
5. setelah selesai, kemudian klik OK, yang akan menutup window string list editor dan kemudian kembali aktif di komponen TForm.
6. perhatikan bahwa sekarang telah terdapat 3 item pilihan pada komponen TRadioGroup.



7. Untuk menggunakan komponen ini sebagai kendali pemilihan, maka isi perintah berikut pada event OnClick.

```

procedure TForm1.RadioGroup1Click(Sender: TObject);
begin
  If radiogroup1.ItemIndex = 0 then
    form1.Caption := 'Saya suka mangga';
  If radiogroup1.ItemIndex = 1 then
    form1.Caption := 'Saya suka jeruk';
  If radiogroup1.ItemIndex = 2 then
    form1.Caption := 'Saya suka durian';
end;

```

8. Atau bisa juga dituliskan dengan menggunakan case of menjadi

```

procedure TForm1.RadioGroup1Click(Sender: TObject);
begin

```

```

case radiogroup1.ItemIndex of
0 : form1.Caption := 'Saya suka '+
    radiogroup1.Items.Strings [0];
1 : form1.Caption := 'Saya suka '+
    radiogroup1.Items.Strings [1];
2 : form1.Caption := 'Saya suka '+
    radiogroup1.Items.Strings [2];
end;
end;

```

9. Hasil yang sama juga dapat diperoleh hanya dengan menggunakan satu baris perintah yaitu:

```

procedure TForm1.RadioGroup1Click(Sender: TObject);
begin
form1.Caption := 'Saya suka '+
radiogroup1.Items.Strings [radiogroup1.itemindex];
end;

```

## 1.2 Komponen TCheckBox

- Berbeda dengan RadioGroup yang hanya mengijinkan satu pilihan dari beberapa pilihan, pada CheckBox dimungkinkan untuk memilih lebih dari satu pilihan secara bersamaan.
- Drag tiga komponen CheckBox ke dalam komponen TForm, kemudian isi properties masing-masing komponen dengan isian sebagai berikut:

Komponen	Name	Caption
CheckBox1	Chbx_Mangga	Mangga
CheckBox2	Chbx_Jeruk	Jeruk
CheckBox3	Chbx_Durian	Durian

- Isialisasi variable global YgDisukai bertipe string. Penulisan inialisasi pada code editor window. Posisi terletak setelah deklarasi tipe.

```

var
    Form1: TForm1;
    YgDisukai: string;

implementation

```

4. Kembali ke 3 komponen CheckBox, tambahkan satu komponen Button dan sisipkan perintah berikut ini pada prosedur

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

```
Begin
```

```
    If Chbx_mangga.Checked = true then
```

```
        YgDisukai := YgDisukai + 'Mangga'+',';
```

```
    If Chbx_jeruk.Checked = true then
```

```
        YgDisukai := YgDisukai + 'Jeruk'+',';
```

```
    If Chbx_durian.Checked = true then
```

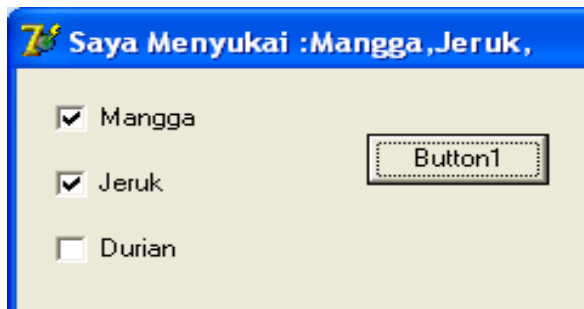
```
        YgDisukai := YgDisukai + 'Durian'+',';
```

```
Form1.Caption := 'Saya Menyukai :'+YgDisukai;
```

```
YgDisukai :='';
```

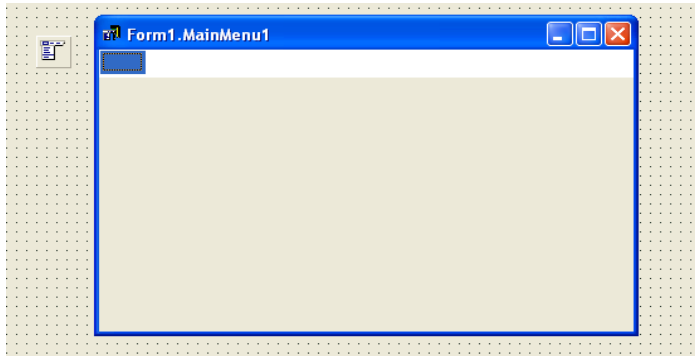
```
End;
```

5. Jalankan program, pilih satu atau lebih pilihan nama buah, dan kemudian klik button untuk menampilkan item yang dipilih pada caption komponen CheckBox.

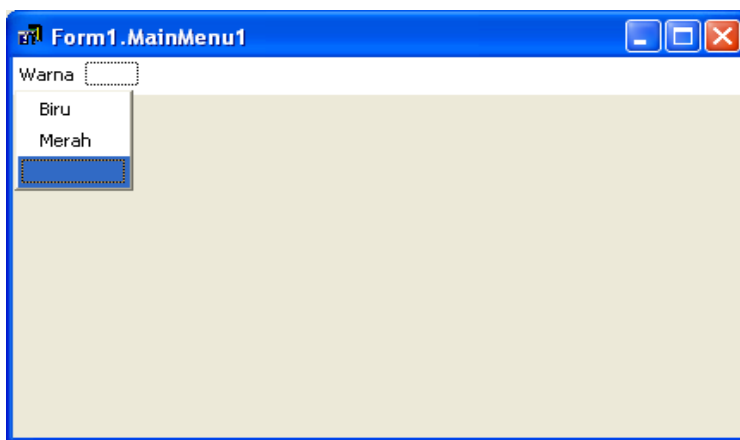


### 1.3 Komponen TMainMenu

1. Komponen ini digunakan untuk menempatkan menu-menu pilihan yang dapat dijalankan pada suatu interface program. Penggunaan menu akan lebih menghemat tempat jika dibandingkan dengan menggunakan button. Juga memungkinkan pengelompokan fungsi-fungsi suatu program. Kelemahannya adalah fungsi-fungsi tersebut menjadi tersembunyi karena tidak langsung terlihat pada form utama.
2. contoh berikut ini akan dicoba membuat menu untuk fungsi yang digunakan mengubah warna form.
3. Masukkan komponen TMainMenu kedalam TForm, pilih komponendan klik 2 kali, sehingga muncul window untuk perancangan menu seperti gambar berikut:



4. Klik boks biru pada ujung kiri atas, isi properties caption dengan Warna.
5. Menu warna kemudian diturunkan menjadi 2 pilihan menu yaitu, Biru dan Merah. Sehingga diperoleh seperti gambar.



6. Kembali ke TForm, Klik masing-masing menu pilihan untuk warna BIRU dan Merah, kemudian isi dengan perintah-perintah seperti berikut ini.

```
procedure TForm1.Biru1Click(Sender: TObject);
begin
form1.Color :=clblue;
end;
```

```
procedure TForm1.Merah1Click(Sender: TObject);
begin
form1.Color :=clred;
end;
```

7. Jalankan program, dan warna TForm akan berubah sesuai dengan pilihan pada menu. Sebagai penanda juga dapat digunakan tanda check pada setiap menu yang sedang terpilih. Perintah yang ditambahkan adalah sebagai berikut.

```
procedure TForm1.Biru1Click(Sender: TObject);
begin
```

```

form1.Color :=clblue;
mainmenu1.Items[0].Items [0].Checked :=true;
mainmenu1.Items[0].Items [1].Checked :=False;
end;

procedure TForm1.Merah1Click(Sender: TObject);
begin
form1.Color :=clred;
mainmenu1.Items[0].Items [0].Checked :=false;
mainmenu1.Items[0].Items [1].Checked :=true;
end;

```

#### 1.4 Komponen TMemo

1. TMemo dapat digunakan untuk menampilkan teks yang lebih banyak dibandingkan TEdit. TMemo memiliki fasilitas scroll bar.

```

procedure TForm1.Button1Click(Sender: TObject);
var
kalimat:string;
begin
memo1.Lines.Clear ;
kalimat:='The !if directive works like C if statements. As shown here'+
'the syntax of !if and the other conditional directives resembles compiler';
memo1.Lines.Add(kalimat);
end;

```

2. TMemo dapat juga digunakan untuk menampilkan teks dalam satu rangkaian indeks, contoh untuk menampilkan 10 suku pertama dari persamaan deret:

$$1 + 2^2 + 3^2 + \dots + 10^2$$

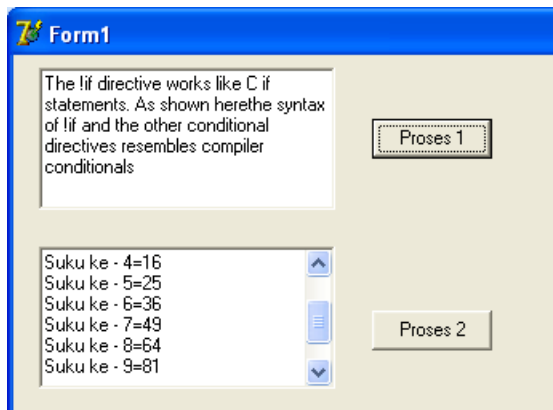
```

procedure TForm1.Button2Click(Sender: TObject);
var
i : integer;
begin
memo2.Lines.Clear ;
for i :=1 to 10 do
memo2.Lines.Add('Suku ke - '+inttostr(i)+'='
+inttostr(i*i));

```

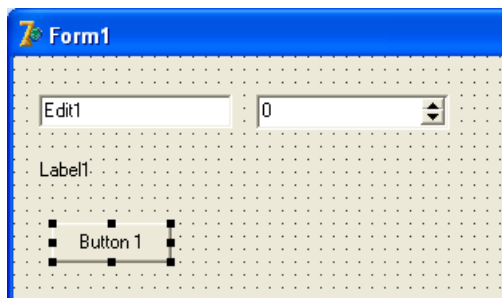
```
memo2.ScrollBars :=ssvertical;
```

```
end;
```



## 1.5 Fungsi Polinomial

1. Letakan 4 komponen pada TForm yaitu TEdit, TSpinEdit ( dari component pallet samples), TLabel dan Tbutton.



2. Program ini akan menyelesaikan suatu bentuk persamaan:

$$F(x) = x^n + x^{n-1} + x^{n-2} + \dots + x^3 + x^2 + x^1$$

3. Pertama kali deklarasikan variable global seperti berikut:

```
var
```

```
Form1: TForm1;
```

```
x,fx:real;
```

```
n : integer;
```

```
implementation
```

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

4. Kemudian untuk komponen TSpinEdit, isi nilai properties MinValue dengan nilai 1, dan pada event OnChange gunakan perintah berikut:

```
procedure TForm1.SpinEdit1Change(Sender: TObject);
```

```
begin
```

```
n:=spinedit1.Value ;
```

```
end;
```

5. Pada button tuliskan prosedur:

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

```
var
```

```
i: integer;
```

```
sum:real;
```

```
begin
```

```
x:=strtofloat (edit1.Text );
```

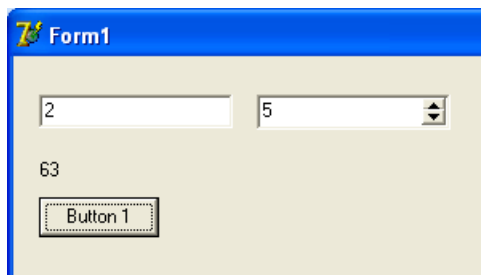
```
sum:=1;
```

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

```
    sum := sum * x + 1;
```

```
    label1.Caption :=floattostr(sum);
```

```
end;
```



## Latihan

1. Dengan menggunakan 2 buah komponen Button yang masing-masing button diberi caption **warna 1** dan **warna 2**, berilah 2 warna yang berbeda pada TForm, serta pada caption masing-masing button diubah ditambahkan tanda '&' didepan angka 1 dan 2 agar masing-masing caption pada button terdapat garis bawah pada karakter yang diisi tanda '&' ini digunakan untuk mengeksekusi button dengan menggunakan keyboard (Alt + 'karakter yang bergaris bawah').
2. Tambahkan komponen TTimer yang terdapat pada Tab System

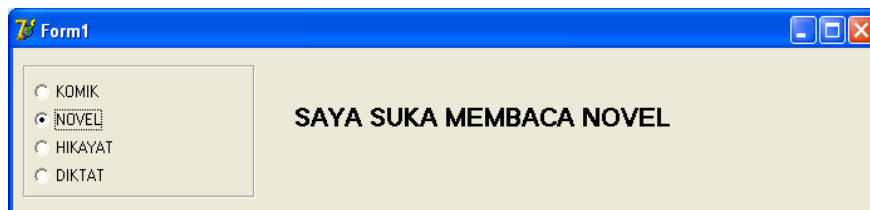




atur warna pada TForm agar mampu berubah dengan menggunakan TTimer sebagai pengatur durasi perubahan setiap warna, minimal menggunakan 4 buah warna, gunakan perintah pencabangan.

