

TIK

BAB 1-3

KATA PENGANTAR

Hai Kilaters!

Wah kerasa banget ya, habis ini udah mau PAS. Tapi tenang aja kilaters! Team Kilat kembali dan akan menemani kamu sepanjang menghadapi PAS ini, dengan Rangkuman Kilat dan Tutor Kilat. Bahannya banyak? Bukan masalah! Selama kita yakin, kita pasti bisa.

Pada Rangkuman ini, kami sudah menyesuaikan materinya dengan kisi-kisi yang ada.

Perlu diketahui bahwa **Rangkuman Kilat bukan berasal dari guru**. Jadi, gunakanlah rangkuman ini sebagai sarana/fasilitas untuk mendukung proses pembelajaranmu. Jangan jadikan rangkuman ini sebagai satu-satunya peganganmu.

Jika Kilaters ada pertanyaan, saran, kritik, pendapat, atau apapun mengenai Rangkuman Kilat ini, kalian dapat menghubungi *contact person* yang tertera di paling bawah setiap halaman. Akhir kata, selamat belajar dan sukses selalu!

26 November 2020,

Team Kilat

**TEAM
KILAT**



JUJU. / ALGORYTHM



CYNN / XNYSZ



KAK HARTO / MARKOVNIKOV



CENTRINO / NERDSQUARED

RANGKUMAN TIK

1. Pemrograman Visual

- a. Memungkinkan pengguna untuk berkreasi lebih baik dalam membuat program aplikasi
- b. Dasar pembuatannya adalah dengan form yang tampilannya dapat diatur lalu dijalankan dengan script yang sangat mudah
- c. Contoh: VB, Delphi, C++ Builder, Gambas, Kylix, Lazarus
- d. Bahasa pemrograman visual dibagi menjadi dua jenis:
 - i. **BASIC** (Beginners All-Purpose Symbolic Instruction Code)
Menawarkan Integrated Development Environment (IDE) visual untuk membuat program berbasis sistem operasi Windows dan menawarkan pengembangan software komputer berbasis grafik dengan cepat
Cth. VB dan Gambas
 - ii. **Delphi**
Umumnya lebih banyak digunakan untuk pengembangan aplikasi desktop dan enterprise berbasis database, tapi dapat juga digunakan dalam pengembangan software
Cth. Delphi, Kylix, Lazarus














2. IDE VB6

- a. Tabulasi New: membuat program aplikasi, project standar adalah **Standard.exe**
- b. Tabulasi Existing: membuka project yang pernah dibuat sebelumnya
- c. Tabulasi Recent: berisi project yang pernah dibuka sebelumnya















3. Elemen-elemen VB6

- a. Title bar: batang judul pada bagian atas window VB6, menunjukkan nama yang sedang aktif
- b. Menu bar: deretan menu perintah untuk menjalankan aksi-aksi khusus
- c. Main toolbar: deretan tombol perintah untuk mengakses perintah khusus dengan cepat

| TOMBOL | NAMA | FUNGSI |
|---|--------------------------|---|
|  | Add Standart Exe Project | Menambahkan Project Baru tipe standart |
|  | Add From | Menambahkan From baru kedalam proyek yang sedang aktif |
|  | Menu Editor | Menampilkan Menu Editor yang digunakan sebagai pembuat menu |
|  | Open Project | Membuka Project yang sudah pernah dibuat |
|  | Save Project | Menyimpan Project Yang sedang Aktif |
|  | Cut | Memotong Objek yang sedang dipilih |

| | | |
|---|--------------------|--|
|  | Copy | Menyalin Object yang sedang dipilih |
|  | Paste | Menampilkan object yang sudah dipotong dengan perintah Cut atau disalin dengan perintah Copy |
|  | Find | Mencari Teks Pada jendela Copy |
|  | Undo | Membatalkan Suatu perintah yang dijalankan sebelumnya |
|  | Redo | Mengulangi suatu perintah yang pernah dibatalkan |
|  | Start | Menjalankan Program |
|  | Break | Menghentikan sementara program yang sedang dijalankan |
|  | End | Menghentikan Program yang sedang dijalankan |
|  | Project Expoler | Menampilkan jendela Project Expoler |
|  | Properties Window | Menampilkan Jendela Properties |
|  | From Layout Window | Menampilkan Jendela Form Layout |
|  | Object Browser | Menampilkan Jendela Object Browser |
|  | Toolbox | Menampilkan Jendela Toolbox |

