

**DASAR PEMROGRAMAN VISUAL
BASIC 6.0**

M. Miftakul Amin, S. Kom., M. Eng.

Jurusan Teknik Komputer

Politeknik Negeri Sriwijaya Palembang

DAFTAR ISI

- Modul 1** INTREGATED DEVELOPMENT ENVIRONMENT (IDE)
MICROSOFT VISUAL BASIC 6.0
- Modul 2** BEKERJA DENGAN FORM DAN KONTROL
VISUAL BASIC 6.0
- Modul 3** VARIABEL, TIPE DATA, KONSTANTA DAN OPERATOR
- Modul 4** FUNGSI DAN PROSEDUR, DECISIONS, ERROR TRAPPING
- Modul 5** PENGGUNAAN .OCX DAN REPETITIONS
- Modul 6** MENGAkses DATABASE MS. ACCESS MENGGUNAKAN ADO DATA CONTROL
- Modul 7** MENGGUNAKAN ADO LIBRARY & SQL
- Modul 8** PASSING VARIABEL ANTAR FORM
- Modul 9** MEMBUAT LAPORAN DENGAN CRYSTAL REPORT
- Modul 10** VALIDASI INPUT
- Modul 11** PEMROGRAMAN GRAFIK
- Modul 12** MENU BAR, STATUS BAR, TOOLBAR

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 1	INTREGATED DEVELOPMENT ENVIRONMENT (IDE) MICROSOFT VISUAL BASIC 6.0

MICROSOFT VISUAL BASIC 6.0

Visual Basic 6.0 adalah salah satu produk bahasa pemrograman yang dikeluarkan Microsoft, salah satu perusahaan *software* terkemuka di dunia. Visual basic 6.0 merupakan bahasa pemrograman yang mudah digunakan untuk pengembangan aplikasi, baik itu aplikasi kecil maupun aplikasi besar. Dengan banyaknya komponen kontrol yang disediakan oleh visual basic 6.0, membuat para programmer dan pengembang aplikasi lebih mudah dalam pembuatan aplikasi.

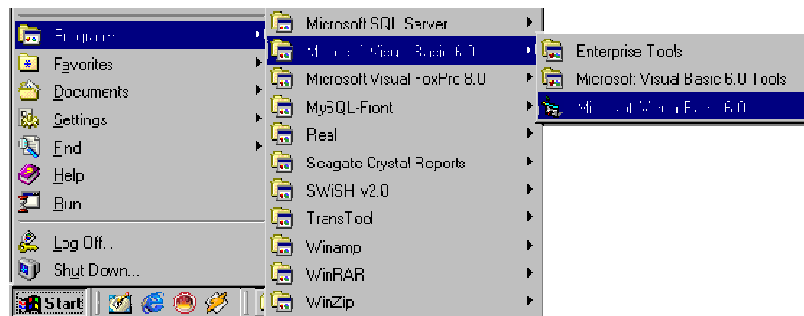
Visual Basic 6.0 memiliki beberapa versi/edisi yang disesuaikan dengan kebutuhan pemakainya. Beberapa versi Visual Basic 6.0 yang sering ditemui di pasaran antara lain:

- Standard Edition (*Learning Edition*)
Versi ini berisi sarana dasar dari Visual Basic 6.0 untuk mengembangkan aplikasi.
- Professional Edition
Versi ini berisi tambahan sarana yang dibutuhkan oleh para programmer professional.
- Enterprise Edition

versi ini dikhususkan bagi para programmer yang ingin mengembangkan aplikasi *Remote computing* atau *Client/Server*.

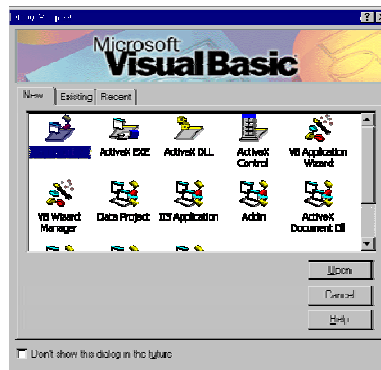
1.1 Memulai Program Visual Basic 6.

1. Pada tampilan utama Window, klik **Start**.
2. Pilih **Programs > Microsoft Visual Basic 6.0 > Microsoft Visual Basic 6.0**.



Gambar 1.1 Cara Membuka Visual Basic

3. Pada kotak dialog **New Project** pastikan Standart.EXE dalam keadaan terpilih, lalu klik tombol open, atau dapat juga dengan menekan **Existing** untuk membuka project yang sudah ada dan **Recent** untuk membuka project yang sudah terbuka sebelumnya.

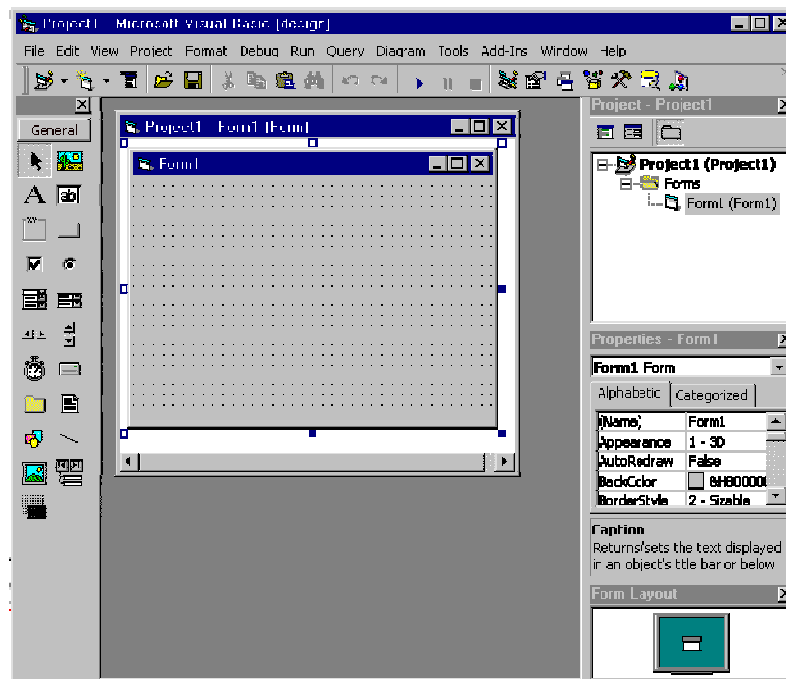


Gambar 1.2 Kotak Dialog New Project

1.2 Tampilan Dasar Visual Basic 6.0

Setelah program Visual Basic 6.0 aktif, maka bidang kerja tempat membuat program aplikasi akan ditampilkan. Tempat ini disebut dengan **Integrated Development Integration (IDE)**, yang terdiri dari komponen-komponen antara lain:

- Menu Bar
- Toolbar
- Toolbox
- Form Window
- Code Window
- Project Explorer
- Properties Window
- Form Layout Window
- Immediate Window
- Watch Window



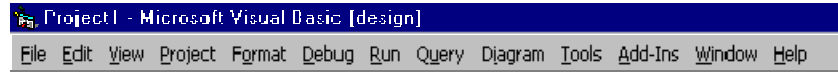
Gambar 1.3 Tampilan bidang kerja Visual Basic 6.0

1.2.1 Menu Bar

Menu bar merupakan kumpulan perintah-perintah yang dikelompokkan dalam kriteria operasinya. Saat bekerja dengan menu bar dapat dilakukan dengan dua cara, yaitu :

- Dengan Mouse :
Klik tombol mouse sebelah kiri pada menu atau sub menu.
- Dengan Keyboard :

Untuk memilih menu, tekan kombinasi tombol : **Alt+F** untuk membuka menu **F**ile, **Ctrl+N** untuk membuat proyek baru, **Ctrl+P** untuk mencetak ke printer dan lain-lain.



Gambar 1.4 Menu Bar

1.2.2 Toolbar

Toolbar fungsinya sama seperti fungsi dari menu bar, hanya saja pada Toolbar pilihan-pilihan berbentuk **Icon**. Untuk memilih suatu proses yang akan dilakukan tinggal mengklik icon yang sesuai dengan proses yang akan diinginkan.



Gambar 1.4 Toolbar

Dengan adanya Toolbar akan memudahkan untuk memilih proses yang sering dilakukan tanpa harus memilihnya pada menu bar.

1.2.3 Toolbox






Toolbox adalah tempat di mana control-control diletakkan. Control-control yang terdapat pada toolbox dipakai dalam pembuatan program aplikasi. Saat pertama kali dijalankan program Visual Basic 6.0 akan menempatkan toolbox di sebelah kiri layar dan berisi 21 control standar.



Gambar 1.5 Toolbox

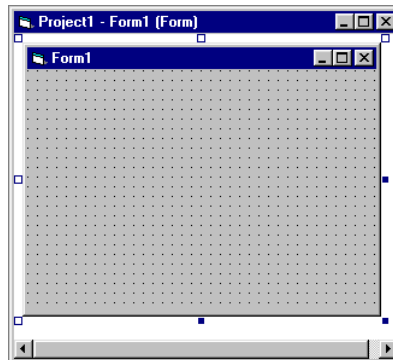
Berikut ini penjelasan dan fungsi dari masing-masing control yang ada dalam Toolbox Visual Basic 6.0, yaitu :

BENTUK	NAMA KONTROL	FUNGSI
	Pointer	Kontrol yang berfungsi untuk memindahkan atau mengubah ukuran control yang ada pada form.
	PictureBox	Untuk menampilkan file gambar (Bitmaps, Icon, Gif, Jpeg dsb).
	Label	Untuk menampilkan teks, tetapi pemakai tidak bisa berinteraksi dengannya.
	TextBox	Untuk menempatkan teks pada form dan pemakai dapat mengedit teks tersebut.

BENTUK	NAMA KONTROL	FUNGSI
	Frame	Untuk mengelompokkan beberapa control (Group) pada suatu form.
	Command Button	Untuk membuat tombol pelaksana suatu perintah atau tindakan ketika digunakan.
	CheckBox	Untuk membuat kotak check yang dapat memilih satu atau banyak pilihan.
	Option Button	Untuk memilih dan mengaktifkan satu pilihan dari banyak pilihan yang ada.
	ComboBox	Sebagai tempat mengetikkan pilihan atau memilih suatu pilihan lewat DropDown – List.
	ListBox	Untuk menampilkan daftar pilihan yang dapat digulung secara horizontal maupun vertical.
	HScrollBar	Untuk menggulung suatu area kerja dengan jangka lebar pada posisi horizontal.
	VScrollBar	Untuk menggulung suatu area kerja dengan jangka lebar pada posisi vertical.
	Timer	Untuk mengoperasikan waktu kejadian pada rutin program dalam interval yang ditentukan.
	DriveListBox	Untuk menampilkan daftar drive computer yang aktif dan dapat dipilih sebuah drive.
	DirListBox	Untuk menampilkan daftar direktori dan path pada drive kerja terpilih.
	FileListBox	Untuk menampilkan daftar file pada direktori dan drive yang aktif.
	Shape	Untuk membentuk obyek dua dimensi seperti square, oval, ellips dan ain-lain.
	Line	Untuk menggambar garis lurus dengan banyak variasi dengan ketebalan yang bisa diatur.
	Image	Untuk menampilkan gambar icon bitmap, atau metafile pada form.
	DataControl	Sebagai sarana akses data dalam suatu database.
	OLE	Untuk menghasilkan proses <i>link</i> dan <i>Embed</i> obyek antar aplikasi.

1.2.4 Form Window

Form window (jendela form) merupakan area keruntuk merancang program aplikasi. Pada jendela form inilah diletakkan control-kontrol (obyek) seperti command button, textbox, label dan lain-lain. Ukuran dari jendela form pada mulanya dapat kecil namun bisa diubah sesuai dengan kebutuhan.

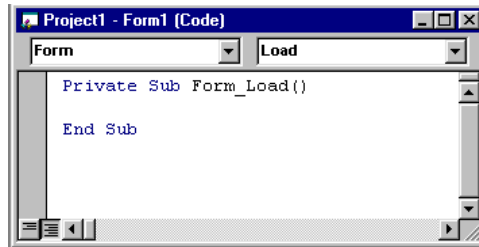


Gambar 1.6 Form

1.2.5 Code Window

Code Window/Kode Editor adalah tempat untuk menuliskan kode program dari aplikasi yang dibuat. Ada lima cara untuk mengaktifkan kode editor, sebagai berikut :

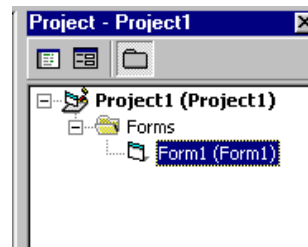
1. Klik menu **View** lalu klik **Code**.
2. Klik ganda pada form atau objek control yang terdapat pada form.
3. Klik kanan pada form lalu klik **View Code**.
4. Klik kanan pada Project Explorer lalu klik **View Code**.
5. Klik pada icon **View Code** pada Project Explorer.



Gambar 1.7 Code Window

1.2.6 Project Explorer

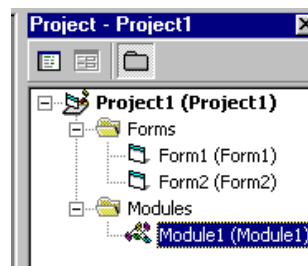
Project Explorer merupakan area berisi semua file program aplikasi Visual Basic 6.0. Suatu aplikasi Visual Basic disebut dengan *Project* (proyek), dan setiap proyek bisa terdiri dari satu atau lebih file misalnya form, modul, class dan lain-lain.



Gambar 1.8 Project Explorer

Pada gambar di atas terlihat bahwa suatu proyek hanya terdiri dari satu form. Untuk menambahkan form atau module ke dalam proyek dapat dilakukan dengan langkah-langkah sebagai berikut :

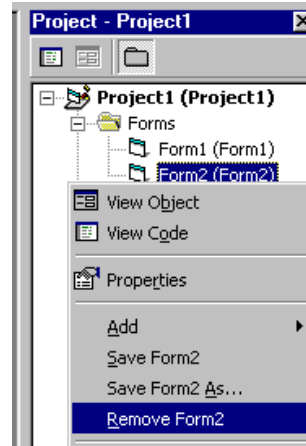
1. Klik menu **Project**, pilih **Add Form**.
2. pada kotak dialog **Add Form**, klik tab **New**, dan pilih **Form**.
3. Klik tombol **Open** untuk menutup kotak dialognya. Sekarang pada Project Explorer ada tambahan satu form baru(Form2).
4. Lakukan hal yang sama untuk menambahkan Module.



Gambar 1.9 Penambahan Form dan Module pada Project Explorer

Selain proses penambahan obyek baru, dapat juga dilakukan penghapusan obyek yang sudah terpasang di Project Explorer. Langkah yang harus dilakukan sebagai berikut :

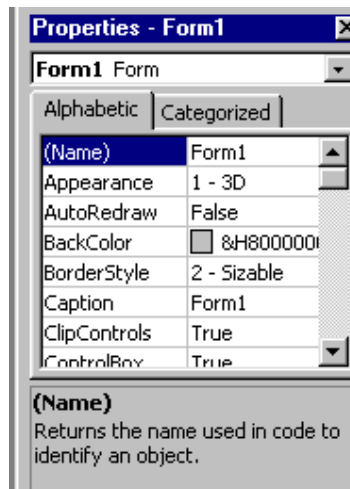
1. Klik kanan mouse pada obyek yang akan dihapus.
2. selanjutnya sebuah menu Pop-Up ditampilkan dan pilih menu **Remove** sesuai dengan nama obyek yang akan dihapus.



Gambar 1.10 Menghapus Form2 pada Project Explorer

1.2.7 Properties Window

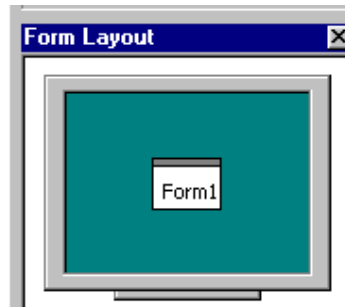
Window ini berisi semua informasi mengenai control (obyek) yang dibuat, dan bertugas menyiapkan segala property dari control yang diperlukan dalam perancangan **User Interface** maupun pemrograman.



Gambar 1.11 Tab Alphabetic pada Properties Windows

1.2.8 Form Layout Window

Merupakan jendela jendela yang menunjukkan tataletak form saat ditampilkan pada layer monitor. Dengan bantuan Form Layout Window ini, akan dapat diketahui posisi aplikasi saat dijalankan.



Gambar 1.12 Form Layout Window

1.2.9 Immediate Window

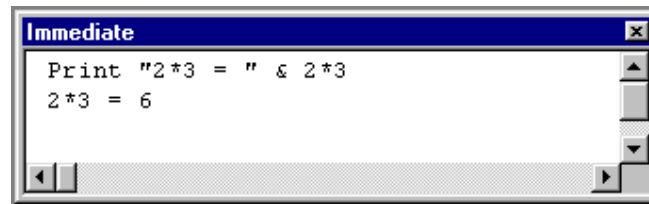
Digunakan untuk untuk mencoba suatu instruksi program, terutama yang berkaitan dengan operasi Aritmatik. Pada saat menguji program, **Immediate Window** juga bisa digunakan sebagai window dialog.

Umumnya saat program Visual Basic 6.0 dijalankan, **Immediate Window** tidak aktif. Untuk mengaktifkan **Immediate Window**, langkah-langkah yang harus dilakukan sebagai berikut :

1. Klik menu **View** pada menu bar.
2. klik **Immediate Window** atau lakukan kombinasi penekanan tombol **Ctrl+G**.

Ketikkan potongan program berikut dalam **Immediate Window** yang akan memproses operasi Aritmatika, yaitu :

1. Ketik Print "2*3=" & 2*3 diikuti dengan menekan tombol **Enter**.
2. Sekarang perhatikan tampilan pada Immediate Window.



Gambar 1.13 Menguji Program pada Immediate Window

1.2.10 Watch Window

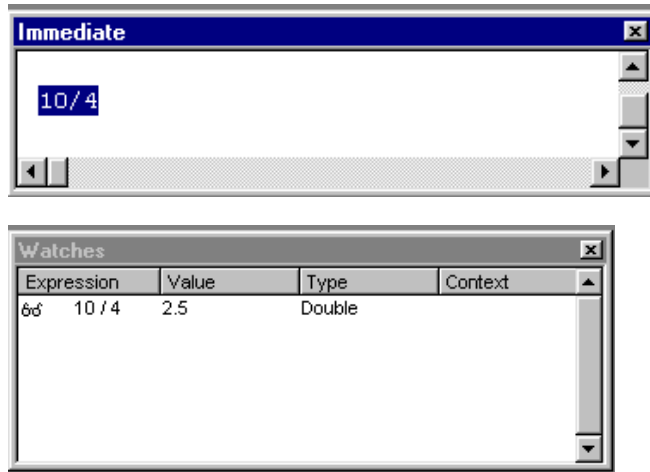
Untuk menampilkan hasil ekspresi yang didefinisikan pada suatu proyek. Dengan **Watch Window**, dapat dihasilkan suatu proses yang ditulis pada **Immediate Window**. Untuk mengaktifkan **Watch Window**, langkah-langkah yang harus dilakukan adalah :

1. Klik menu **View** pada menu bar.
2. Klik **Watch Window**.

Sebagai contoh aktifkan **Immediate Window** dan **Watch Window** kemudian ikuti langkah-langkah berikut :

1. Ketik rumus 2/3 di Immediate Window.
 2. lakukan blok pada rumus **10/4**.
-

3. Selanjutnya lakukan klik pada blok tersebut dan jangan dilepaskan, tarik mouse pointer mouse dan tempatkan ke dalam **Watch Window** dan lepas klik mouse.

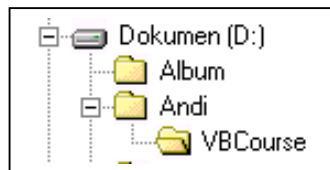


Gambar 1.14 Melihat hasil proses di Watch Window

1.3 Membuat Aplikasi

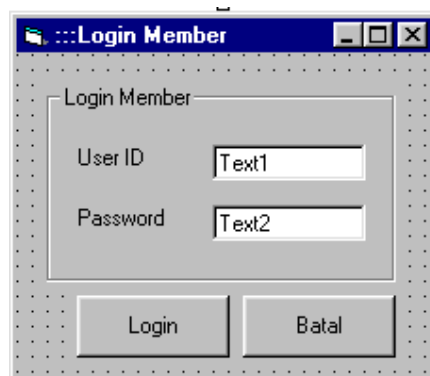
Agar aplikasi yang dibuat dapat terintegrasi dengan baik, terlebih dulu buat folder anda di drive D. Simpan semua pekerjaan di folder anda tersebut.

Sebagai contoh :



Gambar 1.15 Direktori Kerja

Buatlah form seperti berikut :



Gambar 1.16 Form Login


Setting Properties dari masing-masing kontrol sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	Frm_login :::Login Member
Frame1	Caption	Login Member
Label1	Caption	User ID
Label2	Caption	Password
Text1	Text Name	Dikosongkan txtUser
Text2	Text Name PasswordChar	Dikosongkan TxtPass *
Command1	Name Caption	Cmdlogin &Login
Command2	Name Caption	Cmdbatal &Batal

Tulis kode program sebagai berikut :

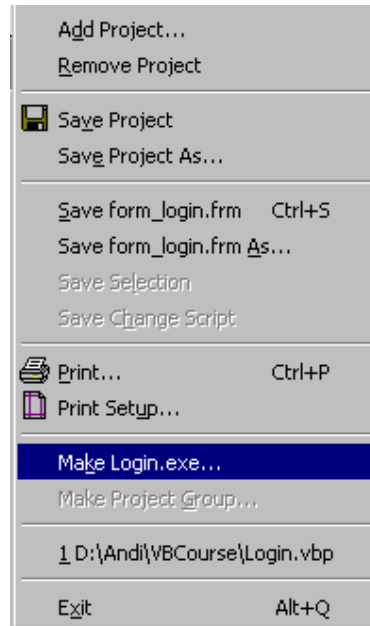
```
Private Sub cmdlogin_Click()  
If txtUser.Text = "" Or txtPass.Text = "" Then  
    MsgBox "Anda tidak berhak mengikuti kelas ini", vbOKOnly + vbExclamation, "Warning"  
Else  
    MsgBox "Selamat Datang : " _  
        & vbCrLf & "User : " & txtUser.Text _  
        & vbCrLf & "Password : " & txtPass.Text _  
        & vbCrLf & "Di Pemrograman Visual Basic 6.0", vbOKOnly + vbInformation,  
    "Congratulation"  
End If  
End Sub  
-----  
Private Sub cmdbatal_Click()  
Unload Me  
End Sub
```

Selanjutnya simpan form dengan nama **frm_login.frm** dan proyek dengan nama **Login.vbp**.

Kemudian untuk menjalankan program di atas dengan menekan icon Run  pada menu **Toolbar** atau dari menu **Run > Start** atau dengan menekan tombol **F5** pada keyboard.

1.4 Membuat File .EXE

Dalam membuat aplikasi Visual Basic 6.0, setelah seluruh program selesai dibuat, langkah selanjutnya adalah mengompilasi project kita menjadi file .EXE, sehingga kita tidak perlu membuka Visual Basic 6.0 untuk menjalankan aplikasi tersebut. Klik menu **File>Make Login.exe**

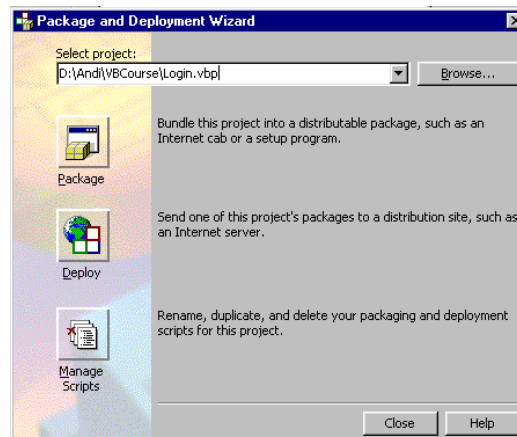


Gambar 1.17 Menu membuat file .EXE

1.5 Membuat File Setup Program

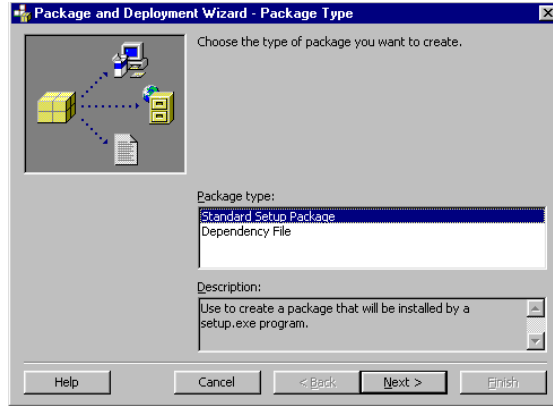
Program .EXE yang baru saja kita buat tersebut dapat berjalan dengan dengan baik di komputer kita yang sudah terinstall Visual Basic 6.0, tetapi belum tentu dapat dijalankan pada komputer lain. Sebuah program .EXE akan memerlukan file-file pendukung yang dipergunakan oleh program tersebut, misalnya file-file kontrol ActiveX atau file-file DLL (**Dynamic Link Library**). Visual Basic 6.0 telah menyediakan fasilitas pembuatan file setup program sehingga program dapat diinstall di komputer lain. Langkah-langkahnya adalah sebagai berikut :

1. Klik **Start > Programs > Microsoft Visual Basic 6.0 > Microsoft Visual Basic 6.0 Tools > Package & Deployment Wizard**. Tentukan nama program yang akan kita buat file setup-nya dengan menekan tombol **Browse**. Dalam hal ini adalah program pada direktori D:\Andi\VB\Course\Login.vbp atau sesuai dengan aplikasi yang anda buat. Setelah itu klik tombol **Package**.



Gambar 1.18 Package and Deployment Wizard

2. Tentukan tipe paket, dalam hal ini **Standart Type Package**
-



Gambar 1.19 Kotak Dialog Tipe Paket

3. Klik Tombol **Next**, untuk menentukan folder tempat penyimpanan file paket. Kita dapat menambahkan folder dengan klik tombol **New Folder**. Misalnya dengan menambahkan folder **Setup**.



