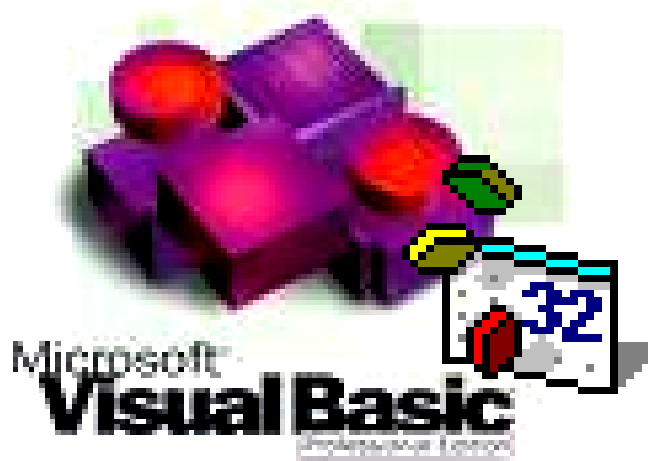


Modul Praktek Laboratorium Komputer

Visual Basic

Version 6.0

Microsoft



Di susun oleh :

Sopiyan Dalis

Akademi Manajemen Informatika & Komputer

BINA SARANA INFORMATIKA

Jakarta

2014

Kata Pengantar

Tidak bisa dipungkiri lagi, semua *software* mengacu ke sistem operasi Windows. Terlebih setelah diluncurkannya Windows 95, membuat para pengembang piranti lunak berlomba-lomba untuk membuat aplikasi berbasis Windows.

Tidak ketinggalan di bidang pemrograman, yang mau tidak mau harus menyesuaikan dengan Windows. Untungnya perusahaan *Microsoft* sebagai pelopor pemrograman berbasis Windows telah menciptakan Visual Basic, yang terbukti kehandalannya untuk membuat program hampir di segala bidang baik yang berkaitan dengan ilmu-ilmu pasti, database dan aplikasi multimedia.

Modul Visual Basic ini disusun untuk melengkapi kebutuhan belajar mahasiswa di Bina Sarana Informatika, khususnya bagi mereka yang ingin menekuni bidang pemrograman berbasis Windows. Agar bisa mendalami isi modul ini, maka siswa harus mengerjakan setiap contoh dan latihan soal yang ada di modul ini.

Akhirnya, mudah-mudahan modul ini bermanfaat bagi semua pembaca. Tidak lupa kritik dan saran demi kesempurnaan modul sangat diharapkan. Selamat mencoba.

Jakarta, Maret 2014

Penyusun

Daftar Isi

Kata Pengantar	ii
Daftar Isi	iii
BAB I PENGENALAN VISUAL BASIC.....	1
BAB II LINGKUNGAN KERJA VISUAL BASIC.....	3
2.1. Komponen Visual Basic	3
2.2. Latihan	4
BAB III KONSTANTA & VARIABEL.....	8
3.1. Aturan Penamaan Konstanta & Variabel	8
3.2. Deklarasi Konstanta & Variabel	8
3.3. Ruang Lingkup Konstanta & Variabel	9
3.4. Latihan	11
BAB IV OPERATOR	15
4.1. Operator Aritmatika	15
4.2. Operator Penggabungan String	15
4.3. Operator Pembandingan	16
4.4. Operator Logika	17
4.5. Operator Penugasan	17
4.6. Latihan	18
BAB V STRUKTUR KENDALI (BRANCHING).....	20
5.1. If	20
5.2. Select Case	21
5.3. Latihan	22
BAB VI PERULANGAN (LOOPING).....	23
6.1. Do ... Loop	23
6.2. For ... Next	24
6.3. While ... Wend	25
6.4. Latihan	26
BAB VII MODUL & PROSEDUR SUB.....	27
7.1. Mengenal Modul	27
7.2. Pembuatan Modul	28

	7.3. Mengenal Sub Procedure	28
	7.4. Pembuatan Prosedur Sub dalam Modul & Form	29
BAB VIII	PROCEDURE FUNCTION	30
	8.1. Mengenal Function Procedure	30
	8.2. Pembuatan Prosedur Fuction dalam Modul & Form	31
BAB IX	PENGERTIAN KOLEKSI OBJEK.....	32
	9.1. Pengertian Koleksi Objek	32
	9.2. Cara Mengakses Koleksi Objek	32
	9.3. Array	33
	9.4. Array Multi Dimensi	33
	9.5. Pembuatan Program Dengan Koleksi Objek & Array	34
BAB X	WINDOWS API.....	36
	10.1. Pengertian Windows API	36
	10.2. Pengertian DLL	37
	10.3. Aplikasi API Viewer	39
	10.4. Penggunaan Windows API Dalam Program	39
BAB XI	DATABASE	44
	11.1. Pengertian Database	44
	11.2. Tipe Field	44
	11.3. Membuat Database & Tabel	45
	11.4. Membuat Index	47
	11.5. Mengisi Data pada Tabel	48
BAB XII	KONTROL DATA	49
BAB XIII	RECORD SET.....	50
BAB XIV	PENCARIAN DATA BERURUTAN	56
BAB XV	PENCARIAN DATA DENGAN INDEX	59
BAB XVI	MENAMPILKAN DATA DENGAN DBGRID	67
BAB XVII	MENGGUNAKAN DATABASE LEWAT KODE PROGRAM	70
	17.1. Deklarasi Variabel Database	70
	17.2. Deklarasi Variabel Recordset	70
	17.3. Open Database	70
	17.4. Open Recordset	71
	17.5. Close	71

BAB XVIII MENGENAL SQL	73
18.1. Pengertian SQL	73
18.2. Menjalankan SQL Data Manager	73
18.3. Select ... From	73
18.4. Select ... From ... Where	75
18.5. Select ... From ... Where ... Order By	77
BAB XIX PEMBUATAN LAPORAN DENGAN DATA REPORT	79
BAB XX MENU	87
Daftar Pustaka	129

BAB I

PENGENALAN VISUAL BASIC

Basic adalah salah satu development tools untuk membangun aplikasi dalam lingkungan Windows. Dalam pengembangan aplikasi, Visual Basic menggunakan pendekatan Visual untuk merancang user interface dalam bentuk form, sedangkan untuk kodingnya menggunakan dialek bahasa Basic yang cenderung mudah dipelajari (Alam, 1999a). Visual Basic telah menjadi tools yang terkenal bagi para pemula maupun para developer. Visual Basic adalah bahasa pemrograman berbasis Microsoft Windows yang merupakan *Object Oriented Programming* (OOP), yaitu pemrograman berorientasi objek, Visual Basic menyediakan objek-objek yang sangat kuat, berguna dan mudah (Alam, 1999b).

Dalam lingkungan Windows, *User-interface* sangat memegang peranan penting, karena dalam pemakaian aplikasi yang kita buat, pemakai senantiasa berinteraksi dengan *User-interface* tanpa menyadari bahwa dibelakangnya berjalan instruksi-instruksi program yang mendukung tampilan dan proses yang dilakukan (Hadi, 2002).

Pada pemrograman Visual, pengembangan aplikasi dimulai dengan pembentukan *user interface*, kemudian mengatur properti dari objek-objek yang digunakan dalam user interface, dan baru dilakukan penulisan kode program untuk menangani kejadian-kejadian (event). Tahap pengembangan aplikasi demikian dikenal dengan istilah pengembangan aplikasi dengan pendekatan *Bottom Up* (Kurniadi, 2000).

Ada beberapa hal yang harus dipahami dalam mempelajari Visual Basic (Kurniadi, 2000):

- **Objek**

Sering disebut entity adalah sesuatu yang bisa dibedakan dengan lainnya. Pada dasarnya seluruh benda didunia ini bisa dikatakan sebagai objek, contoh : mobil, komputer, radio, dan lain-lain.

Dalam Visual Basic objek-objek yang dimaksud disebut kontrol. Jenis-jenis kontrol antara lain ; *Label, Text Box, Combo Box, List Box*, dan masih banyak lagi.

- **Properti**

Sering disebut atribut, adalah ciri-ciri yang menggambarkan suatu objek. Misalnya disebut objek mobil jika mempunyai ban, spion, rem, dan lain-lain.

- **Event**

Suatu kejadian yang menimpa objek. Bagaimana jika mobil didorong, ditabrak, dicat, dan sebagainya.

- **Metode**

Kemampuan yang dimiliki oleh suatu objek. Contohnya jika mobil berbelok, mundur, maju.

BAB II

LINGKUNGAN KERJA VISUAL BASIC

2.1. Komponen Visual Basic

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

Ada tiga icon dalam projek

- ☒ View Code : tampilkan jendela editor program
 - ☒ View Object : tampilkan bentuk formulir
 - ☒ Togle Folder : tampilkan folder tempat penyimpanan file
- b. **Form** adalah objek yang dipakai untuk tempat bekerja program aplikasi. Di dalamnya tempat diletakan objek-objek lainnya. Di dalam form ada grid (garis titik-titik yang berguana untuk pengaturan letak).
- c. **Toolbox** kotak alat berisi icon-icon atau komponen untuk memasukan objek tertentu kedalam form.

Tambah komponen : klik kanan pada toolbox -> komponen atau add tab

- d. **Propertis** adalah untuk menentukan setting suatu objek, menentukan cara kerja dari objek saat program dijalankan misal warna, hurup dan lain-lain.
- e. **Kode program** adalah serangkaian tulisan perintah yang akan dilaksanakan jika suatu objek dijalankan

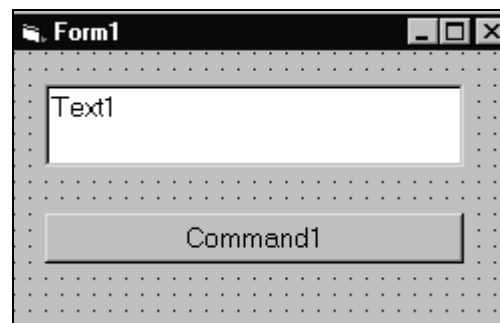
2.2. Latihan

Soal 1 (perancangan interface):

Perancangan interface dilakukan 3 tahap :

1) Membuat interface :

- Siapkan form baru.
- Letakkan TextBox dan CommandButton di dalam form.



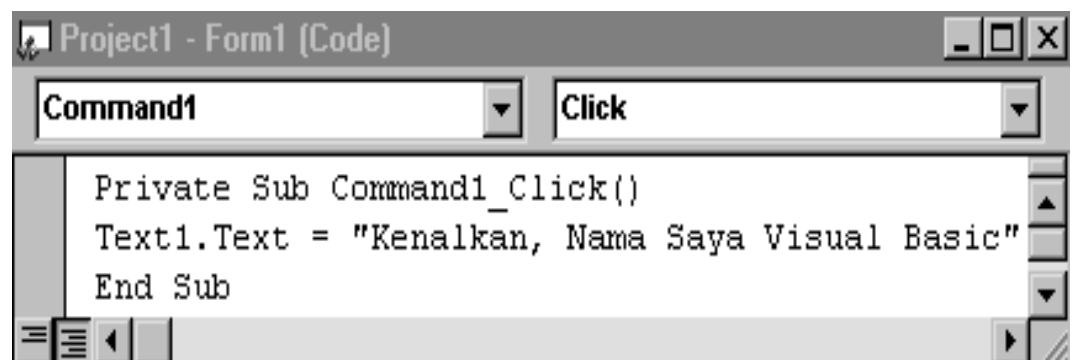
2) Mengatur properti :

Untuk mengatur properti, sesuaikan dengan tabel di bawah ini :

Objek	Properti	Nilai
Form	Caption	Program pengenalan
TextBox	Text	(kosongkan)
CommandButton	Caption	Perkenalan

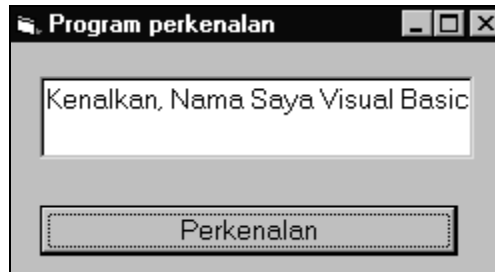
3) Menulis kode

Untuk menuliskan program, klik ganda pada tombol CommandButton (Perkenalan), lalu ketikkan program di bawah ini :



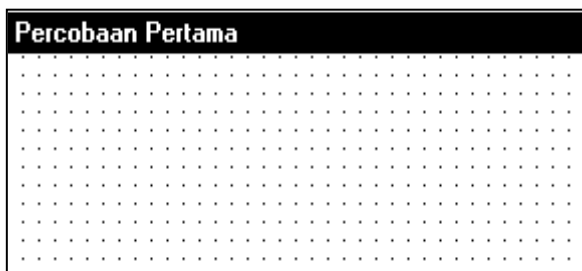
Untuk menjalankan program, tekan tombol F5.

Jika program sudah tampil, klik pada tombol Perkenalan, dan hasilnya seperti tampilan di bawah ini :



Soal 2 :

Buatlah form seperti di bawah ini

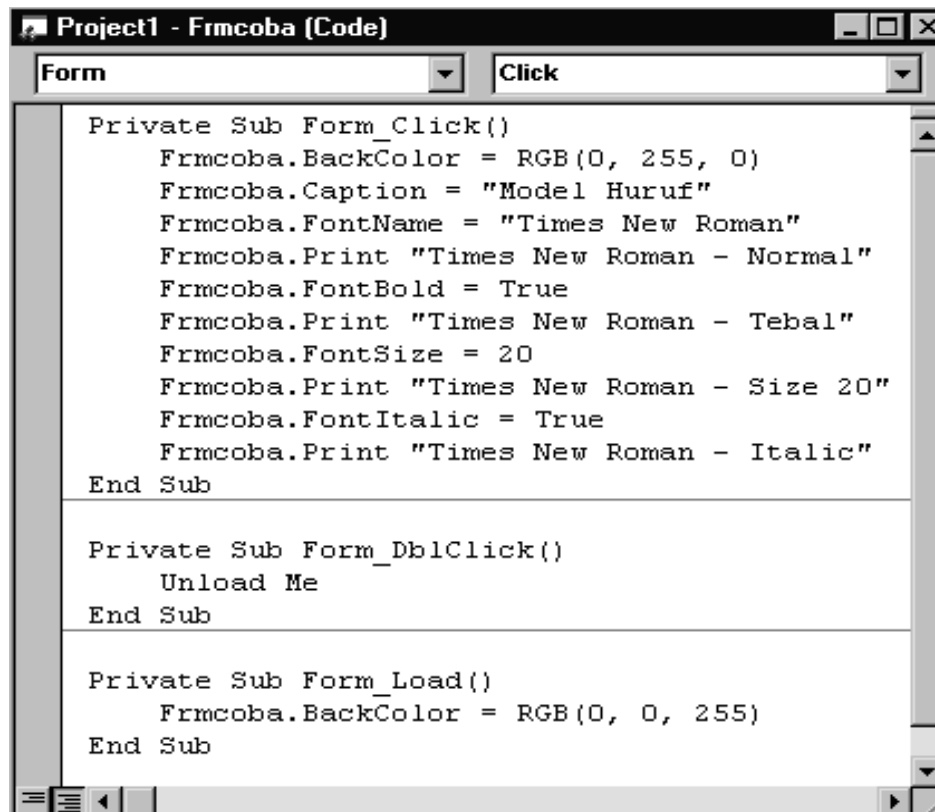


Dari jendela properti, ubahlah :

Name : frmCoba
Caption : Percobaan Pertama
ControlBox : False
MaxButton : False
MinButton : False

Pada form tersebut tampilkan 3 kejadian :

- 1) Saat program dijalankan : Latar belakang form berwarna biru.
- 2) Jika form di klik :
 - Latar belakang form berwarna hijau.
 - Judul form berubah : “Model Huruf”.
 - Tampil tulisan “Times New Roman” dengan variasinya.
- 3) Jika melakukan klik ganda pada form : Program dihentikan.



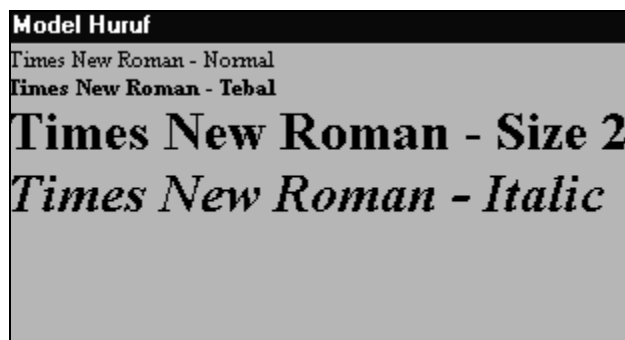
```

Project1 - FrmCoba [Code]
Form Click
Private Sub Form_Click()
    FrmCoba.BackColor = RGB(0, 255, 0)
    FrmCoba.Caption = "Model Huruf"
    FrmCoba.FontName = "Times New Roman"
    FrmCoba.Print "Times New Roman - Normal"
    FrmCoba.FontBold = True
    FrmCoba.Print "Times New Roman - Tebal"
    FrmCoba.FontSize = 20
    FrmCoba.Print "Times New Roman - Size 20"
    FrmCoba.FontItalic = True
    FrmCoba.Print "Times New Roman - Italic"
End Sub

Private Sub Form_DblClick()
    Unload Me
End Sub

Private Sub Form_Load()
    FrmCoba.BackColor = RGB(0, 0, 255)
End Sub

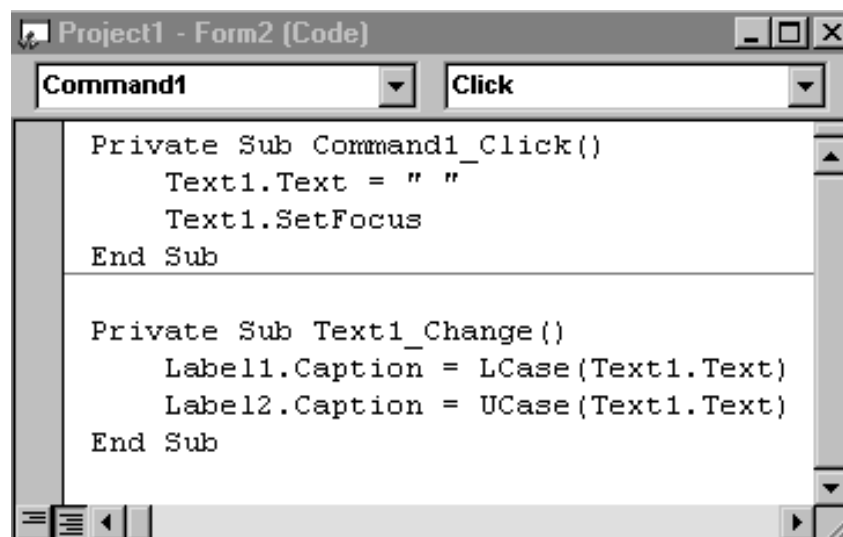
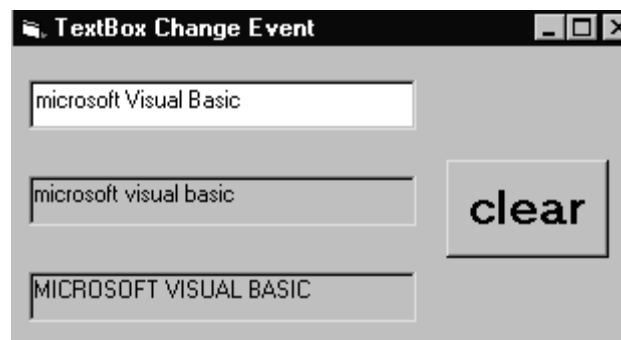
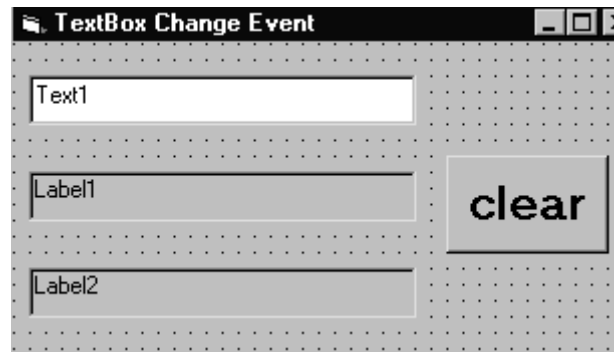
```



Soal 3 :

Contoh di bawah ini menggunakan Textbox dengan event Change dan metode Setfocus. Event Change adalah kejadian yang menyatakan objek tertentu sedang berubah nilainya. Seangkan metode Setfocus berfungsi untuk memindahkan mouse pointer ke objek yang dituju.

Pada saat Text1 diisi dengan kalimat, maka Label1 dan Label2 akan menampilkan isi text tersebut masing-masing huruf kecil dan huruf kapital. Jika tombol clear (Command1) ditekan, isi text dibersihkan.



BAB III

KONSTANTA & VARIABEL

Variabel adalah suatu tempat untuk menampung data yang nilainya selalu berubah. Sedangkan konstanta adalah suatu tempat untuk menampung data yang nilainya selalu tetap dan tidak pernah berubah.

3.1. Aturan Penamaan Konstanta & Variabel

- Harus diawali huruf.
- Boleh terdiri dari huruf, angka, dan garis bawah.
- Maksimal 255 karakter.
- Tidak boleh menggunakan reserved word.

3.2. Deklarasi Konstanta & Variabel

Deklarasi konstanta

Bentuk umum :

[Public | Private] Const nama_konstanta [AS tipe_konstanta] = eksplisit

Contoh :

```
Public Const POTONGAN = 0.05
Const NAMAPERSH = "PT ABC"
```

Contoh penggunaan dalam program

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

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
Dim Gaji as Long
```

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 "
Alamat$ = " Kelapa Gading "
Gaji$ = 1750000
```

3.3. Ruang Lingkup Konstanta & Variabel

Ruang lingkup konstanta atau variabel adalah ruang atau daerah dimana konstanta atau variabel yang dibuat dikenal (dapat dipakai) biasanya juga berhubungan dengan umur (waktu hidup) konstanta atau variabel.

