

**MODUL PRAKTIKUM**

**PANDUAN PRAKTIS PEMROGRAMAN  
VISUAL FOXPRO**

**DIGUNAKAN UNTUK PRAKTIKUM  
MATAKULIAH PRAKTIKUM PEMROGRAMAN BASIS DATA**



**Disusun Oleh:  
Z A I D I R**

**PROGRAM STUDI MANAJEMEN INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS RESPATI YOGYAKARTA  
2013**

## KETENTUAN KEGIATAN PRAKTIKUM

### A. Kelengkapan Praktikum

1. Kegiatan praktikum dilengkapi dengan modul praktikum
2. Perlu memiliki flashdisk
3. Hasil kegiatan praktikum disimpan pada *folder* masing-masing (bisa dibuat pada *drive* komputer atau *flashdisk* atau dibuat di *server*).

### B. Tata Tertib Praktikum

1. Peserta praktikum (praktikan) harus memenuhi atau membawa kelengkapan praktikum setiap kali mengikuti kegiatan praktikum.
2. Untuk mengikuti praktikum harus berpakaian rapi dan sopan.
3. Pada saat kegiatan praktikum, peserta praktikum harus menjaga ketenangan, ketertiban, kebersihan dan kerapian.
4. Peserta praktikum harus datang tepat waktu, dengan waktu toleransi keterlambatan 15 menit.
5. Alat komunikasi dinyalakan dalam mode silent atau dimatikan.
6. Selama praktikum, peserta praktikum tidak diperkenankan meninggalkan ruangan praktikum/lab tanpa seizin pengampu/dosen praktikum.
7. Peserta praktikum harus bertutur kata yang baik dan sopan kepada pengampu/dosen praktikum atau asisten dosen (jika ada).
8. Peserta praktikum harus dapat menunjukkan sikap kejujuran. Apabila menemukan perlengkapan praktikum peserta praktikum lain yang tertinggal di komputer atau meja yang digunakan, maka dapat memberitahukan kepada petugas laboratorium untuk selanjutnya akan dibuat pengumuman.
9. Jadwal kegiatan praktikum yang karena sesuatu sebab tidak bisa dilaksanakan, maka akan dicarikan jadwal pengganti dihari lain berdasarkan kesepakatan pengampu/dosen praktikum dengan peserta praktikum.
10. Peserta praktikum harus memenuhi minimal kehadiran yaitu 75% dari total pertemuan.
11. Pada saat dilaksanakan ujian responsi, peserta praktikum diwajibkan menggunakan **jas almamater**.
12. Hal-hal yang belum diatur pada tata tertib ini akan diatur kemudian.

### C. Sangsi

1. Pengampu/dosen praktikum berhak memperingatkan bahkan mengeluarkan peserta praktikum yang tidak dapat menjaga ketenangan, ketertiban, kebersihan dan kerapian.
2. Apabila peserta praktikum datang lebih dari 15 menit, maka tidak diperkenankan mengikuti praktikum.
3. Peserta praktikum yang tidak memenuhi minimal kehadiran 75% dari total pertemuan, maka tidak diperkenankan mengikuti responsi/UAS. Dengan demikian maka nilai akhiri untuk mata kuliah praktikum adalah E.

#### **D. Penilaian**

Kegiatan praktikum akan dinilai dengan beberapa komponen sebagai berikut:

<b>No</b>	<b>Komponen Penilaian</b>	<b>Bobot Penilaian</b>
1	Praktikum Harian	30%
2	Tugas	30%
3	Responsi/UAS	40%

## DAFTAR ISI

	Hal
JUDUL	i
KETENTUAN KEGIATAN PRAKTIKUM	ii
DAFTAR ISI	iv
KATA PENGANTAR	v
BAB I Mengenal Visual FoxPro	1
BAB II Project dan Komponennya	15
BAB III Pengelolaan Form	23
BAB IV Bekerja dengan Database dan Free Table	35
BAB V Mengelola Data	47
BAB VI Merancang Form dan Mengaitkan Dengan Database	57
BAB VII Tipe Data, Variabel, Operator, Ekspresi, dan Algoritma Pengolahan Data	62
BAB VIII Pengujian, Perulangan, Prosedur dan Fungsi	79
BAB IX Pembuatan Form Transaksi	94
BAB X Membuat Laporan	99
BAB XI Laporan Tingkat Lanjut dan Form Laporan	112
BAB XII Menu, Form Menu, Form Login dan Program Utama	118
Daftar Pustaka	127

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah S.W.T, karena berkat Rahmat dan Karunia-Nya modul praktikum matakuliah “Pemrograman Basis Data” bisa diselesaikan. Tujuan dari penyusunan modul ini adalah agar kegiatan praktikum dapat berjalan dengan panduan yang jelas dan target pembelajaran dapat tercapai dengan maksimal.

Penyelesaian modul ini juga tidak terlepas dari peran pihak-pihak memberikan inspirasi dan dukungan. Pada kesempatan ini penulis tidak lupa mengucapkan terima kasih kepada:

1. Dekan Fakultas Sains dan Teknologi yang selalu memberikan dukungan agar setiap kegiatan praktikum memiliki modul.
2. Ketua Program Studi Manajemen Informatika yang telah memberikan arahan dan dukungan agar semua dosen memiliki modul dengan standard dan kualitas yang baik agar kualitas pebelajaran di Program Studi Manajemen Informatika dapat terjamin.
3. Semua rekan dosen Program Studi Manajemen Informatika yang telah memberikan inspirasi dan koreksi terhadap penyelesaian modul ini.
4. Semua pihak yang tidak bisa disebutkan satu persatu.

Penulis menyadari, bahwa modul ini masih banyak kekurangannya. Penulis berharap kritik dan saran pembaca demi peningkatan kualitas modul ini dimasa yang akan datang.

Akhirnya penulis mengucapkan, semoga modul ini bisa bermanfaat bagi siapa saja yang telah menggunakannya.

Yogyakarta, April 2013  
Penulis

Zaidir

# Bab 1

## Mengenal Lingkungan Visual FoxPro

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat menggunakan Visual FoxPro, mengenali lingkungan Visual FoxPro, mengetahui langkah-langkah yang harus dilakukan untuk membuat aplikasi dengan Visual FoxPro.

Pertemuan ini akan menjelaskan secara singkat tentang konsep pemrograman **Visual FoxPro** dan lingkungannya dan konsep-konsep penting dalam membangun sebuah aplikasi dengan menggunakan **Visual FoxPro** seperti model pemrogram **event-driven**.

### 1.1 Pengertian Pemrograman

Pemrograman merupakan sekumpulan instruksi-instruksi yang terorganisasi dan ketika dieksekusi, menyebabkan komputer berkelakuan seperti apa yang diinginkan oleh pembuat program. Tanpa program, komputer tidak berguna.

Program ibarat suatu resep, yang berisi daftar bahan-bahan (disebut *variabel*) dan petunjuk-petunjuk (disebut *statement*) yang memberitahu komputer apa yang harus dikerjakan dengan variabel-variabel. Variabel-variabel tersebut dapat berupa data numeric, teks, maupun gambar (*graphical images*).

### 1.2 Uraian Singkat tentang Visual FoxPro

Untuk menggunakan Visual FoxPro bisa bermacam-macam, tergantung instalasi program dari komputer yang sedang digunakan. Untuk modul yang dibahas ini menggunakan Visual FoxPro 9.0

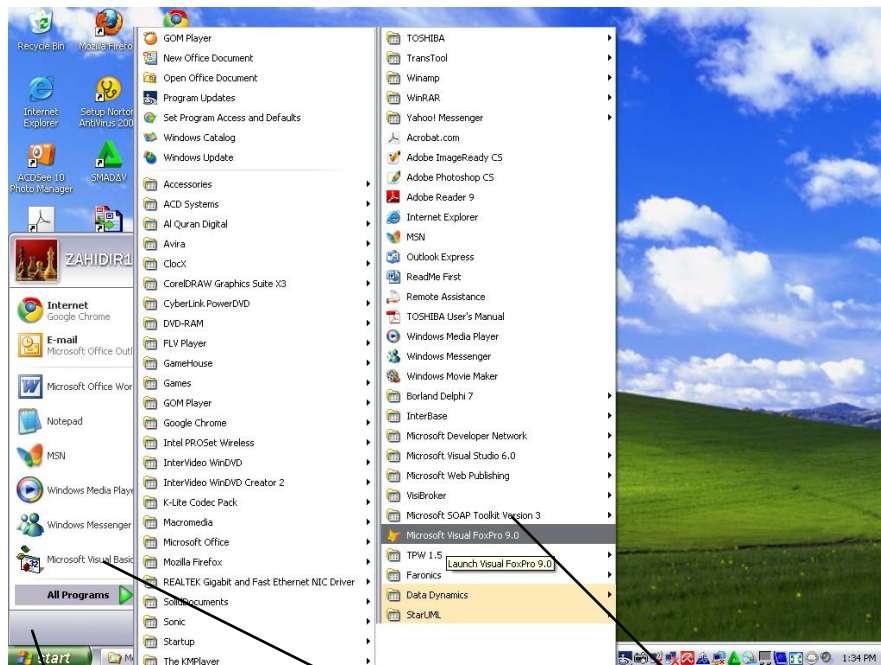
Visual FoxPro 9.0 adalah bahasa pemrograman yang digunakan untuk membuat sebuah aplikasi berhubungan dengan *database*. Kemampuannya dalam membuat aplikasi, membangun *database* dengan cepat dan mudah, menjadikan bahasa pemrograman ini banyak diminati. Selain menyediakan penanganan aplikasi data dengan kemampuan yang kuat dan cepat, Visual FoxPro 9.0 juga mempunyai fleksibilitas yang diperlukan untuk membangun semua jenis *database*.

Visual FoxPro merupakan Bahasa Pemrograman Visual dan *events driven*. Dalam Visual FoxPro, program dikerjakan dalam lingkungan grafis. Karena pengguna (*user*) dapat meng-*click* pada satu obyek tertentu secara random, maka setiap obyek harus diprogram secara independen dapat merespon terhadap tindakan-tindakan itu (*event*). Oleh karena itu, program Visual FoxPro dibuat dalam banyak subprogram, masing-masing mempunyai kode program sendiri-sendiri, dan dieksekusi secara independen dan pada saat yang sama masing-masing dapat di-*link* secara bersama-sama.

### 1.3 Membuka Visual FoxPro

Untuk membuka program Visual FoxPro, dapat mengikuti langkah di bawah ini :

1. Klik tombol **Start** pada *windows taskbar*.
2. Pilih **All Program**, sehingga akan ditampilkan semua aplikasi yang telah diinstal pada komputer.
3. Pilih atau sorot kemudian klik **Microsoft Visual FoxPro 9.0**.



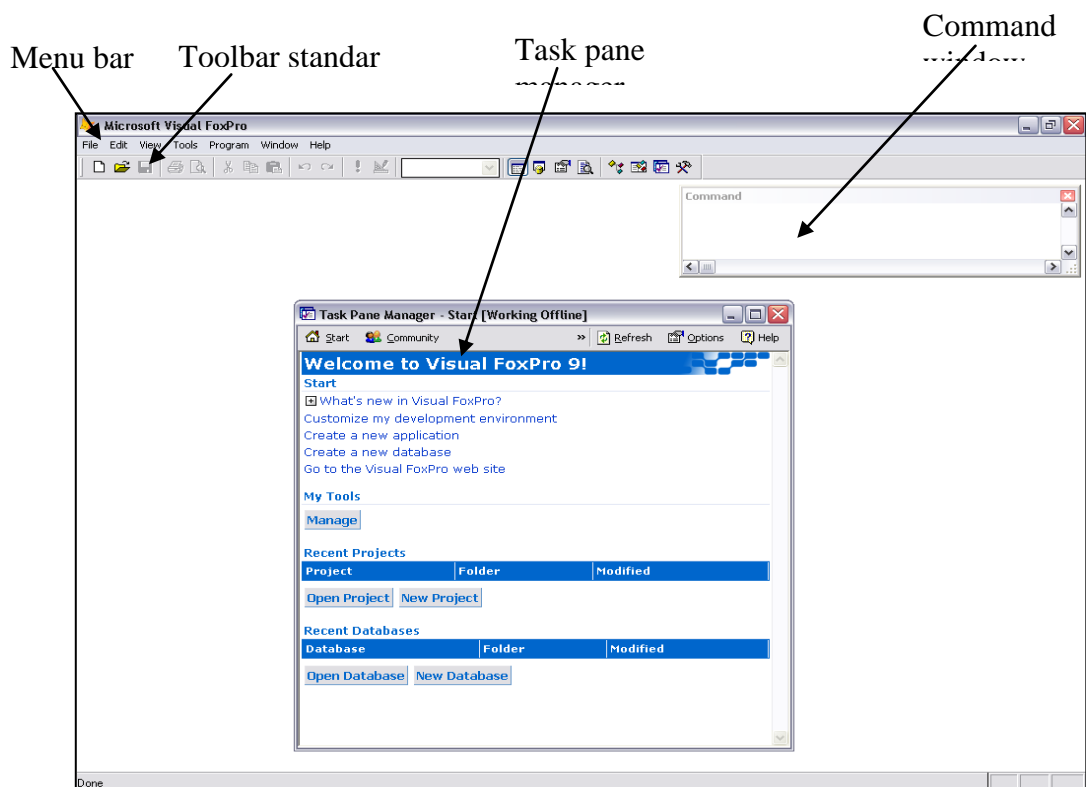
Langkah 1:  
Klik tombol Start

Langkah 2:  
Pilih All Program

Langkah 3:  
Klik Microsoft Visual FoxPro  
9.0

Gambar 1.1 Cara Membuka Visual FoxPro

Pada saat dipanggil pertama kali, maka akan ditampilkan area kerja seperti terlihat pada Gambar 1.2. Pada kondisi tampilan ini, ada beberapa hal yang perlu diperhatikan, khususnya elemen-elemen yang ditampilkan yaitu: **Menu bar**, **Toolbar standar**, **Task pane manager**, dan **Command window**.



Gambar 1.2 Tampilan awal

#### A. Menu Bar (menu batang)

Menu Bar adalah baris menu yang terdiri dari 7 menu utama yang masing-masing berisikan submenu dan perintah lagi. Menu Bar dapat dimanfaatkan untuk memberikan instruksi-instruksi yang diinginkan dalam melakukan operasi dengan Visual FoxPro.

Untuk menampilkan masing-masing pilihan menu dapat dilakukan dengan cara mengarahkan *mouse* pada menu yang bersangkutan, lalu diklik. Cara lainnya yaitu dengan cara menekan tombol ALT bersama-sama dengan tombol huruf awal dari masing-masing menu atau huruf yang diberi garis bawah.

#### B. Toolbar Standar (kumpulan tombol standar)

Toolbar Standar adalah kumpulan tombol berupa icon yang berisikan perintah. Dengan Toolbar Standar, maka akan mempercepat dalam memberikan instruksi dibanding menggunakan Menu Bar. Hanya saja tidak semua perintah yang tersedia pada Toolbar Standar.

Jika Toolbar Standar tidak tampil, maka dapat ditampilkan dengan cara mengklik menu View pada Menu Bar, lalu pilih atau tandai Standar dan klik tombol OK.

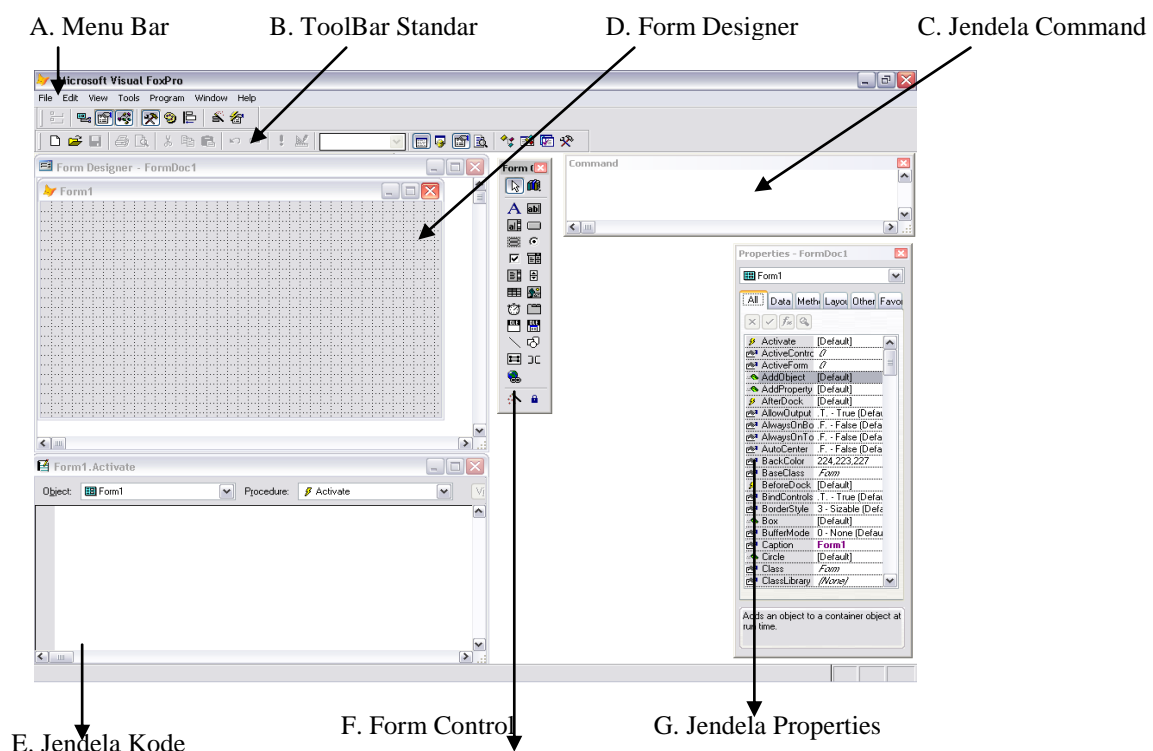
#### C. Task Pane Manager (pengelola panel operasi/tugas)

Task Pane merupakan layar yang disediakan untuk dapat melakukan tugas-tugas tertentu pada Visual FoxPro, misalnya membuat Project baru atau membuka Project yang telah ada. Fasilitas ini bisa digunakan atau bisa juga ditutup saja, karena perintah-perintah tersebut juga ada pada Menu Bar.

#### D. Command Window (jendela perintah)

Jendela perintah berfungsi sebagai tempat penginputan perintah secara langsung kepada program dari Visual FoxPro, dengan kata lain sebagai alternatif selain menggunakan Menu Bar dan Toolbar Standar.

### 1.4 Jendela Kerja Visual FoxPro 9.0

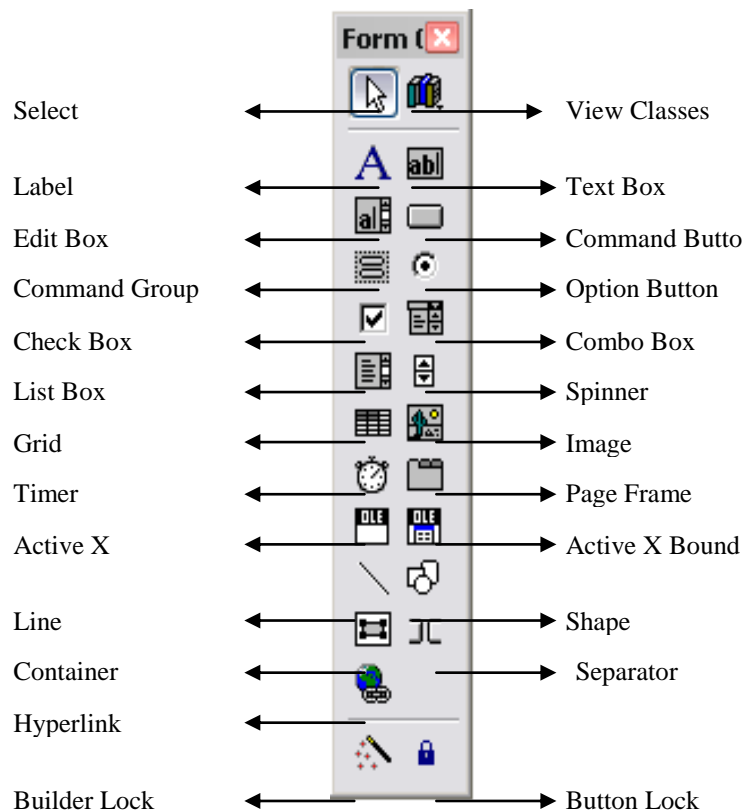


Gambar 1.3 Jendela Kerja Visual FoxPro 9.0

### 1.5 Form Controls

Form Control adalah kumpulan komponen yang digunakan untuk menambahkan teks atau kontrol input, tombol dan lainnya ke dalam *form*. Sangatlah penting untuk mengenal komponen, agar nantinya Anda tidak bingung dalam mencari dan menggunakannya (lihat Gambar 1.4)





Gambar 1.4 Komponen-komponen Form Control Visual FoxPro 9.0

- **Pointer** bukan sebuah kontrol, pointer digunakan untuk memilih control yang sudah ada pada sebuah form.
- **View Classes**, untuk menampilkan class-class yang pernah dibuat.
- **Label Control**, digunakan sebagai text static yang tidak bisa diedit, biasanya digunakan untuk memberi nama pada control yang lain seperti TextBox.
- **TextBox Control**, control yang berisi string dari karakter yang dapat diisi oleh user. Biasanya dimanfaatkan untuk mengisi data atau menampilkan data.
- **EditBox Control**, mirip seperti fungsi textbox, hanya saja dengan control ini data yang diisikan bisa menggulung barisnya, sehingga cocok untuk data-data yang cukup panjang. Cocok dimanfaatkan dengan jenis data Memo.
- **Command Button Control**, command button (tombol perintah) hampir muncul pada setiap form, biasanya digunakan untuk menangkap kejadian berupa mouse click.
- **Command Group Control**, sama seperti command button, hanya saja dengan control ini dapat diciptakan beberapa command button sekaligus.
- **Option Group Control**, digunakan dalam suatu group, dimana seorang user dapat memilih salah satu saja dalam suatu group.
- **Check Box Control**, control ini digunakan untuk pilihan ya/tidak atau benar/salah. Pilihan bisa lebih dari satu.
- **ComboBox Control**, control ini merupakan gabungan antara TextBox dan ListBox Control, dimana daftar item hanya akan ditampilkan jika user men-click pada panah kebawah disebelah kanan Control tersebut.
- **ListBox Control**, berisi sejumlah item dimana seorang user dapat memilih salah satu atau beberapa item.
- **Spinner**, dapat digunakan untuk mengelola data angka yang dapat diisikan dengan cara menambah atau mengurangi sebanyak satu dengan memanfaatkan tombol panah atas atau panah bawah yang ada disebelah kanan control ini.
- **Grid Control**, merupakan control untuk menampilkan data dalam bentuk baris dan kolom. Biasanya banyak dimanfaatkan untuk menampilkan record-record pada database.
- **Image Control**, merupakan control untuk menampilkan data-data berupa gambar.

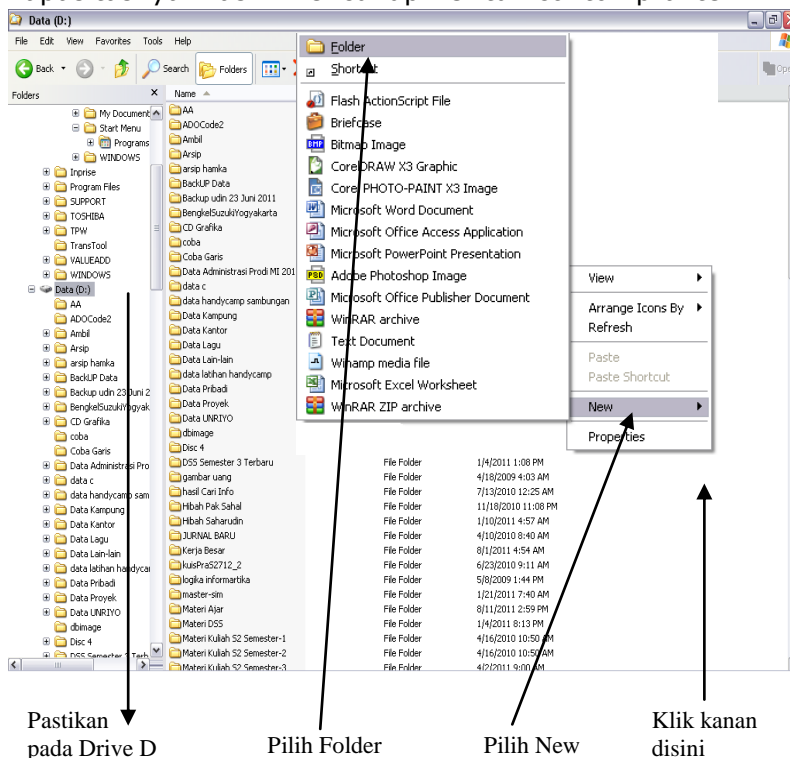
- **Timer Control**, merupakan control yang tidak terlihat pada saat program dijalankan. Control tersebut membangkitkan kejadian pada selang waktu tertentu.
- **Page Frame Control**, digunakan pemanfaatan multi halaman dalam satu form. Biasanya digunakan apabila data yang ditampilkan pada form cukup banyak, dimana bila digunakan kondisi form biasa area tidak mencukupi.
- **Active X Control**, digunakan sebagai tempat untuk program eksternal yang ada pada windows seperti spread sheet yang dihasilkan oleh Microsoft Excel. Dengan menggunakan control tersebut kita dapat menampilkan program lain pada sebuah aplikasi.
- **Active X Bound Control**, sama seperti Active X Control, hanya saja tidak otomatis.
- **Line Control**, digunakan untuk membuat garis sehingga dapat memperindah tampilan dari suatu form.
- **Shape Control**, digunakan untuk membuat objek berupa kotak, lingkaran, oval atau bentuk lain untuk mempercantik tampilan form.
- **Builder Lock Control**, digunakan untuk mengaktifkan fasilitas builder (proses otomatisasi untuk control-control tertentu, seperti Grid, Combo, dll).
- **Button Lock Control**, digunakan untuk mengunci kontrol yang akan digunakan sehingga bisa lebih cepat apabila kontrol yang akan digunakan jenis yang dengan jumlah yang banyak. Untuk mengembalikan ke kondisi normal, klik Pointer.

## 1.6 Praktek

### 1.6.1 Menyiapkan Folder Kerja

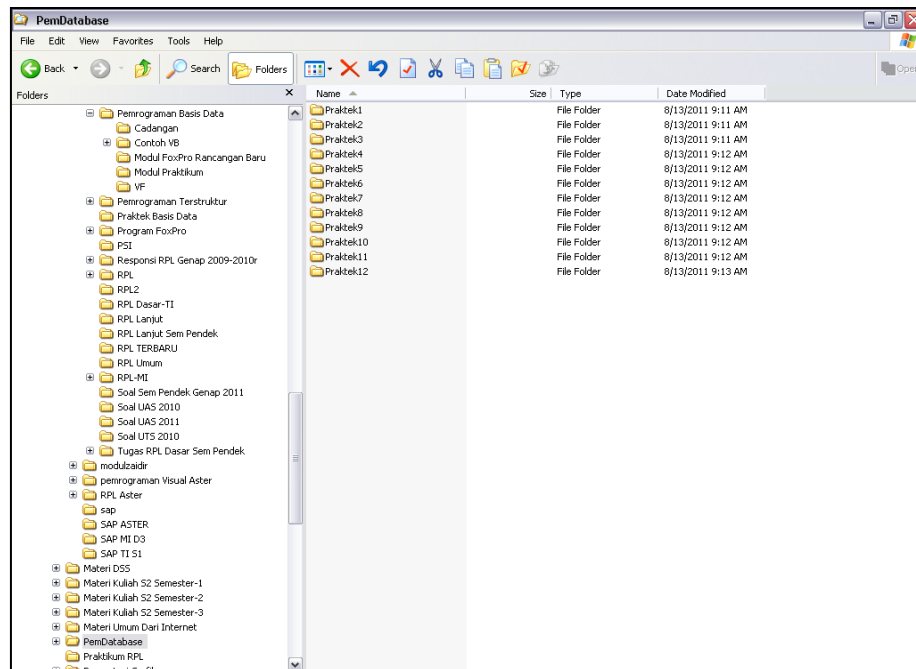
Folder kerja perlu disiapkan dengan seksama agar hasil praktek yang dilakukan dapat tersimpan dengan baik dan mudah untuk ditemukan kembali. Untuk keperluan praktek kita mulai saat ini sampai beberapa pertemuan berikutnya, maka siapkanlah folder kerja dengan langkah sebagai berikut:

1. Tampilkanlah window explorer dengan cara klik kanan pada tobol Start pada windows taskbar, kemudian pilih Explorer.
2. Setelah window explorer ditampilkan, klik drive (drive C, Drive D, drive E, dll) tempat folder/directory akan diciptakan. Sebaiknya pada drive yang aman dan kapasitasnya masih mencukupi. Untuk contoh praktek ini kita pakai drive D.



Gambar 1.5 Window Explorer

3. Klik kanan pada belahan bagian kanan window explorer pada bagian yang kosong, sehingga akan ditampilkan menu layang. Pada menu tersebut pilihlah New, lalu pilih Folder. Ketiklah nama folder yang diinginkan, misalnya **PemDatabase**.
4. Klik dua kali pada folder **PemDatabase**, lalu lakukan langkah 3 untuk menciptakan sub directory pada folder tersebut.



Gambar 1.6 Tampilan Window Explorer Saat Folder Kerja Sudah Terbentuk

5. Misalnya folder/sub directory yang diciptakan berturut-turut adalah: Praktek1, Praktek2, Praktek3, dan seterusnya.

### 1.6.2 Memulai dan menjalankan Visual FoxPro 9.0

Untuk membuka program Visual FoxPro, dapat mengikuti langkah di bawah ini :

1. Klik tombol **Start** pada windows taskbar
2. Pilih **All Program**, sehingga akan ditampilkan semua aplikasi yang telah diinstal pada komputer.
3. Pilih atau sorot kemudian klik **Microsoft Visual FoxPro 9.0**. Tunggulah beberapa saat sampai ditampilkan seperti gambar 1.2.

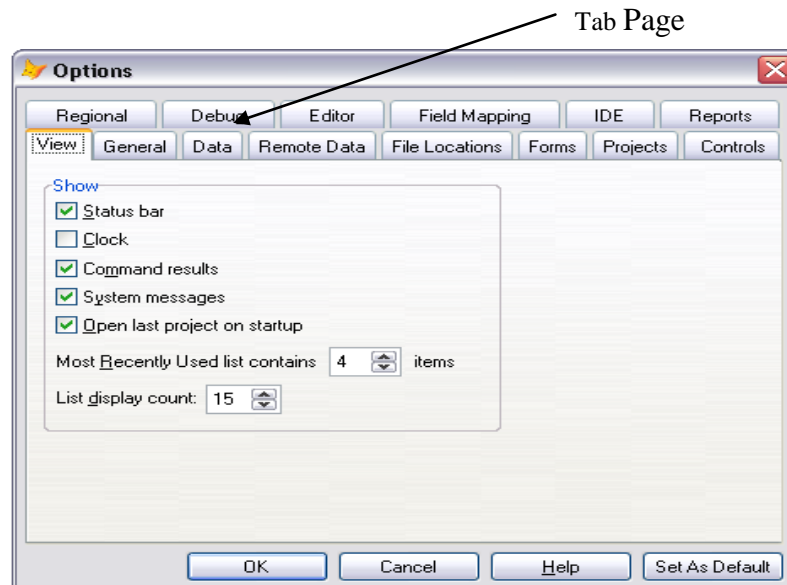
Pada saat tampilan awal tersebut, kita sudah dapat bekerja dengan Visual FoxPro 9.0, baik dengan perintah langsung (**Command Windows**) maupun dengan menggunakan menu (Menu Bar). Perintah langsung dapat diketikkan pada bagian dalam kotak perintah, misalnya Create Pegawai (untuk membuat tabel yang bernama Pegawai), dll.

Untuk tidak membingungkan, Task Pane dapat ditutup dengan mengklik tanda silang pada pojok kanan atas window-nya (tombol **Close**). Sebelum lebih jauh bekerja dengan Visual FoxPro, sebaiknya perlu dilakukan pengaturan-pengaturan dasar, sehingga dalam membuat project atau aplikasi dengan Visual FoxPro bisa lebih teratur dan sesuai keinginan.

### 1.6.3 Menentukan Aturan Dasar

Menentukan aturan dasar yang dimaksud disini adalah melakukan penetapan pada sistem, sehingga saat menyimpan, membuka ataupun efek-efek khusus dalam bekerja dapat ditentukan. Untuk itu ikutilah langkah-langkah sebagai berikut:

1. Klik menu **Tools**
2. Pilih menu **Options**, sehingga akan ditampilkan kotak dialog seperti Gambar 1.7



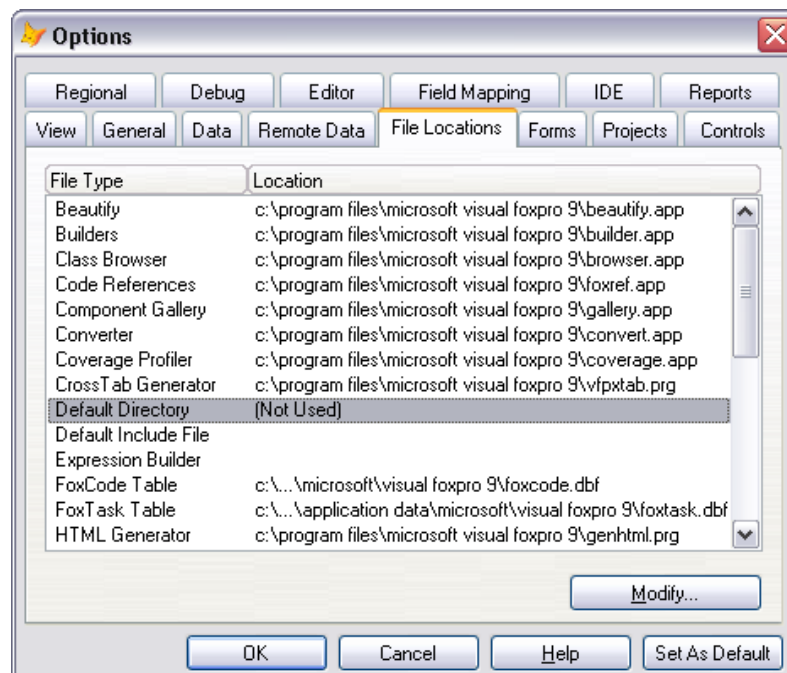
Gambar 1.7 Kotak Dialog Tools

3. Klik Tabs Page yang sesuai dengan keinginan untuk melakukan perubah-perubahan. Beberapa Tabs Page yang cukup penting yaitu: **File Location**, **Regional**, dan **Form**. Setelah selesai melakukan semua perubahan yang diinginkan, akhiri dengan cara mengklik tombol **Set As Default** dilanjutkan dengan tombol **OK**.

#### A. Mengarahkan Direktori Kerja

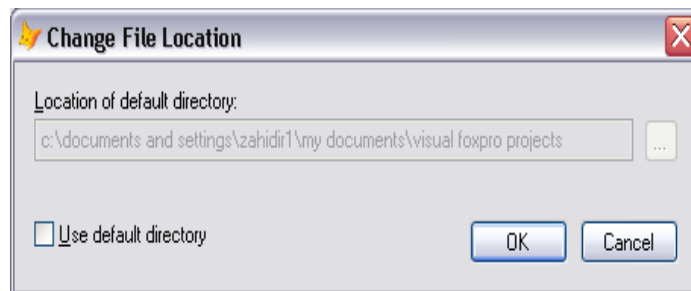
Proses ini perlu dilakukan agar proses dalam bekerja dengan project atau aplikasi tetap terarah.

- 1) Klik Tabs Page **File Location**
- 2) Klik pada baris **Default Directory**
- 3) Klik tombol **Modify**



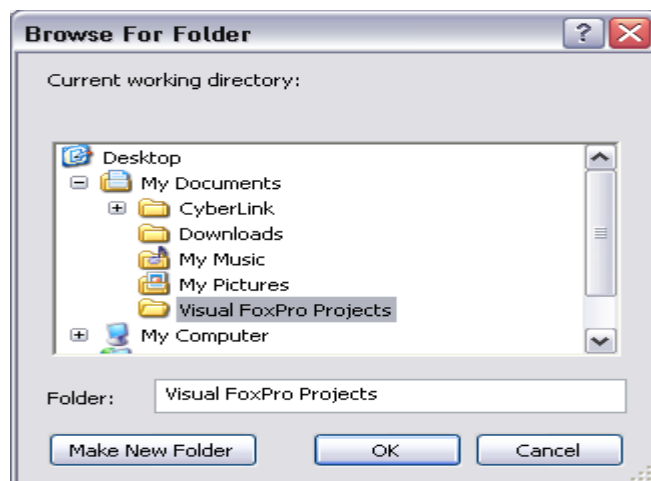
Gambar 1.8 Mengarahkan Direktori Kerja

- 4) Jika penentuan direktori untuk yang pertama kali, maka akan ditampilkan seperti Gambar 1.9, sedangkan jika sudah pernah ditentukan sebelumnya, akan ditampilkan seperti Gambar 1.12.



