

MODUL

VB FUNDAMENTAL

Disusun oleh:
Team Penyusun Modul



Akademik Manajemen Informatika dan Komputer

Bina Sarana Informatika

Jakarta 2010

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, yang telah memberikan rahmat dan hidayahnya sehingga modul Microsoft Visual Basic ini dapat terselesaikan dengan baik. Selanjutnya modul ini disusun untuk memberikan gambaran bagi mahasiswa yang mempelajari program Visual Basic versi 6.0. Dengan menggunakan metode “*Belajar Cepat Dan Menyenangkan*” karena modul ini disertai contoh kasus, sehingga lebih memudahkan anda dalam memahami Visual Basic.

Akhir kata penulis berharap semoga modul Visual Basic 6.0 ini dapat dipergunakan sebaik-baiknya dan dapat dijadikan referensi untuk mahasiswa umum yang ingin mempelajari program Visual Basic 6.0. Tak lupa penulis mengucapkan banyak terima kasih kepada semua pihak yang telah membantu dengan tenaga dan pikirannya, terima kasih juga kepada rekan–rekan instruktur, dosen dan semuanya yang tidak bisa disebutkan satu persatu, yang selalu mendukung penulis sehingga modul ini sehingga dapat selesai sesuai yang kita inginkan semua.

Penulis menyadari masih banyak kekurangan dalam penyusunan modul ini. Untuk itu saran dan kritik yang membangun sangat penulis harapkan guna perbaikan dan pengembangan modul ini kedepan.

Jakarta, April 2010

Penulis

DAFTAR ISI

Kata Pengantar	2
Daftar Isi.....	3
BAB I PENGENALAN VISUAL BASIC 6.0.....	6
1.1 Mengenal Microsoft Visual Basic 6.0.....	6
1.2 Object Oriented Programming (OOP) Visual Basic.....	6
BAB II STRUKTUR PROGRAM VISUAL BASIC.....	15
2.1. Pengenalan Form	15
2.2. Pengenalan Warna Pada Teks	16
BAB III TIPE DATA, VARIABEL DAN KONSTANTA	19
3.1. Tipe Data.....	19
3.2. Variabel dan Konstanta.....	20
3.3. Ruang Lingkup Variabel dan Konstanta.....	21
3.4. Latihan Variabel dan Konstanta.....	21
BAB IV OPERATOR	23
4.1. Operator Perbandingan Relasi	23
4.2. Operator Logika.....	24
4.3. Latihan Operator.....	24
BAB V FUNGSI STANDAR PADA VISUAL BASIC	31
5.1. Fungsi String.....	31
5.2. Fungsi Matematika.....	31
5.3. Fungsi Tanggal dan Waktu.....	31
5.4. Latihan Fungsi Standar.....	32

BAB VI	PERINTAH PERCABANGAN (<i>Branching</i>).....	35
	6.1. If	35
	A. If Then.....	35
	B. If Then Else	36
	C. If Bersarang (Nested IF)	43
	6.2. Select Case.....	43
BAB VII	PERULANGAN (<i>Looping</i>).....	49
	7.1. Do Loop.....	49
	7.2. For Next.....	50
	7.3. While Wend	51
BAB VIII	PROCEDURE DAN FUNCTION.....	54
	8.1. Jenis-jenis Procedure dan Function.....	54
	8.2. Pembuatan Procedure dan Function.....	56
BAB IX	MODUL.....	58
	9.1. Pengenalan Modul Standar.....	58
	9.2. Penggunaan Modul Standar.....	59
BAB X	ARRAY.....	61
	10.1. Pengertian Koleksi Objek	61
	10.2. Cara Mengakses Koleksi Objek.....	61
	10.3. Array Satu Dimensi	61
	10.4. Array Multi Dimensi	62
	10.5. Pembuatan Program Array	62
BAB XI	WINDOWS API	69
	11.1. Pengenalan Windows Api	69
	11.2. Struktur Windows Api	69
	11.3. Contoh Penggunaan Windows API	71

BAB XII	EFFECT FORM DAN KONTROL KOMPONENT.....	74
	12.1. Form membuka dari kiri ke kekanan.....	74
	12.2. Text Muncul Berjalan.....	76
	12.3. Penggunaan Effect Form.....	78
BAB XIII	DATABASE.....	81
	13.1 Pengertian Database.....	81
	13.2 Pembuatan Database.....	81
	13.3 Membuat Database dan Tabel.....	81
	13.4 Langkah Pembuatan Database dengan Visdata	82
	13.5 Membuat Tabel.....	84
	13.6 Pembuatan Form.....	87
	13.6.1. Pembuatan Form Barang.....	87
	13.6.2. Pembuatan Form One to One.....	94
	13.6.3. Pembuatan Form One to Many.....	100
	13.6.4. Pembuatan Form Customer.....	106
BAB XIV	LAPORAN DENGAN DATA REPORT	117
BAB XV	MENU UTAMA.....	121
	15.1 Desain Menu.....	121
	15.2 Koneksi Antar Form.....	123
BAB XVI	SQL (STRUCTURED QUERY LANGUAGE).....	124
BAB XVII	KONEKSI DATABASE DENGAN MODUL.....	127

DAFTAR PUSTAKA

BAB I

PENGENALAN VISUAL BASIC 6.0

1.1. Mengetahui Microsoft Visual Basic 6.0

Bahasa basic pada dasarnya adalah bahasa yang mudah dimengerti sehingga pemrograman di dalam Bahasa Basic dapat dengan mudah dilakukan. Hal ini lebih mudah lagi setelah hadirnya **Microsoft Visual Basic** yang dibangun dari IDE (*Integrated Development Environment*) untuk **membuat bahasa yang sederhana dan mudah dalam pembuatan scriptnya** (*Simple Scripting Language*) untuk graphic user interface dalam bentuk form sedangkan untuk codingnya menggunakan dialek bahasa basic yang dikembangkan dalam sistem operasi Microsoft Windows.

Visual Basic adalah bahasa pemrograman berbasis Windows yang merupakan **Object Oriented Programming (OOP)**, yaitu pemrograman berorientasi objek, Visual Basic menyediakan objek-objek yang sangat kuat, berguna dan sangat mudah dalam penggunaannya.

Pemakaian Visual Basic ditandai dengan kemampuan Visual Basic untuk dapat berinteraksi dengan aplikasi lain di dalam sistem operasi windows dengan komponen **ActiveX Control**. Dengan komponen ini memungkinkan pengguna untuk memanggil dan menggunakan semua model data yang ada di dalam sistem operasi windows. Hal ini juga ditunjang dengan teknik pemrograman di dalam **Visual Basic yang mengadopsi dua macam jenis pemrograman yaitu Pemrograman Visual dan Object Oriented Programming (OOP)**.

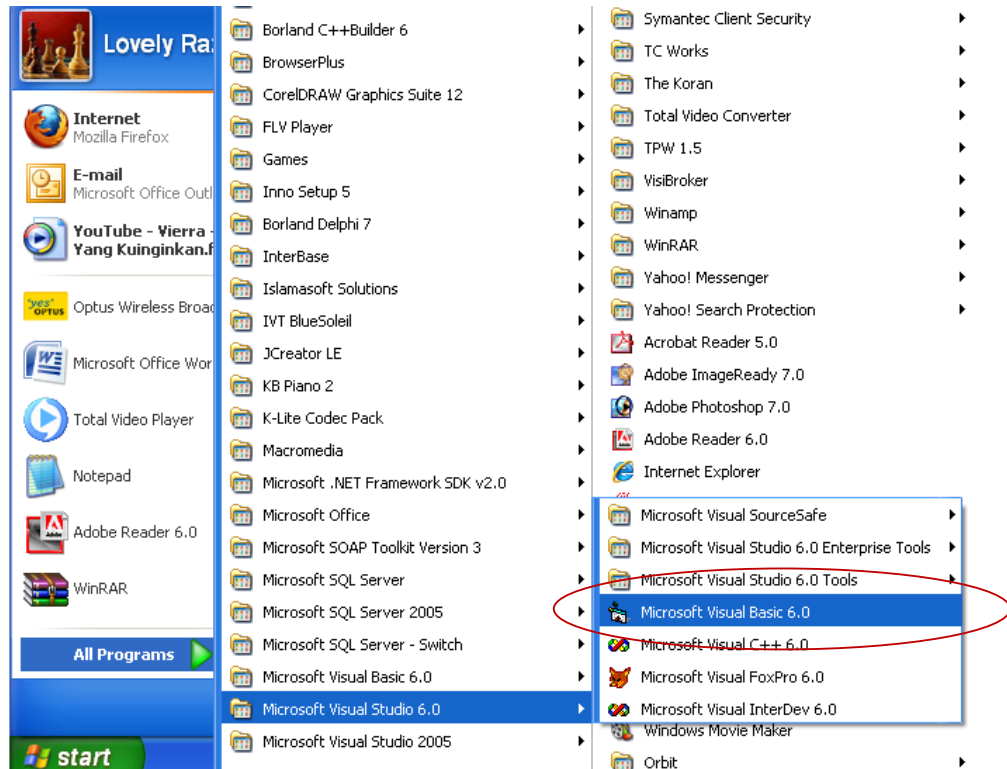
Dalam penggunaan *User-interface* tidak akan menyadari bahwa di belakangnya berjalan instruksi-instruksi program yang mendukung tampilan dan proses yang dilakukan.

Pada pemrograman Visual, pengembangan aplikasi dimulai dengan pembentukan *user interface*. kemudian *mengatur properti dari objek-objek* yang digunakan dalam *user interface*, dan baru dilakukan *penulis kode program* untuk menangani kejadian-kejadian (*event*). Tanpa pengembangan aplikasi demikian

dikenal dengan dengan istilah pengembangan aplikasi dengan *pendekatan Bottom Up*.

A. Menjalani IDE

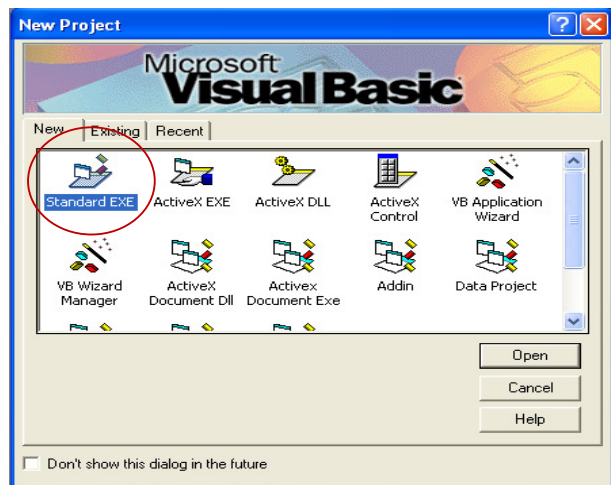
Salah satu cara untuk mengaktifkan IDE Visual Basic adalah menjalankannya dari Menu **Start**, pilih **All Programs**, lalu pilih **Microsoft Visual Studio 6.0** kemudian pilih **Microsoft Visual Basic 6.0**



Gambar 1.1 Mengaktifkan IDE Visual Basic 6.0

B. Memilih jenis Project

Sesaat setelah aktif di IDE Visual Basic, maka anda akan dihadapkan kepada suatu pilihan terhadap jenis Project yang ingin anda buat sebagaimana yang ditunjukkan oleh Gambar 1.2. Sebagai langkah awal dari proses belajar adalah memilih project **Standard EXE**.

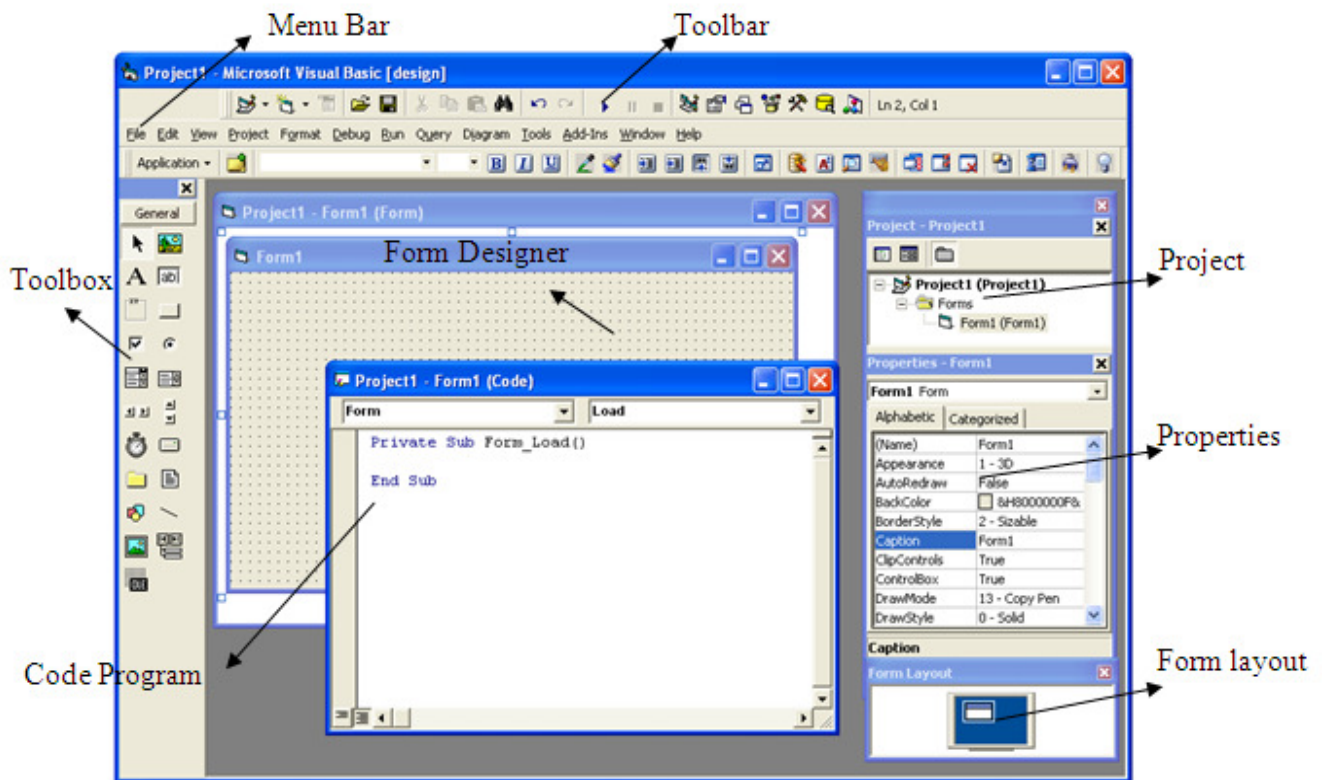


Gambar 1.2 Tampilan jendela New Project

C. Jendela IDE


IDE Visual Basic 6.0 menggunakan model MDI (*Multiple Document Interface*). Berikut ini adalah gambar yang menunjukkan bagian-bagian dan nama-nama jendela yang dapat tampil pada IDE Visual Basic. Mungkin pada IDE anda hanya ditampilkan sebagian jendela dibandingkan dengan Gambar 1.3 anda tidak perlu terkejut, karena penampilan dari jendela-jendela tersebut dapat diatur dengan perintah pada menu **View**.

Sebagai langkah awal dari proses belajar, tidak semua jendela akan kita gunakan, tetapi hanya beberapa yang penting, sedangkan yang lainnya bersifat khusus.



Gambar 1.3 IDE Visual Basic dengan jendela-jendela yang terbuka.

Sebagaimana dengan proses belajar ini, kita akan fokus pada beberapa jendela yang penting terlebih dahulu sehingga konsentrasi tidak menjadi pecah, dan peserta belajar menjadi bingung. Adapun jendela-jendela yang perlu anda perhatikan adalah sebagai berikut :

-
- **Menu Bar**, digunakan untuk memilih tugas-tugas tertentu seperti menyimpan project, membuka project, dll.
 - **Main Toolbar**, digunakan untuk melakukan tugas-tugas tertentu dengan cepat atau dengan menubar View kemudian pilih Toolbar
 - **Jendela Project**, jendela ini berisi gambaran dari semua model yang terdapat dalam aplikasi anda. Anda dapat menggunakan icon Toggle Folder untuk menampilkan model-model dalam jendela tersebut secara di group atau berurut berdasarkan nama. Anda dapat menggunakan Ctrl+R untuk menampilkan jendela project, ataupun menggunakan icon Project Explorer.
 - **Jendela Form Designer**, jendela ini merupakan tempat anda untuk merancang user interface dari aplikasi anda. Jadi jendela ini menyerupai kanvas bagi seorang pelukis. Cara Membuka View kemudian Form Layout Windows
 - **Jendela Toolbox**, jendela ini berisi komponen-komponen yang dapat anda gunakan untuk mengembangkan *user interface*. Membuka View kemudian pilih Toolbox atau klik  pada menu toolbar
 - **Jendela Code**, merupakan tempat bagi anda untuk menulis koding. Anda dapat menampilkan jendela ini dengan menggunakan kombinasi Shift-F7.
 - **Jendela Properties**, merupakan daftar properti-properti object yang sedang terpilih. Sebagai contohnya anda dapat mengubah warna tulisan (foreground). Anda dapat menggunakan F4 untuk menampilkan jendela properti atau dari menubar View kemudian pilih Property Windows.
 - **Jendela Color Palette**, adalah fasilitas cepat untuk mengubah warna suatu object. Untuk menampilkannya melalui View pada Menubar lalu pilih Color Pallete
-

- *Jendela Form Layout*, akan menunjukkan bagaimana form bersangkutan ditampilkan ketika runtime.

1.2. *Object Oriented Programming (OOP) pada Microsoft Visual Basic*

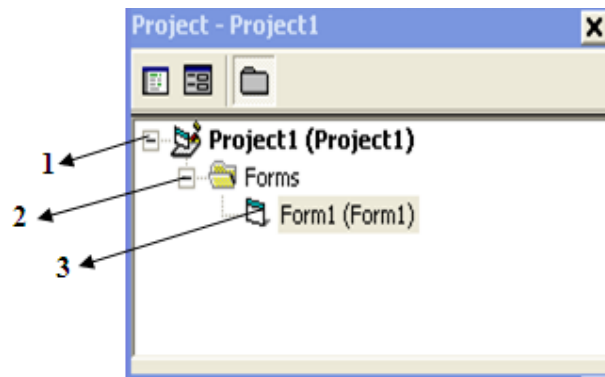
Ada beberapa istilah dan komponen dalam Visual Basic yang digunakan untuk membuat program aplikasi. Komponen–komponen yang akan dibahas ini hanya dasar-dasarnya saja agar dalam pembahasan selanjutnya anda sudah beradaptasi dan memahami penggunaan istilah dan komponen tersebut.

1. **Project**

Project adalah sekumpulan modul atau program aplikasi itu sendiri, project disimpan dalam file berextension **.VBP** biasanya berisi form-form.

2. **Ada tiga icon dalam Project Properties**

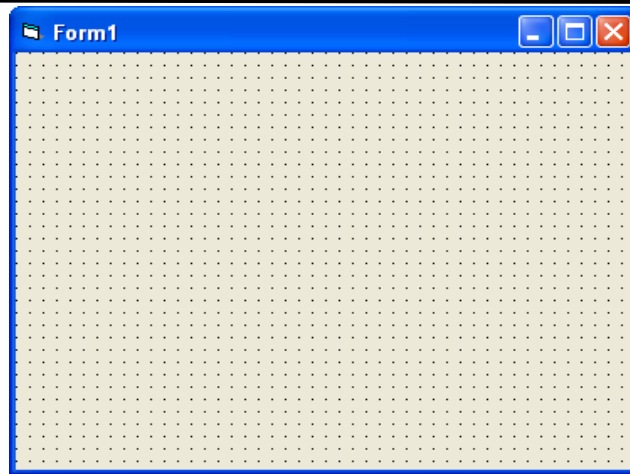
- View Code* : tampilkan jendela editor program
- View Object* : tampilkan bentuk formulir
- Toggle Folders* : Tampilkan folder tempat penyimpanan file



Gambar 1.4 *Jendela Project Properties*

3. **Form**

Form adalah objek yang dipakai untuk tempat bekerja program aplikasi, didalamnya tempat diletakan objek-objek lainnya, didalam form ada grid (garis titik-titik yang berguna untuk pengaturan letak). Form disimpan dalam file berextension **.FRM**
























Gambar 1.5 Tampilan Form

4. Toolbox

Toolbox kotak alat berisi icon-icon atau komponen untuk memasukan objek tertentu kedalam form. Jendela Toolbox merupakan jendela yang sangat penting bagi anda. Dari toolbox ini anda dapat mengambil komponen-komponen (object) yang akan ditanamkan pada form untuk membentuk *user interface*.

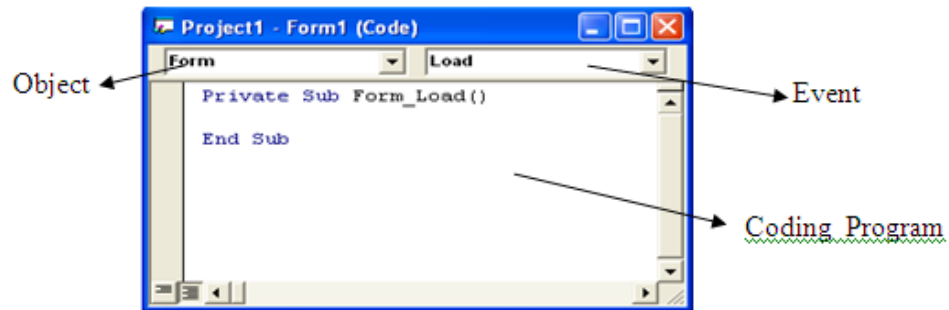
Adapun secara garis besar fungsi dari masing-masing intrinsic kontrol tersebut adalah sebagai berikut:

- a.  **Pointer** bukan merupakan suatu kontrol, gunakan icon ini ketika anda ingin memilih kontrol yang sudah berada pada form.
- b.  **PictureBox** adalah kontrol yang digunakan untuk menampilkan image dengan format: BMP, DIB (bitmap), ICO (icon), CUR (Cursor WMF (metafile), ENF (enhanced metafile), GIF dan JPEG.
- c.  **Label** adalah kontrol yang digunakan untuk menampilkan teks yang tidak dapat diperbaiki oleh pemakai.
- d.  **TextBox** adalah kontrol yang mengandung string yang dapat diperbaiki oleh pemakai, dapat berupa satu baris tunggal atau banyak baris.
- e.  **Frame** adalah kontrol yang digunakan sebagai container bagi kontrol lainnya.
- f.  **Command Button** merupakan kontrol hampir ditemukan pada setiap form, dan digunakan untuk membangkitkan event proses tertentu ketika pemakai melakukan klik padanya.

-
- g.  **CheckBox** digunakan untuk pilihan yang isinya bernilai Yes/No atau True/False.
- h.  **OptionButton** sering digunakan lebih dari satu sebagai pilihan terhadap beberapa option yang hanya dapat dipilih satu.
- i.  **ListBox** mengandung sejumlah item, dan user dapat memilih lebih dari satu (bergantung pada property *MultiSelect*).
- j.  **ComboBox** merupakan kombinasi dari **TextBox** dan suatu **ListBox** dimana pemasukan data dapat dilakukan dengan pengetikan maupun pemilihan.
- k.  **HscrollBox** dan  **VscrollBox** digunakan untuk membentuk scrollbar berdiri sendiri
- l.  **Timer** digunakan unntuk proses background yang diaktifkan berdasarkan interval waktu tertentu dan merupakan kontrol non visual.
- m.  **DriveListBox**,  **DirListBox** ,  **FileListBox** sering digunakan untuk membentuk dialog box yang berkaitan dengan file.
- n.  **Shape** dan  **Line** digunakan untuk menampilkan bentuk garis, persegi, bulatan dan Oval
- o.  **Image** berfungsi menyerupai **Imagebox**, tetapi tidak dapat digunakan sebagai kontainer bagi kontrol lainnya. Suatu yang perlu diketahui bahwa kontrol image menggunakan resource yang lebih kecil dibandingkan dengan **PictureBox**
- p.  **Data** digunakan untuk *data binding*.
- q.  **OLE** dapat digunakan sebagai tempat bagi program eksternal seperti Microsoft Excel, Word dan lain-lain.

Sebenarnya masih banyak komponen-komponen yang lain yang tidak dibahas disini, karena komponen vb begitu kompleks. Tapi kita bisa menambahkan komponen lainnya dengan cara memilih pada menu **Project** lalu pilih **Components** atau **CTRL+T**

5. **Kode Program** adalah serangkaian tulisan perintah yang akan dilaksanakan jika suatu dijalankan.

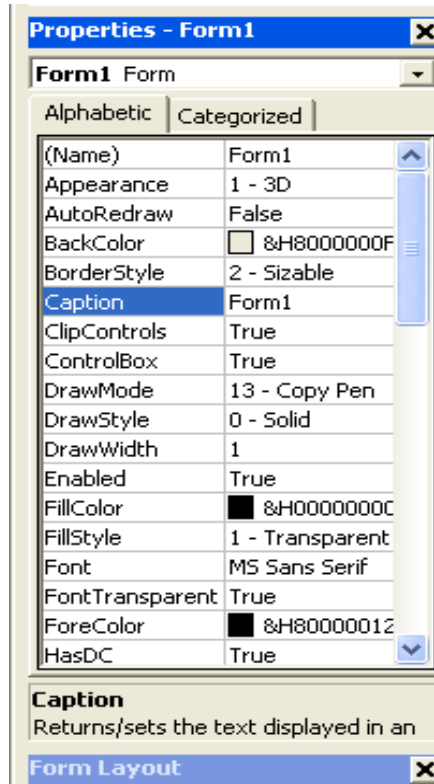


Gambar 1.6 Tampilan Jendela Coding Program

Didalam jendela coding terdapat istilah-istilah sebagai berikut:

- a. **Event** adalah peristiwa atau kejadian yang diterima oleh suatu objek, misalnya klik, seret dan lain-lain yang akan memicu program Visual Basic menjalankan kode program seperti coding program diatas.
- b. **Objek** sering disebut entity merupakan sesuatu yang biasa dibedakan dengan lainnya. Pada dasarnya seluruh benda didunia ini bisa dikatakan sebagai objek, contoh : motor, Komputer, TV dan lain-lain. Dalam Visual Basic objek-objek yang dimaksud disebut kontrol. Jenis-jenis kontrol antara lain : *Label*, *TextBox*, *ComboBox*, *ListBox*, *Command Button* dan masih banyak lagi.
- c. **Metoda** (method) adalah suatu perintah seperti halnya fungsi dan prosedur. Metode inilah tempat untuk mengekspresikan logika pemrograman dari pembuatan suatu program aplikasi. Contohnya Jika Mobil bergerak Maju, Mundur atau berbelok dan sebagainya.
- d. **Module** dapat disejajarkan dengan form, contoh form itu merupakan salah satu modul, yang berisi kode atau procedure.

6. **Properties** adalah untuk menentukan setting suatu object, menentukan cara kerja dari objek saat proram dijalankan misalnya warna, huruf dan lain-lain.



Gambar 1.7 Tampilan Properties Object Form

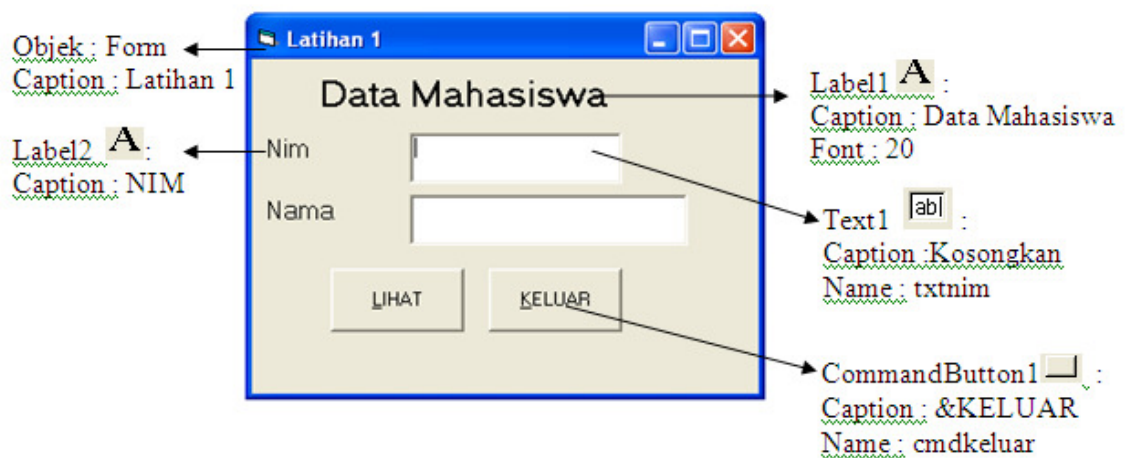
BAB II

STRUKTUR PROGRAM VISUAL BASIC

Pada pertemuan pertama kita sudah belajar mengenai pengenalan Visual Basic 6.0. Sekarang kita coba untuk membuat Struktur Program pada Visual Basic 6.0.

