

MODUL  
PEMROGRAMAN VISUAL BASIC



SEKOLAH TINGGI ILMU KOMPUTER AL- KHAIRIYAH

# PENGANTAR

Modul Pemrograman Visual Basic ini dibuat dengan tujuan untuk memperkenalkan Lingkungan Visual Basic, pembuatan *interface* dengan menggunakan *tools* yang ada serta mengimplementasikan program sederhana dengan bahasa Visual Basic. Disamping itu juga untuk memahami struktur dan perintah-perintah program bahasa Visual Basic. Terutama sekali untuk memberikan tambahan literatur bagi Mahasiswa STIKOM AL-KHAIRIYAH dalam menempuh matakuliah Pemrograman Visual Basic.

Dalam modul ini terdiri dari 12 Bab yang diberikan untuk satu semester. Pada setiap pertemuan diberikan contoh-contoh program dan latihannya, diharapkan dengan mencoba contoh program yang ada dan mengerjakan latihannya, mahasiswa lebih mudah untuk memahami materi yang diberikan.

Penyusun berharap semoga modul ini bisa membantu pembacanya khususnya bagi mahasiswa STIKOM Al-Khairiyah dalam belajar bahasa pemrograman Visual Basic.

Tentunya dalam modul ini juga masih terdapat kekurangan-kekurangan maka dalam kesempatan ini penyusun menerima segala macam kritik yang bersifat konstruktif demi perbaikan materi pada modul inidi masa mendatang. Terima Kasih.

Cilegon,

Penyusun

## BAB I

### PENDAHULUAN

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman Visual Basic dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
1. Mengetahui perkembangan bahasaBasic	<ul style="list-style-type: none"> <li>▪ Menjelaskan perkembangan bahasaBasic</li> <li>▪ Mengenal Visual Basic dan menjelaskan perkembangan versinya mulai dari versi 3.0 sampai versi 6.0</li> </ul>
2. Mengenal lingkungan Visual Basic	<ul style="list-style-type: none"> <li>▪ Mengenal IDE Visual Basic</li> <li>▪ Memahami Property, Method danEvent</li> </ul>
3. Menggunakan VisualBasic	<ul style="list-style-type: none"> <li>▪ Membuka program Visual Basic</li> <li>▪ Mengetahui langkah-langkah membuat aplikasi dengan Visual Basic</li> <li>▪ Menjalankan, menghentikan dan menyimpan program yang dibuat diVisual Basic</li> </ul>

#### A. Perkembangan Bahasa Basic

Bahasa Basic merupakan bahasa populer dan disukai banyak programmer karena kemudahannya serta bahasanya yang cukup familier tanpa mengurangi performance kinerjanya. Sejak kemunculannya pada tahun 1960, bahasa Basic telah mengalami perkembangan yang pesat sekali. Di tahun 1970 digunakan oleh Bill Gates dan Paul Allen untuk mengontrol mikrokomputer Altair dengan menggunakan pita kaset. Kemudian bahasa Basic diikuti oleh pengembang-pengembang software lain dengan nama yang berbeda, namun aturan dan bahasa yang digunakan adalah sama. Munculnya GW-Basic, Qbasic, Quick Basic dan lain sebagainya semakin mempopulerkan bahasa Basic ini untuk digunakan pada mikrokomputer sebagai bahasa pemrograman untuk membuat aplikasi.

## B. Mengetahui Visual Basic

Visual Basic untuk DOS dan Windows diperkenalkan pada tahun 1991. Versi 3.0 dari Visual Basic dikeluarkan pada tahun 1993 dan lebih mengalami kemajuan yang pesat dibandingkan dengan versi sebelumnya. Visual Basic 3.0 masih menggunakan kode-kode yang bekerja dalam 16 bit. Kemudian pada akhir tahun 1995 dikeluarkan Visual Basic versi 4.0 yang mendukung proses 32 bit. Pada akhir tahun 1996 dikeluarkan Visual Basic versi 5.0 dengan kelebihan yang dapat mendukung control Activex dan mulai menghapus atau menghilangkan dukungan terhadap proses 16 bit. Dan versi yang dipakai dalam modul ini adalah Visual Basic versi 6.0 yang dikemas dalam satu paket Microsoft Visual Studio 6.0.

Visual Basic 6.0 ialah bahasa pemrograman *event-driven* yang berasal dari BASIC. *Event driven* artinya program menunggu sampai adanya respons dari pemakai berupa kejadian tertentu, misalnya tombol diklik atau menu dipilih. Ketika event terdeteksi, event yang berhubungan akan melakukan aksi sesuai dengan kode yang diberikan. Ada tiga edisi yang dikeluarkan Microsoft, yaitu:

### 1. Standard Edition

Standard Edition sangat merekomendasikan bagi pemula yang ingin mempelajari Visual Basic 6.0 dan mempunyai fasilitas sebagai berikut:

- Kemampuan aplikasi 32 bit yang berjalan di Microsoft Win 9x dan Win NT untuk pemula.
- Terdiri dari control seperti grid, tab, dan Data Bound.
- Memuat Learn Visual Basic Now dan Online Help.
- Microsoft Developer Network CD berisi dokumentasi.

### 2. Professional Edition

Professional Edition umumnya digunakan oleh para profesional yang sudah cukup mendalami Visual Basic 6.0. Tidak terlalu banyak perbedaan dengan Standard Edition, hanya ada beberapa tambahan, diantaranya:

- ActiveX Control, termasuk Internet Control
- IIS (Internet Information Server)

- Dynamic HTML Page Designer

### 3. Enterprise Edition

Lebih ditekankan untuk membuat aplikasi yang bersifat *server based*, tapi program-program aplikasi standard dapat berjalan dengan baik jika menggunakan versi ini.

Fasilitas tambahan antara lain:

- Application Performance Explorer
- IIS (Internet Information Server)
- Support for Microsoft Transaction Server 2.0
- SQL Debugging
- Visual Component Manager
- Visual Database Tool

Visual Basic merupakan salah satu *Development Tool* yaitu alat bantu untuk membuat berbagai macam program komputer, khususnya yang menggunakan sistem operasi Windows. Visual Basic merupakan salah satu bahasa pemrograman komputer yang mendukung object (*Object Oriented Programming* = OOP). Sayangnya, Visual Basic sampai saat ini hanya dapat berjalan diatas lingkungan sistem operasi Windows. Untuk kalangan sistem operasi yang lain seperti Linux misalnya, Visual Basic masih belum bisa berjalan optimal walaupun saat ini sudah mulai dirintis sebuah framework berbasis .NET agar dapat menjalankan aplikasi VB.NET diatas platform Linux.

### C. Mengenal Integrated Development Environment (IDE)

Kepopuleran Visual Basic sebenarnya datang dari lingkungannya yang sering disebut *Integrated Development Environment* atau IDE. IDE membantu membangun sebuah aplikasi besar, menulis sebuah program, menjalankan program, dan menghasilkan sebuah *executable file*. Executable File yang dihasilkan oleh Visual Basic bersifat independen, dan karena itu file tersebut dapat dijalankan pada komputer tanpa harus menginstall Visual Basic.

Pemrograman visual merupakan dimensi baru dalam pembuatan aplikasi karena dapat langsung menggambarkan objek-objek ke layar sebelum dieksekusi. Dalam lingkungan pengembangan visual, sekarang objek yang anda buat hasilnya langsung tampil di layar. Objek yang

dibuat itu akan sama hasilnya pada saat program dijalankan. Dengan demikian tidak perlu lagi melakukan perubahan kode program secara manual. Setelah semua objek diletakkan dalam suatu form, maka semua atribut objek tersebut akan disimpan dalam suatu kode program yang dapat langsung dijalankan.

#### 1. Menjalankan IDE

Untuk membuka program Visual Basic 6.0 (yang dikemas dalam Microsoft Visual Studio 6.0), ada beberapa cara yang bisa dilakukan diantaranya sebagai berikut:

- ✓ Klik Start – Program - Microsoft Visual Studio 6.0 – Microsoft Visual Basic 6.0. Maka pada layar awal akan muncul tampilan seperti pada Gambar 1..
- ✓ Dengan membuat *short cut* pada jendela desktop dan untuk memulainya cukup melakukan *double-click* pada short cuttersebut.
- ✓ Ketika Visual Basic diinstall, file-file Visual Basic (\*.frm, \*.bas, \*.vpb) di daftarkan pada sistem operasi Window, karena itu untuk memulai visual basic anda dapat melakukan double-click pada file-file tersebut

#### 2. Memilih Tipe Project

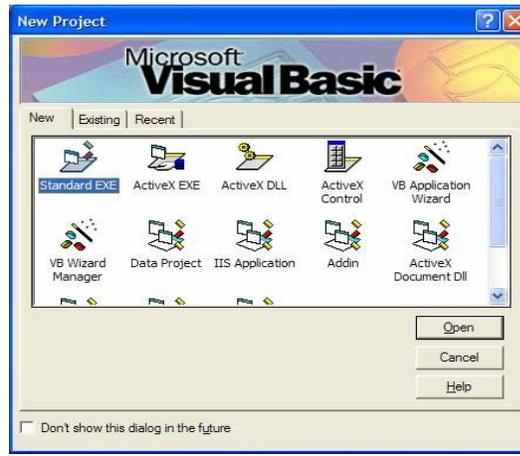
Visual Basic menyediakan banyak jenis modul aplikasi. Beberapa pilihan yang terdapat pada kotak dialog New Project adalah sebagai berikut.

- ✓ Standard EXE : membuat aplikasi Visual Basic Standar
- ✓ Active EXE : membuat aplikasi ActiveX
- ✓ Active DLL : membuat library ActiveX
- ✓ ActiveX Control : membuat kontrol ActiveX
- ✓ VB Application Wizard : membuat aplikasi dengan bantuan Wizard
- ✓ VB Wizard Manager : pusat pengelolaan Wizard Visual Basic
- ✓ IIS Application : membuat aplikasi IIS (Internet Information Server)

✓ DHTML Application : membuat aplikasi DHTML (Dynamic Hypertext Mark-up Language) untuk internet.

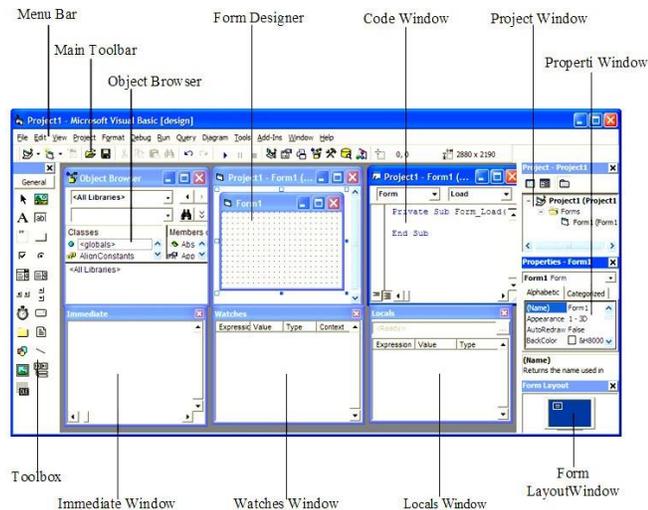
Tab Existing untuk menampilkan aplikasi yang sudah ada dan Tab Recent menampilkan aplikasi yang pernah dibuka terakhir kali.

Untuk memulai program standar pilihlah Standard EXE, kemudian klik pada tombol Open. Setelah itu akan muncul window Project1- Microsoft Visual Basic seperti pada gambar 1.2.



Gambar 1. Kotak Dialog New Project

Sekarang kita akan mengenal bagian-bagian dari IDE ( Integrated Development Environment ) yang kita gunakan seperti pada gambar 1.2



Gambar 2. IDE Visual Basic

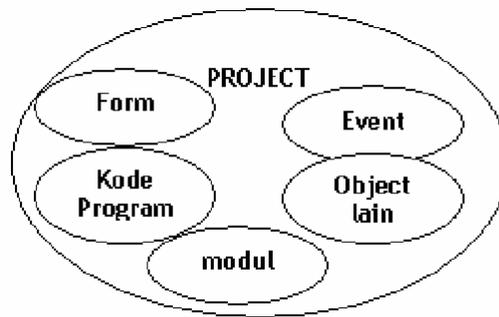
#### D. Komponen IDE dalam VB 6.0

Jendela IDE Visual Basic memiliki sejumlah menu, toolbar, dan window seperti yang terlihat pada gambar 1.2. Berikut ini akan dijelaskan kegunaan masing-masing item :

##### 1. Project Window

Jika membuat program aplikasi, akan terdapat jendela project yang berisi semua file yang dibutuhkan untuk menjalankan program aplikasi Visual Basic yang dibuat. Pada jendela project terdapat tiga icon yaitu icon View Code untuk menampilkan jendela editor, icon View Object untuk menampilkan bentuk formulir (Form) dan icon Toggle Folders digunakan untuk menampilkan folder (tempat penyimpanan file).

Pertama kali ketika menggunakan program Visual Basic maka komponen project yang akan diload, selanjutnya adalah menambah form-form atau membuat modul atau mungkin membuat kode program. Secara lebih sederhana posisi project dalam setiap komponen yang ada adalah seperti terlihat pada gambar dibawah ini :

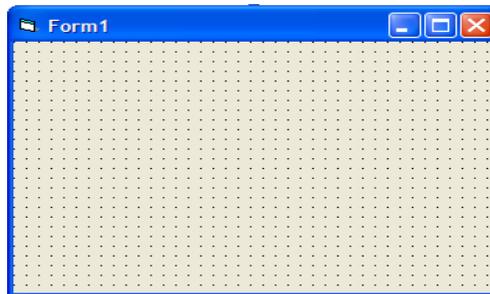


*Gambar 3. Keterkaitan Project dengan komponen lainnya*

##### 2. Form Designer

Form digunakan ketika akan meletakkan object-object apa saja yang akan digunakan dalam program, object-object yang terdapat dalam toolbox, diletakkan dan didesain dalam bagian form. Form sebenarnya adalah suatu objek yang dipakai sebagai tempat bekerja program aplikasi.

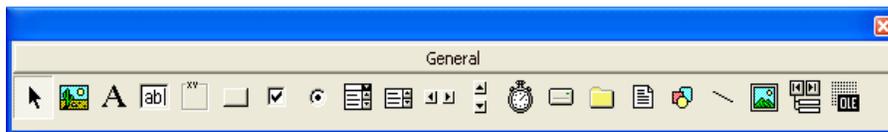
Secara otomatis akan tersedia form yang baru jika kita membuat program aplikasi baru, yaitu dengan nama Form1. Umumnya dalam suatu form terdapat garis titik-titik yang disebut dengan Grid.



Gambar 4. Form Kosong

### 3. Toolbox

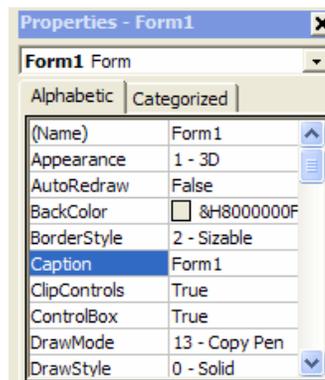
Toolbox adalah kotak alat yang berisi icon-icon untuk memasukkan objek tertentu ke dalam jendela form. Kita dapat memodifikasi toolbox, misalnya menambah komponen icon dengan cara melakukan klik kanan pada toolbox lalu memilih Components atau Add Tab.



Gambar 5. Kumpulan Toolbox

### 4. Properties Window

Properties digunakan untuk menentukan setting suatu objek. Suatu objek biasanya mempunyai beberapa properti yang dapat diatur langsung dari jendela Properties atau lewat kode program.



Gambar 6. Property untuk Form 1

5. Code Window

Digunakan untuk menulis kode program yang menentukan tingkah laku dari form dan objek-objek yang ada pada aplikasi bersangkutan. Kode program adalah serangkaian tulisan perintah yang akan dilaksanakan jika suatu objek dijalankan. Kode program ini akan mengontrol dan menentukan jalannya suatu objek.

6. Module

Module dapat disejajarkan dengan form, tetapi tidak mengandung object. Module dapat berisi kode-kode program atau procedure yang dapat digunakan dalam program aplikasi.

7. Color Pallete Window

Digunakan untuk menentukan warna dari suatu objek seperti warna dari objek yang sedang aktif saat ini.

8. Form Layout Window

Menunjukkan bagaimana peletakan sebuah form akan ditampilkan pada saat dijalankan.

9. Immediate Window

Digunakan untuk memasukkan ekspresi untuk melihat hasilnya dengan menggunakan perintah "print" atau "?". Jendela ini biasanya digunakan bersama watch window pada saat sebuah program di-debug. Short-cut untuk jendela ini adalah Ctrl-G.

10. Object Browser

Digunakan untuk menyelusuri external libraries sehingga anda dapat mempelajari objek-objek dan properti, kejadian (events), dan method yang dimilikinya. Short-cut untuk jendela ini adalah F2.

11. Locals Window

Akan aktif hanya pada saat program di jalankan, berisi nilai dari sejumlah variabel yang bersifat lokal pada sebuah prosedur atau module.

12. Watch Window

Digunakan untuk memonitor nilai dari suatu variabel baik bersifat lokal maupun global.

### 13. Call Stack Window

Hanya ditampilkan jika suatu program yang dijalankan dihentikan untuk sementara (break) dan menekan Ctrl-L. Menunjukkan semua prosedur yang menunggu prosedur aktif telah selesai. Jendela ini penting untuk proses debugging untuk mengetahui jalur eksekusi program hingga sampai situasi sekarang.

## E. Menu, Toolbar dan ToolBox

### 1. Menu

Menu yang akan dibahas secara garis besar, sebab sebagian dari menu berkaitan dengan Visual Basic lanjutan. Berikut ini akan dibahas mengenai menu yang sifatnya umum dan sering dipakai.

- ✓ *Menu File* berisi perintah antara lain, *open* untuk membuka suatu project dan *save* untuk menyimpan project atau form, *make* digunakan untuk mengkompile project menjadi Executable file.
- ✓ *Menu Edit* berkaitan dengan perintah-perintah editing seperti *cut*, *copy*, *paste*, *find*, *replace*, *undo* dan *redo*.
- ✓ *Menu View* digunakan untuk menampilkan window-window pada IDE Visual Basic yang telah dijelaskan pada gambar 1.3.
- ✓ *Menu Project* digunakan untuk menambahkan objek-objek baru seperti pada form, standart module (bas), class module, User control module, dll.
- ✓ *Menu Format* digunakan untuk mengatur posisi dan ukuran satu atau beberapa kontrol.
- ✓ *Menu Debug* berisi perintah-perintah yang digunakan pada saat program sedang di debug. Aplikasi yang dieksekusi tahap demi tahap, nilai dari variabel yang bersangkutan ditampilkan, dan menambahkan *break point* untuk memonitor jalannya program.
- ✓ *Menu Run* berisi perintah-perintah untuk menjalankan aplikasi yang dibangun.

### 2. Toolbar

Visual Basic memiliki sejumlah toolbar. Toolbar-toolbar tersebut dapat kita letakkan pada posisi sebelah atas pada IDE Visual Basic atau

dibiarkan menjadi window di dalam IDE Visual Basic. Toolbar-toolbar tersebut antara lain :

- ✓ *Standard Toolbar* berisi tool yang digunakan untuk perintah-perintah seperti membuka atau menyimpan sebuah project.



Gambar 7. Standar Toolbar

- ✓ *Debug Toolbar* digunakan berisi perintah seperti yang terdapat pada menu debug, jika toolbar tersebut tidak terdapat pada IDE, anda dapat memilih menu view - toolbars untuk



menampilkan toolbar tersebut.

Gambar 8. Debug Toolbar

- ✓ *Edit Toolbar* berfungsi pada saat kita mengedit sebuah kode, menambahkan break point, dan bookmarks.



Gambar 9. Edit Toolbar

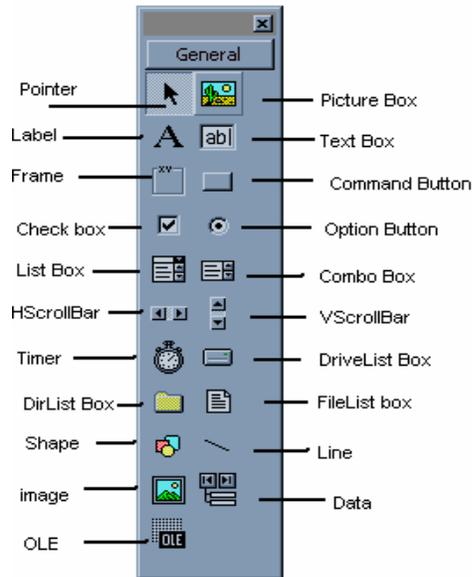
- ✓ *Form Editor Toolbar* digunakan untuk mengatur posisi control yang ada pada form.



Gambar 10. Form Editor toolbar

### 3. ToolBox

Selain Toolbar, Visual Basic juga mempunyai Jendela Toolbox. Jendela ini berisi sejumlah control yang digunakan untuk mendesain antar muka (*User Interface*) yang akan diletakkan pada Form. Berikut ini akan dijelaskan masing-masing dari control yang ada pada toolbox :



Gambar 11. Form Editor toolbar

- ✓ *Pointer* bukan sebuah kontrol, pointer digunakan untuk memilih control yang sudah ada pada sebuah form.
- ✓ *Picture Box Control*, control ini digunakan untuk menampilkan gambar seperti BMP, DIB (bitmap), ICO (icon), CUR (cursor), WMF (metafile), EMF (enhanced metafile), GIF, and JPEG.
- ✓ *Label Control*, digunakan sebagai text static yang tidak bisa diedit, biasanya digunakan untuk memberi nama pada control yang lain seperti TextBox.
- ✓ *TextBox Control*, control yang berisi string dari karakter yang dapat diisi oleh user.
- ✓ *Frame Control*, digunakan untuk mengelompokkan control-control yang lain.
- ✓ *Command Button Control*, command button hampir muncul pada setiap form, biasanya digunakan untuk menangkap kejadian berupa mouse click.
- ✓ *Check Box Control*, control ini digunakan untuk pilihan ya/tidak atau benar/salah.
- ✓ *Option Button Control*, digunakan dalam suatu group, dimana seorang user dapat memilih salah satu saja dalam suatu group.

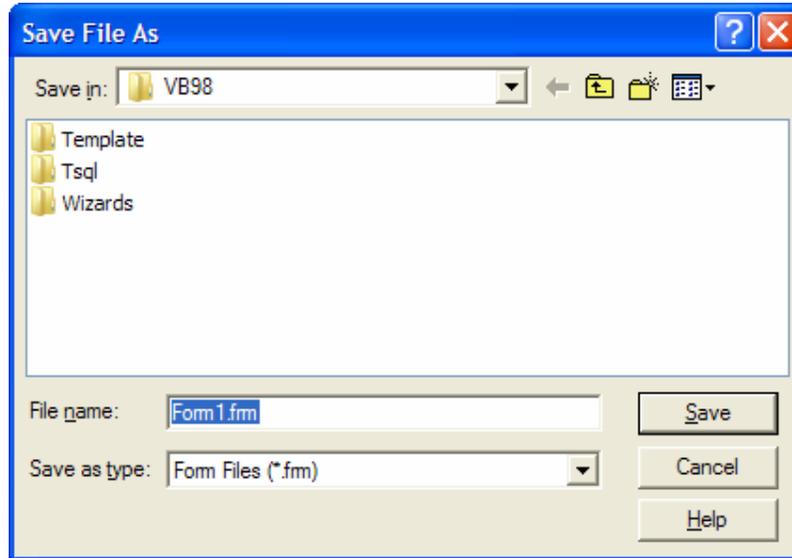
- ✓ *ListBox Control*, berisi sejumlah item dimana seorang user dapat memilih salah satu atau beberapa item.
- ✓ *ComboBox Control*, control ini merupakan gabungan antara TextBox dan ListBox Control, dimana daftar item hanya akan ditampilkan jika user men-click pada panah kebawah disebelah kanan Control tersebut.
- ✓ *HscrollBar* dan *VscrollBar Control*, kedua control ini digunakan untuk ScrollBar.
- ✓ *Timer Control*, merupakan control yang tidak terlihat pada saat program dijalankan. Control tersebut membangkitkan kejadian pada selang waktu tertentu.
- ✓ *DriveList Box*, *DirList Box*, dan *FileList Box Control*, ketiganya digunakan untuk membuat suatu dialog untuk sistem file. Control tersebut jarang digunakan sebab ada Common dialog Control yang akan kita bicarakan nanti sebagai ganti dari ketiga control diatas.
- ✓ *Shape* dan *Line Control*, digunakan untuk memperindah tampilan dari suatu form.
- ✓ *Image Control*, control ini mirip dengan PictureBoxControl tetapi digunakan sebagai container control- control yang lain.
- ✓ *Data Control*, merupakan kunci dari data binding pada Visual Basic. Dengan menggunakan control ini, kita dapat menghubungkan sebuah database pada Visual Basic.
- ✓ *OLE Control*, digunakan sebagai tempat untuk program eksternal yang ada pada windows seperti spread sheet yang dihasilkan oleh Microsoft Excel. Dengan menggunakan control tersebut kita dapat menampilkan program lain pada sebuah aplikasi.

#### F. Menyimpan Project

Langkah-langkah menyimpan:

- ✓ Pada menu File, klik perintah Save Project As kemudian akan muncul Kotak dialog File Project seperti terlihat pada gambar 1.12. Melalui kotak dialog tersebut dapat menyimpan program Visual Basic tersebut.

- ✓ Pilih direktori kerja anda misalkan D:\VB\Latihan01 dengan mengklik pada kontrol combo box.
- ✓ Simpan form dengan nama Form1, kemudian klik tombol Save. Simpan Project dengan nama Project1, kemudian klik tombol Save.



Gambar 12. Save File Dialog

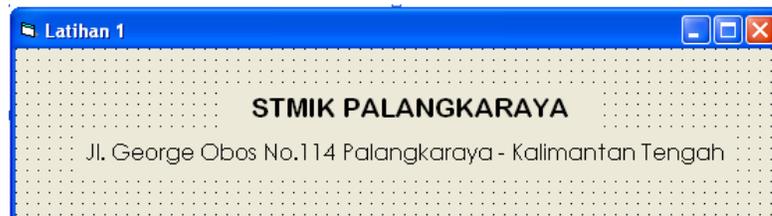
#### G. Menjalankan dan Menghentikan Program

Langkah-langkah menjalankan/menghentikan program:

- ✓ Klik tombol *Start* (▶) pada *ToolBar* atau dengan menekan tombol F5 atau juga dapat menggunakan *Start* pada menu *Run*.
- ✓ Sedangkan untuk menghentikan program yang sedang berjalan dapat menggunakan tombol *End* (■) pada *ToolBar* atau menggunakan *End* pada menu *Run*.

### Soal Latihan

1. Letakkan object Label pada Form. Atur property Form dan Label seperti terlihat pada gambar di bawah ini:



2. Jalankan program yang Anda buat, lihat hasilnya
3. Hentikan program kemudian simpan program yang Anda buat
4. Untuk menyimpan pilih menu File - Save As Project pada. Pada saat project tersebut disimpan ada dua macam file yaitu :
  - File Project (\*.vbp) beri nama prLat01.vbp
  - File Form1 (\*.frm) beri nama frLat01.frm

## BAB II

### TIPE DATA, VARIABEL dan KONSTANTA, OPERATOR dan KONTROL PROGRAM

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman Visual Basic dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
1. Mengenal tipe data, Variabel dan Konstanta dalam Visual Basic	<ul style="list-style-type: none"><li>Mengetahui macam-macam tipe data beserta rangenya</li><li>Memahami istilah Variabel Lokal dan Variabel Global</li><li>Memahami istilah konstanta intrinsik dan konstanta yang dibuat oleh pemrogram</li></ul>
2. Mengenal Operator dan Kontrol Program dalam Visual Basic	<ul style="list-style-type: none"><li>Mengetahui macam-macam operator dalam Visual Basic</li><li>Mengetahui macam-macam kontrol program dalam Visual Basic</li></ul>

#### A. Tipe Data dalam Visual Basic

Data memiliki tipe yang berbeda-beda dan biasanya data dikelompokkan pada kelompok yang sejenis agar tidak terjadi operasi matematika diantara data yang berbeda jenis, contohnya kita tidak bisa menjumlahkan nilai suatu jarak (Km) dengan Massa (Kg) karena keduanya memiliki tipe data yang berbeda. Begitu juga dalam suatu program, setiap nilai harus dikelompokkan pada jenis-jenis tertentu yang disebut dengan tipe data.

Microsoft Visual Basic menyediakan beberapa tipe data seperti Integer, Long, Single, Double, Currency, String, Byte, Boolean, Date, Object dan Variant.

Tabel 1. Tipe Data dalam Visual Basic

Type	Range
Integer	-32768 s/d 32767
Long	-2147483.648 s/d 2147483647
Single	Negatif : -3.40282E38 s/d -1.401298E-45 Positif : 1.401298E-45 s/d 3.402823E38
Double	Negatif : -1.7976931348232E308 s/d -4.94065645841247e-324 Positif : 4.94065645841247e-324 s/d 1.79769313486232E308
Currency	-922337203685477.5808 s/d 922337203685477.5807
String	0 s/d 2 milyar karakter
Byte	0 s/d 255
Boolean	True (benar) atau false (salah)
Date	1 January 100 s/d 31 Desember 9999
Object	Referensi Object
Variant	Null, Error dan tipe seluruh tipe data yang lain

Keterangan :

Integer, Long : tipe data untuk angka bulat

Single, Double : tipe data untuk angka pecahan/desimal  
Currency : tipe data untuk angka mata uang

String : tipe data untuk teks Boolean :

tipe data logika ( True/False)  
Date : tipe data waktu/tanggal

Object : tipe data untuk sebuah objek misalnya gambar Variant : tipe data variant

## B. Variabel dalam Visual Basic

Variabel digunakan untuk menyimpan nilai atau data yang dimiliki program aplikasi yang kita buat. Nilai yang ditampung atau disimpan oleh suatu variabel dapat berubah selama program berjalan. Misalnya kita bisa menyimpan nilai ujian mid di variabel A dan nilai ujian akhir di variabel B dan setiap mahasiswa nilainya pasti berbeda

### 1. Deklarasi Variabel

Deklarasi variabel harus diletakkan sebelum baris-baris perintah yang menggunakan variabel tersebut. Ada dua cara dalam pendeklarasian variabel, yaitu Deklarasi Eksplisit dan Deklarasi Implisit.

