

PEMOGRAMAN VISUAL FOXPRO 1

Disusun Oleh :
M. Arif Soeleman, M.Kom



FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Jl. Nakula I No. 5 - 11 Semarang Telp. 024 - 3517261, 3520165 Fax. 3569684
Homepage : <http://www.dinus.ac.id> E-mail : sekretariat@dinus.ac.id

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, yang telah memberikan rahmat dan nikmat berfikir, serta banyak karunia sehingga penulisan buku ini dapat diselesaikan dengan baik.

Buku ini ditulis untuk memberikan wawasan dasar dalam mempelajari dan mengembangkan pemrograman database menggunakan Visual Foxpro. Dalam buku ini diberikan penjelasan dan contoh-contoh serta latihan pada akhir praktikum. Diharapkan bagi pembaca khususnya mahasiswa dapat mempelajari dengan baik serta dapat mencoba menyelesaikan tugas praktikum yang ada dengan acuan contoh dan bahasan penggalan program. Disamping itu mahasiswa dituntut untuk mandiri dalam mempelajari bahasan demi bahasan, dengan cara memanfaatkan fasilitas bantuan (*help*) yang ada pada visual foxpro. Sehingga peranan pengajar dapat menjadi *partner* dalam diskusi dengan mahasiswa dapat semakin meningkat.

Akhirnya dengan segala kerendahan hati, penulis mengharapkan kesediaannya untuk memberikan kritik dan saran yang konstruktif demi kesempurnaan buku ini. Sebab penulis menyadari sebagai seorang manusia tidak lepas dari segala kesalahan dan keterbatasan. Semoga apa yang ada ini merupakan titik awal untuk menuju semakin sukses selanjutnya, Amin.

Semoga buku ini dapat bermanfaat.

Semarang, Pebruari 2006

Penulis

DAFTAR ISI

Kata Pengantar.....	iii
Daftar Isi	v
PERTEMUAN 1 Instalasi Software VFP 8.0 dan Konsep Pemrograman..	1
1.1 Instalasi Software	1
1.2 Setting Konfigurasi	6
1.2.1 Option Data	6
1.2.2 Option Field Mapping	7
1.2.3 Option Editor.....	9
1.2.4 Option Controls	9
1.3 Konsep Dasar pemrograman	10
1.4 Data	10
1.5 Tipe Data	10
1.5.1 Tipe Karakter	12
1.5.2 Tipe Numerik dan Bilangan Pecahan	13
1.5.3 Tipe Bilangan Bulat	13
1.5.4 Tipe Tanggal	14
1.5.5 Tipe Logika (L)	14
1.5.6 Tipe Data Memo (M)	14
1.5.7 Tipe Umum (G)	15
1.6 Mengolah data dengan Operator	15
1.7 Menggunakan Function	16
1.8 Jenis File dalam Foxpro	17
PERTEMUAN 2 Project Manager, Database dan Tabel.....	19
2.1 Project Manager	19
2.1.1 Membuat Project Baru.....	19
2.1.2 Membuat dan memodifikasi Files	20
2.1.3 Menambahkan sebuah Description pada file	21
2.2 Konsep Database	22
2.2.1 Menciptakan Database Baru	22
2.2.2 Toolbox Database	24
2.2.3 Menambahkan Tabel ke dalam database	25
2.2.4 Membebaskan Tabel dari Database	26
2.2.5 Menghapus Tabel Secara permanent.....	26
2.3 Tabel	26
2.3.1 Konsep Tabel	26
2.3.2 Membuat Tabel Baru	27
2.3.4 Mengubah Struktur Tabel	31
Latihan Praktikum II	32

.... 35	PERTEMUAN 6 Mendesain Form	75
.... 35	6.1 Menenal Form	75
.... 37	6.2 Data Environment	80
... 39	6.3 Menambahkan Object pada Form	80
... 40	6.4 Fasilitas pada form Control	81
... 41	6.4.1 Label	81
.. 41	6.4.2 Text Box	82
.. 42	6.4.3 Menspesifikasi format berbasis karakter	83
.. 44	6.4.4 Menetapkan Tipe Data pada Kontrol Text Box	84
.. 44	6.4.5 Command Button	84
. 47	6.4.6 Combo Box	86
. 47	6.4.7 Kotak Edit (Edit Box)	87
. 47	6.4.8 GRID	88
. 48	6.4.9 Kotak List (List Box)	90
. 51	6.4.10 Object Option Group	92
. 51	6.4.11 Object Citra (Image)	93
. 52	Latihan Praktikum VI	95
. 54	PERTEMUAN 7 & 8 Mengelola Methode dan Event Form	97
. 55	7.1 Memahami Objek	97
55	7.2 Properties pada Object	97
56	7.3 Metode	98
56	7.4 Event	99
57	7.5 Mendefinisikan Methode	100
57	7.6 Mengakses Properties	100
58	7.6.1 Jalan Pintas untuk Mengakses Objek	101
59	7.6.2 Objek _VFP	101
59	7.6.3 Pengaksesan Relatif	102
60	7.7 Memanggil Metode	103
63	7.7.1 Bereaksi Terhadap Event	103
63	7.7.2 Mengedit Metode dan Event Objek	103
63	7.7.3 Menciptakan Metode Baru	104
63	7.7.4 Memanfaatkan procedure AfterRowColChange	109
64	Latihan Praktikum VII	110
65	PERTEMUAN 9 Mendesain Form Supplier (Tugas Mandiri)	113
66	PERTEMUAN 10 Mendesain Laporan (<i>Report</i>)	115
67	10.1 Memahami Laporan	115
68	10.2 Menciptakan Laporan Baru	115
68	10.2.1 Menambahkan Title Band pada Laporan	116
69	10.2.2 Menspesifikasi Lingkungan Data	116
70	10.2.3 Menspesifikasi Toolbar Laporan	117
71	10.2.4 Menambahkan control label pada Laporan	117
73		

PERTEMUAN 3 Perintah-perintah Manipulasi Data	3
3.1 Membuat Tabel	3
3.2 Menampilkan Data	3
3.2.1 Perintah LIST	3
3.2.2 Melihat Data dengan Browse pada Tabel	4
3.2.3 Menampilkan Data dengan Operator AND, OR	4
3.3 Mencari Sebuah Data dengan LOCATE	4
3.3.1 Mencari Sebuah Data	4
3.3.2 Mencari Sekelompok Data	4
3.4 Menghapus Data	4
3.4.1 Menghilangkan Tanda Hapus Record Data	4
3.4.2 Menghapus Record Secara permanent	4
3.5 Meng-Update isi field dengan Perintah Replace	4
Latihan Praktikum III	4
PERTEMUAN 4 Bekerja Dengan View dan Query	5
4.1 Konsep VIEW	5
4.2 Membuat sebuah Local View	5
4.3 Jenis kondisi penggabungan data (Join Condition)	5
4.4 QUERY	5
4.4.1 Konsep Query	5
4.4.2 Menutup dan Menyimpan Query	5
4.4.3 Menjalankan Query	5
4.4.4 Memodifikasi Query	5
4.4.5 Mengurutkan hasil Query	5
4.4.6 Mengelompokkan Hasil Query	5
4.4.7 Menambahkan Ringkasan pada Pengelompokan Data	5
4.4.8 Mengarahkan Hasil Query	5
4.4.9 Mengarahkan Hasil Query ke Cursor	5
Latihan Praktikum IV	6
PERTEMUAN 5 Perintah LOGIKA	6
5.1 Perintah Percabangan	6
5.1.1 Perintah IF..ENDIF	6
5.1.2 Fungsi IIF()	6
5.2 Perintah DO..CASE	6
5.3 Perintah Perulangan	6
5.3.1 Perintah FOR .. ENDFOR	6
5.3.2 Menampilkan dalam bentuk Array	6
5.3.3 Perintah EXIT	6
5.3.4 Perintah Perulangan DO..WHILE	6
5.3.5 Perintah SCAN..ENDSCAN	6
5.3.6 Perintah WITH..ENDWITH	7
5.4 Membuat Program dengan Editor Program	7
Latihan Praktikum V	7