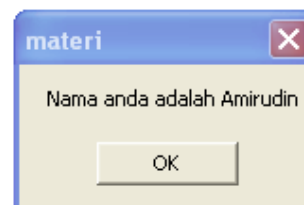
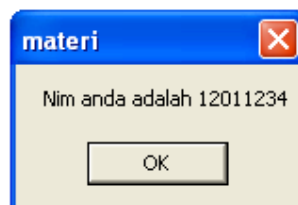
2.1. Pengenalan Form

Buat tampilan Form dibawah ini:

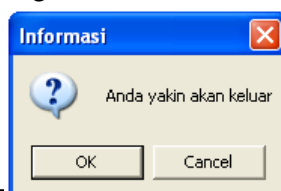


Logika Program :

1. Pada saat form di jalankan background ungu kemudian nim dan nama di input dan tampil warna tulisan merah dan biru dan background berwarna kuning pada textbox tersebut (perubahan dilakukan melalui properties)
2. Tombol lihat berfungsi untuk melihat hasil dari tampilan yang diinput



3. Tombol keluar digunakan untuk keluar dari program



Penyelesaian :**Private Sub cmdkeluar_Click()**

```
p = MsgBox("Anda yakin akan keluar", vbQuestion + vbOKCancel, "Informasi")
```

```
If p = vbOK Then
```

```
End
```

```
End If
```






End Sub**Private Sub cmdlihat_Click()**

```
MsgBox "Nim anda adalah " + txtnim.Text
```

```
MsgBox "Nama anda adalah " + txtnama.Text
```

End Sub**2.2. Pengenalan Warna Pada Teks**

Objek	Property	Nilai
Form1	BorderStyle	1- FixedSingle
	Caption	Latihan 2
	StartPosition	2- CenterScreen

Label1 	Caption Autosize Font Font Style Size	Pengenalan Warna dan Tulisan True Times New Roman Bold 18
TextBox1 	Name TextBox	txthasil Kosongkan
Frame1 	Caption	Pilihan
Option1 	Caption Name	Yellow OptYellow
Option2	Caption Name	Blue OptBlue
Option3	Caption Name	Red Optred
Option4	Caption Name	Bold Optbold
Option5	Caption Name	Italic Optitalic
Option6	Caption Name	Underline Optunderline
CommandButton1 	Caption Name	&BERSIH cmdbersih
CommandButton2	Caption Name	&KELUAR cmdkeluar

Ketentuan Soal :

1. Pada saat form di jalankan tampilan background berwarna hitam kemudian untuk textbox dengan pengaturan warna &HFFFF80
2. Menu Pilihan digunakan untuk memodifikasi bentuk tulisan di dalam textbox (gunakan coding program untuk merubah)
3. Button Bersih digunakan untuk membersihkan tampilan
4. Button keluar digunakan untuk keluar dari proram tanpa menggunakan pesan

Penyelesaian :**Private Sub Form_Activate()**

```
Form2.BackColor = vbBlack  
txthasil.BackColor = &HFFFF80
```

End Sub**Private Sub Optyellow_Click()**

```
txthasil.ForeColor = vbYellow
```

End Sub**Private Sub Optblue_Click()**

```
txthasil.ForeColor = vbBlue
```

End Sub**Private Sub Optred_Click()**

```
txthasil.ForeColor = vbRed
```

End Sub**Private Sub Optbold_Click()**

```
txthasil.FontBold = True  
txthasil.FontUnderline = False  
txthasil.FontItalic = False
```

End Sub**Private Sub Optitalic_Click()**

```
txthasil.FontItalic = True  
txthasil.FontBold = False  
txthasil.FontUnderline = False
```

End Sub**Private Sub Optunderline_Click()**

```
txthasil.FontUnderline = True  
txthasil.FontBold = False  
txthasil.FontItalic = False
```

End Sub**Private Sub cmdbersih_Click()**

```
txthasil = ""  
txthasil.SetFocus
```

End Sub**Private Sub cmdkeluar_Click()**

```
End
```

End Sub

BAB III

TIPE DATA, VARIABEL DAN KONSTANTA

Visual basic mengenal tipe-tipe data yang secara default sudah ada dalam program visual basic. Dan berikut ini adalah jenis-jenis tipe data visual basic.

3.1. Tipe Data

Type Data	Ukuran Storage	Jangkauan
Byte	1 Byte	0 s/d 255
Boolean	2 Byte	True atau False
Integer	2 Byte	-32,768 s/d 32767
Long	4 Byte	-2,147,483,648 s/d 2,147,483,647
Single	4 Byte	-3.402823E38 s/d -1.401298E-45 (-) 1.401298E-45 s/d 3.402823E38 (+)
Double	8 Byte	-1.79769313486232E308 s/d 922,337,203,685,477.5807
Currency	8 Byte	-922,337,203,685,477.5808 s/d 922,337,203,685,477.5807
Decimal	14 Byte	++ 79,228,162,514,264,337,593,543,950,335
Date	8 Byte	1 januari 100 s/d 31 desember 9999
Object	4 Byte	Mengacu pada objek tertentu
String (panjang variabel)	10 Byte + panjang string	0 sampai dengan 2 milyar
String (panjang tetap)	Panjang dari string	1 sampai lebih kurang 65,400
Variant (dengan angka)	16 Byte	Sembarang angka sampai jangkauan jenis double
Variant (dengan karakter)	22 Byte + panjang string	Sama dengan jangkauan variable string

Tabel 3.1 Tipe data, ukuran dan jangkauan masing-masing

3.2. Variabel dan Konstanta

Variabel adalah suatu tempat untuk menampung data yang nilainya selalu berubah. Sedangkan konstanta adalah suatu tempat untuk menampung data yang nilainya tetap dan tidak berubah. Aturan penamaan konstanta dan variabel :

1. Harus diawali huruf
2. Boleh terdiri dari huruf, angka dan garis bawah
3. Maksimal 225 karakter
4. Tidak boleh menggunakan reserve word

A. Deklarasi Variabel dan Konstanta

Deklarasi Variabel

Bentuk umum : `Dim <nama_variabel> [AS <tipe_variabel>]`

Pendeklarasian variabel ada 2 jenis :

1) *Deklarasi Eksplisit*

Untuk mendeklarasikan nama variabel beserta tipe datanya pada awal procedure (menempatkan alokasi nama variabel didalam memori), contohnya sebagai berikut :

```
Dim Nama as String
Dim Alamat as String *30 'Panjang maksimal 30
```

2) *Deklarasi Implisit*

Untuk mendeklarasikan nama variabel beserta tipe datanya dan langsung mengisi nilai variabelnya (menempatkan alokasi nilai variabel langsung didalam memori), contohnya sebagai berikut :

```
Nama$ = "Adi"
Gaji$ = 1750000
```

Deklarasi Konstanta:

```
[Public | Private] Const nama_konstanta [AS tipe_konstanta]
```

Contoh :

```
Const HARGA As Currency = 1000
Dim Biaya As Currency
Dim JUMLAH As Integer
... 'proses
Biaya = HARGA * JUMLAH
... 'proses
```

3.3. Ruang Lingkup Variabel dan Konstanta

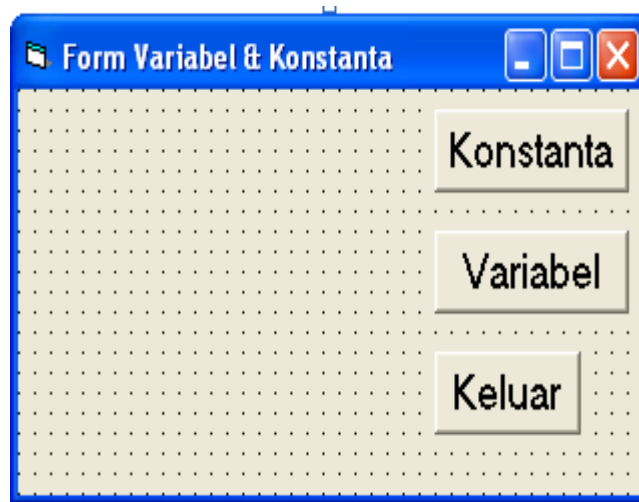
Variabel dapat dikenal diseluruh bagian program, dan ada yang hanya dikenal di procedure tempat variabel tersebut dibuat. Ruang lingkup paling dalam adalah lingkup procedure dengan perintah *Dim*, *Private* atau *Static* maka variabel tersebut mempunyai ruang lingkup procedure


Deklarasi variabel pada Visual Basic dapat dilakukan dengan Keyword berikut :

Keyword	Digunakan pada
Public	Berlaku pada level modul
Private	Berlaku pada level modul
Dim	Berlaku pada level modul dan level procedure
Static	Berlaku pada level procedure

3.4. Latihan Variabel dan Konstanta

1. Buat Form seperti tampilan berikut ini :



Objek	Property	Nilai
CommandButton1 	Caption Name	&KONSTANTA cmdkonstanta
CommandButton2	Caption Name	&VARIABEL cmdvariabel
CommandButton3	Caption Name	&KELUAR cmdkeluar

2. Ketikkan Listing Program berikut :

```

General           Declarations   Sub nilai2()
Public x As Single           Dim a As Integer, b As Integer
Public y As Single           a = InputBox("Ketikan Nilai1 :", "Input")
Public z As Single           b = InputBox("Ketikan Nilai2 :", "Input")
Sub nilai1()           c = a * b
x = 10                       FVar.Print "A dalam Nilai2 :", a
y = 15                       FVar.Print "B dalam Nilai2 :", b
z = x * y                     FVar.Print "C dalam Nilai2 :", c
FVar.Print "X dalam Nilai1 :", x
FVar.Print "Y dalam Nilai1 :", y
FVar.Print "Z dalam Nilai1 :", z
FVar.Print ""
End Sub               End Sub
Private Sub cmdkeluar_Click()   Private Sub cmdnilai1_Click()
Unload Me                     nilai1
End Sub               End Sub
Private Sub cmdkeluar_Click()   Private Sub cmdnilai2_Click()
Unload Me                     nilai2
End Sub               End Sub

```


BAB IV OPERATOR

Visual basic menyediakan operator aritmatika, komparasi dan logika, salah satu hal yang harus dipahami oleh programmer adalah tata urutan operasi dari masing-masing operator tersebut sehingga mampu membuat ekspresi yang akan menghasilkan nilai yang benar. Pada tabel dibawah ini menunjukkan operator dan urutan operasinya dari atas ke bawah

Aritmatika		Logika		Logika
Nama	Operator	Nama	Operator	Operator
Pangkat	^	Sama dengan	=	Not
Negatif	-	Tidak sama dengan	<>	And
Perkalian	*	Kurang dari	<	Or
Pembagian	/	Lebih dari	>	Xor
Sisa bagi	Mod	Kurang dari atau sama dengan	<=	Eqv
Penjumlahan	+	Lebih dari atau sama dengan	>=	Imp
Pengurangan	-	Like		
Penggabungan string	&			

Tabel 4.1 Contoh operator aritmatika

4.1. Operator Perbandingan Relasi

Operator ini digunakan untuk membandingkan suatu data atau ekspresi dengan data yang akan menghasilkan nilai logika (Boolean) benar atau salah.

Operator	Keterangan
=	Sama dengan
<>	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil atau sama dengan
>=	Lebih besar atau sama dengan
Like	Mempunyai ciri yang sama
Ls	Sama referensi object

Tabel 4.2 contoh operator perbandingan (Relasi)

4.2. Operator Logika

Operator logika digunakan untuk mengekspresikan salah satu atau lebih data logika menghasilkan data logika baru.

Operator	Keterangan
Not	Tidak
And	Dan
Or	Atau
Xor	Exklusif or
Eqv	Ekivalen
imp	Impikasi

Tabel 4.3 contoh operator Logika

4.3. Latihan Operator

1. Desain Form seperti tampilan dibawah ini :



bjek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1- FixedSingle Operator 2- CenterScreen
Label1	Caption AutoSize Font Font Style Size	Fungsi Operator True Ravie Bold 18
Frame1	Caption	Data
Label2	Caption	Nilai 1
TextBox1	Name Text	txtnil1 Kosongkan
Label3	Caption	Nilai2
TextBox2	Name Text	txtnil2 Kosongkan
Label4	Caption	Hasil
TextBox3	Name Text	Txthasil Kosongkan
Frame2	Caption	Operator Aritmatika
CommandButton1	Caption Name	+ Cmdtambah
CommandButton2	Caption Name	- Cmdkurang
CommandButton3	Caption Name	* Cmdkali
CommandButton4	Caption Name	/ Cmdbagi
CommandButton5	Caption Name	Mod Cmdmod
CommandButton6	Caption Name	Div Cmddiv
Frame3	Caption	Operator Relasi
CommandButton7	Caption Name	= cmd1
CommandButton8	Caption Name	< cmd2
CommandButton9	Caption Name	> cmd3
CommandButton10	Caption Name	<= cmd4
CommandButton11	Caption Name	=> cmd5
CommandButton12	Caption Name	<> cmd6
Frame4	Caption	Operator Logika
Label5	Caption	Nilai

ComboBox1	Text List	Kosongkan True False
CommandButton13	Caption Name	NOT Cmdnot
CommandButton14	Caption Name	AND Cmdand
CommandButton15	Caption Name	OR Cmdor
CommandButton16	Caption Name	&LAGI Cmdlagi
CommandButton17	Caption Name	&KELUAR Cmdkeluar
Label6	Caption	Nilai diinput True dan false untuk penggunaan Operator Logika

Ketentuan Soal :

1. Nilai1 dan Nilai2 di input dengan aturan :
 - a. Operator aritmatika dan operator relasi diinput nilai angka
 - b. Operator logika and dan or dengan true or false untuk pengisian nilai1 dan nilai2
 - c. Operator logika not melakukan pilihan suatu nilai true or false
 2. Button lagi digunakan mengulangi langkah penginputan dari awal
 3. Button keluar digunakan untuk keluar dari aplikasi
-

Langkah Pengerjaan :

```
Private Sub cmdtambah_Click()  
txthasil = Val(txtnil1.Text) + Val(txtnil2.Text)  
End Sub  
Private Sub cmdkurang_Click()  
txthasil = Val(txtnil1.Text) - Val(txtnil2.Text)  
End Sub  
Private Sub cmdkali_Click()  
txthasil = Val(txtnil1.Text) * Val(txtnil2.Text)  
End Sub  
Private Sub cmdbagi_Click()  
txthasil = Val(txtnil1.Text) / Val(txtnil2.Text)  
End Sub  
Private Sub cmdmod_Click()  
txthasil = Val(txtnil1.Text) Mod Val(txtnil2.Text)  
End Sub  
Private Sub cmdtambah_Click()  
txthasil = Val(txtnil1.Text) + Val(txtnil2.Text)  
End Sub  
Private Sub cmdkurang_Click()  
txthasil = Val(txtnil1.Text) - Val(txtnil2.Text)  
End Sub  
Private Sub cmdkali_Click()  
txthasil = Val(txtnil1.Text) * Val(txtnil2.Text)  
End Sub  
Private Sub cmdbagi_Click()  
txthasil = Val(txtnil1.Text) / Val(txtnil2.Text)  
End Sub  
Private Sub cmdmod_Click()  
txthasil = Val(txtnil1.Text) Mod Val(txtnil2.Text)  
End Sub
```

```
Private Sub cmd1_Click()  
txthasil = Val(txtnil1.Text) = Val(txtnil2.Text)  
End Sub  
Private Sub cmd2_Click()  
txthasil = Val(txtnil1.Text) < Val(txtnil2.Text)  
End Sub  
Private Sub cmd3_Click()  
txthasil = Val(txtnil1.Text) > Val(txtnil2.Text)  
End Sub  
Private Sub cmd4_Click()  
txthasil = Val(txtnil1.Text) <= Val(txtnil2.Text)  
End Sub  
Private Sub cmd5_Click()  
txthasil = Val(txtnil1.Text) >= Val(txtnil2.Text)  
End Sub  
Private Sub cmd6_Click()  
txthasil = Val(txtnil1.Text) <> Val(txtnil2.Text)  
End Sub  
Private Sub cmdkeluar_Click()  
End  
End Sub  
Private Sub cmdand_Click()  
If txtnil1 = "true" And txtnil2 = "true" Then  
    txthasil = "true"  
Elseif txtnil1 = "true" And txtnil2 = "false" Then  
    txthasil = "false"  
Elseif txtnil1 = "true" And txtnil2 = "false" Then  
    txthasil = "false"  
Elseif txtnil1 = "false" And txtnil2 = "true" Then  
    txthasil = "false"  
Else  
    txthasil = "false"  
End If  
End Sub
```

Private Sub cmdor_Click()

```
If txtnil1 = "true" Or txtnil2 = "true" Then
    txthasil = "true"
Elseif txtnil1 = "true" Or txtnil2 = "false" Then
    txthasil = "true"
Elseif txtnil1 = "false" Or txtnil2 = "true" Then
    txthasil = "true"
Else
    txthasil = "false"
End If
```

End Sub**Private Sub cmdNot_Click()**

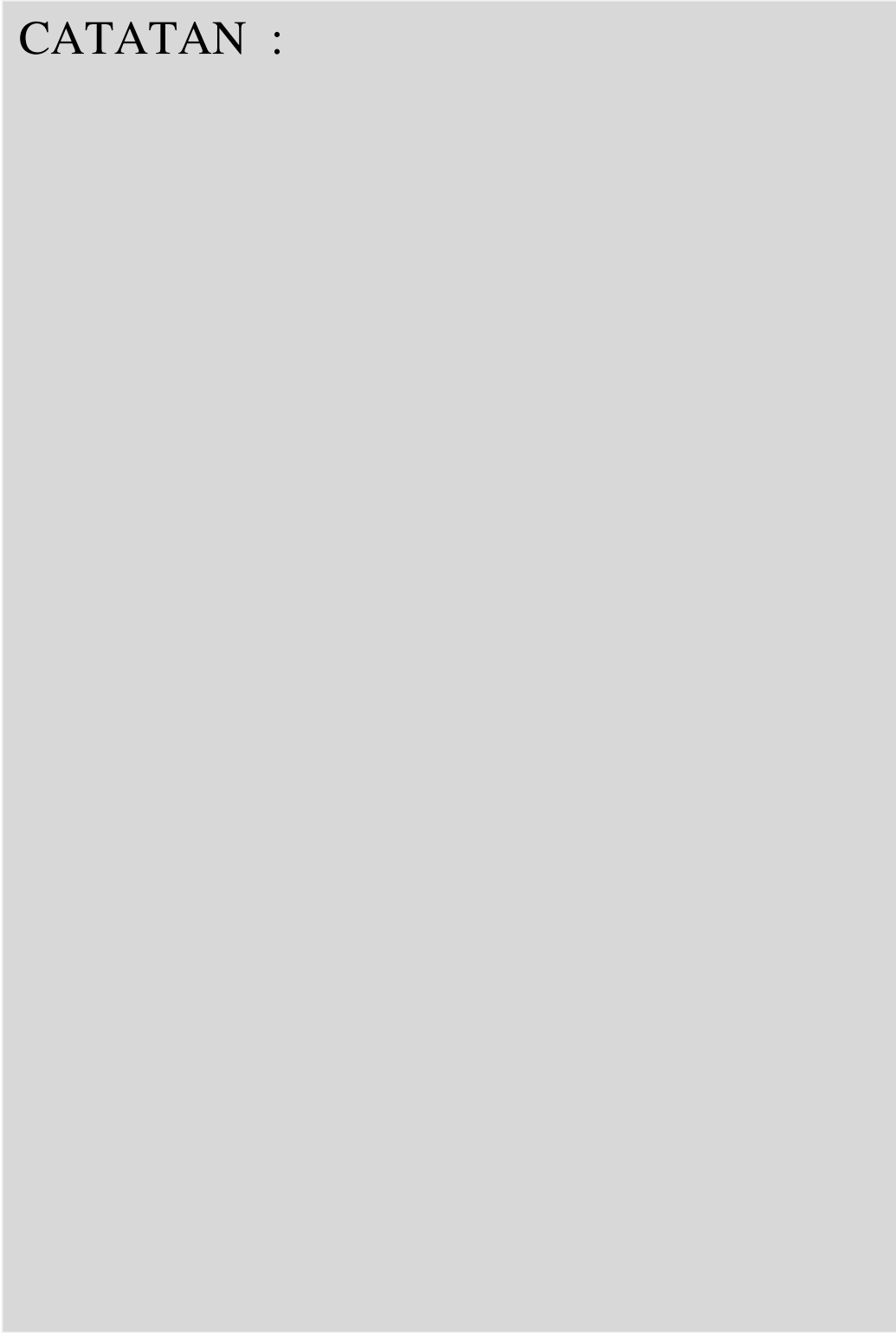
```
If cmbnot = True Then
    txtnot = Not True
Else
    txtnot = Not False
End If
```

End Sub**Private Sub cmdlagi_Click()**

```
txtnil1.Text = ""
txtnil2.Text = ""
txthasil = ""
txtnot = ""
txtnil1.SetFocus
```

End Sub

CATATAN :



BAB V

FUNGSI STANDAR DALAM VISUAL BASIC

5.1. Fungsi String

- a. **Left**, mengambil Karakter dari sebelah kiri
- b. **Right**, mengambil Karakter dari sebelah kanan
- c. **Mid**, mengambil Karakter dari tengah
- d. **LTrim**, Membuang semua awal spasi yang ada di sebelah kiri
- e. **RTrim**, Membuang semua spasi yang ada di sebelah kanan
- f. **Trim**, Membuang semua spasi yang ada di sebelah kiri dan kanan
- g. **Len**, Memberikan nilai balik Integer sebagai panjang string
- h. **LCase**, Mengubah String menjadi huruf kecil semua
- i. **UCase**, Mengubah String menjadi huruf besar semua
- j. **Str (int)**, Mengubah bilangan menjadi string (konversi)
- k. **Chr (Ascii)**, Konversi dari nilai ascii menghasilkan karakter

5.2. Fungsi Matematika

- a. **Val(str)** → Konversi String menjadi Integer
- b. **Asc(Char)** → Konversi Karakter Menghasilkan Nilai Ascii
- c. **Abs(Number)** → Memberikan Nilai Mutlak dari suatu Nilai Negatif
- d. **Round(Number, Decimal)** → Membulatkan suatu Angka sampai beberapa digit decimal
- e. **Sqr(Number)** → memberikan hasil akar kuadrat dari suatu nilai

5.3. Fungsi Tanggal dan Waktu

- a. **Date()** → Mengambil Tanggal Sistem komputer
 - b. **IsDate** → Memberikan Nilai balik Boolean. True jika parameter adalah sebuah Tanggal
 - c. **DateValue** → Konversi String Menjadi Date
-

- d. Date(Date), Month(Date), Year(Date)→Memberikan Nilai Integer dari Variabel Date
- e. untuk hari , bulan, dan Tahun.
- f. Time()→Mengambil Waktu dari system Komputer
- g. Hour(Date), Minute(Date), Second(Date)→ Memberikan Nilai Integer dari Variabel
- h. Date untuk jam, menit, dan detik.
- i. TimeValue(str)→Konversi String ke Variabel Date

5.4. Latihan Fungsi Standar

1. Desain Form seperti tampilan dibawah ini :

STRING

Monday, 12 April 2010 10:45:35 AM

Input Kata

BINA SARANA INFORMATIKA

Hasil

LEFT	BIN
RIGHT	IKA
MID	NA
LEN	23
LTRIM	BINA SARANA INFORMATIKA
RTRIM	BINA SARANA INFORMATIKA
LCASE	bina sarana informatika
UCASE	BINA SARANA INFORMATIKA

BERSIH KELUAR

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1- FixedSingle String 2- CenterScreen
Label1	Caption Name	Kosongkan ltanggal
Label2	Caption Name	Kosongkan Ljam
Frame1	Caption	Input Kata
TextBox1	Name Text	txtkata Kosongkan
Frame2	Caption	Hasil
Label3	Caption	LEFT
TextBox2	Name Text	txtleft Kosongkan
Label4	Caption	RIGHT
TextBox3	Name Text	txtright Kosongkan
Label5	Caption	MID
TextBox4	Name Text	txtmid Kosongkan
Label6	Caption	LEN
TextBox5	Name Text	txtlen Kosongkan
Label7	Caption	LTRIM
TextBox6	Name Text	txtltrim Kosongkan
Label8	Caption	RTRIM
TextBox7	Name Text	txtrtrim Kosongkan
Label9	Caption	LCASE
TextBox8	Name Text	txtlcase Kosongkan
Label10	Caption	UCASE
TextBox7	Name Text	txtucase Kosongkan
CommandButton1	Caption Name	&BERSIH cmdbersih
CommandButton2	Caption Name	&KELUAR cmdkeluar

Ketentuan Pengerjaan :

1. Pada saat form di jalankan maka tanggal dan jam tampil
2. Textbox input kata berisi kata yang akan di jadikan fungsi
3. Pada saat di input kata yang akan di jadikan fungsi maka Textbox di kolom hasil otomatis akan terisi
4. Button bersih berguna untuk membersihkan isi di dalam Textbox
5. Button keluar berfungsi untuk keluar dari program aplikasi

Langkah Pengerjaan :**Private Sub txtkata_Change()**

```
txtleft = Left(txtkata, 3)
txtright = Right(txtkata, 3)
txtmid = Mid(txtkata, 3, 3)
txtlen = Len(txtkata)
txtltrim = LTrim(txtkata)
txtright = RTrim(txtkata)
txtlcase = LCase(txtkata)
txtucase = UCase(txtkata)
```

End Sub**Private Sub cmdbersih_Click()**

```
txtkata.Text = ""
txtleft.Text = ""
txtright.Text = ""
txtmid.Text = ""
txtlen.Text = ""
txtltrim.Text = ""
txtright.Text = ""
txtlcase.Text = ""
txtucase.Text = ""
txtkata.SetFocus
```

End Sub**Private Sub cmdkeluar_Click()**

```
x = MsgBox("Anda Akan keluar dari form ini ?", vbQuestion +
          vbOKCancel, "Informasi")
If x = vbOK Then
    End
End If
```

End Sub**Private Sub Timer1_Timer()**

```
ljam.Caption = Time
ltanggal.Caption = Format(Date, "dddd,d mmmm yyyy")
```

End Sub

BAB VI

PERINTAH PERCABANGAN (BRANCHING)

Struktur kendali memungkinkan anda untuk mengatur jalannya program anda, Jika membiarkan tanpa di periksa oleh statement control-flow, suatu logika program akan berjalan dari kiri ke kanan dan dari atas kebawah. Hanya program yang sangat sederhana dapat ditulis tanpa statement control-flow. Struktur KENDALI yang didukung oleh Visual Basic adalah sebagai berikut:

6.1 If..

A. If...Then

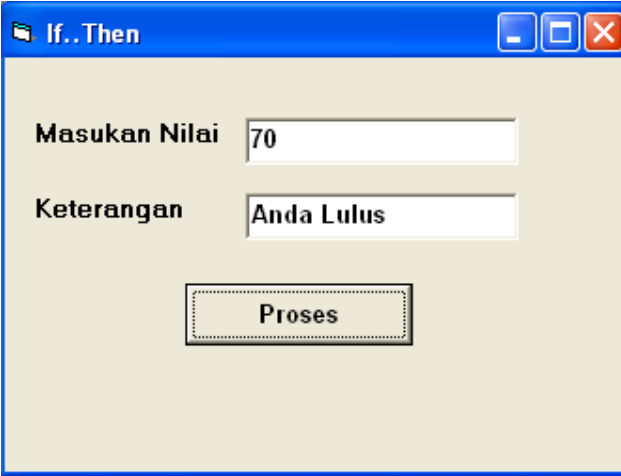
Gunakan suatu struktur If...Then untuk menjalankan suatu pernyataan secara kondisional. Anda dapat menggunakan syntax satu baris ataupun syntax banyak baris.