Deklarasi Eksplisit dilakukan dengan cara menuliskan sebuah kata kunci diikuti nama variabel serta tipe datanya. Aturan penulisan deklarasi variabel dengan cara eksplisit sbb :

<katakunci> <namavariabel> As <TipeData>

Tabel 2. Kata Kunci Deklarasi Variabel Secara Eksplisit

Kata Kunci	Penggunaan
Static	Berlaku pada level prosedur
Dim	Berlaku pada level prosedur dan modul
Private	Berlaku pada level modul/form
Public	Berlaku pada level modul dan aplikasi
Global	Berlaku pada level modul dan aplikasi

Contoh deklarasi variabel secara eksplisit adalah :

```
Dim Nama As String
Private Nilai As Integer
Static Jumlah As Integer
Public Alamat As Variant
```

Deklarasi Implisit dilakukan tanpa menggunakan kata kunci. Pada deklarasi implisit, sebuah variabel langsung digunakan disertai sebuah karakter khusus yang menandakan tipe data variabel tersebut.

Tabel 3. Beberapa Karakter Untuk Deklarasi Variabel Implisit

Tipe Data	Karakter
Integer	%
Long	&
Single	!
Double	#
Currency	@
String	\$

Contoh deklarasi secara implisit :

```
Judul$ = "Cepat Mahir Visual Basic 6.0"
Harga% = 25000
```

Sebaiknya setiap variabel yang digunakan dalam kode program dideklarasikan terlebih dahulu, agar alur jalannya program lebih terkontrol dan mudah dipahami.

## 2. Aturan Penamaan Variabel

Visual Basic 6.0 tidak memperhatikan penulisan huruf besar atau kecil. Variabel dengan nama Alamat akan dianggap sama dengan ALAMAT atau aLaMAT

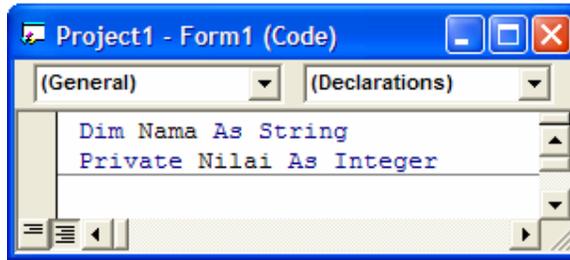
Beberapa aturan yang digunakan dalam penamaan variabel adalah sebagai berikut :

- ✓ Harus unik, tidak boleh ada variabel dengan nama sama pada satu ruang lingkup yang sama.
- ✓ Tidak boleh lebih dari 255 karakter, tetapi hanya 40 karakter pertama yang dianggap sebagai nama variabel. Karakter sisanya diabaikan.
- ✓ Tidak boleh menggunakan spasi, tanda +, -, \*, /, <, >, :, =, #, koma dll
- ✓ Harus dimulai dari huruf, bukan angka atau karakter lainnya
- ✓ Tidak boleh menggunakan reserved word milik Visual Basic 6.0

## 3. Ruang Lingkup Variabel

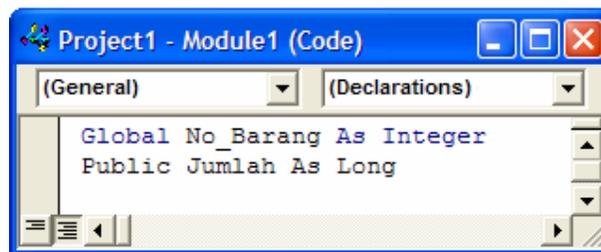
Ada tiga level ruang lingkup variabel, yaitu : level lokal, level form/modul, dan level global.

- ✓ Variabel level lokal adalah variabel yang hanya dikenali dan dapat digunakan dalam prosedur tempat variabel tersebut dideklarasikan. Prosedur di bagian lain tidak dapat mengakses variabel tersebut. Variabel ini dideklarasikan menggunakan perintah Dim, Private, dan Static.
- ✓ Variabel dengan level form/modul dideklarasikan dibagian general declarations dalam jendela kode program form/modul. Nilai variabel ini dapat dikenali dan dipakai pada semua prosedur yang ada dalam form/modul tersebut. Deklarasi variabel ini juga menggunakan pernyataan Dim. Perbedaan dengan variabel lokal adalah tempat deklarasi variabel tersebut. Contoh deklarasi variabel level form/modul :



Gambar 13. Deklarasi variabel level form/modul

- ✓ Variabel level global (aplikasi) dideklarasikan di bagian general declarations dalam jendela kode program form/modul menggunakan pernyataan Public atau Global. Variabel ini dapat dipanggil dan dipakai oleh semua form/modul dalam program aplikasi yang kita buat, dan juga dapat dipanggil dan dipakai oleh seluruh prosedur yang ada dalam form/modul tersebut. Contoh deklarasi variabel level global adalah sbb:



Gambar 14. Deklarasi variabel level global (aplikasi)

### C. Konstanta dalam Visual Basic

Konstanta adalah sejenis variabel yang nilainya tetap dan tidak dapat diubah selama program berjalan. Ada dua jenis konstanta yaitu konstantaintrinsik dan konstanta yang dibuat pemrogram.

Konstanta intrinsik adalah konstanta yang sudah tersedia secara otomatis dalam Visual Basic 6.0. Contoh konstanta intrinsik adalah vbYes, vbModal, dan lain-lain. Pemrogram juga dapat membuat sebuah konstanta sendiri dengan cara mendeklarasikannya terlebih dahulu.

#### 1. Deklarasi Konstanta

Aturan penulisan deklarasi konstanta adalah sbb:

```
[<KataKunci> Const <NamaKonstanta> [As <TipeData>] = <nilai>
```

Kata Kunci : Pilihan kata kunci yang dapat digunakan pada deklarasi konstanta adalah Private dan Public

Nama Konstanta : Nama konstanta yang dideklarasikan

Tipe Data : Tipe data dari konstanta tersebut

Bagian yang diberi tanda kurung siku "[" dan "]" pada aturan penulisan di atas berarti boleh dituliskan, dan boleh juga tidak dituliskan.

Contoh deklarasi konstanta :

```
Const NamaUsaha = "CV. Digdoyo Cellular"  
Public Const Jumlah = 500000  
Private Const Total = 5  
Const Nama As String = "Digdoyo P. Nurkholis"
```

## 2. Ruang Lingkup Konstanta

Konsep ruang lingkup konstanta hampir sama dengan konsep ruang lingkup variabel. Yaitu terdiri dari level lokal, level form/modul dan level global. Ketiga level tersebut hanya dibedakan berdasarkan kata kunci yang digunakan atau letak deklarasi konstanta.

## D. Operator

Operator adalah suatu tanda yang digunakan untuk menghubungkan satu variabel atau konstanta dengan variabel atau konstanta lain dengan tujuan melakukan berbagai manipulasi dan pengolahan data. Pada Ms. Visual Basic 6.0 terdapat bermacam-macam operator :

### 1. Operator Penugasan (assignment)

Operator penugasan ditimbulkan dengan tanda sama dengan (=) dan berfungsi untuk memasukkan suatu data ke dalam suatu variabel.

Contoh :

```
A = 12  
A = A+1  
Text1.text = "STMIK Palangkaraya"
```

## 2. Operator Aritmatika

Operator aritmatika digunakan untuk melakukan operasi aritmatika. Operator aritmatika mempunyai hirarki paling tinggi dibanding operator pembanding dan operator logika. Penulisan operator aritmatika dengan hirarki dari paling tinggi ke paling rendah, sebagaiberikut :

Tabel 4. Operator Aritmatika

Operator	Operasi
$\wedge$	Pemangkatan
-	Tanda Negatif
*, /	Perkalian & Pembagian
\	Pembagian Integer
Mod	Modulus (Sisa hasil bagi)
+, -	Penambahan dan Pengurangan
+, &	Penggabungan String

Contoh :

$$A = (2^4) * 5, B = 25 \text{ Mod } 4$$

## 3. Operator Pembanding

Operator pembanding digunakan untuk membandingkan suatu data (ekspresi) dengan data (ekspresi) lain dan menghasilkan nilai logika (boolean) Benar atau Salah. Tentu saja antara kedua data yang dibandingkan harus mempunyai tipe data yang sama. Bentuk dari operator relasional/pembanding seperti pada tabel berikut :

Tabel 5. Operator Pembanding

Operator	Operasi
=	Sama dengan
$\langle \rangle$	Tidak sama dengan
<, >	Lebih kecil, lebih besar,
<=, >=	lebih kecil sama dengan, lebih besar samadengan
Like	Mempunyai ciri yang sesuai
Is	Sama referensi objek

Contoh :

$$A \langle \rangle B, B > C, C < A$$

#### 4. Operator Logika

Operator logika digunakan untuk mengekspresikan satu atau lebih data (ekspresi) logika (boolean) yang menghasilkan data logika baru. Tabel operator logika dengan hierarki dari atas ke bawah adalah sebagai berikut :

Tabel 6. Operator Logika

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

Contoh :

`X > 5 And X < 10, X = 3 Or x = 5 dst`

#### E. Kontrol Program

Ada banyak perintah di Visual Basic 6.0. yang digunakan untuk mengontrol jalannya program yang akan dibuat. Fungsi kontrol program ini dibentuk dengan logika pemograman yang nantinya akan berguna sebagai validasi terhadap data-data yang masuk maupun yang keluar dari program tersebut.

##### 1. Pencabangan On Error

Pencabangan ini dipakai untuk penanganan kesalahan (error) dalam program. Bentuk penulisannya ada tiga macam, yaitu : On Error GoTo

<baris>, On Error Resume Next, On Error GoTo 0

##### ❖ On Error GoTo <baris>

Menyebabkan penanganan error aktif, sehingga jika terjadi kesalahan program maka kesalahan yang terjadi tersebut akan menunjuk ke <baris> untuk proses selanjutnya.

Contoh :

```
Private Sub Command1_Click()  
    On Error GoTo Pesan  
    .....  
    .....  
    Pesan:  
        MsgBox "Ada Kesalahan  
        Program"  
End Sub
```

Kesalahan apapun yang terjadi, baik kesalahan dalam program maupun kesalahan dalam logika pemrograman, maka kesalahan tersebut tidak akan terlihat dimana letak kesalahannya karena kesalahan apapun yang muncul maka kesalahan tersebut akan memunculkan tulisan pada baris <Pesan>

❖ On Error Resume Next

Menyebabkan jika terjadi kesalahan, program akan melanjutkan ke perintah yang mengikuti (dibawah) perintah yang salah.

Contoh :

```
Private Sub Command1_Click()  
    On Error Resume Next  
    MsgBox "Pesan dikerjakan setelah  
    mengabaikan Error"  
End Sub
```

❖ On Error GoTo 0

Menyebabkan perangkat kesalahan yang sebelumnya dipasang menjadi tidak aktif (membatalkan penanganan kesalahan).

Contoh :

```
Private Sub Command1_Click()  
    On Error GoTo 0  
    MsgBox "Jika Error lanjutkan ke baris  
    berikutnya"  
End Sub
```

## 2. Statement End

Statement End dipakai untuk memaksa kontrol program berhenti dari suatu procedure atau suatu blok program. Beberapa contoh bentuk pernyataan End.. adalah sebagai berikut :

❖ End

Statement End biasanya digunakan untuk mengakhiri penggunaan suatu program, misalkan pada program terdapat suatu tombol command button bercaption "Keluar" maka isi untuk perintah command button keluar tersebut adalah dengan "End". Contoh :

```
Private sub cmdExit_click()  
    End  
End sub
```

❖ End Function

Jika dalam program menggunakan function, terutama function yang akan dibentuk sendiri, maka diakhiri penulisan function tersebut harus ditutup dengan end function.

Contoh :

```
Private Function Tambah(a,b as Integer) as Integer
    Tambah = a + b
End Function
```

❖ End if

Eksresi End If digunakan untuk mengakhiri penggunaan ekspresi If.. Then... Else..., setiap perintah If harus ditutup atau diakhiri dengan End If

Contoh :

```
Private sub cmdOK_click()
    If Text1.Text = 1 then
        MsgBox "Nilainya Satu"
    Else
        MsgBox " Nilainya selain Satu"
    End If
End sub
```

Ekspresi If di atas akan menunjukkan jika nilai yang terdapat dalam object TextBox bernilai 1, maka akan muncul tulisan pesan "Nilainya Satu" selain itu muncul tulisan "Nilainya selain Satu".

❖ End Property

End property biasanya digunakan ketika akan bekerja dengan menggunakan class.

❖ End Select

Salah satu pernyataan percabangan yang bersyarat adalah menggunakan Select Case, diakhir pernyataan tersebut jangan lupa dituliskan End Select yang menyatakan akhir dari percabangan tersebut.

Contoh :

```
Select Case Text1.Text
  Case "1"
    MsgBox "Data anda satu"
  Case "2"
    MsgBox "Data anda dua"
End Select
```

❖ End Sub

Setiap objek yang diletakkan dalam sebuah form akan membentuk suatu modul atau procedure sendiri, proses selanjutnya tergantung dari event yang akan dilakukan dengan objek tersebut. Diakhir pendeklarasian sebuah modul harus ditutupi dengan End Sub.

Contoh :

```
Private Sub Command1_Click()
  .....
  .....
  .....
End Sub
```

❖ End Type

Pendeklarasian End Type digunakan jika mendeklarasikan sekumpulan data bertipe record data, dan disetiap akhir pendeklarasian tersebut harus diakhiri dengan End Type.

Contoh :

```
Type DATASISWA
  NIM as string
  NAMA as string
  Alamat as string
End Type
```

❖ End With

Pendeklarasian End With digunakan jika kita ingin mengakhiri penggunaan with diawal sebuah pendeklarasian, perintah with dan end with dapat digunakan untuk menyingkat suatu penulisan objek yang berulang-ulang. Contoh berikut menyatakan object textbox yang digunakan dan beberapa atribut yang digunakan dalam object tersebut.

Contoh :

```
With Text1
    .Text = "Sani"
    .Font = Arial
    .FontBold = True
    .FontSize = 15
End With
```

### 3. Komentar Program

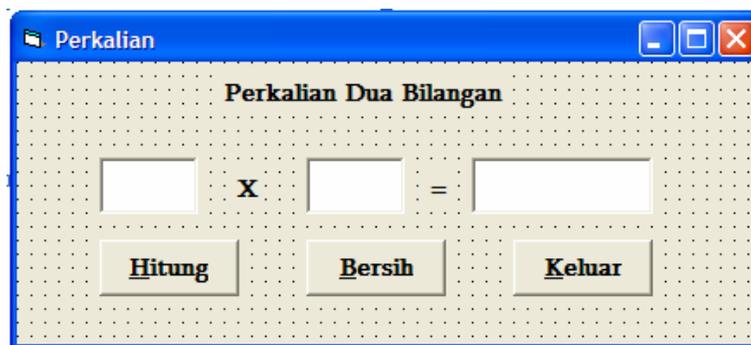
Komentar dapat ditambahkan pada suatu baris program dengan menuliskan tanda petik satu ( ' ) didepan statement yang ingin dinyatakan sebagai suatu komentar, sifat komentar ini tidak akan dikerjakan dan hanya sebagai komentar bagi program dan visual basic tidak akan menganggapnya sebagai kode sehingga tidak akan dijalankan.

Contoh :

```
Private Sub Command1_click()
    A = 3      'Set nilai A = 3
    B = 5      'Set Nilai B = 5
    C = A + B  'Nilai A dan B akan ditambahkan
               'dan disimpan dalam variabel C.
End Sub
```

### F. Contoh Program

1. Kita akan mencoba membuat program perkalian dua bilangan. Letakkan beberapa object yang ada dalam form, atur beberapa setting property yang ada seperti terlihat pada gambar di bawah ini:



Control	Property	Setting Value
Form1	Name	FrKali
	Caption	Perkalian
Label1	Name	LblJudul
	Caption	Perkalian Dua Bilangan
Label2	Name	LblKali
	Caption	X
Text1	Name	txtBill
	Text	Blank
Text2	Name	txtBil2
	Text	Blank
Text3	Name	txtHasil
	Text	Blank
Command1	Name	cmdHitung
	Caption	&Hitung
Command2	Name	cmdBersih
	Caption	&Bersih
Command3	Name	cmdKeluar
	Caption	&Keluar

Ketikkan kode program seperti berikut:

```

Private Sub cmdBersih_Click()
'Mengosongkan isi TextBox
txtBill1.Text = ""
txtBil2.Text = ""
txtHasil.Text = ""
End Sub

Private Sub cmdHitung_Click()
'Deklarasi Variabel Lokal
Dim a As Single
Dim b As Single
Dim Hasil As Single

a = txtBill1.Text
b = txtBil2.Text
Hasil = a * b
txtHasil.Text = Hasil
End Sub

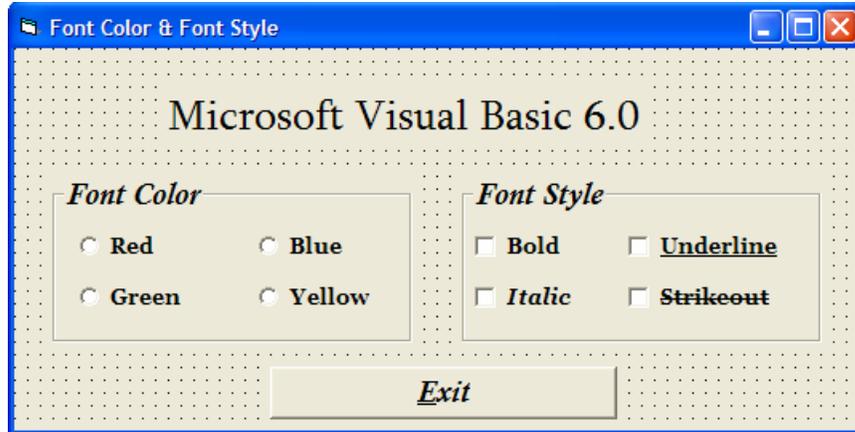
Private Sub cmdKeluar_Click()
'Mengakhiri Program
End
End Sub

```

Simpan program yang Anda buat dengan nama :

- File Project (\*.vbp) beri nama prjCoba01.vbp
- File Form1 (\*.frm) beri nama frmCoba01.frm

2. Membuat program untuk memilih font color dan font style. Letakkan beberapa object yang ada dalam form, atur beberapa setting property yang ada seperti terlihat pada gambar di bawah ini:



Atur property objek tersebut menjadi seperti berikut :

Kontrol	Property	Setting Value
Form1	Name	frFont
	Caption	Font Color & Font Style
Label1	Name	lblTeks
	Caption	Microsoft Visual Basic 6.0
Option1	Name	optRed
	Caption	Red
Option2	Name	optGreen
	Caption	Green
Option3	Name	optBlue
	Caption	Blue
Option4	Name	optYellow
	Caption	Yellow
Check1	Name	chkBold
	Caption	Bold
Check2	Name	chkItalic
	Caption	Italic
Check3	Name	chkUnder
	Caption	Underline
Check4	Name	chkStrike
	Caption	Strikeout
Frame1	Caption	Font Color
Frame2	Caption	Font Style
Command1	Name	cmdExit
	Caption	&Exit

Ketikkan kode program seperti berikut:

```
Private Sub optRed_Click()  
  'Membuat teks bewarna merah  
  lblTeks.ForeColor = vbRed  
End Sub  
  
Private Sub chkBold_Click()  
  'Membuat teks Tebal  
  If chkBold.Value = 1 Then  
    lblTeks.FontBold = True  
  Else  
    lblTeks.FontBold = False  
  End If  
End Sub  
  
'Untuk pilihan font color yang  
'lain silakan Anda coba membuat  
'kode programnya  
  
'Untuk pilihan font style yang  
'lain silakan Anda buat kode  
'programnya
```

Simpan program yang Anda buat dengan nama:

- File Project (\*.vbp) beri nama prCoba02.vbp
- File Form1 (\*.frm) beri nama frCoba02.frm

## Soal Latihan

- Buat program penghitungan dengan menggunakan beberapa operator. Atur tampilan seperti di bawah ini dengan property yang Anda tentukan sendiri.



- Simpan program yang Anda buat dengan nama:
  - File Project (\*.vbp) beri nama prLat02.vbp
  - File Form1 (\*.frm) beri nama frLat02.frm

## BAB III

### PROPERTY, METHOD dan EVENT

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
1. Mengetahui Property, Method dan Event dalam Visual Basic	<ul style="list-style-type: none"><li>Menjelaskan pengertian property dalam Visual Basic</li><li>Menyebutkan contoh-contoh property form dan objek pada Visual Basic</li><li>Menjelaskan dan memberi contoh Method dalam Visual Basic</li><li>Menjelaskan dan memberi contoh Event dalam Visual Basic</li></ul>
2. Menggunakan Property, Method dan Event pada Visual Basic	<ul style="list-style-type: none"><li>Membuat form (interface) sebagai antarmuka untuk sebuah program</li><li>Mengubah nilai property</li><li>Mengenal <i>Event-Driven programming</i>, dan menjalankan program.</li></ul>

Pemrograman Visual Basic menggunakan suatu teknik pemrograman OOP (Object Oriented Programming) yaitu suatu teknik pemrograman yang memodelkan program sebagai sekumpulan objek yang saling memiliki hubungan. Setiap kontrol (objek) memiliki properti, method dan event yang berbeda-beda.

#### A. Memahami Property, Method dan Event

##### 1. Properti

Properti adalah atribut-atribut yang melekat pada sebuah kontrol (objek) yang biasanya merupakan karakteristik penampilannya seperti warna, jenis huruf, ukuran dan sebagainya

Properti sebuah kontrol (objek) dapat diubah pada saat memprogram (menggunakan jendela properti) atau pada saat program dijalankan (menggunakan kode-kode program pada jendela kode).

Contoh : Command1.Caption="OK"

## 2. Method

Method adalah aksi atau perbuatan yang bisa dimiliki oleh kontrol (objek) sehingga user (programmer) dapat memakainya untuk memanipulasi sesuatu. Method tergantung dari instruksi yang diberikan oleh programmer melalui penulisan kode

Contoh : Command1.Click

## 3. Event

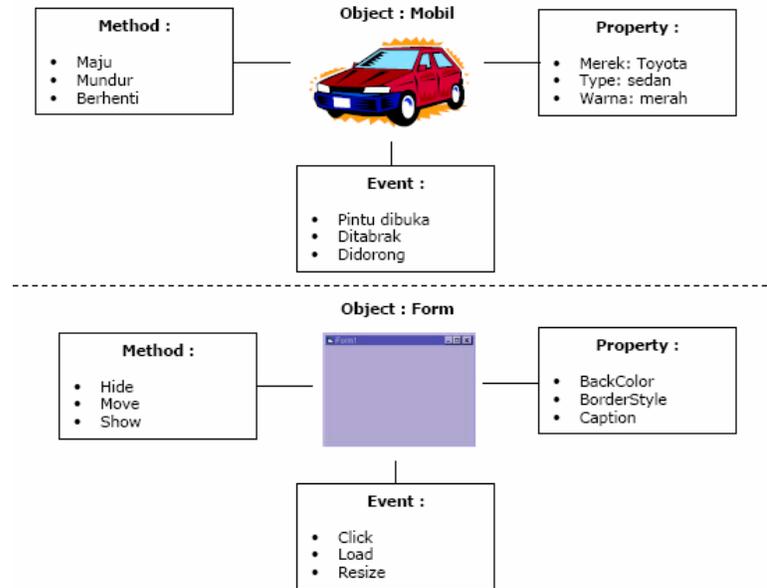
Event adalah kejadian atau segala sesuatu yang dapat dialami oleh sebuah objek. Suatu kontrol (objek) dapat memiliki banyak event.

Contoh : Command1\_Click( )

Secara ringkasnya dapat dijelaskan sebagai berikut :

- Property : karakteristik yang dimiliki object
- Method : aksi yang dapat dilakukan oleh object
- Event : kejadian yang dapat dialami oleh object

Sebagai ilustrasi anda dapat menganggap sebuah mobil sebagai obyek yang memiliki property, method dan event. Perhatikan gambar berikut :



Gambar 15. Ilustrasi untuk menggambarkan Property, Method dan Event

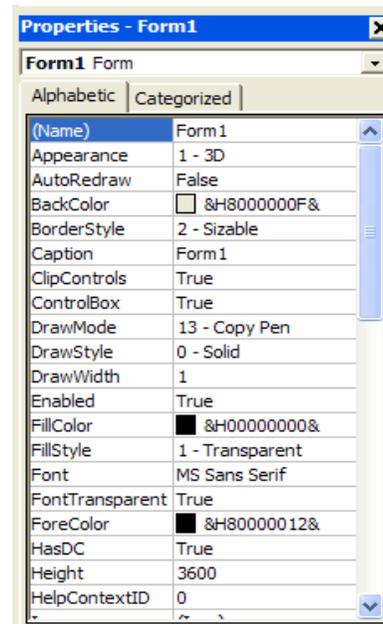
## B. Property

Form merupakan objek utama dalam perprograman Visual Basic, sebab form merupakan tempat dimana objek-objek lain seperti kontrol diletakkan seperti terlihat pada gambar 3.2. Form dan Kontrol merupakan objek-objek yang ada dalam Visual Basic yang memiliki properti, metode dan dapat menangkap suatu kejadian (*event*). Pada bagian ini akan dibahas tentang properti, metode (*methods*), dan kejadian yang berhubungan dengan objek-objek tersebut.



Gambar 16. Tampilan Form dari program Perkalian

Visual Basic mendukung properti-properti yang hanya dimiliki oleh sejumlah objek tertentu saja. Akan tetapi ada beberapa properti-properti yang sifatnya umum yaitu properti-properti ini hampir dimiliki oleh semua objek yang ada dalam Visual Basic. Properti-properti umum tersebut antara lain properti *name*, *top*, *left*, *height*, *weight*, *Foreground*, *Background*, *Font*, *Caption*, *Text*, *Enable*, *Visible*, *TabStop*, *TabIndex*, dan properti-properti umum lainnya.



## 1. Property Name

Semua objek dalam Visual Basic mempunyai properti *Name*. Properti *Name* digunakan untuk memberikan identitas pada objek yang dipakai. Secara *default* sebuah form mempunyai properti *Name* “Form1”, “Form2”, dan seterusnya, lihat gambar 3.3. Dengan mengubah nilai properti ini dengan nama lebih deskripsi akan memudahkan kita untuk mengingat nama dari objek-objek yang dipakai dalam project yang bersangkutan.

Seorang programmer Visual Basic biasanya menggunakan prefik untuk penamaan sebuah kontrol atau form. Penggunaan prefik memudahkan seorang programer untuk mengetahui kontrol yang bersangkutan. Dengan kata lain penggunaan prefik adalah untuk membedakan sebuah kontrol dengan kontrol yang lain seperti nama *frmLatihan* untuk sebuah kontrol form dan *lblNama* untuk sebuah kontrol label. Tabel 7 berisi daftar penamaan prefik yang biasanya digunakan dalam Visual Basic.

Tabel 7. Daftar Penamaan Prefik

Prefik	Nama Elemen	Prefik	Nama Elemen
cbo	Combo Box	lin	Line
chk	Check Box	lst	List Box
cmd	Command	mnu	Menu
dat	Data	mod	Module
dir	Directory List Box	ole	OLE
drv	Drive List Box	opt	Option
fil	File List Box	pic	Picture
fra	Frame	res	Resource
fr	Form	shp	Shape
grd	Grid	tmr	Timer
hsb	Horizontal ScrollBar	txt	Text Box
img	Image	typ	User-Defined Data Type
lbl	Label	vsb	Vertical Scrool Bar

## 2. Property Left, Top, Width, Height

Semua objek-objek yang terlihat (*visible*) memiliki properti *Left*, *Top*, *Width*, dan *Height*. Properti-properti ini digunakan untuk menempatkan posisi dan ukuran suatu objek. Nilai dari properti ini

selalu relatif terhadap objek kontainernya (tempat dimana objek tersebut menempel) dan satuan ukuran default yang digunakan adalah *twips*.

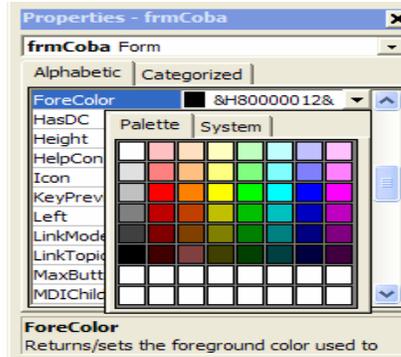
Properti-properti tersebut dapat diubah melalui jendela properti (*properties window*) seperti terlihat pada gambar 3.3 dengan memasukkan suatu nilai numerik pada saat waktu merancang atau dengan memindahkan atau mengubah ukuran form secara interaktif dengan menggunakan mouse. Selain itu perubahan properti juga dapat dilakukan pada saat program dijalankan (*run time*) melalui kode seperti contoh dibawah ini.

```
'Membuat lebar frmCoba menjadi 2 kali lebar semula  
frmCoba.Width = frmCoba.Width * 2  
  
'Membuat tinggi frmCoba menjadi 3 kali tinggi semula  
frmCoba.Height = frmCoba.Height * 3  
  
'Meletakkan frmCoba di sudut kiri atas  
frmCoba.Left = 0  
frmCoba.Top = 0
```

Tidak semua kontrol dalam Visual Basic memiliki properti Left, Top, Width, dan Height. Sebagai contoh Timer tidak memiliki properti – properti tersebut.