Ada variabel yang dikenal diseluruh bagian program, 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, contoh :

```
Private Sub CmdTambah_Click()
    Dim Nama as String
    Dim Alamat as String * 30
    Static No as Integer
End Sub
```

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

Tabel 1-2. *Keyword pada deklarasi variabel*

3.4. Latihan

Soal 1:

Buatlah form baru dan tambahkan kode untuk event klik pada form tersebut.

```

Option Explicit

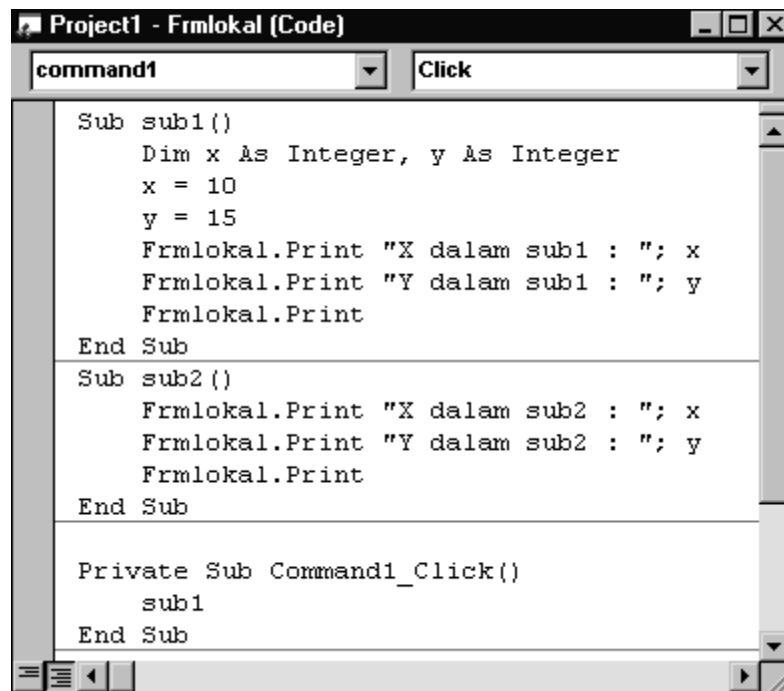
Private Sub Form_Click()
    Dim biaya As Currency
    Dim pajak As Currency
    Dim total As Currency
    biaya = 5000
    pajak = 500
    total = biaya + pajak
    Form3.Print "biaya : "; biaya
    Form3.Print "pajak : "; pajak
    Form3.Print "----- +"
    Form3.Print "total : "; total
End Sub

```

Soal 2 (variabel lokal)

Buatlah form dan ketik kode program seperti di bawah ini :

Setelah tombol Sub2 diklik, x dan y dalam modul Sub2 tidak ada isinya karena tidak bisa memanggil variabel yang ada dalam modul Sub1.



```

Project1 - Frmlokal [Code]
command1 Click

Sub sub1()
    Dim x As Integer, y As Integer
    x = 10
    y = 15
    Frmlokal.Print "X dalam sub1 : "; x
    Frmlokal.Print "Y dalam sub1 : "; y
    Frmlokal.Print
End Sub

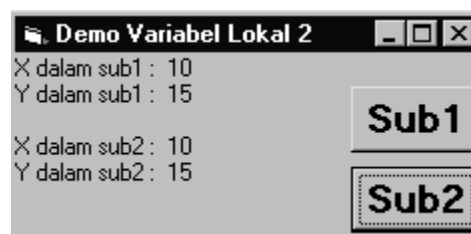
Sub sub2()
    Frmlokal.Print "X dalam sub2 : "; x
    Frmlokal.Print "Y dalam sub2 : "; y
    Frmlokal.Print
End Sub

Private Sub Command1_Click()
    sub1
End Sub

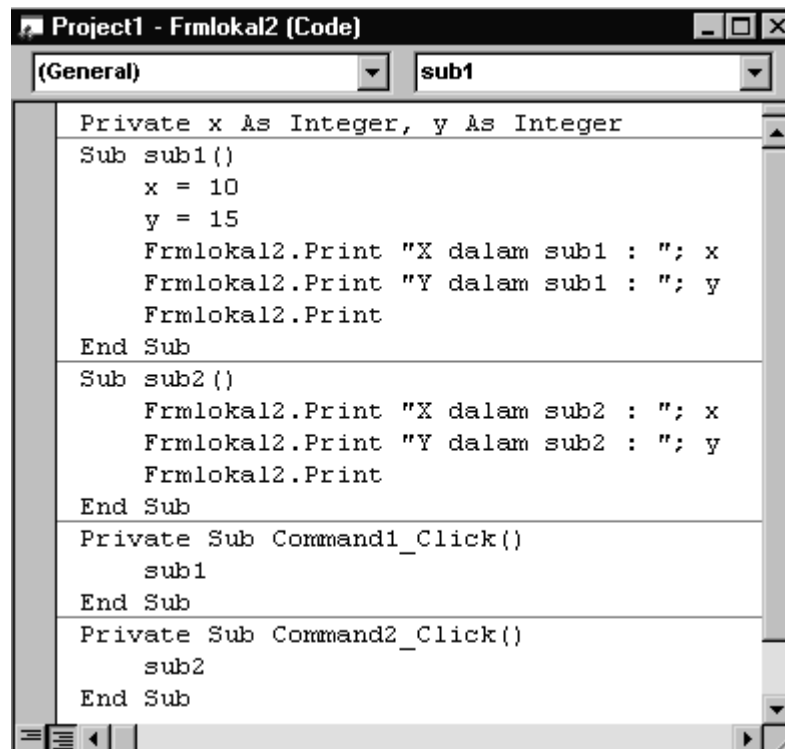
```

Soal 3 (variabel lokal)

Buatlah form dan ketik kode program seperti di bawah ini :



Dalam modul Sub2 x dan y telah didefinisikan, sehingga jika Sub2 diklik maka akan menampilkan hasil.



```

Project1 - Frmlokal2 [Code]
(General) sub1

Private x As Integer, y As Integer
Sub sub1()
    x = 10
    y = 15
    Frmlokal2.Print "X dalam sub1 : "; x
    Frmlokal2.Print "Y dalam sub1 : "; y
    Frmlokal2.Print
End Sub

Sub sub2()
    Frmlokal2.Print "X dalam sub2 : "; x
    Frmlokal2.Print "Y dalam sub2 : "; y
    Frmlokal2.Print
End Sub

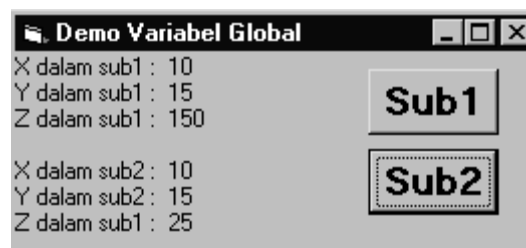
Private Sub Command1_Click()
    sub1
End Sub

Private Sub Command2_Click()
    sub2
End Sub

```

Soal 4 (variabel global)

Buatlah form dan ketik kode program seperti di bawah ini :



```

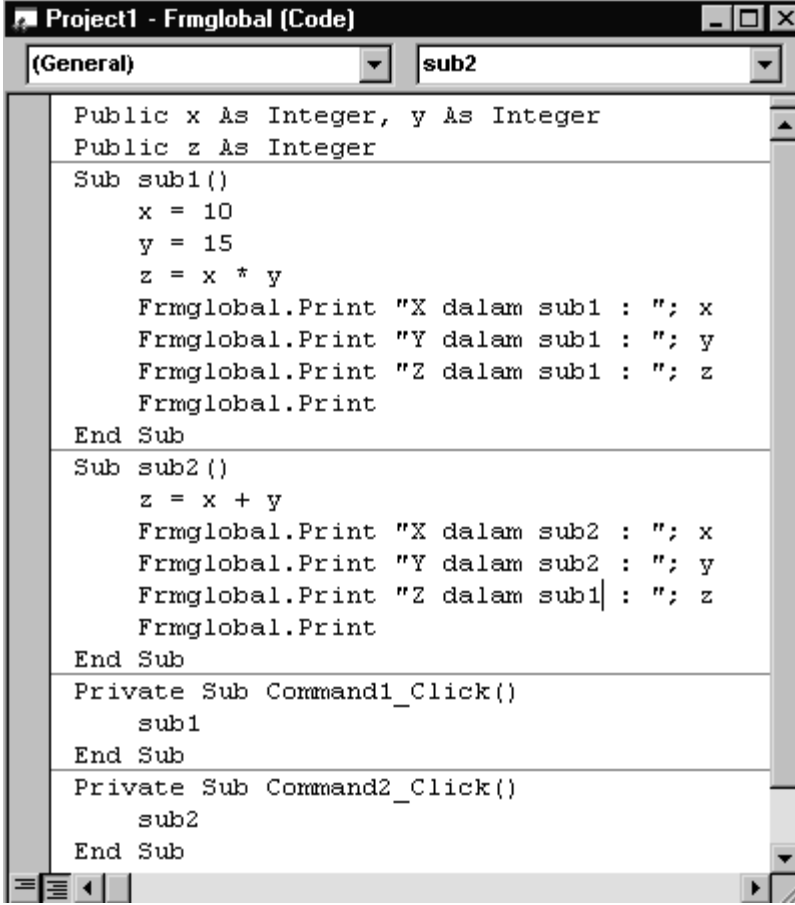
Demo Variabel Global

X dalam sub1 : 10
Y dalam sub1 : 15
Z dalam sub1 : 150

X dalam sub2 : 10
Y dalam sub2 : 15
Z dalam sub1 : 25

Sub1
Sub2

```



```
Public x As Integer, y As Integer
Public z As Integer

Sub sub1()
    x = 10
    y = 15
    z = x * y
    Frmglobal.Print "X dalam sub1 : "; x
    Frmglobal.Print "Y dalam sub1 : "; y
    Frmglobal.Print "Z dalam sub1 : "; z
    Frmglobal.Print
End Sub

Sub sub2()
    z = x + y
    Frmglobal.Print "X dalam sub2 : "; x
    Frmglobal.Print "Y dalam sub2 : "; y
    Frmglobal.Print "Z dalam sub1 : "; z
    Frmglobal.Print
End Sub

Private Sub Command1_Click()
    sub1
End Sub

Private Sub Command2_Click()
    sub2
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, Tabel 1-1, menunjukkan operator dan urutan operasinya dari atas kebawah.

4.1. Operator Aritmatika

Contoh :

$A = 1 + 2 * 3$ 'Akan menghasilkan 7

$B = (1 + 2) * 3$ 'Akan menghasilkan 9

Aritmatika	Komparasi	Logika
Pangkat (^)	Sama (=)	Not
Negatif (-)	Tidak sama (<>)	And
Kali dan Bagi (*, /)	Kurang dari (<)	Or
Pembagian bulat (\)	Lebih dari (>)	Xor
Sisa Bagi (Mod)	Kurang dari atau sama (<=)	Eqv
Tambah dan Kurang (+, -)	Lebih dari atau sama (>=)	Imp
Pengabungan String (&)	Like	

Tabel 1-1. Contoh operator Aritmatika

4.2. Operator Penggabungan String (Like)

Salah satu operator yang menarik untuk dibahas adalah operator Like. Operator Like digunakan untuk operasi pencocokan pola pada string yang akan sangat membantu programmer.

Syntax :

result = string Like pattern

Operator	Keterangan
?	Sembarang karakter tunggal
*	Nol atau lebih karakter
#	Sembarang digit tunggal (0-9)
[charlist]	Sembarang karakter yang berada dalam charlist
[!charlist]	Sembarang karakter yang tidak berada dalam charlist

Tabel 1-2. Contoh operator penggabungan string (Like)

Contoh :

- MyCheck = "aBBa" Like "a*a" ' Returns True.
- MyCheck = "F" Like "[A-Z]" ' Returns True.
- MyCheck = "F" Like "[!A-Z]" ' Returns False.
- MyCheck = "a2a" Like "a#a" ' Returns True.
- MyCheck = "aM5b" Like "a[L-P]#[!c-e]" ' Returns True.
- MyCheck = "BAT123khg" Like "B?T*" ' Returns True.
- MyCheck = "CAT123khg" Like "B?T*" ' Returns False.

4.3. Operator Pembandingan (Relasi)

Operator ini digunakan untuk membandingkan suatu data atau ekspresi dengan data ekspresi pula dan 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 objek

Tabel 1-3. Contoh operator pembandingan (relasi)

4.4. Operator Logika

Digunakan untuk mengekspresikan satu atau lebih data logika menghasilkan data logika baru.

Operator	Keterangan
Not	Tidak
And	Dan
Or	Atau
Xor	Exclusive Or
Eqv	Ekivalen
Imp	Implikasi

Tabel 1-4. Contoh operator Logika

4.5. Operator Penugasan

Operator ini disimbolkan dengan tanda sama dengan (=) dan berfungsi untuk memasukan suatu data kedalam suatu variabel, bentuknya sbb:

< Variabel > = < ekspresi >

Contoh : Bilangan = 0
 Harga = 500
 Banyak = 5
 Jumlah = Harga * Banyak

4.6. Latihan

Buatlah program sederhana dibawah ini dengan menggunakan CommandButton, Label, dan TextBox. Event yang digunakan Change, Click, Keypress Load, dan SetFocus.

Urutan Proses :

- Pertama form dipanggil, tombol Hitung, Bersihkan, dan Keluar tidak aktif.
- User mengisi Jumlah Beli, dan jika ditekan enter, kursor akan terfokus menuju Harga Satuan.
- Saat mengisi Harga Satuan, tombol Hitung aktif, dan jika ditekan maka hasil perhitungan muncul pada label Bayar. (Bayar = Jumlah Beli * Harga Satuan)
- Saat itu tombol Bersihkan, dan Keluar bisa diakses.
- Objek-objek terkait :

Objek	Properti	Nilai
Form1	Caption	Latihan
Label1	Aligment	2 – Center
	BackStyle	1 – Opaque
	BorderStyle	1 – Fixed Single
	Caption	STRUK BELANJA
	Font – Size	Arial Bold – 20
Label2	Aligment	1 – Right Justify

	BackStyle BorderStyle Caption Font – Size	0 – Transparent 0 – None Jumlah Beli MS San Serif – 14
Label3	Aligment BackStyle BorderStyle Caption Font – Size	1 – Right Justify 0 – Transparent 0 – None Harga Satuan MS San Serif – 14
Label4	Aligment BackStyle BorderStyle Caption Font – Size	1 – Right Justify 0 – Transparent 0 – None Bayar MS San Serif – 14
Label5	Caption Name	(kosong) lblHasil
Text1	Name	txtJumlah
Text2	Name	txtHarga
Command1	Caption Name ToolTipText	&Hitung cmdHitung Jumlah * Harga
Command2	Caption Name	&Bersihkan cmdBersih
Command1	Caption Name ToolTipText	&Keluar cmdKeluar Keluar dari aplikasi

BAB V

STRUKTUR KENDALI (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

5.1. If

If...Then

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

```
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 bukan nol.

If...Then...Else

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

```

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 :

```

If JlhRec > 0 Then
    Posisi = 1
Else
    Posisi = 0
End If

```

5.2. 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

```

5.3. Latihan

Buatlah form seperti berikut (nama objek disesuaikan sendiri) :

Ketentuan :

- Pada waktu mengisi nilai ujian 1 dan 2, tombol hitung aktif.
- Setelah Rata-rata dan Keterangan terisi, tombol Hitung hilang berganti menjadi Clear.
- $\text{Rata-Rata} = \frac{\text{Nilai Ujian 1} + \text{Nilai Ujian 2}}{2}$
- Keterangan: Jika Rata-Rata antara 85 s/d 100 maka Sangat Memuaskan

Jika Rata-Rata antara 75 s/d 84 maka Cukup Memuaskan

Jika Rata-Rata antara 60 s/d 74 maka Memuaskan

Selain itu Kurang Memuaskan

BAB VI

PERULANGAN (LOOPING)

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

6.1. 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 condition
    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 condition

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

Do Until condition
 statements
Loop

Do
 statements
Loop Until condition

6.2. 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 counter = start **To** end [Step increment]
 statements
Next [counter]

Argumen counter, start, end, dan increment semuanya adalah numerik. Yang dilakukan Visual Basic ketika melakukan For Next adalah:

- a) Menset nilai counter sama dengan start.
- b) Mencoba apakah counter lebih besar dari end. Jika ya, Visual Basic meninggalkan looping tersebut.
- c) Menjalankan statements tersebut.

- d) Menaikkan counter dengan 1 atau dengan nilai step yang ditentukan
- e) Mengulangi langkah 2 sampai 4

Code berikut akan mencetak semua item yang terdapat pada suatu daftar list.

```
Private Sub Form_Click ()
Dim l As Integer
For i = 0 To List1.ListCount-1
Print List1.List(l)
Next
End Sub
```

6.3. While ... Wend

While – Wend digunakan untuk mengeksekusi suatu blok statement dengan jumlah perulangan yang tak terhingga.

Berikut ini adalah statement While –Wend, yang dijalankan selama kondisi benar :

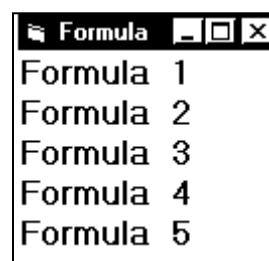
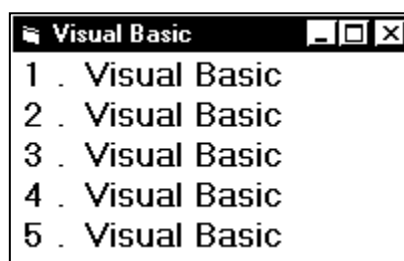
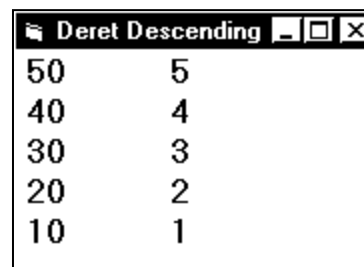
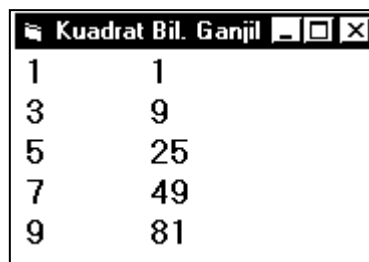
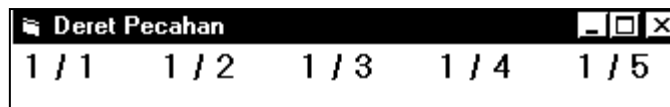
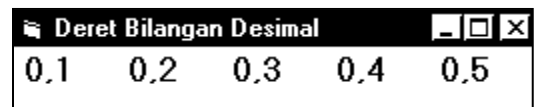
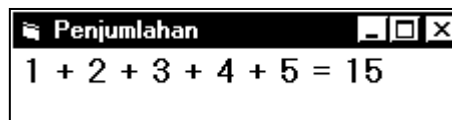
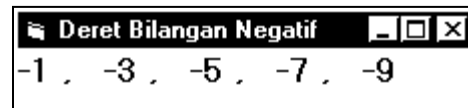
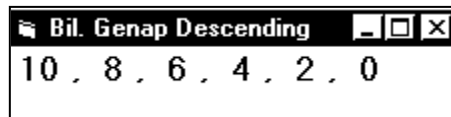
```
While condition
statements
Wend
```

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

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

6.4. Latihan

Buatlah program dengan menggunakan perulangan For atau Do atau While, sehingga bisa menampilkan deret bilangan seperti di bawah ini :

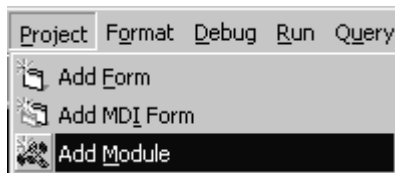


BAB VII

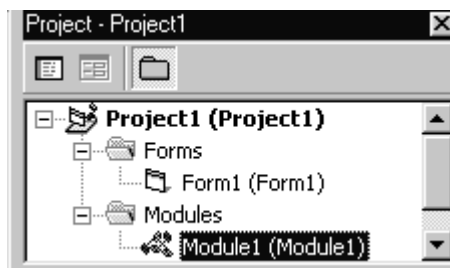
MODUL & PROSEDUR SUB

7.1. Mengenal Modul

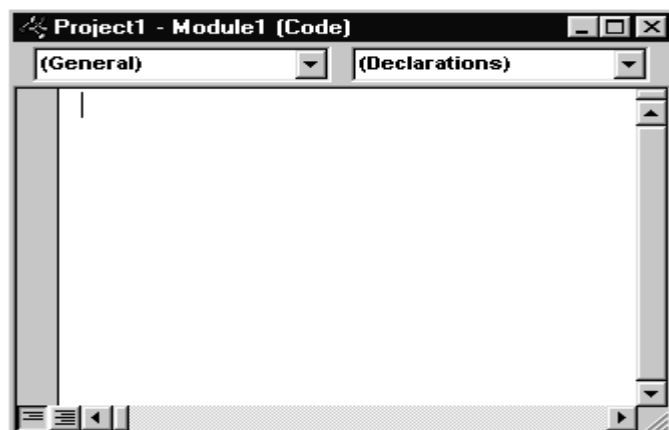
Visual Basic menyediakan module yang dapat digunakan untuk memuat fungsi, subrutin, konstanta, variabel dan type ciptaan yang akan disediakan bagi keseluruhan project. Untuk membuat module dalam project anda dapat menggunakan perintah **Project, Add Module**.



Sesaat setelah anda menambahkan module, maka pada Jendela Project Explorer akan nampak Module1



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



7.2. Pembuatan Modul

Pada Modul anda dapat mendeklarasikan variabel dengan awalan DIM, PRIVATE maupun PUBLIC, dimana awalan DIM dan PRIVATE anda membentuk variabel Module Level (hanya berlaku didalam pemakaian module bersangkutan), sedangkan awalan PUBLIC akan menghasilkan variabel Global yang akan berfungsi bagi keseluruhan Program.

Contoh :

```
Dim A as Integer      'Variabel A adalah Module Level
Private B as Integer  'Variabel B adalah Module Level
Public C as Integer   'Variabel C dapat digunakan oleh program
                      keseluruhan
```

7.3. Mengenal Sub Procedure

Syntax penulisan Sub procedure:

```
[Private Public] [Static] Sub namaprosedur (argumen-argumen)
  pernyataan-pernyataan
End Sub
```

Setiap kali procedure dipanggil, maka pernyataan-pernyataan yang berada di antara Sub dan End Sub akan dijalankan.

Argumen pada procedure adalah nilai yang akan dilewatkan saat pemanggilan procedure.

Di Visual Basic Sub Procedure dapat dibagi atas dua yaitu :

- **General Procedure**, procedure yang diaktifkan oleh aplikasi
- **Event Procedure**, Procedure yang diaktifkan oleh system sebagai respon terhadap event.

Contoh, sub Tengah yang dapat digunakan untuk menampilkan form ketengah Layar, dimana x adalah parameter yang merupakan form yang akan dibuat ketengah layar.

```
Sub Tengah(x As Form)
    x.Top = (Screen.Height - x.Height) \ 2
    x.Left = (Screen.Width - x.Width) \ 2
End Sub
Private Sub Form_Load()
    Call Tengah(Me)
End Sub
```

7.4. Pembuatan Prosedur Sub dalam Modul & Form

Berikut ini adalah contoh prosedur untuk menghitung gaji dan tunjangan yang dibuat dalam modul.

```
Private Sub Form_Click()
    Hitung_Gaji
End Sub

Sub Hitung_Gaji()
    Gaji = Inputbox("Masukkan Gaji Pokok", "Input Gaji")
    Tunj = Inputbox("Masukkan Tunjangan", "Input Tunjangan")

    Gaber = Val(Gaji) + Val(Tunj)

    If Gaji = "" Or Tunj = "" Then
        MsgBox ("Tidak Ada Angka Yang Dimasukkan")
        Exit Sub
    Else
        MsgBox ("Gaji Bersi = " & Str(Gaber))
    End If
End Sub
```

BAB VIII

PROCEDURE FUNCTION

8.1. Mengetahui Function Procedure

Pada Visual Basic telah tersedia berbagai fungsi bawaan seperti Sqr, Cos, dan Chr, tetapi fungsi-fungsi yang tersedia tersebut bersifat umum dan kadang-kadang tidak memenuhi kebutuhan programmer, untuk keperluan tersebut anda dapat menciptakan fungsi-fungsi sendiri yang dikenal dengan Function procedure.