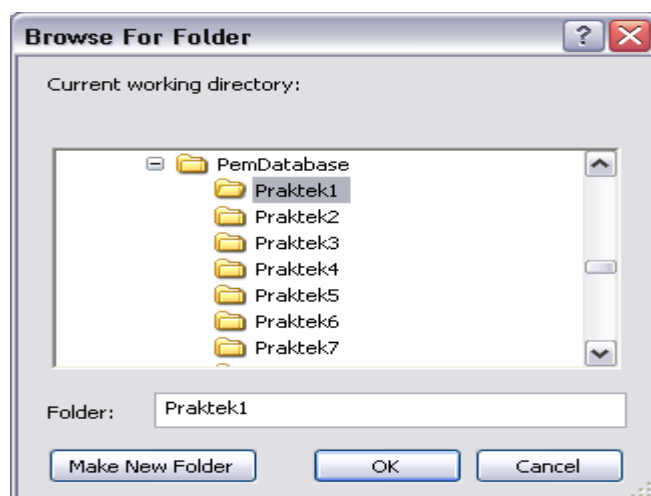
Gambar 1.9 Kondisi perubahan yang pertama kali

- Tandai checkbox yang ada tulisan **Use default directory**, lalu klik tombol titik tiga (...).



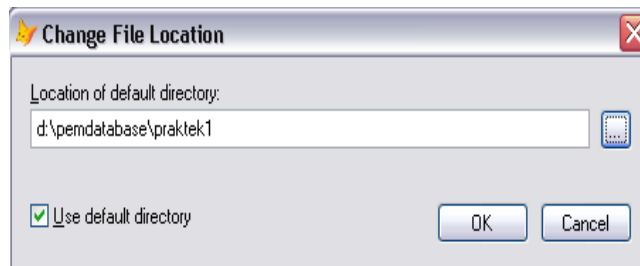
Gambar 1.10 Menentukan Direktori/Drive

- Tetapkan direktori yang akan dijadikan direktori kerja. Misalnya klik pada pilihan My Computer, lalu klik Drive D, pilih folder PemDatabase, kemudian klik folder Praktek1, dan akhiri dengan mengklik tombol **OK**.



Gambar 1.11 Mengarahkan Direktori Kerja

- Saat ditampilkan gambar seperti di bawah ini, klik kembali tombol **OK**, sehingga akan kembali ke tampilan awal seperti Gambar 1.8, namun direktori sudah sesuai dengan yang ditetapkan.

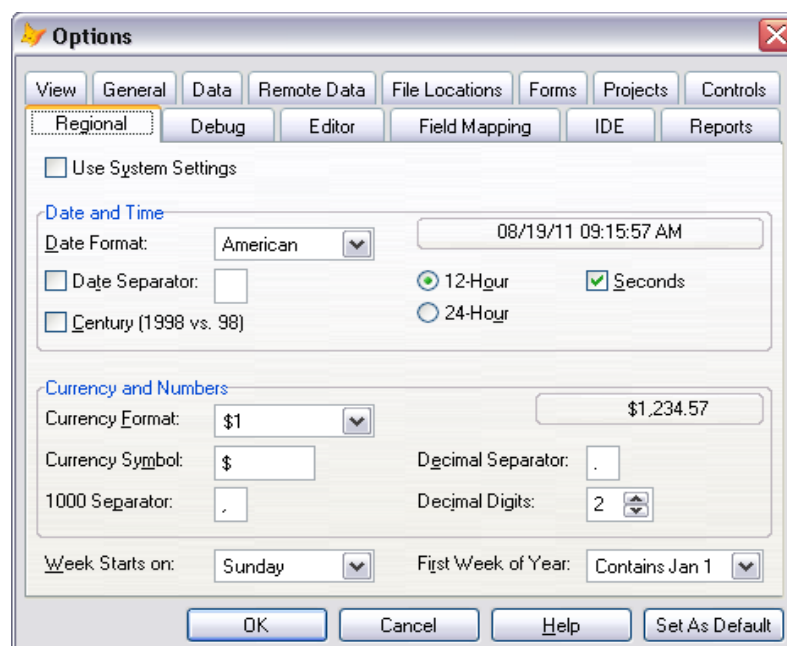


Gambar 1.12 Mengarahkan Direktori Kerja

- 5) Lakukan proses penentuan seting yang lain dengan mengklik **Tabs Page** yang lain. Tapi jika sudah selesai semua pengaturan, maka klik tombol **Set As Default**, selanjutnya klik tombol **OK**.

### B. Menentukan Seting Regional

Banyak hal yang dapat dilakukan pada seting Regional ini, misalnya untuk menentukan format tanggal, penentuan lambang mata uang, titik desimal, dll. Untuk itu klik **Tabs Page Regional**.

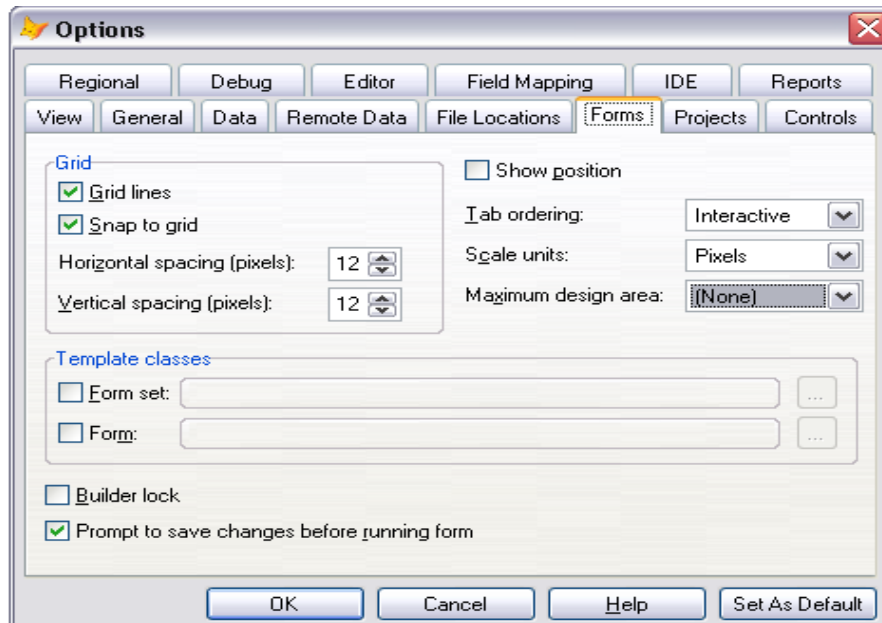


Gambar 1.13 Melakukan Seting Regional

- Untuk mengganti tampilan tanggal ke bentuk seperti aturan tanggal di Indonesia, maka : klik Combobox pada baris Date Format, lalu pilih **Italian** atau **British**
- Untuk menampilkan tahun dengan 4 digit, maka aktifkan **checkbox Century**.
- Untuk mengganti aturan keuangan, rubahlah pada bagian **Currency and Number**.

### C. Menentukan Seting Batasan Form

Batasan seting **form** ini diperlukan untuk menetapkan seperti apa tampilan rancangan **form** yang diinginkan pada saat melakukan implementasi pengolahan data. Untuk itu kliklah Tabs Page **Form**.



Gambar 1.14 Melakukan Seting Batasan Form

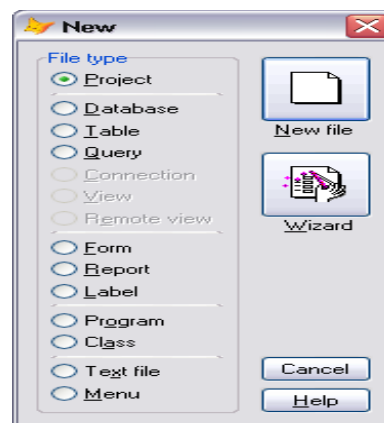
Supaya garis-garis area pada form tidak terlalu besar, sehingga akan mempengaruhi posisi penempatan komponen pada form, maka rubahlah ukuran pada **Horizontal spacing** dan **Vertical spacing**. Secara default ukuran adalah **12 pixel**. Supaya lebih kecil rubahlah ke ukuran **1 pixel**.

Untuk menentukan lebar area form, maka klik combobox pada **Maximum design area**. Secara default pilihannya adalah default. Sedangkan jika Anda akan mencoba boleh dipilih **640 X 480** atau **800 X 600** atau **1024 X 768**.

#### 1.6.4 Membuat Project Baru

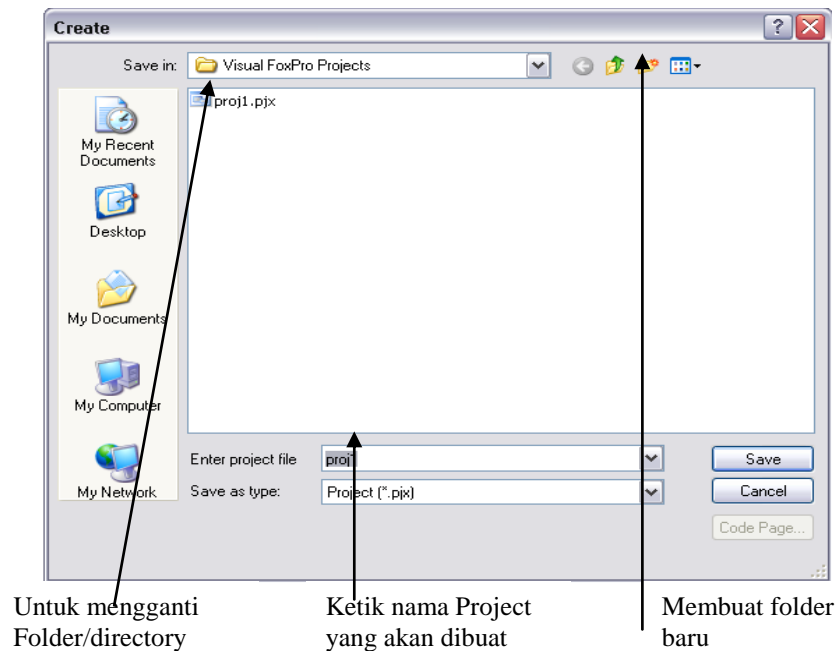
Saat posisi sudah berada pada jendela kerja Visual FoxPro seperti diperlihatkan pada gambar 1.2, kita sudah siap untuk bekerja dengan berbagai perintah atau melakukan sesuatu dengan memanfaatkan menu atau icon yang tersedia. Salah satunya adalah membuat Project baru, dengan langkah sebagai berikut:

1. Klik icon New atau tampilkan menu File, lalu pilih New. Berikutnya ditampilkan kotak dialog seperti Gambar 1.15



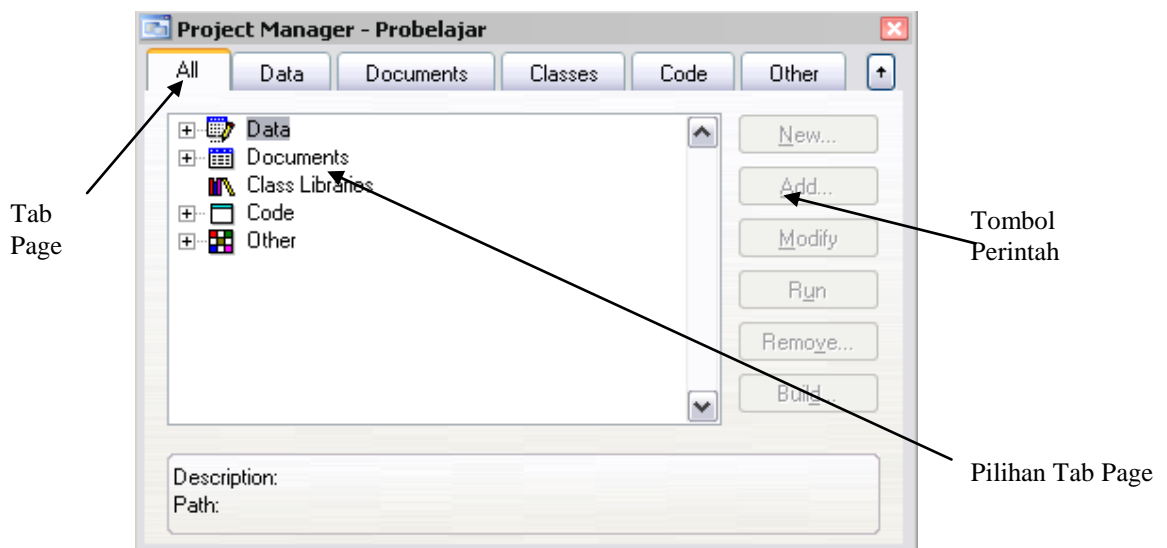
Gambar 1.15 Saat Membuat Project Baru

2. Klik Project lalu lanjutkan dengan mengklik **New File**. Berikut akan ditampilkan kotak dialog untuk menentukan tempat penyimpanan Project yang akan dibuat.



Gambar 1.16 Saat Membuat Project Baru

- Arahkan penyimpanan Project yang dibuat pada *folder* yang telah dibuat sebelumnya (folder Praktek1). Caranya adalah dengan memindahkan folder pada combobox di baris **Save In** (lihat Gambar 1.16)
  - Ketik nama Project pada baris **Enter project file**. Nama project bebas, dan dapat juga menggunakan nama yang ditawarkan (Proj1).
  - Akhiri dengan mengklik tombol **Save**.
3. Setelah selesai mengisi kotak dialog penyimpanan Project, akan diteruskan ke tampilan **Project Manager** seperti terlihat pada Gambar 1.17.



Gambar 1.17 Project Manajer

Klik masing-masing **Tab Page** untuk menentukan kelompok proses yang akan digunakan. Saat ini yang aktif adalah **Tab Page All**, yang berarti semua Tab Page aktif.

Pada masing-masing pilihan jika ada simbol "+" berarti ada sub pilihan di dalamnya. Untuk membukanya klik tanda "+" tersebut, sehingga akan berubah jadi tanda "-".



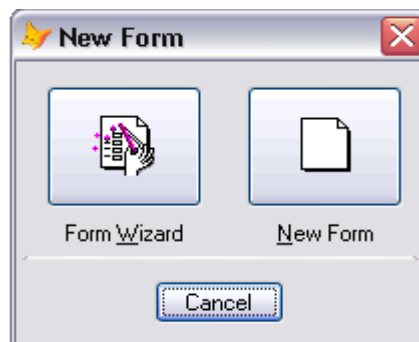
Sedangkan untuk menutupnya klik tanda "-" tersebut, sehingga akan berubah menjadi tanda "+"

1. Pada Tab Page **Data**, terdiri dari 3 pilihan yaitu: Databases, Free Table, Queries.
2. Pada Tab Page **Documents**, terdiri dari 3 pilihan yaitu: Form, Report, Labels.
3. Pada Tab Page **Classes** tidak ada pilihan
4. Pada Tab Page **Code**, terdiri dari 3 pilihan yaitu: Programs, API Libraries, Applications.
5. Pada Tab Page **Other**, terdiri dari 3 pilihan yaitu: Menus, Text Files, Other Files.

### 1.6.5 Membuat Form Baru

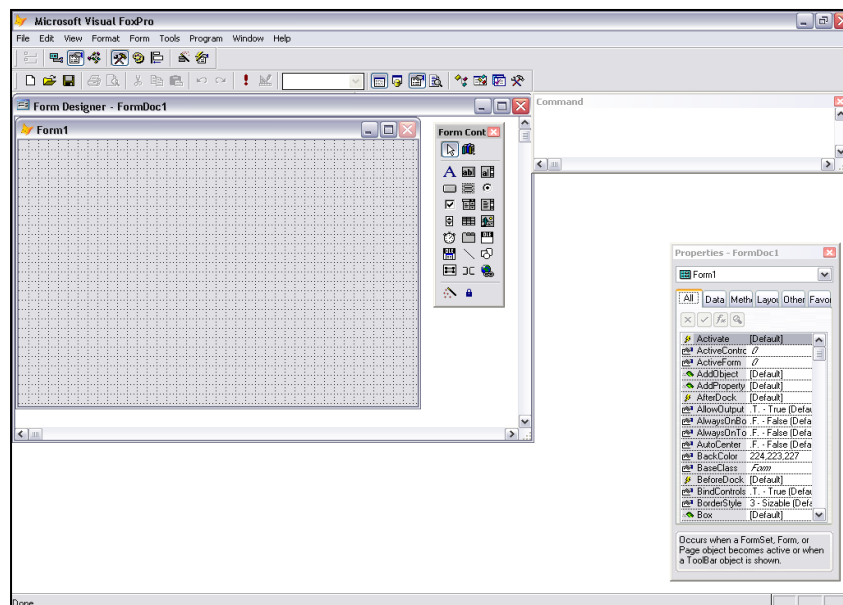
Form merupakan bagian penting pada sebuah aplikasi yang dibangun dengan Visual FoxPro, yang digunakan sebagai *interface*.

1. Klik TabPage **Documents**
2. Klik Pilihan **Forms**
3. Klik tombol **New**



Gambar 1.18 Pilihan Pembuat Form Baru

4. Pilih New Form, sehingga akan ditampilkan kondisi seperti yang terlihat pada Gambar 1.19



Gambar 1.19 Pilihan Pembuat Form Baru

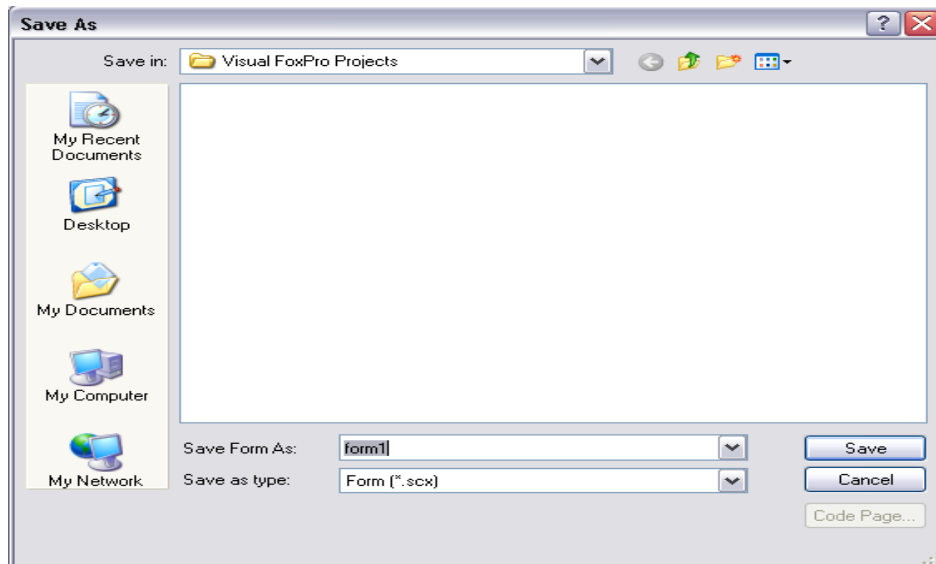
5. Lakukan proses perancangan form dengan menggunakan komponen yang sudah disediakan
6. Atur properties yang sesuai untuk masing-masing komponen

7. Tulis kode program jika diperlukan untuk masing-masing kejadian (*events*) dari komponen yang bersangkutan.

### 1.6.6 Menyimpan Form

Saat pertama kali dilakukan penyimpanan, maka yang disimpan tidak hanya Form, tetapi juga Project-nya. Untuk melakukan penyimpanan dapat dilakukan langkah-langkah sebagai berikut:

1. Tampilkan menu File, lanjutkan memilih **Save**. Atau dapat juga dengan cara langsung mengklik icon **Save** (gambar disket), sehingga akan ditampilkan kotak dialog penyimpanan seperti Gambar 1.20.

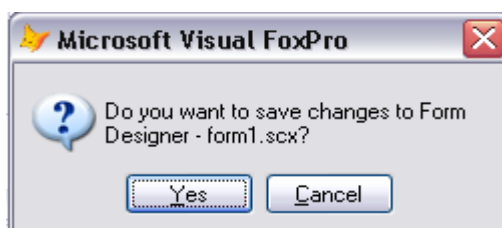


Gambar 1.20 Kotak Dialog Penyimpanan

2. Ketikkan nama *file* yang akan diberikan untuk *Form* atau *Project* yang akan disimpan. Nama dapat menggunakan nama yang ditawarkan. Jika sudah selesai memberi nama, langsung mengklik tombol **Save**.

### 1.6.7 Menjalankan Form

Untuk menjalankan Form, dapat dilakukan dengan cara menekan **icon Run** (tanda seru) atau menekan tombol Ctrl+E. Setiap kali menjalankan *Form*, bila sebelumnya melakukan perubahan terhadap *Form*, maka akan ditampilkan kotak dialog seperti Gambar 1.21.



Gambar 1.21 Kotak Dialog Saat Menjalankan *Form*

Klik tombol **Yes** untuk memastikan perubahan yang dilakukan akan disimpan dan langsung menjalankan *Form*.

# Bab 2

## Project dan Komponennya

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami **Project** dan komponen-komponennya serta dapat membuat *form (interface)* sebagai antarmuka untuk sebuah program, mengubah nilai properti, mengenal *Event-Driven programming*, menjalankan program.

Pada pertemuan ini akan diperkenalkan bagaimana membuat **Project** dan komponen-komponennya. Setelah mengenal semua komponen **Project** akan dilanjutkan untuk membuat *form* sebagai antar muka dengan menempatkan objek-objek dan mengubah nilai-nilai properti dari setiap objek. Dalam pertemuan ini juga diperkenalkan pemrograman *event-driven* dan bagaimana menjalankannya.

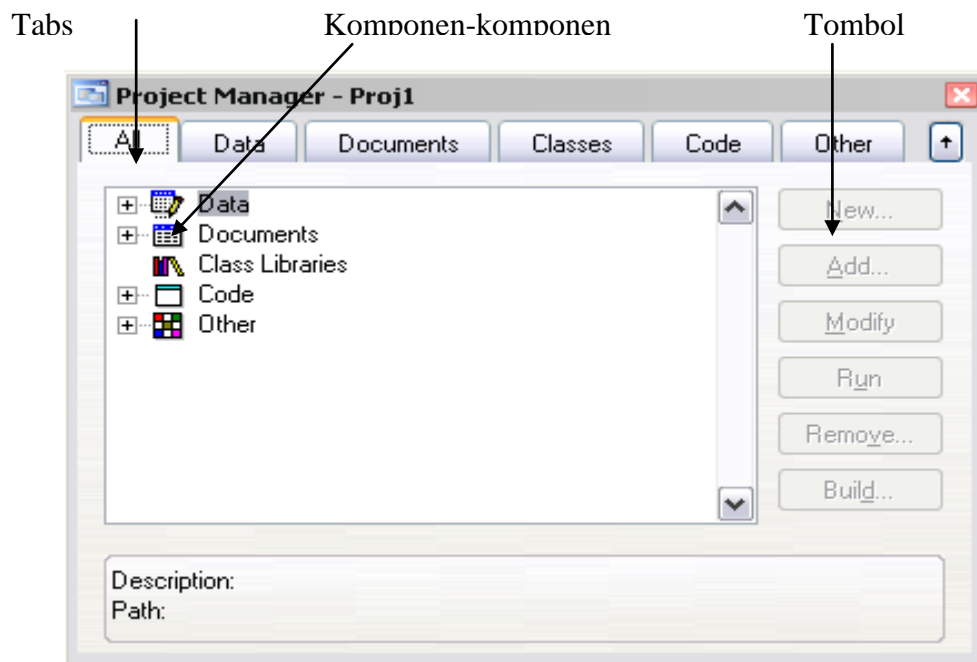
### 2.1 Project

Project merupakan cikal-bakal aplikasi pada Visual FoxPro yang terdiri dari beberapa komponen. Dalam proses pembuatannya dikendalikan oleh Project Manajer. Pada Project Manajer, komponen-komponen Project dikelompokkan sedemikian rupa, sehingga mempermudah dalam pengelolaannya.

Walaupun komponen pembentuk Project cukup banyak, namun dalam implementasinya, tidak harus semua komponen harus dibuat atau digunakan. Semuanya tergantung tingkat kompleksitas aplikasi yang akan dibangun.

Untuk membuat Project bisa dilakukan diawal, baru kemudian membuat atau melengkapi komponen-komponennya. Selain itu bisa juga membuat komponen-komponennya dulu baru digabungkan ke dalam Project yang dibuat kemudian.

Saat membangun sebuah **Project** kita akan dihadapkan pada sebuah **Project Manajer** seperti Gambar 2.1.



Gambar 2.1 Tampilan Project Manajer

Secara default Tabs Page yang terpilih adalah All, yang berarti semua komponen **Project** dapat ditampilkan. Jika di depannya komponen ada simbol **+**, maka komponen itu bisa dibuka dengan mengklik simbol **+** (plus) tersebut. Sebaliknya jika didepan komponen adalah tanda **-** (minus), maka saat diklik akan berdampak menutup komponen tersebut.

### 2.1.1 Data

Pada tabs page Data terdiri dari 3 (tiga) macam komponen Project yaitu *Databases*, *Free Tables*, dan *Queries*

#### 1. Databases

*Database* dapat diibaratkan sebuah gudang yang merupakan tempat penyimpanan data suatu informasi. Dengan satu *database* dapat menciptakan beberapa tabel dan dapat mengorganisasikannya dengan mudah.

#### 2. Free Tables

*Free tables* merupakan tabel-tabel yang berdiri sendiri, tidak terangkum dalam database. Dengan demikian apabila ingin menggunakannya langsung memakai tabelnya tanpa harus membuka *datasenya* terlebih dahulu.

#### 3. Queries

*Queries* merupakan fasilitas yang disediakan untuk menggabungkan beberapa tabel guna melakukan manipulasi data dan dapat digunakan untuk menampilkan informasi atau laporan.

### 2.1.2 Documents

Pada tabs page Documents terdiri dari 3 (tiga) macam komponen Project yaitu **Forms**, **Reports**, dan **Labels**.

#### 1. Forms

Form merupakan objek utama dalam permrograman Visual FoxPro, sebab form merupakan tempat dimana objek-objek lain seperti kontrol diletakkan seperti terlihat pada gambar 2.1. Form dan Kontrol merupakan objek-objek yang ada dalam Visual FoxPro yang memiliki properti, metode dan dapat menangkap suatu kejadian (*event*). Pada bagian ini akan dibahas tentang properti, metode (*methods*), dan kejadian yang berhubungan dengan objek-objek tersebut. Visual FoxPro mendukung properti-properti yang hanya dimiliki oleh sejumlah objek tertentu saja. Akan tetapi ada beberapa properti-properti yang sifatnya umum yaitu properti-properti ini hampir dimiliki oleh semua objek yang ada dalam Visual FoxPro. Properti-properti umum tersebut antara lain properti *name*, *top*, *left*, *height*, *weight*, *Foreground*, *Background*, *Font*, *Caption*, *Text*, *Enable*, *Visible*, *TabStop*, *TabIndex*, dan properti-properti umum lainnya.

#### 2. Reports

Report berfungsi untuk menampilkan data dalam bentuk dokumen yang tercetak pada suatu media. Tampilan data bisa diatur sedemikian rupa, misalnya dengan orientasi potrait maupun landscape.

#### 3. Labels

Hampir sama dengan Report, Label juga digunakan untuk menampilkan data dalam bentuk dokumen yang tercetak pada suatu media. Perbedaannya adalah label menampilkan data dalam bentuk kolom-kolom kecil, dimana dalam satu halaman bisa diciptakan beberapa kolom dan beberapa baris. Perumpamaannya seperti KTP/SIM, Judul Buku, dll.

### 2.1.3 Clases

Pada tabs page Class tidak seperti tabs page documents, karena pada tabs page ini tidak ada pilihan lainnya. Komponen ini sangat dibutuhkan untuk menunjang pengembangan sistem secara lebih cepat karena dengan memanfaatkan class-class yang sudah ada, maka operasi-operasi tertentu tinggal memanggilnya.

### 2.1.4 Code

Pada tabs page **Code** terdiri dari 3 (tiga) macam komponen **Project** yaitu **Programs**, **API Libraries**, dan **Applications**.

#### 1. Programs

Fasilitas ini disediakan untuk membuat program utama dari aplikasi yang dibuat. Pada programs selain membuat program utama (berisikan aturan-aturan penting untuk aplikasi dan menjadi Main), juga bisa menciptakan prosedur atau fungsi-fungsi tertentu.

#### 2. API Libraries

Digunakan untuk mengambil API Libraries untuk dimasukkan ke dalam project guna kepentingan tertentu.

#### 3. Applications

Digunakan untuk menambahkan file aplikasi (file dengan tipe .App atau .Exe)

### 2.1.5 Other

Pada tabs page Other terdiri dari 3 (tiga) macam komponen **Project** yaitu **Menus**, **Text Files**, dan **Other Files**.

#### 1. Menus

Fasilitas ini digunakan untuk merancang menu, baik menu batang maupun menu layang. Dengan fasilitas ini, semua fasilitas fungsional dari sistem yang dihasilkan dapat diintegrasikan sehingga memudahkan dalam proses pengoperasiannya.

#### 2. Text Files

Fasilitas ini disediakan untuk membuat file berjenis Teks sama seperti file yang dibuat pada NotePad atau MS Word. File ini dapat dimanfaatkan untuk menampilkan informasi-informasi yang panjang

#### 3. Other Files

Pada bagian ini merupakan kelompok untuk jenis-jenis file yang lain, seperti file-file jenis gambar dan sebagainya.

## 2.2 Konsep Penting Dalam Pemrograman Visual

Setelah mengenal dan memahami komponen-komponen Project pada Visual FoxPro, hal yang penting lainnya perlu untuk dipahami dan dicermati adalah konsep dalam pemrograman Visual. Dari judul sub bab dapat dilihat bahwa pemahaman kita tidak hanya berlaku untuk pemrograman Visual FoxPro tetapi juga dapat dimanfaatkan untuk pemrograman visual yang lain.

Ada tiga tahapan umum yang dapat dilakukan dalam melakukan pemrograman visual yaitu:

#### 1. Mengambil dan mengatur (mendesain) komponen-komponen atau objek yang digunakan.

Komponen atau objek yang digunakan tentu saja bervariasi untuk setiap aplikasi yang dibuat. Namun hal yang penting diusahakan adalah ketepatan untuk objek yang digunakan sehingga akan meningkatkan kualitas aplikasi yang dihasilkan. Komponen atau objek yang dimaksud adalah seperti yang terlihat pada Gambar 1.4 pada modul 1 (satu). Form juga termasuk sebagai objek.

Kemampuan nilai seni sangat mendukung agar rancangan atau tata letak komponen pada form bisa terlihat menarik dan komunikatif.

## 2. Mengatur properties

Untuk mengatur properties, tentu saja perlu kiat dan pemahaman yang baik. Walaupun masing-masing komponen atau objek memiliki properties yang banyak dan juga ada yang berbeda, maka disarankan aturlah properties yang penting-penting saja. Tidak semua properties harus dirubah atau diatur.

## 3. Menulis kode program.

Supaya aplikasi dapat melaksanakan operasi-operasi ataupun tindakan-tindakan tertentu, maka diperlukan upaya untuk memberikan perintah-perintah yang sesuai. Perintah-perintah tersebut tulis berdasarkan kejadian (*events*) menurut objek yang akan dikendalikan.

## 2.3 Praktek

### 2.3.1 Membuat Project Baru

Untuk keperluan pertemuan kedua, setelah memanggil sistem Visual FoxPro, lanjutkan membuat Project baru. Bila saat memanggil Visual FoxPro bila sudah tertampil Project Manajer dari pekerjaan sebelumnya, mohon ditutup terlebih dahulu. Setelah itu lakukan proses pembuatan **Project** baru dengan cara:

1. Tampilkan menu File, pilih menu New atau klik icon **New**.
2. Saat ditampilkan kotak dialog, pilih Project dan klik tombol **New File**.
3. Beri nama Project yang dibuat dengan nama Proj2 dan simpan pada folder kerja Anda yang bernama Sesi2, lanjutkan dengan mengklik tombol **Save**.

### 2.3.2 Membuat Form

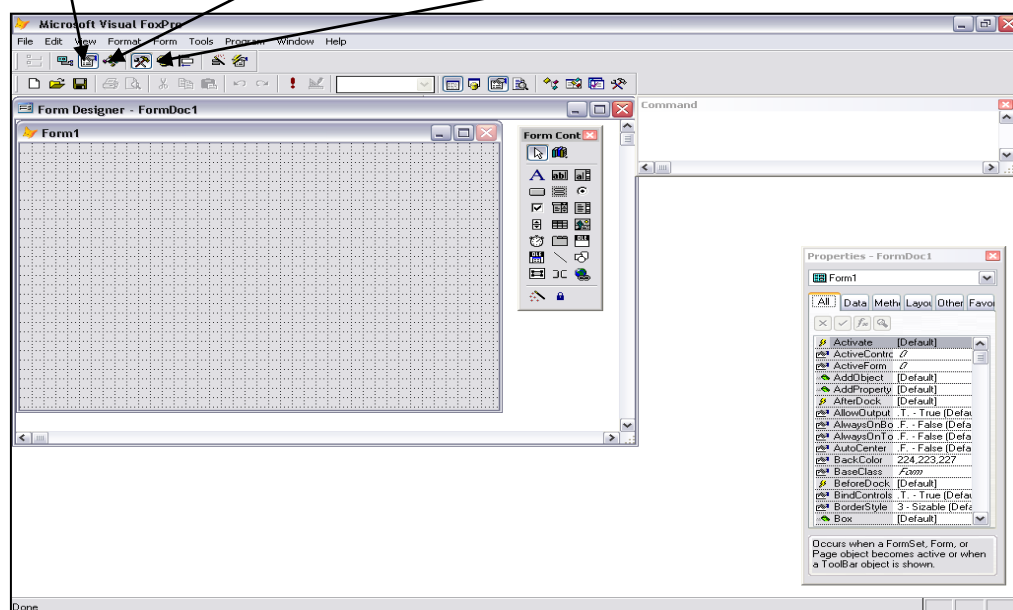
Setelah berhasil melakukan pembuatan Project baru, dan berada pada posisi Project Manajer, buatlah form baru dengan langkah sebagai berikut:

1. Klik Tab Page **Documents**
2. Klik pilihan **Form**
3. Klik tombol **New**
4. Pilihan New Form, sehingga akan ditampilkan seperti terlihat pada Gambar 2.2

Untuk menampilkan  
atau menutup

Untuk berpindah dari desain  
form ke window Code dan

Untuk menampilkan  
atau menutup Form



Gambar 2.2 Tampilan Desain Form

### 2.3.3 Menggunakan Komponen Label, Shape, TextBox, OptionGroup, ComboBox, CheckBox, Command Button

Setelah masuk pada desain *form* seperti Gambar 2.2, gunakanlah beberapa komponen untuk menghasilkan *interface* seperti terlihat pada Gambar 2.3 dengan langkah-langkah sebagai berikut:

The image shows a Windows application window with the title bar "Sistem Keorganisasian Mahasiswa". Inside the window, the form is titled "PENDAFTARAN ANGGOTA HIMAMIKA". The form contains the following fields and controls:

- No. Pendaftaran: Text box labeled "Text1".
- Nama Lengkap: Text box labeled "Text2".
- Tanggal Lahir: Text box labeled "Text3".
- Jenis Kelamin: Radio button group with "Laki-Laki" (selected) and "Perempuan".
- Agama: Dropdown menu labeled "Combo1".
- Alamat: Text box labeled "Text4".
- Bidang Hobi: Three checkboxes labeled "Olah Raga", "Seni", and "Membaca".
- A "Tutup" button is located at the bottom right of the form.