### 3. Property ForeColor dan BackColor

Hampir semua kontrol-kontrol Visual Basic juga mendukung kedua properti *ForeColor* dan *BackColor*, lihat gambar 3.4. Properti *ForeColor* digunakan untuk mempengaruhi warna tulisan dan properti *BackColor* digunakan untuk memilih warna dasar dari objek. Dalam kasus tertentu, properti tersebut tergantung pada properti lain. Sebagai contoh mengganti warna dasar (*background*) tidak akan berpengaruh jika nilai dari properti *backstyle* bernilai *0-Transparent*.



Gambar 18. Properti ForeColor

Ada 2 macam warna yang dapat digunakan pada properti *BackColor* dan *ForeColor* yaitu Standart Color (tab System) dan Custom Color (tab Palette). Kedua jenis warna tersebut dapat dipilih dengan menggunakan tab, namun sebaiknya digunakan warna standar (*standart color*), kecuali ada alasan yang kuat untuk menggunakan warna khusus (*custom color*). Alasan penggunaan warna standar adalah warna ini akan bekerja dengan baik pada semua komputer dengan Sistem Operasi *Windows*.

Visual Basic telah menyediakan sejumlah konstanta simbolik yang menyatakan sebuah warna. Konstanta tersebut dapat dilihat pada tabel 3.2 atau dapat juga dilihat pada jendela *Object Browser*.

Ada beberapa cara untuk mengisi nilai properti *BackColor* dan *ForeColor* melalui kode:

```
'Mengatur property ForeColor dan BackColor
'Pada lblCoba
lblCoba.ForeColor = vbHighlightText
lblCoba.BackColor = vbHighlight
```

Properti ini juga dapat diisi dengan menggunakan konstanta simbol yang lain seperti *vbBlack*, *vbBlue*, *vbCyan*, *vbGreen*, *vbMagenta*, *vbRed*, *vbWhite*, and *vbYellow* atau menggunakan nilai konstanta desimal/heksadesimal.

```
'Ketiga kode program di bawah ini mempunyai hasil sama
txtCoba.BackColor = vbCyan
txtCoba.BackColor = 16776960
txtCoba.BackColor = &HFFFFFF00
```

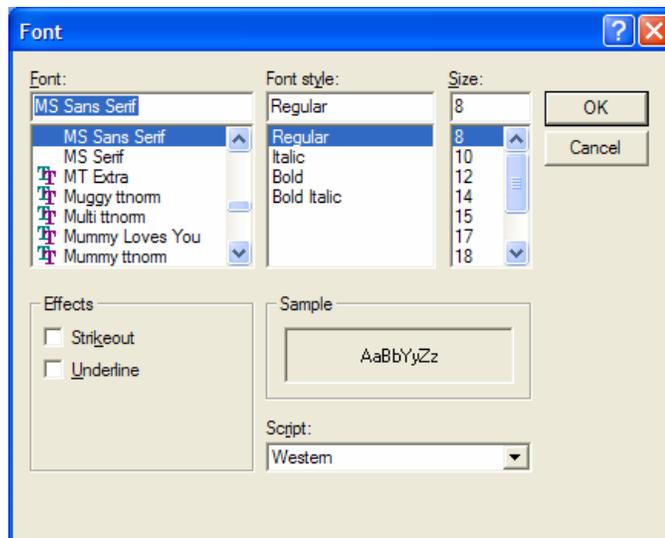
Atau dapat pula digunakan fungsi RGB yang membangun warna berdasarkan warna Merah (R), Hijau (G). dan Biru(B) (Red, Green, dan Blue) atau fungsi QBColor yang merupakan fungsi dari Quick Basic.

Tabel 8. Nilai Konstanta Warna

Konstantan	Nilai Heksadesimal	Keterangan
vb3DDKShadow	&H80000015	Darkest shadow
vb3Dface	&H8000000F	Dark shadow color for 3-D display elements
vb3Dhighlight	&H80000014	Highlight color for 3-D display elements
vb3Dlight	&H80000016	Second lightest of the 3-D colors after vb3Dhighlight
vb3Dshadow	&H80000010	Color of automatic window shadows
vbActiveBorder	&H8000000A	Active window border color
vbActiveTitleBar	&H80000002	Active window caption color
vbActiveTitleBarText	&H80000009	Text color in active caption, size box, scroll bar arrow box
vbApplicationWorkspace	&H8000000C	Background color of multiple-document interface (MDI) applications
vbButtonFace	&H8000000F	Face shading on command buttons
vbButtonShadow	&H80000010	Edge shading on command buttons
vbButtonText	&H80000012	Text color on push buttons
vbDesktop	&H80000001	Desktop color
vbGrayText	&H80000011	Grayed (disabled) text
vbHighlight	&H8000000D	Background color of items selected in a control
vbHighlightText	&H8000000E	Text color of items selected in a control
vbInactiveBorder	&H8000000B	Inactive window border color
vbInactiveCaptionText	&H80000013	Color of text in an inactive caption
vbInactiveTitleBar	&H80000003	Inactive window caption color
vbInactiveTitleBarText	&H80000013	Text color in inactive window caption, size box, scroll bar arrow box
vbInfoBackground	&H80000018	Background color of ToolTips
vbInfoText	&H80000017	Color of text in ToolTips
vbMenuBar	&H80000004	Menu background color
vbMenuText	&H80000007	Text color in menus
vbScrollBars	&H80000000	Scroll bar gray area color
vbTitleBarText	&H80000009	Text color in active caption, size box, scroll bar arrow box
vbWindowBackground	&H80000005	Window background color
vbWindowFrame	&H80000006	Window frame color
vbWindowText	&H80000008	Text color in windows

#### 4. Property Font

Pada waktu merancang, Properti Font dapat diisi dengan menggunakan *Font Dialog* seperti terlihat pada gambar 3.5.



Gambar 19. Kotak dialog Font

Cara lain untuk mengubah nilai properti bisa dilakukan melalui kode program, seperti contoh di bawah ini:

```
txtCoba.Font.Name = "Sylfaen"  
txtCoba.Font.Size = 12  
txtCoba.Font.Bold = True  
txtCoba.Font.Italic = True
```

Font merupakan objek gabungan (*Compound Object*) yang masih mempunyai properti-properti yang terpisah yaitu *Font Name*, *Font Style*, *Font Size* dan *Effects*

#### 5. Property Caption dan Text

Properti *Caption* digunakan untuk menentukan kata atau kalimat yang ditampilkan pada sebuah kontrol, seperti judul dari *Form* dan kalimat pada sebuah *Label*. Sedangkan Properti *Text* sama seperti properti *Caption* hanya saja kalimat dalam properti *Text* dapat diganti. Tidak ada kontrol yang mendukung kedua properti ini secara bersamaan. Kontrol *Label*, *CommandButton*, *CheckBox*, *OptionButton*, *Data*, dan *Frame* mendukung properti *Caption* sedangkan kontrol *TextBox*, *ListBox*, dan *ComboBox* mendukung properti *Text*. Khusus untuk properti *Caption* mendukung penggunaan karakter ampersand (&) untuk menentukan *hotkey* atau *shortcut* dari kontrol.

6. ~~Property Enabled dan Visible~~  
~~txtCoba.Text = Text1.Text~~  
~~txtCoba.Enabled = True~~  
~~txtCoba.Visible = True~~
- Kedua kode program di bawah ini mempunyai hasil sama. Secara default nilai dari properti *Enabled* dan *Visible* adalah True, tetapi mungkin pada saat program dijalankan mungkin sebuah kontrol ingin disembunyikan (*hide*) atau ditampilkan dengan keadaan tidak bisa digunakan (*disabled*). Nilai properti *Enable* dan *Visible* dapat diubah melalui kode program seperti terlihat pada kotak berikut ini.

```
'Membuat txtCoba disembunyikan  
txtCoba.Visible = False  
'Membuat txtCoba tidak bisa digunakan  
txtCoba.Enabled = False
```

### C. Metode Umum (*Common Methods*)

Pada bagian ini akan dijelaskan metode-metode yang sering digunakan antara lain Metode *Move*, *Refresh*, *SetFocus*.

#### 1. Metode *Move*

Jika suatu kontrol mempunyai properti *Left*, *Top*, *Width*, dan *Height*, maka kontrol tersebut pasti juga mendukung metode *move*. Metode ini digunakan untuk mengubah properti *Left*, *Top*, *Width*, dan *Height* dengan menggunakan sebuah operasi tunggal. Seperti terlihat pada kotak dibawah ini.

```
'Lebar Form menjadi dua kali lipat dan  
'posisinya berada pada pojok kiri atas.  
'Syntaxnya : Move(Left As Single, [Top], [Width], [Height])  
  
frmCoba.Move 0, 0, frmCoba.Width * 2, frmCoba.Width * 2
```

#### 2. Metode *Refresh*

Metode *Refresh* merupakan metode yang digunakan untuk menggambar kembali suatu kontrol. Pada keadaan normal biasanya kita tidak perlu memanggil metode ini, sebab Visual Basic secara otomatis akan menyegarkan kembali penampilan suatu kontrol ketika sudah diubah. Metode ini digunakan jika suatu kontrol ingin segera disegarkan kembali tanpa menunggu proses lain yang masih harus dilakukan. Penggunaan metode tersebut dalam kode program dapat dilihat pada kotak dibawah ini.

```
'Menyegarkan kembali DataGridView (kontrol di VB yang  
'digunakan untuk menampilkan data yang disimpan di  
'database)  
  
grdCoba.Refresh
```

#### 3. Metode *SetFocus*

Metode *SetFocus* digunakan untuk memindahkan fokus dari input ke suatu kontrol tertentu. Metode digunakan untuk memodifikasi nilai *TabOrder* yang sudah ditetapkan sebelumnya.

#### D. Kejadian (*Event*)

Selain Properti dan Metode, Visual Basic juga menyediakan sejumlah kejadian (*events*). Berikut ini merupakan kejadian-kejadian umum yang disediakan:

- Kejadian *Click* dan *DoubleClick*  
Kejadian *Click* dan *DoubleClick* terjadi pada saat seorang pengguna melakukan *click* atau *double click* pada sebuah kontrol.
- Kejadian *Change*  
Kejadian *Change* dibangkitkan pada saat isi dari suatu kontrol mengalami perubahan.
- Kejadian *GotFocus* dan *LostFocus*  
Kejadian *GotFocus* akan dibangkitkan pada saat sebuah kontrol menerima fokus input, sedangkan *LostFocus* merupakan kebalikan dari kejadian *GotFocus*, dibangkitkan pada saat kontrol memindahkan fokusnya menuju ke kontrol yang lain.
- Kejadian *KeyDown*, *KeyUp*, dan *KeyPress*  
Kejadian *KeyDown*, *KeyUp*, dan *KeyPress* merupakan kejadian yang berhubungan dengan *keyboard*. Masing-masing kejadian dibangkitkan pada saat *keyboard* di tekan ke bawah, dilepas, dan ditekan.
- Kejadian *MouseDown*, *MouseUp*, dan *MouseMove*  
Kejadian-kejadian ini dibangkitkan pada saat mouse di-*click*, dilepas, dan digerakkan di atas sebuah kontrol

#### E. Contoh Program

Kita akan mencoba membuat program yang meminta user untuk menekan tombol Yes sampai dapat. Letakkan beberapa object yang ada dalam form, atur beberapa setting property yang ada seperti terlihat pada gambar di bawah ini.



Control	Property	Setting Value
Form1	Name	FrYes
	Caption	Klik Yes
Label1	Caption	Are you handsome??
Image1	Picture	Gambar bebas
Command1	Name	CmdYes
	Style	1-Graphical
	Picture	Gambar bebas

Ketikkan kode program berikut ini :

```

`Kode program berikut dijalankan pada saat user
`menggerakkan pointer di atas cmdYes
`cmdYes akan bergerak secara random

Private Sub cmdYes_MouseMove(Button As Integer, Shift As
Integer, X As Single, Y As Single)
cmdYes.Left = Int(Rnd * (Me.Width - cmdYes.Width))
cmdYes.Top = Int(Rnd * (Me.Height - cmdYes.Height))
End Sub

```

## Soal Latihan

Buat program yang bisa merubah warna tombol (*command*). Misalnya, ketika pointer berada di atas tombol maka tombol berwarna kuning, dan ketika pointer tidak berada di atas tombol (meninggalkan tombol) maka warna tombol menjadi biru.

## BAB IV

### KONTROL INTRINSIK

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal kontrol intrinsik dalam Visual Basic	<ul style="list-style-type: none"><li>• Dapat menggunakan kontrol intrinsik yang terdapat di Toolbox</li><li>• Mengenal nilai properti kontrol intrinsik</li><li>• Mengenal fungsi dan kejadian kontrol intrinsik</li></ul>

#### A. Pengantar

Dalam Visual Basic yang dimaksud dengan kontrol-kontrol intrinsik adalah kontrol-kontrol yang tampil pada Toolbox saat Visual Basic pertama kali dijalankan. Disamping kontrol-kontrol intrinsik tersebut, Visual Basic juga mempunyai kontrol-kontrol tambahan yang disebut *Microsoft ActiveX Control (OCX controls)*. Meskipun OCX mempunyai banyak keunggulan dibanding kontrol intrinsik, namun penggunaan kontrol intrinsik tetap memiliki keunggulan dibanding OCX antara lain:

- Kontrol intrinsik disertakan pada file MSVBVM60.DLL yang terdapat pada setiap Visual Basic sehingga tidak memerlukan tambahan file yang lain.
- Kontrol intrinsik ditampilkan lebih cepat dibanding OCX pada saat program dieksekusi (*run time*).

#### B. Command

*Command button* dapat digunakan dengan mudah yaitu hanya dengan diletakkan pada form dan memberikan *caption* (properti *caption*) dan nama (properti *name*) yang sesuai. Kode untuk Prosedur kejadian klik (*Click event procedure*) perlu ditulis agar sebuah command button dapat berfungsi (menerima kejadian ketika ditekan).

```
Private Sub cmdTest_Click()  
    ' Menyimpan data, menutup form yang aktif.  
    Call SaveDataToDisk 'Memanggil prosedur SaveDataToDisk  
    Unload Me 'menutup form  
End Sub
```

Selain kejadian klik (*click event*), *command* juga menerima kejadian dari *keyboard* dan *mouse* (seperti *KeyDown*, *KeyPress*, *KeyUp*, *MouseDown*, *MouseMove*, *MouseUp*, tetapi tidak menerima kejadian *DblClick*).

#### C. Label

Kontrol label digunakan untuk memberikan keterangan untuk kontrol- kontrol lain seperti *TextBox*, *ListBox*, dan *ComboBox*. Biasanya properti *caption* pada kontrol tersebut diisi dengan kalimat yang sesuai dan dilengkapi karakter *ampersand*(&) untuk memberikan *hot key*. Selain properti *caption*, properti lain yang cukup berguna adalah *BorderStyle* (digunakan untuk menampilkan kontrol label secara 3 Dimensi) dan *Alignment* (Jika align dari caption tampil pada pinggir kiri, kanan, atau tengah).

Jika *caption* sebuah label terlalu panjang, maka properti *WordWrap* dapat diberi nilai *true* sehingga label tersebut memiliki lebih dari satu baris. Atau juga dapat mengubah nilai properti *AutoSize* menjadi *true* agar kontrol secara otomatis mengubah ukurannya sesuai dengan panjang *caption*.

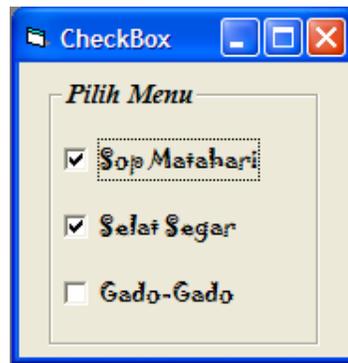
#### D. TextBox

Kontrol *TextBox* digunakan untuk menerima masukan dari pengguna. Pengguna dapat memasukan data berupa angka, huruf, dan karakter- karakter khusus. Pada umumnya kontrol tersebut diletakkan di sebelah kontrol label yang berguna untuk memberikan keterangan. Setelah kontrol *TextBox* tersebut diletakkan pada form, umumnya properti *text* pada kontrol tersebut dihapus. Selain properti *text*, juga terdapat properti *multiline* yang digunakan agar *Textbox* dapat menerima masukan karakter yang panjang. Seperti *label*, *TextBox* juga mendukung properti *alignment*.

Jika sebuah kontrol *TextBox* menerima masukan yang terbatas panjangnya, maka properti *maxlength* dapat diberi nilai yang menunjukkan panjang maksimum yang diperbolehkan. Jika kontrol *TextBox* menerima masukan berupa *password*, maka properti *passwordChar* dapat diisi dengan dengan karakter tertentu, biasanya karakter asterik(\*).

#### E. CheckBox

Kontrol *Checkbox* mempunyai banyak kegunaan ketika suatu dialog menawarkan pilihan. Memungkinkan user memilih lebih dari satu pilihan. Ketika kontrol tersebut diklik maka kontrol tersebut menunjukkan sebuah keadaan benar atau salah (True atau False).



Gambar 20. CheckBox

Peletakan kontrol tersebut pada sebuah form biasanya dilakukan dengan mengisi properti *Caption* untuk menjelaskan keadaan dari kontrol. Sedangkan kejadian (Event) yang penting dari kontrol tersebut adalah kejadian klik (*click event*).

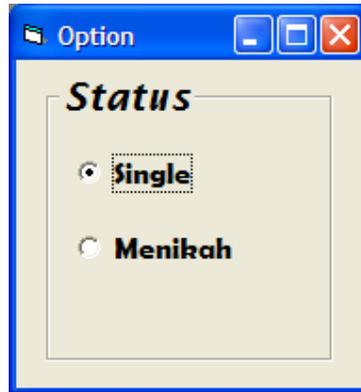
Sebagai contoh ketika sebuah checkbox diklik, maka akan memberikan dampak pada kontrol yang lain. Berikut contoh penggunaan pada suatu kode program :

```
Private Sub chkSop_Click()  
    fraTest.Enabled = (chkSop.Value = 1)  
End Sub
```

CheckBox memiliki nilai 1 dan 0 (1 = True, 0 = False). Ketika CheckBox dipilih (dicentang) maka akan bernilai 1 dan sebaliknya.

#### F. Option

Kontrol Option selalu digunakan dalam suatu group berjumlah 2 atau lebih yang menawarkan pilihan yang bersifat *mutually exclusive* (hanya diperbolehkan untuk memilih salah satu pilihan dari beberapa opsi yang ditawarkan), lihat gambar 4.2.



Gambar 21. Option

Apabila sebuah *Option* dipilih, maka *Option* yang lain dalam group yang sama tidak akan terpilih. *Option* yang dipilih bernilai 1, dan yang tidak dipilih bernilai 0. *Option* dalam satu group tersebut biasanya diletakkan pada sebuah *frame* yang sama.

Contoh dalam pemrogramannya seperti berikut ini:

```
\ Jika status single dipilih, mendapat tunjangan 300000
\ Jika status menikah dipilih, mendapat tunjangan 500000

If optSingle.Value = 1 Then
    tunjangan = 300000
ElseIf optNikah.Value = 1 Then
    tunjangan = 500000
End If
```

#### G. Frame

Kontrol *Frame* mempunyai fungsi yang mirip dengan kontrol *label* digunakan untuk memberikan keterangan pada kontrol lainnya. Bedanya, kontrol *frame* juga digunakan sebagai kontainer (tempat) dari kontrol-kontrol yang lain. Contoh kontrol frame, lihat gambar 4.2.

Pada umumnya *frame* diletakkan pada *form* dan properti *caption*-nya diberi nilai untuk memberikan keterangan tentang kontrol-kontrol yang menempel. Setelah kontrol *frame* diletakkan, kontrol anak (*child control*)

dari *frame* tersebut dapat diletakkan secara langsung diatas *frame* tersebut. Alternatif yang lain adalah dengan membuat kontrol-kontrol yang sudah ada menjadi kontrol anak dari sebuah *frame* dengan cara memilih kontrol-kontrol tersebut dengan melakukan klik disertai dengan menekan tombol *ctrl* kemudian mengambilnya (*cut : ctrl+x*) dan meletakkan (*paste : ctrl+v*) diatas kontrol *frame*.

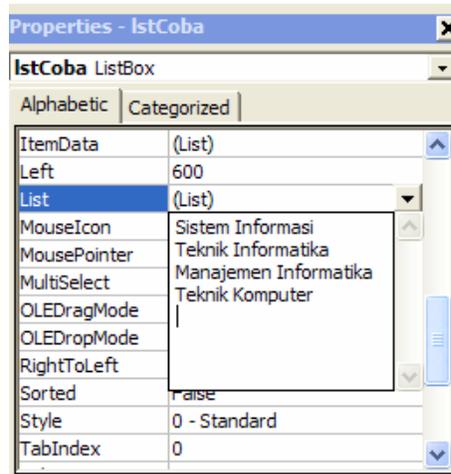
Kontrol *frame* mempunyai 2 karakteristik yang menarik yaitu :

- Jika *Frame* kita gerakkan, maka semua kontrol didalamnya akan mengikuti.
- Properti *enable* dan *visible* dari kontrol *Frame* akan mempengaruhi kontrol didalamnya. Dengan kata lain apabila suatu *Frame* diset *enable : false* (biasa disebut : *disable*), maka semua kontrol yang ada didalamnya secara otomatis akan *disable* pula.

#### H. List Box

Ketika sebuah Kontrol *ListBox* diletakkan pada sebuah *form*, maka beberapa properti dari kontrol tersebut harus diberikan seperti atribut *sorted* agar secara otomatis item yang ada didalamnya diurutkan berdasarkan alpabetik.

Jika item-item yang harus muncul pada kontrol *listbox* sudah diketahui pada saat *design time*, item-item tersebut dapat dimasukkan langsung melalui properti *list*. Cukup mengetikkan isinya, untuk berpindah ke barisberikutnya tekan *Ctrl+Enter*.



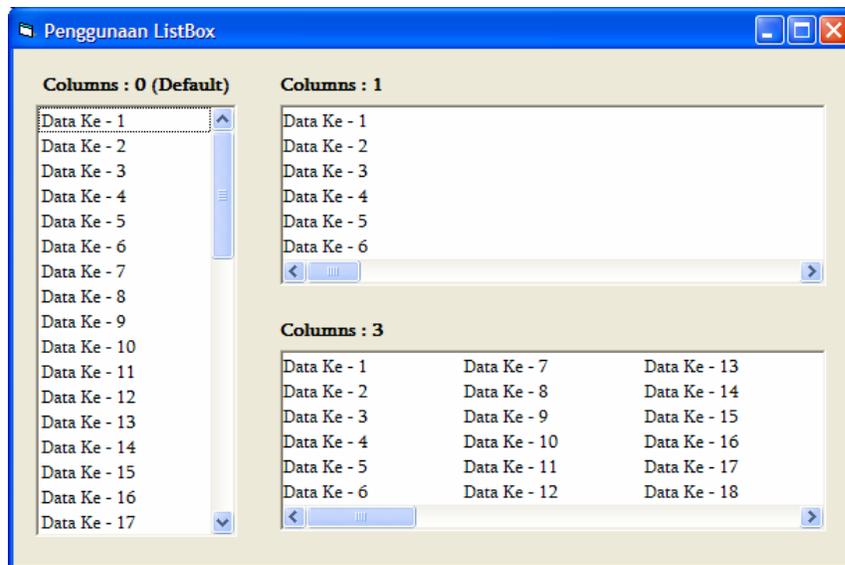
Gambar 22. Properti List dari Kontrol ListBox

Namun hal ini tidak menutup kemungkinan untuk memasukkan item data melalui kode program seperti contoh kode dibawah ini :

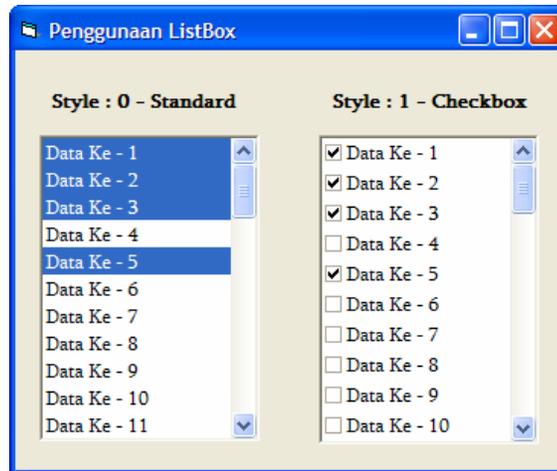
```
lstCoba.AddItem "Sistem Informasi"  
lstCoba.AddItem "Teknik Informatika"  
lstCoba.AddItem "Manajemen Informatika"  
lstCoba.AddItem "Teknik Komputer"
```

Properti *columns* digunakan untuk menentukan tipe-tipe yang berbeda dari *listbox* dengan beberapa kolom dan sebuah *horizontal scroll bar* pada border sebelah bawah atau sebuah kolom dan sebuah *vertical scroll bar* pada sisi sebelah kanan.

Pada gambar 4.4 ditunjukkan perbedaan penggunaan nilai properti *columns* 0, 1 dan 3



Gambar 23. Kontrol *ListBox* dengan Nilai Properti *Columns* 0, 1 dan 3 Kontrol *ListBox* juga memperbolehkan seorang pengguna untuk memilih lebih dari satu item pada suatu waktu. Untuk mengaktifkan, properti *MultiSelect* diberi nilai *1-Simple* atau *2-Extended*. Pada mode *1-Simple* pemilihan dapat dilakukan dengan menggunakan menekan spasi (*space bar*) atau *mouse*. Pada mode *2-Extended* pemilihan dilakukan dengan menekan tombol *shift*. Selain properti *MultiSelect*, kontrol *ListBox* mempunyai properti *Style* yang diberi nilai *0-Standard* dan *1-Checkbox* seperti ditunjukkan Gambar 4.5



Gambar 24. Kontrol ListBox dengan properti Style0–Standard dan 1-Checkbox

Contoh : menggunakan ListBox untuk memilih kota yang pernah Andakunjungi :



Gambar 25. Contoh Program Menggunakan ListBox

- ✓ Atur form seperti pada gambar
- ✓ Ketikkan kode program seperti berikut :

```

Private Sub cmdAdd_Click()
'Menampilkan nama kota di lstKunjung sesuai yang dipilih di
lstKota
Dim CurItem As Integer
CurItem = 0
Do
    'Jika item yang dipilih
    If lstKota.Selected(CurItem) Then
        'Tambahkan ke lstKunjung.Jika Anda menembakkannya ke
ComboBox,
        'ganti "lstKunjung" di bawah dengan nama ComboBox yang ada.
        'Contoh: cboKunjung.AddItem lstKota.List(CurItem)
        lstKunjung.AddItem lstKota.List(CurItem)
        'Lalu hapus dari lstKota
        lstKota.RemoveItem (CurItem)
    Else
        CurItem = CurItem + 1
    End If
Loop Until CurItem = lstKota.ListCount
End Sub

Private Sub cmdAddAll_Click()
'Memindahkan semua kota ke lstKunjung
For i = 0 To lstKota.ListCount - 1
    lstKunjung.AddItem lstKota.List(i)
Next i
lstKota.Clear
End Sub

```

## I. Combo Box

Kontrol *ComboBox* merupakan kontrol yang mirip dengan kontrol *Listbox*, jadi apa yang dapat bekerja pada kontrol *Listbox* juga bekerja dengan baik pada kontrol *comboBox*. Seperti pada kontrol *listbox* juga memiliki properti *sorted* yang digunakan untuk mengurutkan item data secara otomatis dan properti *list* yang digunakan untuk menambahkan data pada saat design time. Kebanyakan metode yang digunakan pada *ListBox* juga terdapat pada *ComboBox* seperti *AddItem*, *RemoveItem*, dan *Clear*.

Kontrol *ComboBox* sebenarnya merupakan gabungan antara sebuah *TextBox* dan *ListBox*.

Kontrol *ComboBox* juga memiliki properti *Style* memberikan 3 pilihan yaitu :

- 0 – Dropdown Combo : pemakai diperbolehkan mengetik tulisan yang tidak ada di *list* (daftar)
- 1 – Simple Combo : pemakai boleh mengetik atau memilih dengan tombol ke atas dan ke bawah pada *keyboard* dan daftar tidak bias terbuka.

- 2 – Dropdown List : pemakai hanya dapat memilih tulisan yang ada di *list* (daftar) dan tidak bisa mengetik di *list*

## J. Image

Digunakan untuk menampilkan gambar dalam format bitmaps (BMP), device independent bitmaps (DIB), metafiles (WMF), enhanced metafiles (EMF), GIF dan JPEG compressed files, dan icons (ICO dan CUR).

Properti *Stretch* digunakan untuk menentukan apakah gambar disesuaikan dengan ukuran kontrol (gambar dapat mengalami pengecilan maupun pembesaran)

### 1. Menampilkan gambar kedalam kontrol image

Pada saat *design* anda dapat mengisikan gambar ke dalam *image box* dengan menggunakan properti *Picture*, sedangkan pada runtime anda dapat menggunakan fungsi *LoadPicture(namafile)* untuk memuat gambar ke properti *Picture* dari kontrol image, contoh:

```
imgGambar.Picture = LoadPicture("D:\Picture\oryn.jpg")
```

Catatan : Fungsi *LoadPicture(namafile)* digunakan untuk memuat file grafik dengan format grafik bitmap (.bmp), icon (.ico), run-length encoded (.rle), metafile (.wmf), enhanced metafiles (.emf), GIF, JPEG (.jpg).

### 2. Mengosongkan kontrol image

Untuk mengosongkan kontrol image pada saat runtime, anda dapat menggunakan fungsi *LoadPicture*, tanpa menggunakan argumen nama file, contoh :

```
imgGambar.Picture = LoadPicture
```

### 3. Menyimpan gambar dalam kontrol *Image* ke File

Anda dapat menggunakan perintah *SavePicture* gambar, namafile untuk menyimpan gambar kedalam file dengan format BMP, contoh :

```
SavePicture imgGambar.Picture, "D:\Picture\oryn.bmp"
```

Catatan : Perintah *SavePicture* akan selalu menyimpan gambar ke format bitmap (.bmp), tanpa memperhatikan format sumber gambar.