Adapun syntax penulisan function procedure :

```
[Private|Public] [Static] Function namaprocedure (argumen-argumen)  
[As type] statements  
End Function
```

Ada tiga perbedaan antara function dan procedure :

- Umumnya anda dapat memanggil suatu function dengan mengikuti nama function sisi kanan dari statement atau ekspresi. (returnvalue = function()).
- Function memiliki type data seperti suatu variabel. Ini menentukan type yang dari nilai yang dikembalikan.
- Nilai kembali dimasukkan ke namafunction itu sendirinya, dan suatu function dapat menjadi bagian dari suatu ekspresi yang panjang.

Contoh Fungsi ciptaan untuk mengembalikan nama bulan dari suatu tanggal dalam bahasa Indonesia.

```
Function Bulan(x As Date)  
Dim sRet As String  
Select Case Month(x)  
Case 1: sRet = "Januari"  
Case 2: sRet = "Februari"  
Case 3: sRet = "Maret"  
Case 4: sRet = "April"  
Case 5: sRet = "Mei"
```

```

Case 6: sRet = "Juni"
Case 7: sRet = "Juli"
Case 8: sRet = "Agustus"
Case 9: sRet = "September"
Case 10: sRet = "Oktober"
Case 11: sRet = "Nopember"
Case 12: sRet = "Desember"
Case Else
    sRet = "tidak sah"
End Select
    Bulan = sRet
End Function

```

8.2. Pembuatan Prosedur Fuction dalam Modul & Form

Program di bawah ini adalah contoh program pencarian kelulusan siswa berdasarkan nilai rata-rata dengan menggunakan fungsi.

```

Function Grade(Rata As Integer) As String
    Select Case Rata
        Case 95 To 100
            Ket = "Sangat Memuaskan"
        Case 85 To 94
            Ket = "Cukup Memuaskan"
        Case 60 To 84
            Ket = "Memuaskan"
        Case 45 To 59
            Ket = "Tidak Memuaskan"
        Case Else
            Ket = "Mengecewakan"
    End Select
    Grade = Ket
End Function

```

Untuk menjalankan fungsi diatas, gunakan program pemanggilan fungsi di bawah ini:

```

Dim N As Integer
N = 90
Print "Nilai Rata-Rata      : " & Str(N)
Print "Keterangan      : " & Grade(N)

```

BAB IX

KOLEKSI OBJEK & ARRAY

9.1. Pengertian Koleksi Objek

Objek-objek yang digunakan dalam form biasanya berdiri sendiri. Dengan demikian jika anda ingin mengubah property ke semua objek tersebut, hal tersebut menyebabkan 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.

9.2. 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.

9.3. Array

Array adalah sebuah variabel memori yang berisi beberapa data sejenis. Data-data tersebut disimpan dengan nomor urut tertentu yang disebut index. Jika X adalah variabel array, maka bisa 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

Pendefinisian Variabel Array :

Dim nama_variabel (index) **As** tipe_variabel

Contoh

Dim hari(6) As String '7 index : 0, 1, 2, ..., 6
Dim bulan(11) As String '12 index : 0, 1, 2, ..., 12

Index (jumlah elemen pada array) selalu diulai dari 0. Jika diinginkan nilai awal bisa diubah dengan perintah *option base* pada bagian *declaration*.

Option Base 1
Dim hari(7) 'index : 1, 2, ..., 7

Tanpa *Option Base* cara yang sama adalah :

Dim bulan(1 to 12) 'index : 1, 2, ..., 12

9.4. Array Multi Dimensi

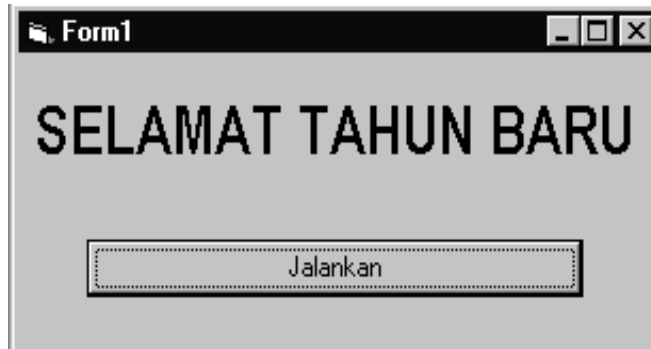
Array berdimensi 2 terdiri dari baris dan kolom, contohnya pada tabel atau matrik. Pendefinisiannya adalah sebagai berikut :

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

9.5. Pembuatan Program Dengan Koleksi Objek & Array

Contoh program koleksi dengan controls :

Buatlah form dengan lay-out sbb :



Caranya :

1. Buat project baru
2. Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Label1	Caption Font	Selamat tahun Baru Arral Narrow, Bold , 22
Command1	Caption	Jalankan

3. Klik ganda kontrol Command1, lalu ketikkan kode programnya :


```
Private Sub Command1_Click()
    Form2.Show
    Form2.Visible = False
    Command1.Visible = False
End Sub
```
4. Tambahkan form baru dengan memilih menu : **Project, Add Form**
5. Letakkan sebuah kontrol timer pada form2.
6. Klik ganda form2, lalu ketikkan kode programnya :


```
Private Sub Form_Load()
    Timer1.Interval = 100
End Sub
```
7. Klik ganda kontrol timer1 pada form2, lalu ketikkan kode programnya :

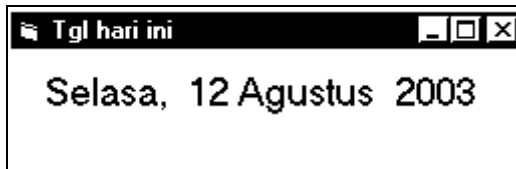

```
Private Sub Timer1_Timer()
    For Each kontrol In Form1.Controls
        'form1.control adalah contoh penggunaan koleksi ini.
        kontrol.Left = kontrol.Left + 120
```

Next kontrol
End Sub

8. Jalankan aplikasi tersebut dengan menekan **F5**

Contoh program dengan array

Buatlah form yang berisi tampilan hari, dan tanggal saat ini.



Option Base 1

Private Sub Form_Load()

Dim hari(7) As String

Dim bulan(12) As String

hari(1) = "Minggu"

hari(2) = "Senin"

hari(3) = "Selasa"

hari(4) = "Rabu"

hari(5) = "Kamis"

hari(6) = "Jumat"

hari(7) = "Sabtu"

bulan(1) = "Januari"

bulan(2) = "Februari"

bulan(3) = "Maret"

bulan(4) = "April"

bulan(5) = "Mei"

bulan(6) = "Juni"

bulan(7) = "Juli"

bulan(8) = "Agustus"

bulan(9) = "September"

bulan(10) = "Oktober"

bulan(11) = "November"

bulan(12) = "Desember"

har = Weekday(Date)

tgl = Day(Date)

bul = Month(Date)

thn = Str(Year(Date))

Label1.FontSize = 16

Label1.Caption = hari(har) & ", " & Str(tgl) & " " & bulan(bul) & " " & thn

End Sub

BAB X

WINDOWS API

10.1. Pengertian Windows API

Windows API (*Application Programming Interface*) merupakan sekumpulan fungsi-fungsi eksternal yang terdapat dalam file-file perpustakaan Windows (library windows) atau file library lainnya yang dapat digunakan. Fungsi ini dapat menangani semua yang berhubungan dengan Windows, seperti pengaksesan disk, interface printer, grafik windows, kotak dialog (buka file, simpan file, memilih font, memilih warna, dll), Windows shell, setting sistem informasi, penanganan file, mengakses sistem registry, memainkan musik, dsb. Fungsi ini menyediakan banyak fitur-fitur standar untuk semua program yang berbasis Windows.

Semua fungsi Windows API hampir terdapat dalam direktori sistem milik Windows (C:\Windows\System) dan paling banyak berekstensi .DLL yang digunakan oleh sistem operasi Windows. Selain itu fungsi ini juga memastikan secara konsisten penggunaan semua sumber yang terdapat dalam Windows. File-file itulah yang disebut dengan **Windows API**.

Windows API digunakan ketika program harus melakukan pengkodean yang tidak terdapat dalam bahasa Visual Basic. Misalnya menentukan kecepatan kursor berkedip dalam Visual Basic tidak ada. Dengan memanggil fungsi Windows API yaitu **GetCaretBlinkTime** (untuk mengembalikan nilai kursor berkedip dalam satuan milidetik) dapat ditentukan kecepatan kursor berkedip.

10.2. Pengertian DLL

File library Windows DLL (*Dynamic Link Library*) yang selanjutnya disebut DLL adalah kode yang sudah dikompilasi dan dapat digunakan oleh program lain. DLL biasanya ditulis dengan bahasa C/C++, Delphi atau bahasa lainnya yang mendukung sistem operasi Windows.

Dengan memanggil fungsi yang terdapat dalam DLL, pemakai dapat mengakses ribuan fungsi yang berhubungan dengan sistem Windows, dengan kualitas sebaik yang digunakan dalam bahasa yang digunakan. Berikut ini nama-nama library milik Windows yang sering dan paling banyak digunakan dalam Windows API.

Nama File DLL	Deskripsi File
Advapi32.DLL	Library yang mendukung fungsi-fungsi keamanan dan rutin-rutin registry.
Comdlg32.DLL	Standar kotak dialog Windows.
Gdi32.DLL	Penanganan grafik Windows.
Kernel32.DLL	Fungsi sistem operasi Windows 32-bit.
Lz32.DLL	Fungsi kompresi file.
Mpr.DLL	Fungsi Internet.
Netapi32.DLL	Fungsi jaringan.
Shell32.DLL	Library shell 32-bit.
User32.DLL	Penanganan rutin user interface.
Version.DLL	Versi Windows.
Winmm.DLL	Fungsi-fungsi multimedia Windows.
Winspool.drv	Fungsi-fungsi printer spooler.

Hal yang perlu dilakukan untuk menggunakan fungsi-fungsi dalam file library Windows yaitu dengan menspesifikasikan di mana fungsi tersebut ditemukan dan menyediakan informasi yang dibutuhkan fungsi pada bagian pendeklarasian fungsi Windows

Langkah pertama adalah mendeklarasikan fungsi di dalam bagian modul, dengan sintaks sbb :

Declare Function NamaFungsi Lib “NamaLibrary” [Alias “AliasFungsi”]
 [[[ByVal/ByRef] variabel [As type] [, [ByVal/ByRef] variabel [As type]]...]] As Type

Jika fungsi tersebut tidak mengembalikan nilai, maka gantilah kata

Function dengan kata **Sub**.

Declare Sub NamaFungsi Lib “NamaLibrary” [Alias “AliasFungsi”]
 [[[ByVal/ByRef] variabel [As type] [, [ByVal/ByRef] variabel [As type]]...]]

Keterangan :

Sintaks	Keterangan
Declare Function/Sub	Sintaks deklarai fungsi / prosedur.
NamaFungsi	Nama fungsi yang ada dalam library.
Lib	Sintaks deklarai fungsi / prosedur.
NamaLibrary	Nama library yang terdapat nama fungsi.
Alias	Sintaks alias.
AliasFungsi	Penamaan fungsi, untuk membedakan dengan fungsi yang sama.
ByVal/ByRef	Type dari parameter fungsi.
Variable	Nama variabel yang digunakan fungsi.
As tipe	Tipe dari variabel.
As Type	Nilai yang dikembalikan oleh fungsi.

Contoh dari pendeklarasian fungsi API adalah sbb :

Public Declare Function SetWindowText Lib “User32” Alias
 “SetWindowTextA” (ByVal hwnd As Long, ByVal lpString As
 String) As Long

Untuk mencari nama fungsi API, tipe data dan konstanta, gunakan aplikasi viewer yang disertakan dalam Visual Basic. Jika mendeklarasikan dalam form, modul maupun class module, perintah Declare diawali dengan kata Private atau Public untuk mendefinisikan jangkauan fungsi dalam kode. Setelah fungsi tersebut dideklarasikan, pemakai dapat menggunakannya dalam program.

10.3. Aplikasi API Viewer

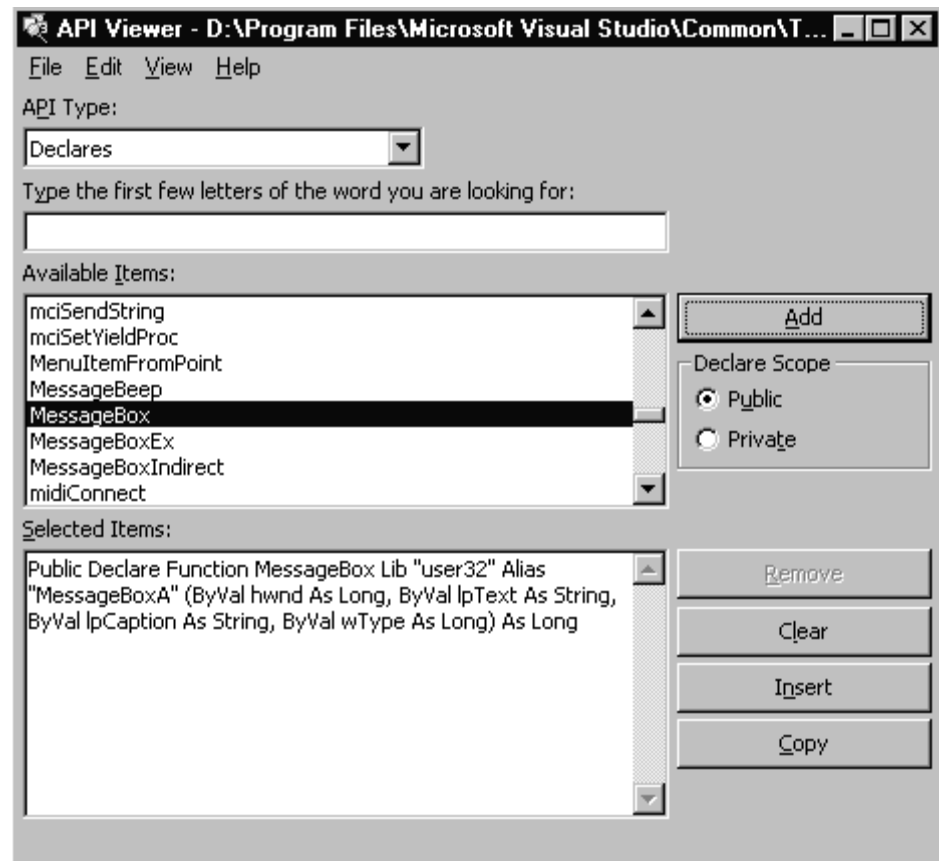
Aplikasi API Viewer menyediakan hampir seluruh deklarasi fungsi, konstanta, dan tipe data yang terdapat dalam file teks (.TXT) maupun dalam file database (.MDB). Setelah menemukan fungsi yang diinginkan, pemakai dapat mengkopinya ke dalam modul yang nantinya dapat digabungkan dengan kode program Visual Basic. Untuk memanggil API Viewer lakukan langkah-langkah berikut ini :

- ☒ Dari menu Add-Ins, pilih Add-Ins Manager
- ☒ Pilih VB API Viewer
- ☒ Pada Load Behavior klik Loaded/Unloaded
- ☒ Klik Ok
- ☒ Klik API Viewer dari Add-Ins Menu
- ☒ Buka Text atau Database File yang ingin dilihat.
- ☒ Pilih item yang ingin dilihat.

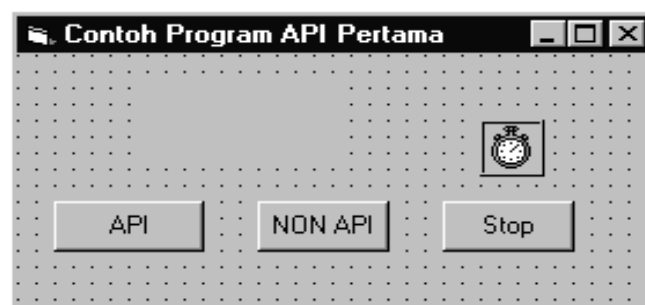
10.4. Penggunaan Windows API dalam Program.

Contoh program 1 :

1. Buat projek baru dan tambahkan modul dengan memilih **Project, Add Module, Open**.
2. Aktifkan **API Viewer**.
3. Klik menu **Add-Ins**, pilih **API Viewer**.
4. Klik menu **File**, pilih **Load Text File**.
5. Pilih file **win32api**, klik **Open**.
6. Pada **Available Items**, pilih **MessageBox**.
7. Klik **Add** dan **Insert**. Klik **Yes**.



8. Tutup jendela Windows API.
9. Tutup jendela module.
10. Tambahkan kontrol timer, label dan tiga buah command button.



11. Atur property objeknya sbb :

Kontrol	Properti	Nilai
Form1	Caption	Contoh Program Pertama API
Label1	Caption	kosongkan
Timer	Interval	2
Command1	Caption	API
Command2	Caption	NON API
Command3	Caption	Stop

12. Ketikkan listing berikut :

```
Private Sub Command1_Click()
    MsgBox Me.hwnd, "API menyebabkan timer tidak berhenti",
    "Msgbox", vbOKOnly
End Sub

Private Sub Command2_Click()
    MsgBox "Non API menyebabkan timer berhenti", vbOKOnly,
    "msgbox"
End Sub

Private Sub Command3_Click()
    End
End Sub

Private Sub Timer1_Timer()
    Label1.Caption = Time
End Sub
```

13. Simpan Projek.

14. Jalankan program.

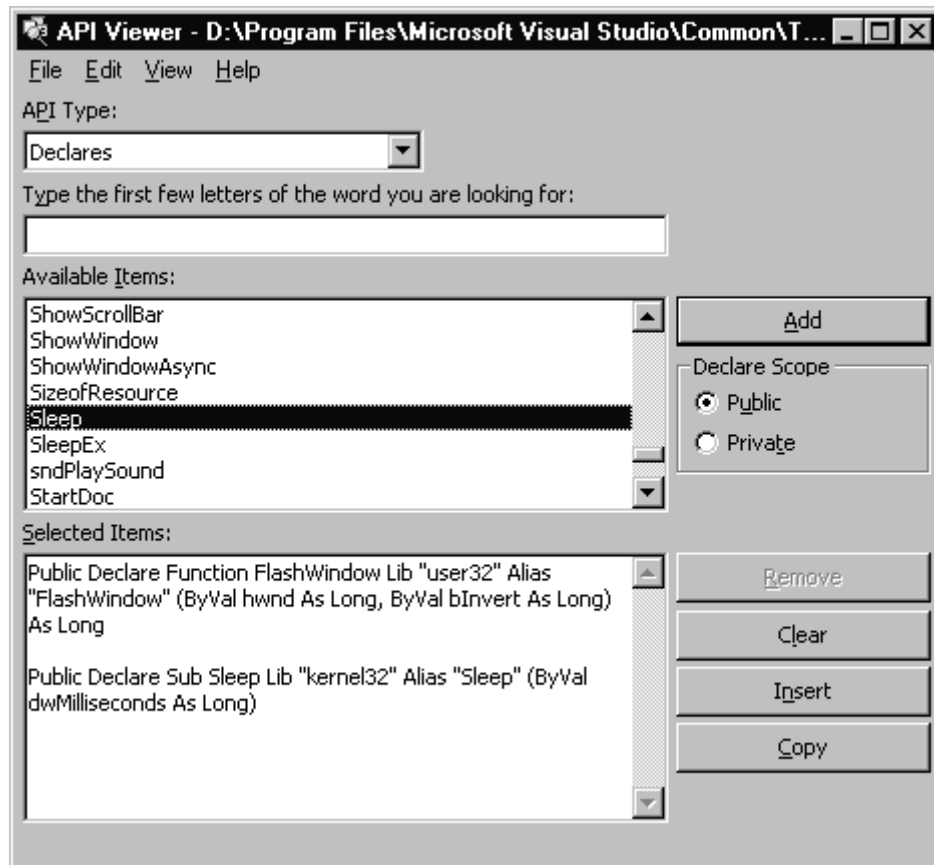
15. Hasilnya sbb :



Contoh program 2 :

1. Buat projek baru dan tambahkan modul dengan memilih **Project, Add Module, Open**.
2. Aktifkan **API Viewer**.
3. Klik menu **Add-Ins**, pilih **API Viewer**.

4. Klik menu **File**, pilih **Load Text File**.
5. Pilih file **win32api**, klik **Open**.
6. Pada **Available Items**, klik double **FlashWindow** dan **Sleep**
7. Klik **Insert**. Klik **Yes**.



8. Tutup jendela Windows API.
9. Tutup jendela module.
10. Tambahkan kontrol command button.



11. Atur property objeknya sbb :

Kontrol	Properti	Nilai
Form1	Caption	Form Berkedip
Command1	Caption	Buat Form Berkedip

12. Ketikkan listing berikut :

```
Private Sub Command1_Click()  
Dim x As Long  
For c = 1 To 10  
    x = FlashWindow(Form1.hwnd, 1)  
    Sleep 500  
Next c  
x = FlashWindow(Form1.hwnd, 0)  
End Sub
```

13. Simpan Projek.

14. Jalankan program.

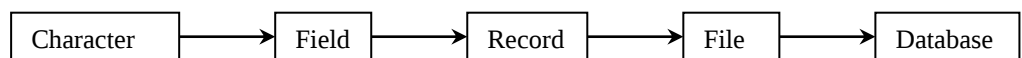
BAB XI

DATABASE

11.1. 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.

11.2. Tipe Field

Ada beberapa tipe field yang digunakan dalam pembuatan tabel :

- **Long**

Digunakan untuk menampung data numerik untuk bilangan bulat.

- **Currency**

Digunakan untuk menampung data numerik. Kelebihannya adalah currency tidak akan melakukan pembulatan bilangan pada operasi matematik.

- **Single**

Digunakan untuk menampung data numerik untuk bilangan pecahan.

- **Double**

Fungsinya sama dengan Single.

- **Date/Time**

Digunakan untuk menampung data tanggal atau jam.

- **Text**

Digunakan untuk menyimpan data karakter, dengan daya tampung sebesar 255 karakter.

- **Memo**

Sama dengan text, tetapi dapat menampung sampai dengan 64.000 karakter.