Gambar 2.3 Rancangan Form Latihan Pertama

1. Sebelum mengambil komponen, atur dulu properties dari form yaitu :
  - Caption : ditulis Sistem Keorganisasian Mahasiswa
  - AutoCenter : True
  - MDIForm : True
  - ShowTips : True
  - ShowWindow : 2 As Top Level Form
2. Ambil Shape lalu bentuk kotak seperti gambar
3. Ambil komponen label sebanyak 7 buah untuk membuat tulisan, berturut-turut: untuk Label1 ganti captionnya dengan No. Pendaftaran, Label2 diganti captionnya dengan Nama Lengkap, Label3 diganti captionnya dengan Tanggal Lahir, Label4 diganti captionnya dengan Jenis Kelamin, Label5 diganti captionnya dengan Agama, Label6 diganti captionnya dengan Alamat, Label7 diganti captionnya dengan Bidang Hobi.
4. Ambil 3 buah TextBox berturut-turut untuk No.Pendaftaran, Nama, Tanggal Lahir. Letakkan di bagian kanannya.
5. Ambil satu komponen OptionGroup, lalu tempatkan di kanan Jenis Kelamin. Lakukan klik kanan pada OptionGroup tersebut, pilih Builder, pada kolom Caption tuliskan pilihan yang ingin dibuat misalkan untuk jenis kelamin diisi Laki-laki pada baris pertama dan Perempuan untuk baris kedua. Klik tab Layout lalu pilih horizontal jika diinginkan tampilan pilihan dalam bentuk datar. Akhiri dengan mengklik tombol OK.
6. Ambil komponen ComboBox dan tempatkan di kanan Agama. Klik pada ComboBox tersebut lalu pilih Builder. Pada baris *Fill the list with* ganti pilihannya dengan pilihan yang kedua yaitu : Data Entered by hand. Isikan pilihan yang akan ditampilkan pada ComboBox pada coloum1, yaitu : Islam, Katholik, Kristen, Hindu, Budha.
7. Ambil komponen TextBox, letakkan di bagian kanan dari Alamat.
8. Ambil 3 buah CheckBox, lalu susun di bagian kanan Bidang Hobi dan ganti masing-masing Captionnya. CheckBox1 diganti Captionnya dengan Olah Raga. CheckBox2 diganti Captionnya dengan Seni. CheckBox3 diganti Captionnya dengan Membaca.
9. Ambil komponen Command Button dan letakkan di bagian bawah seperti terlihat pada gambar. Gantilah Captionnya dengan Tutup, lalu ganti juga properties Name dengan tbltutup
10. Klik dua kali pada tombol Tutup untuk menulis kode program. Ketikkan kode program.

#### **Thisform.Release()**

11. Jalankan program dengan cara meng-klik tombol Run atau dapat juga dengan cara menekan tombol Ctrl+E. Jika saat menjalankan program pertama kali, tentu akan ditanyakan form akan disimpan dimana dan dengan nama apa. Untuk itu arahkan tempat penyimpanan sama dengan tempat Project yaitu folder Sesi2.

### **2.3.4 Menggunakan Komponen Label, Line, Shape, Timer, Container, TextBox, Edit Box, ListBox, Spinner, Image, Command Button**

Untuk lebih memantapkan materi mengenai penggunaan komponen atau objek pada form, maka ciptakan form baru untuk mengimplementasikannya. Untuk mempersiapkan untuk penyelesaian masalah yang kompleks, pada bagian ini akan digunakan komponen-komponen seperti tertera di atas.

Jika saat ini Anda berada pada *form design* (form yang aktif saat ini) maka tutuplah terlebih dahulu supaya posisi kembali ke Project Manajer. Dalam kondisi berada pada Project Manajer, buatlah form baru dengan langkah sebagai berikut:



1. Klik Tab Page **Documents**
2. Klik pilihan **Form**
3. Klik tombol **New**
4. Pilihan **New Form**
5. Sebelum menggunakan komponen, aturlah terlebih dahulu properties **form** yaitu:
  - Caption : ditulis Sistem Informasi Penggajian
  - AutoCenter : True
  - MDIForm : True
  - MaxButton : False
  - MinButton : False
  - ShowTips : True
  - ShowWindow : 2 As Top Level Form
6. Setelah mengatur properties **form**, gunakanlah komponen-komponen yang sesuai untuk membuat form seperti Gambar 2.4

The image shows a Java Swing window titled "Sistem Informasi Penggajian". Inside the window, there is a header section with the text "UNIVERSITAS RESPATI YOGYAKARTA" and "JL. LAKSDA ADISUCIPTO KM 6.3, DEPOK, SLEMAN, YOGYAKARTA". Below this is a section titled "DATA KARYAWAN". The form contains several input fields: "N.I.P" (Text1), "Nama" (Text2), "Unit Kerja" (List1), "Jumlah Staff" (nner1), "Nama Pimpinan" (Text3), and "Catatan" (Edit1). There is also a "Pas Foto" label next to a placeholder image of a cartoon fox. On the right side of the form, there are six buttons: "Baru", "Simpan", "Batal", "Ubah", "Hapus", and "Selesai". A clock icon labeled "Jam" is located in the top right corner of the form area.

Gambar 2.4 Rancangan Form Latihan Kedua

Keterangan:

1. Tulisan bagian atas dibuat dengan 3 label. Label1 diganti captionnya dengan UNIVERSITAS RESPATI YOGYAKARTA. Label2 diganti captionnya dengan JL. LAKSDA ADISUCIPTO KM 6.3, DEPOK, SLEMAN, YOGYAKARTA. Label3 diganti captionnya dengan DATA KARYAWAN. Untuk ketiga label properties yang perlu diubah lagi yaitu Aligment diisi Center, FontBold diganti True, FontSize diganti dengan ukuran yang sesuai.
2. Bagian kanan atas dibuat dengan satu **Shape**. Properties yang diatur yaitu **BorderWidth** diganti dengan ukuran 4. **Curvatur** diisikan dengan angka 90. Setelah itu ditambahkan satu label (label4). Properties yang diatur yaitu Captionnya diganti dengan Jam, Aligment diganti dengan Center, FontBold diganti dengan True, Fontsize diganti dengan 12. Komponen berikutnya yaitu **Timer**. Properties yang perlu diganti yaitu **Interval** dan diisikan dengan angka **1000**. Setelah itu klik dua kali pada komponen Timer, sehingga akan masuk ke window editor dan tulislah perintah sebagai berikut:

```

IF thisform.label4.Caption<>TIME()
  thisform.label4.Caption=TIME()
ENDIF

```

3. Berikutnya dibuat pembatas dengan 2 (dua) komponen Line. Line1 dibuat untuk garis mendatar, Line2 dibuat untuk garis miring. Properties yang perlu diganti yaitu BorderWidth diisi dengan angka 3.
4. Untuk kotak besar ditengah dibuat dengan **Container**. Properties yang diubah yaitu **SpecialEffect** diganti dengan **Sunken**.
5. Di dalam Container digunakan label untuk tulisan-tulisan berturut-turut adalah : Label5 captionnya diubah menjadi N.I.P, Label6 captionnya diubah menjadi Nama, Label7 captionnya diubah menjadi Unit Kerja, Label8 captionnya diubah menjadi Jumlah Staf, Label9 captionnya diubah menjadi Nama Pimpinan, Label10 captionnya diubah menjadi Catatan, Label11 captionnya diubah menjadi Pas Foto.
6. Komponen berikutnya adalah **Text1** diletakkan di kanan N.I.P, Text2 diletakkan di kanan Nama.
7. Komponen selanjutnya adalah **Listbox1** yang diletakkan di kanan Unit kerja. Untuk mengisi Listbox hampir sama seperti mengisi ComboBox.
8. Komponen berikutnya adalah **Spinner1** yang diletakkan di kanan Jumlah Staf. Properties yang diubah adalah KeyboardHighValue diubah menjadi 20, KeyboardLowValue diubah menjadi 1. *SpinnerHighValue* diubah menjadi 20 dan *SpinnerLowValue* diubah menjadi 1.
9. Komponen berikutnya adalah **Text3** yang diletakkan di kanan Nama Pimpinan.
10. Komponen berikutnya adalah **Edit1** yang diletakkan di kanan Catatan.
11. Komponen berikutnya adalah Image1 yang diletakkan di kanan Pas Foto. Properties yang diubah adalah **Stretch** menjadi Stretch. Kemudian properties Picture diklik pada tombol titiktiga untuk mencari lokasi gambar.
12. Komponen berikutnya adalah **6 CommandButton**. Berturut-turut dari Comamand1 sampai Command6. Untuk Command1 Captionnya diubah menjadi **Baru** dan Name diubah menjadi **tblbaru**. Untuk Command2 Captionnya diubah menjadi **Simpan** dan Name diubah menjadi **tblsimpan**. Untuk Command3 Captionnya diubah menjadi **Batal** dan Name diubah menjadi **tblbatal**. Untuk Command4 Captionnya diubah menjadi **Ubah** dan Name diubah menjadi **tblubah**. Untuk Command5 Captionnya diubah menjadi **Hapus** dan Name diubah menjadi **tblhapus**. Untuk Command6 Captionnya diubah menjadi **Selesai** dan Name diubah menjadi **tblselesai**.
13. Klik dua kali pada tombol **Selesai**, lalu ketik kode program sebagai berikut:

```

Thisform.Release()

```

14. Jalankan program dengan cara meng-klik tombol Run atau dapat juga dengan cara menekan tombol Ctrl+E. Jika saat menjalankan program pertama kali, tentu akan ditanyakan *form* akan disimpan dimana, dan dengan nama apa. Untuk itu arahkan tempat penyimpanan sama dengan tempat Project yaitu folder Sesi2

# Bab 3

## Pengelolaan Form

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami lebih dalam mengenai Form dan komponen-komponen yang digunakan sebagai bahan untuk menghasilkan interface yang bagus, lebih fokus dalam mengatur properties komponen yang bersangkutan serta lebih memahami *Event-Driven programming*.

Pada pertemuan ini akan dijelaskan mengenai pengaturan properties penting untuk setiap komponen atau objek, event-driven yang paling sering digunakan pada komponen atau objek serta dilengkapi dengan soal-soal latihan.

### 3.1 Properties Penting

Seperti yang telah diketahui bahwa setiap komponen atau objek memiliki properties yang berbeda dengan jumlah yang cukup banyak. Apabila tidak melakukan perubahan, maka yang digunakan adalah nilai properties default (nilai bawaan yang diberikan sistem). Namun untuk melakukan perubahan, tentu saja tidak harus semua. Untuk melakukan perubahan properties disesuaikan dengan kebutuhan.

Untuk tahap awal (pemula) lakukan perubahan untuk properties yang penting-penting saja dari objek yang bersangkutan.

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
1	Form	<ul style="list-style-type: none"> <li>• Caption</li> <li>• AutoCenter</li> <li>• MDIForm</li> <li>• MaxButton</li> <li>• MinButton</li> <li>• ShowTips</li> <li>• ShowWindow</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menuliskan judul window/Form</li> <li>• Supaya form tampil ditengah (diganti True)</li> <li>• Untuk menentukan apakah mejadikan form Multi Docoment Interface (diganti True jika dijadikan MDI)</li> <li>• Untuk menentukan apakah tombol Maximaze ditampilkan atau tidak (diganti False jika tidak ditampilkan)</li> <li>• Untuk menentukan apakah tombol Manimaze ditampilkan atau tidak (diganti False jika tidak ditampilkan)</li> <li>• Untuk mengaktifkan TooltipsText (diganti True jika ingin ditampilkan)</li> <li>• Untuk menentukan cara tampil form (diganti As Top Level Form supaya tampil selalu di atas)</li> </ul>
2	Label	<ul style="list-style-type: none"> <li>• Caption</li> <li>• BackColor</li> <li>• BackStyle</li> <li>• FontBold</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan tulisan yang akan ditampilkan</li> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan apakah warna latar belakang berlaku atau tidak (Opaque = BackColor berlaku, Transparent = BackColor tidak berlaku sehingga langsung tembus warna form)</li> <li>• Untuk mengaktifkan cetak tebal atau</li> </ul>

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
		<ul style="list-style-type: none"> <li>• FontName</li> <li>• FontSize</li> <li>• ForeColor</li> <li>• WordWrap</li> </ul>	<p>tidak (diganti True jika ingin cetak tebal)</p> <ul style="list-style-type: none"> <li>• Untuk menentukan nama huruf (pilih jenis huruf yang diinginkan)</li> <li>• Untuk menentukan ukuran huruf (masukkan)</li> <li>• Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>• Untuk menjadikan apakah tulisan akan dilanjutkan ke bars berikutnya jika sudah melebihi lebarnya (diganti True)</li> </ul>
3	TextBox	<ul style="list-style-type: none"> <li>• Aligment</li> <li>• BackColor</li> <li>• DisabledBackColor</li> <li>• DisabledForeColor</li> <li>• FontBold</li> <li>• FontSize</li> <li>• ForeColor</li> <li>• Format</li> <li>• InputMask</li> <li>• MaxLenght</li> <li>• TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan data akan ditampilkan dengan posisi apa (rata kiri, tengah, rata kanan)</li> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan efek latar belakang saat kondisi enabled=.F. (False)</li> <li>• Untuk menentukan efek warna tulisan saat kondisi enabled=.F. (False)</li> <li>• Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak tebal)</li> <li>• Untuk menentukan ukuran huruf (masukkan)</li> <li>• Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>• Untuk menentukan format tampilan data (untuk data numerik bisa diberikan seperti : 99,999,999)</li> <li>• Untuk menentukan efek saat mengisi data, biasanya dibuat sama dengan Format.</li> <li>• Untuk menentukan lebar maksimal data yang bisa dimasukkan (untuk data yang dijadikan primary sebaiknya perlu memperhatikan ini untuk diisikan sama dengan lebar field)</li> <li>• Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot.</li> </ul>
4	EditBox	<ul style="list-style-type: none"> <li>• Aligment</li> <li>• BackColor</li> <li>• DisabledBackColor</li> <li>• DisabledForeColor</li> <li>• FontBold</li> <li>• FontSize</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan data akan ditampilkan dengan posisi apa (rata kiri, tengah, rata kanan)</li> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan efek latar belakang saat kondisi enabled=.F. (False)</li> <li>• Untuk menentukan efek warna tulisan saat kondisi enabled=.F. (False)</li> <li>• Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak tebal)</li> <li>• Untuk menentukan ukuran huruf (masukkan)</li> </ul>

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
		<ul style="list-style-type: none"> <li>• ForeColor</li> <li>• TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>• Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot</li> </ul>
5	CommandButton	<ul style="list-style-type: none"> <li>• Aligment</li> <li>• BackColor</li> <li>• Caption</li> <li>• DisabledBackColor</li> <li>• DisabledForeColor</li> <li>• FontBold</li> <li>• FontSize</li> <li>• ForeColor</li> <li>• TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan data akan ditampilkan dengan posisi apa (rata kiri, tengah, rata kanan)</li> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan tulisan yang akan ditampilkan. Untuk Caption ini bisa ditambahkan simbol \&lt; untuk memberikan efek supaya huruf yang ada setelah simbol tersebut diberi garis bawah.</li> <li>• Untuk menentukan efek latar belakang saat kondisi enabled=.F. (False)</li> <li>• Untuk menentukan efek warna tulisan saat kondisi enabled=.F. (False)</li> <li>• Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak tebal)</li> <li>• Untuk menentukan ukuran huruf (masukkan)</li> <li>• Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>• Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot</li> </ul>
6	CommandGroup	<ul style="list-style-type: none"> <li>• BackColor</li> <li>• ButtonCount</li> <li>• Name</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan jumlah tombol yang akan dibuat (bisa juga dirubah dengan klik kanan, lalu pilih builder)</li> <li>• Untuk mementukan nama CommandGroup (nama masing-masing juga bisa dirubah dengan klik kanan, lalu pilih edit.</li> </ul>
7	OptionGroup	<ul style="list-style-type: none"> <li>• BackColor</li> <li>• ButtonCount</li> <li>• Name</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan warna latar belakang</li> <li>• Untuk menentukan jumlah pilihan yang akan dibuat (bisa juga dirubah dengan klik kanan, lalu pilih builder)</li> <li>• Untuk mementukan nama OptionGroup (nama masing-masing juga bisa dirubah dengan klik kanan, lalu pilih edit.</li> </ul>
8	CheckBox	<ul style="list-style-type: none"> <li>• Caption</li> <li>• FontBold</li> </ul>	<ul style="list-style-type: none"> <li>• Untuk menentukan tulisan yang akan ditampilkan.</li> <li>• Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak</li> </ul>

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
		<ul style="list-style-type: none"> <li>FontSize</li> <li>ForeColor</li> </ul>	tebal) <ul style="list-style-type: none"> <li>Untuk menentukan ukuran huruf (masukkan)</li> <li>Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> </ul>
9	ComboBox	<ul style="list-style-type: none"> <li>BackColor</li> <li>DisabledBackColor</li> <li>DisabledForeColor</li> <li>FontBold</li> <li>FontSize</li> <li>ForeColor</li> <li>TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan warna latar belakang</li> <li>Untuk menentukan efek latar belakang saat kondisi enabled=.F. (False)</li> <li>Untuk menentukan efek warna tulisan saat kondisi enabled=.F. (False)</li> <li>Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak tebal)</li> <li>Untuk menentukan ukuran huruf (masukkan)</li> <li>Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot</li> </ul>
10	Spinner	<ul style="list-style-type: none"> <li>BackColor</li> <li>DisabledBackColor</li> <li>DisabledForeColor</li> <li>FontBold</li> <li>FontSize</li> <li>ForeColor</li> <li>KeyboardHighValue</li> <li>KeyBoardLowValue</li> <li>SpinnerHighValue</li> <li>SpinnerLowValue</li> <li>TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan warna latar belakang</li> <li>Untuk menentukan efek latar belakang saat kondisi enabled=.F. (False)</li> <li>Untuk menentukan efek warna tulisan saat kondisi enabled=.F. (False)</li> <li>Untuk mengaktifkan cetak tebal atau tidak (diganti True jika ingin cetak tebal)</li> <li>Untuk menentukan ukuran huruf (masukkan)</li> <li>Untuk menentukan warna huruf (klik tombol titik tiga untuk masuk ke pemilihan warna)</li> <li>Untuk menentukan nilai tertinggi yang bisa diisikan melalui keyboard</li> <li>Untuk menentukan nilai terendah yang bisa diisikan melalui keyboard</li> <li>Untuk menentukan nilai tertinggi yang bisa diisikan melalui penekanan panah atas melalui mouse</li> <li>Untuk menentukan nilai terendah yang bisa diisikan melalui penekanan panah bawah melalui mouse</li> <li>Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot</li> </ul>
11	Grid	<ul style="list-style-type: none"> <li>BackColor</li> <li>ColumnCount</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan warna latar belakang</li> <li>Untuk menentukan jumlah kolom</li> </ul>

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
		<ul style="list-style-type: none"> <li>DeleteMark</li> <li>Enabled</li> <li>TooltipsText</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menetapkan apakah kolom untuk tanda penghapusan record ditampilkan atau tidak (default =.T. atau True)</li> <li>Untuk menentukan apakah objek bisa diakses (dimodifikasi atau tidak). Secara default nilai .T. atau True</li> <li>Untuk menuliskan keterangan untuk objek, dan akan terlihat saat objek tersebut disorot</li> </ul>
12	Image	<ul style="list-style-type: none"> <li>Stretch</li> <li>Picture</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan bentuk penyajian gambar. Ada tiga pilihan yaitu: 0 Clip = penyajian gambar mengikuti ukuran asli gambar, jika area gambar kecil dari ukuran gambar maka gambar tidak bisa tampil penuh, 1 Isometric = penyajian gambar bisa tampil penuh tapi tidak selalu mengikuti ukuran area gambar, 2 Stretch = penyajian gambar selalu mengikuti ukuran area gambar.</li> <li>Untuk memasukkan gambar yang akan ditampilkan. Sebaiknya gambar-gambar yang akan dipakai sudah berada pada directory yang sama dengan project yang dibuat.</li> </ul>
13	Timer	<ul style="list-style-type: none"> <li>Interval</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan ukuran interval yang mengakibatkan pergerakan jam. Jika diisi 1000 maka perubahan terjadi setiap 1 detik. Untuk menghasilkan jam isilah 1000, tetapi untuk animasi isilah sesuai kebutuhan. Semakin kecil isian, maka perubahan semakin cepat.</li> </ul>
14	PageFrame	<ul style="list-style-type: none"> <li>PageCount</li> <li>TabOrientation</li> <li>TabStyle</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan jumlah halaman</li> <li>Untuk menentukan letak TabPage (Top=Diatas, Bottom=Dibawah, Left=Dikiri, Right=Dikanan)</li> <li>Untuk menentukan cara TabPage di tampilan (Justified=seukuran lebar pageframe, Nonjustified= tidak seukuran pageframe)</li> </ul>
15	Line	<ul style="list-style-type: none"> <li>BorderColor</li> <li>BorderStyle</li> <li>BorderWidth</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan warna garis</li> <li>Untuk menentukan jenis garis</li> <li>Untuk menentukan ukuran atau lebar garis. Semakin besar angka yang diisikan, semakin tebal garis yang dihasilkan.</li> </ul>
16	Shape	<ul style="list-style-type: none"> <li>BackColor</li> <li>BackStyle</li> <li>BorderColor</li> <li>BorderStyle</li> <li>Curvature</li> </ul>	<ul style="list-style-type: none"> <li>Untuk menentukan warna latar belakang</li> <li>Untuk menentukan apakah warna latar belakang berlaku atau tidak (Opaque = BackColor berlaku, Transparent = BackColor tidak berlaku sehingga langsung tembus warna form)</li> <li>Untuk menentukan warna garis dari Shape</li> <li>Untuk menentukan jenis garis dari Shape</li> <li>Untuk menentukan bentuk tampilan</li> </ul>

No	Nama Objek	Properties Pokok	Kegunaan/Keterangan
		<ul style="list-style-type: none"> <li>• SpecialEffect</li> <li>• Style</li> </ul>	Shape. Jika diisi dengan suatu nilai maka bentuk bangun dari Shape akan berubah. Jika diisi 90 akan berbentuk lingkaran. <ul style="list-style-type: none"> <li>• Menentukan efek dari shape</li> <li>• Untuk menentukan jenis shape</li> </ul>

### 3.2 Event-Driven yang Sering Digunakan

Karena prinsip kerja program dikendalikan berdasarkan kejadian, maka diperlukan pemahaman terhadap *event-event* yang sering digunakan untuk beberapa object atau komponen. Berikut ini adalah *event-event* yang dimaksudkan:

1. Click

Kejadian (*event*) ini dimaksudkan untuk mengendalikan saat objek yang bersangkutan dipilih (diklik).

2. DbClick

Kejadian (*event*) ini dimaksudkan untuk mengendalikan saat objek yang bersangkutan dilakukan klik dua kali (*double click*).

3. InteractiveChange

Kejadian (*event*) ini dimaksudkan untuk mengendalikan saat objek yang bersangkutan diisikan datanya (datanya diubah-ubah nilainya).

4. LostFocus

Kejadian (*event*) ini dimaksudkan untuk mengendalikan objek saat fokus telah berpindah dari objek yang bersangkutan. Misalnya yang tadinya mengisi data, lalu telah selesai mengisinya dan pindah ke objek lain.

5. Load

Kejadian (*event*) ini dimaksudkan untuk mengendalikan objek (khusus *form*) saat form dibangkitkan atau dipanggil untuk pertama kali.

6. Active

Kejadian (*event*) ini dimaksudkan untuk mengendalikan objek (khusus *form*) saat setiap kali form aktifkan.

### 3.3 Praktek

Sebelum memulai membuat aplikasi untuk praktek pada kesempatan ini, mohon pastikan *directory* yang aktif adalah *directory* kerja Anda yang sesuai saat ini (sesi3). Jika belum, mohon lakukan dengan cara menetapkan *directory* yang aktif yang telah dipelajari pada modul terdahulu.

Pastikan juga saat ini tidak sedang membuka sebuah **Project** (tampil project manajer). Hal ini dibutuhkan karena praktek kali ini akan membuat project yang baru.

Setelah semua dalam kondisi normal, lakukanlah proses mebuat Project baru dengan langkah sebagai berikut:

1. Tampilkan menu **File**, pilih menu **New** atau klik icon New
2. Saat ditampilkan kotak dialog, pilih **Project** dan klik tombol **New File**
3. Beri nama Project yang dibuat dengan nama Proj3 dan simpan pada folder kerja yang bernama Sesi3, lanjutkan dengan mengklik tombol **Save**.



1. Penggunaan Label, TextBox, OptionGroup, Command Button

5. Klik Tab Page Documents
6. Klik pilihan Form
7. Klik tombol New
8. Pilihan New Form
9. Rancanglah form seperti gambar di bawah. Pada form tersebut digunakan sebanyak 7 label (semua yang berupa tulisan mulai dari tulisan Nomor Induk Karyawan sampai Jumlah Terima). Selain Label juga digunakan sebanyak 5 TextBox, 2 Optiongroup dan 1 CommandButton.

The image shows a Windows application window titled "Program Penggunaan OptionGroup". Inside the window, there is a form with the following elements:

- Nomor Induk Karyawan**: A label followed by a text box labeled "Text1".
- Nama Karyawan**: A label followed by a text box labeled "Text2".
- Golongan**: A label followed by a radio button group with four options: I, II, III, and IV. Option I is selected.
- Gaji Pokok**: A label followed by a text box labeled "Text3".
- Status Pegawai**: A label followed by a radio button group with two options: "Tetap" (selected) and "Honorer".
- Bonus**: A label followed by a text box labeled "Text4".
- Jumlah Terima**: A label followed by a text box labeled "Text5".
- Tutup**: A button located at the bottom right of the form.

Gambar 3.1 Rancangan Form Menggunakan Label, TextBox, OptionGroup, Command Button

- a. Untuk Label, yang perlu dirubah hanya bagian Caption saja, yaitu sesuai tulisan yang diinginkan
- b. Untuk TextBox1 yang dirubah MaxLength menjadi 9
- c. Untuk TextBox2 yang dirubah MaxLength menjadi 30
- d. Untuk OptionGroup1 diganti isinya dengan cara klik kanan, lalu pilih Builder, lalu pada bagian Number of Button dirubah menjadi 4 (empat). Pada bagian kolom Caption diisikan pilihan yang akan ditampilkan (I, II, III, IV)
- e. Untuk OptionGroup2 juga diganti isinya dengan cara klik kanan, lalu pilih Builder, pada bagian kolom Caption diisikan dengan pilihan yang akan ditampilkan (Tetap, Honorer)
- f. Untuk TextBox3 sampai TextBox5 yang dirubah adalah Alignment diganti Right, selanjutnya yang diatut adalah Format dan InputMask diisikan 99,999,999
- g. Untuk Command Button yang dirubah adalah Caption (ditulis (\<Tutup) dan Name diganti dengan TblTutup

10. Tulislah kode program sesuai dengan aturan berikut ini

- i. Saat Form ditampilkan, maka setiap object diberi nilai awal
- ii. Saat memilih golongan, maka Gaji Pokok akan tampil secara otomatis dengan ketentuan:
  - b. Jika Golongan I maka Gaji Pokok adalah 800000
  - c. Jika Golongan II maka Gaji Pokok adalah 600000
  - d. Jika Golongan III maka Gaji Pokok adalah 500000
  - e. Jika Golongan IV maka Gaji Pokok adalah 400000
- 3) Saat memilih Status Pegawai, maka Bonus dan Jumlah Terima akan tampil secara otomatis.
- 4) Saat memilih tombol Tutup maka Form akan ditutup

Untuk menjawab ketentuan di atas lakukan langkah-langkah sebagai berikut:

- 1) Pastikan yang aktif adalah Form
- 2) Cari pilihan Active, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut ini

```
thisform.text1.Value=" "
thisform.text2.Value=" "
thisform.text3.Value=0
thisform.text4.Value=0
thisform.text5.Value=0
thisform.optiongroup1.Value=0
thisform.optiongroup2.Value=0
```

- 3) Klik OptionGroup1 (pilihan Golongan)
- 4) Cari pilihan Click pada Properties, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut ini

```
DO case
CASE thisform.optiongroup1.Value=1
  thisform.text3.Value=800000
CASE thisform.optiongroup1.Value=2
  thisform.text3.Value=600000
CASE thisform.optiongroup1.Value=3
  thisform.text3.Value=500000
CASE thisform.optiongroup1.Value=4
  thisform.text3.Value=400000
ENDCASE
```

- 5) Klik OptionGroup2 (pilihan Satus Pegawai)
- 6) Cari pilihan Click pada Properties, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut ini:

```
IF thisform.optiongroup2.Value=1
  thisform.text4.Value=0.1*thisform.text3.Value
ELSE
  thisform.text4.Value=0
ENDIF
thisform.text5.Value=thisform.text3.Value+thisform.text4.Value
```

- 7) Klik CommandButto1 (tombol Tutup)
- 8) Cari pilihan Click pada Properties, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut ini

```
thisform.Release()
```

## 11. Jalankan program

### 2. Penggunaan Label, TextBox, ListBox, Command Button

Form yang akan dibuat berikut ini hampir sama dengan form sebelumnya. Hanya saja yang berbeda adalah pada pilihan golongan yang tadinya menggunakan OptionGroup, sekarang akan digunakan ListBox dengan tujuan supaya dapat dilihat perbedaannya baik dari segi tampilan maupun kode program yang harus ditulis.

Untuk mempercepat menghasilkan Form yang harus dibuat, dimana bentuk dan komponen yang digunakan hampir sama dapat dilakukan dengan dua cara. Cara pertama adalah dengan cara **SaveAs**, dan cara kedua adalah meng-Copy.

Untuk cara pertama, maka bukalah Form1 yang sudah selesai dikerjakan pada praktek terdahulu. Lalu tampilkan menu File, selanjutnya pilih menu **SaveAs**. Berilah nama dengan **Form2**. Setelah itu tutup Form1 dan kembali ke **Project Manajer**.

- a. Klik menu **Add** untuk menambahkan **Form2** hasil Save As ke Project.
- b. Bukalah Form2 yang telah dihasilkan dari menyimpan Form1 ke nama baru.
- c. Hapus objek OptionGroup1 (pilihan untuk golongan)
- d. Ambil Objek **ListBox**, lalu letakkan pada bagian golongan sebagai pengganti OptionGroup1 yang telah dihapus
- e. Isilah Listbox1 dengan cara klik kanan, lalu pilih Builder. Saat ditampilkan kotak dialog, gantilah pilihan pada **Fill the list with** dengan **Data entered by hand**, lalu pada bagian Column1 isikan pilihan yang akan ditampilkan pada ListBox yaitu I, II, III, IV. Setelah selesai, klik tombol OK untuk kembali ke desain form.
- f. Karena aturan urutan untuk gaji adalah tampil secara otomatis saat memilih golongan, maka klik objek **ListBox1**, lalu cari pilihan Click pada Properties, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut:

```
DO case
CASE thisform.list1.ListIndex=1
  thisform.text3.Value=800000
CASE thisform.list1.ListIndex=2
  thisform.text3.Value=1000000
CASE thisform.list1.ListIndex=3
  thisform.text3.Value=1500000
CASE thisform.list1.ListIndex=4
  thisform.text3.Value=1800000
ENDCASE
```

Gambar 3.2 Rancangan Form Label, TextBox, ListBox, Command Button

- g. Lakukan perubahan kode pada Form Active dengan menghilangkan kode terkait OptionGroup1 diubah dengan object List1, sehingga kode pada From Active menjadi:

Object: Form1 Procedure: Activate

```

thisform.text1.Value=""
thisform.text2.Value=""
thisform.text3.Value=0
thisform.text4.Value=0
thisform.text5.Value=0
thisform.list1.Value=""
thisform.optiongroup2.Value=0

```

Bagian ini saja yang diubah

- h. Jalankan Program

### 3. Penggunaan Label, TextBox, ComboBox, Command Button

Form yang akan dibuat berikut ini hampir sama dengan form sebelumnya. Hanya saja yang berbeda adalah pada pilihan golongan yang tadinya menggunakan **ListBox**, sekarang akan digunakan **ComboBox** dengan tujuan supaya dapat dilihat perbedaannya baik dari segi tampilan maupun kode program yang harus ditulis.

Untuk mempercepat menghasilkan Form maka bukalah Form2 yang sudah selesai dikerjakan pada praktek terdahulu. Lalu tampilkan menu File, selanjutnya pilih menu **SaveAs**. Berilah nama dengan **Form3**. Setelah itu tutup Form2 dan kembali ke Project manajer.

- Klik menu **Add** untuk menambahkan **Form3** hasil Save As ke Project
- Bukalah Form3 yang telah dihasilkan dari menyimpan Form2 ke nama baru.
- Hapus objek **ListBox1** (pilihan untuk golongan)
- Ambil Objek **ComboBox**, lalu letakkan pada bagian golongan sebagai pengganti ListBox1 yang telah dihapus
- Isilah Combo1 dengan cara klik kanan, lalu pilih **Builder**. Saat ditampilkan kotak dialog, gantilah pilihan pada **Fill the list with** dengan **Data entered by hand**, lalu pada bagian Column1 isikan pilihan yang akan ditampilkan pada ComboBox yaitu I, II, III, IV. Setelah selesai, klik tombol **OK** untuk kembali ke desain form.



- f. Karena aturan urutan untuk gaji adalah tampil secara otomatis saat memilih golongan, maka klik objek Combo1, lalu cari pilihan Click pada Properties, kemudian klik dua kali. Saat ditampilkan layar penulisan kode, ketikkan kode program berikut:

DO case

CASE thisform.combo1.Value="IV"

thisform.text3.Value=1800000

CASE thisform.combo1.Value="III"

thisform.text3.Value=1500000

CASE thisform.combo1.Value="II"

thisform.text3.Value=1000000

CASE thisform.combo1.Value="I"

thisform.text3.Value=800000

ENDCASE

Gambar 3.3 Rancangan Form Label, TextBox, ComboBox, Command Button

- g. Lakukan perubahan kode pada Form Active dengan menghilangkan kode terkait List1 diubah dengan object Combo1, sehingga kode pada Form Active menjadi:

```
Object: Form1 Procedure: Activate

thisform.text1.Value=""
thisform.text2.Value=""
thisform.text3.Value=0
thisform.text4.Value=0
thisform.text5.Value=0
thisform.combo1.Value=""
thisform.optiongroup2.Value=0
```

- h. Jalankan program

4. Kasus tentang layanan pelanggan di PDAM (buatlah rancangan form seperti Gambar 3.4, dan kerjakan program berdasarkan ketentuan yang ditetapkan).

Form5

**Perusahaan Daerah Air Minum (PDAM)**  
Jl. Jetis No. 32 Yogyakarta

Nama Pelanggan: Text1

Alamat Pelanggan: Text2

Status Pelanggan: Combo1

Jumlah Pemakaian: Text3

Biaya Tarif Dasar: Text4

Biaya Pemakaian: Text5

Jumlah Biaya: Text6

Potongan: Text7

Jumlah Bayar: Text8

Gambar 3.4 Rancangan Form Untuk Studi Kasus di PDAM

Ketentuan :

- Nama pelanggan, alamat pelanggan, status pelanggan dan jumlah pemakaian diisi melalui *keyboard*.
- Pada combo untuk isian Status Pelanggan diisi dengan pilihan : Umum, Perusahaan, Yayasan
- Saat memilih (**Click**) pada combo Status Pelanggan, maka Biaya Tarif Dasar akan tampil secara otomatis dengan aturan :
  - Jika Status Pelanggan="Umum" maka Biaya Tarif Dasar = 35000
  - Jika Status Pelanggan ="Perusahaan" maka Biaya Tarif Dasar = 55000
  - Jika Status Pelanggan ="Yayasan" maka Biaya Tarif Dasar = 15000
- Saat mengisi Jumlah Pemakaian (**InteractiveChange**) maka Biaya akan tampil secara otomatis, dengan ketentuan :
  - Jika Jumlah Pemakaian lebih kecil dari 100 M<sup>3</sup>, maka Biaya Pemakaian 20% dari Biaya Tarif Dasar, selain itu jika Jumlah Pemakaian lebih besar atau sama dengan 100 M<sup>3</sup>, maka Biaya Pemakaian = 10% dari Biaya Tarif Dasar.
  - Jumlah Biaya = Biaya Tarif Dasar + Biaya Pemakaian
  - Jika Jumlah Pemakaian lebih besar dari 500 M<sup>3</sup>, maka diberikan Potongan = 5% dari Jumlah Biaya, selain itu Potongan = 0
  - Jumlah Bayar = Jumlah Biaya - Potongan

# Bab 4

## Bekerja Dengan *Database* dan *Free Table*

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami tentang bekerja dengan *Database* dan *Free Table* sehingga mampu membangun aplikasi dengan dukungan database.

Pada pertemuan ini akan diperkenalkan bagaimana membuat *Database*, dan membuat Tabel. Selain itu juga diberikan juga beberapa latihan tambahan.

### 4.1 Database dan Free Tabel

Kedua fasilitas di atas merupakan fasilitas yang tersedia pada Visual FoxPro untuk dapat mengelola data secara terstruktur dan dinamis. Dalam implementasinya, dapat menggunakan salah diantaranya, sesuai dengan kebutuhan aplikasi. Pada fasilitas database akan diciptakan beberapa tabel, dengan kata lain semua tabel yang dibuat dibungkus pada sebuah *database*. Sedangkan fasilitas **Free Table**, semua tabel yang dibuat tidak ada pembungkusnya, namun proses-proses yang dilakukan terhadap tabel semuanya sama.

### 4.2 Database

Database merupakan tempat penyimpanan data suatu informasi. Dengan *database* Anda dapat mengorganisasikan dan menggunakan beberapa tabel secara bersamaan.