## Tugas

1. Buat Form baru, atur tampilannya agar menghasilkan interface sebagai berikut:



2. Buat algoritmanya dan kumpulkan beserta screenshootnya

## BAB 4. FILE TEKS & PESAN

### Capaian Pembelajaran dan Indikator

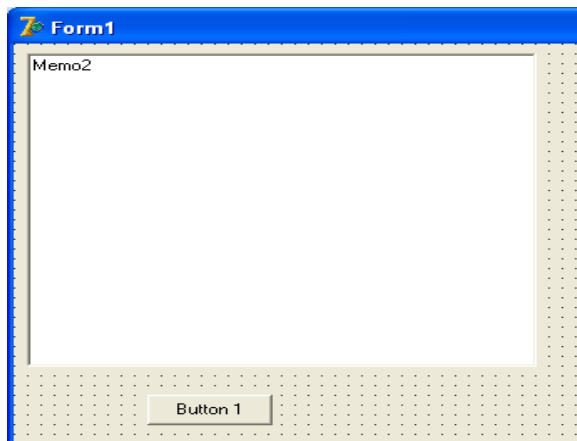
Capaian Pembelajaran : Dapat membuat program yang mampu membaca dan menulis pada suatu file teks serta dapat menampilkan pesan sesuai kebutuhan.

Indikator : - Mampu membuat program yang mampu membaca dan menulis pada suatu file teks.  
- Mampu memilih bentuk-bentuk pesan yang dapat digunakan dalam delphi

### Praktek

#### Membaca File Teks

1. Letakkan 2 komponen pada TForm yaitu TButton dan TMemo. Simpan sebuah file teks pada lokasi yang dapat diketahui, misalnya tersimpan pada :  
c::\coba.txt.



2. Deklarasikan variable globalnya:

var

Form1: TForm1;

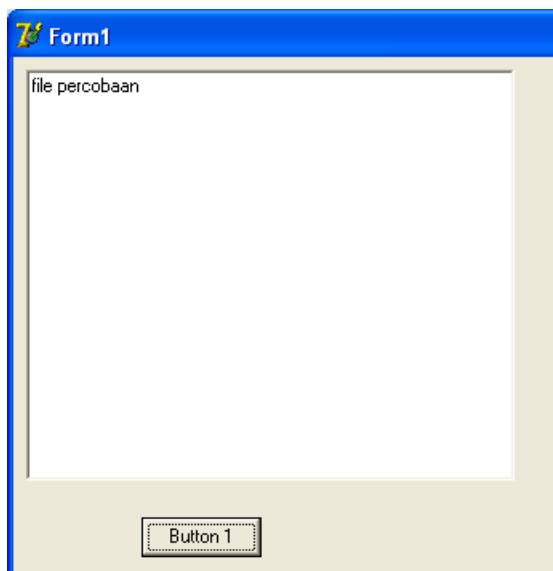
text1 : textfile;

bariskata : string;

implementation

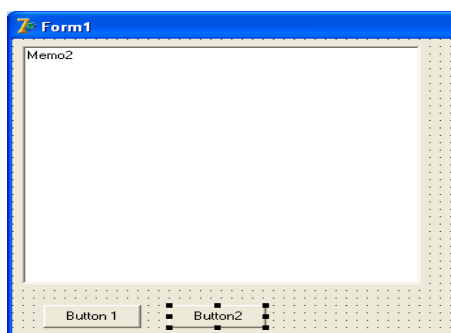
{ \$R \*.dfm }

3. Double klik TButton dan isi prosedurnya:
- ```
procedure TForm1.Button1Click(Sender: TObject);
begin
memo2.Lines.Clear ;
assignfile(text1,'c:\coba.txt');
reset(text1);
while not eof (text1) do
begin
readln(text1,bariskata);
memo2.Lines.Add(bariskata);
end;
closefile(text1);
end;
```



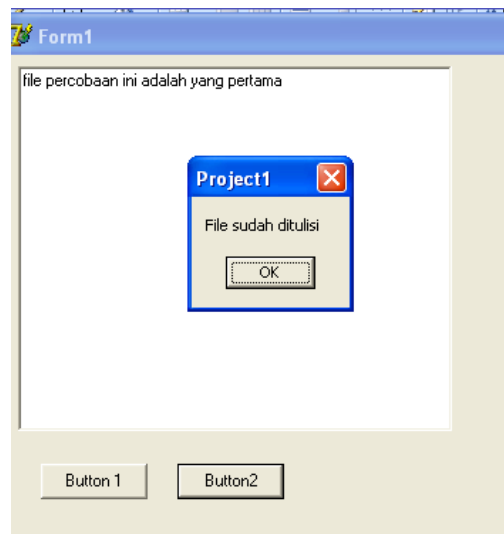
### Menulis Pada File Teks

1. Tambahkan satu komponen TButton pada TForm untuk menempatkan perintah penulisan pada file teks.



2. Isi prosedur TButton2:

```
procedure TForm1.Button2Click(Sender: TObject);  
var  
    i,jumlahbaris : integer;  
begin  
    jumlahbaris:=memo2.Lines.Count ;  
    assignfile(text1,'c:\coba.txt');  
    rewrite(text1);  
    for i :=0 to jumlahbaris-1 do  
    begin  
        bariskata:=memo2.Lines.Strings [i];  
        writeln(text1,bariskata);  
    end;  
    closefile(text1);  
    showmessage('File sudah ditulisi');  
end;
```



**Fungsi pesan/peringatan mempunyai empat parameter**

- Parameter pertama: kalimat yang akan dimunculkan(pesannya)
- Parameter kedua: tipe message box seperti mtWarning, mtError, mtInformation dan mtConfirmation.
- Parameter ketiga: kumpulan tombol yang akan digunakan seperti mbYes, mbNo, mbOK, mbCancel atau mbHelp.

- Parameter keempat: untuk help context atau nomor halaman pada Help, beri angka nol jika anda tidak mempunyai file help.

Contoh:

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  messagedlg('Hello, guys',mtinformation,[mbok],0);  
end;
```



### Latihan

Buat tipe message yang lain (mtWarning, mtError dan mtConfirmation) sertakan tombol-tombol yang digunakan (mbYes, mbNo, mbOK, mbCancel)

### Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya

## BAB 5. OLE


### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Mengetahui fungsi OLE dan memahami cara pengimplementasiannya

Indikator : - Mampu membuat Aplikasi yang melibatkan OLE

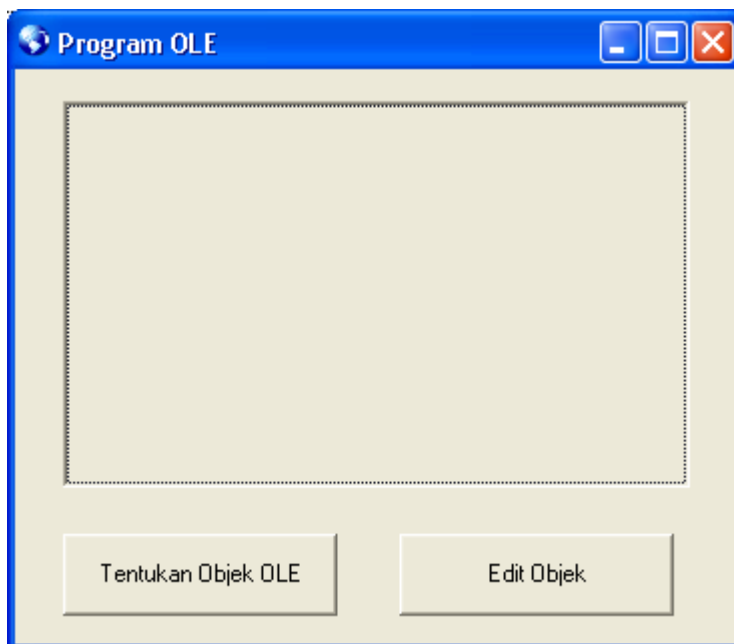
### Dasar Teori

OLE (*Object Linking and Embedding*) memungkinkan suatu aplikasi lain disisipkan pada aplikasi yang dibuat. Sebagai contoh, kita dapat meletakkan dokumen yang dibuat dengan menggunakan Microsoft Word, Excel ataupun gambar yang dibuat dengan Paint pada form. Selain aplikasi-aplikasi lain tersebut diletakkan pada form, juga kita dapat melakukan penyuntingan melalui form yang dibuat.

Delphi menyediakan OleContainer untuk mewujudkan OLE. Komponen ini berada pada tab System di jendela Component Pallet dengan ikon 

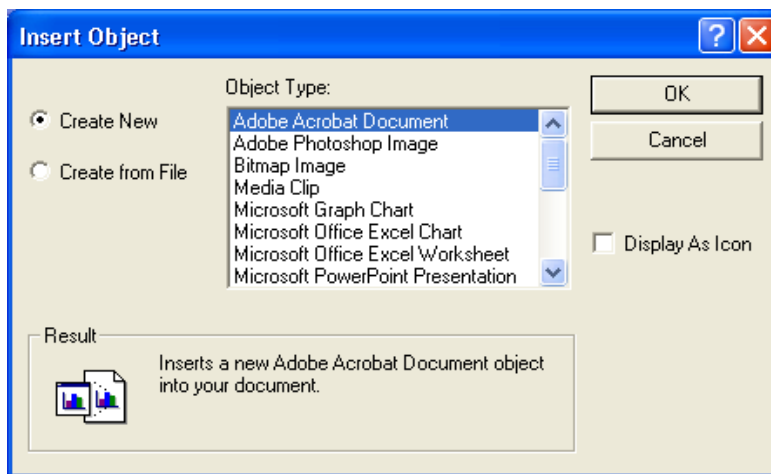
### Latihan

1. Siapkan project/proyek baru, ubah caption formnya menjadi : Program OLE
2. Buat Formnya sebagai berikut:



### 3. Tentukan Objek OLE

- a. Klik 2 kali pada Button Tentukan Objek OLE
- b. Kemudian tuliskan perintah sebagai berikut:  
procedure TForm1.bpilihClick(Sender: TObject);  
begin  
olecontainer1.InsertObjectDialog ; end;
- c. Kemudian uji coba programnya dengan me-RUN, klik pada Button Tentukan Objek OLE. Apakah sudah muncul jendela sebagai berikut?, kalau sudah lanjutkan langkah point 4 berikutnya !



### 4. Mengedit dan menutup OLE

- a. Klik 2 kali pada button Edit Objek
- b. Tuliskan prosedur berikut:

```
procedure TForm1.beditClick(Sender: TObject);  
begin  
//cek objek OLE sudah ada atau belum  
if olecontainer1.State =osempty then  
begin  
messagedlg('Objek OLE belum ditentukan',mtinformation,[mbok],0);  
exit;  
end;  
//membuat kombinasi caption tombol dan tindakan sesuai dengan caption  
if bedit.Caption = 'Edit Objek' then  
begin  
bedit.Caption :='Tutup Objek';
```

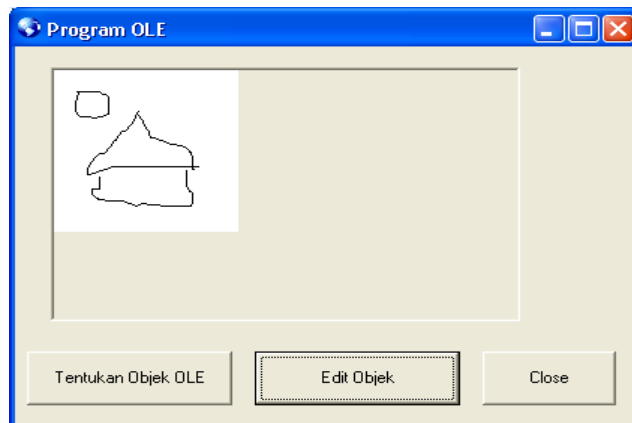
```

olecontainer1.DoVerb(ovprimary);
end
else
begin
bedit.Caption :='Edit Objek';
olecontainer1.Close ;
end;
end;

```

## 5. Menguji Program

a. Tambahkan sebuah Button untuk Close form, sebagai berikut:



b. Tuliskan prosedur untuk close formnya

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

```
begin
```

```
close;
```

```
end;
```

c. Jalankan program dengan me-RUN

d. Gunakan tombol Tentukan Objek untuk memilih Objek yang digunakan

e. Gunakan tombol Edit Objek untuk mengubah isi Objek yang dibuat

f. Tutup Objek digunakan untuk mengakhiri pengubahan data objek

## 6. Menyimpan dan Memuat Objek OLE

a. Tuliskan prosedur untuk pengecekan file awal ketika form create

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



```
begin
if fileexists('ole.tmp') then
olecontainer1.LoadFromFile('ole.tmp');
end;
```

b. Tuliskan prosedur untuk penyimpanan file baru ketika form close

```
procedure TForm1.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
if olecontainer1.State <>osempty then
olecontainer1.SaveToFile('ole.tmp');
end;
```

## Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya

## BAB 6. GRAFIK

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami dasar-dasar pemrograman grafik

Indikator : - Mampu menggambar piksel  
- Mampu membuat garis, mengatur bentuk garis, kurva, dll.

### Dasar Teori

Beberapa hal dasar yang perlu diketahui terkait pemrograman grafik, diantaranya:

- a. Kanvas
- b. Piksel
- c. Koordinat

Kanvas dapat dibayangkan seperti suatu lembaran kertas atau kain, yang pada lembaran tersebut kita dapat menggambar. Pada form kanvas diimplementasikan melalui property yang disebut **canvas**. Kanvas tersusun atas sejumlah piksel. Piksel adalah elemen terkecil pada kanvas yang dapat dimanipulasi.

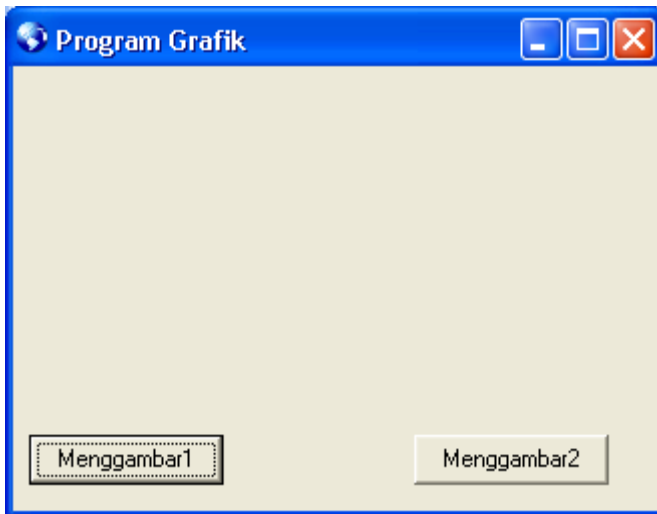
Setiap piksel mempunyai koordinat, yang dinyatakan dengan X,Y. X dapat dibayangkan sebagai kolom dan Y sebagai baris. Dengan demikian piksel pojok kiri atas memiliki koordinat (0,0) sedangkan pojok kanan bawah memiliki koordinat berupa (client width, client height).

Canvas memiliki property bernama Pen, atribut ini berguna untuk pengaturan color, style, width dll. Untuk property style pada pen terdapat beberapa macam bentuk garis yang dihasilkan, diantaranya dengan perintah: pssolid, psdash, psdot, psdashdot, psdashdotdot.

Canvas juga menyediakan property untuk menggambar kotak (rectangle), lingkaran (ellipse), busur (arc), potongan kue (pie), polygon/polyline dan kurva Bezier (polybezier)

## Latihan 1

1. Siapkan form baru, kemudian setting seperti form dibawah ini:



2. Klik 2 kali pada button yang ber caption menggambar1
3. Tuliskan prosedurnya, sbb:

```
procedure TForm2.bsatuClick(Sender: TObject);  
var x:integer;  
begin  
for x:=0 to clientwidth- 1 do  
canvas.Pixels [x,clientheight div 2]:=clred;  
end;
```

4. Klik 2 kali pada button yang ber caption menggambar2
5. Tuliskan prosedurnya, sbb:

```
procedure TForm2.bduaClick(Sender: TObject);  
var x,y,s,a:integer;  
r : real;  
begin  
a:= clientheight div 3;  
r:= clientheight div 2;  
for x:= 1 to clientwidth do  
begin
```

```

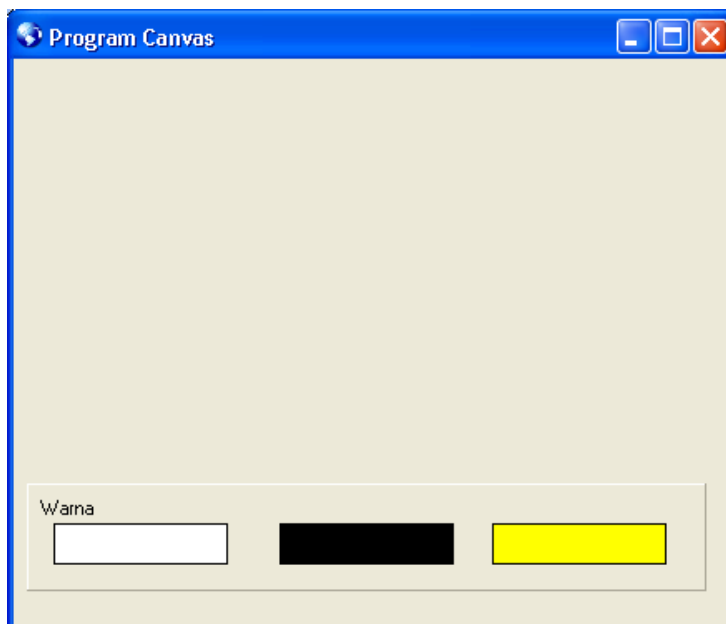
r:=x/clientwidth * 2*pi;
y:=s + trunc(a*sin(r));
canvas.Pixels [x,y]:=clgreen;
end;
end;

```

6. RUN programnya dan lihat hasilnya melalui klik button yang ada

## Latihan 2

1. Siapkan form baru, kemudian setting seperti form dibawah ini:



2. Siapkan sebuah panel, label dan 3 buah shape, yang masing-masing shape dibedakan color pada property brushnya.
3. Tuliskan perintah-perintah berikut untuk setiap prosedurnya

```

private
  { Private declarations }
  tomboltekan:boolean;
public
  { Public declarations }
end;

```

```

var

```

```
Form3: TForm3;
```

implementation

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

```
procedure TForm3.FormMouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  canvas.MoveTo(x,y);  
  tomboltekan:=true;  
end;
```

```
procedure TForm3.FormMouseUp(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  tomboltekan:=false;  
end;
```

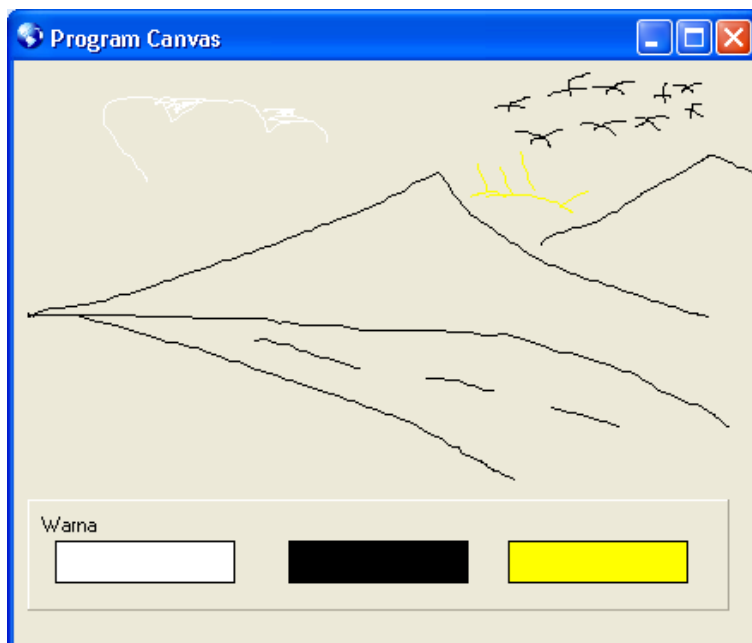
```
procedure TForm3.FormMouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
  if tomboltekan then  
  begin  
    canvas.LineTo(x,y);  
  end;  
end;
```

```
procedure TForm3.sputihMouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  canvas.pen.Color :=sputih.Brush.Color ;  
end;
```

```
procedure TForm3.shitamMouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  canvas.pen.Color :=shitam.Brush.Color ;  
end;
```

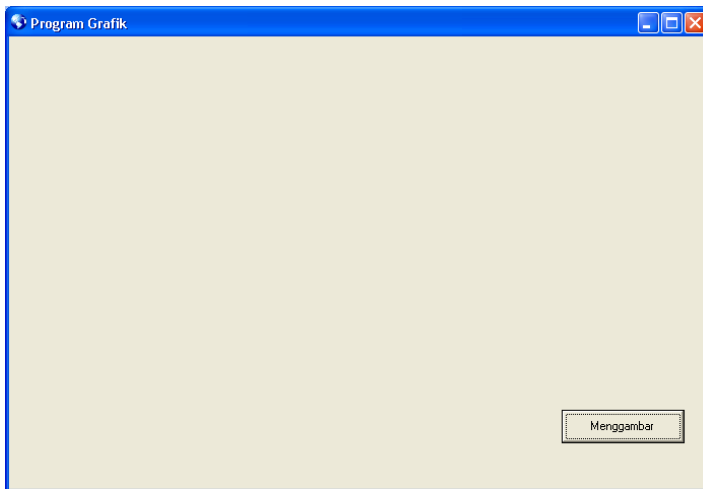
```
procedure TForm3.skuningMouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  canvas.pen.Color :=skuning.Brush.Color ;  
end;
```

4. RUN programnya, pilih salah satu warna dan gambarkan pada canvas



### Latihan 3

1. Siapkan sebuah form baru, kemudian setting form seperti dibawah ini:



2. Siapkan sebuah button yang akan digunakan sebagai tempat eksekusi prosedurnya.
3. Tuliskan prosedur berikut ketika button klik

```
procedure TForm4.bgambarClick(Sender: TObject);
begin
//menggambar kotak
canvas.Rectangle(10,10,20,20);
canvas.Rectangle(50,50,75,75);

//menggambar kotak dengan garis putus-putus
canvas.Pen.Style :=psdot;
canvas.Rectangle(100,30,150,80);

//menggambar kotak dengan bagian pinggir melengkung
canvas.Pen.Style :=pssolid;
canvas.RoundRect(125,110,200,250,20,20);

