

VARIABEL, OPERATOR DAN EKSPRESI

Di susun Oleh :
H. Ary Setyadi

Di dukung oleh :
Portal edukasi Indonesia
Open Knowledge and Education
<http://oke.or.id>



3

VARIABEL, OPERATOR DAN EKSPRESI

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 pengenalan 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)
- 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 satu 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

Operatot 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 \Leftarrow **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

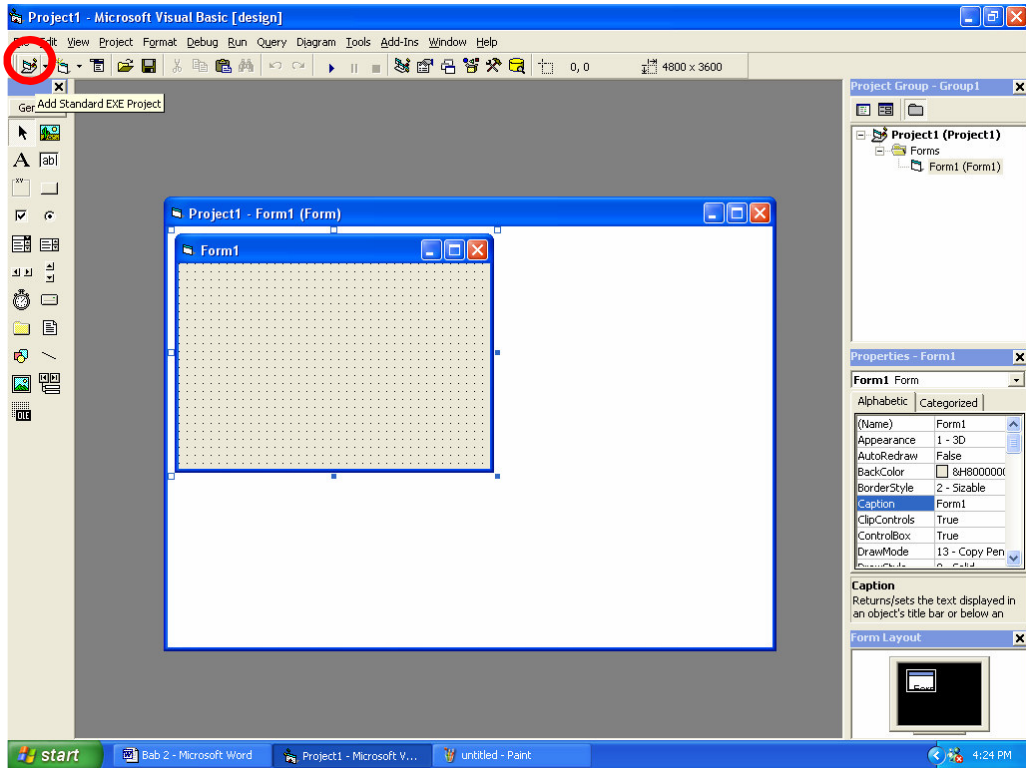
3.4. Project Sederhana Dengan Ekspresi

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

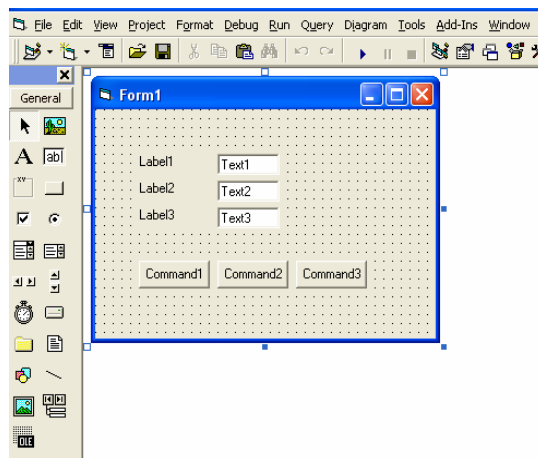
Untuk membuat project yang menyajikan operasi aritmatika sederhana seperti penjumlahan, pengurangan dan perkalian dapat dilakukan dengan langkah-langkah sebagai berikut:

- (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

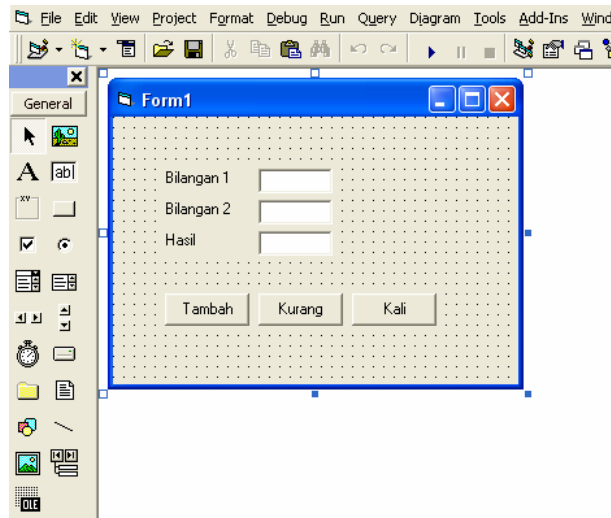
- (2) Tambahkan komponen-komponen: 3 komponen label, 3 komponen textbox dan 3 komponen commandButton yang ada di [toolbox]. Atur tempatnya pada form seperti gambar 3.2. berikut.



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)
```

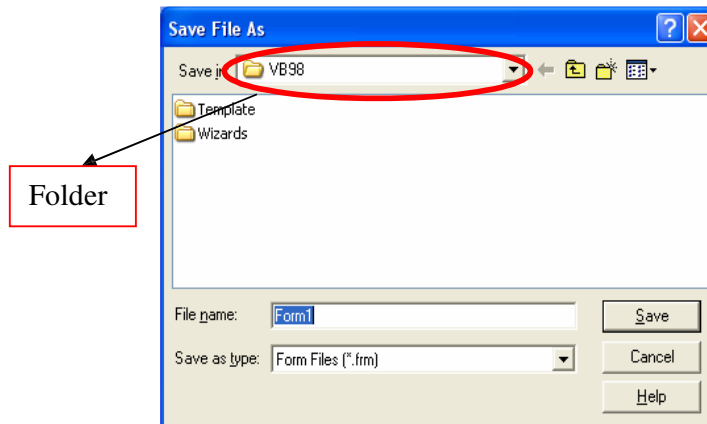
```
b=val(bil2)
```

```
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:

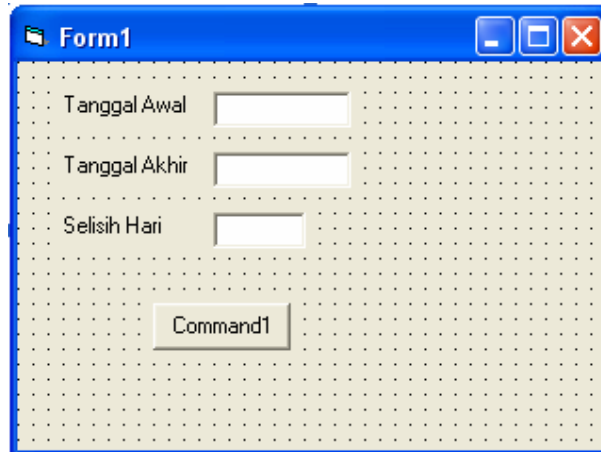
- (1) Buat project baru dengan Standard EXE
- (2) Tambahkan komponen-komponen: 3 label, 3 textbox, dan 1 command button.
- (3) Untuk label1, isikan property [Caption] dengan Tanggal Awal. Untuk label2 isikan property [Caption] dengan Tanggal Akhir. Dan untuk label3 isikan property [Caption] dengan Selisih 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 mengurangkan, hasil pengurangan berupa jumlah hari selisih hari.