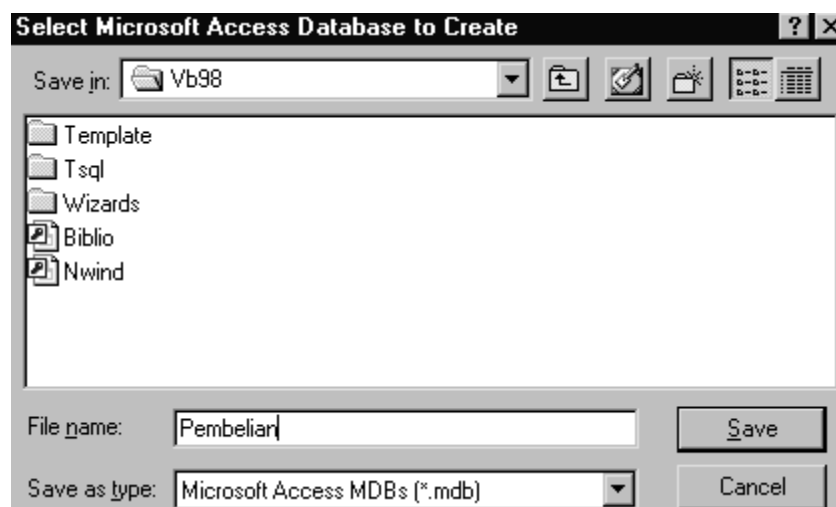
11.3. Membuat Database dan Tabel

Untuk membuat database dengan Visual Data Manager, lakukan langkah-langkah berikut:

- Pilih menu Add-Ins pada menu utama Visual Basic, lalu pilih Visual Data Manager.



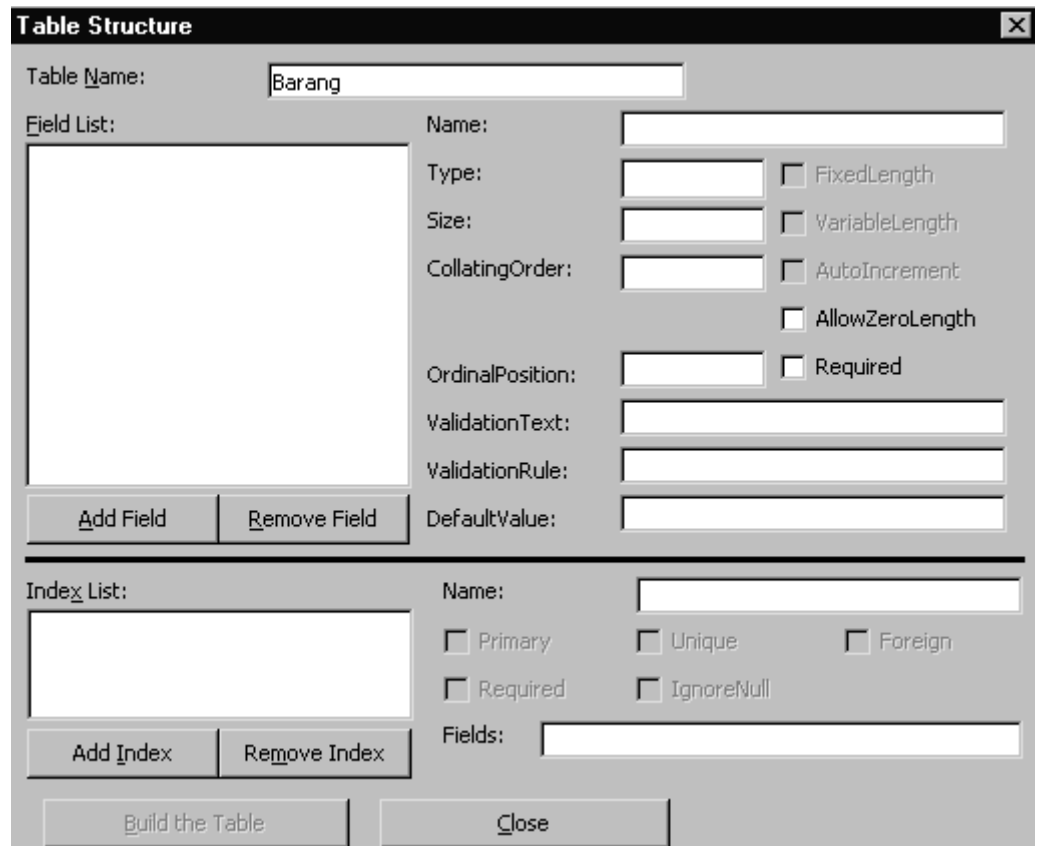
- Pilih : File/New/Microsoft Access/Version 7.0 MDB.



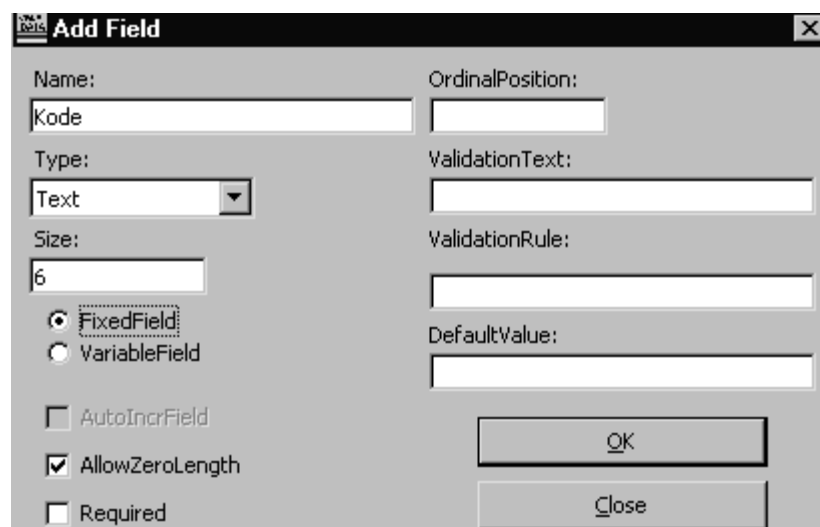
- c) Berikanlah nama pada database yang akan dibuat.



- d) Klik kanan pada Properties, pilih New Table.



- e) Ketik tulisan Barang pada Table Name, lalu klik Add Field.



f) Buatlah beberapa field sebagai berikut :

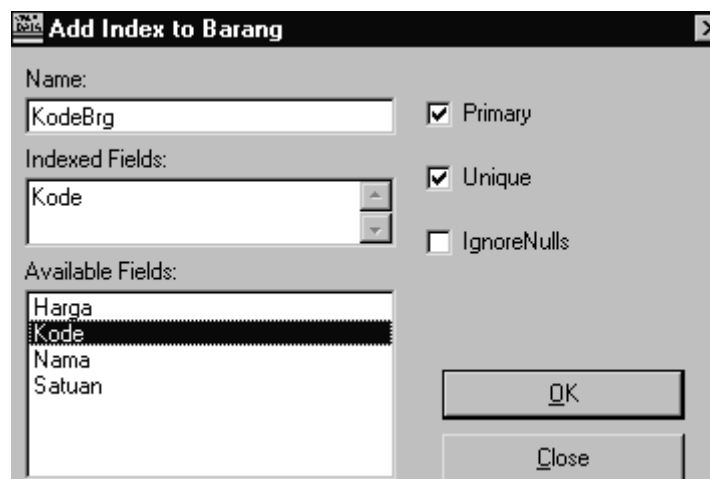
Name	Type	Size
Kode	Text	6
Nama	Text	30
Satuan	Text	5
Harga	Single	

g) Jika telah selesai mengisi field, klik Build the Table, klik Close (yang terdapat pada Table Structure).

11.4. Membuat index

Index data adalah urutan data pada tabel database. Index data diperlukan terutama untuk pencarian data dengan cepat dan pengelompokan data.

- Untuk membuat index data gunakan cara sebagai berikut :
- Buka database pembelian.
- Setelah tampil Structure Tabel, pilih Add Index.



- Ketik nama index (misalnya kodebrg) pada kolom Name.
- Klik kunci index pada Available Fields.
- Tentukan pilihan index (Primary, Unique, IgnoreNulls). Primary digunakan sebagai pengenal suatu record, jika tabel yang bersangkutan dihubungkan dengan tabel lain (dalam satu tabel hanya boleh ada satu primary key). Unique

dipilih jika ingin index bersifat unik. IgnoreNulls digunakan untuk mengabaikan field yang kosong.

11.5. Mengisi Data pada Tabel

Untuk mengisi data pada tabel, dobel klik pada nama tabel tersebut.



Fungsi menu yang ada pada tabel barang :

Add : menambah record baru.

Update : mengoreksi record.

Delete : menghapus record aktif.

Find : mencari record.

Refresh : menampilkan ulang setelah record mengalami perubahan.

Close : menutup tabel.

BAB XII

KONTROL DATA

Kontrol data merupakan interface (antar muka) yang menghubungkan sebuah form dengan file data.

Di dalam kontrol data memiliki beberapa properti yaitu :

- ***Connect***

Hubungan file data, bisa bersumber dari Access, Dbase, Excel, Foxpro, Lotus, Paradox, dan file Text.

- ***Database Name***

Menyatakan nama file database (.mdb) yang digunakan.

- ***Record Source***

Diisi tabel yang digunakan.

- ***EOF Action***

Pengecekan pergerakan kursor. Jika kursor sampai pada akhir file, maka terdapat 3 pilihan properti ; MoveLast (record terakhir), EOF (akhir file), AddNew (menambah record baru).

- ***Record Set Type***

Tipe record ; 0 – Tabel, 1 – Dynaset, 2 – Snapshot.

BAB XIII

RECORD SET

Record set adalah status record yang menunjukkan tindakan apa yang akan dilakukan terhadap record tersebut dengan beberapa metode antara lain :

Metode	Keterangan
AddNew	Menambah record baru
MoveFirst	Pindah ke record pertama
MovePrevious	Pindah 1 record ke depan
MoveNext	Pindah 1 record ke belakang
MoveLast	Pindah ke record terakhir
Delete	Menghapus record
FindFirst	Mencari dari record pertama
NoMatch	Record yang dicari tidak ada yang sama
EOF	Berada di record yang terakhir
BOF	Berada di record yang pertama
Edit	Merubah record
Update	Menyimpan perubahan record
CancelUpdate	Membatalkan perubahan record

Untuk lebih memahami mengenai metode record set, lakukan langkah-langkah berikut ini :

Buatlah form dengan lay-out sbb :

Hasil program adalah sbb :

Caranya :

1. Buat project baru
2. Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Caption	Formulir Data Barang
Data1	Name Caption Connect DatabaseName RecordsetType RecordSource	dbPembelian Database Pembelian Acces C:\Zidan\Pembelian.mdb 0 – Table Barang
Label1	Caption	Kode Barang
Label2	Caption	Nama
Label3	Caption	Satuan
Label4	Caption	Harga Satuan
Text1	Name DataField DataSource Text	Txtkode Kode dbPembelian <kosongkan>
Text2	Name DataField DataSource Text	Txtnama Nama dbPembelian <kosongkan>
Text3	Name DataField DataSource Text	Txtsatuan Satuan dbPembelian <kosongkan>

Text4	Name DataField DataSource Text	Txtharga Harga dbPembelian <kosongkan>
Command1	Name Caption Picture Style ToolTipText	CmdTop <kosongkan> C:\Program Files\Microsoft Visual Studio\Vfp98\Wizard\Graphics\Top 1 – Graphical Top
Command2	Name Caption Picture Style ToolTipText	CmdPrevious <kosongkan> C:\Program Files\Microsoft Visual Studio\Vfp98\Wizard\Graphics\Previous 1 – Graphical Previous
Command3	Name Caption Picture Style ToolTipText	CmdNext <kosongkan> C:\Program Files\Microsoft Visual Studio\Vfp98\Wizard\Graphics\Next 1 – Graphical Next
Command4	Name Caption Picture Style ToolTipText	CmdLast <kosongkan> C:\Program Files\Microsoft Visual Studio\Vfp98\Wizard\Graphics\Bottom 1 – Graphical Last
Command5	Name Caption Picture Style ToolTipText	CmdNew <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Bitmaps\Off CtlBr\Large\Color\New 1 – Graphical New
Command6	Name Caption Picture Style ToolTipText	CmdSave <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Bitmaps\Off CtlBr\Large\Color\Save 1 – Graphical Save
Command7	Name Caption Picture Style ToolTipText	CmdUndo <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Bitmaps\Off CtlBr\Large\Color\Undo 1 – Graphical Undo

Command8	Name Caption Picture Style ToolTipText	CmdEdit <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Bitmaps\Off CtlBr\Small\Color\Spell 1 – Graphical Edit
Command9	Name Caption Picture Style ToolTipText	CmdDel <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Bitmaps\Off CtlBr\Large\Color\Cut 1 – Graphical Delete
Command10	Name Caption Picture Style ToolTipText	CmdFind <kosongkan> C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icon\Misc\Binocular 1 – Graphical Find
Command11	Name Caption Picture Style ToolTipText	CmdExit E&xit <None> 1 – Graphical Selesai

3.Listing Programnya sbb :

```

Dim tambah As Boolean
Sub nonaktif()
Dim ctl As Control
For Each ctl In FormBrg
    If TypeName(ctl) = "textbox" Then
        ctl.Locked = True
    End If
    If TypeName(ctl) = "combobox" Then
        ctl.Locked = True
    End If
Next
End Sub

Private Sub cmdback_Click()
dbpembelian.Recordset.MovePrevious
If dbpembelian.Recordset.BOF Then
    dbpembelian.Recordset.MoveFirst
    MsgBox "Sudah di awal record", 16, "Informasi"
End If
End Sub

```

```

Private Sub cmdfind_Click()
Dim mkode As String * 6
mkode = InputBox("Ketik Kode Barangnya :", "Pencarian Data Barang")
msyarat = "Kode = " & mkode & " "
dbpembelian.Recordset.Index = "Kodebrg"
dbpembelian.Recordset.Seek "=", mkode
If dbpembelian.Recordset.NoMatch Then
    MsgBox "Record Kode Barang " + cr + "Tidak ada", 16, "Informasi"
End If
End Sub

```

```

Private Sub cmddel_Click()
del = MsgBox("Record ini dihapus ?", 36, "Konfirmasi")
If del = 6 Then
    dbpembelian.Recordset.Delete
    dbpembelian.Recordset.MoveNext
    If dbpembelian.Recordset.EOF Then
        dbpembelian.Recordset.MoveLast
    End If
End If
End Sub

```

```

Private Sub cmdedit_Click()
    dbpembelian.Recordset.Edit
    aktif
End Sub

```

```

Private Sub cmdexit_Click()
    Unload Me
End Sub

```

```

Private Sub cmdlast_Click()
    dbpembelian.Recordset.MoveLast
    MsgBox "Sudah di akhir record", 16, "Informasi"
End Sub

```

```

Private Sub cmdnew_Click()
    tambah = True
    dbpembelian.Recordset.AddNew
    kosong
    aktif
End Sub

```

```

Private Sub cmdnext_Click()
dbpembelian.Recordset.MoveNext
If dbpembelian.Recordset.EOF Then
    dbpembelian.Recordset.MoveLast
    MsgBox "Sudah di akhir record", 16, "Informasi"
End If
End Sub

```

```

Private Sub cmdsave_Click()
    If tambah = True Then
        dbpembelian.Recordset.Update
    Else
        dbpembelian.Recordset.Update
    End If
    nonaktif
End Sub

Private Sub cmdtop_Click()
    dbpembelian.Recordset.MoveFirst
    MsgBox "Sudah di awal record", 16, "Informasi"
End Sub

Private Sub cmdundo_Click()
    dbpembelian.Recordset.CancelUpdate
    dbpembelian.Recordset.MoveLast
    nonaktif
End Sub

Private Sub Form_Load()
    nonaktif
    tambah = False
End Sub

Sub kosong()
    Dim ctl As Control
    For Each ctl In FormBrg
        If TypeName(ctl) = "textbox" Then
            ctl.Text = ""
        End If
        If TypeName(ctl) = "combobox" Then
            ctl.Text = ""
        End If
    Next
End Sub

Sub aktif()
    Dim ctl As Control
    For Each ctl In FormBrg
        If TypeName(ctl) = "textbox" Then
            ctl.Locked = False
        End If
        If TypeName(ctl) = "combobox" Then
            ctl.Locked = False
        End If
    Next
End Sub

```

4. Jalankan aplikasi tersebut dengan menekan **F5**
5. Simpan program diatas dengan nama LAT01.

BAB XIV

PENCARIAN DATA BERURUTAN

Pencarian berurutan (sekuensial) adalah mencari data secara terurut dari suatu lokasi (record) sampai ke lokasi lainnya dengan memeriksa record per record. Pencarian ini akan memakan waktu yang cukup lama jika jumlah datanya besar. Untuk melakukan pencarian data berurutan, user dapat menggunakan metode **Find** sbb :

`object.recordset.Find <string criteria>`

Contoh :

`dbpembelian.recordset.Find "Kode = ' PS.02 ' "`

Metode **Find** akan menyebabkan pencarian dilakukan dari posisi record sekarang ke record terakhir. Oleh karena itu, jika user akan mencari dari awal record, maka penunjuk record harus digerakkan ke awal record dengan metode **Move First**.

Untuk memahami pencarian data berurutan lakukan langkah-langkah sebagai berikut :

1. Buka LAT01.
2. Simpan proyek dengan nama baru LAT02 (File, Save Project As)
3. Simpan form dengan nama baru LAT02 (File, Save Lat01.frm As)
4. Tambahkan objek seperti gambar berikut ini (Perhatikan bahwa objek Frame1 dibuat sebelum objek di dalamnya).



5. Ubah setting property objeknya sbb :

Objek	Properti	Setting
Frame	Caption	Cari Data
Label	Caption	Kode Barang
Text	Name	Txtcari
	Caption	(kosong)
Command1	Name	Cmdcari
	Caption	Cari
Command2	Name	Cmdexit
	Caption	Exit

6. Tampilkan editor kode program dan ketikkan kode programnya :

```

Private Sub cmdcari_Click()
    Dim mkode As String * 6
    Dim msyarat As String
    mkode = txtcari.Text
    msyarat = "Kode = " & mkode & " "
    'cari data pertama yang nomornya sama dengan mkode
    dbpembelian.Recordset.Find msyarat
    If dbpembelian.Recordset.EOF Then
        X = MsgBox("Data tidak ditemukan!", vbOKOnly, "Pencarian  
Kode Barang")
        dbpembelian.Recordset.MoveFirst
    End If
End Sub

Private Sub cmdexit_Click()
    End
End Sub

Private Sub txtcari_GotFocus()

```

```
cmdcari.Default = True  
End Sub
```

```
Private Sub txtcari_LostFocus()  
cmdcari.Default = False  
End Sub
```

7. Simpan lagi proyek dan form dengan nama sama
8. Jalankan program.

BAB XV

PENCARIAN DATA DENGAN INDEX

Untuk melakukan pencarian dengan cepat diperlukan index data. Dengan adanya index data, maka data akan terurut sehingga pencarian tidak harus berurutan, tetapi dapat dengan melakukan pencarian biner, yaitu selalu mencari data dengan cara membandingkan data yang dicari dengan data tengah suatu set data. Jika data tidak sama, maka set data akan dipecah lagi dan membandingkan data yang dicari dengan tengah set data yang sesuai setelah dipecah. Demikian seterusnya, sehingga langkah pencarian akan kecil (sedikit) dan didapatkan kecepatan pencarian yang baik (cepat), walaupun datanya besar.

Untuk memahami pencarian data dengan index lakukan langkah-langkah sebagai berikut :

1. Buka LAT02.
2. Simpan proyek dengan nama baru LAT03 (File, Save Project As)
3. Simpan form dengan nama baru LAT03 (File, Save Lat02.frm As)
4. Karena objek **Adodc** tidak mempunyai metoda pencarian data dengan index, maka harus diganti dengan objek **Data**.
5. Tentukan setting property objek data menjadi sbb :

Objek	Setting
Name	dbPembelian
Caption	Database Pembelian
Connect	Acces
Database Name	C:\Zidan\Pembelian.mdb
RecordsetType	0 – Table (karena property Index dan metode Seek hanya dapat digunakan pada setting ini)
RecordSource	Barang

6. Tampilkan editor kode program dan editlah kode programnya menjadi sbb :

```
Private Sub cmdcari_Click()
    Dim mkode As String * 6
    mkode = txtcari.Text
    msyarat = "Kode = " & mkode & " ' '"
    dbpembelian.Recordset.Index = "Kodebrg"
    dbpembelian.Recordset.Seek "=", mkode
    If dbpembelian.Recordset.NoMatch Then
        X = MsgBox("Data tidak ditemukan!", vbOKOnly,
            "Pencarian Kode Barang")
    End If
End Sub

Private Sub cmdexit_Click()
    End
End Sub

Private Sub Form_Activate()
    txtcari.SetFocus
End Sub

Private Sub txtcari_GotFocus()
    cmdcari.Default = True
End Sub

Private Sub txtcari_LostFocus()
    cmdcari.Default = False
End Sub
```

7. Simpan lagi proyek dan form dengan nama sama.
8. Jalankan program.

Contoh program 2:

- Buatlah form dengan lay-out sbb :

- Hasil program adalah sbb :

- Caranya :

- 1) Buka LAT03.
- 2) Simpan proyek dengan nama baru LAT04 (File, Save Project As)
- 3) Simpan form dengan nama baru LAT04 (File, Save Lat03.frm As)
- 4) Hapus kontrol yang tidak diperlukan.
- 5) Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Caption	Contoh Kontrol Data

Data1	Name	dbPembelian
	Caption	Database Pembelian
	Connect	Acces
	Database Name	C:\Zidan\Pembelian.mdb
	RecordsetType	0 – Table
	RecordSource	Barang
Text1	Name	TxtKode
	Text	Kosongkan
	DataSource	dbPembelian
	DataField	Kode
Text2	Name	TxtNama
	Text	Kosongkan
	DataSource	dbPembelian
	DataField	Nama
Text3	Name	TxtSatuan
	Text	Kosongkan
	DataSource	dbPembelian
	DataField	Satuan
Text4	Name	TxtHarga
	Text	Kosongkan
	DataSource	dbPembelian
	DataField	Harga
Command1	Name	CmdTop
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Vfp98\Wizard\Graphic
	ToolTipText	Top
Command2	Name	CmdBack
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Vfp98\Wizard\Graphic
	ToolTipText	Back
Command3	Name	CmdNext
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Vfp98\Wizard\Graphic
	ToolTipText	Next
Command4	Name	CmdLast
	Caption	Kosongkan
	Style	1 - Graphical

	Picture	C:\Program Files\Visual Studio\Vfp98\Wizard\Graphic
	ToolTipText	Last
Command5	Name	CmdNew
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	New
Command6	Name	CmdSave
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Save
Command7	Name	CmdUndo
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Undo
Command8	Name	CmdEdit
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Edit
Command9	Name	CmdDel
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Del
Command10	Name	CmdCari
	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Find
Command11	Name	CmdExit