10.2.5	Menambahkan Kontrol Field	118
10.2.6	Menambahkan Gabungan Field ke dalam Laporan.....	120
10.3	Menambahkan Variabel pada Laporan	120
10.4	Men-Subtotal Kelompok Record	121
10.5	Menset Layout Halaman Cetak	122
10.6	Mencetak Laporan	122
10.6.1	Mencetak Laporan ke Printer	123
10.6.2	Mencetak Laporan ke File.....	124
10.6.3	Mem-Preview Laporan.....	125
	Laporan Praktikum X	125
	PERTEMUAN 11 Menampilkan Laporan Lewat FORM	127
11.1	Form Pencetakan Data	127
11.2	Membuat Function Cetak	130
	Latihan Praktikum XI	132
	PERTEMUAN 12 Tugas Mandiri Mendesain Laporan Supplier	135
	PERTEMUAN 13 Mendesain Menu	137
13.1	Memahami Menu	137
13.2	Membuat Menu Baru	138
13.3	Option-option pada Menu View	141
13.4	Generate Menu	142
13.5	Mengaitkan Item Menu dengan perintah tertentu	143
13.6	Menciptakan Tombol Pintas	143
13.5.1	Langkah menciptakan Tombol Pintas	144
13.7	Mensetting property Enable dan Disabled Item menu	144
13.8	Mengaitkan Menu SDI dengan Form Utama	145
	Latihan Praktikum XIII	146

LAMPIRAN

PERTEMUAN 1

Instalasi Software VFP 8.0 dan Konsep Pemrograman

Pokok Bahasan :

- ✓ Instalasi Visual Foxpro 8.0
- ✓ Setting Konfigurasi
- ✓ Konsep Pemrograman dan Tipe Data

- TIU** : Mahasiswa dapat membuat aplikasi VFP dengan tabel tunggal dalam form beserta laporan dan diaplikasikan ke dalam menu.
- TIK** : Mahasiswa dapat melakukan instalasi software visual Foxpro 8.0 dan mengatur setting konfigurasi secara benar.

1.1 Instalasi Software

Untuk melakukan instalasi Software Visual Foxpro 8.0 ke dalam komputer, ada beberapa persyaratan teknis yang minimal tersedia dalam komputer, antara lain :

- **Komputer:** IBM-compatible dengan Pentium class processor
- **Peripherals:** Mouse atau pointing device
- **Memory:** 64 MB RAM (128 MB atau lebih tinggi dianjurkan)
- **Hard disk space:**
 - Visual FoxPro Prerequisites: 20 MB
 - Visual FoxPro Typical Install: 165 MB
 - Visual FoxPro Maximum Install: 165 MB
- **Video:** 800 x 600 resolution, 256 colors (High color 16-bit recommended)
- **Operating system: Developing applications dengan Visual FoxPro 8.0 support pada** Microsoft Windows 2000 Service Pack 2, dan Windows XP. Anda dapat membuat dan mendistribusikan aplikasi untuk Windows 98, Windows Me, Windows 2000 Service Pack 2 or later, dan Windows XP.

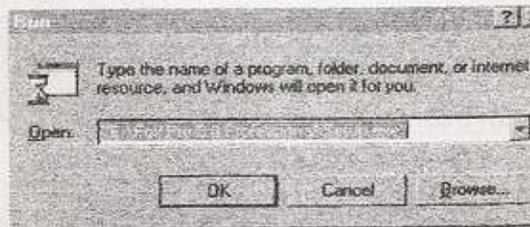
Catatan : Installation pada Windows NT 4.0 Terminal Server Edition tidak support.

Dengan persyaratan minimal tersebut diatas, sistem akan berjalan baik apabila proses instalasi berhasil.

Tahap pertama untuk melakukan instalasi sebaiknya software Visual Foxpro tersebut dicopy kan dulu ke dalam hardisk komputer anda, hal ini dilakukan untuk menghindari hambatan-hambatan yang muncul ketika melakukan instalasi. Salah satu yang terjadi jika kita langsung melakukan install dari CD, adanya permintaan Insert Disk 1 ketika terjadi proses pencopian data.

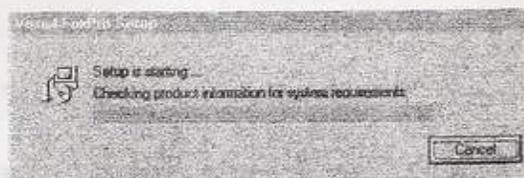
Kemudian setelah proses copy software selesai, ada beberapa tahapan yang akan dilakukan. Adapun langkah-langkah yang dilakukan sebagai berikut

1. Dari Menu Utama START, *click* kemudian pilih menu RUN, *click*
2. Kemudian muncul tampilan seperti dalam Gambar 1, dimana kita diminta menentukan file Setup yang akan kita install dalam komputer. Untuk contoh dibawah ini CD program berada pada Drive E:\Foxpro 8.0 Profesional. Untuk mencari directory program, *click* tombol Browse.
3. Kemudian pilih file SETUP.EXE yang ada pada direktori Visual Foxpro tersebut. Klik OK untuk melanjutkan.



Gambar 1.1 Memanggil File Setup.Exe

4. Pada proses selanjutnya akan muncul tampilan seperti dalam gambar 1.2, dimana terjadi proses pengecekan terhadap informasi system hardware yang ada pada komputer tersebut.

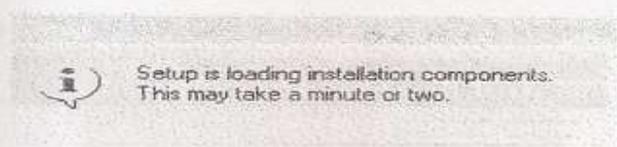


Gambar 1.2 Chek Komponen

5. Tahap selanjutnya setelah chek informasi, akan muncul tampilan berikut (Gambar 3) dimana Tahap 1, menunjukkan suatu proses untuk mengecek komponen atau software Foxpro yang sudah ada dalam system komputer. Untuk melanjutkan proses, klik menu 1 atau Prerequisites



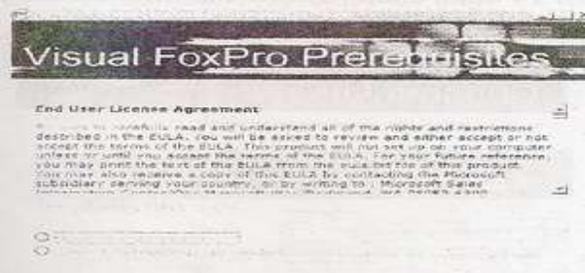
Gambar 1.3 Pilihan Setup



Gambar 1.4 Load Komponen

akan muncul sebagai proses loading componen yang akan diinstalasi dalam komputer, proses ini memerlukan waktu satu atau dua menit.