#### 4.2.1 Membuat *Database*

Untuk membuat *database* dapat dilakukan dengan dua cara. Cara pertama dengan menggunakan **Command Prompt**, dan cara kedua dengan menggunakan fasilitas yang ada di **Project Manajer**.

##### 1. Menggunakan Command Prompt

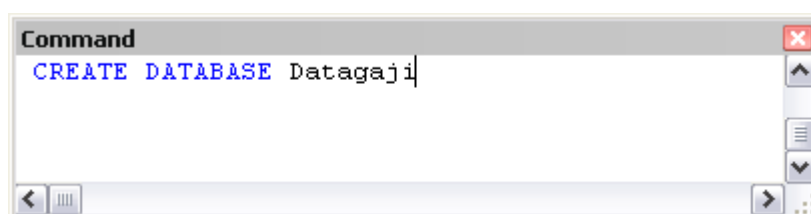
Jika menggunakan cara pertama, maka ketikkan perintah pada **Command Prompt** yaitu:

**Create Database <namadatabase>**

Misalkan akan dibuat sebuah *database* yang diberi nama DATAGAJI, maka ketikkan perintah:

**Create Database Datagaji**

Untuk lengkapnya dapat dilihat pada Gambar 4.1

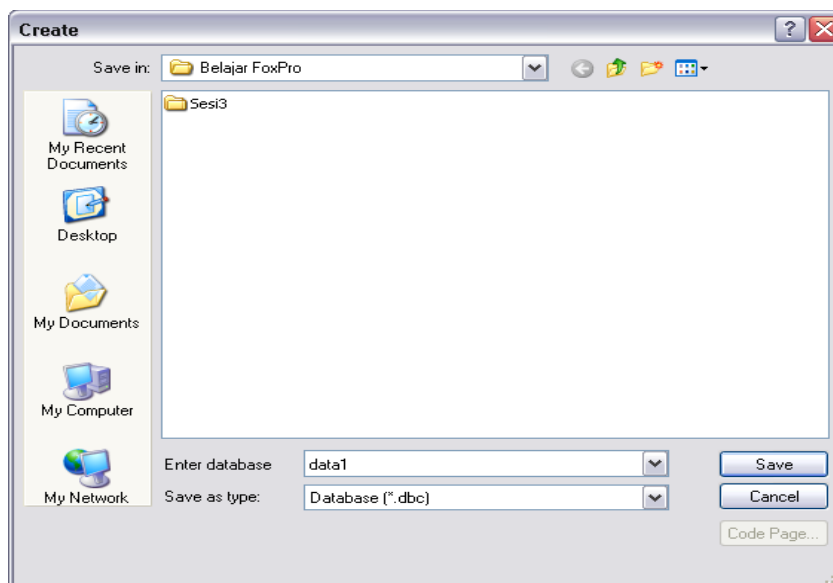


Gambar 4.1 *Command Prompt* Untuk Membuat *Database*

## 2. Menggunakan Project Manajer

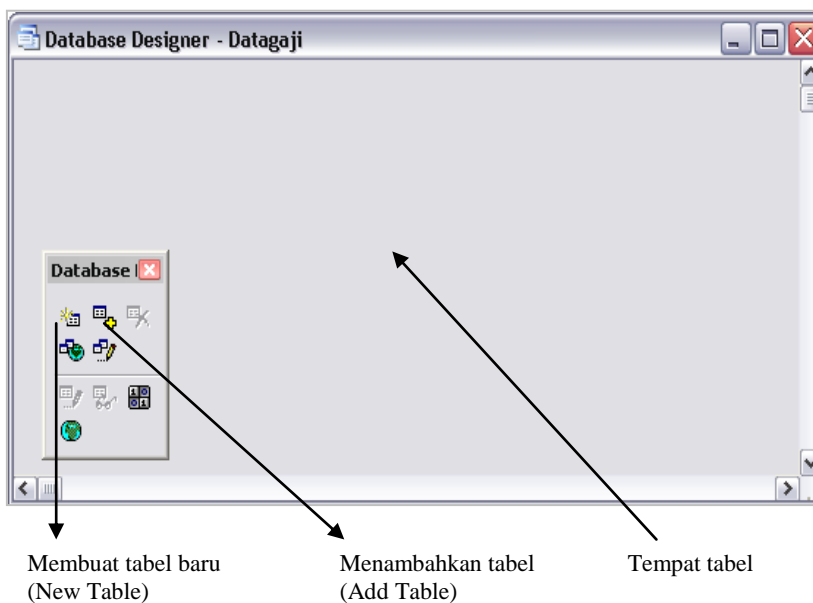
Saat berada pada Project Manajer, untuk membuat database lakukanlah langkah sebagai berikut:

- Klik Tab **Data**
- Klik pilihan **Database**
- Klik tombol **New**
- Pilih **New Database**, sehingga akan ditampilkan kotak dialog untuk memberi nama file *database* yang akan dibuat seperti terlihat pada Gambar 4.2.



Gambar 4.2 Kotak Dialog Membuat Database

- Ketikkan nama *Database* yang akan dibuat (secara *default* akan ditawarkan nama Data1 atau Data2, Data3, dst). Setelah mengetik nama lanjutkan dengan mengklik tombol **Save**, sehingga akan ditampilkan **Database Designer** seperti terlihat pada Gambar 4.3.



Gambar 4.3 Database Designer

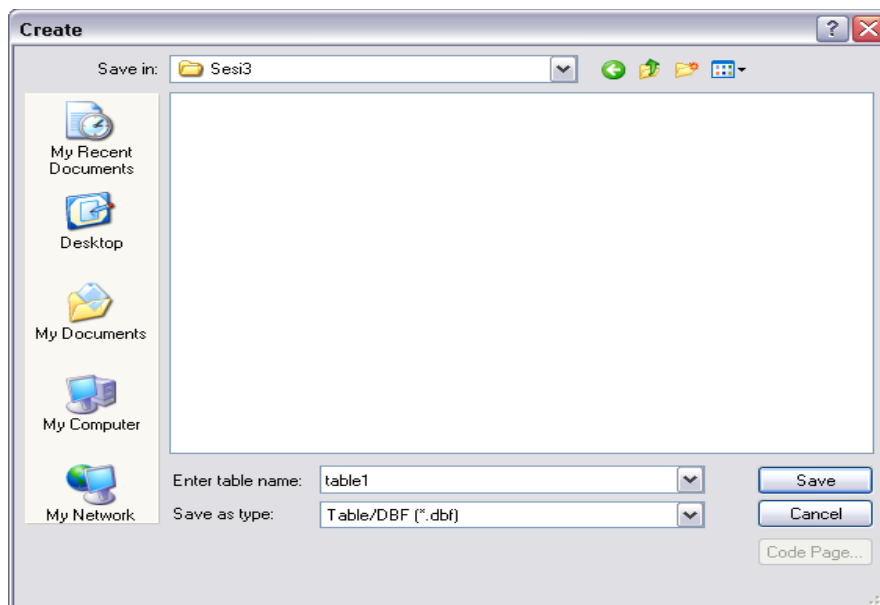


- f. Pada saat ditampilkan *database designer*, maka sudah bisa melakukan proses-proses yang terkait dengan *database*, misalnya membuat tabel, menambahkan tabel, dan lain sebagainya.

#### 4.2.2 Membuat Tabel Baru

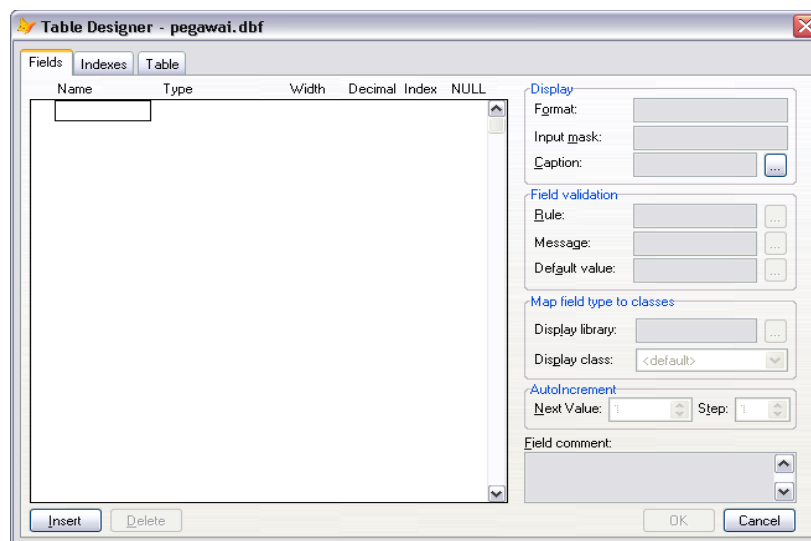
Setelah *database* tercipta, maka selanjutnya perlu disiapkan tabel-tabel yang akan digunakan untuk menampung data sesuai dengan struktur yang diinginkan.

1. Untuk itu klik icon **New Tabel** (lihat Gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan sistem) dan **New Table** (membuat tabel secara manual). Untuk tahap pemula sebaiknya pilih saja **New Table** agar lebih tahu seperti apa proses yang sesungguhnya.
2. Setelah memilih **New Table**, akan ditampilkan kotak dialog seperti Gambar 4.4



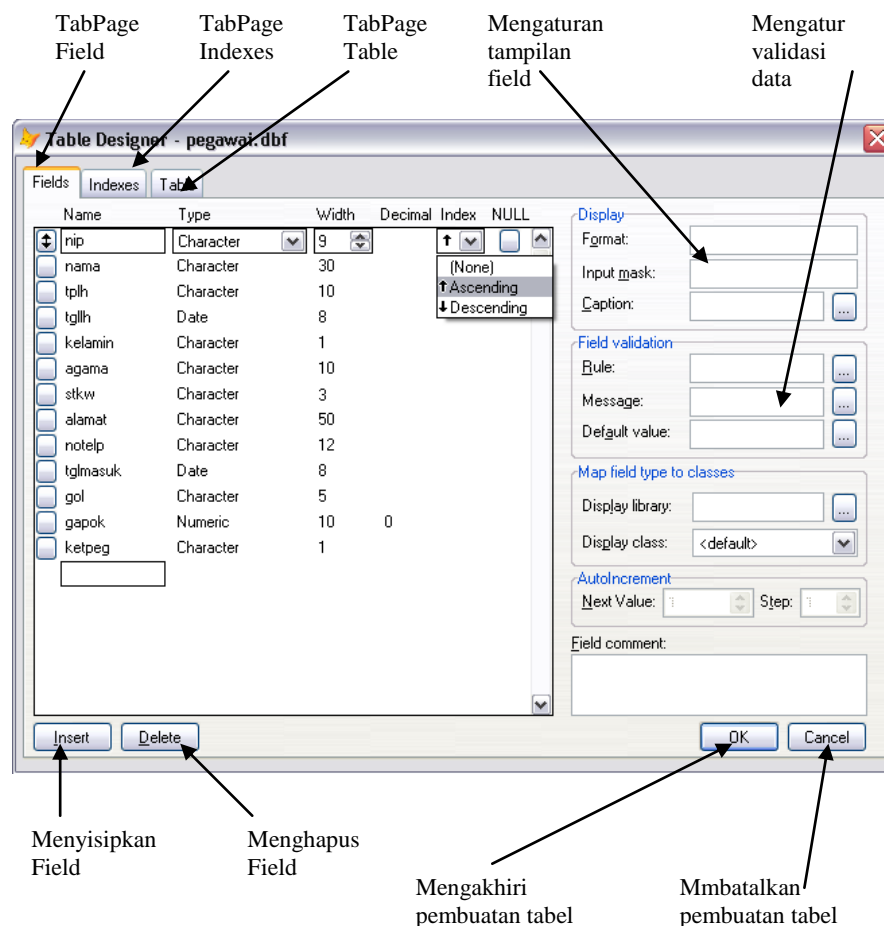
Gambar 4.4 Kotak Dialog Membuat Tabel

3. Ketikkan nama tabel yang akan dibuat (secara *default* ditawarkan nama Table1, Table2, table3, dst). Setelah mengetik nama lanjutkan dengan mengklik tombol **Save**, sehingga akan ditampilkan **Table Designer** seperti terlihat pada Gambar 4.5.



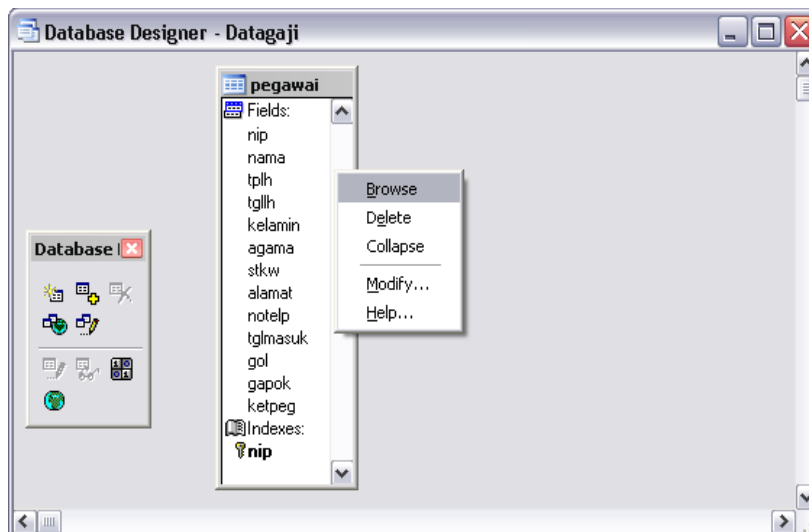
Gambar 4.5 Kotak Dialog Table Designer

4. Buatlah nama field (pada kolom Name) , jenis data (pada kolom Type), ukuran data (pada kolom Width), titik desimal (pada kolom **Decimal**, khusus data **Numeric**), metode pengurutan (pada kolom Index) sesuai dengan kebutuhan.
5. Pada kolom Index perlu ditetapkan metode pengurutan yang diinginkan dari field yang bersangkutan apakah **A (Ascending)** = dari urutan kecil ke urutan besar ( A – Z)) atau **D (Descending)** =dari urutan besar ke kecil (Z – A)). Sebagai catatan field yang menjadi kunci (baik kunci utama (*primary key*) maupun kunci tamu (*foreign key*)) perlu diindex. Tapi perlu juga diingat bahwa untuk *field* yang dijadikan kunci utama perlu ditindaklanjuti dengan memilih jenis kunci pada Tab Page **Indexes**. Secara lengkap bentuk proses pengisian *field* dan lain sebagai dapat dilihat pada Gambar 4.6



Gambar 4.6 Proses Pengelolaan Pembuatan Tabel

6. Setelah selesai menentukan hal-hal yang berkaitan dengan tabel, maka untuk mengakhiri klik tombol **OK** yang menunjukkan bahwa proses pembuatan tabel sudah selesai. Berikutnya akan kembali ke *database designer*, seperti Gambar 4.7.



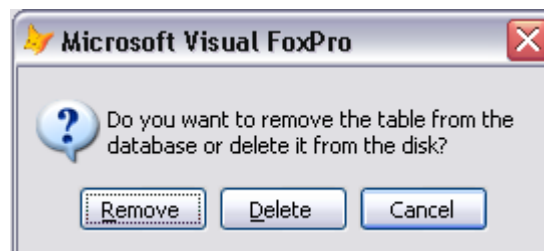
Gambar 4.7 Database Designer Setelah Membuat Tabel

Pada saat ditampilkan Database Designer seperti gambar di atas, dapat melakukan proses pembuatan tabel yang lain atau tabel berikutnya dengan langkah-langkah seperti yang sudah dicobakan di atas.

Namun bila dianggap tabel untuk *database* yang dibuat sudah mencukupi, maka dapat melakukan proses pengelolaan tabel seperti menampilkan data (**browse**), menghapus tabel (**delete**), merubah struktur tabel (**modify**).

### 4.2.3 Menghapus Tabel

Tabel yang sudah diciptakan pada *database*, memungkinkan untuk dihapus. Saat berada seperti Gambar 4.7, klik kanan pada tabel yang diinginkan, lalu pilih **Delete**, sehingga akan ditampilkan kotak dialog seperti Gambar 4.8



Gambar 4.8 Kotak Dialog Menghapus Tabel

Jika dipilih **Remove**, maka tabel akan dibuang dari *database*, tetapi tidak dihilangkan dari disk. Sedangkan jika dipilih **Delete**, maka tabel akan dibuang dari *database* sekaligus dibuang dari disk.

### 4.2.4 Merubah Struktur Tabel

Tabel yang sudah terlanjur dibuat dengan struktur tertentu, masih memungkinkan untuk dilakukan perubahan sesuai dengan kondisi yang terjadi dalam pengembangan aplikasi. Saat tampilan seperti Gambar 4.7, klik kanan pada tabel yang akan dirubah, sehingga akan masuk lagi ke tampilan proses pengelolaan tabel seperti Gambar 4.6. Setelah masuk ke tampilan proses pengelolaan tabel, silahkan melakukan perubahan yang diinginkan, misalnya menambahkan *field*, menghapus *field*, atau merubah hal-hal yang lain.

### 4.3 Free Table

Seperti yang telah dikemukakan di atas, bahwa yang menjadi perbedaan mendasar antara database dengan free table adalah kalau pada **database**, tabel-tabel dibungkus dengan sebuah database, sedangkan pada **free table**, tabel-tabel tidak dibungkus. Dari pernyataan demikian, maka jika pada database untuk menggunakan tabel terlebih dahulu harus dibuka *databasenya*. Sedangkan pada *free table* jika menggunakan tabel langsung dibuka. Proses-proses yang lain sama seperti yang telah dibahas di atas.

### 4.4 Praktek

Kegiatan praktek ini akan berkelanjutan untuk beberapa sesi berikutnya, maka sebaiknya kerjakanlah pada *folder* baru, dalam hal ini **Sesi4**. Pada sesi ini, akan mempraktekkan membuat *database* dan tabel.

Untuk itu sebelumnya awailah dengan cara membuat Project baru dan beri nama Project dengan **ProHotel**.

#### 4.4.1 Membuat Database Baru

Saat berada pada Proejct Manajer, untuk membuat *database* lakukanlah langkah berikut:

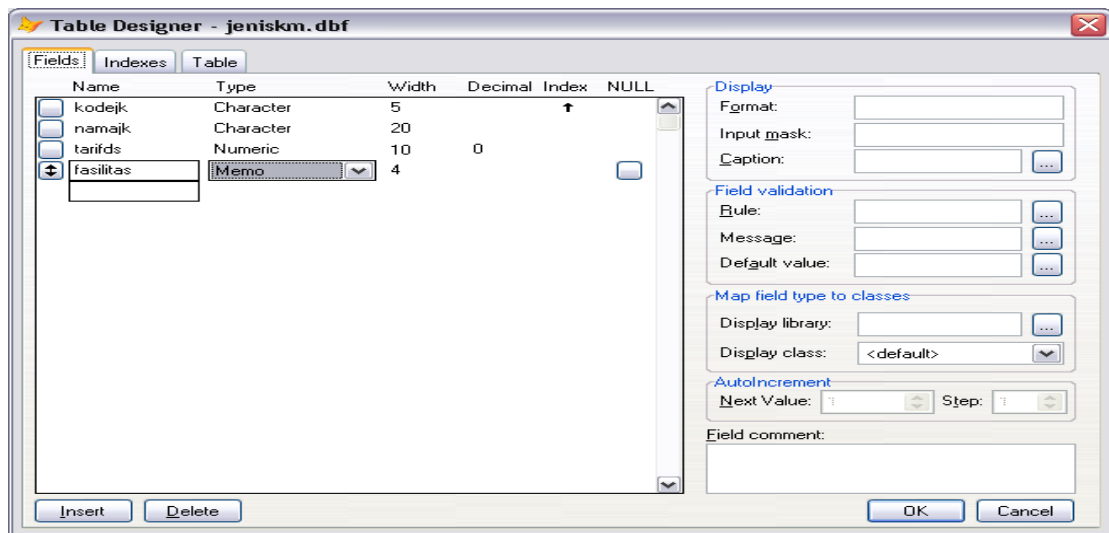
- a. Klik Tab **Data**
- b. Klik pilihan **Database**
- c. Klik tombol **New**
- d. Pilih **New Database**, sehingga ditampilkan kotak dialog untuk memberi nama file *database* yang akan dibuat.
- e. Ketikkan nama *Database* yang akan dibuat (secara default akan ditawarkan nama Data1 atau Data2, Data3, dst). Gantilah dengan nama **DataHotel**, sehingga akan ditampilkan Database Designer (lihat Gambar 4.3).
- f. Pada saat ditampilkan *database designer*, maka sudah bisa melakukan proses-proses yang terkait dengan *database*, misalnya membuat tabel, menambahkan tabel, dan lain sebagainya.

#### 4.4.2 Membuat Tabel

Tabel yang diperlukan untuk **Project** hotel yang akan dibuat terdiri dari beberapa tabel. Untuk itu buatlah secara berturut-turut seluruh tabel yang dibutuhkan. Langkah untuk menciptakan tabel dari posisi database designer adalah sebagai berikut:

##### 1. Membuat Tabel Jenis Kamar

- a. Klik icon **New Tabel** (lihat Gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu Tabel Wizard (membuat tabel dengan bantuan sistem) dan *New Table* (membuat tabel secara manual). Untuk itu pilih saja **New Table**.
- b. Setelah memilih **New Table**, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama JENISKM dan lanjutkan dengan mengklik tombol Save, sehingga akan ditampilkan **Table Designer**.
- c. Buatlah nama field (pada kolom Name) , jenis data (pada kolom Type), ukuran data (pada kolom Width), titik desimal (pada kolom Decimal, khusus data Numeric), metode pengurutan (pada kolom Index) sesuai dengan kebutuhan. Field-field untuk tabel jenis kamar adalah: **kodejk, namajk, tarifds, fasilitas**. Field yang diindeks adalah **kodejk**

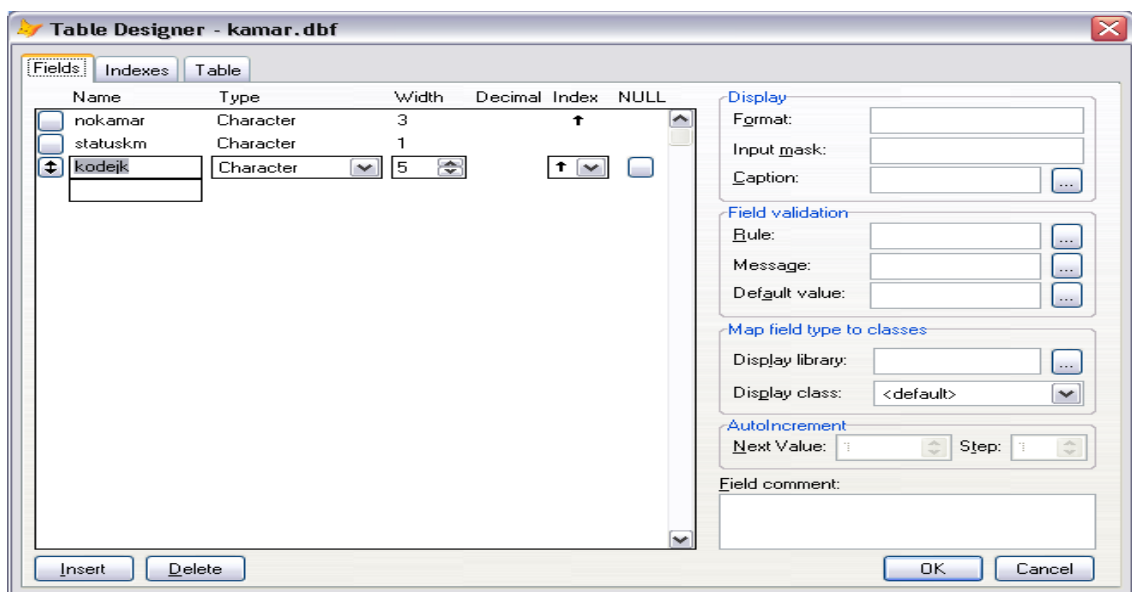


Gambar 4.9 Kotak Dialog Pembuatan Tabel Jenis Kamar

- d. Klik Tab Indexes untuk menentukan jenis kunci. Klik pada kolom **Type** dan pilih **Primary** pada baris kodejk karena *field kodejk* sebagai kunci utama.
- e. Klik tombol **OK** jika sudah selesai membuat semua *field* dan ketentuan lain, dan kembali ke *database designer*.

## 2. Membuat Tabel Kamar

- a. Klik icon New Tabel (lihat Gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan system) dan **New Table** (membuat tabel secara manual). Untuk itu pilih saja **New Table**.
- b. Setelah memilih **New Table**, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama KAMAR dan lanjutkan dengan mengklik tombol Save, sehingga akan ditampilkan Table Designer.
- c. Buatlah nama *field* (pada kolom Name) , jenis data (pada kolom Type), ukuran data (pada kolom Width), titik desimal (pada kolom Decimal, khusus data Numeric), metode pengurutan (pada kolom Index) sesuai dengan kebutuhan. Field yang diindeks adalah field **nokamar** dan field **kodejk**

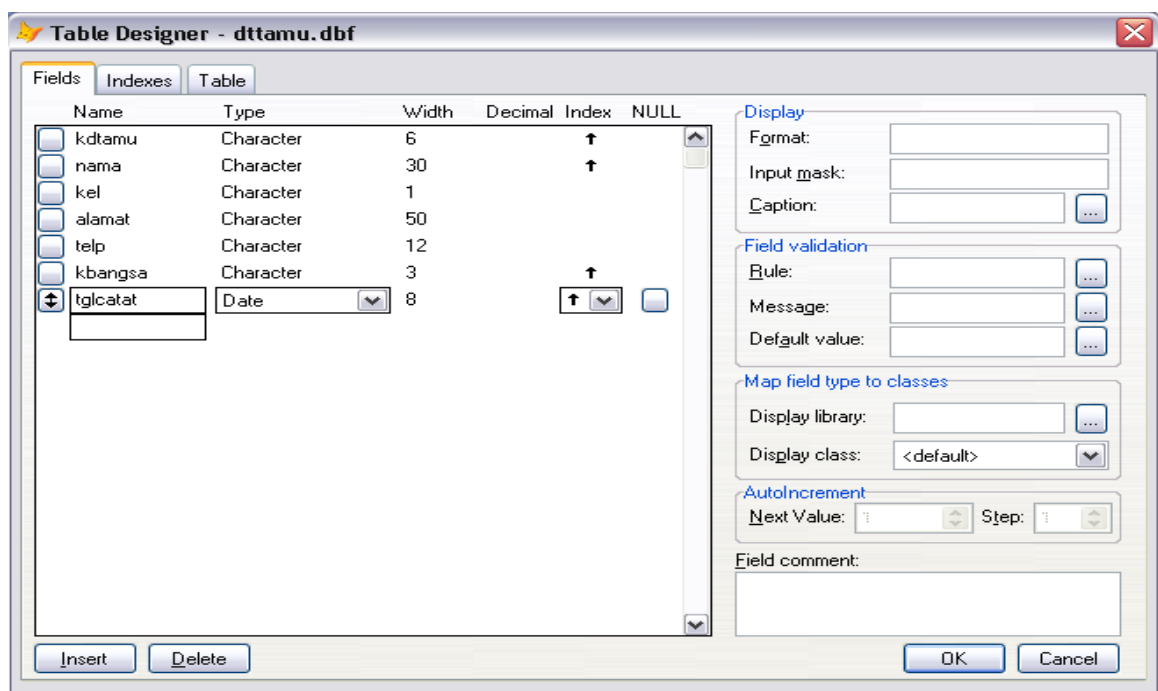


Gambar 4.10 Kotak Dialog Pembuatan Tabel Kamar

- d. Klik Tab **Indexes** untuk menentukan jenis kunci. Klik pada kolom **Type** dan pilih **Primary** pada baris nokamar karena field nokamar sebagai kunci utama. Sedangkan pada baris kodejk biarkan tetap regular karena field kodejk sebagai kunci tamu.
- e. Klik tombol **OK** jika sudah selesai membuat semua field dan ketentuan lain, dan kembali ke database designer.

### 3. Membuat Tabel Data Tamu

- a. Untuk itu klik icon **New Tabel** (lihat Gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan sistem) dan **New Table** (membuat tabel secara manual). Untuk itu pilih saja **New Table**.
- b. Setelah memilih **New Table**, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama DTTAMU dan lanjutkan dengan mengklik tombol **Save**, sehingga akan ditampilkan *Table Designer*.
- c. Buatlah nama field (pada kolom **Name**) , jenis data (pada kolom **Type**), ukuran data (pada kolom **Width**), titik desimal (pada kolom **Decimal**, khusus data **Numeric**), metode pengurutan (pada kolom **Index**) sesuai dengan kebutuhan. Field yang diindeks adalah field **kdtamu**, field **nama**, field **kbangsa**, field **tgldatat**.

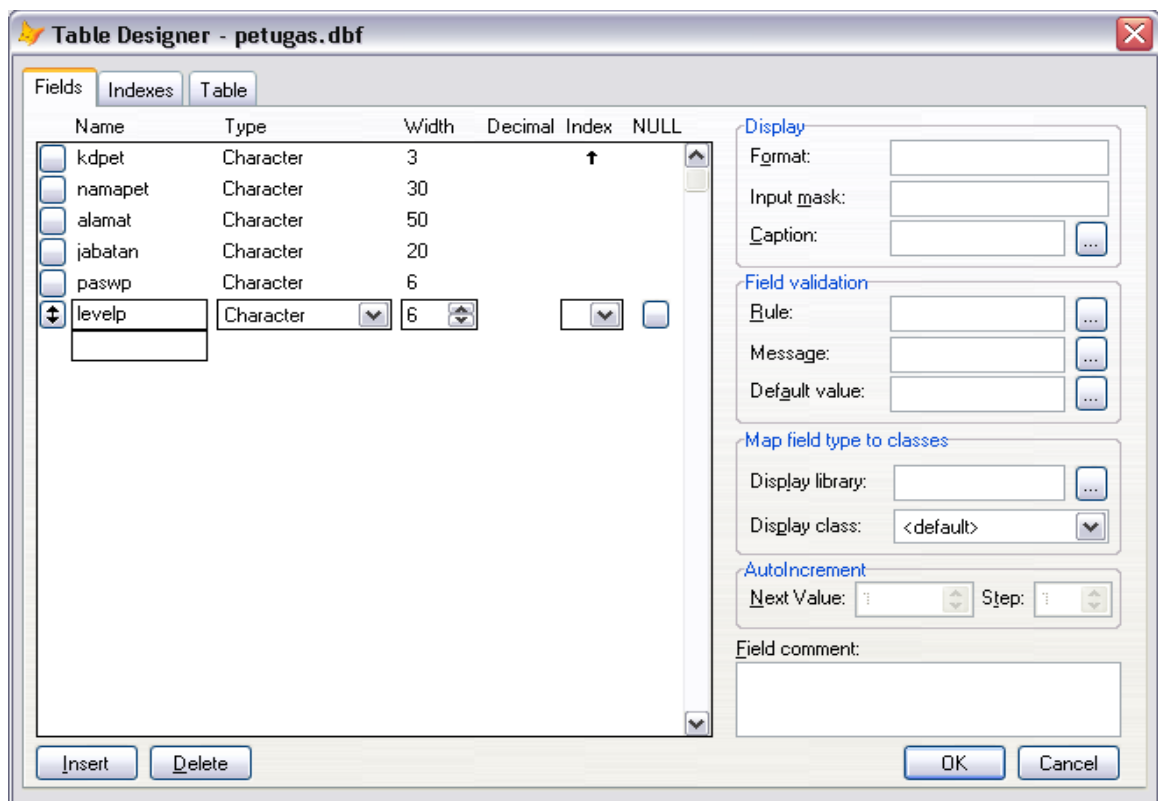


Gambar 4.11 Kotak Dialog Pembuatan Tabel Tamu

- d. Klik Tab **Indexes** untuk menentukan jenis kunci. Klik pada kolom **Type** dan pilih **Primary** pada baris kdtamu karena *field* kdtamu sebagai kunci utama. Pada baris nama, baris kbangsa, baris tgldatat biarkan saja tetap regular karena sifatnya untuk proses pengurutan biasa.
- e. Klik tombol **OK** jika sudah selesai membuat semua *field* dan ketentuan lain, dan kembali ke *database designer*.

#### 4. Membuat Tabel Petugas

- Untuk itu klik icon **New Tabel** (lihat Gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan sistem) dan **New Table** (membuat tabel secara manual). Untuk itu pilih saja **New Table**.
- Setelah memilih **New Table**, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama PETUGAS dan lanjutkan dengan mengklik tombol **Save**, sehingga akan ditampilkan **Table Designer**.
- Buatlah nama *field* (pada kolom Name) , jenis data (pada kolom **Type**), ukuran data (pada kolom Width), titik desimal (pada kolom Decimal, khusus data Numeric), metode pengurutan (pada kolom **Index**) sesuai dengan kebutuhan. *Field* yang diindeks adalah *field kdpet*.

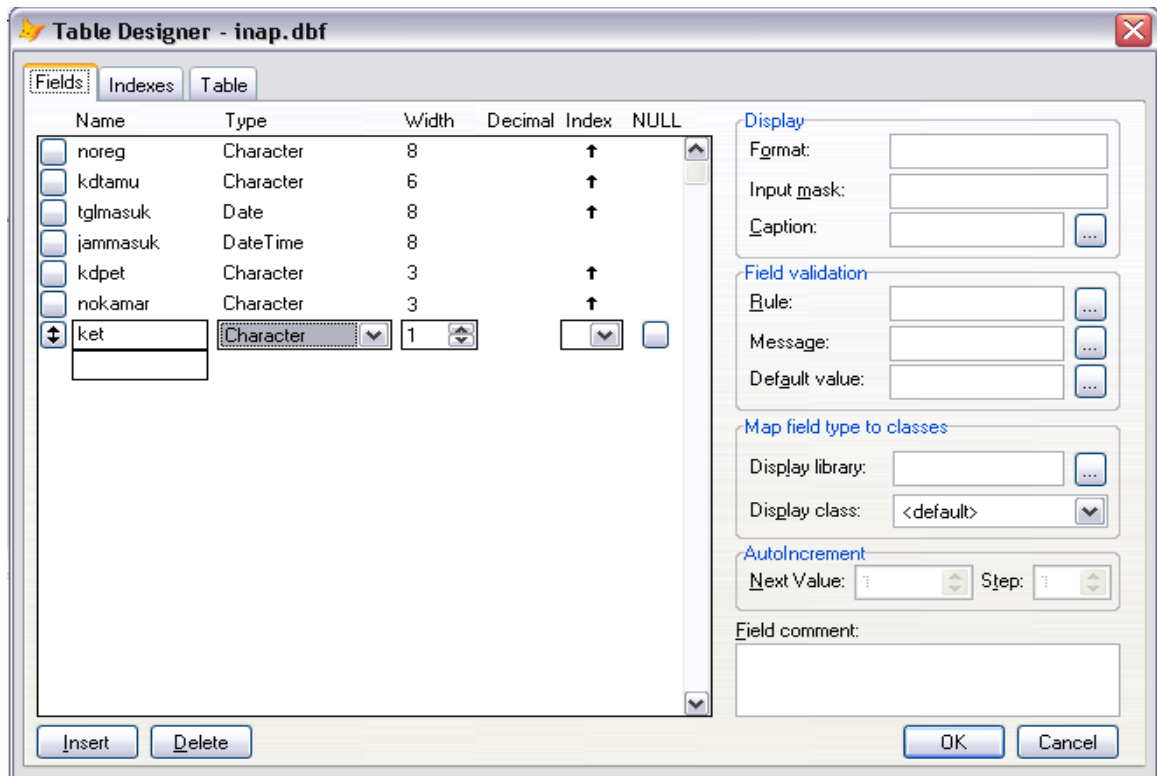


Gambar 4.12 Kotak Dialog Pembuatan Tabel Petugas

- Klik Tab **Indexes** untuk menentukan jenis kunci. Klik pada kolom **Type** dan pilih **Primary** pada baris kdpet karena *field* kdpet sebagai kunci utama.
- Klik tombol **OK** jika sudah selesai membuat semua *field* dan ketentuan lain, dan kembali ke *database designer*.

## 5. Membuat Tabel Inap

- Untuk itu klik icon **New Tabel** (lihat gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan sistem) dan **New Table** (membuat tabel secara manual). Untuk itu pilih saja **New Table**.
- Setelah memilih **New Table**, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama INAP dan lanjutkan dengan mengklik tombol **Save**, sehingga akan ditampilkan **Table Designer**.
- Buatlah nama *field* (pada kolom **Name**) , jenis data (pada kolom **Type**), ukuran data (pada kolom Width), titik desimal (pada kolom Decimal, khusus data Numeric), metode pengurutan (pada kolom **Index**) sesuai dengan kebutuhan. *Field* yang diindeks adalah *field* **noreg**, **kdtamu**, **tglmasuk**, **kdpet**, **nokamar**.