- d. Toolbox: kontrol-kontrol standar yang bisa dipasang ke form dengan double click

| TOMBOL | NAMA | FUNGSI |
|---|----------------|---|
|  | Pointer | Memilih, Mengatur ukuran, dan memindahkan posisi control yang terpasang pada bagian form |
|  | Picture Box | Menampilkan Gambar dari sebuah file |
|  | Label | Menampilkan teks, dimana pengguna program tidak bisa mengubah teks tersebut |
|  | TextBox | MembuatText, dimana text tersebut dapat diubah oleh pengguna program |
|  | Frame | Mengelompokkan beberapa control, frame ini harus dibuat terlebih dahulu sebelum di isi dengan control-kontrol yang akan dikelompokkan |
|  | Command Button | Membuat Tombol Perintah |
|  | CheckBox | Membuat Kotak Periksa, dimana pengguna program dapat memilih beberapa pilihan sekaligus |
|  | Option Button | Membuat Tombol Pilihan dimana pengguna program dapat memilih satu dari beberapa pilihan yang disediakan |
|  | ComboBox | Menambahkan Control kotak Combo yang merupakan control gandingan antara TextBox dan Listbox |
|  | ListBox | Membuat Daftar Pilihan |
|  | HScrollBar | Membuat control penggulung horizontal |
|  | VScroolBar | Membuat Kontrol Penggulung Vertikal |
|  | Timer | Membuat Kontrol waktu dan interval yang diperlukan |
|  | Drive ListBox | Menampilkan Disk Drive yang dimiliki oleh komputer |

| | | |
|---|-------------|--|
|  | DirListBox | Menampilkan Direktori dan Path yang dimiliki oleh Driver Aktif |
|  | FileListBox | Menampilkan Daftar File dari direktori aktif |
|  | Shape | Menambahkan control gambar berupa lingkaran, oval, persegi panjang, bujur sangkar, dan lain-lain |
|  | Line | Membuat Garis Lurus |
|  | Image | Menampilkan gambar pada form dalam format bitmap, icon, atau metafile |
|  | Data | Menghubungkan program dengan database dan menampilkan informasinya pada form |
|  | OLE | Membuat link antarprogram aplikasi |

- e. Jendela project: menampilkan nama project, form, atau modul yang terlibat
- f. Form: lembar kerja untuk menempatkan kontrol-kontrol yang diperlukan dalam membuat aplikasi. Ukuran dapat diatur dengan menggunakan handle dengan cara klik, tahan, dan geser. Diaktifkan dengan 3 cara:
 - i. Klik view object pada jendela project
 - ii. Klik menu view, pilih object
 - iii. Shift + F7 pada keyboard
- g. Jendela code: untuk menampilkan atau menuliskan kode program. Diaktifkan dengan:
 - i. Klik view code pada jendela project
 - ii. Klik menu view, pilih code
 - iii. Klik ganda pada form atau control pada form
- h. Jendela properties: untuk menentukan setting suatu objek
- i. Jendela form layout: untuk mengubah letak form saat menjalankan form. Klik, tahan dan geser ikon form dalam monitor untuk mengubah letak form, lalu tekan F5 untuk melihat hasil perubahan

4. Menggunakan VB6

- a. Komentar (Comment)
 - i. Suatu teks dalam badan program yang **tidak dieksekusi**
 - ii. Sangat memudahkan untuk mendokumentasikan perintah atau menelusuri error
 - iii. Dinyatakan dengan **menambahkan petik tunggal** (') di awal baris
- b. Variable
 - i. **Lokasi atau tempat penyimpanan sementara** dalam memori untuk data program
 - ii. Aturan pendeklarasian nama:
 1. Harus diawali huruf
 2. Tidak boleh menggunakan %, &, !, #, @, \$, dan spasi
 3. Panjang variabel tidak lebih dari 255 karakter