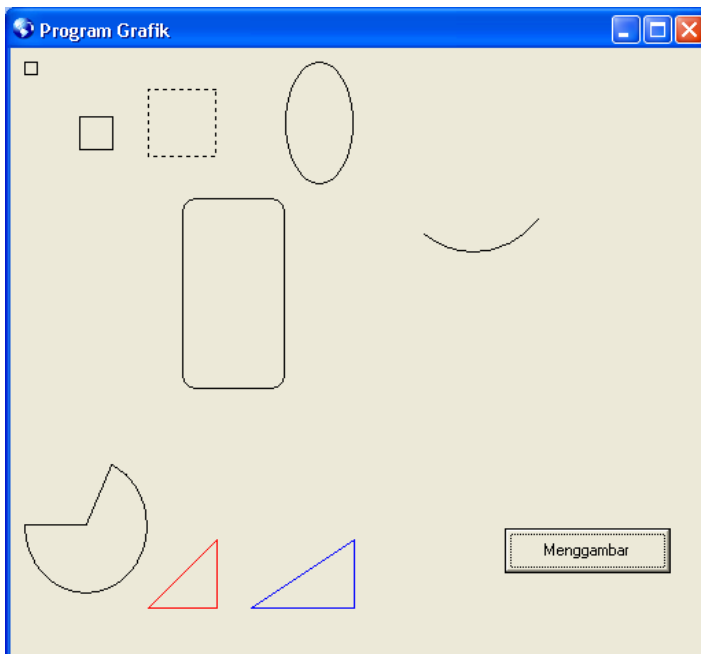
//menggambar lingkaran(ellips)
canvas.Ellipse(200,10,250,100);

//menggambar busur
canvas.Arc(275,10,400,150,260,200,390,130);
```

```
//menggambar potongan kue
canvas.Pie(10,300,100,400,15,350,150,125);

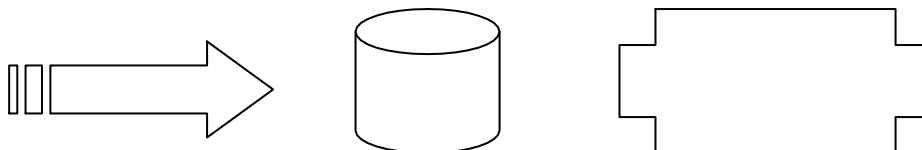
//menggambar poligon(polyline)
canvas.Pen.Color :=clred;
canvas.Polyline([point(100,410),point(150,360),point(150,410),point(100,410)]);
canvas.Pen.Color :=clblue;
canvas.Polygon([point(175,410),point(250,360),point(250,410),point(175,410)])
end;
```

4. RUN program dan lihat hasilnya dengan mengklik button



### Tugas

Buatlah gambar-gambar seperti berikut, serta buat laporan dari soal ini dan sertakan screenshotsnya





## BAB 7. TOOL BAR DAN STATUS BAR

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami dasar-dasar pembuatan tool bar dan status bar

Indikator : - Mampu membuat tool bar dan status bar  
- Mampu mengaplikasikannya dalam pembuatan program.


### Dasar Teori

Tool Bar (kadang-kala disebut speed bar atau control bar) seringkali dijumpai pada aplikasi di lingkungan Windows. Kegunaannya adalah untuk menyingkat pengaksesan terhadap tugas-tugas yang tersedia pada menu aplikasi.

Status bar merupakan baris yang mengandung keterangan status yang terjadi pada aplikasi. Seringkali aplikasi windows dilengkapi dengan baris status ini.


### Latihan 1

Siapkan sebuah project baru

Letakkan sebuah komponen cool bar  yang digunakan sebagai tempat dari tool bar (pada komponen pallet tab Win32)

Atur form sehingga berbentuk sebagai berikut:

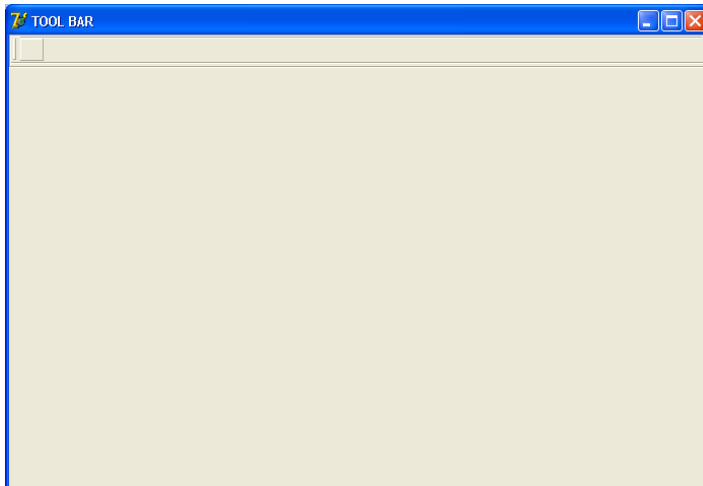


Biasanya komponen coolbar digunakan bersama dengan tool bar apabila jumlah tool bar lebih dari 1. Untuk memasukkan komponen tool bar pada form, klik ikon menu tool bar pada komponen pallet tab win32 , letakkan didalam komponen cool bar,

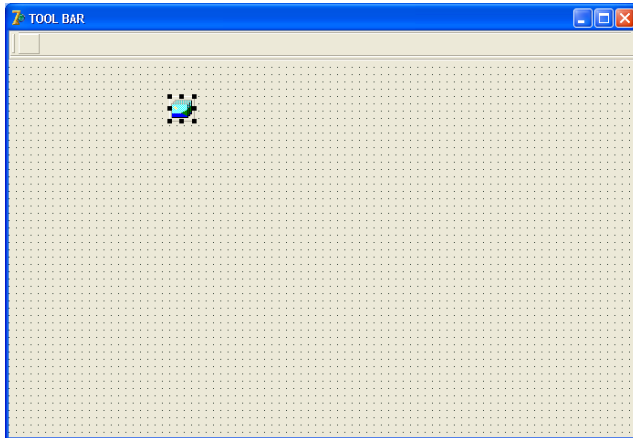
klik kanan mouse pilih new button pada saat menu pop-up muncul sehingga form berbentuk sebagai berikut:



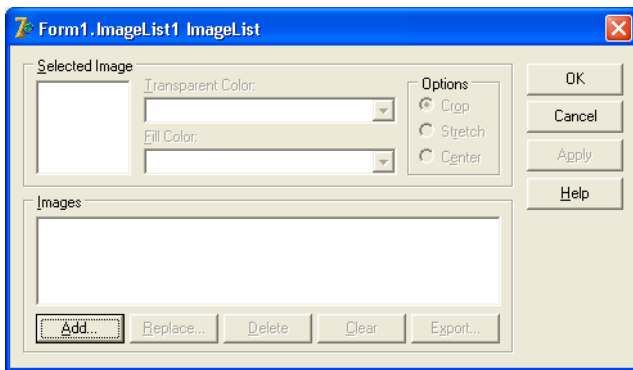
Agar ketika dijalankan tombol tidak menonjol maka ubahlah property flat tool bar menjadi true (hal ini akan menyebabkan tombol akan terlihat hanya ketika penunjuk mouse diarahkan pada button tersebut)



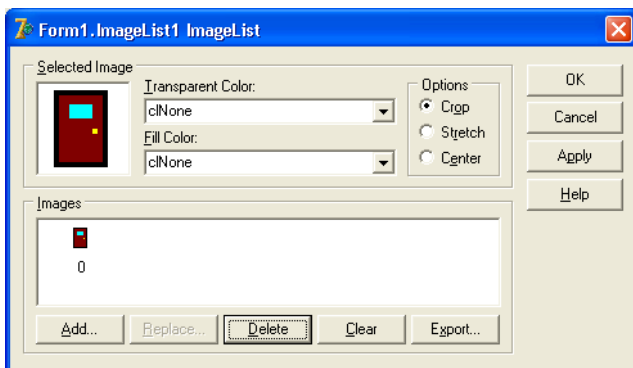
Tool bar biasanya berisi gambar, yang menggambarkan proses apa yang akan dilakukan ketika button tersebut di klik, oleh karena itu untuk memasukkan gambar maka kita menggunakan komponen image list pada tab Win32 (🖼️), klik komponen tersebut dan letakkan pada sembarang dibidang form.



Klik ganda komponen image list tersebut, dan akan muncul kotak dialog baru sebagai berikut:



Klik tombol Add, pilih direktori C:\Prpgram Files\Common Files\Borland Shared\Images\Buttons dan kemudian pilih berkas doorshut sebagaimana terlihat dibawah ini:



Agar hanya terdapat 1 buah gambar doorshut maka pilih gambar doorshut yang tidak digunakan kemudian tekan tolbol Delete, untuk menambahkan gambar lain maka tekan tombol Add kemudian tambahkan gambar lainnya. Dan kalau sudah selesai klik tombol OK

Agar gambar-gambar tersebut dapat digunakan tool bar, maka pilih dan klik komponen tool bar yang mengandung button, kemudian pada property images pilih imagelist1.

Untuk mengatur gambar maka ubah index gambar di property imageindex pada toolbuttonnya.

Agar toolbutton dapat dieksekusi maka berikan perintah pada tool button sebagai berikut:


```
procedure TForm1.ToolButton1Click(Sender: TObject);  
begin  
close;  
end;
```

Agar tool bar dapat dipindah atur property dragkind isi dkDock dan dragmode isi dmAutomatic.

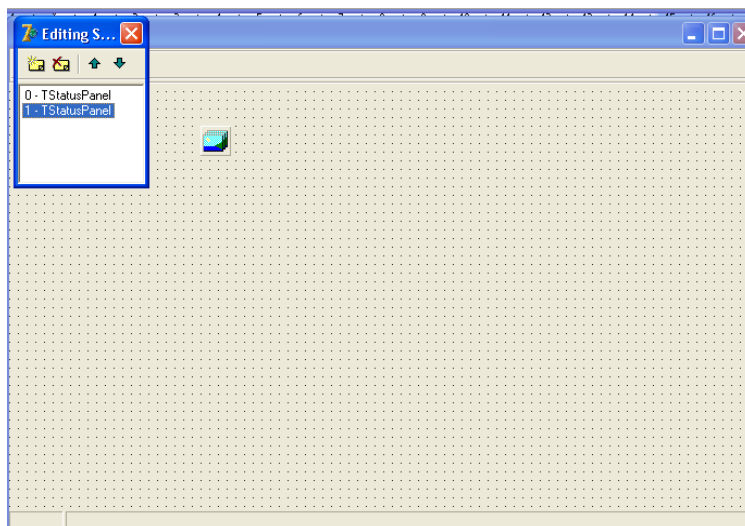
Agar tool bar dapat diletakkan lagi pada posisinya, ubah property cool bar pada Docksitenya menjadi True

## Latihan 2

Untuk membuat status bar, buka form yang mengandung tool bar diatas.

Klik ikon status bar pada tab win32  letakkan pada form (klik sembarang pada form)

Untuk membagi area status bar, klik ganda pada komponen status bar, sehingga dimunculkan layar sebagai berikut:



Untuk mengatur lebar setiap bagian, klik ganda pada status bar, pilih TstatusPanel yang akan diubah ukurannya. Pada property Width diubah nilainya.

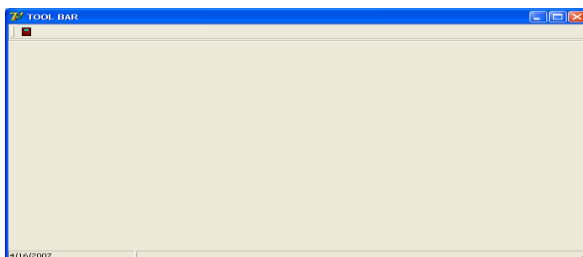
Tambahkan prosedur pada programnya sebagai berikut:

```
procedure TForm1.FormActivate(Sender: TObject);
begin
statusbar1.Panels [0].Text :=datetostr(date);
end;
```

```
procedure TForm1.FormMouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
statusbar1.Panels [1].Text := 'Mouse Sedang ditekan';
end;
```

```
procedure TForm1.FormMouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
statusbar1.Panels [1].Text := "";
end;
```

Sehingga akan menghasilkan form sebagai berikut:



## Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya


## BAB 8. TREEVIEW DAN LISTVIEW


### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami dasar-dasar pembuatan treeview dan listview

Indikator : - Mampu membuat treeview dan listview  
- Mampu mengaplikasikannya dalam pembuatan program.

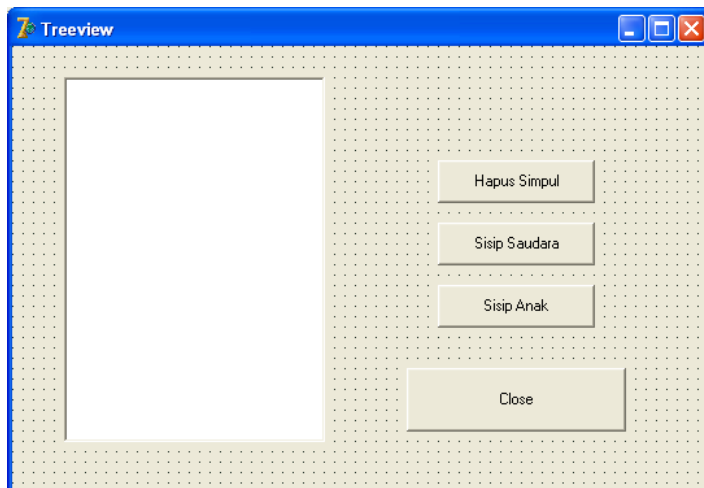
### Dasar Teori

Treeview adalah komponen berbentuk jendela yang berguna untuk menampilkan daftar item secara hirarkis, misalnya berupa daftar berkas dan direktori. Treeview terletak pada tab win32 dengan gambar sebagai berikut : .

Komponen listview berguna untuk menampilkan daftar item dengan berbagai model. Masing-masing item dapat diatur dan dilengkapi dengan teks atau gambar. Terletak pada tab win32 komponen pallet dengan gambar sebagai berikut: .

### Latihan 1

Siapkan sebuah form baru dan format form sebagai berikut:



Tuliskan prosedur untuk form diatas, sebagai berikut:

```
private
    { Private declarations }
    posisi:integer;
public
    { Public declarations }
```

```

    end;
var
    Form1: TForm1;
implementation

{$R *.dfm}
procedure TForm1.FormCreate(Sender: TObject);
var jumsimpul,indeks:integer;
begin
    jumsimpul:=treeview1.Items.Count ;
    for indeks:=0 to jumsimpul-1 do
    begin
        treeview1.Items [indeks].SelectedIndex :=treeview1.Items[indeks].Level ;
    end;
end;

procedure TForm1.TreeView1Change(Sender: TObject; Node: TTreeNode);
begin
    posisi:=node.AbsoluteIndex ;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
    if treeview1.Items.Count =0 then
    begin
        messagedlg('simpul habis',mtInformation,[mbok],0);
    exit;
    end;
    treeview1.Items [posisi].Delete ;
    treeview1.SetFocus ;
end;

procedure TForm1.Button2Click(Sender: TObject);
var simpul:treenode;

```

```
begin
if treeview1.Items.Count =0 then
simpul:=nil
else
simpul:=treeview1.Items [posisi];
treeview1.Items.Insert(simpul,'Baru...');
treeview1.SetFocus ;
end;
```

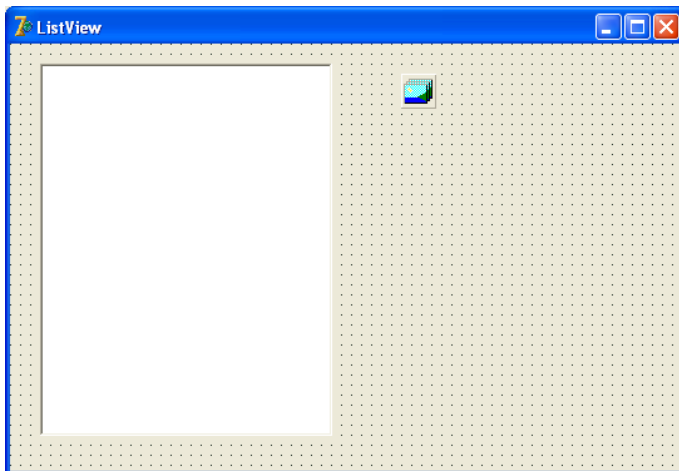
```
procedure TForm1.Button3Click(Sender: TObject);
var simpul:ttreenode;
begin
if treeview1.Items.Count = 0 then
simpul :=nil
else
simpul:=treeview1.Items[posisi];
treeview1.Items.AddChild(simpul,'Baru..');
treeview1.SetFocus ;
end;
```

```
procedure TForm1.Button4Click(Sender: TObject);
begin
close;
end;
```

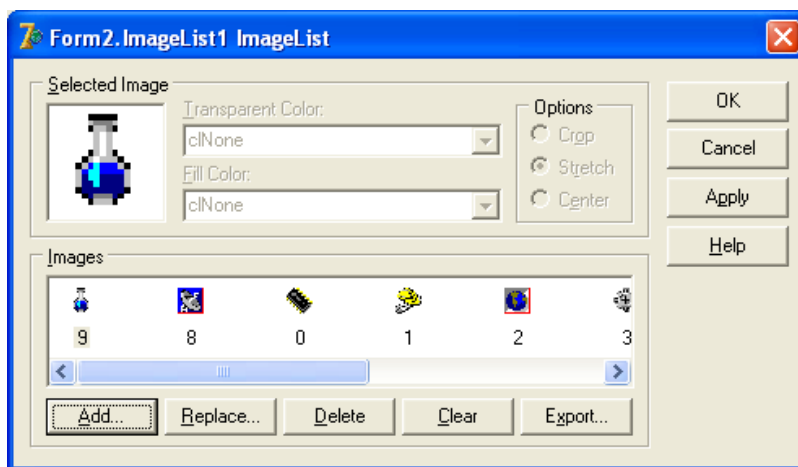
## Latihan 2

Siapkan sebuah form baru dengan format sebagai berikut:





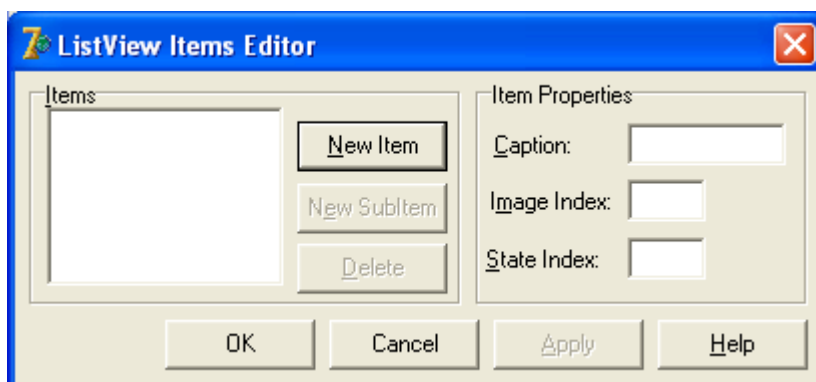
Klik ganda pada Imagelist sehingga muncul layar baru, sbb:



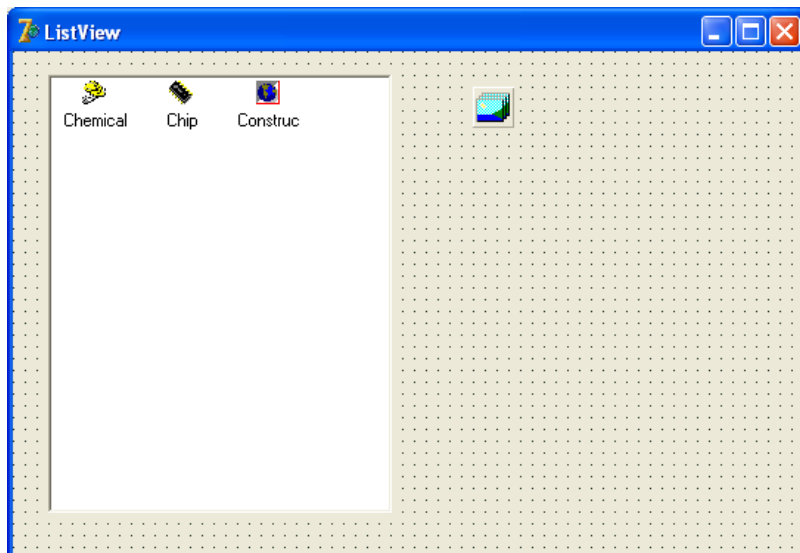
Untuk memunculkan gambar ikon seperti gambar diatas, klik pada button Add kemudian pilih direktori C:\Program Files\Common Files\Borland Shared\Images\Icons, setelah ikon yang dibutuhkan dimasukkan, klik tombol OK.

Pada komponen ListView pilih properti LargeImage, isikan dengan ImageList1

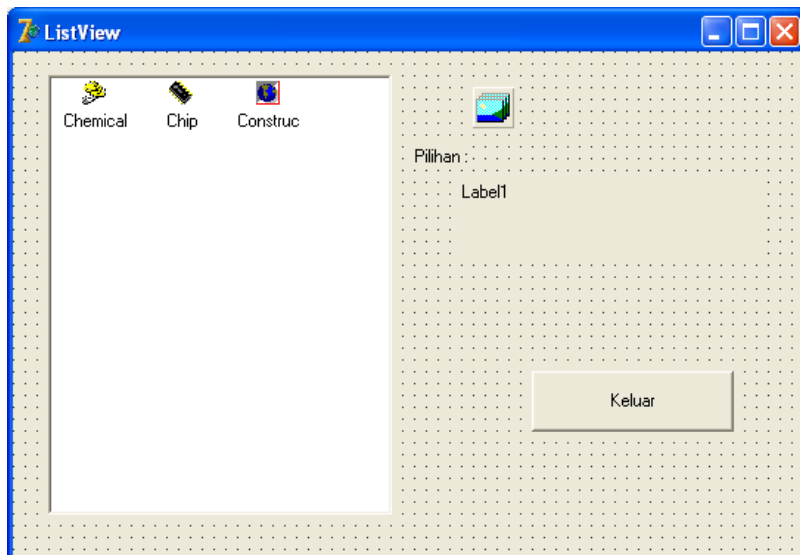
Klik kanan Pada ListView, pilih ItemEditor, sehingga muncul jendela baru, sbb:



Klik New Item, kemudian isi Caption dan Image Index pada Item Properties. Sesuaikan isian caption dan nomor index yang mengikuti list image-nya. Setelah selesai klik OK, sehingga akan menjadi sebagai berikut:



Tambahkan 1 buah button pada form dan 2 buah label, sehingga form menjadi sebagai berikut:



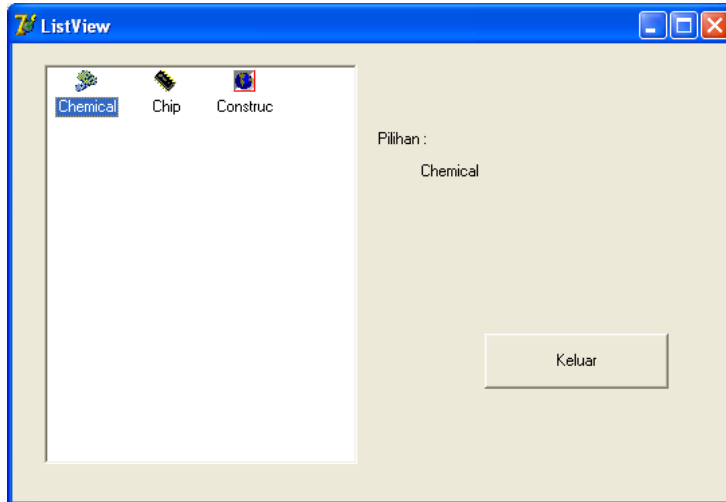
Tuliskan Prosedur berikut pada masing-masing event komponen :