#### K. Picture Box

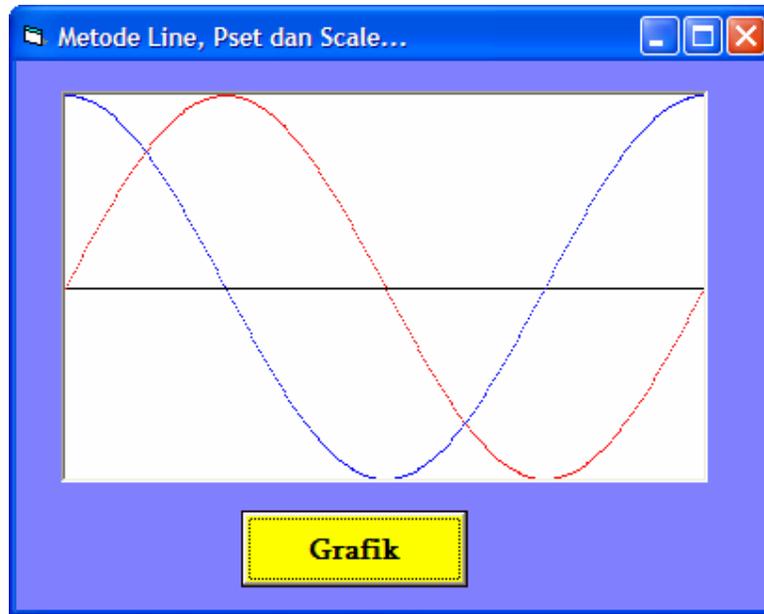
Kalau *Image* digunakan untuk menampilkan gambar, demikian juga *PictureBox* Selain menampilkan gambar, *Picture Box* mendukung berbagai metoda untuk Untuk operasi grafik, dan dapat berfungsi sebagai kontainer bagi kontrol-kontrol lain, seperti fungsi *Frame*.

Contoh penggunaan metode *Line*, *Pset*, *Scale* pada *PictureBox* untuk menampilkan grafik :

Ketikkan kode program seperti berikut :

```
Private Sub cmdGrafik_Click()  
picGrafik.ScaleMode = 0  
picGrafik.ScaleWidth = 360  
picGrafik.ScaleHeight = 2  
picGrafik.ScaleLeft = 0  
picGrafik.ScaleTop = -1  
picGrafik.ForeColor = vbBlack `Set ForeColor Jadi Hitam  
picGrafik.Line (0, 0)-(360, 0) `Garis Hitam  
  
For i = 0 To 360  
    picGrafik.ForeColor = vbRed    `Titik Merah  
    picGrafik.PSet (i, -Sin(i * 3.14 / 180))  
  
    picGrafik.ForeColor = vbBlue  `Titik Biru  
    picGrafik.PSet (i, -Cos(i * 3.14 / 180))  
Next i  
End Sub
```

Setelah program dijalankan, maka hasilnya seperti terlihat pada gambar4.7



Gambar 26. Membuat Grafik dengan PictureBox

❖ Scale Mode

Digunakan untuk menentukan unit skala yang digunakan (0 - User, 1 - Twip, 2 - Point, 3 - Pixel, 4 - Character, 5 - Inch, 6 - Milimeter, 7 - Centimeter)

- 1 inchi = 1440 Twip
- 1 cm = 567 Twip
- 1 inchi = 72 point
- 1 character = (120 Twip untuk lebar, 240 untuk tinggi)
- 1 cm = 1000 unit (Himetric)

Anda dapat membuat modus skala sendiri dengan menset properti ini menjadi 0 - User, dan skala anda dapat ditentukan pada ScaleWidth dan ScaleHeight, perhatikan kembali contoh Grafik sinus sebelumnya.

❖ Scale Left

Digunakan untuk menentukan nilai koordinat horizontal paling kiri, Anda dapat menggunakan properti ini untuk menentukan koordinat paling kiri dari suatu sumbu X.

❖ Scale Top

Digunakan untuk menentukan nilai koordinat vertikal paling atas, anda dapat menggunakan properti ini untuk menentukan koordinat paling atas dari suatu sumbu Y.

❖ Scale Height

Digunakan untuk menentukan tinggi sumbu vertikal. Menentukan panjang sumbu Y.

❖ ScaleWidth

Digunakan untuk menentukan panjang sumbu horizontal. Menentukan panjang sumbu X.

Metode Grafik Pada PictureBox

Salah satu perbedaan antara Image dengan PictureBox adalah tersedianya berbagai metoda penggambaran grafik pada *PictureBox*, antara lain:

➤ Circle(x,y),r,warna,awal,akhir,Aspek

Menggambarkan sebuah lingkaran dengan berpusat pada koordinat x,y dan jari-jari r dengan warna garis, mulai dari sudut awal, sampai sudut akhir (yang dinyatakan dalam radian), serta aspek perbandingan tinggi dengan lebar

➤ Cls

Membersihkan *PictureBox* dengan warna *BackColor*

➤ Line (x1,y1) - (x2, y2),warna

Menggambarkan garis tunggal dari koordinat x1,y1 sampai dengan x2,y2

➤ Line (x1,y1) - (x2, y2),warna,B

Menggambarkan kotak dari koordinat x1,y1 sampai dengan x2,y2

➤ Line (x1,y1) - (x2, y2),warna,BF

Menggambarkan kotak berisi dari koordinat x1,y1 sampai dengan x2,y2

➤ Pset(x,y)

Mencetak dot (titik) pada koordinat tertentu pada *Form*, *Picturebox*, dan *Printer*, pada koordinat yang ditentukan

#### L. Scroll Bar

Ada 2 macam kontrol *ScrollBar* yaitu *HscrollBar* dan *VscrollBar*. Keduanya mempunyai sifat yang sama namun berbeda orientasi. Properti dari kontrol tersebut yang cukup penting adalah properti *Min* dan *Max* yang merepresentasikan nilai jangkauan minimum dan maksimum. Properti lain yang penting pada waktu run-time adalah *Value*, yang selalu mengembalikan nilai posisi indikator yang terdapat pada *scrollbar*.

Ada 2 kejadian (Event) penting dari *scrollbar* yaitu kejadian *Change* yang akan dipanggil pada saat *scrollbar* diklik dan kejadian geser (*Scroll*) yang dipanggil ketika indikator pada *scrollbar* mengalami perubahan.

Contoh : Menggeser posisi *Image* ke atas dan ke bawah



Gambar 27. M

Kode programnya seperti berikut ini:

```
Private Sub vsbGambar_Change()  
imgGambar.Top = -vsbGambar.Value  
End Sub  
  
Private Sub vsbGambar_Scroll()  
imgGambar.Top = -vsbGambar.Value  
End Sub
```

Event *Scroll*, akan dibangkitkan ketika pemakai melakukan pergeseran terhadap Bar dengan menggunakan drag pada tombol kiri mouse, jadi Event *Scroll* akan terjadi ketika pemakai melakukan pergeseran dengan menekan tombol kiri mouse, dan diakhiri dengan Event *Change* ketika

pemakai melepas penekanan mouse. Jadi anda harus memanfaatkan kedua event tersebut untuk mendapatkan hasil yang baik dalam pemakaian ScrollBar.

#### M. Timer

Komponen Timer sangat baik untuk mengimplementasikan pengaruh waktu terhadap suatu proses seperti proses animasi atau dalam pembuatan game supaya kecepatan dari game bisa diatur. Komponen Timer bersifat non-visual, pada saat program dijalankan, Timer tidak kelihatan. Beberapa properti *Timer* adalah sebagai berikut :

☒ Enabled

Menentukan apakah kontrol dapat efektif terhadap Event Timer.

☒ Index

Digunakan untuk menentukan nomor index, jika kontrol tersebut merupakan kontrol array.

☒ Interval

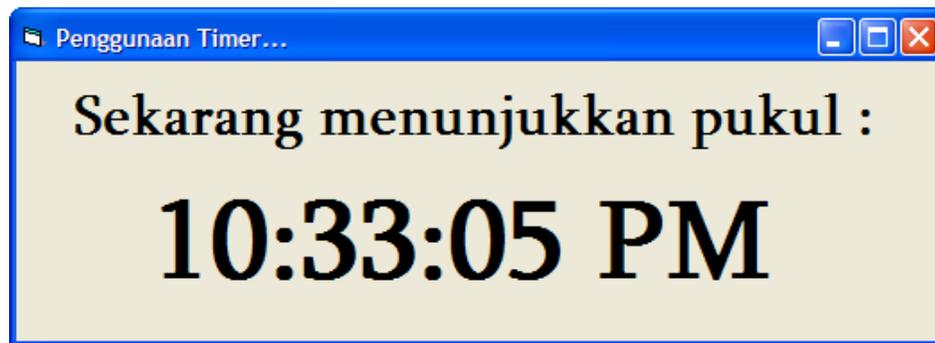
Menentukan nilai interval dalam mili detik (1/1000) antar pemanggilan Event Timer.

☒ Tag

Properti ini dapat digunakan sebagai tempat menyimpan data sementara yang berkaitan dengan kontrol label tersebut

*Event Timer* : Event yang dibangkitkan oleh kontrol timer berdasarkan interval waktu yang telah ditentukan.

Contoh : Membuat Jam Digital



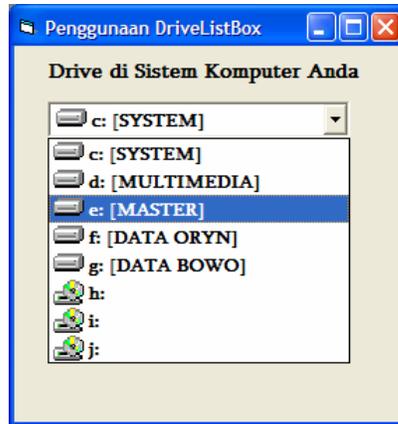
Gambar 28. Menggunakan Timer untuk membuat Jam Digital

Letakkan Label dan Timer pada Form seperti pada gambar 4.9. Atur properti Timer, (*Enabled : True, Interval : 1000*). Dan ketikkan kode program seperti berikut :

```
Private Sub tmrJam_Timer()  
lblJam.Caption = Format(Now, "hh:mm:ss")  
End Sub
```

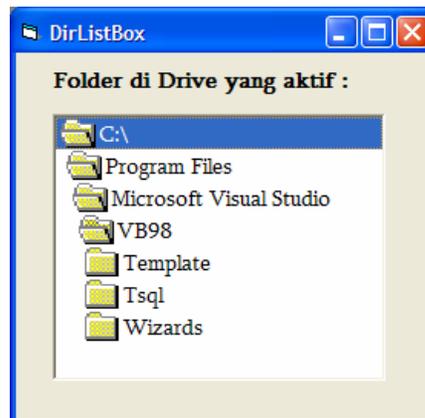
N. DriveListBox, DirListBox, FileListBox

◇ *DriveListBox* digunakan untuk menampilkan drive yang terdapat pada sistem komputer



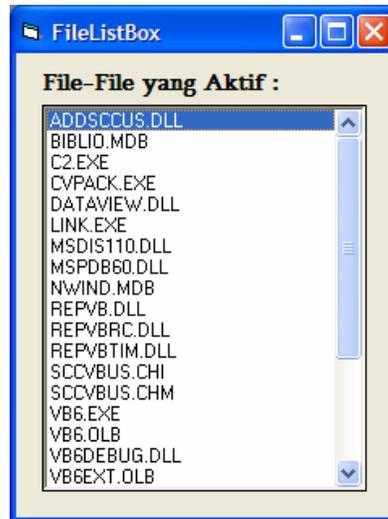
Gambar 29. Menggunakan *DriveListBox*

◇ *DirListBox* dapat digunakan untuk menampilkan folder-folder yang terdapat di drive yang aktif.



Gambar 30. Menggunakan *DirListBox*

◇ *FileListBox* digunakan untuk menampilkan file-file yang terdapat pada folder yang aktif.



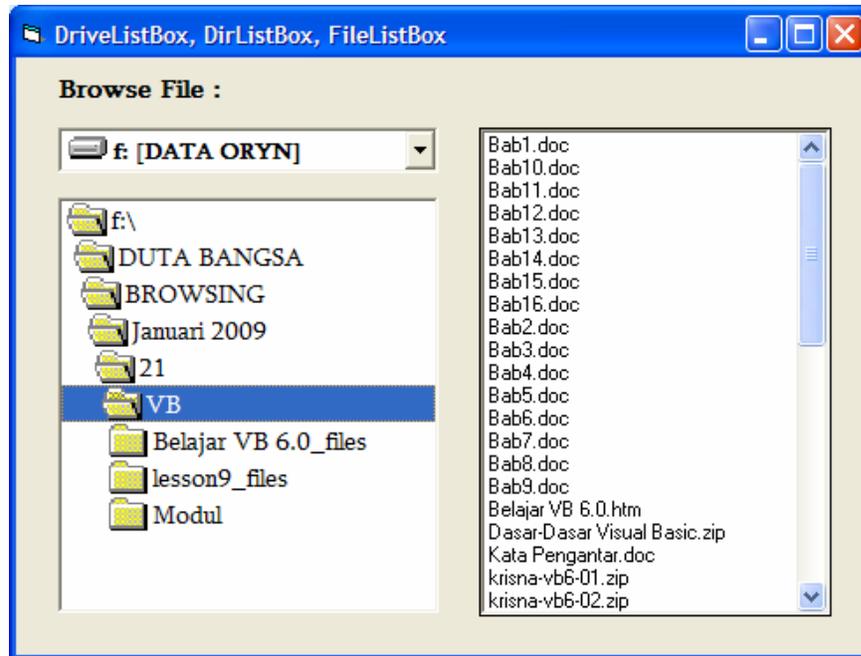
Gambar 31. Menggunakan FileListBox

1. Membatasi file pada FileListBox

Untuk membatasi file berdasarkan nama file, anda dapat menggunakan properti *Pattern*, misalnya kalau file yang ingin ditampilkan hanya berupa file bitmap, maka kita dapat menggunakan (\*.bmp), kalau file yang ingin ditampilkan berupa file grafik kita dapat menggunakan (\*.bmp;\*.jpg;\*.gif;\*.wmf;\*.ico) Untuk membatasi file berdasarkan atributnya, anda dapat menggunakan properti Archive, Hidden, Normal, ReadOnly dan System, dengan menentukan masing-masing menjadi True atau False.

2. Menghubungkan DriveListBox, DirListBox dan FileListBox

Misalnya kita memiliki tiga buah kontrol yaitu DriveListBox, DirListBox, dan FileListBox dimana perubahan pada DriveListBox akan menyebabkan perubahan tampilan pada DirListBox, dan FileListBox.



Gambar 32. Menghubungkan DriveListBox, DirListBox, FileListBox

Ketikkan kode program seperti berikut :

```

Private Sub dirTest_Change ()
filTest.Path = dirTest.Path
End Sub

Private Sub drvTest_Change ()
dirTest.Path = drvTest.Drive
End Sub

```

## 0. Common Dialog Box

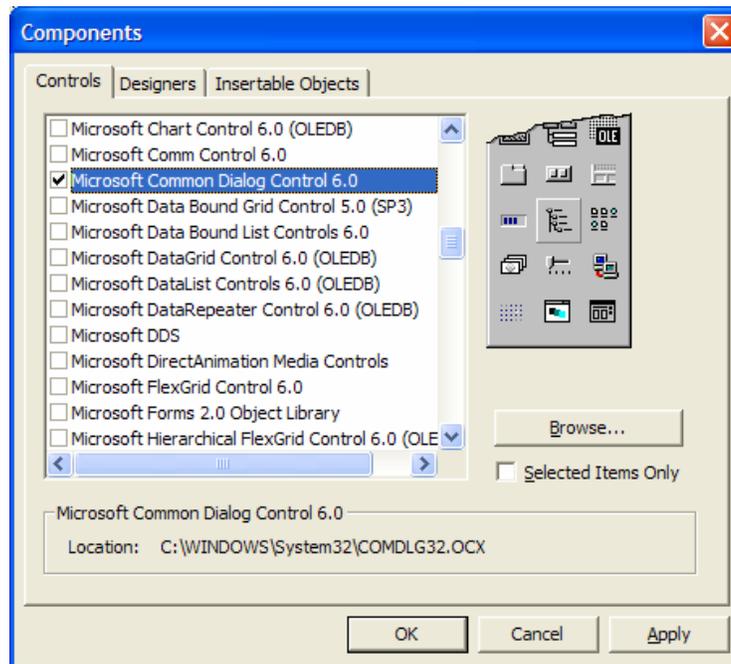
### 1. Metode Common Dialog

Common Dialog Box merupakan kontrol yang menyediakan fasilitas dialog-dialog umum yang sering digunakan pada lingkungan Windows seperti *File Dialog*, *Font Dialog*, dan *Printer Dialog*.

Pada saat kontrol tersebut diletakkan pada form, kontrol tersebut tidak dapat diubah ukurannya seperti kontrol timer, namun kontrol tersebut dapat dilihat pada saat program dijalankan (*runtime*) dalam bentuk dialog.

Meskipun kontrol tersebut merupakan kontrol yang umum, namun kontrol ini tidak diletakkan pada Toolbox. Untuk menambahkan kontrol tersebut pada Toolbox ikutilah langkah-langkah berikut ini:

- a. Pilih menu *Project-Components* atau dengan menekan *Ctrl-T*, maka akan ditampilkan *Components Dialog box*, lihat gambar 4.14.



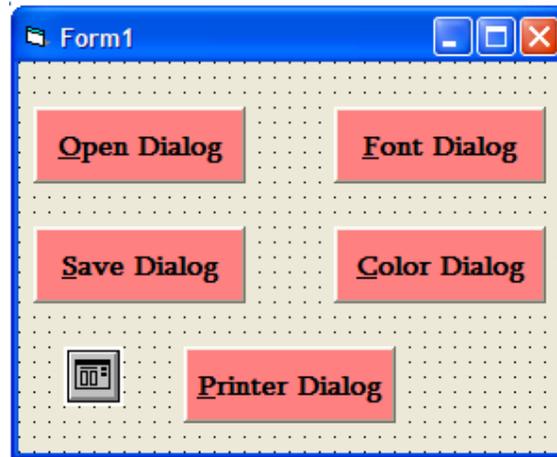
Gambar 33. Window Components-Common Dialog

- b. Pilih Microsoft Common Dialog Box Control pada ListBox dan klikOK, maka Kontrol Dialog Box akan ditampilkan pada ToolBox.
- c. Untuk menambahkan pada form, double click pada kontroltersebut  
Untuk menampilkan Dialog box dapat digunakan salah satu Methodberikut ini:
- *ShowColor* untuk menampilkan sebuah color dialog.
  - *ShowFont* untuk menampilkan sebuah font dialog.
  - *ShowHelp* untuk menampilkan sebuah help dialog.
  - *ShowOpen* untuk menampilkan sebuah open file dialog.
  - *ShowPrinter* untuk menampilkan sebuah printer dialog.
  - *ShowSave* untuk menampilkan sebuah save file dialog

```
cdbDialog.DialogTitle = "File Open"  
cdbDialog.Filter = "*.txt" 'Tampilkan file text saja  
cdbDialog.FileName = "*.txt" 'Default nama file  
cdbDialog.ShowOpen ' menampilkan dialog open file
```

## 2. Contoh Program

Tambahkan 1 Common Dialog Control dan 5 Command di FormAnda. Atur seperti berikut :



Gambar 34. Contoh Program menggunakan Common Dialog

Ketikkan kode program berikut :

```
Private Sub cmdColor_Click()
    cdbDialog.DialogTitle = "Select a Color"
    cdbDialog.ShowColor      ' Display the dialog box
End Sub

Private Sub cmdFont_Click()
    cdbDialog.DialogTitle = "Font"
    cdbDialog.FontName = "Arial"
    cdbDialog.FontBold = cdLCFBoth
    cdbDialog.ShowFont
End Sub

Private Sub cmdPrinter_Click()
    cdbDialog.DialogTitle = "Select a Printer"
    cdbDialog.ShowPrinter
End

Private Sub cmdOpen_Click()
    cdbDialog.DialogTitle="File Open"
    cdbDialog.Filter="*.txt" ' Tampilkan file text saja
    cdbDialog.FileName="*.txt" ' Default nama file
    cdbDialog.ShowOpen ' menampilkan dialog open file
End Sub

Private Sub cmdSave_Click()
    cdbDialog.DialogTitle="File Save"
    cdbDialog.Filter="*.*" ' Show all files
    cdbDialog.FileName="test.txt" ' Default filename
    cdbDialog.ShowSave ' Trigger the dialog box
End Sub
```

#### P. Kontrol Animasi

Kontrol Animasi dapat digunakan untuk menjalankan file AVI dan untuk menambahkan animasi pada sebuah program aplikasi. Kontrol ini hanya mendukung file AVI saja yang tidak memiliki suara dan tidak terkompres. File AVI ini dapat digemukan pada subdirektori \Common\Graphics

\Video pada Microsoft Visual Studio 6.0

Kontrol animasi menyediakan 3 properti utama. Dua diantaranya adalah *Center* dan *BackStyle* yang hanya dapat diset pada saat design time dan bersifat *read only* pada saat run time. Jika properti *Center* diset true, maka file AVI akan ditampilkan tepat ditengah.

Sedangkan properti *BackStyle* dapat diberi nilai 0-cc2 (*BackStyle Transparent*) atau 1-cc2 (*BackStyle Opaque*). Properti yang ketiga adalah *AutoPlay* yang dapat diberi nilai kapanpun juga. Jika properti ini diberi nilai *True* maka secara otomatis akan dimainkan segera pada saat kontrol tersebut ditampilkan.

Untuk membuka file AVI yang akan digunakan untuk ditampilkan dapat digunakan methods *Open*.

```
Animation1.Open "C:\vb6\Graphics\AVIs\filecopy.avi"
```

Untuk menjalankan file avi dapat digunakan methods *Play* dengan format sebagai berikut:

```
Play [RepeatCount], [StartFrame], [EndFrame]
```

⊗ *RepeatCount* merupakan banyaknya perulangan animasi yang dijalankan (nilai defaultnya adalah 1 yang akan menjalankan file AVI terus menerus).

⊗ *StartFrame* menunjukkan awal frame yang dijalankan.

⊗ *EndFrame* menunjukkan akhir frame.

Untuk menghentikan animasi dapat digunakan salah satu dari 2 cara tergantung pada cara memulai yaitu :

⇒ Jika animasi dalam mode *AutoPlay*, maka dapat dihentikan dengan memberi nilai properti *AutoPlay* menjadi false.

⇒ Jika animasi dijalankan dengan menggunakan method *play* maka dapat dihentikan dengan menggunakan method *stop*.

Untuk menghemat memori, kontrol animasi dapat di-*unload* dengan menggunakan metoda *close*



Gambar 35. Contoh Program menggunakan Animation 1

Q. Contoh Program

Berikut contoh program untuk menghitung durasi waktu pada saat user klik tombol START sampai user klik tombol STOP. Ketika program dijalankan, waktu akan berjalan. Ketika user klik tombol START maka durasi waktu akan berjalan, ketika user klik tombol STOP maka waktu berhenti dan user mendapat informasi durasi waktu. Dan ketika klik tombol EXIT maka program selesai.

Buatlah tampilan form seperti gambar 4.17. Tambahkan sebuah Timer. Ketik kode program seperti yang telah disediakan di kotak *source code*.



Gambar 36. Contoh program menghitung durasi waktu

Ketikkan kode program seperti berikut :

```
Dim Awal As Date
Dim Akhir As Date
Dim Lama As Single

Private Sub cmdProses_Click()
'Jika tombol mula-mula adalah Start
If cmdProses.Caption = "START" Then
    Awal = Now
    'Tampung waktu pertama kali dimulai
    lblStart.Caption = Format(Awal, "hh:mm:ss")
    cmdProses.Caption = "STOP"
    'Jika tombol dengan tulisan Stop ditekan
ElseIf cmdProses.Caption = "STOP" Then
    'Matikan Timer
    tmrWaktu.Enabled = False
    Akhir = Now
    Lama = Akhir - Awal
    lblDurasi.Caption = Format(Lama, "hh:mm:ss")
    cmdProses.Caption = "EXIT"
ElseIf cmdProses.Caption = "EXIT" Then
    Unload Me
End If
End Sub

Private Sub tmrWaktu_Timer()
lblJam.Caption = "Sekarang Jam : " & Format(Now, "hh:mm:ss")
If cmdProses.Caption = "STOP" Then
    'Hitung kembali durasi waktu
    lblDurasi.Caption = Format(Now - Awal, "hh:mm:ss")
End If
End Sub
```

## Soal Latihan

Buka kembali contoh program seperti pada gambar 36 di atas. Tambahkan satu textbox/label untuk menginformasikan berapa jumlah yang harus dibayarkan setiap durasi waktu tertentu. Misalnya setiap durasi 30 detik harus membayar Rp. 750,00. Jadi biaya akan otomatis bertambah terus menerus selama program belum di-STOP.

## BAB V

### REMARKS, MESSAGE BOX, INPUTBOX

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal Remarks, Message Box dan Input Box	<ul style="list-style-type: none"><li>▪ Dapat menggunakan Remarks untuk memberikan keterangan pada kodeprogram</li><li>▪ Dapat menggunakan Message Box untuk menampilkan pesan</li><li>▪ Dapat menggunakan Input Box untuk memberikan <i>inputan</i> pada pesan</li></ul>

#### A. Remarks

*Remarks* digunakan untuk membantu memberikan keterangan seorang programmer lain yang nanti akan memodifikasi program aplikasi dikemudian hari. *Remarks* memberikan pesan – pesan yang penting berkaitan dengan program yang dibuat. *Remarks* tidak harus ditulis dengan format tertentu atau bahasa Inggris, melainkan dapat ditulis dalam bahasa Indonesia sekalipun.

Jadi *remarks* merupakan pesan atau keterangan yang ditulis dalam kode program. *Remarks* digunakan untuk membantu menjelaskan tentang kode yang ditulis dan Visual Basic akan mengabaikan semua *Remarks* yang ada pada kode program. Tujuan menambahkan *Remarks* dalam program :

- ❖ Memberikan keterangan nama programmer dan tanggal pembuatan program.
- ❖ Memberikan keterangan umum mengenai prosedur dan fungsi yang digunakan
- ❖ Memberikan keterangan pada perintah – perintah yang sulit dimengerti sehingga jika ada programmer lain yang akan memodifikasi akan mengerti maksud dari kode yang ditulis.

Visual Basic mendukung 2 macam penggunaan *Remarks* yaitu *Remarks* yang dimulai dengan menggunakan kata *Rem* dan *Remarks* yang dimulai dengan tanda petik (\*).

Berikut ini format penggunaan statement *Rem*.

```
Rem Programmer: STMIK-DB , Tanggal : 24-Feb-2009
Rem Program untuk menghitung luas segitiga.
Rem Dengan menggunakan masukan berupa
Rem 1. Tinggi segitiga (t)
Rem 2. Alas segitiga (a).
Rem Tombol hitung untuk menghitung luas segitiga
Rem Tombol exit untuk keluar dari aplikasi.
```

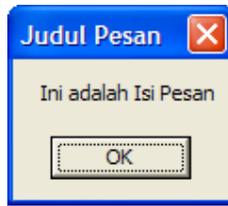
Pada contoh tersebut terdapat sejumlah *Remarks* yang menjelaskan nama *programmer* yang membuat dan tanggal pembuatannya serta kegunaan dari program tersebut. Selain menggunakan *Rem* juga dapat digunakan tanda petik satu untuk melakukan remark.

```
` Programmer: STMIK-DB , Tanggal : 24-Feb-2009
` Program untuk menghitung luas segitiga.
` Dengan menggunakan masukan berupa
` 1. Tinggi segitiga (t)
` 2. Alas segitiga (a).
` Tombol hitung untuk menghitung luas segitiga
` Tombol exit untuk keluar dari aplikasi.
```

Kedua contoh di atas memberikan pesan dan keterangan yang sama, hanya saja cara yang kedua lebih mudah.

## B. Message Box

Adakalanya sebuah program ingin menampilkan pesan kesalahan atau bertanya pada user, sebab kontrol – kontrol yang ada pada *form* kurang jelas. *Message box* tidak seperti kontrol yang melekat pada form. *Message Box* berisi sebuah message akan ditampilkan tepat diatas sebuah form dan akan hilang ketika mendapat respon dari user dengan menklik tombol yang ada pada *message box* tersebut (lihat gambar 5.1). Visual Basic menyediakan 2 cara untuk menampilkan *message box* yaitudengan menggunakan statement *MsgBox* dan menggunakan fungsi *MsgBox*



Gambar 37. Contoh pesan dengan MessageBox

1. Statement MsgBox

Pesan yang ditampilkan melalui statement *Msgbox* menampilkan tombol Ok. Ketika user selesai membaca pesan tersebut, maka user akan menekan tombol Ok untuk menutup pesan tersebut. Berikut ini Format statement MsgBox :

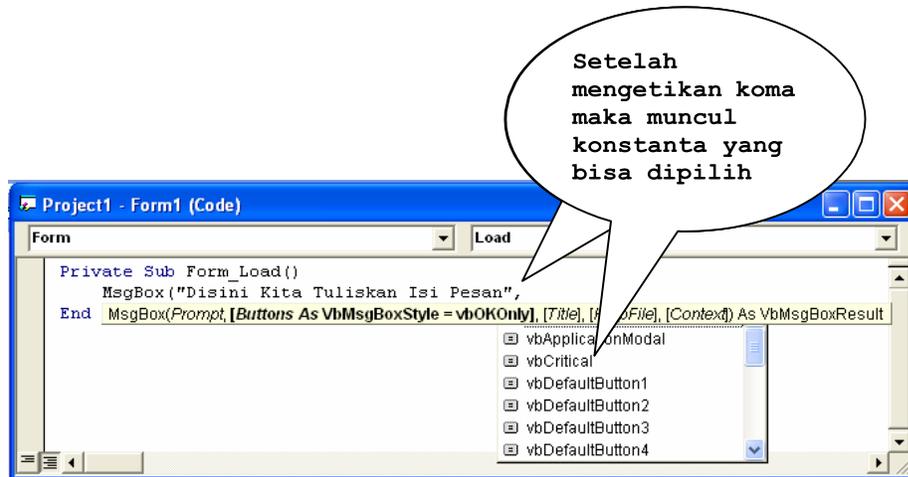
```
MsgBox Prompt, Style Value, Title
```

*Prompt* merupakan kalimat atau variabel yang berisi string yang akan ditampilkan. *Style* menentukan tipe dari command button yang akan terlihat pada message box dan nilainya dapat dipilih salah satu seperti pada tabel 9 Sedangkan *Title* menunjukkan judul dari message box.