- 6. Tahap selanjutnya setelah loading komponen selesai, muncul display seperti yang tampak pada gambar 5. Dimana proses ini menanyakan kepada user, apakah setuju dengan kesepakatan yang ada dalam penggunaan software ini. *Jika user setuju, click pilihan pertama (I accept the agreement)* dan untuk melanjutkan proses instalasi *click Continue.*



Gambar 1.5 Persetujuan Lisensi

- 7. tahap ini, user diminta mengisikan password, dan konfirmasi ulang password. Kalo tidak menghendaki pengisian password, pada tanda chek Automatically log on dikosongkan.



Gambar 1.6 Isi Nama User



Gambar 1.7 Proses Instalasi

Setelah tahap diatas, selesai akan muncul pesan Restart Now atau Restart Later. Otomatis kalau dipilih restart computer akan restart otomatis, kalau dipilih later, dalam 14 detik akan restart. Dan muncul tampilan berikut ini :

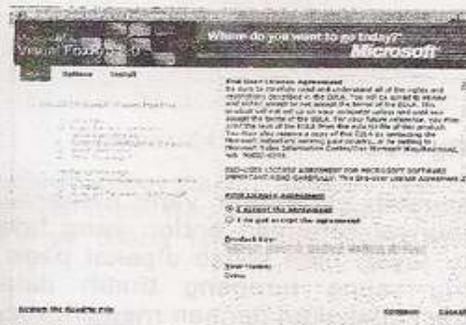


Gambar 1.8



Gambar 1.9 Pilihan setup

Pada tampilan gambar 1.9, Klik pilihan no 2 untuk instalasi selanjutnya



Gambar 1.10 Entry CD Key

Pada gambar 1.10, click Accept the agreement, dan klik continue untuk melanjutkan.



Gambar 1.11 Kebutuhan Space

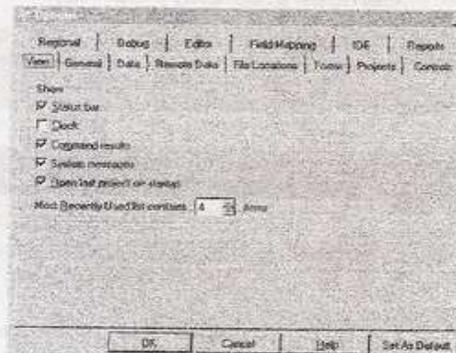
Klik **install now**, dan jika berhasil akan muncul pesan **Successfully installed**.



Gambar 1.12 Install Selesai

1.2 Setting Konfigurasi

VFP memberikan sejumlah parameter yang menentukan bagaimana cara kerjanya. VFP mempunyai sejumlah option yang tidak dapat ditampilkan sekaligus dalam satu layar, oleh sebab dipakai page frame style, artinya beberapa page yang saling tumpang tindih dalam sebuah jendela. Perubahan konfigurasi dilakukan dengan memilih menu **Tools|Option**



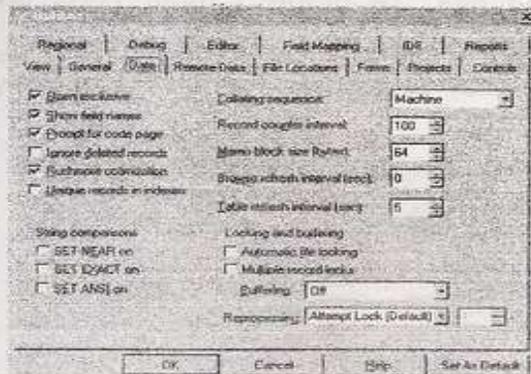
Gambar 1.13. Kotak Dialog Tools/Options

1.2.1 Option Data

Page Data dipakai untuk menyatakan :

- Pengaksesan data termasuk metode urutan pengaksesan

- Perbandingan pencarian string
- Parameter lock dan buffer untuk pengaksesan secara bersama-sama.
- Ukuran blok memo
- Rate dari refresh



Gambar 14. Option Data

Option **Open Exclusive** menyatakan apakah VFP akan membuka tabel dalam lingkungan single user atau multi user. Defaultnya tabel dibuka secara exclusive, artinya orang lain tidak boleh membuka tabel secara bersama-sama.

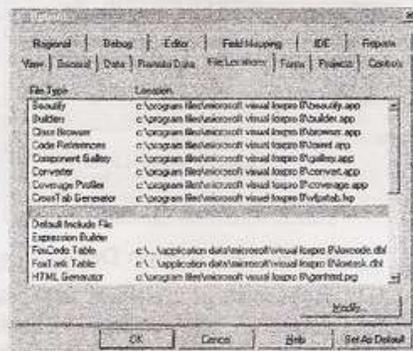
1.2.2 Option Field Mapping

Option ini dipakai untuk mengatur cara kerja form builder. Pada VFP sebelum versi 5.0, tipe object dari setiap tipe field sudah ditentukan yaitu sebuah text box. Jika anda menambahkan sebuah field numerik anda akan memerlukan objek lain misalnya sebuah spinner. Dalam kasus lain anda ingin memakai check box untuk field tipe logical. Dengan page *field mapping* anda dapat mengatur control default untuk tiap tipe field.



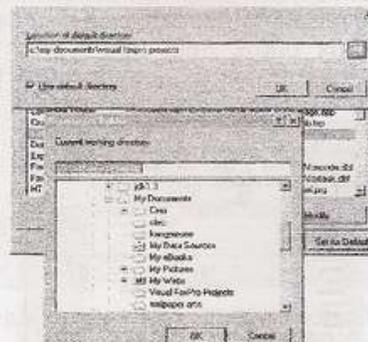
Gambar 1.15. Option Field Mapping

- **Drag and drop field caption.** Option ini dipakai agar VFP memakai nama field seperti caption dari class yang sedang dibuat.
- **Copy field comment.** Pilih option ini agar VFP menyalin komentar dari sebuah field ke property Comment dari class yang baru dibuat.
- **Copy field input mask.** Dipakai untuk mengkopli input mask dari sebuah field (jika ada) ke properti Input Mask dari class yang baru dibuat.
- **Copy field formset.** Dipakai untuk mengkopli spesifikasi format dari sebuah field (jika ada) ke properti Format dari class yang baru dibuat.



Gambar 1.16. Option File Location

Pada option diatas, posisi **Default Directory** belum digunakan, untuk melakukan setting direktori kerja, klik modify, kemudian aktifkan tanda check use default directory. Untuk menseleksi direktori kerja click tombol elipsis, kemudian pilih direktory aktif. Klik OK, kemudian Set As Default.



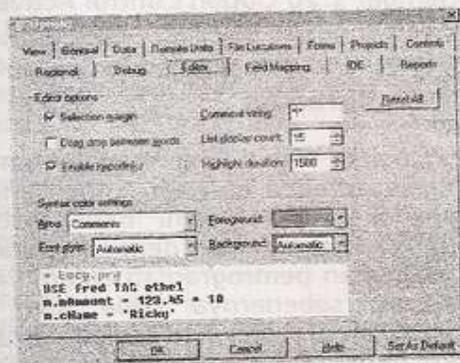
Gambar 1.17. Option Working Ditektori

1.2.3 Option Editor

Option ini memungkinkan Anda mengubah warna yang dipakai untuk menampilkan tipe-tipe teks pada saat anda bekerja dengan editor. Pemakaian warna membuat program lebih mudah dibaca. Anda dapat memberi efek warna pada kata kunci (*keyword*) variabel atau komentar.