```
procedure TForm2.Button1Click(Sender: TObject);  
begin  
close;  
end;
```

```
procedure TForm2.ListView1Change(Sender: TObject; Item: TListItem;  
Change: TItemChange);
```

```
begin  
label1.Caption :=item.Caption ;  
end;
```

Apabila program di Run, Akan menghasilkan tampilan sebagai berikut:



## Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya

## BAB 9. DATABASE DESKTOP

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami cara membuat file table melalui fasilitas Database Desktop

Indikator : - Mampu menggunakan komponen yang digunakan untuk mengakses database/tabel

### Dasar Teori

Definisi *database* sangatlah bervariasi. *Database* dapat dianggap sebagai tempat untuk sekumpulan berkas data terkomputerisasi.

implementasi *database* dapat dibagi menjadi 2 :

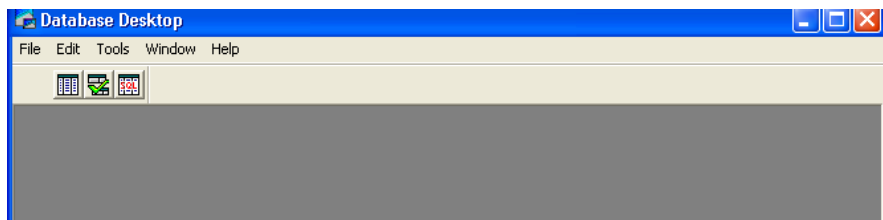
1. sebuah database disimpan pada sebuah direktori dan **terdiri atas sejumlah berkas table.**
2. sebuah database dinyatakan dengan sebuah berkas.

Dalam terminologi *database* yang bersifat relasional, suatu database tersusun atas sejumlah tabel.

### Latihan

#### Langkah Langkah Mebuat Database Desktop

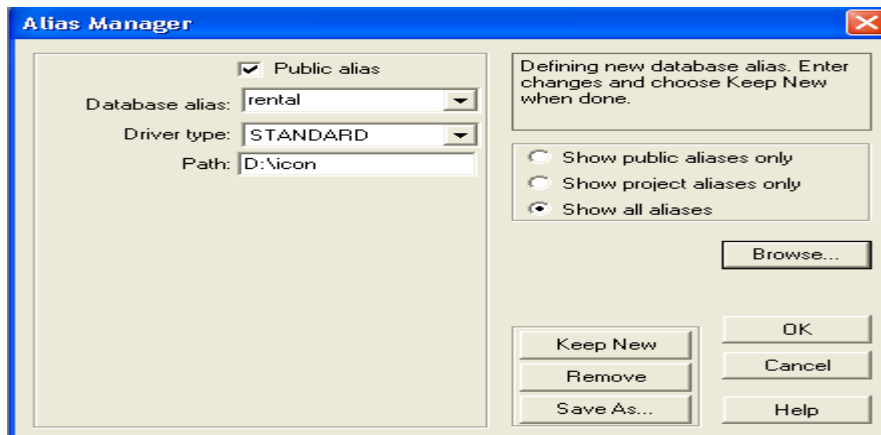
Masuk keprogram Delphi, pilih menu tools kemudian pilih database desktop akan muncul tampilan seperti berikut:



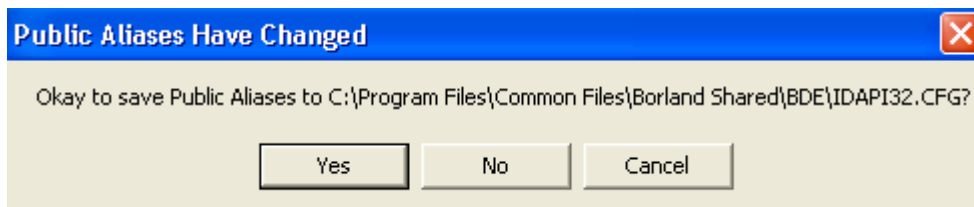
sekarang kita berada pada tampilan database destop ,sebelum anda membuat tabel anda terlebih dahulu harus membuat nama database.

## Membuat alias

pada database desktop pilih menu tools lalu pilih alias manager, klik **new**



sekarang anda harus memberi nama database alias **rental** kemudian tentukan path yaitu lokasi dimana akan menyimpan database alias lalu **ok** lalu **ok** lagi maka akan muncul kotak konfirmasi seperti berikut

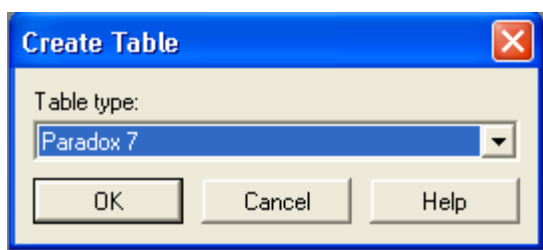


tekan **yes** bila setuju akan menyimpan databasenya

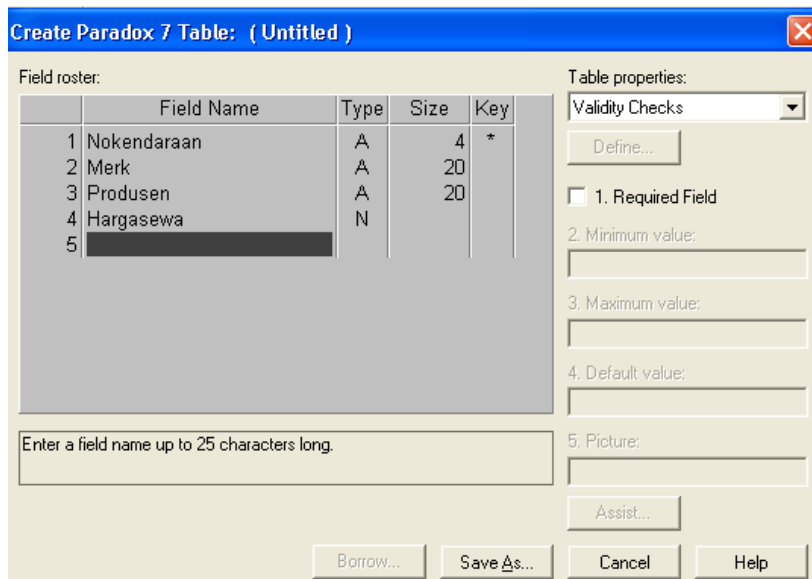
## Membuat Tabel

caranya (masih berada dalam tampilan database desktop)

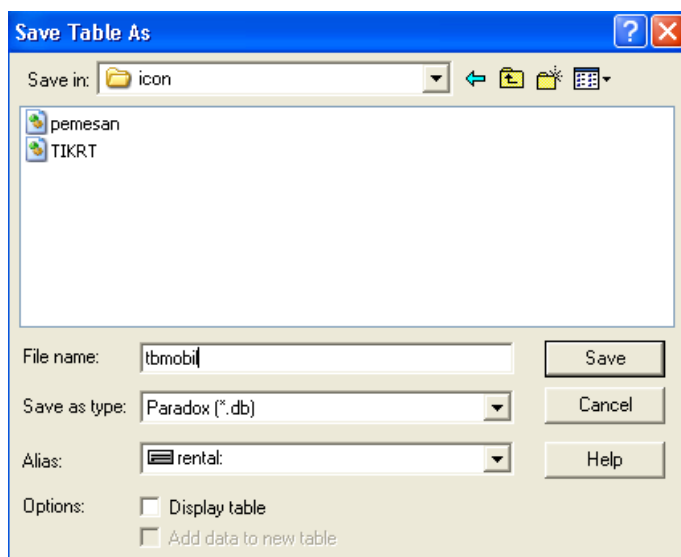
pilih menu file\new\ table maka akan muncul tampilan sebagai berikut



karena kita menggunakan database desktop pilihlah paradox 7 lalu tekan ok maka akan masuk dalam sebuah form untuk membuat tabel seperti berikut



sebagai contoh anda masukan seperti tampilan di atas setelah itu tekan **save as** akan muncul tampilan seperti berikut



sekarang anda cari nama alias yang anda buat sebagai contoh anda cari alias dengan nama rental kemudia filename harus anda isi contoh diatas dengan nama **tbmobil** setelah itu anda **save**

setelah anda membuat tabel tentunya anda harus mengisi datanya cara mengisi tabel pada database desktop anda

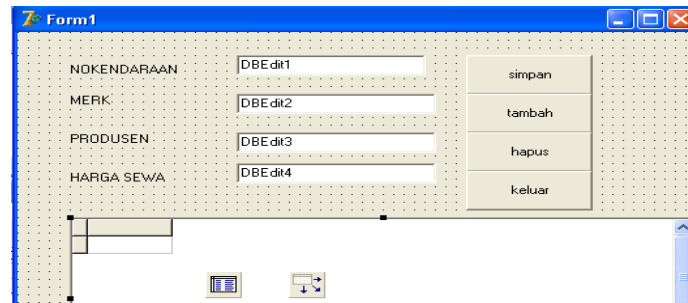
pilih menu file /open/table anda cari nama **aliasnya** kemudian pilih tabel apa yang anda ingin buka lalu **open** sekarang saatnya mengisi datanya pada database destop pilih menu **table** lalu **editdata** isi data sebagai berikut :

| tbmobil | Nokendaraan | Merk   | Produsen | Hargasewa  |
|---------|-------------|--------|----------|------------|
| 1       | B001        | ESCUDO | SUZUKI   | 500.000,00 |
| 2       | B002        | BALENO | SUZUKI   | 700.000,00 |

sekarang tutup database desktop anda sekarang kita akan mencoba untuk membuat aplikasinya menggunakan Delphi

### Komponen yang anda perlukan

| Komponen    | Properties   | Nilai       |
|-------------|--------------|-------------|
| Label1      | Caption      | Nokendaraan |
| Label2      | Caption      | Merk        |
| Label3      | Caption      | Produsen    |
| Label4      | Caption      | Hargasewa   |
| Dbedit1     | Datasource   | Datasource1 |
|             | Datafield    | Nokendaraan |
| Dbedit2     | Datasource   | Datasource1 |
|             | Datafield    | Merk        |
| Dbedit3     | Datasource   | Datasource1 |
|             | Datafield    | Produsen    |
| Dbedit4     | Datasource   | Datasource1 |
|             | Datafield    | Hargasewa   |
| Table1      | Databasename | Rental      |
|             | Tablename    | Tbmobil     |
|             | Active       | True        |
| Datasource1 | Dataset      | Table1      |
| Dbgrid1     | Datasource   | Datasource1 |
| Button1     | Caption      | Simpan      |
| Button2     | Caption      | Tambah      |
| Button3     | Caption      | Hapus       |
| Button4     | Caption      | Keluar      |