Tabel 9. Nilai Konstanta Message Box

STYLE VALUE	KONSTANTA	TOMBOL YG DITAMPILKAN
0	vbOkOnly	Ok button
1	vbOkCancel	Ok dan Cancel buttons
2	vbAbortRetryIgnore	Abort, Retry dan Ignore buttons.
3	vbYesNoCancel	Yes, No dan Cancel buttons
4	vbYesNo	Yes dan No buttons
5	vbRetryCancel	Retry dan Cancel buttons

Kita bisa menggunakan nama konstanta atau syle value untuk menggantikan nilai integer pada argumen kedua. Penggunaan nama konstanta akan lebih mudah dibaca dibandingkan menggunakan style value. Visual Basic akan menampilkan list dari nama konstanta tersebut begitu anda mengetikan tanda koma setelah argumen pertama.



Gambar 38. Memilih Konstanta pada Message Box

## 2. Fungsi MsgBox

Untuk menampung tombol mana yang ditekan oleh user, maka digunakan fungsi MsgBox. Format pada fungsi MsgBox() sedikit berbeda dengan statement MsgBox. Fungsi ini menyediakan type yang lebih luas dibanding Statement MsgBox.

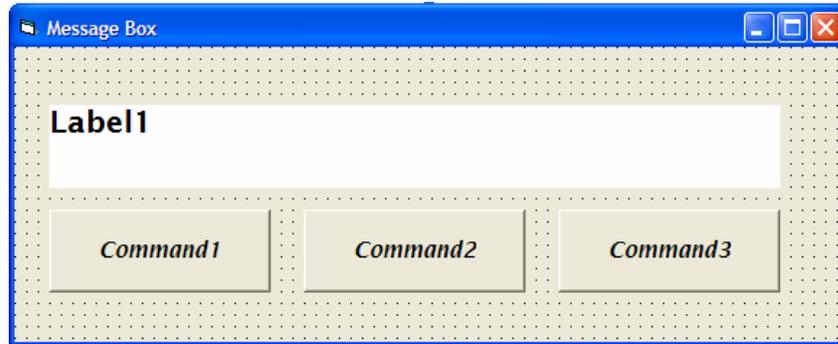
**PesanAnda=MsgBox(Prompt, Style Value, Title)**

PesanAnda adalah variable yang akan menampung nilai dari fungsi MsgBox(). Nilainya menentukan type dari tombol yang diklik oleh user. Variable tersebut haruslah dideklarasikan sebagai tipe data integer pada general declaration section.

Tabel 10. Nilai Konstanta Saat Tombol MessageBox diklik

NILAI	KONSTANTA	TOMBOL YG DIKLIK
1	vbOk	Ok button
2	vbCancel	Cancel button
3	vbAbort	Abort button
4	vbRetry	Retry button
5	vbIgnore	Ignore button
6	vbYes	Yes button
7	vbNo	No button

Contoh : Buat project baru dengan tiga command button dan label.



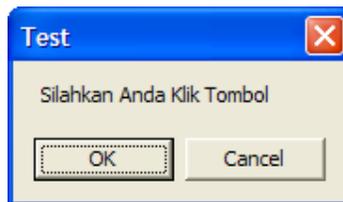
Gambar 39. Contoh program menggunakan MessageBox

Kemudian klik view code dan ketikkan kode berikut ini :

```
Private Sub Command1_Click()
    Dim testMsg As Integer
    testMsg = MsgBox("Silahkan Anda Klik Tombol", 1,
    "Test")
    If testMsg = 1 Then
        Label1.Caption = "Anda meng-klik tombol OK"
    Else
        Label1.Caption = "Anda meng-klik tombol Cancel"
    End If
End Sub
```

Kode program di atas menggunakan Style Value untuk menampilkan tombol pada message. Kode program berikut menggunakan Konstanta untuk menampilkan tombol pada message. Coba Anda ketikkan di Command2\_Click dan lihat hasilnya!

```
Private Sub Command2_Click()
    Dim testMsg As Integer
    testMsg = MsgBox("Silahkan Anda Klik
    Tombol", vbOKCancel, "Test")
    If testMsg = vbOK Then
        Label1.Caption = "Anda meng-klik tombol OK"
    Else
        Label1.Caption = "Anda meng-klik tombol Cancel"
    End If
End Sub
```



Gambar 40. Tampilan pesan menggunakan MessageBox

Ketika user meng-klik OK pada test button, maka akan muncul pesan "Anda meng-klik tombol OK" sedangkan jika user meng-klik Cancelbutton maka akan muncul pesan "Anda meng-klik tombol Cancel" Untuk membuat message box anda lebih menarik, anda bisamenambahkan icon pada message box tersebut. Ada empat tipeicon yang disediakan oleh Visual Basic seperti yang tertera padatable berikut ini :

Tabel 11 Nilai Konstanta dan Icon pada MessageBox

NILAI	KONSTANTA	ICON
16	<b>vbCritical</b>	
32	<b>vbQuestion</b>	
48	<b>vbExclamation</b>	
64	<b>vbInformation</b>	

Masukan kode berikut ini pada project yang telah anda buat sebelumnya.

```
Private Sub Command3_Click()
Dim testMsg As Integer
testMsg = MsgBox("Silahkan Anda Klik", vbYesNoCancel _
+ vbExclamation, "Test")
If testMsg = 6 Then
Label1.Caption = "Testing Successful"
ElseIf testMsg = 7 Then
Label1.Caption = "Are You Sure?"
Else
Label1.Caption = "Testing Fail"
End If
End Sub
```

### C. Input Box

InputBox akan menampilkan message dimana user dapat menginputkan suatu nilai atau message pada form tersebut. Format penggunaannya adalah sebagai berikut:

```
MyMessage=InputBox (Prompt,Title,default_text,x-position,y-position)
```

MyMessage adalah tipe data variant yang dideklarasikan sebagai string. Argumen yang tersedia dapat dijelaskan sebagai berikut:

- ☑ Prompt : Pesan yang ditampilkan.
- ☑ Title : Judul dari Input Box.
- ☑ default-text : Default text yang ditampilkan pada field dimana user dapat menggunakannya atau menggantinya.
- ☑ x-position and y-position : menunjukkan posisi dimana input box tersebut akan ditampilkan pada form

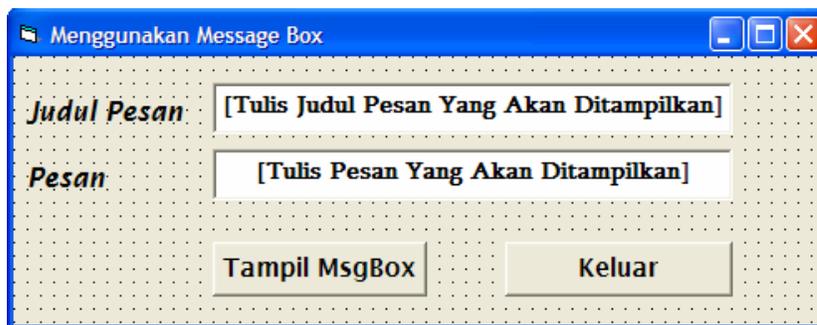
Tambahkan satu Command lagi pada project yang telah anda buat sebelumnya. Masukan kode berikut ini :

```
Private Sub Command4_Click()
Dim userMsg As String
userMsg = InputBox("Apa pesan anda?", "Message Entry Form", _
"Masukan message anda disini", 500, 700)
If userMsg <> "" Then
Label1.Caption = userMsg
Else
Label1.Caption = "No Message"
End If
End Sub
```

Ketika user meng-klik tombol OK pada Input Box, pada label akan menampilkan message yang telah anda ketik, akan tetapi jika anda meng-klik tombol Cancel maka akan ditampilkan pesan "No Message".

#### D. Contoh Program

Contoh berikut ini adalah untuk menerapkan penggunaan MessageBox. Tampilan yang akan dibuat seperti pada gambar 5.4.



Gambar 41. Contoh MessageBox

Langkah-langkah :

1. Jalankan IDE Visual Basic dan pilihlah project Standart Exe.
2. Simpanlah project tersebut dengan nama cthPesan.frm untuk form dan cthPesan.vbs untuk projectnya.
3. Pada form letakkan kontrol seperti pada gambar 5.5.

4. Edit properti dari kontrol-kontrol seperti tabel berikut :

Tabel 12. Nilai Property Contoh MessageBox

Control	Properti	Setting Value
Form1	Name	frmMsgBox
	Caption	Menggunakan Message Box
Command1	Name	cmdMsgBox
	Caption	Tampil MsgBox
Command2	Name	cmdKeluar
	Caption	Keluar
Label1	Name	lblJudulPesan
	Caption	Judul Pesan
	Font	Size = 12
Label2	Name	LblPesan
	Caption	Pesan
	Font	Size = 12
Text1	Name	TxtJudulPesan
	Font	Size = 12
	Text	[Tulis Judul Pesan Yang Akan Ditampilkan]
Text2	Name	TxtPesan
	Font	Size = 12
	Text	[Tulis Pesan Yang Akan Ditampilkan]

5. Kemudian ketik kode berikut ini :

```

Rem * ..... *
Rem * Nama Program = Contoh Message Box *
Rem * Programmer = Digdoyo Komputer *
Rem * Tgl Pembuatan = 01 Agustus 2011 *
Rem * STMIK PALANGKARAYA *
Rem * ..... *

Private Sub cmdKeluar_Click()
    End
End Sub

Private Sub cmdMsgBox_Click()
    Dim JudulPesan As String
    Dim Pesan As String
    JudulPesan = txtJudulPesan.Text
    Pesan = txtPesan.Text
    MsgBox Pesan, vbOKOnly, JudulPesan
End Sub

```

## SOAL LATIHAN

Buatlah sebuah form dengan sebuah label, sebuah Textbox, dan 2 buah command Button. Textbox tersebut digunakan untuk memasukan data berupa angka dengan range antara 0 sampai 100000. Jika seorang user memasukan huruf atau angka yang berada diluar range maka akan ditampilkan pesan kesalahan dan meminta user untuk memasukkan data kembali. Jika masukan seorang user sudah benar maka akan ditampilkan pesan bahwa masukan sudah benar dan keluar dari aplikasi.

## BAB VI

### PERCABANGAN DAN PERULANGAN

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
1. Mengenal OperatorKondisi	<ul style="list-style-type: none"><li>▪ Dapat menggunakan operator kondisi</li><li>▪ Dapat melakukan perbandingan untuk pengambilan keputusan</li></ul>
2. Mengenal OperatorLogika	<ul style="list-style-type: none"><li>▪ Dapat menggunakan operator logika</li></ul>
3. Mengenal Ekspresi Percabangan	<ul style="list-style-type: none"><li>▪ Dapat menggunakan perintah If...Then...Else</li><li>▪ Dapat menggunakan perintah Select Case</li></ul>
4. Mengenal Ekspresi Perulangan	<ul style="list-style-type: none"><li>▪ Dapat menggunakan perintah Do...Loop</li><li>▪ Dapat menggunakan perintah While...Wend</li><li>▪ Dapat menggunakan perintah For...Next</li></ul>

#### A. Operator Kondisi

Untuk mengontrol alur program dalam Visual Basic, kita bisa menggunakan bermacam-macam operator kondisi. Pada dasarnya, operator kondisi ini mirip dengan operator aritmatik. Operator kondisi merupakan alat yang sangat hebat, dengan operator tersebut kita dapat melakukan perbandingan untuk memutuskan tindakan selanjutnya. Tabel 13 menunjukkan operator kondisi yang digunakan dalam Visual Basic.

Tabel 13. Operator Kondisi

OPERATOR	MAKSUD
=	Sama dengan
>	Lebih besar
<	lebih kecil
>=	Lebih besar sama dengan
<=	Lebih kecil sama dengan
<>	Tidak sama dengan

## B. Operator Logika

Sebagai tambahan untuk operator kondisi, ada beberapa operator logika yang ditambahkan pada Visual Basic sehingga memudahkan pembuatan alur program.

Tabel 14. Operator Logika

OPERATOR	MAKSUD
And	Keduanya harus bernilai True
or	Salah satu saja yang bernilai True
Xor	Salah satu boleh bernilai True tetapi tidak boleh keduanya bernilai True
Not	Bernilai False

## C. Pernyataan If...Then...Else

### 1. Sintaks Umum

Syntax umum untuk pernyataan if...then...else adalah

```
IF <kondisi> THEN <kode program>
```

Bila <kondisi> bernilai True maka <kode program> akan dikerjakan.

```
IF <kondisi> THEN  
    <blok kode program 1>  
ELSE  
    <blok kode program 2>  
END IF
```

Bila <kondisi> bernilai True maka <blok kode program 1> akan dikerjakan, tetapi bila <kondisi> bernilai False maka <blok kodeprogram 2> yang akan dikerjakan.

```
IF <kondisi 1> THEN  
    <blok kode program 1>  
ELSEIF <kondisi 2> THEN  
    <blok kode program 2>  
ELSE  
    <blok kode program 3>  
END IF
```

Bila <kondisi 1> bernilai True maka <blok kode program 1> akan dikerjakan, kemudian bila <kondisi 2> bernilai True maka <blok kodeprogram 2> akan dikerjakan, tetapi bila <kondisi 1> dan <kondisi 2> bernilai False maka <blok kode program 3> yang akan dikerjakan.

## 2. Contoh Program

Membuat Program Input Password

Letakkan kontrol Image, TextBox, Label dan Command. Atur seperti gambar berikut :



Gambar 42. Program Input Password dengan Fungsi IF

Pengaturan property setiap object-nya adalah sebagai berikut :

Tabel 15. Property Object untuk Program Input Password

Object	Properties	Value
Form1	Caption	Struktur If...Then...
	StartUpPostion	2-CenterScreen
Image1	Stretch	True
	Picture	Lock.jpg
	Visible	False
Label1	Caption	Input Password
Text1	Name	txtPass
	PasswordChar	*
	Text	<kosong>
Command1	Name	cmdOK
	Caption	&OK

Buka Jendela Code dan pada bagian Code Editor ketikkan kode programnya sebagai berikut :

```

Private Sub cmdOK_Click()
  If txtPass.Text = "digdoyo" Then
    Image1.Visible = True
  End If
End Sub

```

Simpan program dan jalankan :

1. Ketikkan sembarang teks pada txtPass lalu klik tombol OK, maka tidak terjadi apa-apa.
2. Ketikkan "digdoyo" pada txtPass lalu klik tombol OK, maka gambar kunci (lock) akan muncul

Penjelasan kode program :

```

If txtPass.Text = "dana" Then Image1.Visible = True

```

**Kondisi**
**Kode program yang dikerjakan jika kondisi True**

Modifikasi programnya menjadi seperti berikut :

```

Private Sub cmdOK_Click()
  If txtPass.Text = "dana" Then
    Image1.Visible = True
    MsgBox "Password Benar", vbOKOnly, "Sukses"
  Else
    Image1.Visible = False
    MsgBox "Password Salah", vbOKOnly, "Gagal"
    txtPass.Text = ""
    txtPass.SetFocus
  End If
End Sub

```

Jalankan program :

1. Ketikkan sembarang teks pada txtPass lalu klik tombol OK, maka gambar tidak muncul dan muncul kotak pesan "Password Salah". Klik tombol OK pada kotak pesan tersebut. Maka txtPass dikosongkan dan kursor akan aktif di txtPass (txtPass.Setfocus)
2. Ketikkan "dana" pada txtPass lalu klik tombol OK maka gambar akan muncul dan muncul kotak pesan "Password Benar".

Catatan Tambahan :

- Teks "dana" harus diketik huruf kecil semua. Ingat : data string bersifat *case sensitive* !

- ☑ Agar password-nya tidak bersifat *case sensitive*, modifikasi pernyataan kondisinya menjadi : `If LCase(txtPass.Text) = "dana"Then`
- ☑ Fungsi `LCase` adalah untuk mengkonversi semua string yang diinput ke `txtPass.Text` menjadi huruf kecil, walaupun user menginputnya dengan huruf kapital.

TANTANGAN :

1. Modifikasilah struktur kontrol dan kode program di atas sehingga seorang user hanya mempunyai 3 kali kesempatan untuk mengetikkan password. Setelah 3 kali kesempatan dan password masih salah maka program akan otomatis berhenti.
2. Membuat program menentukan nilai seorang mahasiswa dengan kriteria sebagai berikut :

Tabel 16. Interval Nilai untuk Program Konversi Nilai

Interval Nilai	Nilai Huruf	Keterangan
80-100	A	Lulus
68-79	B	Lulus
56-67	C	Lulus
41-55	D	Remidi
0-40	E	Gagal

Atur form beserta kontrol-kontrol yang diperlukan seperti berikut :



Gambar 43. Program Konversi Nilai dengan Statement IF

Ketikkan kode program sebagai berikut :

```

Private Sub txtNilai_Change()
nilai = Val(txtNilai.Text)
If nilai >= 80 And nilai <= 100 Then
    lblHuruf.Caption = "A"
    lblKet.Caption = "Lulus"
ElseIf nilai >= 68 And nilai <= 79 Then
    lblHuruf.Caption = "B"
    lblKet.Caption = "Lulus"
ElseIf nilai >= 56 And nilai <= 67 Then
    lblHuruf.Caption = "C"
    lblKet.Caption = "Lulus"
ElseIf nilai >= 41 And nilai <= 55 Then
    lblHuruf.Caption = "D"
    lblKet.Caption = "Remidi"
Else
    lblHuruf.Caption = "E"
    lblKet.Caption = "Gagal"
End If
End Sub

```

#### D. Pernyataan Select...Case

Cara lain untuk menangani pengambilan keputusan dalam sebuah program adalah dengan menggunakan Select...Case Statement, yang mampu menangani sejumlah kondisi dari satu variabel. Select...Case serupa dengan If ... Then ..... Elseif, tetapi lebih efisien apabila percabangan bergantung kepada satu kondisi saja. Dengan menggunakan Select ... Case sebagai pengganti dari If ... Then ... Else If, akan membuat program menjadi lebih sederhana.

##### 1. Syntaks Umum

Format penggunaan *Select Case* :

```

Select Case nama_variabel
Case nilai_1
    Perintah yang akan dieksekusi jika memenuhi nilai_1
Case nilai_2
    Perintah yang akan dieksekusi jika memenuhi nilai_2
Case Else
    Perintah yang akan dieksekusi jika tidak memenuhi semua
End Select

```

Type data pada nama\_variabel harus sama dengan nilai pada Case.

##### 2. Contoh Program

1. Membuat program untuk mengetahui harga barang berdasarkan jenis barang yang dipilih di ListBox :

Atur Form beserta kontrol lain yang dibutuhkan seperti tampilanberikut:



Gambar 44. Program Harga Barang dengan StatementSelect Case

Ketikkan kode program berikut :

```
Private Sub lstBarang_Click()  
Dim Harga As Long  
Dim Satuan As String  
  
Select Case lstBarang.Text  
  Case "CD Blank"  
    Harga = 85000  
    Satuan = "Box"  
  
  Case "Buku Tulis"  
    Harga = 36000  
    Satuan = "Lusin"  
  
  Case "Kertas"  
    Harga = 28000  
    Satuan = "Rim"  
  
  Case "Bolpoin"  
    Harga = 15000  
    Satuan = "Pak"  
End Select  
  
lblHarga.Caption = Harga  
lblSatuan.Caption = Satuan  
  
End Sub
```

Catatan : variabel nilai (lstBarang.Text) adalah string sehingga nilai yang ada pada Case seperti "CD Blank" adalah tipe data string juga.

TANTANGAN :

1. Modifikasi program Harga Barang di atas (gambar 6.3).
  - ⇒ Tambahkan input jumlah barang, tambahkan pula output outputdiskon dan total yang harus dibayar.
  - ⇒ Diskon sebesar 5% dari TotalHarga. Diberikan diskon jika jumlahbeli $\geq$ 5, selain itu tidak mendapat diskon.
  - ⇒ Total Bayar = Total Harga-Diskon
2. Buka kembali contoh program konversi nilai pada gambar 6.2. Anda dapat mengganti statement IF dengan statement Select...Case. Berikut kode programnya:

```
Private Sub txtNilai_Change ()  
Dim Nilai As Single  
Nilai = txtNilai.Text  
  
Select Case Nilai  
  Case Is  $\geq$  80  
    lblHuruf.Caption = "A"  
    lblKet.Caption = "Lulus"  
  Case Is  $\geq$  68  
    lblHuruf.Caption = "B"  
    lblKet.Caption = "Lulus"  
  Case Is  $\geq$  56  
    lblHuruf.Caption = "C"  
    lblKet.Caption = "Lulus"  
  Case Is  $\geq$  41  
    lblHuruf.Caption = "D"  
    lblKet.Caption = "Remidi"  
  Case Else  
    lblHuruf.Caption = "E"  
    lblKet.Caption = "Gagal"  
End Select  
End Sub
```

Keyword Is disini untuk menentukan kondisi. Biasa digunakan untuk datanumerik.

Atau bisa juga memakai kode program seperti berikut:

```

Private Sub txtNilai_Change()
  Dim Nilai As Single
  Nilai = txtNilai.Text
  Select Case Nilai
    Case 80 To 100
      lblHuruf.Caption = "A"
      lblKet.Caption = "Lulus"
    Case 68 To 79
      lblHuruf.Caption = "B"
      lblKet.Caption = "Lulus"
    Case 56 To 67
      lblHuruf.Caption = "C"
      lblKet.Caption = "Lulus"
    Case 41 To 55
      lblHuruf.Caption = "D"
      lblKet.Caption = "Remidi"
    Case Else
      lblHuruf.Caption = "E"
      lblKet.Caption = "Gagal"
  End Select
End Sub

```

#### E. Pernyataan Do...Loop

Visual Basic mendukung beberapa versi statement *Do*. Looping (perulangan) dengan menggunakan *While* mungkin yang paling populer digunakan dalam pemrograman Visual Basic. Seperti Statement *If...Then*, *do While* juga membutuhkan ekspresi perbandingan untuk keluar dari looping

##### 1. Sintaks Umum

Ada beberapa sintak penggunaan untuk pernyataan Do..Loop

```

a) Do While <kondisi>
    <VB statement>
Loop

b) Do
    <VB statement>
Loop While <kondisi>

```

*Keterangan* : VB Statement akan diulang selama <kondisi> bernilai TRUE. Pengulangan berhenti bila <kondisi> sudah bernilai FALSE.

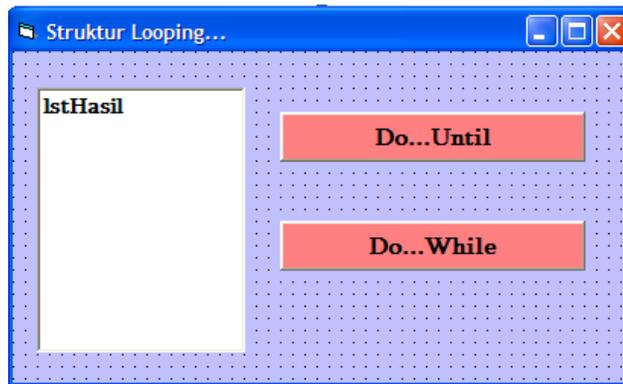
- c) **Do Until** kondisi  
 VB statement  
**Loop**
- d) **Do**  
 VB statement  
**Loop Until** kondisi

*Keterangan* : VB Statement akan diulang sampai <kondisi> bernilai TRUE. Pengulangan berhenti bila <kondisi> sudah bernilai FALSE.

## 2. Contoh Program

- a. Menampilkan angka 0 sampai 10 di ListBox dengan perintah Do...While dan Do...Until.

Atur Form beserta kontrol yang dibutuhkan seperti pada gambar 6.4



*Gambar 45. Program Menampilkan Angka dengan Do...Loop*

Ketik kode programnya seperti berikut ini :

```

Private Sub cmdUntil_Click()
  lstHasil.Clear
  i = 0
  Do Until i > 10
    lstHasil.AddItem "Angka Ke- " & i
    i = i + 1
  Loop
End Sub

Private Sub cmdWhile_Click()
  lstHasil.Clear
  i = 0
  Do While i =< 10
    lstHasil.AddItem "Angka Ke- " & i
    i = i + 1
  Loop
End Sub

```

Penjelasan program :

- ❖ Kode program tersebut akan menampilkan hasil yang sama.
  - ❖ Perhatikan pada cmdUntil\_Click(). Program akan dijalankan sampai  $i > 10$
  - ❖ Perhatikan pada cmdWhile\_Click(). Program akan dijalankan selama  $i \leq 10$
  - ❖ Kedua statement tersebut memiliki arti yang sama
- b. Cobalah ganti syntaks statement Do Until dan Do While di atas, dengan kode program berikut ini, lihat hasilnya:

```
Private Sub cmdUntil_Click()  
lstHasil.Clear  
i = 0  
Do  
    lstHasil.AddItem "Angka Ke- " & i  
    i = i + 1  
Loop Until i > 10  
End Sub  
  
Private Sub cmdWhile_Click()  
lstHasil.Clear  
i = 0  
Do  
    lstHasil.AddItem "Angka Ke- " & i  
    i = i + 1  
Loop While i <= 10  
End Sub
```

#### F. Pernyataan While...Wend

Pengulangan While...Wend akan mengeksekusi sekumpulan statement- statement perintah selama suatu kondisi itu benar. Looping atau perulangan yang menggunakan While..Wend ini mempunyai syntak sebagai berikut :

##### 1. Syntaks Umum

Ada beberapa syntak penggunaan untuk pernyataan While...Wend

```
While <kondisi>  
    VB Statement  
Wend
```

*Keterangan* : Jika kondisi benar, maka semua statement akan dieksekusi dan ketika mencapai baris Wend, control akan kembali lagi ke statement While untuk mengevaluasi kembali nilai dari kondisi,

jika nilai dari kondisi masih memenuhi syarat atau benar maka proses loop/perulangan akan terjadi lagi. Jika nilai kondisinya Salah, maka program akan keluar dari loop dan mengeksekusi perintah-perintah yang ada setelah Wend.

## 2. Contoh Program

- a. Contoh Statement While berikut akan mengevaluasi nilai numerik yang dimasukkan user lewat keyboard, dan kondisi yang dievaluasi adalah selama nilainya lebih besar atau sama dengan nol, jika nilainya negatif maka program akan berhenti.

```
Number = 0
While Number >= 0
    Number = InputBox("Silakan masukkan nilai yang lain ?")
Wend
```

- b. Contoh 1 di atas bisa dimodifikasi. Letakkan sebuah command button dan sebuah textbox pada form kemudian, ketikkan listing program berikut dalam tombol command button tersebut. (atur posisinya dengan benar).

```
Private Sub Command1_click()
Number = 0
While number >= 0
    Total = Total + Number
    Number = InputBox("Silakan masukkan nilai yang lain ?")
Wend
Text1.Text = Total
End Sub
```

## G. Pernyataan For...Next

Perulangan dengan For ... Next merupakan salah satu struktur perulangan yang sering terdapat pada banyak bahasa pemrograman, perulangan For ... Next menggunakan suatu variabel yang disebut counter untuk melakukan penambahan secara otomatis, sesuai dengan nilai awal dari variabel tersebut. Perulangan dengan For digunakan untuk mengulang statement atau satu blok statement berulang kali, sejumlah yang ditentukan.

### 1. Sintaks Umum

```
FOR <pencacah> = <awal> TO <akhir> [STEP <langkah>]
<blok kode program>
NEXT <pencacah>
```

- ⇒ <pencacah> adalah variabel (tipe: integer) yang digunakan untuk menyimpan angka pengulangan.
- ⇒ <awal> adalah nilai awal dari <pencacah>.
- ⇒ <akhir> adalah nilai akhir dari <pencacah>.
- ⇒ <langkah> adalah perubahan nilai <pencacah> setiap pengulangan. Sifatnya optional (boleh ditulis ataupun tidak). Bila tidak ditulis maka nilai <langkah> adalah 1.

## 2. Contoh Program

- a. Program menampilkan angka dari 0 sampai 10 dan sebaliknya (10 sampai 0)

Atur Form dan Kontrol yang dibutuhkan seperti gambar 6.4. Ganti Caption pada Command, masing-masing menjadi "For Next 1" dan "For Next 2".

Ketikkan kode program berikut ini :

```
Private Sub cmdFor1_click()
'Menampilkan angka 0 sampai 10 di ListBox
lstHasil.Clear
  For i = 0 To 10
    lstHasil.AddItem "Angka " & i
  Next i
End Sub

Private Sub cmdFor2_click()
'Menampilkan Angka 10 sampai 0 di ListBox
lstHasil.Clear
  For i = 10 To 0 STEP -1
    lstHasil.AddItem "Angka " & i
  Next i
End Sub
```

Jika Anda menginginkan angka dengan kelipatan 2 dst, maka tambahkanlah perintah STEP, contoh :

For i = 0 To 10 STEP 3, akan menampilkan angka dari 0 sampai 10

dengan kelipatan 3. Hasilnya : 0, 3, 6, 9

- b. Menampilkan angka genap pada ListBox. Bilangan genap adalah bilangan yang habis dibagi 2. Memungkinkan user untuk menginputkan batas awal dan akhir. Atur form dengan meletakkan ListBox, 2 TextBox, dan Command untuk proses. Buat properti Column = 3 untuk ListBox



Gambar 46. Menampilkan bilangan genap dengan For Next

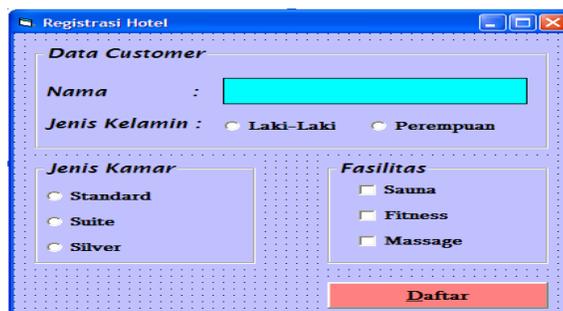
Ketikkan kode program seperti berikut

```
Private Sub cmdProses_Click()
    Dim i, Awal, Akhir As Single
    lstHasil.Clear
    Awal = txtAwal.Text
    Akhir = txtAkhir.Text
    For i = Awal To Akhir
        If i Mod 2 = 0 Then
            lstHasil.AddItem i
        End If
    Next i
End Sub
```

Bagaimana jika Anda diminta untuk menampilkan bilangan ganjil?? Silakan Anda coba sendiri. Okey 😊

## SOAL LATIHAN

Buat program Registrasi Hotel, atur tampilan form beserta kontrol yang dibutuhkan sebagai berikut :



Pada program di atas user dipersilahkan untuk mengisi namanya, jenis kelaminnya laki-laki atau perempuan, kemudian memilih jenis kamar yang diinginkan, kemudian memilih fasilitas apa saja yang diinginkan.