- Tidak boleh terdapat duplikasi nama
4. Tidak boleh mengandung reserved word (input, print dkk)
- iii. Didefinisikan dengan 3 perintah (**Dim**, **Public**, atau **Static**) dengan **menyebutkan jenis data** yang akan disimpan
Cth. Public namaBarang As String
 - iv. Jenis-jenis
 1. Global:
Ruang lingkup **seluruh project**, akan selalu aktif dimanapun, didefinisikan dengan perintah **Public**
 2. Lokal Dinamis:
Ruang lingkup **hanya pada sebuah prosedur**, karena bersifat dinamis setiap kali prosedur diproses ulang variabel akan bernilai kosong, didefinisikan dengan **Dim**
 3. Lokal Statis:
Ruang lingkup **hanya pada sebuah prosedur**, karena bersifat statis setiap kali prosedur diproses ulang variabel akan bernilai sesuai dengan nilai terakhir diproses, didefinisikan dengan **Static**
- c. Konstanta
 - i. Semacam variabel yang digunakan untuk **memasukkan data yang tidak dapat diubah nilainya**
 - ii. Jenis-jenis
 1. Konstanta Intrinsik: disediakan oleh aplikasi dan kontrol dalam VB6, cth. vbRed, vbKeyF10, dll.
 2. Konstanta Symbolic atau User-Defined: dideklarasikan sendiri oleh user dengan statemen Const
Cth. Private Const vbPembelian = 0
 - d. Tipe Data
 - a) **Integer**: bilangan bulat dengan besar penyimpanan 16 bit
 - b) **Long**: bilangan bulat, dengan penyimpanan 32 bit
 - c) **Byte**: bilangan positif dengan besar penyimpanan 8 bit
 - d) **Single**: bilangan bebas single precision, besar penyimpanan 4 byte
 - e) **Double**: bilangan bebas double precision, besar penyimpanan 8 bytes
 - f) **Currency**: bilangan bebas dalam format fixed-point, besar penyimpanan 8 bytes
 - g) **Boolean**: bilangan bulat yang hanya bernilai 0 dan -1 (True and False), besar penyimpanan 1 bit
 - h) **String**: data karakter berformat unicode, besar penyimpanan 32 bit
 - i) **Date**: data numeric yang dikonversikan dalam tanggal, besar penyimpanan 8 bit
 - j) **Object**: data abstract berupa object yang dilewatkan dalam sebuah reference, misal Form, Class, dll.
 - k) **Variant**: belum diketahui jenisnya, dapat digunakan untuk semua tipe data, besar penyimpanan 16 bytes

e. Event
Kejadian pada sebuah objek atau kontrol, misal saatu diklik atau menekan tombol keyboard

f. Property
Karakteristik atribut-atribut pada suatu object. Properti yang umum pada object:

- i. **Name** : nama object itu sendiri
- ii. **Alignment** : pengaturan posisi teks
- iii. **BackColor** : warna latar object
- iv. **Border** : tentang kondisi border atau tepi pembatas object
- v. **Enabled** : status object dapat diakses tidak
- vi. **Visible** : status object terlihat atau tidak

g. Function
Prosedur yang **dapat menghasilkan suatu nilai** dari operasi fungsi tersebut. Pendeklarasian fungsi dilakukan dengan cara seperti pendeklarasian method. Di dalamnya terdapat return value, perintah return diberikan untuk menunjukkan nilai yang dikembalikan. Contoh:

```
Function hitung_luas() as Integer
..... (perhitungan)
Return hasil
End
```

h. Operator sederhana

- i. Operator Logika: **membandingkan sebuah perbandingan data** yang menghasilkan nilai TRUE atau FALSE. Simbol yang digunakan adalah OR, AND, dan NOT
Cth. NOT (3 > 2), hasilnya FALSE
- ii. Operator Aritmatika: **mengolah matematis nilai data**. Simbol yang digunakan adalah * (perkalian), / (pembagian pecahan), \ (pembagian bulat), + (penambahan), - (pengurangan), ^ (perpangkatan), Mod (sisa), & (penggabungan string)
- iii. Operator Perbandingan: **membandingkan nilai data** yang akan menghasilkan nilai TRUE atau FALSE. Simbol yang digunakan adalah > , < , = , >= , <= , <> (tidak sama dengan)
Cth. 3 <> 2, hasilnya TRUE

5. Fungsi & Aplikasi pada percabangan dan perulangan

a. Fungsi Aritmatika

i. Rnd

Fungsi yang banyak digunakan dalam pemrograman ketika bergelut dengan masalah peluang (probability)

Contoh : $\text{rnd} * 6 =$ menghasilkan nilai acak mulai 1 sampai < 6

ii. Sqr

Sebuah fungsi akar kuadrat sebuah angka.

Contoh : $\text{Sqr}(4) = 2$, $\text{Sqr}(49) = 7$

iii. Int/Fix

Merubah angka desimal menjadi angka bulat (integer) dengan memotong angka dibelakang koma.