Syntax:

```
If kondisi Then pernyataan
Atau
If kondisi Then
    pernyataan-pernyataan
End If
```

Kondisi biasanya berupa suatu perbandingan, maupun ekspresi yang menghasilkan nilai numerik. Visual Basic menginterpretasikan False sebagai nol (0), dan True sebagai satu.

Contoh :



The screenshot shows a Windows-style window titled "If.. Then". Inside the window, there are two text input fields. The first is labeled "Masukan Nilai" and contains the number "70". The second is labeled "Keterangan" and contains the text "Anda Lulus". Below these fields is a button with the text "Proses". The window has a blue title bar and standard minimize, maximize, and close buttons.

Gambar 6.1 Form latihan If..Then

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle If..Then 2- CenterScreen
Label1	Caption	Masukan Nilai
TextBox1	Name Text	txtnilai Kosongkan
Label2	Caption	Keterangan
TextBox2	Name Text	txtket Kosongkan
CommandButton1	Caption Name	Proses Cmdproses

Ketentuan Logika :

Logika dari program diatas adalah ketika diinput nilai lebih dari atau sama dengan 60 (≥ 60) maka akan menghasilkan keterangan “Anda Lulus”.

Listing kodenya:

```
Private sub cmdhitung_click()
  If txtnilai1.text >=6 then
    Txtket.text="Anda Lulus"
  End IF
End Sub
```

B. If...Then...Else

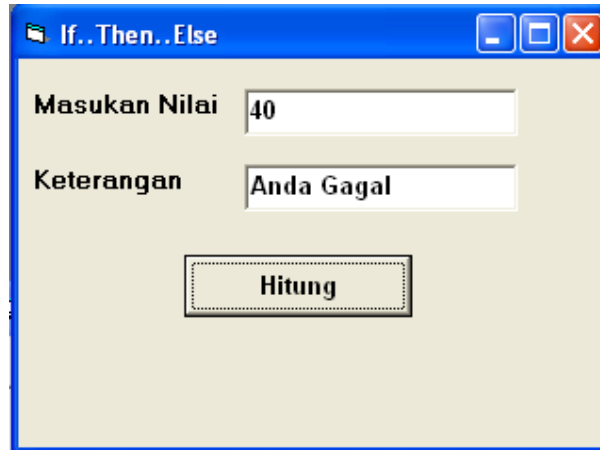
Gunakan If...Then...Else untuk mendefinisikan beberapa blok pernyataan yang akan dijalankan salah satu berdasarkan kondisi yang memenuhi syarat

Syntax:

```
If kondisi1 Then
  [blok pernyataan-1]
[Elseif kondisi2 Then
  [blok pernyataan-2]] ...
[Else
  [blok pernyataan-n]]
End If
```

Visual Basic awalnya akan mencoba kondisi1. Jika False, maka Visual Basic akan memeriksa kondisi2, dan seterusnya sampai menemukan suatu kondisi True untuk dijalankan blok pernyataannya.

Contoh :



Gambar 6.2 Form latihan If..Then Else

Ketentuan Logika :

Dari kondisi diatas jika untuk menampilkan keterangan gagal, kita cukup menambahkan kondisi yang kedua saja yaitu else.

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle If..Then..Else 2- CenterScreen
Label1	Caption	Masukan Nilai
TextBox1	Name Text	Txtnilai Kosongkan
Label2	Caption	Keterangan
TextBox2	Name Text	Txtket Kosongkan
CommandButton1	Caption Name	Hitung Cmdhitung

Listing kodenya:

```
Private sub cmdhitung_click()  
If txtnilai1.text >=6 then  
Txtket.text="Anda Lulus"  
Else  
Txtket.text="Anda Gagal"  
End IF  
End Sub
```

Contoh Kasus :

Buat tampilan form dibawah ini :

Percabangan

PERHITUNGAN NILAI MAHASISWA

Niai Absen	<input type="text" value="100"/>	× 10% =	<input type="text" value="10"/>	→ habsen
Niai Tugas	<input type="text" value="70"/>	× 20% =	<input type="text" value="14"/>	→ htgs
Niai UTS	<input type="text" value="85"/>	× 30% =	<input type="text" value="25.5"/>	→ huts
Niai UaS	<input type="text" value="100"/>	× 40% =	<input type="text" value="40"/>	→ huas

Total Nilai

Grade

Keterangan

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Percabangan 2- CenterScreen
Label1	Caption	Perhitungan Nilai Mahasiswa
Label2	Caption	Nilai Absen
TextBox1	Name Text	txtabsen Kosongkan
Label3	Caption	Nilai Tugas
TextBox2	Name Text	txttgs Kosongkan
Label4	Caption	Nilai UTS
TextBox3	Name Text	Txtuts Kosongkan
Label5	Caption	Nilai UAS
TextBox4	Name Text	Txtuas Kosongkan
Label6	Caption	X 10%
TextBox5	Name Text	habsen Kosongkan
Label7	Caption	X 20%
TextBox6	Name Text	Htgs Kosongkan
Label8	Caption	X 30%
TextBox7	Name Text	Huts Kosongkan
Label9	Caption	X 40%
TextBox8	Name Text	Huas Kosongkan
Label10	Caption	Total Nilai
TextBox9	Name	Txttotnil

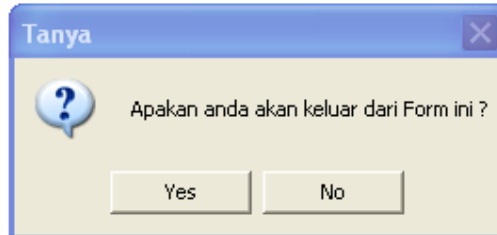
	Text	Kosongkan
Label11	Caption	Grade
TextBox10	Name Text	Txtgrade Kosongkan
Label12	Caption	Keterangan
TextBox11	Name Text	Txtket Kosongkan
CommandButton1	Name Caption	Cmdhitung &HITUNG
CommandButton2	Name Caption	Cmdbersih &BERSIH
CommandButton3	Name Caption	Cmdkeluar &KELUAR

Ketentuan Soal :

1. Saat Program di jalankan maka TextBox habsen, htgs, huts, dan huas dalam kondisi tidak aktif dan kursor berada dalam Textbox txtabsen
2. Pengisian nilai berdasarkan ketentuan berikut pada saat nilai diinput :
 - a. habsen didapat dari 10% X Nilai Absen
 - b. htgs didapat dari 20% X Nilai Tugas
 - c. huts didapat dari 30% X Nilai UTS
 - d. huas didapat dari 40% X Nilai UAS
3. Button hitung di klik maka akan tampil total nilai, grade dan keterangan dengan ketentuan :
 - a. Total Nilai didapat dari habsen + htgs + huts + huas
 - b. Grade didapat berdasarkan ketentuan dibawah ini :

Total Nilai	Grade
100 – 80	A
79 – 69	B
68 – 56	C
55 – 40	D
<40	E

- c. Keterangan dengan berdasarkan grade jika A atau B atau C maka keterangan “LULUS” selain itu maka akan “GAGAL”
4. Button Bersih untuk mengulang kembali
5. Button keluar untuk keluar dari aplikasi dan tampil pesan konfirmasi seperti gambar berikut :



Langkah Pengerjaan :

Private Sub Form_Activate()

```
habsen.Enabled = False
```

```
htgs.Enabled = False
```

```
huts.Enabled = False
```

```
huas.Enabled = False
```

```
txtabsen.SetFocus
```

End Sub

Private Sub txtabsen_Change()

```
habsen = 0.1 * Val(txtabsen)
```

End Sub

Private Sub txttgs_Change()

```
htgs = 0.2 * Val(txttgs)
```

End Sub

Private Sub txttuts_Change()

```
huts = 0.3 * Val(txttuts)
```

End Sub

Private Sub txtuas_Change()

```
huas = 0.4 * Val(txtuas)
```

End Sub

```
Private Sub cmdhitung_Click()  
'Menghitung Total Nilai  
txttotnil = Val(habsen.Text) + Val(htgs.Text) + Val(huts.Text) + Val(huas.Text)  
'menghitung grade  
If txttotnil >= 80 And txttotnil <= 100 Then  
    txtgrade = "A"  
Else  
If txttotnil >= 69 And txttotnil <= 79 Then  
    txtgrade = "B"  
Else  
If txttotnil >= 56 And txttotnil <= 68 Then  
    txtgrade = "C"  
Else  
If txttotnil >= 40 And txttotnil <= 55 Then  
    txtgrade = "D"  
Else  
    txtgrade = "E"  
End If  
End If  
End If  
End If  
'Menghitung keterangan  
If txtgrade = "A" Or txtgrade = "B" Or txtgrade = "C" Then  
    txtket = "LULUS"  
Else  
    txtket = "GAGAL"  
End If  
End Sub
```

C. If Bercabang / Bersarang (Nested IF)

If yang terletak di dalam IF

Bentuk Umum :

```
If kondisi ke-1 Then
    If kondisi ke-1 Then
        Pernyataan ke-1
    Else
        Pernyataan Ke-2
    endif
Else
    If kondisi ke-1 Then
        Pernyataan ke-1
    Else
        Pernyataan Ke-2
    endif
End IF
```

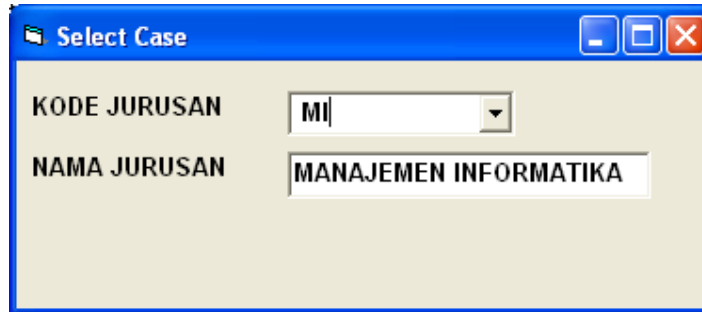
6.2. Select Case

Visual Basic menyediakan struktur Select Case sebagai suatu alternatif terhadap *If...Then...Else*. Suatu Select Case statement memiliki kemampuan yang sama dengan *If...Then...Else...*, tetapi membuat code lebih mudah dibaca.

Struktur Select Case bekerja dengan suatu percobaan tunggal yang hanya dievaluasi satu kali pada bagian atas struktur. Visual Basic then membandingkan hasil ekspresi dengan nilai pada setiap Case didalam struktur tersebut, jika ada yang sesuai, akan dijalankan blok statement yang sesuai.

```
Select Case ekspresi yang dicoba
    [Case ekspresi1
        [blok pernyataan-1]]
    [Case ekspresi2
        [blok pernyataan-2]]
    ...
    [Case Else
        [blok pernyataan-n]]
End Select
```

Contoh:



Gambar 6.3 Form latihan Case

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Select Case 2- CenterScreen
Label1	Caption	Kode Jurusan
ComboBox1	Name Text	Kosongkan Kosongkan
Label2	Caption	Nama Jurusan

Setelah membuat desain formnya tambahkan listing berikut:

```

Private sub combo1_click()
  Select case combo1
  Case "MI"
    Text1.text = "Manajemen Informatika"
  Case "KA"
    Text1.text = "Komputer Akuntansi"
  Case else
    Text1.text = "Teknik Informatika"
  End select
End sub

```

Contoh Kasus :

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Percabangan 2- CenterScreen
Label1	Caption	Café YUNFA
Label2	Caption	Pilih Menu
ComboBox1	Name Text	Cmbmenu Kosongkan
Label3	Caption	Pilih Paket
ComboBox1	Name Text	Cmbpaket Kosongkan
Label4	Caption	List Menu
List1	Name	List1
Label5	Caption	Harga
TextBox1	Name Text	txthrgmenu Kosongkan

Frame1	Caption	Fasilitas
Option1	Caption Name	Delivery Option1
Option2	Caption Name	Makan Ditempat Option2
Label6	Caption	Harga
TextBox2	Name Text	txthrgfasilitas Kosongkan
Frame2	Caption	Daftar Minuman
Check1	Name Caption	Check1 Es Teh Manis
Check2	Name Caption	Check2 Air Mineral
TextBox3	Name Text	txthrgminum Kosongkan
TextBox4	Name Text	txtjumbel Kosongkan
TextBox5	Name Text	txttotbay Kosongkan
TextBox6	Name Text	txtubay Kosongkan
TextBox7	Name Text	txtukem Kosongkan

Ketentuan Pengerjaan :

- Form dijalankan ada pilihan Menu dan Pilih Paket dengan ketentuan :

Pilih Menu	Pilih Paket	List Menu	Harga
Sarapan pagi	Paket 1	Nasi Uduk Telur Tempe Orek	5000
	Paket 2	Nasi Goreng	7000

		Telur Mata Sapi Tempe Goreng	
Makan Siang	Paket 1	Nasi Putih Ayam Bakar Lalapan	8000
	Paket 2	Nasi Putih Rendang Sambal Ijo Sayur	10000

2. Fasilitas jika delivery ditambah biaya 5000 dan apabila tidak maka tidak ada penambahan biaya
3. Pilihan Minuman berdasarkan pilihan
4. Jumlah beli di input di tekan enter maka total bayar muncul otomatis didapat dari jumlahbeli X (hargamenu + hargafasilitas + harga minuman)
5. Uang kembali tampil dan didapat dari uang bayar – total bayar pada saat diinput uang bayar

Langkah Pengerjaan :

```

Private Sub Form_Load()
cmbmenu.AddItem ("Sarapan Pagi")
cmbmenu.AddItem ("Makan Siang")
cmbpaket.AddItem ("Paket 1")
cmbpaket.AddItem ("Paket 2")
End Sub

Private Sub Option1_Click()
txthrgfasilitas = 5000
End Sub

Private Sub Option2_Click()
txthrgfasilitas = 0
End Sub

Private Sub txtjumbel_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
txttotbay = Val(txtjumbel) * (Val(txthrgmenu) + Val(txthrgfasilitas) + Val(txthrgminum))
txtubay.SetFocus
End If
End Sub

```

Private Sub cmbpaket_Click()

```
If cmbmenu.Text = "Sarapan Pagi" Then
```

```
    Select Case (cmbpaket.Text)
```

```
        Case "Paket 1"
```

```
            List1.Clear
```

```
            List1.AddItem ("Nasi Uduk")
```

```
            List1.AddItem ("Telur")
```

```
            List1.AddItem ("Tempe Orek")
```

```
            txthrgmenu = 5000
```

```
        Case Else
```

```
            List1.Clear
```

```
            List1.AddItem ("Nasi Goreng")
```

```
            List1.AddItem ("Telur Mata Sapi")
```

```
            List1.AddItem ("Tempe goreng")
```

```
            txthrgmenu = 7000
```

```
        End Select
```

```
Else
```

```
    Select Case (cmbpaket.Text)
```

```
        Case "Paket 1"
```

```
            List1.Clear
```

```
            List1.AddItem ("Nasi Putih")
```

```
            List1.AddItem ("Ayam Bakar")
```

```
            List1.AddItem ("Lalapan")
```

```
            txthrgmenu = 8000
```

```
        Case Else
```

```
            List1.Clear
```

```
            List1.AddItem ("Nasi Putih")
```

```
            List1.AddItem ("Rendang")
```

```
            List1.AddItem ("Sayur")
```

```
            List1.AddItem ("Sambal Ijo")
```

```
            txthrgmenu = 10000
```

```
        End Select
```

```
End If
```

```
End Sub
```

Private Sub Check1_Click()

```
If Check1 = 1 And Check2 = 1 Then
```

```
    txthrgminimum = 3000
```

```
Elseif Check1 = 1 And Check2 = 0 Then
```

```
    txthrgminimum = 2500
```

```
Elseif Check1 = 0 And Check2 = 1 Then
```

```
    txthrgminimum = 500
```

```
Else
```

```
    txthrgminimum = 0
```

```
End If
```

```
End Sub
```

Private Sub Check2_Click()

```
If Check1 = 1 And Check2 = 1 Then
```

```
    txthrgminimum = 3000
```

```
Elseif Check1 = 1 And Check2 = 0 Then
```

```
    txthrgminimum = 2500
```

```
Elseif Check1 = 0 And Check2 = 1 Then
```

```
    txthrgminimum = 500
```

```
Else
```

```
    txthrgminimum = 0
```

```
End If
```

```
End Sub
```

Private Sub txtubay_Change()

```
txtukem = Val(txtubay) - (txtotbay)
```

```
End Sub
```

```
End Sub
```

BAB VII

PERULANGAN (LOOPING)

Struktur loop memperbolehkan anda untuk melaksanakan sekelompok baris terlebih dari satu kali.

7.1. Do...Loop

Gunakan Do Loop untuk mengeksekusi suatu blok statement dengan jumlah perulangan yang tak terhingga, ada beberapa Do...Loop statement, tetapi masing-masing mengevaluasi suatu kondisi untuk menentukan apakah melanjutkan eksekusi Berikut ini adalah statement Do Loop, yang dijalankan selama kondisi benar :

```
Do While <kondisi>  
    statements  
Loop
```

Ketika Visual menjalankan Do Loop ini, pertama kali akan di coba kondisinya, jika kondisi False (zero), akan diloncati semua statement yang mengikuti kondisi tersebut. Visual Basic **akan menjalankan statement jika kondisi benar dan kembali ke Do...Loop berikutnya.**

```
Function Faktorial (x)  
    Dim Hasil As Double  
    Dim Count As Long  
    Hasil = 1  
    count = 2  
    Do While Count <= x  
        Hasil = Hasil * count  
        count = count + 1  
    Loop  
    Faktorial = count  
End Function
```

Variasi lain dari statement Do...Loop, yang menjamin minimal satu kali statement dijalankan :

```
Do
  statements
Loop While <Kondisi>
```

Dua variasi yang analog dengan dua contoh sebelumnya adalah perulangan minimal satu kali. Dan berikut ini adalah berulang atau tidak sama sekali :

```
Do Until <Kondisi>
  statements
Loop
Do
  statements
Loop Until <Kondisi>
```

7.2. For...Next

Do Loop bekerja dengan baik, ketika anda tidak tahu berapa banyak kali untuk butuhkan untuk menjalankan statement. Ketika anda **mengetahui harus menjalankan statement sejumlah kali**, bagaimanapun For...Next adalah pilihan yang lebih baik. Tidak seperti Do Loop, For...Next menggunakan suatu variabel yang disebut counter yang mana akan bertambah atau berkurang pada setiap perulangan. Syntaxnya adalah

```
For < pencacah > = <awal> To <akhir> [step<langkah>]
  Statement
Next <pencacah>
```

Keterangan :

Pencacah : Variabel (tipe : integer) yang digunakan untuk menyimpan angka perulangan.

awal : Nilai awal dari **pencacah**

akhir : Nilai akhir dari **pencacah**

langkah : Perubahan nilai **pencacah** setiap pengulangan. Sifatnya optional (boleh ditulis ataupun tidak). Bila tidak ditulis maka nilai **langkah** adalah 1.

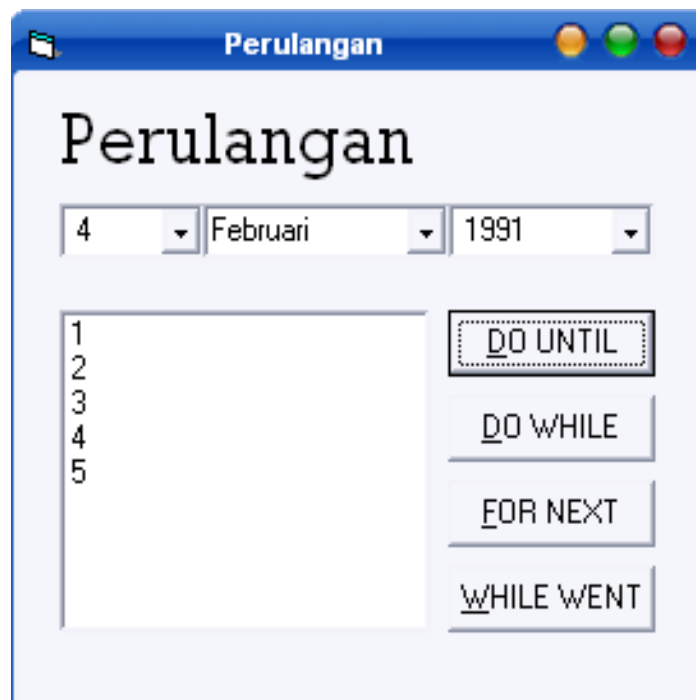
7.3. While ... Wend

While – Wend digunakan untuk **mengeksekusi suatu blok statement dengan jumlah pengulangan yang tak terhingga**. Berikut ini adalah statement While –Wend, yang dijalankan selama kondisi benar :

```
WHILE <Kondisi>  
    <Statements/Blok Kode Program>  
WEND
```

Jika **kondisi** FALSE (Zero), maka semua <Ststements/blok kode program> yang mengikitu **kondisi** akan diloncati. Program akan menjalankan <Ststements/blok kode program> jika **kondisi** benar, dan akan kembali ke **WHILE** berikutnya.

Contoh :



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Perulangan 2- CenterScreen
Label1	Caption	Perulangan
ComboBox1	Name Text	Kosongkan Kosongkan
ComboBox2	Name Text	Kosongkan Kosongkan
ComboBox3	Name Text	Kosongkan Kosongkan
Label2	Caption	Input Kata
TextBox1	Name Text	txtkata Kosongkan
List1	Name	List1
CommandButton1	Name Caption	cmddountil &DO UNTIL
CommandButton2	Name Caption	cmddowhile &DO WHILE
CommandButton3	Name Caption	cmdfornext &FOR NEXT
CommandButton4	Name Caption	cmdwhilewent &WHILE WENT

Ketentuan pengerjaan :

1. ComboBox Berisi tanggal, bulan dan tahun yang akan di pilih gunakan looping untuk membuat formatnya
2. TextBox input kata berisi data yang akan diiput dan akan diolah untuk perulangan dimana hasilnya akan di tampilkan dalam ListBox

Langkah Mengerjakan :

```

Dim i, jumlah As Variant

Private Sub Form_Load()
For i = 1 To 31
Me.cmbtgl.AddItem (Str(i))
Next i

For i = 1 To 12
Me.cmbbln.AddItem (MonthName(i))
Next i

For i = 1990 To Year(Now)
Me.cmbthn.AddItem (Str(i))
Next i
End Sub

```

Private Sub cmdddountil_Click()

```
List1.Clear
i = 1
Do Until i > 5
List1.AddItem (i)
i = i + 1
Loop
```

End Sub**Private Sub cmdddowhile_Click()**

```
List1.Clear
i = 10
Do While i > 6
List1.AddItem (i)
i = i - 1
Loop
```

End Sub**Private Sub cmdfornext_Click()**

```
List1.Clear
Dim a As Integer
a = 1
jumlah = 0
For i = 1 To 10
List1.AddItem (i)
jumlah = jumlah + a
a = a + 2
Next
```

End Sub**Private Sub cmdwhilewent_Click()**

```
List1.Clear
i = 20
While i <= 60
List1.AddItem (i)
i = i + 10
Wend
```

End Sub

BAB VIII

PROCEDURE DAN FUNCTION

Procedure adalah blok kode program yang berisi perintah-perintah untuk mengerjakan tugas tertentu. Bila di dalam kode program yang kita buat ada perintah-perintah untuk melakukan tugas yang sama di beberapa tempat, maka akan lebih baik perintah-perintah tersebut dibuat dalam sebuah procedure. Kemudian, procedure itu bisa di-‘panggil’ bila diperlukan.

Penggunaan procedure sangat menghemat penulisan kode program, karena kode-kode program yang sama di beberapa tempat cukup dibuat pada satu bagian saja. Selain itu, procedure akan memudahkan perbaikan kode program bila terjadi perubahan atau kesalahan, karena perbaikan cukup dilakukan pada satu bagian saja.

8.1 Jenis-jenis Procedure

Pada VB6 ada 4 jenis procedure, yaitu :

- **Procedure Sub** – procedure yang tidak mengembalikan nilai setelah ‘tugas’-nya selesai.
- **Procedure Function** – procedure yang mengembalikan nilai setelah ‘tugas’-nya selesai.
- **Procedure Event** – procedure untuk suatu event pada sebuah object. Digunakan di dalam class module.
- **Procedure Property** – procedure untuk mengubah (let) atau mengambil (get) nilai property pada sebuah object. Digunakan di dalam class module.

Bentuk penulisan (syntax) procedure sub :

```
[Public | Private] Sub <nama_sub>
    ([<argumen>])
    ...
    <isi procedure>
    ...
End Sub
```

Sedangkan bentuk penulisan (syntax) procedure function :

```
[Public | Private]      Function
<nama_function> ([<argumen>]) As
<tipe_data>
...
<isi procedure>
...
End Function
```

Penjelasan :

Pernyataan **[Public | Private]** menentukan ruang lingkup (scope) procedure. Sebuah procedure dengan *scope public* bisa digunakan dalam lingkup project. Sedangkan procedure dengan *scope private* hanya bisa digunakan dalam lingkup form saja.

<nama_sub> atau **<nama_function>** dibuat sebagai pengenalan procedure saat di-'panggil'. Aturan penamaan sebuah procedure sama dengan aturan penamaan sebuah variabel.

<argumen> merupakan serangkaian nilai dan tipe data yang dipakai oleh procedure untuk mengerjakan 'tugas'-nya. Sebuah procedure bisa saja tidak memakai argumen sama sekali.

Pernyataan **As <tipe_data>** pada *procedure function* menentukan tipe data nilai yang akan dikembalikan (return value) setelah 'tugas'-nya selesai.

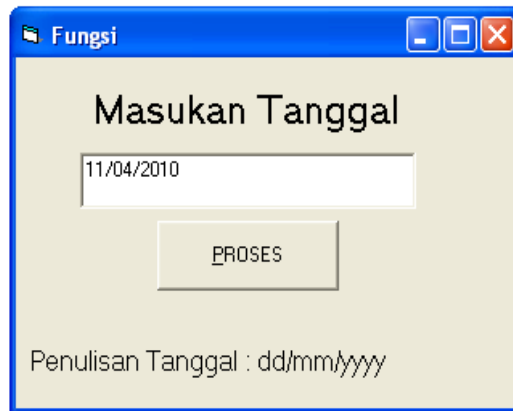
Untuk menggunakan sebuah procedure, maka procedure tersebut harus di-'panggil' pada bagian tertentu dari kode program. Procedure sub di-'panggil' dengan pernyataan :

```
Call <nama_sub>([<argumen>])
```

Sedangkan procedure function bisa di-'panggil' langsung dengan menyisipkannya di dalam kode program yang memanggilnya.

Procedure boleh ditulis dimana saja dalam kode program, tetapi biasanya ditulis dibagian atas atau bawah agar mudah ditangani.