Area. Elemen program yang akan diubah font dan warnanya. Pilihan area

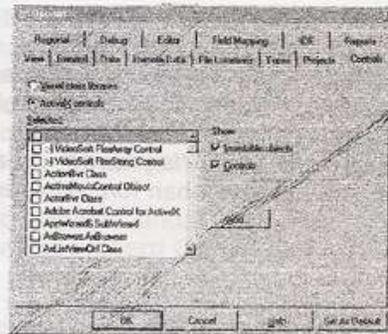
- Comments,
- Keywords
- Literals
- Normal, Operators, strings dan variables



Gambar 1.19. Option Editor

1.2.4 Option Controls

Option ini digunakan untuk menyeleksi penggunaan AxtiveX Control. Dengan melakukan pengaktifan terhadap pilihan ini maka dapat digunakan beberapa class yang sudah tersedia untuk digunakan. Misalnya penggunaan tanggal dengan model seperti setup windows, microsoft progress bar untuk membuat tampilan index.



Gambar 1.20 Option Control Activex

1.3 Konsep Dasar Pemrograman

Sebenarnya tujuan diciptakan Visual Foxpro adalah untuk mengurangi porsi pemrograman dalam menciptakan aplikasi. Apa yang dulunya harus kita lakukan dengan menuliskan baris demi baris instruksi dalam sebuah program, sekarang dapat kita lakukan dengan merancang form, laporan atau menu. Ini menyebabkan pemrograman dalam visual foxpro kelihatan tidak penting. Namun, sebenarnya pada level tertentu saat kita menciptakan form, menu atau laporan beberapa baris program harus kita sisipkan ke dalam form, menu atau laporan yang sedang dibuat. Ini menyebabkan penguasaan bahasa Foxpro tetap menjadi faktor yang penting dalam menciptakan aplikasi.

Program pada dasarnya tersusun dari dua bagian, yaitu :

- Bagian data yang diolah
- Bagian instruksi yang menyatakan cara mengolah data tersebut.

1.4. Data

Data dalam Visual Foxpro dibagi menjadi dua macam:

- Data permanent yang disimpan dalam field-field dalam table **.DBF**
- Data temporer yang disimpan di memori computer. Data yang disimpan di memori computer disebut juga **memvar** (*memory variable*)

Khusus untuk memvar yang digunakan untuk objek, Visual Foxpro memberikan nama khusus, yaitu property. Setiap data selalu memiliki tipe. Tipe data menyatakan cara komputer memperlakukan data tersebut. Sebagai contoh, jika suatu tipe data dinyatakan sebagai data numeric maka komputer akan mengetahui bahwa operasi penjumlahan, perkalian dan operasi matematis lainnya sah untuk data tersebut.

1.5 Tipe Data

Tabel 1.1 Tipe Data

Tipe	Keterangan	Ukuran	Rentang Data
Character	Alphanumeric text. Contoh, sebuah alamat customer	1 byte per character to 254	Semua characters
Currency	Jumlah uang. Contoh, harga dari sebuah item	8 bytes	- \$922337203685477.580 7 to \$922337203685477.580 7
Date	Chronological data dari bulan, hari, dan tahun	8 bytes	Date formats, {^0001-01-01}, January 1st, 1 A.D to {^9999-12-31}, December 31st, 9999 A.D.
Date Time	Chronological data dari bulan, hari, tahun, jam, minutes, and detik.	8 bytes	Date formats, {^0001-01-01}, January 1st, 1 A.D to {^9999-12-31}, December 31st, 9999 A.D., plus 00:00:00 a.m. to 11:59:59 p.m.
Logical	Boolean value of True or False	1 byte	True (.T.) or False (.F.)
Numeric	Integers atau bilangan decimal	8 bytes in memory; 1 to 20 bytes in table	- .9999999999E+19 to .9999999999E+20

Tabel 1.2 Tipe Data Khusus Field Visual Foxpro

Tipe	Keterangan	Ukuran	Rentang Data
Character (Binary)	Semua type data karakter	1 byte per character to 254	Any characters
Double	double-precision floating-point number	8 bytes	+/- 4.94065645841247E-324 to +/- 8.9884656743115E307
Float	Sama dengan Numeric	8 bytes dalam memory;	- .9999999999E+19 to .9999999999E+20

		1 to 20 bytes dalam tabel	
General	Referensi untuk sebuah OLE object Contoh, sebuah Microsoft Excel worksheet	4 bytes in table	Limited by available memory
Integer	Numeric value dengan tanpa decimals.	4 bytes	-2147483647 2147483647
Memo	Alphanumeric text untuk data panjang atau block data Contoh, catatan tentan daftar telepon	4 bytes dalam table	Limited by available memory

1.5.1 Tipe Karakter

Tipe karakter, yang sering pula disebut tipe data string, menyatakan data berupa deretan karakter alphabet A sampai Z, karakter angka 0-9 dan sejumlah karakter tertentu. Menyimpan data angka ke dalam tipe karakter memboroskan memori dan disk karena setiap karakter menempati ukuran satu byte sehingga informasi angka akan menyimpan sebanyak 1 byte pada data angka tersebut. Padahal jika angka yang sama disimpan dalam tipe bilangan numeric, hanya akan dihabiskan 4 byte saja. Jadi kita sebaiknya menggunakan data karakter hanya apabila diperlukan. Data karakter diciptakan dengan mengagip data yang dimaksud dengan salah satu dari tanda di bawah ini :

- Tanda kutip tunggal ` `
- Tanda kutip ganda " "
- Tanda kurung []

Untuk melakukan percobaan terhadap berbagai tipe data, dapat digunakan Command windows. Jika tidak aktif dapat dilakukan pengaktifan pilihan lewat pilihan : **menu Windows, Command Windows atau Ctrl + F2**
Maka akan ditampilkan windows seperti dibawah ini :



Gambar 1.21 Jendela Window Command

Berikut adalah contoh data karakter. Perhatikan bahwa perintah ? kita gunakan untuk menampilkan data yang dispesifikasikan di belakangnya ke layar.

Ouput dari perintah Command

```
DO FORM d:\penjualan\form1.scx
QUIT
? Belajar Foxpro'
? 'Visual Foxpro 8.0'
```

1.5.2 Tipe Numerik dan Bilangan Pecahan

Tipe data numeric (numeric atau float menyatakan bilangan - 0.9999999999999999E+19 sampai dengan 0.9999999999999999E+20). Data numeric paling cocok digunakan untuk menyimpan data-data yang perlu dijumlahkan, dikalikan dan dikurangkan. Untuk menuliskan data numeric, kita langsung menuliskan data yang dimaksud tanpa perlu menggunakan tanda apit tertentu. Perhatikan contoh :

```
? 7
? 7 + 1
&& 8
```

Bila kita mengapit data numeric dengan tanda ` ` , " " atau [] tanpa sengaja data tersebut akan diperlakukan sebagai data karakter dan bukan data numeric lagi.

```
? '7'
? '7' + '1'
&& 71
```

1.5.3 Tipe Bilangan Bulat

Tipe bilangan bulat (*integer*) menyatakan data bilangan bulat yang berkisar dari -2147483647 sampai dengan 2147483647. Manfaat tipe ini sebenarnya sama saja seperti tipe bilangan numeric. Bedanya operasi matematis terhadap data bilangan bulat sedikit lebih cepat daripada operasi matematis terhadap data bilangan numeric dan memori yang dipakai untuk menyimpan data bilangan bulat lebih sedikit.