Contoh : $\text{Int}(0.88) = 0$, $\text{Int}(1.618033988749895) = 1$

iv. Abs

Menghasilkan angka mutlak/absolut

Contoh : $\text{Abs}(-9) = 9$

v. Log

Fungsi logaritma

Contoh : $\text{Log}(69) = 1.83884909074$

vi. Cos

Fungsi yg menghitung nilai cos sebuah sudut dalam satuan radian

vii. Sin

Fungsi yang menghitung nilai sin sebuah sudut dlm satuan radian

viii. Tan

Fungsi yg menghitung nilai tangent sebuah sudut dlm satuan radian

ix. Atn

Fungsi yang menghitung nilai arc tangent sebuah sudut dalam satuan radian

x. Round

Fungsi untuk melakukan pembulatan keatas dengan menentukan jumlah decimalnya. Format fungsi round adalah (m, n) membulatkan angka m sebanyak n decimal.

Contoh : $\text{Round}(3.14159265358979, 2) = 3,14$

xi. Exp

Fungsi eksponensial, misalnya $\text{Exp}(x) = e^x$

Contoh : $\text{Exp}(1) = e^1 = 2.7182818284590$

b. Fungsi String

i. Left

Mengambil karakter dari string x yang dimulai dari kiri sebanyak n

Contoh : $s = \text{Left}(\text{"Fakta Nyata"}, 3) = \text{"Fak"}$

ii. Right

Sama dengan fungsi left namun dimulai dari kanan sebanyak n

Contoh : $s = \text{Right}(\text{"Barrack Obama"}, 5) = \text{"Obama"}$

- iii. Mid
Mengambil karakter dari string x yang dimulai dari karakter m sebanyak n. (x, m, n)
Contoh : Mid("Joe Biden", 2, 2) = "oe"
- iv. Ltrim
Menghapus karakter spasi di bagian kiri dari string x.
Contoh : s = Ltrim(" Donald Trump "), s = "Donald Trump "
- v. Rtrim
Menghapus karakter spasi di bagian kanan string x
Contoh : s = Rtrim(" Donald Trumpet "), s = " Donald Trumpet"
- vi. Trim
Menghapus spasi di bagian kanan dan kiri string x.
Contoh : s = Trim(" Soeharto "), s = "Soeharto"
- vii. Len
Menghitung jumlah karakter dari string x
Contoh : s = Len("Prabowo"), s = 7
- viii. Lcase
Mengubah semua huruf di dlm string x menjadi kecil s'mua
Contoh : s = Lcase("TIDAK UNTUK ANAK DIBAWAH UMUR"),
s = "tidak untuk anak dibawah umur"
- ix. Ucase
Mengubah semua huruf dalam string x menjadi kapital.
Contoh : s = Ucase("yang balap monyet"), s = "YANG BALAP MONYET"
- x. Instr
Memberitahu posisi karakter n dalam string x.
Contoh : s = InStr("Astaghfirullah", "g"), s = 5
- xi. Chr
Memberitahu karakter dari nilai ASCII x.
Contoh : s = Chr(65) , s = A
- xii. Replace
Menggantikan karakter m di dalam string x dengan karakter n.
Contoh : s = Replace("owowo", "o", "u"), s = "uwuwu"
- c. Fungsi waktu
 - i. Date
Menunjukkan tanggal sekarang
Contoh : d = date , → 26/11/2020
 - ii. Now
Memberikan waktu dan tanggal sekarang
Contoh : d = Now , → 26/11/2020 20:22:34
 - iii. Time
Berguna untuk mendapatkan waktu saat ini.
Contoh : d = Time , → 20:22:38

- iv. Day(x)
Guna mengambil nilai tanggal saja dari x
Contoh : d = Day(#26/11/2904#) , hasilnya 26
- v. Month(x)
Untuk mengambil nilai bulan x.
Contoh : d = Month(#26/11/2904#) , hasilnya 11
- vi. Year(x)
Untuk mengambil nilai tahun saja dari x
Contoh : d = Year(#26/11/2904#) , hasilnya 2904
- vii. Hour(x)
Untuk mengambil nilai jam dari x
Contoh : d = Hour(#20:22:38#) , hasilnya 20
- viii. Minute(x)
Untuk mengambil nilai menit dari x.
Contoh : d = Minute(#20:22:38#) , hasilnya 22
- ix. Second(x)
Untuk mengambil nilai detik dari x.
Contoh : d = Second(#20:22:38#) , hasilnya 38
- x. DateSerial(x,y,z)
Guna menghasilkan tanggal dari kombinasi x, y, z.
Contoh : d = DateSerial(2012, 1, 13), hasilnya 13/01/2012
- xi. TimeSerial(x,y,z)
Menghasilkan waktu dari kombinasi x, y, z.
Contoh : d = TimeSerial(12, 15, 56) , hasilnya 12:15:56
- xii. MonthName(x)
Digunakan untuk mendapatkan nama bulan dari x.
Contoh : d = MonthName(12), hasilnya Desember.
- xiii. WeekdayName(x)
Mendapatkan nama hari dari x.
Contoh : d = weekday(2, , vbSunday) , hasilnya Senin