Gambar 1.20 Kotak Dialog Folder penyimpanan Paket

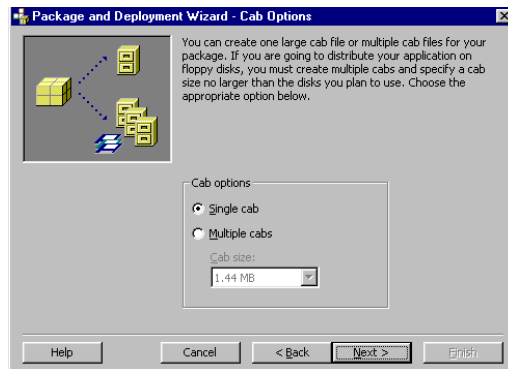
4. Gunakan tombol **Add** untuk menambahkan file anggota. Klik tombol **Next**, akan muncul kotak dialog penentuan file **.CAB**.



Gambar 1.20 Kotak Penentuan file anggota Paket

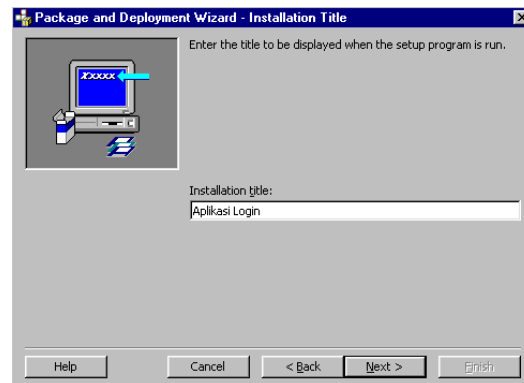
5. Tentukan file **CAB**, jika kita ingin mendistribusikan aplikasi kita dengan floppy disk, kita harus membuat **multiple cabs**, jika kita ingin mendistribusikan program dalam media berkapasitas
-

besar seperti CD-ROM, kita bisa memilih **Single CAB**. Klik tombol **Next**, akan muncul tampilan kotak dengan **Installation Title**.



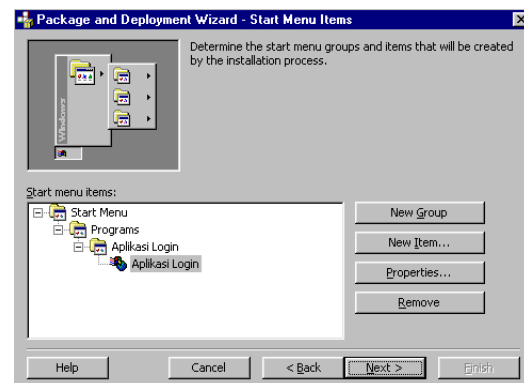
Gambar 1.21 Kotak Penentuan ukuran file .CAB

6. Isi judul Instalasi dengan Aplikasi Login, kemudian klik tombol **Next**,



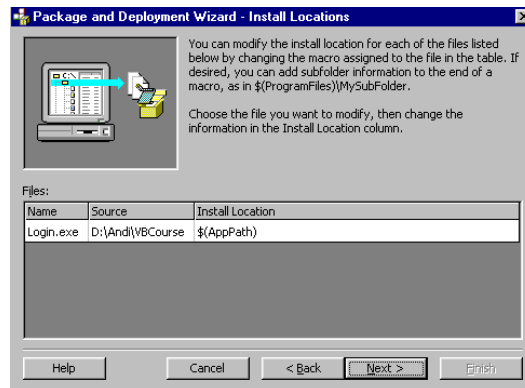
Gambar 1.22 Kotak Penentuan Judul Instalasi

7. selanjutnya akan menampilkan kotak dialog lokasi program hasil setup yang nantinya akan ditampilkan pada komputer baru. Klik tombol **Next**.



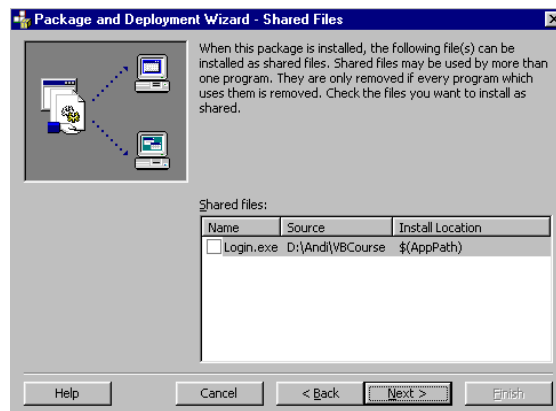
Gambar 1.23 Kotak Dialog lokasi program hasil setup

8. Tampilan selanjutnya adalah lokasi program disimpan di media penyimpanan pada komputer baru. Klik tombol **Next** lagi, kemudian muncul kotak dialog pilihan **Shared files**.



Gambar 1.24 Kotak Dialog lokasi program hasil setup

9. Berilah tanda cek jika ingin program hasil setup dapat dipakai bersama-sama oleh beberapa user. Klik tombol **Next** untuk melanjutkan proses.



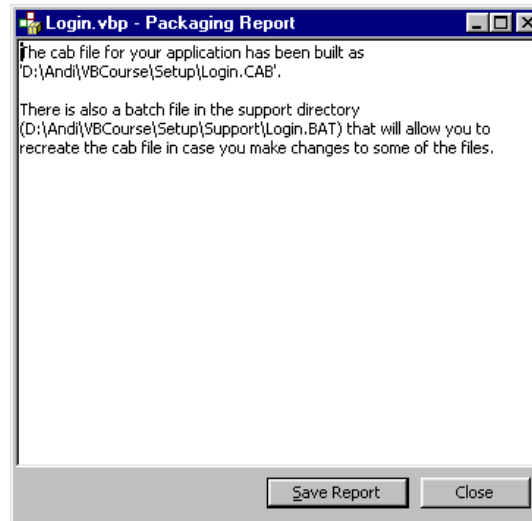
Gambar 1.25 Kotak Dialog pilihan shared file

10. Tampilan terakhir adalah kotak dialog Script Name.



Gambar 1.26 Kotak Dialog Script Name

11. Klik tombol **finish**. Program akan mulai membuat file setup. Tunggu sampai proses selesai, setelah itu akan tampil report. Klik tombol **Save Report** untuk menyimpan catatan atau **Close** untuk menutup report tanpa menyimpannya.



Gambar 1.27 Tampilan Report

Sampai di sini, proses pembuatan file setup telah selesai. Kita dapat copy file-file hasil setup ini ke media lain (floppy disk atau CD-ROM) untuk diinstall ke komputer lain.

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 2	BEKERJA DENGAN FORM DAN CONTROL VISUAL BASIC 6.0

1. FORM DAN CONTROL

Program aplikasi yang dibuat dengan Visual Basic 6.0 bisa dikatakan tidak pernah lepas dari pemakaian form dan kontrol. Sebenarnya kontrol pada Visual Basic 6.0 tidak hanya yang ada pada **Toolbox** saja, masih banyak lagi yang tersimpan sebagai komponen tambahan(.OCX).

Kontrol standard terdapat pada semua versi Visual Basic 6.0, baik pada Learning Edition, Profesional Edition, ataupun Enterprise Edition.

Properti

Setiap kontrol/objek akan mempunyai sifat, seperti misalnya seberapa lebar form yang akan kita buat, apa judul yang akan kita berikan ke form, di posisi mana kontrol akan kita tempatkan, hal-hal semacam inilah yang disebut dengan property/sifat dari kontrol. Dalam pemrograman Visual Basic 6.0 disediakan berbagai macam property yang dipergunakan untuk menyeting kontrol yang kita gunakan.

Contoh :

Form1.Caption “:::Recording Manual Book”

Memberikan property judul kepada sebuah form

Event

Kontrol-kontrol tersebut akan bekerja apabila pada kontrol tersebut dikenai sebuah kejadian, misalnya tombol diklik, textbox diisi dan lain sebagainya. Kejadian-kejadian inilah yang sering disebut dengan event. Pada saat kapan suatu objek/kontrol bekerja, tentunya pada saat kontrol/objek tersebut dikenai event.

Berikut ini beberapa event yang sering digunakan oleh pemakai program, yaitu :

- **Event untuk Mouse**
 - Click**
Event ini terjadi bila tombol kiri mouse ditekan dan dilepas dengan cepat pada saat posisi pointer berada di atas objek.
 - DbIClick**
Even ini terjadi bila tombol kiri ditekan dan dilepas dengan cepat sebanyak dua kali saat pointer berada di atas objek.
 - DragDrop**
Event ini terjadi bila tombol kiri mouse ditekan dan ditahan kemudian menggeser objek dari satu tempat ke tempat yang lain, kemudian dilepas tombol kiri mouse tersebut.

 - **Event untuk Keyboard**
 - KeyPress**
Event ini terjadi bila sebuah tombol keyboard ditekan. Nilai parameter dari KeyPress adalah kode ASCII untuk menyatakan tombol keyboard yang ditekan.
 - KeyDown**
Event ini terjadi bila anda menekan dan menahan sebuah tombol keyboard.
 - KeyUp**
Event ini terjadi bila anda melepaskan sebuah tombol keyboard.

 - **Event untuk perubahan**
 - Activate**
Event ini terjadi apabila sebuah form menjadi window yang aktif
-

Gotfocus

Event ini terjadi bila sebuah kontrol menjadi satu-satunya focus. Sebuah form dapat membuat objek fokus bila form tersebut menjadi window aktif.

LostFocus

Event ini terjadi bila sebuah kontrol kehilangan fokus karena ada objek lain yang mendapat fokus.

Load

Event ini terjadi bila sebuah form dibuka atau dipanggil oleh aplikasi.

Unload

Event ini terjadi bila sebuah form ditutup.

Resize

Event ini terjadi bila semua sebuah form diubah ukurannya.

- **Even-event lainnya**

Timer

Event ini terjadi ketika event lain dijalankan sebelum event timer ini. Event ini hanya berhubungan dengan kontrol timer.

Metode(Method)

Adalah suatu set perintah seperti halnya fungsi dan prosedur, tetapi sudah tersedia di dalam objek. Metode biasanya akan mengerjakan tugas khusus pada suatu objek.

Contoh metode :

```
Private sub Form_Activate()
```

```
.....
```

```
End Sub
```

Modul

Modul dapat disejajarkan dengan form, tetapi dalam modul tidak terdapat objek. Modul hanya berisi kode program atau prosedur yang dapat digunakan dalam program aplikasi. Modul ini disimpan dengan ekstensi .BAS.

Pada bagian ini akan dibahas beberapa kontrol standard yang sering dipakai beserta properti-properti, event-event dan method-method yang bekerja pada masing-masing kontrol. Dengan memahami property, event dan method akan sangat membantu dalam memanfaatkan kontrol-kontrol tersebut secara efektif dan efisien.

2. PEMAKAIAN FORM DAN KONTROL

Buatlah desain form seperti berikut :

1. Buat Project baru
2. Rancang form seperti berikut, kemudian simpan Form dengan nama **frmFormControl.frm** dan Project dengan nama **prjFormControl.frm**.



Gambar 2.1 Form frmFormControl

Setting Properties dari masing-masing kontrol sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	FrmFormControl :::Recording Manual Book
Label	Caption	Sesuaikan
Text1	Text Name	Dikosongkan TxtNewItem
Option1	Name Caption	OptInterMediate InterMediate
Option2	Name Caption	OptMedium Medium
Option3	Name Caption	OptAdvance Advance
Check1	Name Caption	ChkReferensi Referensi
Check2	Name Caption	ChkTutorial Tutorial
Check3	Name Caption	ChkLatihan Latihan
Combo1	Name	CmbProduk
List1	Name	LstData
Frame1	Caption	[Daftar Manual Book]
List1	Name	LstData
Command1	Name Caption	CmdTambah Tambah
Command2	Name Caption	CmdNewItem Item Baru

Command3	Name Caption	CmdHapus Hapus Item
Command4	Name Caption	CmdSelesai Selesai

3. Tulis kode program sebagai berikut :

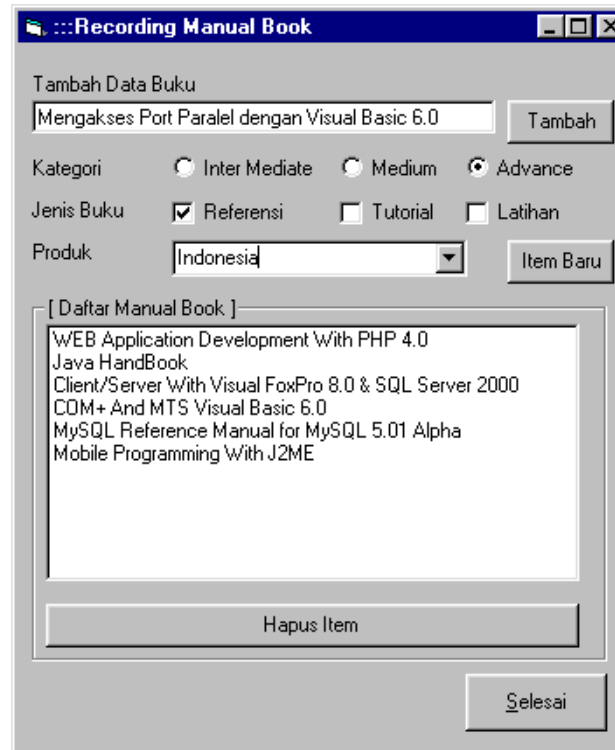
```
Option Explicit
-----
Private Sub Form_Load()

    With cmbProduk
        .AddItem "Indonesia"
        .AddItem "Luar Negeri"
        .ListIndex = 0
    End With
End Sub
-----
Private Sub cmdTambah_Click()
    If Len(txtNewItem.Text) = 0 Then
        MsgBox "Isikan Item yang akan ditambahkan", _
            vbOKOnly + vbInformation, "Konfirmasi"
    Else
        lstData.AddItem txtNewItem.Text
        cmdNewItem_Click
    End If
End Sub
-----
Private Sub cmdHapus_Click()
    Dim nomor_seleksi As Integer
    nomor_seleksi = lstData.ListIndex

    If nomor_seleksi < 0 Then
        MsgBox "Pilih Item yang akan dihapus", vbOKOnly + vbCritical,
            "Informasi"
        Exit Sub
    End If

    lstData.RemoveItem (nomor_seleksi)
End Sub
-----
Private Sub cmdNewItem_Click()
    txtNewItem.Text = ""
    optIntermediate.Value = False
    optMedium.Value = False
    optAdvance.Value = False
    chkReferensi.Value = 0
    chkTutorial.Value = 0
    chkLatihan.Value = 0
    cmbProduk.Text = ""
    txtNewItem.SetFocus
End Sub
-----
Private Sub cmdSelesai_Click()
    Unload Me
End Sub
```

4. Jalankan program sehingga diperoleh tampilan seperti berikut :



Gambar 2.2 Hasil program pada saat dijalankan

5. Keterangan :

- Ketika anda bekerja dengan form, hal yang perlu menjadi perhatian adalah bahwa form tersebut disimpan dengan nama file berekstensi **.frm**. file berekstensi **.frm** ini sebenarnya adalah file text yang berisi kode yang dipakai visual basic untuk membuat form. Kalau anda mempunyai program text editor seperti **Notepad**, anda dapat membuka file berekstensi **.frm** tersebut.


Kalau form yang anda buat diatas dibuka dengan **Notepad**, maka akan tampak kode program sebagai berikut :

```

frmFormControl - Notepad
File Edit Search Help
VERSION 5.00
Begin VB.Form Form1
    Caption = ":::Recording Manual Book"
    ClientHeight = 6060
    ClientLeft = 60
    ClientTop = 345
    ClientWidth = 5235
    LinkTopic = "Form1"
    ScaleHeight = 6060
    ScaleWidth = 5235
    StartupPosition = 2 'CenterScreen
    Begin VB.CommandButton cmdSelesai
        Caption = "&Selesai"
        Height = 495
        Left = 3840
        TabIndex = 17
        Top = 5400
        Width = 1215
    End
    Begin VB.CheckBox chkLatihan
        Caption = "Latihan"
        Height = 255
        Left = 3840
        TabIndex = 16
        Top = 1320
        Width = 975
    End
    Begin VB.CommandButton cmdNewItem
        Caption = "Item Baru"
        Height = 375
        Left = 4200
        TabIndex = 15
    End
End
    
```

Gambar 2.3 Isi dari file yang berekstensi .frm

- Beberapa properti yang sering dipergunakan oleh form adalah :

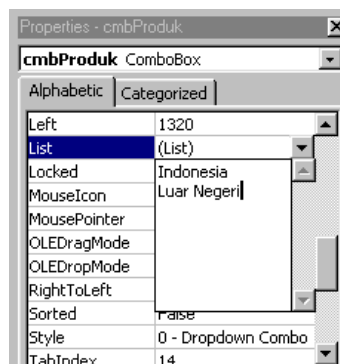
Properti	Fungsi
Name	Digunakan sebagai identifikasi nama dari form, pemberian nama dari form ini sangat penting, apalagi jika dalam project yang anda buat terdapat banyak form. Anda dapat memberikan nama form yang sama dengan nama file form yang berekstensi .frm . Ketika anda bekerja dengan form nantinya bukan file yang berekstensi .frm yang anda panggil, tetapi adalah properti name dari form.
BorderStyle	<p>0 – none , tidak ada border, title, icon, ataupun control box</p> <p>1 – fixed single, user tidak dapat mengubah ukuran form</p> <p>2 – sizeable, seperti default form</p> <p>3 – fixed dialog, tidak ada minimize,maximize button,icon dan tidak dapat diubah ukurannya oleh user. Property ini dipakai jika form yang anda buat dipergunakan sebagai dialog box.</p> <p>4 – Fixed ToolWindow, Tool window memiliki title bar, dan tidak ada minimize dan mazimize button.</p> <p>5 – sizeable ToolWindow, sama seperti fixed ToolWindow, tetapi form dapat diubah ukurannya.</p>
Icon, Caption, MinButton, MaxButton, ControlBox	<p>Jika Controlbox disetup False, maka akan menghapus property icon, minimize, maximize dan close button.</p> <p>Icon, caption, minimize button, maximize button, close button</p> 

Windowstate	Menentukan properti form pada saat ditampilkan 0 – normal sesuai ukuran form pada saat desain 1 - Minimized form akan ditampilkan minimize sebagai ikon pada statusbar 2 – Maximized form akan ditampilkan memenuhi layar
Startup Position	Menentukan property form pada layar monitor saat aplikasi dijalankan : 0 – manual , tampilan form pada layar monitor sesuai dengan posisi form yang tampak pada form layout window. Perubahan posisi form pada layar monitor bisa diubah dengan cara melakukan Drag and Drop pada objek yang ada di form layout window. 1 - CenterOwner , menempatkan form pada posisi tengah-tengah owner. 2. – Center Screen , menempatkan form pada tengah-tengah layar monitor 3 – Windows default , menempatkan form pada posisi yang tidak beraturan pada layar monitor sesuai dengan defaultnya windows.

- syntax **WITH ... END WITH** dipergunakan untuk menyeting properti dari suatu objek yang sama.
- Kontrol **Option** dan **Check** akan diinisialisasi sebagai True/1 jika dalam kondisi terpilih dan sebaliknya akan diinisialisasi sebagai False/0 jika dalam kondisi tidak terpilih.
- Untuk menambahkan item ke dalam **Listbox** atau **Combobox** dapat dilakukan dengan methode **Additem** diikuti String Item yang akan ditambahkan ataupun melalui properties Windows pada property **List**.

cmbProduk.Additem "Indonesia"
 cmbProduk.Additem "Luar Negeri"

Akan sama hasilnya dengan :



Gambar 2.3 Mengisi Properti dari Combobox

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 3	VARIABEL, TIPE DATA, KONSTANTA DAN OPERATOR

1. VARIABEL

Variabel adalah suatu tempat dalam memori yang digunakan untuk menampung data. Mendeklarasikan variabel dalam Visual Basic 6.0 tidak diwajibkan, tetapi sangat dianjurkan. Untuk mendeklarasikan variabel dipergunakan perintah :

Dim nama_variable [**As Type**]

Contoh : Dim jumlah As Integer
 Dim Kota As String

Jika **As Type** tidak disertakan maka tipe data dianggap variant.

2. TIPE DATA

Nilai yang disimpan dalam sebuah variabel akan memiliki nama dan tipe data tertentu. Berikut ini adalah tipe data pada Visual Basic 6.0

Tabel 3.1 Tipe Data Pada Visual Basic 6.0

Tipe Data	Jangkauan Nilai
Integer	-32.768 sampai 32.767
Long Integer	-2.147.483.648 sampai 2.147.483.647
Single	-3.402823E38 s/d 3.402823E38
Double	-1.7976931348232E308 s/d 1.79769313486232E308
Currency	- 922337203685477,5808 s/d - 922337203685477,5807
String	0 s/d 65.535
Boolean	True atau False
Date	1 januari 100 s/d 31 desember 9999
Byte	0 s/d 255
Object	Referensi Objek
Variant	Semua tipe data

Selain tipe data di atas, kita dapat membuat tipe data baru dengan mengambil salah satu atau beberapa tipe data standar yang telah ada. Visual Basic 6.0 menyediakan cara untuk membuat tipe data sendiri dengan menggunakan pernyataan '**Type**'. Tipe data ini biasa disebut sebagai UDT (*User Defined Type*). Cara menuliskannya sebagai berikut :

```
Type <nama_tipe_data>
    Nama_var1 [As tipe_data]
    Nama_var2 [As tipe_data]
    .....
    Nama_varN [As tipe_data]
End Type
```

Contoh : Type data_karyawan
 NIK As String * 5
 NAMA As String * 30
 ALAMAT As String * 20
 TGL_LAHIR As Date
 End Type

3. KONSTANTA

Konstanta adalah nama yang menyimpan suatu nilai yang tetap selama program dieksekusi (*running*). Agar konstanta yang dideklarasikan dapat dikenal di semua module, deklarasikan dengan menggunakan kata kunci **PUBLIC**. Cara menuliskannya :

Const nama_konstanta [As tipe_data] = ekspresi

Contoh: Const Diskon As Single = 0.1

4. ARRAY DAN KONTROL ARRAY

Array merupakan tipe terstruktur yang terdiri dari sejumlah komponen yang mempunyai tipe data yang sama, disinilah perbedaan Array dengan UDT. Banyaknya komponen dalam suatu Array ditunjukkan oleh suatu indeks. Setiap komponen di dalam Array dapat diakses dengan menunjukkan nilai indeks dari Array tersebut.

Contoh dari penggunaan Array : Dim bilangan(10) As Integer
Dim bilangan(1 to 10) As Integer

Kontrol Array adalah kontrol yang memiliki indeks seperti Array. Kontrol ini sangat membantu sekali pada saat kita membuat aplikasi, untuk menyederhanakan kode program.

5. OPERATOR

Operator adalah suatu tanda yang digunakan untuk menghubungkan satu variabel atau konstanta dengan variabel atau konstanta lain dengan tujuan melakukan berbagai manipulasi data dan pengolahan data. Berikut adalah beberapa operator yang ada dalam Visual Basic 6.0

Tabel 3.2 Tipe Data Pada Visual Basic 6.0

Operator	Operasi
^	Pemangkatan
-	Tanda negatif
*, /	Perkalian dan pembagian
\	Pembagian integer
Mod	Modulus (sisa hasil bagi)
+, -	Penambahan pengurangan

Tabel 3.3 operator perbandingan

Operator	Operasi
=	Sama dengan
<>	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil sama dengan
>=	Lebih besar sama dengan
Like	Mempunyai cirri yang sama
Is	Mempunyai referensi objek yang sama

Tabel 3.3 Operator Logika

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

6. PROGRAM REGISTRASI KURSUS

Program berikut memberikan contoh tentang bagaimana membuat aplikasi memanfaatkan tipe data, operator, konstanta, *User Defined Type* (UDT), dan penggunaan operator.

1. Buat Project baru
2. Rancang form seperti berikut, kemudian simpan Form dengan nama **frmRegKursus** dan Project dengan nama **prjRegKursus**.