Tiap kamar memiliki harga yang berbeda-beda. Untuk pilihan jenis kamar gunakan Option, di sini user hanya dapat memilih salah satu saja. Sedangkan untuk fasilitas dapat menggunakan Check, di mana user dapat tidak memilih dan bebas untuk memilih yang mana saja.

Kita asumsikan harga kamar adalah sebagai berikut :

Jenis Kamar	Harga
Standard	150000
Suite	250000
Silver	450000

Jenis Fasilitas	Harga
Sauna	50000
Fitness	75000
Massage	150000

- ✓ Kemudian tampilkan harga total pembayarannya dalam bentuk Message Box dengan klik tombol Daftar.
- ✓ Minta input pembayaran dengan menggunakan InputBox. Validasilah agar pembayaran mencukupi total pembayaran. Jika pembayaran tidak mencukupi total pembayaran, maka InputBox akan muncul terus samapi pembayaran mencukupi.
- ✓ Jika terdapat uang kembali tampilkan jumlah uang kembaliannya dengan MesageBox.
- ✓ Dan ucapkan terima kasih. Setelah itu resetlah kembali form ke keadaan semula sebelum diisi.

## BAB VII

### FUNGSI-FUNGSI BAWAAN VB

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal Fungsi-Fungsi Bawaan VB	<ul style="list-style-type: none"><li>• Dapat menggunakan fungsi Numerik</li><li>• Dapat menggunakan fungsi String</li><li>• Dapat menggunakan fungsi tanggal danwaktu</li></ul>

Dalam Visual Basic, fungsi bawaan adalah suatu fungsi yang sudah ada dalam Visual Basic dan dapat digunakan secara langsung. Ada sejumlah fungsi-fungsi bawaan dalam VB diantaranya adalah fungsi untuk mengubah suatu tipe data String dari suatu TextBox menjadi tipe data numerik agar dapat digunakan untuk melakukan operasi perhitungan, fungsi yang berkaitan dengan matematika, dll.

#### A. Fungsi-Fungsi Numerik

Dengan menggunakan fungsi-fungsi bawaan yang disediakan, waktu pembuatan sebuah program dapat disingkat. Fungsi-fungsi numerik merupakan fungsi-fungsi yang berkaitan dengan tipe data numerik. Fungsi-fungsi ini dapat dikelompokkan sebagai berikut :

##### 1. Fungsi-Fungsi Konversi ke Integer

Ada tiga macam fungsi yang melakukan konversi ke tipe data integer yaitu seperti terlihat pada tabel berikut :

Tabel 17. Fungsi Konversi ke Integer

Fungsi	Keterangan
CInt()	Pembulatan nilai desimal 0.5 dan lebih ke integer yang lebih dekat
Fix()	Pemotongan suatu nilai desimal menjadi integer
Int()	Pembulatan ke bawah suatu nilai desimal

Untuk bilangan positif, fungsi Fix() dan Int() memberikan perlakuan yang sama.

Sebagai contoh, kedua pernyataan berikut ini mengembalikan nilai yang sama:

```
ans1 = Int(14.5)      `mengembalikan 14
ans2 = Fix(14.5)     `mengembalikan 14
```

Namun fungsi Fix() dan Int() memberikan perlakuan berbeda pada bilangan negatif.

Kedua pernyataan berikut memberikan pengembalian yang berbeda :

```
ans1 = Int(-14.5)    `mengembalikan -15
ans2 = Fix(-14.5)   `mengembalikan -14
```

Sedangkan fungsi CInt() mengembalikan pembulatan angka yang terdekat. Berikut ini contoh penggunaan CInt() :

```
ans1 = CInt(14.1)    `mengembalikan 14
ans2 = CInt(14.5)    `mengembalikan 14
ans3 = CInt(14.6)   `mengembalikan 15
ans2 = CInt(-14.5)  `mengembalikan -14
ans3 = CInt(-14.6)  `mengembalikan -15
```

## 2. Fungsi – Fungsi Konversi Tipe Data

Tabel 18 merupakan fungsi – fungsi yang melakukan konversi ke tipe data.

Tabel 18. Fungsi-Fungsi Konversi ke Tipe Data

Fungsi	Keterangan
CCur()	Mengkonversi argumen menjadi tipe data Currency
CDBl()	Mengkonversi argumen menjadi tipe data double(presisi ganda)
CLng()	Mengkonversi argumen menjadi tipe data Long Integer
CSng()	Mengkonversi argumen menjadi tipe data Single
CStr()	Mengkonversi argumen menjadi tipe data String
CVar()	Mengkonversi argumen menjadi tipe data Variant

Sebagai contoh misalkan sebuah data yang merupakan hasil bagi(1/7).

```
lblNilai = CSng(1/7) `ditampilkan 0.1428571
lblNilai = CDBl(1/7) `ditampilkan 0.142857142857143
```

Contoh yang lain, ketika Anda akan menghitung 2/0.5 :

```

'Contoh 1:
MsgBox 2 / CInt(0.5) '<-- error Division by zero

'Contoh 2:
MsgBox 2 / Val(0.5) '<-- error Division by zero

'Contoh 3:
MsgBox 2 / CDbl(0.5) '<-- benar menghasilkan 4

```

Mengapa contoh 1 dan 2 menghasilkan error Divison by zero? Karenabilangan pecahan di kedua contoh tersebut dikonversi ke bilangan bulat, sehingga nilai 0.5 (nol koma lima) dianggap sebagai angka nol saja. Semua angka yang dibagi dengan nol maka akan menghasilkan error . Jadi, solusinya gunakan cara pada contoh 3.

### 3. Fungsi – Fungsi Matematika

Fungsi-fungsi yang berkaitan dengan matematika seperti terlihat padatabel 19.

Tabel 19. Fungsi-Fungsi Matematika

Fungsi	Keterangan
Abs()	Mengembalikan nilai absolut dari argumen
Atn()	Mengembalikan nilai Arc Tangen dari argumen dalam bentuk radian
Cos()	Mengembalikan nilai Cosinus dari argumen dalam radian
Exp()	Mengembalikan nilai Eksponensial dari argumen
Log()	Mengembalikan nilai Logaritma dari argumen
Sin()	Mengembalikan nilai Sinus dari argumen dalam radian
Sqr()	Mengembalikan nilai Akar dari argumen
Tan()	Mengembalikan nilai Tengen dari argumen dalam radian

### 4. Fungsi – Fungsi String

Selain fungsi-fungsi numerik, Visual Basic juga mempunyai sejumlah fungsi-fungsi String diantaranya seperti pada tabel 20.

Tabel 20. Fungsi-Fungsi String

Fungsi	Keterangan
LCase()	Mengembalikan argumen string sebagai tipe data string huruf kecil
UCase()	Mengembalikan argumen string sebagai tipe data string huruf besar
Val()	Mengembalikan nilai number dari argumen string
Len()	Mengembalikan nilai number yang menunjukkan panjang string

Substring merupakan bagian dari suatu string. Untuk memperoleh substring dari suatu string dapat digunakan 3 fungsi berikut :

a. Left(StringVal, length)

Fungsi ini mengembalikan nilai string dari sebelah kiri sebanyak *length* karakter .

b. Right(StringVal, length)

Fungsi ini mengembalikan nilai string dari sebelah kanan sebanyak *length* karakter .

c. Mid(StringVal, startVal, length)

Fungsi ini mengembalikan nilai string dari *startVal* sebanyak *length* karakter .

Contoh :

Hurufkecil = LCase("Dwi Apri")	'HurufKecil = "dwi apri"
hurufbesar = UCase("Setyorini")	'HurufBesar = "SETYORINI"
Title = "STMIK Duta Bangsa"	
lTitle = Left(Title, 3)	'lTitle = "STM"
rTitle = Right(Title, 5)	'RTitle = "angsa"
mTitle = Mid(Title, 3, 8)	'MTitle = "MIK Duta"
Panjang = Len(Title)	'length = 17

5. Fungsi – Fungsi Tanggal dan Waktu

Jika dalam suatu aplikasi diperlukan pengambilan tanggal atau waktu dari sistem, Visual Basic menyediakan fungsi – fungsi Now(), Date(), dan Time(). Lihat tabel berikut :

Tabel 21. Fungsi-fungsi Tanggal dan Waktu

Fungsi	Keterangan
Now()	Mengembalikan tanggal dan waktu sistem
Date()	Mengembalikan tanggal dari sistem
Time()	Mengembalikan waktu dari sistem
DateDiff	Mendapatkan selisih dari dua buah tanggal

Untuk mencari selisih dari dua buah tanggal tidaklah sulit, karena dalam visual basic telah disediakan fasilitas untuk melakukan hal tersebut dengan menggunakan fungsi "DateDiff ", yaitu sebuah fungsi yang digunakan untuk mendapatkan selisih dari dua buah tanggal. Dari fungsi *DateDiff* kita bisa mendapatkan selisih hari, bulan dan tahun dari dua buah tanggal.

Cara penggunaan dari fungsi ini adalah sbb:

```

Dim hari, bulan, tahun

'Untuk mencari selisih hari
hari=DateTime.DateDiff("d",CDate(Text1.Text),CDate(Text2.Text))

'Untuk mencari selisih bulan
bulan = DateTime.DateDiff("m",CDate(Text1.Text),CDate(Text2.Text))

'Untuk mencari selisih tahun
tahun = DateTime.DateDiff("yyyy",CDate(Text1.Text),CDate(Text2.Text))

```

Selain fungsi-fungsi diatas terdapat sebuah fungsi yaitu format() yang berkaitan dengan fungsi-fungsi Tanggal dan waktu.

```

Format(Ekpresi[,format[,firstdayofweek[, firstweekofyear]])

```

Fungsi ini mengembalikan nilai variant atau string sesuai dengan format yang ditentukan.

## B. Contoh Program

1. Membuat Marquee untuk Form Caption, dimana caption dari form yang kita gunakan akan berjalan dari kanan kekiri secara terus menerus. Kontrol yang digunakan adalah satu buah kontrol timer yang intervalnya kita set menjadi 100 atau bisa kita ganti sesuai keinginan kita.

Ketikkan kode program seperti berikut :

```

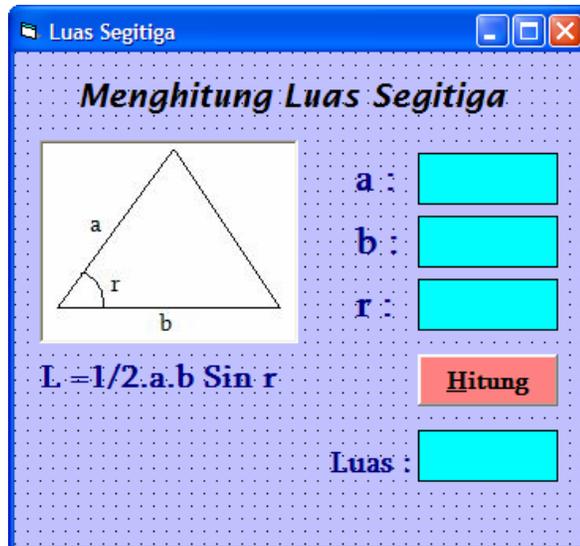
Private Sub Form_Load()
    Form1.Caption = " [ Belajar VB 6.0 ] "
End Sub

Private Sub Timer1_Timer()
    a = Left(Form1.Caption, 1)
    b = Len(Form1.Caption)
    c = Right(Form1.Caption, b - 1)
    Form1.Caption = c + a
End Sub

```

2. Menghitung Luas Segitiga, jika diketahui panjang dua sisi segitiga dan besar sudut yang diapit oleh segitiga tersebut. Untuk menyelesaikan masalah tersebut dapat digunakan fungsi berikut :  $L = \frac{1}{2} \cdot a \cdot b \cdot \sin r$  Catatan: sudut r yang Anda masukkan dalam ukuran derajat, untuk itu perlu dirubah terlebih dahulu sebelum dimasukkan dalam fungsi yang disediakan oleh VB. (Rad = Phi/180)

Rancang formnya seperti berikut ini :



Ketik kode programnya sebagai berikut :

```

Private Sub cmdUsia_Click()
A = Val(txtA.Text)
B = Val(txtB.Text)
R = Val(txtR.Text)
hasil = 0.5 * A * B * Sin(R * (3.14 / 180))
txtLuas.Text = Round(hasil, 2)
End Sub

```

## SOAL LATIHAN

1. Buatlah program yang dapat menampilkan hari dan tanggal saat ini :Contoh Formnya :



2. Buat program untuk menghitung usia seseorang. Ketika program dijalankan muncul *inputbox* dan user diminta untuk memasukkan tanggal lahir. Tampilannya seperti gambar berikut:

Tanggal Lahir

Masukkan tanggal lahir Anda  
Format :dd/mm/yyyy

17/04/1983

OK  
Cancel

Setelah user memasukkan tanggal lahir, dan mengklik tombol OK, muncul *messagebox* yang menampilkan usia Anda. Tampilannya sepertiberikut :

Usia

Usia Anda 25 Tahun 11 Bulan

OK

## BAB VIII

### SUBROUTINE PROCEDURE, FUNCTION PROCEDURE and MODULE

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal <i>subroutine procedure, function procedure</i> dan <i>module</i>	<ul style="list-style-type: none"><li>• Dapat menjelaskan prosedur fungsi, subrutin dan modul</li><li>• Dapat membuat dan menggunakan subrutin dan fungsi</li></ul>

#### A. Pengantar

Dalam kenyataan seringkali program yang harus ditulis cukup panjang, sehingga kesalahan yang mungkin dibuat oleh seorang *programmer* semakin besar. Untuk mengatasi masalah tersebut dapat dilakukan dengan memecah program tersebut menjadi bagian-bagian kecil (rutin) tetapi tetap logis. Rutin-rutin kecil tersebut akan membuat penelusuran dan perawatan program menjadi lebih mudah dan terstruktur. Rutin-rutin kecil tersebut sering disebut dengan nama prosedur.

Dalam Visual Basic terdapat 2 macam prosedur yaitu :

1. Prosedur umum (*general purpose procedure*) merupakan prosedur yang ditemukan di dalam daftar *Drop Down* pada jendela kode.
2. Prosedur Kejadian (*Event procedure*), prosedur yang berisi kode yang dijalankan ketika suatu kejadian dari kontrol di bangkitkan.

Sedangkan prosedur-prosedur yang ditambahkan dalam sebuah program tersebut disebut *subprogram*. Ada 2 jenis subprogram dalam Visual Basic yaitu prosedur subrutin (*subroutine procedures*) dan prosedur fungsi (*function procedures*)

## B. Sub Rutin

Subrutin merupakan prosedur umum (*general porpuse procedure*) yang ditulis dan ditambahkan dalam program. Format penulisan sebuah subrutin adalah sebagai berikut:

```
Sub nama_subRutin [(argumen)]  
    .  
    .  
    .  
End Sub
```

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

Contoh :

```
Sub TotalPembelian()  
    ' Inisialisasi variabel  
    Dim Total As Currency  
    Dim Disc As Single  
  
    'mendefinisikan nilai variabel  
    Total = txtTotal.Text  
    Disc = 0.2  
  
    lblTotal.Caption = Total - Total * Disc  
End Sub
```

Sedangkan untuk memanggil sebuah subrutin digunakan format sebagai berikut :

```
[Call] SubName [(Argumen list)]
```

Untuk memanggil subrutin dari contoh diatas dapat digunakan kode sebagai berikut:

```
Call TotalPembelian()
```

atau

```
TotalPembelian
```

Untuk memanggil sebuah subrutin dapat digunakan perintah *Call* diikuti dengan nama subrutin diikuti dan tanda kurung atau langsung memanggil nama subrutin saja.

## C. Fungsi

Subrutin dan Fungsi sebenarnya mirip yaitu prosedur umum (*general purpose procedure*) yang ditulis dan ditambahkan dalam program.

Bedanya Fungsi mengembalikan nilai sedangkan Subrutin tidak. Format penulisan sebuah Fungsi adalah sebagai berikut.

```
Function nama_Fungsi [(argumen)] as [ReturnValType]
    .
    .
    .
End Function
```

Contoh :

```
Function Total_Pembelian (Total As Currency)
' Inisialisasi variabel
  Dim Disc As Single

' Mendefinisikan nilai variabel
  Disc = 0.2
  Total_Pembelian = Total - Total * Disc

End Function
```

Pada contoh di atas merupakan fungsi TotalPembelian yang dimodifikasi. Total tidak lagi diambil dari nilai textbox (txtTotal), melainkan dikirim melalui suatu argumen. Argumen yang dikirimkan melalui suatu subrutin atau fungsi dapat digunakan langsung seperti sebuah variabel. Untuk memanggil fungsi dari contoh di atas dapat digunakan kode sebagai berikut:

```
'Memasukkan nilai Total sebesar 200
Total_Pembelian (200)
```

#### D. Modul (Module)

Penulisan sebuah modul dapat dilakukan pada sebuah file berekstensi .Bas. File tersebut dapat dibuat melalui menu *Project-Add Module*. Secara default Visual Basic memberikan nama *Module1*.

Pada Module, Anda dapat mendeklarasikan variabel dengan awalan DIM, PRIVATE maupun PUBLIC, dimana awalan DIM dan PRIVATE membentuk variabel modul level (hanya berlaku didalam pemakaian variabel bersangkutan), sedangkan awalan PUBLIC akan membentuk variabel global yang akan berfungsi bagi keseluruhan program.

Contoh :

```
Dim A As Integer 'Variabel A adalah modul level  
Private B As Integer 'Variabel B adalah modul level  
Public C As Integer 'Variabel C dapat digunakan  
                        'oleh program keseluruhan
```

Subrutin dapat ditulis pada module. Subrutin tersebut dapat dipanggil kapan saja dari aplikasi yang sama dengan format sebagai berikut:

```
[nama_module.]nama_subrutin
```

Contoh :

```
Module1.TotalPembelian
```

atau

```
TotalPembelian
```

Pemanggilan dapat dilakukan dengan menyebut nama module terlebih dahulu diikuti dengan tanda titik kemudian nama subrutin atau langsung nama subrutin.

#### E. Contoh Program

Membuat program untuk mengkonversi temperatur Celcius ke temperatur Fahrenheit. User diminta untuk memasukkan suhu dalam Fahrenheit dengan fasilitas `InputBox`, setelah diklik `OK` muncul `messagebox` yang menampilkan konversi suhu dalam Celcius.

Ketikkan kode program berikut ini :

```
'Fungsi Konversi Ke Celcius  
Function Celcius(fDerajat)  
Celcius = (fDerajat - 32) * 5 / 9  
End Function  
  
Private Sub Form_Load()  
'Menampilkan InputBox untuk Input Suhu dalam Fahrenheit  
'Celcius(Temp): memanggil fungsi Celcius dengan besar suhu  
                'sesuai yang dimasukkan diinputbox  
'Perintah Round untuk menampilkan angka desimal dengan  
                'mengatur berapa angka di belakang koma  
  
Dim Temp  
Temp = InputBox("Silakan masukkan suhu dalam Fahrenheit",  
"Suhu")  
MsgBox "Suhu dalam Celcius adalah : " & _  
Round(Celcius(Temp), 2) & " Derajat", vbOKOnly, "Suhu  
dalam Celcius"  
End Sub
```

## SOAL LATIHAN

Modifikasi contoh program konversi suhu di atas, buat supaya user bisa memilih menu untuk konversi suhu. Contoh tampilan programnya seperti berikut :

Program Konversi Suhu

**Program Konversi Suhu**

Saya ingin mengkonversikan

Satuan Suhu dalam derajat

Angka di belakang koma

**Hasil**

Celcius :  Kelvin :

Fahrenheit :  Reamur :

User memasukkan nilai dan memilih satuan suhu yang akan dikonversi. User juga menentukan nilai desimal dari hasil yang akan ditampilkan. Kemudian secara otomatis, hasil akan ditampilkan dalam Celcius, Fahrenheit, Kelvin dan Reamur

# BAB IX

## DATABASE

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal Kontrol Database, Fungsi-fungsisederhana kontrol database	<ul style="list-style-type: none"><li>• Dapat menggunakan kontrol database</li><li>• Dapat menggunakan fungsi-fungsi sederhana dari kontrol database</li><li>• Dapat menghubungkan textbox dengandatabase melalui kontrol database</li><li>• Dapat menghubungkan databasedengan kode program</li></ul>

### A. Pengertian Database

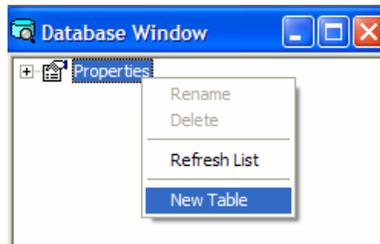
Database merupakan bagian dari kehidupan kita sehari-hari meskipun sering tidak disadari. Sebagai contoh di STMIK Duta Bangsa, database digunakan untuk menyimpan data para mahasiswa, dosen, jadwal kuliah, nilai masing-masing mahasiswa, dan lain-lain.

Database merupakan sekumpulan data yang saling berhubungan, didesain untuk menyediakan informasi pada sebuah organisasi. Sedangkan DBMS (The Database Management System) merupakan perangkat lunak yang digunakan untuk mendefinisikan, membuat, mengatur, dan menyediakan akses pada database. Microsoft Access merupakan contoh software RDBMS (Relation DBMS).

#### 1. Membuat Database dan Tabel dengan Visual Data Manager

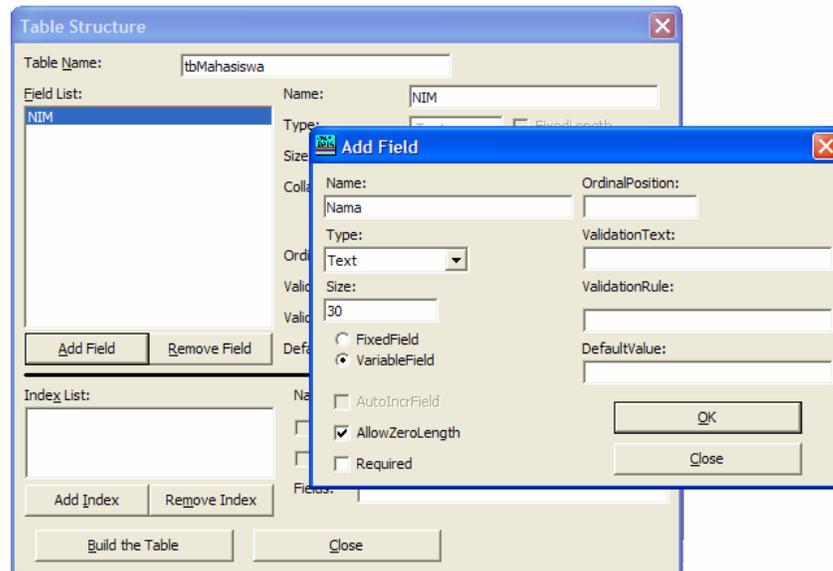
Untuk membuat database dengan Visual Data Manager, lakukan langkah-langkah sebagai berikut :

- a. Pilih menu Add Ins pada menu utama Visual Basic, kemudian pilih Visual Data Manager
- b. Pilih File – New – Microsoft Access – Version 7.0 MDB
- c. Berikan nama database(misal dbAkademik), tentukan tempat dimana Anda akan menyimpan database, kemudian klik Save.
- d. Untuk membuat tabel, klik kanan Properties, pilih New Table



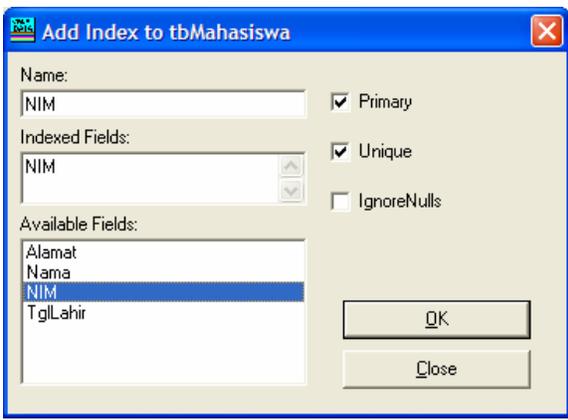
Gambar 48. Membuat tabel

- e. Isikan nama tabel, klik Add Field, isikan nama field, tipe data dan tentukan Filed Sizenya, Klik OK.



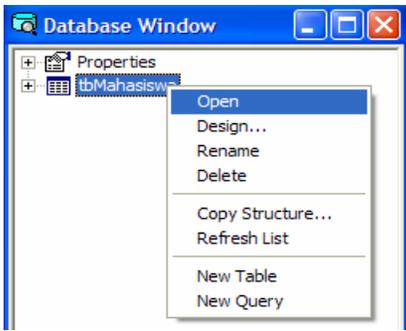
Gambar 49. Membuat Struktur Tabel

- f. Untuk membuat Primary Key pada Field, klik Add Index, isikan Name (misal: NIM), pada Available Fields pilih Field yang akan dijadikan Primary Key (misal pilih NIM)
- g. Tentukan pilihan Index (Primary, Unique, IgnoreNulls). Primary digunakan sebagai pengenal suatu record, jika tabel yang dihubungkan berhubungan dengan tabel yang lain (dalam satu tabel hanya boleh ada satu primary key). Unique dipilih jika ingin indeks bersifat unik, IgnoreNulls digunakan untuk mengabaikan field yang kosong.



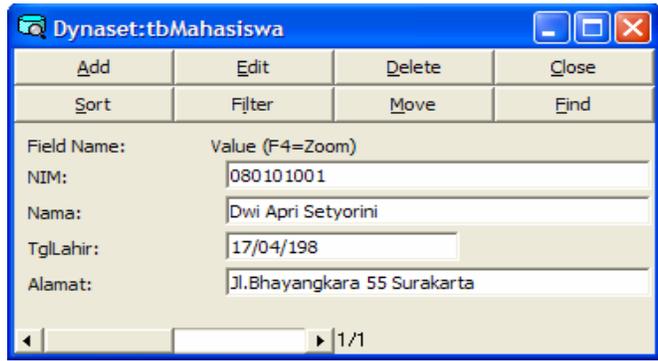
Gambar 50. Menentukan Primary Key

- h. Jika Anda sudah membuat semua Field yang dibutuhkan, maka klik Build the Table
2. Mengisi Data Pada Tabel
- a. Klik kanan pada nama tabel (tbMahasiswa, pilih Open



Gambar 51. Membuka Tabel

- b. Muncul window Dynaset : Mahasiswa



Gambar 52. Mengisikan data pada tabel

Fungsi menu yang ada pada tabel tbMahasiswa

- Add : menambah record baru

- Edit : mengoreksi record
- Delete: menghapus record aktif
- Find : mencari record
- Refresh : menampilkan ulang setelah record mengalami perubahan
- Close : menutup tabel

B. Data Access Object (DAO)

DAO (Data Access Object) merupakan suatu Object Data Access Yang berukuran besar meliputi kemampuan Data Definition, Data Manipulation dan Database Maintenance. Untuk mengakses database menggunakan DAO dibutuhkan object Data. Jika dikoneksikan dengan Microsoft Access DAO hanya bisa dikoneksikan dengan Access 97.



Gambar 53. Objek Data

Beberapa properti yang dimiliki oleh objek Data dapat Anda lihat pada tabel berikut :

Tabel 22. Property Objek Data

Properti	Keterangan
Connect	Hubungan file data, bias bersumber dari Access,dBase, Excel, Foxpro, Lotus, Paradox, dan file Text.
DatabaseName	Nama file database
RecordSource	Tabel dalam database
EOF Action	Pengecekan pergerakan kursor. Jika kursor sampai pada akhir file, maka terdapat pilihan property :MoveLast (record terakhir), EOF (akhir file), AddNew (menambah record baru)
Recordset Type	Tipe record : 0-Table, 1-Dynaset, 2-SnapShot

1. Koneksi Database dengan DAO

Untuk koneksi database dengan DAO, Anda harus menambahkan objek Data pada Form. Ada dua cara koneksi database menggunakan DAO

- a. Dengan mengatur property dari objek Data

Tabel 23. Properti Kontrol Data dbAkademik

Property	Setting Value
Name	datAkademik
Caption	dbAkademik
Connect	Access
DatabaseName	dbAkademik
Recordsource	tbMahasiswa

- b. Pengaturan dengan menetikkan kode program seperti berikut :

```
Private Sub Form_Load()  
datAkademik.DatabaseName = App.Path &  
"\dbAkademik.mdb"  
datAkademik.RecordSource = "tbMahasiswa"  
End Sub
```

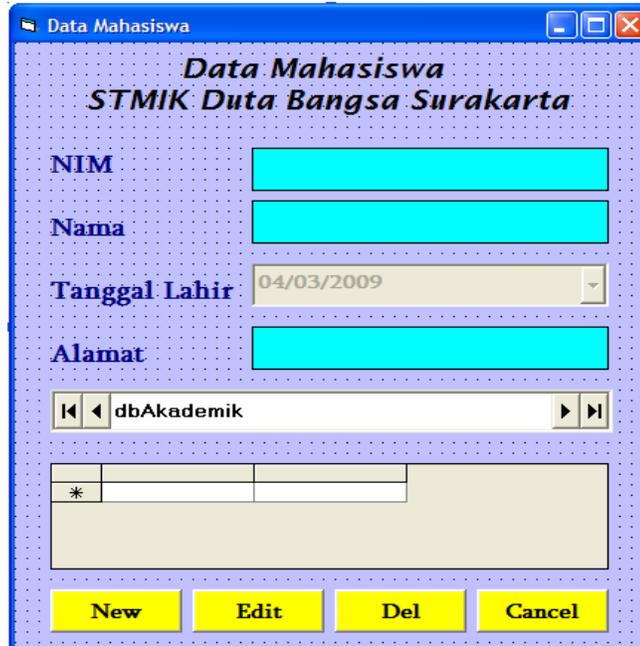
Koneksi database cara pertama masih bersifat statis yang artinya jika program/file dipindah ke folder atau drive lain, maka pengaturan properti DatabaseName harus disesuaikan dengan program/file berada.

