



ALGORITMA PEMROGRAMAN 2

Menggunakan

VISUAL BASIC 6.0

Achmad Basuki

Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember Surabaya
Surabaya 2006

DAFTAR ISI

Kata Pengantar

Daftar Isi

1. Mengetahui Visual Basic	1
1.1. Mengetahui Visual Basic 6.0	1
1.2. Interface Antar Muka Visual Basic 6.0	2
1.3. Konsep Dasar Pemrograman Dalam Visual Basic 6.0	3
1.4. Membuat Project Baru	4
2. Event dan Property	9
2.1. Membuat User Interface	9
2.2. Mengatur Property	10
2.3. Event Dalam Kode Program	12
2.4. Membuat Project Mengatur Property	13
2.5. Latihan	17
3. Variabel, Operator dan Ekspresi	20
3.1. Variabel	20
3.2. Operator	21
3.3. Ekspresi	22
3.4. Project Sederhana Dengan Ekspresi	25
3.5. Latihan	30
4. Kondisi dan Keputusan	33
4.1. Percabangan Dengan If..Then..Else..	33
4.2. Percabangan Dengan If..Then..Else.. Berantai	37
4.3. Pemilihan Dengan Select Case	41
4.4. Latihan	44
5. Pengulangan	47
5.1. Pengulangan Dengan For..Next	47
5.2. Pengulangan Dengan While...Wend	51
5.3. Pengulangan Dengan Do..Loop Until	52

5.4. Pengulangan Bertingkat	53
5.5. Latihan	58
6. Array	60
6.1. Konsep Array	60
6.2. Array 2 Dimensi	66
6.3. Control Array	69
6.4. Latihan	72
7. Subroutine dan Fungsi	76
7.1. Subroutine	76
7.2. Fungsi	79
7.3. Latihan	82
8. Bermain Dengan Waktu	84
8.1. Tipe Data Waktu	84
8.2. Komponen Timer	86
8.3. Latihan	90
9. Operasi File	91
9.1. Operasi File Dalam Visual Basic	91
9.2. Membuka File	92
9.3. Menyimpan Data Ke File	93
9.4. Membaca Data Dari File	95
9.5. Komponen-komponen Untuk Operasi File	96
9.6. Latihan	99

1.1.Mengenal Visual Basic 6.0

Bahasa Basic pada dasarnya adalah bahasa yang mudah dimengerti sehingga pemrograman di dalam bahasa Basic dapat dengan mudah dilakukan meskipun oleh orang yang baru belajar membuat program. Hal ini lebih mudah lagi setelah hadirnya Microsoft Visual Basic, yang dibangun dari ide untuk membuat bahasa yang sederhana dan mudah dalam pembuatan scriptnya (*simple scripting language*) untuk graphic user interface yang dikembangkan dalam sistem operasi Microsoft Windows.

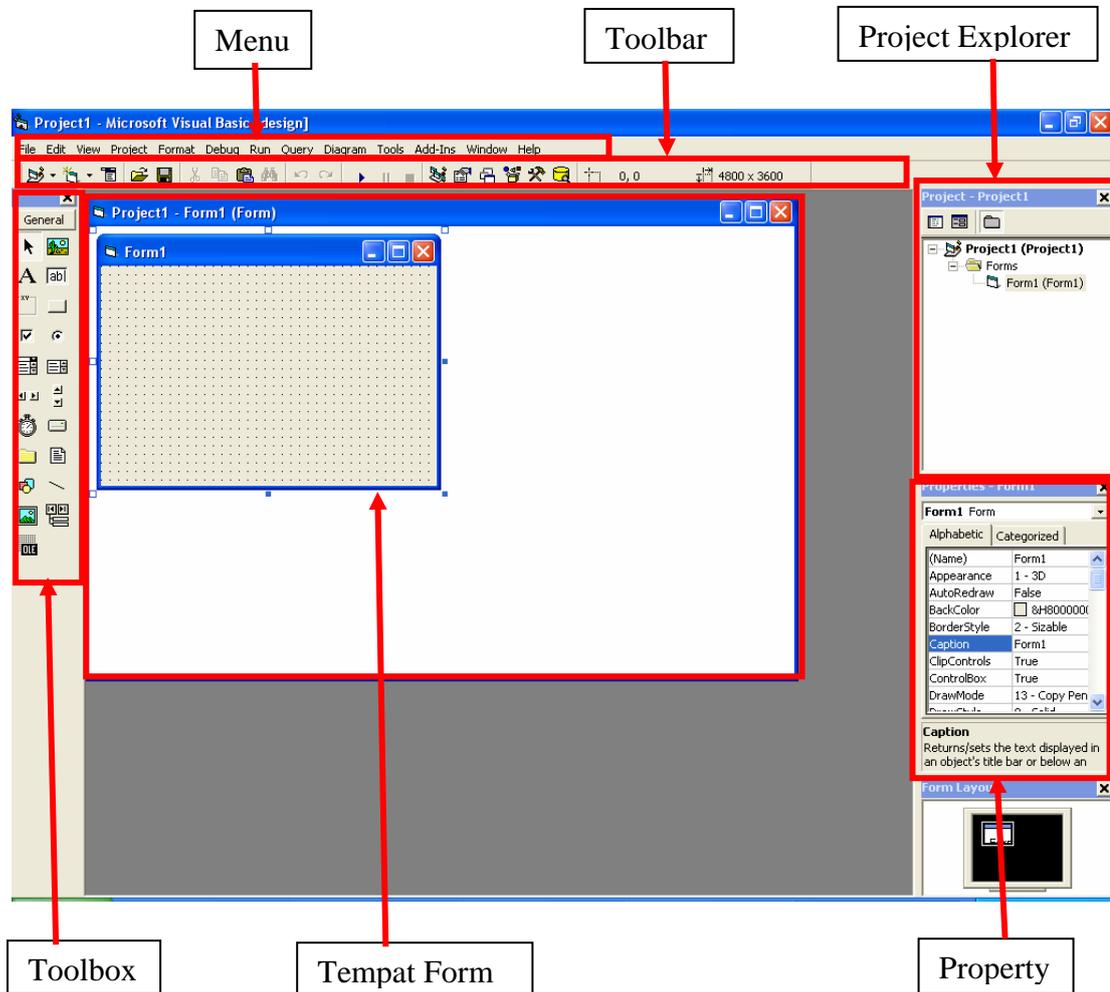
Visual Basic merupakan bahasa pemrograman yang sangat mudah dipelajari, dengan teknik pemrograman visual yang memungkinkan pengguna untuk berkreasi lebih baik dalam menghasilkan suatu program aplikasi. Ini terlihat dari dasar pembuatan dalam visual basic adalah FORM, dimana pengguna dapat mengatur tampilan form kemudian dijalankan dalam script yang sangat mudah.

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

Visual Basic 6.0 sebetulnya perkembangan dari versi sebelumnya dengan beberapa penambahan komponen yang sedang tren saat ini, seperti kemampuan pemrograman internet dengan DHTML (*Dynamic HyperText Mark Language*), dan beberapa penambahan fitur database dan multimedia yang semakin baik. Sampai saat buku ini ditulis bisa dikatakan bahwa Visual Basic 6.0 masih merupakan pilih pertama di dalam membuat program aplikasi yang ada di pasar perangkat lunak nasional. Hal ini disebabkan oleh kemudahan dalam melakukan proses *development* dari aplikasi yang dibuat.

1.2. Interface Antar Muka Visual Basic 6.0

Interface antar muka Visual Basic 6.0, berisi menu, toolbar, toolbox, form, project explorer dan property seperti terlihat pada gambar 1.1. berikut:

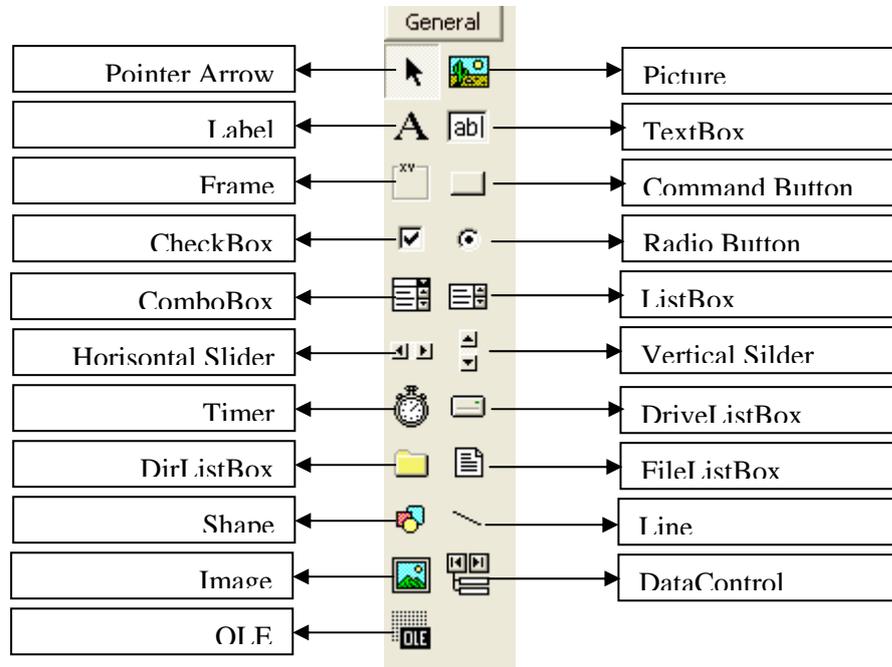


Gambar 1.1. Interface antar muka Visual Basic 6.0

Pembuatan program aplikasi menggunakan Visual Basic dilakukan dengan membuat tampilan aplikasi pada form, kemudian diberi script program di dalam komponen-komponen yang diperlukan. Form disusun oleh komponen-komponen yang berada di [Toolbox], dan setiap komponen yang dipakai harus diatur propertinya lewat jendela [Property].

Menu pada dasarnya adalah operasional standar di dalam sistem operasi windows, seperti membuat form baru, membuat project baru, membuka project dan menyimpan project. Di samping itu terdapat fasilitas-fasilitas pemakaian visual basic pada menu. Untuk lebih jelasnya Visual Basic menyediakan bantuan yang sangat lengkap dan detail dalam MSDN.

Toolbox berisi komponen-komponen yang bisa digunakan oleh suatu project aktif, artinya isi komponen dalam toolbox sangat tergantung pada jenis project yang dibangun. Komponen standar dalam toolbox dapat dilihat pada gambar 1.2 berikut ini.



Gambar 1.2. Komponen standar dalam Toolbox

1.3. Konsep Dasar Pemrograman Dalam Visual Basic 6.0

Konsep dasar pemrograman Visual Basic 6.0, adalah pembuatan form dengan mengikuti aturan pemrograman **Property**, **Metode** dan **Event**. Hal ini berarti:

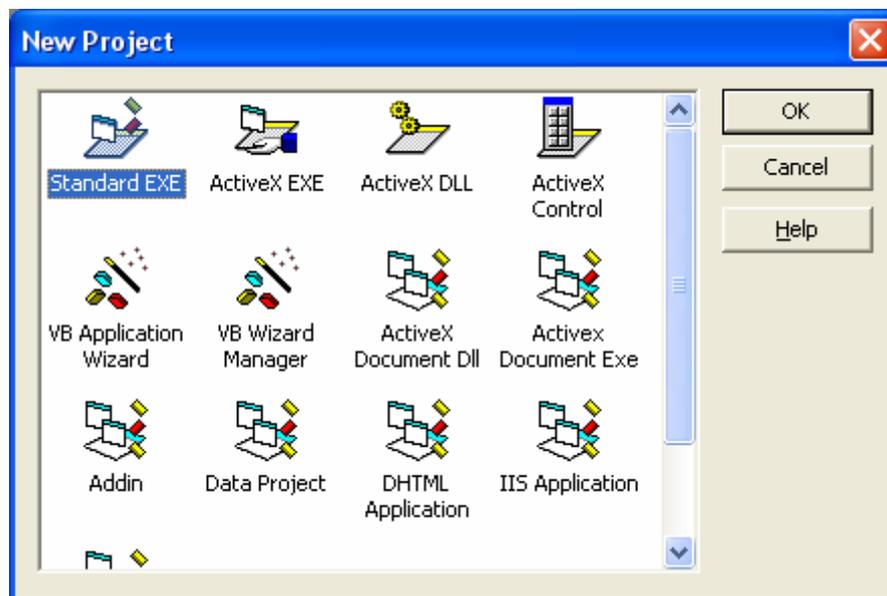
- (1) **Property**: Setiap komponen di dalam pemrograman Visual Basic dapat diatur propertinya sesuai dengan kebutuhan aplikasi. Property yang tidak boleh dilupakan pada setiap komponen adalah “**Name**”, yang berarti nama variabel (komponen) yang akan digunakan dalam scripting. Properti “Name” ini hanya bisa diatur melalui jendela Property, sedangkan nilai peroperti yang lain bisa diatur melalui script seperti

```
Command1.Caption="Play"
Text1.Text="Visual Basic"
Label1.Visible=False
Timer1.Enable=True
```

- (2) **Metode:** Bahwa jalannya program dapat diatur sesuai aplikasi dengan menggunakan metode pemrograman yang diatur sebagai aksi dari setiap komponen. Metode inilah tempat untuk mengekspresikan logika pemrograman dari pembuatan suatu program aplikasi.
- (3) **Event:** Setiap komponen dapat beraksi melalui event, seperti event click pada command button yang tertulis dalam layar script `Command1_Click`, atau event `Mouse Down` pada picture yang tertulis dengan `Picture1_MouseDown`. Pengaturan event dalam setiap komponen yang akan menjalankan semua metode yang dibuat.

1.4. Membuat Project Baru

Untuk memulai pembuatan program aplikasi di dalam Visual Basic, yang dilakukan adalah membuat project baru. Project adalah sekumpulan form, modul, fungsi, data dan laporan yang digunakan dalam suatu aplikasi. Membuat project baru dapat dilakukan dengan memilih menu `[File] >> [New Project]` atau dengan menekan ikon `[new project]` pada Toolbar yang terletak di pojok kiri atas. Setelah itu akan muncul konfirmasi untuk jenis project dari program aplikasi yang akan dibuat seperti terlihat pada gambar 1.3. berikut.

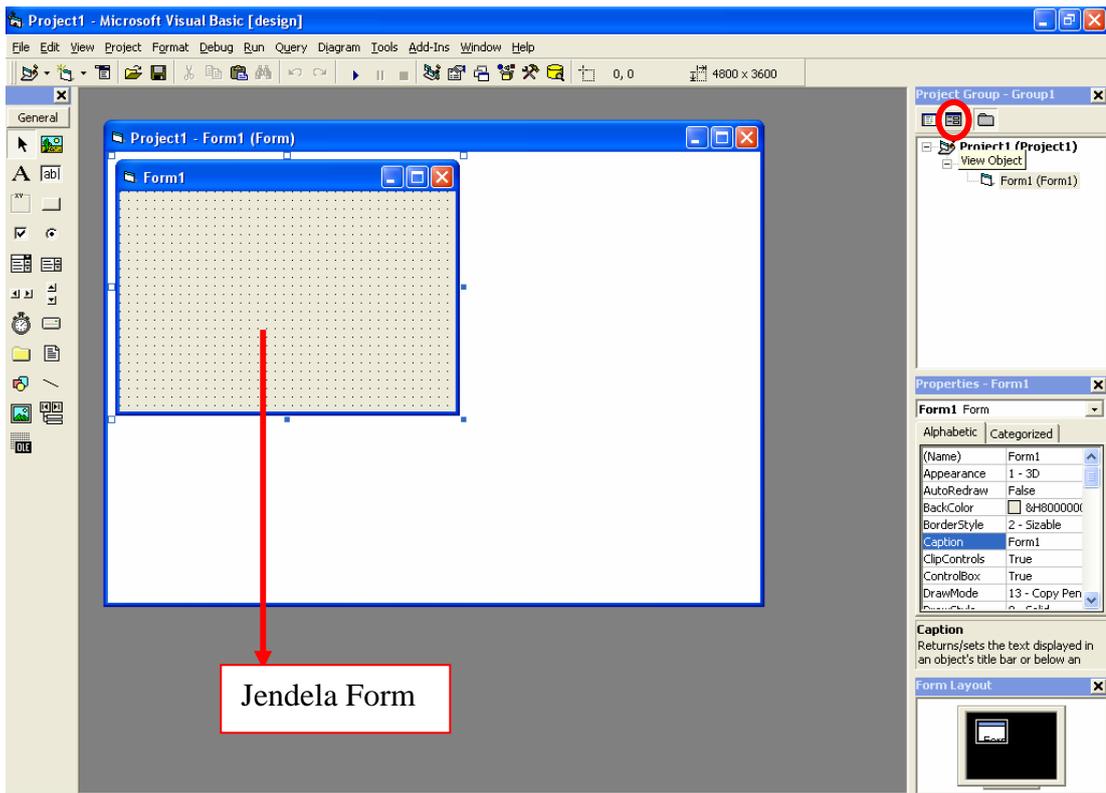


Gambar 1.3. Layar pemilihan jenis project

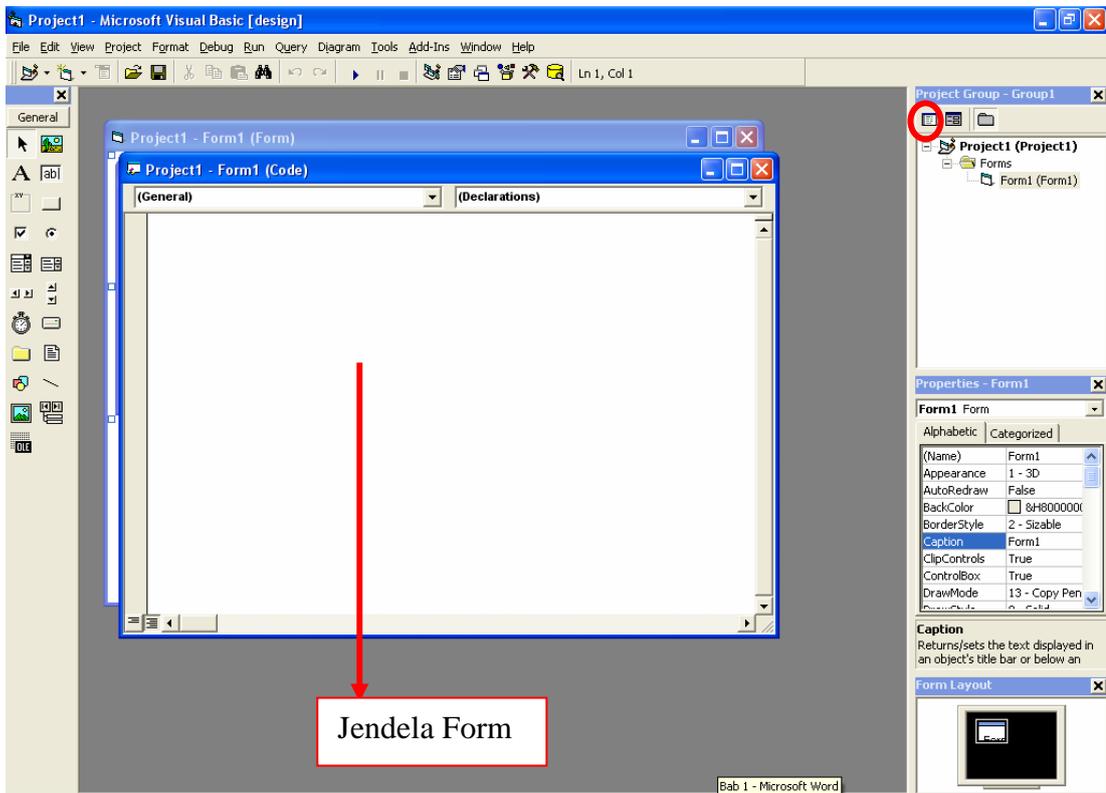
Visual Basic 6.0 menyediakan 13 jenis project yang bisa dibuat seperti terlihat pada gambar 1.3 di atas. Ada beberapa project yang biasa digunakan oleh banyak pengguna Visual Basic, antara lain:

- (1) Standard EXE: Project standar dalam Visual Basic dengan komponen-komponen standar. Jenis project ini sangat sederhana, tetapi memiliki keunggulan bahwa semua komponennya dapat diakui oleh semua unit komputer dan semua user meskipun bukan administrator. Pada buku ini akan digunakan project Standard EXE ini, sebagai konsep pemrograman visualnya.
- (2) ActiveX EXE: Project ini adalah project ActiveX berisi komponen-komponen kemampuan untuk berinteraksi dengan semua aplikasi di sistem operasi windows.
- (3) ActiveX DLL: Project ini menghasilkan sebuah aplikasi library yang selanjutnya dapat digunakan oleh semua aplikasi di sistem operasi windows.
- (4) ActiveX Control: Project ini menghasilkan komponen-komponen baru untuk aplikasi Visual Basic yang lain
- (5) VB Application Wizard: Project ini memandu pengguna untuk membuat aplikasi secara mudah tanpa harus pusing-pusing dengan perintah-perintah pemrograman.
- (6) Addin: Project seperti Standard EXE tetapi dengan berbagai macam komponen tambahan yang memungkinkan kebebasan kreasi dari pengguna.
- (7) Data project: Project ini melengkapi komponennya dengan komponen-komponen database. Sehingga bisa dikatakan project ini memang disediakan untuk keperluan pembuatan aplikasi database.
- (8) DHTML Application: Project ini digunakan untuk membuat aplikasi internet pada sisi client (*client side*) dengan fungsi-fungsi DHTML.
- (9) IIS Application: Project ini menghasilkan aplikasi internet pada sisi server (*server side*) dengan komponen-komponen CGI (*Common Gateway Interface*).

Selanjutnya pilih Standard EXE dan tekan [Ok]. Lalu muncul tampilan dari Standard Exe seperti pada gambar 1.1. Dengan demikian project sudah siap dibuat. Dalam pembuatan project sebelumnya double click pada form yang terbuat maka akan terlihat jendela tersembunyi (*hidden windows*) yang berupa jendela untuk pembuatan program atau jendela kode (*code windows*). Hal ini Dapat dilakukan dengan cara memilih ikon jendela form atau jendela kode yang ada di [Project Explorer]. Hal ini dapat dilihat pada gambar 1.4 dan gambar 1.5

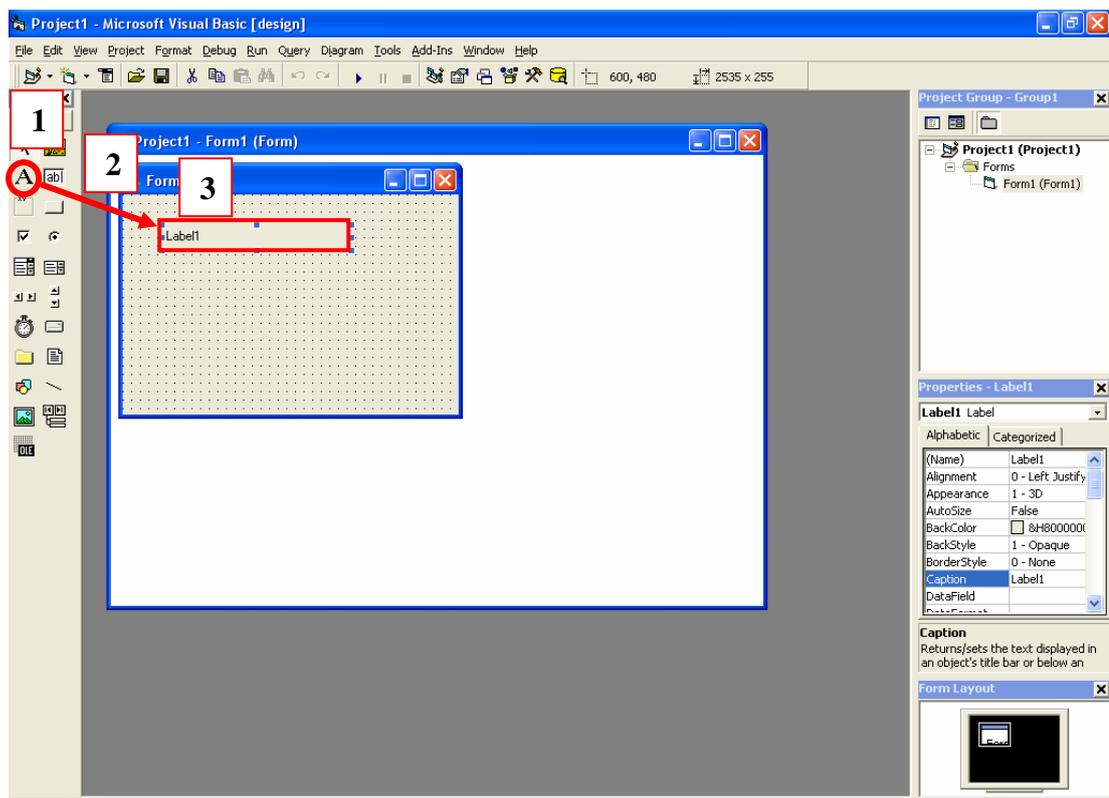


Gambar 1.4. Jendela Form



Gambar 1.5. Jendela Kode

Pada jendela form, pengguna dalam membangun tampilan dari program aplikasi yang akan dibuat dengan mengatur komponen-komponen baik letak, properti dan eventnya. Untuk mengambil suatu komponen dari [Toolbox] dapat dilakukan dengan click komponen tersebut, kemudian klik atau tarik pada posisi yang benar pada form. Sebagai contoh mengambil label dari Toolbox dapat dilakukan dengan cara seperti gambar 1.6 di bawah ini.



Gambar 1.6. Cara mengambil label dari Toolbox

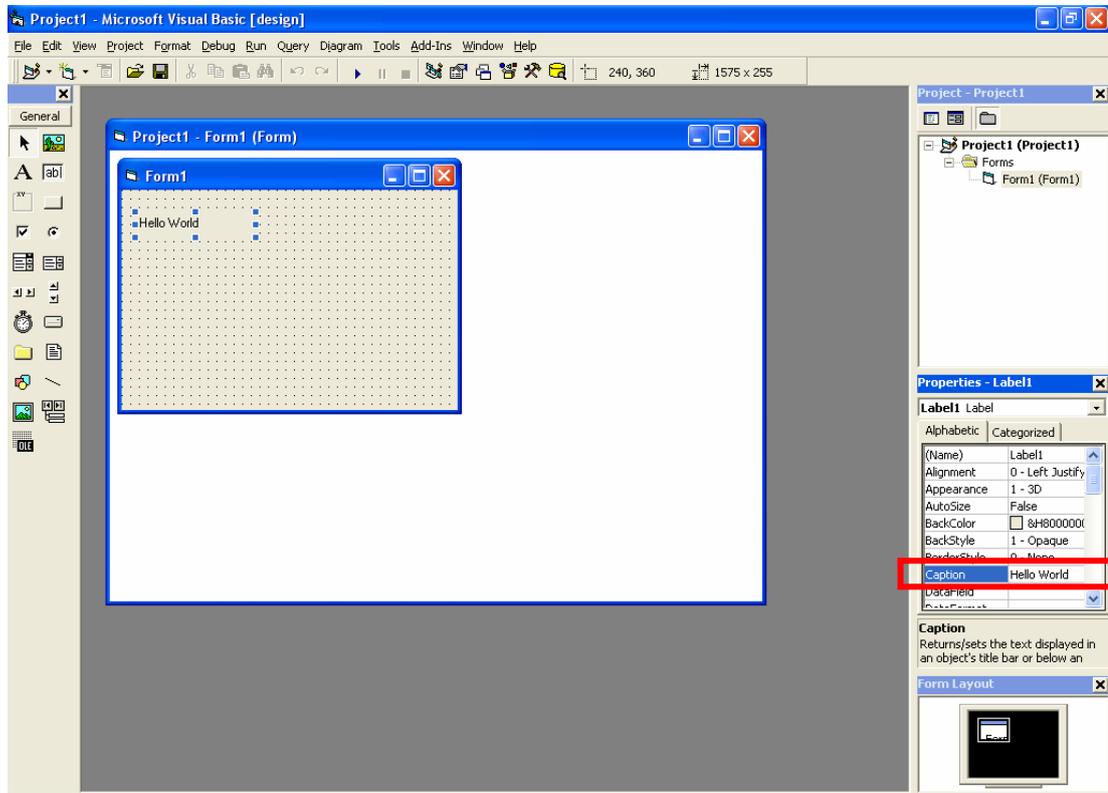
Langkah-langkah mengambil label dari toolbox untuk dipasangkan dalam form adalah sebagai berikut:

- (1) Click ikon [Label] pada [ToolBox]
- (2) Pindahkan ke posisi dimana label itu akan diletakkan
- (3) Klik dan tarik sampai ukurannya benar lalu lepaskan

Catatan:

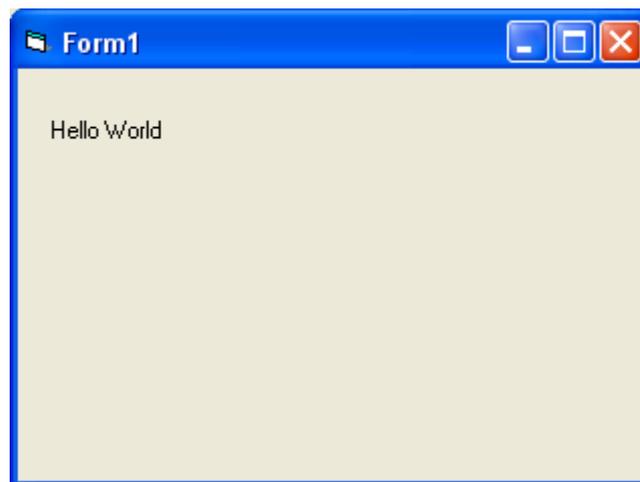
Jangan lupa untuk mengatur property name dari setiap komponen yang digunakan, karena name merupakan identitas obyek yang akan digunakan dalam menulis program.