	Caption	Kosongkan
	Style	1 - Graphical
	Picture	C:\Program Files\Visual Studio\Common\Graphic\Bitmaps\OffC tlrBr\Large\Color
	ToolTipText	Exit

- 6) Tampilkan editor kode program dan ketiklah kode programnya sbb :

```

Dim tambah As Boolean
Sub nonaktif()
Dim ctl As Control
For Each ctl In FormBrg
If TypeName(ctl) = "textbox" Then
ctl.Locked = True
End If
If TypeName(ctl) = "combobox" Then
ctl.Locked = True
End If
Next
End Sub

Private Sub cmdback_Click()
dbpembelian.Recordset.MovePrevious
If dbpembelian.Recordset.BOF Then
dbpembelian.Recordset.MoveFirst
MsgBox "Sudah di awal record", 16, "Informasi"
End If
End Sub

Private Sub cmdcari_Click()
Dim mkode As String * 6
mkode = InputBox("Ketik Kode Barangnya :", "Pencarian Data  
Barang")
msyarat = "Kode = " & mkode & " "
dbpembelian.Recordset.Index = "Kodebrg"
dbpembelian.Recordset.Seek "=", mkode
If dbpembelian.Recordset.NoMatch Then
MsgBox "Record Kode Barang " + cr + "Tidak ada", 16, "Informasi"
End If
End Sub

Private Sub cmddel_Click()
del = MsgBox("Record ini dihapus ?", 36, "Konfirmasi")
If del = 6 Then
dbpembelian.Recordset.Delete
dbpembelian.Recordset.MoveNext
If dbpembelian.Recordset.EOF Then

```

```

        dbpembelian.Recordset.MoveLast
    End If
End If
End Sub

Private Sub cmdedit_Click()
    dbpembelian.Recordset.Edit
    aktif
End Sub

Private Sub cmdexit_Click()
    Unload Me
End Sub

Private Sub cmdlast_Click()
    dbpembelian.Recordset.MoveLast
    MsgBox "Sudah di akhir record", 16, "Informasi"
End Sub

Private Sub cmdnew_Click()
    tambah = True
    dbpembelian.Recordset.AddNew
    kosong
    aktif
End Sub

Private Sub cmdnext_Click()
    dbpembelian.Recordset.MoveNext
    If dbpembelian.Recordset.EOF Then
        dbpembelian.Recordset.MoveLast
        MsgBox "Sudah di akhir record", 16, "Informasi"
    End If
End Sub

Private Sub cmdsave_Click()
    If tambah = True Then
        dbpembelian.Recordset.Update
    Else
        dbpembelian.Recordset.Update
    End If
    nonaktif
End Sub

Private Sub cmdtop_Click()
    dbpembelian.Recordset.MoveFirst
    MsgBox "Sudah di awal record", 16, "Informasi"
End Sub

Private Sub cmdundo_Click()
    dbpembelian.Recordset.CancelUpdate

```

```
dbpembelian.Recordset.MoveLast
nonaktif
End Sub
```

```
Private Sub Form_Load()
nonaktif
tambah = False
End Sub
```

```
Sub kosong()
Dim ctl As Control
For Each ctl In FormBrg
    If TypeName(ctl) = "textbox" Then
        ctl.Text = ""
    End If
    If TypeName(ctl) = "combobox" Then
        ctl.Text = ""
    End If
Next
End Sub
```

```
Sub aktif()
Dim ctl As Control
For Each ctl In FormBrg
    If TypeName(ctl) = "textbox" Then
        ctl.Locked = False
    End If
    If TypeName(ctl) = "combobox" Then
        ctl.Locked = False
    End If
Next
End Sub
```

- 7) Jalankan aplikasi tersebut dengan menekan **F5**

BAB XVI

MENAMPILKAN DATA DENGAN DBGRID

Penanganan data dalam bentuk table sangatlah penting karena bentuk table memberikan beberapa keuntungan, antara lain dapat melihat beberapa record (baris) sekaligus dalam satu table dan lebih jelas keterkaitan serta urutan antardatanya.

Untuk memahami menampilkan data dengan DBGrid lakukan langkah-langkah sebagai berikut :

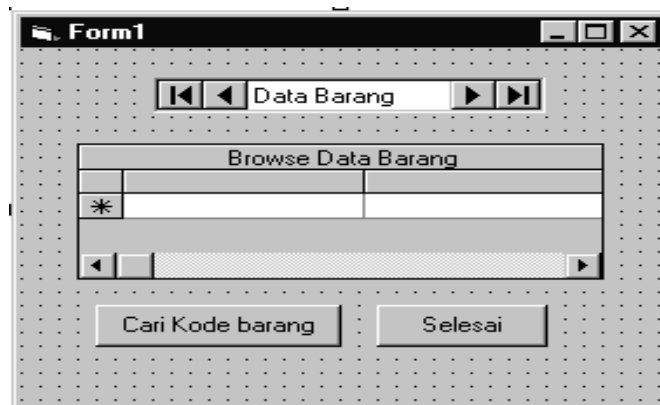
1. Buat form baru dengan menu File lalu New Project, lalu pilihlah VB Enterprise / Profesional Edition Controls.
2. Tampilkan jendela Form.
3. Klik ganda pada icon Data pada jendela Toolbox.
4. Atur property objek Data sbb :

Objek	Setting
Name	Data1
Appearance	3D
Caption	Data Barang
Connect	Acces
Database Name	C:\Zidan\Pembelian.mdb
EOFAction	AddNew
RecordsetType	Dynaset
RecordSource	Barang

5. Klik ganda icon DBGrid pada jendela toolbox dan aturlah. Jika tidak ada icon DBGrid, maka tambahkan melalui menu Project lalu Component (Ctrl + T), kemudian pilih Microsoft Data Bound Control 5.0 (SP3)
6. Atur property objek DBGrid sbb :

Objek	Setting
Name	DBGrid
AllowAddNew	True
AllowArrows	True
AllowDelete	True
AllowUpdate	True
Caption	Browse Data Barang
ColumnHeaders	True
DataSource	Data1
TabAction	Grid Navigation

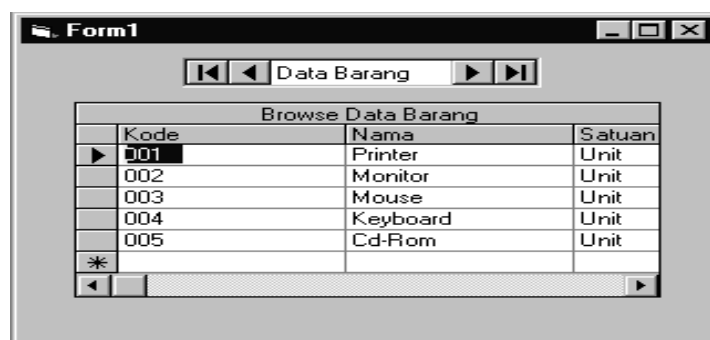
7. Atur letak dan ukuran seperti berikut :



8. Simpan lagi proyek dan form dengan nama sama.

9. Jalankan program.

10. Hasilnya sbb :



a. Menambahkan Record

- Klik icon Last [] pada objek Data, lalu tekan tombol panah ke bawah.
- Ketik data yang dikehendaki.

b. Mengedit Record

- Klik pada data yang ingin di edit sampai blok hilang.
- Ketik data yang dikehendaki

c. Menghapus Record

- Klik disebelah kiri record sehingga tersorot satu baris record yang akan dihapus.
- Tekan tombol **Del** / **Delete** pada keyboard.

BAB XVII

MENGGUNAKAN DATABASE LEWAT KODE PROGRAM

Untuk membuka database lewat kode program, ikuti langkah-langkah sebagai berikut:

17.1. Deklarasi Variabel Database

Level deklarasi variabel database bisa pada level form atau level lain, tetapi biasanya tidak pada level prosedur. Penulisannya adalah sebagai berikut :

Dim [nama_variabel] As Database

Contoh : Dim dbPembelian As Database

Jika perintah diatas tidak dikenal, maka dapat menambahkan dulu objek data dari toolbox ke dalam form. Objek data tersebut tidak perlu dikaitkan ke dalam database, karena hanya sebagai syarat agar dapat menggunakan perintah deklarasi variabel database di atas. Setelah program dijalankan satu kali, maka objek data tersebut dapat dihapus.

17.2. Deklarasi Variabel Recordset

Level deklarasi recordset ini bisa pada level form atau level lain, tetapi biasanya tidak pada level prosedur. penulisannya adalah sebagai berikut :

Dim [nama_variabel] As Recordset

Contoh : Dim rsBarang As Recordset

17.3. Open Database

Setelah mendeklarasikan variabel database dan variabel recordset, berilah untuk membuka database pada suatu prosedur, misalnya prosedur Form_Load. Penulisannya adalah sebagai berikut :

Set [nama_variabel] = OpenDatabase (“[namaFileDatabase]”)

Contoh :

Set dbPembelian = OpenDatabase (“C:\VB6\Pembelian.mdb”)

Untuk penulisan database harus disertakan nama direktori atau folder atau path secara lengkap. Penulisan perintah ini sama dengan menentukan properti DatabaseName pada objek Data.

17.4. Open Recordset

Tentukan recordset dari database yang telah dibuka. Penulisannya adalah sebagai berikut :

Set [nmVar] = [nmVarDatabase].OpenRecordset (“[nmTabel]”)

Contoh

Set rsBarang = dbPembelian.OpenRecordset (“Barang”)

Perintah ini sama dengan menentukan properti RecordSource pada objek Data.

17.5. Close

Setelah databse dan recordset diakses, tutuplah dengan perintah sebagai berikut :

[namaVarRecordset].Close

[namaVarDatabase].Close

Contoh :

rsBarang.Close

dbPembelian.Close

Jika menutup suatu database, maka seluruh recordset yang mengacu pada database tersebut otomatis tertutup. Jadi perintah di atas boleh ditulis dengan satu baris saja, yaitu dbPembelian.Close.

Contoh penulisan selengkapnya sebagai berikut :

```
Dim dbPembelian As Database  
Dim rsBarang As Recordset
```

```
Private Sub cmdSelesai_Click()  
    rsBarang.Close  
    dbPembelian.Close  
End  
End Sub
```

```
Private Sub Form_Load()  
    Set dbPembelian = OpenDatabase ("C:\VB6\Pembelian.mdb")  
    Set rsBarang = dbPembelian.OpenRecordset ("Barang")  
End Sub
```

BAB XVIII

MENGENAL SQL

18.1. Pengetian SQL

SQL adalah singkatan dari Structured Query Language. SQL adalah bahasa yang digunakan untuk memuat, mengurutkan, menyaring, meng-update, dan mengolah relasi antardatabase suatu data sehingga dihasilkan suatu data yang spesifik dari suatu database.

18.2. Menjalankan SQL Data Manager

Berikut ini akan dibahas penerapan SQL dalam property **RecordSource** pada suatu objek data.

Bentuk Umum :

<objekdata>.RecordSource=<stringSQL>

Keterangan :

<stringSQL> adalah string yang berisi pernyataan dalam SQL.

Contoh :

Perintah dibawah ini akan mengambil data yang berupa field **KodeBrg** dan field **NoBon** dari table data **Beli**.

Data1.RecordSource = "SELECT KodeBrg, NoBon FROM Beli"

18.3. Select ... From

Digunakan untuk memilih field dan table data dengan SQL .

Bentuk Umum :

SELECT <daftarField> FROM <namaTabel> IN <namadatabase>

Keterangan :

<daftarField> adalah nama-nama field yang diinginkan, bisa satu atau lebih field (* = untuk semua field).

<namaTabel> adalah salah satu nama table yang ada dalam nama database di sebelah kanan pernyataan **IN**.

<namaDatabase> adalah string yang berisi nama database.

Untuk memahami Select ... From lakukan langkah-langkah sebagai berikut :

1. Pilih menu **File** lalu **New Project**.
2. Pilih **VB Enterprise Edition Controls**, lalu **Ok**.
3. Tambahkan satu buah objek **Data** dan ubahlah property **DatabaseName**-nya menjadi **C:\Zidan\Penjualan.mdb** atau nama database lainnya. Properti **RecorSourceType**-nya tidak boleh berupa **0 – Table** (agar dapat menggunakan index dalam operasi databasenya).
4. Tambahkan sebuah objek **DBGrid** dan ubahlah property **DataSource**-nya menjadi **Data1**.
5. Tambahkan objek **Label**, **TextBox**, dan **CommandButton**. Atur letak dan ukuran seperti berikut :

The screenshot shows a Visual Basic form titled 'Form1'. At the top, there is a 'Data1' control with navigation buttons. Below it is a table with two columns: 'KodeBrg' and 'NoBon'. The table contains 10 rows of data. Below the table is a text box labeled 'Bentuk SQL' containing the following SQL query:
 SELECT KodeBrg.NoBon FROM Jual IN 'C:\Lat\VB6\Penjualan.mdb'
 At the bottom of the form, there are three buttons: 'Data Barang', 'Data Penjualan', and 'Selesai'.

KodeBrg	NoBon
BT.001	F001
BT.001	B006
BT.001	B008
BT.001	B009
BT.002	B008
BT.002	B009
BT.003	B005
BT.003	B006
BT.003	B008

Bentuk SQL

```
SELECT KodeBrg.NoBon FROM Jual IN 'C:\Lat\VB6\Penjualan.mdb'
```

Data Barang Data Penjualan Selesai

6. Tampilkan editor kode program dan ketikkan kode programnya sbb:

```

Private Sub Command1_Click()
    'menampilkan seluruh field dan seluruh record tabel data Beli
    Data1.RecordSource = "SELECT * FROM Beli IN
    'C:\LatVB6\Pembelian.mdb' "
    'obyek data mengambil data sesuai SQL
    Data1.Refresh
    Text1.Text = Data1.RecordSource
End Sub

Private Sub Command2_Click()
    Data1.RecordSource = "SELECT * FROM Jual IN
    'C:\LatVB6\Penjualan.mdb' "
    Data1.Refresh
    Text1.Text = Data1.RecordSource
End Sub

Private Sub Command3_Click()
    End
End Sub

Private Sub Form_Load()
    'pada saat program dijalankan, program menampilkan
    'seluruh record data Jual dengan kolom KodeBrg dan NoBon saja.
    Data1.RecordSource = "SELECT KodeBrg,NoBon FROM Jual IN
    'C:\LatVB6\Penjualan.mdb' "
    Data1.Refresh
    Text1.Text = Data1.RecordSource
End Sub

```

7. Simpan lagi proyek dan form lalu jalankan program.

18.4. Select ... From ... Where

Digunakan untuk memilih field dan table data hanya untuk record tertentu, sehingga SQL dapat bekerja sebagai penyaring / filter data.

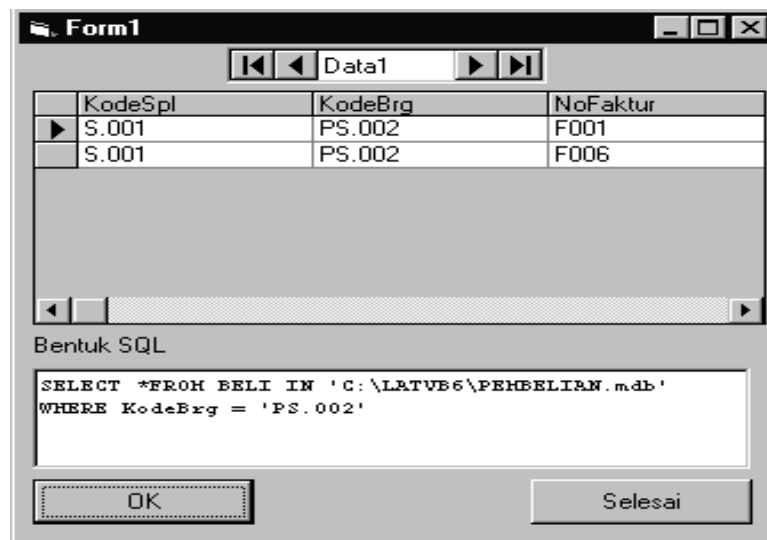
Bentuk Umum :

SELECT <daftarField> **FROM** <namaTabel> **IN** <namadatabase> **WHERE**
<syarat>

Contoh : **SELECT * FROM BELI IN 'C:\LATVB6\PEMBELIAN.mdb' WHERE**
KodeBrg = 'PS.002'

Untuk memahami Select ... From ... Where lakukan langkah-langkah sebagai berikut:

1. Pilih menu **File** lalu **New Project**.
2. Pilih **VB Enterprise Edition Controls**, lalu **Ok**.
3. Tambahkan satu buah objek **Data** dan ubahlah property **DatabaseName**-nya menjadi **C:\Zidan\Penjualan.mdb** atau nama database lainnya. Properti **RecorSourceType**-nya tidak boleh berupa **0 – Table** (agar dapat menggunakan index dalam operasi databasenya).
4. Tambahkan sebuah objek **DBGrid** dan ubahlah property **DataSource**-nya menjadi **Data1**.
5. Tambahkan objek **Label**, **TextBox**, dan **CommandButton**. Atur letak dan ukuran seperti berikut :



6. Tampilkan editor kode program dan ketikkan kode programnya sbb:

```
Private Sub Command1_Click()
    Dim X As Integer
    On Error GoTo Salah
    'ambil isi Text1 sebagai <stringSQL>
    Data1.RecordSource = Text1.Text
    Data1.Refresh
    On Error GoTo 0
    Exit Sub
Salah:
```

```

Salah:
    Beep
    X = MsgBox("Tulisan SQL salah!", vbOKOnly)
    Text1.SetFocus
End Sub

```

```

Private Sub Command2_Click()
    End
End Sub

```

```

Private Sub Form_Activate()
    Text1.SetFocus
    Command1.Default = True
End Sub

```

7. Simpan lagi proyek dan form dengan nama **Lat30** lalu jalankan program.
8. Isilah isian bentuk SQL dengan <stringSQL> yang diinginkan, misal **SELECT * FROM BELI IN 'C:\LATVB6\PEMBELIAN.mdb' WHERE KodeBrg = 'PS.002'**

18.5. Select ... From ... Order By

SQL dapat digunakan untuk mengakses suatu data dengan hasil yang teratur. Kunci pengurutannya bahkan bisa beberapa field sekaligus.

Bentuk Umum :

```

SELECT <daftarField> FROM <namaTabel> IN <namadatabase> ORDER BY
<fieldKunci1>, <fieldKunci2>,

```

Contoh :

```

SELECT * FROM Jual IN 'C:\LatVB6\Penjualan.mdb' ORDER BY KodeCst,
NoBon

```

Anda dapat menggunakan program **Lat30** untuk mencoba contoh tersebut.

Form1

Navigation: [Previous] [Data1] [Next]

	KodeCst	KodeBrg	NoBon
▶	C.001	BT.003	B006
	C.001	PS.002	B006
	C.001	BT.001	B006
	C.001	BT.002	B009
	C.001	BT.001	B009
	C.001	BT.003	B009
	C.001	PS.002	B009

Bentuk SQL

```
SELECT * FROM Jual IN 'C:\LatVB6\Penjualan.mdb'  
ORDER BY KodeCst, NoBon
```

OK Selesai

BAB XIX

PEMBUATAN LAPORAN DENGAN DATA REPORT

Data Report adalah suatu laporan mengenai database yang disediakan oleh Visual Basic. Laporan ini dapat terbentuk jika Data Environment sudah didesain. **Data Environment** merupakan penghubung antara database yang telah didesain dengan Data Report yang akan ditampilkan.

Untuk membuat Report dengan Data Report di Visual Basic, lakukan langkah-langkah berikut ini :

1) Buat Database baru.

Caranya :

- a) Buat Projek Baru : File, New, Projek, Pilih **Data Project**, Ok
- b) Dari menu bar **Add-Ins** pilih **Visual Data Manager**.
- c) Klik File, New, Microsoft Access, Version 7.0 MDB
- d) Tentukan direktori penyimpanan data dan ketikkan nama filenya. Misal :

Tesdatabase



- e) Pada **Database Windows**, klik kanan di **Properties**.
- f) Pilih **New Table** sehingga jendela **Table Structure** terbuka.

- g) Ketikkan nama table (misal : **Data Teman**)
- h) Klik **Add Field** sehingga jendela Add Field terbuka.

- i) Isikan nama field, tipe dan ukuran field. Klik **Ok** untuk menambah filed lagi. Isi seperti berikut :

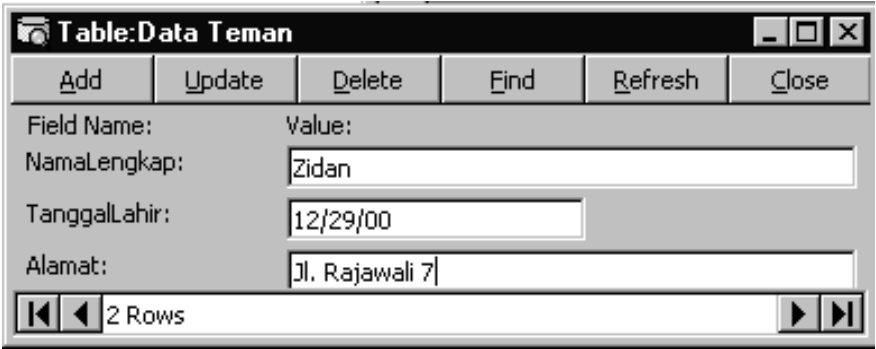
Name	Type	Size
NamaLengkap	Text	30
TanggalLahir	Date/Time	8
Alamat	Text	50

- j) Setelah membuat filed-field tersebut, klik **Close**.
- k) Klik **Build The Table** pada jendela Table Structure. Nama table yang telah dibuat akan ditampilkan di bawah Properties pada jendela kerja Visual Data Manager.



- l) Klik kanan pada Data Teman.
- m) Pilih **Open**. Ketikkan data-datanya.
- n) Klik **Update** untuk menyimpan data.

- o) Klik **Add** untuk menambah data.



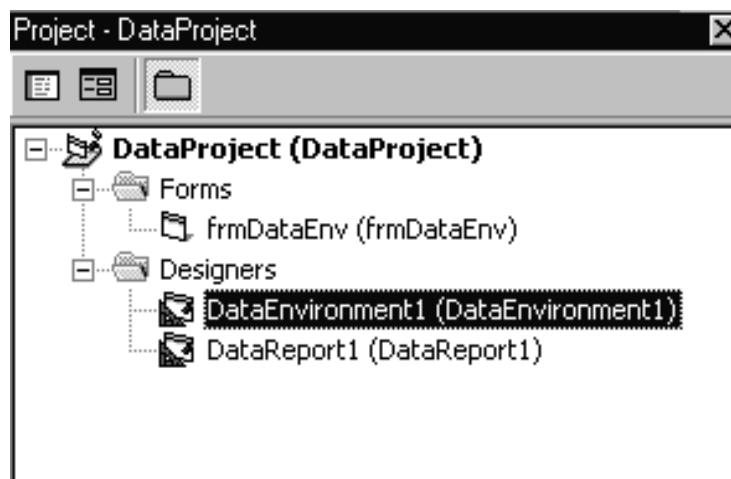
The screenshot shows a window titled "Table: Data Teman" with a toolbar containing buttons for Add, Update, Delete, Find, Refresh, and Close. Below the toolbar, there are four input fields labeled "Field Name:" and "Value:". The first field is "NamaLengkap:" with the value "Zidan". The second field is "TanggalLahir:" with the value "12/29/00". The third field is "Alamat:" with the value "Jl. Rajawali 7". At the bottom, there is a status bar showing "2 Rows" and navigation arrows.