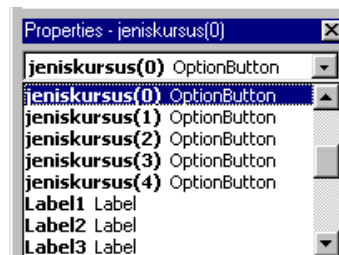
Gambar 3.1 Form frmRegKursus

Setting Properties dari masing-masing kontrol sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	frmRegKursus Kalkulator
Label1	Caption	Bilangan 1
Label2	Caption	Bilangan 2
Text1	Name	txtNama
Text2	Text Name	Dikosongkan txtTanggalReg
Combo1	Text	Dikosongkan
Option1	Name Caption (0) Caption (1) Caption (2) Caption (3) Caption (4)	jenisKursus Visual Basic 6.0 Visual FoxPro 8.0 Borland Delphi 7.0 Java Linux
Text3	Text Name	Dikosongkan TxtNamaKursus
Text4	Text Name	Dikosongkan TxtWaktuKursus
Text5	Text Name	Dikosongkan TxtBiayaKursus

Text6	Text Name	Dikosongkan TxtTotalPendaftaran
Text7	Text Name	Dikosongkan TxtTotalBiaya
Command1	Name Caption	CmdRekam Rekam
Command2	Name Caption	CmdReport Report
Command3	Name Caption	CmdSelesai Selesai

3. Ganti nama Option1 (Visual Basic 6.0) menjadi jenisKursus. Lakukan pengkopian terhadap kontrol option 2,3,4,5 jika ada pesan **You already have a control named 'jenisKursus'**. **Do you want to create control array ?** tekan Yes. Jendela properties akan nampak sebagai berikut :



Gambar 3.2 Kontrol Array bernama jenisKursus dengan 5 item(0-4)

4. Tulis kode program sebagai berikut :

```
Option Explicit
-----
' mendefinisikan variabel global
Dim waktukursus, biayakursus As Variant
Dim i As Byte
Dim totalBiaya As Single

'Mendefinisikan konstanta
Const diskon As Single = 0.1
Const biayadaftar As Single = 25000
-----
'mendefinisikan tipe data UDT
Private Type registrasi
    nama As String
    kelas As String
    pilihan As String
End Type
-----
'memakai tipe data UDT
Dim rec_registrasi As registrasi
-----
Sub setup_kelas()
With cmbKelas
    .AddItem "A"
    .AddItem "B"
    .AddItem "C"
    .AddItem "D"
    .ListIndex = 0
End With
End Sub
```

```
-----
Private Sub Form_Load()
'inisialisasi Array
waktukursus = Array("Senin 08.30-10.00 WIB", "Selasa 08.30-10.00 WIB", _
    "Rabu 10.00-11.30 WIB", "Kamis 10.00-11.30 WIB", "Rabu 10.00-
11.30 WIB")
biayakursus = Array(325000, 300000, 450000, 750000, 275000)
'nonaktifkan pilihan pada saat running pertama kali
For i = 0 To 4
    jeniskursus(i).Value = False
Next i
'isi tanggal dengan tanggal sekarang
txtTanggalReg.Text = Format(Date, "dd-mmm-yyyy")
'panggil prosedur tampilkan kelas
Call setup_kelas
'tampilkan konstanta pendaftaran
txtBiayaPendaftaran.Text = Format(biayadaftar, "###,###,###.00")
End Sub
-----

Private Sub jeniskursus_Click(Index As Integer)
    Dim biaya, administrasi As Long
    txtNamaKursus.Text = jeniskursus(Index).Caption
    txtWaktuKursus.Text = waktukursus(Index)
    txtBiayaKursus.Text = Format(CLng(biayakursus(Index)),
"###,###,###.00")
    biaya = CLng(biayakursus(Index))
    administrasi = CLng(txtBiayaPendaftaran.Text)
    totalBiaya = biaya + administrasi
    txtTotalBiaya = CStr(Format(totalBiaya, "###,###,###.00"))
End Sub
-----

Private Sub chkDiskon_Click()
    Dim selisihDiskon As Single
    If chkDiskon.Value = 1 Then
        selisihDiskon = totalBiaya * diskon
        txtTotalBiaya.Text = CStr(Format(totalBiaya - selisihDiskon,
"###,###,###.00"))
    Else
        txtTotalBiaya.Text = CStr(Format(totalBiaya, "###,###,###.00"))
    End If
End Sub
-----

Private Sub cmdRekam_Click()
    rec_registrasi.nama = txtNama.Text
    rec_registrasi.kelas = cmbKelas.Text
    rec_registrasi.pilihan = txtNamaKursus.Text
End Sub
-----

Private Sub cmdReport_Click()
    MsgBox "Nama : " & rec_registrasi.nama & vbCrLf _
        & "Kelas : " & rec_registrasi.kelas & vbCrLf _
        & "Pilihan Kursus : " & rec_registrasi.pilihan
End Sub
-----

Private Sub cmdSelesai_Click()
    Unload Me
End Sub
-----
```

5. Jalankan program sehingga diperoleh tampilan sebagai berikut :



Gambar 3.3 Hasil program pada saat dijalankan

6. Keterangan :

- Pada program diatas ditemui penggunaan fungsi-fungsi untuk mengkonversi tipe data dari suatu variabel.
 - CBool = untuk mengkonversi tipe data ke Boolean
 - CDate = untuk mengkonversi tipe data ke Date
 - CDBl = untuk mengkonversi tipe data ke Double
 - CCur = untuk mengkonversi tipe data ke Currency
 - CSng = untuk mengkonversi tipe data ke Single
 - CLng = untuk mengkonversi tipe data ke Long Integer
 - CStr = untuk mengkonversi tipe data ke string
- Fungsi format() dipergunakan untuk menampilkan variabel dalam format tertentu.

Bentuk pemakaian fungsi format adalah **format**(ekspresi,format)

Contoh :

format(Date,"w") mengembalikan nilai minggu(1=Sunday,7=Saturday)

format(Date,"yyyy") mengembalikan nilai tahun dalam 4 digit

format(Date,"dd-mm-yyyy") mengembalikan nilai tanggal-bulan-tahun misal : 25-05-2005

Tombol **Rekam** dipergunakan untuk menyimpan dahulu variabel ke dalam tipe data UDT, untuk kemudian ditampilkan melalui tombol **Report**. Jika tombol rekam belum ditekan, maka UDT belum akan terisi data yang terbaru.

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 4	- FUNGSI DAN PROSEDUR - DECISIONS (IF...THEN, SELECT...CASE) - ERROR TRAPPING

1. PROSEDUR DAN FUNGSI

Program yang besar biasanya dipecah menjadi beberapa prosedur dan atau fungsi untuk menyederhanakan logika. Prosedur dan fungsi ini sangat bermanfaat untuk mengelompokkan kode-kode program yang seringkali digunakan atau dipakai bersama-sama di dalam program, seperti kalkulasi, operasi basis data dan lain sebagainya. Dengan demikian proses penelusuran kesalahan (Debugging) dapat dikonsentrasikan pada prosedur atau fungsi yang telah dibuat.

Perbedaan utama prosedur dan fungsi adalah bahwa prosedur tidak mengembalikan nilai sedangkan function mengembalikan nilai.

PROSEDUR

Bentuk umum prosedur : **[Private|Public] [Static] Sub** nama_prosedur(arguments)
statement
End Sub

Setiap kali prosedur dipanggil, maka pernyataan-pernyataan yang berada diantara sub dan End Sub akan dijalankan. Arguments pada prosedur adalah nilai yang akan dilewatkan pada saat pemanggilan prosedur.

Pada visual Basic sub prosedur dapat dibagi atas dua, yaitu :

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

Contoh :

Sub tengah yang dapat digunakan untuk menampilkan form ke tengah layar, dimana x adalah parameter yang merupakan form yang akan dibuat ke tengah 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

FUNGSI

Pada Visual Basic telah tersedia berbagai fungsi bawaan seperti Len,Chr,Str, Val dan lain-lain, tetapi fungsi-fungsi yang tersedia tersebut bersifat umum dan kadang-kadang tidak bisa memenuhi kebutuhan programmer pada saat membuat aplikasi. Untuk keperluan tersebut anda dapat menciptakan fungsi-fungsi sendiri.

Bentuk umum function : **[Private|Public] [Static] Function** nama_fungsi(arguments) **[As type]**
statement
End Function

2. PERCABANGAN

Percabangan digunakan untuk mengatur alur program berdasarkan kondisi (logika) tertentu. Dengan demikian alur program tidak harus berurutan, tetapi dapat melompat-lompat.

STRUKTUR PERCABANGAN IF..THEN

Merupakan struktur percabangan dimana ekspresi akan dikerjakan bila kondisinya terpenuhi. Tetapi jika kondisinya tidak terpenuhi, maka ekspresi di dalam struktur IF tidak akan dijalankan dan blok IF akan dilompati serta program akan melakukan tindakan berikutnya.

Bentuk umum : **if** ekspresi **Then**
[statement – 1]
[Elseif ekspresi-n **Then]**
[statement – n] ...
[Else]
[statement – else]
End if

SELECT...CASE

Struktur SELECT...CASE digunakan untuk menjalankan satu blok perintah yang jumlahnya banyak atau bertingkat-tingkat. Struktur ini lebih sederhana bila dibandingkan dengan struktur IF...THEN.

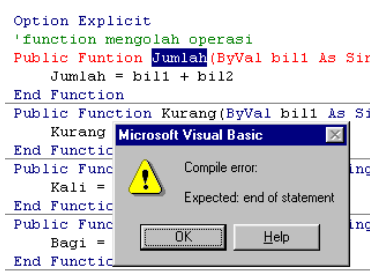
Bentuk umum : **Select Case** ekspresi
[Case ekspresi uji-n]
[statement-n] ...
[Case Else] ...
[statement else]
End Select

3. Error Trapping

Adakalanya kita perlu menangani kesalahan karena berbagai macam hal. Suatu kesalahan seringkali menyebabkan program menampilkan pesan kesalahan yang akan membingungkan pemakai. Jadi, alangkah baik kalau kesalahan-kesalahan yang terjadi tidak ditampilkan atau mungkin kita dapat membuat deskripsi-deskripsi kesalahan yang terjadi akibat kesalahan user, atau bisa jadi kita mengantisipasi kesalahan-kesalahan yang kemungkinan akan terjadi pada saat program dijalankan.

Pada dasarnya, kesalahan itu dapat dikelompokkan menjadi tiga jenis, yaitu :

- 1) Kesalahan Kaidah/syntax Error
Biasanya karena adanya kesalahan dalam penulisan syntax atau aturan perintah.
Contoh :



Gambar 4.1 Pesan Kesalahan Syntax Error

Kesalahan yang terjadi adalah karena keyword **Funtion** yang seharusnya **Function**.

- 2) Kesalahan Runtime Error
Kesalahan yang terjadi pada saat program sedang berjalan atau *run-time*. Kesalahan ini karena biasanya program tidak menemukan object yang sedang dijalankan. Dalam hal seperti ini anda dapat mendebug program yang saat itu sedang berjalan, baru kemudian akan nampak baris mana yang salah, sehingga anda mudah untuk memperbaikinya.
- 3) Kesalahan Logika (Logic Error)
Kesalahan logika adalah jenis kesalahan yang sulit untuk ditelusuri dan dicari penyebabnya. Kesalahan seperti ini adakalanya tidak menimbulkan pesan kesalahan, tetapi memberikan hasil yang keliru.

4. Program penggunaan fungsi dan prosedur serta Error Trapping

Pada program berikut akan diberikan contoh penggunaan prosedur dan fungsi untuk melakukan operasi aritmatika, dimana prosedur dan fungsi ditempatkan pada sebuah module. Sehingga fungsi dan prosedur tersebut dapat dipergunakan pada semua form.

1. buat project baru dan desain form seperti berikut ini :



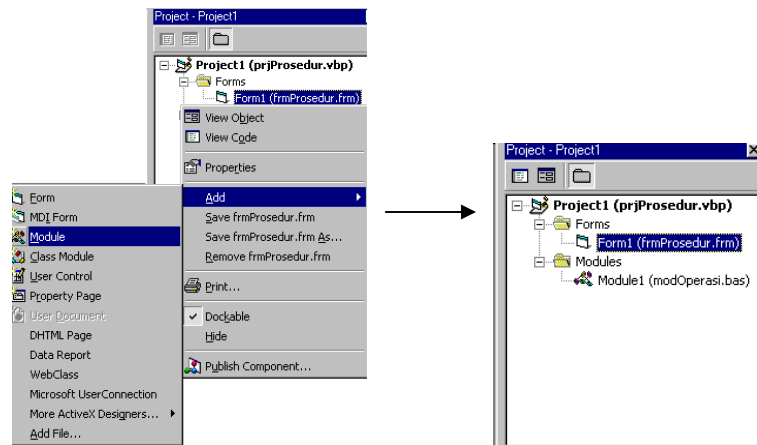
Gambar 4.2 Form frmProsedur

2. kemudian simpan project dengan nama **frmProsedur** dan Project dengan nama **prjProsedur**.
3. setting propertiesnya seperti berikut :

Setting Properties dari masing-masing kontrol sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	frmProsedur Kalkulator
Frame1	Caption	Perincian Jenis Kursus
Frame2	Caption	Perincian Biaya Kursus
Text1	Text Name	Dikosongkan TxtBil1
Text2	Text Name	Dikosongkan TxtBil2
Command1	Name Caption (0) Caption (1) Caption (2) Caption (3)	cmdOpertor + - * /
Text3	Text Name	Dikosongkan TxtHasil
Command2	Name Caption	CmdUlangi &Ulangi
Command3	Name Caption	CmdSelesai &Selesai

4. Kemudian tambahkan 1 buah module, dan simpan dengan nama **modOperasi**, kemudian tuliskan kode berikut dalam module :



Gambar 4.3 Menambahkan module kedalam project

```

Option Explicit

'function mengolah operasi
Public Function Jumlah(ByVal bill As Single, ByVal bil2 As Single) As Single
    Jumlah = bill + bil2
End Function

-----

Public Function Kurang(ByVal bill As Single, ByVal bil2 As Single) As Single
    Kurang = bill - bil2
End Function

-----

Public Function Kali(ByVal bill As Single, ByVal bil2 As Single) As Single
    Kali = bill * bil2
End Function

-----

Public Function Bagi(ByVal bill As Single, ByVal bil2 As Single) As Single
    Bagi = bill / bil2
End Function

-----

'function verifikasi entry
Public Function cek_entry(teks1 As TextBox, teks2 As TextBox) As Boolean
    If Len(teks1.Text) = 0 Then
        MsgBox "Mohon Bilangan 1 diisi", vbOKOnly + vbInformation, "Konfirmasi"
        teks1.SetFocus
        cek_entry = False
    ElseIf Len(teks2.Text) = 0 Then
        MsgBox "Mohon Bilangan 2 diisi", vbOKOnly + vbInformation, "Konfirmasi"
        teks2.SetFocus
        cek_entry = False
    ElseIf IsNumeric(teks1.Text) = False Then
        MsgBox "Bilangan 1 bukan Numeric", vbOKOnly + vbInformation, "Konfirmasi"
        teks1.SetFocus
        cek_entry = False
    ElseIf IsNumeric(teks2.Text) = False Then
        MsgBox "Bilangan 2 bukan Numeric", vbOKOnly + vbInformation, "Konfirmasi"
        teks2.SetFocus
        cek_entry = False
    Else
        cek_entry = True
    End If
End Function
    
```

5. kemudian lanjutkan dengan menulis kode program untuk frmProsedur sebagai berikut :

```
Option Explicit
-----
Sub kosong_entry()
    txtBill1.Text = ""
    txtBil2.Text = ""
    txtHasil.Text = ""
End Sub
-----
Private Sub cmdOperator_Click(Index As Integer)
On Error GoTo Salah
Select Case Index
    Case 0
        If cek_entry(txtBill1, txtBil2) = True Then
            txtHasil.Text = CStr(Jumlah(Val(txtBill1.Text), Val(txtBil2.Text)))
        End If
    Case 1
        If cek_entry(txtBill1, txtBil2) = True Then
            txtHasil.Text = CStr(Kurang(Val(txtBill1.Text), Val(txtBil2.Text)))
        End If
    Case 2
        If cek_entry(txtBill1, txtBil2) = True Then
            txtHasil.Text = CStr(Kali(Val(txtBill1.Text), Val(txtBil2.Text)))
        End If
    Case 3
        If cek_entry(txtBill1, txtBil2) = True Then
            txtHasil.Text = CStr(Bagi(Val(txtBill1.Text), Val(txtBil2.Text)))
        End If
End Select

Exit Sub
Salah:
MsgBox "Terjadi Kesalahan : " _
    & vbCrLf & "Error Number : " & Err.Number _
    & vbCrLf & "Error Description : " & Err.Description, vbOKOnly + vbInformation,
"INFO"
End Sub

-----
Private Sub cmdUlangi_Click()
    Call kosong_entry
End Sub
-----
Private Sub cmdSelesai_Click()
Unload Me
End Sub
```

6. Keterangan :

- fungsi **IsNumeric**(ekspresi) dipergunakan untuk mendeteksi apakah suatu variabel/ekspresi merupakan nilai Numeric. Disamping fungsi **IsNumeric**(), masih terdapat fungsi-fungsi yang lain untuk menguji suatu ekspresi atau variabel, yaitu :
 - **IsDate**(ekspresi) – menguji apakah ekspresi berupa tipe data tanggal.
 - **IsEmpty**(variabel) - menguji apakah suatu variabel telah terinisialisasi.
 - **IsArray**(variabel) – menguji apakah sebuah variabel merupakan tipe data Array.
 - **IsNull**(ekspresi) – Menghasilkan nilai True, jika ekspresi adalah NULL.
 - **IsError**(ekspresi) – menunjukkan bahwa suatu ekspresi adalah nilai Error
 - **IsObject**(ekspresi) – menunjukkan bahwa suatu ekspresi mengacu pada object

Biasanya fungsi-fungsi tersebut dipergunakan untuk mendeteksi kemungkinan terjadinya **Error** dalam program yang kita buat.

- Object **Err** adalah system, yang dihasilkan VB 6.0 jika sebuah **Error** terjadi. Untuk menangani **Error** (*Error-Handling*), dimana jika terjadi suatu **Error**, maka alur program akan meloncat ke baris yang ditentukan. Juga bisa digunakan untuk menon-aktif penanganan **Error**.

Bentuk-bentuk Error-handling :

Bentuk	Keterangan
On Error GoTo line	Jika ada Error, melompat ke label line
On Error Resume Next	Jika ada Error, eksekusi dilanjutkan ke baris program selanjutnya
On Error GoTo 0	Menon-aktifkan penanganan Error.

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 5	PENGGUNAAN .OCX REPETITIONS (FOR...NEXT,DO...LOOP)

1. FILE .OCX

ActiveX Control merupakan komponen ekstensi dari komponen-komponen yang ada di dalam **Toolbox** Visual Basic 6.0, Artinya kita dapat menambah komponen-komponen yang tidak dimiliki Visual Basic 6.0 sesuai dengan keinginan kita. File-file ActiveX Control ini berekstensi **.OCX**, yang nantinya akan kita gunakan untuk membangun Aplikasi dengan VB. Jika kita masuk ke Visual Basic 6.0 pada mode Standard .EXE, maka komponen-komponen **Toolbox** hanya berisi komponen-komponen Standard Visual Basic 6.0. Kita dapat menambahkan komponen .ActiveX Control ke dalam project kita dengan mudah.

2. PERULANGAN

Perulangan dipergunakan untuk mengulang suatu blok program atau untuk mempersingkat perulangan kode program.

STRUKTUR FOR ... NEXT

Mengulang sekelompok statement sebanyak n-kali.

Bentuk umum : **FOR** counter = start **To** End [**Step** step]
 [statements]
 [**Exit For**]
 [statements]
 NEXT [counter]

Contoh :

```
For bil_1 = 1 to 9  
  Print bil_1;  
Next bil_1
```

Akan menghasilkan output : **1 2 3 4 5 6 7 8 9**

```
For bil_2 = 9 to 1 step -1  
  Print bil_2;  
Next bil_2
```

Akan menghasilkan output : **9 8 7 6 5 4 3 2 1**

STRUKTUR DO ... LOOP

Mengulang sekelompok statement selama kondisi yang ditentukan True atau hingga True.

Bentuk-bentuk :

Do While condition [statements] [Exit Do] Loop
--

Do Until condition [statements] [Exit Do] Loop
--

Do [statements] [Exit Do] Loop While condition
--

Do [statements] [Exit Do] Loop Until condition
--

Selain bentuk strukturnya Do While ... Loop dan Do ... Loop While, terdapat bentuk lain yang fungsinya sama juga yaitu **While – Wend**. Struktur While . .. Wend ini sama persis dengan struktur Do While ... Loop, hanya bentuk penulisannya saja yang berbeda.

Bentuk penulisan While ... Wend, sebagai berikut :

```
While (statements)  
    statements  
Wend
```

Contoh :

```
bil = 1  
Do While bil <= 9  
    Print bil;  
    bil = bil + 1  
Loop
```

Akan menghasilkan output : **1 2 3 4 5 6 7 8 9**

```
bil = 1  
Do  
    Print bil;  
    bil = bil + 1  
Loop While bil <= 9
```

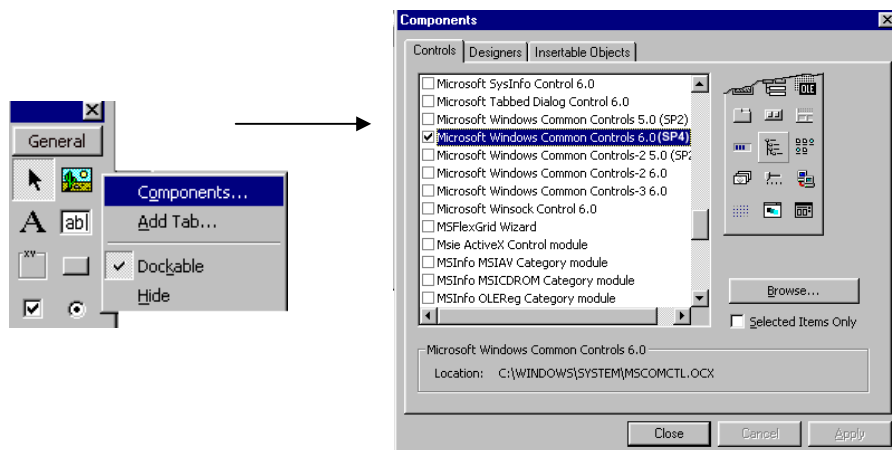
Akan menghasilkan output : **1 2 3 4 5 6 7 8 9**

```
bil = 1  
While bil <= 9  
    Print bil;  
    bil = bil + 1  
Wend
```

Akan menghasilkan output : **1 2 3 4 5 6 7 8 9**

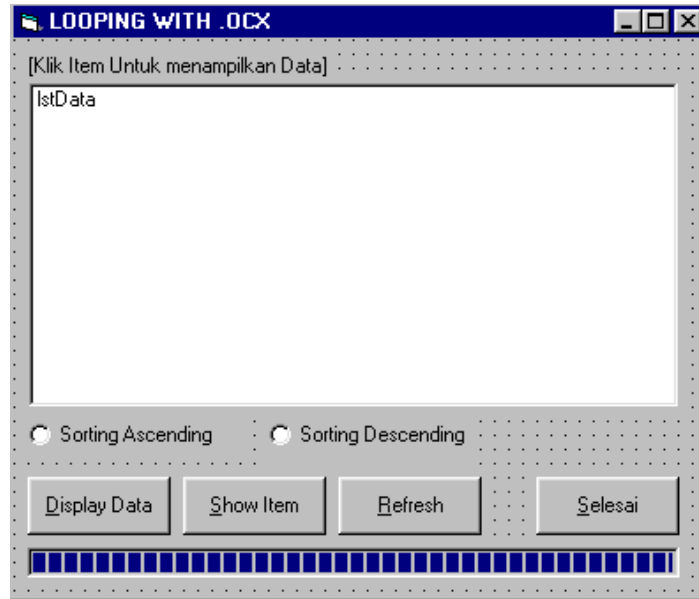
3. Program menggunakan file .OCX

1. Buat project baru.
2. Tambahkan komponen **Microsoft Windows Common Controls 6.0 (SP4)** ke project yang baru saja kita buat.



Gambar 5.1 Menambahkan komponen .OCX

3. Selanjutnya desain form sebagai berikut :



Gambar 5.2 Form frmLoopOcx

4. kemudian simpan project dengan nama **frmLoopOcx** dan Project dengan nama **prjLoopOcx**.
 5. setting propertiesnya seperti berikut :

Setting Properties dari masing-masing kontrol sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	frmLoopingOcx Kalkulator
Label1	Caption	[Klik Item Untuk menampilkan Data]
ListView1	Name	LstData
Option1	Name Caption	OptAscending Sorting Ascending
Option2	Name caption	OptDescending Sorting Descending
Command1	Name Caption	CmdDisplayData &Display Data
Command2	Name Caption	CmdShowItem &Show Item
Command3	Name Caption	CmdRefresh &Refresh
Command4	Name Caption	CmdSelesai &Selesai
Progressbar1	Name	PgbBar

6. Kemudian tuliskan kode berikut dalam module :

```
Option Explicit
Dim i As Integer
Dim nomor(1 To 9) As Integer
Dim nama(1 To 9) As String
Dim klik As Boolean
Dim nomor_tampil, nama_tampil As String
Dim panbar(3000) As String
```

```
-----  
'prosedur menampilkan data  
Sub list_data()  
  
    For i = 1 To 9  
        nomor(i) = i  
        nama(i) = "Nama " & i  
    Next i  
  
    With lstData  
        .ColumnHeaders.Clear  
        .ListItems.Clear  
        .View = lvwReport  
        .LabelEdit = lvwManual  
        .GridLines = True  
        .FullRowSelect = True  
        .ListItems.Clear  
        .ColumnHeaders.Add 1, , "No. Reg"  
        .ColumnHeaders.Add 2, , "Nama"  
        .ColumnHeaders(1).Width = 1000  
        .ColumnHeaders(2).Width = 4400  
        For i = LBound(nomor) To UBound(nomor)  
            .ListItems.Add 1, , nomor(i)  
            .ListItems(1).SubItems(1) = nama(i)  
        Next i  
    End With  
End Sub  
  
-----  
'fungsi mengecek ada tidaknya data dalam list  
Function cek_list() As Boolean  
    If lstData.ListItems.Count < 1 Then  
        cek_list = False  
    Else  
        cek_list = True  
    End If  
End Function  
  
-----  
'prosedur showprogress  
Sub showProgress()  
    pgbBar.Min = LBound(panbar)  
    pgbBar.Max = UBound(panbar)  
    pgbBar.Value = pgbBar.Min  
    pgbBar.Visible = True  
    For i = LBound(panbar) To UBound(panbar)  
        pgbBar.Value = i  
    Next i  
    pgbBar.Value = pgbBar.Min  
    pgbBar.Visible = False  
End Sub  
  
-----  
'prosedur menyimpan variabel yang dipilih  
Private Sub lstData_ItemClick(ByVal Item As MSComctlLib.ListItem)  
    klik = True  
    nomor_tampil = Item.Text  
    nama_tampil = Item.SubItems(1)  
End Sub  
  
-----  
Private Sub optAscending_Click()  
    If cek_list = True Then  
        lstData.SortOrder = lvwAscending  
        lstData.Sorted = True  
    Else  
        MsgBox "Belum Ada data yang disortir", vbOKOnly + vbInformation,  
"Konfirmasi"  
        optAscending.Value = False  
    End If  
End Sub
```

```
-----
Private Sub optDescending_Click()
    If cek_list = True Then
        lstData.SortOrder = lvwDescending
        lstData.Sorted = True
    Else
        MsgBox "Belum Ada data yang disortir", vbOKOnly + vbInformation,
"Konfirmasi"
        optDescending.Value = False
    End If
End Sub
-----

Private Sub cmdDisplayData_Click()
    'memanggil prosedur
    Call list_data
    'memanggil progressbar
    Call showProgress
End Sub
-----

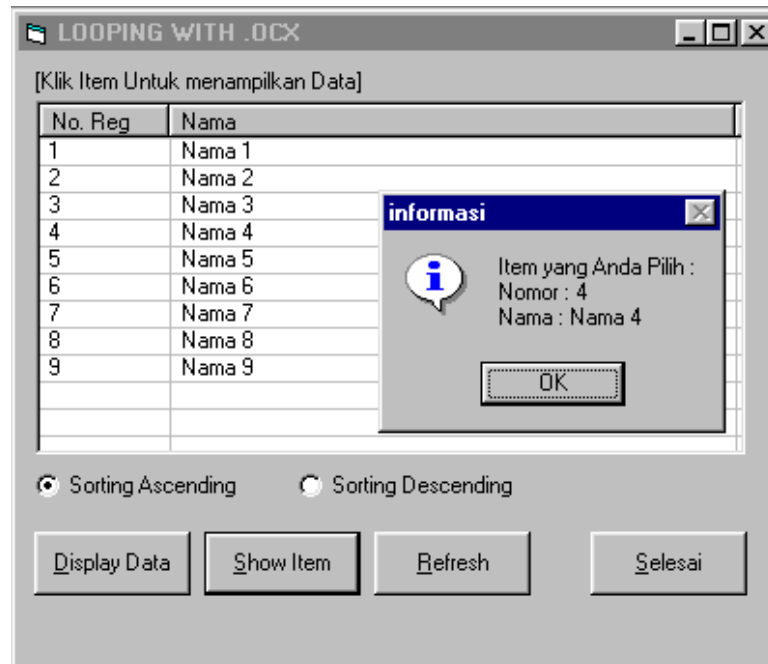
Private Sub cmdShowItem_Click()
    If klik = True Then
        MsgBox "Item yang Anda Pilih : " _
            & vbCrLf & "Nomor : " & nomor_tampil _
            & vbCrLf & "Nama : " & nama_tampil, vbOKOnly + vbInformation,
"informasi"
        klik = False
    Else
        MsgBox "Anda Belum Memilih Item Untuk ditampilkan", vbOKOnly +
vbInformation, "informasi"
        klik = False
    End If
End Sub

-----

Private Sub cmdRefresh_Click()
    'memanggil prosedur
    lstData.ListItems.Clear
End Sub
-----

Private Sub cmdSelesai_Click()
    Unload Me
End Sub
```


7. Jalankan program sehingga diperoleh tampilan sebagai berikut :



Gambar 5.3 Running Form frmLoopOcx

8. Keterangan :

- fungsi **LBound**(var_Array) dipergunakan untuk mengetahui indeks terkecil dari suatu variabel Array. sedangkan fungsi **UBound**(var_Array) digunakan untuk mengetahui indeks terbesar dari varaiabel Array.