Koneksi database cara kedua, sifatnya dinamis. App.Path digunakan untuk mendapatkan folder aktif.

## 2. Contoh Program Menggunakan DAO

Berikut adalah contoh program olah data Mahasiswa di STMIK Palangkaraya. Atur Form seperti pada contoh gambar 9.7.

- Tanggal lahir menggunakan DTPicker. Untuk menambahkan DTPicker di toolbox adalah dari menu Project - Components, pilih tab Controls- kemudian pilih Microsoft Windows Common Controls-2 6.0
- Grid menggunakan DBGrid. Untuk menambahkan DBGrid di toolbox adalah dari menu Project - Components, pilih tab Controls- kemudian pilih Microsoft Data Bound Grid Control 5.0. Atur properti Datasource dari DBGrid, pilih datAkademik.



Gambar 54. Program olah data mahasiswa dengan DAO

Ketik kode program berikut :

1. Koneksi Database

```

Private Sub Form_Load()
    'Koneksi database
    dataAkademik.DatabaseName = App.Path & "\dbAkademik.mdb"
    dataAkademik.RecordSource = "tbMahasiswa"
End Sub

```

```

Private Sub cmdNew_Click()
    If cmdNew.Caption = "New" Then
        txtNIM.Enabled = True
        txtNIM.SetFocus
        cmdNew.Caption = "Save"
        cmdEdit.Enabled = False
        cmdDel.Enabled = False
    Else
        With dataAkademik.Recordset
            .AddNew 'Menambahkan data baru
            !NIM = txtNIM.Text
            !Nama = txtNama.Text
            !Alamat = txtAlamat.Text
            !TglLahir = dtpLahir.Value
            .Update
        End With
        cmdCancel_Click ' memanggil cmdCancel_Click
    End If
End Sub

```

```

Private Sub cmdEdit_Click()
If cmdEdit.Caption = "Edit" Then
    txtNIM.Enabled = True
    txtNIM.SetFocus
    cmdEdit.Caption = "Update"
    cmdNew.Enabled = False
    cmdDel.Enabled = False
Else
    With datAkademik.Recordset
        .Edit 'Perintah untuk edit data
        !Nama = txtNama.Text
        !Alamat = txtAlamat.Text
        !TglLahir = dtpLahir.Value
        .Update
    End With
    cmdCancel_Click ' memanggil cmdCancel_Click
End If
End Sub

Private Sub cmdDel_Click()
If cmdDel.Caption = "Del" Then
    txtNIM.Enabled = True
    txtNIM.SetFocus
    cmdDel.Caption = "Delete"
    cmdNew.Enabled = False
    cmdEdit.Enabled = False
Else
    pesan = MsgBox("Apakah Anda yakin menghapus NIM " & _
txtNIM.Text & " ?", vbYesNo, "Hapus Data")
    If pesan = vbYes Then
        datAkademik.Recordset.Delete
    End If
    cmdCancel_Click ' memanggil cmdCancel_Click
End If
End Sub

```

## 2. Menghapus data

```

Private Sub txtNIM_KeyPress(KeyAscii As Integer)
Dim Cari
Cari = "NIM = '" & txtNIM.Text & "'"
datAkademik.Recordset.FindFirst Cari 'Cari NIM dari record pertama

```

```

If KeyAscii = 13 Then 'Jika ditekan Enter
With datAkademik.Recordset
  If cmdNew.Caption = "Save" Then
    If .NoMatch Then 'Jika tidak ditemukan NIM
      txtNama.Enabled = True
      txtAlamat.Enabled = True
      txtNama.Text = ""
      txtAlamat.Text = ""
      dtpLahir.Enabled = True
      dtpLahir.Value = Now
      txtNama.SetFocus
    Else
      txtNama.Text = !Nama
      txtAlamat.Text = !Alamat
      dtpLahir.Value = !TglLahir
      MsgBox "NIM " & txtNIM.Text & " sudah ada"
      txtNIM.SetFocus
    End If

  ElseIf cmdEdit.Caption = "Update" Then
    If .NoMatch Then
      MsgBox "NIM " & txtNIM.Text & " belum ada"
      txtNIM.SetFocus
    Else
      txtNama.Enabled = True
      txtAlamat.Enabled = True
      dtpLahir.Enabled = True
      txtNama.SetFocus
      txtNama.Text = !Nama
      txtAlamat.Text = !Alamat
      dtpLahir.Value = !TglLahir
    End If

  ElseIf cmdDel.Caption = "Delete" Then
    If .NoMatch Then
      MsgBox "NIM " & txtNIM.Text & " belum ada"
      txtNIM.SetFocus
    Else
      txtNama.Text = !Nama
      txtAlamat.Text = !Alamat
      dtpLahir.Value = !TglLahir
    End If
  End If
End With
End If

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

```

```

Private Sub cmdCancel_Click()
'Mengembalikan ke keadaan awal

txtNIM.Text = ""
txtNama.Text = ""
txtAlamat.Text = ""
dtpLahir.Value = Now

txtNIM.Enabled = False
txtNama.Enabled = False
txtAlamat.Enabled = False
dtpLahir.Enabled = False

cmdNew.Caption = "New"
cmdEdit.Caption = "Edit"
cmdDel.Caption = "Del"

cmdNew.Enabled = True
cmdEdit.Enabled = True
cmdDel.Enabled = True

DBGrid1.Refresh

End Sub

```

### C. Active X Data Object

ADO (*ActiveX Data Object*) merupakan suatu Object Data yang mempunyai kemampuan Data Definition, Data Manipulation dan Database Maintenance serta dapat untuk membangun koneksi dengan beberapa jenis database. Untuk mengakses database menggunakan ADO ada dua fasilitas yang dapat dipakai yaitu

- ADODC

ADODC adalah sebuah object sehingga dapat dilihat atau ditambahkan di toolbox. Untuk menambahkan objek ini maka lakukan langkah sebagai berikut :

Klik menu Project, pilih Component (atau tekan Ctrl-T), tandai dengan memberi tanda cek pada Microsoft ADO Data Control 6.0 (OLEDB). Akhiri dengan tekan tombol OK, maka pada Toolbox akan muncul objek ADODC.

- ADODB

Salah satu cara menghubungkan aplikasi dengan database melalui kode program, tanpa menggunakan objek.

Database yang dapat dikoneksi dengan ADO selain Microsoft Access, dapat juga dengan Foxpro, ODBC, SQLServer atau MySQL.

## 1. Koneksi Database dengan ADODB

Di dalam Visual Basic, Object ADODB yang sering digunakan adalah ADODB.Connection dan ADODB.Recordset. Untuk dapat membuat suatu variable dari ADODB ini kita harus terlebih dahulu memilih references di dalam Visual Basic yaitu Microsoft ActiveX Data Object. Lintasan langkah berikut :

- Pilih menu Project – References
- Pilih Microsoft ActiveX Data Objects 2.1 Library – Klik OK

Module sering digunakan di dalam deklarasi Connection dan Recordset untuk penggunaan database , sehingga Object Connection dan Recordset tersebut dapat digunakan padasemua bagian di dalam project yang anda buat.

- Connection digunakan untuk melakukan koneksi ke database yang dipilih dengan perintah open ataupun mengeksekusi sintaks-sintaks SQL dengan perintah execute. Untuk membuka koneksi ke database gunakan perintah open yang diikuti oleh connection string yang dapat dihapalkan ataupun menggunakan bantuan seperti control ADODC atau file extension .UDL
- Recordset digunakan untuk menampung data (bisa berasal dari 1/ lebih tabel) yang merupakan hasil eksekusi perintah sql select,syntax:

*select [nama kolom] from [nama tabel] where [kondisi]*

Tambahkan Module pada program VB Anda (Project – Add Module), kemudian ketikkan kode program koneksi database berikut di Module Anda

```
Public Con As ADODB.Connection
Public Sub OpenConnection()
    Set Con = New ADODB.Connection
    Con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
        "Data Source=" & App.Path & "\dbAkademik.mdb;" & _
        "Persist Security Info=False"
    Con.CursorLocation = adUseClient
    Con.Open
End Sub
```

Hasil dari koneksi di atas adalah Con akan membuka koneksi ke database dengan nama dbAkademik.mdb yang terletak di folder aktif dbAkademik.mdb.

## 2. Contoh Program Menggunakan ADODB

Berikut adalah contoh program olah data Mahasiswa di STMIK Duta Bangsa. Atur Form seperti pada contoh gambar 9.8.

- NIM akan diinput secara otomatis berdasarkan Program Studi, Jenjang, Tahun Masuk dan urutan mahasiswa saat daftar ulang.
- Jenjang Sarjana menawarkan dua program studi yaitu : Sistem Informasi dan Teknik Informatika. Jenjang Diploma Tiga menawarkan dua program studi yaitu : Manajemen Informatika dan Teknik Komputer.
- Kita akan menggunakan MSFlexGrid untuk menampilkan data. Cara menambahkan project MSFlexGrid : Pilih menu Project – Components, Pilih Microsoft FlexGrid Control 6.0 – Klik OK

Berikut tampilan Form yang digunakan :

The screenshot shows a Windows application window titled "Data Mahasiswa" for "STMIK PALANGKARAYA". The form contains the following fields:

- NIM: A text input field with a cyan background.
- NAMA: A text input field.
- JENJANG: A dropdown menu.
- TGL LAHIR: A date picker.
- PRODI: A dropdown menu.
- ALAMAT: A text input field.

Below the form is a table with two columns and one row. At the bottom of the window are five buttons: NEW, EDIT, DELETE, CANCEL, and EXIT.

Gambar 55. Program olah data mahasiswa dengan ADODB

Ketik kode program berikut :

### a. Koneksi Database

Ketik kode program koneksi database berikut pada Module

```

Public Con As ADODB.Connection
Public rsMahasiswa As ADODB.Recordset
Public Sub OpenConnection()
    Set Con = New ADODB.Connection
    Con.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0;" & _
        "Data Source=" & App.Path & "\dbAkademik.mdb;" & _
        "Persist Security Info=False"
    Con.CursorLocation = adUseClient
    Con.Open
End Sub

```

b. Kode program di Form\_Load

```

Private Sub Form_Load()
OpenConnection 'Memanggil koneksi database dari Module

'Mengkoneksikan rsMahasiswa ke tabel tbMahasiswa
Set rsMahasiswa = New ADODB.Recordset
rsMahasiswa.Open "Select * From tbMahasiswa", Con, adOpenStatic,
adLockOptimistic

'Memanggil sub TampilData
'Digunakan untuk menampilkan data di Grid
TampilData

'Mengisikan item di cbJenjang
cbJenjang.AddItem "Sarjana"
cbJenjang.AddItem "Diploma 3"
End Sub

```

c. Kode program untuk mengatur Grid

```

Sub AturGrid()
    Grid.Rows = 1
    Grid.Cols = 7
    Grid.ColWidth(0) = 500
    Grid.ColWidth(1) = 1000
    Grid.ColWidth(2) = 2000
    Grid.ColWidth(3) = 2000
    Grid.ColWidth(4) = 2000
    Grid.ColWidth(5) = 2000
    Grid.ColWidth(6) = 2000

    Grid.TextMatrix(0, 0) = "No"
    Grid.TextMatrix(0, 1) = "NIM"
    Grid.TextMatrix(0, 2) = "Jenjang"
    Grid.TextMatrix(0, 3) = "Program Studi"
    Grid.TextMatrix(0, 4) = "Nama Mahasiswa"
    Grid.TextMatrix(0, 5) = "Tanggal Lahir"
    Grid.TextMatrix(0, 6) = "Alamat"
End Sub

```

d. Kode program untuk menampilkan data di Grid

```
Sub TulisData()  
Dim i As Integer  
i = 0  
Do While Not rsMahasiswa.EOF  
i = i + 1  
Grid.AddItem (i & vbTab & rsMahasiswa(0) & vbTab & _  
rsMahasiswa(1) & vbTab & rsMahasiswa(2) & vbTab & _  
rsMahasiswa(3) & vbTab & rsMahasiswa(4) & vbTab & _  
rsMahasiswa(5))  
rsMahasiswa.MoveNext  
Loop  
End Sub  
Sub TampilData()  
Set rsMahasiswa = New ADODB.Recordset  
rsMahasiswa.Open "Select * From tbMahasiswa " & _  
"Order By NIM", Con, adOpenStatic, adLockOptimistic  
AturGrid  
TulisData  
End Sub
```

e. Kode program saat cbJenjang dan cbProdi diklik

```
Private Sub cbJenjang_Click()  
cbProdi.Enabled = True  
cbProdi.SetFocus  
Select Case cbJenjang.ListIndex  
Case 0  
cbProdi.Clear  
cbProdi.AddItem "Sistem Informasi"  
cbProdi.AddItem "Teknik Informatika"  
Case 1  
cbProdi.Clear  
cbProdi.AddItem "Manajemen Informatika"  
cbProdi.AddItem "Teknik Komputer"  
End Select  
If cbProdi.Text <> "" Then  
BuatNIM  
Else  
txtNIM.Text = ""  
End If  
End Sub  
Private Sub cbProdi_Click()  
If cbJenjang.Text <> "" Then  
BuatNIM  
txtNama.Enabled = True  
dtpLahir.Enabled = True  
txtAlamat.Enabled = True  
txtNama.SetFocus  
End If  
End Sub
```

f. Kode program untuk menampilkan membuat NIM secara otomatis

```
Sub BuatNIM()  
Dim Prodi, Jenjang, Tahun, NIM As String  
Tahun = Right(Year(Now), 2)  
Select Case cbJenjang.ListIndex  
    Case 0  
        Jenjang = "01"  
        Select Case cbProdi.ListIndex  
            Case 0  
                Prodi = "01"  
            Case 1  
                Prodi = "02"  
        End Select  
    Case 1  
        Jenjang = "02"  
        Select Case cbProdi.ListIndex  
            Case 0  
                Prodi = "03"  
            Case 1  
                Prodi = "04"  
        End Select  
End Select  
Set rsMahasiswa = New ADODB.Recordset  
rsMahasiswa.Open "Select NIM From tbMahasiswa Where Jenjang= '"  
& cbJenjang.Text & "' And " & _  
"Prodi= '" & cbProdi.Text & "' Order By NIM ", Con,  
adOpenStatic, adLockOptimistic  
With rsMahasiswa  
    If .EOF Then  
        NIM = Tahun & Jenjang & Prodi & "001"  
    Else  
        .MoveLast  
        A = rsMahasiswa!NIM  
        B = Right(A, 3)  
        C = Val(B)  
        If C > 0 And C < 10 Then  
            NIM = Tahun & Jenjang & Prodi & "00" & (C + 1)  
        ElseIf C >= 10 And C < 100 Then  
            NIM = Tahun & Jenjang & Prodi & "0" & (C + 1)  
        ElseIf C >= 100 And C < 1000 Then  
            NIM = Tahun & Jenjang & Prodi & C  
        End If  
    End If  
End With  
txtNIM.Text = NIM  
End Sub
```

g. Kode program untuk pencarian NIM

```
Private Sub txtNIM_KeyPress(KeyAscii As Integer)  
Set rsMahasiswa = New ADODB.Recordset  
rsMahasiswa.Open "Select * From tbMahasiswa Where NIM = '" &  
txtNIM.Text & "'", Con, adOpenStatic, adLockOptimistic
```

```

If KeyAscii = 13 Then 'Jika ditekan Enter
  With rsMahasiswa
    If cmdEdit.Caption = "Update" Then
      If .EOF Then
        MsgBox "NIM " & txtNIM.Text & " belum ada"
        txtNIM.SetFocus
      Else
        txtNama.Enabled = True
        txtAlamat.Enabled = True
        dtpLahir.Enabled = True
        txtNama.SetFocus
        cbJenjang.Text = !Jenjang
        cbProdi.Text = !Prodi
        txtNama.Text = !Nama
        txtAlamat.Text = !Alamat
        dtpLahir.Value = !TglLahir
      End If
    ElseIf cmdDel.Caption = "Delete" Then
      If .EOF Then
        MsgBox "NIM " & txtNIM.Text & " belum ada"
        txtNIM.SetFocus
      Else
        cbJenjang.Text = !Jenjang
        cbProdi.Text = !Prodi
        txtNama.Text = !Nama
        txtAlamat.Text = !Alamat
        dtpLahir.Value = !TglLahir
      End If
    End If
  End With
End If
'hanya boleh diisi angka atau backspace
If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9"))
Or KeyAscii = vbKeyBack) Then
  Beep
  KeyAscii = 0
End If
End Sub

```

h. Kode program untuk menambah data baru dan edit data

```

Private Sub cmdNew_Click()
If cmdNew.Caption = "New" Then
  cbJenjang.Enabled = True
  cbJenjang.SetFocus
  cmdNew.Caption = "Save"
  cmdEdit.Enabled = False
  cmdDel.Enabled = False
Else
  Set rsMahasiswa = New ADODB.Recordset
  rsMahasiswa.Open "Select * From tbMahasiswa", Con,
  adOpenStatic, adLockOptimistic

```

```

        With rsMahasiswa
            .AddNew 'Menambahkan data baru
            !NIM = txtNIM.Text
            !Prodi = cbProdi.Text
            !Jenjang = cbJenjang.Text
            !Nama = txtNama.Text
            !Alamat = txtAlamat.Text
            !TglLahir = Format(dtpLahir.Value, "dd/mm/yyyy")
            .Update
        End With
        cmdCancel_Click ' memanggil cmdCancel_Click
    End If
End Sub

Private Sub cmdEdit_Click()
    If cmdEdit.Caption = "Edit" Then
        txtNIM.Enabled = True
        txtNIM.SetFocus
        cmdEdit.Caption = "Update"
        cmdNew.Enabled = False
        cmdDel.Enabled = False
    Else
        Set rsMahasiswa = New ADODB.Recordset
        rsMahasiswa.Open "Select * From tbMahasiswa where NIM = '" &
        txtNIM.Text & "'", Con, adOpenStatic, adLockOptimistic
        With rsMahasiswa
            !Prodi = cbProdi.Text
            !Jenjang = cbJenjang.Text
            !Nama = txtNama.Text
            !Alamat = txtAlamat.Text
            !TglLahir = Format(dtpLahir.Value, "dd/mm/yyyy")
            .Update
        End With
        cmdCancel_Click ' memanggil cmdCancel_Click
    End If
End Sub

```

i. Kode program untuk menghapus data

```

Private Sub cmdDel_Click()
    If cmdDel.Caption = "Del" Then
        txtNIM.Enabled = True
        txtNIM.SetFocus
        cmdDel.Caption = "Delete"
        cmdNew.Enabled = False
        cmdEdit.Enabled = False
    Else
        pesan = MsgBox("Apakah Anda yakin menghapus NIM " & _
        txtNIM.Text & " ?", vbYesNo, "Hapus Data")
        If pesan = vbYes Then
            Set rsMahasiswa = New ADODB.Recordset
            rsMahasiswa.Open "Select * From tbMahasiswa where
            NIM = '" & txtNIM.Text & "'", Con, adOpenStatic,
            adLockOptimistic
            rsMahasiswa.Delete
        End If
        cmdCancel_Click ' memanggil cmdCancel_Click
    End If
End Sub

```

j. Kode program pada saat klik Cancel

```
Private Sub cmdCancel_Click()  
    txtNIM.Text = ""  
    txtNama.Text = ""  
    txtAlamat.Text = ""  
    cbProdi.Text = ""  
    cbJenjang.Text = ""  
    dtpLahir.Value = Now  
  
    txtNIM.Enabled = False  
    txtNama.Enabled = False  
    txtAlamat.Enabled = False  
    cbProdi.Enabled = False  
    cbJenjang.Enabled = False  
    dtpLahir.Enabled = False  
  
    cmdNew.Caption = "New"  
    cmdEdit.Caption = "Edit"  
    cmdDel.Caption = "Del"  
  
    cmdNew.Enabled = True  
    cmdEdit.Enabled = True  
    cmdDel.Enabled = True  
  
    TampilData  
End Sub
```

## Soal Latihan

Buka kembali program yang sudah Anda buat seperti gambar 55.

- Tambahkan Tabel Dosen dan Tabel Mata Kuliah pada Database Akademik yang sudah Anda buat. Atur Field yang dibutuhkan untuk masing-masing tabel.
- Tambahkan Form untuk Olah Data Dosen dan Form untuk Olah Data Mata Kuliah dan ketik kode programnya.

## BAB X

### DATA REPORT dan CRYSTAL REPORT

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal Data Report dan Crystal Report untuk pembuatan Laporan	<ul style="list-style-type: none"><li>• Dapat menggunakan Data Report</li><li>• Dapat menggunakan Crystal Report</li></ul>

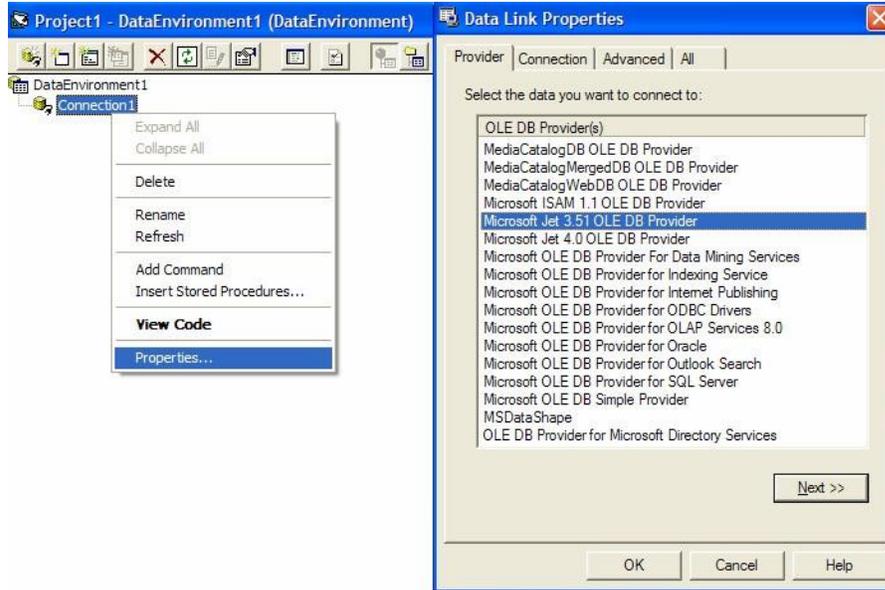
#### A. Data Report

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

##### 1. Menggunakan Data Environment

Buka kembali program yang Anda buat seperti pada gambar 9.7 (Bab 9). Tambahkan Data Environment dengan cara sebagai berikut :

- Pastikan Data Environment dan Data Report sudah diaktifkan dengan cara : pilih Project – Components, pilih tab Designers, Aktifkan (centang) Dta Environent dan Data Report- Klik OK
- Pilih Project – Add Data Environment. Klik kanan Connection1-Pilih Properties. Pada bagian tab Provider pilih Microsoft Jet 3.51 OLE DB Provider (untuk Access 1997), Microsoft Jet 4.0 OLE DB Provider(untuk Access 2000-2003)



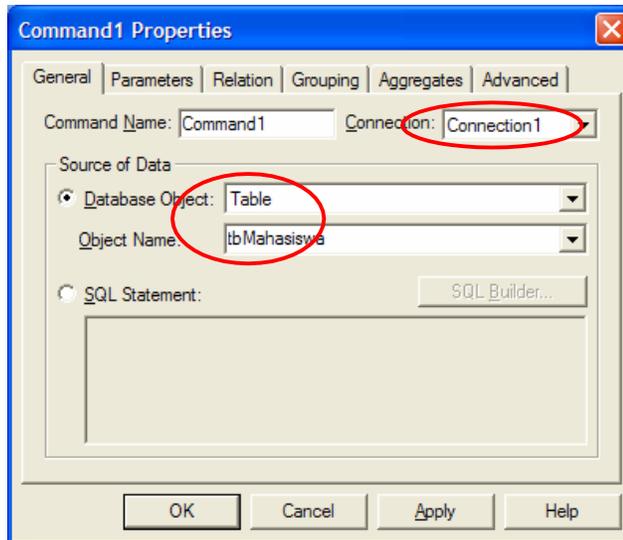
Gambar 56. Menggunakan Data Environment

- Klik Next>>, pada bagian *Select or enter a database name*, browse nama database (dbAkademik.mdb)
- Klik tombol Test Connection, jika sukses maka muncul messagebox seperti berikut :



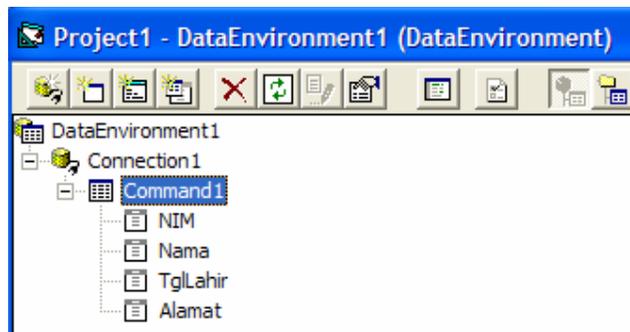
Gambar 57. MessageBox Koneksi database sukses

- Tambahkan Command untuk koneksi ke tabel, Klik kanan pada Connection1, pilih Add Command.
- Klik kanan pada Command1 pilih properties. Pada Connection pilih Connection1. Pada Database Object pilih Table. Pada Object Name pilih tbMahasiswa. Klik OK



Gambar 58. Properti Command1 untuk koneksi ke tabel

- Klik OK. Klik tanda  pada Command1, maka field-field dari tabel tbMahasiswa akan ditampilkan



Gambar 59. Field-field pada Command1 (tbMahasiswa)

## 2. Menggunakan Data Report

Setelah Data Environment kita atur, kemudian buat laporannya menggunakan Data Report. Ikuti langkah berikut :

- Pilih menu Project – Add Data Report

Tabel 24. Bagian Data Report

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

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

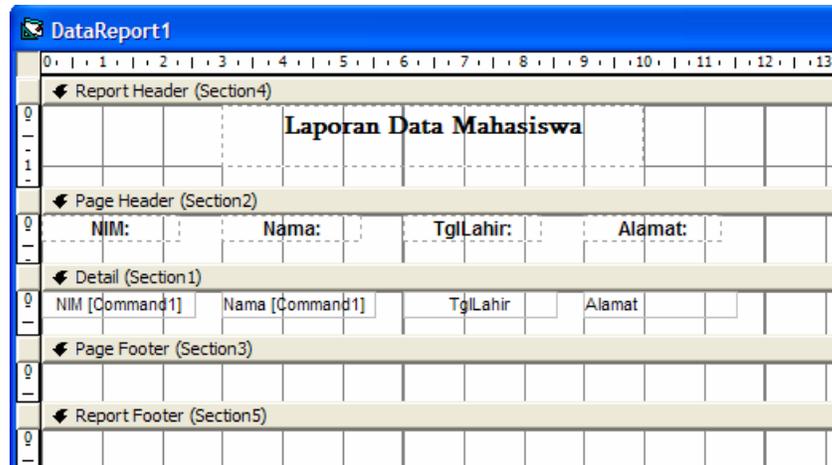


Gambar 60. Toolbox pada tab Data Report

Tabel 25. Toolbox pada tab Data Report

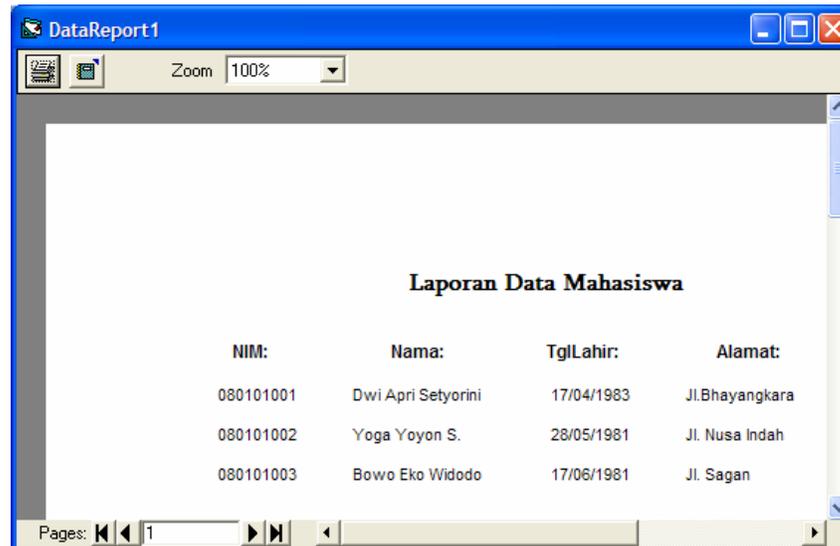
Nama Kontrol	Keterangan
RptLabel	Kontrol untuk memberikan label pada report
RptTextBox	Kontrol serupa textbox yang hanya menampilkan teks database saat runtime
RptImage	Kontrol untuk menempatkan image padareport
RptLine	Kontrol yang dapat menggambar garis secara horizontal, vertical maupun diagonal
RptShape	Kontrol untuk menggambar shape padareport
RptFunction	Kontrol yang dapat menset hasil perhitungan data

- Letakkan kontrol RptLabel pada bagian Report Header. Ubah Property Caption menjadi "Laporan Data ahasiswa", dan property Aligment menjadi 2-rptJustifyCenter
- Klik DataReport1, ubah property DataSource, pilih DataEnvironment1
- Isi DataMember denga nama Command1
- Klik menu bar Window, Pilih Cascade. Letakkan Jendela DataEnvironment1 di atas jendela Data Report.
- Drag and Drop field-filed yang ada di Jendela DataEnvironent1 keJendela Data Report di bagian Detail.
- Pindahkan bagian yang diakhiri tanda ":" ke bagian Page Header



Gambar 61. Meletakkan Field-Field ke Data Report

- Desain akhir Data Report adalah sebagai berikut :



Gambar 62. Desain Akhir Data Report

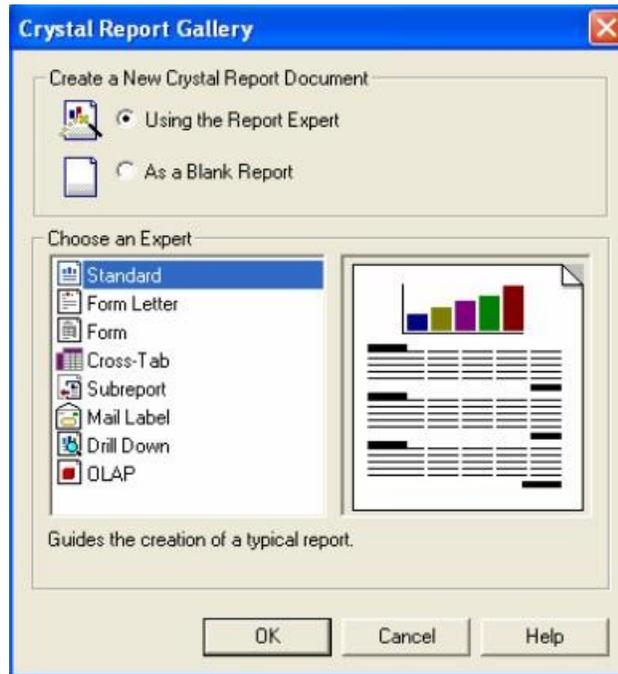
## B. Crystal Report

Selain Data Report, kita juga bisa menggunakan Crystal Report dalam pembuatan laporan. Crystal Report adalah suatu form khusus berbentuk seperti lembaran format naskah yang ingin dicetak.