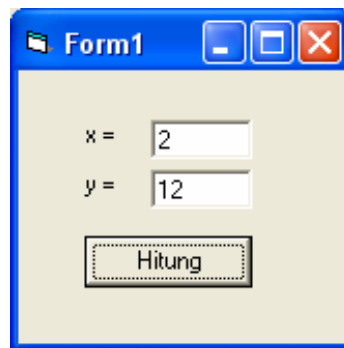
Gambar 3.5. Tampilan project selisih hari

- (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:

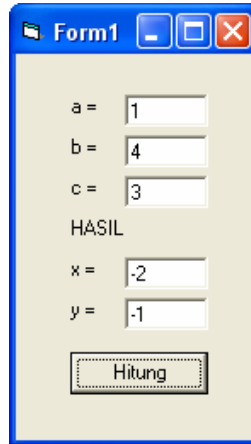
- (1) Menghitung nilai persamaan $y=x^2+3x+2$ dengan x diketahui, dan tampilan seperti gambar 3.6. berikut:



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:

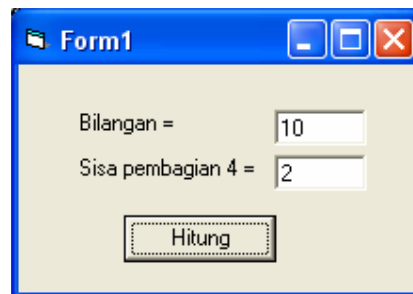
$$x = -\frac{b}{2a} \text{ dan masukkan nilai x kedalam persamaan } y = ax^2 + bx + c$$



The screenshot shows a Windows application window titled "Form1". It contains several input and output fields. The input fields are labeled "a =", "b =", and "c =", with values 1, 4, and 3 respectively. Below these is a label "HASIL" followed by output fields for "x =" (value -2) and "y =" (value -1). At the bottom center is a button labeled "Hitung".

Gambar 3.7. Contoh hasil latihan 2

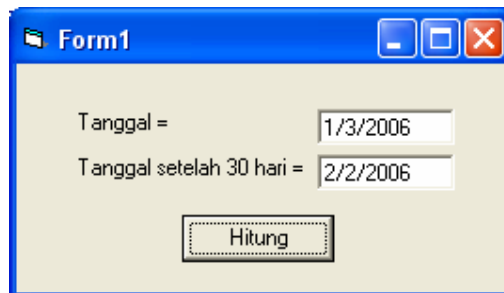
- (3) Menghitung sisa pembagian bilangan dibagi 4 dengan bilangan diketahui menggunakan rumus sisa=bilangan mod 4.



The screenshot shows a Windows application window titled "Form1". It contains two input fields: "Bilangan =" with the value 10, and "Sisa pembagian 4 =" with the value 2. At the bottom center is a button labeled "Hitung".

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 two input fields: "Tanggal =" with the value 1/3/2006, and "Tanggal setelah 30 hari =" with the value 2/2/2006. At the bottom center is a button labeled "Hitung".

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

HITUNG

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, kondisi2 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:

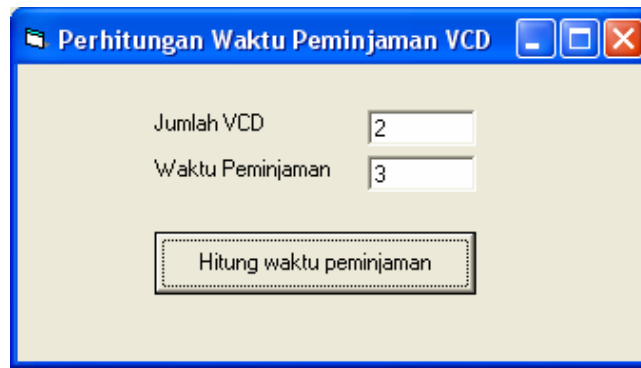
Sebuah perusahaan memberikan hadiah dan bonus pada karyawan yang jam kerjanya di atas 40 jam per-minggu dan tidak pernah terlambat dalam satu bulan. Karyawan yang jumlah jamnya di atas 40 jam per-minggu mendapat bonus sebesar 25.000 per kelebihan jamnya, dan bila dia juga tidak pernah terlambat akan 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
```

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 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 30m³ 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