Langkah berikutnya adalah memberikan teks pada label, misalkan “Hello world”, maka pilih properti Caption, dan isi dengan Hello world. Hasil tampilan program adalah sebagai berikut:



Gambar 1.7. Layout pada form

Dan untuk menjalankan program click ikon Run (▶) pada toolbar atau pilih menu [Run] >> [Start], atau dengan tekan tombol [F5]. Sehingga hasil program adalah:



Gambar 1.8. Hasil program

2

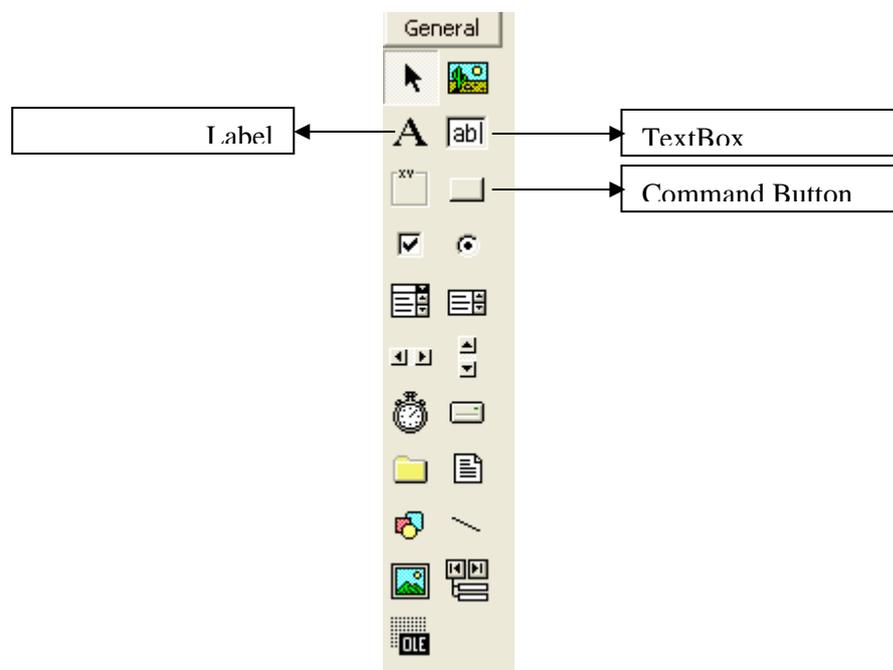
EVENT DAN PROPERTY

2.1. Membuat User Interface

Pemrograman Visual Basic adalah suatu pemrograman visual, dimana pembuatan program dilakukan menggunakan media visual atau sering disebut dengan *user-interface*. Yang artinya bahwa pembuatan program berdasarkan tampilan yang dihasilkan program, dengan kode-kode program (Script) diletakkan masing-masing komponen.

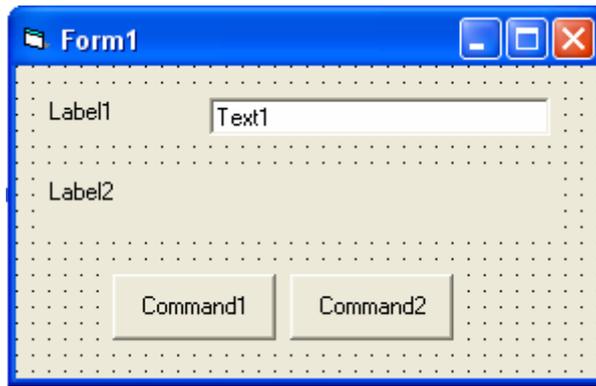
Contoh 1:

Buat project baru dengan StandartEXE untuk membuat User Interface sederhana dengan melibatkan komponen Label, Textbox dan CommandButton yang ada di Toolbox pada sebelah kiri dari antar muka Visual Basic seperti gambar 2.1 dan gambar 2.2 berikut:



Gambar 2.1. Komponen yang digunakan

Gunakan komponen-komponen seperti pada gambar 2.1 untuk membuat form pada gambar 2.2. berikut:

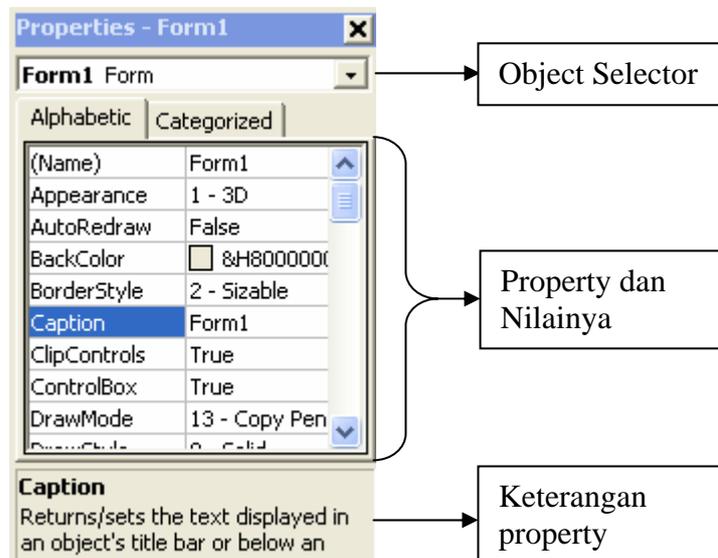


Gambar 2.2

Untuk mengatur letaknya tinggal menggunakan “*drag & drop*” dengan mouse. Program ini belum selesai karena masih perlu pengaturan **property** dan penambahan **event** pada masing-masing komponen untuk dapat memberikan tampilan yang dapat diterima oleh user dan dapat menjalankan proses.

2.2. Mengatur Property

Property pada tampilan antar muka Visual Basic terletak di sebelah kanan, seperti gambar 2.3 berikut:



Gambar 2.3. Tampilan property

Contoh 2:

Pada contoh 1 di atas, komponen-komponen yang sudah digunakan adalah Label1, Label 2, Text1, Command1 dan Command2. Atur property masing-masing komponen tersebut sebagai berikut, sehingga menghasilkan tampilan seperti gambar 2.4.

Tabel 2.1. Pengaturan property contoh 2

Object	Property	Nilai
Form1	Caption	Perkenalan
	StartPosition	2 – CenterScreen
Label1	Capton	Masukkan nama :
Label2	Alignment	- 3 Center
	Background	Pallete – Putih
	Font	Size: 24
Text1	Text	Kosong
Command1	Caption	OK
Command2	Caption	Keluar

Hasil form setelah propertynya di atur adalah:



Gambar 2.4. Hasil tampilan form contoh 2

Masing-masing komponen mempunyai property yang berbeda dan jumlahnya banyak, tetapi ada beberapa property yang sering digunakan pada setiap komponen, antara lain [Caption]. Property yang sering digunakan untuk Form antara lain:

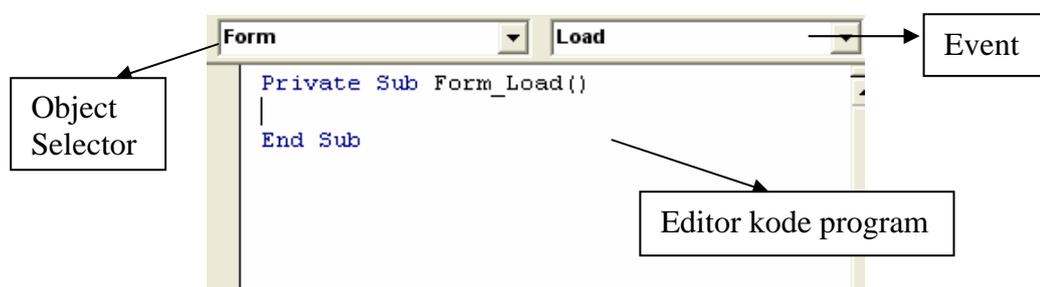
- Name: menyatakan nama obyek form yang sangat berguna untuk memanggil dan menyimpan form.
- Caption: digunakan untuk memberikan *title* pada form.
- StartUpPosition: digunakan untuk meletakkan form ketika form tersebut dipanggil atau aktif. Ada empat pilihan yaitu: Manual, CenterOwner, CenterScreen, Windows Default,

2.3. Event Dengan Kode Program

Pemrograman visual bersifat *event-driver*, yang artinya program bekerja berdasarkan event yang terjadi ketika suatu object diberikan aksi misalkan tombol ditekan, option dipilih, atau setelah mengetikkan sesuatu pada text kemudian di tekan [Enter]. Untuk membuat event ini tinggal click pada komponen dari tampilan user interface yang sudah dibuat.

Contoh 3:

Pada tampilan contoh 2, click pada form (bagian kosong yang tidak digunakan komponen-komponen lainnya), maka akan muncul tampilan seperti gambar 5 berikut:



Gambar 2.5. Tampilan untuk kode program

Perhatikan gambar 2.5, ketika form di-click maka akan muncul event Load, ini disebabkan event default untuk form adalah load. Dan secara otomatis di bagian kode program sudah disediakan fungsi untuk event load pada form yang tertulis:

```
Private Sub Form_Load()  
|  
End Sub
```

Di dalam fungsi ini dituliskan kode program. Kode program ini dijalankan ketika form dipanggil. Event ini bisa diganti di bagian [Event], perhatikan bahwa event pada setiap komponen termasuk form jumlahnya banyak, tinggal dipilih sesuai kebutuhan aplikasi.

Contoh 4:

Pada tampilan contoh 2, click pada command1, sehingga muncul event pada bagian kode program sebagai berikut:

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

Tambahkan program pada kode program ini sehingga menjadi:

```
Private Sub Command1_Click()  
Label2.Caption = Text1.Text  
End Sub
```

Program ini berarti bahwa apa yang diketikkan pada text1 akan ditampilkan pada label2.

Kemudian click pada command2, dan tambahkan kode program sehingga menjadi:

```
Private Sub Command2_Click()  
End  
End Sub
```

Perintah End, berarti program keluar dan selesai.

Simpan form dan project ini dengan cara pilih menu [File] >> [Save Project], beri nama form dengan formLatihan21 dan nama project dengan projectLatihan21.

Jalankan dengan menekan ikon Run (▶) pada toolbar. Masukkan nama misalnya “Achmad Basuki” pada text1, dan tekan tombol Ok. Hasilnya adalah seperti gambar 2.6 berikut.

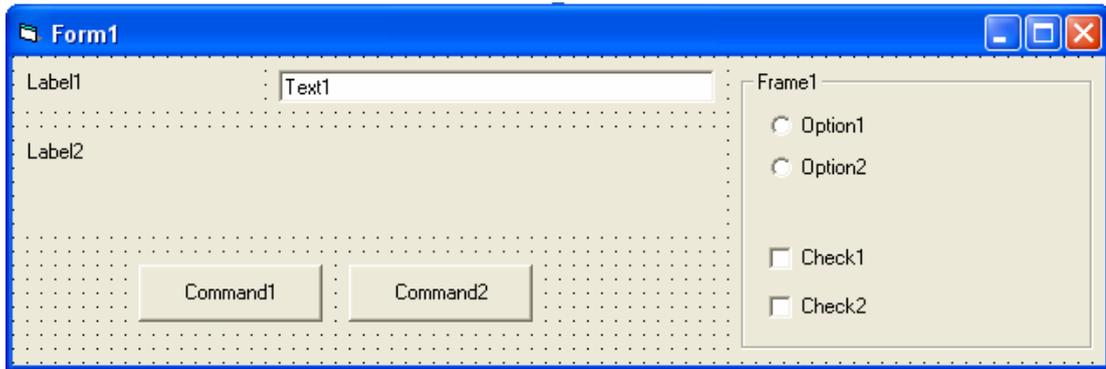


Gambar 2.6. Hasil program latihan 2.1

2.4. Membuat Project Mengatur Property

Contoh 5:

Buat project baru dengan StandardEXE untuk mencoba bermain dengan property dan event lebih jauh lagi. Tambahkan komponen-komponen Label, Text, Frame, OptionButton, CheckBox dan Command. Atur tampilannya seperti gambar 2.7 berikut:



Gambar 2.7. Contoh tampilan awal latihan 2.2

Bila menggunakan tampilan di dalam frame, maka frame harus dibuat terlebih dahulu baru komponen-komponen di dalamnya. Atur property setiap komponen seperti tabel 2.2. berikut:

Tabel 2.2 Pengaturan property latihan 2.2

Object	Property	Nilai
Form1	BorderStyle	1 – Fixed Single
	Caption	Test Untuk Property
	StartPosition	2 – CenterScreen
Label1	Caption	Masukkan nama :
Label2	Alignment	- 3 Center
	Background	Pallette – Putih
	Font	Size: 18
Text1	Text	Kosong
Command1	Caption	OK
Command2	Caption	Keluar
Frame1	Caption	Pilihan property:
Option1	Caption	Merah
Option2	Caption	Biru
Check1	Caption	Miring
Check2	Caption	Tebal

Dari pengaturan ini akan dihasilkan tampilan program aplikasi seperti pada gambar 2.8 di bawah ini.



Gambar 2.8. Tampilan untuk latihan 2.2

Click pada command1, tambahkan kode program sehingga menjadi:

```
Private Sub Command1_Click()
    Label2.Caption = Text1.Text
End Sub
```

Click pada command2, tambahkan kode program sehingga menjadi:

```
Private Sub Command2_Click()
    End
End Sub
```

Click pada Option1, tambahkan kode program sehingga menjadi:

```
Private Sub Option1_Click()
    Label2.ForeColor = vbRed
End Sub
```

Label2.ForeColor adalah properti untuk mengganti warna teks pada label2, dan vbRed adalah warna merah yang sudah disediakan oleh Visual Basic, untuk warna yang lain seperti biru menggunakan vbBlue.

Click pada Option2, tambahkan kode program sehingga menjadi:

```
Private Sub Option2_Click()
    Label2.ForeColor = vbBlue
End Sub
```

Click pada check1, tambahkan kode program sehingga menjadi:

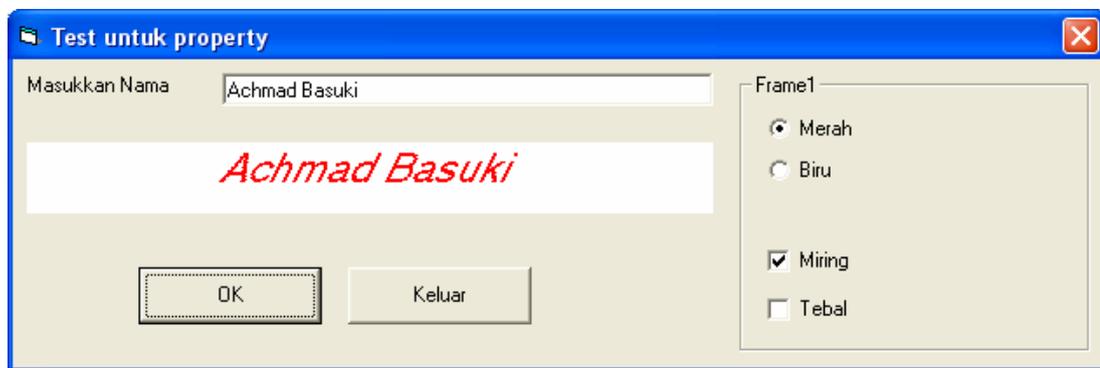
```
Private Sub Check1_Click()
    Label2.FontItalic = Check1.Value
End Sub
```

Click pada check2, tambahkan kode program sehingga menjadi:

```
Private Sub Check2_Click()  
Label2.FontBold = Check2.Value  
End Sub
```

Label2.FontItalic adalah property untuk mengatur apakah teks pada label2 dibuat miring atau tidak, bila nilainya True maka teks menjadi miring. Label2.FontBold adalah property untuk mengatur apakah teks pada label2 dibuat tebal atau tidak, bila nilainya True maka teks menjadi tebal.

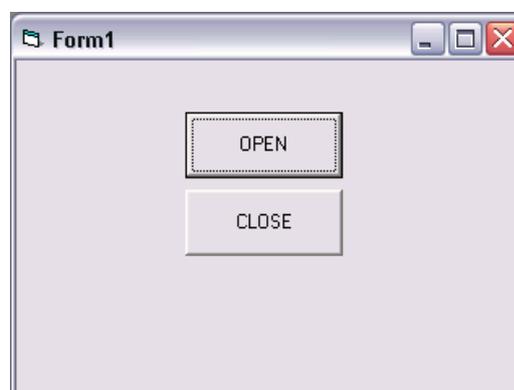
Simpan project dengan nama projectLatihan22. Dan jalankan program, masukkan nama Achmad Basuki, atur teksnya menjadi merah dan miring seperti gambar 2.9.



Gambar 2.9. Hasil dari latihan 2.2

Contoh 6:

Buat project baru dengan StandardEXE untuk membuat aplikasi yang menggunakan event dan property untuk bermain dengan command button. Tambahkan komponen-komponen Command Button. Atur tampilannya seperti gambar 2.10. berikut:



Gambar 2.10. Tampilan form contoh 6

Program ini bertujuan bila ditekan OPEN maka tombol OPEN tidak kelihatan, dan tinggal tombol CLOSE. Bila ditekan CLOSE maka tombol CLOSE tidak kelihatan dan tinggal tombol OPEN.

Click pada command1, tambahkan program untuk event click sebagai berikut:

```
Private Sub Command1_Click()  
    Command1.Visible = False  
    Command2.Visible = True  
End Sub
```

Click pada command2, tambahkan program untuk event click sebagai berikut:

```
Private Sub Command2_Click()  
    Command2.Visible = False  
    Command1.Visible = True  
End Sub
```

Simpan form dalam formLatihan24, dan simpan project dalam projectLatihan24.

2.5. Latihan

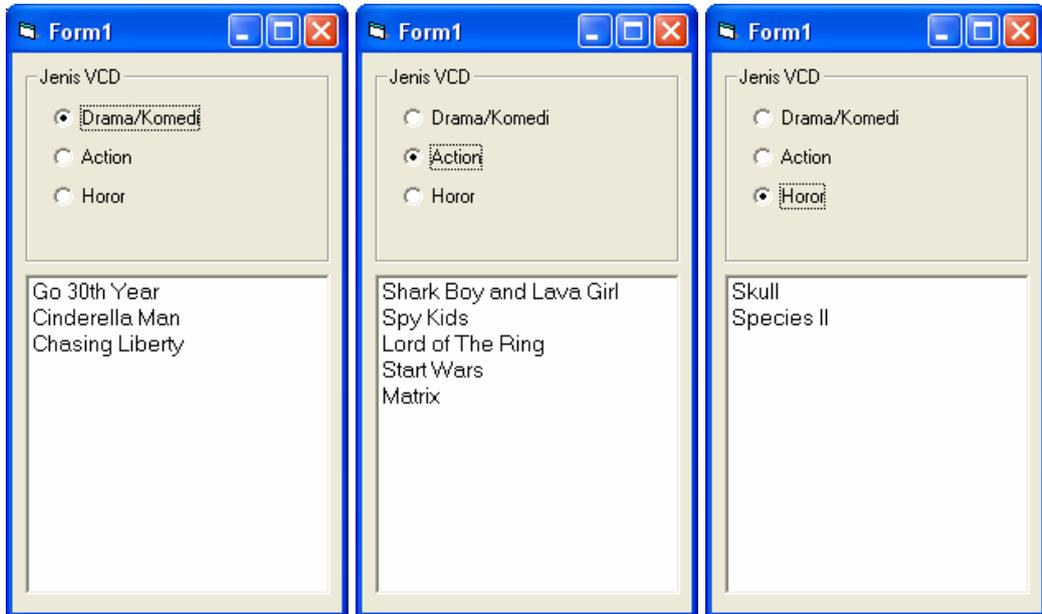
(1) Buat program untuk memilih menu makanan dan minuman menggunakan `OptionButton` dan `Label` tampilan seperti gambar 2.10 berikut.



The screenshot shows a Windows application window titled "Form1". It contains two columns of radio buttons for selecting food and drinks. The "Makanan" column has options: Soto Ayam, Rawon, Gado-gado, Krengsengan (selected), and Pecel Madiun. The "Minuman" column has options: Teh Manis, Es Teh (selected), Es Jeruk, and Kopi. Below the columns, there are two large labels: "Krengsengan" and "Es Teh".

Gambar 2.10. Soal latihan 1

- (2) Buat program untuk menampilkan judul VCD sesuai dengan jenis VCD yang dipilih menggunakan Option Button dan ListBox seperti gambar 2.11 berikut. Gunakan property Clear pada listbox untuk membersihkan tampilan sebelumnya pada listbox, dan property AddItem pada listbox untuk menambahkan teks pada listbox.



Gambar 2.11. Soal latihan 2

- (3) Buat program untuk mengganti warna latar belakang form menggunakan Frame dan optionButton seperti gambar 2.12 berikut. Gunakan event change pada combobox, dan property bgcolor pada form.



Gambar 2.12. Soal Latihan 3

- (4) Buat program untuk Mengatur alignment teks menggunakan OptionButton dan label seperti gambar 2.13 berikut. Gunakan property alignmet pada label untuk mengatur alignment.



Gambar 2.13. Soal Latihan 4

3.1. Variabel

Variabel adalah suatu tempat untuk menampung suatu nilai pada memory komputer. Untuk lebih mudah diakses, variabel diberi nama. Nama inilah yang akan menjadi pengenal suatu variabel, misalkan $a=4$, ini berarti bahwa nama variabelnya adalah a dan nilainya adalah 4.

Variabel di dalam visual basic mempunyai beberapa macam berdasarkan nilai yang ada di dalam variabel tersebut, antara lain:

- (1) Boolean : menampung nilai biner, True atau False
- (2) Byte : Menampung nilai bulat kecil antara 0-256
- (3) Integer : Menampung nilai bulat antara -32768 s/d 32768 (15 bit)
- (4) Long : Menampung nilai bulat dengan bit yang panjang (31 bit)
- (5) Single : Menampung nilai pecahan dari 10^{-38} sampai dengan 10^{38} pada bagian positif, dan -10^{-38} sampai dengan -10^{38} pada bagian negatif.
- (6) Double : Menampung nilai pecahan dari 10^{-108} sampai dengan 10^{108} pada bagian positif, dan -10^{-108} sampai dengan -10^{108} pada bagian negatif.
- (7) String : Menampung nilai non numerik atau string, misalkan untuk menyimpan alamat. Variabel ini tidak bisa dioperasikan secara aritmatika.
- (8) Date : Menampung nilai tanggal
- (9) Variant : Merupakan variabel bebas, yang menampung nilai tergantung nilai apa yang ditampung pertama kali. Variabel ini yang merupakan kelebihan dari visual basic (bahkam bisa menjadi kekurangan untuk pemrograman yang lebih tinggi) karena bersifat seperti bunglon.

Kelebihan dari visual basic adalah bahwa setiap obyek baik berupa form, report (laporan), data atau komponen dapat menjadi variabel.

Variabel di dalam visual basic dibedakan menjadi 3 macam variabel yaitu variabel lokal, variabel global terhadap form dan variabel global terhadap aplikasi (project).

- (1) Variabel lokal: adalah variabel yang hanya aktif dalam suatu fungsi atau subroutine di dalam visual basic. Caranya dituliskan secara langsung di dalam suatu fungsi atau subroutine tanpa pendefinisian atau dengan definisi variabel di dalam fungsi.
- (2) Variabel global dalam form: adalah variabel yang aktif selama satu form berjalan, bila pindah ke form yang lain maka variabel ini tidak aktif. Ini dapat dilakukan dengan mendefinisikan variabel di luar subroutine, biasanya diletakkan pada baris paling atas dari suatu form. Contohnya **Dim a as integer** diletakkan pada baris paling atas, maka variabel a selalu aktif selama form aktif.
- (3) Variabel global dalam aplikasi: adalah variabel yang aktif selama aplikasi masih aktif. Variabel ini masih aktif meskipun form yang berjalan sudah berganti. Untuk mendefinisikan variabel ini dilakukan dengan menambahkan modul dan pendefinisian dengan **global**.

3.2. Operator

Operator adalah suatu simbol atau tanda untuk menyatakan suatu operasi atau proses. Pada dasarnya komputer dengan ALUnya (Aritmethic Logical Unit), mempunyai dua macam operator yaitu operator Aritmatika dan operator logika (perbandingan). Operator-operator di dalam Visual Basic antara lain:

Jenis	Operator	Kegunaan
Aritmatika	+	Penjumlahan
	-	Pengurangan
	*	Perkalian
	/	Pembagian
	^	Pangkat
	mod	Sisa pembagian
	\	Hasil bulat pembagian
	&	Penggabungan string

Jenis	Operator	Kegunaan
Relasi	=	Sama dengan
	<	Lebih kecil
	<=	Lebih kecil atau sama dengan
	>	Lebih besar
	>=	Lebih besar atau sama dengan
	<>	Tidak sama
Logika	AND	Dua kondisi harus dipenuhi
	OR	Dari dua kondisi, akan benar bila ada salah satu atau lebih kondisi yang dipenuhi
	NOT	Invers dari kondisi yang diberikan

Penjelasan lebih lanjut mengenai operator ada di bagian ekspresi, karena operator ini merupakan bagian dari ekspresi. Pemakaian operator ini sangat berhubungan dengan penulisan rumus di dalam pemrograman.

3.3. Ekspresi

Ekspresi adalah suatu cara penulisan untuk memberikan atau memasukkan nilai ke dalam variabel. Ekspresi secara umum dalam *computer statement* dituliskan sebagai:

Variabel ← Nilai

Di dalam Visual Basic ekspresi menggunakan tanda sama dengan (=). Dengan aturan sebelah kiri adalah variabel penampung (hasil) dan sebelah kanan adalah nilai yang dimasukkan ke variabel

Variabel = Nilai

Sebagai contoh untuk memasukkan nilai 5 ke dalam variabel a dapat dilakukan dengan **a=5**. Atau memasukkan nama 'widya' ke dalam variabel nama dapat dituliskan dengan **nama="widya"** (Pada tipe data string, penulisan diberi tanda petik dua). Atau untuk memasukkan nilai keputusan benar dapat dituliskan dengan **keputusan=True**, Pada tipe data boolean nilainya hanya True dan False. Sehingga dapat dikatakan bahwa cara memasukkan nilai ini sangat tergantung dari jenis nilai apakah itu boolean, numerik, string atau date. Sedangkan format bilangan dan tanggal

secara default menggunakan format yang ada pada sistem operasi, meskipun tidak menutup kemungkinan diubah menggunakan format tersendiri.

Berikut ini contoh-contoh penulisan ekspresi untuk memasukkan nilai ke dalam variabel:

Ekspresi	Arti dari ekspresi
<code>a=10.5</code>	Nilai 10.5 dimasukkan ke dalam variabel a, tanda titik berarti nilai pecahan dalam desimal
<code>tg="10-02-2006"</code>	Memasukkan tanggal 2 Oktober 2006 ke dalam variabel tg, format tanggal secara umum menggunakan format tanggal sistem operasi yaitu bulan-tanggal-tahun, sebelum melakukan ekspresi ini sebaiknya didefinisikan variabel tg dengan cara: Dim tg as date
<code>Nrp="7403030010"</code>	Memasukkan format string "7403030010" ke dalam variabel Nrp
<code>Ketemu=True</code>	Memasukkan nilai True ke dalam variabel boolean Ketemu

Ekspresi merupakan suatu proses yang bersifat *sequential*, yang artinya bahwa proses dilakukan dari baris paling atas sampai baris terakhir. Sebagai contoh bila dituliskan:

a=10

a=5

Maka artinya pada baris pertama a bernilai 10, dan pada baris kedua a bernilai 5, sehingga nilai 10 diganti dengan nilai 5. Sehingga hasilnya a bernilai 5.

Ekspresi bukan hanya seperti diatas, tetapi dapat juga merupakan *penulisan* suatu formula dengan melibatkan variabel-variabel yang sudah ada sebelumnya.

Contoh 1:

a=5

b=10

c=a+b

Hasilnya variabel a bernilai 5, b bernilai 10 dan c bernilai 15 sebagai hasil dari $a+b=5+10=15$.

Ekspresi dapat digunakan untuk melakukan “*counting*” yaitu perhitungan penjumlahan secara terus menerus terhadap suatu variabel

Contoh 2:

$$a=5$$

$$a=a+2$$

Pada baris pertama a bernilai 5, pada baris kedua a bernilai 7, karena a yang sebelumnya bernilai 5 ditambah dengan 2 sehingga nilai akhir a bernilai 7.

Contoh 3:

$$\text{jumlah}=0$$

$$\text{jumlah}=\text{jumlah}+5$$

$$\text{jumlah}=\text{jumlah}-3$$

$$\text{jumlah}=\text{jumlah}+2$$

Pada baris pertama variabel jumlah bernilai 0, pada baris kedua variabel jumlah bernilai 5, pada baris ketiga variabel jumlah bernilai 2 dan pada baris keempat variabel jumlah bernilai 4.

Contoh 4:

Untuk menuliskan ekspresi dari rumus: $x = \sin(t)$ dengan t diketahui misalkan $t=0.5$ dapat dilakukan dengan:

$$t=0.5$$

$$x=\sin(t)$$

Contoh 5:

Untuk menuliskan rumus ABC dalam menyelesaikan persamaan kuadrat sebagai berikut:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

dengan nilai a, b dan c diketahui misalkan 1, 3 dan 2 adalah:

$$a=1: b=3: c=2$$

$$x1=(-b+(b^2-4*a*c)^{0.5})/(2*a)$$

$$x2=(-b-(b^2-4*a*c)^{0.5})/(2*a)$$

Contoh 6.

Untuk memasukkan rumus $y = x^2 + 3x + 2$, dimana $x=2$ dapat dituliskan dengan:

$$x = 2$$

$$y = x^2 + 3*x + 2$$

Contoh 7.

Untuk menukar nilai $a=10$ dan $b=5$ maka diperlukan proses swap. Proses swap ini dapat dijelaskan dengan proses menukar benda pada dua tangan dimana masing-masing tangan hanya boleh memegang satu benda

Untuk melakukan proses swap antara a dan b , maka diperlukan variabel penampung temp, dan dapat dituliskan dengan:

temp=a

a=b

b=temp

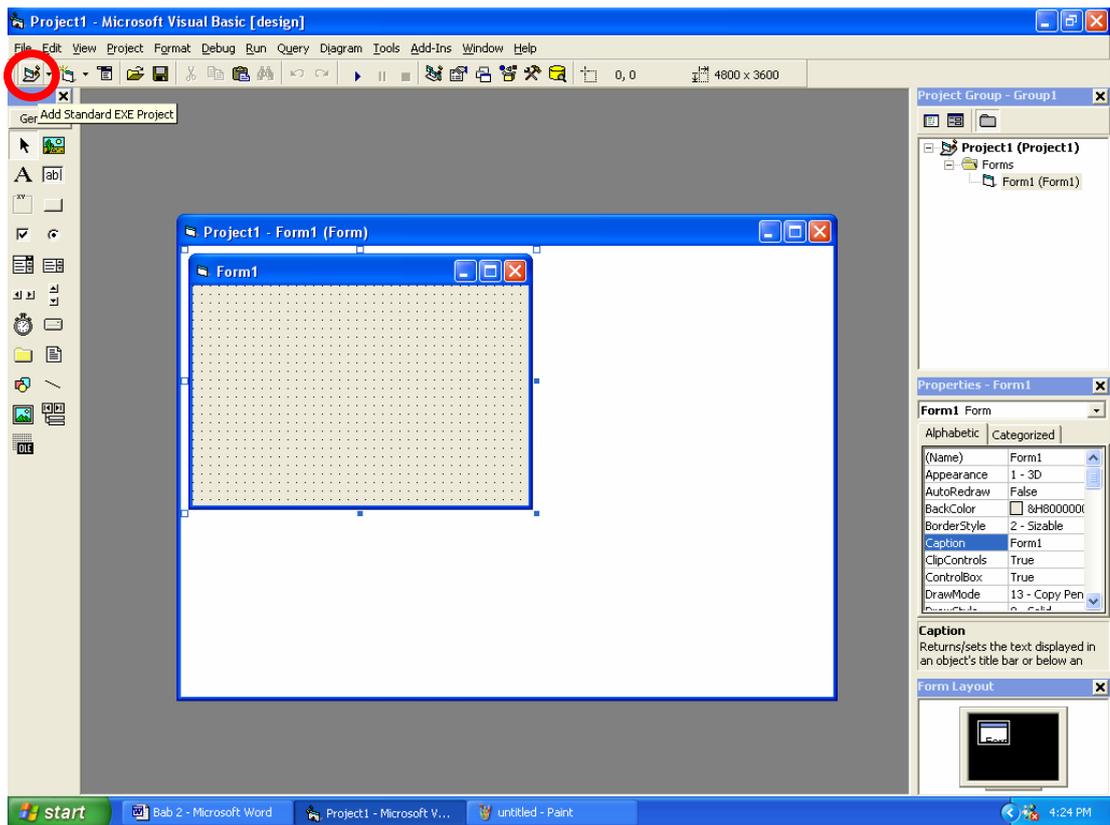
3.4. Project Sederhana Dengan Ekspresi

Pembuatan project sederhana dalam bab ini mempunyai arti pembuatan project dengan input/output terbatas tanpa penggunaan komponen yang rumit. Hal ini digunakan untuk menjelaskan kegunaan variabel dan ekspresi di dalam visual basic, sekaligus memperkenalkan *event* dan *subroutine*, yang secara detail penggunaan event dan subroutine ini akan dijelaskan pada bab 6 (fungsi dan subroutine).