- p) Klik **Close** untuk keluar dari jendela table.
- q) Klik **File, Exit** untuk kembali ke jendela desain Visual Basic

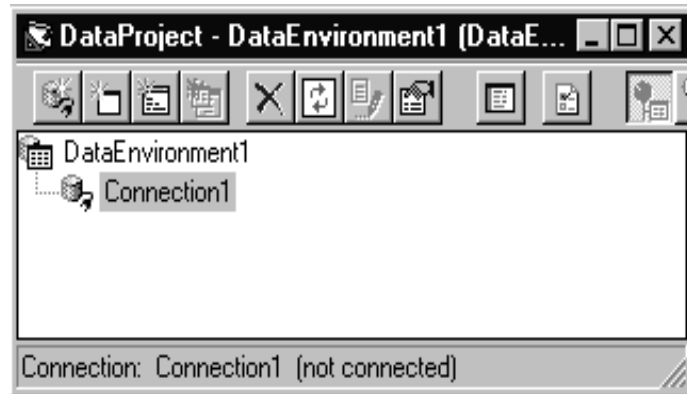
3) Buat Data Environment.

Caranya :

- a) Pada window Project, klik double **DataEnvironment1**



- b) Klik pada **Connection1**. Kemudian dari jendela **Properties**, ganti property **namenya** menjadi **Dataku**

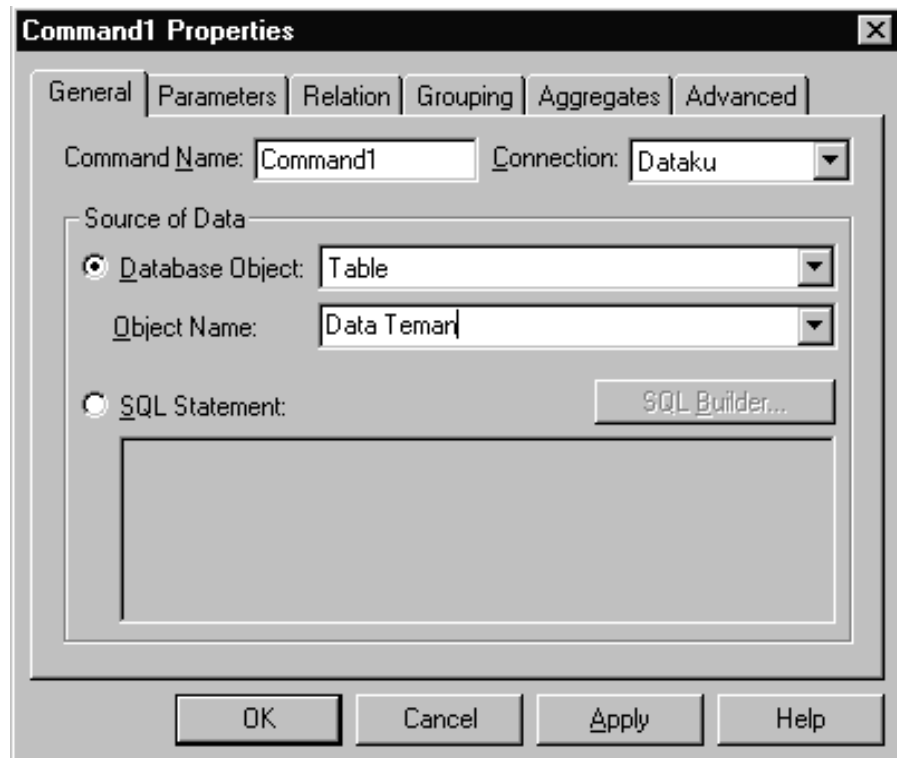


- c) Untuk menghubungkan dengan database, klik kanan pada Dataku, pilih **Properties** hingga tampil jendela **Data Link Properties**.
- d) Pilih **Microsoft Jet 3.51 OLE DB PROVIDER**, klik **Next**.
- e) Browse Nama Database yang akan dihubungkan.. (**Tesdatabase.mdb**)



- f) Klik **Test Connection**. Jika koneksi berhasil, berarti database telah terkoneksi. Klik **Ok**.
- g) Untuk menambahkan command pada database, klik kanan **Dataku**, pilih **Add Command**. Akan muncul Command1 di bawah Dataku.

- h) Klik kanan Command1, pilih **Properties**.
- i) Perhatikan bagian **Connection**, pilih **Dataku**.
- j) Pada bagian **Database Objek**, ubah **Stored Procedure** menjadi **Table**.
- k) Pada bagian objek **name**, pilih nama table yang telah didesain (**Data Teman**).



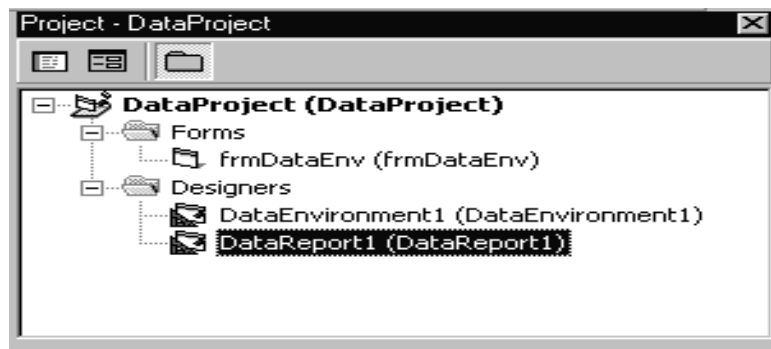
- l) Klik **Ok**.
- m) Klik tanda (+) di depan **Dataku**. Field-field database akan ditampilkan.



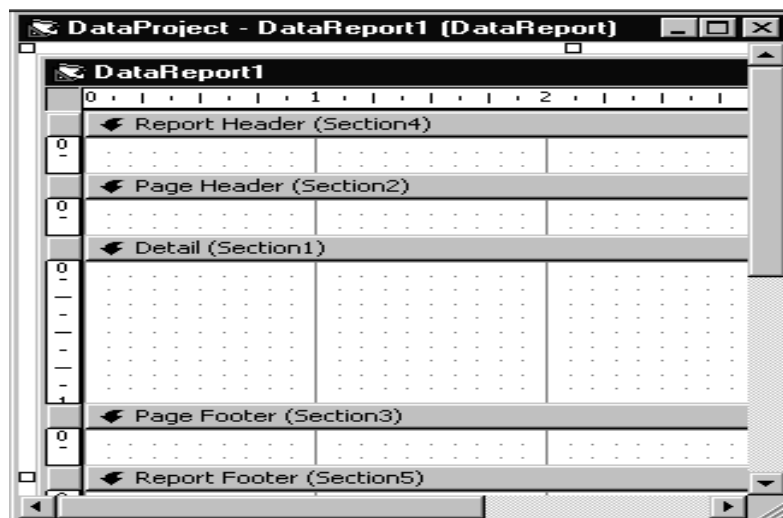
4) Buat Data Report.

Caranya :

- a) Pada window Project, klik double **DataReport1**



- b) Klik kanan pada jendela Data Report. Klik **Show Report Header/Footer**.



Bagian-bagiannya :

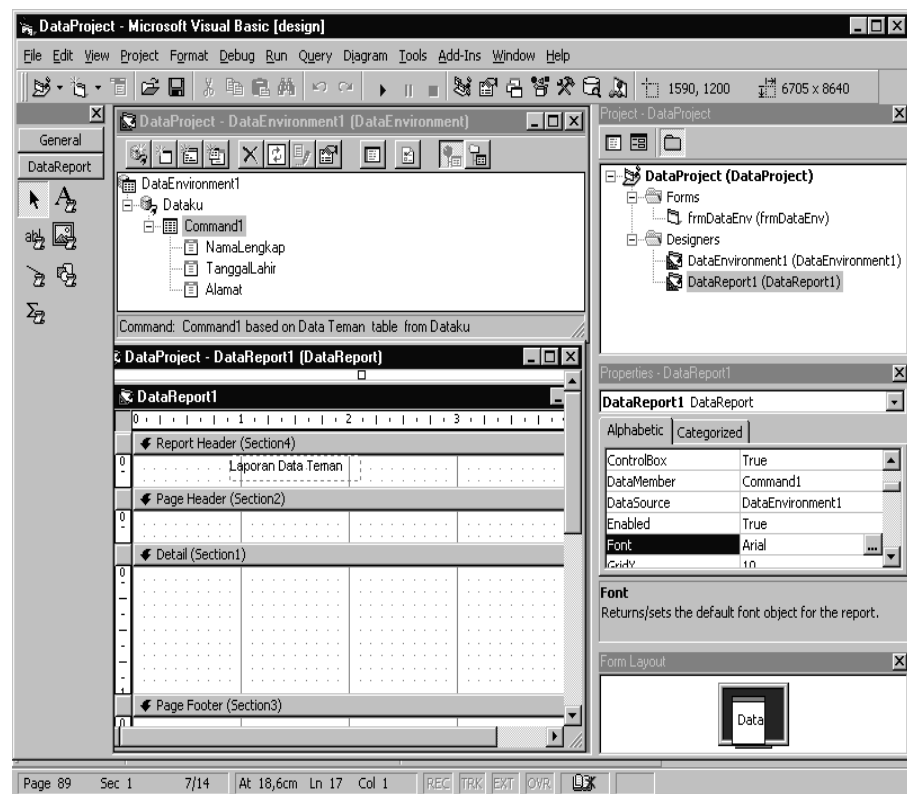
Bagian	Keterangan
Report Header	Berisi judul laporan.
Page Header	Berisi judul kolom data yang akan ditampilkan.
Detail	Berisi nama field yang akan ditampilkan pada kolom.
Page Footer	Berisi catatan kaki di akhir kolom-kolom report.
Report Footer	Berisi catatan kecil/keterangan report

Selain jendela Data Report, pada toolbox akan muncul sebuah tab Data Report yang berisi :

Nama Kontrol	Keterangan
RtpTextBox	Kontrol serupa textbox yang hanya menampilkan teks database saat runtime.
RptLine	Kontrol yang dapat menggambar garis secara horizontal, vertical, maupun diagonal.
RptFunction	Kontrol yang dapat menset hasil perhitungan pada

	saat runtime (misalnya jumlah toatal, standar deviasi, dsb)
RptLabel	Kontrol untuk memberikan label pada report.
RptImage	Kontrol untuk menempatkan image pada report
RptShape	Kontrol untuk menggambar shape pada report

- c) Letakkan kontrol **RptLabel** pada bagian **Report Header**. Ubah property **Caption** menjadi 'Laporan Data Teman', dan property **Alignment** menjadi rptjustcenter.
- d) Klik **Data Report1**.
- e) Untuk menghubungkan data report ke database yang telah dibuat, isi property **Datasource** dengan nama data environment yang telah dibuat. (**DataEnvironment1**)
- f) Isi property **Datamember** dengan nama command yang telah dibuat. (**Command1**)
- g) Klik menu bar **Windows**, pilih **Cascade**.
- h) Letakkan jendela **Data Environment** di atas jendela **Data Report**.



- i) **Drag & drop** field-filed pada jendela data environment ke jendela Data Report di bagian **Detail**
- j) Pindahkan bagian yang diakhiri tanda “:” ke bagian **page header**.

The screenshot shows the design view of a report titled 'DataReport1'. It features a grid with columns numbered 0 to 4. The report is divided into several sections:

- Report Header (Section4):** Contains the text 'Laporan Data Teman'.
- Page Header (Section2):** Contains labels 'NamaLengkap:', 'TanggalLahir:', and 'Alamat:'.
- Detail (Section1):** Contains data fields for 'NamaLengkap', 'TanggalLahir', and 'Alamat'.
- Page Footer (Section3):** Empty.
- Report Footer (Section5):** Empty.

- k) Desain akhir Data Report adalah sbb :

The screenshot shows the final preview of the report 'DataReport1'. The report is titled 'Laporan Data Teman' and displays the following data:

NamaLengkap:	TanggalLahir:	Alamat:
Zidan	12/29/00	Jl. Rajawali 7
Hanif	5/20/02	Jl. Jeruk 1

The report is displayed on a single page, as indicated by the 'Pages: 1' indicator at the bottom.

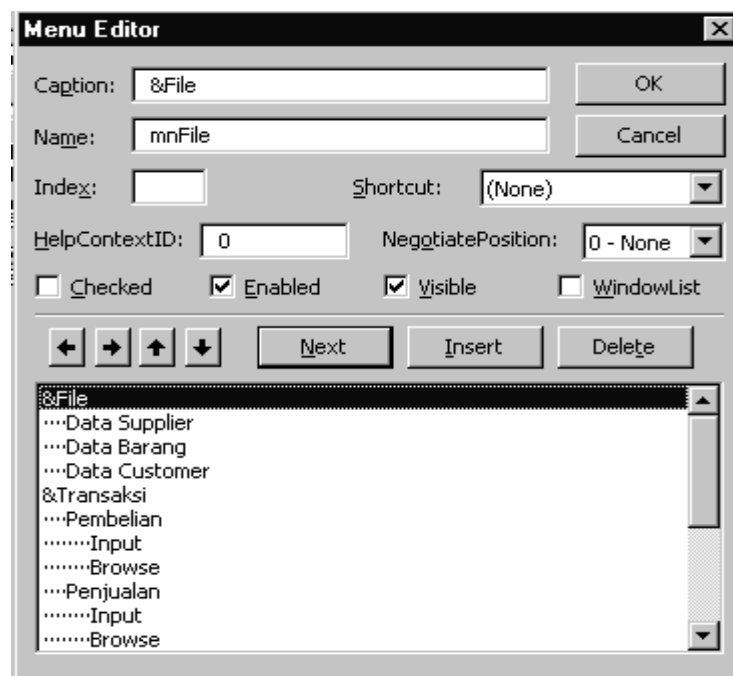
BAB XX

MENU

Menu utama / menu program digunakan untuk mengintegrasikan program-program sehingga pemakai program tidak kesulitan menggunakan program yang telah dibuat. Ms. Visual Basic 6.0 menyediakan fasilitas untuk membuat menu program, yaitu Menu Editor. Dengan fasilitas ini, anda dapat membuat menu program dengan cepat dan mudah. Berikut ini langkah-langkah pembuatan menu :

1. Buat project baru dan tampilkan form
2. Klik Form lalu ubahlah property **Name**-nya menjadi **frmMenu** dan property **Caption**-nya menjadi **Menu Program Pembelian & Penjualan**.

3. Klik Icon **Menu Editor** 



4. Isilah **Caption** dengan tampilan menu (Contoh : **&File**)
5. Isilah **Name** dengan nama pengenal menu sebagai procedure atau objek untuk kode program (Contoh : **mnFile**)

6. Beri tanda pilih pada pilihan **Enable** agar menu ini bisa dipilih.
7. Beri tanda pilih pada pilihan **Visible** agar menu ini ditampilkan dilayar.
8. Gunakan icon tanda panah untuk mengubah letak / level menu.
9. Klik **Next** untuk membuat pilihan / menu berikutnya
10. Pilih **Insert** untuk menyisipkan pilihan baru pada posisi yang sekarang ditunjuk.
11. Pilih **Delete** untuk menghapus menu yang ditunjuk.
12. Lengkapilah struktur menu menjadi seperti table berikut ini :

Caption			Name
Level 1	Level 2	Level 3	
&File			mnFile
	Data Supplier		mnSupplier
	Data Barang		mnBarang
	Data Customer		mnCustomer
&Transaksi			mnTransaksi
	Pembelian		mnBeli
		Input	mnInputBeli
		Browse	mnBrowseBeli
	Penjualan		mnJual
		Input	mnInputJual
		Browse	mnBrowseJual
&Cetak			mnCetak
	Supplier		mnCetakSupplier
	Barang		mnCetakBarang
	Customer		mnCetakCustomer
	Pembelian		mnCetakPembelian
	Penjualan		mnCetakPenjualan
&Selesai			mnSelesai
	Tutup Program		mnTutup

13. Setelah selesai klik **OK** dan simpanlah proyek dan formya dengan nama **frmMenu**.
14. Jalankan aplikasi tersebut dengan menekan **F5**
15. Setelah Form Menu Utama terbentuk langkah berikutnya adalah membuat program untuk sub menu.

a) Program Data Supplier

- Klik sub menu Data Supplier pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmSupplier. Ketikkan kode programnya seperti berikut ini :

```
Private Sub mnSupplier_Click()  
    frmSupplier.Show  
End Sub
```

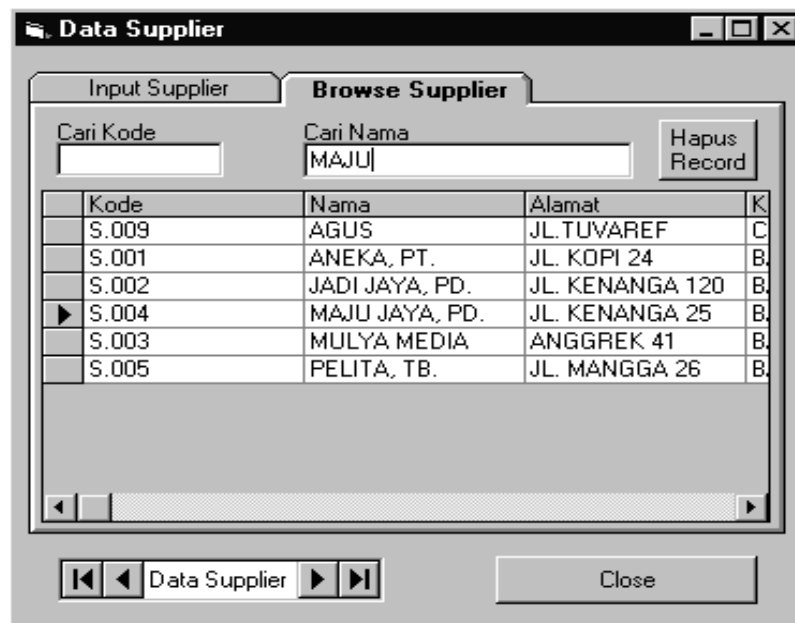
- Klik menu Project, Add Form, Form, Open.
- Buat form seperti berikut :

- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Name	frmSupplier
	Caption	Data Supplier
SSTab	Tabs	2
	Caption [Tab 0]	Input Supplier
	Caption [Tab 1]	Browse Supplier
Command1	Name	CmdClose
	Caption	Close
Data	Caption	Data Supplier
	DatabaseName	C:\Pembelian.mdb
	RecordsetType	0-Table

	RecordSource	Supplier
Text1	Name	txtKode
Text2	Name	txtNama
Text3	Name	txtAlamat
Text4	Name	txtKota
Text5	Name	txtHubungan
Text6	Name	txtTelepon
Command2	Name	cmdSimpan
Command3	Name	cmdBatal

- Klik tab Browse Supplier kemudian tambahkan objek-objeknya seperti berikut :



- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Text1	Name	txtCariKode
Text2	Name	txtCariNama
Command1	Name	cmdHapusRecord
DBGrid	DataSource	Data1

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```
Private Sub cmdBatal_Click()
    txtKode.Text = ""
```

```

BlankForm
TidakSiapIsti
txtKode.SetFocus
End Sub

```

```

Private Sub cmdClose_Click()
    frmSupplier.Hide
    frmMenu.Show
End Sub

```

```

Private Sub cmdHapusRecord_Click()
    Dim X As Byte
    X = MsgBox("Betul akan dihapus?", vbOKCancel, "Menghapus record")
    If X = vbOK Then
        Data1.Recordset.Delete
    End If
End Sub

```

```

Private Sub cmdSimpan_Click()
    Data1.Recordset.AddNew
    Data1.Recordset!Kode = txtKode.Text
    Data1.Recordset!Nama = txtNama.Text
    Data1.Recordset!Alamat = txtAlamat.Text
    Data1.Recordset!Kota = txtKota.Text
    Data1.Recordset!Hubungan = txtHubungan.Text
    Data1.Recordset!Telepon = txtTelepon.Text
    Data1.Recordset.Update
    cmdBatal_Click
End Sub

```

```

Private Sub Form_Activate()
    txtKode.SetFocus
    txtKode.MaxLength = 5
    txtNama.MaxLength = 30
    txtAlamat.MaxLength = 30
    txtKota.MaxLength = 10
    txtHubungan.MaxLength = 30
    txtTelepon.MaxLength = 20
    TidakSiapIsti
End Sub

```

```

Private Sub SSTab1_GotFocus()
    If SSTab1.Caption = "Input Supplier" Then
        txtKode.SetFocus
    Else
        txtCariKode.SetFocus
    End If
End Sub

```

```

Private Sub txtCariKode_Change()
    Data1.Recordset.Index = "KodeSpl"
    Data1.Recordset.Seek "<=", txtCariKode.Text
    If Data1.Recordset.NoMatch Then
        If Len(Trim(txtCariKode.Text)) = 5 Then
            Beep
        End If
    End If
End Sub

Private Sub txtCariNama_Change()
    Data1.Recordset.Index = "NamaSpl"
    Data1.Recordset.Seek "<=", Trim(txtCariNama.Text) & "zzz"
    If Data1.Recordset.NoMatch Then
        If Len(Trim(txtCariKode.Text)) = 30 Then
            Beep
        End If
    End If
End Sub

Private Sub txtKode_Change()
    If Len(Trim(txtKode.Text)) < 5 Then
        Exit Sub
    End If
    Data1.Recordset.Index = "KodeSpl"
    Data1.Recordset.Seek "=", txtKode.Text
    If Not Data1.Recordset.NoMatch Then
        On Error Resume Next 'jika field kosong, abaikan!
        txtNama.Text = Data1.Recordset!Nama
        txtAlamat.Text = Data1.Recordset!Alamat
        txtKota.Text = Data1.Recordset!Kota
        txtHubungan.Text = Data1.Recordset!Hubungan
        txtTelepon.Text = Data1.Recordset!Telepon
        TidakSiapIsi
        On Error GoTo 0
        Exit Sub
    End If
    SiapIsi
    BlankForm
    txtNama.SetFocus
End Sub