<p style="text-align: center;">JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG</p>		
<p>MICROSOFT VISUAL BASIC 6.0</p>	<p>MODUL 6</p>	<p>MENGAKSES DATABASE MS. ACCESS MENGUNAKAN ADO DATA CONTROL</p>

1. ADO DATA CONTROL

Microsot ActiveX Data Object.(ADO) Data Control merupakan komponen yang dipergunakan untuk melakukan koneksi dengan database. Cara yang lain adalah dengan menggunakan **Reference Library** atau menggunakan **DataEnvironment**.

Dengan menggunakan **ADO Data Control** anda cukup menyeting properti dari ADO Data Control tersebut untuk melakukan koneksi dengan database. Namun hal yang terlebih dahulu harus dilakukan adalah membangun dan merancang database.

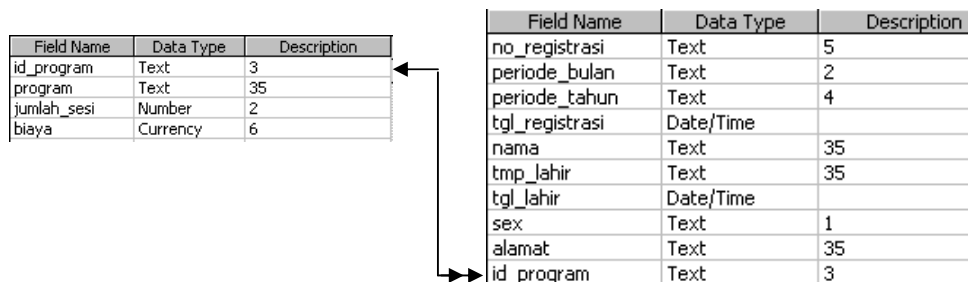
2. MS. ACCESS

MS. Acces merupakan produk Microsoft untuk membangun database. Versi MS. Access yang akan digunakan dalam aplikasi ini adalah MS. Access 2000. Dengan menggunakan Access 2000 kita dapat merancang, membuat dan mengelola database dengan sangat mudah.

3. Membangun Aplikasi dengan ADO DATA CONTROL

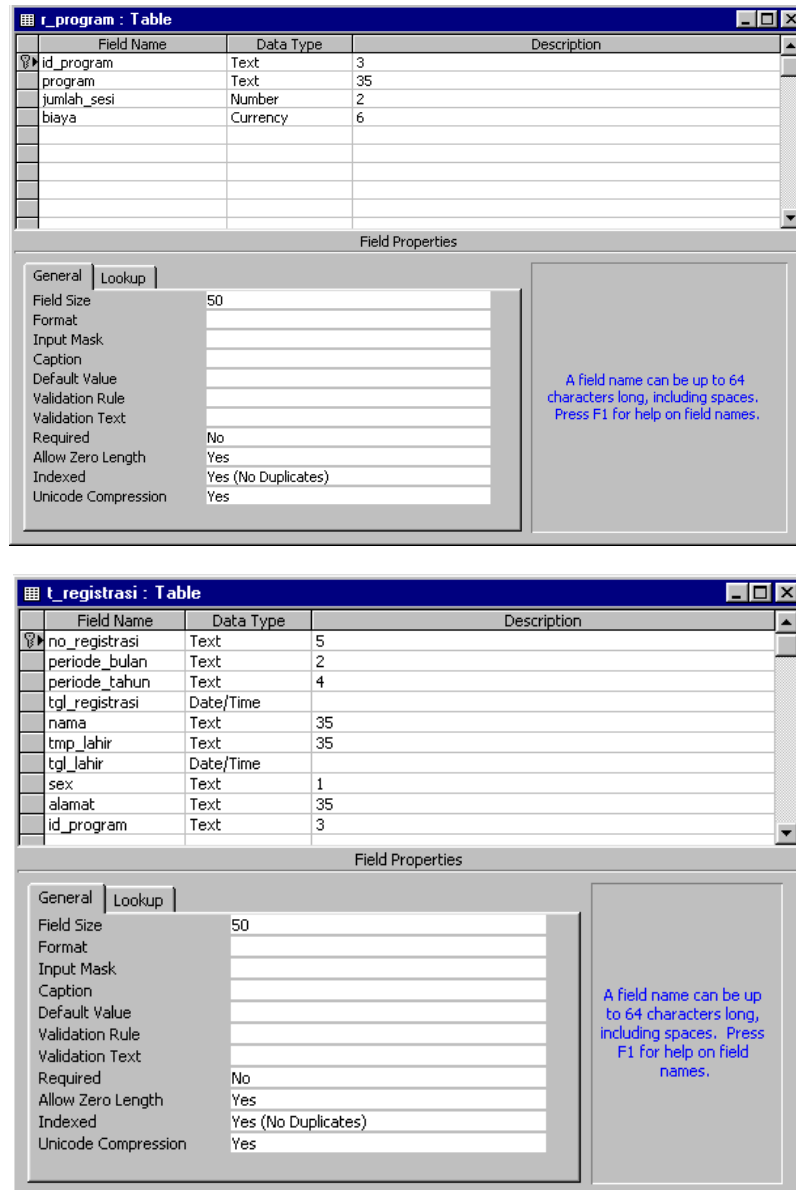
Untuk membangun file database yang akan digunakan sebagai sumber data aplikasi dengan Visual Basic 6.0, dapat menggunakan Microsoft Access, ataupun program Visual Data Manager yang sudah *add-ins* di dalam program Visual Basic 6.0. Sebagai ilustrasi, akan dibuat sebuah program yang akan mengelola data registrasi.

Buatlah table-table di bawah ini dan relasikan sesuai dengan field-field yang berhubungan.



Untuk merancang dan membuat tabel baru , ikuti langkah berikut :

1. Buka Microsoft Access, pilih **Blank Access database**, kemudian jika muncul menu dialog **file New Database**, simpan dengan nama **dbcourse.mdb**, simpan di direktori **c:\data\dbcourse.mdb**.
2. Setelah itu tampil jendela kerja database dengan pilihan obyek tables tersebut, klik dua kali **Create table in Design View**, jendela tabel design akan ditampilkan.
3. pada kotak dialog tersebut definisikan struktur tabel dengan cara mengisi nama field (*Field Name*), jenis data (*Data Type*) dan keterangan (*Description*) bila ada. Pada saat mendefinisikan *Field Name*, anda menentukan *field Name* mana yang akan dijadikan *Primary Key*. Anda cukup mengklik toolbar dengan gambar kunci untuk mendefinisikan *Primary Key*.



Gambar 6.1 desain tabel

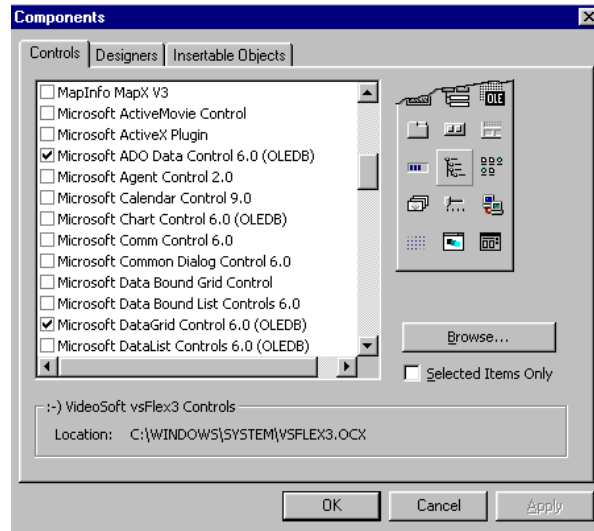
4. Isikan data dari masing-masing tabel seperti berikut ini :

id_program	program	jumlah_sesi	biaya
DLP	Borland Delphi 7,0	12	Rp350,000
JAV	JAVA	15	Rp450,000
LNK	LINUX	12	Rp500,000
VB6	MS. Visual Basic 6.0	12	Rp350,000
VFP	MS. Visual FoxPro 6,0	12	Rp275,000

no_registrasi	periode bulan	periode tahun	tgl registrasi	nama	tmp_lahir	tgl_lahir	sex	alamat	id_program
05021	02	2005	1/30/05	Helmalia Putri	Jakarta	10/26/82	P	Menteng no. 18	JAV
05022	02	2005	1/25/05	Jeckie Chan	Hongkong	12/8/82	L	Jl. Chiong Ping no. 15	JAV
05023	02	2005	1/26/05	Anisa Kristin	Yogyakarta	10/5/75	L	Pengok 28 A	JAV
05031	03	2005	2/2/05	Ikhwani Arisandi	Bantul	2/6/85	P	Perum Polri no. 25	VB6
05032	03	2005	2/13/05	Cakra Kelana	Sleman	3/8/82	P	Purwomartani 3A	VB6

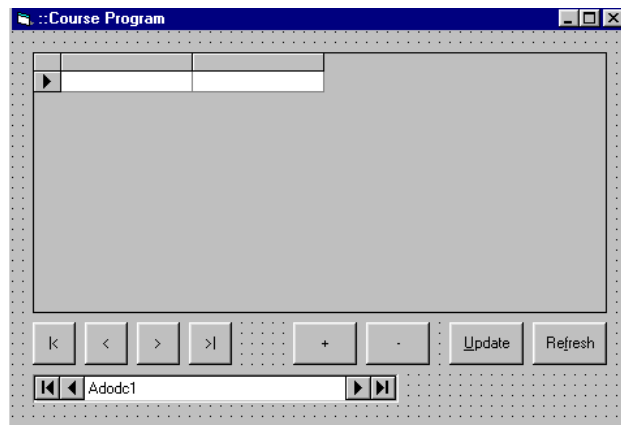
Gambar 6.2 isi tabel

5. Sampai di sini kita sudah cukup untuk bekerja dengan database MS. Access, langkah selanjutnya adalah mendesain interface dengan Visual Basic 6.0
6. Buka Visual Basic 6.0, Untuk menggunakan komponen ADO Data kontrol terlebih dahulu tambahkan komponen tersebut ke dalam project, dan tambahkan juga komponen Data Grid Control untuk menampilkan data dari database.



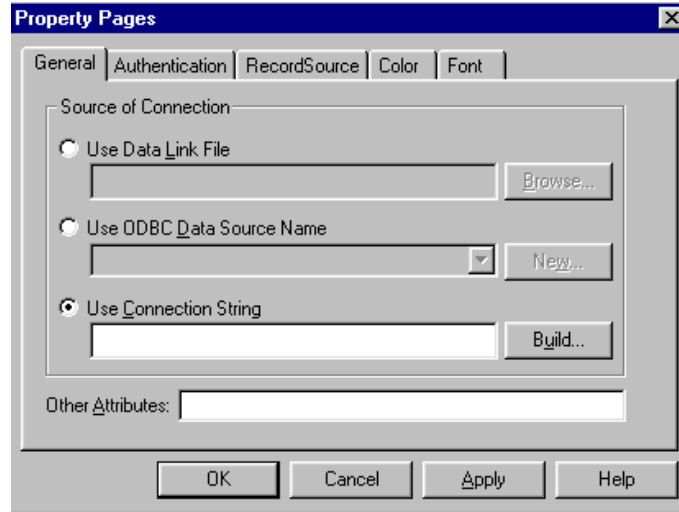
Gambar 6.3 Jendela untuk menambahkan komponen

7. Selanjutnya desain form sebagai berikut, dan simpan form dengan nama frmADODC.frm dan project dengan nama prjADODC.vbp.



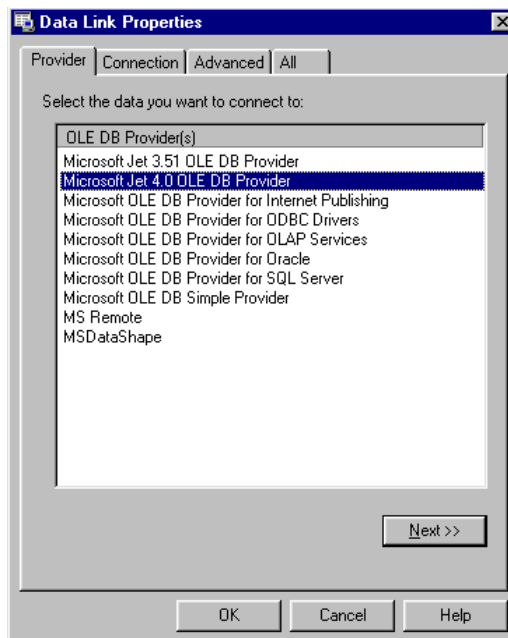
Gambar 6.4 Form frm

8. Untuk membuat koneksi dengan database, ikuti langkah-langkah berikut :
 1. Pilih dan klik kanan pada object ADODC yang ada di form, maka akan ditampilkan sederetan menu pop-up, pilih dan klik dengan tombol kiri mouse **ADODC Properties**. Kotak dialog Property Pages akan ditampilkan.
-



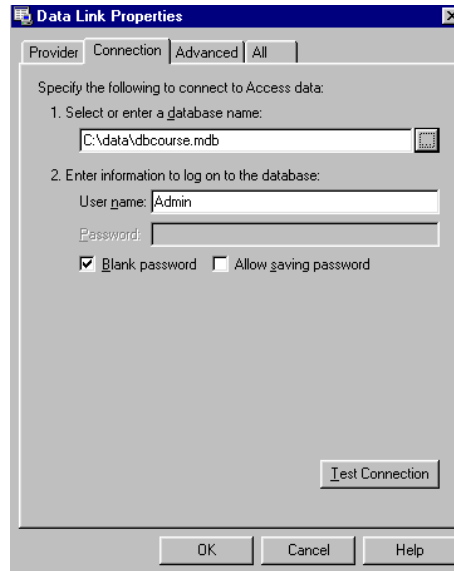
Gambar 6.5 Jendela Property pages

2. Klik tombol **Build**, maka kotak dialog Data Link Properties akan ditampilkan. Kotak dialog ini berisi daftar Provider koneksi. Pilih "Microsoft Jet 4.0 OLE DB Provider".

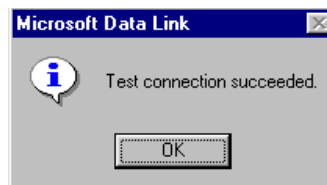


Gambar 6.6 Jendela Property pages

3. Selanjutnya klik tombol **Next >>** atau klik tombol tab **Connection**. Pada **Select Or Enter a Database Name**, anda cukup mengklik tombol (...), maka kotak dialog **Select Access Database** akan ditampilkan. Pada kotak ini anda arahkan di mana posisi database disimpan. Dalam contoh ini database yang dipakai, yaitu **dbcourse.mdb** yang terletak di direktori **C:\data\dbcourse.mdb**. Setelah anda memilih database, klik tombol **Open**.
 4. Selanjutnya untuk memastikan apakah database sudah terkoneksi atau belum, klik tombol **Test Connection**. Jika koneksi berhasil akan ditampilkan pesan sukses.
-

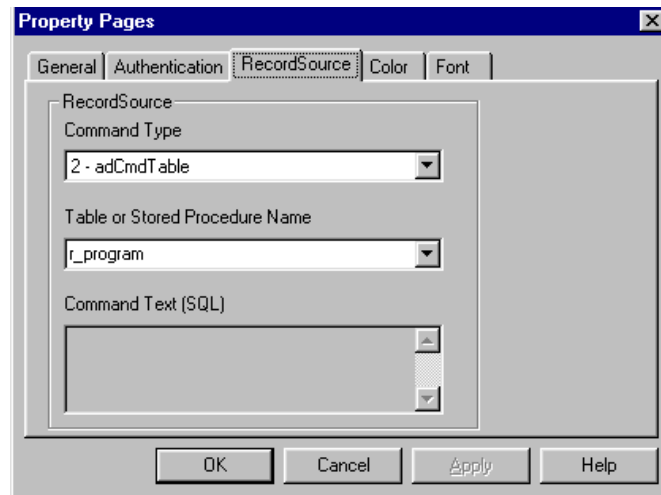


Gambar 6.7 Jendela Data Link untuk memilih sumber database



Gambar 6.8 MessageBox jika koneksi dengan Database telah sukses

5. Selanjutnya klik tombol OK, kemudian klik tombol OK satu kali lagi untuk menuju Property pages untuk menentukan **RecordSource** yang dipakai dari database **dbcourse.mdb**.
6. Klik Tab **RecordSource**, pada Command Type pilih 2-adCmdTable, dan pada Table or Stored Procedure Name pilih **r_program**, sebagai table sumber untuk kontrol ADO.



Gambar 6.9 Jendela RecordSource untuk menentukan tabel yang digunakan.

7. Sampai disini anda telah berhasil membuat koneksi dengan database menggunakan ADO Data Control.

9. Setting properties dari masing-masing kontrol yang ada dalam form sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	FrmADODC :::Course Program
DataGrid1	Name DataSource	DataGrid1 Adodc1
Command1	Name Caption(0) Caption(1) Caption(2) Caption(3) Caption(4) Caption(5) Caption(6) Caption(7)	CmdManipulasi < < > > + - &Update Re&fresh
Command2	Name Caption	CmdSelesai &Exit
Adodc1	Name	Adodc1

10. Kemudian tuliskan kode program sebagai berikut :

```

Option Explicit

Private Sub cmdManipulasi_Click(Index As Integer)
On Error GoTo salah
    Select Case Index
        Case 0
            Adodc1.Recordset.MoveFirst
        Case 1
            Adodc1.Recordset.MovePrevious
        Case 2
            Adodc1.Recordset.MoveNext
        Case 3
            Adodc1.Recordset.MoveLast
        Case 4
            Adodc1.Recordset.AddNew
        Case 5
            If MsgBox("Apakah Record ini akan dihapus ? ", vbYesNo + vbQuestion,
"Konfirmasi") = vbYes Then
                Adodc1.Recordset.Delete
            End If
        Case 6
            Adodc1.Recordset.Update
        Case 7
            Adodc1.Refresh
    End Select
Exit Sub
salah:
MsgBox "Pesan Kesalahan : " & vbCrLf & Err.Description
End Sub

Private Sub cmdSelesai_Click()
    Unload Me
End Sub

```

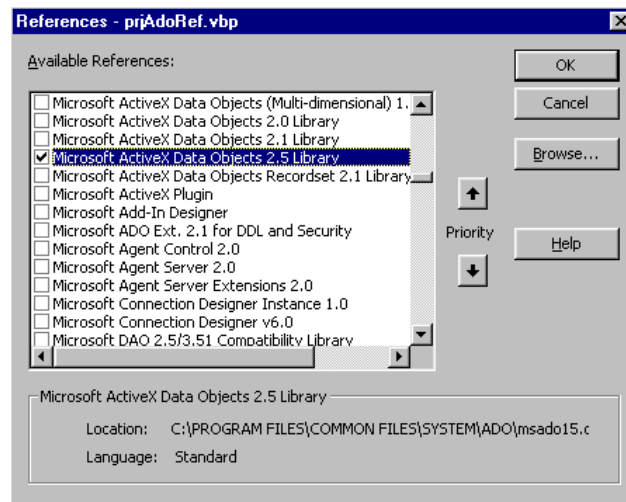
11. Keterangan :

- Adodc memiliki method **Addnew** untuk menambah data, **Update** untuk menyimpan perubahan dalam record secara permanen dan **Delete** untuk menghapus record.
- Untuk melakukan navigasi terhadap record, dipergunakan methode sebagai berikut :
 - **MoveFirst** : untuk pindah ke record pertama
 - **MovePrevious** : untuk berpindah ke record sebelumnya
 - **MoveNext** : untuk berpindah ke record selanjutnya
 - **MoveLast** : untuk berpindah ke record terakhir
 - **Refresh** :dipergunakan untuk menyegarkan data yang ada dalam tabel

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 7	MENGGUNAKAN ADO LIBRARY & SQL

1. MENGAITKAN ADO LIBRARY DENGAN VISUAL BASIC 6.0

Agar Visual Basic dapat menggunakan library ADO, maka library ADO harus terdaftar terlebih dahulu pada registry windows dan memberi tanda cek pada library tersebut. Untuk melakukannya buka menu **Project > References**.



Gambar 7.1 Mengaitkan Visual Basic dengan ADO Library

Setelah library ADO dicek, maka library tersebut sudah dapat digunakan secara default, biasanya library ADO sudah terinstall pada saat Visual Basic 6.0 diinstall. Untuk distribusi ADO library dapat diperoleh dari Instalasi Microsoft Data Access Componen (MDAC) versi 2.x, sehingga tidak perlu menginstall VB.

Object ADO yang sering dipergunakan untuk membuat aplikasi database adalah :

- **Connection**
Dipergunakan untuk membuat koneksi dengan database, property penting yang dipergunakan adalah :
 - Open : metode open bertanggung jawab untuk membuka koneksi dengan database.
 - Execute : metode execute dipergunakan untuk menjalankan perintah query, pernyataan SQL, atau prosedur tersimpan.
 - State : property ini dipergunakan untuk mengetahui apakah object connection dalam kondisi terbuka (adStateOpen) atau tidak.
- **Command**
Dipergunakan untuk mengirim perintah ke database, umumnya perintah yang dikirim tersebut berupa **Stored Procedure** dan perintah SQL.

- **Recordset**

Objek recordset dipergunakan untuk menampung hasil eksekusi dalam bentuk table. Obyek recordset memiliki beberapa property dan metode yang penting sbb :

1. property **cursorType** : menentukan arah pergerakan ketika recordset dibaca, nilainya :
 - a. **AdOpenDynamic** : jenis kursor ini memungkinkan segala perubahan yang dilakukan pada record(insert,update,delete) dapat dilihat user lain.
 - b. **AdOpenKeyset** : mirip seperti AdOpenDynamic, hanya saja penambahan record baru yang dilakukan oleh user lain tidak dapat diketahui
 - c. **AdOpenStatic** : jenis kursor ini dapat dibaca ke segala arah.
 - d. **AdOpenForwardOnly** : arah pembacaan record hanya bergerak maju saja
2. property **LockType** : menangani aktifitas penguncian record, nilainya :
 - a. **AdLockReadOnly** : record hanya bisa dibaca.
 - b. **AdLockPessimistic** : mengunci sebuah record selama proses perubahan dan tidak mengijinkan pengguna lain untuk merubah record pada saat yang sama.
 - c. **AdLockOptimistic** : tipe ini banyak dipergunakan dalam pemrograman, karena record hanya akan dikunci sesaat saja ketika perubahan akan dilakukan sehingga memberi kesempatan user lain untuk merubah record.
 - d. **AdLockBatchOptimistic** : dipergunakan untuk pemrograman recordset terputus.

2. MEMBUAT KONEKSI

Hal yang pertama kali harus dilakukan setelah mengaitkan Visual Basic dengan ADO Library adalah membuat koneksi.

Untuk melakukan koneksi antara Visual Basic dengan database digunakan object **Connection** dari ADO.

Property dan method yang penting pada obyek ini yaitu property **ConnectionString** dan metode **Open**.

Property **ConnectionString** adalah sebuah string yang terdiri dari pasangan parameter dan nilai parameter yang dipisahkan oleh titik koma (;). Sedangkan method **Open** digunakan untuk membuka koneksi sesuai dengan **ConnectionString**-nya.

Untuk database MS.ACCESS 2000, ConnectionStringnya adalah :

```
ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" _  
    & "Persist Security Info=false;" _  
    & "Data Source = c:\data\dbcourse.mdb;"
```

Untuk membuat koneksi, anda dapat membuat dan kemudian menyimpannya dalam sebuah module khusus, sehingga object **Connection** tersebut dapat dipanggil oleh modul-modul/form yang lain.

3. Menggunakan SQL (Structured Query Language)

Bahasa SQL adalah sarana dasar yang dibutuhkan untuk mengakses data di dalam database relasional. Menguasai bahasa SQL adalah langkah pertama dan terpenting dalam memahami database relasional.

Untuk memanipulasi data, bahasa SQL yang dipergunakan adalah DML (*Data Manipulation Language*). Perintah-perintah ini bertanggung jawab untuk melakukan query dan perubahan yang dilakukan di dalam tabel.

Perintah DML meliputi :

- **INSERT**

Pernyataan **INSERT** dipergunakan untuk menambahkan baris baru pada sebuah tabel.