3.4.1. Membuat Penjumlahan, Pengurangan dan Perkalian

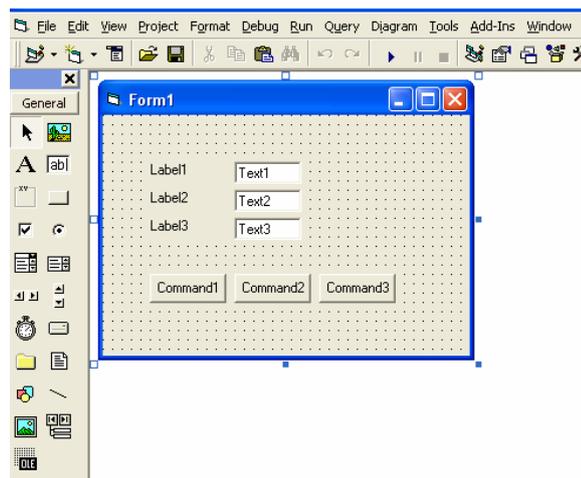
Untuk membuat project yang menyajikan operasi aritmatika sederhana seperti penjumlahan, pengurangan dan perkalian dapat dilakukan dengan langkah-langkah sebagai berikut:

- (1) Membuat project baru, dengan click ikon add standard EXE project () yang adalah pada bagian sebelah kiri toolbar (tanda lingkaran merah pada gambar 3.1 di bawah ini).



Gambar 3.1 Contoh pembuatan project baru

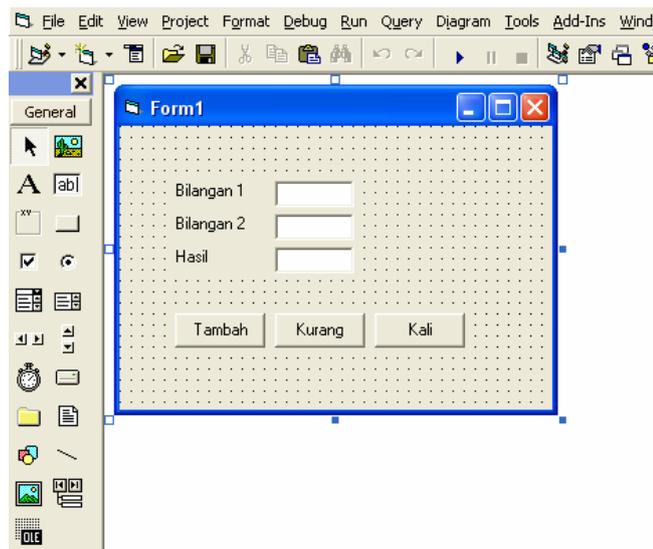
(2) Tambahkan komponen-komponen: 3 komponen label, 3 komponen textbox dan 3 komponen commandButton yang ada di [toolbox]. Atur tempatnya pada form seperti gambar 3.2. berikut.



Gambar 3.2. Contoh penaturan letak setiap komponen

(3) Pada komponen label1, isikan property [Caption] dengan Bilangan 1. Pada label2, isikan property [Caption] dengan Bilangan 2. Dan pada label 3 isikan property [Caption] dengan Hasil.

- (4) Pada komponen textbox1, isikan property [Name] dengan bil1 dan property [Text] dikosongkan. Pada textbox2, isikan property [Name] dengan bil2 dan property [Text] dikosongkan. Dan pada textbox3, isikan property [Name] dengan bil3 dan property [Text] dikosongkan.
- (5) Pada komponen command1, isikan property [Caption] dengan Tambah. Pada command2, isikan property [Caption] dengan Kurang. Dan pada command3, isikan property [Caption] dengan Kali. Sehingga hasil tampilan form menjadi seperti gambar 3.3.



Gambar 3.3. Hasil tampilan project 1

- (6) Click pada command1 (Tambah), sehingga muncul tampilan script dengan disediakan perintah dari event command_click

```
Private Sub Command1_Click()
```

```
End Sub
```

Isikan program penjumlahan dari bil1 dan bil2, hasilnya ditampung dalam bil3 sebagai berikut

```
Private Sub Command1_Click()
```

```
a=val(bil1)
```

```
b=val(bil2)
```

```
c=a+b
```

```
bil3 = c
```

```
End Sub
```

Val merupakan suatu fungsi yang digunakan untuk mengkonversi tipe data string atau variant seperti pada textbox menjadi tipe data numerik sehingga bisa dihitung dengan operasi aritmatika.

- (7) Click pada command1 (kurang), sehingga muncul tampilan script dengan disediakan perintah dari event command_click

```
Private Sub Command2_Click()
```

```
End Sub
```

Isikan program pengurangan dari bil1 dan bil2, hasilnya ditampung dalam bil3 sebagai berikut

```
Private Sub Command2_Click()
```

```
a=val(bil1)
```

```
b=val(bil2)
```

```
c=a-b
```

```
bil3 = c
```

```
End Sub
```

- (8) Click pada command3 (kali), sehingga muncul tampilan script dengan disediakan perintah dari event command_click

```
Private Sub Command3_Click()
```

```
End Sub
```

Isikan program perkalian dari bil1 dan bil2, hasilnya ditampung dalam bil3 sebagai berikut

```
Private Sub Command1_Click()
```

```
a=val(bil1)
```

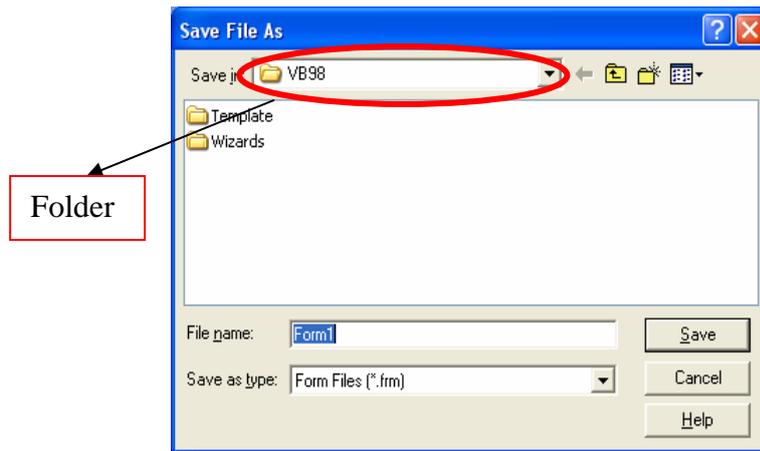
```
b=val(bil2)
```

```
c=a*b
```

```
bil3 = c
```

```
End Sub
```

- (9) Project sudah selesai, untuk menjalankan tekan F5. Cobalah untuk mengisi Bilangan 1 dan Bilangan 2 dengan sembarang nilai numerik, lalu click tombol-tombol tambah, kurang dan kali. Bila hasilnya benar maka project aplikasi dikatakan benar. Untuk menyimpan project pilih menu [File] >> [Save Project]



Gambar 3.4. Contoh jendela untuk menyimpan form

Pilih Folder mana untuk menyimpan file, sebaiknya buat folder baru di My Documents. Isikan nama file dengan FormLatihan31 untuk form lalu tekan [Save]. Setelah itu muncul jendela yang sama untuk project, isikan nama project dengan ProjectLatihan31 dan tekan [Save].

3.4.2. Menghitung Hari

Salah satu aplikasi yang banyak digunakan adalah menentukan selisih hari, biasanya digunakan untuk menentukan kurang berapa hari suatu pekerjaan harus selesai (*deadline*) atau berapa hari lagi masa berlaku suatu kartu dan lainnya. Untuk membuat aplikasi untuk menghitung hari adalah sebagai berikut:

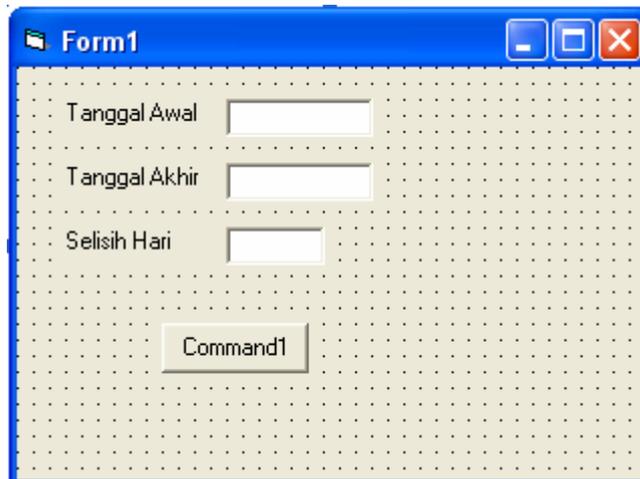
- (1) Buat project baru dengan Standard EXE
- (2) Tambahkan komponen-komponen: 3 label, 3 textbox, dan 1 command button.
- (3) Untuk label1, isikan property [Caption] dengan Tanggal Awal. Untuk label2 isikan property [Caption] dengan Tanggal Akhir. Dan untuk label3 isikan property [Caption] dengan Selisih hari.
- (4) Untuk setiap textbox, kosongkan property [Text].
- (5) Untuk command1, isikan property [Caption] dengan Hitung.
- (6) Atur posisinya seperti gambar 3.5 di bawah ini.
- (7) Click pada command1, pada layar code isikan program berikut:

```
Private Sub Command1_Click()
    Dim tg1 As Date
    Dim tg2 As Date
    tg1 = Text1
    tg2 = Text2
```

Text3 = tg2 - tg1

End Sub

Untuk tipe data tanggal, variabel perlu didefinisikan terlebih dahulu sehingga tipe data variant pada textbox dapat diterjemahkan menjadi date. Langkah berikutnya tinggal mengurangi, hasil pengurangan berupa jumlah hari selisih hari.



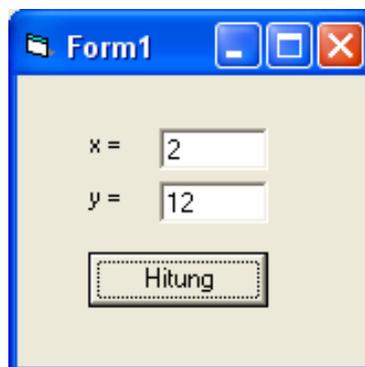
Gambar 3.5. Tampilan project selisih hari

- (8) Program selesai, simpan form dengan FormLatihan32 dan project dengan ProjectLatihan32. Untuk menjalankan program tekan F5. Jalankan program yang sudah dibuat, bila hasilnya adalah selisih hari antara tanggal akhir dan tanggal awal maka hasilnya benar.

3.5. Latihan

Buatlah project sederhana untuk:

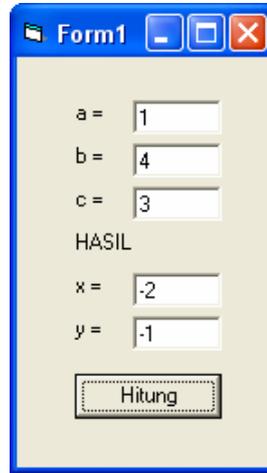
- (1) Menghitung nilai persamaan $y=x^2+3x+2$ dengan x diketahui, dan tampilan seperti gambar 3.6. berikut:



Gambar 3.6. Contoh hasil latihan 1

- (2) Menghitung nilai puncak persamaan kuadrat $y=ax^2+bx+c$ dengan a, b dan c diketahui menggunakan rumus:

$$x = -\frac{b}{2a} \text{ dan masukkan nilai } x \text{ kedalam persamaan } y = ax^2 + bx + c$$

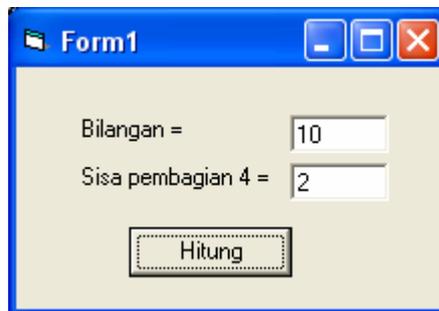


The screenshot shows a Windows application window titled "Form1". It contains a form with the following elements:

- Input field for 'a' with the value '1'.
- Input field for 'b' with the value '4'.
- Input field for 'c' with the value '3'.
- Text label "HASIL".
- Output field for 'x' with the value '-2'.
- Output field for 'y' with the value '-1'.
- A button labeled "Hitung" at the bottom.

Gambar 3.7. Contoh hasil latihan 2

- (3) Menghitung sisa pembagian bilangan dibagi 4 dengan bilangan diketahui menggunakan rumus sisa=bilangan mod 4.



The screenshot shows a Windows application window titled "Form1". It contains a form with the following elements:

- Input field for "Bilangan =" with the value "10".
- Output field for "Sisa pembagian 4 =" with the value "2".
- A button labeled "Hitung" at the bottom.

Gambar 3.8. Contoh hasil latihan 3

- (4) Menghitung tanggal setelah 30 hari dari tanggal yang dimasukkan.



The screenshot shows a Windows application window titled "Form1". It contains a form with the following elements:

- Input field for "Tanggal =" with the value "1/3/2006".
- Output field for "Tanggal setelah 30 hari =" with the value "2/2/2006".
- A button labeled "Hitung" at the bottom.

Gambar 3.9. Contoh hasil latihan 4

(5) Menghitung harga pembelian komputer dengan spesifikasi sebagai berikut:

harga komputer Rp. 4.000.000,-

harga printer Rp. 550.000,-

harga monitor Rp. 750.000,-

Dengan memasukkan jumlah komputer, monitor dan printer. Hitung total pembelian sengan tampilan sebagai berikut.

JUMLAH		
Jumlah komputer =	3	12000000
Jumlah monitor =	3	2250000
Jumlah printer =	1	550000
JUMLAH PEMBELIAN		14800000

Gambar 3.10. Contoh hasil latihan 5

4

KONDISI DAN KEPUTUSAN

4.1. Percabangan dengan If...Then...Else...

Percabangan berdasarkan kondisi suatu variabel atau nilai di dalam pemrograman Visual Basic dinyatakan dengan perintah:

```
If <kondisi> Then
    < Keputusan kondisi benar >
Else
    < Keputusan kondisi salah >
End If
```

Penulisan percabangan seperti di atas disebut dengan “**kondisi dan keputusan**”.

Dalam pengertian yang umum bisa dikatakan:

Jika memenuhi kondisi maka lakukan proses keputusan untuk kondisi benar, jika tidak lakukan proses keputusan untuk kondisi salah

Contoh 1:

“Jika bilangan habis dibagi dua maka bilangan itu adalah bilangan genap, jika tidak bilangan itu adalah bilangan ganjil”, yang dituliskan dengan:

```
If bilangan mod 2 = 0 Then
    Ket = "Bilangan genap"
Else
    Ket = "Bilangan ganjil"
End If
```

Contoh 2:

“Jika peminjaman lebih dari 7 hari maka dikenakan denda sebesar 2000”, yang dituliskan dengan

```
If peminjaman>7 Then
    Denda = 2000
```

```

Else
    Denda = 0
End If

```

Contoh Aplikasi 4.1:

Membuat aplikasi untuk menentukan apakah bilangan yang dimasukkan adalah bilangan genap atau ganjil.

- (1) Buat project baru
- (2) Pada form1, tambahkan komponen-komponen: 2 Label, 2 TextBox dan 1 command button
- (3) Pada label1, isi property [Caption] dengan Bilangan. Pada label2, isi property [Caption] dengan Keterangan.
- (4) Pada semua textbox, kosongkan property [Text].
- (5) Pada Command1, isi property [Caption] dengan Cek.
- (6) Atur tampilannya seperti gambar 4.1 berikut:



Gambar 4.1. Tampilan contoh 4.1

- (7) Click Command1, tambahkan program untuk mengecek apakah bilangan yang dimasukkan bilangan genap atau ganjil dengan cara : “Jika bilangan habis dibagi dua maka bilangan tersebut genap, jika tidak bilangan tersebut ganjil”

```

Private Sub Command1_Click()
    bil = Val(Text1)
    If bil Mod 2 = 0 Then
        ket = "GENAP"
    Else
        ket = "GANJIL"
    End If
    Text2 = ket
End Sub

```

(8) Simpan project ini dengan memilih menu [File] >> [Save Project]. Maka simpan form dengan nama formLatihan41, dan simpan project dengan nama projectLatihan41.

(9) Jalankan program dan uji dengan beberapa bilangan.

Penulisan percabangan di dalam Visual Basic mempunyai beberapa format, antara lain:

(1) Jika kondisi dipenuhi maka dilakukan proses yang hanya satu baris perintah

```
If kondisi Then <Proses Kondisi Benar>
```

(2) Jika kondisi dipenuhi maka dilakukan proses yang lebih dari satu baris perintah

```
If kondisi Then  
    <Proses Kondisi Benar>  
End If
```

(3) Jika kondisi dipenuhi maka dilakukan proses A, jika tidak dilakukan proses B yang masing-masing hanya satu baris perintah

```
If kondisi Then <Proses A> Else <Proses B>
```

(4) Jika kondisi dipenuhi maka dilakukan proses A, jika tidak dilakukan proses B yang masing-masing hanya satu baris perintah

```
If kondisi Then  
    <Proses A>  
Else  
    <Proses B>  
End If
```

Contoh 3:

“Jika nilai<60 maka tidak lulus, jika tidak lulus” dapat dituliskan dengan:

```
If Nilai<60 Then Ket="Lulus" Else Ket="Tidak Lulus"
```

Contoh 4:

“Jika pembelian > 50000 maka mendapat hadiah cangkir”, dapat dituliskan dengan:

```
If pembelian > 50000 Then Hadiah = "Cangkir"
```

Contoh 5:

“Jika nilai<41 maka keterangan diisi dengan tidak lulus dan nilai huruf diberi E”, dapat dituliskan dengan:

```

If nilai<41 Then
    Ket = " Tidak Lulus"
    NilaiHuruf = "E"
End If

```

Contoh 6:

“Jika pembelian>500000 maka mendapat potongan 10% dan hadiah berupa tas cantik, jika tidak potongan 5%”, dapat dituliskan dengan:

```

If pembelian > 500000 Then
    Potongan = (10/100) * Pembelian
    Hadiah = "Tas Cantik"
Else
    Potongan = (5/100) * Pembelian
End If

```

Contoh Aplikasi 4.2:

Sebuah toko memberikan bonus sebesar 15% dari total pembelian setelah dikurangi 200000 dan hadiah berupa gelas cantik bagi pelanggannya untuk pembelian di 200000. Untuk membuat aplikasi tersebut lakukan langkah-langkah berikut:

- (1) Buat project baru
- (2) Untuk form1, isikan property [Name] dengan formLatihan32.
- (3) Tambahkan komponen-komponen: 3 label, 3 textbox dan 1 command button.
- (4) Untuk label1, isi property [Caption] dengan “Jumlah pembelian”. Untuk label2, isi property [Caption] dengan “Bonus”. Dan untuk label3, isi property [Caption] dengan “Hadiah”
- (5) Untuk semua textbox, kosongkan property [Text]. Untuk textbox1, isi property [Name] dengan pembelian. Untuk textbox2, isi property [Name] dengan bonus. Dan untuk textbox3, isi property [Name] dengan hadiah
- (6) Untuk command1, isi property [Caption] dengan “Hitung Bonus”, dan atur posisi setiap komponen seperti gambar 4.2 berikut:



Gambar 4.2. Tampilan contoh 4.2

- (7) Click pada command1, tambahkan program untuk percabangan : “Jika pembelian>200000 maka bonus 15% dari pembelian dan hadiah berupa gelas cantik”

```
Private Sub Command1_Click()
    jmlBeli = Val(pembelian)
    bonus = 0
    hadiah = " "
    If jmlBeli > 200000 Then
        sisa = jmlBeli - 200000
        bonus = (15 / 100) * sisa
        hadiah = "gelas cantik"
    End If
End Sub
```

- (8) Simpan project tersebut dengan nama projectLatihan42. Dan jalankan dengan mencoba beberapa jumlah pembelian di bawah 200000 dan di atas 200000.

4.2. Percabangan dengan If...Then...Else... Berantai

Percabangan tidak harus hanya memiliki satu kondisi, bisa juga percabangan memiliki banyak kondisi dengan banyak proses keputusan, hal ini sering dinamakan dengan percabangan berantai. Percabangan berantai secara umum ada dua macam yang dituliskan dengan:

- (1) Jenis percabangan berantai dengan hanya satu kondisi yang bisa terpenuhi

```
If <kondisi1> Then
    <Keputusan dalam kondisi1>
else
    if <Kondisi2> Then
```

```

        <Keputusan dalam kondisi2>
    else
        if <Kondisi3> Then
            <Keputusan dalam kondisi3>
        Else
            If <Kondisi4> Then
                <Keputusan dalam kondisi 4>
            Else
                --- Keputusan yang lain ---
            End If
        End If
    End If
End If

```

(2) Jenis percabangan berantai dengan banyak kondisi yang harus dipenuhi.

```

If <kondisi1> Then
    <Keputusan dalam kondisi1>
    if <Kondisi2> Then
        <Keputusan dalam kondisi2>
        if <Kondisi3> Then
            <Keputusan dalam kondisi1, kondisi 2
            dan kondisi 3>
        Else
            <Keputusan dalam kondisi1 dan kondisi2
            tetapi tidak dalam kondisi 3>
        End If
    End If
End If

```

Contoh 7:

Menentukan nilai A,B,C,D,E dengan syarat nilai A>80, nilai B>65 dan kurang dari 81, nilai C>55 dan kurang dari 66, nilai D>40 dan kurang dari 56, dan nilai E kurang dari 41, dapat dituliskan dalam percabangan berantai sebagai berikut:

```

If nilai < 41 Then
    nilaiHuruf = "E"
Else
    If nilai < 56 Then

```

```

        nilaiHuruf = "D"
    Else
        If nilai < 66 Then
            nilaiHuruf = "C"
        Else
            If nilai<81 Then
                nilaiHuruf = "B"
            Else
                nilaiHuruf = "A"
            End If
        End If
    End If
End If

```

Contoh 8:

Sebuah perusahaan memberikan hadiah dan bonus pada karyawan yang jam kerjanya di atas 40 jam per-minggu dan tidak pernah terlambat dalam satu bulan. Karyawan yang jumlah jamnya di atas 40 jam per-minggu mendapat bonus sebesar 25.000 per kelebihan jamnya, dan bila dia juga tidak pernah terlambat akan mendapat hadiah berupa 10% dari gaji pokoknya. Percabangan bersyarat untuk kejadian ini dapat dituliskan dengan:

```

    If jumlahJam > 40 Then
        kelebihanJam = jumlahJam - 40
        bonus = 25000 * kelebihanJam
        If terlambat = 0 Then
            hadiah = (10/100) * gajiPokok
        End If
    End If
End If

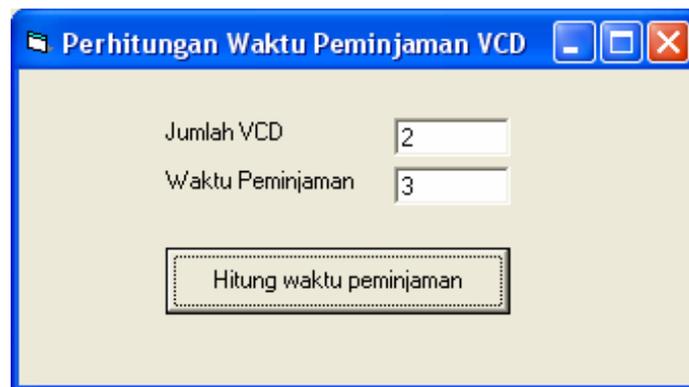
```

Contoh Aplikasi 4.3.

Sebuah rental VCD memberikan persyaratan waktu peminjaman. Jika jumlah VCD yang dipinjam 1 buah maka waktu peminjamannya 2 hari, jika 2 VCD, waktu peminjamannya 3 hari, jika 3 VCD waktu peminjamannya 4 hari, jika 4 VCD waktu peminjamannya 5 hari, jika 5 VCD waktu peminjamannya 6 hari dan jika lebih dari 4

VCD waktu peminjamannya 7 hari. Langkah-langkah pembuatannya adalah sebagai berikut:

- (1) Buat project baru
- (2) Pada form, isi property [Name] dengan formLatihan43 dan isi property [Caption] dengan Perhitungan Waktu Peminjaman VCD
- (3) Tambahkan komponen-komponen: 2 label, 2 textbox dan 1 command button.
- (4) Untuk label1, isi property [Caption] dengan Jumlah VCD. Untuk label2, isi property [Caption] dengan Waktu peminjaman.
- (5) Untuk semua textbox, kosongkan property [Text]. Untuk textbox1 isi property [Name] dengan jumlahVCD dan untuk textbox2 isi property [Name] dengan waktuPinjam.
- (6) Untuk command1, isi property [Caption] dengan Hitung waktu peminjaman. Atur posisi seperti gambar 4.3 berikut.



Gambar 4.3. Tampilan contoh 4.3

- (7) Click pada command1, tambahkan program untuk menghitung waktu peminjaman dari jumlah VCD yang dipinjam sebagai berikut:

```
Private Sub Command1_Click()  
    jmlVCD = Val(jumlahVCD)  
    If jmlVCD = 1 Then  
        waktuPinjam = 2  
    Else  
        If jmlVCD = 2 Then  
            waktuPinjam = 3  
        Else  
            If jmlVCD = 3 Then  
                waktuPinjam = 4  
            Else
```

```

        If jmlVCD = 4 Then
            waktuPinjam = 5
        Else
            If jmlVCD = 5 Then
                waktuPinjam = 6
            Else
                waktuPinjam = 7
            End If
        End If
    End If
End If
End Sub

```

Pemakaian If Then Else untuk persoalan ini sebenarnya tidak cocok, karena programnya akan panjang. Sebaiknya untuk persoalan ini digunakan perintah Select..Case yang akan dijelaskan pada sub bab berikutnya.

- (8) Simpan project dengan nama projectLatihan43. Jalankan dengan mengganti jumlah VCD yang dipinjam untuk menguji apakah program tersebut sudah benar.

4.3. Pemilihan dengan Select Case

Salah satu model percabangan adalah percabangan index atau yang dikenal dengan pemilihan berindex, dalam visual Basic percabangan index ini menggunakan Select Case dengan format:

```

Select Case <variable>
Case 0:
    <Proses untuk keadaan variabel=0>
Case 1:
    <Proses untuk keadaan variabel=1>
Case 2:
    <Proses untuk keadaan variabel=2>
Case 3:
    <Proses untuk keadaan variabel=3>
Case 4:
    <Proses untuk keadaan variabel=4>
.....
End Select

```

Contoh 9:

Departemen di sebuah perusahaan dikodekan dengan nomor urut sebagai berikut:

- 1: Akuntansi
- 2: Marketing
- 3: Gudang
- 4: Kasir

Dengan memasukkan kode departemen, akan ditampilkan nama departemen. Hal ini dapat dilakukan dengan:

```
Select Case kodeDepartemen
```

```
Case 1:
```

```
namaDepartemen = "Akuntansi"
```

```
Case 2:
```

```
namaDepartemen = "Marketing"
```

```
Case 3:
```

```
namaDepartemen = "Gudang"
```

```
Case 4:
```

```
namaDepartemen = "Kasir"
```

```
End Select
```

Contoh 10:

Dalam pembuatan program, dibuat menu dengan pilihan berupa nomor proses sebagai berikut

- 1: Entry Data
- 2: Cetak Laporan
- 3: Keluar

Dimana untuk masing-masing program entry data dan cetak laporan dibangun dari form dengan nama formEntry dan formLaporan. Maka programnya adalah:

```
Select Case pilihan
```

```
Case 1:
```

```
formEntry.Show
```

```
Case 2:
```

```
formLaporan.Show
```

```
Case 3:
```

```
Unload Me
```

```
End Select
```

Perintah `formEntry.Show` berarti menjalankan dan menampilkan form dengan nama `formEntry` pada project yang dibuat. Perintah `Unload Me` berarti keluar dari form yang sedang aktif.

Contoh Aplikasi 4.4.

Seorang pelanggan di sebuah restoran tinggal memilih menu makanan berdasarkan nomor dari menu makanan yang sudah disediakan dan akan ditampilkan nama menu beserta harganya seperti menu makanan berikut:

Nomor	Nama Menu	Harga
1	Soto Ayam	Rp. 4000,-
2	Soto Daging	Rp. 5000,-
3	Sate Ayam	Rp. 6000,-
4	Bali Ayam+Telor	Rp. 5000,-
5	Rawon	Rp. 5000,-

Untuk membuat aplikasi semacam itu, lakukan langkah-langkah sebagai berikut:

- (1) Buat project baru
- (2) Pada form, isi property [Name] dengan `formLatihan44`
- (3) Tambahkan komponen-komponen: 3 label, 3 textbox dan 1 command button
- (4) Untuk `label1`, isi property [Caption] dengan Nomor Menu. Untuk `label2` isi property [Caption] dengan Nama Menu. Dan untuk `label3` isi property [Caption] dengan Harga.
- (5) Untuk semua textbox, kosongkan property [Text]. Untuk `textbox2`, isi property [Name] dengan `namaMenu`. Untuk `textbox3`, isi property [Name] dengan `harga`
- (6) Untuk `command1`, isi property [Caption] dengan Lihat Menu. Atur tampilannya seperti gambar 3.4 berikut.