Crystal Report dapat berdiri sendiri dapat pula menjadi satu dengan project Visual Basic yang anda buat. Bila berdiri sendiri, report tersebut pun dapat dipanggil dari project Visual Basic dengan Crystal Report Control sehingga report yang telah anda buat dapat digunakan oleh beberapa project sekaligus.

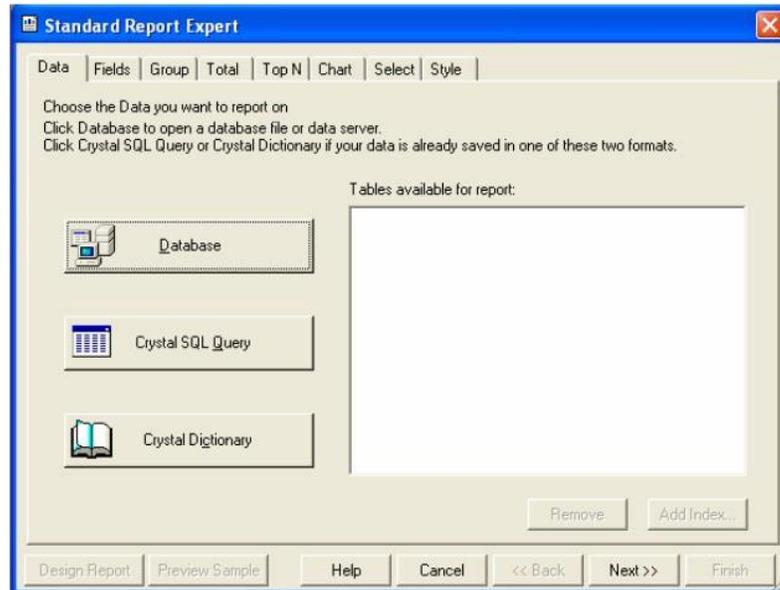
1. Membuat Laporan dengan Crystal Report

- Pertama-tama Anda harus Install dulu Software Crystal Report. Pada kesempatan kali ini menggunakan Crystal Report 8.5.
- Buka program Crystal Report , maka pada tampilan windows akan muncul tampilan seperti gambar di bawah ini



*Gambar 63. Crystal Report Gallery*

- Pilihlah Report Expert dan Standard Expert untuk laporan standard dan pilih OK untuk melanjutkan. Setelah itu akan muncul semua property dari report yang akan kita buat, yang akan ditampilkan seperti gambar :



*Gambar 64. Standard Report Expert*

- Pertama kali kita harus memasukkan datasource dari report yang akan kita buat (darimana data yang mau kita tampilkan) dengan memilih 1 dari ketiga jenis data yang telah tersedia (Database, Crystal SQL Query, dan Crystal Dictionary). Untuk menampilkan data dari database seperti Microsoft Access, kita akan memilih Database.
  - Pilih DatabaseFile dan pilih find database file dan click add, cari database yang akan ditampilkan.
  - Pilihlah datasource yang diinginkan dan click Add, setelah itu clickClose, maka dengan ini, report yang akan kita buat telah ditentukan datasourcenya. Setelah itu pilihlah tombol Next untuk menuju bagian field yang akan kita tampilkan di dalam report.
  - Pilihlah field-field yang akan kita tampilkan di report, dan bila sudah selesai, click finish untuk menuju design report kita.
  - Buat desainya, simpan Report, misal rptMahasiswa
2. Koneksi Crystal Report dengan Visual Basic
- Pastikan Crystal Report sudah terinstall dan Anda sudah selesai membuat desain reportnya dan sudah Anda simpan dalam satu folder dengan program Visual Basic yang Anda buat.

- Buka kembali program yang telah Anda buat di Bab 9 (Gambar 9.8). Tambahkan tombol untuk cetak data.
- Tambahkan object Crystal Report di toolbox dengan cara pilih Project - Components - Pilih tab Controls - pilih Crystal Report Viewer Control tekan OK. Kemudian tambahkan objek tersebut kedalam Form
- Berikut kode program untuk mencetak data di VB

```

'Mencetak semua data mahasiswa
Sub CetakAll ()
rptMhs.Reset
rptMhs.Connnect = Con
rptMhs.ReportFileName = App.Path & "\RptMahasiswa.rpt"
rptMhs.WindowState = crptMaximized
rptMhs.RetrieveDataFiles
rptMhs.Action = 1
End Sub

'Mencetak berdasarkan Program Studi
Sub CetakProdi ()
rptMhs.Reset
rptMhs.Connnect = Con
rptMhs.ReportFileName = App.Path & "\RptDep.rpt"
rptMhs.SelectionFormula = ("{tbMahasiswa.Prodi}='" &
cbProdi.Text & "'")
rptMhs.WindowState = crptMaximized
rptMhs.RetrieveDataFiles
rptMhs.Action = 1
End Sub

```

Beberapa property dari Crystal Report yang biasa digunakan di dalam aplikasi, diantaranya adalah :

.Reset - Digunakan untuk mengembalikan kondisi report ke dalam kondisi semula, biasanya properti ini digunakan pertama kali sebelum properti lainnya .

.Connect - Digunakan untuk menentukan jenis conection yang digunakan Oleh report tersebut, biasanya diisi oleh variabel ADODB.Connection yang kita buat.

.ReportFileName - Digunakan untuk menentukan report yang akan kita tampilkan Beserta path dari report tersebut.

.StoreProcedureParam - Digunakan bila datasource dari report yang akan kita panggil adalah storeProcedur yang memiliki parameter

.Formula - Digunakan apabila di dalam report yang kita buat, terdapat Formula yang ingin kita isi dari VB.

.SelectionFormula - Digunakan untuk memfilter data dari report yang ingin kita tampilkan dengan menyebutkan namaDataSource>Nama Field

.WindowState -Digunakan untuk menentukan windowstate dari reportpada saat pertama kali tampil apakah fullscreen, minimize dsb.

.Action -Digunakan untuk memunculkan report yang akan kita tampilkan dengan memberinya angka 1 (.Action=1)

## **Soal Latihan**

Buka kembali program yang sudah Anda buat pada soal Latihan Bab 9. Buatlah Laporan untuk Data Dosen dan Data Mata Kuliah dengan Crystal Report.

# BAB XI

## MDI FORM dan MENU

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal MDI Form dan Menu Editor	<ul style="list-style-type: none"><li>• Dapat menggunakan MDI Form sebagai form utama dalam program</li><li>• Dapat menggunakan fasilitas MenuEditor untuk membuat menu</li></ul>

### A. MDI Form

Pada Visual Basic, anda dapat mengembangkan aplikasi dengan interface sebagai berikut :

- SDI (Single Document Interface)
- MDI (Multiple Document Interface)

Pada aplikasi SDI, setiap form merupakan form-form yang berdiri sendiri, Aplikasi SDI pada windows terdapat pada aplikasi seperti Notepad, WordPad dan Paint.

Sedangkan aplikasi seperti Visual Basic menggunakan MDI, yaitu terdiri dari suatu MDIForm, dan didalamnya merupakan form-form anak (MDIChild).

Ada beberapa hal yang harus diperhatikan dalam penggunaan MDIForm adalah :

- ✓ Didalam satu project hanya dapat terdiri dari satu MDIForm
- ✓ Anda tidak dapat menempatkan kontrol-kontrol secara langsung pada MDIForm, kecuali kontrol yang memiliki properti Alignment, atau menempatkannya diatas kontainer seperti PictureBox.
- ✓ Anda tidak dapat menggunakan metode penggambaran (Print, Line, Circle, dan PSet) seperti pada form umumnya.

### 1. Membuat MDI Form di Visual Basic

- Kita harus tambahkan dengan klik pada menu Project - Add MDI Form. Klik Open.
- Pada Project Explorer akan muncul dua buah form yaitu Form1 dan MDIForm1. Ubahlah properties MDIChild dari Form1 menjadi true. Hal ini menandakan Form1 adalah sebagai form anak dari MDIForm1. Settinglah pada Project Properties agar Form utamanya adalah MDIForm1.

### 2. Karakteristik dari MDI Form

- Semua Cchild Form tidak dapat dipindahkan keluar dari MDI Form.
- Ketika suatu Child Form diminimize, akan menjadi icon dibawah MDI Form.
- Anda dapat menentukan apakah Child Form secara otomatis ditampilkan atau tidak dengan menggunakan properti AutoShowChildren pada MDIForm.
- Jika pada Child Form ada menu, maka menu akan ditampilkan pada MDIform menu.

Pada MDIform anda tidak dapat menempatkan kontrol-kontrol visible yang tidak mendukung alignment, kecuali kalau anda menempatkannya kedalam suatu kontainer seperti PictureBox, dan Toolbar. Sedangkan kontrol non-visible seperti Timer dan CommonDialog box dapat ditempatkan diatas MDIForm.

### 3. Mendapatkan MDI Child yang sedang aktif

Untuk mendapatkan form-form yang sedang aktif di dalam suatu MDIForm, anda dapat menggunakan properti ActiveForm, contoh :

```
Private Sub MDIForm_QueryUnload(Cancel As Integer,  
UnloadMode As Integer)  
If Not Me.ActiveForm Is Nothing Then  
    MsgBox "Masih ada Form yang aktif"  
    Cancel = True  
End If  
End Sub
```

### 4. Mengatur MDI Child dalam jendela MDI form

Anda dapat menggunakan metoda Arrange untuk mengatur penyusunan form-form yang sedang aktif didalam suatu MDI Form.

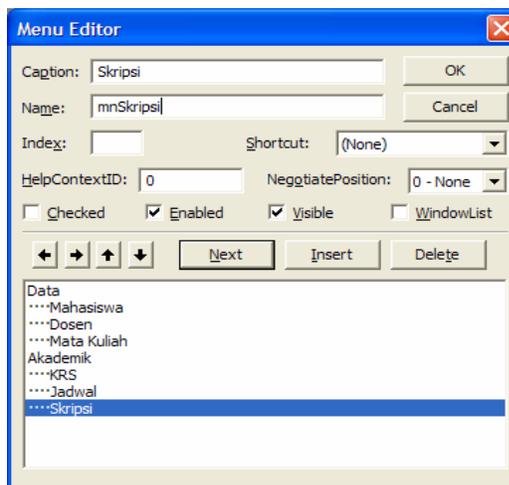
Metoda Arrange ini diikuti oleh suatu parameter yang menentukan jenis penyusunan yang akan dilakukan, contoh :

```
Private Sub mnuTileHorizontally_Click()  
    Arrange vbTileHorizontal  
End Sub  
  
Private Sub mnuTileVertically_Click()  
    Arrange vbTileVertical  
End Sub  
  
Private Sub mnuCascade_Click()  
    Arrange vbCascade  
End Sub  
  
Private Sub mnuArrangeIcons_Click()  
    Arrange vbArrangeIcons  
End Sub
```

## B. Menu Editor

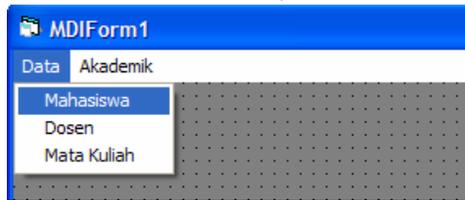
Pembuatan menu pada Visual Basic dapat dilakukan dengan bantuan Menu Editor yang terdapat pada Menu Tools-Menu Editor. Pada dasarnya setiap item menu memiliki Caption dan sebuah Nama. Anda dapat membentuk Kunci Akses dengan menggunakan tanda & (ampersand) pada Caption dari menu tersebut. Untuk membuat menu anda cukup mengetikkan Caption dan Name, selanjutnya klik pada Next, dan ketikkan menu yang berikutnya, sampai selesai. Selanjutnya adalah membuat Sub Menu dengan melakukan klik pada panah kanan dan sebaliknya.

### 1. Membuat Menu di MDI Form



Gambar 65. Window Menu Editor

Setelah menu diatur pada Window Menu Editor, maka berikut hasilnya :



Gambar 66. Tampilan menu yang dibuat di MDIForm

Untuk memunculkan frMahasiswa pada saat klik menu Mahasiswa maka ketikkan kode berikut pada jendela MDIForm kode editor

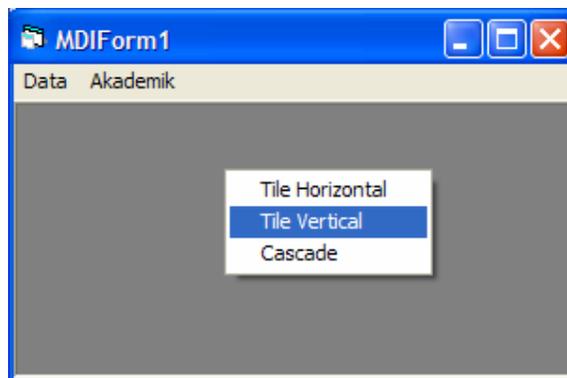
```
Private Sub mnMhs_Click()  
    frMahasiswa.Show  
End Sub
```

## 2. Membuat Pop Up Menu

Pada Visual Basic, anda dapat membuat menu pop up dengan memanfaatkan menu Editor untuk mendefinisikan nama kelompok menu Pop Up beserta Sub Menunya, dan menonaktifkan option Visible dari kelompok menu tersebut.

```
Private Sub MDIForm_MouseDown(Button As Integer, Shift  
As Integer, X As Single, Y As Single)  
    If Button And vbRightButton Then  
        PopupMenu mnAtur  
    End If  
End Sub
```

Sehingga kalau dilakukan klik kanan pada form akan menampilkan suatu PopUp menu yang berupa Sub Menu dari mnAtur.



Gambar 67. Tampilan Pop Up Menu di MDIForm

## **Soal Latihan**

Buka kembali program yang sudah Anda buat pada Bab 9. Koneksikan dengan Crystal Report (rptMahasiswa). Tambahkan MDIForm dan tambahkan menu Laporan untuk menampilkan laporan Mahasiswa, Dosen, Mata Kuliah dan laporan lain yang dibutuhkan.

## BAB XII

### MENGCOMPILE DAN MEMBUAT FILE SETUP

*Standar Kompetensi :*

Mahasiswa dapat mengetahui, memahami, menguasai dan mampu mengimplementasi teori, konsep dan prinsip pemrograman visual dengan logika pemrograman yang benar, ringkas, dan tepat dalam penerapannya di bidang teknologi informasi

Kompetensi Dasar	Indikator
Mengenal Fasilitas File Exe dan Fasilitas Package and Deployment Wizard	<ul style="list-style-type: none"><li>• Dapat fasilitas File Exe untuk mengcompile file dalam bentuk exe</li><li>• Dapat menggunakan fasilitas Package and Deployment Wizard untuk membuat file Setup</li></ul>

#### A. Mengcompile atau Membuat File Executable (EXE)

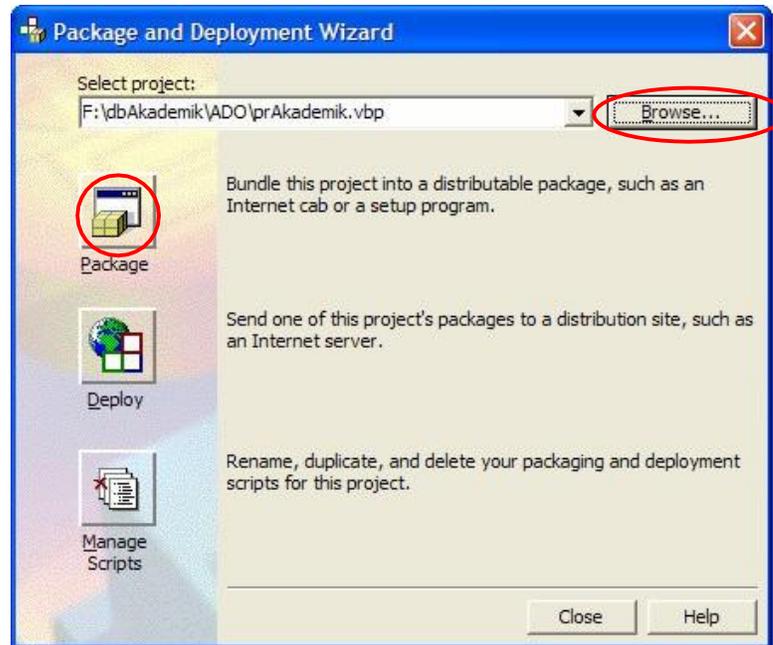
Setelah selesai membuat project anda dapat menyimpannya dan membuat file exe nya. Jika program aplikasi yang sudah dibuat belum dcompile dalam bentuk EXE maka akan dibutuhkan VB setiap kali menjalankannya. Agar user yang lain bisa juga menjalankan program yang telah kita buat, maka program yang telah selesai kita buat harus di compile dulu dalam bentuk exe.

- Untuk membuat File EXE buka kembali projek (\*.VBP) yang telah selesai anda buat (misalnya projek mahasiswa yang telah kita buat)
- kemudian klik menu File- Make Project.exe lalu simpan file .exe tersebut dan lanjutkan mengklik OK
- Tunggu sampai proses kompilasi selesai dan coba keluar dari Visual Basic. Lalu jalankan file.exe yang telah anda buat.

#### B. Setup Exe

Selain file exe yang tak kalah penting lagi adalah file SETUP.EXE. Membuat file setup.exe tidaklah sesulit yang dibayangkan hanya dengan beberapa klik Anda sudah dapat membuat file setup.exe.

- Untuk membuat Setup.exe klik Start - Program - Microsoft Visual Studio 6 - Microsoft Visual Studio 6.0 Tools - Package & Deployment Wizard  
| sehingga akan tampil window seperti gambar dibawah ini.



Gambar 68. Package and Deployment Wizard

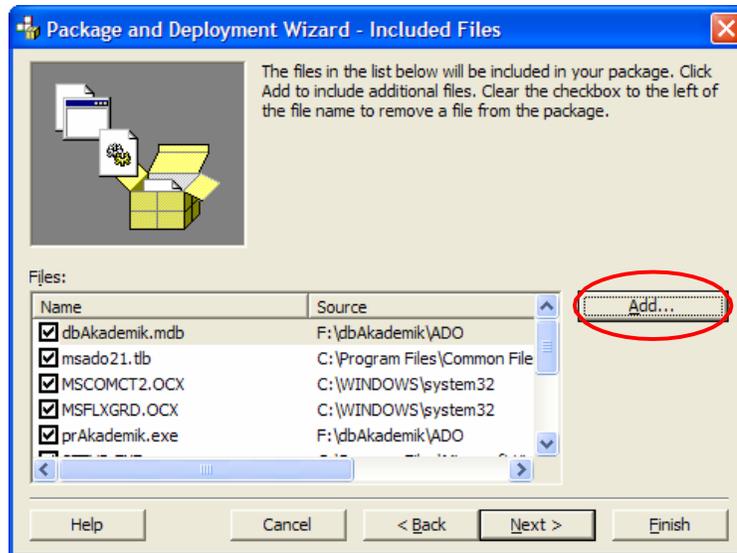
- Kemudian klik browse untuk mencari tempat folder proyek yang telah kita buat dan simpan. Cari file .vbp yang telah anda buat, pastikan anda juga telah membuat file .exe dalam satu folder yang sama. Kemudian lanjutkan dengan mengklik Package
- Setelah diklik Package akan muncul window seperti gambar 12.2



Gambar 69. Package and Deployment Wizard-Package Type

- Pilih Package type = Standard Setup Package. Lanjutkan dengan mengklik Next. Setelah itu akan muncul window yang meminta anda untuk menyimpan hasil kompilasi Setup.exe. Pilih tempat folder yang anda inginkan. Kemudian klik Next. Maka muncul Window yang

menampilkan file-file yang akan dipackage. Jika Anda menggunakan database (misal : dbAkademik.mdb) dan file database tersebut belum ada di list, maka tekan tombol Add untuk menambahkan file database tersebut.



Gambar 70. Package and Deployment Wizard-Included Files

- Setelah anda klik Next akan muncul Window Cab Options, pilih pada defaultnya yaitu Single Cab, kemudian lanjutkan dengan klik Next.
- Setelah anda klik Next akan muncul Window Installation Title, masukkan sesuai dengan nama project yang kita buat. Kemudian lanjutkan dengan mengklik Next.
- Setelah anda klik Next akan muncul Window Start Menu Items, pilih pada defaultnya, lanjutkan dengan mengklik Next
- Setelah anda klik Next akan muncul Window Install Location yang memberitahu ke kita tempat / lokasi instalasi, biarkan pada defaultnya, lanjutkan dengan mengklik Next
- Setelah anda klik Next akan muncul Window Shared Files, lanjutkan dengan mengklik Next. Sehingga akan muncul window Finished, biarkan pada defaultnya, kemudian klik Finish.
- Tunggu beberapa saat sampai hasil kompilasinya selesai. Install hasil setup file yang sudah Anda buat. Jalankan dari start menu windows.

## **Soal Latihan**

Buatlah setup file dari program yang Anda buat. Install setup file tersebut, dan jalankan program yang sudah Anda install dari Start Menu

## Referensi

<http://54ll1.wordpress.com/2008/08/02/program-sederhana-pada-visual-basic>  
<http://absanka.files.wordpress.com/2008/07/pemrograman-visual.pdf>  
<http://absanka.wordpress.com/2008/07/05/pdf-program-visual-basic>  
<http://books.google.co.id/books?id=zXFAlSg4rM8C&printsec=frontcover#PPA19,M1>  
<http://bsi.purworejo.asia/2008/07/penggunaan-combo-dan-option.html>  
[http://download.evopage.com/vb6/Visual Basic 6.0 skripsi.pdf](http://download.evopage.com/vb6/Visual%20Basic%206.0%20skripsi.pdf) <http://dwi.its-sby.edu/PIKTI/Visual%20Basic/last/Bab1.doc>  
<http://dwi.its-sby.edu/PIKTI/Visual%20Basic/last/Bab3.doc> <http://dwi.its-sby.edu/PIKTI/Visual%20Basic/last/Bab4.doc>  
<http://dwi.its-sby.edu/PIKTI/Visual%20Basic/last/Bab5.doc> <http://dwi.its-sby.edu/PIKTI/Visual%20Basic/last/Bab11.doc>  
[http://elearning.upnjatim.ac.id/courses/FKK1002/work/48ca972c47ddcvaria\\_bel\\_dan\\_konstanta.htm](http://elearning.upnjatim.ac.id/courses/FKK1002/work/48ca972c47ddcvaria_bel_dan_konstanta.htm)  
[http://elista.akprind.ac.id/staff/catur/Pemrograman Basisdata/Visual Basic/modulvbdao.pdf](http://elista.akprind.ac.id/staff/catur/Pemrograman%20Basisdata/Visual%20Basic/modulvbdao.pdf)  
<http://forum.vb-bego.com/viewtopic.php?f=8&t=559>  
<http://ikc.cbn.net.id/berseri/krisna-vb6/krisna-vb6-06.zip>  
<http://lecturer.eepis-its.edu/~tessy/tutorial/bab5.pdf>  
<http://leo.apeaje.info/vb/tutor.html>  
[http://ma2dxl.blogspot.com/2009\\_01\\_01\\_archive.html](http://ma2dxl.blogspot.com/2009_01_01_archive.html)  
<http://mercusian.com/visual-basic/praktek-visual-basic-database-ado-data-control.html>  
<http://mercusian.com/visual-basic/praktek-visual-basic-database-dao.html>  
<http://mercusuar.ac.id/download/modul-kuliah/modul-visual-basic/modul-database-dengan-dao.html>  
<http://mugi.or.id/blogs/elang/archive/2008/08/15/penggunaan-dao-data-access-objects.aspx> [http://ns1.cic.ac.id/~marsani.asfi/tulisan/Modul\\_1.htm](http://ns1.cic.ac.id/~marsani.asfi/tulisan/Modul_1.htm)

[http://ns1.cic.ac.id/~marsani.asfi/tulisan/Modul\\_1.htm](http://ns1.cic.ac.id/~marsani.asfi/tulisan/Modul_1.htm)  
<http://ns1.cic.ac.id/~marsani.asfi/tulisan/Temu%231.htm> <http://one.indoskripsi.com/judul-skripsi-makalah-tentang/kumpulan-koleksi-fungsi-visual-basic-60> [http://organisasi.org/ilmu\\_pengetahuan/matematika](http://organisasi.org/ilmu_pengetahuan/matematika)  
[http://students.itb.ac.id/~hadi102/E-books/Computer\\_6\\_Programming/Visual Basic \(VB\)/test.pdf](http://students.itb.ac.id/~hadi102/E-books/Computer_6_Programming/Visual_Basic_(VB)/test.pdf)  
[http://tutorial.smkn6dki.or.id/index.php?action=downloadfile&filename=05\\_Visual\\_Basic.pdf&directory=public\\_downloads/Modul\\_alih\\_Profesi\\_UNJ&PHPSESSID=433dcd7d02ffa25ac38d7c471105b17](http://tutorial.smkn6dki.or.id/index.php?action=downloadfile&filename=05_Visual_Basic.pdf&directory=public_downloads/Modul_alih_Profesi_UNJ&PHPSESSID=433dcd7d02ffa25ac38d7c471105b17)  
[http://unilinet.unila.ac.id/~gigih/belajar\\_komputer/VB/Pelajaran vb- 6.0/krisna-vb6-08.pdf](http://unilinet.unila.ac.id/~gigih/belajar_komputer/VB/Pelajaran_vb-6.0/krisna-vb6-08.pdf)  
[http://www.ajibsusanto.site88.net/aplikasi\\_bisnis/MODUL\\_VB\\_6.pdf](http://www.ajibsusanto.site88.net/aplikasi_bisnis/MODUL_VB_6.pdf)  
[http://www.box.net/index.php?rm=box\\_v2\\_download\\_shared\\_file&file\\_id=f\\_70947083](http://www.box.net/index.php?rm=box_v2_download_shared_file&file_id=f_70947083)  
<http://www.convertworld.com/id/sudut/Radian.html>  
[http://www.geocities.com/gen\\_xeost/BAB\\_6.pdf](http://www.geocities.com/gen_xeost/BAB_6.pdf)  
[http://www.geocities.com/visiweb/vb/dt\\_env/index.html](http://www.geocities.com/visiweb/vb/dt_env/index.html)  
<http://www.kuliahit.com/kuliahit/article/13/Perintah-MsgBox-dan-InputBox-pada-VB>  
<http://www.kuliahit.com/kuliahit/article/16/Fungsi-Date-and-Time-pada-VB>  
<http://www.masinosinaga.com/index.php>  
<http://www.masinosinaga.com/index.php?name=Forums&file=viewtopic&t=130>  
<http://www.masinosinaga.com/index.php?name=News&file=article&sid=234>  
<http://www.masinosinaga.com/index.php?name=News&file=article&sid=40>  
<http://www.mikron123.com/index.php/Tutorial-VB/Tutorial-1-Mengenal-VB.htm>  
[http://www.smkwikrama.net/learning2/ebook/Pengenalan Visual Basic kom8.doc](http://www.smkwikrama.net/learning2/ebook/Pengenalan_Visual_Basic_kom8.doc)  
[http://www.stmik-budidarma.ac.id/Download/D3 - Pemrograman Visual Basic II.pdf](http://www.stmik-budidarma.ac.id/Download/D3-Pemrograman_Visual_Basic_II.pdf)

Tim Penyusun. 2002. Modul Praktikum Visual Basic. Yogyakarta. FMIPA Universitas Gadjah  
Mada.

Tim Penyusun. 2006. Modul Praktek Laboratorium Komputer Visual Basic 6.0.  
Jakarta. AMIK Bina Sarana Informatika.