Private Sub txtKode_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

```

```

Private Sub txtAlamat_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtKota_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtHubungan_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Function TidakSiapIpsi()
    txtNama.Enabled = False
    txtAlamat.Enabled = False
    txtKota.Enabled = False
    txtHubungan.Enabled = False
    txtTelepon.Enabled = False
    txtNama.BackColor = &H8000000F
    txtAlamat.BackColor = &H8000000F
    txtKota.BackColor = &H8000000F
    txtHubungan.BackColor = &H8000000F
    txtTelepon.BackColor = &H8000000F
    cmdSimpan.Enabled = False
End Function

Private Function SiapIpsi()
    txtNama.Enabled = True
    txtAlamat.Enabled = True
    txtKota.Enabled = True
    txtHubungan.Enabled = True
    txtTelepon.Enabled = True
    txtNama.BackColor = &H8000000E
    txtAlamat.BackColor = &H8000000E
    txtKota.BackColor = &H8000000E
    txtHubungan.BackColor = &H8000000E
    txtTelepon.BackColor = &H8000000E
    cmdSimpan.Enabled = True
End Function

Private Function BlankForm()
    txtNama.Text = ""
    txtAlamat.Text = ""
    txtKota.Text = ""
    txtHubungan.Text = ""
    txtTelepon.Text = ""
End Function

```

- Simpan project dan form lalu jalankan.

b) Program Data Barang

- Klik sub menu Data Barang pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmSupplier. Ketikkan kode programnya seperti berikut ini :

```
Private Sub mnBarang_Click()
    frmBarang.Show
End Sub
```

- Klik menu Project, Add Form, Form, Open.
- Buat form seperti berikut :

- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	frmBarang Data Barang
SSTab	Tabs	2
	Caption [Tab 0]	Input Barang
	Caption [Tab 1]	Browse Barang
Command1	Name	CmdClose
	Caption	Close

Data	Caption	Data Barang
	DatabaseName	C:\Pembelian.mdb
	RecordsetType	0-Table
	RecordSource	Barang
Text1	Name	txtKode
Text2	Name	txtNama
Text3	Name	txtSatuan
Text4	Name	txtHarga
Command2	Name	cmdSimpan
Command3	Name	cmdBatal

- Klik tab Browse Supplier kemudian tambahkan objek-objeknya seperti berikut :



- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Text1	Name	txtCariKode
Text2	Name	txtCariNama
Command1	Name	cmdHapusRecord
DBGrid	DataSource	Data1

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```

Private Sub cmdBatal_Click()
    txtKode.Text = ""
    BlankForm
    TidakSiapIsi
    txtKode.SetFocus
End Sub

Private Sub cmdClose_Click()
    frmBarang.Hide
    frmMenu.Show
End Sub

Private Sub cmdHapusRecord_Click()
    Dim X As Byte
    X = MsgBox("Betul akan dihapus?", vbOKCancel, "Menghapus record")
    If X = vbOK Then
        Data1.Recordset.Delete
    End If
End Sub

Private Sub cmdSimpan_Click()
    On Error Resume Next
    Data1.Recordset.AddNew
    Data1.Recordset!Kode = txtKode.Text
    Data1.Recordset!Nama = txtNama.Text
    Data1.Recordset!Satuan = txtSatuan.Text
    Data1.Recordset!Harga = txtHarga.Text
    Data1.Recordset.Update
    On Error GoTo 0
    cmdBatal_Click
End Sub

Private Sub Form_Activate()
    txtKode.SetFocus
    txtKode.MaxLength = 6
    txtNama.MaxLength = 30
    txtSatuan.MaxLength = 5
    txtHarga.MaxLength = 12
    TidakSiapIsi
End Sub

Private Sub SSTab1_GotFocus()
    If SSTab1.Caption = "Input Barang" Then
        txtKode.SetFocus
    Else
        txtCariKode.SetFocus
    End If

```

End Sub

```
Private Sub txtCariKode_Change()
    Data1.Recordset.Index = "KodeBrg"
    Data1.Recordset.Seek "<=", txtCariKode.Text
    If Data1.Recordset.NoMatch Then
        If Len(Trim(txtCariKode.Text)) = 6 Then
            Beep
        End If
    End If
End Sub
```

```
Private Sub txtCariNama_Change()
    Data1.Recordset.Index = "NamaBrg"
    Data1.Recordset.Seek "<=", Trim(txtCariNama.Text) & "zzz"
    If Data1.Recordset.NoMatch Then
        If Len(Trim(txtCariKode.Text)) = 30 Then
            Beep
        End If
    End If
End Sub
```

```
Private Sub txtKode_Change()
    If Len(Trim(txtKode.Text)) < 6 Then
        Exit Sub
    End If
    Data1.Recordset.Index = "KodeBrg"
    Data1.Recordset.Seek "=", txtKode.Text
    If Not Data1.Recordset.NoMatch Then
        On Error Resume Next 'jika field kosong, abaikan!
        txtNama.Text = Data1.Recordset!Nama
        txtSatuan.Text = Data1.Recordset!Satuan
        txtHarga.Text = Data1.Recordset!Harga
        TidakSiapIsi
        On Error GoTo 0
        Exit Sub
    End If
    SiapIsi
    BlankForm
    txtNama.SetFocus
End Sub
```

```
Private Sub txtKode_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCCase(Chr(KeyAscii)))
End Sub
```

```
Private Sub txtNama_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCCase(Chr(KeyAscii)))
End Sub
```

```
Private Sub txtSatuan_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub
```

```
Private Function TidakSiapIpsi()
    txtNama.Enabled = False
    txtSatuan.Enabled = False
    txtHarga.Enabled = False
    txtNama.BackColor = &H8000000F
    txtSatuan.BackColor = &H8000000F
    txtHarga.BackColor = &H8000000F
    cmdSimpan.Enabled = False
End Function
```

```
Private Function SiapIpsi()
    txtNama.Enabled = True
    txtSatuan.Enabled = True
    txtHarga.Enabled = True
    txtNama.BackColor = &H8000000E
    txtSatuan.BackColor = &H8000000E
    txtHarga.BackColor = &H8000000E
    cmdSimpan.Enabled = True
End Function
```

```
Private Function BlankForm()
    txtNama.Text = ""
    txtSatuan.Text = ""
    txtHarga.Text = ""
End Function
```

```
Private Sub txtHarga_KeyPress(KeyAscii As Integer)
    'hanya boleh diisi angka atau backspace
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack) Then
        Beep
        KeyAscii = 0
    End If
End Sub
```

- i. Simpan project dan form lalu jalankan.

c) Program Data Customer

- Klik sub menu Data Barang pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmSupplier. Ketikkan kode programnya seperti berikut ini :

```
Private Sub mnCustomer_Click()
```

frmCustomer.Show
End Sub

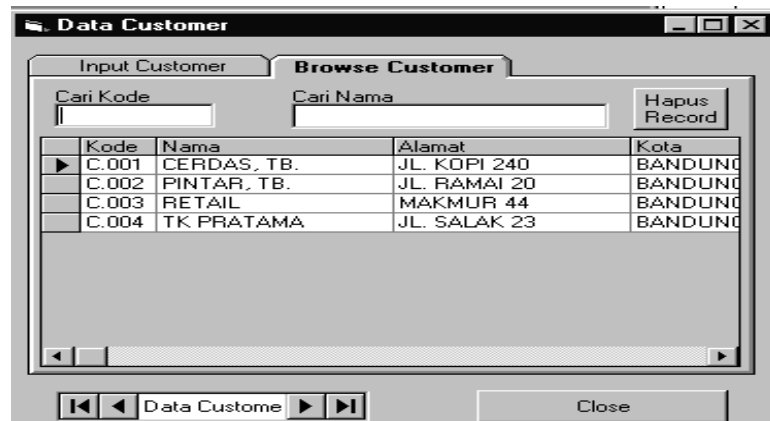
- Klik menu Project, Add Form, Form, Open.
- Buat form seperti berikut :

- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	frmCustomer Data Customer
SSTab	Tabs	2
	Caption [Tab 0]	Input Customer
	Caption [Tab 1]	Browse Customer
Command1	Name	CmdClose
	Caption	Close
Data	Caption	Data Customer
	DatabaseName	C:\Pembelian.mdb
	RecordsetType	0-Table
	RecordSource	Customer
Text1	Name	txtKode
Text2	Name	txtNama
Text3	Name	txtAlamat
Text4	Name	txtKota

Text5	Name	txtHubungan
Text6	Name	txtTelepon
Command2	Name	CmdSimpan
Command3	Name	CmdBatal

- Klik tab Browse Customer kemudian tambahkan objek-objeknya seperti berikut



- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Text1	Name	txtCariKode
Text2	Name	TxtCariNama
Command1	Name	cmdHapusRecord
DBGrid	DataSource	Data1

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```

Private Sub cmdBatal_Click()
    txtKode.Text = ""
    BlankForm
    TidakSiapsi
    txtKode.SetFocus
End Sub
Private Sub cmdClose_Click()
    frmCustomer.Hide
    frmMenu.Show
End Sub

Private Sub cmdHapusRecord_Click()
    Dim X As Byte
    X = MsgBox("Betul akan dihapus?", vbOKCancel, "Menghapus record")

```

```

    If X = vbOK Then
        Data1.Recordset.Delete
    End If
End Sub

Private Sub cmdSimpan_Click()
    Data1.Recordset.AddNew
    Data1.Recordset!Kode = txtKode.Text
    Data1.Recordset!Nama = txtNama.Text
    Data1.Recordset!Alamat = txtAlamat.Text
    Data1.Recordset!Kota = txtKota.Text
    Data1.Recordset!Hubungan = txtHubungan.Text
    Data1.Recordset!Telepon = txtTelepon.Text
    Data1.Recordset.Update
    cmdBatal_Click
End Sub

Private Sub Form_Activate()
    txtKode.SetFocus
    txtKode.MaxLength = 5
    txtNama.MaxLength = 30
    txtAlamat.MaxLength = 30
    txtKota.MaxLength = 10
    txtHubungan.MaxLength = 30
    txtTelepon.MaxLength = 20
    TidakSiaplsi
End Sub

Private Sub SSTab1_GotFocus()
    If SSTab1.Caption = "Input Customer" Then
        txtKode.SetFocus
    Else
        txtCariKode.SetFocus
    End If
End Sub

Private Sub txtCariKode_Change()
    Data1.Recordset.Index = "KodeCst"
    Data1.Recordset.Seek "<=", txtCariKode.Text
    If Data1.Recordset.NoMatch Then
        If Len(Trim(txtCariKode.Text)) = 5 Then
            Beep
        End If
    End If
End Sub

Private Sub txtCariNama_Change()
    Data1.Recordset.Index = "NamaCst"
    Data1.Recordset.Seek "<=", Trim(txtCariNama.Text) & "zzz"
    If Data1.Recordset.NoMatch Then

```

```

        If Len(Trim(txtCariKode.Text)) = 30 Then
            Beep
        End If
    End If
End Sub

Private Sub txtKode_Change()
    If Len(Trim(txtKode.Text)) < 5 Then
        Exit Sub
    End If
    Data1.Recordset.Index = "KodeCst"
    Data1.Recordset.Seek "=", txtKode.Text
    If Not Data1.Recordset.NoMatch Then
        On Error Resume Next 'jika field kosong, abaikan!
        txtNama.Text = Data1.Recordset!Nama
        txtAlamat.Text = Data1.Recordset!Alamat
        txtKota.Text = Data1.Recordset!Kota
        txtHubungan.Text = Data1.Recordset!Hubungan
        txtTelepon.Text = Data1.Recordset!Telepon
        TidakSiaplsi
        On Error GoTo 0
        Exit Sub
    End If
    Siaplsi
    BlankForm
    txtNama.SetFocus
End Sub

Private Sub txtKode_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtAlamat_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtKota_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub txtHubungan_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Function TidakSiaplsi()
    txtNama.Enabled = False

```

```

txtAlamat.Enabled = False
txtKota.Enabled = False
txtHubungan.Enabled = False
txtTelepon.Enabled = False
txtNama.BackColor = &H8000000F
txtAlamat.BackColor = &H8000000F
txtKota.BackColor = &H8000000F
txtHubungan.BackColor = &H8000000F
txtTelepon.BackColor = &H8000000F
cmdSimpan.Enabled = False
End Function

```

```

Private Function SiapIsi()
    txtNama.Enabled = True
    txtAlamat.Enabled = True
    txtKota.Enabled = True
    txtHubungan.Enabled = True
    txtTelepon.Enabled = True
    txtNama.BackColor = &H8000000E
    txtAlamat.BackColor = &H8000000E
    txtKota.BackColor = &H8000000E
    txtHubungan.BackColor = &H8000000E
    txtTelepon.BackColor = &H8000000E
    cmdSimpan.Enabled = True
End Function

```

```

Private Function BlankForm()
    txtNama.Text = ""
    txtAlamat.Text = ""
    txtKota.Text = ""
    txtHubungan.Text = ""
    txtTelepon.Text = ""
End Function

```

- i. Simpan project dan form lalu jalankan.

d) Program Input Pembelian

- Klik sub menu Pembelian, Input pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmInBeli. Ketikkan kode programnya seperti berikut ini :

```

Private Sub mnInputBeli_Click()
    frmInBeli.Show
End Sub

```

- Klik menu Project, Add Form, Form, Open.
- Buat form seperti berikut :

- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	frmInBeli Pemasukan Data Pembelian
Data1	DatabaseName	C:\Pembelian.mdb
	RecordsetType	0-Table
	RecordSource	Supplier
	Visible	False
Data2	DatabaseName	C:\Pembelian.mdb
	RecordsetType	0-Table
	RecordSource	Barang
	Visible	False
DBCombo1	RowSource	Data1
	ListField	Kode
DBCombo2	RowSource	Data2
	ListField	Kode
Label1	Caption	Nomor Faktur
Label2	Caption	Kode Supplier
Label3	Caption	Kode Barang
Label4	Caption	Harga Satuan
Label5	Caption	Banyak
Label6	Caption	Tgl. Faktur
Label7	Caption	Nama

Label8	Caption	Nama
Label9	Caption	Jumlah
Text1	Name	txtNoFak
	Caption	kosongkan
Text2	Name	txtTgFak
	Caption	kosongkan
Text3	Name	txtNamaSpl
	Caption	kosongkan
Text4	Name	txtNamaBrgl
	Caption	Kosongkan
Text5	Name	txtHarga
	Caption	kosongkan
Text6	Name	TxtBanyak
	Caption	kosongkan
Text7	Name	TxtJumlah
	Caption	kosongkan
Command1	Name	CmdSimpan
	Caption	Simpan
Command2	Name	CmdBatal
	Caption	Batal
Command3	Name	CmdSelesai
	Caption	Selesai

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```
Dim wrkJet As Workspace
Dim dbPembelian As Database
Dim rsBeli As Recordset
```

```
Private Sub cmdBatal_Click()
    BlankForm
    txtNoFak.SetFocus
End Sub
```

```
Private Sub cmdSelesai_Click()
    dbPembelian.Close
    frmInBeli.Hide
    frmMenu.Show
End Sub
```

```

Private Sub cmdSimpan_Click()
    'tambahkan record baru pada tabel pembelian
    rsBeli.AddNew
    'simpan isian form ke dalam tabel pembelian
    rsBeli!NoFaktur = txtNoFak.Text
    rsBeli!TgFaktur = txtTgFak.Text
    rsBeli!KodeSpl = DBCombo1.Text
    rsBeli!KodeBrg = DBCombo2.Text
    rsBeli!Harga = txtHarga.Text
    rsBeli!Banyak = txtBanyak.Text
    'update record
    rsBeli.Update
    Beep
    BlankForm
    txtNoFak.SetFocus
End Sub

Private Sub DBCombo1_Change()
    If Len(Trim(DBCombo1.Text)) < 5 Then
        Exit Sub
    End If
    'cari kode supplier dalam index tabel supplier
    Data1.Recordset.Seek "=", DBCombo1.Text
    If Data1.Recordset.NoMatch Then
        'jika tidak ditemukan, data tidak valid
        X = MsgBox("Kode supplier tidak ada!", vbOKOnly)
        txtNamaSpl.Text = ""
        DBCombo1.SetFocus
        Exit Sub
    End If
    txtNamaSpl.Text = Data1.Recordset!Nama
End Sub

Private Sub DBCombo1_KeyPress(KeyAscii As Integer)
    'mengubah huruf kecil jadi huruf besar
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub DBCombo2_KeyPress(KeyAscii As Integer)
    'mengubah huruf kecil jadi huruf besar
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub DBCombo2_Change()
    If Len(Trim(DBCombo2.Text)) < 5 Then
        Exit Sub
    End If
    'cari kode barang dalam index tabel barang
    Data2.Recordset.Seek ">=", DBCombo2.Text
    If Data2.Recordset.NoMatch Then

```

```

        'jika tidak ditemukan, data tidak valid
        X = MsgBox("Kode barang tidak ada!", vbOKOnly)
        txtNamaBrg.Text = ""
        DBCombo2.SetFocus
        Exit Sub
    End If
    txtNamaBrg.Text = Data2.Recordset!Nama
    txtHarga.Text = Str(Data2.Recordset!Harga)
End Sub

Private Sub Form_Activate()
    Set wrkJet = CreateWorkspace("", "admin", "", dbUseJet)
    Set dbPembelian = OpenDatabase("C:\LatVB6\Pembelian.mdb")
    Set rsBeli = dbPembelian.OpenRecordset("Beli")
    'tentukan index supplier
    Data1.Recordset.Index = "KodeSpl"
    'tentukan index barang
    Data2.Recordset.Index = "KodeBrg"
    txtNoFak.MaxLength = 10
    txtTgFak.MaxLength = 12
    txtHarga.MaxLength = 10
    txtBanyak.MaxLength = 10
    txtNamaSpl.Enabled = False
    txtNamaBrg.Enabled = False
    txtJumlah.Enabled = False
    txtNamaSpl.BackColor = &H8000000F
    txtNamaBrg.BackColor = &H8000000F
    txtJumlah.BackColor = &H8000000F
    cmdSimpan.Enabled = False
    DBCombo1.ToolTipText = "Ketik atau pilih kode supplier"
    DBCombo2.ToolTipText = "Ketik atau pilih kode barang"
    BlankForm
End Sub

Private Sub BlankForm()
    DBCombo1.Text = ""
    DBCombo2.Text = ""
    txtNoFak.Text = ""
    txtTgFak.Text = ""
    txtNamaSpl.Text = ""
    txtNamaBrg.Text = ""
    txtHarga.Text = ""
    txtBanyak.Text = ""
    txtJumlah.Text = ""
End Sub

Private Sub txtBanyak_KeyPress(KeyAscii As Integer)
    'hanya boleh diisi angka atau backspace

```

```

    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack) Then
        Beep
        KeyAscii = 0
    End If
End Sub

```

```

Private Sub txtHarga_Change()
    Dim MJumlah As Single
    MJumlah = Val(txtHarga.Text) * Val(txtBanyak.Text)
    On Error GoTo Salah
    txtJumlah.Text = Format(MJumlah, "Rp ###,###,###") & ",-"
    On Error GoTo 0
    Exit Sub
Salah:
End Sub

```

```

Private Sub txtBanyak_Change()
    Dim MJumlah As Single
    MJumlah = Val(txtHarga.Text) * Val(txtBanyak.Text)
    On Error GoTo Salah
    txtJumlah.Text = Format(MJumlah, "Rp ###,###,###") & ",-"
    On Error GoTo 0
    Exit Sub
Salah:
End Sub

```

```

Private Sub txtHarga_KeyPress(KeyAscii As Integer)
    'hanya boleh diisi angka atau backspace
    If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack) Then
        Beep
        KeyAscii = 0
    End If
End Sub

```

```

Private Sub txtNoFak_LostFocus()
    cmdSimpan.Enabled = True
End Sub

```

```

Private Sub txtTgFak_LostFocus()
    On Error GoTo SalahTanggal
    CekTanggal = CDate(txtTgFak.Text)
    On Error GoTo 0
    Exit Sub
SalahTanggal:
    If Len(Trim(txtTgFak.Text)) = 0 Then
        Exit Sub
    End If

```

```

X = MsgBox("Format tanggal: dd-mm-yyyy" & Chr(13) & "Misalnya:
22-01-1999", vbOKOnly)
txtTgFak.SetFocus
End Sub

Private Sub txtNoFak_KeyPress(KeyAscii As Integer)
'mengubah huruf kecil jadi huruf besar
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

```

- i. Simpan project dan form lalu jalankan

e) Program Browse Pembelian

- Klik sub menu Pembelian, Browse pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmBrowseBeli.

Ketikkan kode programnya seperti berikut ini :

```

Private Sub mnBrowseBeli_Click()
frmBrowseBeli.Show
End Sub

```

- Klik menu Project, Add Form, Form, Open.
- Ubah properti Form1 sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	FrmBrowseBeli Informasi Data Pembelian

- Klik ganda pada objek **SSTab** lalu atur tampilannya seperti gambar dibawah ini.
Jika objek SSTab tidak ada dalam toolbox, tambahkan lewat menu **Project, Components**, pilih **Microsoft Tabbed Dialog Control 6.0**.