Contoh : **INSERT INTO r_program(id_program,program,jumlah_sesi,biaya)
VALUES('VFP','ORACLE 9i',15,400000)**

- **UPDATE**

Pernyataan **UPDATE** dipergunakan untuk memperbaharui data yang ada dalam sebuah tabel.

Contoh : **UPDATE r_program
SET program ='MS. Viual FoxPro 8.0'
WHERE id_program='VFP'**

- **DELETE**

Pernyataan **DELETE** dipergunakan untuk menghapus sebuah baris yang ada di dalam tabel.

Contoh : **DELETE FROM r_program
WHERE id_program='VFP'**

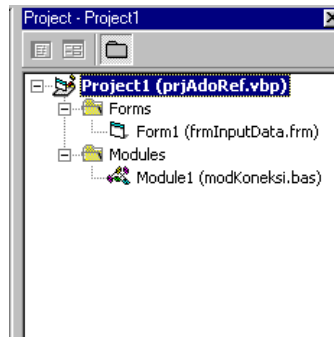
- **SELECT**

Pernyataan **SELECT** dipergunakan untuk melakukan pengambilan data dari sebuah tabel.

Contoh : **SELECT * FROM r_program**

4. MEMBUAT APLIKASI DENGAN ADO LIBRARY DAN SQL

2. Tambahkan sebuah module untuk menyimpan string koneksi dari object **Connection**, sehingga dapat dipergunakan oleh modul/form lain, sehingga cukup sekali saja mendefinisikan **Connection**.



Gambar 7.2 Menambahkan Module

Tuis kode program untuk module tersebut :

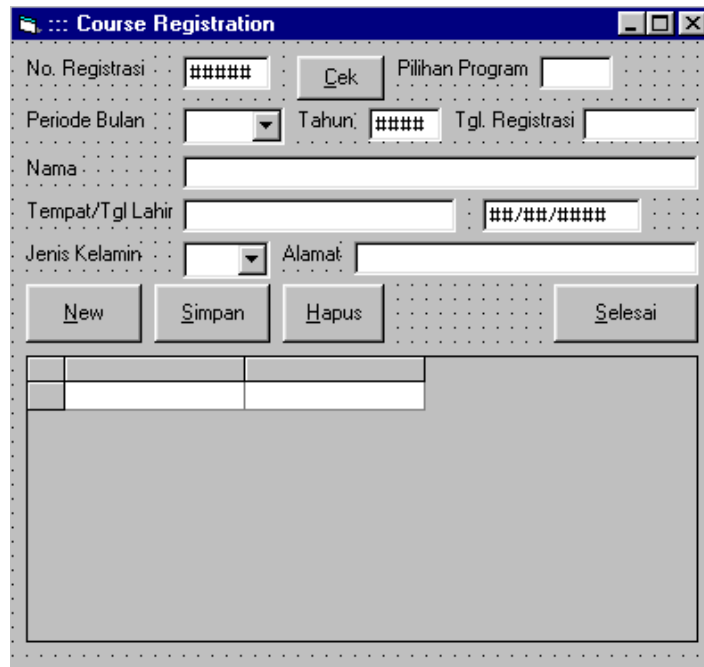
```
`modKoneksi.bas
`-----

Option Explicit
Public koneksi As New ADODB.Connection
'-----

Public Sub buka_koneksi()
On Error GoTo error_handel
koneksi.CursorLocation = adUseClient
koneksi.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" _
    & "Persist Security Info=false;" _
    & "Data Source = c:\data\dbcourse.mdb;"
koneksi.Open
If Not koneksi.State = 1 Then
    MsgBox "koneksi KE SERVER GAGAL", vbCritical + vbOKOnly, "Konfirmasi"
End
End If
Exit Sub
error_handel:
MsgBox "Gagal Koneksi Ke Server ...." & Chr(13) _
    & "Silahkan Menghubungi Administrator" & Chr(13) _
    & "Laporkan Komentar Berikut : " & Chr(13) & Chr(13) _
    & Err.Description, vbOKOnly + vbInformation, "Konfirmasi"
End Sub
Public Sub tutup_koneksi()
On Error GoTo salah

If koneksi.State = adStateOpen Then
    koneksi.Close
    Set koneksi = Nothing
End If
Exit Sub
salah:
MsgBox "Ada Kesalahan : " & vbCrLf _
    & "Silahkan Menghubungi Administrator" & Chr(13) _
    & "Laporkan Komentar Berikut : " & Chr(13) & Chr(13) _
    & Err.Description, vbOKOnly + vbInformation, "Konfirmasi"
End Sub
```

3. Desain form seperti berikut .:



Gambar 7.3 Desain Form

4. Setting properties dari masing-masing kontrol yang ada dalam form sebagai berikut :

Komponen	Properties	Nilai
MaskedTextBox1	Name Mask	mskNoreg #####
Text1	Name	TxtProgram
Combo1	Name	CmbBulan
MaskedTextBox2	Name Mask	MskTahun ####
Text2	Name	TxtTglReg
Text3	Name	TxtNam
Text4	Name	TxtTmpLahir
MaksEditBox3	Name Mask	MskTglLahir ##/##/####
Combo2	Name	CmbSex
Text5	Name	TxtAlamat
Command1	Name Caption	CmdCek &cek
Command2	Name Caption	CmdNew &New
Command3	Name Caption	CmdSimpan &Simpan
Command4	Name Caption	CmdHapus &Hapus
Command1	Name Caption	CmdSelesai &Selesai
DataGrid1	Name	DataGrid1

5. Kemudian tuliskan kode program sebagai berikut :

```
Option Explicit
Dim rsregistrasi As New ADODB.Recordset
-----
'prosedur menampilkan data
Sub browse_data()
    If rsregistrasi.State = adStateOpen Then rsregistrasi.Close
    rsregistrasi.Open "t_registrasi", koneksi, adOpenStatic, adLockOptimistic
    Set DataGrid1.DataSource = rsregistrasi
End Sub
-----
'fungsi untuk mengecek data
Function cek_data() As Boolean
    If rsregistrasi.State = adStateOpen Then rsregistrasi.Close
    rsregistrasi.Open "SELECT * FROM t_registrasi WHERE no_registrasi='" &
mskNoreg.Text & "'", koneksi, adOpenStatic, adLockOptimistic
    If rsregistrasi.RecordCount > 0 Then
        cek_data = True
    Else
        cek_data = False
    End If
End Function
-----
Private Sub Form_Load()
    If Not koneksi.State = adStateOpen Then
        buka_koneksi
    End If
    browse_data
    txtTglReg.Text = Format(Date, ("dd/mm/yyyy"))
End Sub
-----
Private Sub Form_Unload(Cancel As Integer)
    If koneksi.State = adStateOpen Then
        tutup_koneksi
    End If
End Sub

If rsregistrasi.State = adStateOpen Then
    rsregistrasi.Close
    Set rsregistrasi = Nothing
End If
End Sub
-----
Private Sub cmdCek_Click()
    If cek_data() = True Then
        MsgBox "Data Tersebut sudah ada", vbOKOnly + vbInformation, "Konfirmasi"
    End If
    browse_data
End Sub
-----
Private Sub cmdNew_Click()
    mskNoreg.Mask = "#####"
    mskNoreg.Mask = "#####"
    txtProgram.Text = ""
    cmbBulan.Text = ""
    mskTahun.Mask = "#####"
    mskTahun.Mask = "#####"
    txtNama.Text = ""
    txtTmpLahir.Text = ""
    mskTglLahir.Mask = "###/###/####"
    mskTglLahir.Mask = "###/###/####"
    cmbSex.Text = ""
    txtAlamat.Text = ""
    mskNoreg.SetFocus
End Sub
-----
Private Sub cmdSimpan_Click()
    On Error GoTo salah
    If cek_data() = True Then
        MsgBox "No. Registrasi Telah Terdaftar, Cek Ulang", vbOKOnly +
vbInformation, "Konfirmasi"
    Else
```

```

        koneksi.Execute "INSERT INTO
t_registrasi(no_registrasi,periode_bulan,periode_tahun," _
        & "tgl_registrasi,nama,tmp_lahir,tgl_lahir,sex,alamat,id_program)" _
        & "VALUES('" & mskNoreg.Text & "','" & cmbBulan.Text & "','" &
mskTahun.Text _
        & "','" & CDate(txtTglReg.Text) & "','" & txtNama.Text _
        & "','" & txtTmpLahir.Text & "','" & CDate(mskTglLahir.Text) _
        & "','" & cmbSex.Text & "','" & txtAlamat.Text & "','" &
txtProgram.Text & "')"
        End If
        browse_data
Exit Sub
salah:
MsgBox "Cek Inputan" & vbCrLf _
        & "Mungkin ada data yang salah atau belum terisi"

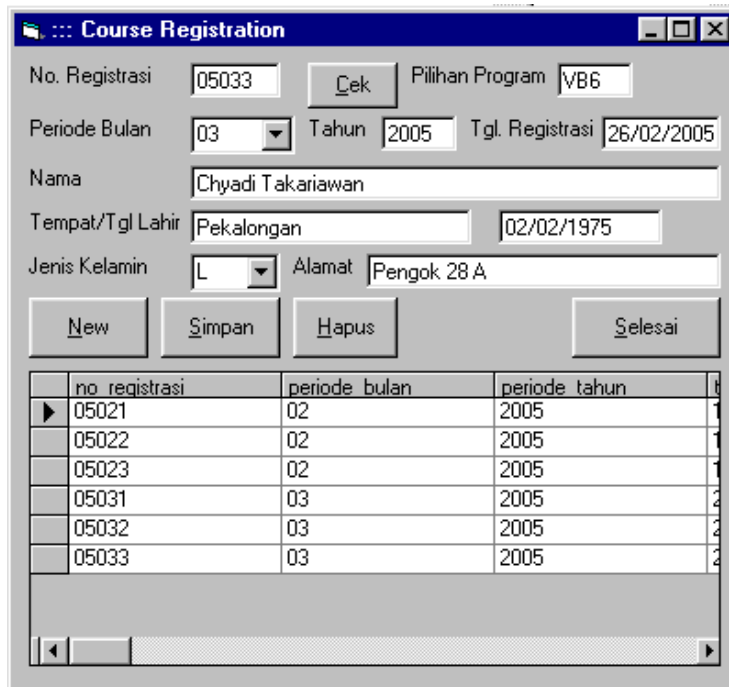
End Sub

-----
Private Sub cmdHapus_Click()
    If cek_data() = True Then
        If MsgBox("Apakah Data Akan dihapus ? ", vbYesNo + vbQuestion, "DELETE
RECORD") = vbYes Then
            koneksi.Execute "DELETE FROM t_registrasi WHERE no_registrasi='" &
mskNoreg.Text & "'"
            End If
        End If
        browse_data
    End Sub

-----
Private Sub cmdSelesai_Click()
    Unload Me
End Sub

```

6. Jalankan program sehingga diperoleh tampilan sebagai berikut :



Gambar 7.4 Hasil Running Program

7. Keterangan :

- Untuk menciptakan sebuah objek koneksi dan recorset adalah sebagai berikut :

```
Dim koneksi as ADODB.Connection
Set koneksi = New ADODB.Connection
```

```
Dim rsregistrasi As ADODB.Recordset
Set rsregistrasi = New ADODB.Recordset
```

Atau

```
Dim koneksi As New ADODB.Recordset
Dim rsregistrasi As New ADODB.Recordset
```

Setelah seluruh objek Connection dan Recordset dipakai dalam aplikasi, maka objek-objek tersebut harus segera dibebaskan dari memori, caranya adalah sebagai berikut :

```
Koneksi.close
Set koneksi = Nothing
```

```
Rsregistrasi.close
Set rsregistrasi = Nothing
```

- Untuk membuat koneksi dengan database, dibuat sebuah prosedur yang akan menangani koneksi, dan jika terjadi kesalahan, maka akan segera ditampilkan pesan kesalahan, tanpa harus keluar dari program.

```
Public Sub buka_koneksi()
On Error GoTo error_handle

koneksi.CursorLocation = adUseClient
koneksi.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" _
    & "Persist Security Info=false;" _
    & "Data Source = c:\data\dbcourse.mdb;"

koneksi.Open
If Not koneksi.State = 1 Then
    MsgBox "koneksi KE SERVER GAGAL", vbCritical + vbOKOnly, "Konfirmasi"
End
End If

Exit Sub
error_handel:
MsgBox "Gagal Koneksi Ke Server ..." & Chr(13) _
    & "Silahkan Menghubungi Administrator" & Chr(13) _
    & "Laporkan Komentar Berikut : " & Chr(13) & Chr(13) _
    & Err.Description, vbOKOnly + vbInformation, "Konfirmasi"

End Sub
```

Jika koneksi yang diciptakan lewat **ConnectionString** berhasil, maka objek **Connection** akan mengembalikan nilai 1 atau konstanta **AdStateOpen**. Hal ini ditandai dengan property state dari objek **Connection**.

- Setiap kali sebuah objek akan dibuka, maka terlebih dahulu objek tersebut dalam kondisi tertutup, jika masih terbuka maka harus ditutup dahulu, seperti objek recordset harus ditutup terlebih dahulu sebelum dibuka.

```
If rsregistrasi.State = adStateOpen Then rsregistrasi.Close
rsregistrasi.Open "SELECT * FROM t_registrasi WHERE no_registrasi=" _
    & mskNoreg.Text & "'", koneksi, adOpenStatic, adLockOptimistic
```


- Untuk menampilkan seluruh data yang ada dalam tabel, kemudian menampilkan datanya, dibuat sebuah prosedur sebagai berikut :

```
Sub browse_data()  
  If rsregistrasi.State = adStateOpen Then rsregistrasi.Close  
  rsregistrasi.Open "t_registrasi", koneksi, adOpenStatic, adLockOptimistic  
  Set DataGrid1.DataSource = rsregistrasi  
End Sub
```

- Dalam program di atas dibuat sebuah fungsi untuk mendeteksi apakah sebuah record dengan nomor tertentu sudah ada dalam tabel. Fungsi tersebut akan mengembalikan nilai True jika data yang dimaksud sudah ada dalam table, demikian juga sebaliknya. Fungsi tersebut adalah :

```
Function cek_data() As Boolean  
  If rsregistrasi.State = adStateOpen Then rsregistrasi.Close  
  rsregistrasi.Open "SELECT * FROM t_registrasi WHERE no_registrasi='" & _  
    & mskNoreg.Text & "'", koneksi, adOpenStatic, adLockOptimistic  
  If rsregistrasi.RecordCount > 0 Then  
    cek_data = True  
  Else  
    cek_data = False  
  End If  
End Function
```

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 8	PASSING VARIABEL ANTAR FORM

1. BEKERJA DENGAN VARIABEL PUBLIC

Seringkali ketika membangun aplikasi yang kompleks dengan Visual Basic 6.0, kita akan sering bekerja dengan menggunakan banyak form. Hal ini dikarenakan agar aplikasi yang kita bangun tersebut mudah untuk dioperasikan dan akan lebih memudahkan user.

Pada saat kita bekerja dengan banyak form tersebut, kita akan sering bekerja dengan menggunakan variabel yang didefinisikan secara public. Bisa juga nantinya akan ditemui banyak sekali prosedur ataupun fungsi yang kita ciptakan secara public. Hal ini dimaksudkan agar perpindahan data dan variabel serta fungsi dan prosedur dapat dikenali oleh semua form yang kita gunakan untuk aplikasi.

Biasanya untuk mendefinisikan variabel, fungsi dan prosedur secara public, kita akan menempatkannya pada sebuah modul Visual Basic yang berekstensi .BAS, atau dapat juga kita definisikan secara public pada form tertentu, sehingga kita akan mudah untuk mengaksesnya, dengan hanya cukup sekali saja mendefinisikannya.

Perbedaan ketika kita mendefinisikan variabel secara public di form dan di modul adalah pada bagaimana cara kita mengaksesnya. Ketika variabel public kita definisikan pada modul (.BAS) kita cukup memanggilnya sesuai dengan nama variabel tersebut, tetapi jika pendefinisian variabel public tersebut pada form, maka cara untuk mengaksesnya adalah dengan menyebutkan nama dari form tersebut, diikuti dengan nama variabelnya.

Contoh : `frmCourseProgram.pldProgram`

Nama form variabel public

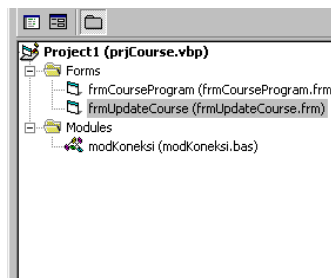
2. MEMBANGUN PROGRAM PASSING VARIABEL ANTAR FORM

Dalam aplikasi yang dibuat ini nantinya akan diperkenalkan bagaimana kita membangun sebuah aplikasi yang akan bekerja dengan lebih dari satu buah form untuk mengirimkan variabel antar form.

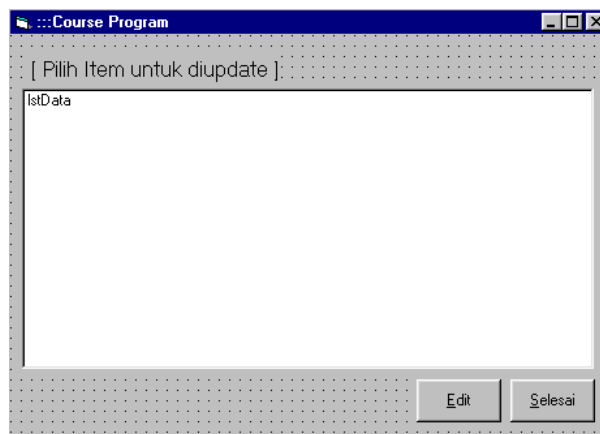
Pendefinisian variabel dilakukan pada form pertama, dan kemudian dikirimkan pada form kedua untuk diproses.

Untuk keperluan tersebut kita akan membutuhkan 2 buah form dan 1 buah modul serta tabel `r_program` yang ada pada database `dbcourse`, yang sudah pernah dibuat pada modul sebelumnya.

1. Untuk keperluan tersebut buatlah desain form dan modul, serta beri nama sebagai berikut :



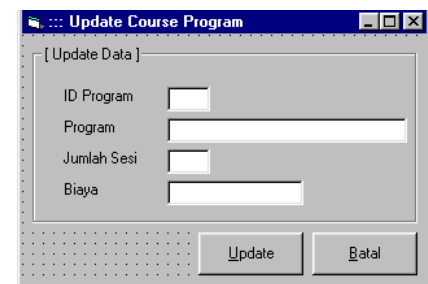
Gambar 8.1 Komponen form dan modul



Gambar 8.2 frmCourseProgram

Komponen	Property	Nilai
Form1	Name	frmCourseProgram
	Caption	::: Course Program
Label1	Caption	[Pilih Item utnuk diUpdate]
ListView1	Name	LstData
Command1	Name	CmdEdit
	Caption	&Edit
Command2	Name	CmdSelesai
	caption	&Selesai

Komponen	Property	Nilai
Form1	Name	frmUpdateCourse
	Caption	::: Course Program
Shape1	Caption	[Update Update]
Text1	Name	txtIdProgram
Text2	Name	txtProgram
Text3	Name	txtJumlahSesi
Text4	Name	txtBiaya
Command1	Name	CmdUpdate
	Caption	&Update
Command2	Name	CmdBatal
	Caption	&Batal



Gambar 8.3 frmUpdateCourse

2. Selanjutnya tambahkan sebuah modul dan simpan dengan nama modKoneksi, jika anda sudah membuat modul koneksi tersebut pada modul sebelumnya, anda cukup mengkopinya saja, karena isinya sama persis dengan modul sebelumnya.

Namun jangan lupa untuk menambahkan ADO Reference.

Isi modul tersebut adalah :

```
Option Explicit
Public koneksi As New ADODB.Connection
'-----

Public Sub buka_koneksi()
On Error GoTo error_handel
koneksi.CursorLocation = adUseClient
koneksi.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" _
    & "Persist Security Info=false;" _
    & "Data Source = c:\data\dbcourse.mdb;"
koneksi.Open
If Not koneksi.State = 1 Then
    MsgBox "koneksi KE SERVER GAGAL", vbCritical + vbOKOnly, "Konfirmasi"
End If
Exit Sub
error_handel:
MsgBox "Gagal Koneksi Ke Server ...." & Chr(13) _
    & "Silahkan Menghubungi Administrator" & Chr(13) _
    & "Laporkan Komentar Berikut : " & Chr(13) & Chr(13) _
    & Err.Description, vbOKOnly + vbInformation, "Konfirmasi"
End Sub
'-----

Public Sub tutup_koneksi()
On Error GoTo salah
If koneksi.State = adStateOpen Then
    koneksi.Close
    Set koneksi = Nothing
End If
Exit Sub
salah:
MsgBox "Ada Kesalahan : " & vbCrLf _
    & "Silahkan Menghubungi Administrator" & Chr(13) _
    & "Laporkan Komentar Berikut : " & Chr(13) & Chr(13) _
    & Err.Description, vbOKOnly + vbInformation, "Konfirmasi"
End Sub
```

3. Selanjutnya untuk form frmCourseProgram anda dapat menuliskan kodenya sebagai berikut :

```
Option Explicit
Dim rsprogram As New ADODB.Recordset
Public pIdProgram As String
Public pProgram As String
Public pJumlahSesi As String
Public pBiaya As Currency
```

```
-----  
Sub tampil_data()  
    If rsprogram.State = adStateOpen Then rsprogram.Close  
    rsprogram.Open "SELECT * FROM r_program", koneksi, adOpenStatic,  
adLockOptimistic  
    With lstData  
        .View = lvwReport  
        .LabelEdit = lvwManual  
        .GridLines = True  
        .FullRowSelect = True  
        .ListItems.Clear  
        .ColumnHeaders.Clear  
        .ColumnHeaders.Add 1, , "ID Program"  
        .ColumnHeaders.Add 2, , "Program"  
        .ColumnHeaders.Add 3, , "Jumlah Sesi"  
        .ColumnHeaders.Add 4, , "Biaya"  
        .ColumnHeaders(1).Width = 1000  
        .ColumnHeaders(2).Width = 3000  
        .ColumnHeaders(3).Width = 1000  
        .ColumnHeaders(4).Width = 1500  
        Do Until rsprogram.EOF  
            .ListItems.Add 1, , rsprogram.Fields("id_program").Value  
            .ListItems(1).SubItems(1) = rsprogram.Fields("program").Value &  
"  
"  
            .ListItems(1).SubItems(2) =  
rsprogram.Fields("jumlah_sesi").Value & "  
            .ListItems(1).SubItems(3) = rsprogram.Fields("biaya").Value & "  
            rsprogram.MoveNext  
        Loop  
    End With  
End Sub  
-----  
Private Sub Form_Load()  
    If koneksi.State <> adStateOpen Then  
        buka_koneksi  
    End If  
    tampil_data  
End Sub  
-----  
Private Sub Form_Unload(Cancel As Integer)  
    If koneksi.State = adStateOpen Then  
        tutup_koneksi  
    End If  
    If rsprogram.State = adStateOpen Then  
        rsprogram.Close  
        Set rsprogram = Nothing  
    End If  
End Sub  
-----  
Private Sub cmdEdit_Click()  
    If Len(pIdProgram) = 0 Then  
        MsgBox "Belum ada item data yang akan diedit", vbOKOnly +  
vbInformation, "Konfirmasi"  
    Else  
        frmUpdateCourse.Show vbModal  
    End If  
End Sub
```

```
-----  
Private Sub lstData_Click()  
    pIdProgram = lstData.SelectedItem.Text  
    pProgram = lstData.SelectedItem.SubItems(1)  
    pJumlahSesi = lstData.SelectedItem.SubItems(2)  
    pBiaya = lstData.SelectedItem.SubItems(3)  
End Sub
```

```
-----  
Private Sub cmdSelesai_Click()  
    Unload Me  
End Sub
```

4. Selanjutnya kode program untuk frmUpdateCourse

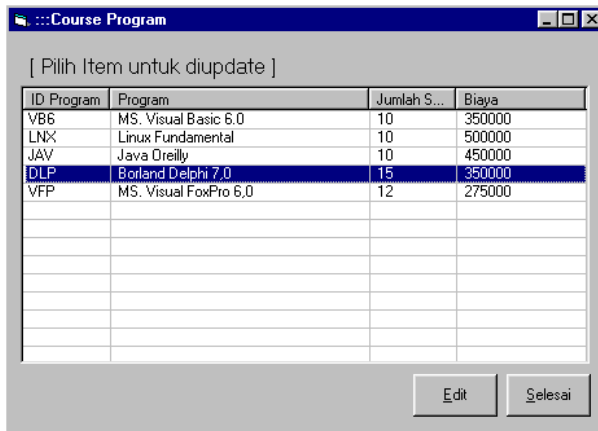
Option Explicit

```
-----  
Private Sub Form_Load()  
    If koneksi.State <> adStateOpen Then  
        buka_koneksi  
    End If  
    If Len(frmCourseProgram.pIdProgram) <> 0 Then  
        txtIdProgram.Text = frmCourseProgram.pIdProgram  
        txtProgram.Text = frmCourseProgram.pProgram  
        txtJumlahSesi.Text = frmCourseProgram.pJumlahSesi  
        txtBiaya.Text = frmCourseProgram.pBiaya  
    End If  
End Sub
```

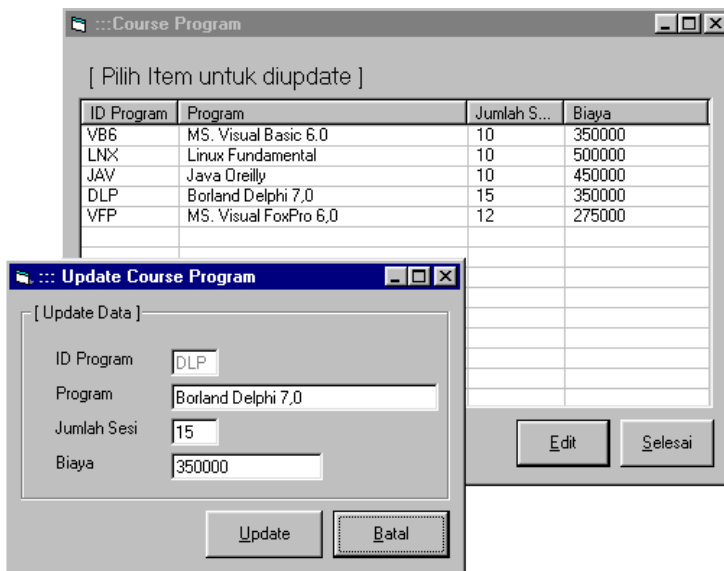
```
-----  
Private Sub cmdUpdate_Click()  
    koneksi.Execute "UPDATE r_program " _  
        & "SET program='" & txtProgram.Text & "'," _  
        & "jumlah_sesi='" & txtJumlahSesi.Text & "'," _  
        & "biaya='" & txtBiaya.Text & "'" _  
        & "WHERE id_program='" & txtIdProgram.Text & "'" _  
        frmCourseProgram.tampil_data  
    Unload Me  
End Sub
```

```
-----  
Private Sub cmdBatal_Click()  
    Unload Me  
End Sub
```

5. Selanjutnya anda dapat menjalankan program tersebut, dan akan nampak hasil programnya sebagai berikut :



Klik pada record yang mempunyai ID Program DLP, kemudian tekan tombol "Edit".