6. Pernyataan Percabangan

a. Percabangan dengan menggunakan **IF**

Sintaks percabangan IF:

```
IF [syarat/kondisi] THEN
    [pernyataan/perintah]
ELSEIF [ekspresi] THEN
    [Pernyataan/perintah]
ELSE
    [Pernyataan/perintah]
ENDIF
```

Keterangan :

- Syarat yang dituliskan boleh melibatkan semua jenis tipe data.
- Perintah yang diberikan boleh lebih dari satu
- ELSEIF bersifat opsional, digunakan pada percabangan dengan >1 kemungkinan
- Else juga opsional. Pernyataan dibawah ELSE bekerja ketika kondisi tidak ada yang sesuai dengan syarat/ekspresi dalam IF dan ELSEIF

Syarat yang bernilai 0 dan "" (String kosong) = FALSE

Selain itu, = TRUE

CONTOH :

“Jika uang di atas 100ribu, nongkrong dan berfoya-foya, Jika uang dibawah 100 ribu dan di atas 50 ribu, makan selusin Indomie, jika tidak punya uang maka bermimpilah”

Program :

```
PUBLIC SUB Main ()
DIM uang AS INTEGER
IF uang>100000 THEN
    PRINT “Nongkrong foya-foya.”
ELSE IF 100000>uang>50000 THEN
    PRINT “Makan INdomie”
```

```
ELSE
    PRINT"Ngidam aja sampe ngiler."
ENDIF
END
```

b. Percabangan dengan menggunakan **SELECT**

Sintaks percabangan SELECT :

```
SELECT CASE variabel
CASE syarat_1
    Pernyataan_1/Perintah_1
CASE syarat_2
    Pernyataan_2/Perintah_2
CASE ELSE
    Pernyataan_n/Perintah_n
END SELECT
```

Keterangan :

- Syarat yang dituliskan boleh melibatkan semua tipe data
- Perintah yang diberikan boleh lebih dari satu
- CASE ELSE bersifat opsional. Berfungsi dalam menangani semua syarat yang diatas yang tidak terpenuhi.
- Pernyataan Perulangan / Looping statement

c. Perulangan **FOR**

Sintaks FOR :

```
FOR nama_variabel = awal TO akhir [STEP angka_lompatan]
    pernyataan...
NEXT
```

Keterangan :

- nama_variabel adalah nama variabel yang akan dijadikan sebagai counter
- awal adalah nilai awal bertipe data numerik yg diberikan pada counter
- akhir adalah nilai akhir bertipe data numerik yg diberikan pada counter
- Angka_lompatan adalah nilai selisih/perubahan pada counter dalam setiap perulangan yang terjadi. Bila tidak dituliskan, maka secara default akan dianggap bernilai positif satu
- Jumlah pernyataan boleh lebih dari satu.

d. Perulangan **DO WHILE**

Sintaks DO WHILE LOOP :

DO WHILE syarat / kondisi

 Pernyataan/Perintah...

LOOP

Keterangan :

- Syarat yang dituliskan boleh melibatkan semua jenis tipe data
- Perintah yang diberikan boleh lebih dari satu

e. Perulangan dengan **REPEAT**

Sintaks pernyataan REPEAT :

REPEAT

 Pernyataan...

UNTIL syarat / kondisi

Keterangan :

- Syarat yang dituliskan boleh melibatkan semua jenis tipe data.
- Perintah yang diberikan boleh lebih dari satu.

f. Perulangan lain

Sintaks lain:

DO

 Pernyataan

LOOP WHILE syarat / kondisi

DO UNTIL syarat / kondisi

 Pernyataan

LOOP

WHILE syarat / kondisi

 pernyataan

WEND