Informasi Data Pembelian

No. Faktur Kode Supplier Kode Barang

No. Faktur: F005 Jumlah Barang: 84 Nilai Pembelian: 206.400

Mulai Cari Data Pembelian

Tabel Data Pembelian

	KodeSpl	KodeBrg	NoFaktur
▶	S.002	BT.003	F005
	S.002	PS.001	F005
	S.003	KP.003	F005
	S.002	KP.003	F005
	S.001	PS.002	F006
	S.002	PS.001	F007
	S.002	KP.003	F008

Close

- Klik pada tab pertama (**TAB 0**) lalu ubah properti **Caption**nya menjadi **No. Faktur**.
- Klik pada tab pertama (**TAB 1**) lalu ubah properti **Caption**nya menjadi **Kode Supplier**.
- Klik pada tab pertama (**TAB 2**) lalu ubah properti **Caption**nya menjadi **Kode Barang**.
- Klik kembali tab pertama (**No. Faktur**) lalu tambahkan objek-objeknya seperti gambar di atas (cara penambahan objek ke dalam **SSTab** tidak boleh menggunakan cara klik ganda).
- Klik pada objek data (**Data 1**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Pembelian
DatabaseName	C:\Pembelian.mdb
RecordSource	Beli

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Pembelian
DataSource	Data 1

- Klik tab ke dua (**Kode Supplier**) lalu tambahkan objek-objek seperti gambar berikut ini :

- Klik pada objek data (**Data 2**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Pembelian
DatabaseName	C:\Pembelian.mdb
RecordSource	Beli

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Pembelian
DataSource	Data 2

- Klik tab ke tiga (**Kode Barang**) lalu tambahkan objek-objek seperti gambar berikut ini :

- Klik pada objek data (**Data 3**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Pembelian
DatabaseName	C:\Pembelian.mdb
RecordSource	Beli

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Pembelian
DataSource	Data 3

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```
Private Sub Command2_Click()
    Dim MBanyak, MNilai As Single
    Dim MCari As String * 5
    Data2.Recordset.Index = "KodeSpl"
    MCari = Text4.Text
    Data2.Recordset.Seek "=", MCari
    If Data2.Recordset.NoMatch Then
        Text5.Text = ""
        Text6.Text = ""
        Beep
    End If
    Exit Sub
    End If
    MBanyak = 0
    MNilai = 0
    Do While Data2.Recordset!KodeSpl = MCari
```

```

        MBanyak = MBanyak + Data2.Recordset!Banyak
        MNilai   =   MNilai   +   (Data2.Recordset!Banyak   *
                        Data2.Recordset!Harga)
        Data2.Recordset.MoveNext
        If Data2.Recordset.EOF = True Then
            Exit Do
        End If
    Loop
    Data2.Recordset.Seek "=", MCari
    Text5.Text = Format(MBanyak, "##,###,###")
    Text6.Text = Format(MNilai, "##,###,###,###")
End Sub

Private Sub Command4_Click()
    frmBrowseBeli.Hide
    frmMenu.Show
End Sub

Private Sub Text4_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub Command1_Click()
    Dim MBanyak, MNilai As Single
    Dim MCari As String * 10
    Data1.Recordset.Index = "NoFaktur"
    MCari = Text1.Text
    Data1.Recordset.Seek "=", MCari
    If Data1.Recordset.NoMatch Then
        Text2.Text = ""
        Text3.Text = ""
        Beep
        Exit Sub
    End If
    MBanyak = 0
    MNilai = 0
    Do While Data1.Recordset!NoFaktur = MCari
        MBanyak = MBanyak + Data1.Recordset!Banyak
        MNilai   =   MNilai   +   (Data1.Recordset!Banyak   *
                                Data1.Recordset!Harga)
        Data1.Recordset.MoveNext
        If Data1.Recordset.EOF = True Then
            Exit Do
        End If
    Loop
    Data1.Recordset.Seek "=", MCari
    Text2.Text = Format(MBanyak, "##,###,###")
    Text3.Text = Format(MNilai, "##,###,###,###")
End Sub

```

```

Private Sub SStab1_GotFocus()
    If SStab1.Caption = "No. Faktur" Then
        Text1.SetFocus
        Command1.Default = True
    ElseIf SStab1.Caption = "Kode Supplier" Then
        Text4.SetFocus
        Command2.Default = True
    Else
        Text7.SetFocus
        Command3.Default = True
    End If
End Sub

Private Sub Text1_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub Command3_Click()
    Dim MBanyak, MNilai As Single
    Dim MCari As String * 6
    Data3.Recordset.Index = "KodeBrg"
    MCari = Text7.Text
    Data3.Recordset.Seek "=", MCari
    If Data3.Recordset.NoMatch Then
        Text8.Text = ""
        Text9.Text = ""
        Beep
        Exit Sub
    End If
    MBanyak = 0
    MNilai = 0
    Do While Data3.Recordset!KodeBrg = MCari
        MBanyak = MBanyak + Data3.Recordset!Banyak
        MNilai = MNilai + (Data3.Recordset!Banyak *
Data3.Recordset!Harga)
        Data3.Recordset.MoveNext
        If Data3.Recordset.EOF = True Then
            Exit Do
        End If
    Loop
    Data3.Recordset.Seek "=", MCari
    Text8.Text = Format(MBanyak, "##,###,###")
    Text9.Text = Format(MNilai, "##,###,###,###")
    On Error GoTo TAdaGambar
    Image1.Stretch = True
    Image1.Picture = LoadPicture("C:\LatVB6\" & Text7.Text & ".BMP")
    Exit Sub
TAdaGambar:
    Image1.Picture = LoadPicture("")
End Sub

```

```
Private Sub Text7_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub
```

- Simpan project dan form lalu jalankan

f) Program Input Penjualan

- Klik sub menu Penjualan, Input pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmInJual. Ketikkan kode programnya seperti berikut ini :

```
Private Sub mnInputJual_Click()
    frmInJual.Show
End Sub
```

- Klik menu Project, Add Form, Form, Open.
- Buat form seperti berikut :

Nama Barang	Kode	Harga Satuan	Banyak	Nilai Penjualan
Buku Tulis AA	BT.003	3.740	120	448.800
Buku Tulis Lokomotif	BT.001	3.300	60	198.000

Jumlah 646.800

Simpan Batal Selesai Data1 Data2 Data3 Data4

- Atur property untuk kontrol-kontrolnya sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	FrmInJual Pemasukan Data Penjualan
Data1	Name	dbCustomer
	Connect	Access
	DatabaseName	C:\Penjualan.mdb
	RecordsetType	0-Table
	RecordSource	Customer
	Visible	False
Data2	Name	DbJual
	Connect	Access
	DatabaseName	C:\Penjualan.mdb
	RecordsetType	0-Table
	RecordSource	Jual
	Visible	False
Data3	Name	DbSemjual
	Connect	Access
	DatabaseName	C:\Penjualan.mdb
	RecordsetType	0-Table
	RecordSource	Semjual
	Visible	False
Data4	Name	DbBarang
	Connect	Access
	DatabaseName	C:\Penjualan.mdb
	RecordsetType	0-Table
	RecordSource	Barang
	Visible	False
Label1	Caption	Nomor Bon
Label2	Caption	Kode Customer
Label3	Caption	Tgl. Bon
Label4	Caption	Nama
Label5	Caption	Jumlah
Text1	Name	txtNoBon
	Caption	kosongkan
Text2	Name	TxtTgBon
	Caption	Kosongkan

Text3	Name	TxtNamaCst
	Caption	Kosongkan
Text4	Name	TxtJumlah
	Alignment	1 – Right Justify
	Caption	Kosongkan
	MultiLine	True
DBCombo	Name	dbCombo1
	RowSource	DbCustomer
	ListField	Kode
	Style	0 – dbcDropDownCombo
DBGrid	Name	DBGrid1
	AllowAddNew	True
	AllowUpdate	True
	DataSource	DBSemJual
	TabAction	1 – Column Navigation
Command1	Name	CmdSimpan
	Caption	Simpan
Command2	Name	CmdBatal
	Caption	Batal
Command3	Name	CmdSelesai
	Caption	Selesai

- Klik pada objek DBGrid1 lalu klik kanan sampai muncul menu tambahan, kemudian pilih Retrieve Fields untuk memuat field tabel Semjual ke dalam objek DBGrid1.
- Klik pada objek DBGrid1 lalu klik kanan sampai muncul menu tambahan, kemudian pilih Properties sampai muncul kotak dialog Property Pages.
- Klik tab Columns, lalu pilihlah kolom Harga dan isilah kolom NumberFormat dengan karakter #,### agar bisa menampilkan pemisah ribuan. Lakukan hal yang sama pada kolom Banyak dan kolom Nilai.
- Klik lembar Layout dan tentukan lebar masing-masing kolom secara proporsional agar seluruh kolom dapat tampil di layar monitor.

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```

Dim MJumlah As Single
Private Sub cmdBatal_Click()
    BlankForm
    txtNoBon.SetFocus
End Sub

Private Sub cmdSelesai_Click()
    frmInJual.Hide
    frmMenu.Show
End Sub

Private Sub cmdSimpan_Click()
    'bawa ke record pertama dari tabel sementara
    dbSemJual.Recordset.MoveFirst
    'ulang sampai record terakhir
    Do While Not dbSemJual.Recordset.EOF
        'jika record kosong, jangan disimpan
        If Not dbSemJual.Recordset!Nama = Space(30) Then
            'tambahkan record baru ke data penjualan
            dbJual.Recordset.AddNew
            'isi record penjualan lalu update
            dbJual.Recordset!NoBon = txtNoBon.Text
            dbJual.Recordset!TgBon = txtTgBon.Text
            dbJual.Recordset!KodeCst = DBCombo1.Text
            dbJual.Recordset!KodeBrg = dbSemJual.Recordset!Kode
            dbJual.Recordset!Harga = dbSemJual.Recordset!Harga
            dbJual.Recordset!Banyak = dbSemJual.Recordset!Banyak
            dbJual.Recordset.Update
        End If
        'proses record berikutnya
        dbSemJual.Recordset.MoveNext
    Loop
    BlankForm
    Form_Activate
End Sub

Private Sub DBCombo1_Change()
    If Len(Trim(DBCombo1.Text)) < 5 Then
        Exit Sub
    End If
    'cari kode customer dalam index tabel customer
    dbCustomer.Recordset.Index = "KodeCst"
    dbCustomer.Recordset.Seek "=", DBCombo1.Text
    If dbCustomer.Recordset.NoMatch Then
        'jika tidak ditemukan, data tidak valid
        X = MsgBox("Kode customer tidak ada!", vbOKOnly)
        txtNamaCst.Text = ""
        DBCombo1.SetFocus
    End If
End Sub

```

```

Exit Sub
End If
txtNamaCst.Text = dbCustomer.Recordset!Nama
End Sub

Private Sub DBCombo1_KeyPress(KeyAscii As Integer)
    'mengubah huruf kecil jadi huruf besar
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub DBGrid1_AfterColEdit(ByVal ColIndex As Integer)
    'simpan data harga dan banyak ke variabel memory
    MHarga = dbSemJual.Recordset!Harga
    MBanyak = dbSemJual.Recordset!Banyak
    'jika yang diedit adalah kolom nama barang
    If DBGrid1.Col = 0 Then
        dbBarang.Recordset.Index = "NamaBrg"
        dbBarang.Recordset.Seek "=", UCase(DBGrid1.SelText)
        If dbBarang.Recordset.NoMatch Then
            'jika nama barang tidak ada, ke isian kode barang
            Beep
            DBGrid1.Col = 1
            Exit Sub
        End If
        'jika nama barang ada, update nama, kode dan harga
        'pada tabel isian sementara
        dbSemJual.Recordset.Edit
        dbSemJual.Recordset!Nama = dbBarang.Recordset!Nama
        dbSemJual.Recordset!Kode = dbBarang.Recordset!Kode
        dbSemJual.Recordset!Harga = dbBarang.Recordset!Harga * 1.1
        dbSemJual.Recordset.Update
        'ke isian harga
        DBGrid1.Col = 2
        DBGrid1.Refresh
        Exit Sub
    End If
    'jika yang diedit adalah kolom kode barang
    If DBGrid1.Col = 1 Then
        dbBarang.Recordset.Index = "KodeBrg"
        dbBarang.Recordset.Seek "=", DBGrid1.SelText
        If dbBarang.Recordset.NoMatch Then
            'jika kode barang tidak ada, tetap di isian kode barang
            Beep
            DBGrid1.Col = 1
            Exit Sub
        End If
        'jika kode barang ada, update nama, kode dan harga
        'pada tabel isian sementara
        dbSemJual.Recordset.Edit
        dbSemJual.Recordset!Nama = dbBarang.Recordset!Nama

```

```

dbSemJual.Recordset!Kode = dbBarang.Recordset!Kode
dbSemJual.Recordset!Harga = dbBarang.Recordset!Harga * 1.1
dbSemJual.Recordset.Update
'ke isian harga
DBGrid1.Col = 2
DBGrid1.Refresh
Exit Sub
End If
'jika yang diedit adalah kolom harga barang
If DBGrid1.Col = 2 Then
    'update harga dan nilai pada tabel isian sementara
    dbSemJual.Recordset.Edit
    dbSemJual.Recordset!Harga = DBGrid1.SelText
    dbSemJual.Recordset!Nilai      =      DBGrid1.SelText      *
dbSemJual.Recordset!Banyak
    dbSemJual.Recordset.Update
    'update jumlah penjualan
    MJumlah = MJumlah + (DBGrid1.SelText - MHarga) * MBanyak
    txtJumlah.Text = Format(MJumlah, "#,###")
    DBGrid1.Col = 3
    Exit Sub
End If
'jika yang diedit adalah kolom banyak barang
If DBGrid1.Col = 3 Then
    'update banyak dan nilai pada tabel isian sementara
    dbSemJual.Recordset.Edit
    dbSemJual.Recordset!Banyak = DBGrid1.SelText
    dbSemJual.Recordset!Nilai      =      DBGrid1.SelText      *
dbSemJual.Recordset!Harga
    dbSemJual.Recordset.Update
    'update jumlah penjualan
    MJumlah = MJumlah + (DBGrid1.SelText - MBanyak) * MHarga
    txtJumlah.Text = Format(MJumlah, "#,###")
    DBGrid1.Col = 4
End If
'jika yang diedit adalah kolom nilai
If DBGrid1.Col = 4 Then
    'ke record berikutnya
    dbSemJual.Recordset.MoveNext
    'mulai isi nama barang
    DBGrid1.Col = 0
End If
End Sub

Private Sub Form_Activate()
    MJumlah = 0
    txtNoBon.MaxLength = 10
    txtTgBon.MaxLength = 12
    txtNamaCst.Enabled = False
    txtNamaCst.BackColor = &H8000000F

```

```

cmdSimpan.Enabled = False
'jika tabel sementara sudah ada isinya,
'hapus seluruh recordnya
If Not dbSemJual.Recordset.RecordCount = 0 Then
    dbSemJual.Recordset.MoveFirst
    Do While Not dbSemJual.Recordset.EOF
        dbSemJual.Recordset.Delete
        dbSemJual.Recordset.MoveNext
    Loop
End If
'isi dengan 15 record kosong
For i = 1 To 15
    dbSemJual.Recordset.AddNew
    dbSemJual.Recordset!Harga = 0
    dbSemJual.Recordset!Banyak = 0
    dbSemJual.Recordset.Update
Next i
'ke record pertama
dbSemJual.Recordset.MoveFirst
'siap mengisi tabel sementara
txtNoBon.SetFocus
End Sub

Private Sub BlankForm()
    DBCombo1.Text = ""
    txtNoBon.Text = ""
    txtTgBon.Text = ""
    txtNamaCst.Text = ""
    txtJumlah.Text = ""
End Sub

Private Sub txtNoBon_LostFocus()
    cmdSimpan.Enabled = True
End Sub

Private Sub txtTgBon_LostFocus()
    On Error GoTo SalahTanggal
    CekTanggal = CDate(txtTgBon.Text)
    On Error GoTo 0
    Exit Sub
SalahTanggal:
    If Len(Trim(txtTgBon.Text)) = 0 Then
        Exit Sub
    End If
    X = MsgBox("Format tanggal: dd-mm-yyyy" & Chr(13) & "Misalnya:
22-01-1999", vbOKOnly)
    txtTgBon.SetFocus
End Sub

Private Sub txtNoBon_KeyPress(KeyAscii As Integer)

```

```

'mengubah huruf kecil jadi huruf besar
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

```

- Simpan project dan form lalu jalankan

g) Program Browse Penjualan

- Klik sub menu Penjualan, Browse pada form frmMenu.
- Buat procedure untuk menghubungkan frmMenu dengan frmBrowseJual.

Ketikkan kode programnya seperti berikut ini :

```

Private Sub mnBrowseJual_Click()
    frmBrowseJual.Show
End Sub

```

- Klik menu Project, Add Form, Form, Open.
- Ubah properti Form1 sbb :

Kontrol	Properti	Nilai
Form1	Name Caption	FrmBrowseJual Informasi Data Penjualan

- Klik ganda pada objek **SSTab** lalu atur tampilannya seperti gambar dibawah ini.

Jika objek SSTab tidak ada dalam toolbox, tambahkan lewat menu **Project**, **Components**, pilih **Microsoft Tabbed Dialog Control 6.0**.

KodeCst	KodeBrng	NoBon	TgBon	Banyak	Harga
C.002	BT.003	B005	1/15/00	120	3.740
C.002	KP.001	B005	1/15/00	24	1.650
C.001	PS.002	B006	1/15/00	60	4.950
C.001	BT.003	B006	1/15/00	120	3.740
C.001	BT.001	B006	1/15/00	120	3.300
C.002	PS.002	B008	1/15/00	60	4.950
C.002	BT.003	B008	1/15/00	120	3.740

- Klik pada tab pertama (**TAB 0**) lalu ubah properti **Caption**nya menjadi **No. Bon**.
- Klik pada tab pertama (**TAB 1**) lalu ubah properti **Caption**nya menjadi **Kode Customer**.
- Klik pada tab pertama (**TAB 2**) lalu ubah properti **Caption**nya menjadi **Kode Barang**.
- Klik kembali tab pertama (**No. Bon**) lalu tambahkan objek-objeknya seperti gambar di atas (cara penambahan objek ke dalam **SSTab** tidak boleh menggunakan cara klik ganda).
- Klik pada objek data (**Data 1**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Penjualan
DatabaseName	C:\Penjualan.mdb
RecordSource	Jual

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Penjualan
DataSource	Data 1

- Klik tab ke dua (**Kode Customer**) lalu tambahkan objek-objek seperti gambar berikut ini :

Informasi Data Penjualan

Tab: **Kode Customer**

Kode Customer: C.002 Jumlah Barang: 852 Nilai Pembelian: 3.517.800

Mulai Cari Data Penjualan

	KodeCst	KodeBrg	NoBon	TgBon	Banyak	Harga
▶	C.002	PS.002	F001	12/20/99	60	4.950
	C.002	BT.001	F001	12/20/99	120	3.300
	C.002	PS.002	B003	1/5/00	48	4.950
	C.002	PS.002	B005	1/15/00	60	4.950
	C.002	BT.003	B005	1/15/00	120	3.740
	C.002	KP.001	B005	1/15/00	24	1.650
	C.002	PS.002	B005	1/15/00	60	4.950

Close

- Klik pada objek data (**Data 2**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Penjualan
DatabaseName	C:\Penjualan.mdb
RecordSource	Jual

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Penjualan
DataSource	Data 2

- Klik tab ke tiga (**Kode Barang**) lalu tambahkan objek-objek seperti gambar berikut ini :

- Klik pada objek data (**Data 3**) lalu ubah propertinya menjadi sbb :

Properti	Nilai
Caption	Data Penjualan
DatabaseName	C:\Penjualan.mdb
RecordSource	Jual

- Klik pada objek **DBGrid1** lalu ubah propertinya menjadi sbb:

Properti	Nilai
Caption	Tabel Data Penjualan
DataSource	Data 3

- Tampilkan editor kode program dan tulislah kode programnya seperti berikut :

```
Private Sub Command2_Click()
    Dim MBanyak, MNilai As Single
    Dim MCari As String * 5
    Data2.Recordset.Index = "KodeCst"
    MCari = Text4.Text
    Data2.Recordset.Seek ">=", MCari, CDate("01-01-1900")
    If Data2.Recordset.NoMatch Then
        Text5.Text = ""
        Text6.Text = ""
    End If
End Sub
```

```

        Beep
    Exit Sub
End If
MBanyak = 0
MNilai = 0
Do While Data2.Recordset!KodeCst = MCari
    MBanyak = MBanyak + Data2.Recordset!Banyak
    MNilai = MNilai + (Data2.Recordset!Banyak *
Data2.Recordset!Harga)
    Data2.Recordset.MoveNext
    If Data2.Recordset.EOF = True Then
        Exit Do
    End If
Loop
Data2.Recordset.Seek ">=", MCari, CDate("01-01-1900")
Text5.Text = Format(MBanyak, "##,###,###")
Text6.Text = Format(MNilai, "##,###,###,###")
End Sub

Private Sub Command4_Click()
    frmBrowseJual.Hide
    frmMenu.Show
End Sub

Private Sub Text4_KeyPress(KeyAscii As Integer)
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

Private Sub Command1_Click()
    Dim MBanyak, MNilai As Single
    Dim MCari As String * 10
    Data1.Recordset.Index = "NoBon"
    MCari = Text1.Text
    Data1.Recordset.Seek "=", MCari
    If Data1.Recordset.NoMatch Then
        Text2.Text = ""
        Text3.Text = ""
        Beep
    Exit Sub
    End If
    MBanyak = 0
    MNilai = 0
    Do While Data1.Recordset!NoBon = MCari
        MBanyak = MBanyak + Data1.Recordset!Banyak
        MNilai = MNilai + (Data1.Recordset!Banyak *
Data1.Recordset!Harga)
        Data1.Recordset.MoveNext
        If Data1.Recordset.EOF = True Then
            Exit Do
        End If
    End If

```

```

Loop
Data1.Recordset.Seek "=", MCari
Text2.Text = Format(MBanyak, "##,###,###")
Text3.Text = Format(MNilai, "##,###,###,###")
End Sub

```

```

Private Sub SSTab1_GotFocus()
If SSTab1.Caption = "No. Bon" Then
Text1.SetFocus
Command1.Default = True
ElseIf SSTab1.Caption = "Kode Customer" Then
Text4.SetFocus
Command2.Default = True
Else
Text7.SetFocus
Command3.Default = True
End If

```

```
End Sub
```

```

Private Sub Text1_KeyPress(KeyAscii As Integer)
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

```

```

Private Sub Command3_Click()
Dim MBanyak, MNilai As Single
Dim MCari As String * 6
Data3.Recordset.Index = "KodeBrg"
MCari = Text7.Text
Data3.Recordset.Seek ">=", MCari, CDate("01-01-1900")
If Data3.Recordset.NoMatch Then
Text8.Text = ""
Text9.Text = ""
Beep
Exit Sub
End If
MBanyak = 0
MNilai = 0
Do While Data3.Recordset!KodeBrg = MCari
MBanyak = MBanyak + Data3.Recordset!Banyak
MNilai = MNilai + (Data3.Recordset!Banyak *
Data3.Recordset!Harga)
Data3.Recordset.MoveNext
If Data3.Recordset.EOF = True Then
Exit Do
End If
Loop
Data3.Recordset.Seek ">=", MCari, CDate("01-01-1900")
Text8.Text = Format(MBanyak, "#,###")
Text9.Text = Format(MNilai, "#,###")

```

```
On Error GoTo TAdaGambar
Image1.Stretch = True
Image1.Picture = LoadPicture("C:\LatVB6\" & Text7.Text & ".BMP")
Exit Sub
TAdaGambar:
Image1.Picture = LoadPicture("")
End Sub

Private Sub Text7_KeyPress(KeyAscii As Integer)
KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub
```

- Simpan project dan form lalu jalankan

Daftar Pustaka

- Alam, M. Agus. J. 1999. Belajar Sendiri Microsoft Visual Basic 6.0. Elex Media Komputindo. Jakarta.
- Alam, M. Agus. J. 1999. Belajar Sendiri Manajemen Database Dengan Microsoft Visual Basic 6.0. Elex Media Komputindo. Jakarta.
- Hadi, Rahardian. 2002. Pemrograman Windows API Dengan Microsoft Visual Basic. Elex Media Komputindo. Jakarta.
- Kristanto, Harianto, Ir. 2002. Konsep Perancangan Database. Andi. Yogyakarta.
- Kurniadi, Adi. 2000. Pemrograman Visual Basic 6.0. Elex Media Komputindo. Jakarta.
- Pramono, Djoko. 2001. Mudah Menguasai Visual Basic 6.0. Elex Media Komputindo. Jakarta.
- Yuswanto. 2001. Panduan Belajar Microsoft Visual Basic 5.0. Prestasi Pustaka. Jakarta.