Maka akan dimunculkan sebuah form yang akan menampung seluruh baris record dari item yang kita pilih.

Selanjutnya anda dapat mengupdate beberapa field data yang ditampilkan di form, dan kemudian menekan tombol "Update"

6. Keterangan :

- Pada form frmCourseProgram didefinisikan variabel public untuk menampung data yang akan dikirimkan ke frmUpdateCourse

```
Public pIdProgram As String
Public pProgram As String
Public pJumlahSesi As String
Public pBiaya As Currency
```

- Kemudian setelah program berjalan, pengisian variabel public diisi dengan memilih item yang ada pada listView.

```
Private Sub lstData_Click()
    pIdProgram = lstData.SelectedItem.Text
    pProgram = lstData.SelectedItem.SubItems(1)
    pJumlahSesi = lstData.SelectedItem.SubItems(2)
    pBiaya = lstData.SelectedItem.SubItems(3)
End Sub
```

- Setelah variabel public diisi dengan item yang dipilih, user dapat menekan tombol Edit untuk menampilkan form frmUpdateCourse. Pada saat frmUpdateCourse diload secara otomatis mengambil variabel public dari frmCourseProgram dan menampilkannya pada TextBox yang ada di form frmUpdateCourse.

```
If Len(frmCourseProgram.pIdProgram) <> 0 Then
    txtIdProgram.Text = frmCourseProgram.pIdProgram
    txtProgram.Text = frmCourseProgram.pProgram
    txtJumlahSesi.Text = frmCourseProgram.pJumlahSesi
    txtBiaya.Text = frmCourseProgram.pBiaya
End If
```

- Dan selanjutnya proses update data dapat dijalankan.

```
Private Sub cmdUpdate_Click()
    koneksi.Execute "UPDATE r_program " _
        & "SET program='" & txtProgram.Text & "'," _
        & "jumlah_sesi='" & txtJumlahSesi.Text & "'," _
        & "biaya='" & txtBiaya.Text & "'" _
        & "WHERE id_program='" & txtIdProgram.Text & "'"
    frmCourseProgram.tampil_data
    Unload Me
End Sub
```


JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 9	MEMBUAT LAPORAN DENGAN CRYSTAL REPORT

1. BEKERJA DENGAN CRYSTAL REPORT

Laporan adalah output akhir dari sebuah system informasi. laporan yang dihasilkan tersebut dapat berupa informasi yang tertulis (tercetak pada selembar kertas) dan tidak tertulis (tampil pada monitor dan dapat pula berupa sebuah file).

Pembuatan laporan pada modul ini menggunakan Crystal Report yang merupakan salah satu produk dari **Seagate Software**, sebuah perusahaan software yang menangani perkembangan teknologi penyajian laporan. Crystal report merupakan salah satu produk yang selalu ter-update dan memiliki berbagai keunggulan.

Membuka program Crystal Report

Langkah untuk membuka jendela Crystal Report, yaitu dari **Start > Programs > Seagate Software > 32 bit Crystal Report Designer**. Selanjutnya, akan muncul tampilan **Seagate Crystal Report Registration**. Tekan tombol **Register Later**, maka muncul kotak dialog **Welcome To**. Dari kotak dialog **Welcome To** pilih **New Report** untuk membuat laporan baru atau pilih **Open Report** untuk membuka laporan. Dan **Cancel** untuk keluar dari kotak dialog **Welcome To**.

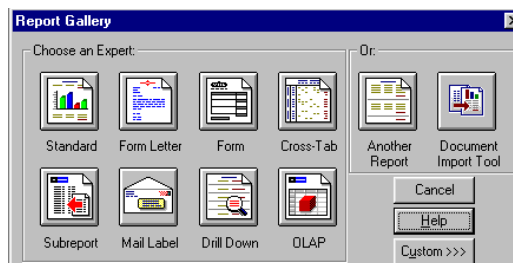


Gambar 9.1 Kotak dialog Welcome Seagate Crystal Report

Membuat Laporan dengan Crystal Report

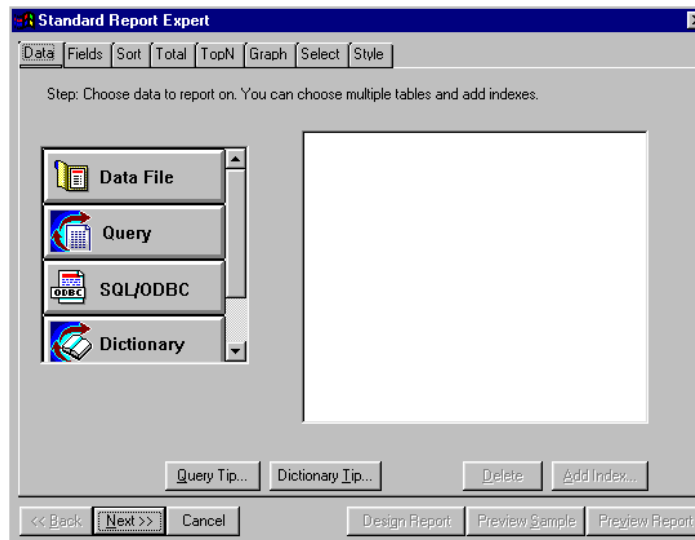
Prosedur untuk membuat laporan adalah sebagai berikut :

1. Dari kotak dialog **Welcome to** pilih **New Report**, atau bila sudah keluar dari kotak dialog **Welcome to**, pilih menu **File>New**.
2. Pada kotak dialog **Report Gallery**, klik **Standard**.



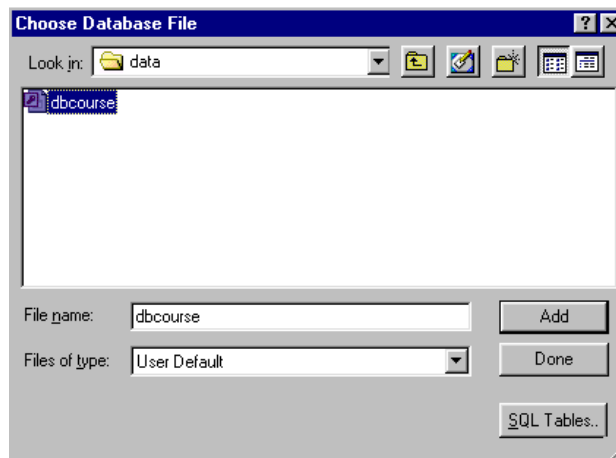
Gambar 9.2 Kotak dialog Report Gallery

3. Maka muncul kotak dialog **Standard Report Expert**.



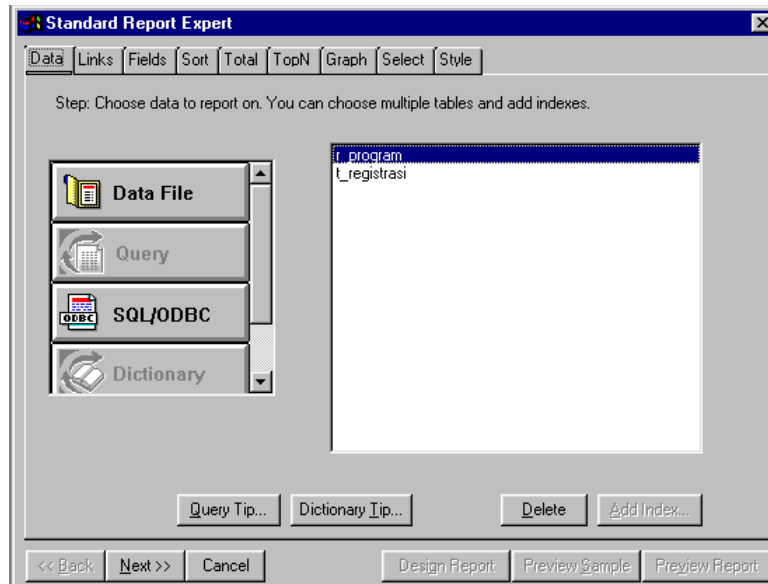
Gambar 9.3 Kotak dialog Standard Report Expert

4. Klik **Data File** dan selanjutnya akan ditampilkan kotak dialog **Choose Database File**, cari lokasi database file yang akan digunakan, yaitu C:\data\dbcourde.mdb, kemudian tekan tombol **Add**.



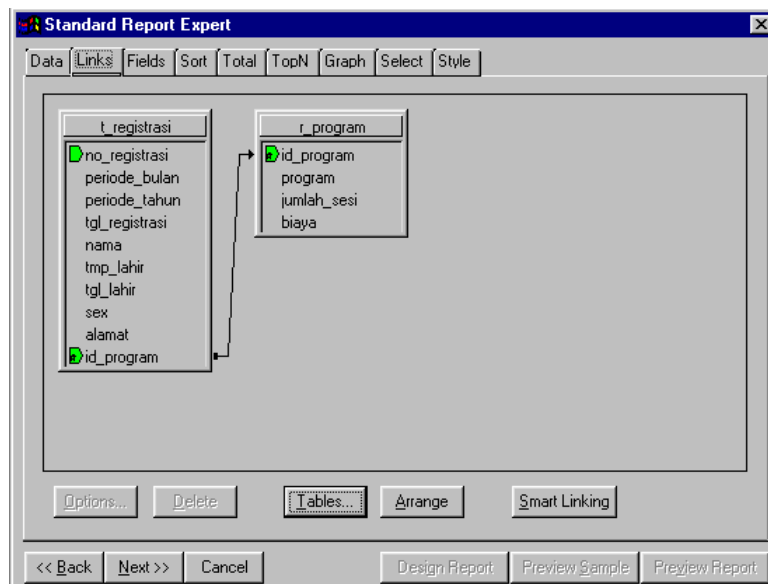
Gambar 9.4 Kotak dialog Choose Database File

5. Selanjutnya akan muncul kotak dialog **Select Tables**. Pilih nama table yang akan dibuat laporan. Pilih seluruh tabel t_registrasi dan t_program.
 6. Selanjutnya kedua tabel tersebut sudah dapat diolah untuk dibuat laporannya.
-



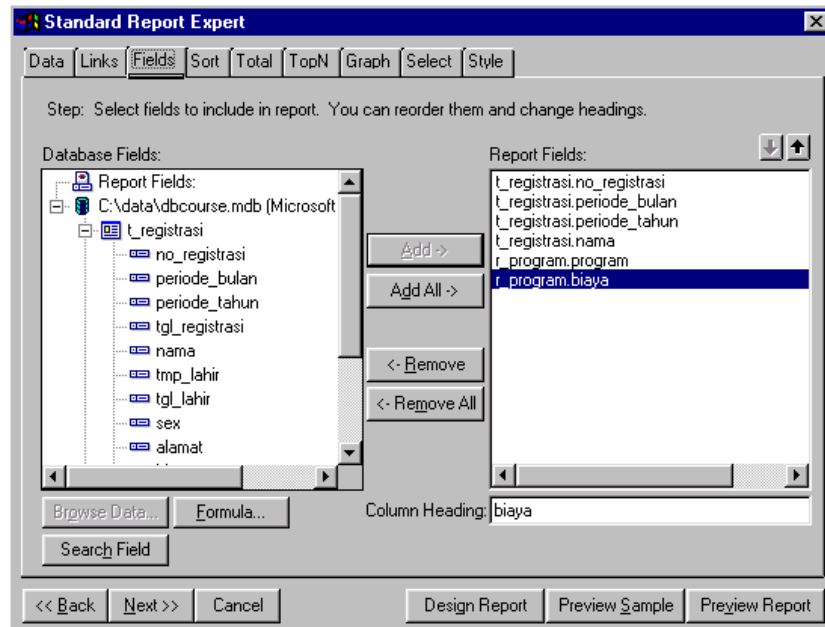
Gambar 9.5 Kotak dialog Standard Report Expert

7. Klik **Next>>** untuk nampak bahwa Crystal Report secara otomatis membuat relasi antar file table dalam database sesuai dengan field yang bersesuaian.



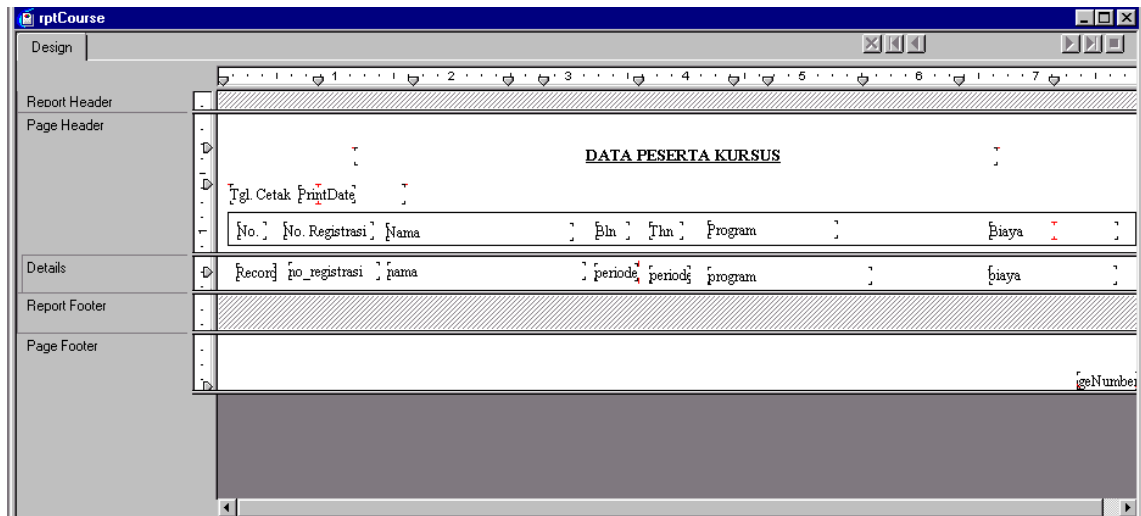
Gambar 9.6 Kotak dialog Standard Report Expert Auto Link

8. Selanjutnya pilih field-field yang akan ditampilkan pada laporan, pilih field-field :
- t_registrasi.no_registrasi
 - t_registrasi.periode_bulan
 - t_registrasi.periode_tahun
 - t_registrasi.nama
 - t_program.program,
 - t_program.biaya



Gambar 9.7 Memilih Field sebagai sumber laporan

9. Dengan tool-tool yang tersedia, Aturlah tampilan seperti berikut :



Gambar 9.8 Tampilan Desain Laporan

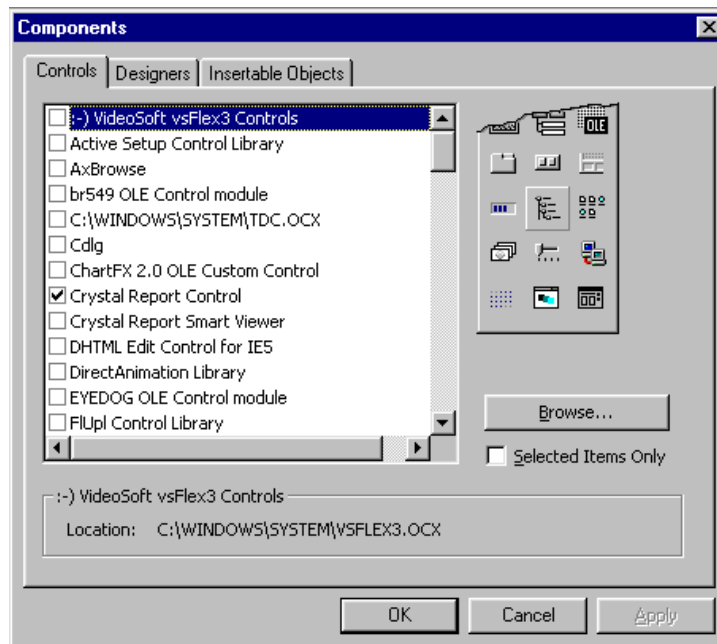
Untuk menambahkan objek line caranya dari menu **Insert** pilih **Line**, kemudian klik tahan geser dan letakkan pada posisi yang dikehendaki.

Untuk menambahkan sebuah teks caranya dari menu **Insert** pilih **Text Object**, kemudian klik pada posisi dimana anda ingin menyisipkan teks. Sebuah kotak dengan kursor akan dimunculkan, ketik teks yang diinginkan, setelah selesai klik diluar kotak teks.

2. MENGUHUBUNGKAN CRYSTAL REPORT DENGAN FORM

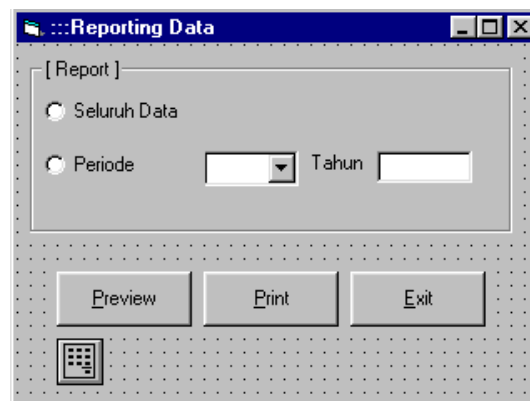
Karena pencetakan akhir dilakukan pada sebuah form, maka object Crystal Report harus ditambahkan ke dalam Project Visual Basic.

1. Buka Visual Basic dan tambahkan Komponen Crystal Report



Gambar 9.9 Menambahkan komponen Crystal Report

2. Kemudian desain form seperti berikut :



Gambar 9.10 Desain Form Laporan

3. Setting propertiesnya sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	frmReportData :::Reporting Data
Shape1	Caption	[Report]
Option1	Name Caption	OptAll Seluruh
Option2	Name Caption	OptPeriode Periode
Combo1	Name	CmbBulan
Text1	Name	TxtTahun
Command1	Name Caption	CmdPreview Previe
Command2	Name Caption	CmdPrint Print
Command3	Name Caption	CmdSelesai Exit
CrystalReport1	Name	Cr_report

4. Tulis Kode Program sebagai berikut :

```

Option Explicit

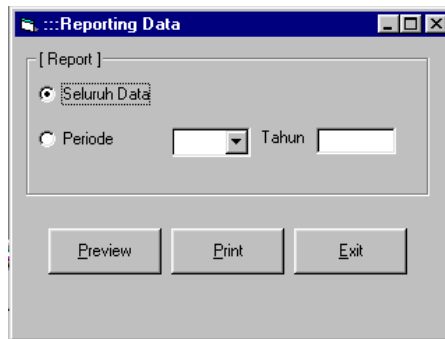
Private Sub cmdPreview_Click()
    If optAll.Value = True Then
        cr_report.ReportFileName = "rptCourse.rpt"
        cr_report.WindowState = crptMaximized
        cr_report.Destination = crptToWindow
        cr_report.Action = 1
    ElseIf optPeriode.Value = True Then
        cr_report.ReportFileName = "rptCourse.rpt"
        cr_report.SelectionFormula = "{t_registrasi.periode_bulan} = '" _
            & cmbBulan.Text & "' AND {t_registrasi.periode_tahun} = '" _
            & txtTahun.Text & "'"
        cr_report.WindowState = crptMaximized
        cr_report.Destination = crptToWindow
        cr_report.Action = 1
    Else
        MsgBox "Tentukan Pencetakan", vbOKOnly + vbInformation, "Konfirmasi"
    End If
End Sub

Private Sub cmdPrint_Click()
    If optAll.Value = True Then
        cr_report.ReportFileName = "rptCourse.rpt"
        cr_report.WindowState = crptMaximized
        cr_report.Destination = crptToPrinter
        cr_report.Action = 1
    ElseIf optPeriode.Value = True Then
        cr_report.ReportFileName = "rptCourse.rpt"
        cr_report.SelectionFormula = "{t_registrasi.periode_bulan} = '" _
            & cmbBulan.Text & "' AND {t_registrasi.periode_tahun} = '" _
            & txtTahun.Text & "'"
        cr_report.WindowState = crptMaximized
        cr_report.Destination = crptToPrinter
        cr_report.Action = 1
    Else
        MsgBox "Tentukan Pencetakan", vbOKOnly + vbInformation, "Konfirmasi"
    End If
End Sub

Private Sub cmdSelesai_Click()
    Unload Me
End Sub

```

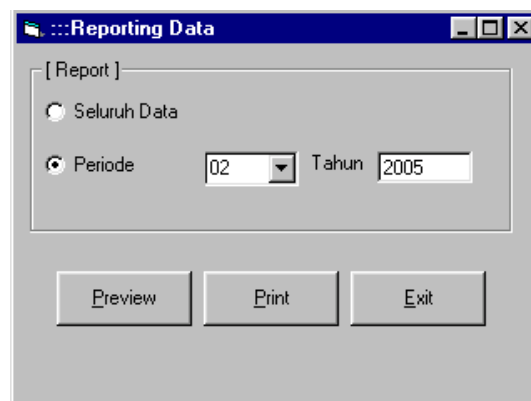
5. Jalankan program

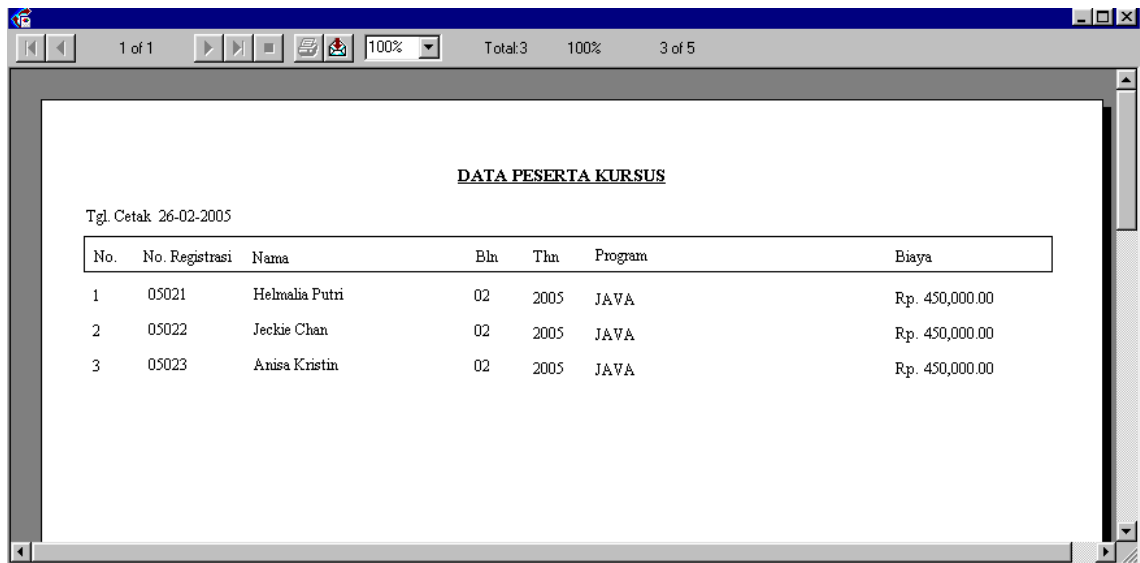


The report window displays the following data:

Tgl. Cetak 26-02-2005

No.	No. Registrasi	Nama	Bln	Thn	Program	Biaya
1	05021	Helmalia Putri	02	2005	JAVA	Rp. 450,000.00
2	05022	Jeckie Chan	02	2005	JAVA	Rp. 450,000.00
3	05023	Anisa Kristin	02	2005	JAVA	Rp. 450,000.00
4	05031	Ikhwan Arisandi	03	2005	MS. Visual Basic 6.0	Rp. 350,000.00
5	05032	Cakra Kelana	03	2005	MS. Visual Basic 6.0	Rp. 350,000.00





The screenshot shows a window titled "DATA PESERTA KURSUS" with a table of course participants. The window includes a toolbar with navigation and zoom controls, and a status bar showing "1 of 1", "Total:3", "100%", and "3 of 5". The table has columns for No., No. Registrasi, Nama, Bln, Thn, Program, and Biaya. The data is as follows:

No.	No. Registrasi	Nama	Bln	Thn	Program	Biaya
1	05021	Helmalia Putri	02	2005	JAVA	Rp. 450,000.00
2	05022	Jeckie Chan	02	2005	JAVA	Rp. 450,000.00
3	05023	Anisa Kristin	02	2005	JAVA	Rp. 450,000.00

Gambar 9.11 Tampilan hasil program

3. KETERANGAN

- Crystal Report Control memiliki property-property penting untuk berhubungan dengan Laporan, yaitu :

ReportFileName : Menentukan nama dari file laporan
WindowState : Tampilan file laporan ketika dipanggil
Destination : Tujuan dari output file laporan
SelectionFormula : Menyeleksi record berdasarkan kondisi tertentu
Action : Menampilkan laporan

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 10	VALIDASI INPUT

Validasi data sangat diperlukan terutama saat pemasukan data. Hal-hal yang menyebabkan dibutuhkan validasi data diantaranya untuk mengani kesalahan yang karena kecerobohan atau ketidak telitian dalam memasukkan data. Misalnya memasukkan huruf pada textbox yang seharusnya digunakan untuk memasukkan jumlah uang yang akan dibayar. Kesalahan ini menyebabkan program tidak dapat mengeksekusi penyimpanan data, karena adanya perbedaan pada tipe data yang akan dipakai. Untuk itu, sangat diperlukan sekali data yang valid melalui validasi data.