Gambar 4.4. Tampilan contoh 4.4

- (7) Click pada command1, tambahkan program seleksi menu dengan Select Case sebagai berikut:

```
Private Sub Command1_Click()  
    nomor = Val(Text1)  
    Select Case nomor  
    Case 1:  
        namaMenu = "Soto Ayam"  
        harga = 4000  
    Case 2:  
        namaMenu = "Soto Daging"  
        harga = 5000  
    Case 3:  
        namaMenu = "Sate Ayam"  
        harga = 6000  
    Case 4:  
        namaMenu = "Bali Ayam+Telor"  
        harga = 5000  
    Case 5:  
        namaMenu = "Rawon"  
        harga = 5000  
    End Select  
End Sub
```

- (8) Simpan project dengan nama projectLatihan44. Jalankan dengan mengganti-ganti nomor menu untuk menguji apakah program sudah benar atau belum.

4.4. Latihan

- (1) Buatlah program untuk menentukan apakah bilangan yang dimasukkan adalah bilangan kelipatan tiga atau bukan.
- (2) Buatlah program untuk menentukan apakah bilangan yang dimasukkan adalah bilangan kelipatan dua, tiga dan lima atau bukan. Misalkan 20 adalah kelipatan 2 dan 5, 12 adalah kelipatan 2 dan 3, 60 adalah kelipatan 2, 3 dan 5, 125 adalah kelipatan 5, 11 bukan kelipatan ketiganya.
- (3) Buatlah program untuk menghitung akar persamaan kuadrat (x_1 dan x_2) dengan memasukkan nilai a,b dan c, menggunakan rumus ABC sebagai berikut:

$$d = b^2 - 4ac$$

Jika $d < 0$ maka jawabnya adalah akar imajiner, dan jika $d > 0$ gunakan rumus berikut untuk menghitung x_1 dan x_2 :

$$x_1 = \frac{-b + \sqrt{d}}{2a} \quad \text{dan} \quad x_2 = \frac{-b - \sqrt{d}}{2a}$$

- (4) Buatlah program untuk mengurutkan 3 bilangan yang dimasukkan dengan If..Then..Else
- (5) Buatlah program untuk mengurutkan 4 bilangan yang dimasukkan dengan If..Then..Else
- (6) PDAM akan menarik tagihan berdasarkan pemakaian air pelanggan. Pada 30 m³ pertama biaya pemakaian adalah Rp. 120/m³. Jika pemakaian lebih dari 30 m³, maka biaya kelebihan untuk 30 m³ pertama adalah Rp. 150/m³, 30 m³ kedua adalah Rp. 190/m³ dan 30 m³ ketiga dan seterusnya biaya pemakaiannya Rp. 240/m³.

Sebagai contoh pelanggan A pemakaiannya adalah 75 m³, biayanya adalah:

$$30 \times 120 = 3600$$

$$30 \times 150 = 4500$$

$$15 \times 190 = 2850$$

Total biayanya adalah $3600 + 4500 + 2850 = 10950$

- (7) Buatlah program untuk menentukan nama bulan berdasarkan nomor bulan yang dimasukkan dengan If...Then...Else... berantai dan dengan Select Case.
- (8) Seorang pelanggan akan membeli product pada sebuah toko. Toko tersebut akan memberikan diskon pada pelanggan dengan ketentuan:

Diskon 1: Pembelian antara 100000 sampai dengan 250000, setiap kelebihan dari 100000 mendapat diskon 15%

Diskon 2: Pembelian lebih 250000 sampai dengan 500000, setiap kelebihan dari 100000 mendapat diskon 12%

Diskon 3: Pembelian lebih dari 500000, setiap kelebihan dari 500000 mendapat diskon 10%

Sebagai contoh si Fulan membeli setotal 300000, maka diskonnya adalah:

$$(300000 - 100000) * 12/100 = 24000$$

(9) Sebuah rental VCD menentukan jenis VCD sebagai berikut:

1: VCD Action

2: VCD Komedi

3: VCD Horor

Buatlah program untuk dapat menentukan jenis VCD bila dimasukkan nomor dari jenis VCD tersebut dengan Select Case

5.1. Pengulangan Dengan For...Next

Pengulangan proses menggunakan For...Next adalah bentuk pengulangan terkendali dengan variabel kendali yang terus berjalan maju atau mundur. Format penulisan pengulangan ini adalah sebagai berikut:

```
For variabel = nilai_awal To nilai_akhir [Step langkah]
    <Proses dalam pengulangan>
Next variable
```

Contoh 1:

Untuk menentukan barisan bilangan 1, 2, 3, ..., 10 maka bilangan mempunyai nilai awal = 1, nilai akhir=10 dan step antar bilangan = 1, sehingga bisa ditulis dengan

```
For bilangan=1 To 10 Step 1
    <Proses untuk barisan bilangan>
Next bilangan
```

Karena step=1, maka secara default tidak perlu dituliskan sehingga perintahnya bias ditulis dengan:

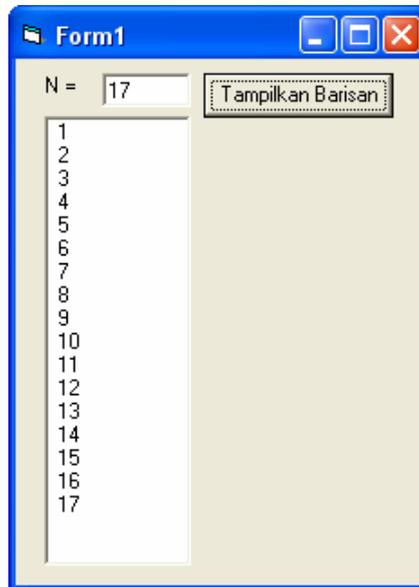
```
For bilangan=1 To 10
    <Proses untuk barisan bilangan>
Next bilangan
```

Contoh Aplikasi 5.1:

Untuk menampilkan barisan bilangan 1 sampai dengan N, dengan N dimasukkan. Hasil barisan ditampilkan pada Label.

- (1) Buat project baru. Pada form, isi property [Name] dengan formLatihan51
- (2) Tambahkan komponen-komponen: 1 label, 1 textbox, 1 buah listBox dan 1 command button

- (3) Untuk label1, isi property [Caption] dengan "N=". Untuk textbox1, kosongkan property [Text]. Untuk command1, isi property [Caption] dengan Barisan. Atur posisi masing-masing komponen seperti gambar 5.1. berikut:



Gambar 5.1. Tampilan latihan 5.1

- (4) Click pada command1, tambahkan program untuk membuat barisan bilangan dari 1 sampai dengan N sebagai berikut:

```
Private Sub Command1_Click()  
    N = Val(Text1)  
    List1.Clear  
    For i = 1 To N  
        List1.AddItem i  
    Next i  
End Sub
```

Fungsi Val digunakan untuk mengubah tipe data text menjadi tipe data numerik, sedangkan fungsi Str digunakan untuk mengubah tipe data numerik menjadi tipe data text.

Property clear pada List1 digunakan untuk membersihkan isi list1, dan property AddItem pada List1 digunakan untuk menambahkan isi list1. Dengan demikian sebelumnya dibersihkan, kemudian ditambah isinya dengan barisan bilangan yang dibuat.

- (5) Simpan project dengan nama projectLatihan51. jalankan dan coba isi N yang berbeda-beda, perhatikan hasilnya.

Contoh 2:

Menampilkan barisan bilangan 20, 19, 18, ..., 3, 2, 1 dapat dilakukan dengan:

```
For bilangan=20 To 1 Step -1
    List1.AddItem bilangan
Next bilangan
```

Contoh 3:

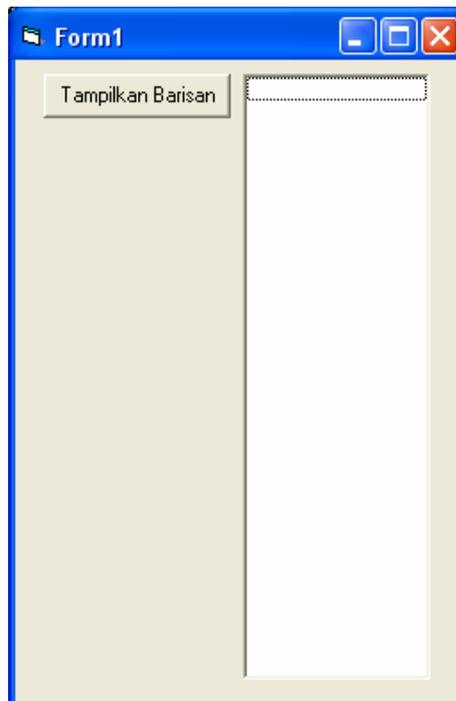
Menampilkan barisan bilangan 1, 4, 7, 10, 13, 16, 19, 22, $3N+1$ dapat dilakukan dengan:

```
For bilangan=0 To N Step -1
    List1.AddItem 3*bilangan+1
Next bilangan
```

Contoh Aplikasi 5.2:

Untuk menampilkan huruf A sampai dengan Z menggunakan pengulangan For..Next, dapat dilakukan dengan langkah-langkah sebagai berikut:

- (1) Buat project baru. Pada form, isi property [Name] dengan formLatihan52
- (2) Tambahkan komponen-komponen: 1 buah listBox dan 1 command button
- (3) Untuk command1, isi property [Caption] dengan Alfabet. Atur tampilannya seperti gambar 5.2 berikut.



Gambar 5.2. Tampilan latihan 5.2

- (4) Click pada command1, tambahkan kode untuk menampilkan barisan alfabet menggunakan For..Next. Teknik yang digunakan adalah menggunakan kode ASCII untuk masing-masing karakter, dimana kode ASCII untuk 'A' adalah 65, dan demikian seterusnya kode ASCII ditambah satu-persatu sampai 26 kali.

```
Private Sub Command1_Click()  
List1.Clear  
For i = 1 To 26  
    kodeASCII = 64 + i  
    List1.AddItem Chr(kodeASCII)  
Next i  
End Sub
```

Fungsi CHR adalah fungsi untuk mengambil karakter dari kode ASCII.

- (5) Simpan project dengan nama projectLatihan52. Jalnkan untuk mengetahui apakah hasilnya sudah benar atru tidak.

Contoh 4:

Menampilkan barisan fibbonanci sampai N kali (N bilangan bulat positif) dimana nilai saat ini adalah jumlah dari nilai dua suku sebelumnya. Contoh barisan fibbonanci: 1 1 2 3 5 8 13 21 34 ...

```
sukul=1  
suku2=1  
List1.AddItem suku1  
List1.AddItem suku2  
For i=3 To N  
    bil=sukul+suku2  
    List1.AddItem bil  
    suku1=suku2  
    suku2=bil  
Next i
```

Contoh 5:

Menampilkan barisan bilangan kelipatan 3 yang lebih kecil dari 100 dan berhenti saat bilangan tersebut juga kelipatan 7. Untuk menghentikan pengulangan For..Next sebelum mencapai nilai akhir dapat dilakukan dengan perintah **Exit For**.

```

For i=3 To 100 Step 3
    If i mod 7 = 0 then Exit For
Next i

```

Contoh 6:

Menampilkan barisan N bilangan yang berganti-ganti 1 dan -1 seperti :

1 -1 1 -1 1 -1 1 -1

adalah sebagai berikut:

```

bilangan = -1;
List1.Clear
For i=1 To N
    bilangan = -1 * bilangan
    List1.AddItem bilangan
Next i

```

Atau dengan cara:

```

List1.Clear
For i=1 To N
    If i mod 2 = 0 then bilangan = -1 else bilangan = 1
    List1.AddItem bilangan
Next i

```

Contoh 7:

Menampilkan barisan N bilangan yang berganti-ganti positif dan negatif seperti :

1 -2 3 -4 5 -6 7 -N

dapat dilakukan dengan cara mengalikan variable kontrol dari looping (i) dan bilangan sebagai berikut:

```

List1.Clear
For i=1 To N
    If i mod 2 = 0 then bilangan = -1 else bilangan = 1
    List1.AddItem i*bilangan
Next i

```

5.2. Pengulangan Dengan While...Wend

Pengulangan dengan While... Wend adalah bentuk pengulangan terkendali dengan proses akan diulang selama kondisi yang diberikan masih dipenuhi. Format penulisan pengulangan ini adalah:

```
While <Kondisi>
    <Proses selama kondisi dipenuhi>
Wend
```

Variabel kontrol pada jenis looping tidak otomatis bertambah/berkurang seperti pada looping dengan For...Next, sehingga penambahan atau pengurangan nilai dilakukan dengan menggunakan akumulator yang dituliskan dengan

var = var + step

Meskipun hal ini bukan suatu keharusan. Karena tidak setiap looping membutuhkan akumulator, variabel yang terus-menerus bertambah atau berkurang.

Contoh 8:

Menampilkan bilangan bulat ganjil dari satu sampai dengan 11, ini merupakan looping dengan akumulator yang dapat dituliskan sebagai berikut:

```
bilangan=1
While bilangan<=11
    List1.AddItem bilangan
    bilangan = bilangan + 2
Wend
```

Contoh 9:

Mencari nilai fungsi $f(x) = -x^2 + 3x - 2$ dari $x=-1$ sampai dengan $f(x)$ mendekati nol, atau dalam arti $|f(x)| < 0.001$, dengan step 0.01, dapat dituliskan dengan:

```
x = -1
fx = -x ^ 2 + 3 * x - 2
while abs(fx)>0.001
    x=x+0.01
    fx = -x ^ 2 + 3 * x - 2
wend
label1.Caption = fx
```

5.3. Pengulangan Dengan Do Loop Until

Pengulangan dengan Do Loop Until adalah bentuk pengulangan terkendali dengan proses akan diulang sampai kondisi yang diberikan dipenuhi. Format penulisan pengulangan ini adalah:

```
Do
    <Proses selama kondisi belum dipenuhi>
Loop Until <Kondisi>
```

Contoh 10:

Menampilkan bilangan bulat ganjil dari satu sampai dengan 11, ini merupakan looping dengan akumulator yang dapat dituliskan sebagai berikut:

```
bilangan=1
do
    List1.AddItem bilangan
    bilangan = bilangan + 2
Loop until bilangan>11
```

Contoh 11:

Mencari nilai fungsi $f(x) = -x^2 + 3x - 2$ dari $x=-1$ sampai dengan $f(x)$ mendekati nol, atau dalam arti $|f(x)| < 0.001$, dengan step 0.01, dapat dituliskan dengan:

```
x = -1
do
    x=x+0.01
    fx = -x ^ 2 + 3 * x - 2
loop until abs(fx) <= 0.001
label1.Caption = fx
```

5.4. Pengulangan Bertingkat

Pada contoh-contoh di atas, hanya dilakukan satu pengulangan. Padahal dalam banyak aplikasi pemrograman menggunakan pengulangan yang lebih dari satu misalkan pada pengolahan matrik, diperlukan pengulangan untuk baris dan pengulangan untuk kolom pada setiap baris. Pengulangan yang adalah di dalam pengulangan inilah yang dinamakan pengulangan bertingkat.

Pengulangan bertingkat secara umum dapat dituliskan dengan bentuk penulisan sebagai berikut:

```
Pengulangan 1
.....
Proses di dalam pengulangan 1
Pengulangan 2
.....
Proses dalam pengulangan 2
.....
Akhir pengulangan 2
.....
Akhir pengulangan 1
```

Contoh 12.

Menampilkan n baris dari barisan bilangan 1 2 3 4 5 sebagai berikut:

```
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
.....
```

Dapat ditulis dengan:

```
For baris = 1 To N
    Kal=""
    For i = 1 To 5
        Kal = kal & str(i)
    Next i
    List1.AddItem kal
Next baris
```

Contoh 13.

Menampilkan 5 baris dari kombinasi barisan bilangan 1 2 3 4 5 sebagai berikut:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Perhatikan bahwa pada baris ke 1 hasilnya adalah dari 1 sampai dengan 1, pada baris ke 2 hasilnya adalah dari 1 sampai dengan 2, pada baris ke 3 hasilnya adalah dari 1 sampai dengan 3. Sehingga dapat dikatakan untuk baris ke i hasilnya adalah dari 1 sampai dengan i. Dan dapat dituliskan sebagai berikut:

```

For i = 1 To 5
    kal = ""
    For j=1 To i
        kal = kal & str(j)
    Next j
    List1.AddItem kal
Next i

```

Contoh 12.

Menampilkan n baris dari barisan bilangan 1 2 3 4 5 sebagai berikut:

```

1 2 3 4 5
5 4 3 2 1
1 2 3 4 5
5 4 3 2 1
.....

```

Perhatikan bahwa pada baris ganjil hasilnya 1 2 3 4 5 dan pada baris yang genap hasilnya adalah 5 4 3 2 1, maka perlu untuk mengetahui apakah baris itu baris genap/ganjil untuk menentukan nilai awal, nilai akhir dan step. Dapat ditulis dengan:

```

For baris = 1 To N
    If baris mod 2 =0 Then
        Awal = 1 : Akhir = 5
        Langkah = 1
    Else
        Awal = 5 : Akhir = 1
        Langkah = -1
    End If
    Kal=""
    For i = awal To akhir Step langkah
        Kal = kal & str(i)
    Next i
    List1.AddItem kal
Next baris

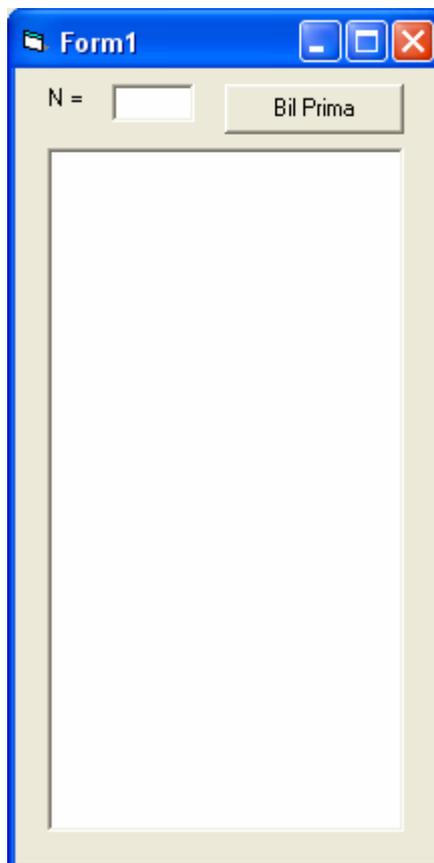
```

Contoh Aplikasi 5.3:

Membuat program untuk menghitung N bilangan prima, definisi bilangan prima adalah bilangan yang hanya habis dibagi oleh 1 dan bilangan itu sendiri atau bisa dikatakan bahwa bilangan prima adalah bilangan yang tidak habis dibagi oleh bilangan 2 sampai sebelum bilangan itu sendiri.

- (1) Buat project baru dengan StandardEXE
- (2) Pada form1, isi property [Name] dengan formLatihan53.
- (3) Tambahkan komponen-komponen: 1 buah label, 1 textbox, 1 listbox dan 1 command button.
- (4) Isi property masing-masing komponen seperti tabel di bawah ini dan atur posisinya seperti gambar 5.3 berikut

Komponen	Property	Nilai
Label1	Caption	N =
Text1	Text	(kosong)
Command1	Caption	Bil Prima



Gambar 5.3. Tampilan latihan 5.3

(5) Click pada command1, tambahkan program untuk membangkitkan N bilangan prima pertama.

```
Private Sub Command1_Click()  
n = Val(Text1)  
List1.Clear  
k = 0  
bil = 2  
While k < n  
    prima = True  
    For pembagi = 2 To bil - 1  
        If bil Mod pembagi = 0 Then prima = False  
    Next pembagi  
    If prima Then  
        List1.AddItem bil  
        k = k + 1  
    End If  
    bil = bil + 1  
Wend  
End Sub
```

(6) Simpan project dalam projectLatihan53.

Contoh 13:

Menampilkan hasil penjumlahan dari deret berikut:

$$1 + 2 + 3 + 4 + \dots + N$$

dapat dilakukan dengan:

```
s=0  
For i=1 To n  
    s = s + i  
next i
```

Penulisan $s=s+i$ dinamakan dengan akumulator, yaitu suatu variabel yang terus menerus bertambah sesuai data. Akumulator ini banyak digunakan dalam penulisan kode program dari rumus-rumus yang menggunakan jumlahan (sigma) sebagai berikut:

$$\sum_x f(x)$$

Contoh 14:

Menampilkan hasil penjumlahan dari rumus deret yang dituliskan dengan

$$s = \sum_{x=1}^n (x-1)^2$$

dapat dilakukan dengan:

```
s=0
For x=1 To n
    s = s + (x - 1) ^ 2
next x
```

5.5. Latihan

1. Buatlah program untuk menampilkan barisan N buah bilangan yang dimulai dengan 1, bilangan kedua berselisih 1 dari bilangan pertama, bilangan ketiga berselisih 2 dari bilangan kedua, ... bilangan ke n berselisih (n-1) dari bilangan ke n-1 sebagai berikut:

1 2 4 7 11 16 22 ...

2. Sebuah obyek berjalan dari posisi x=0 sampai dengan x=10 dia kembali ke x=0, kemudian di kembali x=10, dan demikian seterusnya sampai dia berputar (dari x=0 kebalik ke x=0) sebanyak N kali. Buatlah program untuk menampilkan posisi obyek tersebut.
3. Buatlah program untuk menampilkan barisan N bilangan sebagai berikut:

1 -2 3 -4 5 -1 2 -3 4 -5 1 -2 3

4. Buatlah program untuk menampilkan barisan N buah bilangan biner sebagai berikut:

1 0 1 0 1 0 1 0 ...

5. Buatlah program untuk menampilkan barisan N buah bilangan biner sebagai berikut:

1 0 0 0 1 0 0 0 1 0 0 0 ...

6. Buatlah program untuk menampilkan barisan N buah bilangan biner sebagai berikut:

1 1 0 0 1 1 0 0 1 1 0 0 ...

7. Buatlah program untuk menampilkan nilai x yang berjalan dari 0 sampai 1 dengan step 0.1, dan $y = x^2 - 3x + 2$.

8. Buatlah program menampilkan n baris dari barisan bilangan sebagai berikut

```
1  2  3  4  5  6  7  8
2  3  4  5  6  7  8  1
3  4  5  6  7  8  1  2
4  5  6  7  8  1  2  3
.....
```

9. Buatlah program untuk menentukan dimana x dapat menyebabkan nilai fungsi $f(x) = x^3 - 2x^2 + x - 1$ menjadi $|f(x)| < 0.001$, dengan x berjalan dari -4 sampai dengan 4 dengan step 0.01.

10. Buatlah program untuk menghitung rumus deret sebagai berikut:

$$s = \sum_{x=0}^{10} (x - 5)^2$$

11. Buatlah program untuk menghitung rumus deret sebagai berikut:

$$s = \sum_{y=0}^{10} \sum_{x=0}^{10} (-1)^{x+y}$$

6.1. Konsep Array

Array dalam bahasa Indonesia adalah indeks, maka variabel array adalah variabel yang mempunyai indeks. Sehingga penulisannya adalah *var(indeks)*. Array sangat penting di dalam pemrograman, karena array mampu menampung banyak data yang bertipe sama. Sebagai contoh variabel array x menampung nilai-nilai bilangan bulat {3, 6, 8, 7, 5, 1} berarti indeks untuk variabel x ini adalah 1 sampai dengan 6, dan ditulis sebagai x(1), x(2), x(3), x(4), x(5) dan x(6).

Array dapat diibaratkan sebagai kereta api dalam komputer, yang di dalamnya terdapat gerbong-gerbong memory yang berisi data-data yang mempunyai tipe dan perlakuan yang sama. Untuk mengambil atau menampilkan nilai array dapat dianalogikan dengan mencari kursi dalam gerbong kereta api, maka pada karcis perlu dituliskan nomor gerbong, Array juga demikian, untuk menampilkan nilai array tinggal menyebutkan indeks-nya. Misalkan untuk menampilkan nilai variabel x yang ke 5 dituliskan dengan x(5).

Untuk dapat membuat variabel array maka terlebih dahulu harus didefinisikan nama variabel array dan berapa jumlah maksimalnya dengan cara:

```
Dim nama_variabel(jumlah_array) as Tipe_Data
```

Perintah ini diletakkan sesuai kebutuhan apakah array ini hanya untuk subroutine (event) lokal atau pada seluruh event di form.

Sebagai contoh untuk membuat array bilangan bulat yang dapat menampung 10 bilangan dapat dituliskan dengan:

```
Dim bilangan(10) as integer
```

Bila jumlah array sudah ditentukan 10 maka tidak boleh menggunakan data lebih dari 10, karena 10 menyatakan jumlah maksimum dari data yang akan ditampung adalah suatu array. Sebagai contoh:

Bilangan(4) = 100 → Perintah ini benar

Bilangan(12) = 8 → Perintah ini salah karena batasnya sudah ditentukan 10.

Contoh 1:

Memasukkan 6 nama sebagai berikut “Basuki”, “Achmad”, “Rizki”, “Widya”, “Dian”, dan “Teguh” ke dalam array dan menampilkan semua nama ke dalam List saat form di panggil.

```
Dim nama(6) as String

Private Sub Form_Load()
nama(0) = "Basuki"
nama(1) = "Achmad"
nama(2) = "Rizki"
nama(3) = "Widya"
nama(4) = "Dian"
nama(5) = "Teguh"
List1.Clear
For i=0 To 5
    List1.AddItem nama(i)
Next i
End Sub
```

*Array pada Visual Basic dimulai dari indeks 0. Sehingga data pertama dituliskan dengan **var(0)**.*

Visual Basic juga mengenal array dinamis, dimana jumlahnya bisa tak terbatas. Untuk mendefinisikan array dinamis dapat dilakukan dengan mendefinisikan array tanpa menuliskan jumlah maksimum arraynya sebagai berikut:

Dim variabel() as tipe_data

Untuk contoh nama di atas, dapat dituliskan dengan:

Dim nama() as string

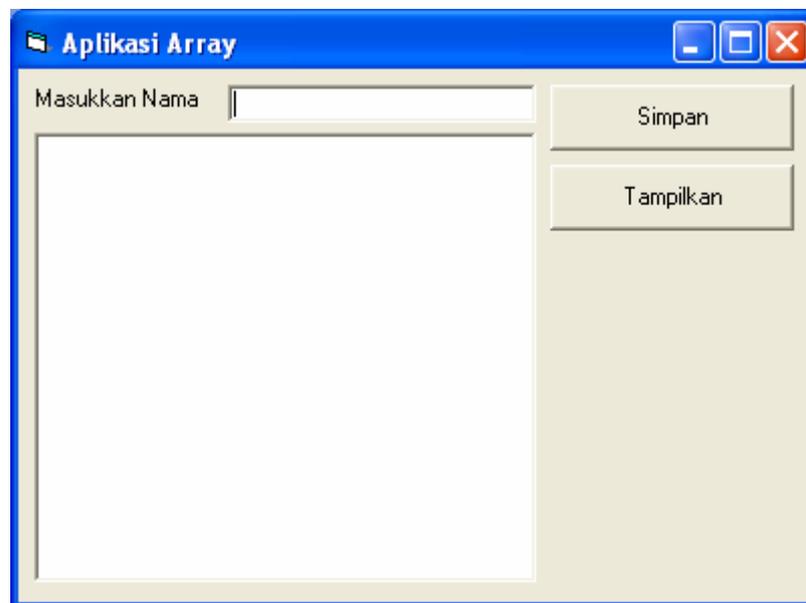
Contoh Aplikasi 6.1:

Membuat project untuk menuliskan data-data nama siswa dengan nomor dibuat otomatis, dengan jumlah data siswa maksimum adalah 20. Kemudian menampilkan semua data nama siswa yang sudah dimasukkan.

- (1) Buat project baru dengan StandardEXE
- (2) Tambahkan komponen-komponen: 1 label, 1 textbox, 1 listbox dan 2 command button

- (3) Masukkan nilai property dari masing-masing komponen dan form seperti tabel di bawah ini, dan atur penempatan setiap komponen seperti gambar 6.1.

Komponen	Property	Nilai
Form1	Name	formLatihan61
	Caption	Latihan Array 1
	StartUpPosition	2 – CenterSreen
Label1	Caption	Masukkan Nama
Text1	Text	(Kosong)
Command1	Caption	Simpan
Command2	Caption	Tampilkan



Gambar 6.1. Tampilan latihan 6.1

- (4) Tambahkan kode program pada jendela kode sebagai berikut:

```

` Mendefinisikan variabel global nama sebagai array
` dan jumlah data
Dim nama(20) As String
Dim n As Integer

Private Sub Command1_Click()
` Menambah nomor array dan memindahkan isi text1
` ke array nama
n = n + 1
nama(n) = Trim(Text1.Text)

```

```

` Mengosongkan isi text1 dan menaruh kursor ke text1
Text1.Text = ""
Text1.SetFocus
End Sub

Private Sub Command2_Click()
` Menampilkan semua isi array nama
List1.Clear
For i = 1 To n
    List1.AddItem nama(i)
Next i
End Sub

Private Sub Form_Load()
` Saat form dipanggil jumlah array diisi dengan nol
n = 0
End Sub

```

(5) Simpan project dengan nama projectLatihan61.

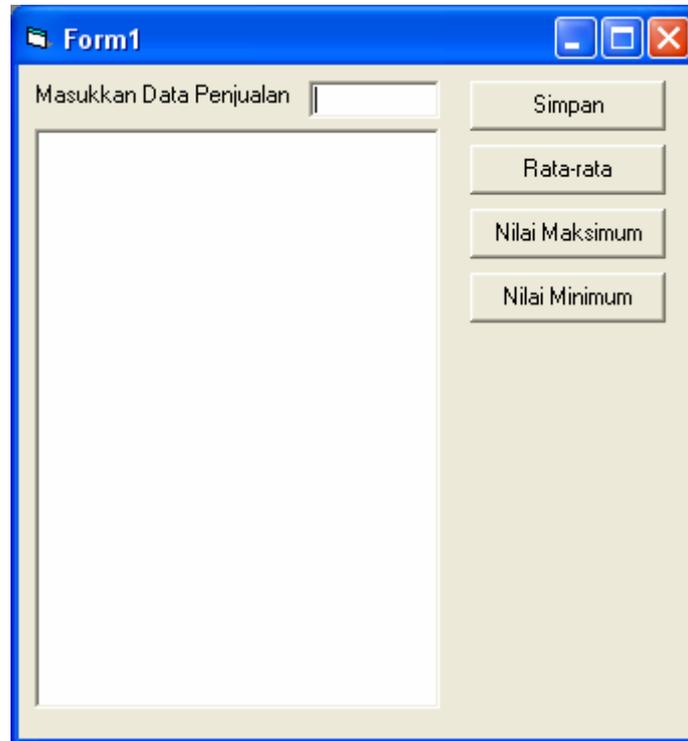
Contoh Aplikasi 6.2:

Membuat project untuk memasukkan data-data penjualan yang berisi jumlah produk penjualan dalam satu bulan. Kemudian dihitung rata-rata penjualan, bulan yang mengalami penjualan tertinggi dan bulan yang mengalami penjualan terendah.