```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
table1.Post;  
end;  
procedure TForm1.Button2Click(Sender: TObject);  
begin  
table1.Append;  
end;  
procedure TForm1.Button3Click(Sender: TObject);  
begin  
table1.Delete;  
end;  
procedure TForm1.Button4Click(Sender: TObject);  
begin  
close;  
end;
```

## Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya



## BAB 10. DATABASE ACCESS DENGAN DELPHI & PENCARIAN DATA

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami tentang Database Access dengan Delphi

Indikator : - Mampu membuat koneksi antara database Access dengan Delphi  
- Mampu membuat aplikasi untuk pencarian data

### Dasar Teori

#### 1. Mekanisme Pengaksesan Database

Pada dasarnya database dapat dibedakan menjadi dua macam :

##### 1. Local database.

**Berada** pada drive local atau sebuah jaringan local ( Local Area Network ). Yang termasuk pada local database di antaranya adalah *Paradox, dBase, FoxPro, dan Access*.

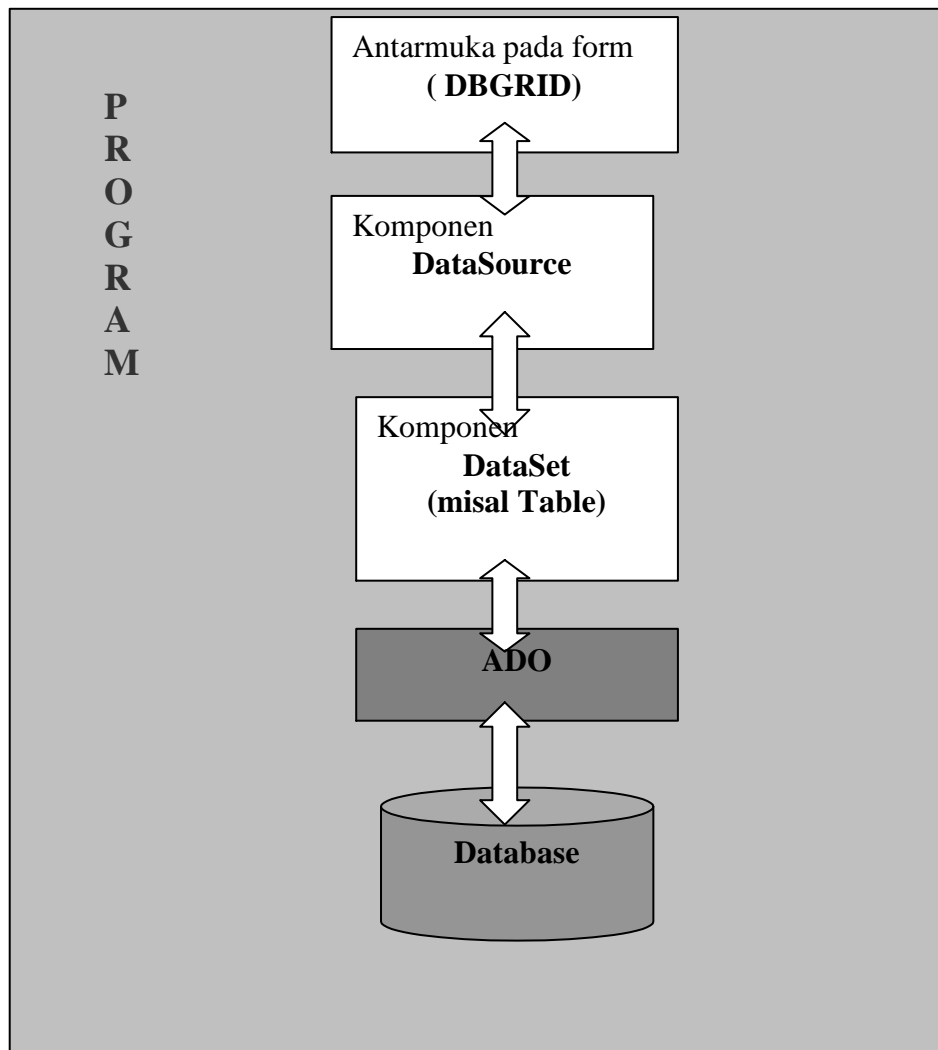
##### 2. Remote database servers

Dapat di golongan pada *two-tiered applications* atau *multi-tiered application*, karena aplikasi terletak pada system (tiers) yang berbeda. Yang termasuk di antaranya adalah *InterBase, Oracle, Sybase, Microsoft SQL Serve, dan DB2*.

Delphi menyatakan banyak komponen untuk membuat aplikasi database. Komponen-komponen tersebut diantaranya :

- a. Page BDE.
- b. Page ADO.
- c. Page Interbase.
- d. Page dbExpres.
- e. Page DataAccess

Sebenarnya terdapat beberapa mekanisme untuk mengakses *database*. Salah satu diantaranya adalah melalui **ADO**, yang dapat digunakan untuk mengakses **database MS Access**.

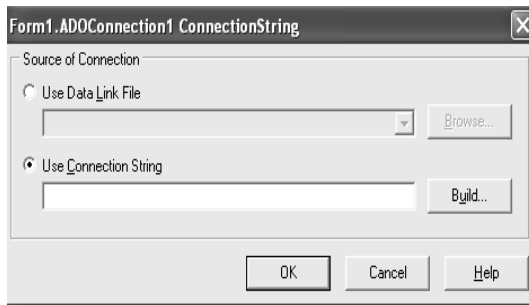


**Gambar** Mekanisme Pengaksesan database melalui ADO

## 2. Memahami Komponen-Komponen Database dalam Delphi

Untuk lebih memahami penggunaan komponen TADOConnection dan TADOTable coba **latihan** berikut :

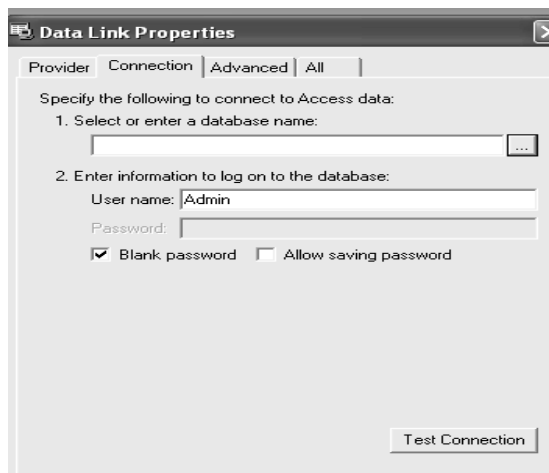
- Buat sebuah objek baru dan tambahkan **TADOConnectiaon** pada form. **Klik tombol elipsis** yang terdapat pada property **ConnectionString**.
- Pada kotak dialog **Data Link Properties**, pilih **Use Connection String** dan klik pada tombol **Build**.



- c. Pada tab **Provider** terdapat beberapa provider . pilih tab **Microsoft Jet 4.0 OLE DB Provider** klik **Next**.



- d. Pada **Tab Connection**, klik tombol **ellipsis button** ( tombol yang mengarahkan ketempat penyimpanan *database* access ), dan klik tombol **Test Connection**.



- e. Jika sudah terconnecti klik **tombol OK**.
- f. Tambahkan komponen **TADOTable**. Ubah **property Connection** menjadi **ADOConnection1**. ubah **property TableName** menjadi **table** yang

diinginkan, dan aktifkan tabel dengan mengubah property **Active menjadi True**.

- g. Tambahkan komponen **TDataSource** dan ubah property **DataSet** menjadi **ADOTable1**.
- h. Tambahkan komponen **TDBGrid** dan **TDBNavigator** dan ubah property **DataSource** dari dua komponen tersebut menjadi **DataSource1**.

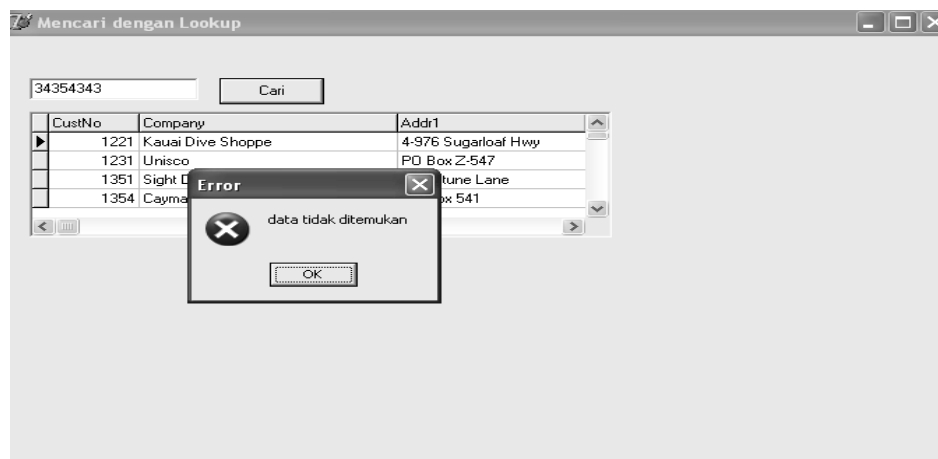
### 3. Pencarian Data

Mencari data sangat di perlukan dalam pembuatan database. Maka di dalam penggunaan **ADO dikenal 3 model pencarian** data antara lain menggunakan **method Locate, Lookup, dan seek**.

#### Latihan

Buat aplikasi untuk pencarian dengan menggunakan **Lookup**. **lookup** dipakai untuk mengambil nilai dari field yang sesuai dengan kriteria pencarian, contoh, seperti dibawah ini

1. Buat sebuah proyek baru dalam Delphi dan tambahkan pada form, komponen **TADOTable, TDataSource, TDBGrid, Tedit, dan TButton**.
2. Ubah beberapa property dari masing-masing komponen. Database yang di gunakan yaitu **dbdemos.mdb**. sedangkan tabelnya adalah **customer**.
3. Pada aplikasi ini jika user mengklik **tombol cari** maka akan dilakukan **proses pencarian**. **Ditampilkan kotak dialog** . jika pencarian gagal, akan muncul pesan kesalahan.



| KOMPONEN    | PROPERTY         | NILAI                                                                                                                 |
|-------------|------------------|-----------------------------------------------------------------------------------------------------------------------|
| TADOTable   | ConnectionString | Provider=Microsoft.jet.OLEDB.4.0<br>;Data Source= C:\Program<br>Files\Common Files\Borland<br>Shared\Data\dbdemos.mdb |
|             | Tablename        | Customer                                                                                                              |
|             | Active           | True                                                                                                                  |
| TdataSource | DataSet          | ADOTable1                                                                                                             |
| TDBGrid     | DataSource       | DataSource1                                                                                                           |
| TEdit       | Name             | EditCari                                                                                                              |
| TButton     | Caption          | Cari                                                                                                                  |

**Listing program sebagai berikut :**

```

procedure TForm1.Button1Click(Sender: TObject);
var
hasilpencarian:variant;
begin
if Trim(editcari.Text)<>"then
begin
hasilpencarian :=ADOTable1.lookup('custno',editcari.Text,'company');
if not varIsNull (hasilPencarian) then
ShowMessage(varToStr(hasilpencarian))
else
MessageDlg('data tidak ditemukan',mterror, [mbOK], 0);
end
// editcari masih kosong maka letakkan fokus pada editcari
else
editcari.setfocus;
end;

```

**Tugas**

1. Aplikasikan program pencarian dengan menggunakan method Locate dan seek
2. Buat laporan dengan menyertakan screenshootnya dari soal nomor 1

## BAB 11. NAVIGASI RECORD


### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami tentang Navigasi untuk Record

Indikator : - Mampu membuat navigasi dengan dbnavigator  
- Mampu membuat sendiri tombol navigator

### Dasar Teori

**Navigasi Record** digunakan untuk mengelola record dalam tabel, diantaranya berpindah kerecord paling awal, berpindah kerecord terakhir, maju persatu record, mundur persatu record.

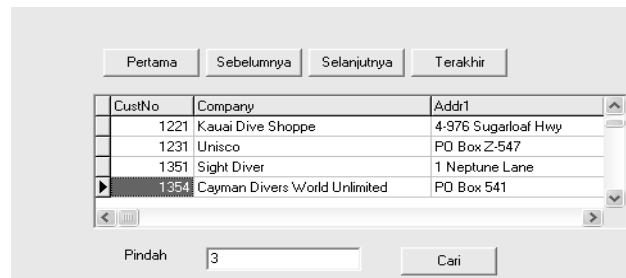
Dalam Delphi sudah tersedia komponen untuk navigator record seperti gambar ini . Selain komponen navigator yang sudah disediakan oleh Delphi, programmer juga dapat membuat komponen navigasi sendiri, oleh karena itu dalam praktek disampaikan tentang tentang cara pembuatan tombol navigasi.

### Latihan

Siapkan 2 form. Form pertama digunakan untuk tombol navigasi seperti praktek 1 dengan menggunakan database Paradox dan komponen TTable. dan form ke 2 digunakan untuk tombol simpan, batal, baru, tutup dengan database Access dan komponen ADO.

#### 1. Tombol Navigasi

- a. Buat form baru
- b. Tambahkan komponen **TTable**, ubah property **TDataBase** name menjadi **DBDemos**, dan property **TableName** menjadi **Country.db**, aktifkan tabel dengan mengubah property **Active** menjadi **True**.
- c. Tambahkan komponen **TDataSource**, ubah property **DataSet** menjadi **Table1**.
- d. Tambahkan komponen **TDBGrid**, ubah property **DataSource** menjadi **DataSource1**.



- e. Tambahkan lima buah tombol **TButton** dan **satu buah TEdit**, kemudian susun tampilannya.
- f. Lengkapi kode program sebagai berikut :