Berikut ini akan diperkenalkan bagaimana membuat validasi pada form pemasukan data, meliputi :

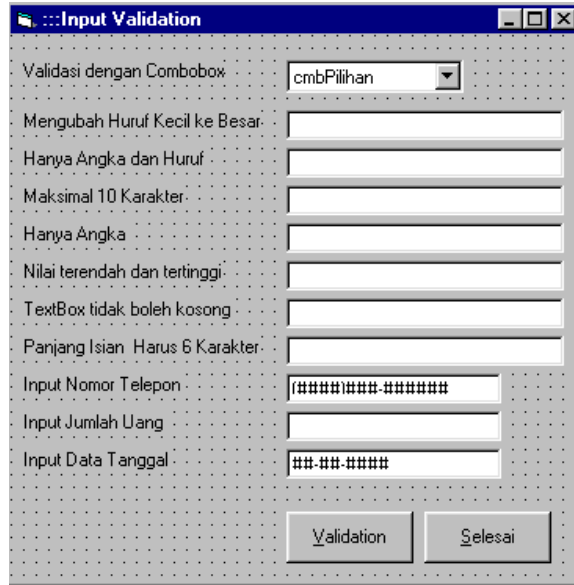
1. Validasi melalui pemilihan combo box.
2. Validasi mengubah huruf kecil ke huruf besar.
3. Validasi angka dan huruf.
4. Validasi maksimal karakter
5. Validasi hanya untuk angka
6. Validasi nilai terendah dan tertinggi
7. Validasi data input yang tidak boleh kosong
8. Validasi pengecekan panjang karakter
9. Validasi input nomor telepon
10. Validasi jumlah uang
11. Validasi data tanggal

Dalam membuat validasi ini biasanya kita memerlukan kontrol tambahan yang tidak ada dalam kontrol standard visual basic 6.0 yang ada dalam toolbox.

Membuat program validasi input

Program validasi input ini akan menunjukkan bagaimana cara kita menerima input dari user, untuk kemudian menyeleksi keabsahan dari data tersebut. Untuk membuatnya ikuti langkah-langkah berikut ini :

1. Pada project yang baru tambahkan komponen Microsoft Masked Edit Control 6.0 (**MSMASK32.OCX**).
 2. Selanjutnya buatlah form seperti berikut :
-

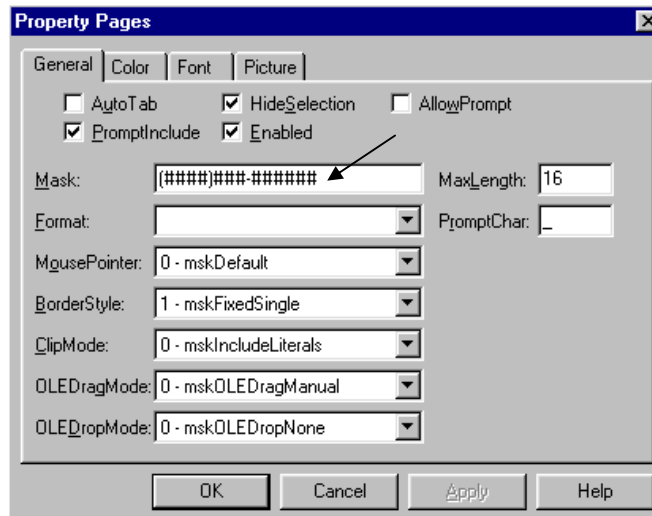


Gambar 10.1 Desain form input validasi

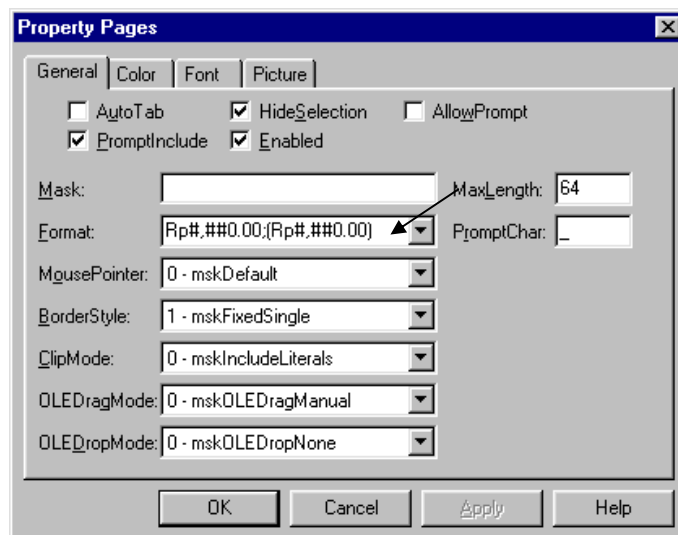
3. Selanjutnya buatlah form seperti berikut :

Komponen	Properti	Nilai
Label1	Caption	Validasi dengan Combobox
Label2	Caption	Mengubah huruf kecil ke huruf besar
Label3	Caption	Hanya angka dan huruf
Label4	Caption	Maksimal 10 Karakter
Label5	Caption	Hanya Angka
Label6	Caption	Nlai Terendah dan Tertinggi
Label7	Caption	TextBox tidak boleh kosong
Label8	Caption	Panjang isian Harus 6 karakter
Label9	Caption	Input Nomor Telepon
Label10	Caption	Input Jumlah Uang
Label11	Caption	Input Data Tanggal
Combo1	Name	CmbPilihan
Text1	Name	TxtHurufBesar
Text2	Name	TxtAngkaHuruf
Text3	Name	TxtMakHuruf
Text4	Name	TxtAngka
Text5	Name	TxtRendahTinggi
Text6	Name	TxtIsi
Text7	Name	TxtPanjangKarakter
MaskedTextBox1	Name	MskTelepon
MaskedTextBox2	Name	MskUang
MaskedTextBox3	Name	MskTanggal
Command1	Name Caption	CmdValidation &Validation
Command2	Name Caption	CmdSelesai &Selesai

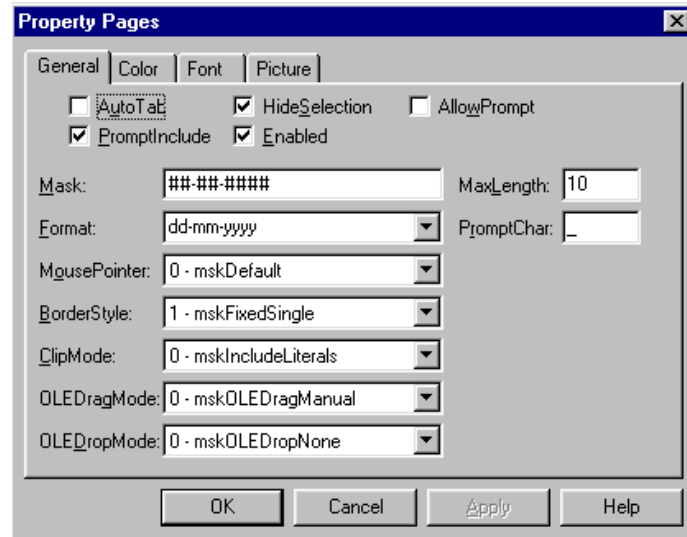
4. Khusus untuk objek MaskedTextBox, atur property-nya sesuai dengan gambar di bawah ini, dengan cara klik kanan pada objek MaskedTextBox lalu pilih properties.



Gambar 10.2 Setting property untuk MaskedTextBox1



Gambar 10.2 Setting property untuk MaskedTextBox2



Gambar 10.3 Setting property untuk MaskedTextBox3

5. Kemudian tulis kode programnya sebagai berikut :

Option Explicit

'membuat fungsi validasi secara general

Public Function validasi() As String

Dim strmessage As String

'validasi nilai tertinggi dan terendah

Dim intRendah As Integer

Dim intTinggi As Integer

Dim strpesan As String

intRendah = 1

intTinggi = 100

strpesan = "Nilai Harus diantara " & _

CStr(intRendah) & _

" - " & CStr(intTinggi)

With txtRendahTinggi

If Val(.Text) < intRendah Or Val(.Text) > intTinggi Then

strmessage = strpesan

.SetFocus

End If

End With

'validasi untuk input yang harus diisi

If Len(txtIsi.Text) = 0 Then

strmessage = "Text ini harus diisi"

```
txtIsi.SetFocus
End If

'validasi untuk pengecekan panjang karakter
With txtPanjangKarakter
  If Len(.Text) <> 6 Then
    strmessage = "INPUT harus 6 karakter"
  End If
End With
'pengembalian nilai validasi
validasi = strmessage
End Function

-----

Private Sub Form_Load()
'validasi berupa pilihan item data dengan Combobox
With cmbPilihan
  .AddItem "A"
  .AddItem "B"
  .AddItem "C"
End With
cmbPilihan.ListIndex = 0
End Sub

-----

Private Sub mskUang_keypress(keyascii As Integer)
'validasi untuk input nilai uang
Dim strAngka As String
strAngka = "0123456789"

If keyascii > 26 Then
  If InStr(strAngka, Chr(keyascii)) = 0 Then
    keyascii = 0
  End If
End If
End Sub

-----

Private Sub txtAngka_keypress(keyascii As Integer)
'validasi untuk input hanya angka
Dim strAngka As String
strAngka = "0123456789+-"

If keyascii > 26 Then
  If InStr(strAngka, Chr(keyascii)) = 0 Then
    keyascii = 0
  End If
End If
End Sub
```

```
-----  
Private Sub txtAngkahuruf_keypress(keyascii As Integer)  
    'validasi untuk input hanya angka dan huruf saja  
    Dim strAngkaHuruf As String  
    strAngkaHuruf = "0123456789" & _  
        "ABCDEFGHIJKLMNOPQRSTUVWXYZ" & _  
        "abcdefghijklmnopqrstuvwxyz"  
    If keyascii > 26 Then  
        If InStr(strAngkaHuruf, Chr(keyascii)) = 0 Then  
            keyascii = 0  
        End If  
    End If  
End Sub
```

```
-----  
Private Sub txtHurufBesar_keypress(keyascii As Integer)  
    'validasi data mengubah huruf kecil ke besar  
    keyascii = Asc(UCCase(Chr(keyascii)))  
End Sub
```

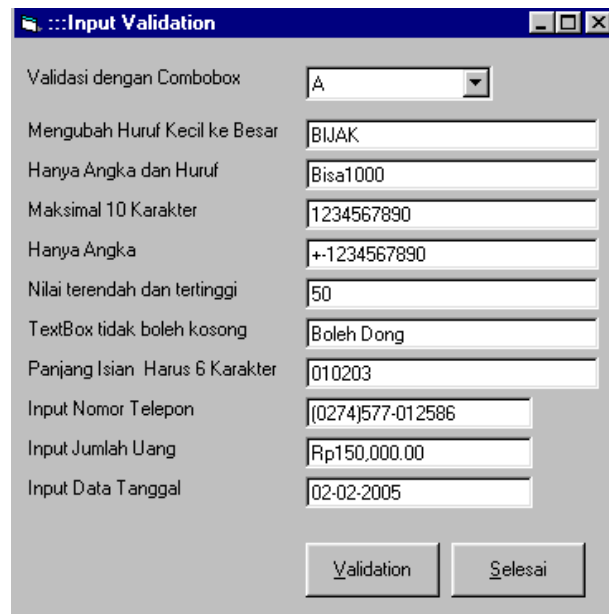
```
-----  
Private Sub txtMakHuruf_keypress(keyascii As Integer)  
    'validasi untuk membatasi maksimal 10 karakter  
    txtMakHuruf.MaxLength = 10  
End Sub
```

```
-----  
Private Sub txtPanjangKarakter_keypress(keyascii As Integer)  
    'validasi untuk membatasi maksimal 10 karakter  
    txtMakHuruf.MaxLength = 10  
End Sub
```

```
-----  
Private Sub cmdValidasi_Click()  
    'fungsi memanggil validasi data  
    Dim strErr As String  
    strErr = validasi()  
  
    If strErr = "" Then  
        Unload Me  
    Else  
        MsgBox "Ada Kesalahan : " &  
            & vbCrLf & strErr, _  
            vbOKOnly + vbInformation, "Konfirmasi"  
    End If  
End Sub
```

```
-----  
Private Sub cmdSelesai_Click()  
    Unload Me  
End Sub
```

6. Kemudian anda dapat menjalankan program, sehingga akan nampak hasil program seperti berikut :



Gambar 10.4 Hasil tampilan program

7. Keterangan :

Fungsi validasi secara general akan dipanggil saat tombol **Validation** diklik. Di dalam fungsi tersebut terdapat beberapa kondisi persyaratan yang harus dipenuhi oleh suatu objek. Misalnya pada pernyataan berikut :

```
If Len(txtIsi.Text) = 0 Then  
    strmessage = "Text ini harus diisi"  
    txtIsi.SetFocus  
End If
```

Artinya txtIsi tidak boleh kosong. Apabila kosong, akan menampilkan pesan "Textbox ini harus diisi" yang telah dideklarasikan dengan variabel strMessage. Fungsi validasi tersebut akan mengembalikan nilai berupa suatu variabel strMessage. Jika nilai yang dikembalikan kosong, berarti data sudah valid.

JURUSAN TEKNIK KOMPUTER		
POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 11	PEMROGRAMAN GRAFIK

1. MENGGAMBAR GRAFIK

Visual Basic 6.0 menyediakan cara untuk menggambar grafik. Dengan Grafik informasi yang disajikan akan mudah untuk dilihat, daripada menggunakan angka-angka yang rumit. Sejumlah fungsi yang disediakan Visual Basic 6.0 sudah mencukupi untuk merepresentasikan data dengan grafik, baik yang bersumber dari data statis maupun dinamis, misalnya data dari tabel-tabel dari sebuah database.

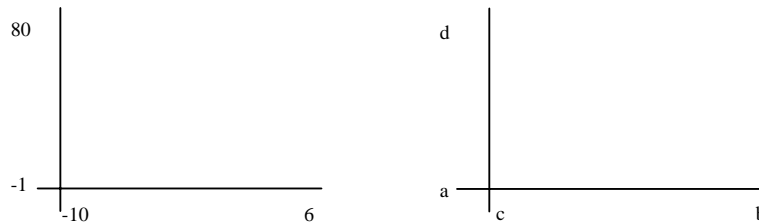
Untuk menggambar Grafik secara manual, dilakukan dengan langkah-langkah sebagai berikut :

1. Definisikan sistem koordinat
2. Gunakan metode untuk menggambar grafik apakah metode garis(*line*), kotak(*rectangle/bar*), lingkaran(*circle/Pie*).
3. Memposisikan Text untuk keterangan/legend dari Grafik.

Langkah mudah membuat grafik :

1. Membuat Sistem Koordinat

- 1) Buat sumbu X dan sumbu Y
- 2) Buat skala sebagai batas sumbu X dan sumbu Y, sebagai contoh kita tentukan untuk sumbu X dari -10 s/d 80 dan sumbu Y dari -1 s/d 6.



Dengan Visual Basic 6.0 hal yang serupa dapat dilakukan dengan cara :

```
Scale(a,d)-(b,c)
Line(a,0)-(b,0)
Line(0,c)-(0,d)
```

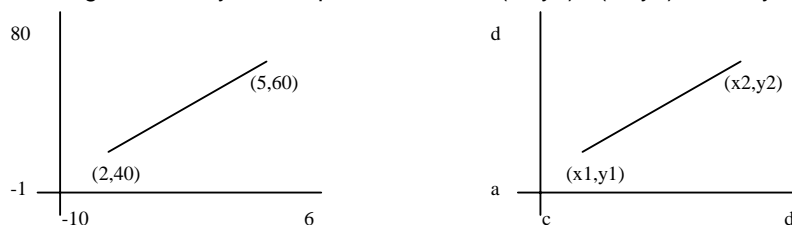
Sehingga grafik diatas dapat digambar dengan cara :

```
pGrafik.Scale (-1, 80)-(6, -10)
pGrafik.Line (-1, 0)-(6, 0)
pGrafik.Line (0, -10)-(0, 80)
```

2. Metode Grafik

1) Metode Line

Tarik garis misalnya ambil posisi koordinat (x1,y1) - (x2,y2), misalnya (2,40)-(5,60)

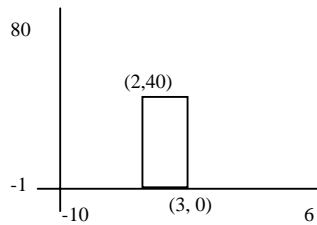


Dengan Visual Basic Hal serupa dapat dilakukan dengan
Line(x1,y1)-(x2,y2)

Sehingga dapat digambar dengan :
pGrafik.Scale (-1, 80)-(6, -10)
pGrafik.Line (-1, 0)-(6, 0)
pGrafik.Line (0, -10)-(0, 80)
pGrafik.Line (2, 40)-(5, 60)

2) Metode Bar

Buat kotak misalnya ambil posisi koordinat (x1,y1) - (x2,y2), misalnya (2,40)-(3,0) dari posisi atas ke bawah

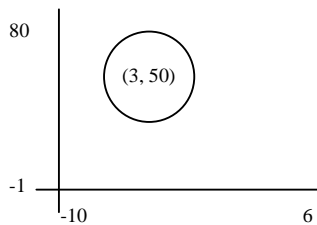


Dengan Visual Basic Hal serupa dapat dilakukan dengan
Line(x1,y1)-(x2,y2) , , B

Sehingga dapat digambar dengan :
pGrafik.Scale (-1, 80)-(6, -10)
pGrafik.Line (-1, 0)-(6, 0)
pGrafik.Line (0, -10)-(0, 80)
pGrafik.Line (2, 40)-(3, 0) , , B

3) Metode Circle

Buat lingkaran dengan pusat lingkaran (x1,y1) dan jari-jarinya n, misal pusat lingkaran (3,50) dan jari-jari 1

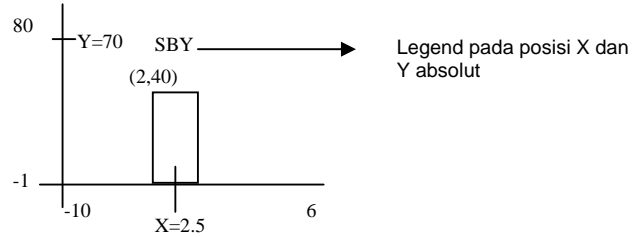


Dengan Visual Basic hal serupa dapat dilakukan dengan :
Pset(x1,y1)
Circle(x1,y1) , r

Sehingga dapat digambarkan dengan :
pGrafik.Scale (-1, 80)-(6, -10)
pGrafik.Line (-1, 0)-(6, 0)
pGrafik.Line (0, -10)-(0, 80)
pGrafik.PSet (3, 50)
pGrafik.Circle (3, 50), 1

3. Memposisikan Text.

Untuk menempatkan keterangan dalam Grafik cukup diberikan pada posisi sumbu X dan Y secara absolut.



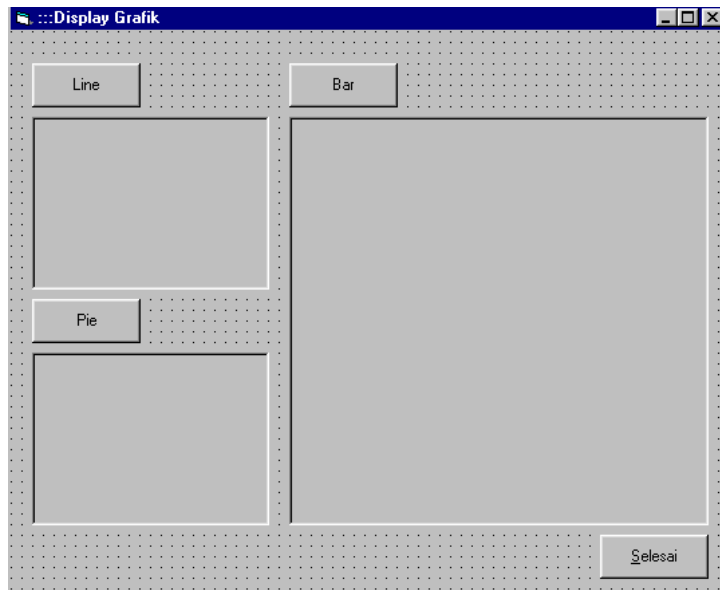
Dengan Visual Basic hal serupa dapat dilakukan dengan cara :
CurrentY=70
CurrentX=2.5

Sehingga dapat digambar dengan Visual Basic dengan cara :
pGrafik.Scale (-1, 80)-(6, -10)
pGrafik.Line (-1, 0)-(6, 0)
pGrafik.Line (0, -10)-(0, 80)
pGrafik.Line (2, 40)-(3, 0), , BF
pGrafik.CurrentY = 70
pGrafik.CurrentX = 2.5
pGrafik.Print "SBY"

2. PROGRAM MENGGAMBAR GRAFIK

Untuk menggambar grafik dengan Visual Basic, anda dapat menggunakan object form secara langsung, atau dengan menggunakan object PictureBox.

1. Desain form seperti tampak pada gambar berikut :



Gambar 11.1 Desain frmGrafik

2. Setting propertiesnya sebagai berikut :

Komponen	Properties	Nilai
Form1	Name Caption	FrmGrafik :::Display Grafik
Command1	Name Caption	CmdLine Line
Command2	Name Caption	CmdBar Bar
Command3	Name Caption	CmdPie Pie
Command4	Name Caption	CmdSelesai Selesai
Picturebox1	Name	Pline
Picturebox2	Name	Pbar
Picturebox3	Name	Ppie

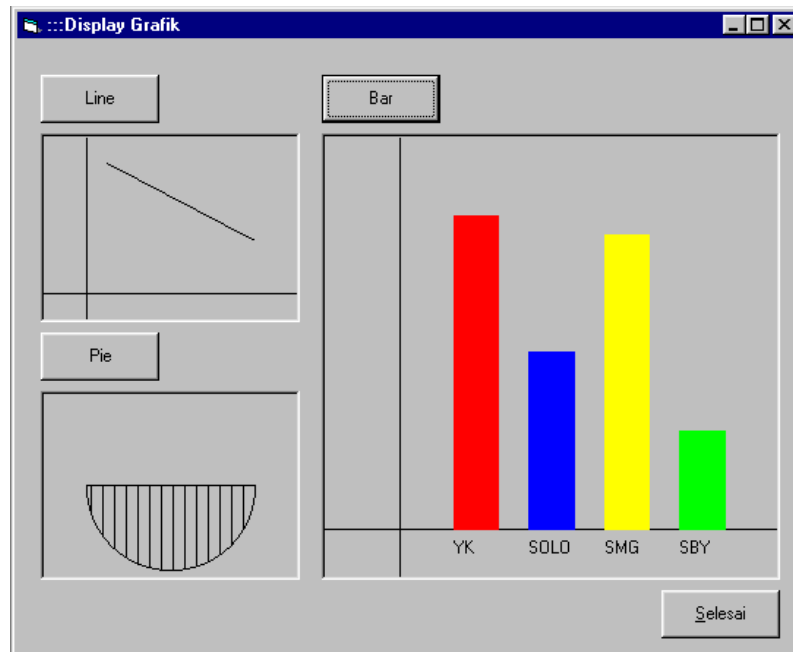
3. Tulis kode program

```

Private Sub cmdBar_Click()
    pBar.Scale (-1, 40)-(5, -5)
    pBar.Line (-1, 0)-(5, 0)
    pBar.Line (0, -5)-(0, 40)
    pBar.Line (0.7, 32)-(1.3, 0), vbRed, BF
    pBar.Line (1.7, 18)-(2.3, 0), vbBlue, BF
    pBar.Line (2.7, 30)-(3.3, 0), vbYellow, BF
    pBar.Line (3.7, 10)-(4.3, 0), vbGreen, BF
    pBar.CurrentY = -1
    pBar.CurrentX = 0.7
    pBar.Print "YK"
    pBar.CurrentY = -1
    pBar.CurrentX = 1.7
    pBar.Print "SOLO"
    pBar.CurrentY = -1
    pBar.CurrentX = 2.7
    pBar.Print "SMG"
    pBar.CurrentY = -1
    pBar.CurrentX = 3.7
    pBar.Print "SBY"
End Sub
-----
Private Sub cmdLine_Click()
    pLine.Cls
    pLine.Scale (-2, 18)-(10, -3)
    pLine.Line (-2, 0)-(10, 0)
    pLine.Line (0, -3)-(0, 18)
    pLine.Line (1, 15)-(8, 6)
End Sub
-----
Private Sub cmdPie_Click()
    Dim c As Single
    c = 2 * 3.14159
    pPie.Cls
    pPie.Scale (-3, 3)-(3, -3)
    pPie.FillStyle = 3
    pPie.Circle (0, 0), 2, , -0.5 * c, -1 * c
End Sub
-----
Private Sub cmdSelesai_Click()
    Unload Me
End Sub

```

4. Jalankan program sehingga nampak hasil sebagai berikut :



Gambar 11.2 Desain frmGrafik

3. KETERANGAN

- Metode CLS
Digunakan untuk menghapus teks atau grafis yang dijalankan dari suatu form atau picturebox.
- Metode Line dipergunakan untuk membuat garis, persegi atau mengisi kotak. Bentuk penulisan dari metode Line sebagai berikut :

Object. **Line[Step] (x1,y1), [Color],[B][F]**

- Step : kata kunci ini sifatnya pilihan yang khusus digunakan pada koordinat titik awal dari posisi grafis yang diberikan oleh sumbu X dan sumbu Y.
- (x1,y1) : sifatnya pilihan yaitu nilai single yang menyatakan koordinat dari titik awal pada suatu garis atau segiempat.
- Step : kata kunci ini sifatnya pilihan yang khusus yang khusus digunakan pada koordinat titik akhir dari titik awal suatu garis.
- Color : sifatnya pilihan yaitu nilai Long Integer yang menyatakan warna RGB untuk digunakan pada pewarnaan suatu garis.
- B : sifatnya pilihan dan jika diikutkan menyebabkan kotak yang digambar dengan menggunakan koordinat untuk menentukan ujung yang dihadapan dari suatu kotak.
- F : sifatnya pilihan. Jika option B digunakan maka option F digunakan untuk menentukan kotak yang diisi dengan warna yang sama dengan warna untuk menggambar kotak. Option F tidak dapat digunakan tanpa option B.
-

- Metode Circle digunakan untuk menggambar lingkaran, ellips atau garis lengkung. Bentuk penulisan dari method Circle, sebagai berikut :

Object. **Circle[Step] (x,y), Radius,[Color,start,end,aspect]**

- Step : kata kunci ini sifatnya pilihan yang khusus sebagai pusat lingkaran, ellips atau garis lengkung dimana arah koordinatnya diberikan oleh CurrentX dan CurrentY pada properties obyek.
- (x1,y1) : nilai single yang diindikasikan sebagai koordinat titik pusat dari lingkaran, ellips atau garis lengkung.
- Radius : nilai single yang diindikasikan sebagai jari-jari dari lingkaran, ellips atau garis lengkung.
- Color : sifatnya pilihan yaitu nilai long integer yang diindikasikan pada fungsi warna RGB dari garis lingkaran.
- Start,end : ketika garis lengkung, bagian bagian lingkaran, ellips digambar, *start* dan *end* menentukan posisi di awal dan akhir dari suatu garis lengkung. Nilai standard untuk *start* adalah 0 radian dan nilai standar untuk *end* adalah $2 * \pi$ radian. Sedang batasan nilainya mulai $-2 * \pi$ radian s/d $2 * \pi$ radian.
- Aspect : nilai single precision yang diindikasikan sebagai perbandingan aspek dari suatu lingkaran. Nilai standardnya adalah 1.0

- Metode Pset dipergunakan untuk menentukan pusat lingkaran.
 - Anda selanjutnya dapat berkreasi dengan menggunakan sumber data dari Database.
-

JURUSAN TEKNIK KOMPUTER POLITEKNIK NEGERI SRIWIJAYA PALEMBANG		
MICROSOFT VISUAL BASIC 6.0	MODUL 12	- MENU BAR - STATUS BAR - TOOL BAR

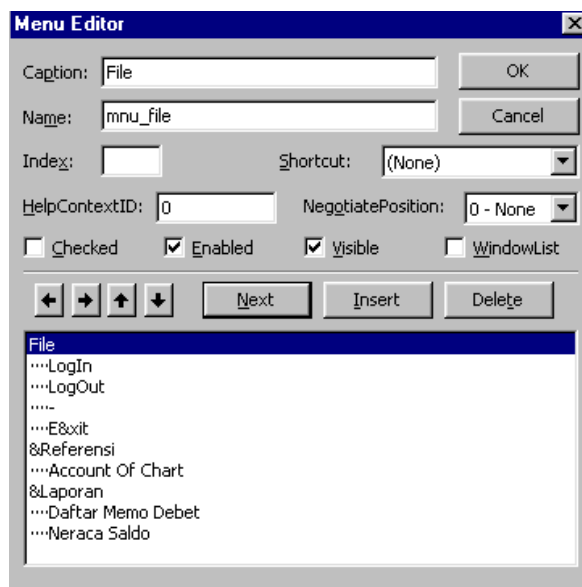
Visual Basic 6.0 menyediakan sarana untuk membuat menu tanpa terlebih dahulu membuat dasar pemrogramannya. Pembuatan menu mutlak diperlukan, mengingat program yang sudah siap pakai bisa menjadi lebih efektif dan praktis. Selain itu, bentuk menu yang cantik bisa menjadi daya tarik tersendiri bagi user.