- (1) Buat project baru dengan StandardEXE
- (2) Tambahkan komponen-komponen: 2 buah label, 2 textbox, 1 listbox dan 4 command button
- (3) Isi property dari setiap komponen seperti tabel di bawah ini dan atur posisi tiap komponen seperti gambar 6.2.

Komponen	Property	Nilai
Form1	Name	formLatihan62
	Caption	Latihan Array 2
	StartPosition	2 – CenterScreen
Label1	Caption	Masukkan Nama
Text1	Text	(Kosong)
Command1	Caption	Simpan

Komponen	Property	Nilai
Command2	Caption	Rata-Rata
Command3	Caption	Nilai Maksimal
Command4	Caption	Nilai Minimal



Gambar 6.2. Tampilan latihan 6.2

(4) Pada jendela kode, tambahkan kode program berikut:

```
'Mendefinisikan array data penjualan jual(n)
'Dan jumlah data penjualan n
Dim jual(100) As Single
Dim n As Integer

Private Sub Command1_Click()
'Jumlah data bertambah dan ambil data dari text1
n = n + 1
jual(n) = Val(Text1)
'Tampilkan ke list
List1.AddItem jual(n)
End Sub

Private Sub Command2_Click()
```

```

'Menghitung nilai rata-rata dari data
jumlah = 0
For i = 1 To n
    jumlah = jumlah + jual(i)
Next i
rata2 = jumlah / n
List1.AddItem ""
List1.AddItem "Rata-rata = " & rata2
End Sub

Private Sub Command3_Click()
'Menghitung nilai maksimal dari data
jualMaks = jual(1)
For i = 2 To n
    If jual(i) > jualMaks Then jualMaks = jual(i)
Next
List1.AddItem ""
List1.AddItem "Penjualan maksimum = " & jualMaks
End Sub

Private Sub Command4_Click()
'Menghitung nilai minimal dari data
jualMin = jual(1)
For i = 2 To n
    If jual(i) < jualMin Then jualMin = jual(i)
Next
List1.AddItem ""
List1.AddItem "Penjualan maksimum = " & jualMin
End Sub

Private Sub Form_Load()
'Nilai awal jumlah data diberi nol
n = 0
List1.Clear
End Sub

```

(5) Simpan project dengan nama projectLatihan62.

Contoh 2:

Membuat barisan fibbonanci menggunakan konsep array. Definisi dari barisan fibbonanci adalah suku ke n adalah jumlah dari dua suku sebelumnya (suku ke n-1 dan suku ke n-2) atau dituliskan dengan:

$$x(n) = x(n-1) + x(n-2)$$

Dan kode programnya adalah:

```
x(1) = 1
x(2) = 1
For i=1 To N
    x(i) = x(i-1) + x(i+1)
Next i
` Menampilkan hasilnya
For i=1 To N
    List1.AddItem x(i)
Next i
```

6.2. Array 2 Dimensi

Array bisa menggunakan satu indeks yang disebut dengan satu dimensi seperti yang sudah dijelaskan di atas. Dan juga bisa menggunakan lebih dari satu, misalkan 2 indeks yang dinamakan dengan array 2 dimensi dan ditulis dengan:

variabel(indeks1,indeks2)

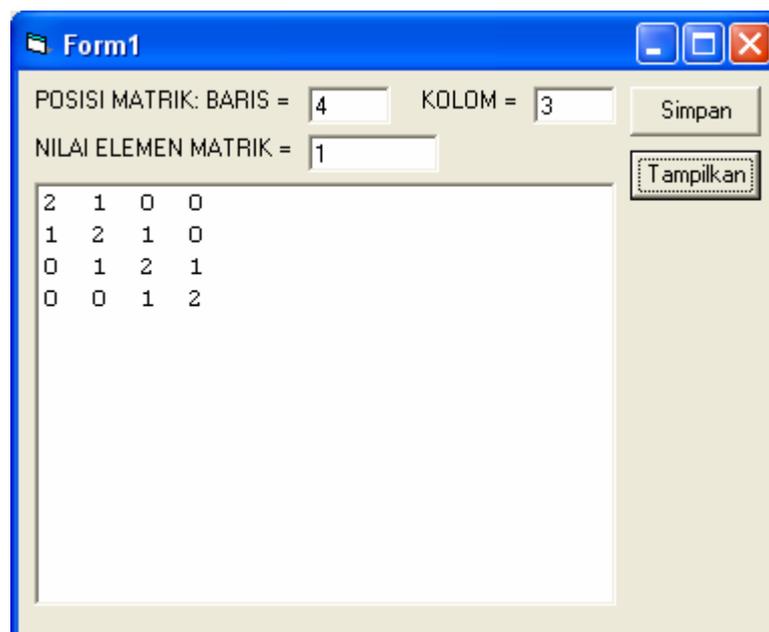
Array 2 dimensi ini biasa digunakan untuk keperluan matrik, dimana indeks pertama menyatakan baris dan indeks kedua menyatakan kolom. Sebagai contoh a(2,3) adalah nilai a pada baris 2 dan kolom 3.

Contoh Aplikasi 6.3:

Membuat program untuk memasukkan matrik dan menampilkan matrik sebagai berikut:

- (1) Buat project baru dengan standard EXE
- (2) Tambahkan komponen-komponen: 1 buah labe, 1 buah textbox, 1 buah listbox, dan 2 buah command button
- (3) Isi property setiap komponen seperti pada tabel d bawah ini, dan atur posisinya seperti gambar 6.3

Komponen	Property	Nilai
Form1	Name	formLatihan63
	Caption	Latihan Array 3
	StartPosition	2 – CenterScreen
Label1	Caption	POSISI MATRIK: BARIS =
Label2	Caption	KOLOM=
Label3	Caption	NILAI ELEMEN MATRIK =
Text1	Text	(Kosong)
Text2	Text	(Kosong)
Text3	Text	(Kosong)
Command1	Caption	Simpan
Command2	Caption	Tampilkan



Gambar 6.3. Tampilan latihan 6.3

(4) Pada jendela code, tambahkan kode program sebagai berikut:

```
'Definisi matrik berukuran 4x4
Dim m(4, 4) As Single

Private Sub Command1_Click()
'Memasukkan nilai matrik sesuai dengan
'baris dan kolom yang dimasukkan
baris = Val(Text1)
```

```

kolom = Val(Text2)
m(baris, kolom) = Val(Text3)
End Sub

Private Sub Command2_Click()
'Menampilkan semua isi matrik
List1.Clear
For i = 1 To 4
    kal = ""
    For j = 1 To 4
        kal = kal & m(i, j) & "  "
    Next j
    List1.AddItem kal
Next i
End Sub

```

(5) Simpan project ini dengan nama projectLatihan63.

Contoh 3:

Membuat penjumlahan dua buah matrik a dan b yang berukuran sama (mxn) menggunakan rumus:

$$c(i, j) = a(i, j) + b(i, j)$$

dapat dilakukan dengan:

```

For baris=1 To m
    For kolom=1 To n
        c(baris,kolom) = a(baris,kolom) + b(baris,kolom)
    Next kolom
Next baris

```

Contoh 4:

Membuat perkalian dua buah matrik a dan b yang berukuran sama (nxn) menggunakan rumus:

$$c(i, j) = \sum_{k=1}^n a(i, k) * b(k, j)$$

dapat dilakukan dengan:

```

For baris=1 To n
    For kolom=1 To n

```

```

z = 0
For i=1 To n
    z = z + a(baris,i) * b(i,kolom)
Next i
c(baris,kolom) = z
Next kolom
Next baris

```

6.3. Control Array

Pada visual Basic, array tidak hanya berlaku untuk variabel tetapi juga berlaku untuk komponen. Pemakaian array pada komponen ini dinamakan dengan **control array**. Control array ini banyak terjadi ketika membuat komponen dengan cara meng-copy dari komponen yang sudah ada, misalkan untuk membuat command button dengan meng-copy command1 akan muncul konfirmasi "*You already have a control named command1. Do you want to create a control array ?*". Bila pertanyaan ini dijawab dengan Yes, maka hasil command yang kedua mempunyai nama command1(1), sedangkan command yang pertama mempunyai nama command1(0). Control array ini sangat berguna bila membuat suatu project dengan banyak komponen yang mempunyai model dan kegunaan yang sama. Control array juga berguna untuk membuat project dengan komponen serupa yang sangat banyak misalkan untuk lampu indikator yang menyatakan volume atau kekuatan.

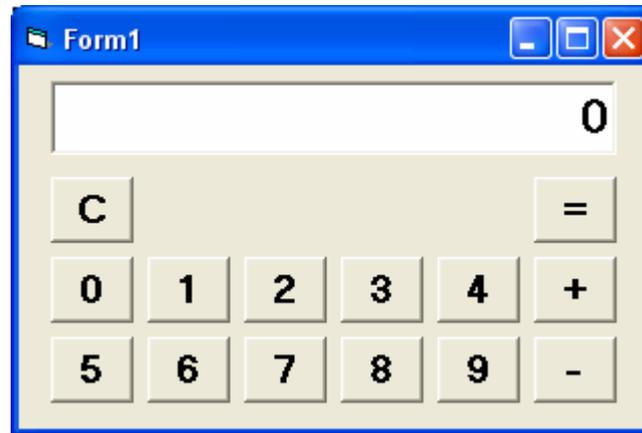
Contoh Aplikasi 6.4:

Membuat kalkulator dengan kemampuan penjumlahan dan pengurangan, dapat dilakukan dengan memanfaatkan control array pada tombol-tombol yang digunakan.

- (1) Buat project baru dengan StandardEXE. Untuk form, isi property [Name] dengan formLatihan64.
- (2) Tambahkan komponen TextBox, atur property [Font] dengan size 18 dan bold, atur property alignment dengan 1-Right Justify. Isi property [Text] dengan 0 (nol).
- (3) Tambahkan command Button, atur property [Font] dengan size 14 dan bold. Buatlah 14 buah command button dengan cara meng-copy dan jadikan control

array sehingga menjadi `command1(0)`, `command1(1)`, `command1(2)` sampai dengan `command1(13)`.

(4) Atur posisinya seperti gambar 6.4 berikut.



Gambar 6.4. Tampilan latihan 6.4 membuat kalkulator

(5) Pada jendela code, tambahkan kode program sebagai operasi kalkulator sebagai berikut:

```
% Definisi variabel global
Dim kal, operasi As String
Dim bil As Integer

Private Sub Command1_Click(Index As Integer)
% Penambahan karakter setiap tombol angka ditekan
kal = kal + Trim(Str(Index))
Text1 = kal
End Sub

Private Sub Command2_Click()
% Membersihkan memory dan text1
Text1 = "0"
kal = ""
End Sub

Private Sub Command3_Click(Index As Integer)
Select Case Index
Case 0:
% Operasi perhitungan bila tombol = ditekan
bil2 = Val(kal)
If operasi = "+" Then bil = bil + bil2
```

```

        If operasi = "-" Then bil = bil - bil2
        Text1 = bil
        kal = Trim(Str(bil))
    Case 1:
        %Penampungan variabel dan operator untuk penjumlahan
        bil = Val(kal)
        operasi = "+"
        kal = ""
    Case 2:
        %Penampungan variabel dan operator untuk pengurangan
        bil = Val(kal)
        operasi = "-"
        kal = ""
    End Select
End Sub

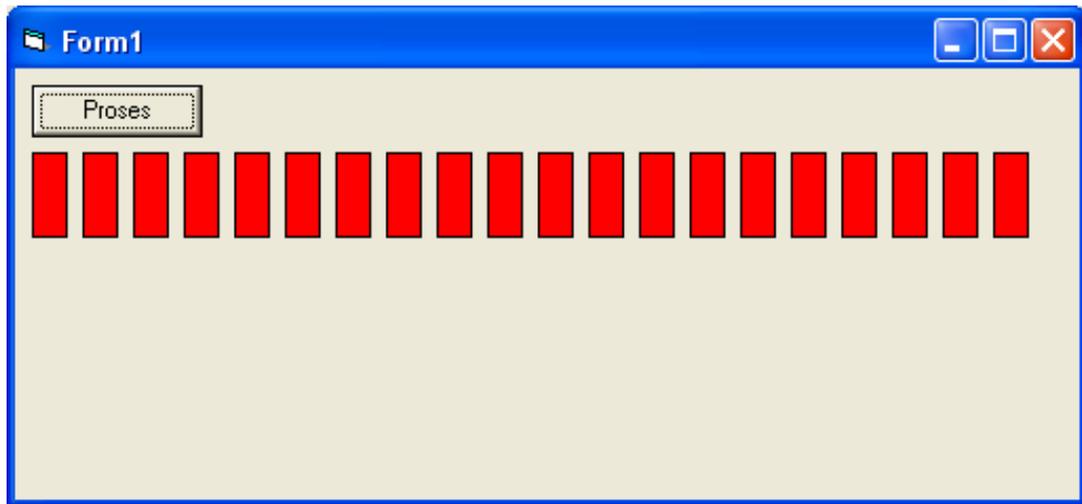
```

- (6) Simpan project dengan nama projectLatihan64. Cobalah lakukan proses perhitungan seperti kalkulator biasa. Kalkulator ini memang masih sederhana, dan bisa ditambah fasilitasnya dengan menambah command button dan setiap eventnya.

Contoh Aplikasi 6.5:

Membuat indikator proses dengan memanfaatkan control array. Lakukan pengulangan 1 sampai dengan 20, setiap pengulangan lakukan pengulangan lagi sebanyak 1000000 dn lakukan proses perhitungan sederhana misalkan $x=2*j+1$. Buatlah indikator yang menunjukkan sudah berapa pengulangan pertama yang selesai dengan indikator dibuat memanfaatkan property [Visible] pada shape. Untu membuat program ini lakukan langkah-langkah sebagai berikut:

- (1) Buat project baru dengan StandardEXE. Untuk form1, isi property [Name] dengan formLatihan65
- (2) Tambahkan komponen Command, isi property [Caption] dengan Proses.
- (3) Tambahkan komponen shape, isi property [FillColor] dengan Pallette Merah, dan isi property [FillStyle] dengan 0-Solid. Copy komponen shape ini sebanyak 19 kali sehingga totalnya adalah 20 shape dan letakkan berjajar, seperti pada gambar 6.5 berikut.



Gambar 6.5. Tampilan latihan 6.5

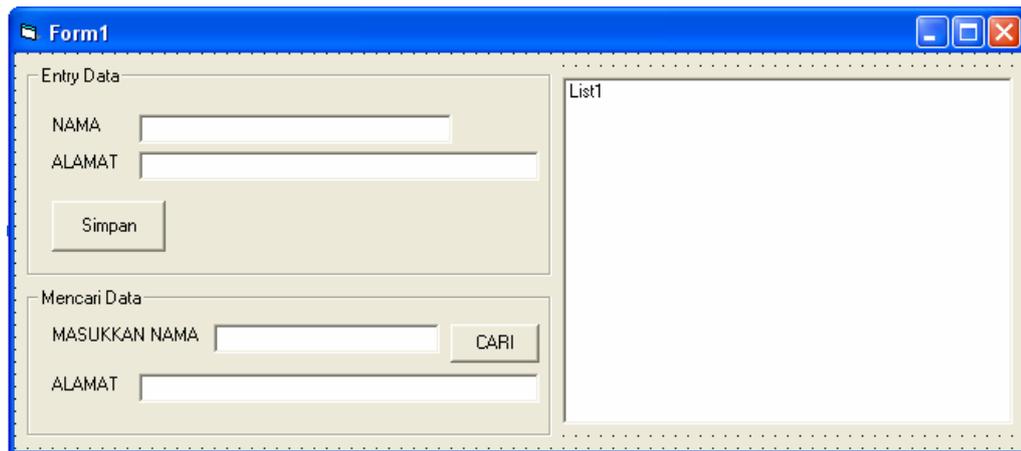
- (4) Pada jendela code, tambahkan kode program untuk menampilkan indikator pengulangan sebagai berikut:

```
Private Sub Command1_Click()
    'Hilangkan semua shape dengan mengatur visible
    For i = 0 To 19
        Shape1(i).Visible = False
    Next i
    'Lakukan pengulangan
    For i = 1 To 20
        'Lakukan pengulangan sebanyak 1000000
        For j = 1 To 1000000
            'Lakukan proses perhitungan sederhana
            x = 2 * i + 1
        Next j
        'Tampilkan shape ke i-1
        Shape1(i - 1).Visible = True
        Shape1(i - 1).Refresh
    Next i
End Sub
```

- (5) Simpan project dengan nama projectLatihan65

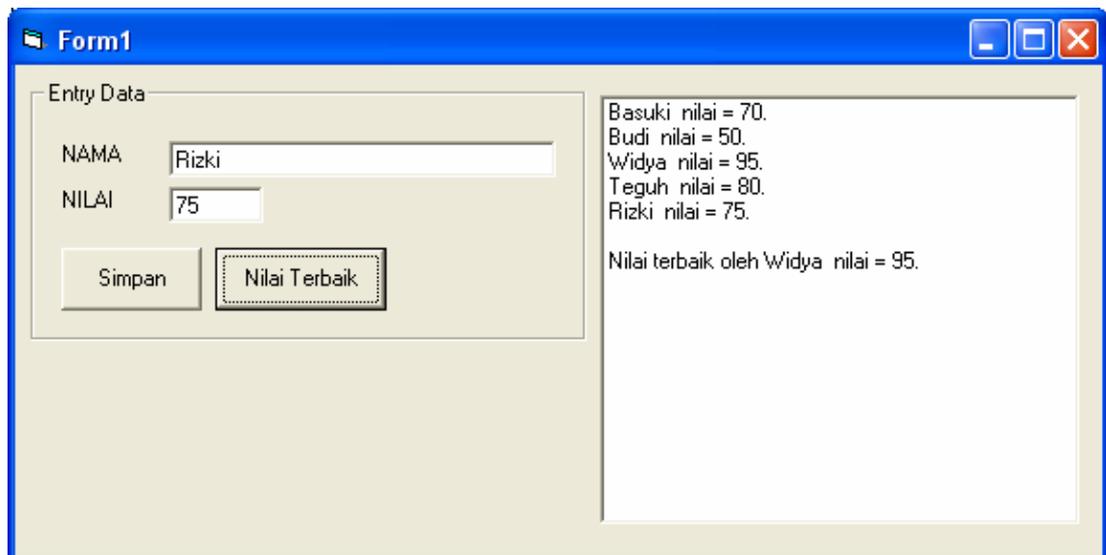
6.4. Latihan

- (1) Buat program untuk memasukkan nama dan alamat menggunakan array, dan ditampilkan di listbox. Masukkan nama dan tekan tombol Cari, tampilkan alamat dari nama yang dipanggil, dengsn tampilan seperti gambar 6.6 berikut:



Gambar 6.6. Contoh tampilan soal 6.1

- (2) Buat program untuk memasukkan nama siswa dan nilai, dan cari nama siswa yang mempunyai nilai terbaik, dengan tampilan seperti gambar 6.7 berikut:



Gambar 6.7. Contoh tampilan soal 6.2

- (3) Buat program untuk memasukkan penjualan bulanan, dan menampilkan seluruh selisih penjualan saat ini dikurangi dengan penjualan bulan yang lalu, yang ditulis dengan:

$$\text{selisih(bulan)} = \text{penjualan(bulan)} - \text{penjualan(bulan-1)}$$

Dan menampilkan selisih penjualan terbesar untuk menunjukkan pada bulan apa terjadi kemajuan penjualan terbesar. Dengan tampilan seperti gambar 6.8 berikut.



Gambar 6.8. Contoh tampilan soal 6.3

- (4) Buat program untuk memasukkan jumlah penjualan komputer dalam bulanan dan mingguan, dengan array 2 dimensi sebagai berikut:

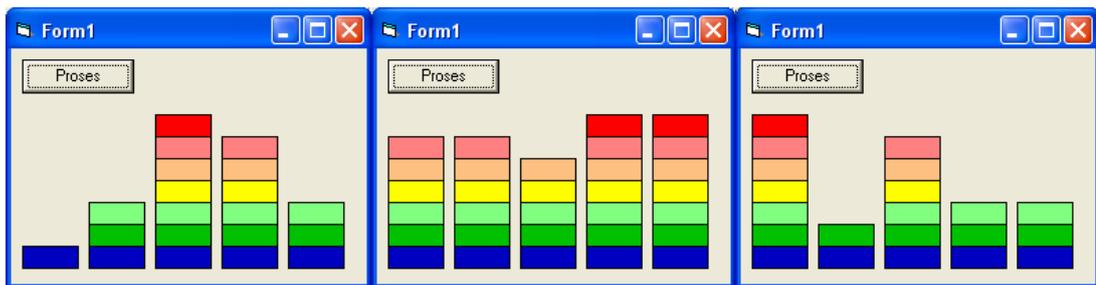
Tampilkan semua jumlah penjualan komputer, dengan tampilan seperti gambar 6.9 berikut:



Gambar 6.9. Contoh tampilan soal 6.4

- (5) Buat program untuk memasukkan jumlah user yang menggunakan internet di sebuah warnet pada setiap harinya, dan menampilkan semua jumlah user per-hari. Tampilkan juga, rata-rata dan pada hari ke berapa usernya yang paling banyak.

- (6) Pada contoh aplikasi 6.4 untuk membuat kalkulator, operatornya hanya penjumlahan (+) dan pengurangan (-), tambahkan operator perkalian (*), operator pembagian (/) dan operator pangkat (^). Selain itu tambahkan tombol titik (.) untuk bisa mengoperasikan bilangan pecahan.
- (7) Buatlah aplikasi yang dapat menunjukkan indikator dari 5 volume (Diulang sebanyak 200 kali), dimana setiap nilai volume berikisar dari 0 sampai dengan 6. Besarnya volume setiap periode berganti secara acak, seperti terlihat pada gambar 6.10 berikut.



Gambar 6.10. Contoh tampilan soal 6.7

7.1. Subroutine

Subroutine atau procedure adalah suatu blok program terpisah yang digunakan untuk mengerjakan suatu pekerjaan tertentu. Kegunaan yang umum dari subroutine adalah menghemat kode program bila terjadi proses yang sama diulang berkali-kali. Salah satu bentuk subroutine di dalam visual basic dikenal dengan event dari komponen. Pada bab-bab sebelumnya sudah banyak digunakan subroutine ini. Penulisan subroutine di dalam Visual Basic dibedakan menjadi 3 macam:

- (1) Subroutine yang bersifat event dari komponen tertentu (subroutine ini sudah tersedia dalam library Visual Basic sebagai modul OOP dalam Visual Basic) dituliskan dengan:

```
Private Sub Komponen_Event(Input)
    <blok program dalam subroutine>
    .....
End Sub
```

- (2) Subroutine yang bersifat metode private dimana pemanggilan subroutinanya hanya ditulis dengan nama subroutine (Subroutine ini adalah subroutine yang dibuat sendiri), dituliskan dengan:

```
Private Sub Nama_Fungsi(Input)
    <blok program dalam subroutine>
    .....
End Sub
```

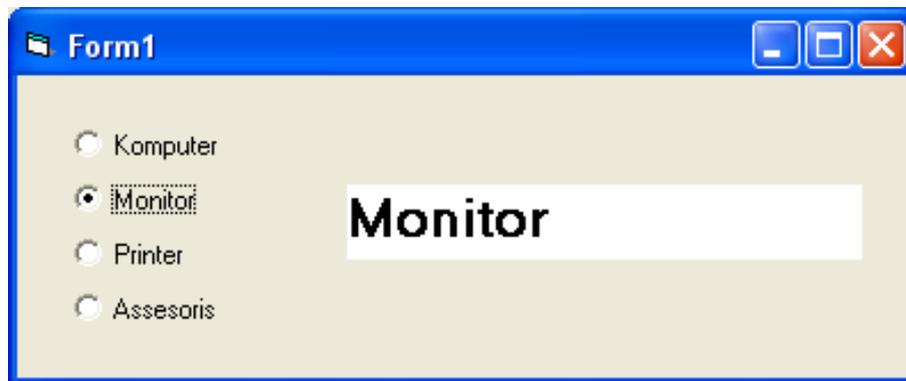
- (3) Subroutine yang bukan event atau metode, dimana pemanggilannya menggunakan **call NamaSubroutine**, dituliskan dengan:

```
Sub Komponen_Event(Input)
    <blok program dalam subroutine>
    .....
End Sub
```

Pemakaian Subroutine semacam ini banyak ditemui pada pemrograman game dalam Visual Basic.

Contoh Aplikasi 7.1:

Membuat subroutine pilihan option yang dipilih ketika option di-click seperti gambar 7.1 berikut:



Gambar 7.1. Tampilan latihan subroutine 7.1

Lakukan langkah-langkah berikut:

- (1) Buat project baru dengan standardEXE. Pada form, isi property [Name] dengan formLatihan71.
- (2) Tambahkan label, isi property [Appereance] dengan 0-Flat, isi property [Font] size dengan 18 bold.
- (3) Tambahkan 4 komponen option menggunakan control array, isi [Caption] dari masing-masing option dengan {"Komputer", "Monitor", "Printer", "Assesoris"}. Atur posisinya seperti gambar 7.1.
- (4) Pada layar code, tambahkan subroutine berupa event ketika option dipilih:

```
Private Sub Option1_Click(Index As Integer)
    Label1 = Option1(Index).Caption
End Sub
```
- (5) Simpan project dengan nama projectLatihan71.

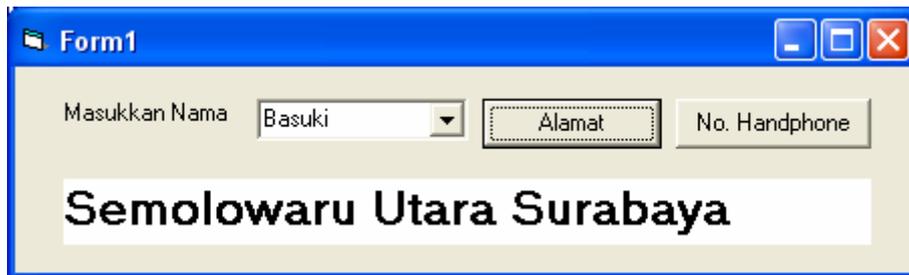
Subroutine selain berupa event yang sudah disediakan oleh Visual Basic, bisa juga berupa subroutine yang dibuat sendiri untuk keperluan khusus. Misalkan untuk menampilkan hasil proses ke dalam listbox yang merupakan hasil dari proses yang berulang.

Contoh Aplikasi 7.2:

Membuat program untuk menyimpan nama, alamat dan nomor handphone dalam array saat Form dipanggil,

nama={"Basuki", "Widya", "Teguh", "Dian"},

alamat={“Semolowaru Utara Surabaya”, “Semolowaru Elok Surabaya”, “Mulyosari Selatan Surabaya”, “Kompleks Laguna Surabaya”}, dan
 handphone={“081331344000”, ”0315912190”, “0315997878”, “08111562145”}
 Bila dipilih tombol alamat akan ditampilkan alamatnya, dan bila dipilih tombol nomor handphone maka akan ditampilan nomor handphone seperti gambar 7.2 berikut:



Gambar 7.2. Tampilan latihan 7.2

Langkah-langkah pembuatan sebagai berikut:

- (1) Buat project baru dengan StandardExe. Pada form1, isi property [Name] dengan formLatihan72
- (2) Tambahkan komponen-komponen label, combobox dan command button. Atur posisinya seperti gambar 7.2 dan property setiap komponen seperti tabel di bawah ini

Komponen	Property	Nilai
Label1	Caption	Masukkan Nama
Label2	Appereance	0 – Flat
	Font	size 18 Bold
Command1	Caption	Alamat
Command2	Caption	No. Handphone
Combo1	Text	Basuki
	List	Buat samna dengan nama: Basuki Widya Teguh Dian

(3) Pada jendela kode, tambahkan kode program berikut:

```
Dim nama(4), alamat(4), hp(4) As String

Private Sub Command1_Click()
Tampilkan 1
End Sub

Private Sub Command2_Click()
Tampilkan 2
End Sub

Private Sub Form_Load()
'Ketika form dipanggil, data didefinisikan
nama(1) = "Basuki"
alamat(1) = "Semolowaru Utara Surabaya"
hp(1) = "081331344000"
nama(2) = "Widya"
alamat(2) = "Semolowaru Elok Surabaya"
hp(2) = "0315912190"
nama(3) = "Teguh"
alamat(3) = "Mulyosari Selatan Surabaya"
hp(3) = "0315997878"
nama(4) = "Dian"
alamat(4) = "Kompleks Laguna Surabaya"
hp(4) = "08111562145"
End Sub

'Subroutine yang dibuat sendiri
Private Sub Tampilkan(kondisi As Integer)
indeks = Combo1.ListIndex + 1
If kondisi = 1 Then Label2 = alamat(indeks)
If kondisi = 2 Then Label2 = hp(indeks)
End Sub
```

(4) Simpan project dengan nama projectLatihan72.

7.2. Fungsi

Sama seperti subroutine, fungsi adalah suatu blok program yang digunakan untuk suatu pekerjaan tertentu. Beda fungsi dan subroutine adalah subroutine tidak menghasilkan nilai sedangkan fungsi menghasilkan nilai. Fungsi sering kali digunakan untuk proses-proses perhitungan. Pernyataan fungsi secara umum dituliskan dengan:

$$\text{Output} = \text{Fungsi}(\text{Input})$$

Sebagai contoh untuk menyatakan fungsi f dengan input x dan output y dituliskan dengan $y=f(x)$.