8.2. Pembuatan Procedure dan Function



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Fungsi 2- CenterScreen
Label1	Caption	Masukan Tanggal
TextBox1	Name T0dffffpd pext	Text1 Kosongkan
CommandButton1	Name Caption	cmdproses &PROSES
Label2	Caption	Penulisan Tanggal : dd/mm/yyyy

Ketentuan Pengerjaan :

Pada saat program akan di jalankan pada Text1 Ketikan tanggal yang akan di cari harinya dengan format dd/mm/yyyy, contoh : 11/04/2010

Penyelesaian :

Function Hari(tgl As Date) As String

hari_ke = Weekday(tgl, vbSunday)

Select Case hari_ke

Case 1: Hari = "Minggu"

Case 2: Hari = "Senin"

Case 3: Hari = "Selasa"

Case 4: Hari = "Rabu"

Case 5: Hari = "Kamis"

Case 6: Hari = "Jum'at"

Case 7: Hari = "Sabtu"

End Select

End Function

Private Sub cmdproses_Click()

MsgBox "Pada Tanggal " & txthari.Text & " adalah hari " & Hari(txthari.Text), , "Hari"

txthari.SetFocus: SendKeys "{home}+{end}"

End Sub

Contoh Program Selanjutnya :

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Prosedur 2- CenterScreen
Label1	Caption	Nim
TextBox1	Name Text	Txtnim Kosongkan
TextBox2	Name Text	Txtnama Kosongkan
CommandButton1	Name Caption	cmdbersih Bersih
CommandButton2	Name Caption	cmdkeluar Keluar

Ketentuan Pengerjaan :

1. Form dirunning kemudian nim dan nama diinput
2. Button bersih untuk membersihkan TextBox dan button Keluar untuk keluar dari aplikasi dan tampil pesan informasi

Langkah Penyelesaian :

```

Sub bersih()
txtnim.Text = ""
txtnama.Text = ""
txtnim.SetFocus
End Sub

Private Sub cmdbersih_Click()
bersih
End Sub

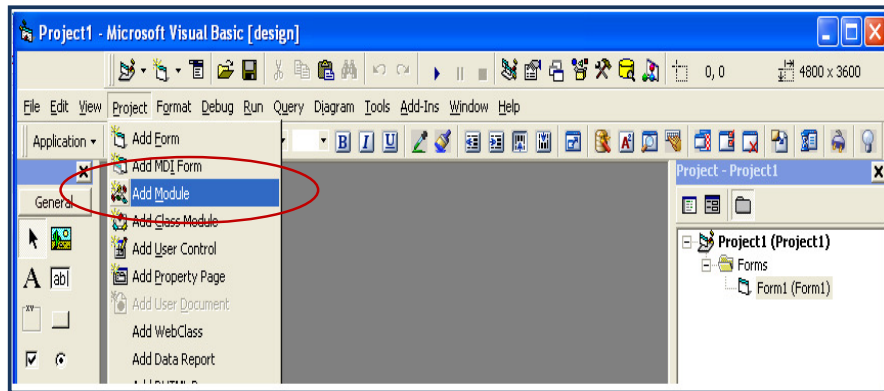
Private Sub cmdkeluar_Click()
p = MsgBox("Anda mau keluar dari form ini", vbQuestion + vbYesNo, "Info")
If p = vbYes Then
    End
End If
End Sub

```

BAB IX MODUL

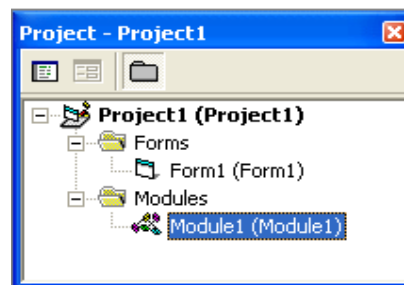
9.1 Pengenalan Modul Standar

Visual Basic menyediakan module yang dapat digunakan untuk memuat fungsi,subrutin, konstanta, variabel dan type ciptaan yang akan disediakan bagi keseluruhan project. Untuk membuat module dalam project anda dapat menggunakan perintah *Project, Add Module*.



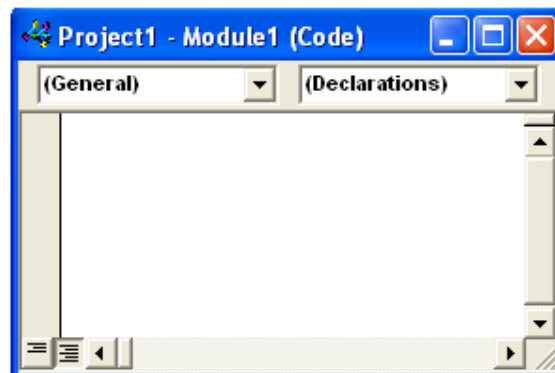
Gambar 8.1 Cara menjalankan Module

Sesaat setelah anda menambahkan module, maka pada Jendela Project Explorer akan nampak Module 1



Gambar 8.2 Module

Untuk melakukan koding, anda dapat melakukan double klik pada Module tersebut.



Gambar 8.2 Jendela Kode Module

9.2. Penggunaan Modul Standar



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Fungsi dan Modul 2- CenterScreen
Label1	Caption	Luas Alas
TextBox1	Name Text	Txtalas Kosongkan
TextBox2	Name Text	Txttinggi Kosongkan
CommandButton1	Name Caption	cmdhitung Hitung Luas Segitiga
CommandButton2	Name Caption	cmdbersih &Bersih
CommandButton2	Name Caption	cmdkeluar &Keluar

Ketentuan Program :

1. Luas alas dan tinggi diinput, Button Hitung diklik tampil hasil pada txthasil didapat dari alas X tinggi / 2, gunakan fungsi untuk menyelesaikan dan tempatkan pada module
2. Button Bersih gunakan prosedur
3. Button Keluar untuk keluar dari aplikasi dan tampil pesan

Langkah Penyelesaian :**Private Sub cmdbersih_Click()**

Call bersih

txtalas.Text = ""

End Sub**Private Sub cmdkeluar_Click()**

If MsgBox("Mau Keluar??", vbQuestion + vbOKCancel, "message") Then End

End Sub**Private Sub cmdluas_Click()**

Dim a, t As Integer

a = Val(txtalas.Text)

t = Val(txttinggi.Text)

txthasil.Text = luassegitiga(a, t)

End Sub**Public Sub bersih()**

Dim x As Control

For Each x In Me

If TypeName(x) = "TextBox" Then x.Text = ""

If TypeName(x) = "ListBox" Then x.clear

Next

End Sub**Letakan pada modul :****Public Function luassegitiga(alas, tinggi As Integer) As Double**

luassegitiga = (alas * tinggi) / 2

End Function

BAB X

ARRAY

10.1. Pengeritan Koleksi Objek

Objek-objek yang digunakan dalam form biasanya berdiri sendiri. Dengan demikian jika anda ingin mengubah property ke semua objek tersebut, hal tersebut meny ebabkan listing menjadi panjang. Untuk itu digunakan koleksi dan array yang dapat menghemat penulisan listing program. **Koleksi adalah** suatu kumpulan komponen yang dapat berfungsi sejalan. Hal ini sangat berguna saat akan mendefinisikan sebuah proses yang menyangkut kontrol-kontrol yang ada pada form.

10.2. Cara Mengakses Koleksi Objek

➤ ***Koleksi Sementara***

Dapat dilakukan dengan mengklik kontrol-kontrol pada form yang akan dijadikan sebagai satu group koleksi sambil menekan tombol **Ctrl**. Koleksi sementara biasanya digunakan untuk memindahkan objek secara bersamaan hanya pada saat tersebut.

➤ ***Koleksi dengan Controls***

Untuk mempermudah koleksi semua objek dalam satu form digunakan property controls.

Sintaknya : [Objek].controls

Dimana [objek] lebih umum dinyatakan sebagai nama form yang kontrol-kontrolnya dijadikan koleksi proses.

10.3. Array Satu Dimensi

Array adalah variable yang berisi beberapa data sejenis. Data-data tersebut disimpan dengan nomor urut tertentu yang disebut index. Jika X adalah variable array, maka bias digambarkan sebagai berikut :

Variabel X
Data-1 Elemen/index ke-1
Data-2 Elemen/index ke-2
Data-3 Elemen/index ke -3
Data-n Elemen/index ke-n

10.4. Array Multi Dimensi

Array berdimensi 2 terdiri dari baris dan kolom, contohnya pada tabel atau matrik. Pendefinisiannya adalah sebagai berikut :

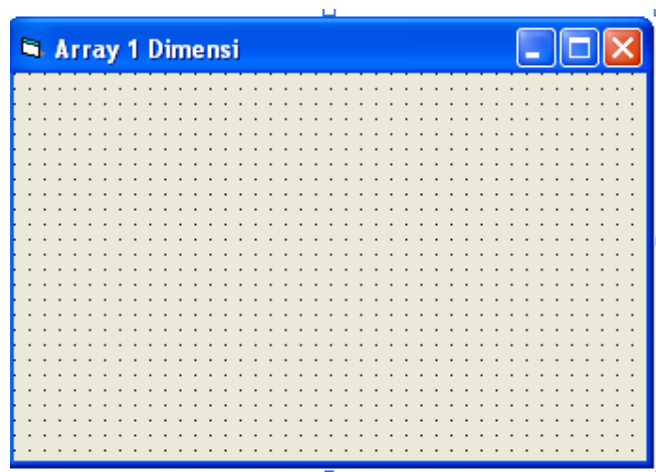
```
Dim nama_variabel (n1, n2) As tipe_variabel (n1=baris, n2=kolom)
```

10.5. Pembuatan Program Array

1. Array Satu Dimensi

Ketentuan Program :

Buat tampilan form seperti dibawah ini



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Array 1 Dimensi 2- CenterScreen

Langkah Pengerjaan :

```
Private Sub Form_Load()
Dim hari(6) As String
hari(0) = "Senin"
hari(1) = "Selasa"
hari(2) = "Rabu"
hari(3) = "Kamis"
hari(4) = "Jum'at"
hari(5) = "Sabtu"
hari(6) = "Minggu"

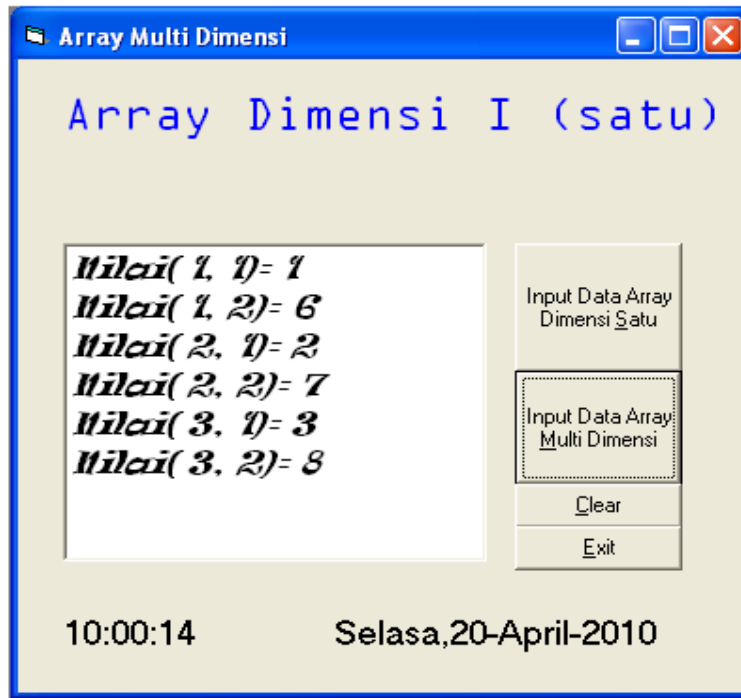
For i = 0 To 6
MsgBox "Hari " & hari(i), "Nama Hari"
Next i
End Sub
```

Tampilan Output akan mencetak seperti tampilan dibawah ini dan mengulang sebanyak jumlah hari

2. Array Multi Dimensi

Ketentuan Program :

Buat tampilan Form seperti dibawah ini :



Objek	Property	Nilai
Form1	BorderStyle Caption StartUpPosition	1-FixedSingle Array Multi Dimensi 2- CenterScreen
Label1	Caption	Kosongkan
List1	Name	List1
CommandButton1	Caption Name	Input Data Array Dimensi & Satu cmdsatudimensi
CommandButton2	Caption Name	Input Data Array & Multi Dimensi cmdmultidimensi
CommandButton3	Caption Name	&Clear Cmdclear
CommandButton4	Caption Name	&Exit Cmdexit
Label2	Caption Name	Kosongkan Ltime
Label3	Caption Name	Kosongkan Ltgl
Timer1	Interval	1000
Timer2	Interval	500

Langkah Pengerjaan :

```
General                Deklarations
Dim nama(10) As String
Dim nilai(5, 5) As Byte
Dim r As Byte
Dim m As Byte
Dim w As Byte
Dim i As Byte

Private Sub cmdmultidimensi_Click()
List1.clear
w = InputBox("Masukan Jumlah Baris", "Input Baris", 0)
i = InputBox("Masukan Jumlah Kolom", "Input Baris", 0)
For r = 1 To w
    For m = 1 To i
        nilai(r, m) = InputBox("Nilai Pada (" + Str(r) + ", " + Str(m) + ")", "Input Nilai Matrik", 0)
    Next
Next
For r = 1 To w
    For m = 1 To i
        List1.AddItem "Nilai(" + Str(r) + ", " + Str(m) + ")=" + Str(nilai(r, m))
    Next
Next
Next
End Sub

Private Sub cmdsatudimensi_Click()
List1.clear
m = InputBox("Masukan Jumlah Data Teman Anda", "Input Jumlah Data", 0)
For r = 1 To m
    nama(r) = InputBox("Masukan nama teman anda yang ke- " + Str(r), "Input Jumlah Data", "Nama")
Next
For r = 1 To m
    List1.AddItem UCase(nama(r))
Next r
End Sub

Private Sub cmdclear_Click()
Dim ctrl As Control
For Each ctrl In Me
    If TypeName(ctrl) = "ListBox" Then
        ctrl.clear
    End If
Next
End Sub
```

```
Private Sub Timer2_Timer()  
If Label1.Caption = "Array Dimensi I(satu)" Then  
    Label1.Caption = "Array Multi Dimensi"  
    Label1.ForeColor = &HFF&  
Else  
    Label1.Caption = "Array Dimensi I (satu)"  
    Label1.ForeColor = &HFF0000  
End If  
End Sub  
  
Private Sub Timer1_Timer()  
Itgl.Caption = Format(Date, "dddd,dd-mmmm-yyyy")  
ltime.Caption = Time  
End Sub  
  
Private Sub Command4_Click()  
End  
End Sub
```

Latihan :

Buat tampilan seperti dibawah ini :

Perhitungan Matrik 2 X 2

Perhitungan Matrik Dinamis

Matrik 1 Matrik 2

Perkalian
 Penambahan
 Pengurangan

1 2	3 4
1 2	3 4

Input Matrik 1 Input Matrik 2

Hasil Perhitungan Matriks

9 12
9 12

Perhitungan

Keluar

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Perhitungan Matrik 2 X 2 2- CenterScreen
Frame1	Caption	Perhitungan Matrik Dinamis
Label1	Caption	Matrik1
Label2	Caption	Matrik2
Option1	Caption	Perkalian
Option2	Caption	Penambahan
Option3	Caption	Pengurangan
TextBox1	Text Name Multiline Scrollbars	Kosongkan Matrik1 True 2-Vertical
TextBox2	Text Name Multiline Scrollbars	Kosongkan Matrik2 True 2-Vertical
CommandButton1	Caption Name	Input Matrik 1 Cmdinputm1
CommandButton2	Caption Name	Input Matrik 2 Cmdinputm2
Label3	Caption	Hasil Perhitungan Matrik
TextBox3	Text Name Multiline Scrollbars	Kosongkan hasil True 2-Vertical
CommandButton1	Caption Name	Perhitungan cmdhitung
CommandButton2	Caption Name	Keluar cmdkeluar

Ketentuan Pengerjaan :

1. Button matrik1 di klik kemudian input data untuk matrik 1 lalu Button matrik2 di klick kemudian input data untuk matrik2
2. Pilih Option Button untuk proses pengolahan matrik
3. Klik Button Perhitungan untuk melihat hasil
4. Button Keluar digunakan untuk keluar dari aplikasi

Latihan :

Tambahkan lising program untuk pengurangan dan penambahan logika program sama seperti perkalian

General **Deklarations**

```
Private m1(1, 1) As Double
Private m2(1, 1) As Double
Private hsl(1, 1) As Double
Private Sub cmdhitung_Click()
Dim i, j As Integer
Private Sub cmdhitung_Click()
Dim i, j As Integer
If optkali.Value = True Then
    kali
    For i = 0 To 1
        For j = 0 To 1
            hasil.Text = hasil.Text & " " & hsl(i, j)
        Next j
        pindahbaris$ = Chr(13) + Chr(10)
        hasil.Text = hasil.Text + pindahbaris$
    Next i
End If
End Sub
Private Sub cmdinputm1_Click()
Dim i, j As Integer
For i = 0 To 1
    For j = 0 To 1
        nilai$ = InputBox("masukkan nilai matrik ke 1", "Proses")
        pindahbaris$ = Chr(13) + Chr(10)
        matrik1.Text = matrik1.Text & " " & nilai$
        m1(i, j) = nilai$
    Next j
    matrik1.Text = matrik1.Text & " " & pindahbaris$
Next i
End Sub
```

Private Sub cmdinputm2_Click()

```
Dim i, j As Integer
```

```
For i = 0 To 1
```

```
    For j = 0 To 1
```

```
        nilai$ = InputBox("masukkan nilail matrik ke 2", "Proses")
```

```
        pindahbaris$ = Chr(13) + Chr(10)
```

```
        matrik2.Text = matrik2.Text & " " & nilai$
```

```
        m2(i, j) = nilai$
```

```
    Next j
```

```
matrik2.Text = matrik2.Text & " " & pindahbaris$
```

```
Next i
```

End Sub**Private Sub cmdkeluar_Click()**

```
End
```

End Sub**Private Sub bersih()**

```
'hapus matrik
```

```
For i = 0 To 1
```

```
    For j = 0 To 1
```

```
        hsl(i, j) = 0
```

```
    Next j
```

```
Next i
```

End Sub**Private Sub kali()**

```
Dim i, j, k As Integer
```

```
'perhitungan
```

```
For i = 0 To 1
```

```
    For j = 0 To 1
```

```
        hsl(i, j) = 0
```

```
        For k = 0 To 1
```

```
            hsl(i, j) = hsl(i, j) + (m1(i, k) * m2(k, j))
```

```
        Next k
```

```
    Next j
```

```
Next i
```

End Sub

BAB XI

WINDOWS API

11.1. Pengenalan Windows API

Windows API (Application Programming Interface) merupakan sekumpulan prosedur dan rutin yang terdapat dalam file-file pustaka windows (*library windows*) yang memiliki kemampuan mengakses kekuatan sistem. Microsoft menjadikan API sebagai jembatan satu-satunya yang disediakan untuk mengembangkan system windows.

Hampir semua fungsi Windows API terdapat dalam direktori system milik windows (biasanya terdapat di C:\Windows\System bergantung pada saat pertama kali instalasi windows). Fungsi-fungsi tersebut ditulis pada sebuah file dengan ekstensi *.dll*.

11.2. Struktur Deklarasi Windows API

Pada umumnya file *.dll* yang terinstall pada Visual Basic akan dibaca secara default lokasi filenya tanpa harus menuliskan path secara utuh. Proses pemberitahuan ke visual basic mengenai prosedur eksternal yang hendak dimanfaatkan disebut dengan "Deklarasi". Yaitu dengan menempatkan sebaris kode untuk mendeklarasikan file eksternal yang digunakan, sebagai contoh pendeklarasian fungsi API sebagai berikut:

```
Declare Function ExitWindowsEx Lib "user32" (ByVal _  
uFlags As Long, ByVal dwReserved As Long) As Long
```

Adapun penjelasan dari deklarasi fungsi tersebut adalah sebagai berikut :

1. **Declare**

Fungsi ini merupakan reserved word (kata yang telah baku yang disediakan visual basic untuk menyatakan pendeklarasian prosedur API).

2. **Function**

Fungsi ini juga merupakan sebuah reserved word dari visual basic. API menggunakan prosedur Function agar dihasilkan suatu nilai (returned value) sebagai hasil dari sebuah aksi.

3. **ExitWindowsEx**

Fungsi ini merupakan salah satu nama fungsi yang terdapat pada suatu file pustaka prosedur (*.dll). Setelah penulisan nama fungsi ini, proses harus dilanjutkan dengan nama file pustaka prosedur dimana fungsi tersebut berada.

Terkadang nama fungsi yang tertulis pada file .dll (pustaka prosedur) merupakan nama yang tidak sah (ilegal) yang digunakan visual basic. Untuk menyiasati masalah ini, visual basic mengizinkan untuk menempatkan string teks "Alias" tepat disebelah kanan nama file .dll

4. **Lib "user32"**

Disinilah fungsi untuk memberitahukan visual basic dimana letak fungsi ExitWindowsEx berada, yaitu pada library (pustaka prosedur user32.dll) file user32.dll disebut sebagai pustaka prosedur karena didalamnya terdapat puluhan bahkan ratusan macam fungsi.

Pada pendeklarasian fungsi windows API diatas file user32 tidak menyertakan ekstensinya dikarenakan file tersebut termasuk dalam tiga jenis Special Windows System Dll yang paling banyak digunakan sehingga visual basic secara otomatis akan mengenalinya tanpa harus menulis ekstensi .dll. Adapun tiga jenis Special Windows System DLL itu adalah kernel32,user32, dan gdi32.

5. **(ByVal uFlags As Long, ByVal dwReserved As Long)**

Merupakan argument yang digunakan dalam windows API sama dengan prosedur buatan pada visual basic namun memiliki perbedaan yaitu kebanyakan prosedur API memiliki argument yang membutuhkan suatu nilai tertentu yang bersifat tetap sedangkan pada prosedur visual basic sering argument menggunakan nilai referensi, yaitu nilai (value) yang dapat berubah.

6. **As Long**

Fungsi ini merupakan pendeklarasian tipe nilai yang dikembalikan (returned value) fungsi API.

Pendeklarasian fungsi API harus ditulis secara utuh dalam satu baris pernyataan. Apabila tidak memungkinkan, dapat dipisahkan dengan menggunakan sebuah karakter underscore(_) setelah spasi seperti yang telah dicontohkan pada pendeklarasian fungsi API diatas.

Deklarasi fungsi API dapat dilakukan dengan awalan Private atau Public. Awalan Public selalu ditempatkan dalam sebuah modul (*.bas) dan fungsi yang dideklarasikan pada jendela modul selalu bersifat global yang dapat dipakai pada setiap form atau listing program yang lain, sedangkan fungsi yang dideklarasikan pada jendela selain modul selalu diawali dengan private, contoh :

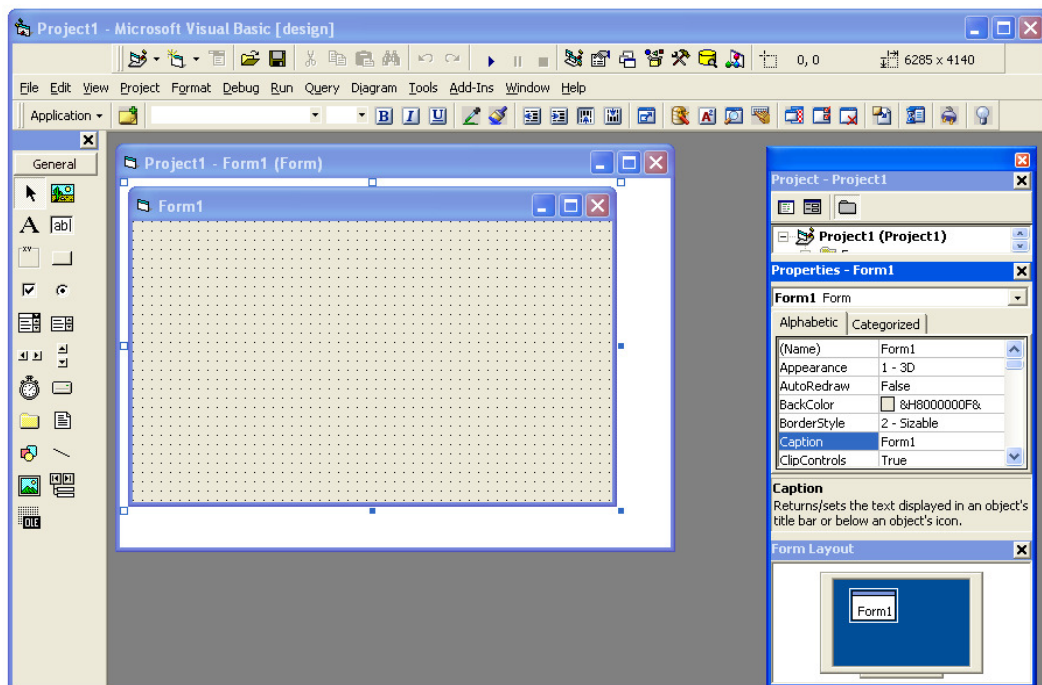
```
Public Declare Function ExitWindowsEx Lib "user32" (ByVal _  
uFlags As Long, ByVal dwReserved As Long) As Long
```

Penulisan nama fungsi atau prosedur yang digunakan dalam windows API bersifat casesensitive. Sebagai contoh visual basic akan membedakan antara penulisan ExitWindowsEx dengan exitwindowsex.

11.1. Contoh Penggunaan Windows API

Pada contoh program dibawah ini kita akan membuat sebuah form tanpa border menjadi berbentuk oval dan dapat digerakan. Normalnya sebuah form tanpa border tidak akan bisa digerakan (digeser kekanan maupun kekiri menggunakan mouse) pada saat program dijalankan.