```

procedure TForm1.Button1Click(Sender: TObject);
begin
    Table1.First;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
    Table1.Prior;
end;
procedure TForm1.Button3Click(Sender: TObject);
begin
    Table1.Next;
end;
procedure TForm1.Button4Click(Sender: TObject);
begin
    Table1.Last;
end;
procedure TForm1.Button5Click(Sender: TObject);
begin
    Table1.MoveBy(StrToInt(Edit1.Text));
end;

```

## 2. Simpan, Batal, Baru, Tutup.

Mula-mula buatlah form baru dengan tampilan seperti berikut :

| KODE_BARANG | NAMA_BARANG           | SATUAN | BANYAK |
|-------------|-----------------------|--------|--------|
| A1A01       | MONITOR SAMSUNG 14 IN | UNIT   | 7      |
| A2B01       | MONITOR YUNDAI 14 INC | UNIT   | 2      |
| A2B02       | MONITOR YUNDAI 15 INC | UNIT   | 4      |
| a1b01       | ioriowiwo             | unit   | 5      |

Pedoman dalam pembuatan form dapat dilihat dibawah ini :

| KOMPONEN     | PROPERTY         | NILAI                                                                             |
|--------------|------------------|-----------------------------------------------------------------------------------|
| Form         | Caption          | Form Data Barang                                                                  |
| Group Box    | Caption          | Input Data Barang                                                                 |
| Label1       | Caption          | Kode Barang                                                                       |
| Label2       | Caption          | Nama Barang                                                                       |
| Label3       | Caption          | Satuan                                                                            |
| Label4       | Caption          | Banyak                                                                            |
| Edit1        | Name             | Edit1                                                                             |
| Edit2        | Name             | Edit2                                                                             |
| Edit3        | Name             | Edit3                                                                             |
| Combobox1    | Name             | Combobox1                                                                         |
| Speedbutton1 | Caption          | Simpan                                                                            |
| Speedbutton2 | Caption          | Batal                                                                             |
| Speedbutton3 | Caption          | Baru                                                                              |
| Speedbutton4 | Caption          | Tutup                                                                             |
| TADOTable    | ConnectionString | Isi alamat database yang akan di hubunngkan. Seperti contoh pertemuan sebelumnya. |
|              | Tablename        | Tentukan Nama tabelnya                                                            |



|             |            |             |
|-------------|------------|-------------|
|             | Active     | True        |
| TdataSource | DataSet    | ADOTable1   |
| TDBGrid     | DataSource | DataSource1 |

**Listing program :**

**TOMBOL SIMPAN**

```

procedure TForm1.SpeedButton1Click(Sender: TObject);
begin
if (edit1.Text<>")and (edit2.Text<>") and
(edit3.Text<>")and(combobox1.Text<>")then
begin
with ADOTable1 do
begin
ADOTable1.Append;
ADOTable1['KODE_BARANG']:=edit1.Text;
ADOTable1['NAMA_BARANG']:=edit2.Text;
ADOTable1['BANYAK']:=edit3.Text;
ADOTable1['SATUAN']:=combobox1.Text;
ADOTable1.Post;
end;
end
else
showmessage('data harus diisi');
end;

```

**TOMBOL BATAL**

```

procedure TForm1.SpeedButton2Click(Sender: TObject);
begin
edit1.Text:="";
edit2.Text:="";
edit3.Text:="";
combobox1.Text:="";
ADOTable1.Cancel;
edit1.SetFocus;
end;

```

### **TOMBOL BARU**

```
procedure TForm1.SpeedButton3Click(Sender: TObject);
begin
edit1.Text:="";
edit2.Text:="";
edit3.Text:="";
combobox1.Text:="";
edit1.Enabled:=true;
edit2.Enabled:=false;
edit1.SetFocus;
speedbutton1.Enabled:=true;
end;
```

### **TOMBOL TUTUP**

```
procedure TForm1.SpeedButton4Click(Sender: TObject);
begin
Application.Terminate;
end;
```

### **Tugas**

Buat laporan sertakan screenshotnya dari latihan diatas

## BAB 12. MEMFORMAT DAN MEMVALIDASI DATA

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami tentang Format dan Validasi Data

Indikator : - Mampu membuat Validasi terhadap suatu data

### Dasar Teori

Validasi data adalah data yang bukan seharusnya diinputkan tidak dapat diinputkan/tidak diterima oleh komponen, semisal data yang **bertipe Number** tidak dapat diinputkan dengan yang **bertipe Text**.

### Latihan

Praktekkan pengisian format pada **TMaskEdit** agar karakter seperti A tidak bisa di masukkan. Ikuti langkah berikut :

- a. Buat form dengan tampilan sebagai berikut :



The screenshot shows a form with three input fields. The first field is labeled 'Nomor Retur' and contains a single character 'A'. The second field is labeled 'Tanggal Retur' and contains the date '7/27/2005'. The third field is labeled 'Banyak' and is empty. Below the fields are three buttons: 'Simpan', 'Batal', and 'Selesai'.

- b. klik pada ... yang terletak disebelah **kanan MaskEdit**, dan pilih Date hasilnya sebagai berikut .



- c. Modifikasikan Input Masks menjadi **!99/99/0000;1;\_**
- d. Klik OK untuk menutup **input mask editor**.

### Tugas

Gunakan validasi data ini terhadap salah satu program yang anda buat sebelumnya, buat laporan dengan sertakan screenshootnya

## BAB 13. DATA MODUL

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami tentang Data Modul

Indikator : - Mampu membuat Data Modul terhadap suatu Aplikasi

### Dasar Teori

Data Modul adalah suatu

Untuk membuat data modul pilihlah File, New Data Modul. Selanjutnya tempatkan komponen lain seperti TADOTable, TQuery dan sebagainya.

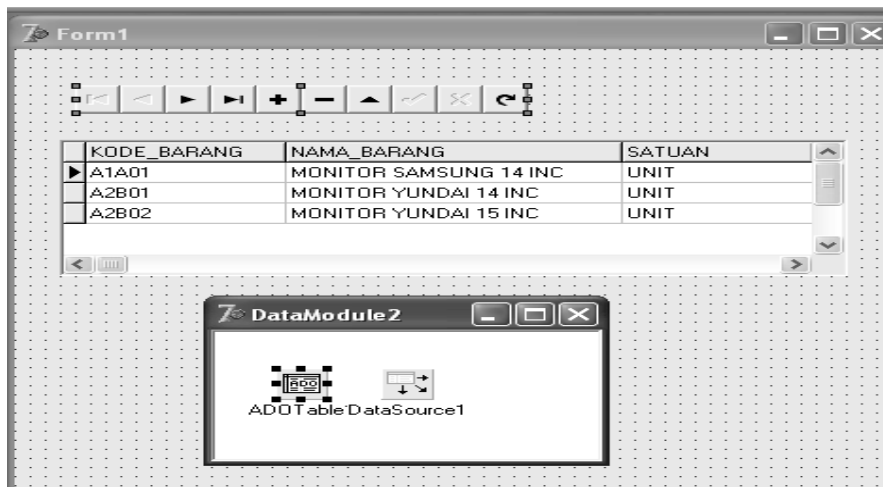
Langkah pembuatan datamodul.

1. Buat aplikasi baru dengan memilih **File, New Application**.
2. Buat sebuah **datamodul** dengan memilih **File, New Data Modul**.
3. Klik **page ADO** pada komponen **pallette**, lalu tempatkan komponen ADOTable di dalam **data modul**.
4. Tempatkan pula **DataSource** didalam **data modul**.
5. Klik page Data Control pada **Component Palette**, lalu tempatkan **DBGrid** pada form.
6. Ubah **Property komponen** Adotable seperti.

| Property         |
|------------------|
| ConnectionString |
| TableName        |
| Active           |

7. Klik komponen **DataSource1**, kemudian ubah property **DataSet** dengan **Table1**.
8. Tempatkan **DBNavigator** pada form.
9. Untuk mengakses dataset yang ada pada Datamodul, form harus menggunakan **Datamodul**. Caranya **Klik menu file, use unit**. Pilih **unitnya lalu OK**.
10. Klik **DBGrid**, ubah property **DataSource** pada Object Inspector menjadi **DataModul2.DataSource1**.

11. Lakukan langkah 10 pada **DBNavigator**.



**Gambar Menggunakan DataModul**

### Tugas

Buat sebuah aplikasi yang melibatkan beberapa table dan gabung semua komponen koneksinya dalam sebuah modul, buat laporannya dan sertakan screenshootnya.

## BAB 14. REPORT DENGAN QREPORT

### Capaian Pembelajaran dan Indikator

Capaian Pembelajaran: Memahami tentang pembuatan laporan dengan QReport

Indikator : - Mampu membuat aplikasi sederhana khususnya laporan dengan menggunakan QuickRep

### Latihan

Langkah-langkah yang dikerjakan

1. siapkan form baru (seperti gambar), dan tambahkan beberapa komponen QReport seperti : QuickRep, Qrband, QRLabel, QRDBText, label, edit, button, dan komponen-komponen untuk koneksi
2. koneksikan komponen-komponen yang ada, termasuk QuickRep dan QRDBText yang dikoneksikan dengan komponen-komponen yang telah terkoneksi pada database dan tabel.

The image shows a screenshot of a QReport form titled "BIODATA". The form is displayed on a grid background with column headers numbered 1 to 5 and row headers numbered 1 to 3. The fields are as follows:

|   | 1                      | 2             | 3        | 4 | 5 |
|---|------------------------|---------------|----------|---|---|
| 1 | <b>BIODATA</b>         |               |          |   |   |
|   | [NIP]                  | [nip]         |          |   |   |
|   | [NAMA]                 | [nama]        |          |   |   |
| 2 | [TEMPAT/TANGGAL LAHIR] | [tempatlahir] | [glahir] |   |   |
|   | [ENIS KELAMIN]         | [kelamin]     |          |   |   |
|   | [AGAMA]                | [agama]       |          |   |   |
| 3 | [ALAMAT]               | [alamat]      |          |   |   |

Column Header

### Tugas

Buat laporan dari soal latihan diatas sertakan screenshootnya

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |



|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |
|---------------------|------|-----------------------|
| <b>RESPONSI</b>     | Nama |                       |
|                     | NIM  |                       |
|                     |      |                       |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |



|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |
|---------------------|------|-----------------------|
| <b>RESPONSI</b>     | Nama |                       |
|                     | NIM  |                       |
|                     |      |                       |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |



|                     |      |                       |  |
|---------------------|------|-----------------------|--|
| <b>RESPONSI</b>     | Nama |                       |  |
|                     | NIM  |                       |  |
|                     |      |                       |  |
| Diperiksa Tanggal : |      | Paraf Dosen/Asisten : |  |

## **DAFTAR PUSTAKA**

Kadir, Abdul, 2004, Dasar Aplikasi Database MySql-Delphi, Andi, Yogyakarta  
Komputer, Wahana, 2003, Pemrograman Borland Delphi 7, Andi, Yogyakarta