1.5.4 Tipe Tanggal

Tipe tanggal (date) digunakan untuk menyimpan data tanggal dalam format yang berlaku saat ini. Formatnya adalah mm/dd/yy, jika SE CENTURY diset OFF, atau mm/dd/yyyy jika diset ON. Data tanggal ditulis dengan mengapitnya dengan tanda kurung { }. Sebagai contoh dibawah ini kita menciptakan dua data tanggal dan menampilkannya ke layer dengan perintah ?

```
? {12/25/90}          && {12/25/90}
? {08/17/45}          && {08/17/45}
```

menggunakan data tanggal perlu hati-hati. Bila tanggal yang kita spesifikasikan tidak sah, misalnya tanggal 12 bulan 25 maka otomatis foxpro akan mengosongkan data tersebut. Perhatikan :

```
? {25/12/95}          && { / / }
```

diatas karena tanggal {25/12/95} adalah tanggal yang tidak sah, sehingga foxpro mengosongkan hasil keluaran tanggal tersebut.. Jika kita menginginkan format tanggal menjadi dd/mm/yy di mana angka tanggal menjadi angka pertama, angka bulan menjadi angka kedua, kita dapat mengubah dulu format tanggal menjadi format BRITISH

```
SET DATE BRITISH
? {25/12/95}          && {25/12/95}
```

1.5.5 Tipe Logika (L)

Tipe logika (*logical*) digunakan untuk menyatakan data yang mengandung dua kondisi, seperti ya/tidak atau benar/salah. Data logika dispesifikasikan dengan cara diapit dengan tanda titik (.). Untuk menyatakan kondisi benar/ya, digunakan symbol .T., .t., .Y. atau .y., sedangkan untuk menyatakan kondisi salah/tidak dapat digunakan symbol .F., .f., .N. atau .n.. Dalam penerapan sehari-hari, data logika dapat kita gunakan, antara lain untuk menyimpan informasi mengenai status, misalnya jenis kelamin status lunas atau belum.

Berikut contoh data logika :

```
? .F.          && .F.
? .n.          && .F.
? .Y.          && .T.
```

1.5.6 Tipe Data Memo (M)

Tipe memo digunakan untuk menyatakan data berupa karakter yang sangat panjang dan hanya berlaku untuk field dan tidak berlaku untuk memvar. Field memo mirip dengan field karakter. Namun berbeda dengan karakter, memo berukuran tidak tetap dan disimpan dalam file teks yang terpisah dari file .DBF

Pada waktu kita mendefinisikan field memo dalam suatu file .DBF, field memo tersebut sendiri hanya berukuran 4 byte yang berisi pointer (penunjuk) memo. Isi dari memo sendiri disimpan dalam file teks terpisah yang normalnya memiliki ekstensi .FPT. Empat byte pointer memo berguna untuk menunjukkan lokasi relative memo dalam file .FPT. Sebaliknya, jika kita mendefinisikan field karakter, maka yang disimpan dalam file .DBF adalah isi dari field itu sendiri.

1.5.7 Tipe Umum (G)

Tipe data *general* (G) menyerupai tipe data memo. Bedanya tipe data *general* selain dapat menyimpan data karakter teks seperti halnya memo, juga dapat menyimpan data OLE (*Object Linking and Embedding*) seperti data spreadsheet, grafik dan suara yang diciptakan aplikasi lain. Sama seperti pada waktu kita mendefinisikan field *general*, sebuah pointer (penunjuk) OLE berukuran 4 byte diciptakan dalam field tersebut dimana pointer tersebut dipakai untuk mencatat lokasi di mana file OLE berada.

1.6 Mengolah data dengan Operator

Untuk melakukan operasi terhadap berbagai tipe data, khususnya tipe data numeric, tanggal dan karakter kita memerlukan operator. Operator - operator yang umum dikenal dalam visual Foxpro tertera pada Tabel di bawah ini :

Tabel 1.3. Jenis Operator

Operator	Fungsi	Berlaku untuk tipe	Contoh
=	Pengujian kesamaan	Semua tipe	? x = 9
+	Penjumlahan	Numeric, Character, Date, dateTime	? "Fox" + "pro" ? date() + 3
! atau NOT	Negasi / kebalikan	Logical	? ! .T.
*	Perkalian / Pembagian	Numeric	? 5 * 5, ? 8/4
^	Perpangkatan	Numeric	? 2 ^ 2

Kita harus selalu menggunakan tipe data yang sama pada waktu melakukan operasi. Sebagai contoh, baris program berikut menyimpan dua buah data numeric ke dalam dua buah memvar.

Tanda n sengaja kita lewatkan di depan kedua nama memvar untuk menunjukkan bahwa keduanya bertipe numeric.

```
N_var1 = 123
N_var2 = 456
```

Baris berikutnya menyimpan dua buah data karakter ke dalam dua buah memvar yang lain. Tanda c sengaja kita letakkan di depan kedua nama variable untuk menunjukkan kedua data ini bertipe character.

```
C_var1 = "123"
C_var2 = "456"
```

Sekarang kita lakukan operasi + yang akan menghasilkan data sebagai berikut :

```
? n_var1 + n_var2           && menghasilkan 579
? c_var1 + c_var2           && menghasilkan 123456
```

1.7 Menggunakan Function

Functions mengembalikan sebuah data yang spesifik. Sebagai contoh functions STR() dan VAL(), digunakan dalam bahasan berikutnya untuk mengembalikan nilai character dan numeric values. Disini ada lima metode untuk memanggil sebuah Visual FoxPro function:

- Menandai nilai balikan dari sebuah function kepada sebuah variabel. Baris dibawah ini adalah menyimpan sebuah current system date pada sebuah variabel yang dinamakan dToday:
dToday = DATE()
- Memasukan function yang dipanggil di dalam sebuah perintah Visual FoxPro. Baris perintah ini adalah men-set default directory terhadap nilai balikan dari GETDIR() function:
CD GETDIR()
- Mencetak / atau menampilkan nilai dari jendela output aktif. Baris perintah di bawah ini merupakan cetakan dari sistem waktu yang akan ditampilkan dalam output aktif window :
? TIME()
- Memanggil function tanpa melakukan penyimpanan nilai balikan. Perintah ini memanggil cursor off:
SYS(2002)
- Memanggil function dalam function lainnya. Baris perintah berikut adalah menampilkan urutan hari dalam tanggal yang dipanggil function :

1.8 Jenis File dalam Foxpro

Di dalam pemrograman Visual Foxpro terdapat beberapa file yang merupakan element pendukung maupun file bentukan dari sebuah aplikasi yang diciptakan. Adapun jenis-jenis file tersebut dapat dilihat dalam tabel berikut :