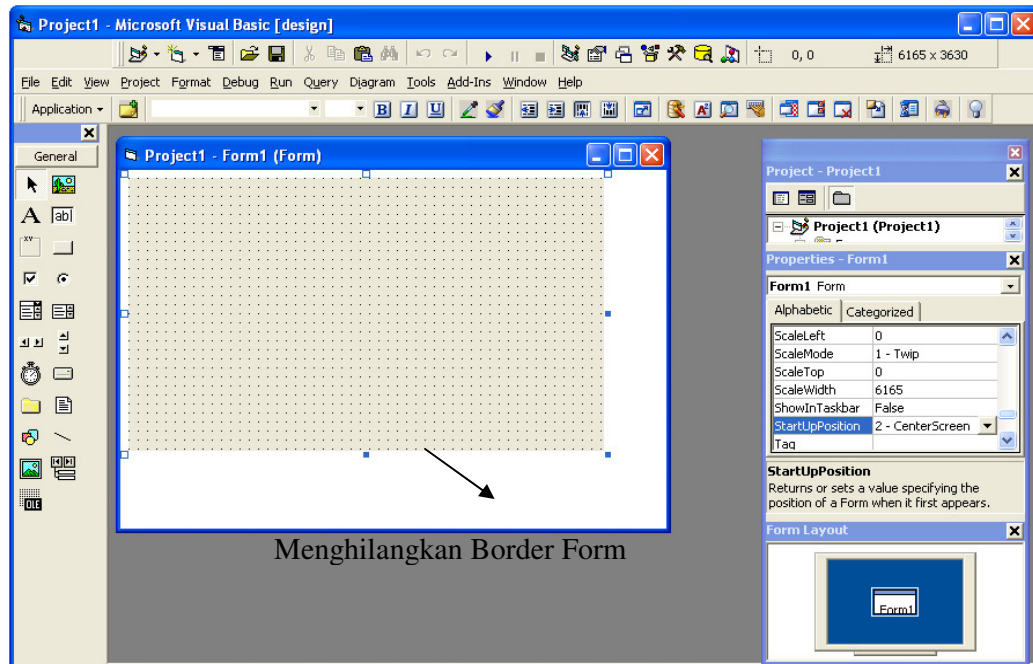
Buatlah sebuah project kemudian tambahkanlah sebuah form



Aturlah properties form seperti pada table dibawah ini :

Property Item	Nilai Property
BorderStyle	0 – None
StartPosition	2 – CenterScreen

Sehingga tampilan form akan menjadi seperti gambar dibawah ini :



Berpindahlah ke jendela kode kemudian buatlah listing kode seperti dibawah ini :

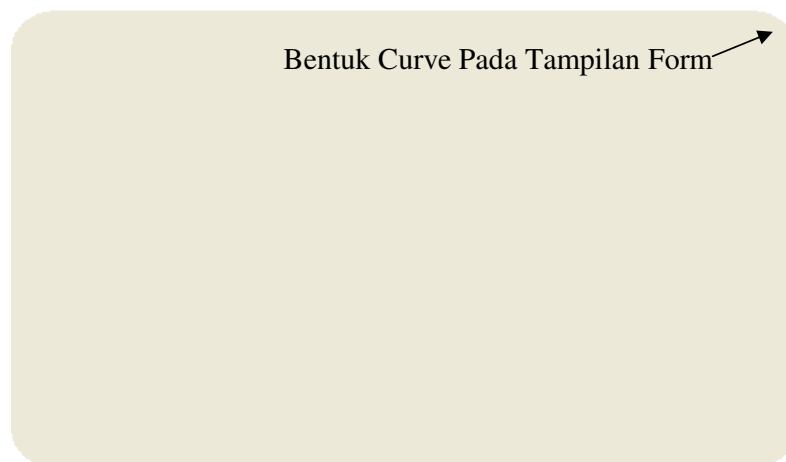
```
'win api untuk membuat lekukan diform
Private Declare Function SetWindowRgn Lib _
    "user32" (ByVal hwnd As Long, _
    ByVal hRgn As Long, ByVal bRedraw As Long) As Long

Private Declare Function CreateRoundRectRgn Lib _
    "gdi32.dll" (ByVal X1 As Long, ByVal Y1 As Long, _
    ByVal X2 As Long, ByVal Y2 As Long, _
    ByVal X3 As Long, ByVal Y3 As Long) As Long

'win api agar form tanpa border dapat digerakan
Private Declare Function SendMessage Lib _
    "user32" Alias "SendMessageA" (ByVal hwnd As Long, _
    ByVal wParam As Long, ByVal lParam As Long, _
    lParam As Any) As Long
Private Declare Function DeleteObject Lib "gdi32" _
    (ByVal hObject As Long) As Long
```

```
Private Declare Sub ReleaseCapture Lib "user32" ()  
Private Const WM_NCLBUTTONDOWN = &HA1  
Private Const HTCAPTION = 2  
  
'sub untuk membuat lakukan diform  
Private Sub buatoval(Frm As Form, Optional ByVal Kurva As Double = 50)  
    Dim hRgn As Long  
    Dim X1 As Long, Y1 As Long  
    X1 = Frm.Width / Screen.TwipsPerPixelX  
    Y1 = Frm.Height / Screen.TwipsPerPixelY  
    hRgn = CreateRoundRectRgn(0, 0, X1, Y1, Kurva, Kurva)  
    SetWindowRgn Frm.hwnd, hRgn, True  
    DeleteObject hRgn  
End Sub  
  
Private Sub Form_Load()  
    buatoval Me  
End Sub  
  
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As Single, _  
Y As Single)  
    If Button = 1 Then  
        Call ReleaseCapture  
        Call SendMessage(Me.hwnd, WM_NCLBUTTONDOWN, HTCAPTION, 0&)  
    End If  
End Sub
```

Listing kode diatas apabila dijalankan akan menghasilkan sebuah form dengan tampilan seperti gambar berikut ini :



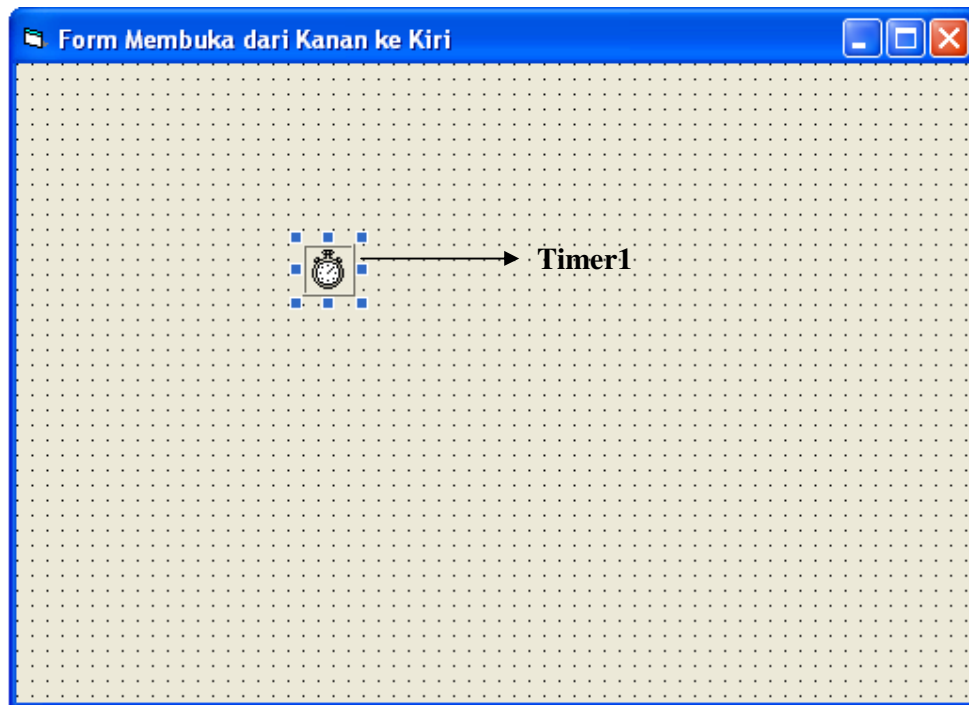
BAB XII

EFFECT FORM DAN KONTROL KOMPONEN

Pada pertemuan ini kita akan membahas masalah tentang animasi sederhana yang berkaitan dengan gerakan object. Dalam animasi ini kita lebih banyak bermain timer (berikut nilai intervalnya di properti)

12.1 Form membuka dari kiri ke kanan

Animasi ini untuk membuka form mulai dari kiri ke kanan ketika program dijalankan. Buatlah rancangan form dibawah ini.



Gambar 12.1 Rancangan Form Animasi

Kemudian ketik listing kode berikut ini:

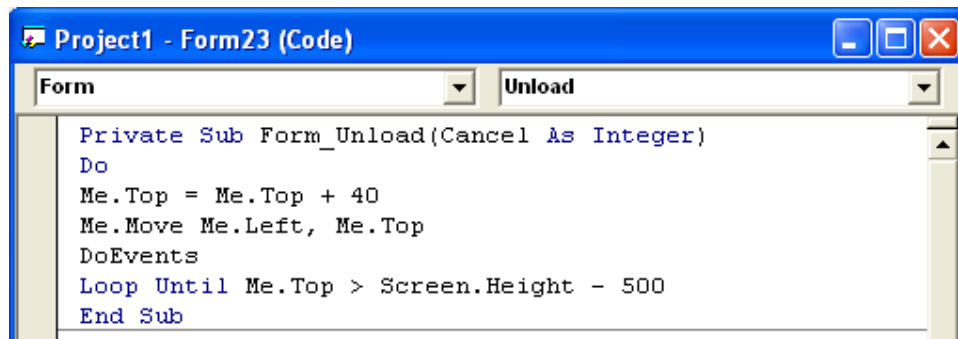
```
Private Sub Timer1_Timer()  
Me.Width = Me.Width + 400  
If Me.Width >= 9000 Then  
    Timer1.Enabled = False  
End If  
End Sub
```

Jangan lupa intervalnya di isi dengan 1



Gambar 12.2 *Form Animasi ketika dijalankan*

Kita juga bisa menambahkan animasi untuk menutup form. Kita tinggal menambahkan listing kode dibawah ini pada form **unload**.

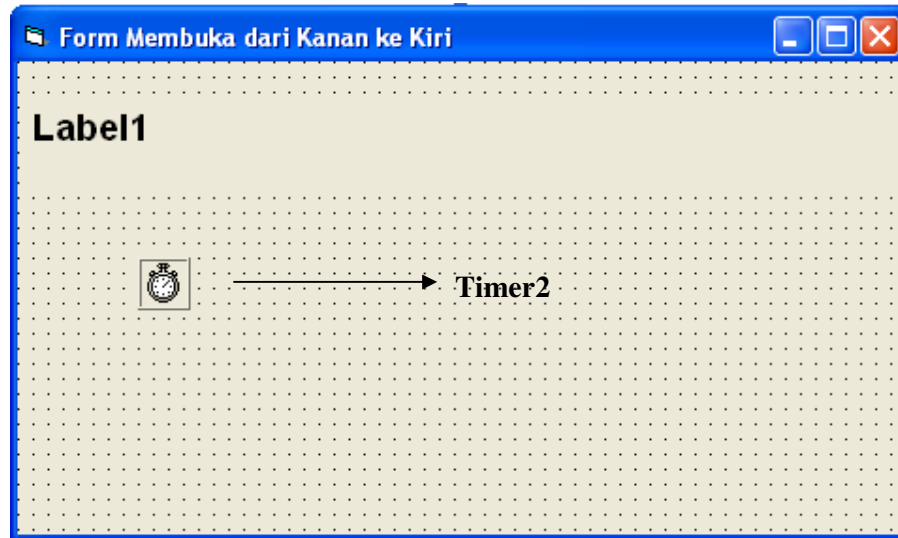
A screenshot of the Visual Basic code editor window titled "Project1 - Form23 (Code)". The "Form" dropdown menu is set to "Form" and the "Unload" dropdown menu is set to "Unload". The code in the editor is as follows:

```
Private Sub Form_Unload(Cancel As Integer)
Do
Me.Top = Me.Top + 40
Me.Move Me.Left, Me.Top
DoEvents
Loop Until Me.Top > Screen.Height - 500
End Sub
```

Gambar 12.3 *Form Animasi untuk menutup program*

12.2. Text Muncul berjalan

Pada form ini, kita akan membuat animasi text berjalan. Caranya kita rancang dulu form dibawah ini kemudian ikuti langkah2 selanjutnya.



Gambar 12.4 *Form Animasi text muncul berjalan*

```
Private Sub Timer2_Timer()
```

```
Timer2.Interval = 100
```

```
i = i + 1
```

```
Label1 = Left(Str1, i)
```

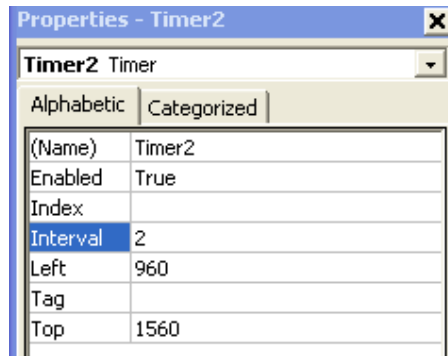
```
If i = Len(Str1) Then
```

```
i = 1
```

```
Timer2.Interval = 3000
```

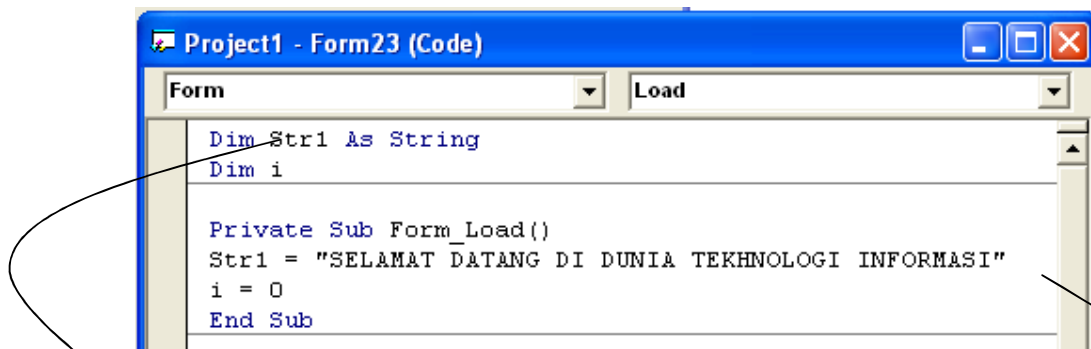
```
End If
```

```
End Sub
```



Gambar 12.5 *Properties Timer 2*

Yang harus diperhatikan disini adalah kita harus mendeklarasikan terlebih dahulu variabel yang akan menjadi perulangannya. Untuk lebih jelasnya bisa dilihat listing kode berikut ini.

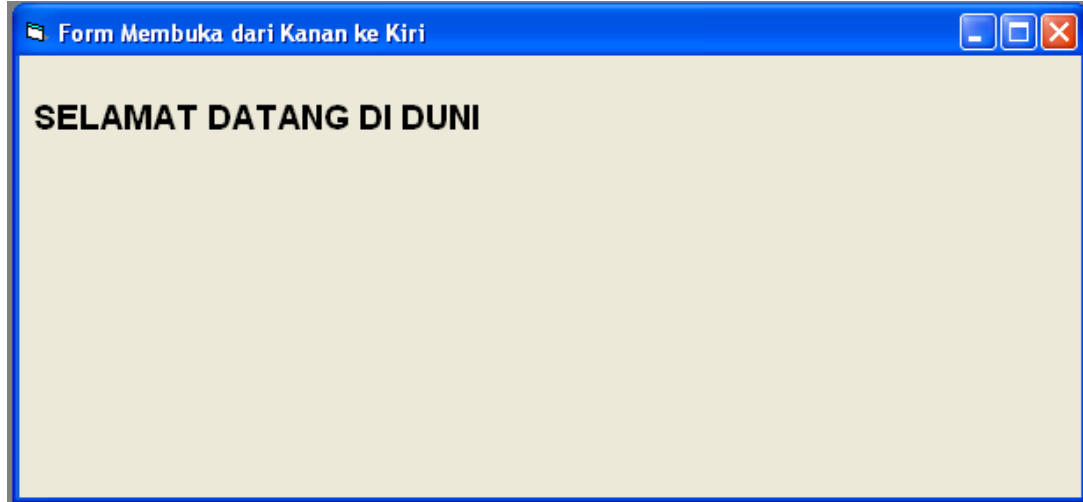


Gambar 12.6 *Properties Timer 2*

Deklarasi variabel **str1** dan **i**

Tambahkan str1 dengan kata yang anda inginkan

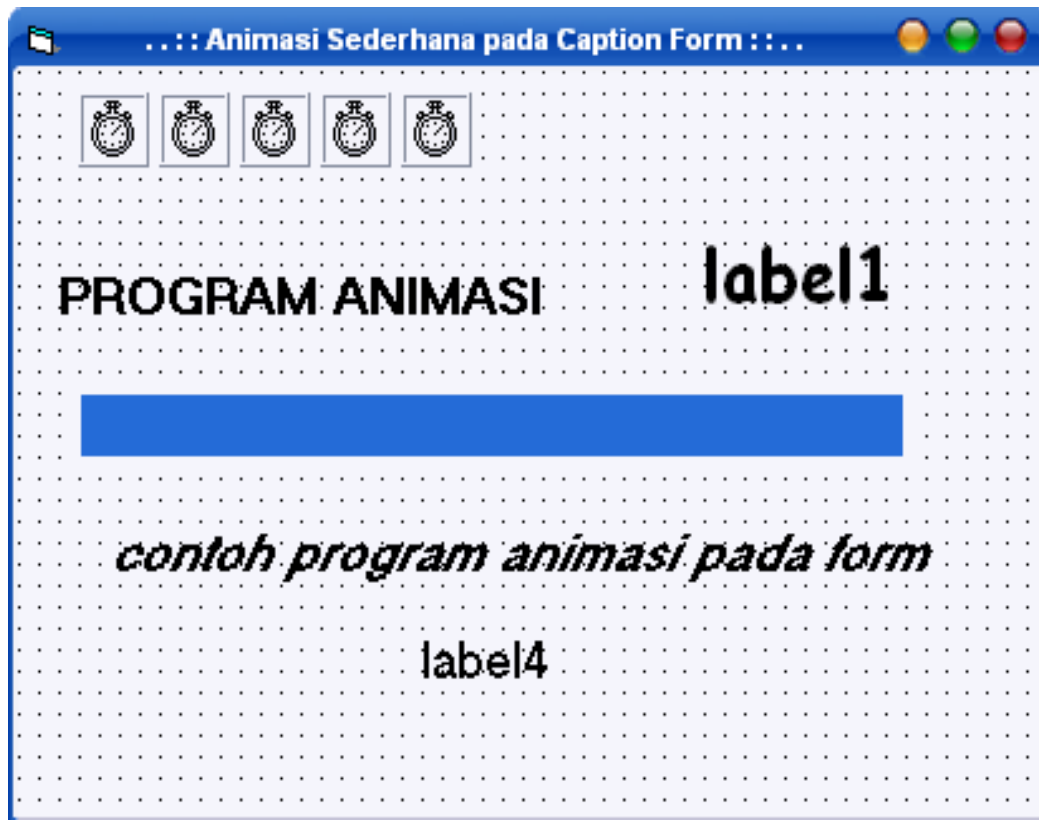
Hasil form ketika dijalankan




Gambar 12.7 Form hasil text muncul berjalan

12.3. Penggunaan Effect Form

1. Design Tampilan seperti dibawah ini:

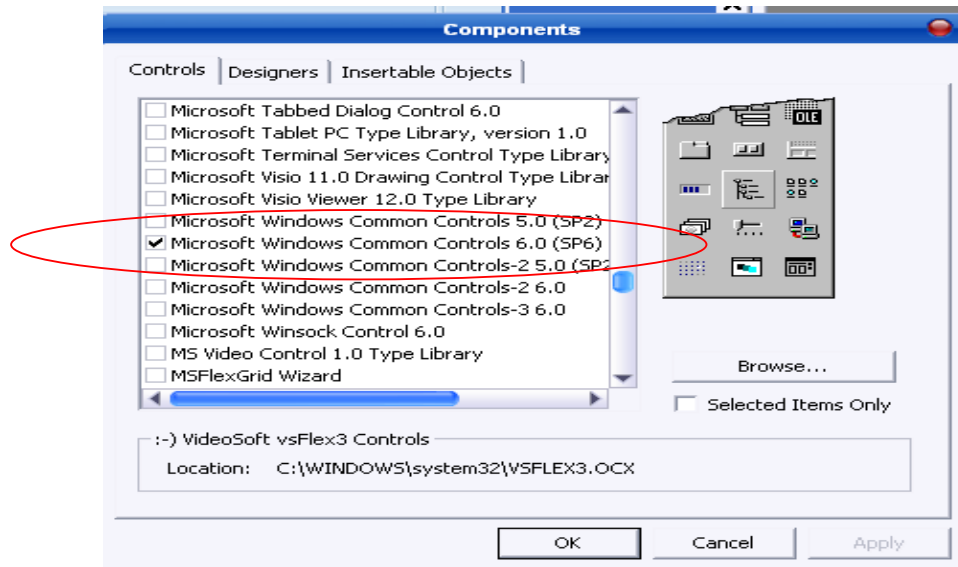


Gambar 12.8 Form Animasi

Objek	Property	Nilai
Form1	BorderStyle	1-FixedSingle
	Caption	...: Animasi Sederhana pada Caption Form : ...
Form1	StartPosition	2- CenterScreen
	Name	Form_Splash
Label1	Caption	Kosongkan
Label2	Caption	Contoh program animasi pada form
Label3	Caption	PROGRAM ANIMASI
Label4	Caption	Kosongkan
ProgressBar 	Name	Pb
	Scrolling	1 – ccScrollingSmooth
Timer1	Interval	100
Timer2	Interval	300
Timer3	Interval	100
Timer4	Interval	100
Timer5	Interval	400

Ketentuan Untuk ProgressBar :

- Klik Kanan pada ToolBox > Components
- Tampil Tampilan Berikut, lalu beri check sesuai dengan tampilan dibawah ini:



Gambar 12.9 Letak Component untuk PrograssBar

```
Option Explicit
Dim X, Y, m, n, i As Integer
Dim nama(6) As String
Dim z As Boolean

Private Sub Timer1_Timer()
If Y < 100 Then
    Y = Y + 1
    Label1.Caption = "(" & Y & "%)"
End If
pb.Value = pb.Value + 1
If pb.Value = pb.Max Then
Unload Me
End If
End Sub

Private Sub Timer2_Timer()
If Label3.Visible = True Then
    Label3.Visible = False
ElseIf Label3.Visible = False Then
    Label3.Visible = True
End If
Label2.Caption = Right(Label2.Caption, Len(Label2.Caption) - 1) & Left(Label2.Caption, 1)
End Sub

Private Sub Timer4_Timer()
Form_Splash.BackColor = RGB(Rnd * 255, Rnd * 255, Rnd * 255)
End Sub

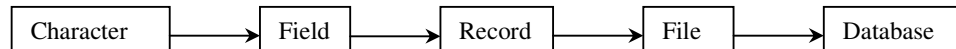
Private Sub Timer5_Timer()
Form_Splash.Caption = Right(Form_Splash.Caption, Len(Form_Splash.Caption) - 1) &
Left(Form_Splash.Caption, 1)
End Sub

Private Sub Timer3_Timer()
If z = False Then
    m = m + 1
    If m = 18 Then z = True
Else
    Label4.Visible = False
    m = 1
    z = False
    Label4.Visible = True
    n = n + 1
End If
Select Case n
Case 0
    Label4.Caption = " contoh animasi"
    Label4.FontName = "arial"
    Label4.FontItalic = True
Case 1
    Timer3.Enabled = True
End Select
If m <= 0 Then m = 1
Label4.FontSize = m
End Sub
```

BAB XIII DATABASE

1.3.1 Pengertian Database

Database adalah kumpulan beberapa data yang saling berhubungan berdasarkan kode-kode tertentu sehingga membentuk sebuah sistem. Secara hirarki database bisa digambarkan sebagai berikut :



Character sebagai satuan data terkecil dikumpulkan menjadi field (kolom), kumpulan field membentuk sebuah record (baris), beberapa record disimpan dalam file (tabel), dan kumpulan file akan membentuk sebuah database.

13.2 Pembuatan Database

Kemudahan Menggunakan microsoft Access sebagai pengolah database

- Dapat melakukan proses penyortiran dan pengatuan data
- Pembuatan label data serta laporan
- Pembuatan data kegiatan sehari-hari misalnya untuk menampung daftar pelanggan, pendataan karyawan dan lain sebagainya

13.3 Membuat Database dan Tabel

Database : kumpulan dari beberapa tabel

Tabel : kumpulan dari beberapa record

Record : isian data yang dimiliki oleh field

Field : atribut yang dimiliki oleh suatu tabel

Character : hanya terdiri dari satu angka atau huruf

Contoh : Tabel Mahasiswa Field

No	Nim	Nama	Jurusan
1	12091234	Amirudin	MI
2	11094321	Cinta Laura	KA
3	13095678	Sri Lestari	TK

Record

Tabel 13.1 Struktur Tabel

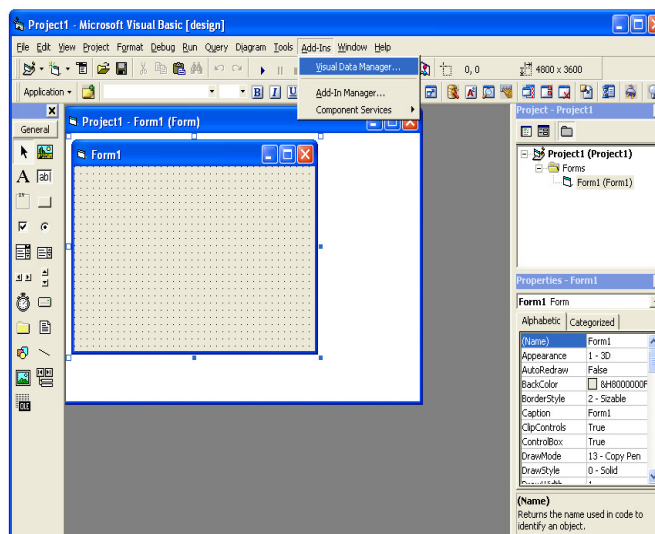
13.4. Langkah Pembuatan Database Dengan Visdata

Microsoft Visual Basic 6.0 dapat memproses file database yang dibuat dengan program database yang sudah ada pada Microsoft Access, Dbase, Foxpro, Paradox, dan lain-lain. Komponen Visual Basic untuk membuat dan memproses file database tersebut dinamakan **Visual Data Manager (Visdata)**.

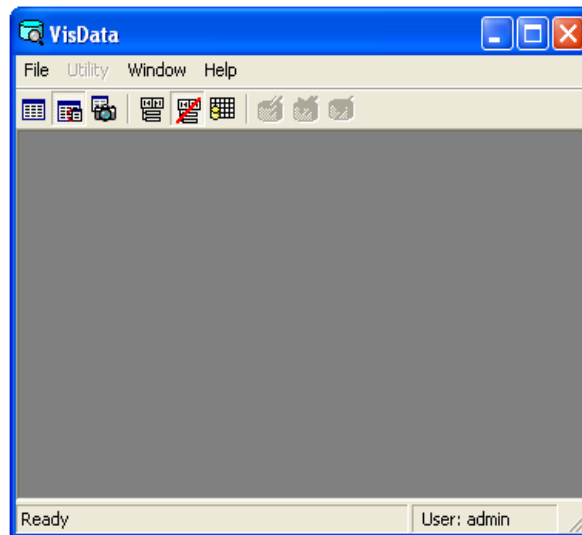
Membuat database Penjualan.Mdb

Untuk membuat database dengan Visdata dengan menggunakan format Microsoft Access dengan langkah-langkah sebagai berikut :

1. Apabila program Microsoft Visual Basic 6-0 belum di jalankan, klik tombol **start** di Taskbar dan jalankan program Visual Basic 6.0
2. Dari jendela Visual Basic pilih menu **Add-Ins** kemudian pilih Visual Data Manager (Visdata). Lihat Gambar 13.5.1 dan Gambar 13.5.2

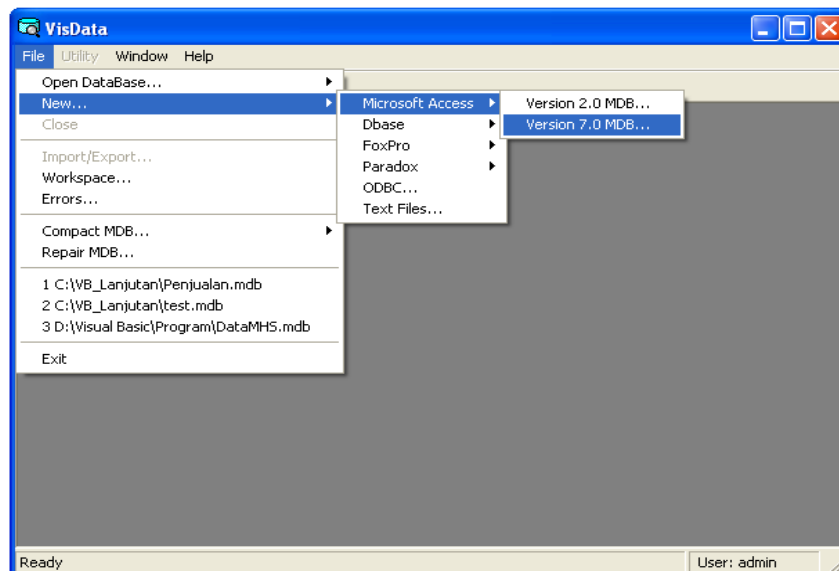


Gambar 13.4.1 Langkah Pembuatan Database

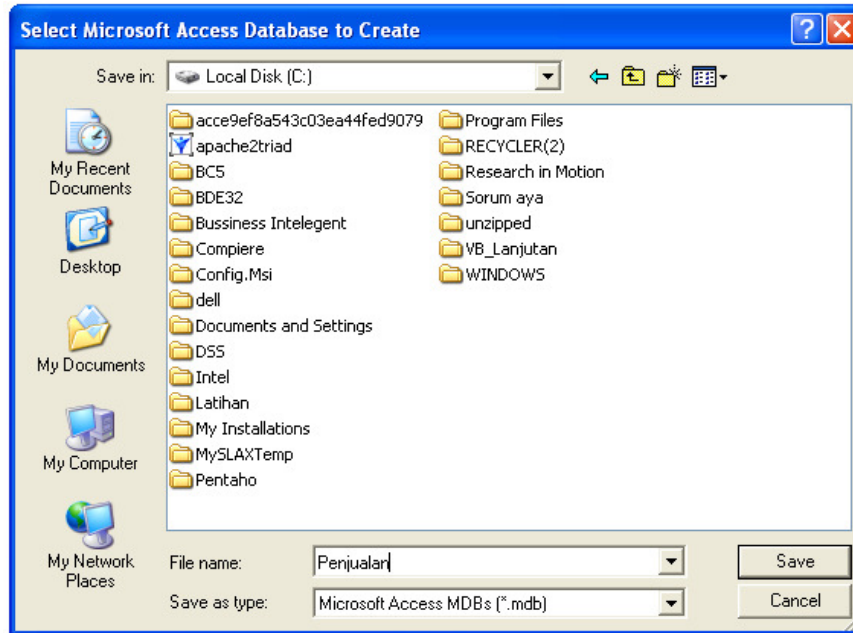


Gambar 13.4.2 Tampilan Visdata

3. Pilih menu **File**, lalu **New** (Gambar 15.3.3)
 4. Pilih menu **Microsoft Access**, pilih **Version 7.0 MDB**
 5. Selanjutnya akan ditampilkan kotak dialog “**Select Microsoft Access Database to Create**” seperti pada gambar 15.3.4. Pilihlah folder atau direktori penyimpanan pada kotak **Save in** untuk menyimpan database yang akan dibuat, lalu ketikkan **Penjualan** pada kotak file name
- Dalam pembuatan satu database anda dapat membuat lebih dari satu table query



Gambar 13.4.3. Membuat database



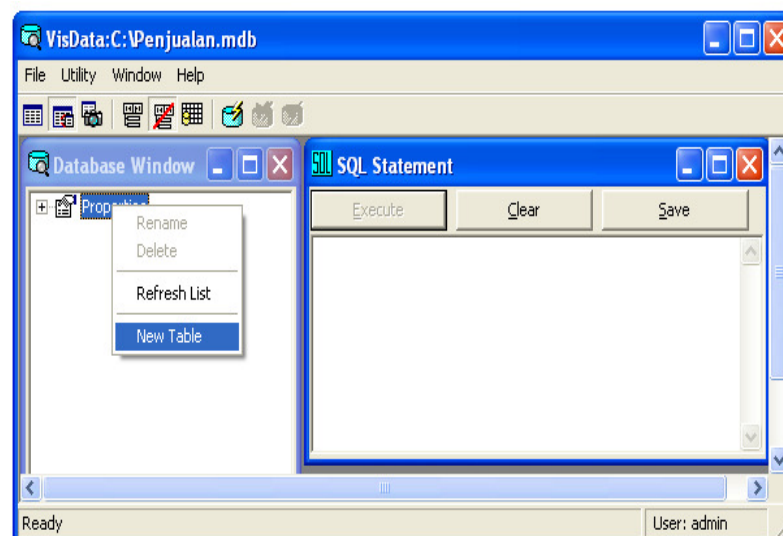
Gambar 13.4.4 Penyimpanan Database

13.5. Membuat Tabel

Untuk membuat table barang, langkah-langkahnya sebagai berikut :

1. Klik tombol mouse sebelah kanan pada jendela database (**Database Windows**), kemudian dari **PopUp Menu** yang tampak pilih **New Table**.