1. MEMBUAT MENU BAR

Menu Bar adalah menu standar yang berlaku sebagai menu utama pada setiap program aplikasi. Untuk membuat menu Bar adalah sebagai berikut :

1. Klik menu Tools pilih Menu Editor.
2. Pada kolom Menu Editor, ketik judul menu dan submenu seperti berikut :

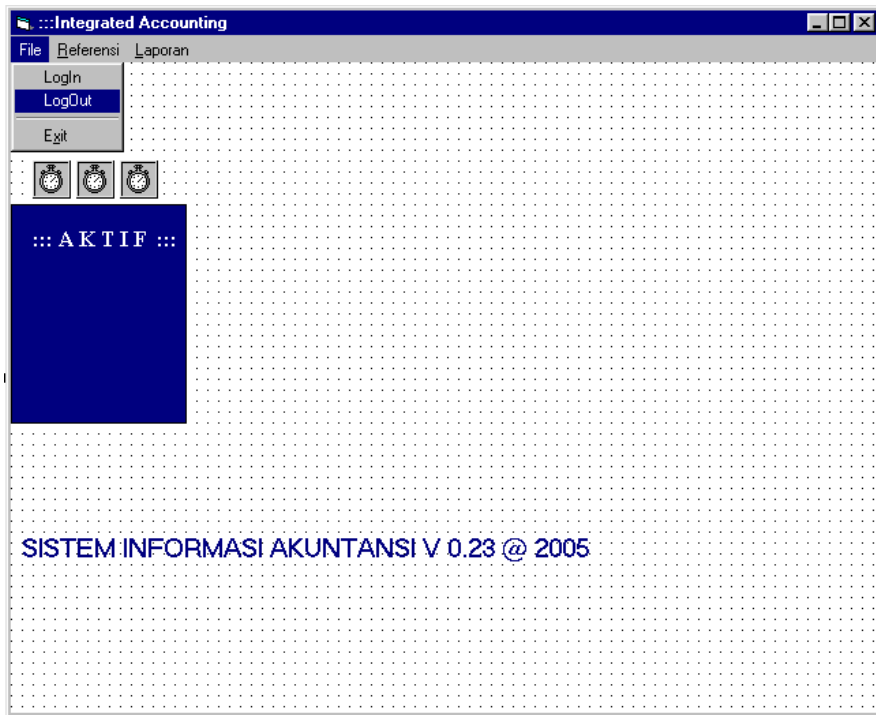
Caption	Name	Shortcut
&File	Mnu_File	(None)
Login	Mnu_login	(None)
Logout	Mnu_logout	(None)
-	Mnu_spt1	(None)
E&Xit	Mnu_Exit	Ctrl+X
&Referensi	Mnu_data	(None)
Account Of Chart	Mnu_chart	(None)
&Laporan	Mnu_laporan	(None)
Daftar Memo Debet	Mnu_meDebet	(None)
Neraca Saldo	Mnu_neracasaldo	(None)



Gambar 12.1 Menu Editor

Cara mengisi input Menu Editor adalah sebagai berikut :

- Untuk menu utama, isi kolom input Caption (&File) dan Name(mnu_file).
 - Untuk pengisian menu berikutnya, klik baris kosong di bawah menu yang telah terisi pada listbox (atau klik tombol **N**ext).
 - Submenu ditandai oleh tanda empat buah titik (...) yang akan muncul bila tombol → ditekan.
 - Untuk garis pemisah antara menu, ketik karakter -.
 - Untuk bantuan tombol keyboard (missal : Ctrl + X), isi kolom "Shortcut".
 - Karakter & memberi arti bahwa karakter sesudahnya ditandai garis bawah.
3. Klik tombol OK setelah selesai.
 4. Anda dapat melihat Menu yang baru saja anda buat, dengan menjalankan form.
 5. Untuk mempercantik form dari menu, berilah warna dasar putih pada form.
 6. Tambahkan 3 buah kontrol Timer, 2 buah kontrol Label dan 1 buah kontrol Shape



Gambar 12.2 Kontrol tambahan pada menu

7. Atur property dari masing-masing kontrol seperti berikut :

Komponen	Properti	Nilai
Timer1	Name Interval	Timer1 1000
Timer2	Name Interval	Timer2 500
Timer3	Name Interval	Timer3 100
Label1	Name Caption	Aktif ::: AKTIF :::
Label2	Name Caption	Marquee Sistem Informasi Akuntansi V 0.25 @ 2005
Shape1	Name Backstyle backColor	Kotak Opaque &H00800000&

8. Selanjutnya agar tampilan dari menu anda kelihatan lebih sempurna, anda dapat menambahkan kontrol **StatusBar**.

2. STATUS BAR

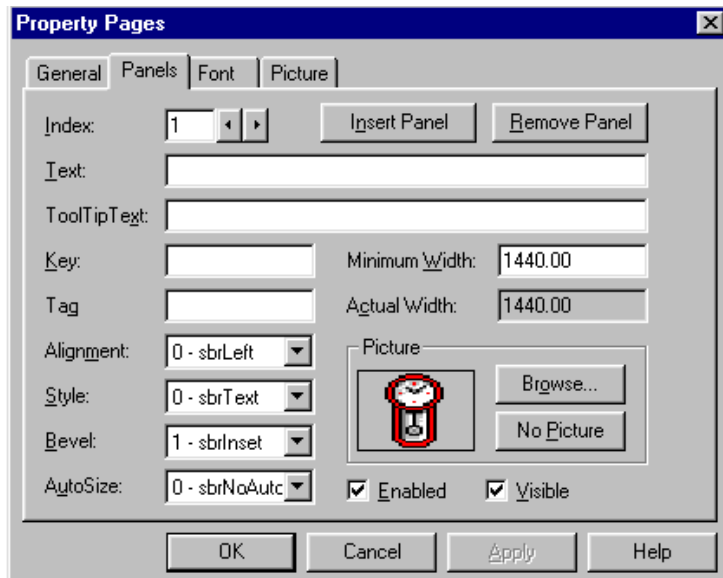
Untuk membuat status bar dipergunakan **StatusBar Control**, kontrol ini merupakan frame yang berisi panel sebagai penunjuk dari suatu aplikasi, seperti penunjuk status **Caps Lock** atau **Num Lock**, penunjuk waktu, penunjuk tanggal ataupun untuk menampilkan status user yang sedang aktif menggunakan program aplikasi.

StatusBar Control terletak pada Microsoft Windows Common Controls 6.0, untuk menggunakannya, StatusBar Control harus diletakkan pada **ToolBox**.

Cara membuat StatusBar (contoh) :

Berikut ini contoh untuk membuat Control Panel, langkah-langkahnya sebagai berikut :

1. Tempatkan **StatusBar Control** pada form.
2. klik kanan bidang **StatusBar Control** dan pilih properties (muncul kotak dialog property pages).
3. Klik tab "Panels".
4. Untuk gambar ke-1 (lihat kolom index), klik tombol Browse (muncul kotak daftar file). Cari pada drive/folder/file mana gambar yang akan disisipkan.



Gambar 12.3 Property Pages StatusBar

5. Klik tombol open, pada frame Picture muncul sebuah gambar
6. Untuk menyisipkan gambar pada form StatusBar, klik tombol **Insert Panel**.
7. Ulangi langkah 4-6 untuk gambar ke-2.
8. Sampai di sini tampilan form anda kurang lebih akan seperti berikut :



Gambar 12.4 Tampilan StatusBar

9. Selanjutnya akan lebih sempurna, jika dalam form yang anda buat ditambahkan **ToolBar** sebagai menu tambahan.

3. TOOL BAR

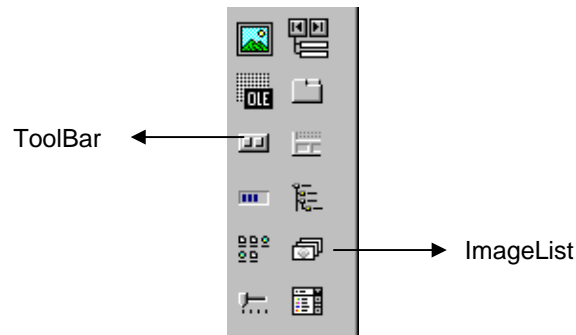
ToolBar control merupakan tombol untuk membuat menu berbentuk tombol yang bisa berisi image icon, tulisan, tooltip, ataupun ketiga-tiganya.

ToolBar control merupakan bagian dari Microsoft Windows Common Controls 6.0. untuk penggunaannya, **ToolBar** control harus diletakkan dalam **ToolBarBox**.

Sebagai kontrol yang bisa menampilkan image, ToolBar control harus disandingkan dengan **ImageList Control**, selain itu, **ComboBox** dan **TextBox Control** dapat diletakkan dalam **ToolBar Control**.

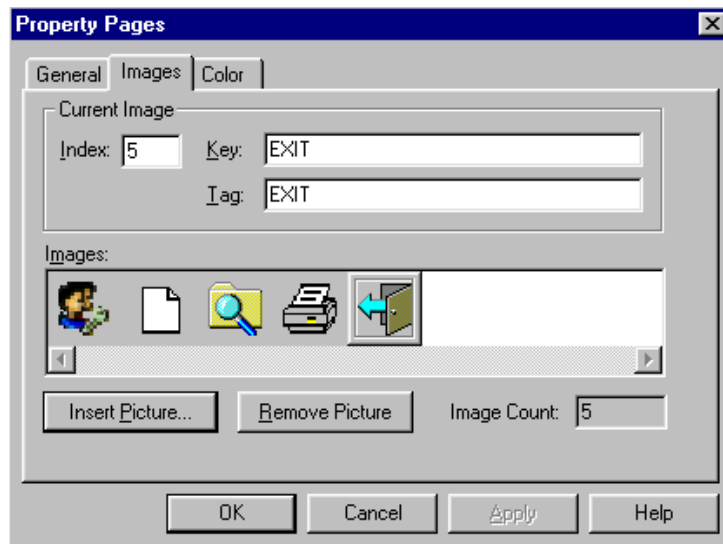
Untuk membuat ToolBar, ikuti langkah-langkah sebagai berikut :

1. Tempatkan **ToolBar Control** dan **ImageList Control** pada bidang form.



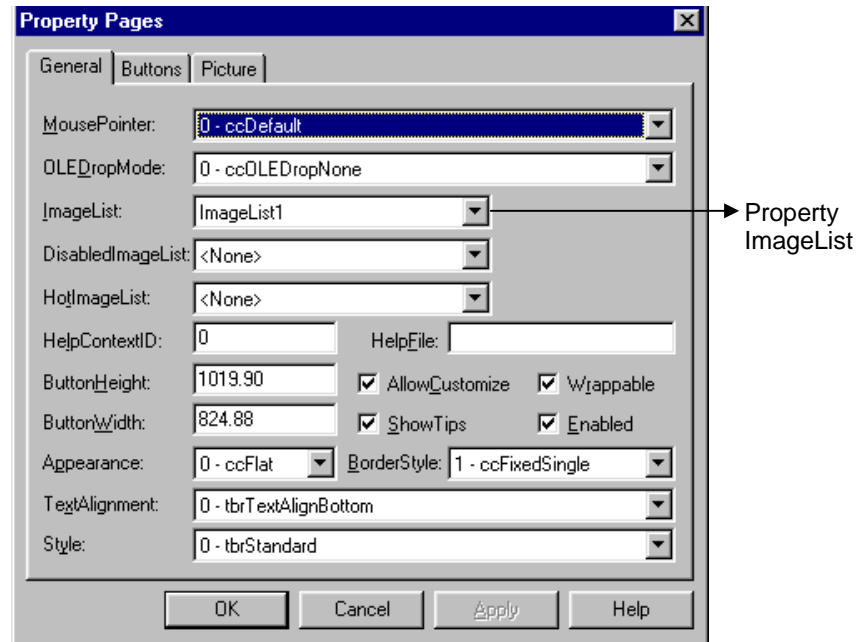
Gambar 12.5 Control ToolBar dan ImageList

2. Klik Kanan pada ImageList Control yang sudah ditempatkan dalam form. Pilih Properties (muncul kotak dialog Properti Pages).
3. Pada kotak dialog Properti Pages pilih tab Image, klik tombol **Insert Picture**, pada kotak dialog **Select Picture**, cari file gambar/icon yang akan ditempatkan pada **ImageList**.



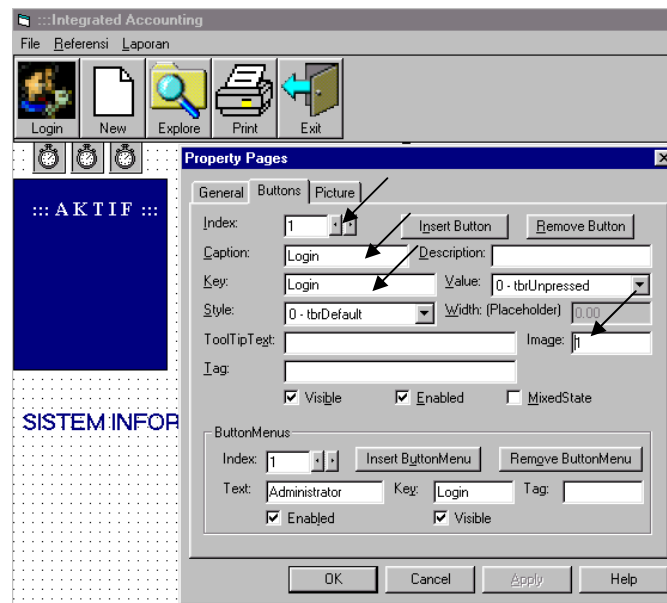
Gambar 12.6 Kotak dialog ImageList Property

4. Isi nama image ke-1 pada kolom Key (misal : index -1 : Key=LOGIN, index -2 : Key=NEW, index -3 : Key=PREVIEW, index -4 : Key=PRINT, index -5 : Key=EXIT)
5. Klik OK untuk menutup kotak dialog **ImageList Property Pages**.
6. Langkah selanjutnya adalah membuat tombol untuk kontrol **ToolBar**.
7. Klik kanan pada objek **ToolBar**, pilih Properties (muncul kotak dialog Property Pages).
8. Klik tab **General**, pada kolom **ImageList**, masukkan **ImageList1**.



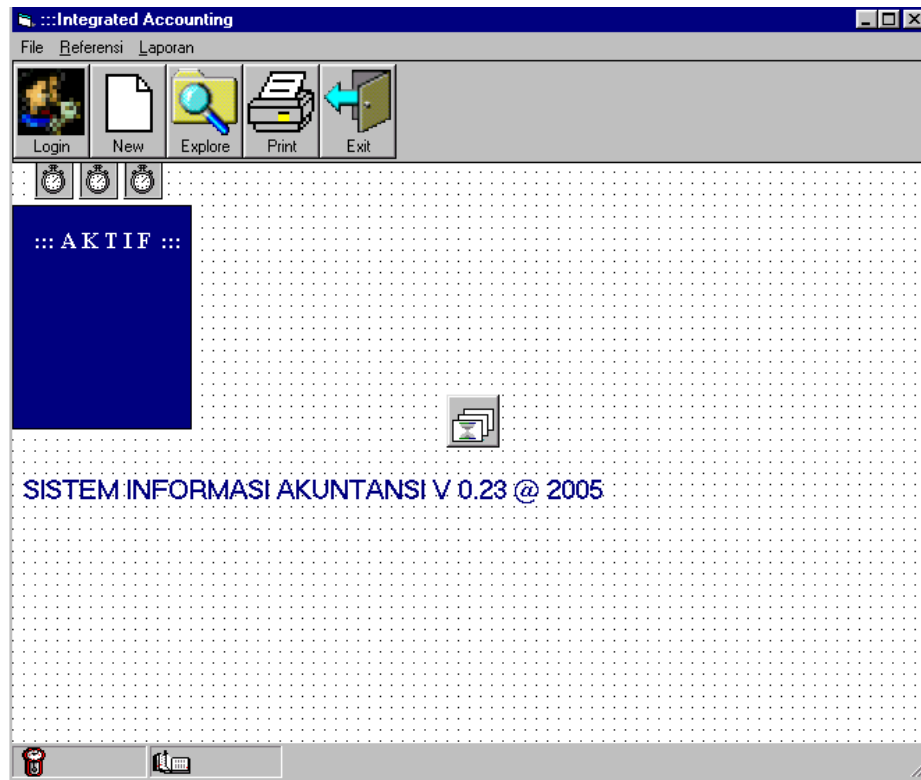
Gambar 12.7 Kotak dialog tab General Toolbar Properti Pages

9. Klik tab "Buttons", isi judul tombol pada kolom caption untuk tombol ke-1 (lihat "index"), misal (index-1: Login, index-2: New, index-3: Explore, index-4: Print, index-5: Exit). Pada kolom Image isi sesuai dengan index yang ada pada **ImageList Property**.
10. Ulangi langkah di atas hingga semua tombol diberi judul dan gambar.



Gambar 12.8 Memberi judul dan gambar pada tombol

11. Selanjutnya form dari menu utama anda kurang lebih akan tampak seperti berikut ini :



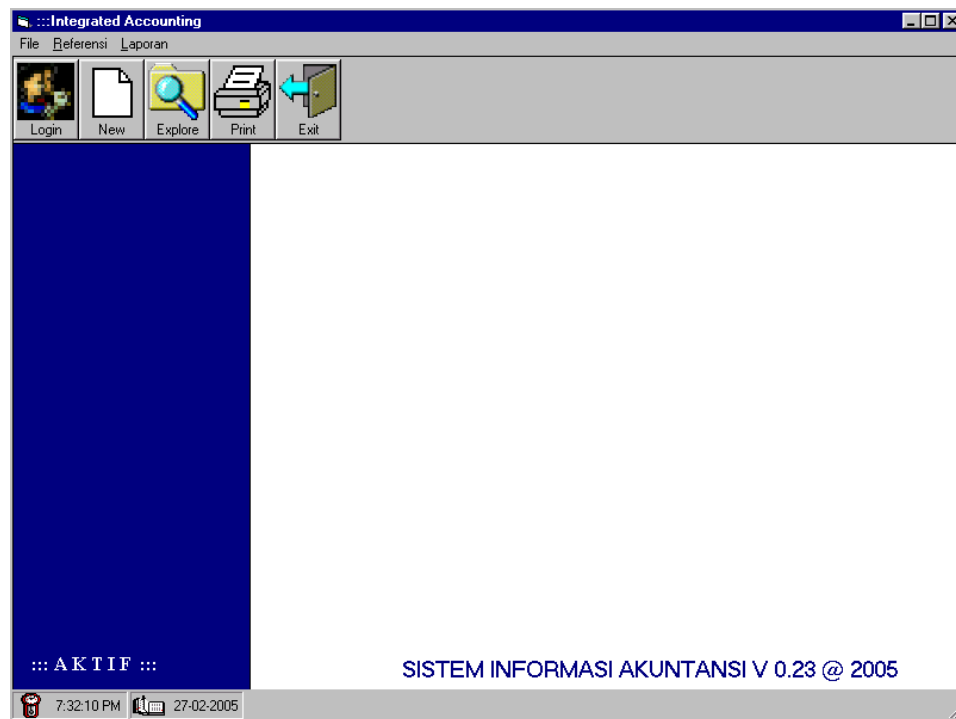
Gambar 12.9 Tampilan form menu utama

12. Langkah selanjutnya adalah menuliskan kode program untuk form, seperti berikut :

```
Option Explicit
-----
Private Sub Form_Load()
kotak.Top = 0
kotak.Left = 0
kotak.Height = Screen.Height
kotak.Width = 3000
End Sub
-----
Private Sub Timer1_Timer()
With StatusBar1
.Panels(1).Text = Time
.Panels(2).Text = Format(Date, "dd-mm-yyyy")
End With
End Sub
-----
Private Sub Timer2_Timer()
If aktif.Visible = True Then
aktif.Top = 1000
aktif.Visible = False
Else
aktif.Top = 7500
aktif.Visible = True
End If
End Sub
```

```
-----  
Private Sub Timer3_Timer()  
    marquee.Refresh  
    marquee.Top = Me.Height - 1500  
    marquee.Move marquee.Left - 300  
    If marquee.Left < Me.Left Then  
        marquee.Left = Me.Width  
    End If  
End Sub  
-----  
Private Sub mnu_exit_Click()  
    If MsgBox("Apakah Anda Akan Mengakhiri" _  
        & vbCrLf & "Aplikasi ini ?", _  
        vbYesNo + vbQuestion, "Konfirmasi") = vbYes Then  
  
        End  
    End If  
End Sub
```

13. Selanjutnya program sudah dapat dijalankan, dan hasilnya kurang lebih akan nampak seperti berikut :



Gambar 12.10 Tampilan Menu Utama ketika Running

14. Keterangan :

- Pada saat form diload, maka otomatis kontrol shape(kotak) akan mempunyai tinggi yang sama dengan objek Screen/layar monitor, dan akan memiliki lebar 3000 pixel, kode program yang digunakan adalah :

```
Private Sub Form_Load()  
    kotak.Top = 0  
    kotak.Left = 0  
    kotak.Height = Screen.Height  
    kotak.Width = 3000  
End Sub
```

- Pada status bar akan muncul informasi jam dan tanggal yang akan selalu berubah, sesuai dengan interval yang diberikan kepada property kontrol Timer1 yaitu selama 1 detik (interval= 1000)

```
Private Sub Timer1_Timer()  
With StatusBar1  
    .Panels(1).Text = Time  
    .Panels(2).Text = Format(Date, "dd-mm-yyyy")  
End With  
End Sub
```

- Tulisan aktif akan nampak berkedip-kedip, sesuai dengan interval yang diberikan kepada objek Timer2, yaitu selama 0.5 detik (interval=500)

```
Private Sub Timer2_Timer()  
If aktif.Visible = True Then  
    aktif.Top = 1000  
    aktif.Visible = False  
Else  
    aktif.Top = 7500  
    aktif.Visible = True  
End If  
End Sub
```

- Tulisan "Sistem Informasi Akuntansi V 0.25 @ 2005" akan nampak berjalan dari sebelah kanan ke kiri dengan pergerakan selama 0.1 detik (interval=100).

```
Private Sub Timer3_Timer()  
marquee.Refresh  
marquee.Top = Me.Height - 1500  
marquee.Move marquee.Left - 300  
If marquee.Left < Me.Left Then  
    marquee.Left = Me.Width  
End If  
End Sub
```

Tentang Penulis



M. Miftakul Amin lahir di Raman Aji, Lampung Timur. Menyelesaikan jenjang Strata 1 di Jurusan Teknik Informatika, Universitas Teknologi Yogyakarta (UTY). Kemudian melanjutkan studi Strata 2 di Jurusan Teknik Elektro dan Teknologi Informasi Universitas Gadjah Mada (UGM) dengan mengambil konsentrasi *Computer and Information System (CIS)*. Aktifitas saat ini adalah sebagai dosen tetap di jurusan teknik informatika/komputer Politeknik Negeri Sriwijaya Palembang. Selain sibuk mengajar penulis juga seorang *programmer* yang cukup berpengalaman mengembangkan sistem informasi yang telah banyak digunakan baik oleh instansi swasta maupun pemerintah.

Informasi kontak :

homepage : <http://mafisamin.web.ugm.ac.id>
e-mail : mafis_amin@yahoo.com
mobile : 0813 790 18 202
y-messenger : mafis_amin
software Dev. : <http://www.masaminsoft.com>