Gambar 4.13 Kotak Dialog Pembuatan Tabel Inap

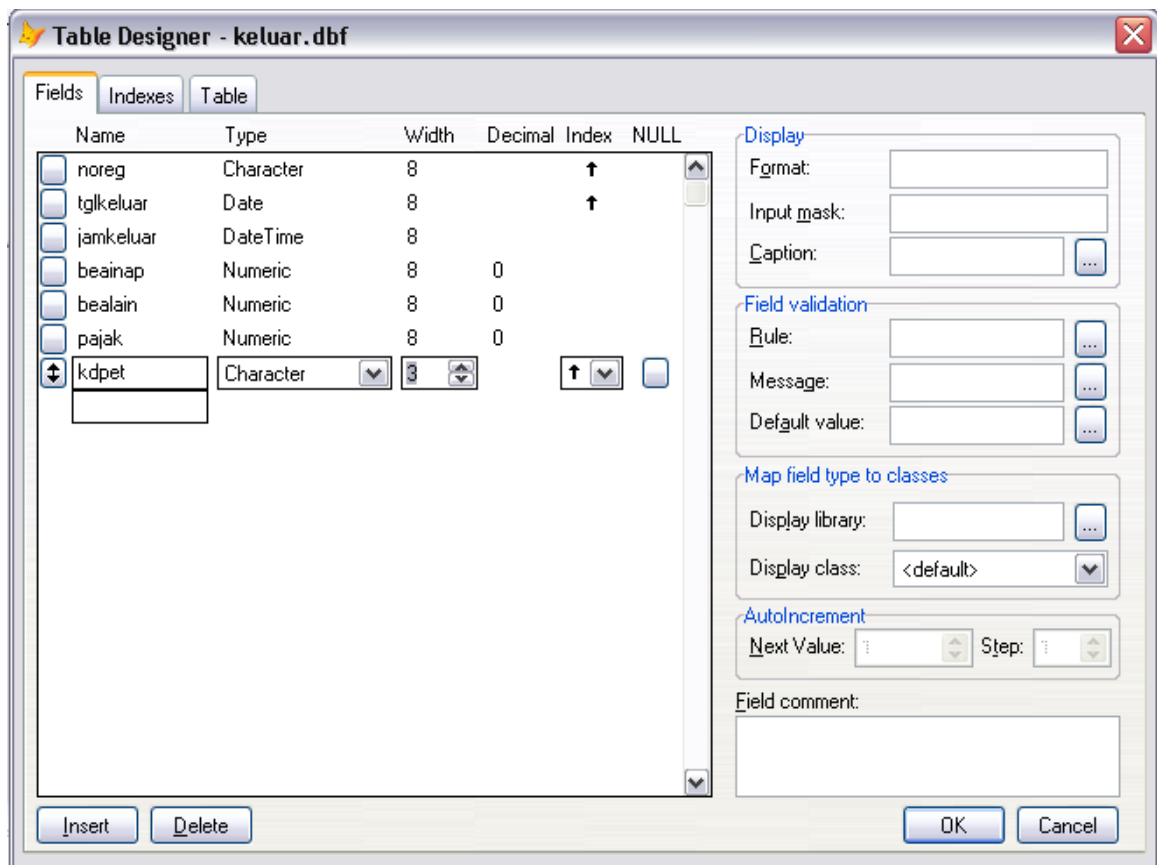
- Klik Tab Indexes untuk menentukan jenis kunci. Klik pada kolom Type dan pilih Primary pada kolom noreg karena field noreg sebagai kunci utama. Sedangkan yang lain biarkan tetap regular karena hanya sebagai foreign key atau hanya pengurutan biasa.
- Klik tombol OK jika sudah selesai membuat semua field dan ketentuan lain, dan kembali ke **database designer**.

## 6. Membuat Tabel Keluar

- Untuk itu klik icon **New Tabel** (lihat gambar 4.3) sehingga akan ditampilkan 2 pilihan yaitu **Tabel Wizard** (membuat tabel dengan bantuan sistem) dan **New Table** (membuat tabel secara manual). Untuk itu pilih saja **New Table**.



- b. Setelah memilih New Table, akan ditampilkan kotak dialog untuk memberikan nama tabel. Ketikkan nama tabel (secara default ditawarkan nama Table1, Table2, table3, dst). Gantilah dengan nama KELUAR dan lanjutkan dengan mengklik tombol Save, sehingga akan ditampilkan **Table Designer**.
- c. Buatlah nama field (pada kolom Name) , jenis data (pada kolom Type), ukuran data (pada kolom Width), titik desimal (pada kolom Decimal, khusus data Numeric), metode pengurutan (pada kolom Index) sesuai dengan kebutuhan. Field yang diindeks adalah field **noreg**, **tglkeluar**, **kdpet**.



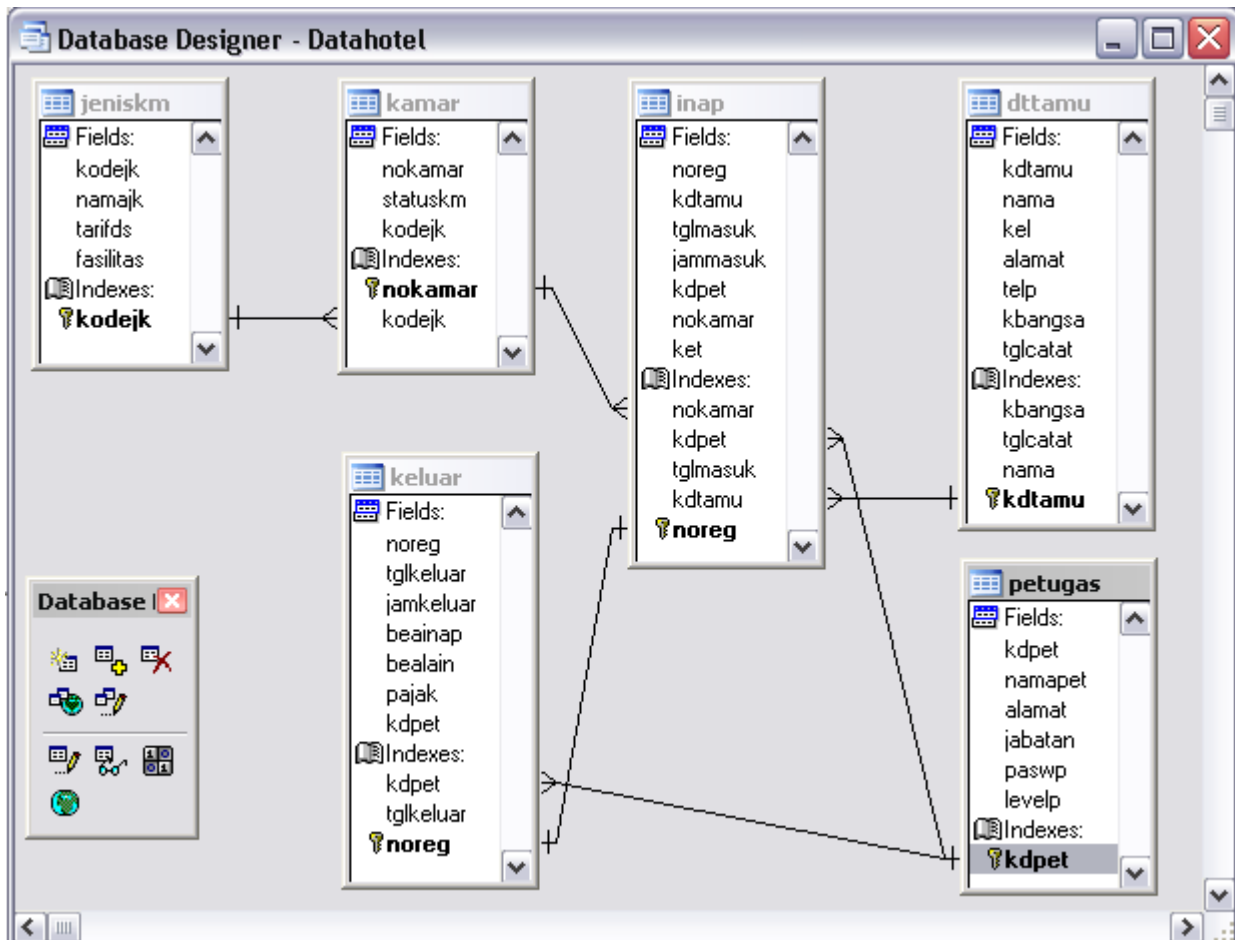
Gambar 4.14 Kotak Dialog Pembuatan Tabel Keluar

- d. Klik Tab **Indexes** untuk menentukan jenis kunci. Klik pada kolom **Type** dan pilih **Primary** pada kolom noreg karena field noreg sebagai kunci utama.
- e. Klik tombol **OK** jika sudah selesai membuat semua field dan ketentuan lain, dan kembali ke *database designer*.

### 4.4.3 Membuat Relasi

Setelah semua kebutuhan tabel yang digunakan untuk aplikasi, maka tabel-tabel tersebut dapat direlasikan. Dalam proses merelasikan tabel ini, hal yang perlu diperhatikan adalah harus paham tabel mana yang menjadi tabel induk dan tabel mana yang menjadi tabel anak.

Untuk merelasikan tabel dapat dilakukan dengan cara, arahkan field kunci pada tabel induk ke field kunci pada tabel anak. Hasil relasi untuk aplikasi perhotelan dapat dilihat pada Gambar 4.15.



Gambar 4.15 Tampilan Relasi Tabel Pada *Database Designer*

# Bab 5

## Mengelola Data

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami tentang bagaimana mengelola data, baik secara interaktif dengan memanfaatkan table maupun menggunakan form.

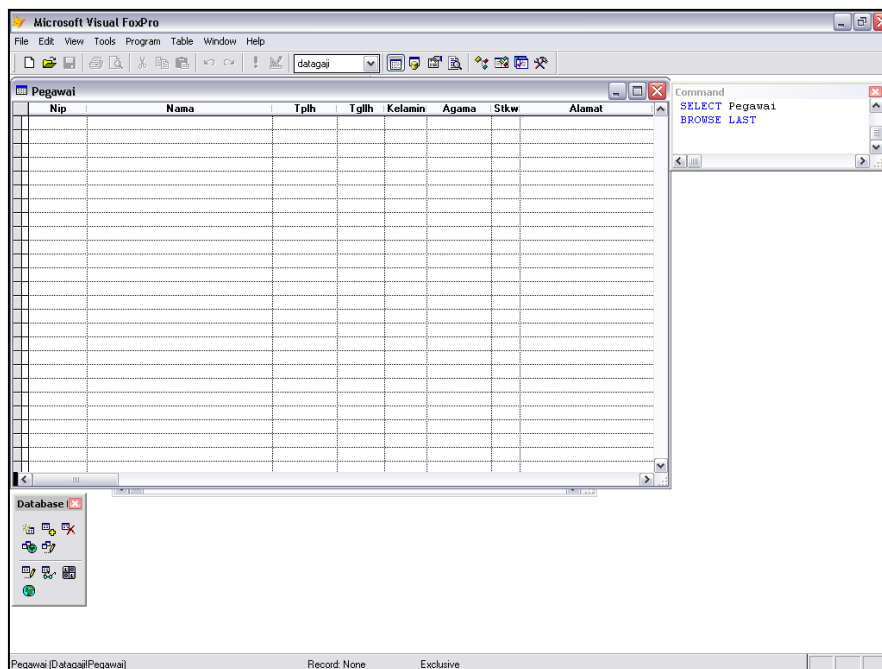
Pada pertemuan ini akan diperkenalkan bagaimana mengelola data dan memanfaatkan *database* dan tabel untuk membangun aplikasi. Selain itu juga diberikan juga beberapa latihan tambahan.

### 5.1 Mengelola Data

Mengelola data sangat penting dilakukan agar data-data yang dimiliki bisa dimasukkan dalam database, bisa diubah kembali, bisa ditampilkan dengan berbagai kondisi, dan bahkan data tersebut bisa dihapus jika tidak dibutuhkan lagi.

Untuk menampilkan data sekaligus dapat mengelola lebih lanjut dapat dilakukan dengan cara:

1. Klik kanan pada tabel yang sesuai
2. Pilih browse sehingga akan ditampilkan lembar kerja seperti Gambar 5.1



Gambar 5.1 Lembar Kerja Pengelolaan Tabel model Browse

3. Saat berada pada tampilan seperti gambar di atas, gunakanlah menu View dan menu Table untuk melakukan pengelolaan tabel. Misalkan mengklik menu View, lalu pilih pilihan Edit, maka tampilan berubah seperti Gambar 5.2.

Kdtamu	Nmtamu	Kel	Alamat	Telp	Kbangsa	Tglcatat
TM0001	Alex Wibowo	L	Jl. Janti No. 43 Yogya	081654545646	WNI	26-05-2015
TM0002	Hermawan	L	Jl. Sudirman No. 43 Yogya	081654545623	WNI	26-05-2015
TM0003	Linawati	P	Jl. Sudirman No. 43 Yogya	081654545677	WNI	26-05-2015
TM0004	Franco	L	Jl. Sudirman No. 43 Yogya	081654545687		

Gambar 5.2 Lembar Kerja Pengelolaan Tabel model Edit

a. Menambah data/record

- Tampilkan menu **Table**
- Pilih menu **Append New Record** (pilihan baris keempat)
- Isikan data yang sesuai untuk setiap field/kolom
- Ulangi langkah di atas untuk menambahkan data/record berikutnya.

b. Menghapus data/record

- Letakkan kursor diposisi mulainya data akan dihapus
- Tampilkan menu Table
- Pilih menu Delete Records, sehingga akan ditampilkan kotak dialog seperti Gambar 5.3

**Delete**

Scope: Next 1

For:

While:

Delete Cancel

Gambar 5.3 Kotak Dialog Proses Hapus Data

- Tentukan batasan dalam proses penghapusan data. Jika dalam posisi normal seperti gambar di atas langsung diklik tombol **Delete**, maka data yang akan dihapus hanya satu *record* dimana posisi kursor berada. Bila yang ingin dihapus adalah beberapa data mulai dari posisi kursor maka dapat mengganti angka 1 (satu) menjadi angka tertentu. Misalnya jika diganti angka 3 berarti yang akan dihapus adalah mulai dari posisi kursor sebanyak 3 record berikutnya. Apabila yang akan dihapus adalah semua data, maka gantilah pilihan pada tulisan **Next** menjadi **All**. Setelah selesai menentukan batasan penghapusan, maka klik tombol Delete untuk melakukan proses. Setelah dilakukan proses penghapusan, maka didepan data tersebut akan terlihat ada tanda hitam, yang menunjukkan data tersebut sudah dihapus, namun belum secara permanen.

c. Memanggil kembali data yang telah terhapus

Data yang pernah dihapus dengan perintah Delete Record, secara fisik belum hilang, dan masih memungkinkan untuk dipanggil kembali (dibatalkan proses hapusnya). Untuk melakukan hal tersebut lakukan langkah-langkah sebagai berikut:

- Letakkan kursor diposisi dimulainya proses pembatalan penghapusan
- Tampilkan menu Table
- Pilih menu Recall Records. Setelah memilih pilihan tersebut akan ditampilkan kotak dialog yang mirip seperti proses penghapusan. Hal-hal yang perlu dilakukan sama dengan proses penghapusan.

d. Menghilangkan data secara permanen

Data yang pernah dihapus dengan perintah Delete Records belum seutuhnya hilang. Untuk menghapus data secara permanen lakukan langkah sebagai berikut:

- Tampilkan menu Table
- Pilih menu Remove Delete Records, sehingga data yang pernah ditandai akan dihapus secara permanen.

e. Merubah tampilan data

Bentuk tampilan seperti Gambar 5.1 merupakan bentuk tampilan *browse*. Jika ingin bentuk tampilan yang lain, maka dapat dilakukan dengan langkah sebagai berikut:

- Tampilkan menu **View**
- Pilih menu **Edit**, sehingga bentuk tampilan data akan berubah. Jika ingin kembali ke bentuk *browse*, tampilkan menu Table, pilih menu **Browse**.

f. Kembali ke Database Designer (klik tanda silang pada tampilan data).

## 5.2 Praktek

Praktek pada pertemuan ini merupakan lanjutan dari sesi sebelumnya. Pada sesi ini, akan dicobakan proses pengelolaan data. Berikutnya akan dicoba untuk membuat pengolahan data dengan memanfaatkan *database* yang telah diciptakan.

Untuk itu bukalah **Project ProHotel** yang telah dibuat pada praktikum sebelumnya.

### 5.2.1 Mengelola Data

Setelah semua tabel selesai dibuat dan sudah dianggap memadai untuk sebuah aplikasi, maka data siap dikelola sesuai dengan kebutuhan. Pengelolaan data dapat dilakukan dengan menggunakan objek lain seperti form sebagai *interface*, atau dapat juga dilakukan secara langsung.

Pada kesempatan ini, yang dilakukan adalah mengelola data secara langsung, dengan harapan, pemakai dapat lebih paham dalam mengelola *database* bila terjadi hal-hal yang bersifat teknis.

Untuk memulai melakukan pengelolaan data, bisa dilakukan dengan dua cara yaitu:

1. Langsung dari posisi **Project Manajer**

Jika proses pengelolaan data dilakukan dari posisi project manajer, maka lakukan langkah-langkah sebagai berikut:

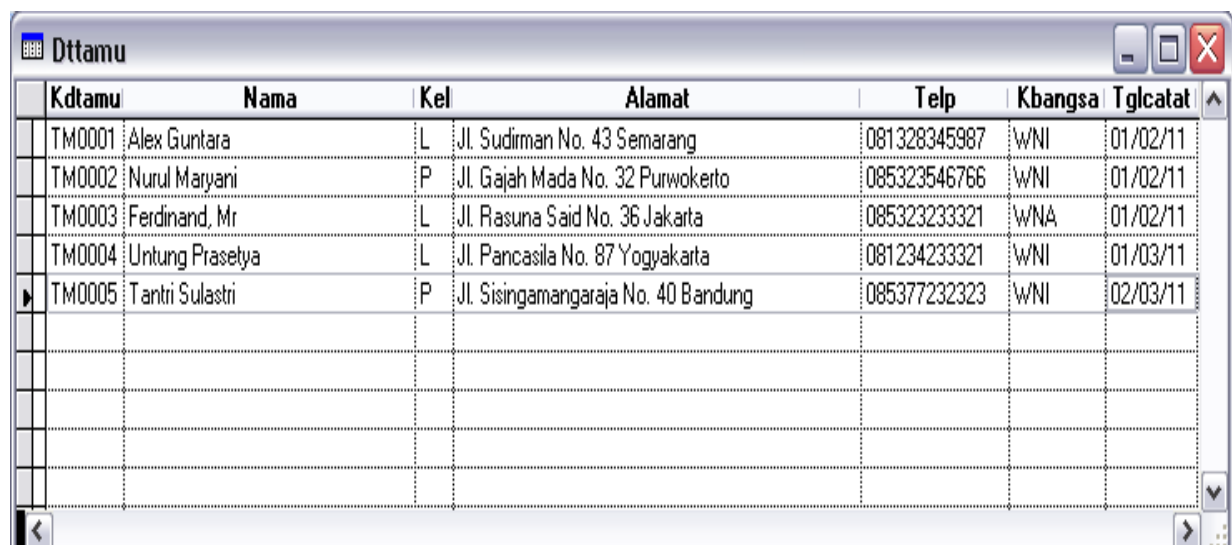
- Klik tab Data
- Tampilkan tabel-tabel yang akan dikelola datanya. Jika belum tampil, maka klik pada simbol tambah (+) pada Databases. Klik pada simbol tambah (+) pada nama database (datahotel). Klik pada sibol tambah (+) pada Tables.
- Klik tabel yang bersangkutan
- Klik menu Browse, sehingga akan ditampilkan layar untuk mengelola data
- Untuk mengelola data gunakanlah menu Table. Misalkan untuk setiap kali akan menambahkan data, tampilkan menu table, pilih Append New Record. Isikanlah data yang sesuai untuk masing-masing filed (kolom)

2. Dari posisi **Database Designer**

Jika proses pengelolaan data dilakukan dari posisi *database designer* (lihat Gambar 4.15), maka lakukan langkah-langkah sebagai berikut:

- Klik kanan pada tabel yang bersangkutan
- Pilih menu Browse, sehingga akan ditampilkan layar untuk mengelola data
- Untuk mengelola data gunakanlah menu Table. Misalkan untuk setiap kali akan menambahkan data, tampilkan menu table, pilih Append New Record. Isikanlah data yang sesuai untuk masing-masing filed (kolom)

Untuk mengisi data pada tabel tamu, lihatlah Gambar 5.4. Silahkan isi minimal 5 data.



Kdtamu	Nama	Kel	Alamat	Telp	Kbangsa	Tglcatat
TM0001	Alex Guntara	L	Jl. Sudirman No. 43 Semarang	081328345987	WNI	01/02/11
TM0002	Nurul Maryani	P	Jl. Gajah Mada No. 32 Purwokerto	085323546766	WNI	01/02/11
TM0003	Ferdinand, Mr	L	Jl. Rasuna Said No. 36 Jakarta	085323233321	WNA	01/02/11
TM0004	Untung Prasetya	L	Jl. Pancasila No. 87 Yogyakarta	081234233321	WNI	01/03/11
TM0005	Tantri Sulastri	P	Jl. Sisingamangaraja No. 40 Bandung	085377232323	WNI	02/03/11

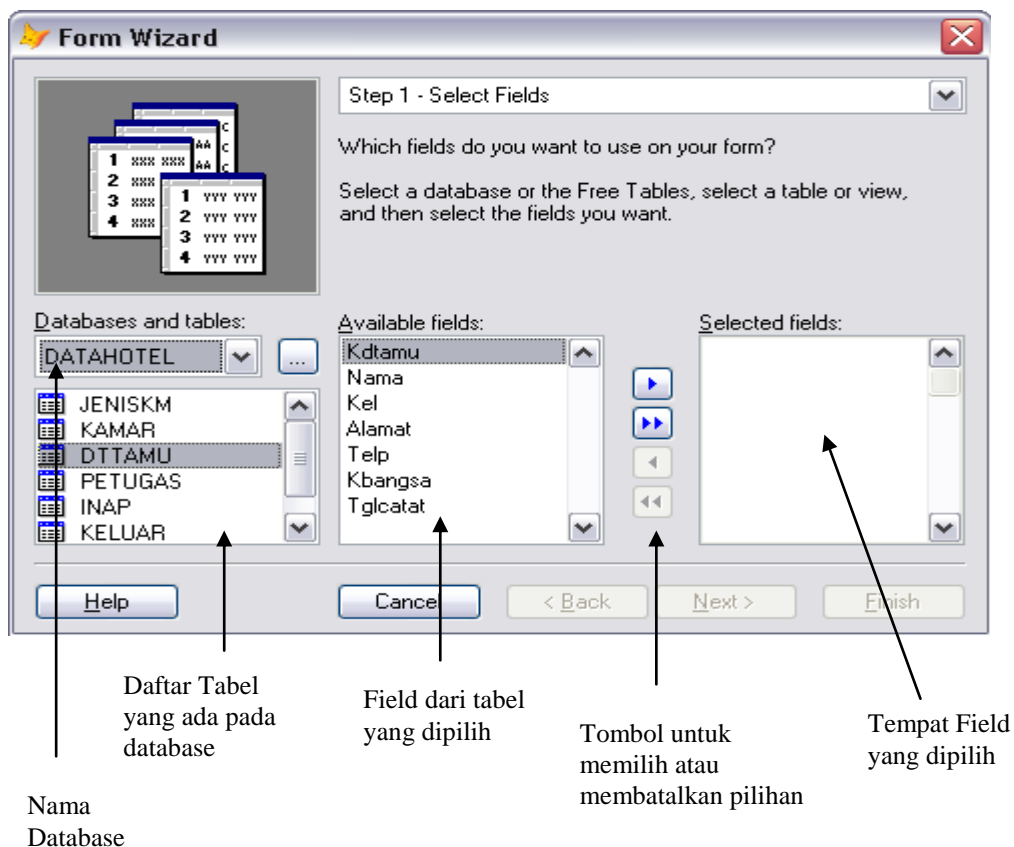
Gambar 5.4 Tampilan Pengelolaan Data Tamu

### 5.2.2 Membuat Form Wizard

Untuk melihat pengelolaan data melalui *interface (form)*, maka buatlah sebuah *form* yang dikaitkan dengan *database*. Untuk mempercepat proses pembuatannya, dilakukan dengan membuat form secara wizard.

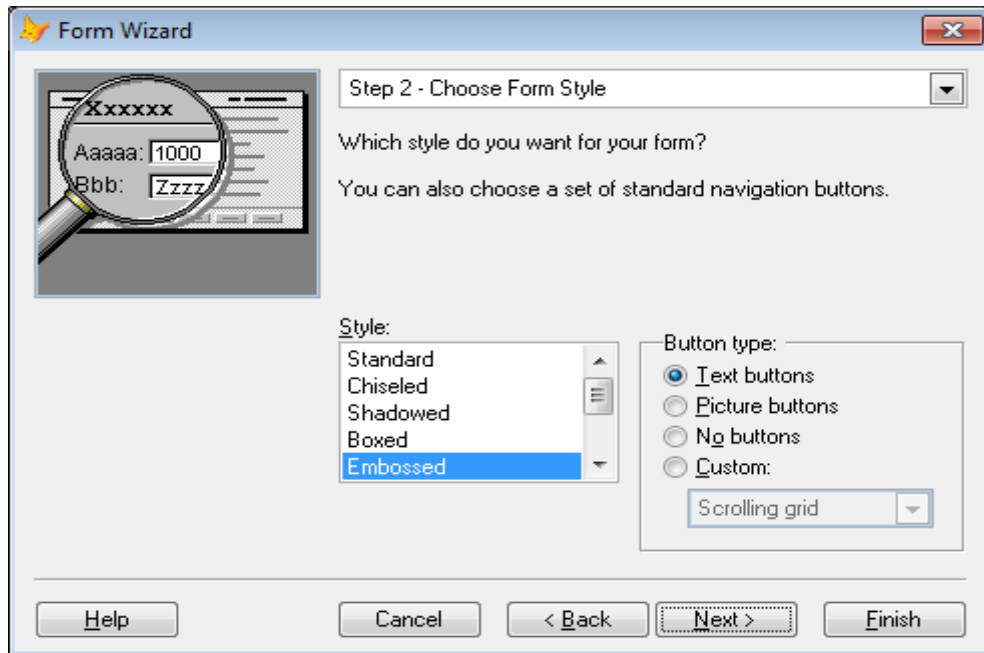
Pastikan posisi berada pada Project Manajer, lalu lakukan langkah-langkah sebagai berikut:

1. Klik Tab **Documents**
2. Klik pilihan **Forms**
3. Klik tombol **New**
4. Pilih/klik **Form Wizard**
5. Pilih Form Wizard dan lanjutkan dengan mengklik tombol **OK**, sehingga akan ditampilkan kotak dialog seperti Gambar 5.5



Gambar 5.5 Tampilan Memilih Field Untuk Form Wizard

6. Klik tabel yang akan dikaitkan dengan *form*
7. Pilih field-field yang akan dipakai (sebaiknya pilih semua)
8. Klik tombol Next



Gambar 5.6 Tampilan Memilih Tombol yang Digunakan

9. Tentukan jenis tombol perintah yang akan digunakan, selanjutnya klik tombol Next lagi
10. Tentukan pengurutan data (bisa tidak dilakukan), klik tombol Next
11. Berikan/ketik judul form, akhiri dengan mengklik tombol Finish.
12. Ketikan nama untuk menyimpan form yang dibuat, lanjutkan mengklik tombol Save, tunggu beberapa saat sampai selesai dan proses kembali ke project manajet
13. Saat berada di project manajer, carilah form yang telah dihasilkan dan klik kemudian klik tombol Run untuk melihat hasil, seperti terlihat pada Gambar 5.7.

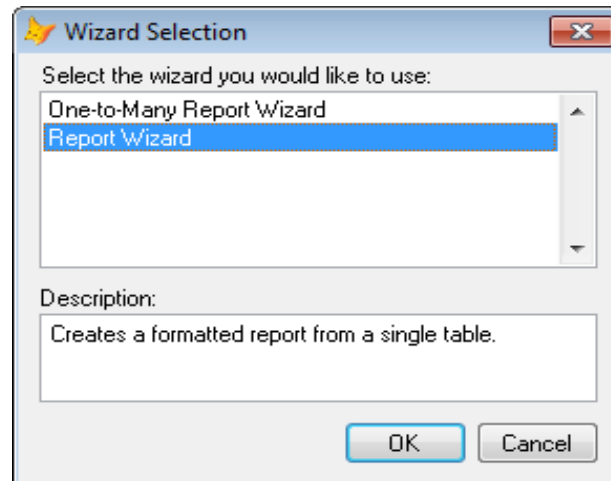
Gambar 5.7 Tampilan Hasil Form Wizard



### 5.2.3 Membuat Report Wizard

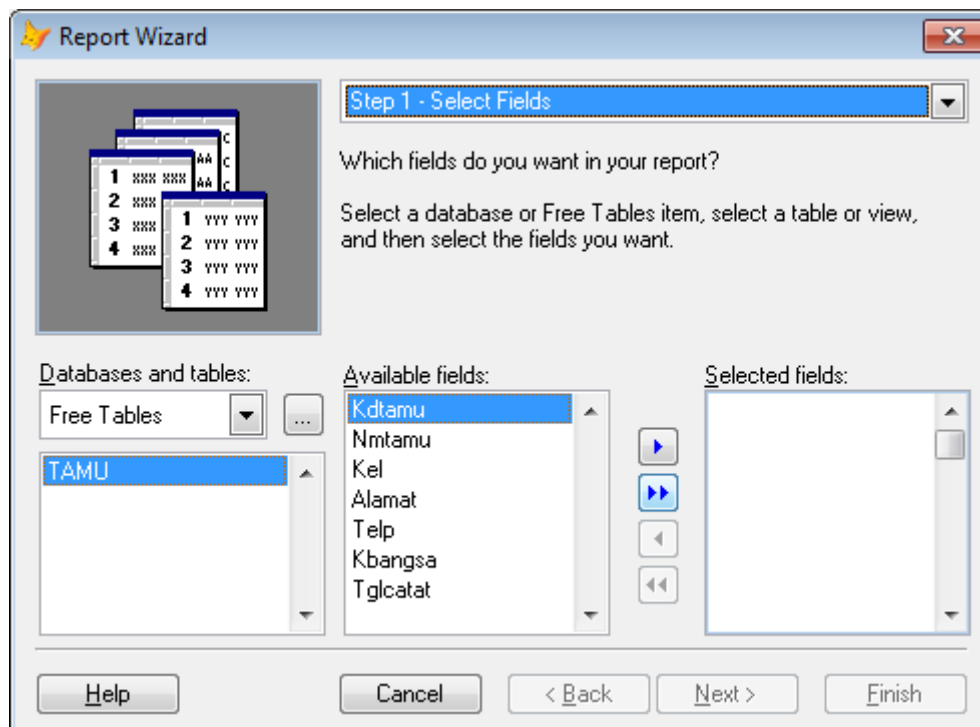
Proses pembuatan report secara wizard hampir sama dengan membuat form secara wizard. Untuk itu lakukanlah langkah sebagai berikut:

1. Klik Tab Documents
2. Klik pilihan Report
3. Klik tombol New
4. Klik pilihan Report Wizard
5. Pilih Report Wizard, dan lanjutkan mengklik tombol OK (lihat Gambar 5.8)



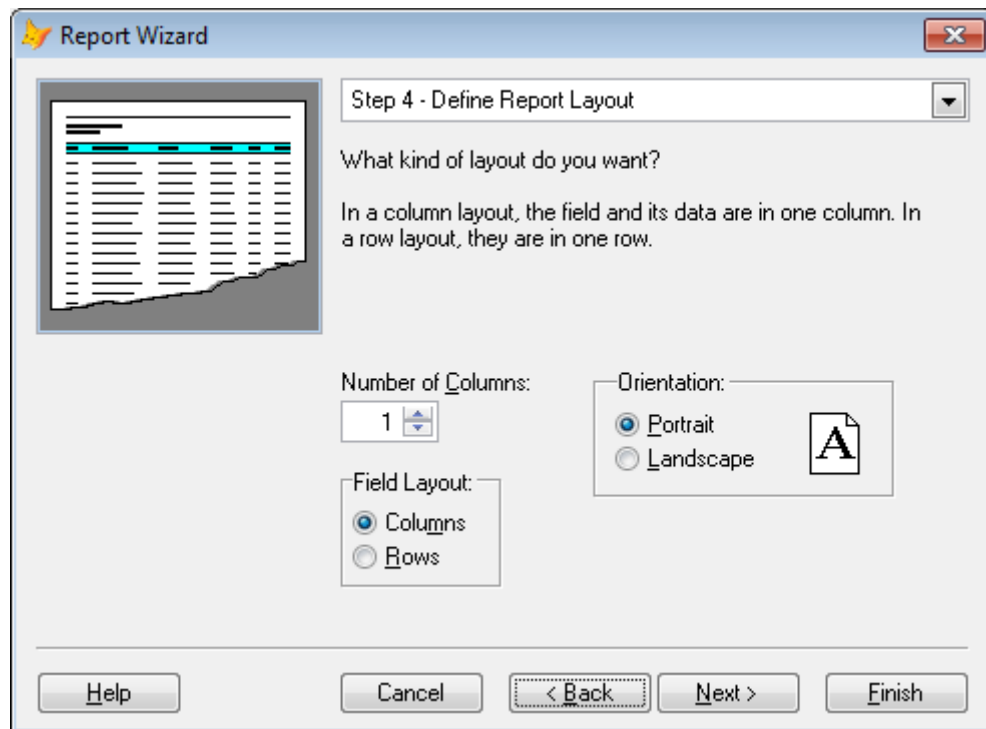
Gambar 5.8 Proses Pilihan Pembuatan Report Wizard

6. Klik tabel yang akan dikaitkan dengan Report
7. Pilih field-field yang akan dipakai untuk tampilan data pada report (untuk memilih semua field, klik tombol anak panah ganda). Lihat Gambar 5.9



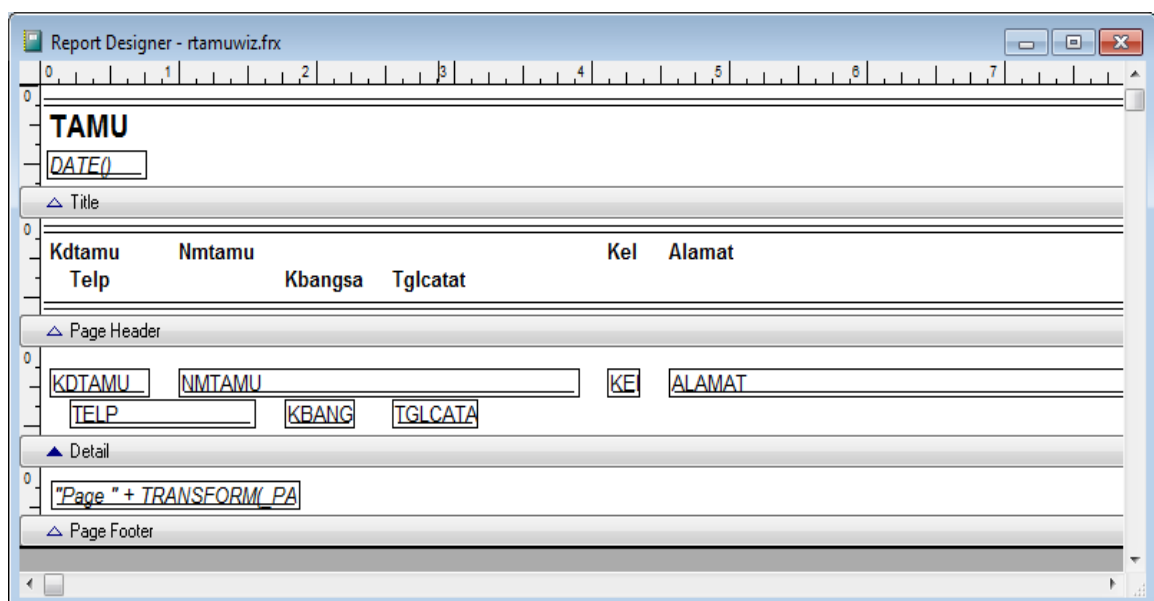
Gambar 5.9 Proses Pemilihan Field untuk Pembuatan Report Wizard

8. Klik tombol Next
9. Tentukan pengelompokan data/group (bisa tidak dilakukan) lalu klik tombol Next.
10. Tentukan jenis tampilan report (nilai default adalah Executive), lalu klik tombol Next
11. Tentukan orientasi laporan : Tegak (potrait) atau melebar (landscape). Jika data cukup banyak, pilih landscape. Lanjutkan mengklik tombol Next (lihat Gambar 5.10).