Tabel 1.4 Jenis File di Visual Foxpro

Extensi	Jenis File
.act	Document Wizard diagram
.app	Generated application atau Active Document
.cdx	Compound index
.chm	Compiled HTML Help
.dbc	Database
.dft	Table
.dbg	Debugger configuration
.dct	Database memo
.dnc	Database index
.dll	Windows Dynamic Link Library
.err	Compilation error
.esl	Visual FoxPro support library
.exe	Executable program
.fox	Macro
.lib	Visual FoxPro Dynamic Link Library
.fmt	Format File
.jct	Table memo
.jrt	Report memo
.jrx	Report atau Laporan
.obj	Compiled program
.h	Header file (for inclusion in a Visual FoxPro atau C/C++ program)
.hlp	WinHelp
.htm	HTML
.idx	Index, compact index
.lbt	Label memo
.lbx	Label

.log	Coverage log
.lst	Documenting Wizard list
.mem	Variable save
.mnt	Menu memo
.mnx	Menu
.mpr	Generated menu program
.mpx	Compiled menu program
.ocx	ActiveX control
.pjt	Project memo
.pjx	Project
.prg	Program
.qpr	Generated query program
.qpx	Compiled query program
.sct	Form memo
.scx	Form
.spr	Generated Screen Program
.spx	Compiled Screen Program
.tbk	Memo backup
.txt	Text
.vct	Visual class library memo
.vcx	Visual class library
.vue	FoxPro 2.x view
.win	Window file

Latihan Praktikum I

- Praktekan perintah setting konfigurasi dan perintah operasi type diatas.

PERTEMUAN 2

Project Manager, Database dan Tabel

Pokok Bahasan :

- ↳ Bekerja Dengan Project Manager
- ↳ Mengelola Database
- ↳ Mengelola Tabel

☐ : Mahasiswa dapat membuat project baru, menambahkan database dan tabel ke dalam project serta mampu mengolah data yang ada dalam tabel.

2.1 Project Manager

Project Manager merupakan fasilitas VFP yang sangat berguna untuk mengorganisasikan komponen-komponen yang menyusun sebuah aplikasi. Dengan menempatkan setiap aplikasi pada proyek tersendiri, Anda tidak perlu lagi mengingat-ingat komponen apa saja yang menyusun aplikasi tersebut. Melalui project manager juga dengan mudah dapat mengkompilasi komponen-komponen yang menyusun aplikasi. Anda bahkan dapat membentuk file *executable* (file EXE)

2.1.1. Membuat Project Baru

Project dapat dibuat dengan dua cara :

1. Lewat *command windows*, dengan menggunakan perintah **CREATE PROJECT [namaProject]**
2. Lewat *Menu* dengan langkah sebagai berikut :

1. Dari menu **File** menu, pilih **New**.

Kemudian setelah memilih **New file**, akan muncul pilihan dalam bentuk *radio button* dimana, masing-masing pilihan merupakan jenis objek yang akan kita buat. Adapun tampilan dari pilihan tersebut tampak dalam gambar :

2. Pilih **Project** dan pilih **New File**

PERTEMUAN 3

Perintah-perintah Manipulasi Data

Pokok Bahasan :

- Membuka Tabel
- Menampilkan Data & Mencari Data
- Mencari Sekelompok Data
- Menghapus Data dan Meng-update Data

23. Mahasiswa dapat mengoperasikan perintah-perintah untuk membuka tabel, menampilkan data, melakukan manipulasi data dan mencari data serta mampu mengupdate data.

Pertemuan ini membahas operasi-operasi yang sering dilakukan pada tabel, yaitu membuka tabel, melihat data di dalamnya dan mencari data. Dengan visual foxpro, cara membuka dan melihat data ditentukan cara penyimpanannya. Mulai dari VFP versi 5.0 hingga 8.0 Anda dapat menyimpan data sebagai tabel bebas (free table) atau bagian dari kumpulan tabel yang dinamakan dengan Database. Database memberikan keuntungan lebih dalam mengorganisasikan data dibandingkan dengan tabel bebas, tetapi bekerja dengan tabel yang terikat dengan database lebih sulit dibanding dengan tabel bebas.

23.1 Membuka Tabel

Sebelum menampilkan data yang ada didalam tabel Anda harus membuka tabel terlebih dahulu. Tabel dibuka pada sebuah area kerja. Setiap area kerja dapat menangani sebuah tabel beserta semua indexnya. VFP dapat menangani 32.767 area kerja. Sebelum membuka tabel, Anda harus memilih salah satu dari 32.767 area kerja yang ada. Kita akan memakai area yang sudah dibuat pada pertemuan 3. Untuk membuka tabel, Anda harus menetapkan dimana data tersebut dibuka. Untuk menetapkan posisi default digunakan perintah **SET DEFAULT TO**

Contoh :

```
SET DEFAULT TO E:\DATA_VFP
```

Contoh diatas menetapkan folder yang aktif pada e:\data_vfp.

PERTEMUAN 4

Bekerja Dengan VIEW dan QUERY

Pokok Bahasan :

- ↳ Konsep VIEW
- ↳ Membuat VIEW Local
- ↳ Query
- ↳ Jenis Penggabungan

TOK : Mahasiswa dapat mengoperasikan perintah-perintah untuk membuat view, menampilkan data, melakukan seleksi kriteria, pengurutan maupun pengelompokan data.

4.1 Konsep VIEW

View merupakan suatu cara untuk menseleksi field-field yang akan ditampilkan dari suatu tabel dan dengan suatu kriteria tertentu untuk memilih record yang akan ditampilkan, selain itu view dapat juga digunakan untuk menseleksi beberapa field dari beberapa buah tabel ditampilkan menjadi satu kesatuan. Prinsip dasarnya view adalah merupakan query tetapi data view dapat disertakan dan diaktifkan pada form ataupun laporan sedangkan query tidak dapat disertakan dalam form atau laporan.

View dapat menampilkan data untuk data lokal artinya data yang digunakan yang berasal dari komputer yang sedang digunakan (*local views*) ataupun untuk menampilkan data dari komputer lain seperti dari server atau workstation lain yang tersemabung dalam jaringan (LAN) sering disebut dengan **Remote Views**

Untuk menggunakan remote view, komputer anda harus terhubung pada server dengan teknologi LAN (*Local Area Network*), terlebih dahulu kita harus melakukan koneksi ke data server. Untuk remote view Microsoft Visual Foxpro dapat berkomunikasi data dengan Oracle atau SQL dan MySQL..

Anda dapat membuat dua jenis tipe view : **local** dan **remote**. Remote views menggunakan **remote SQL** syntax untuk menampilkan informasi dari tabel pada sebuah **remote ODBC data source**. **Local views** menggunakan Visual FoxPro SQL syntax untuk menampilkan informasi dari tabel atau views. Anda dapat menambahkan satu atau lebih remote views ke dalam sebuah local view, membuat ini memungkinkan anda untuk mengakses informasi dari Visual FoxPro dan remote ODBC data source dalam view yang sama.

PERTEMUAN 5 PERINTAH LOGIKA

Pokok Bahasan :

- ✓ Perintah Percabangan (IF..ENDIF, DO..CASE, IIF)
- ✓ Perintah Perulangan (DO..WHILE, SCAN.., FOR..)

TK : Mahasiswa dapat mengoperasikan perintah-perintah percabangan dalam mengelola aplikasi dan dapat menggunakan perintah perulangan dalam mengelola data dalam sebuah aplikasi.