Definisi fungsi di dalam Visual Basic mengikuti aturan fungsi dimana ada input dan ada output dituliskan dengan:

```
Private Function NamaFungsi(Input) As Tipe_Data_Output
    <Blok di dalam fungsi>
    .....
    NamaFungsi=<Nilai yang keluar dari Fungsi>
End Function
```

Contoh Aplikasi 7.3:

Membuat program untuk menghitung BEP (*Break Event Point*) dari perusahaan krupuk dengan ketentuan:

- (1) Harga satuan penjualan Rp 2000,- per-kemasan
- (2) Biaya produk terdiri dari biaya awal Rp 900.000,- biaya produksi per kemasan krupuk Rp. 600,- dan biaya transportasi setiap kemasan krupuk adalah Rp. 25,-

Berarti rumus dengan jumlah kemasan x untuk:

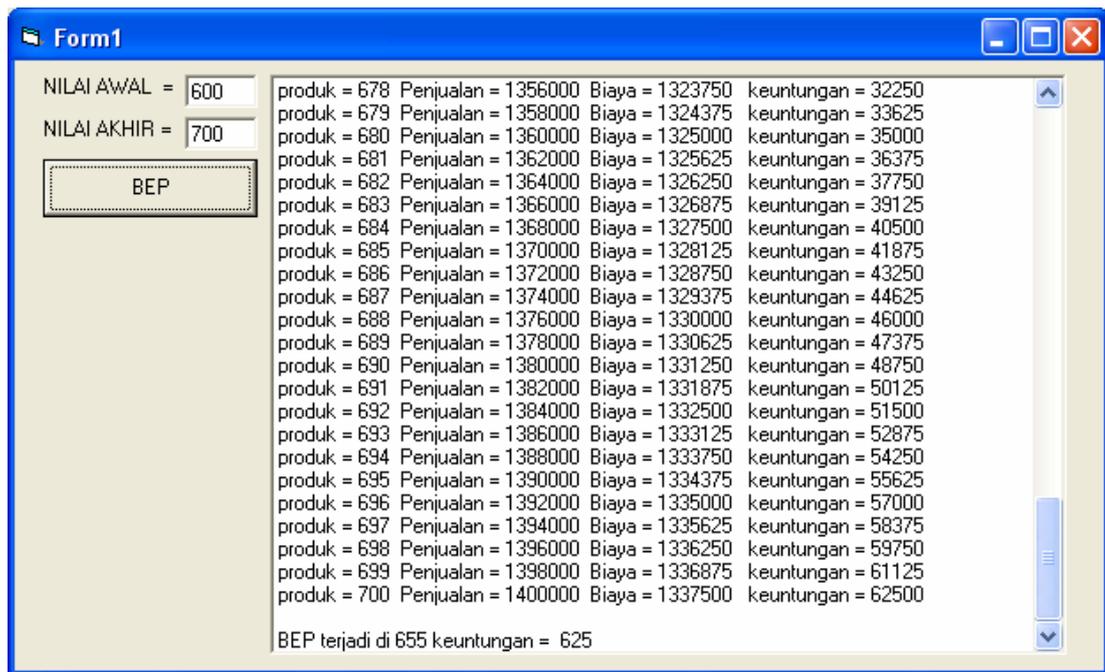
Penghasilan $P \rightarrow P = 2000.x$

Biaya $B \rightarrow B = 900000 + 625.x$

BEP adalah mencari x dimana $P = B$.

Langkah-langkah pembuatan program adalah sebagai berikut:

- (1) Buat project baru dengan StandardEXE. Pada form1, isi property [Name] dengan formLatihan73.
- (2) Tambahkan komponen-komponen: Label, TextBox, ListBox dan Command1. Pada command1, isi property [Caption] dengan BEP.
- (3) Atur posisi masing-masing komponen seperti gambar 7.3 di bawah ini



Gambar 7.3. Tampilan hasil latihan 73

(4) Pada layar code, tambahkan kode program sebagai berikut:

```
Private Function penjualan(x As Single) As Single
    penjualan = 2000 * x
End Function
```

```
Private Function biaya(x As Single) As Single
    biaya = 900000 + 625 * x
End Function
```

```
Private Sub Command1_Click()
    Dim x As Single
    awal = Val(Text1)
    akhir = Val(Text2)
    langkah = Val(Text3)
    List1.Clear
    BEPmin = 900000
    xmin = 0
    For x = awal To akhir
        P = penjualan(x)
        B = biaya(x)
        BEP = P - B
        If BEP < BEPmin And BEP > 0 Then
```

```

        BEPmin = BEP
        xmin = x
    End If
    kal = "produk =" & Str(x) & "  Penjualan =" & Str(P)
    kal = kal & "  Biaya =" & Str(B)
    kal = kal & "  keuntungan =" & Str(BEP)
    List1.AddItem kal
Next x
List1.AddItem ""
kal = "BEP terjadi di" & Str(xmin) & "  keuntungan = " &
Str(BEPmin)
List1.AddItem kal
End Sub

```

(5) Simpan project dengan nama projectLatihan73.

7.3. Latihan

(1) Buat program untuk memasukkan nama, gaji pokok, tunjangan, potongan dan PPH 10%. Setiap pengisian diakhir dengan tekan ENTER untuk melanjutkan pengisian yang berikutnya, misalkan habis mengetikkan nama tekan ENTER (kode ASCII=13) maka kursor mengarah pada pengisian Gaji Pokok. Buatlah fungsi perhitungan gaji.

The image shows a Windows application window titled "Form1". It contains four text input fields labeled "NAMA", "GAJI POKOK", "TUNJANGAN", and "POTONGAN" arranged vertically. To the right of these fields is a button labeled "HITUNG GAJI". Below the input fields is a large text box displaying "Jumlah Gaji Rp. 0". The window has a standard Windows title bar with minimize, maximize, and close buttons.

Gambar 7.4. Contoh tampilan soal 7.1

(2) Buat program untuk menghitung keuntungan penjualan harian dari jumlah produk yang dibeli dan jumlah yang terjual dalam satu hari. Harga satuan beli adalah Rp 3200,-/produk dan harga satuan jual Rp. 4000,-/produk. Bila jumlah produk yang dibeli lebih besar dari jumlah yang terjual, maka sisanya akan

dikenakan biaya sebesar Rp. 500 per-produk. Dengan data yang diketahui dalam satu minggu sebagai berikut:

Hari	Jumlah Pembelian	Jumlah Penjualan
1	250	230
2	210	210
3	300	295
4	300	290
5	210	210
6	210	220
7	250	250

- (3) Buat program untuk menghitung IPS (index Prestasi) mahasiswa menggunakan fungsi bila nilai dan sks dari semua mata kuliah dalam satu semester dimasukkan menggunakan variabel array.

Pada transkrip mahasiswa dalam satu semester terdapat mata kuliah yang jumlahnya tidak lebih dari 20, Masukkan array mata kuliah, jumlah jam, jumlah sks dan nilai huruf. Nilai huruf ini dikonversi ke nilai indeks untuk kemudian dihitung IPS dengan rumus:

$$IPS = \frac{\sum_k N(k) \cdot S(k)}{\sum_k S(k)}$$

Dimana : N(k) adalah nilai indeks dari mata kuliah ke k.

S(k) adalah jumlah SKS dari mata kuliah ke k.

- (4) Buat program dengan fungsi untuk menentukan keputusan apakah seorang calon pegawai akan diterima atau tidak berdasarkan tiga nilai yaitu nilai akademis, nilai attitude, dan nilai manajerial. Calon pegawai akan diterima bilai:
- Nilai akademis di atas 80, nilai attitude dan nilai manajerial di atas 55, ATAU
 - Nilai manajerial diatas 80, nilai akademis dan attitude di atas 60, ATAU
 - Rata-rata nilai di atas 75.
- (5) Buat program untuk menampilkan nilai akhir mahasiswa baik berupa nilai angka dan nilai huruf menggunakan fungsi perhitungan nilai berdasarkan nilai UTS, nilai UAS, dan nilai Tugas yang masing-masing prosentasenya adalah 40%, 50%

dan 10%. Pada program ini dimasukkan data siswa dalam sekelas dan hasilnya adalah laporan nilai kelas sebagai berikut

NAMA	HASIL EVALUASI			NILAI	
	UTS	UAS	TUGAS	ANGKA	HURUF

8.1. Tipe Data Waktu

Di dalam Visual Basic, dikenal adanya tipe data waktu baik itu Date maupun Time. Tipe data waktu ini digunakan untuk keperluan-keperluan pengolahan yang berhubungan dengan waktu, misalkan transaksi, lama proses, periode dan lain-lain. Untuk menyatakan tipe data waktu di dalam Visual Basic seperti halnya tipe data yang lain, asalkan variabelnya bukan array dan bukan global variable, maka tidak perlu didefinisikan. Pada dasarnya satuan yang digunakan dalam tipe data waktu di Visual Basic adalah hari. Untuk mengetahui waktu sekarang dapat menggunakan dua perintah waktu yaitu:

- (1) Time: menyatakan waktu berupa tanggal dan jam
- (2) Timer: menyatakan berapa detik hari ini sudah berjalan, misalkan jam 7.00 maka nilai timernya adalah $7 \times (60 \times 60) + 0 \times 60 + 0 = 18200$.

Contoh Aplikasi 1:

Menghitung tanggal setelah 5 hari dari saat ini, dapat dilakukan dengan :

- (1) Buat project baru dengan standardEXE, pada form isi property [Name] dengan formLatihan81. Buatlah seperti gambar 8.1 di bawah ini:



Gambar 8.1. Hasil tampilan 8.1

- (2) Pada jendela code, tambahkan kode program berikut:

```
Private Sub Command1_Click()  
    tanggal_hari_ini = Date  
    tanggal_5_hari_lagi = tanggal_hari_ini + 5  
    Text1 = tanggal_hari_ini
```

```
Text2 = tanggal_5_hari_lagi
End Sub
```

Perintah Date menghasilkan tanggal hari ini. Bila data waktu (termasuk tanggal dan waktu) ditambahkan dengan a, maka artinya ditambah 5 hari. Bila $a=1/24$ maka artinya waktu ditambah 1/24 hari atau 1 jam. Bila $a=1/(24*60)$ artinya waktu ditambah dengan 1 menit.

(3) Simpan project dengan nama projectLatihan81.

Contoh Aplikasi 2:

Menghitung waktu setelah d detik dari saat ini, dapat dilakukan dengan:

(1) Buat project baru dengan standardEXE, pada form isi property [Name] dengan formLatihan82. Buatlah seperti gambar 8.2 di bawah ini:



Gambar 8.2. Hasil tampilan 8.2

(2) Pada jendela code, tambahkan kode program berikut:

```
Private Sub Form_Load()
    waktu_saat_ini = time
    Text1 = waktu_saat_ini
End Sub

Private Sub Command1_Click()
    waktu_saat_ini = Time
    d = val(text2)
    waktu_tambah = d / (24 * 60 * 60)
    waktu_setelah_tambah = waktu_saat_ini + waktu_tambah
    Text3 = waktu_setelah_tambah
End Sub
```

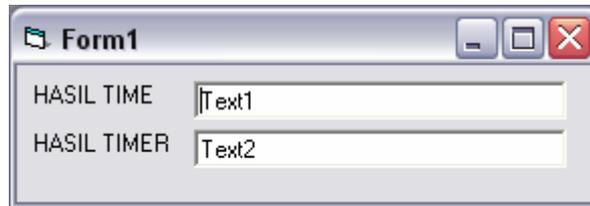
Perintah Time menghasilkan jam hari ini. Setiap penambahan waktu 1 dianggap 1 hari, untuk 1 detik harus dibagi dengan $24*60*60$.

(3) Simpan project dengan nama projectLatihan82.

Latihan Aplikasi 8.3:

Menunjukkan perbedaan hasil dari Time dan Timer, lakukan dengan cara sebagai berikut:

- (1) Buat project baru dengan standard EXE, pada form isi property [Name] dengan formLatihan83. Buatlah seperti gambar 8.3 di bawah ini:



Gambar 8.3. Hasil tampilan 8.3

- (2) Pada jendela code, tambahkan kode program berikut:

```
Private Sub Form_Load()  
    Text1 = Time  
    Text2 = Timer  
End Sub
```

Perintah Time menghasilkan jam hari ini. Setiap penambahan waktu 1 dianggap 1 hari, untuk 1 detik harus dibagi dengan $24*60*60$.

- (3) Simpan project dengan nama projectLatihan83. Hasil programnya adalah seperti gambar 8.4 berikut.



Gambar 8.4. Perbedaan time dan timer

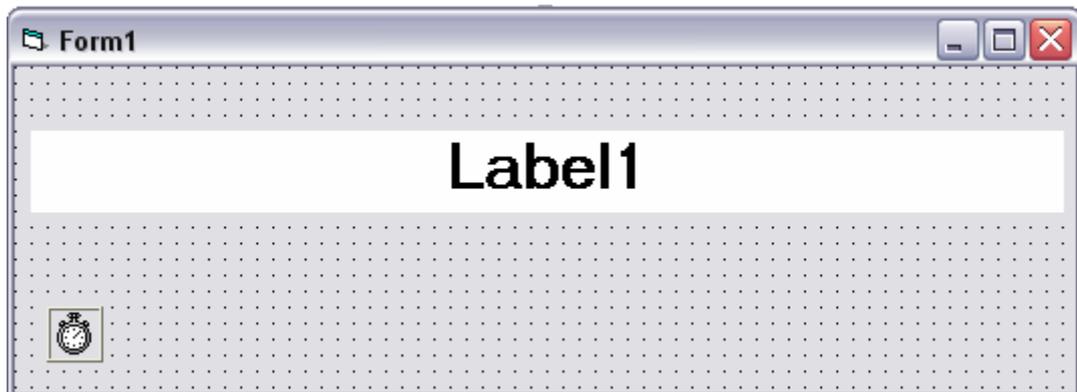
8.2. Komponen Timer

Selain menggunakan perintah Time dan Timer, Visual Basic juga menyediakan komponen Timer dengan ikon  pada toolboxnya. Perintah timer ini sangat baik untuk mengimplementasikan pengaruh waktu terhadap suatu proses seperti proses animasi atau dalam pembuatan game supaya kecepatan dari game bisa diatur.

Contoh Aplikasi 8.4:

Membuat text berputar seperti "SELAMAT DATANG" yang berputar dari kiri ke kanan, lakukan langkah-langkah sebagai berikut:

- (1) Buat project baru dengan standard EXE, pada form isi property [Name] dengan formLatihan84.
- (2) Buatlah seperti gambar 8.5 di bawah ini:



Gambar 8.5. Hasil tampilan 8.4

- (3) Isi property setiap komponen seperti tabel berikut:

Komponen	Property	Nilai
Label1	Alignment	2-Center
	Appearance	0-Flat
	Font	Size: 24 Bold
Timer1	Enabled	False
	Interval	100

Sebelum form jalan, property [Enabled] pada Timer1 dibuat False, sehingga gerakan belum jalan. Ketika form dipanggil property [Enabled] dibuat True untuk menjalankan waktu untuk gerakan. Property [Interval] pada Timer1 digunakan untuk menentukan waktu tunda yang besarnya n gerakan per 1ms, sebagai contoh interval dibuat 50, artinya setiap gerakan mempunyai waktu tunda 50 ms atau dalam 1 detik terjadi 20 kali gerakan.

- (4) Pada jendela code, tambahkan kode program berikut:

```
Dim kalimat As String
Dim panjang As Integer
Dim jalan As Boolean

Private Sub Form_Load()
    jalan = True
```

```

kalimat = "SELAMAT DATANG      "
panjang = Len(kalimat)
Label1 = kalimat
Label1.Refresh
Timer1.Enabled = True
End Sub

Private Sub Form_Unload(Cancel As Integer)
jalan = False
End Sub

Private Sub Timer1_Timer()
If jalan Then
    'Perintah ini untuk menaruh huruf terakhir
    'dari teks yang dibuat
    kalimat = Right(kalimat, 1) & Left(kalimat, panjang - 1)
    Label1 = kalimat
    Label1.Refresh
Else
    Timer1.Enabled = False
End If
End Sub

```

(5) Simpan project dengan nama projectLatihan83.

Catatan:

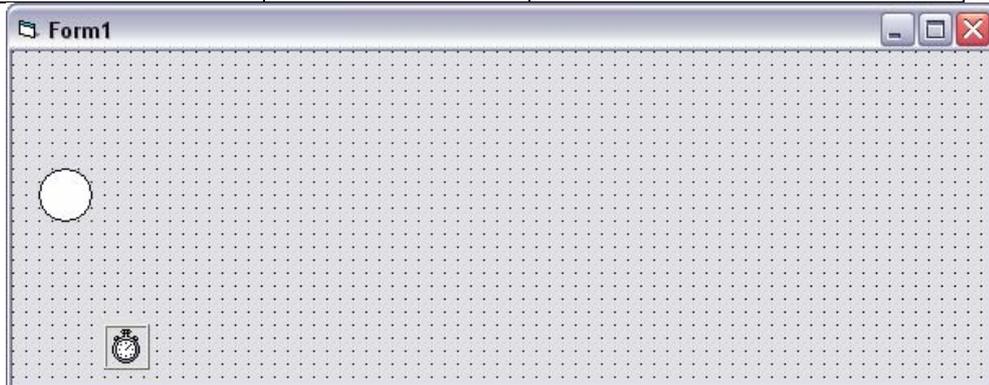
kalimat= Right(kalimat, 1) & Left(kalimat, panjang - 1) menggunakan dua fungsi **Right** dan **Left**. Fungsi **Right(kalimat,1)** digunakan untuk mengambil bil

Contoh Aplikasi 8.5:

Membuat obyek (shape berbentuk lingkaran) bergerak dari kiri ke kanan, dengan langkah-langkah sebagai berikut:

- (1) Buat project baru dengan standardEXE, pada form isi property [Name] dengan formLatihan85.
- (2) Buatlah seperti gambar 8.6, dan atur property setiap komponen seperti tabel di bawah ini.

Komponen	Property	Nilai
Shape1	BackColor	Pallete - Kuning
	BackStyle	1-Opaque
	Shape	3 - Circle
Timer1	Enabled	True
	Interval	100



Gambar 8.6. Tampilan latihan 8.5

- (3) Pada jendela code, tambahkan kode program untuk menggerakkan obyek. Program ini berisi pada saat form dipanggil diletakkan lingkaran dengan posisi(20,50) dengan asumsi skala jendela form adalah (400x100). Setelah itu setiap periode (interval pada timer1 atau pada event Timer1.Timer) posisi x dinaikkan satu persatu, jika x lebih besar dari 380 maka x menjadi 0.

```
Dim x, y As Integer
```

```
Private Sub Form_Load()  
Form1.ScaleWidth = 400  
Form1.ScaleHeight = 100  
x = 20: y = 50  
Shape1.Top = y  
Shape1.Left = x  
Timer1.Enabled = True  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)  
Timer1.Enabled = False  
End Sub
```

```

Private Sub Timer1_Timer()
x = x + 1: If x > 380 Then x = 1
Shapel.Top = y
Shapel.Left = x
Shapel.Refresh
End Sub

```

(4) Simpan project dengan nama projectLatihan85.

8.3. Latihan

- (1) Buat program untuk menghitung lama proses looping variabel x sebanyak 1000 kali untuk menghitung $y=2*x$ dan menampilkannya dalam ListBox, yang ditampilkan oleh komponen label.
- (2) Buat program untuk menampilkan jam digital pada label1 seperti gambar 8.7 di bawah ini.



Gambar 8.7. Tampilkan Soal 8.2

- (3) Buat program untuk menampilkan teks "SELAMAT DATANG" yang berjalan dari kanan ke kiri.
- (4) Buat program untuk menampilkan pesan jika jamnya sama dengan jam di set sebagai acuan, misalkan pada waktu kuliah, jam keluar adalah jam masuk ditambah dengan jumlah jam kuliah. Dengan memasukkan jam masuk dan jumlah jam kuliah, bila waktu jam keluar maka akan muncul pesan dan bunyi dengan perintah BEEP.
- (5) Buat program untuk menampilkan waktu deadline untuk pembayaran hutang dari semua data pelanggan. Hasilnya berupa informasi nama pelanggan, tanggal deadline dan berapa jumlah hari lagi untuk mencapai deadline.
- (6) Buat program untuk memberikan alarm saat seseorang sudah waktunya habis dalam menjawab suatu pertanyaan berhitung yang diberikan komputer. Dimana seseorang diberikan batas waktu untuk menjawab setiap pertanyaan adalah 3 menit, lebih dari itu jawabannya disalahkan.

9.1. Operasi File Dalam Visual Basic

Visual Basic menyediakan suatu fasilitas untuk membaca, menyimpan dan memperbaiki file data, sehingga memungkinkan untuk memisahkan data dengan program, dan program dapat menjadi lebih baik karena bisa tidak bergantung pada data (independent terhadap data), yang artinya datanya bisa diubah-ubah dengan mudah tanpa harus mengubah-ubah file programnya.

Operasi File di dalam Visual Basic terdapat bermacam-macam cara akses file ke dalam program aplikasinya, yaitu:

- (1) Membuka File
- (2) Membaca File
- (3) Menyimpan File

9.2. Membuka File

Untuk membuka file baik untuk keperluan penyimpanan maupun untuk keperluan pembacaan menggunakan perintah umum sebagai berikut::

```
Open "Nama_File" For {Status} As #{n}
```

Dimana: Nama_File : nama file lengkap dengan nama foldernya.

Status : adalah untuk keperluan apa file tersebut baca/simpan.

Status = INPUT untuk membaca File

Status = OUTPUT untuk menyimpan File

n : adalah nomor buffer (integer 1, 2, 3, ...) yang digunakan untuk keperluan pemilihan file yang mana yang akan diproses.

Setiap mengakses file baik untuk membaca maupun untuk menyimpan, maka file tersebut harus dibuka terlebih dahulu dengan status akses yang diberikan. Setelah file terbuka maka langkah berikutnya adalah bagaimana cara membaca dan menyimpan file.

File yang sudah dibaca, seharusnya ditutup saat program sudah selesai sehingga tidak ada data yang tertinggal di memory menyebabkan memory menjadi tidak efektif. Untuk menutup file dapat dilakukan dengan:

Close #n

9.3. Menyimpan Data Ke File

Untuk menyimpan data ke file di dalam program Visual Basic dapat dilakukan dengan membuka file untuk keperluan menyimpan atau menulis ke file:

Open Nama_File For Output As #1

Saat file dibuka dengan cara semacam ini, maka file hanya bisa ditulisi dan bila file tersebut sudah ada isinya sebelumnya, maka dianggap hilang. Atau dengan kata lain file selalu dianggap baru. Cara menyimpan file semacam ini menghasilkan file yang bersifat file text, sehingga bisa dibaca oleh *text editor* seperti notepad, wordpad atau microsoft word.

Setelah file dibuka, langkah berikutnya untuk menyimpan data dalam variabel ke file dapat dilakukan dengan:

Print #1, nama_variabel

Contoh Aplikasi 9.1:

Menyimpan data dalam buku tamu ke dalam file pegawai.txt, yang isinya adalah nama, alamat, dan nomor telepon. Contoh datanya adalah:

Nama	Alamat	No. Telp
Basuki	Semolowaru Utara Surabaya	081331344000
Widya	Semolowaru Elok Surabaya	0317810870
Dian	Mulyosari Selatan Surabaya	0811235623
Teguh	Arif Rahman Hakim 10A Surabaya	0315991010
Rizki	Keputih II/56 Surabaya	0315943201

Langkah-langkah pembuatannya adalah:

- (1) Buat project baru dengan StandardEXE. Untuk form1, isi property [Name] dengan formLatihan91.
- (2) Tambahkan komponen-komponen label, textbox, listbox dan command button. Atur tampilannya seperti gambar 9.1 di bawah ini

Gambar 9.1. Hasil tampilan latihan 9.1

(3) Pada jendela code, tambahkan kode program berikut.

```

Dim Ndata As Integer
Dim nama(100), alamat(100), notelp(100) As String

Private Sub Command1_Click()
'Mengambil data dari komponen text
Ndata = Ndata + 1
nama(Ndata) = Trim(Text1)
alamat(Ndata) = Trim(Text2)
notelp(Ndata) = Trim(Text3)
'Menampilkan ke listbox
kalimat = nama(Ndata) & " : " & alamat(Ndata)
kalimat = kalimat & " : " & notelp(Ndata)
List1.AddItem kalimat
'Kosongkan text
Text1 = ""
Text2 = ""
Text3 = ""
'Letakkan pointer di Text1
Text1.SetFocus

```

```

End Sub

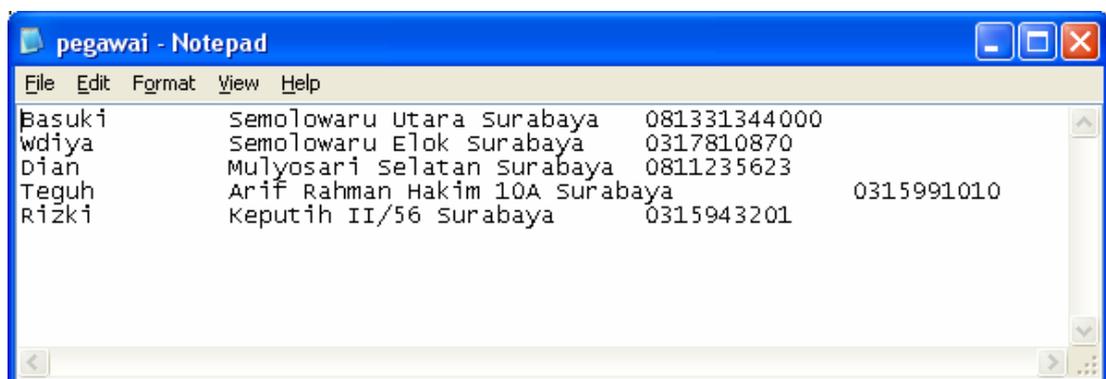
Private Sub Command2_Click()
    'Buka File untuk menyimpan data
    namafile = App.Path & "\pegawai.txt"
    Open namafile For Output As #1
    'Menyimpan semua data
    For i = 1 To Ndata
        Print #1, nama(i), alamat(i), notelp(i)
    Next i
    'Menutup file
    Close #1
End Sub

Private Sub Form_Load()
    Ndata = 0
    List1.Clear
End Sub

```

Perhatikan pada bagian mengambil data dari text terdapat fungsi **Trim** yang berguna untuk membersihkan spasi sebelah kiri dan kanan dari suatu teks. Perhatikan pada namafile ada perintah **App.Path** yang arti file diletakkan pada folder sama dengan program aplikasinya.

- (4) Simpan project dengan nama projectLatihan91. Jalankan program dan isi semua data yang diminta. Setelah itu tekan tombol [Simpan ke File]. Hasilnya adalah file pegawai.txt yang berada pada folder dimana program disimpan.
- (5) Buka file pegawai.txt menggunakan NotePad.



Gambar 9.2. Hasil file teks yang sudah dibuat

9.4. Membaca Data Dari File

Untuk membaca data dari file di dalam program Visual Basic dapat dilakukan dengan terlebih dahulu membuka file untuk keperluan membaca dari file:

```
Open Nama_File For Input As #1
```

Saat file dibuka dengan cara semacam ini, maka file hanya bisa dibaca dan tidak bisa ditulisi. File yang bisa dibaca adalah file teks.

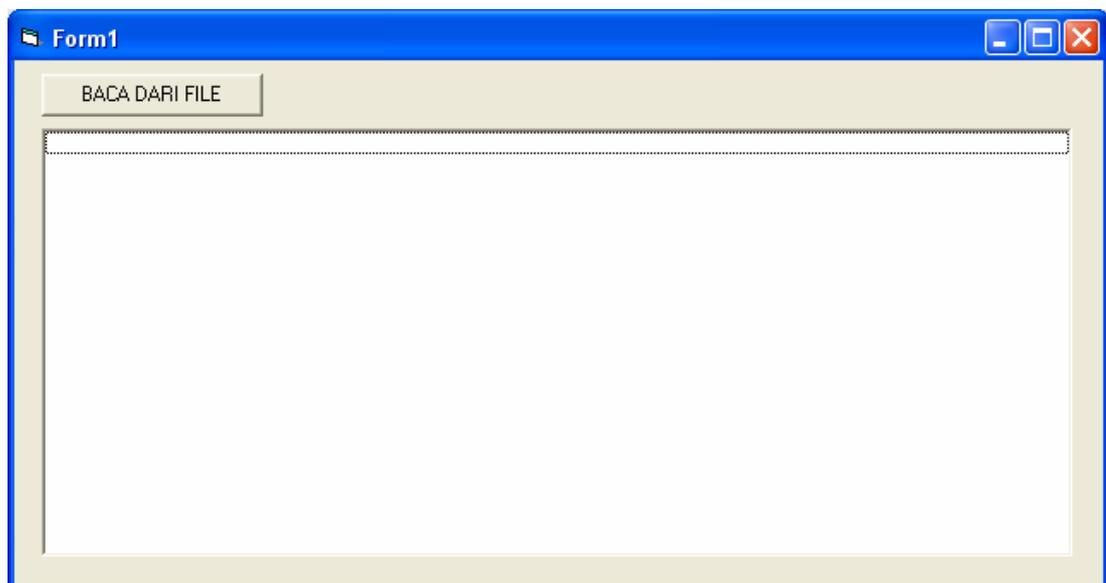
Untuk membaca data variabel ke dalam file, dapat dilakukan menggunakan perintah:

```
Input #1, nama_variabel
```

Contoh Aplikasi 9.2:

Membaca data dalam buku tamu yang sudah diketikkan dalam file pegawai.txt, dilakukan dengan:

- (1) Buat project baru dengan StandardEXE. Untuk form1, isi property [Name] dengan formLatihan92.
- (2) Tambahkan komponen-komponen listbox dan command button. Atur agar tampilannya seperti gambar 9.3 di bawah ini:



Gambar 9.3. Tampilan latihan 9.2

- (3) Pada jendela code, tambahkan kode program sebagai berikut:

```
Private Sub Command1_Click()  
    'Membuka file untuk membaca  
    namafile = App.Path & "\pegawai.txt"  
    Open namafile For Input As #1  
    'Membaca semua data file
```

```

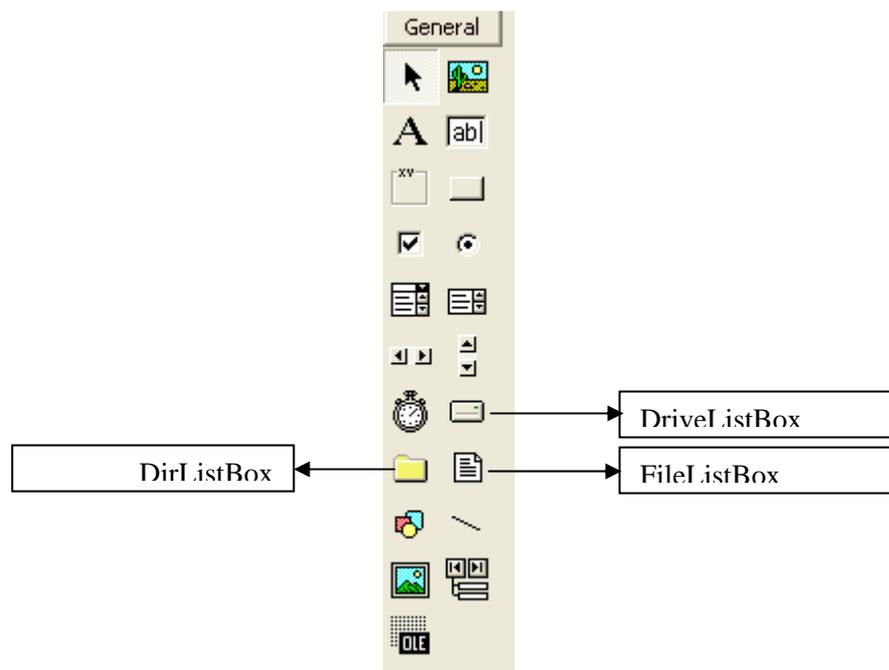
'sampai data terakhir (End Of File)
While Not EOF(1)
    'membaca data
    Input #1, datatamu
    'Menampilkan data di listbox
    kalimat = datatamu
    List1.AddItem kalimat
Wend
'Menutup file
Close #1
End Sub

```

(4) Simpan project dengan nama projectLatihan92.