Gambar 5.10 Pilihan Orientasi Laporan

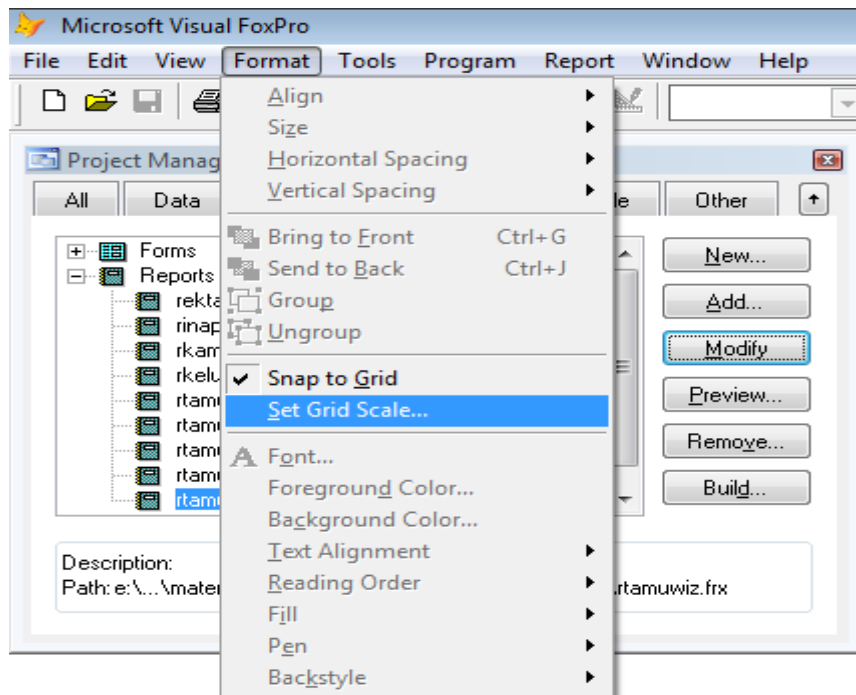
12. Tentukan pengurutan data (bisa tidak dilakukan), klik tombol Next
13. Berikan/ketik judul report, akhiri dengan mengklik tombol Finish.
14. Ketikkan nama untuk menyimpan report yang dibuat, lanjutkan mengklik tombol Save, tunggu beberapa saat sampai report selesai dan kembali ke project manager.
15. Saat berada di project manager, carilah report yang telah dihasilkan dan klik kemudian klik tombol Modify untuk melakukan perbaikan dan layout, seperti terlihat pada Gambar 5.11



Gambar 5.11 Modifikasi Report

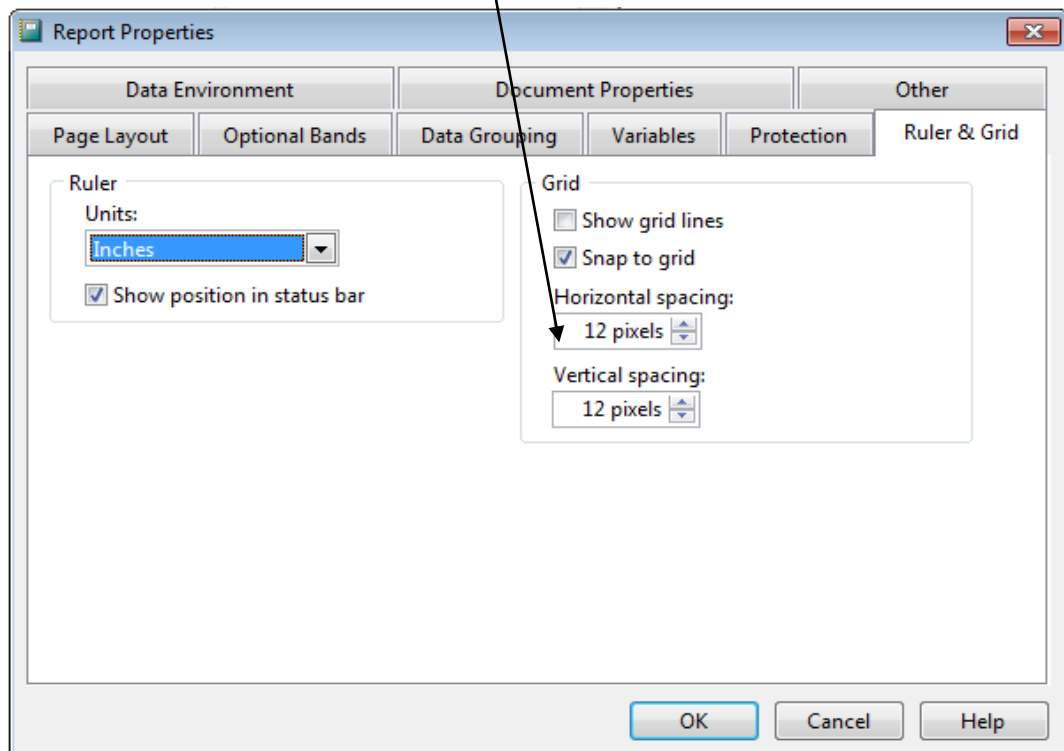
Sebelum melakukan proses modifikasi terhadap laporan (report), atur terlebih dahulu skala grid agar pengaturan laporan menjadi lebih leluasa. Langkah-langkahnya adalah sebagai berikut:

- Tampilkan menu Format (lihat Gambar 5.12)



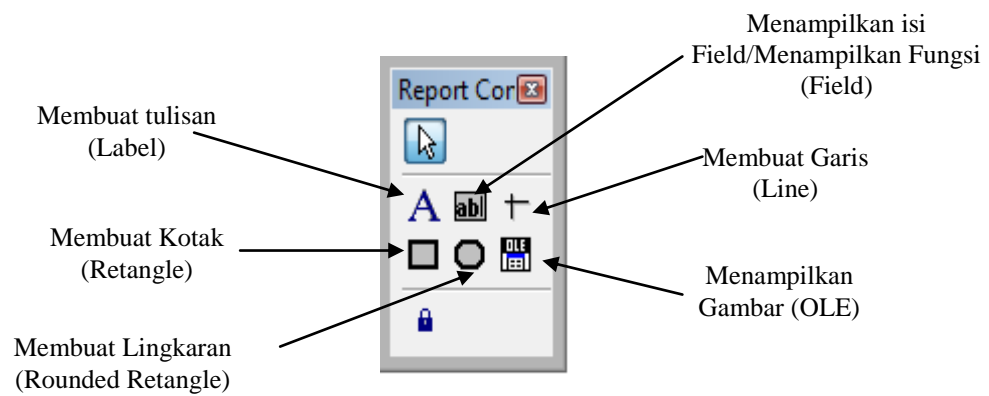
Gambar 5.12 Pilihan Menu Pengaturan Skala

- Pilih menu Set Grid Scale
- Tentukan skala secara horizontal maupun secara vertical (sebaiknya diisi nilai 1 pixels), setelah selesai akhiri dengan mengklik tombol Ok.



Gambar 5.13 Cara Menentukan Skala

Selanjutnya proses modifikasi laporan bisa dilanjutkan dengan memanfaatkan **Report Control** seperti terlihat pada Gambar 5.14.



Gambar 5.14 Report Control

# Bab 6

## Merancang Form dan Mengaitkan Dengan *Database*

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami dan mampu melakukan perancangan form secara manual (desain sendiri) sebagai *interface* dalam pengolahan data dalam pemrograman Visual FoxPro.

Pada pertemuan ini akan dibahas mengenai proses merancangan *form* dengan rancangan sendiri, dan menghubungkan atau mengaitkan *form* dengan *database* yang telah disiapkan. Selain itu juga diberikan juga beberapa latihan tambahan.

### 6.1 Merancang Form

Merancang form bisa menjadi pekerjaan yang mengasyikkan, tapi bisa juga menjadi pekerjaan yang membosankan. Merancangan form membutuhkan keilmuan terkait interaksi manusia dan komputer. Fokus perancangan pada tata letak, pewarnaan dan penggunaan komponen yang tepat. Perancang form yang memiliki nilai seni tentu dapat menghasilkan interface yang menarik dan interaktif. Perancang form yang merasa tidak memiliki nilai seni jangan berkecil hati, karena prinsip yang tidak kalah penting adalah sering berlatih.

Dasar pertimbangan untuk merancang interface (rancangan masukan) adalah adanya pembagian area. Area pertama adalah untuk judul. Area yang kedua adalah untuk tubuhnya suatu interface. Area yang ketiga adalah untuk penempatan instruksi atau keterangan lainnya. Pengaturan area ini disesuaikan dengan selera, dengan tetap mengacu pada estetika yang ada.

### 6.2 Mengaitkan Form Dengan *Database*

Pekerjaan ini membutuhkan kesabaran dan ketelitian. Pemrograman basis data akan berarti apabila program yang dibuat bisa melakukan pengolahan data, dimana datanya bisa tersimpan dengan baik di database. Setelah database dibuat, interface juga sudah dirancang dengan baik, maka tiba saat untuk melakukan proses menghubungkan atau mengaitkan interface (form) dengan database. Dengan terkaitnya form dengan database maka pengolahan data akan leluasa dilakukan.

Untuk mengaitkan form dengan database, bisa hanya memasukkan databasenya saja ke data environment, atau bisa juga dengan menggunakan komponen khusus (misalnya Grid). Proses mengaitkan form dengan database dengan menggunakan Grid akan menjadi lebih interaktif. Apabila tabel yang dipakai hanya satu maka mengaitkan form dengan menggunakan Grid cukup mudah, namun bila tabel yang digunakan lebih dari satu, tentu memerlukan ketelitian agar data dari semua tabel bisa ditampilkan.

### 6.3 Praktek

Praktek pada pertemuan ini merupakan lanjutan dari sesi sebelumnya. Pada sesi ini, akan dicobakan proses pengelolaan data. Berikutnya akan dicoba untuk membuat rancangan form dan proses mengaitkan form dengan *database*.

Untuk itu bukalah **Project ProHotel** yang telah dibuat pada praktikum sebelumnya.

#### 6.3.1 Membuat Rancangan Form Pengolahan Data Tamu

Pastikan saat ini berada di posisi Project Manajer, kemudian lakukan langkah-langkah sebagai berikut:

1. Klik Tab Page **Documents**
2. Klik pilihan **Form**
3. Klik tombol **New**
4. Pilihan **New Form**, sehingga akan ditampilkan Desain Form.
5. Sebelum mengambil komponen-komponen yang dibutuhkan, atur properties Form sebagai berikut:
  - Caption : ditulis Sistem Administrasi Perhotelan
  - AutoCenter : diganti **True**
  - MDIForm : diganti **True**
  - MaxButton : diganti **False**
  - MinButton : diganti **False**
  - ShowTips : diganti **True**
  - ShowWindow : diganti dengan **2 As Top Level Form**
6. Ambil komponen-komponen dan atur seperti terlihat pada Gambar 6.1.

*Gambar 6.1 Rancangan Form Pengolahan Data Tamu*

7. Komponen yang digunakan berturut-turut adalah :
  - a. Image yang diletakkan di kiri atas Form, lalu atur properties
    - Stretch : Stretch
    - Picture : cari gambar yang sesuai (sebaiknya sudah berada pada directori yang sama dengan aplikasi)
  - b. Label1, lalu atur properties
    - Caption : HOTEL PURNAMA
    - FontBold : .T. (True)
    - FontSize : 24
  - c. Label2, lalu atur properties
    - Caption : JL. SISINGAMANGARAJA NO 43 YOGYAKARTA, TELP (0274) 563323
    - FontBold : .T. (True)
    - FontSize : 12
  - d. Label3, lalu atur properties

Caption : PENGOLAHAN DATA TAMU  
FontBold : .T. (True)  
FontSize : 20

- e. Line1 (garis miring sebelah kiri), lalu atur properties  
BorderWidth : 4
- f. Line2 (garis mendatar), lalu atur properties  
BorderWidth : 4
- g. Label4, lalu atur properties  
Caption : Kode Tamu  
FontBold : .T. (True)  
FontSize : 10
- h. Label5, lalu atur properties  
Caption : Nama Tamu  
FontBold : .T. (True)  
FontSize : 10
- i. Label6, lalu atur properties  
Caption : Jenis Kelamin  
FontBold : .T. (True)  
FontSize : 10
- j. Label7, lalu atur properties  
Caption : Alamat  
FontBold : .T. (True)  
FontSize : 10
- k. Label8, lalu atur properties  
Caption : No. Telp  
FontBold : .T. (True)  
FontSize : 10
- l. Label9, lalu atur properties  
Caption : Kewargaan  
FontBold : .T. (True)  
FontSize : 10
- m. Label10, lalu atur properties  
Caption : Tanggal Catat  
FontBold : .T. (True)  
FontSize : 10
- n. Text1, letakkan disebelah kanan Kode Tamu, lalu atur properties  
MaxLength : 6 (sesuaikan dengan lebar pada tabel)
- o. Text2, letakkan disebelah kanan Nama Tamu, lalu atur properties  
MaxLength : 30 (sesuaikan dengan lebar pada tabel)
- p. Optiongroup1, letakkan disebelah kanan Jenis Kelamin. Klik kanan, lalu pilih Builder, isikan pilihan pada kolom Caption yaitu (Laki-laki dan Perempuan). Tentukan Layout menjadi Horizontal. Klik tombol OK. Klik kanan, lalu pilih Edit jika diperlukan.
- q. Text3, letakkan disebelah kanan Alamat, lalu atur properties  
MaxLength : 50 (sesuaikan dengan lebar pada tabel)
- r. Text4, letakkan disebelah kanan No. Telp, lalu atur properties  
MaxLength : 12 (sesuaikan dengan lebar pada tabel)
- s. Optiongroup2, letakkan disebelah kanan Kewargaan. Klik kanan, lalu pilih Builder, isikan pilihan pada kolom Caption yaitu (WNI dan WNA). Tentukan Layout tetap Vertical. Klik tombol OK. Klik kanan, lalu pilih Edit jika diperlukan.
- t. Text5, letakkan disebelah kanan Tanggal Catat, lalu atur properties
- u. Grid1, dan atur ukuran sesuai dengan yang diinginkan. Sesuaikan seperti gambar.  
Lalu atur properti  
Readonly : .T. (True)  
Tabstop : .F. (False)
- v. Command Button (tombol) sebanyak 6 buah yaitu Command Button1 sampai Command Button6. Atur secara berturut-turut properties
  - Untuk Command Button1

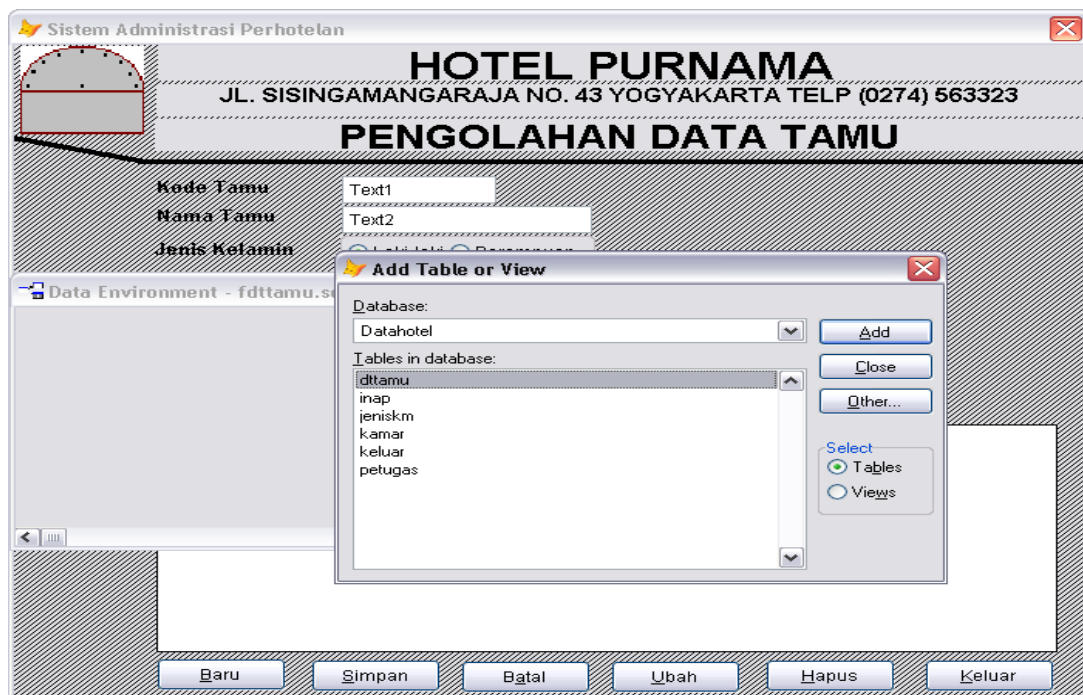
- Caption : \<Baru
- Name : tblbaru
- Untuk Command Button2
- Caption : \<Simpan
- Name : tblsimpan
- Untuk Command Button3
- Caption : B\<atal
- Name : tblbatal
- Untuk Command Button4
- Caption : \<Ubah
- Name : tblubah
- Untuk Command Button5
- Caption : \<Hapus
- Name : tblhapus
- Untuk Command Button6
- Caption : \<Keluar
- Name : tblkeluar

8. Setelah selesai rancangan form seperti gambar di atas, simpanlah form tersebut dengan nama **FdtTamu** atau **FTamu**.

### 6.3.2 Mengaitkan Form dengan Database dan Tabel

Setelah selesai rancangan Form seperti gambar di atas, maka jika Form akan dimanfaatkan untuk pengolahan data yang menggunakan tabel maka dapat dikaitkan dengan cara memasukkan tabel yang dibutuhkan pada form dengan langkah sebagai berikut:

- a. Klik kanan pada daerah yang kosong (yang tidak ada objek) pada form atau dipinggir form
- b. Pilih Data Environment, sehingga akan ditampilkan kotak dialog seperti terlihat pada Gambar 6.2.



Gambar 6. Proses Mengaitkan Form Dengan Database

- c. Klik nama tabel pada kolom **Tables in Database**
- d. Klik tombol Add.

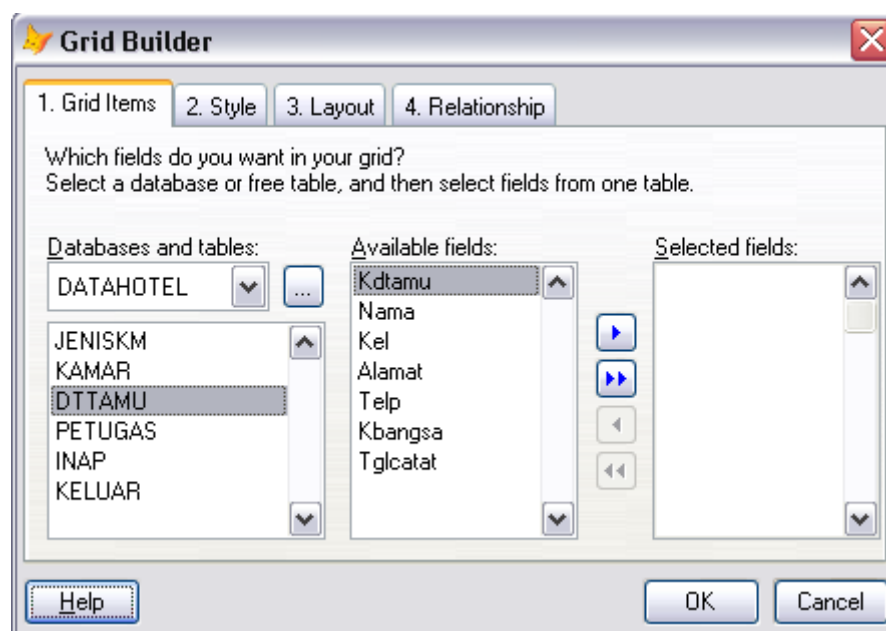


- e. Lakukan langkah c dan d untuk menambahkan tabel yang lain. Jika sudah selesai klik tombol **Close**. Dengan demikian tabel yang sudah ditambahkan sudah bisa diakses pada form. Jika tabel yang digunakan lebih dari satu, maka jika tabel tersebut saling terkait, lakukan proses membuat relasinya.

### 6.3.3 Mengaitkan Grid dengan Tabel

Untuk mengaitkan grid dengan tabel yang dipakai dalam pengolahan data dapat dilakukan seperti proses pembuatan combobox. Untuk itu lakukan langkah-langkah sebagai berikut:

1. Klik kanan pada bagian grid
2. Pilih **Builder**
3. Tentukan *database* dan atau tabel yang akan dipakai dengan memperhatikan pada kolom **Database and Tables**. Jika tabel-tabel belum muncul, klik tombol titik tiga untuk membaca tabel. Jika sudah tampil, maka cukup memilih dengan mengklik nama tabelnya (lihat Gambar 6.3)



Gambar 6.3 Penggunaan Tabel untuk Grid

4. Pilih/seleksi field yang dibutuhkan (bisa dipilih semua sekaligus) dengan menggunakan tombol panah kanan satu (memilih field tertentu) atau tombol panah kanan dua (memilih semua sekaligus)
5. Klik tab **Style** jika menginginkan model tampilan grid (silahkan pilihan salah satu)
6. Klik tab **Layout** jika menginginkan pengaturan lebar kolom.
7. Klik tombol **OK** jika sudah selesai menentukan aturan yang diperlukan.

# Bab 7

## Tipe Data, Variabel, Operator, Ekspresi dan Algoritma Pengolahan Data

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami tentang tipe data, variabel dan operator yang digunakan dalam pemrograman Visual FoxPro.

Pada pertemuan ini akan dibahas mengenai tipe data, pendeklarasian variabel, inputbox dan messagebox, operator matematika, operator perbandingan, operator penggabungan, operator logika. Selain itu juga diberikan juga beberapa latihan tambahan.

### 7.1 Tipe Data

Tipe data digunakan untuk mengatur nilai dari sebuah variabel yang Anda gunakan untuk membuat sebuah aplikasi. Perhatikan Tabel 7.1, mengenai beberapa tipe data yang dapat digunakan.

Tabel 7.1 : Tipe Data

Tipe Data	Keterangan	Nilai
Blob	Bilangan binari dengan ukuran tak tentu	Terbatas oleh memori dan file dengan limit 2 GB
Character	Berupa huruf dan angka serta simbol-simbol	Semua karakter
Currency	Bilangan mata uang	-922337203685477.5807 to 922337203685477.5807
Date	Data mengenai bulan, hari dan tahun	Januari 1st, 1 A.D sampai (^9999-12-31). Desember 31st, 9999 A.D
DateTime	Data mengenai bulan, hari, tahun, jam, menit, dan detik	(Date) Januari 1st, 9999-12-31 sampai Desember 31st, 9999. (Time) 00:00:00 a.m to 11:59:59 p.m
Logical	Hanya mempunyai dua buah nilai yaitu True dan False	True (.T.) or False (.F.)
Numeric	Bilangan Integer atau desimal dari angka	-.9999999999E+19 sampai -.9999999999E+20
Varbinary	Varbinary sama seperti bilangan Varchar, tidak dapat digunakan dengan nilai 0	Bilangan hexadecimal
Variant	Bilangan Variant dapat digunakan untuk seluruh tipe data dan nilai 0	Lihat tipe data lainnya

## 7.2 Pendeklarasian Variabel

Variabel digunakan untuk menyimpan sebuah informasi. Anda dapat membuat variabel, dengan membuat nama pada variabel tersebut dan memberinya sebuah nilai. Perhatikan pendeklarasian variabel di bawah ini.

**A = 1**

Nama variabel tersebut adalah A yang memiliki kandungan nilai 1. Jika variabel yang dibuat mempunyai nilai string (huruf), maka pembuatan nilai tersebut akan ditambah dengan tanda kutip (") pada awal dan akhir karakter. Perhatikan pendeklarasian variabel di bawah ini.

**Nama = "Hadi"**

Agar variabel dapat berfungsi dengan baik, terdapat persyaratan dalam penamaan variabel:

- Karakter dimulai dengan huruf
- Karakter yang berhubungan dengan matematika atau fungsi lainnya (\*, &, ^, ...) tidak dapat digunakan.
- Penggunaan spasi diganti dengan underscore (\_)
- Panjang karakter dari variabel tidak boleh lebih dari 255 karakter.

Visual FoxPro mempunyai dua buah jenis pendeklarasian variabel yaitu:

### 1. Pendeklarasian Global

Dalam mendeklarasikan variabel bertipe global, Anda harus menuliskan perintah **Public** kemudian diikuti nama dari variabel. Contoh:

**Public (namavariabel)**

Misalkan kita akan membuat variabel dengan nama **mkode** dengan tipe **public**, maka harus kita deklarasikan sebagai berikut:

**Public mkode**

Jika ada beberapa variabel yang ingin dibuat, penulisannya dipisah dengan menggunakan tanda koma. Contoh:

**Public mnama, mgol, mgaji, mtunj**

Dengan menggunakan jenis pendeklarasian Global, program yang Anda buat dapat menggunakan variabel yang sama dalam sebuah form (seluruh komponen di dalam form). Dengan kata lain, variabel yang dideklarasikan dengan tipe Global akan dikenal di semua bagian aplikasi.

### 2. Pendeklarasian Lokal

Dalam mendeklarasikan variabel bertipe Lokal, Anda harus menuliskan perintah **Local** kemudian diikuti dengan nama dari variabel. Contoh:

**Local (namavariabel)**

Misalkan kita akan membuat variabel dengan nama **mcari** dengan tipe **local**, maka harus kita deklarasikan sebagai berikut:

**Local mcari**

Jika ada beberapa variabel yang ingin dibuat, penulisannya dipisah dengan menggunakan tanda koma. Contoh:

**Local mnama, mgol, mgaji, mtunj**

Pendeklarasian lokal digunakan untuk mendeklarasikan suatu variabel di dalam sebuah prosedur atau komponen, maka variabel tersebut hanya dapat digunakan di dalam prosedur komponen itu saja.

### 7.3 InputBox dan MessageBox

Visual FoxPro dapat berinteraksi dengan pengguna untuk menampilkan suatu informasi dan peran berupa sebuah kotak dialog. Pesan tersebut digunakan untuk mengingatkan kepada pengguna. Visual FoxPro mempunyai dua buah kotak dialog yang digunakan pengguna yaitu InputBox dan MessageBox.

#### 1. InputBox

InputBox berfungsi menampilkan sebuah kotak dialog yang berisi kotak input yang dapat diisi oleh pengguna.

Bentuk Umum Perintah:

**Var** = InputBox(<**Teks**>,<**JudulWindow**>)

**Var** adalah variabel yang akan menampung masukan

**Teks** adalah tulisan yang akan ditampilkan pada kotak

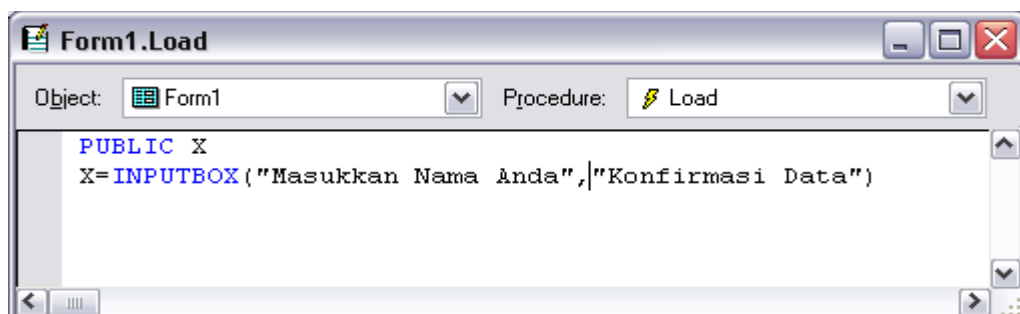
**JudulWindow** adalah judul untuk kotak yang akan ditampilkan

Contoh:

**X=InputBox("Masukkan Nama Anda ","Konfirmasi Data")**

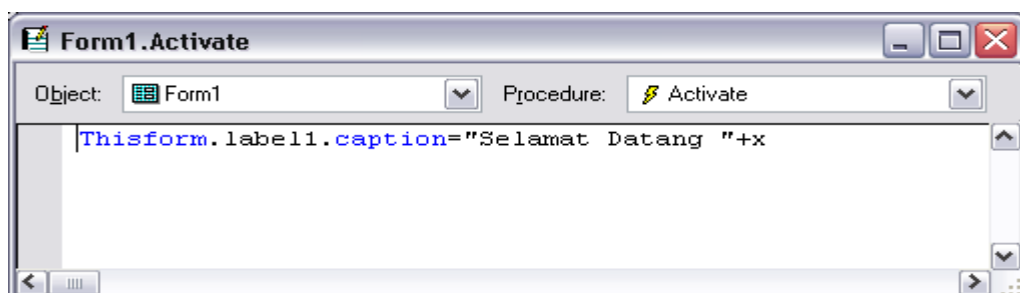
Implementasi:

- Buatlah sebuah form baru
- Tambahlah objek label pada form tersebut
- Tulis kode program pada Form Load



Gambar 7.1 Contoh Kode Program Untuk InputBox di Form Load

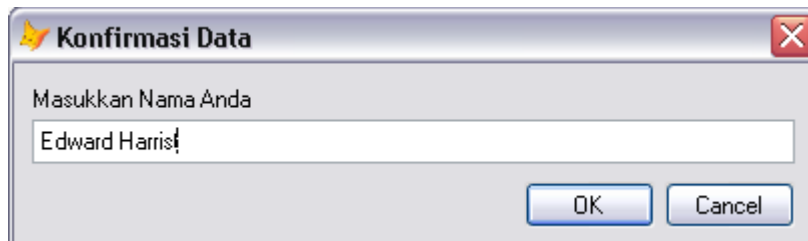
- Tulis kode program pada Form Active



Gambar 7.2 Contoh Kode Program Untuk InputBox di Form Active

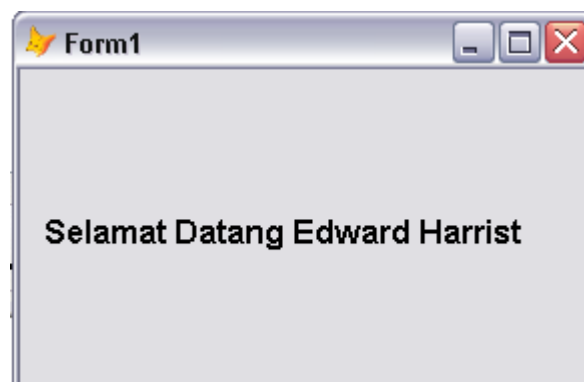
Hasil:

Saat Form dijalankan, maka akan ditampilkan kota dialog seperti Gambar 7.3.



*Gambar 7.3 Proses Pengisian Inputboxx*

Setelah diklik tombol OK akan ditampilkan form seperti Gambar 7.4.



*Gambar 7.4 Hasil Setelah Inputbox Diisi*

## 2. MessageBox

MessageBox merupakan sebuah pesan yang berfungsi untuk mengingatkan kepada pengguna akan sesuatu dengan menampilkan sebuah kotak pesan yang berisi peringatan.

Bentuk Umum Perintah:

**Var** = MessageBox(<**Teks**>,<**Tombol**>+<**Icon**>,<**JudulWindow**>)

**Var** adalah variabel yang akan menampung nilai pesan

**Teks** adalah tulisan yang akan ditampilkan pada kotak pesan

**Tombol** adalah nilai angka dari tombol yang akan digunakan (lihat tabel 7.2)

**Icon** adalah nilai icon yang digunakan (lihat tabel 7.3)

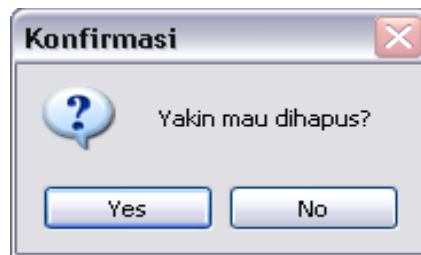
**JudulWindow** adalah judul untuk kotak yang akan ditampilkan

Contoh:

**K=MessageBox("Yakin data akan dihapus?",4+32,"Konfirmasi")**

Keterangan:

Hasil eksekusi dari contoh di atas adalah sebuah kotak pesan yang menampilkan tulisan pertanyaan "Yakin data akan dihapus?", lalu tombol yang ditampilkan adalah tombol Yes dan No (angka 4), icon yang digunakan adalah icon tanda tanya (angka 32), dan judul kotak pesan adalah "Konfirmasi")



*Gambar 7.5 Tampilan Hasil MessageBox*

Tabel 7.2 Angka Tombol

Nilai Angka	Tombol Yang Tampil
0	OK
1	OK dan Cancel
2	Abort, Retry, dan Ignore
3	Yes, No, dan Cancel
4	Yes dan No
5	Retry dan Cancel

Tabel 7.3 Angka Icon

Nilai Angka	Icon Yang Tampil
16	Icon tanda Silang 
32	Icon tanda tanya 
48	Icon tanda Seru 
64	Icon tanda informasi 

## 7.4 Operator Matematika

Operator matematika merupakan jenis operator yang digunakan sebagai operasi matematika. Operator seperti ini biasanya digunakan untuk mengubah suatu bilangan melalui sebuah proses.

Operator matematika memiliki banyak tipe, perhatikan Tabel 7.4

Tabel 7.4 Jenis Operator Matematika

OPerator	Keterangan
^	Operator pemangkatan
*	Operator perkalian
/	Operator pembagian
\	Operator pembagian bilangan bulat
Mod	Operator pembagian (siswa pembagian)
+	Operator penjumlahan
-	Operator pengurangan

## 7.5 Operator Perbandingan

Operator Pembanding merupakan jenis operator yang digunakan untuk membandingkan suatu nilai dengan nilai yang lain. Operator Pembanding mempunyai dua nilai yaitu True (.T.) dan False (.F.). Untuk lebih jelas tentang operator ini lihat Tabel 7.5

Tabel 7.5 Jenis Operator Pembanding

OPerator	Keterangan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil atau sama dengan
>=	Lebih besar atau sama dengan
=	Sama dengan
<>	Tidak sama dengan

## 7.6 Operator Penggabungan

Operator Penggabungan berfungsi untuk menggabungkan dua buah variabel string. Visual FoxPro mempunyai dua buah fungsi operator penggabungan yaitu +.

Sebagai contoh:

Local A, B

A=Alltrim(thisform.text1.value)

B = Thisform.text2.value)

Thisform.text3.value=A + " "+B

Keterangan:

Pendeklarasian variabel A dan B secara Lokal

Alltrim digunakan untuk membaca banyaknya karakter yang Anda inputkan

Penggabungan nilai A dan B yang hasilnya dialihkan ke Text1.value

## 7.7 Operator Logika

Operator Logika merupakan fungsi yang digunakan sebagai pembandingan antara suatu nilai dan dapat mengeluarkan nilai true atau false dari hasil perbandingan tersebut. Untuk jenisnya dapat dilihat pada tabel 5.6.

Tabel 5.6 Jenis Operator Logika

Operator	Keterangan	Penjelasan
And	Dan	Hasil keseluruhan akan benar jika kedua nilai benar
Or	Atau	Hasil keseluruhan akan benar jika salah satu nilai benar
Not	Tidak	Nilai ingkaran dari suatu nilai

## 7.8 Ekspresi

Ekspresi (ungkapan) adalah merupakan proses untuk membandingkan suatu nilai tertentu dengan menggunakan operator tertentu yang akan menghasilkan kondisi benar atau salah. Data yang dibandingkan dengan data pembanding harus memiliki jenis data yang sama.

Bentuk Umum:

N1 OP N1

N1 = nilai yang dibandingkan

OP = operator yang digunakan

N1 = nilai pembanding

Contoh:

A = 30

B = 32

Ekspresi: **A > B**

Ekspresi: **A > 50 .and. B<30**

## 7.9 Aktif/Non Aktif dan Tampil/Tidak Tampil Objek

1. Aktif atau Non Aktif Objek



Aktif (hidup) maksudnya adalah bahwa kontrol tersebut bisa digunakan (diisi, diklik, dan sebagainya). Sebaliknya Non Aktif (mati) maksudnya adalah bahwa kontrol tersebut tidak bisa digunakan.

Dalam membuat aplikasi tentu saja terkadang membutuhkan banyak kontrol, sesuai dengan program yang dibuat. Pada saat tertentu terkadang kontrol tersebut perlu dimatikan (non aktif) atau dihidupkan (aktif). Pengaturan ini dapat dilakukan melalui properties. Namun kondisi yang dialami oleh objek tersebut akan bersifat tetap. Untuk lebih fleksibel, maka proses pengaturan dapat pula dilakukan dengan penulisan kode, sehingga dapat dikendalikan dengan kondisi yang berbeda setiap saat.