Program adalah urutan jalannya sebuah aplikasi. Sebuah program yang ditulis dalam Foxpro pada prinsipnya memiliki alur program sekuensial, artinya program dieksekusi atau dijalankan mulai dari baris instruksi paling atas sampai baris instruksi paling bawah. Namun demikian, program yang berjalan secara sekuensial saja, tidaklah fleksibel. Untuk itu Foxpro menyediakan sejumlah mekanisme yang memungkinkan program berjalan secara sekuensial melainkan melompat ke sana-sini, bergantung kebutuhan. Pada prinsipnya hanya terdapat dua golongan perintah untuk mengontrol jalannya :

- Golongan perintah percabangan (*conditional branching*)
- Golongan perintah perulangan (*looping*)

5.1 Perintah Percabangan (*Conditional Branching*)

Perintah percabangan mengizinkan Anda untuk melakukan test kondisi dan juga, tergantung pada hasil dari test, dengan berbagai bentuk operasi. Ada dua kelompok perintah didalam Visual FoxPro yang dikenal dalam perintah percabangan :

- Perintah IF ... ENDIF
- Perintah DO CASE ENDCASE

5.1.1 Perintah IF...ENDIF

Perintah IF memiliki bentuk umum sebagai berikut :

```
IF IExpression [THEN]
  Commande True Instruction
[ELSE
```

PERTEMUAN 6 MENDESAIN FORM

Pokok Bahasan :

↳ Mengenal Form, Tool Box dan Properties

↳ Data Environment

↳ Membuat object baru pada Form

TUJUAN: Mahasiswa dapat merancang form dan menempatkan object pada form beserta pengaturan properties serta mengelola metode pengaksesan object.

6.1 Mengenal Form

Setiap aplikasi minimal mempunyai sebuah form. Form dapat dipakai untuk menampilkan informasi pada pemakai. Akan tetapi, pada umumnya form dipakai untuk berinteraksi dengan pemakai, artinya pemakai memasukkan data dan melihat hasilnya pada form.

Seiring perkembangan pemrograman sudah berubah, pada pemrograman konvensional program menentukan urutan dan apa yang dikerjakan oleh pemakai. Sekarang, pemakai mengontrol jalannya program. Pemrograman seperti ini merupakan pemrograman berorientasi objek. Pemrograman ini mempunyai 2 tipe dasar objek visual yaitu *container* dan *control*. *Container* adalah objek yang dapat menampung objek-objek lain. Form adalah sebuah container karena dapat menampung objek lain seperti text box, label, check box dan lain-lain.

Dalam perancangan form akan banyak menggunakan object yang disediakan dengan programming, sebelum bagaimana cara merancang form kita terlebih dahulu harus memahami suatu object berikut propertiesnya. Dengan menggunakan pemrograman yang berorientasi pada objek (OOP), akan selalu berhubungan dengan properties yang digunakan untuk melakukan setting yang kita butuhkan sesuai dengan keinginan kita. Setiap objek akan mempunyai properties yang berbeda. Dengan kata lain properties merupakan kumpulan dari parameter untuk menentukan nilai-nilai setting terhadap objek. Dalam Visual Foxpro **properties** akan dikelompokkan menjadi **All, Data, Methodes, Layout dan Other**.

- **All**, akan menampilkan semua jenis properties

PERTEMUAN 7 & 8

Mengelola Methode dan Event Form

Pokok Bahasan :

- ↳ Konsep Objek dalam Pemrograman
- ↳ Mendefinisikan Methode dan Event

TUJUAN : Mahasiswa dapat mengontrol program pada form serta menempatkan perintah prosedur secara tepat dan benar.

7.1 Memahami Objek

Untuk memahami konsep sebuah objek adalah dengan melihat form, tombol perintah, kotak cek, label dan beberapa kontrol lainnya. Dalam Visual Foxpro setiap form dan kontrol merupakan objek. Sebuah objek selalu memiliki apa yang disebut kelas. Sebuah kelas mengandung informasi mengenai karakteristik dari suatu objek dan objek adalah memiliki hubungan sangat dekat, tetapi keduanya tidaklah sama. Sebuah kelas terdiri dari informasi tentang bagaimana sebuah object seharusnya tampak. Sebuah kelas merupakan *blueprint* atau *schematic* dari sebuah objek. Jika komputer diibaratkan suatu objek, maka komponen elektronik tersebut merupakan kelas dari objek tersebut. Saat objek diciptakan, karakteristik yang terdapat dalam kelas digunakan untuk menyusun objek tersebut. Adapun komponen dari sebuah objek adalah properti, metode dan event.

7.2 Properties pada Object

Sebuah object memiliki properties yang pasti, atau pengenal. Sebagai contoh, sebuah komputer adalah memiliki bentuk dan ukuran yang pasti. Misalnya ukuran tinggi casing, jumlah drive, jumlah hardisk, tipe monitor dan sebagainya merupakan properti dari objek komputer desktop. Object yang anda ciptakan di dalam Visual FoxPro juga mempunyai properties yang berfungsi sebagai penunjuk dari class mana object tersebut berasal. Properties dapat di buat ketika saat mendesign atau ketika object tersebut sedang dijalankan, kita dapat mengubah properti objek dengan nilai baru yang kita kehendaki. Sebagai contoh, beberapa dari properties yang mengikuti sebuah object check box dapat dituliskan dalam sebuah tabel berikut :

PERTEMUAN 9

Mendesain Form Supplier (Tugas Mandiri)

Pokok Bahasan :

Pembuatan Form Supplier

Mengelola Procedure

Tugas Praktikum IX

Buat form untuk mengelola data Suplier dengan ketentuan sebagai berikut :

- Saat proses input data terdapat validasi data kembar
- Proses Edit dan Hapus dilakukan dengan bantuan tombol Top, Next, Bottom
- Tombol find, untuk mencari cepat
- Form dilengkapi dengan objek grid atau list.

The image shows a screenshot of a software interface for managing supplier data. It consists of two main parts: a data entry form and a data grid.

Data Entry Form:

Kode	11
Nama Suplier	12
Alamat	13
Kota	14
Telepon	15
Kontak Person	16

Data Grid:

Kode	Name Suplier	Alamat
11	12	13

Navigation buttons are located at the bottom right of the grid: a left arrow, a right arrow, and a refresh/clear button.

Pokok Bahasan :

- Bekerja Dengan Report Designer
- Membuat variabel baru

TK : Mahasiswa mampu membuat laporan dengan report designer serta dapat menambahkan variabel total, no urut dan melakukan operasi logika pada laporan .

10.1 Memahami Laporan

Untuk mencetak laporan dalam Visual Foxpro, yang pertama dilakukan adalah mendesain laporan yang kita perlukan dengan Report Designer. Sebuah laporan yang lengkap secara prinsip dapat terdiri dari beberapa bagian sebagai berikut :

- Bagian judul (*title*)
- Bagian header halaman (*page header*)
- Bagian grup (*group header*)
- Bagian rincian (*detail*)
- Bagian footer grup (*group footer*)
- Bagian footer halaman (*page footer*)
- Bagian ringkasan (*summary*)

Untuk membuat sebuah laporan yang sederhana secara cepat , click **Quick Report** pada **Report** menu. Quick Report menampilkan kepada anda hasil fields dan layout yang anda kehendaki pada laporan, sehingga ketika anda jalankan akan menampilkan data dari tabel yang anda pilih. Ketika *Report Designer* dalam kondisi dibuka, anda dapat mengatur *foreground* dan *background colors* dari elemen laporan yang dapat ditambahkan dari menu. Dalam modus standar, Report Designer menampilkan tiga bands yaitu *Page Header*, *Detail*, dan *Page Footer*. Sebuah garis separator adalah batas bawah dari tiap-tiap band. Type band memperlihatkan lokasi tempat penulisan atau penempatan data ditunjukkan dengan sebuah panah berwarna biru yang mengarah ke atas.