9.5. Komponen-Komponen Untuk Operasi File

Visual Basic sudah menyediakan komponen-komponen untuk keperluan operasi file yang berupa DriveListBox, DirListBox dan FileListBox yang terletak pada toolbox.



Gambar 9.4. Komponen Access File dalam Toolbox

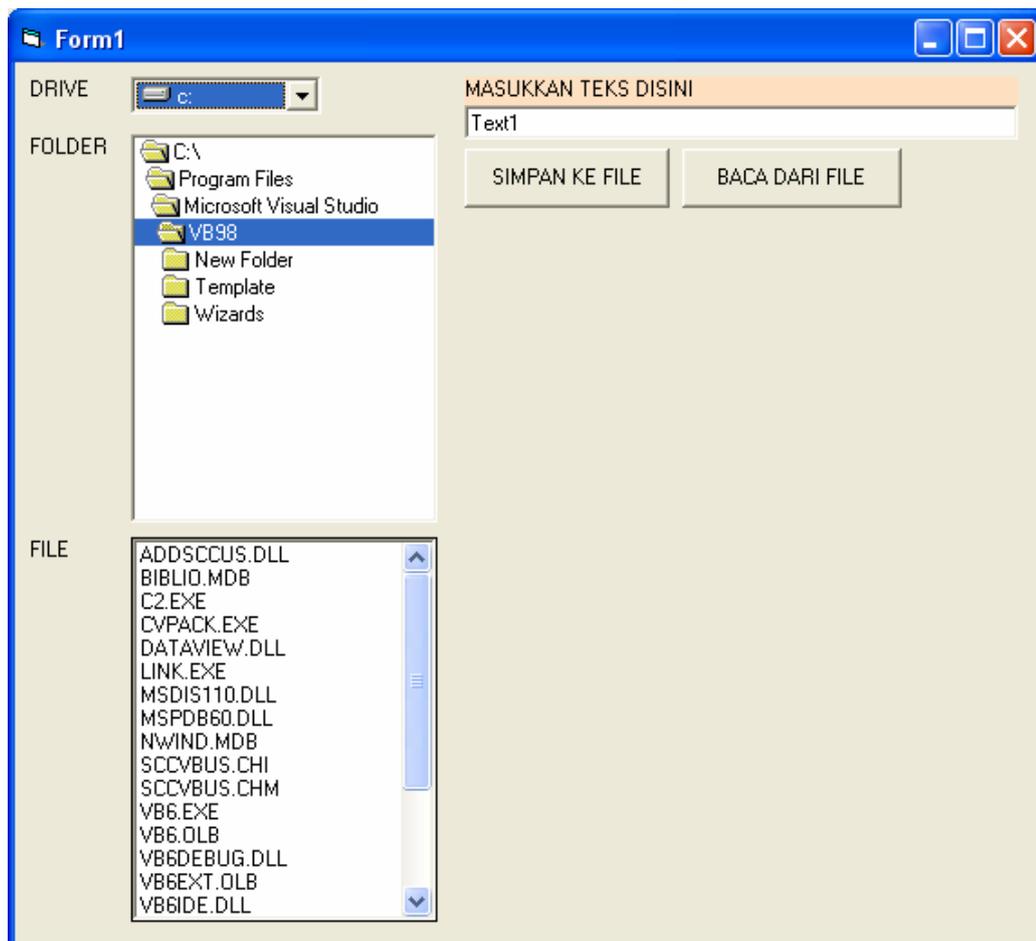
Keterangan setiap komponen File:

- **DriveListBox** adalah komponen yang dapat menampilkan semua drive yang ada di MyComputer.
- **DirListBox** adalah komponen yang dapat menampilkan semua folder yang ada di drive yang ditunjuk oleh DriveListBox
- **FileListBox** adalah komponen yang dapat menampilkan semua file yang ada dalam folder yang ditunjuk oleh DirListBos atau folder path.

Contoh Aplikasi 9.3:

Membuat program untuk menyimpan dan membaca dokumen, dengan langkah-langkah sebagai berikut:

- (1) Buat project baru dengan StandardEXE. Untuk form1, isi property [Name] dengan formLatihan93.
- (2) Tambahkan komponen-komponen listbox dan command button. Atur agar tampilannya seperti gambar 9.5 di bawah ini:



Gambar 9.5. Tampilan latihan 9.3

(3) Pada jendela code, tambahkan kode program berikut:

```
Private Sub Dir1_Change()  
    'Jika Dir berubah maka file akan berubah  
    'sesuai dengan folder yang ditunjuk  
    File1.Path = Dir1  
End Sub  
  
Private Sub Drive1_Change()  
    'Jika drive berubah maka dir dan file akan berubah  
    'sesuai dengan drive yang ditunjuk  
    Dir1.Path = Drive1  
    File1.Path = Dir1  
End Sub  
  
Private Sub Text1_KeyPress(KeyAscii As Integer)  
    'Jika pada text1 ditekan ENTER dan isinya tidak kosong  
    'maka data masuk ke listbox, text1 dikosongkan  
    'dan pointer menunjuk ke text1  
    If KeyAscii = 13 And Trim(Text1) <> "" Then  
        List1.AddItem Trim(Text1)  
        Text1 = ""  
        Text1.SetFocus  
    End If  
End Sub  
  
Private Sub Command2_Click()  
    'Membuka file dengan namafile yang ditunjuk  
    namafile = Dir1.List(Dir1.ListIndex) & "\"  
    namafile = namafile & File1.Left(File1.ListIndex)  
    List1.Clear  
    Open namafile For Input As #1  
    While Not EOF(1)  
        'Membaca data dari file buffer 1  
        Input #1, kal  
        'Menampilkan ke listbox  
        List1.AddItem kal  
    Wend
```

```

Close #1
End Sub

Private Sub Command1_Click()
    'Membuka file dengan namafile yang ditunjuk
    namafile = Dir1.List(Dir1.ListIndex) & "\"
    namafile = namafile & File1.Left(File1.ListIndex)
    Open namafile For Output As #1
    'Menghitung jumlah baris data yang ada di listbox
    n = List1.ListCount
    For i = 1 To n
        'mengambil data ke i dari listbox
        List1.ListIndex = i - 1
        'menyimpan ke file
        Print #1, Trim(List1.Text)
    Next i
    Close #1
End If
End Sub

```

- (4) Simpan project dengan nama projectLatihan93. Dengan program ini juga bisa dilakukan update data, dengan cara baca data yang sudah ada kemudian tambahkan data-data baru dan simpan kembali ke file.

9.6. Latihan

- (1) Dengan program contoh 9.3, buat dokumen berikut, dan simpan ke dalam dokumen1.

```
BERJALAN DALAM GELAP
```

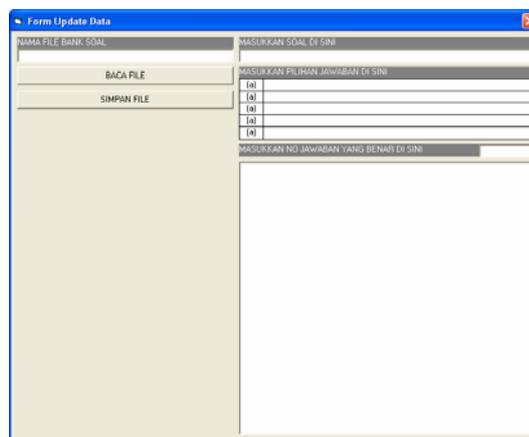
```
Maka tidak ada yang terlihat saat cahaya telah dimatikan
```

```
Kaki merangkak tanpa tujuan yang pasti
```

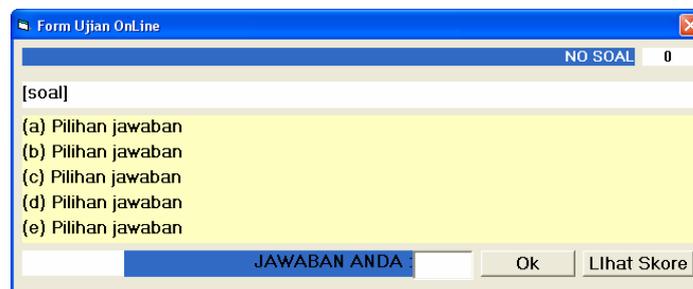
```
Hanya jiwa yang hidup yang dapat melangkah
```

- (2) Buatlah program untuk membaca file teks dokumen1.txt dan tampilkan jumlah kata yang ada dalam dokumen tersebut dengan tidak case-sensitif, artinya huruf besar dan huruf kecil dianggap sama.

- (3) Buatlah program untuk menyimpan salah satu dokumen pada berita dari salah satu koran yang anda sukai. Tampilkan kata-kata yang adalah dalam dokumen tersebut dan berapa kali kata-kata tersebut muncul dalam dokumen.
- (4) Buatlah program untuk ujian on-line sebagai berikut:
- Ujian on-line yang dimaksudkan di sini adalah ujian yang soalnya di generate oleh komputer dari bank soal yang sudah disediakan.
 - Aplikasi ujian on-line yang dibangun menggunakan bank soal yang disimpan dalam file teks.
 - Setiap soal mempunyai 5 pilihan jawaban, dengan satu jawaban yang benar.
 - Aplikasi ujian on-line membutuhkan dua form yaitu form untuk meng-update data dan form untuk ujiannya sendiri.
 - Program pada form update data sama seperti program update file text, karena tugas form ini mengupdate file bank soal yang berupa file text, tampilannya seperti gambar 9.6
 - Program pada form ujian online sama dengan program membaca file text, tampilannya seperti gambar 9.7



Gambar 9.6. Tampilan form update data dari ujian on-line



Gambar 9.7 Form ujian on line.

Bermain Dengan MsChart

Achmad Basuki

Surabaya, Nopember 2005

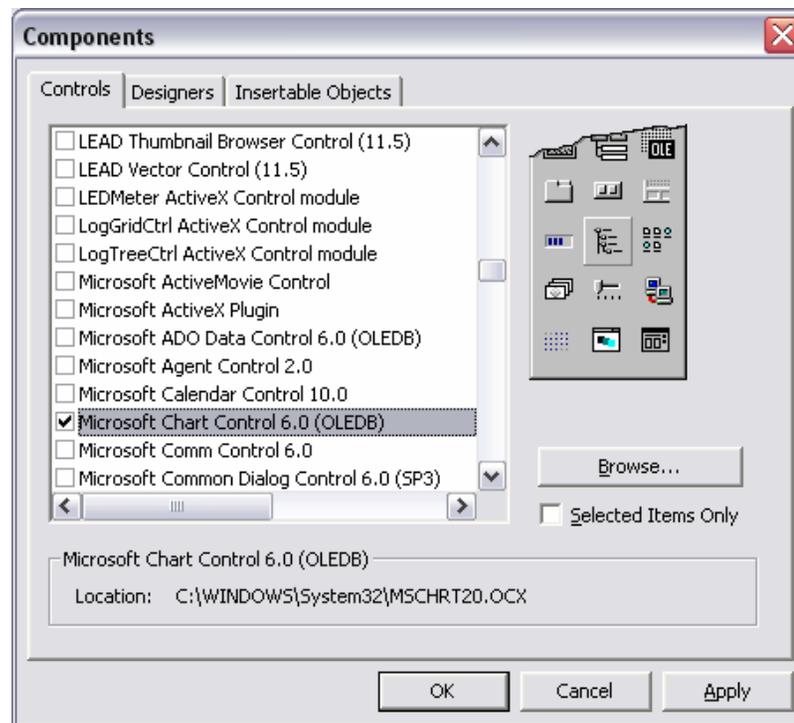


Materi

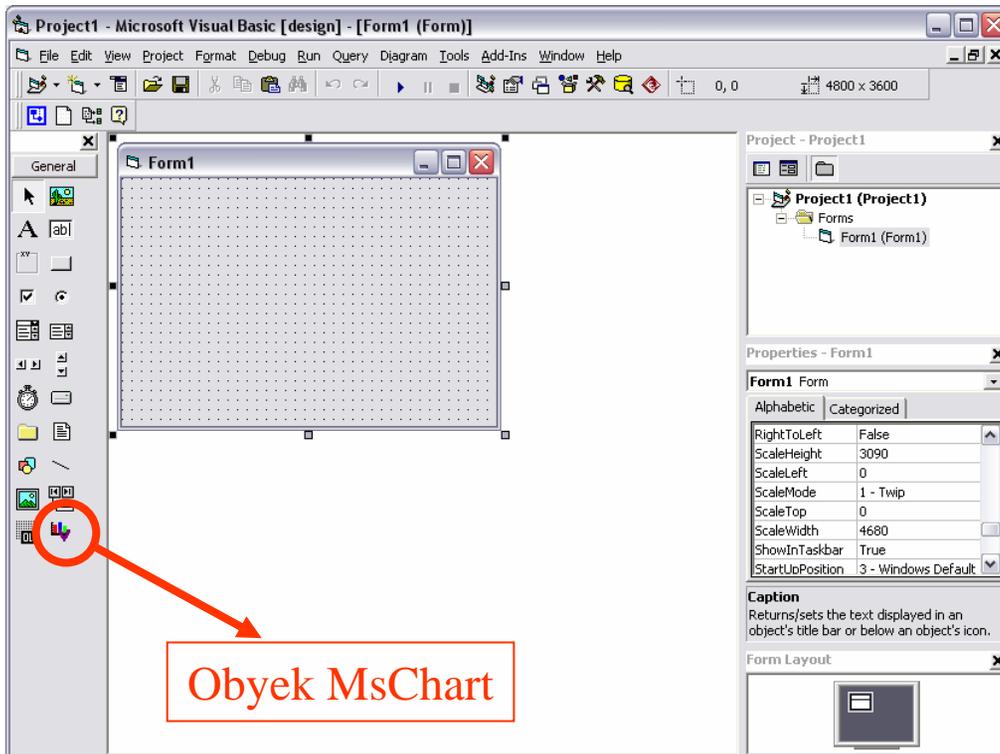
- Obyek MsChart
- Contoh Grafik Data
- Membuat Grafik Fungsi

Obyek MsChart

Menambah komponen MsChart pada Project,
Lakukan dengan menu Project >>Components...
Pilih Microsoft Chart Control 6.0 (OLEDB)



Obyek MsChart



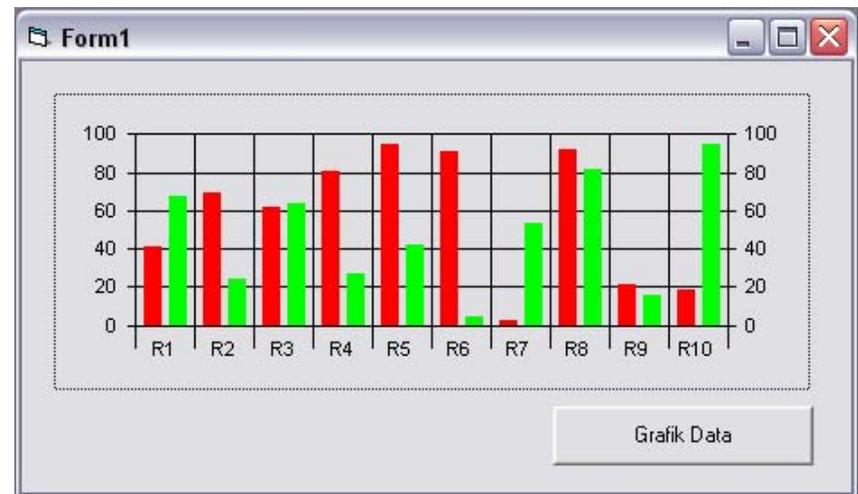
Property yang harus diatur:

- Column Count
- Row Count
- Chart Type

Untuk mengatur tampilan lebih baik, gunakan click kanan dan pilih Properties..

Contoh Grafik Data Dengan MSChart

- ❑ Buat project baru, tambahkan komponen MsChart.
- ❑ Ambil dan letakkan obyek MsChart pada form
- ❑ Atur property column count = 2 dan row count = 10
- ❑ Buat CommandButton, isi property Caption dengan Grafik Data
- ❑ Atur seperti tampilan di samping



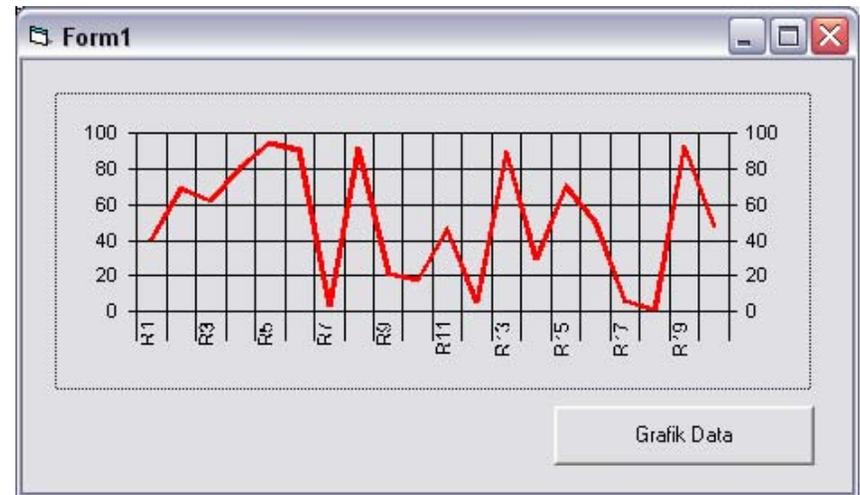
Contoh Grafik Data Dengan MSChart

- Click CommandButton dan tambahkan program berikut:

```
Private Sub Command1_Click()  
For i = 1 To 10  
    dat1 = Int(10 * Rnd)  
    dat2 = Int(10 * Rnd)  
    MSChart1.Row = i  
    MSChart1.Column = 1  
    MSChart1.Data = dat1  
    MSChart1.Column = 2  
    MSChart1.Data = dat2  
Next i  
End Sub
```

Membuat Grafik Fungsi $F(x)=x^2-1$

- Buat project baru, tambahkan komponen MSChart.
- Tambahkan obyek MSChart, atur properti `column count=1`, `row count=20`, `charttype = vtchChartType2DLine`
- Tambahkan `CommandButton`, atur properti `caption = Gambar Grafik`
- Atur letaknya seperti gambar di samping



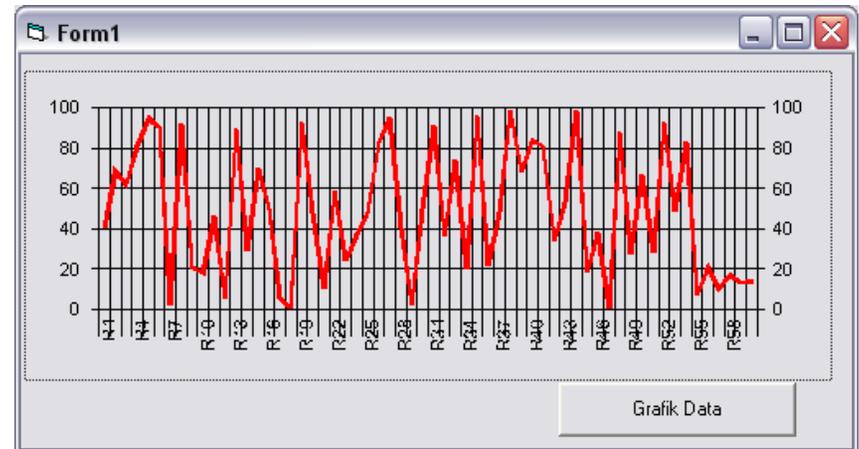
Contoh Grafik Data Dengan MSChart

- Click CommandButton dan tambahkan program berikut:

```
Private Sub Command1_Click()  
For X = 1 To 10  
    MSChart1.Row = X  
    Y = X ^ 2 - 1  
    MSChart1.Column = 2  
    MSChart1.Data = Y  
Next i  
End Sub
```

Membuat Grafik Fungsi $F(x)=\sin(x)$

- Buat project baru, tambahkan komponen MSChart.
- Tambahkan obyek MSChart, atur properti `column count=1`, `row count=60`, `charttype = vtchChartType2DLine`
- Tambahkan `CommandButton`, atur properti `caption = Gambar Grafik`
- Atur letaknya seperti gambar di samping



Contoh Grafik Data Dengan MSChart

- Click CommandButton dan tambahkan program berikut:

```
Private Sub Command1_Click()  
For X = 1 To 10  
    MSChart1.Row = X  
    Y = sin(X)  
    MSChart1.Column = 2  
    MSChart1.Data = Y  
Next i  
End Sub
```

Pengolahan Data Grafik

Achmad Basuki

Surabaya Nopember 2005

Tujuan

- ❑ Mengenalkan prinsip-prinsip pengolahan citra
- ❑ Melakukan proses pengolahan citra untuk keperluan pengolahan data grafik
- ❑ Melakukan perbaikan informasi citra
- ❑ Membuat aplikasi pengolahan citra



Materi

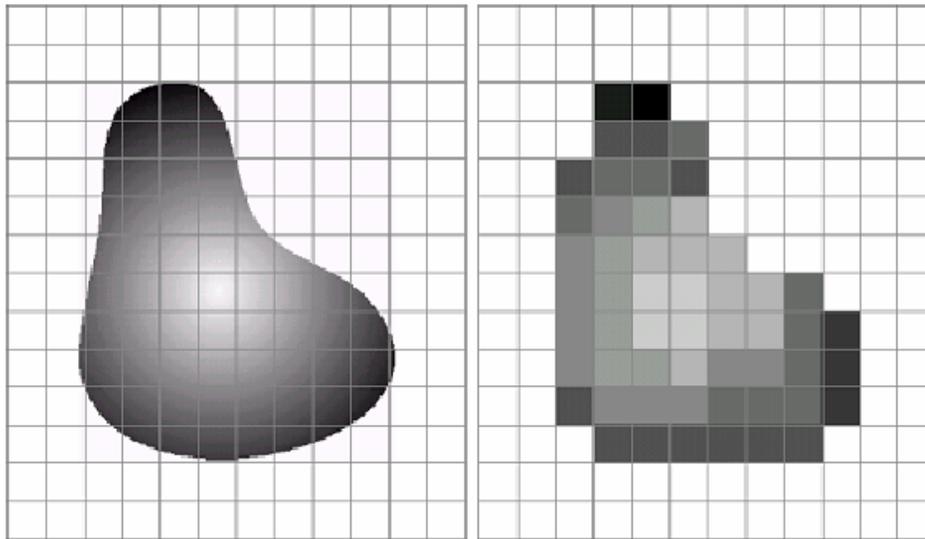
- Prinsip Pengolahan Citra
- Membaca Dan Menampilkan Data Citra
- Memperbaiki Kualitas Citra
- Statistik Data Citra
- Noise Dan Teknik Mereduksi Noise
- Deteksi Tepi

Prinsip Pengolahan Data

Tabel 1.1. Bidang komputer dilihat dari input dan outputnya

		Output	
		Image	Deskripsi
Input	Image	Image Processing	Pattern Recognition, Computer Vision
	Deskripsi	Computer Graphics	Data Processing lainnya

Citra Analog dan Citra Digital

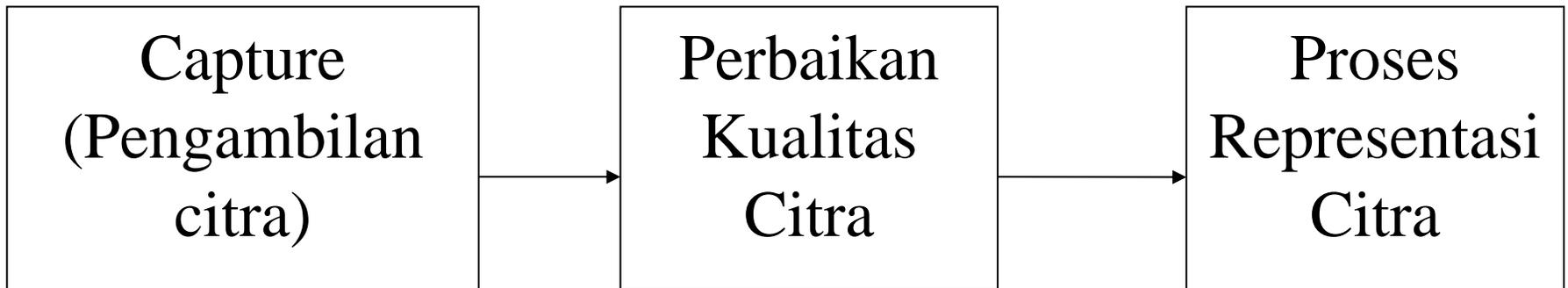


Continuous image
projected onto a
sensor array

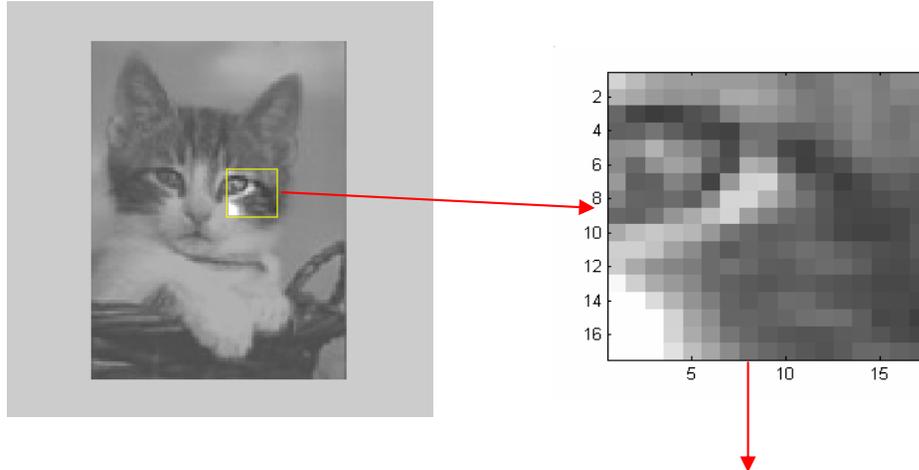
Result of image
sampling and
quantization

Citra analog dan digital tergantung pada perangkat pengambilan citra, apakah berupa perangkat analog atau digital

Proses Pengolahan Citra

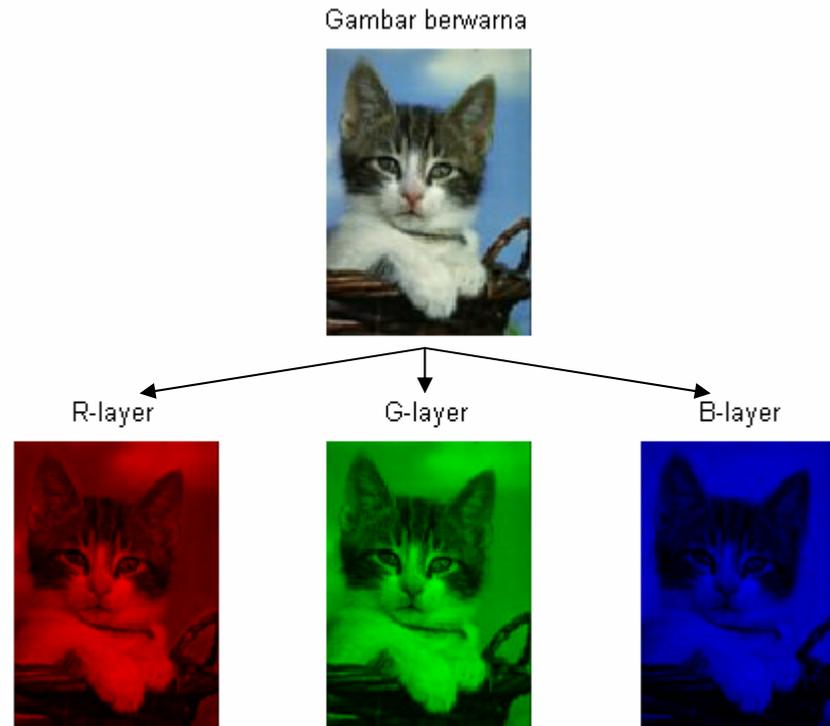


Model Data Citra



210	186	164	156	157	155	154	154	147	130	113	120	136	134	122	137	138
179	166	152	144	148	144	167	168	158	141	123	119	132	136	124	130	128
103	70	60	65	81	90	128	145	144	134	124	117	126	135	124	115	120
97	99	116	95	79	69	66	111	113	100	101	109	122	133	119	114	123
141	145	177	145	127	110	74	123	113	72	65	87	106	126	121	118	121
136	105	144	160	149	96	80	174	163	95	64	73	80	103	112	110	106
152	94	97	130	113	67	150	209	212	144	97	81	62	74	91	96	89
129	101	99	109	95	137	212	213	176	139	113	88	71	72	79	85	85
103	98	116	148	169	192	210	157	129	109	100	86	73	70	71	73	83
169	189	198	189	181	151	117	103	93	95	98	85	71	67	67	73	88
206	206	193	161	138	107	90	101	97	103	111	106	95	87	80	78	88
203	169	144	132	124	107	114	104	93	92	103	108	106	94	77	75	81
247	206	170	152	135	107	97	90	86	91	93	87	83	79	72	72	76
255	253	218	179	146	125	100	92	98	108	107	94	85	82	76	75	72
255	255	255	210	163	140	101	101	98	102	111	114	111	94	72	74	63
255	255	255	211	170	147	117	103	92	87	84	85	92	93	85	76	71
255	255	255	224	189	165	138	115	104	96	88	83	91	104	105	98	94

3 Layer RGB Pada Citra Berwarna



Teknik-Teknik Pengolahan Citra

- ❑ **Image Enhancement:** Proses perbaikan kualitas citra.
- ❑ **Image Segmentation:** Proses pembagian atau pengelompokan obyek pada citra
- ❑ **Image Restoration:** Proses perbaikan model pada citra
- ❑ **Color Image Processing:** Proses yang berhubungan dengan warna
- ❑ **Image Compression:** Proses kompresi pada data citra
- ❑ **Image Pattern Recognition:** Proses pengenalan pola pada citra

Beberapa Aplikasi Pengolahan Citra

- ❑ Biometric: Proses identifikasi manusia berdasarkan apa yang ada di badan manusia, seperti sidik jari, wajah, iris dll.
- ❑ Object Recognition: Proses pengenalan obyek gambar seperti tanda tangan, tulisan dll.
- ❑ Image Retrieval: Proses penyajian citra berdasarkan fitur citra seperti warna, bentuk dan tekstur.
- ❑ Image Compression
- ❑ Computer Vision: Proses penterjemahan obyek berdasarkan citra yang diambil oleh komputer.