Bentuk Umum Perintah :

`Thisform>NamaObjek.Enabled=<kondisi>`

NamaObjek : adalah nama objek/kontrol yang akan diberikan pengaturan  
Enabled : adalah perintah untuk aktif/tidak aktif  
<kondisi> : adalah kondisi yang diberikan (**True (.T.)** untuk aktif)  
atau (**False (.F.)** untuk tidak aktif)

Contoh :

Misalkan pada form ada objek/kontrol TEXT1

Untuk mematikan kontrol tersebut maka kode yang harus ditulis adalah :

**`Thisform.TEXT1.Enabled=.f.`**

Untuk menghidupkan kontrol tersebut maka kode yang harus ditulis adalah :

**`Thisform.TEXT1.Enabled=.t.`**

## 2. Tampil atau Tidak Tampil Objek

Tampil (muncul) maksudnya adalah bahwa kontrol tersebut tampak pada interface. Sebaliknya Tidak Tampil (hilang) maksudnya adalah bahwa kontrol tersebut tidak tampak pada interface.

Untuk keperluan tertentu kondisi-kondisi ini sangat membantu untuk digunakan supaya program lebih interaktif.

Bentuk Umum Perintah :

`NamaObjek.Visible=<kondisi>`

NamaObjek : adalah nama objek/kontrol yang akan diberikan pengaturan  
Visible : adalah perintah untuk tampil/tidak tampil  
<kondisi> : adalah kondisi yang diberikan (**True (.T.)** untuk aktif)  
atau (**False (.F.)** untuk tidak aktif)

Contoh :

Misalkan pada form ada objek/kontrol TEXT1

Untuk menghilangkan kontrol tersebut maka kode yang harus ditulis adalah :

**`Thisform.TEXT1.Visible=.f.`**

Untuk menampilkan kontrol tersebut maka kode yang harus ditulis adalah :

**`Thisform.TEXT1.Visible=.t.`**

## 7.10 Algoritma Pengolahan Data

Pengolahan data merupakan hal yang pasti dilakukan dalam setiap aplikasi. Pengolahan data terdiri dari menambah, menyimpan, membatalkan, merubah, menghapus dan

sebagainya yang menggunakan tombol-tombol sebagai kontrolnya. Untuk memudahkan pemahaman dalam melakukan pengolahan data, berikut ini akan disajikan algoritma untuk proses-proses yang digunakan.

a. Algoritma untuk tombol **Tambah**

- Bersihkan isian data atau berikan nilai awal
- Aktifkan isian data
- Matikan semua tombol kecuali tombol Simpan dan tombol Batal
- Tempatkan kursor pada isian yang pertama

b. Algoritma untuk tombol **Simpan**

- Lakukan pencarian untuk data yang akan disimpan (kalau sudah bekerja dengan data)
- Jika data sudah ada, tampilkan komentar
- Jika belum ada lakukan proses penyimpanan
- Matikan isian data
- Hidupkan semua tombol kecuali tombol Simpan dan tombol Batal

c. Algoritma untuk tombol **Batal** (sama dengan kondisi Form Load)

- Cek apakah data sudah ada atau belum (kalau sudah bekerja dengan data)
- Jika data sudah ada, maka semua tombol dihidupkan kecuali tombol simpan data tombol Batal
- Jika data belum ada, maka semua tombol dimatikan kecuali tombol Tambah dan tombol Keluar. Bersihkan isian data
- Matikan isian data

d. Algoritma untuk tombol **Rubah**

- Matikan semua tombol kecuali tombol Simpan dan tombol Batal
- Aktifkan isian data
- Tempat kursor pada isian pertama yang akan dirubah

e. Algoritma untuk tombol **Hapus**

- Tampilkan kotak pesan untuk menanyakan kepastian proses penghapusan
- Jika dijawab Ya, maka lakukan penghapusan data
- Cek apakah posisi pointer berada diakhir record atau tidak
- Jika berada diakhir record, bawa ke awal record. Jika tidak diakhir record bawa pointer ke record berikutnya

f. Algoritma untuk tombol **Maju**

- Cek apakah pointer berada diakhir record atau tidak
- Jika berada diakhir record tampilkan komentar
- Jika tidak berada diakhir record, pindahkan pointer ke record berikutnya

g. Algoritma untuk tombol **Mundur**

- Cek apakah pointer berada di awal record
- Jia berada di awal record, tampilkan komentar
- Jika tidak berada diakhir record, pindahkan ke record sebelumnya.

## 7.11 Perintah-perintah Dalam Pengolahan Data

### 1. Append Blank

Perintah ini digunakan menambah record baru. Biasanya perintah ini digunakan pada saat proses penyimpanan

### 2. Replace

Perintah ini digunakan untuk mengganti nilai field dari record kosong yang sudah ditambahkan dengan suatu nilai tertentu (biasanya data yang dimasukkan pada objek di form)

3. Delete

Perintah ini digunakan untuk menghapus data (menandai record). Data dihapus tapi belum secara permanen. Maksudnya masih bisa dipanggil kembali.

4. Pack

Perintah ini digunakan untuk menghapus data yang sudah ditandai dari perintah Delete. Dengan perintah ini data yang sudah ditandai akan dihapus secara permanen.

5. Skip

Perintah ini berfungsi untuk memindahkan pointer ke record berikutnya

6. Skip -1

Perintah ini berfungsi untuk memindahkan pointer ke record sebelumnya

7. Goto Top

Perintah ini berfungsi untuk memindahkan pointer ke record paling atas

8. Goto Bottom

Perintah ini berfungsi untuk memindahkan pointer ke record paling bawah

9. Locate

Perintah ini digunakan untuk melakukan pencarian data baik untuk data yang tidak terindex maupun untuk data yang terindex.

10. Find

Perintah ini juga digunakan untuk pencarian data, namun khusus untuk data yang sudah terindex dan berjenis string

11. Seek

Perintah ini juga digunakan untuk pencarian data seperti perintah yang lain, namun hanya berlaku untuk data yang sudah terindex dan jenis data boleh selain string

12. EOF

Perintah ini digunakan untuk mengetahui posisi pointer apakah diposisi akhir record. EOF=**E**nd **O**f **F**ile.

13. BOF

Perintah ini digunakan untuk mengetahui posisi pointer apakah diposisi awal record. BOF=**B**egin **O**f **F**ile.

## 7.12 Praktek

Praktek pada pertemuan ini merupakan lanjutan dari sesi sebelumnya. Pada sesi ini, akan dicobakan proses pengelolaan data. Berikutnya akan dicoba untuk membuat kode program pada form pengolahan data tamu.

Untuk itu bukanlah Project ProHotel, lalu buka form FdtTamu atau Ftamu yang telah dibuat pada praktikum sebelumnya.

### 7.12.1 Membuat Kode-kode Program

Bagian ini biasanya terkadang masih sering didapati kesalahan dalam penempatan event untuk masing-masing objek yang diprogramkan. Untuk memberikan kesan mudah, maka proses pembuatan kode pada kesempatan ini dibuat dengan cara paling sederhana dan belum menerapkan prinsip modular. Jadi kode program ditulis berdasarkan event untuk masing-masing objek dan terkadang ada perintah yang sama ditulis berulang-ulang.

#### 1. Kode pada Form Active

- Klik dua kali pada bagian form yang kosong
- Ganti procedure dari Load ke Active
- Tulis kode program berikut:

No	Perintah
1	SET DELETED ON
2	SET DATE ITALIAN
3	SELECT dttamu
4	SET ORDER TO kdtamu
5	COUNT TO jmldata
6	IF jmldata>0
7	thisform.tblbaru.Enabled=.t.
8	thisform.tblsimpan.Enabled=.f.
9	thisform.tblbatal.Enabled=.f.
10	thisform.tblubah.Enabled=.t.
11	thisform.tblhapus.Enabled=.t.
12	thisform.tblkeluar.Enabled=.t.
13	GOTO top
14	thisform.text1.value=kdtamu
15	thisform.text2.value=nama
16	IF kel="L"
17	thisform.optiongroup1.Value=1
18	ELSE
19	thisform.optiongroup1.Value=2
20	endif
21	thisform.text3.value=alamat
22	thisform.text4.value=telp
23	IF kbangsa="WNI"
24	thisform.optiongroup2.Value=1
25	ELSE
26	thisform.optiongroup2.Value=2
27	endif
28	thisform.text5.value=tglcatat
29	ELSE
30	thisform.tblbaru.Enabled=.t.
31	thisform.tblsimpan.Enabled=.f.
32	thisform.tblbatal.Enabled=.f.
33	thisform.tblubah.Enabled=.f.
34	thisform.tblhapus.Enabled=.f.
35	thisform.tblkeluar.Enabled=.t.
36	thisform.text1.Value=" "
37	thisform.text2.Value=" "
38	thisform.optiongroup1.Value=0
39	thisform.text3.Value=" "
40	thisform.text4.Value=" "
41	thisform.optiongroup2.Value=0
42	thisform.text5.Value=CTOD("../../..")

43	ENDIF
44	thisform.text1.Enabled=.f.
45	thisform.text2.Enabled=.f.
46	thisform.text3.Enabled=.f.
47	thisform.text4.Enabled=.f.
48	thisform.text5.Enabled=.f.
49	thisform.optiongroup1.Enabled=.f.
50	thisform.optiongroup2.Enabled=.f.

#### Keterangan baris program/perintah:

Baris 1 berguna untuk mengaktifkan modus penghapusan (data yang telah dihapus tidak ditampilkan lagi). Baris 2 berguna untuk menetapkan format tanggal ke bentuk italia (sama seperti penulisan Indonesia yaitu tanggal, bulan dan tahun). Baris 3 berguna untuk mengaktifkan tabel dtamu. Baris 4 berguna untuk mengaktifkan index data berdasarkan kdtamu. Baris 5 berguna untuk menghitung jumlah record dari tabel dtamu dimana hasil perhitungan akan disimpan data variabel jmldata.

Baris 6 digunakan untuk menguji apakah tabel dtamu ada datanya (jmldata>0), jika data ada, maka komputer akan melaksanakan perintah yang ada pada baris 7 sampai baris 28 (baris 7 sampai baris 12 adalah untuk mengendalikan tombol, apakah aktif (.t.) atau non aktif (.f.), baris 13 memindah pointer ke posisi paling atas, baris 14 sampai baris 28 adalah mengambil data dari tabel dan menampilkan pada objek isian).

Baris 29 (jika data tidak ada – Else) maka komputer akan melaksanakan perintah yang ada pada baris 30 sampai baris 42 (baris 30 sampai baris 35 adalah untuk mengendalikan tombol, apakah aktif (.t.) atau non aktif (.f.), baris 36 sampai baris 42 untuk memberi nilai awal untuk masing-masing objek isian). Baris 43 adalah penutup perintah IF. Baris 44 sampai baris 50 adalah menon-aktifkan semua objek isian.

#### 2. Kode pada tombol Baru

- Klik dua kali pada tombol Baru
- Tulis kode program berikut:

No	Perintah
1	thisform.text1.Enabled=.t.
2	thisform.text2.Enabled=.t.
3	thisform.text3.Enabled=.t.
4	thisform.text4.Enabled=.t.
5	thisform.text5.Enabled=.t.
6	thisform.optiongroup1.Enabled=.t.
7	thisform.optiongroup2.Enabled=.t.
8	thisform.tblbaru.Enabled=.f.
9	thisform.tblsimpan.Enabled=.t.
10	thisform.tblbatal.Enabled=.t.
11	thisform.tblubah.Enabled=.f.
12	thisform.tblhapus.Enabled=.f.
13	thisform.tblkeluar.Enabled=.f.
14	thisform.text1.Value=""
15	thisform.text2.Value=""
16	thisform.optiongroup1.Value=0
17	thisform.text3.Value=""
18	thisform.text4.Value=""
19	thisform.optiongroup2.Value=0
20	thisform.text5.Value=DATE()
21	thisform.text1.SetFocus()

#### Keterangan baris program/perintah:

Baris 1 sampai baris 7 adalah perintah untuk mengaktifkan semua objek isian. Baris 8 sampai 13 adalah perintah untuk mengendalikan tombol, apakah aktif (.t.) atau non aktif (.f.). Baris 14 sampai baris 20 adalah perintah untuk memberi nilai awal untuk masing-masing objek isian (nilai yang diapit oleh tanda petik adalah untuk data string, nilai diawali dengan 0 adalah untuk data numerik, nilai diawali dengan Date() atau CTOD("..../..") untuk data tanggal). Baris 21 adalah perintah untuk menempatkan kursor di objek Text1.

### 3. Kode pada tombol Simpan

- Klik dua kali pada tombol Simpan
- Tulis kode program berikut:

No	Perintah
1	cekdata=thisform.text1.Value
2	SET DELETED off
3	SELECT dttamu
4	SET ORDER TO kdtamu
5	GOTO top
6	SEEK cekdata
7	IF .not. found()
8	APPEND BLANK
9	replace kdtamu WITH thisform.text1.Value
10	replace nama WITH thisform.text2.Value
11	IF thisform.optiongroup1.Value=1
12	replace kel WITH "L"
13	ELSE
14	replace kel WITH "P"
15	endif
16	replace alamat WITH thisform.text3.Value
17	replace telp WITH thisform.text4.Value
18	IF thisform.optiongroup2.Value=1
19	replace kbangsa WITH "WNI"
20	ELSE
21	replace kbangsa WITH "WNA"
22	endif
23	replace tglcatat WITH thisform.text5.value
24	ELSE
25	kom=MESSAGEBOX("Data sudah ada, mau diperbarui?",4+32,"Konfirmasi")
26	IF kom=6
27	IF DELETED()
28	RECALL
29	endif
30	replace kdtamu WITH thisform.text1.Value
31	replace nama WITH thisform.text2.Value
32	IF thisform.optiongroup1.Value=1
33	replace kel WITH "L"
34	ELSE
35	replace kel WITH "P"
36	endif
37	replace alamat WITH thisform.text3.Value
38	replace telp WITH thisform.text4.Value
39	IF thisform.optiongroup2.Value=1
40	replace kbangsa WITH "WNI"
41	ELSE
42	replace kbangsa WITH "WNA"
43	endif
44	replace tglcatat WITH thisform.text5.value
45	endif
46	ENDIF
47	SET DELETED on
48	thisform.grid1.Refresh()
49	thisform.tblbaru.Enabled=.t.
50	thisform.tblsimpan.Enabled=.f.
51	thisform.tblbatal.Enabled=.f.
52	thisform.tblubah.Enabled=.t.
53	thisform.tblhapus.Enabled=.t.
54	thisform.tblkeluar.Enabled=.t.
55	thisform.text1.Enabled=.f.

56	<code>thisform.text2.Enabled=.f.</code>
57	<code>thisform.text3.Enabled=.f.</code>
58	<code>thisform.text4.Enabled=.f.</code>
59	<code>thisform.text5.Enabled=.f.</code>
60	<code>thisform.optiongroup1.Enabled=.f.</code>
61	<code>thisform.optiongroup2.Enabled=.f.</code>

#### Keterangan baris program/perintah:

Baris 1 digunakan untuk mengisi variabel cekdata dengan nilai yang ada pada objek Text1 (nilai data kode tamu). Baris 2 untuk mematikan modus penghapusan. Baris 3 digunakan untuk mengaktifkan tabel dtamu. Baris 4 digunakan untuk mengaktifkan index data berdasarkan kdtamu. Baris 5 digunakan untuk membawa pointer ke posisi paling awal/atas. Baris 6 digunakan untuk proses pencarian data (seek) berdasarkan data pada variabel cekdata.

Baris 7 digunakan untuk menguji apakah data yang dicari tidak ketemu (.not. found()), jika tidak ketemu akan melaksanakan perintah baris 8 sampai baris 23. Baris 8 adalah perintah untuk menambah record kosong. Baris 9 sampai baris 23 adalah mengganti nilai field yang kosong dengan nilai yang ada pada masing-masing objek isian.

Baris 24 jika data ketemu (kebalikan baris 7), jika data ketemu maka akan melaksanakan perintah baris 25 sampai baris 45. Baris 25 adalah perintah untuk membuat kotak pesan dengan variabel penampungnya bernama kom, tombol yang akan ditampilkan Yes dan No, icon yang digunakan adalah icon tanda tanya. Baris 26 digunakan untuk menguji apakah tombol yang dipilih saat ditawarkan pada kotak pesan adalah tombol (Yes), jika ya maka akan dilaksanakan baris perintah mulai baris 27 sampai baris 45. Baris 27 menguji apakah data yang ketemu itu pernah dihapus. Baris 28 untuk memanggil kembali data yang pernah dihapus itu (recall). Baris 30 sampai 44 adalah mengganti nilai pada field dengan nilai yang ada pada objek isian.

Baris 47 untuk mengaktifkan modus penghapusan. Baris 48 untuk menyegarkan objek grid supaya data yang ditampilkan sesuai dengan kondisi yang terakhir. Baris 49 sampai baris 54 adalah perintah untuk mengendalikan tombol, apakah aktif (.t.) atau non aktif (.f.). Baris 55 sampai baris 61 digunakan untuk menon-aktifkan semua objek isian.

#### 4. Kode pada tombol Batal

- Klik dua kali pada tombol Batal
- Tulis kode program berikut:

No	Perintah
1	<code>SELECT dtamu</code>
2	<code>SET ORDER TO kdtamu</code>
3	<code>COUNT TO jmldata</code>
4	<code>IF jmldata&gt;0</code>
5	<code>    thisform.tblbaru.Enabled=.t.</code>
6	<code>    thisform.tblsimpan.Enabled=.f.</code>
7	<code>    thisform.tblbatal.Enabled=.f.</code>
8	<code>    thisform.tblubah.Enabled=.t.</code>
9	<code>    thisform.tblhapus.Enabled=.t.</code>
10	<code>    thisform.tblkeluar.Enabled=.t.</code>
11	<code>GOTO top</code>
12	<code>    thisform.text1.value=kdtamu</code>
13	<code>    thisform.text2.value=nama</code>
14	<code>    IF kel="L"</code>
15	<code>        thisform.optiongroup1.Value=1</code>
16	<code>    ELSE</code>
17	<code>        thisform.optiongroup1.Value=2</code>
18	<code>    endif</code>
19	<code>    thisform.text3.value=alamat</code>
20	<code>    thisform.text4.value=telp</code>
21	<code>    IF kbangsa="WNI"</code>

22	<code>thisform.optiongroup2.Value=1</code>
23	<code>ELSE</code>
24	<code>thisform.optiongroup2.Value=2</code>
25	<code>endif</code>
26	<code>thisform.text5.value=tglcatat</code>
27	<code>ELSE</code>
28	<code>thisform.tblbaru.Enabled=.t.</code>
29	<code>thisform.tblsimpan.Enabled=.f.</code>
30	<code>thisform.tblbatal.Enabled=.f.</code>
31	<code>thisform.tblubah.Enabled=.f.</code>
32	<code>thisform.tblhapus.Enabled=.f.</code>
33	<code>thisform.tblkeluar.Enabled=.t.</code>
34	<code>thisform.text1.Value=" "</code>
35	<code>thisform.text2.Value=" "</code>
36	<code>thisform.optiongroup1.Value=0</code>
37	<code>thisform.text3.Value=" "</code>
38	<code>thisform.text4.Value=" "</code>
39	<code>thisform.optiongroup2.Value=0</code>
40	<code>thisform.text5.Value=CTOD("../...")</code>
41	<code>ENDIF</code>
42	<code>thisform.text1.Enabled=.f.</code>
43	<code>thisform.text2.Enabled=.f.</code>
44	<code>thisform.text3.Enabled=.f.</code>
45	<code>thisform.text4.Enabled=.f.</code>
46	<code>thisform.text5.Enabled=.f.</code>
47	<code>thisform.optiongroup1.Enabled=.f.</code>
48	<code>thisform.optiongroup2.Enabled=.f.</code>

#### Keterangan baris program/perintah:

Baris perintah batal ini hampir sama dengan baris perintah yang terdapat pada form active. Hanya 2 baris pertama di Form active saja yang dihilangkan, sehingga penjelasannya lihat saja pada Form Active.

#### 5. Kode pada tombol Ubah

- Klik dua kali pada tombol Ubah
- Tulis kode program berikut:

No	Perintah
1	<code>thisform.text1.Enabled=.f.</code>
2	<code>thisform.text2.Enabled=.t.</code>
3	<code>thisform.text3.Enabled=.t.</code>
4	<code>thisform.text4.Enabled=.t.</code>
5	<code>thisform.text5.Enabled=.t.</code>
6	<code>thisform.optiongroup1.Enabled=.t.</code>
7	<code>thisform.optiongroup2.Enabled=.t.</code>
8	<code>thisform.tblbaru.Enabled=.f.</code>
9	<code>thisform.tblsimpan.Enabled=.t.</code>
10	<code>thisform.tblbatal.Enabled=.t.</code>
11	<code>thisform.tblubah.Enabled=.f.</code>
12	<code>thisform.tblhapus.Enabled=.f.</code>
13	<code>thisform.tblkeluar.Enabled=.f.</code>
14	<code>thisform.text2.SetFocus()</code>

#### Keterangan baris program/perintah:

Baris 1 sampai baris 7 adalah perintah untuk mengaktifkan semua objek isian kecuali objek Text1. Baris 8 sampai baris 13 merupakan perintah untuk mengendalikan tombol, apakah aktif (.t.) atau non aktif (.f.). Baris 14 digunakan untuk mengarahkan kursor ke objek text2 (nama tamu).



## 6. Kode pada tombol Hapus

- Klik dua kali pada tombol Hapus
- Tulis kode program berikut:

No	Perintah
1	tanya=MESSAGEBOX("Yakin mau dihapus?",4+32,"Konfirmasi")
2	IF tanya=6
3	DELETE
4	thisform.grid1.Refresh()
5	IF .not. EOF()
6	SKIP
7	IF EOF()
8	GOTO top
9	ENDIF
10	ELSE
11	GOTO top
12	ENDIF
13	thisform.text1.value=kdtamu
14	thisform.text2.value=nama
15	IF kel="L"
16	thisform.optiongroup1.Value=1
17	ELSE
18	thisform.optiongroup1.Value=2
19	endif
20	thisform.text3.value=alamat
21	thisform.text4.value=telp
22	IF kbangsa="WNI"
23	thisform.optiongroup2.Value=1
24	ELSE
25	thisform.optiongroup2.Value=2
26	endif
27	thisform.text5.value=tglcatat
28	ENDIF

### Keterangan baris program/perintah:

Baris 1 adalah perintah untuk membuat kotak pesan dengan variabel penampungnya bernama tanya, tombol yang akan ditampilkan Yes dan No (angka 4), icon yang digunakan adalah icon tanda tanya (angka 32). Baris 2 adalah perintah menguji apakah yang diklik adalah tombol Yes (angka 6), jika ya maka akan melaksanakan perintah baris 3 sampai baris 27.

Baris 3 perintah untuk menghapus data. Baris 4 adalah perintah untuk menyegarkan grid supaya datanya sesuai dengan kondisi terakhir. Baris 5 adalah perintah untuk menguji apakah saat ini berada di akhir file (End of File – EOF). Baris 6 adalah untuk memindah pointer ke record berikutnya. Baris 7 menguji apakah saat ini berada di akhir file (EOF). Baris 8 adalah perintah untuk membawa pointer ke posisi paling awal/atas. Baris 13 sampai baris 27 adalah perintah untuk menampilkan data (mengisi objek isian dengan nilai yang ada pada field dari record yang aktif).

7. Kode pada tombol Keluar
- Klik dua kali pada tombol Keluar
  - Tulis kode program berikut:

No	Perintah
1	<code>SELECT dttamu</code>
2	<code>GOTO top</code>
3	<code>SET DELETED off</code>
4	<code>COUNT TO jdh FOR DELETED()</code>
5	<code>IF jdh&gt;0</code>
6	<code>PACK</code>
7	<code>ENDIF</code>
8	<code>thisform.Release()</code>

**Keterangan baris program/perintah:**

Baris 1 adalah perintah untuk mengaktifkan tabel dttamu. Baris 2 adalah perintah untuk membawa pointer ke posisi paling awal/atas. Baris 3 adalah perintah untuk menon-aktifkan modus penghapusan. Baris 4 adalah perintah untuk menghitung data khusus untuk data yang pernah dihapus (deleted()). Baris 5 adalah perintah untuk menguji apakah ada data yang pernah dihapus, jika ada, maka akan dilakukan proses penghapusan secara permanen (pack). Baris 8 adalah perintah untuk menutup form.

8. Kode pada Grid dipindahkan letak kursor/pointer
- Klik dua kali pada grid
  - Ganti procedure ActivateCell menjadi **AfterRowColChange**
  - Tulis kode program berikut:

No	Perintah
1	<code>thisform.text1.value=kdtamu</code>
2	<code>thisform.text2.value=nama</code>
3	<code>IF kel="L"</code>
4	<code>    thisform.optiongroup1.Value=1</code>
5	<code>ELSE</code>
6	<code>    thisform.optiongroup1.Value=2</code>
7	<code>endif</code>
8	<code>thisform.text3.value=alamat</code>
9	<code>thisform.text4.value=telp</code>
10	<code>IF kbangsa="WNI"</code>
11	<code>    thisform.optiongroup2.Value=1</code>
12	<code>ELSE</code>
13	<code>    thisform.optiongroup2.Value=2</code>
14	<code>endif</code>
15	<code>thisform.text5.value=tglcatat</code>

**Keterangan baris program/perintah:**

Baris 1 sampai baris 15 adalah perintah untuk menampilkan data (mengisi objek isian dengan nilai yang ada pada field dari record yang aktif.

# Bab 8

## Pengujian, Perulangan, Fungsi, dan Prosedur

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami tentang pengujian, perulangan, fungsi, dan prosedur.

Pada pertemuan ini akan dibahas mengenai If .. Tehn, Do Case ... EndCase, For ... EndFor, Do While ... Enddo, For Each ... EndFor, fungsi-fungsi karakter, fungsi numerik, fungsi tanggal, dan prosedur.

### 8.1 Pengujian

#### 1. IF .... THEN

If ... Then merupakan kondisi atau ekspresi dasar untuk mengambil nilai yang bersifat true atau false dan memberi tindakan selanjutnya terhadap sifat dari suatu ekspresi. Adapun listingnya dapat ditulis sebagai berikut:

```
IF (kondisi) Then  
.... statemen/tindakan yang dilakukan  
Else  
.... selain itu  
Endif
```

Catatan : Penulisan perintah bisa juga tidak menggunakan Then.

#### 2. Do Case .... Endcase

Do Case ... EndCase merupakan struktur pengendalian cabang yang digunakan untuk memilih suatu tindakan pada situasi tertentu. Kondisi dari Do Case merupakan alternatif lain dari penggunaan If ... Then.

Listingnya dapat ditulis sebagai berikut:

```
Do Case (nama ekspresi)  
  Case (ekspresi pertama)  
    ... statemen/tindakan pertama  
  Case (ekspresi kedua)  
    ... statemen/tindakan kedua  
  Otherwise  
    ... statemen/tindakan lainnya  
EndCase
```

## 8.2 Perulangan

### 1. FOR ... ENDFOR

For ... EndFor merupakan perintah perulangan, yang digunakan untuk mengulang suatu kondisi sampai batas kondisi tersebut terpenuhi. Adapun listingnya dapat ditulis sebagai berikut:

```
FOR (variabel) = [nilai awal] TO [nilai akhir]  
  ... statemen/tindakan yang dilakukan  
ENDFOR | NEXT (variabel)
```

### 2. Do While .... Endo

Do While ... Enddo mempunyai fungsi yang sama seperti For ... EndFor yaitu perintah pengulangan untuk perulangan berakhir berdasarkan suatu kondisi. Listingnya dapat ditulis sebagai berikut:

```
Do While (nama ekspresi)  
  ....statemen/tindakan  
[Loop]  
[Exit]  
Enddo
```

### 3. For Each .... EndFor

For Each ... EndFor merupakan perintah yang digunakan untuk mengeksekusi pengulangan dari elemen array. Listingnya dapat ditulis sebagai berikut:

```
FOR EACH [nama variabel] IN [nama Array]  
  ... statemen/tindakan  
[Exit]  
[Loop]  
ENDFOR | NEXT
```

Contoh:

```
DIMENSION hari[7]  
Hari[1] = "Senin"  
Hari[2] = "Selasa"  
Hari[3] = "Rabu"  
Hari[4] = "Kamis"  
Hari[5] = "Jum'at"  
Hari[6] = "Sabtu"  
Hari[7] = "Minggu"
```

**FOR EACH A IN hari**

**Thisform.List1.additem ("Hari "+A)**

**ENDFOR**

### **8.3 Fungsi**

Banyak fungsi yang dapat digunakan dalam proses pengolahan data. Penggunaan fungsi tentu saja disesuaikan dengan proses atau hasil yang diinginkan.

#### **1. Fungsi Karakter**

Fungsi karakter yang dimaksudkan adalah fungsi-fungsi yang digunakan dalam hal pengelolaan data karakter, diantaranya:

##### **a. Fungsi STR**

Fungsi ini digunakan untuk merubah data numerik menjadi data karakter/string.

Bentuk umum perintah:

STR(n), dimana n adalah data numerik yang akan dirubah menjadi data karakter.

##### **b. Fungsi LEFT**

Fungsi ini digunakan untuk mengambil bagian kiri dari suatu karakter sejumlah tertentu.

Bentuk umum perintah:

LEFT(X,n), dimana X adalah data karakter yang akan diambil bagian kirinya. Sedangkan n adalah jumlah karakter yang diambil.

##### **c. Fungsi RIGHT**

Fungsi ini digunakan untuk mengambil bagian kanan dari suatu karakter sejumlah tertentu.

Bentuk umum perintah:

RIGHT(X,n), dimana X adalah data karakter yang akan diambil bagian kanannya. Sedangkan n adalah jumlah karakter yang diambil.

##### **d. Fungsi SUBSTR**

Fungsi ini digunakan untuk mengambil bagian tertentu dari suatu karakter (bisa di kiri, di kanan, atau di tengah).

Bentuk umum perintah:

SUBSTR(X,m,n), dimana X adalah data karakter yang akan diambil, m adalah posisi karakter (posisi karakter dihitung 1 untuk posisi paling kiri), sedangkan n adalah jumlah karakter yang diambil.

##### **e. Fungsi LTRIM**

Fungsi ini digunakan untuk menghilangkan data spasi di bagian kiri dari suatu data string.

Bentuk umum perintah:

LTRIM(X), dimana X adalah data string yang akan dihilangkan spasi di bagian kiri.

##### **f. Fungsi RTRIM**

Fungsi ini digunakan untuk menghilangkan data spasi di bagian kanan dari suatu data string.

Bentuk umum perintah:

RTRIM(X), dimana X adalah data string yang akan dihilangkan spasi di bagian kanan.

g. Fungsi TRIM

Fungsi ini digunakan untuk menghilangkan data spasi di bagian kiri dan di bagian kanan dari suatu data string.

Bentuk umum perintah:

TRIM(X), dimana X adalah data string yang akan dihilangkan spasinya.

2. Fungsi Numerik

Fungsi numerik yang dimaksudkan adalah fungsi-fungsi yang digunakan dalam hal pengelolaan data numerik, diantaranya:

a. Fungsi Val

Fungsi ini digunakan untuk merubah data karakter menjadi data numerik.

Bentuk umum perintah:

Val(X), dimana X adalah data karakter yang akan dirubah menjadi data numerik.

b. Fungsi Abs

Fungsi ini digunakan untuk mengambil nilai absolut dari suatu data numerik.

Bentuk umum perintah:

Abs(n), dimana n adalah data numerik yang akan dicari nilai absolutnya.

3. Fungsi Statistik

a. Fungsi Sum

Fungsi ini digunakan untuk menghitung hasil penjumlahan, biasanya digunakan untuk mencari nilai total dari suatu deretan data numerik.

Bentuk umum perintah:

Sum(n), dimana n adalah data yang akan dihitung hasil penjumlahannya

b. Fungsi Count

Fungsi ini digunakan untuk menghitung jumlah data. Biasanya banyak digunakan untuk menghitung data dari suatu deretan tertentu atau untuk menghitung jumlah record dalam basis data.

Bentuk umum perintah:

Count(n), dimana n adalah deretan data atau field yang akan dihitung

c. Fungsi Max

Fungsi ini digunakan untuk mencari nilai terbesar dari suatu deretan data numerik.

Bentuk umum perintah:

Max(n), dimana n adalah data yang akan dicari nilai terbesarnya.

d. Fungsi Min

Fungsi ini digunakan untuk mencari nilai terkecil dari suatu deretan data numerik.

Bentuk umum perintah:

Min(n), dimana n adalah data yang akan dicari nilai terkecilnya.

4. Fungsi Tanggal

Fungsi tanggal yang dimaksudkan adalah fungsi-fungsi yang digunakan dalam hal pengelolaan data tanggal, diantaranya:

a. Fungsi Day

Fungsi ini digunakan untuk menghasilkan nilai hari dari suatu data tanggal.

Bentuk umum perintah:

Day(D), dimana D adalah data tanggal.

b. Fungsi Month

Fungsi ini digunakan untuk menghasilkan nilai bulan dalam bentuk angka dari suatu data tanggal.

Bentuk umum perintah:

Month(D), dimana D adalah data tanggal.

c. Fungsi CMonth

Fungsi ini digunakan untuk menghasilkan nilai bulan dalam bentuk karakter dari suatu data tanggal.

Bentuk umum perintah:

CMonth(D), dimana D adalah data tanggal.

d. Fungsi Year

Fungsi ini digunakan untuk menghasilkan nilai tahun dari suatu data tanggal.

Bentuk umum perintah:

Year(D), dimana D adalah data tanggal.

e. Fungsi DOW

Fungsi ini digunakan untuk menghasilkan nilai hari dalam bentuk angka pada satu minggu dari suatu data tanggal.

Bentuk umum perintah:

DOW(D), dimana D adalah data tanggal.

f. Fungsi CDOW

Fungsi ini digunakan untuk menghasilkan nilai hari dalam bentuk karakter pada satu minggu dari suatu data tanggal.

Bentuk umum perintah:

CDOW(D), dimana D adalah data tanggal.

g. Fungsi CTOD

Fungsi ini digunakan untuk merubah data karakter menjadi data tanggal.

Bentuk umum perintah:

CTOD("..../..../..")

h. Fungsi DTOC

Fungsi ini merupakan kebalikan dari fungsi CTOD yaitu merubah data tanggal menjadi data karakter.

Bentuk umum perintah:

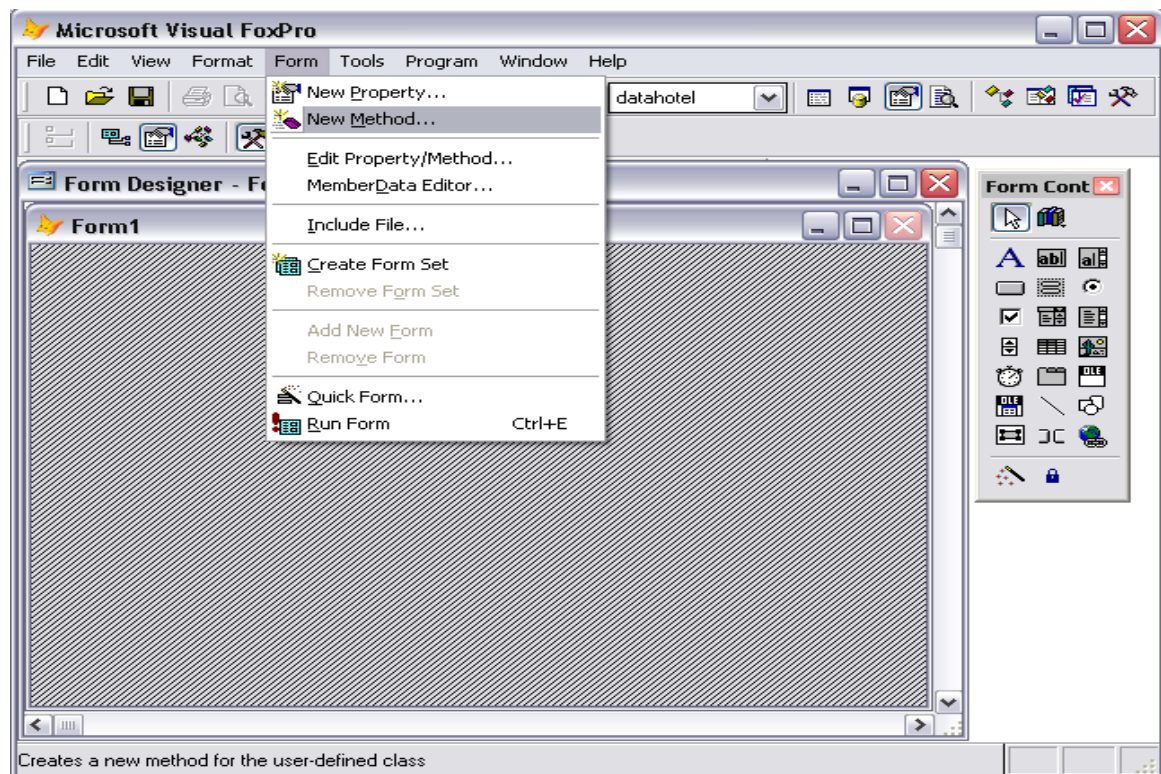
DTOC(D), dimana D adalah data tanggal.