Lihar gambar 13.6.1



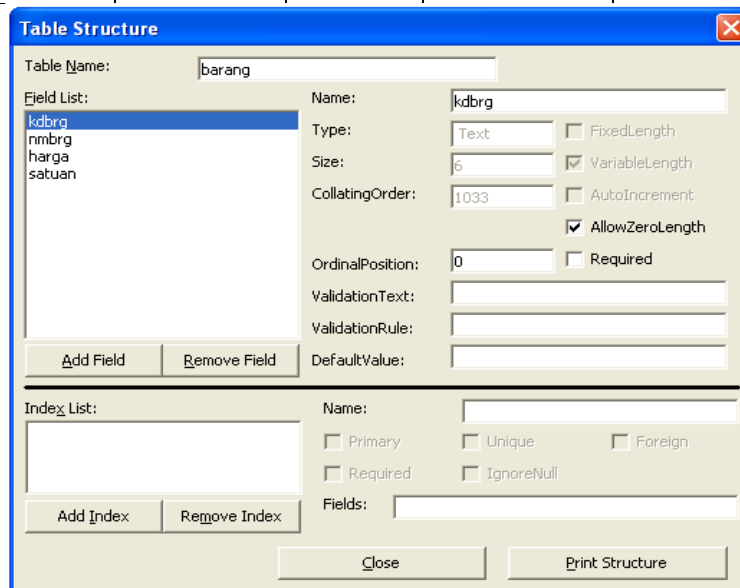
Gambar 13.6.1 Membuat Tabel

Dari **Tabel Structure** (Gambar 13.6.2) ketikkan nama tabelnya="Mahasiswa" pada kotak **Table Name**

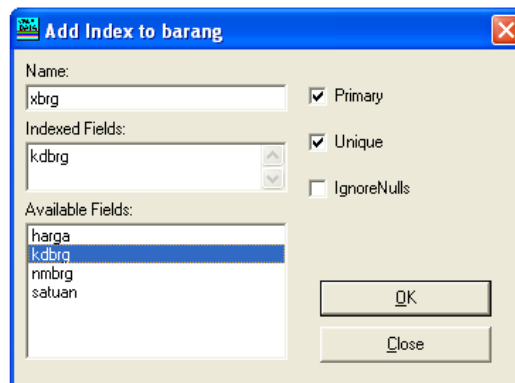
2. Klik tombol **Add Field** untuk membuat field baru dan buatlah field sebagai berikut:

Tabel Barang

Name	Type	Size	Keterangan	Index
Kdbrg	Text	6	<i>Primery Key</i>	Xkdbrg
Nmbrg	Text	20		
Harga	Currency	8		
Satuan	Text	5		

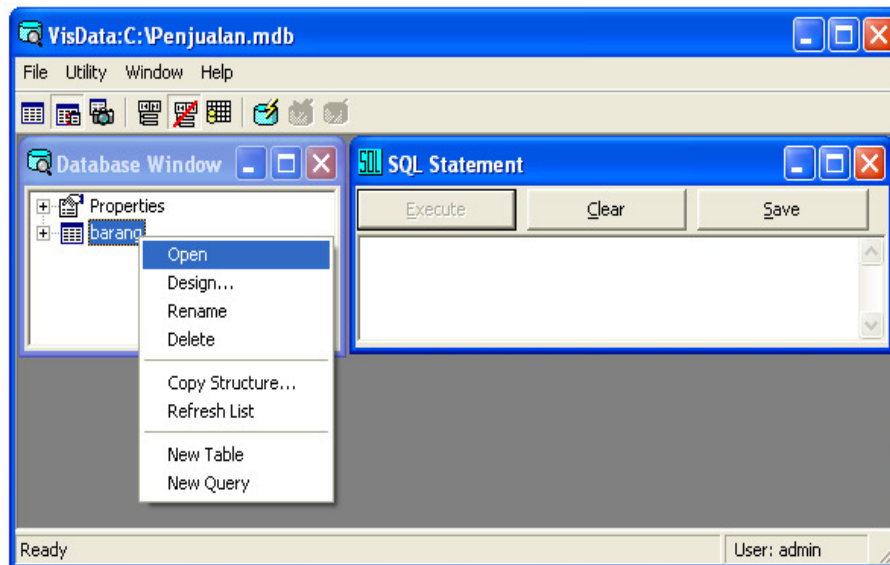


Gambar 13.6.2 Membuat tabel Barang



Gambar 13.6.3 Membuat index pada tabel Barang

3. Setelah selesai klik **OK** lalu klik Close, sampai muncul kembali tampilan seperti pada gambar 13.6.3
4. Kemudian untuk membuat index, klik tombol **Add Index** sampai muncul tampilan gambar 13.6.3. Lalu buatlah index pada tabel Barang seperti gambar 13.6.3
5. Klik **Build the Table** untuk menyimpan struktur table yang telah anda buat
6. Klik **Close** untuk kembali ke database window, sampai muncul tampilan Gambar 13.6.1
7. Untuk penginputan data dapat dilakukan dengan cara double klik di nama tabel atau Klik kanan, lalu pilih Open pada tabel yang akan di input seperti gambar 13.6.4



Gambar 13.6.4 Mengisi Record pada Tabel Barang

8. Kemudian klik tombol **Add** untuk penginputan datanya dan Tombol **Update** untuk proses penyimpanan

Input data tabel barang seperti data berikut :

ode Barang	Nama Barang	Harga	Satuan
MN-001	Monitor LG 15"	500000	Buah
KY-003	Keyboard Komic	25000	Buah
MS-001	Mouse Optik	55000	Buah

9. Jika anda ingin keluar dari Visual Data Manager, pilih menu **File** lalu **Exit**
10. Selanjutnya buatlah tabel Customer, dan Tabel Transaksi dengan struktur sebagai berikut : (Ulangi langkah diatas 1 s/d 9 untuk pembuatan tabel)

Tabel Customer

Name	Type	Size	Keterangan	Index
KdCust	Text	6	<i>Primery Key</i>	Xkdcust
NmCust	Text	25		
Telp	Text	15		
Alamat	Text	30		

Kemudian input datanya sebanyak 5 record

Tabel Transaksi

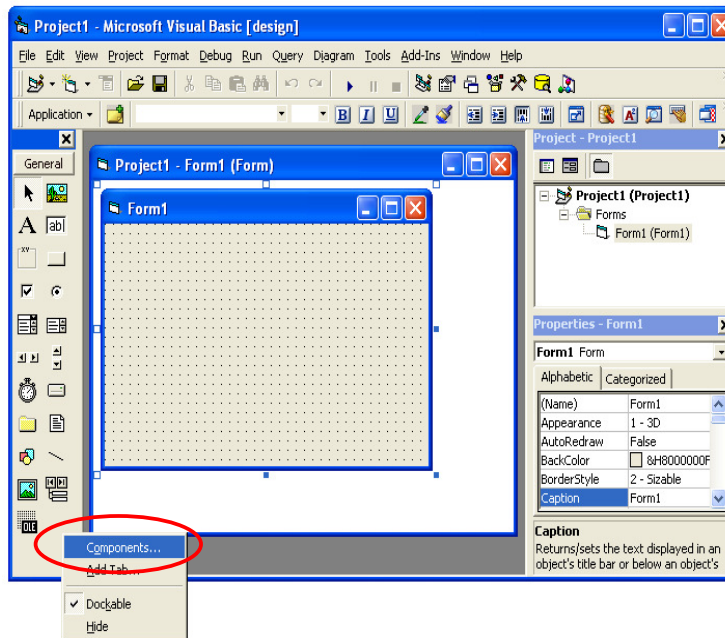
Name	Type	Size	Keterangan	Index
NoTrans	Text	5	<i>Primary Key</i>	xnotrans
TglTrans	Date/Time	8		
KdCust	Text	6		
KdBrg	Text	6		
JmlBeli	Integer	3		
Total	Currency	8		

13.6. Pembuatan Form

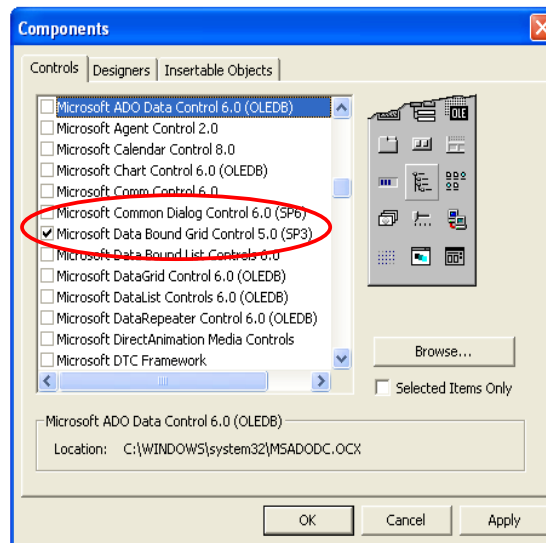
13.6.1. Membuat Form Barang

Form Barang ini dibuat untuk mengelola data barang, langkah-langkah pembuatannya sebagai berikut :

1. Tambahkan sebuah form baru dengan format sebagai berikut :
 Name : FBarang
 Caption : Form Barang
2. Pada form tersebut tambahkan objek DBGRID1, klik kanan pada Toolbox kemudian pilih Components (lihat gambar 13.6.1)., kemudian pilih **Microsoft Data Bound Grid Control 5.0(SP3)**

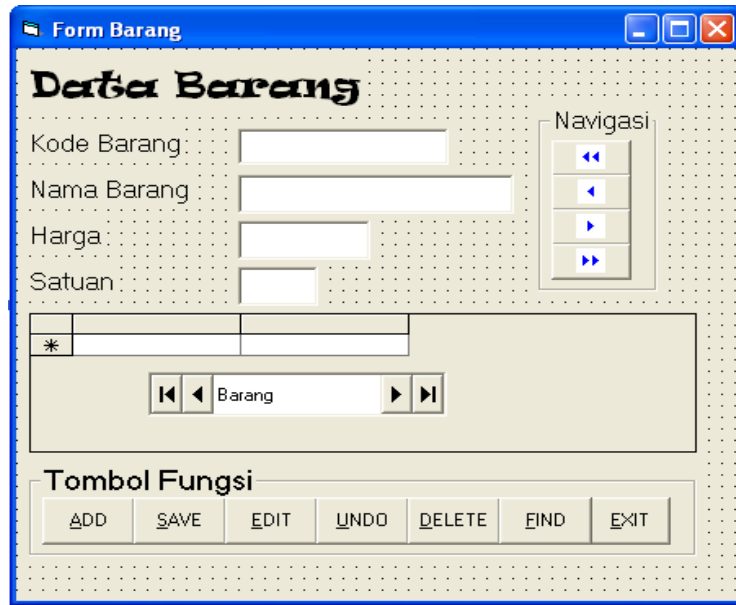


Gambar 13.6.1. Menampilkan komponen tambahan



Gambar 13.6.1. Menampilkan komponen Microsoft data bound Grid

3. Tambahkan objek berikut ini dan aturlah posisinya seperti pada gambar



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Form Barang 2- CenterScreen
Label1	Caption	Data Barang
Label2	Caption	Kode Barang
TextBox1	Name Text	Txtkdbrg Kosongkan
Label3	Caption	Nama Barang
TextBox2	Name Text	Txtnmbrg Kosongkan
Label4	Caption	Harga
TextBox3	Name Text	Txthrg Kosongkan
Label5	Caption	Satuan
TextBox4	Name Text	Txtsatuan Kosongkan
Frame1	Caption	Navigasi
CommandButton1	Caption Name Picture Style	Kosongkan cmdtop C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Graphics\Top 1-Graphical
CommandButton2	Caption Name Picture	Kosongkan cmdprev C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Graphics\Previous

	Style	1-Graphical
CommandButton3	Caption Name Picture Style	Kosongkan cmdnext C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Graphics\Next 1-Graphical
CommandButton4	Caption Name Picture Style	Kosongkan cmdbott C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Graphics\Bottom 1-Graphical
Data1	Name Caption Visible	Dtbrg Barang False
DBGrid1	Data Source	Dtbrg
Frame2	Caption	Tombol Fungsi
CommandButton5	Caption Name	&ADD Cmdadd
CommandButton6	Caption Name	&SAVE Cmdsave
CommandButton7	Caption Name	&EDIT Cmdedit
CommandButton8	Caption Name	&UNDO CmdUNDO
CommandButton9	Caption Name	&DELETE Cmddelete
CommandButton10	Caption Name	&FIND Cmdfind
CommandButton11	Caption Name	&EXIT Cmdexit

Koneksi Database Menggunakan Objek

Objek	Properties	Keterangan
Data1	Databasename Record Source RecordsetType	Cari tempat penyimpanan database anda Barang (Nama tabel) 0-Table
DBGrid1	Data Source	Dtbrg
Txtdbrg	Data Source Data Field	Dtbrg kdbrg
Txtnmbrg	Data Source Data Field	Dtbrg nmbrg
Txtharga	Data Source Data Field	Dtbrg harga
Txtsatuan	Data Source Data Field	Dtbrg satuan

Listing Program

```
General           Declarations  
Dim x As Control  
  
Private Sub cmdbott_Click()  
On Error Resume Next  
dtbrg.Recordset.MoveLast  
MsgBox "data sudah diakhir record!", 16, "Informasi"  
End Sub  
  
Private Sub cmdnext_Click()  
On Error Resume Next  
dtbrg.Recordset.MoveNext  
If dtbrg.Recordset.EOF Then  
    dtbrg.Recordset.MoveLast  
    MsgBox "Sudah diakhir record", vbCritical, "Informasi"  
End If  
End Sub  
  
Private Sub cmdprev_Click()  
On Error Resume Next  
dtbrg.Recordset.MovePrevious  
If dtbrg.Recordset.BOF Then  
    dtbrg.Recordset.MoveFirst  
    MsgBox "Sudah diawal record", vbCritical, "Informasi"  
End If  
End Sub  
  
Private Sub cmdtop_Click()  
On Error Resume Next  
dtbrg.Recordset.MoveFirst  
MsgBox "data sudah diawal record!", 16, "Informasi"  
End Sub  
  
Private Sub cmdadd_Click()  
kosong  
aktif  
txtkdbrg.SetFocus  
cmdadd.Enabled = False  
cmdsave.Enabled = True  
End Sub
```

Private Sub cmdsave_Click()

```
On Error Resume Next
With dtbrg.Recordset
.AddNew
!kdbrg = txtkdbrg.Text
!nmbrg = txtnmbrg.Text
!harga = Val(txtharga.Text)
!satuan = txtsatuan.Text
.Update
.MoveLast
nonaktif
On Error GoTo 0
cmdadd.Enabled = True
cmdsave.Enabled = False
End With
End Sub
```

Private Sub cmdedit_Click()

```
dtbrg.Recordset.Edit
aktif
txtkdbrg.Enabled = False
txtnmbrg.SetFocus
cmdadd.Enabled = True
cmdsave.Enabled = True
cmdedit.Enabled = True
End Sub
```

Private Sub cmdundo_Click()

```
p = MsgBox("Yakin akan membatalkan penginputan ??", vbQuestion + vbOKCancel, "Konfirmasi")
If p = vbOK Then
kosong
nonaktif
End If
cmdsave.Enabled = False
cmdadd.Enabled = True
End Sub
```

Private Sub cmddelete_Click()

```
p = MsgBox("Yakin ingin menghapus record ini ??", vbQuestion + vbOKCancel, "Konfirmasi")
If p = vbOK Then
dtbrg.Recordset.Delete
dtbrg.Recordset.MovePrevious
nonaktif
End If
End Sub
```

Private Sub cmdfind_Click()

```
Dim a As String * 6
```

```
a = InputBox("Ketik kode yang akan dicari :", "Pencarian")
```

```
cari = "Kode=" + a + ""
```

```
On Error Resume Next
```

```
dtbrg.Recordset.Index = "xkdbrg"
```

```
dtbrg.Recordset.Seek "=", cari
```

```
If dtbrg.Recordset.NoMatch Then
```

```
    MsgBox "record kode barang" + cr + "tidak ada", 16, "Informasi"
```

```
End If
```

```
End Sub
```

Private Sub cmdexit_Click()

```
p = MsgBox("Yakin ingin keluar??", 32 + 4, "Informasi")
```

```
If p = 6 Then
```

```
    Unload Me
```

```
End If
```

```
End Sub
```

Private Sub Form_Load()

```
cmdsave.Enabled = False
```

```
nonaktif
```

```
End Sub
```



```
Private Sub kosong()  
For Each x In Me  
If TypeName(x) = "TextBox" Then  
x.Text = ""  
End If  
Next  
End Sub
```

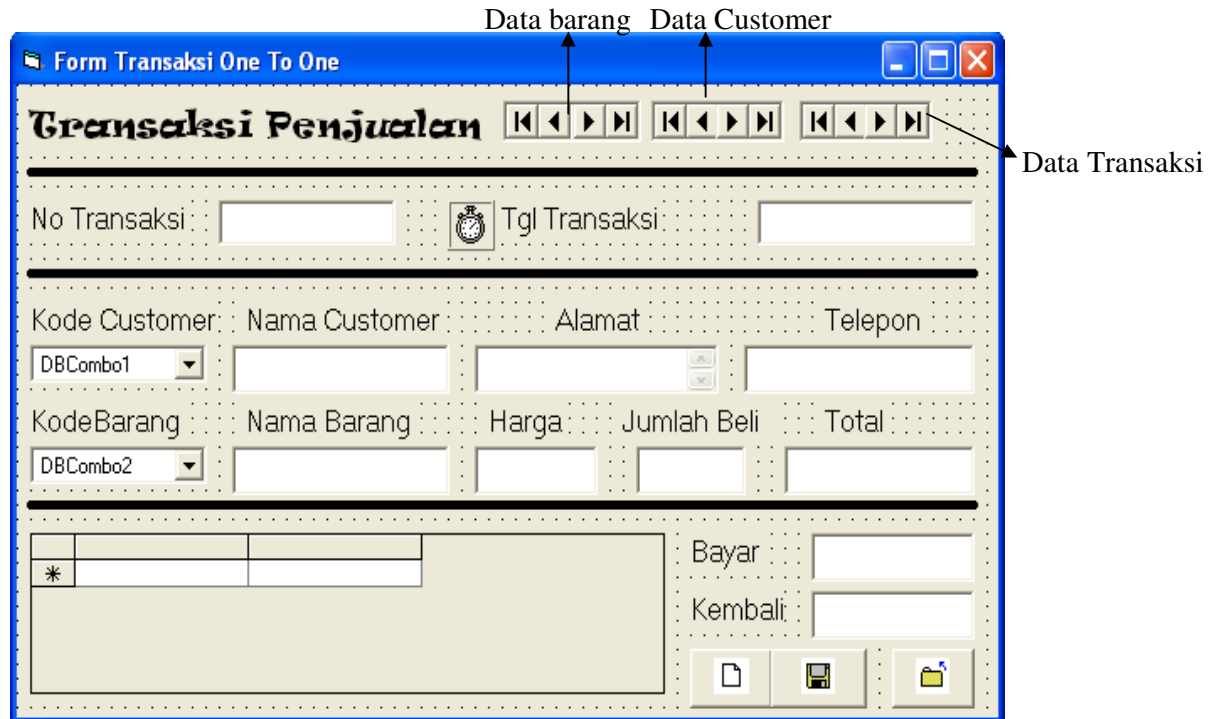
```
Private Sub aktif()  
For Each x In Me  
If TypeName(x) = "TextBox" Then  
x.Enabled = True  
End If  
Next  
End Sub
```

```
Private Sub nonaktif()  
For Each x In Me  
If TypeName(x) = "TextBox" Then  
x.Enabled = False  
End If  
Next  
End Sub
```

13.6.2. Membuat Form Transaksi One to One

Form Transaksi ini dibuat untuk mengelola data transaksi, langkah-langkah pembuatannya sebagai berikut :

1. Tambahkan sebuah form baru dengan format sebagai berikut :
Name : FOnetoOne
Caption : Form One To One
2. Pada form tersebut tambahkan objek DBGRID1, klik kanan pada Toolbox kemudian pilih Components kemudian pilih **Microsoft Data Bound Grid Control 5.0(SP3)**
3. Tambahkan objek berikut ini dan aturlah posisinya seperti pada gambar



Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Form One to One 2- CenterScreen
Label1	Caption	Transaksi Penjualan
Label2	Caption	Kode Customer
DBCombo1	ListField RowSource	Kdcust Dtcust
Label2	Caption	Nama Customer
TextBox1	Name Text	Txtnmcust Kosongkan
Label3	Caption	Alamat
TextBox2	Name Text Multiline Scrollbars	Txtalamat Kosongkan True 2-Vertical
Label4	Caption	Telepon
TextBox3	Name Text	Txtnotelp Kosongkan
DBCombo2	ListField RowSource	Kdbrg Dtbrg
Label5	Caption	Nama Barang
TextBox4	Name Text	Txtnmbrg Kosongkan
Label6	Caption	Harga

TextBox5	Name Text	txtharga Kosongkan
Label7	Caption	Jumlah Beli
TextBox6	Name Text	txtjumbel Kosongkan
Label8	Caption	Total
TextBox7	Name Text	txttotal Kosongkan
Label9	Caption	Bayar
TextBox8	Name Text	txtbayar Kosongkan
Label10	Caption	Kembali
TextBox8	Name Text	txtkembali Kosongkan
Timer	Interval	500
Data1	Name Caption	Dtbrg Barang
Data2	Name Caption	dtcust Customer
Data3	Name Caption	Dttrans Transaksi
DBGrid1	Data Source	Dttrans
CommandButton1	Caption Name Picture Style	Kosongkan cmdnew C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wznew1-Graphical
CommandButton2	Caption Name Picture Style	Kosongkan cmdsave C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzsave1-Graphical
CommandButton3	Caption Name Picture Style	Kosongkan cmdclose C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzclose1-Graphical

Koneksi Database Menggunakan Objek

Objek	Properties	Keterangan
Data1	Databasename Record Source RecordsetType	Cari tempat penyimpanan database anda Barang (Nama tabel) 0-Table
Data2	Databasename	Cari tempat penyimpanan database anda

	Record Source RecordsetType	Customer (Nama tabel) 0-Table
Data3	Databasename Record Source RecordsetType	Cari tempat penyimpanan database anda Transaksi (Nama tabel) 0-Table
DBGrid1	Data Source	Dttrans

Listing Program

```

General
Dim x As Control

Declarations

Private Sub Timer1_Timer()
txttgltrans.Text = Format(Date, "dd/mm/yy")
End Sub

Private Sub DBCombo1_Change()
On Error Resume Next
dtcust.Recordset.Index = "xkdcust"
dtcust.Recordset.Seek "=", DBCombo1
If Not dtcust.Recordset.NoMatch Then
    txtnmcust.Text = dtcust.Recordset!nmcust
    txtalamat.Text = dtcust.Recordset!alamat
    txtnotelp.Text = dtcust.Recordset!telp
End If
End Sub

Private Sub DBCombo2_Change()
dtbrg.Recordset.Index = "xkdbrg"
dtbrg.Recordset.Seek "=", DBCombo2
If Not dtbrg.Recordset.NoMatch Then
    txtnmbrg.Text = dtbrg.Recordset!nmbrg
    txtthrg.Text = dtbrg.Recordset!harga
    txtjumbel.SetFocus
End If
End Sub

Private Sub txtbayar_Change()
txtkembali.Text = Val(txtbayar.Text) - Val(txttotal.Text)
End Sub

```

Private Sub Form_Activate()

cmdsave = False

End Sub**Private Sub Form_Load()**

nonaktif

bersih

End Sub**Private Sub nonaktif()**

For Each x In Me

If TypeName(x) = "TextBox" Then

x.Enabled = 0

End If

Next x

End Sub**Private Sub aktif()**

For Each x In Me

If TypeName(x) = "TextBox" Then

x.Enabled = 1

End If

Next x

End Sub**Private Sub bersih()**

For Each x In Me

If TypeName(x) = "TextBox" Then

x.Text = ""

End If

Next x

DBCombo1.Text = "Pilih Kode"

DBCombo2.Text = "Pilih Kode"

End Sub**Private Sub cmdbatal_Click()**

p = MsgBox("cancel all saving or editing data ??", vbQuestion + vbOKCancel, "Informasi")

If VB = vbOK Then

Adocust.Recordset.MoveFirst

nonaktif

End If

cmdsave.Enabled = False

cmdadd.Enabled = True

cmdundo.Enabled = False

End Sub**Private Sub cmdclose_Click()**

Unload Me

End Sub

Private Sub cmdinput_Click()

```
dttrans.Recordset.AddNew
aktif
bersih
txtnotrans.SetFocus
cmdsave.Enabled = True
cmdinput.Enabled = False
txtnmcust.Enabled = False
txtalamat.Enabled = False
txtnotelp.Enabled = False
txtnmbrg.Enabled = False
txthrg.Enabled = False
txttltrans.Enabled = False
txttotal.Enabled = False
txtkembali.Enabled = False
```

End Sub**Private Sub cmdsave_Click()**

```
On Error Resume Next
With dttrans.Recordset
!notrans = txtnotrans.Text
!tgltrans = txttltrans.Text
!kdcust = DBCombo1
!kdbrg = DBCombo2
!jumbel = txtjumbel.Text
!total = txttotal.Text
.Update
DBGrid1.Refresh
nonaktif
cmdsave.Enabled = False
cmdinput.Enabled = True
End With
```

End Sub**Private Sub txtjumbel_KeyPress(KeyAscii As Integer)**

```
If KeyAscii = 13 Then
txttotal.Text = Val(txthrg.Text * txtjumbel.Text)
txtbayar.SetFocus
End If
```

End Sub

13.6.3. Membuat Form Transaksi One to Many

Form Transaksi ini dibuat untuk mengelola data transaksi, langkah-langkah pembuatannya sebagai berikut :

1. Tambahkan sebuah form baru dengan format sebagai berikut :
Name : FOnetoMany
Caption : Form One To Many
2. Pada form tersebut tambahkan objek DBGRID1, klik kanan pada Toolbox kemudian pilih Components kemudian pilih **Microsoft Data Bound Grid Control 5.0(SP3)**
3. Tambahkan objek berikut ini dan aturlah posisinya seperti pada gambar



The screenshot shows a Visual Basic form titled "Form One To Many" with the title bar "Form One To Many". The form has a title "TRANSAKSI PENJUALAN" and a clock icon. It contains several input fields and controls:

- No Faktur: [Text Box]
- Kode Customer: [DBCombo1]
- Tanggal: [Text Box]
- Nama Customer: [Text Box]
- Kode Barang: [DBCombo2]
- Nama Barang: [Text Box]
- Harga: [Text Box]
- Qty: [Text Box]
- Subtotal: [Text Box]

Below the input fields is a table with a grid and a row of navigation buttons (back, forward, etc.). At the bottom, there are buttons for "NEW", "SAVE", "CANCEL", "DELETE", and "EXIT", and a "Total bayar" [Text Box].