Pemrograman Access File Teks Dengan Visual Basic



Achmad Basuki

Politeknik Elektronika Negeri Surabaya
PENS-ITS 2005



Materi

- Open File
- Menyimpan File Text
- Membaca File Text
- Update File Text
- Membuat Ujian On-Line



Open File

Fungsi untuk membuka file dalam Visual Basic:

```
Open "Nama_File" For {Status} As #{n}
```

Nama_File : Nama file lengkap dengan nama foldernya
Status : INPUT untuk membaca file text
 OUPUT untuk menyimpan file text
N : Nomor buffer (integer 1,2,3,.....)

Untuk menutup file digunakan: `Close #{n}`



Menyimpan File

- Untuk menyimpan file, gunakan status buka filenya OUTPUT sebagai berikut:

Open “Nama_File” For Output As #1

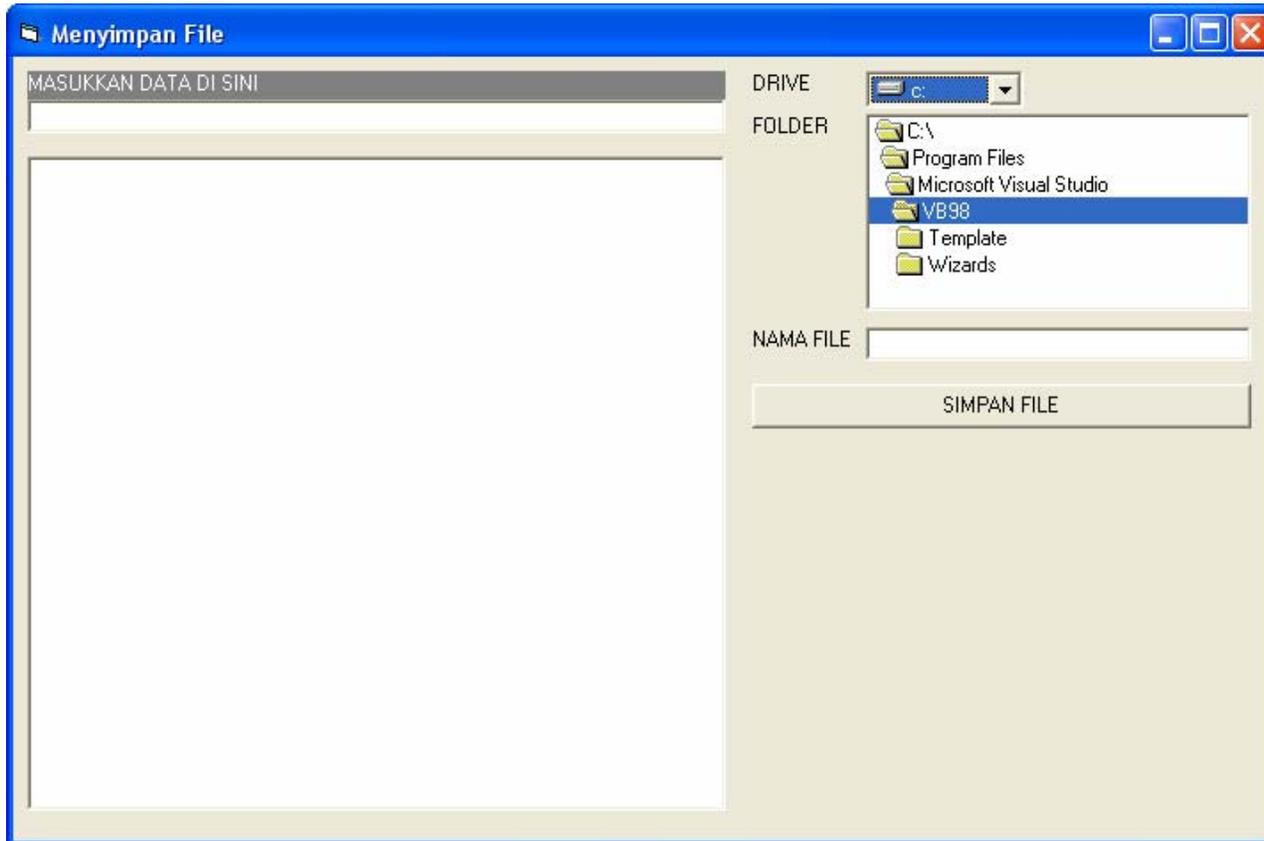
- Saat file yang dibuka dengan status Output, maka bila ada data sebelumnya pada file tersebut maka data akan hilang, jadi selalu dianggap membuka file baru.
- Perintah untuk menyimpan data dari file text baris per-baris adalah:

Print #1, nama_Var



Latihan Menyimpan File Teks

Buat tampilan sebagai berikut dengan menggunakan komponen TextBox untuk masukan data, listBox untuk menampilkan semua data, driveListBox untuk menampilkan dan memilih drive dalam komputer, DirListBox untuk menampilkan dan memilih folder pada drive yang sudah dipilih, TextBox untuk nama file, dan Command Button untuk proses menyimpan file.





Latihan Menyimpan File Text

- Double-click pada DriveBoxList, tambahkan program untuk menentukan drive aktif

```
Private Sub Drive1_LostFocus()  
Drive1.Refresh  
Dir1.Path = Drive1  
End Sub
```

- Double-click pada Text1, untuk menambah data pada listBox setelah tekan ENTER

```
Private Sub Text1_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 And Trim(Text1) <> "" Then  
List1.AddItem Trim(Text1)  
Text1 = ""  
Text1.SetFocus  
End If  
End Sub
```



Latihan Menyimpan File Text

- Double-click pada commandButton Simpan File, tambahkan program untuk menyimpan file

```
Private Sub Command1_Click()  
If Trim(txtNamaFile) <> "" Then  
namafile = Dir1.List(Dir1.ListIndex) & "\" & Trim(txtNamaFile)  
Open namafile For Output As #1  
n = List1.ListCount  
Text1 = n  
For i = 1 To n  
List1.ListIndex = i - 1  
Print #1, Trim(List1.Text)  
Next i  
Close #1  
End If  
End Sub
```



Membaca File

- Untuk membaca file, gunakan status buka filenya INPUT sebagai berikut:

Open “Nama_File” For Input As #1

- Saat file yang dibuka dengan status Input, maka file itu hanya bisa dibaca dan tidak bisa ditambahi.
- Perintah untuk menyimpan data dari file text baris per-baris adalah:

Print #1, nama_Var



Latihan Membaca File

Buat tampilan sebagai berikut dengan menggunakan komponen listBox untuk menampilkan semua data, driveListBox untuk menampilkan dan memilih drive dalam komputer, DirListBox untuk menampilkan dan memilih folder pada drive yang sudah dipilih, TextBox untuk nama file, dan Command Button untuk proses membaca file.

The screenshot shows a Windows application window titled "Form2". The window has a blue title bar with standard minimize, maximize, and close buttons. The main area is divided into several sections:

- DRIVE:** A dropdown menu showing "c:".
- FOLDER:** A list box containing the following folders:
 - C:\
 - Program Files
 - Microsoft Visual Studio
 - VB98** (highlighted)
 - Template
 - Wizards
- NAMA FILE:** An empty text box.
- BACA FILE:** A command button with a dotted border.

The right side of the window is a large, empty white area, likely intended for displaying the contents of the selected file.



Latihan Menyimpan File Text

- Double-click pada DriveBoxList, tambahkan program untuk menentukan drive aktif

```
Private Sub Drive1_LostFocus()  
Drive1.Refresh  
Dir1.Path = Drive1  
End Sub
```

- Double-click pada commandbutton Baca File, untuk membaca file text

```
Private Sub Command1_Click()  
If Trim(txtNamaFile) <> "" Then  
namafile = Dir1.List(Dir1.ListIndex) & "\" & Trim(txtNamaFile)  
List1.Clear  
Open namafile For Input As #1  
While Not EOF(1)  
Input #1, kal  
List1.AddItem kal  
Wend  
Close #1  
End If  
End Sub
```



Update File Text

- Buka file dengan status Input
- Baca semua data dan simpan dalam suatu variabel array
- Tutup file
- Tambahkan nilai-nilai baru (proses update) pada variabel array
- Buka file dengan status Output
- Simpan semua file array ke file yang dibuka
- Tutup File



Latihan Update File

Buat tampilan sebagai berikut dengan menggunakan komponen komponen TextBox untuk masukan data, listBox untuk menampilkan semua data, driveListBox untuk menampilkan dan memilih drive dalam komputer, DirListBox untuk menampilkan dan memilih folder pada drive yang sudah dipilih, TextBox untuk nama file, dan Command Button untuk proses membaca file dan menyimpan file

The screenshot shows a Windows application window titled "Form3". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The main content area is divided into several sections:

- DRIVE:** A dropdown menu showing "c:".
- FOLDER:** A list box containing the following folders: "C:\", "Program Files", "Microsoft Visual Studio", "VB98" (which is selected), "Template", and "Wizards".
- NAMA FILE:** A text box for entering the file name.
- Buttons:** Two buttons labeled "BACA FILE" and "SIMPAN FILE" are located below the file name text box.
- Text Area:** On the right side, there is a large text area with a header "MASUKKAN DATA DI SINI" and a dashed border, indicating it is a text box for data entry.



Latihan Update File Text

- Double-click pada DriveBoxList, tambahkan program untuk menentukan drive aktif

```
Private Sub Drive1_LostFocus()  
Drive1.Refresh  
Dir1.Path = Drive1  
End Sub
```

- Double-click pada Text1, untuk menambah data pada listBox setelah tekan ENTER

```
Private Sub Text1_KeyPress(KeyAscii As Integer)  
If KeyAscii = 13 And Trim(Text1) <> "" Then  
List1.AddItem Trim(Text1)  
Text1 = ""  
Text1.SetFocus  
End If  
End Sub
```



Latihan Update File Text

- Double-click pada commadbutton Baca File, untuk membaca file text

```
Private Sub Command1_Click()  
If Trim(txtNamaFile) <> "" Then  
namafile = Dir1.List(Dir1.ListIndex) & "\" & Trim(txtNamaFile)  
List1.Clear  
Open namafile For Input As #1  
While Not EOF(1)  
    Input #1, kal  
    List1.AddItem kal  
Wend  
Close #1  
End If  
End Sub
```



Latihan Update File Text

- Double-click pada commandButton Simpan File, tambahkan program untuk menyimpan file

```
Private Sub Command1_Click()  
If Trim(txtNamaFile) <> "" Then  
namafile = Dir1.List(Dir1.ListIndex) & "\" & Trim(txtNamaFile)  
Open namafile For Output As #1  
n = List1.ListCount  
Text1 = n  
For i = 1 To n  
List1.ListIndex = i - 1  
Print #1, Trim(List1.Text)  
Next i  
Close #1  
End If  
End Sub
```



Membuat Ujian On-Line

- Ujian on-line yang dimaksudkan di sini adalah ujian yang soalnya di generate oleh komputer dari bank soal yang sudah disediakan.
- Aplikasi ujian on-line yang dibangun menggunakan bank soal yang disimpan dalam file teks.
- Setiap soal mempunyai 5 pilihan jawaban, dengan satu jawaban yang benar.
- Aplikasi ujian on-line membutuhkan dua form yaitu form untuk meng-update data dan form untuk ujiannya sendiri.
- Program pada form update data sama seperti program update file text, karena tugas form ini mengupdate file bank soal yang berupa file text
- Program pada form ujian online sama dengan program membaca file text.



Form Pada Ujian On-line

Form Update Data

NAMA FILE BANK SOAL:

MASUKKAN SOAL DI SINI:

BACA FILE

SIMPAN FILE

MASUKKAN PILIHAN JAWABAN DI SINI

(a)	<input type="text"/>

MASUKKAN NO JAWABAN YANG BENAR DI SINI:

Form Ujian OnLine

NO SOAL:

[soal]

(a) Pilihan jawaban

(b) Pilihan jawaban

(c) Pilihan jawaban

(d) Pilihan jawaban

(e) Pilihan jawaban

JAWABAN ANDA:

Ok Lihat Score

Membaca dan Menyajikan Citra

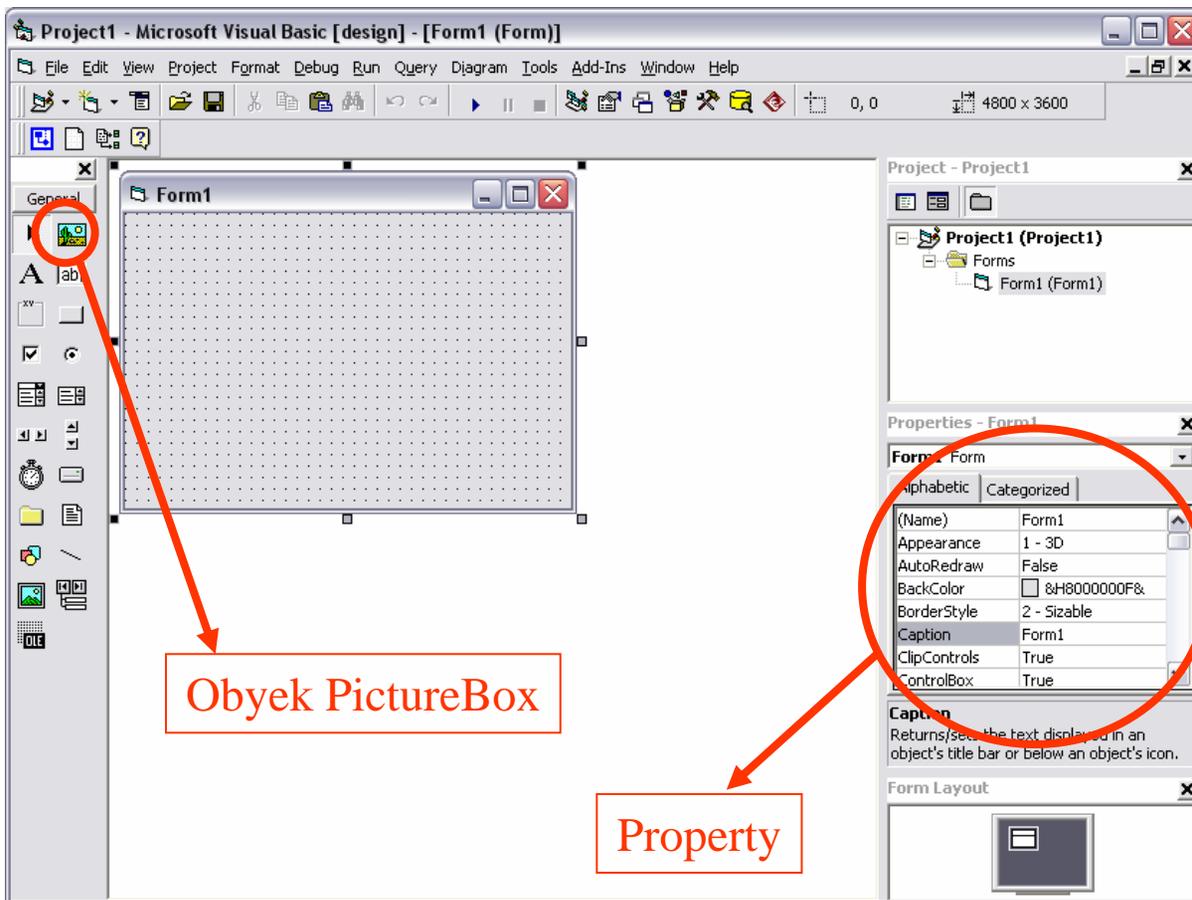
Achmad Basuki

Surabaya, Nopember 2005

Materi

- ❑ Obyek PictureBox
- ❑ Membaca Gambar Dari File
- ❑ Memberikan Warna Pada PictureBox
- ❑ Mengambil Nilai Pixel
- ❑ Capture Gambar Dari PictureBox
- ❑ Memutar Gambar Secara Horizontal
- ❑ Memutar Gambar Secara Vertikal

Obyek PictureBox



Property yang harus diatur:

- Appearance = Flat
- AutoRedraw = True
- Height
- Width
- Picture = Nama file gambar
- ScaleMode = Pixel

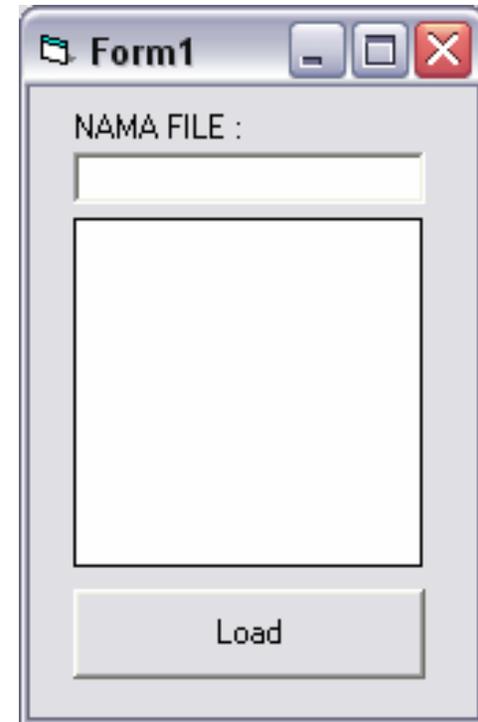
Membaca Gambar Dari File

- Letakkan picturebox pada form
- Fungsi untuk membaca gambar dari file dapat dilakukan dengan mengatur atribut picture pada picturebox atau menggunakan perintah:

Picture1.Picture=LoadPicture(“NamaFile”)

Membaca Gambar Dari File

- ❑ Buat project baru.
- ❑ Pada form, atur property ScaleMode=Pixel
- ❑ Tambahkan pictureBox, atur property Appearance=Flat, auto redraw=True, Height=128 dan Width=128
- ❑ Tambahkan Label, atur property caption>Nama File.
- ❑ Tambahkan TextBox, atur Name=txtFile, Text=kosong
- ❑ Tambahkan CommandButton, atur property Caption=Load
- ❑ Atur posisinya seperti gambar di samping.



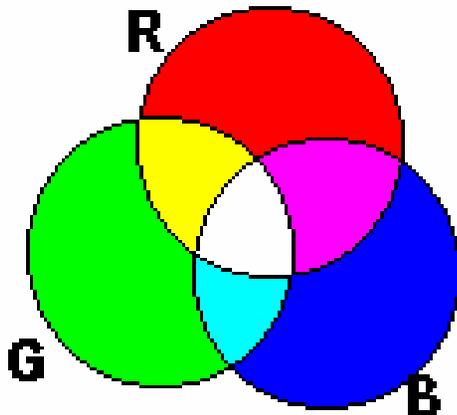
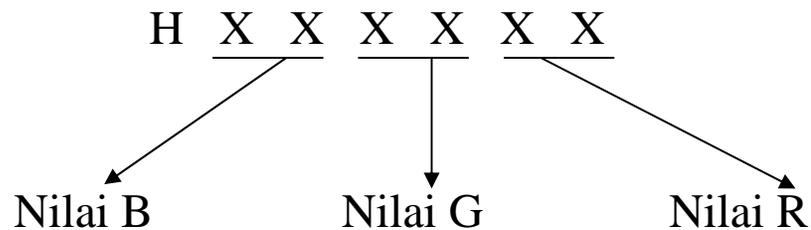
Membaca Gambar Dari File

- Click pada CommandButton, dan tambahkan program sebagai berikut:

```
Private Sub Command1_Click()  
namaFile = App.Path & "\" & Trim(Text1)  
Picture1.Picture = LoadPicture(namaFile)  
End Sub
```

- Program di atas mensyaratkan bahwa file gambar harus berada pada folder yang sama dengan program aplikasinya.

Konsep Warna RGB Pada Picture



Nilai	Warna
H000000	Hitam
H0000FF	Merah
H00FF00	Hijau
HFF0000	Biru
H00FFFF	Kuning
HFF00FF	Magenta
HFFFF00	Cyan
HFFFFFFF	Putih

nilai	Warna
H00AAFF	Orange
H888888	Abu-abu
HFF00AA	Ungu
HAAFF00	Hijau Muda
HAA00FF	Merah Muda
HAAFFFF	Kuning Muda
H0088AA	Coklat
HAA0088	Ungu

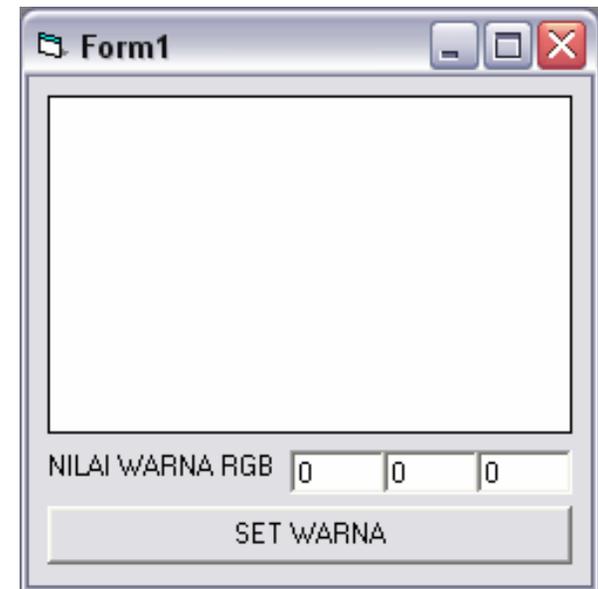
Konsep Warna Pada Picture

Warna	Hexa	R	G	B
HITAM	H000000	0	0	0
MERAH	H0000FF	255	0	0
HIJAU	H00FF00	0	255	0
BIRU	HFF0000	0	0	255
KUNING	H00FFFF	255	255	0
MAGENTA	HFF00FF	255	0	255
CYAN	HFFFFFF00	255	255	255
PUTIH	HFFFFFFF	255	255	255
ABU-ABU	H888888	128	128	128

Memberi Warna Pada PictureBox

- ❑ Buat project baru
- ❑ Tambahkan pictureBox dengan atribut yang sudah dijelaskan di depan
- ❑ Tambahkan label, isi caption = NILAI WARNA
- ❑ Tambahkan TextBox1, isi name = txtR
- ❑ Tambahkan TextBox1, isi name = txtG
- ❑ Tambahkan TextBox1, isi name = txtB
- ❑ Tambahkan commandButton, isi Caption dengan SET WARNA
- ❑ Click pada Command dan tambahkan program:

```
Private Sub Command1_Click()  
    r = Val(txtR)  
    g = Val(txtG)  
    b = Val(txtB)  
    Picture1.BackColor = RGB(r, g, b)  
End Sub
```



Mengambil Nilai Pixel

- Untuk mengambil nilai pixel (x,y) pada pictureBox digunakan perintah:

w = pictureBox.Point(x,y)

- Nilai w adalah nilai warna dalam RGB, untuk menghasilkan nilai r, g dan b diperlukan proses pemisahan bit dari 24 bit menjadi masing2 8 bit sebagai berikut:

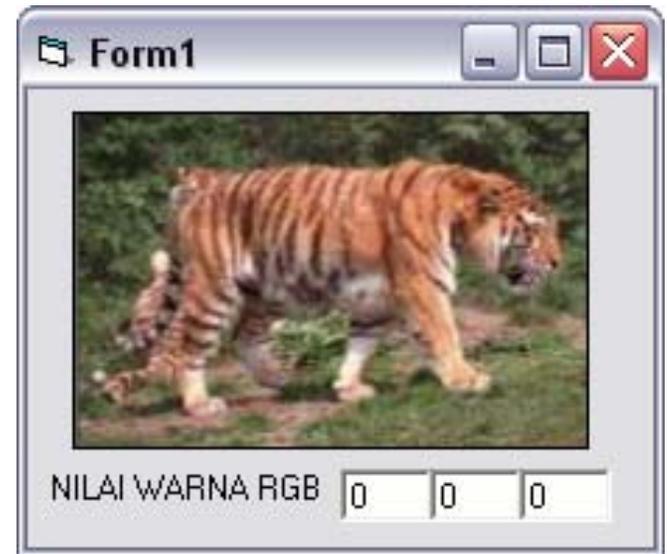
r = w and RGB(255,0,0)

g = int((w and RGB(0,255,0)) /256)

b = int(int((w and RGB(0,0,255)) /256)/256)

Mengambil Nilai Pixel

- ❑ Buat project baru
- ❑ Tambahkan pictureBox dengan atribut yang sudah dijelaskan di depan
- ❑ Tambahkan label, isi caption = NILAI WARNA
- ❑ Tambahkan TextBox1, isi name = txtR
- ❑ Tambahkan TextBox1, isi name = txtG
- ❑ Tambahkan TextBox1, isi name = txtB
- ❑ Isi PictureBox dengan gambar harimau dari property picture



Mengambil Nilai Pixel

- Tambahkan even MouseDown pada PictureBox untuk memberikan aksi ketika mouse diclick pada picture sebagai berikut:

```
Private Sub Picture1_MouseDown(Button As Integer, Shift As Integer, _  
    X As Single, Y As Single)  
    w = Picture1.Point(X, Y)  
    r = w And RGB(255, 0, 0)  
    g = Int((w And RGB(0, 255, 0)) / 256)  
    b = Int(Int((w And RGB(0, 0, 255)) / 256) / 256)  
    txtR = r  
    txtG = g  
    txtB = b  
End Sub
```

Capture Gambar Dari PictureBox

- ❑ Buat project baru
- ❑ Buat 2 pictureBox, picture1 diisi dengan gambar harimau dan picture2 dikosongkan, dengan property seperti telah dijelaskan di depan
- ❑ Buat CommandButton, isi caption=CAPTURE
- ❑ Click commandButton dan tambahkan program capture dari picture1 ke picture2 sebagai berikut:

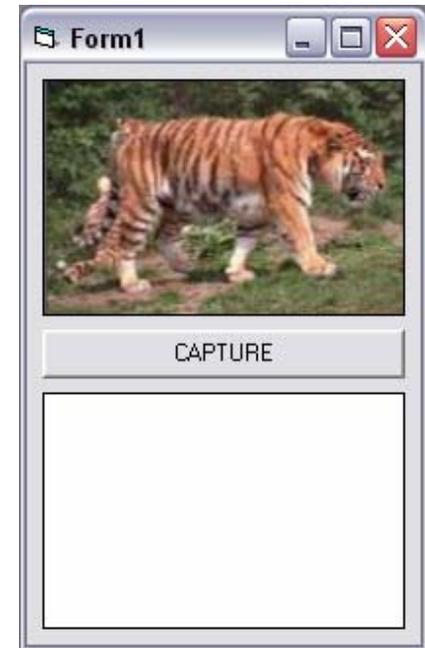
```
Private Sub Command1_Click()  
For i = 1 To Picture1.ScaleWidth  
    For j = 1 To Picture1.ScaleHeight  
        w = Picture1.Point(i, j)  
        r = w And RGB(255, 0, 0)  
        g = Int((w And RGB(0, 255, 0)) / 256)  
        b = Int(Int((w And RGB(0, 0, 255)) / 256) / 256)  
        Picture2.PSet (i, j), RGB(r, g, b)  
    Next j  
Next i  
End Sub
```



Memutar Gambar Secara Horizontal

- ❑ Buat project baru
- ❑ Buat 2 pictureBox, picture1 diisi dengan gambar harimau dan picture2 dikosongkan, dengan property seperti telah dijelaskan di depan
- ❑ Buat CommandButton, isi caption=CAPTURE
- ❑ Click commandButton dan tambahkan program capture dari picture1 ke picture2 sebagai berikut:

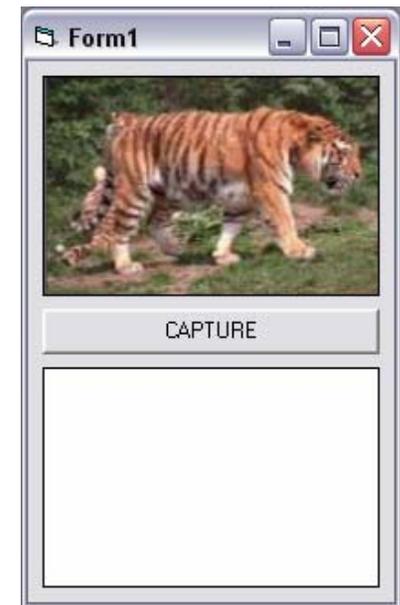
```
Private Sub Command1_Click()  
For i = 1 To Picture1.ScaleWidth  
    For j = 1 To Picture1.ScaleHeight  
        w = Picture1.Point(i, j)  
        r = w And RGB(255, 0, 0)  
        g = Int((w And RGB(0, 255, 0)) / 256)  
        b = Int(Int((w And RGB(0, 0, 255)) / 256) / 256)  
        Picture2.PSet (picture2.ScaleWidth-i, j), RGB(r, g, b)  
    Next j  
Next i  
End Sub
```



Memutar Gambar Secara Vertikal

- ❑ Buat project baru
- ❑ Buat 2 pictureBox, picture1 diisi dengan gambar harimau dan picture2 dikosongkan, dengan property seperti telah dijelaskan di depan
- ❑ Buat CommandButton, isi caption=CAPTURE
- ❑ Click commandButton dan tambahkan program capture dari picture1 ke picture2 sebagai berikut:

```
Private Sub Command1_Click()  
For i = 1 To Picture1.ScaleWidth  
    For j = 1 To Picture1.ScaleHeight  
        w = Picture1.Point(i, j)  
        r = w And RGB(255, 0, 0)  
        g = Int((w And RGB(0, 255, 0)) / 256)  
        b = Int(Int((w And RGB(0, 0, 255)) / 256) / 256)  
        Picture2.PSet (i, picture2.ScaleHeight-j), RGB(r, g, b)  
    Next j  
Next i  
End Sub
```



Daftar Pustaka

- [1] Krisna D. Octovhiana, Cepat Mahir Visual Basic 6.0, Kuliah Berseri IlmuKomputer.Com, Copyright © 2003 IlmuKomputer.Com
- [2] Rahmat Putra, Malik Akbar, 140 Trik Inovatif Visual Basic, Elex Media Komputindo , Jakarta 2004.
- [3] Lukman Hakim, Pemrograman Game Dengan Visual Basic, Penerbit Andi Yogyakarta, 2003.
- [4] Achmad Basuki, Pengolahan Citra Digital Menggunakan Visual Basic, Graha Ilmu Yogyakarta, 2005.