10.2 Menciptakan Laporan Baru

Untuk menciptakan laporan baru, kita harus membuka dulu Report Designer. Langkah-langkah untuk menjalankan Report Designer :

PERTEMUAN 11

Menampilkan Laporan Lewat FORM

Pokok Bahasan :

- ☞ Mendesain Form Bantu Cetak Laporan
- ☞ Membuat Function Cetak

TK : Mahasiswa mampu membuat laporan dengan report designer serta dapat mendesain form bantu untuk mencetak berdasar kriteria tertentu dalam sebuah aplikasi.

Laporan yang telah kita buat pada bahasan sebelumnya sifatnya global dalam menampilkan data-data yang ada dalam suatu tabel, sehingga laporan model seperti ini menjadi kurang efisien dari segi informasi yang akan diperoleh. Dalam laporan model ini juga tidak dapat memberikan keluaran yang sifatnya khusus atau tertentu. Untuk itu diperlukan sebuah form sebagai media perantara antara user dengan laporan yang akan ditampilkan.

11.1 Form Pencetakan Data

Untuk melakukan suatu pekerjaan mencetak laporan yang di panggil lewat sebuah form bantu, maka dapat kita ciptakan form yang akan menjadi perantara untuk melakukan suatu pilihan pencetakan. Sebagai contoh kita ingin mempunyai suatu pilihan terhadap data barang yang akan dicetak berdasarkan Nomor barang, maka desain form yang kita buat sebagai berikut :

The image shows a screenshot of a form titled "LAPORAN STOCK BARANG". The form has a dotted border and contains the following elements:

- Header: **LAPORAN STOCK BARANG**
- Field: **No. Barang**
- Range selection: **Dari** [11] **Sampai** [12]
- Buttons: **Cetak** and **Selesai**

Gambar 11.1 Form Bantu Cetak Stock

PERTEMUAN 12

TUGAS MANDIRI

MENDESAIN LAPORAN SUPPLIER

1. Buat laporan untuk menampilkan data Supplier dengan format berikut dengan berdasarkan **Group** Kota :

Report Designer - Konsolid

LAPORAN DATA SUPPLIER
CV AMERGO SENENG

KODE	NAMA	ALAMAT
------	------	--------

▲ Page Header

KOTA :

▲ Group Header 1:kota

<input type="text" value="kdplg"/>	<input type="text" value="nmplg"/>	<input type="text" value="alamat"/>
------------------------------------	------------------------------------	-------------------------------------

2. Tambahkan variabel No. Urut pada laporan diatas, sehingga laporan tersebut terdapat nomor urut data.
3. Buat laporan dengan menggunakan form bantuan untuk mencetak data per supplier. Objek pilihan kode supplier menggunakan kotak kombo, dengan mengambil data dari tabel supplier. (Gunakan perintah For). Adapun format formnya sebagai berikut :

Pilihan Cetak

Pilih Kode pelanggan Cetak

Tkode

Cetak di Layar
 Cetak di Kertas

Cetak
Batal

4. Buat laporan dengan menggunakan form bantuan untuk mencetak data

PERTEMUAN 13

Mendesain Menu

Pokok Bahasan :

- Bekerja Dengan Menu Designer
- Option Pada Menu View
- Membuat Program Menu

TK : Mahasiswa mampu membuat menu dan menggabungkan seluruh form dan laporan dalam menu aplikasi.

13.1 Memahami Menu

Menu dan toolbar menyediakan cara yang mudah bagi pemakai program untuk menjalankan tugas tertentu dari suatu program. Pada dasarnya menu dan toolbar memiliki fungsi yang sama. Bedanya sementara menu menampilkan teks sebagai informasi mengenai tugas yang akan dieksekusi apabila dipilih, toolbar menampilkan gambar sebagai informasi mengenai tugas yang akan dikerjakan apabila dipilih.

Menu designer adalah tool yang disediakan oleh VFP untuk membuat system menu sesuai standard windows. Menu VFP adalah menu Windows bukan object dari Foxpro. Apa pengaruhnya? Ada dua hal

- Anda tidak dapat mendefinisikan warna menu pada aplikasi VFP. Jika anda ingin mengubah warna menu, Anda harus mengubah warna pada Control panel dari Windows
- Setiap timer pada aplikasi tidak akan aktif pada saat menu aktif.

Istilah-istilah umum yang dipakai pada system menu :

- **Menu Bar** adalah level menu paling atas, item-itemnya akan selalu tampak selama aplikasi dijalankan
- **Menu pad** adalah level dibawah menu bar. Setiap nama item dalam menu bar menyatakan sebuah menu pad. Jika Anda klik sebuah menu Pad, akan ditampilkan sejumlah pilihan atau langsung menjalankan sebuah perintah
- **Prompt menu** adalah caption yang ditampilkan pada sebuah menu pop-up. Dengan click pada sebuah menu prompt, akan menjalankan sebuah perintah atau menampilkan submenu.

LAMPIRAN :

MESSAGEBOX() Function

Menampilkan sebuah kotak dialog untuk pemakai.

Bentuk :

```
MESSAGEBOX(eMessageText [, cTitleBarText][, nDialogBoxType ][, nTimeout])
```

Nilai balikan Numeric

Parameter

eMessageText

Menyatakan tulisan atau text yang tampak pada kotak dialog. Pemakaian sebuah tombol enter (CHR(13)) dalam *eMessageText* untuk memindahkan sebuah bagian dari sebuah pesan pada baris berikutnya dalam kotak dialog. Tinggi dan lebar dari kotak dialog adalah tergantung dengan penambahan pada kalimat atau text yang diinputkan dalam variabel *MessageText*.

Sebagai contoh pesan sederhana sebagai berikut :

```
? MESSAGEBOX(DATE())
```

nDialogBoxType

Menyatakan tombol dan ikon yang nampak dalam kotak dialog, nilai default tombol ketika kotak dialog ditampilkan dan menunjukkan sifat dari kotak dialog tersebut.

Dalam tabel berikut ini, kotak dialog bernilai antara angka 0 hingga 5 menyatakan spesifikasi tombol-tombol yang tampak dalam kotak dialog tersebut. Nilai gambar dari ikon tersebut antara lain bernilai 16, 32, 48, and 64 menyatakan spesifikasi masing-masing gambar yang muncul. Nilai default tersebut adalah 0, 256, dan 512 menyatakan spesifikasi dimana kotak tersebut dan nilai tombol.

Berikut ini nilai tabel dari kotak dialog tombol (*nDialogBoxType*) .

Value	Dialog box buttons
0	OK button only
1	OK and Cancel buttons
2	Abort, Retry, and Ignore buttons
3	Yes, No, and Cancel buttons
4	Yes and No buttons
5	Retry and Cancel buttons