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Form One to Many 2- CenterScreen
Label1	Caption	Transaksi Penjualan
Timer	Interval	500
Label2	Caption	No Faktur
Textbox1	Name Text	Txtnofak Kosongkan
Label3	Caption	Tanggal
Textbox2	Name Text	Txttgl Kosongkan
Label4	Caption	Kode Customer
DBCombo1	ListField RowSource	Kdcust dtcust
Label5	Caption	Nama Customer
TextBox3	Name Text	Txtnmcust Kosongkan
DBCombo2	ListField RowSource	Kdbrg dtbrg
Label6	Caption	Nama Barang
TextBox4	Name Text	Txtnmbrg Kosongkan
Label7	Caption	Harga
TextBox5	Name Text	Txtharga Kosongkan
Label8	Caption	Qty
TextBox6	Name Text	Txtqty Kosongkan
Label9	Caption	Subtotal
TextBox7	Name Text	txtsubtotal Kosongkan
Label10	Caption	Total Bayar
TextBox8	Name Text	Txttotbay Kosongkan
Data1	Name Caption Database Name Data Source Record Source Type	Dtbrg Barang Cari penyimpanan database Pilih tabel Barang 0-Table
Data2	Name Caption Database Name Data Source Record Source Type	Dtcust Customer Cari penyimpanan database Pilih tabel Customer 0-Table
Data3	Name Caption	Dtrans Transaksi

	Database Name Data Source Record Source Type	Cari penyimpanan database Pilih tabel Transaksi 0-Table
Data4	Name Caption Database Name Data Source Record Source Type	dtdetail Detail Cari penyimpanan database Pilih tabel detail 0-Table
Data5	Name Caption Database Name Data Source Record Source Type	dtsementara Sementara Cari penyimpanan database Pilih tabel sementara 0-Table
DBGrid1	Data Source	dtsementara
CommandButton1	Caption Name	NEW cmdnew
CommandButton2	Caption Name	SAVE cmdsave
CommandButton3	Caption Name	CANCEL Cmdcancel
CommandButton4	Caption Name	DELETE cmdcancel
CommandButton5	Caption Name	EXIT cmexit

Listing Program :**General Declarations**

```
Dim x As Control
```

Sub auto()

```
Dimurut As String * 4
```

```
Dim hitung As Integer
```

```
With dtjual.Recordset
```

```
If .RecordCount = 0 Then
```

```
    txtnofak = "F" + "001"
```

```
Else
```

```
    .MoveLast
```

```
    urut = Val(Right(!nofaktur, 3))
```

```
    hitung = urut + 1
```

```
If hitung < 10 Then
```

```
    txtnofak = "F" + "00" + Trim(Str(hitung))
```

```
Else
```

```
    txtnofak = "F" + Trim(Str(hitung))
```

```
End If
```

```
End If
```

```
End With
```

```
End Sub
```

Function hapus_grid()

```
'menghapus untuk tabel sementara
If dtsementara.Recordset.RecordCount = 0 Then
    nonaktif
Else
    dtsementara.Recordset.MoveFirst
    Do While Not dtsementara.Recordset.EOF
        dtsementara.Recordset.Delete
        dtsementara.Recordset.MoveNext
    Loop
End If
```

End Function**Public Sub tombol(tambah, batal, hapus, simpan, keluar As Boolean)**

```
cmdnew.Enabled = tambah
cmdcancel.Enabled = batal
cmddelete.Enabled = hapus
cmdsave.Enabled = simpan
cmdexit.Enabled = keluar
```

End Sub**Private Sub cmdcancel_Click()**

```
bersih
tombol True, False, False, False, True
Call hapus_grid
```

End Sub**Private Sub Form_Activate()**

```
tombol 1, 0, 0, 0, 1
bersih
nonaktif
```

End Sub**Private Sub Timer1_Timer()**

```
txttgl = Date
```

End Sub**Private Sub DBCombo1_Change()**

```
dtcust.Recordset.FindFirst "kdcust=" & DBCombo1 & ""
txtnmcust = dtcust.Recordset!nmcust
```

End Sub**Private Sub DBCombo2_Change()**

```
dtbrg.Recordset.FindFirst "kdbrg=" & DBCombo2 & ""
txtnmbrg = dtbrg.Recordset!nmbrg
txthrg = dtbrg.Recordset!harga
txtqty.SetFocus
```

End Sub

Sub bersih()

```
For Each x In Me
If TypeName(x) = "TextBox" Then x.Text = ""
    DBCombo1 = ""
    DBCombo2 = ""
Next x
End Sub
```

Sub nonaktif()

```
For Each x In Me
If TypeName(x) = "TextBox" Then x.Enabled = False
If TypeName(x) = "DBCOMbo" Then x.Enabled = False
Next x
End Sub
```

Sub aktif()

```
For Each x In Me
If TypeName(x) = "TextBox" Then x.Enabled = True
If TypeName(x) = "DBCOMbo" Then x.Enabled = True
Next x
End Sub
```

Private Sub txtqty_KeyPress(KeyAscii As Integer)

```
If KeyAscii = 13 Then
txtsubtotal = Val(txtqty) * Val(txthrg)
With dtsementara.Recordset
    .AddNew
    !kdbrg = Trim(DBCombo2)
    !nmbrg = txtnmbrg
    !harga = txthrg
    !qty = txtqty
    !subtotal = txtsubtotal
    .Update
    .MoveLast
    txttotbay = Val(txtsubtotal) + Val(txttotbay)

    lagi = MsgBox("ingin menambah lagi?", 36, "Tanya")
If lagi = 6 Then
    DBCombo2 = ""
    txtnmbrg = ""
    txthrg = ""
    txtqty = ""
    txtsubtotal = ""
    DBCombo2.SetFocus
    tombol False, False, True, True, False
Else
    tombol False, True, True, True, False
    'nonaktif
End If
End With
End If
End Sub
```

Private Sub cmddelete_Click()

```
If totbay = "" Then
    txttotbay = ""
    dtsementara.Recordset.Delete
Else
    txttotbay = Val(txttotbay) - dtsementara.Recordset!subtotal
    dtsementara.Recordset.Delete
End If
End Sub
```

Private Sub cmdexit_Click()

```
Unload Me
```

```
End Sub
```

Private Sub cmdnew_Click()

```
bersih
aktif
auto
tombol False, True, False, False, True
End Sub
```

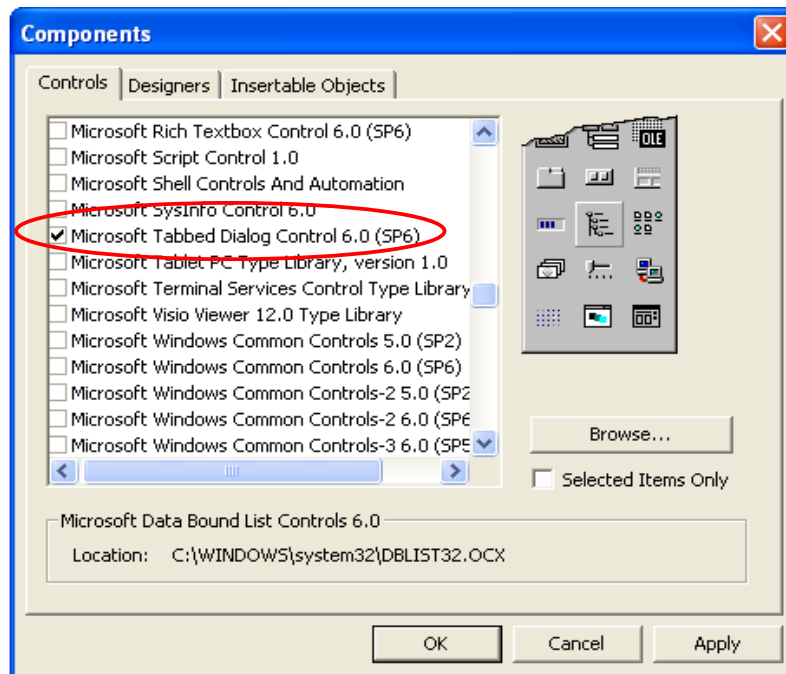
Private Sub cmdsave_Click()

```
On Error Resume Next
With dtjual.Recordset
    .AddNew
    !nofaktur = txtnofak
    !tglfak = txttgl
    !totbay = txttotbay
    !kdcust = DBCombo1
    .Update
End With
With dtsementara.Recordset
    .MoveFirst
    While Not dtsementara.Recordset.EOF
        dtdetail.Recordset.AddNew
        dtdetail.Recordset!nofaktur = txtnofak.Text
        dtdetail.Recordset!kdbrg = !kdbrg
        dtdetail.Recordset!qty = !qty
        dtdetail.Recordset.Update
        .Delete
        .MoveNext
    end
End With
bersih
tombol True, False, False, False, True
nonaktif
End Sub
```

13.6.4. Membuat Form Customer

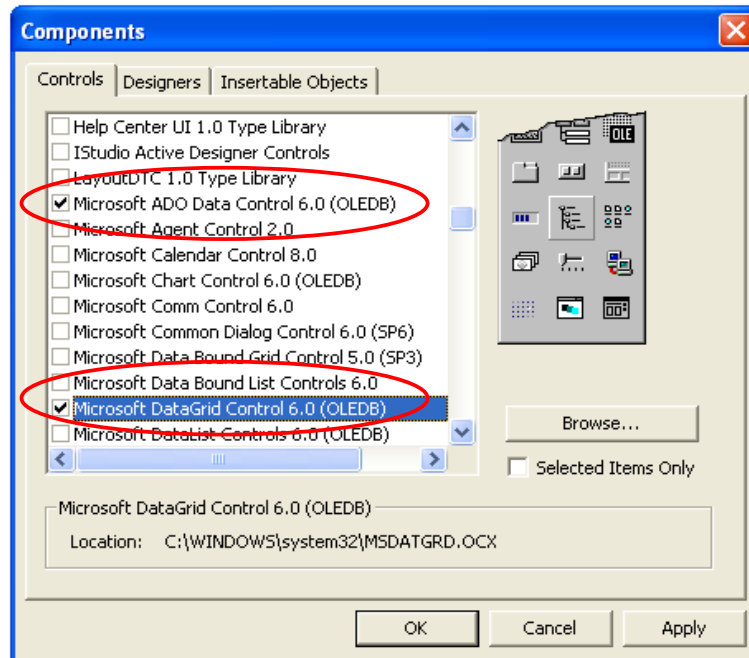
Form Customer ini dibuat untuk mengelola data Customer, Bagian utama dari Form ini adalah sebuah objek SSTab yang terdiri dari 2 tab, yaitu tab **Input** dan tab **Browse**. Untuk membuat form customer seperti gambar 13.7.3, langkah-langkah pembuatannya sebagai berikut :

1. Tambahkan sebuah form baru dengan format sebagai berikut :
Name : FCustomer
Caption : Form Customer
 - a. Pada form tersebut tambahkan objek SSTab, klik kanan pada Toolbox kemudian pilih Components



Gambar 13.7.2

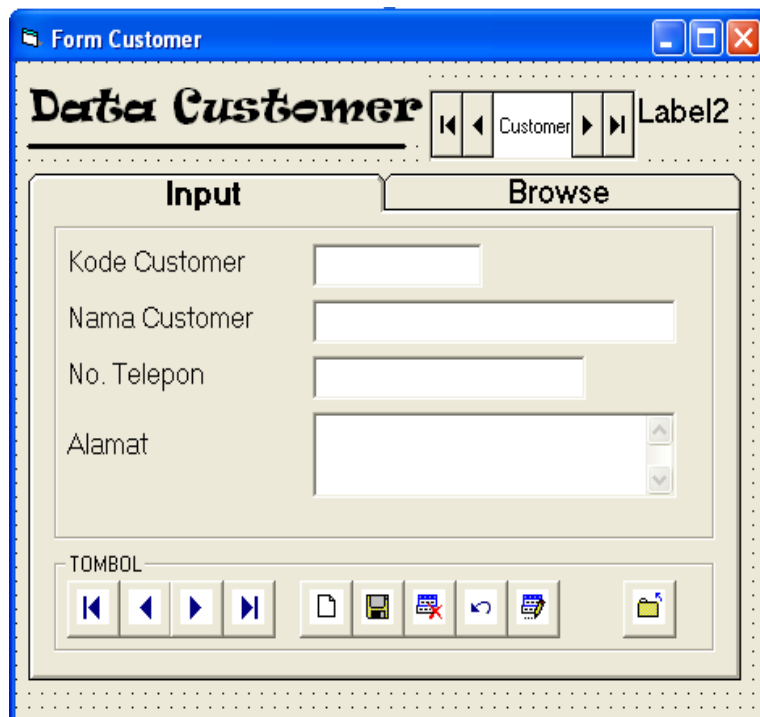
- b. Selanjutnya cari dan checklist objek **Microsoft ADO Data Control 6.0 (OLEDB)** dan **Microsoft DataGrid Control 6.0 (OLEDB)**



2. Pada Form tersebut tambahkan objek SSTab, lalu ubahlah propertinya sebagai berikut:

- Tab Count : 2
- TabPerRow : 2
- TabHeight : 400
- TabCaption : - Tab 0 : Input - Tab 1 : Browse

3. Tambahkan objek berikut ini dan aturlah posisinya seperti pada gambar

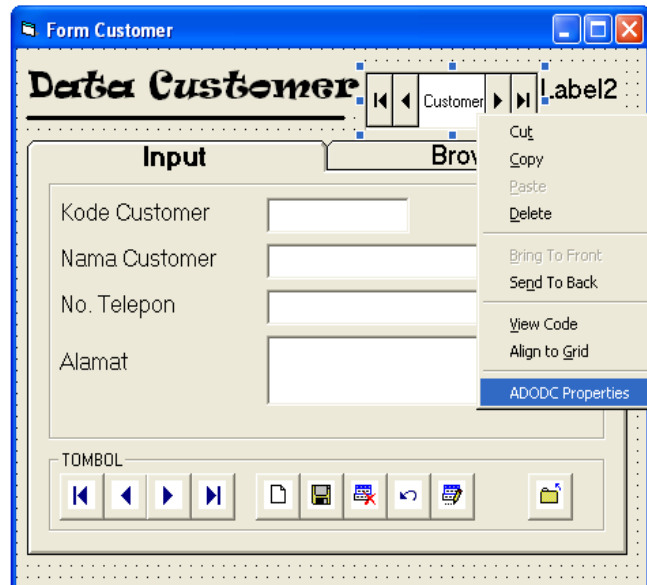


4. Form Customer tambahkan objek berikut dan atur posisinya

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Form Customer 2- CenterScreen
Label1	Caption	Data Customer
Adodc1	Name Caption Recordsource	adocust Customer Customer

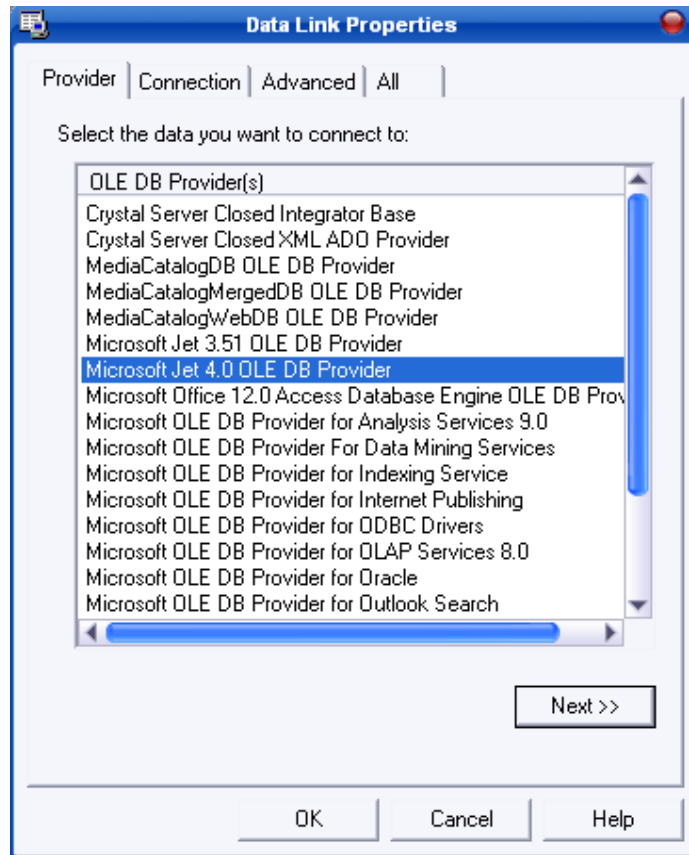
Cara Mengkoneksi :

- a. Klik kanan pada objek adodc kemudian pilih **Adodc Properties**, tampil tampilan berikut :

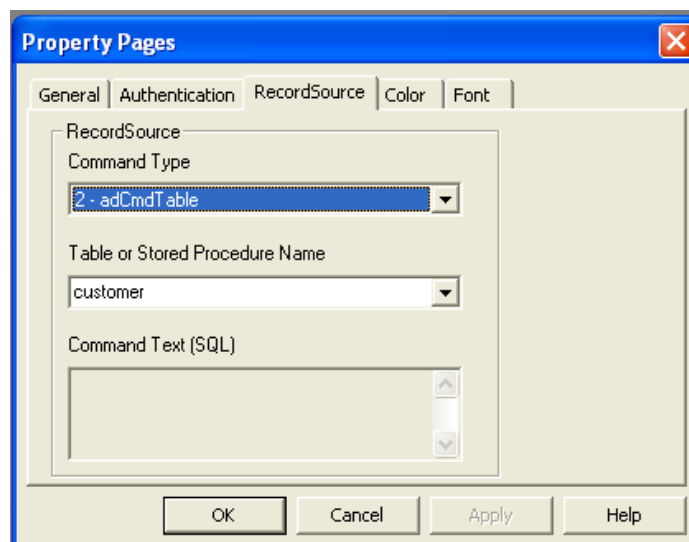


- b. Pada Tab General :
1. Tab General pilih **Use Connection String**, lalu klik **Build**
 2. Tampil **Data Link Properties**, pada tab Provider pilih Microsoft Jet 4.0 OLE DB Provider klik Next kemudian tampil **Select or Enter a database name** klik





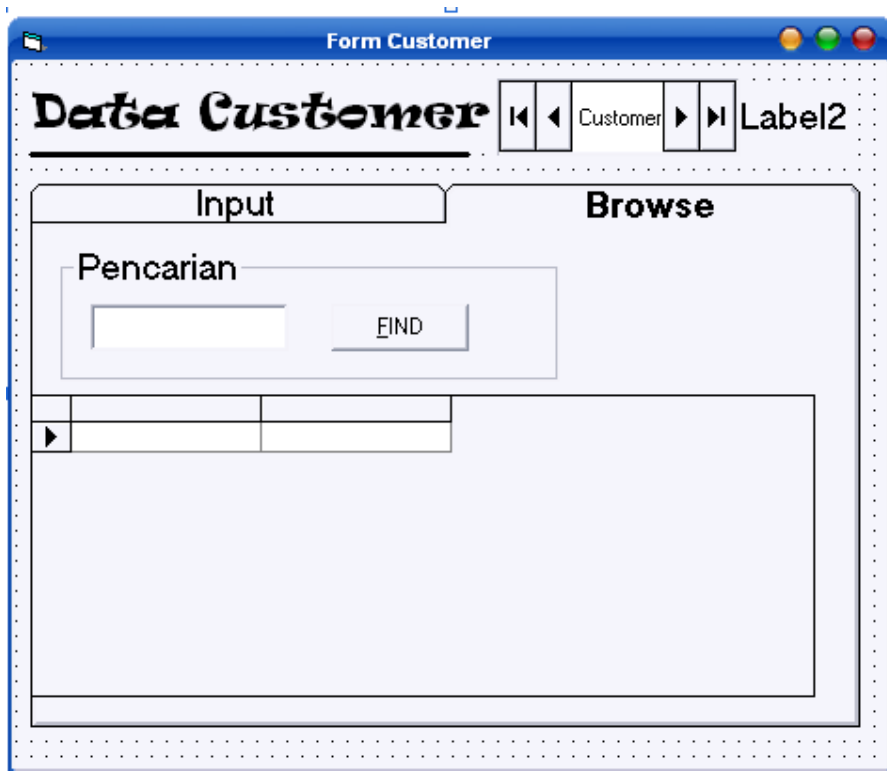
3. Selanjutnya klik Test Connection, langkah selanjutnya klik tombol **OK**
 4. Untuk keluar dari **Data Link Properties** klik tombol **OK**
- c. Pada tab **RecordSource** pada Command Type pilih **2-adCmdTable**, lalu **Table or Stored Procedure Name** pilih **customer** pilih **Apply** kemudian **OK**



Objek	Property	Nilai
Frame1	Caption	Kosongkan
Label1	Caption	Kode Customer
TextBox1	Name Text	Txtkdcust Kosongkan
Label2	Caption	Nama Customer
TextBox2	Name Text	Txtnmcust Kosongkan
Label3	Caption	Txtnotelp
TextBox3	Name Text	Txtnotelp Kosongkan
Label4	Caption	Alamat
TextBox4	Name Text Multiline Scrollbars	Txtalamat Kosongkan True 2-Vertical
Frame2	Caption	Tombol
CommandButton1	Caption Name Picture Style	Kosongkan cmdtop C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wztop 1-Graphical
CommandButton2	Caption Name Picture Style	Kosongkan cmdprev C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzback \Wzback
CommandButton3	Caption Name Picture Style	Kosongkan cmdnext C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wznext 1-Graphical
CommandButton4	Caption Name Picture Style	Kosongkan cmdlast C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzend 1-Graphical
CommandButton5	Caption Name Picture Style	Kosongkan cmdnew C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps \Wznew 1-Graphical
CommandButton6	Caption Name Picture Style	Kosongkan cmdnew C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps \Wzsave 1-Graphical
CommandButton7	Caption	Kosongkan

	Name Picture Style	cmddelete C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzdelete1-Graphical
CommandButton8	Caption Name Picture Style	Kosongkan cmdundo C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzundo1-Graphical
CommandButton9	Caption Name Picture Style	Kosongkan cmdundo C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzedit1-Graphical
CommandButton10	Caption Name Picture Style	Kosongkan cmdclose C:\Program Files\Microsoft Visual Studio\Vfp98\Wizards\Wizbmps\Wzclose1-Graphical

5. Pada Tab **Input** tambahkan objek berikut ini dan aturlah posisinya seperti gambar
6. Pada Tab **Browse** tambahkan objek berikut ini dan aturlah posisinya seperti gambar



Objek	Property	Nilai
Frame 2	Caption	Pencarian
TextBox1	Name Text	txtfind Kosongkan
Datagrid1	Name Control Source	DataGrid1 adocust

A. Koneksi Tabel dengan TextBox

Objek	Properties	Keterangan
txtkdcust	Data Source Data Field	adocust kdcust
txtnmcust	Data Source Data Field	adocust nmcust
txtalamat	Data Source Data Field	adocust alamat
txtnotelp	Data Source Data Field	adocust notelp

Listing Program :

```
General           Declarations
Public isicust As Boolean

Private Sub cmdadd_Click()
aktif
kosong
txtkdcust.SetFocus
cmdsave.Enabled = True
cmdadd.Enabled = False
cmdundo.Enabled = True
End Sub

Private Sub cmdclose_Click()
p = MsgBox("Yakin ingin keluar ??", 32 + 4, "Informasi")
If p = 6 Then
    Unload Me
End If
End Sub

Private Sub cmddelete_Click()
p = MsgBox("Kode" + txtkdcust.Text + "dan seluruh data akan dihapus ??", vbQuestion +
vbOKCancel, "Konfirmasi")
If p = vbOK Then
    Adocust.Recordset.Delete
    Adocust.Recordset.MovePrevious
    nonaktif
End If
End Sub

Private Sub cmdedit_Click()
isicust = False
aktif
txtkdcust.Enabled = False
txtnmcust.SetFocus
cmdadd.Enabled = False
cmdsave.Enabled = True
cmdedit.Enabled = False
cmdundo.Enabled = True
End Sub
```

Private Sub cmdfind_Click()

```
If txtfind.Text = "" Then
    MsgBox "Mohon data diinput terlebih dahulu", vbInformation,
    "Pencarian"
Exit Sub
End If
With adocust.Recordset
    .MoveFirst
    .Find "kdcust=" & txtfind.Text & ""
    If Not .EOF Then
        MsgBox "kode" + txtfind.Text + "ditemukan!", 32, "Pencarian"
    Else
        MsgBox "kode" + txtfind.Text + "tidak ada!", 16, "Kesalahan
        pencarian"
    End If
End With
End Sub
```

Private Sub cmdlast_Click()

```
On Error Resume Next
Adocust.Recordset.MoveLast
MsgBox "data sudah diakhir record!", 16, "Informasi"
End Sub
```

Private Sub cmdnext_Click()

```
On Error Resume Next
Adocust.Recordset.MoveNext
If Adocust.Recordset.EOF Then
    Adocust.Recordset.MoveLast
    MsgBox "Sudah diakhir record", vbCritical, "Informasi"
End If
End Sub
```

Private Sub cmdprev_Click()

```
On Error Resume Next
Adocust.Recordset.MovePrevious
If Adocust.Recordset.BOF Then
    Adocust.Recordset.MoveFirst
    MsgBox "Sudah diawal record", vbCritical, "Informasi"
End If
End Sub
```

Private Sub cmdundo_Click()

```
p = MsgBox("Yakin akan membatalkan penginputan ??", vbQuestion + vbOKCancel, "Konfirmasi")
If p = vbOK Then
    kosong
    nonaktif
End If
```

Private Sub cmdsave_Click()

```
On Error Resume Next
With Adocust.Recordset
If isicust = True Then
    .AddNew
    txtkdcust.Text = !kdcust
    txtnmcust.Text = !nmcust
    txtnotelp.Text = !notelp
    txtalamat.Text = !alamat
    .Update
Else
    txtnmcust.Text = !nmcust
    txtnotelp.Text = !notelp
    txtalamat.Text = !alamat
    .Update
End If
On Error GoTo 0
cmdsave.Enabled = False
cmdadd.Enabled = True
cmdedit.Enabled = True
cmdundo.Enabled = False
nonaktif
End With
End Sub
```

Private Sub cmdtop_Click()

```
On Error Resume Next
Adocust.Recordset.MoveFirst
MsgBox "data sudah diawal record!", 16, "Informasi"
End Sub
```

Private Sub Form_Load()

```
Label2.Caption = Date
cmdsave.Enabled = False
cmdundo.Enabled = False
nonaktif
End Sub
```

Private Sub kosong()

```
txtkdcust.Text = ""
txtnmcust.Text = ""
txtnotelp.Text = ""
txtalamat.Text = ""
End Sub
```

Private Sub nonaktif()

txtkdcust.Enabled = 0

txtnmcust.Enabled = 0

txtnotelp.Enabled = 0

txtalamat.Enabled = 0

End Sub**Private Sub aktif()**

txtkdcust.Enabled = 1

txtnmcust.Enabled = 1

txtnotelp.Enabled = 1

txtalamat.Enabled = 1

End Sub

BAB XIV

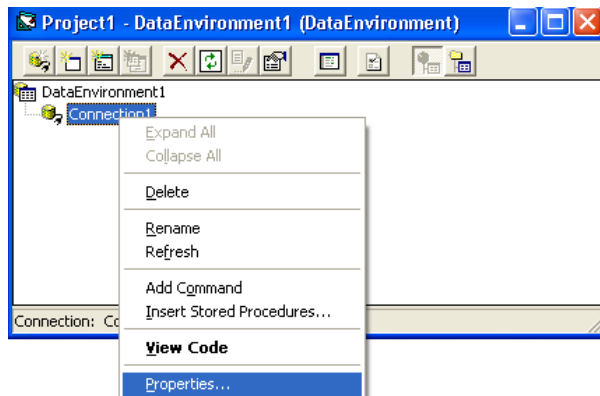
LAPORAN DENGAN DATA REPORT

Salah satu fasilitas Visual Basic ialah menyediakan *Data Report* untuk membuat Laporan. seperti halnya kontrol data yang digunakan sebagai penghubung dengan database, Data Report pula membutuhkan *Data Environment* yang berfungsi sebagai penghubung antara database yang ada dengan Data Report.

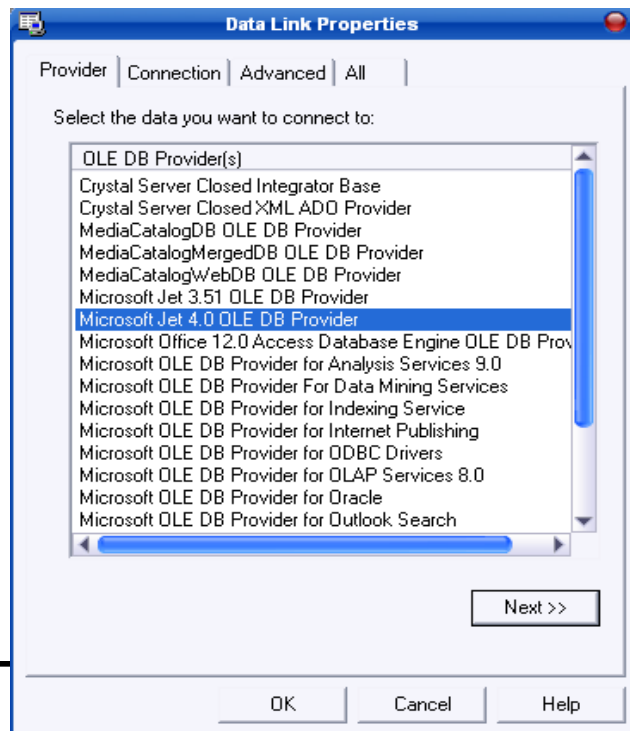
Untuk membuat Report dengan Data Report di Visula Basic, adalah sebagai berikut :


1. Buat Data Environment

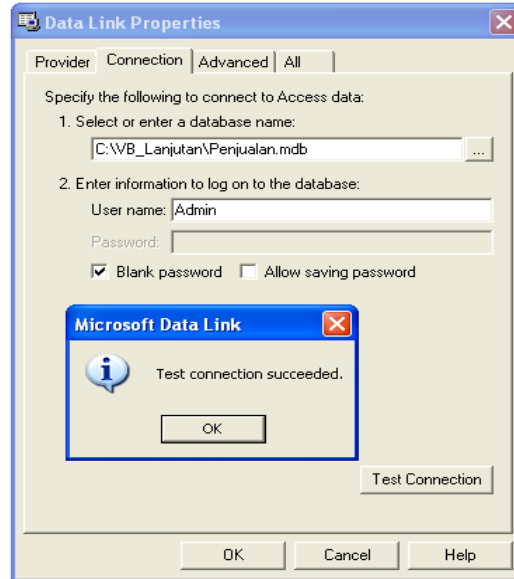
- Klik **Project** pada MenuBar , Klik **Data Add Environment**
- Akan tampil jendela Data Environment, klik kanan pada connection1, pilih properties



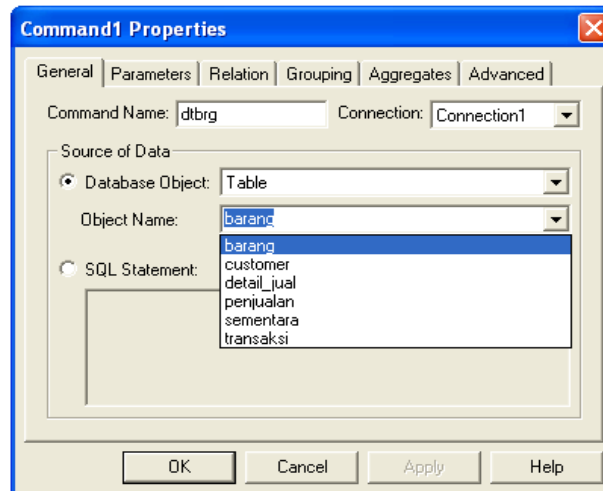
- Akan tampil jendela Data Link Properties (untuk menghubungkan dengan database)
- Pilih **Microsoft Jet 4.0 OLE DB PROVIDER**, klik Next



- e. Muncul tab Connection kemudian pada **Select or enter a database name** pilih nama database klik tombol  cari penyimpanan database yang akan di hubungkan **Penjualan.mdb**
- f. Klik **Test Connection**, jika koneksi berhasil (database telah terkoneksi). Klik OK



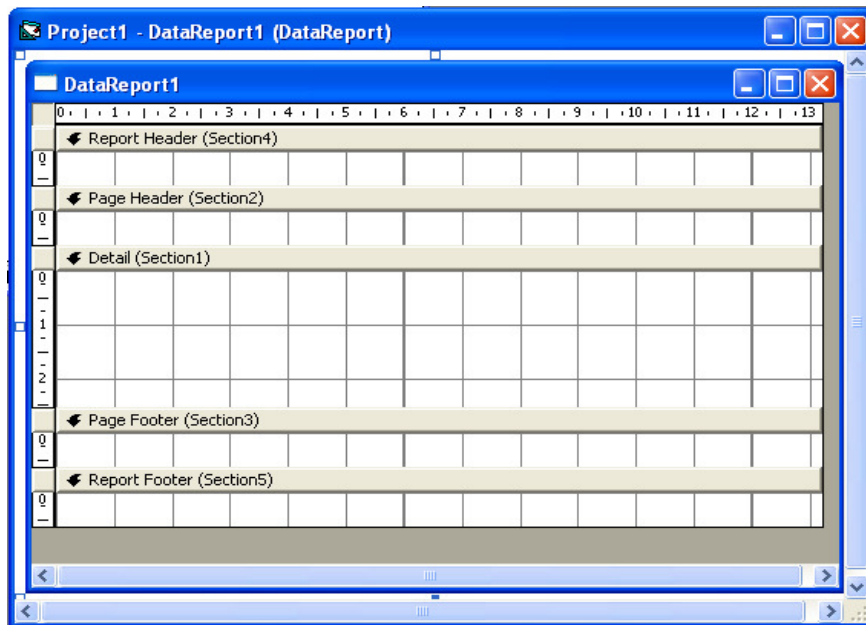
- g. Untuk mewakili tabel, tambahkan command pada **Connection1** dengan cara Klik kanan pada Command1, pilih **Add Command**
- h. Klik kanan pada **Command1** lalu pilih **Properties** Akan tampil jendela **Command1 Properties**
- i. Untuk bagian **Command Name** ketikkan **dtbrg** kemudian pada **connection**, pilih **connection1**
- j. Pada bagian **Database Object** pilih **Table**, langkah selanjutnya **Object Name** pilih nama tabel **barang**



- k. Klik OK

2. Buat Data Report


- a. Klik **Project** pada MenuBar , Klik **Data Add Report**



Penjelasan :

Bagian	Keterangan
Report Header	Berisi Judul Laporan
Page Header	Berisi Judul Kolom data yang akan ditampilkan
Detail	Berisi data field yang akan ditampilkan pada kolom
Page Footer	Berisi catatan kaki di akhir kolom-kolom report
Report Footer	Berisi catatan kecil / keterangan Report

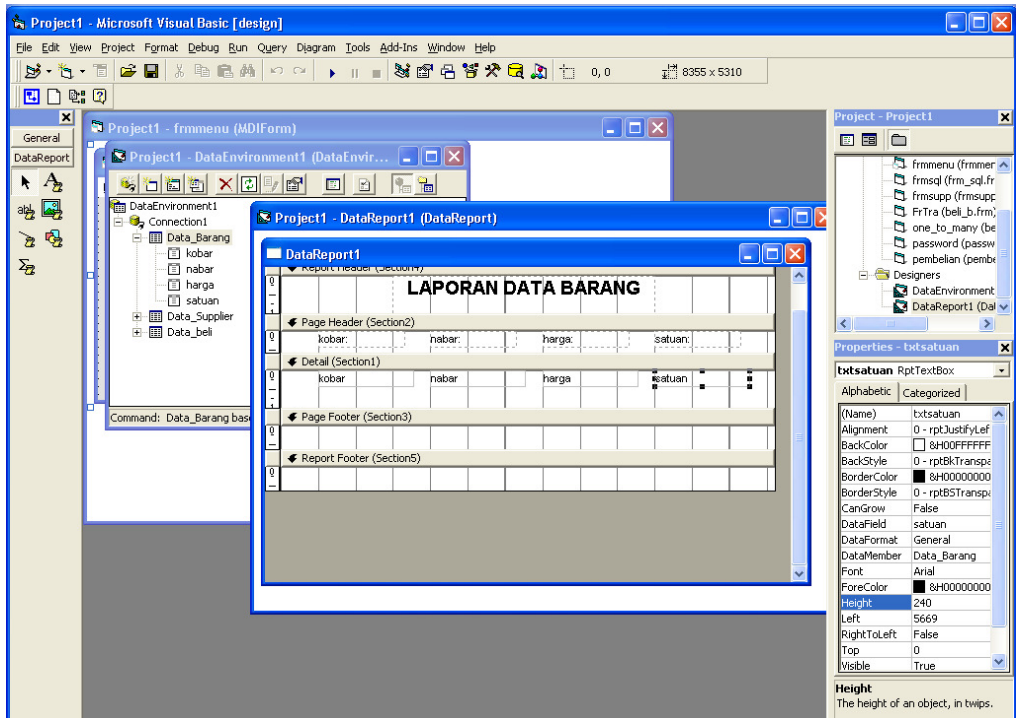
samping jendela report, ada beberapa Toolbox yang mempunyai fungsi yang sama dengan toolbox form.

- b. Design RptLabel  di Report Header. Ubah Property : **Caption** menjadi "Laporan Data Barang" dan proprty : **Alignment** menjadi **rptjustcenter**. Kemudian atur tampilannya sesuai dengan yang anda inginkan gunakan Fungsi **Font** pada **Properties** untuk merubah
- c. klik **Data Report**

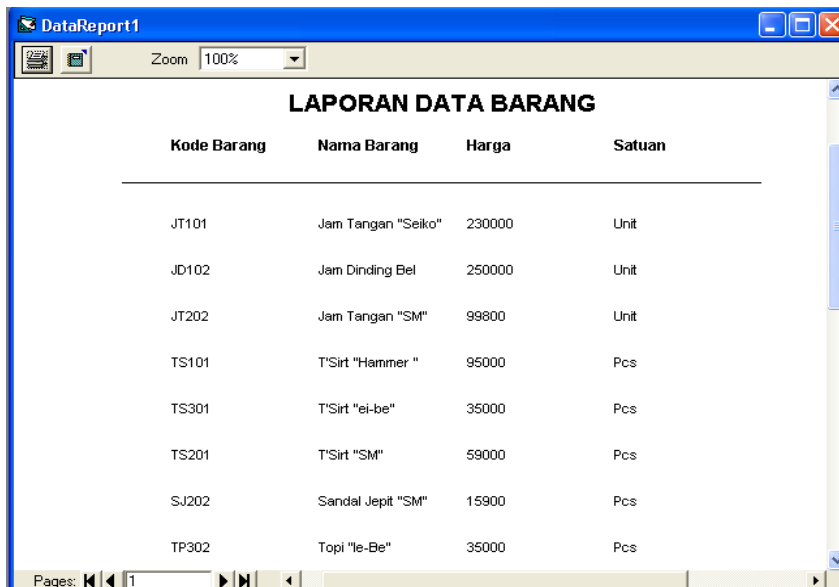
Object	Properties	Value
Data Report	DataSource	Data Environment1
	DataMember	Data_Barang (nama Command)

- d. Letakan jendela **Data Environment** di atas Jendela **Data Report**

- e. **Drag & drop** field-filed pada jendela Data Environment ke jendela Data Report di bagian Detail
- f. Pindahkan bagian yang diakhiri tanda “:” ke bagian **Page Header**



- g. Desain akhir Data Report adalah sebgiaa berikut :



- h. Smpn dengan nama lapbarang
- i. Buatlah Laporan untuk table yang lain

BAB XV

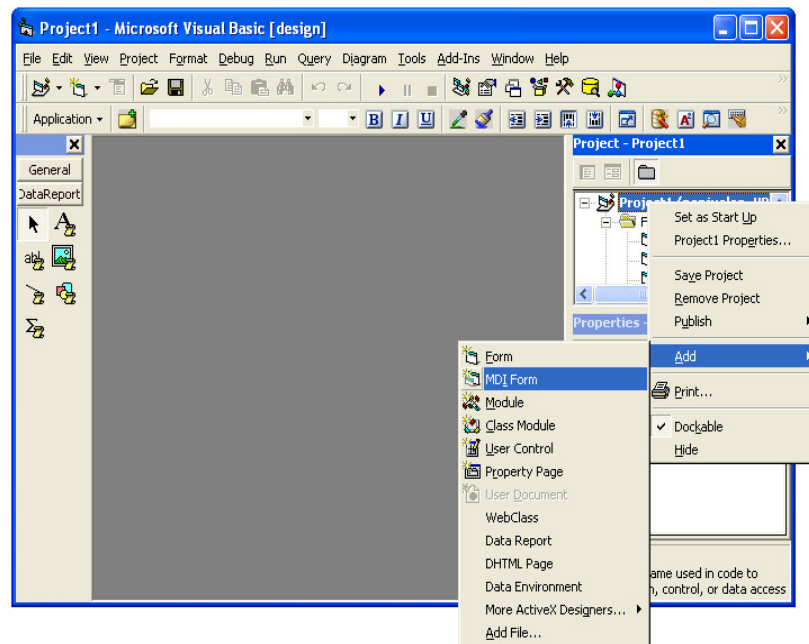
MENU UTAMA

15.1. Desain Menu

Menu digunakan untuk menghubungkan antar form yang ada, dimana antara form satu dengan yang lain saling menunjang dan terintegrasi.

Sebagai contoh kita akan membuat menu utama yang akan menghubungkan form-form yang sudah kita buat pada bab-bab sebelumnya. Untuk itu, langkah-langkah pembuatan sebagai berikut :

1. Pada Project Penjualan yang ada anda tambahkan Form baru yaitu **Klik Kanan Pada Project Pilih Add** kemudian **MDI Form**.



2. Akan Tampil jendela MDI Form

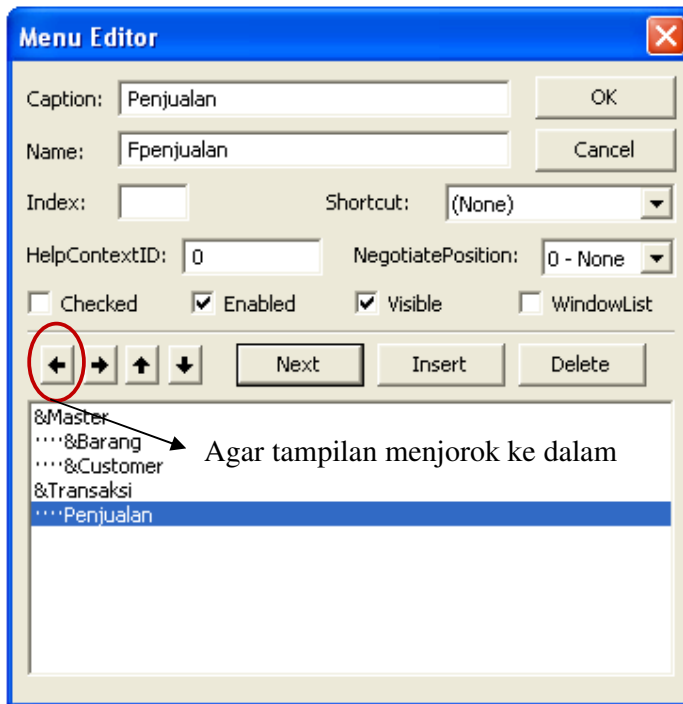


3. Klik Icon **Menu Editor** yang berada di toolbar seperti gambar dibawah ini



Menu Editor

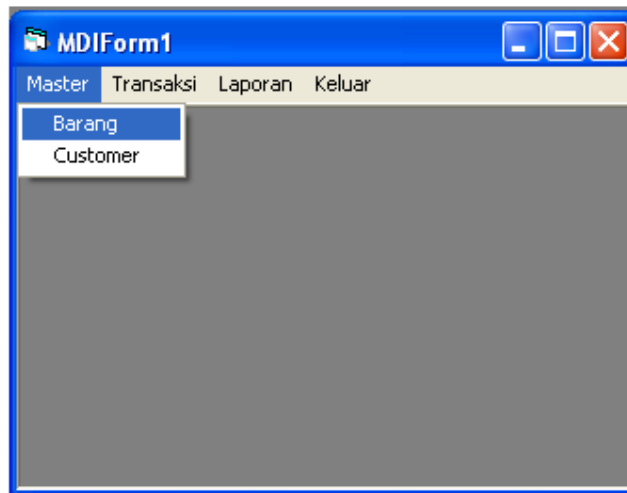
4. **Caption** digunakan sebagai judul dari menu, sedangkan **Name** sebagai pengenal dari menu. **Enable** digunakan agar menu dapat digunakan. **Visible** agar dapat di tampilkan di layar. Klik Next untuk dapat membuat sub menu. Klik Ok jika sudah selesai.



Rancang tampilan Menu Utama berdasarkan tabel berikut :

Caption	
Field1	Field2
&Master	Barang
	Customer
&Transaksi	Penjualan
&Laporan	Lap. Barang
	Lap.Customer
	Lap.Penjualan
&Exit	-

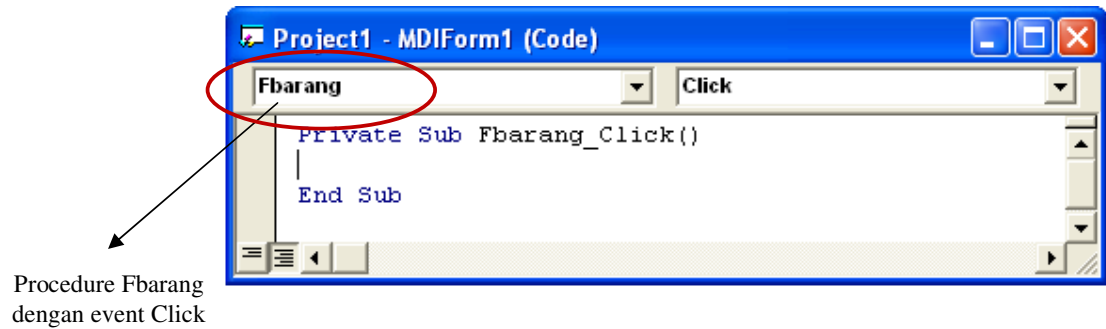
5. Selesai membuat menu utama anda tinggal melihat didalam MDI form hasil dari pembuatan menu utama, kemudian simpan form dengan nama frmmenu.frm



15.2. Koneksi Antar Form

MDI Form hanyalah sekedar form biasa tidak dapat digunakan untuk memanggil form-form yang lainnya. Untuk dapat mengaitkan antar form dengan langkah sebagai berikut

1. Buka Form **Menu utama**
2. Klik File pada **MDIMenu** kemudian pilih **Data Barang**
3. Maka akan tampil seperti gambar dibawah ini :



Dengan lisitng program untuk menampilkan form yang lain sebagai berikut :

```
Private Sub Fbarang_Click()  
frmbarang.show  
End Sub
```

Penjelasan :

Pilih form yang hendak anda tampilkan kemudian tambahkan dengan perintah **SHOW**.

BAB XVI

SQL (*Structured Query Language*)

SQL adalah bahasa yang digunakan untuk memuat, mengurutkan, menyaring, mengupdate dan mengolah relasi antar database suatu data sehingga dihasilkan suatu data yang spesifik dari suatu database.

Bentuk Umum :

```
<Objek Data>. RecordSource = <String SQL>
```

Ket: <String SQL> adalah string yang berisi pernyataan dalam SQL.

Bentuk-Bentuk dalam SQL

1. *Select...From*

Digunakan Untuk memilih field dan tabel data dengan SQL

Bentuk Umum :

```
Select <Daftar Field> From <Nama Tabel> In <Nama Database>
```

Ket : <Field> Adalah nama-nama field yang diinginkan, bisa satu /lebih field

<Nama Tabel> adalah salah satu nama tabel yang ada dalam database (*) untuk semua field

2. *Select...From...Where*

Digunakan Untuk Menyaring data berdasarkan Kriteria tertentu

Bentuk Umum :

```
Select <Daftar Field> From <Nama Tabel> Where <Syarat>
```

3. *Select...From...OrderBy*

Digunakan untuk mengakses suatu data dengan hasil yang terurut (mengurutkan data secara Assembly)

Bentuk Umum :

```
Select <Daftar Field> From <Nama Tabel> In <Nama Database> Order By  
<Field Kunci1>, <Field Kunci2>
```

Contoh Program :

1. Buat Database dengan nama : dtmahasiswa
2. Buat Tabel baru dengan nama : Mahasiswa

Name	Type	Size
Nim	Text	5
Nama	Text	25
Jurusan	Text	25
Kelas	Text	5
IPK	Single	4

3. Isi Record Sebanyak 5 :

Nim	Nama	Jurusan	Kelas	IPK
12011	Nanny	Manajemen Informatika	MI4D	3.12
11025	Manda	Komputerisasi Akutansi	KA4C	3.15
13020	Aldy	Teknik Komputer	TK4E	3.23
12055	Adit	Manajemen Informatika	MI4R	3.19
13021	Andiya	Teknik Komputer	TK4F	3.45

4. Buat tampilan Form seperti gambar dibawah ini :

The screenshot shows a window titled "Perintah SQL" with a data grid titled "Data Mahasiswa". The grid has four columns: "nim", "nama", "jurusan", and "kelas". The data in the grid is as follows:

nim	nama	jurusan	kelas
12055	Adit	Manajemen Informatika	MI4R
13020	Aldy	Teknik Komputer	TK4E
13021	Andiya	Teknik Komputer	TK4F
11025	Manda	Komputerisasi Akutansi	KA4C
12011	Nanny	Manajemen Informatika	MI4D

Below the grid, there is a text box containing the SQL query: "Select * from mahasiswa order by nama". There are two buttons: "Tampil" and "Keluar".

5. Ketikkan Listing Program Berikut :

```
Private Sub Cmdselesai_Click()  
Pesan = MsgBox("Yakin Nih Ingin Keluar ?", 32 + 1, "Info")  
If pesan = vbOK Then  
End  
Else  
Pesan = MsgBox("Ga Jadi Keluar Ah!", vbInformation + vbOkOnly, "Informasi")  
End If  
End Sub  
  
Private Sub CmdTampil_Click()  
Data1.RecordSource = Text1.Text  
Data1.Refresh  
End Sub
```

Proses Program :

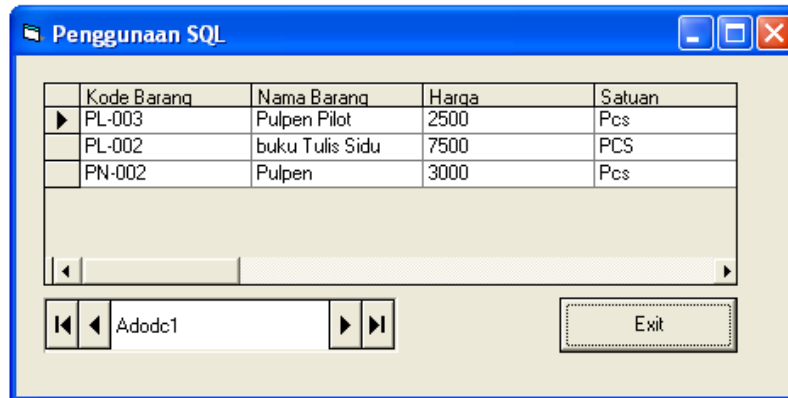
Cara menjalankan program dengan mengetikkan perintah SQL pada Text Box, kemudian klik Command Tampil Data maka data yang dimunculkan pada DBGrid1 akan terseleksi dengan perintah SQL yang diketik. Buatlah perintah SQL dibawah ini, sebelumnya koneksikan dulu database dengan grid melalui jendela properties :

1. Select nim, nama, kelas from Tbl_Mhs
2. Select * from Tbl_Mhs
3. Select * from Tbl_Mhs where jurusan = 'Teknik Komputer'
4. Select * from Tbl_Mhs where ipk > 3.00
5. Select * from Tbl_Mhs order by nama
6. Select * from Tbl_Mhs order by ipk

BAB XVII

KONEKSI DATABASE DENGAN MODUL

1. Desain tampilan Form seperti gambar dibawah ini :



2. Atur tampilan Form dan sesuaikan :

Objek	Property	Nilai
Form1	BorderStyle Caption StartPosition	1-FixedSingle Form One to One 2- CenterScreen
Label1	Caption	Transaksi Penjualan
Label2	Caption	Kode Customer
DBCombo1	ListField RowSource	Kdcust dtecust
Label2	Caption	Nama Customer

3. Listing Program :

Public Sub koneksi()

```
Frmbarang.Adodc1.ConnectionString="Provider=microsoft.jet.OLEDB.4.0;Data Source= 'c:\Penjualan.mdb'; persist security info=false"
```

```
Frmbarang.Adodc1.RecordSource = "select * from barang order by kdbrg"
```

```
Frmbarang.Adodc1.Refresh
```

End Sub

Private Sub cmdexit_Click()

```
Unload Me
```

End Sub

Private Sub Form_Load()

```
koneksi
```

End Sub

DAFTAR PUSTAKA

- Firdaus. 2005. Pemrograman Database dengan Microsoft Visual Basic 6.0 untuk Orang Awam. Maxikom. Palembang
- Kurniawan, Rulianto. 2007. 30 Trik Modifikasi Windows dengan Visual Basic 6.0. Maxikom. Palembang
- Madcom. 2005. Mahir Dalam 7 Hari Pemrograman Visual basic 6.0. Andi. Yogyakarta
- Madcom. 2005. Aplikasi Pemrograman Database dengan Visual Basic 6.0 dan Crystal Report. Andi. Yogyakarta
- Malik, Jaja Jamaludin. 2006. Animasi Form pada Visual Basic. Andi. Yogyakarta
- Nugroho, Bunafit. 2005. Membuat Animasi dan tampilan Cantik pada Interface Form Visual Basic. Gava Media. Yogyakarta
- Prasetyo, Didik Dwi. 2006. 101 Tip & Trik Visual Basic 6.0. Elex Media Komputindo. Jakarta
- Rusmawan, Uus. 2007. Konsep dan Implementasi Visual Basic. Elex Media Komputindo. Jakarta
- Supardi, Yuniar, Ir. 2006. Microsoft Visual Basic 6.0 Untuk Segala Tingkatan. Elex Media Komputindo. Jakarta
-



Modul VB FUNDAMENTAL

Disusun Oleh : Team Penyusun Modul

Akademik Manajemen Informatika dan Komputer

Bina Sarana Informatika

J a k a r t a 2 0 1 0