## 8.4 Prosedur

Dengan menerapkan konsep program secara prosedur, maka program yang dibuat menjadi lebih praktis yaitu program mudah diperbaiki dan mudah untuk dikembangkan. Untuk itu apabila ada perintah-perintah yang sekiranya berulang-ulang digunakan, sebaiknya letakkan dalam sebuah prosedur. Setelah diletakkan dalam sebuah proses, maka setiap kali dibutuhkan tinggal memanggilnya.

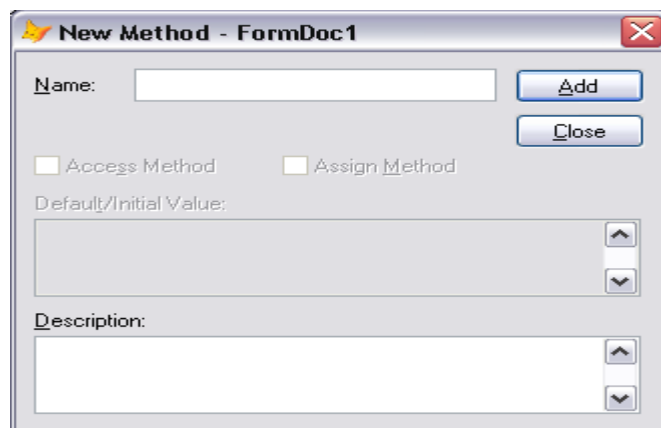
Untuk membuat prosedur lakukan langkah-langkah sebagai berikut:

1. Dalam kondisi berada pada desain form, tampilkan menu Form seperti terlihat pada Gambar 8.1.



Gambar 8.1 Tampilan menu Form untuk membuat prosedur

2. Pilih menu New Method, sehingga akan ditampilkan kotak dialog untuk memberi nama prosedur, seperti Gambar 8.2.



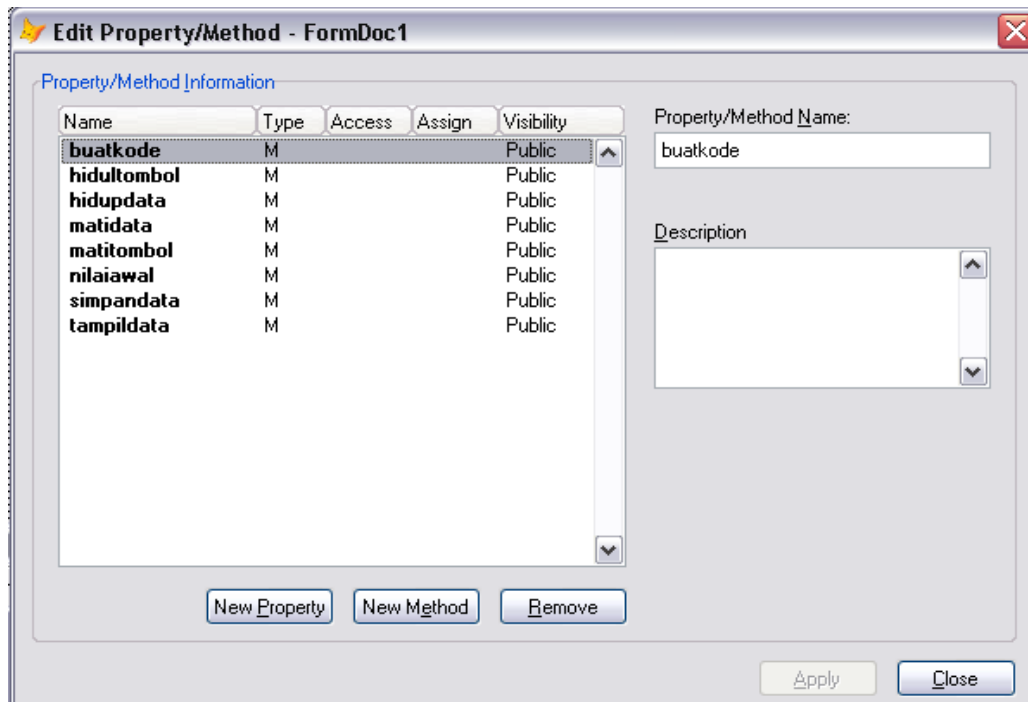
Gambar 8.2 Tampilan kotak dialog pembuatan prosedur

3. Ketikkan nama prosedur yang akan dibuat, kemudian klik tombol Add.
4. Jika masih ingin membuat prosedur yang lain, maka ulangi langkah ke 3 sampai semua prosedur selesai diciptakan. Semua prosedur yang telah dibuat akan masuk ke dalam jendela Properties.
5. Jika sudah selesai, akhiri dengan mengklik tombol Close.

Untuk melakukan perubahan ataupun untuk menghapus prosedur yang sudah dibuat, lakukan langkah sebagai berikut:



1. Tampilkan menu Form
2. Pilih menu Edit Property/Method, sehingga akan ditampilkankotak dialog seperti Gambar 8.3.



Gambar 8.3 Tampilan kotak dialog merubah prosedur

3. Lakukan proses yang diinginkan. Jika untuk merubah, klik nama prosedur yang akan dirubah, lalu perbaiki nama prosedur tersebut pada bagian sebelah kanan di bagian Property/Method Name. Jika ingin menghapus prosedur, maka klik prosedur yang akan dihapus, lalu klik tombol Remove. Jika sudah selesai melakukan proses, akhiri dengan mengklik tombol Close.

Untuk memanggil prosedur yang dibutuhkan dalam program dapat dilakukan dengan cara menuliskan kata `Thisform` lalu diikuti oleh nama prosedur yang diakhiri dengan menambahkan kurung buka dan kurung tutup. Misalkan nama prosedur yang akan dipanggil atau yang digunakan adalah **tampildata**, maka perintah yang digunakan adalah:

**`Thisform.tampildata()`**

## 8.5 Praktek

Praktek pada pertemuan ini merupakan lanjutan dari sesi sebelumnya. Pada sesi ini, akan dicobakan proses pengelolaan data untuk form transaksi, dimana untuk satu form akan melibatkan lebih dari satu tabel.

Untuk itu bukalah **Project ProHotel**, yang telah dibuat pada praktikum sebelumnya.

### 1. Membuat Form transaksi untuk data kamar

Untuk pengolahan data kamar membutuhkan dua tabel yaitu tabel kamar dan tabel jenis kamar. Lakukan langkah-langkah sebagai berikut:

- Rancanglah form seperti terlihat pada Gambar 8.4. Rancangan form bisa dibuat dari awal atau cukup melakukan Save As dari form yang sudah ada sebelumnya agar tidak terlalu lama dalam prosesnya.

The screenshot shows a Windows-style application window titled 'Sistem Administrasi Perhotelan' with a sub-header 'HOTEL PURNAMA'. Below the header is the address 'JL. SISINGAMANGARAJA NO. 43 YOGYAKARTA Telp (0274) 563323'. The main title of the form is 'PENGOLAHAN DATA KAMAR'. The form contains four labeled input fields: 'Nomor Kamar' (Text1), 'Kode Jenis Kamar' (Combo1), 'Nama Jenis Kamar' (Text2), and 'Status Kamar' (radio buttons for 'Kosong' and 'Isi'). Below these fields is a table with three columns: 'Nokamar', 'Kodejk', and 'Statuskm'. The first row of the table contains the text 'abl' in each column. At the bottom of the form are six buttons: 'Baru', 'Simpan', 'Batal', 'Ubah', 'Hapus', and 'Keluar'.

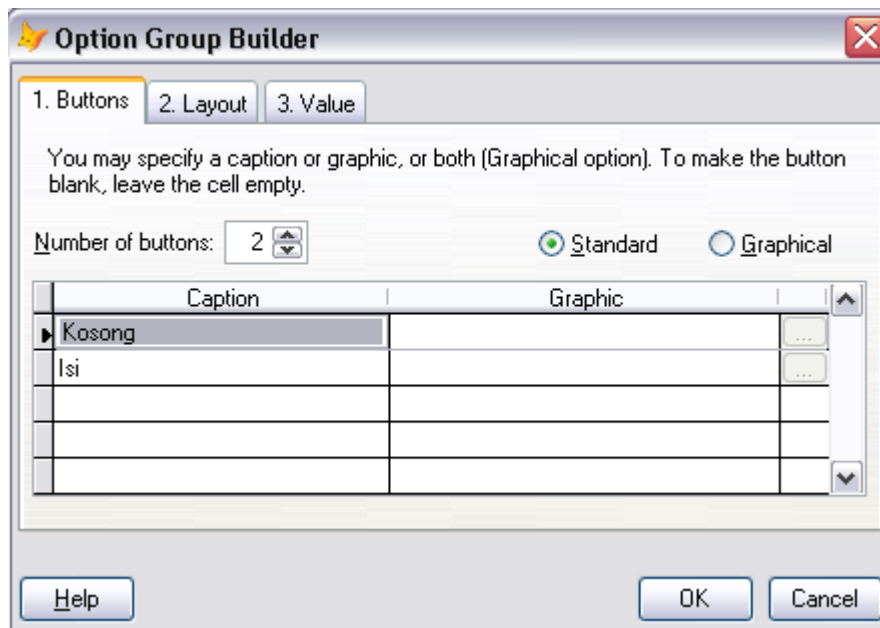
Gambar 8.4 Tampilan kotak dialog merubah prosedur

- Atur properties MaxLength untuk objek Text1. Langkahnya adalah klik objek Text1, cari properties Maxlength, kemudian isikan lebarnya sesuai lebar pada field yaitu 3.
- Atur properties untuk objek Text2. Klik Text2, cari properties Enabled, lalu ganti dengan False.
- Isilah pilihan untuk Kode Jenis Kamar dengan cara Klik kanan pada Combo1, pilih Builder dan atur kotak dialog seperti Gambar 8.5

The screenshot shows a 'Combo Box Builder' dialog box with four tabs: '1. List Items', '2. Style', '3. Layout', and '4. Value'. The '1. List Items' tab is active. It contains the text 'What items do you want in your combo box? Select a database or free table, and then select the fields.' Below this is a dropdown menu 'Fill the list with:' set to 'Fields from a table or view'. There are three panes: 'Databases and tables:' showing 'DATAHOTEL' selected; 'Available fields:' showing 'Tariffs' and 'Fasilitas'; and 'Selected fields:' showing 'Kodejk' and 'Namaik' selected. At the bottom are 'Help', 'OK', and 'Cancel' buttons.

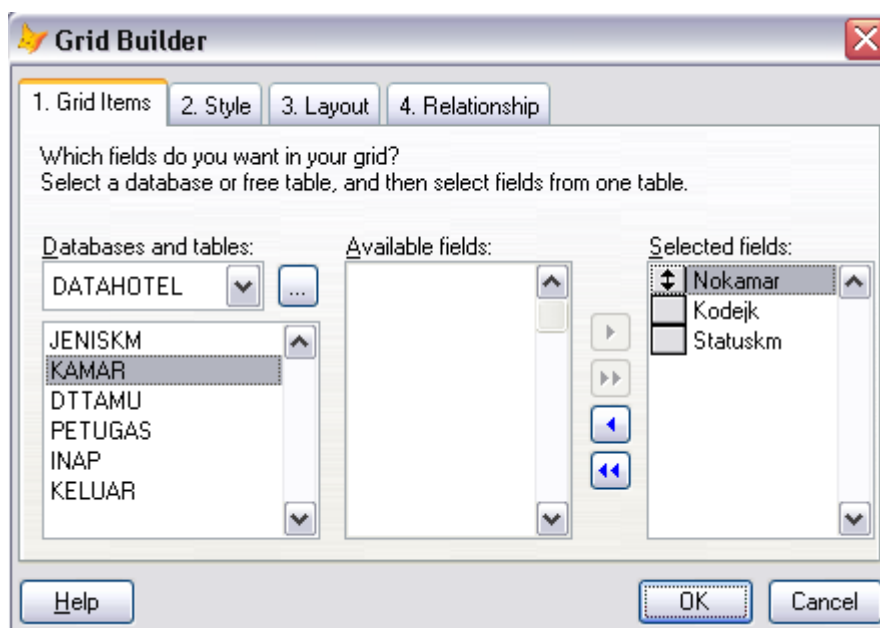
Gambar 8.5 Tampilan kotak dialog pembuatan kotak pilihan (Combo)

- e. Atur pilihan untuk Status Kamar dengan cara Klik kanan pada Optiongroup1, pilih Builder dan atur kotak dialog seperti Gambar 8.6.



Gambar 8.6 Tampilan kotak dialog pembuatan pilihan (Optiongroup)

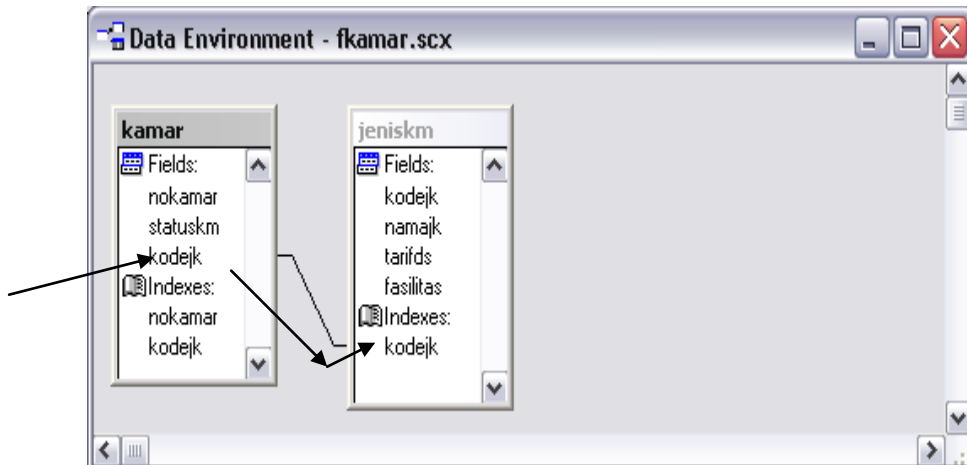
- f. Lakukan pengisian untuk tampilan data dengan cara Klik kanan pada Grid1, pilih Builder dan atur kotak dialog seperti Gambar 8.7



Gambar 8.7 Tampilan kotak dialog mengisi Grid

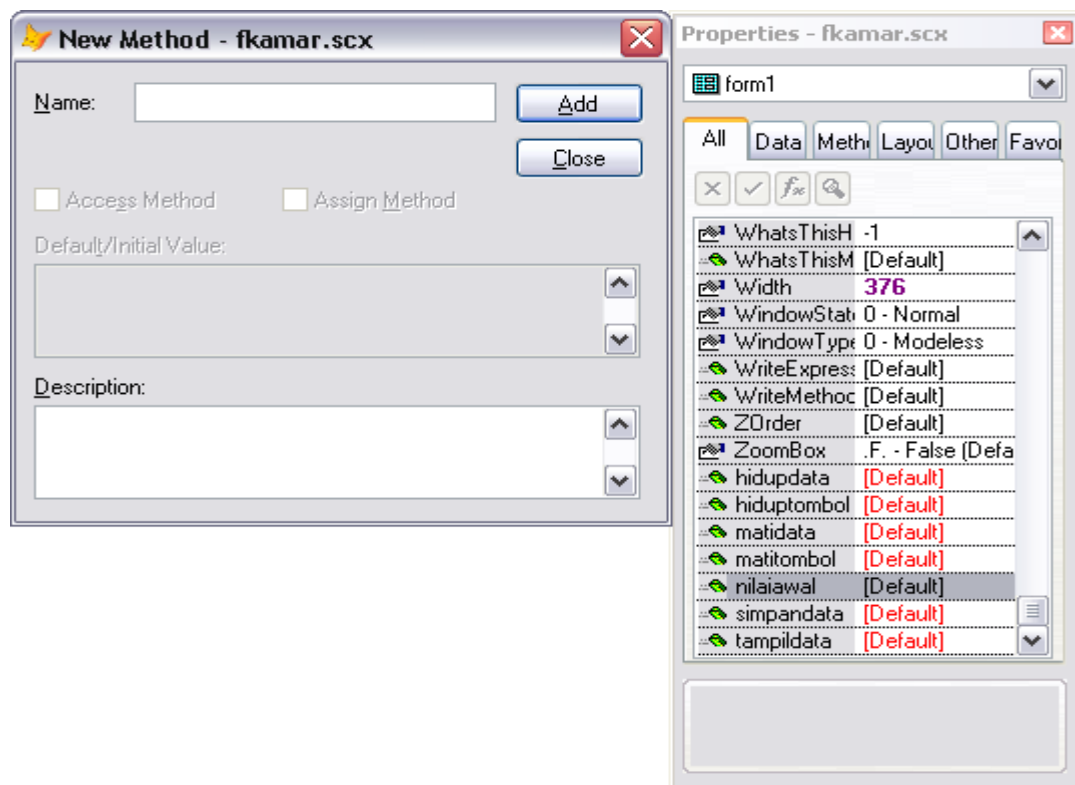
- g. Atur properties untuk masing-masing tombol (yang dirubah Caption dan Name) berturut-turut dari kiri ke kanan :
- 1) Tombol Baru yang dirubah adalah Caption : \<Baru, Name : tblbaru
  - 2) Tombol Simpan yang dirubah adalah Caption : \<Simpan, Name : tblsimpan
  - 3) Tombol Batal yang dirubah adalah Caption : B\<aru, Name : tblbatal
  - 4) Tombol Ubah yang dirubah adalah Caption : \<Ubah, Name : tblubah
  - 5) Tombol Hapus yang dirubah adalah Caption : \<Hapus, Name : tblhapus
  - 6) Tombol Keluar yang dirubah adalah Caption : \<Keluar, Name : tblkeluar

- h. Lakukan proses relasi dari tabel yang digunakan dengan cara Klik kanan pada tempat yang kosong sehingga akan masuk ke DataEnvironment. Drag field kodejk yang ada pada tabel kamar, lalu arahkan ke tabel jeniskm ke bagian indexes seperti terlihat pada Gambar 8.8



Gambar 8..8 Tampilan Data Environmen

- i. Kembali ke desain form, kemudian Ciptakanlah prosedur dengan cara tampilkan menu Form, lalu pilih menu New Method. Saat ditampilkan kotak dialog seperti Gambar 8.9, ketik nama prosedur, lalu klik tombol Add. Nama prosedur yang dibuat berturut-turut : **hidupdata**, **hiduptombol**, **matidata**, **matitombol**, **nilaiawal**, **simpandata**, **tampildata**. Jika sudah selesai, klik tombol Close.



Gambar 8..9 Tampilan pembuatan prosedur

- j. Tulis kode program pada masing-masing prosedur, dengan cara klik dua kali pada prosedur yang akan ditulis kode programnya. Sebagai contoh arahkan mouse ke window properties, lalu klik dua kali pada nama prosedur hidupdata. Tulis kode program sebagai berikut:

```
Object: form1 Procedure: hidupdata

thisform.text1.Enabled=.t.
thisform.combo1.Enabled=.t.
thisform.optiongroup1.enabled=.t.
```

- k. Tulis kode program untuk prosedur hiduptombol sebagai berikut:

```
Object: form1 Procedure: hiduptombol

thisform.tblbaru.enabled=.t.
thisform.tblsimpan.enabled=.t.
thisform.tblbatal.enabled=.t.
thisform.tblubah.enabled=.t.
thisform.tblhapus.enabled=.t.
thisform.tblkeluar.enabled=.t.
```

- l. Tulis kode program untuk prosedur matidata sebagai berikut:

```
Object: form1 Procedure: matidata

thisform.text1.Enabled=.f.
thisform.combo1.Enabled=.f.
thisform.optiongroup1.enabled=.f.
```

- m. Tulis kode program untuk prosedur matitombol sebagai berikut:

```
Object: form1 Procedure: matitombol

thisform.tblbaru.enabled=.f.
thisform.tblsimpan.enabled=.f.
thisform.tblbatal.enabled=.f.
thisform.tblubah.enabled=.f.
thisform.tblhapus.enabled=.f.
thisform.tblkeluar.enabled=.f.
```

- n. Tulis kode program untuk prosedur nilaiawal sebagai berikut:

```
Object: form1 Procedure: nilaiawal  
  
thisform.text1.value=""  
thisform.combo1.Value=""  
thisform.text2.Value=""  
thisform.optiongroup1.Value=0  
|
```

- o. Tulis kode program untuk prosedur simpandata sebagai berikut:

```
Object: form1 Procedure: simpandata  
  
replace nokamar WITH thisform.text1.Value  
replace kodejk WITH thisform.combo1.Value  
IF thisform.optiongroup1.Value=1  
    replace statuskm WITH "K"  
ELSE  
    replace statuskm WITH "I"  
ENDIF
```

- p. Tulis kode program untuk prosedur tampildata sebagai berikut:

```
Object: form1 Procedure: tampildata  
  
SELECT kamar  
thisform.text1.Value=kamar.nokamar  
thisform.combo1.value=kamar.kodejk  
thisform.text2.Value=jeniskm.namajk  
IF kamar.statuskm="K"  
    thisform.optiongroup1.Value=1  
ELSE  
    thisform.optiongroup1.Value=1  
ENDIF
```

- q. Tulis kode program saat Combo di klik sebagai berikut:

```
Object: Combo1 Procedure: Click  
  
IF thisform.combo1.Value<>" "  
    thisform.text2.Value=jeniskm.namajk  
ELSE  
    thisform.text2.Value=""  
endif
```

- r. Tulis kode program untuk form active sebagai berikut:

```
Object: form1 Procedure: Activate

SET DELETED ON
SET DATE ITALIAN
SELECT kamar
SET ORDER TO nokamar
COUNT TO jmldata
IF jmldata>0
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tblbatal.Enabled=.f.
    GOTO top
    thisform.tampildata()
ELSE
    thisform.matitombol()
    thisform.tblbaru.Enabled=.t.
    thisform.tblkeluar.Enabled=.t.
    thisform.nilaiawal()
ENDIF
thisform.matidata()
```

- s. Tulis kode program untuk tombol Baru sebagai berikut:

```
Object: tblbaru Procedure: Click

thisform.hidupdata()
thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.nilaiawal()
thisform.optiongroup1.Value=1
thisform.text1.SetFocus()
```

- t. Tulis kode program untuk tombol Simpan sebagai berikut:

```
Object: tblsimpan Procedure: Click View Parent Code

cekdata=thisform.text1.Value
SET DELETED off
SELECT kamar
SET ORDER TO nokamar
GOTO top
SEEK cekdata
IF .not. found()
    APPEND BLANK
    thisform.simpandata()
ELSE
    kom=MESSAGEBOX("Data sudah ada, mau diperbarui?",4+32,"Konfirmasi")
    IF kom=6
        IF DELETED()
            RECALL
        ENDIF
        thisform.simpandata()
    endif
ENDIF
SET DELETED on
thisform.grid1.Refresh()
thisform.hiduptombol()
thisform.tblsimpan.Enabled=.f.
thisform.tblbatal.Enabled=.f.
thisform.matidata()
```

- u. Tulis kode program untuk tombol Batal sebagai berikut:

```
Object: tlbatal Procedure: Click

SELECT kamar
SET ORDER TO knokamar
COUNT TO jmldata
IF jmldata>0
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tlbatal.Enabled=.f.
    GOTO top
    thisform.tampildata()
ELSE
    thisform.matitombol()
    thisform.tbldbaru.Enabled=.t.
    thisform.tbkeluar.Enabled=.t.
    thisform.nilaiawal()
ENDIF
thisform.matidata()
```

- v. Tulis kode program untuk tombol Ubah sebagai berikut:

```
Object: tlbubah Procedure: Click

thisform.hidupdata()
thisform.text1.Enabled=.f.
thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tlbatal.Enabled=.t.
thisform.combol.SetFocus()
```

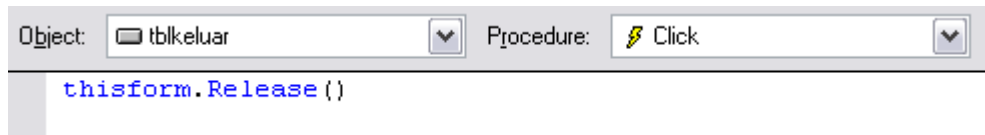
- w. Tulis kode program untuk tombol Hapus sebagai berikut:

```
Object: tlbhapus Procedure: Click

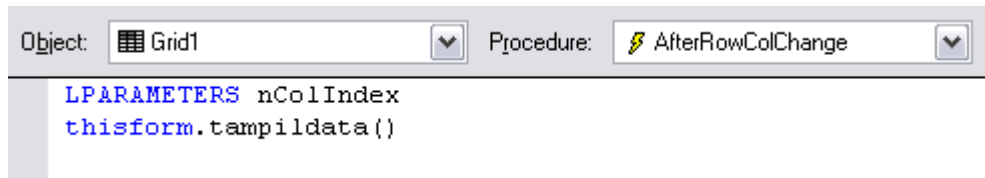
tanya=MESSAGEBOX("Yakin mau dihapus?",4+32,"Konfirmasi")
IF tanya=6
    SELECT kamar
    DELETE
    thisform.grid1.Refresh()
    IF .not. EOF()
        SKIP
        IF EOF()
            GOTO top
        ENDIF
    ELSE
        GOTO top
    ENDIF
    thisform.tampildata()
ENDIF
```



- x. Tulis kode program untuk tombol Keluar sebagai berikut:



- y. Tulis kode program untuk kondisi memindah/merubah tampilan pada Grid sebagai berikut:



Supaya tampilan grid lebih informatif, maka bentuk tampilan grid seperti terlihat pada Gambar 6.2 dapat dimodifikasi. Jika saat ini hanya ada tiga kolom dengan urutan nokamar, kodejk, dan statuskm maka bisa dirubah menjadi lima kolom dengan urutan No Kamar, Kode Jenis, Nama Jenis Kamar, Tarif Dasar, dan Status Kamar. Untuk melakukan hal tersebut ikuti langkah-langkah sebagai berikut:

- Klik Grid, lalu cari properties ColumnCount dan ubah nilainya dari 3 menjadi 5 akhiri dengan Enter.
- Klik kanan pada Grid, lalu pilih menu Edit
- Lakukan perubahan untuk judul kolom berturut-turut dari kiri ke kanan. Cara merubahnya adalah dengan mengklik pada bagian baris atas untuk kolom bersangkutan sampai pada properties akan muncul tulisan Header1 untuk kolom paling kiri. Ubah Caption pada properties dengan tulisan No. Kamar kemudian tekan Enter. Lanjutkan dengan mengubah kolom kedua, dan seterusnya sesuai dengan judul yang semestinya.
- Lakukan perubahan untuk isi data untuk masing-masing kolom berturut-turut dari kiri ke kanan. Cara merubahnya adalah dengan mengklik pada bagian baris kedua untuk kolom yang bersangkutan sampai pada properties akan muncul tulisan Column1 untuk kolom paling kiri. Atur properties ControlSource untuk kolom pertama yaitu kamar.nokamar. Untuk kolom kedua kamar.kodejk. Untuk kolom ketiga jeniskm.namajk. Untuk kolom keempat jeniskm.tarifds. Untuk kolom kelima kamar.statuskm. Selain mengatur properties ControlSource, perlu juga mengatur properties Aligment. Kolom pertama sampai kolom ketiga serta kolom kelima diatur rata kiri (*Midle Left*). Kolom keempat diatur rata kanan (*Midle Right*). Khusus untuk kolom keempat perlu juga mengatur properties InputMask dan properties Format yang diisi dengan 99,999,999.
- Hasil dari perubahan tampilan grid dapat dilihat pada Gambar 8.10.



Gambar 8..10 Tampilan grid yang sudah dimodifikasi

# Bab 9

## Pembuatan Form Transaksi

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami tentang pembuatan *form* transaksi, dimana satu *form* akan menggunakan lebih dari satu tabel.

Pada pertemuan ini akan dibahas mengenai tahap-tahapan rinci mengenai pembuatan form transaksi dengan mengembangkan form yang sudah dibuat pada pertemuan sebelumnya.

### 9.1 Pembuatan Form Transaksi

Pada semua bahasa pemrograman, proses pembuatan *form* transaksi memang menjadi perhatian khusus. Hal yang perlu diperhatikan adalah informasi-informasi yang akan disajikan pada layar/*interface*. Kode program memang agak sedikit lebih rumit jika dibandingkan dengan laporan yang menggunakan satu tabel.

### 9.2 Tampilan/Perhitungan Otomatis

Pada *form* transaksi memang banyak dibutuhkan tampilan-tampilan otomatis maupun proses perhitungan otomatis untuk membantu pengolahan data berikutnya. Contoh nyata untuk pernyataan ini adalah saat membuat form transaksi tamu menginap dan transaksi tamu keluar.

Pada form transaksi tamu menginap (*check in*), maka data petugas sudah otomatis muncul sesuai user yang login. Pada saat mengisi kode tamu, maka nama tamu, alamat tamu, dan telp tamu akan tampil secara otomatis. Pada saat mengisi nomor kamar, maka jenis kamar dan tarif kamar akan tampil secara otomatis.

Pada form transaksi tamu keluar (*check out*), maka data petugas juga sudah otomatis muncul sesuai user yang login. Pada saat mengisi nomor register, maka akan tampil secara otomatis informasi mengenai tamu beserta kamar dan tarifnya. Selanjutnya, pada saat setelah mengisi tanggal keluar dan jam keluar, maka secara otomatis akan keluar hasil perhitungan biaya yang harus dibayarkan.

Dari penjelasan di atas, dapat disimpulkan bahwa karena banyaknya tabel yang terkait, perlunya proses-proses perhitungan otomatis maka pembuatan form transaksi memang sedikit agak rumit. Untuk mengatasi hal tersebut, tentu diperlukan kesabaran dan rasa penasaran yang tinggi agar sistem yang dibangun bisa diselesaikan. Dalam pemrograman basis data (*database*), yang penting ingat nama masing-masing tabel dan ingat juga masing-masing fieldnya.

### 9.3 Praktek

Kegiatan praktek pada kesempatan ini adalah untuk melakukan pembuatan *form* transaksi, dimana untuk satu *form* akan melibatkan lebih dari satu tabel.

1. Membuat Form transaksi untuk data Inap

Untuk pengolahan data inap membutuhkan lima tabel yaitu tabel **inap**, tabel **dtamu**, tabel **kamar**, tabel **jeniskm**, dan tabel **petugas**. Lakukan langkah-langkah sebagai berikut:

- a. Untuk itu rancanglah form seperti terlihat pada Gambar 9.1. Rancangan *form* bisa dibuat dari awal atau cukup melakukan **Save As dari form kamar** agar tidak terlalu lama dalam prosesnya.

Sistem Administrasi Perhotelan

**Hotel Purnama**  
Jl. Adisucipto KM 6.3 Depok Sleman Yogyakarta  
**PENGOLAHAN DATA INAP TAMU**

Kode Petugas: Text1 Nama Petugas: Text2

No. Register: Text3 Tanggal Masuk: Text4  
Kode Tamu: Combo1 Jam Masuk: Text5  
Nama Tamu: Text6 No. Kamar: Combo2  
Alamat: Text7 Jenis Kamar: Text9  
Telepon: Text8 Tarif: Text10

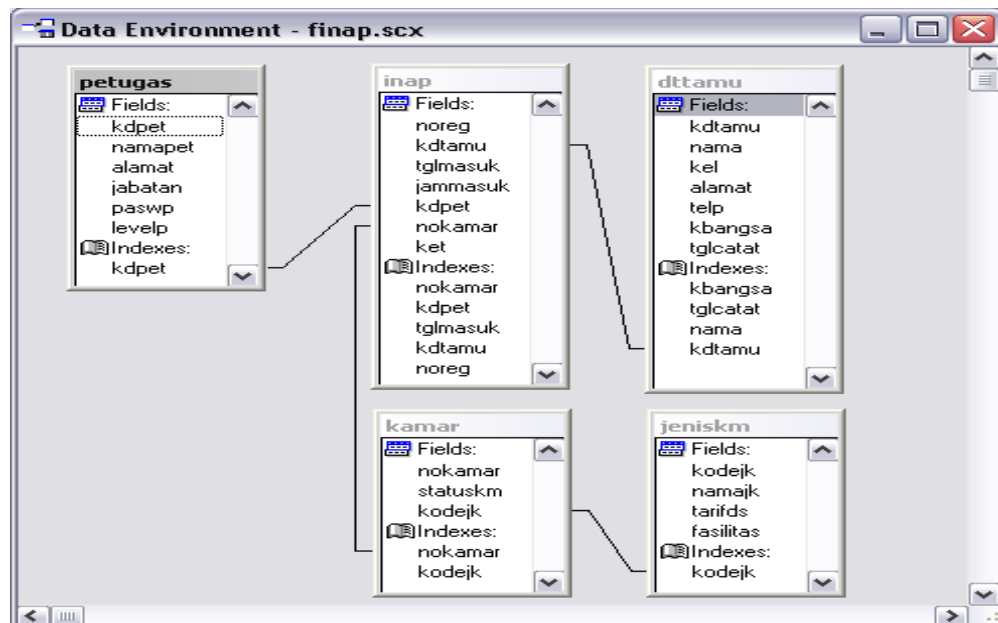
Noreg	Kdtamu	Tglmasuk	Jammasuk	Kdpet	Nokamar	Ket
abl	abl	abl	abl	abl	abl	abl

Baru Simpan Batal Ubah Hapus Keluar

Gambar 9.1 Desain Form Inap

- b. Dibutuhkan sebanyak 10 komponen **Text** dan 2 komponen **Combo**
- c. Aturlah properties **MaxLegth** untuk Text1, dan Text3. Sesuaikan dengan lebar field kdpet (Text1) yang ada di tabel Petugas, dan field Noreg (Text3) yang ada di tabel Inap.
- d. Buatlah daftar pilihan (Combo1) untuk kode tamu yang datanya diambil dari tabel DTTAMU. Klik kanan Combo1, lalu pilih Builder, kemudian pilih tabel DTTAMU sebagai sumber datanya, dimana field yang dipakai cukup field Kdtamu dan field nama.
- e. Buatlah daftar pilihan (Combo2) untuk No. kamar yang datanya diambil dari tabel KAMAR. Klik kanan Combo2, lalu pilih Builder, kemudian pilih tabel KAMAR sebagai sumber datanya, dimana field yang dipakai cukup field nokamar.

- f. Isilah Grid1 dengan klik kanan lalu pilih Builder, dimana tabel yang digunakan adalah tabel **INAP**. Susunlah *field* seperti gambar 9.1.
- g. Aturlah DataEnvironment, sehingga semua tabel yang dibutuhkan sudah masuk. Jangan lupa untuk membuat relasi diantara masing-masing tabel seperti Gambar 9.2.



Gambar 9.2 Tampilan DataEnvironment form Inap

- h. Ketik kode program untuk event Combo1 dipilih (Click) sebagai berikut

Object:  Procedure:

```

thisform.text6.value=dtttamu.nmtamu
thisform.text7.value=dtttamu.alamat
thisform.text8.value=dtttamu.telp

```

- i. Buatlah kode program untuk event Combo2 dipilih (Click) sebagai berikut:

Object:  Procedure:  [View Parent Code](#)

```

IF kamar.statuskm="I" then
    p=MESSAGEBOX("Kamar ini sudah terisi, coba yang lain..",0+6,"Informasi")
    thisform.combo1.SetFocus()
else
    thisform.text9.value=jeniskm.namajk
    thisform.text10.value=jeniskm.tarifs
endif

```

- j. Tambahkan prosedur untuk membuat kode otomatis (dengan catatan, prosedur-prosedur yang lain sudah ikut dari pengolahan data kamar).

- k. Buatlah kode program untuk masing-masing prosedur yang disediakan. Apabila prosedur sudah ada akibat melakukan Save As pada form sebelumnya, maka kode program tinggal disesuaikan. Apabila prosedur belum ada, maka dibuat terlebih dahulu prosedur yang dibutuhkan.
- 1) Ketik kode program untuk prosedur pembuatan kode otomatis

```
Object: Form1 Procedure: buatkode

SET ORDER TO noreg
GOTO bottom
IF noreg=SPACE(8) then
    vnoreg="00000001"
ELSE
    urut=VAL(noreg)+1
    DO case
    case urut<10
        vnoreg="0000000"+LTRIM(STR(urut))
    case urut<100
        vnoreg="000000"+LTRIM(STR(urut))
    case urut<1000
        vnoreg="00000"+LTRIM(STR(urut))
    case urut<10000
        vnoreg="0000"+LTRIM(STR(urut))
    case urut<100000
        vnoreg="000"+LTRIM(STR(urut))
    case urut<1000000
        vnoreg="00"+LTRIM(STR(urut))
    case urut<10000000
        vnoreg="0"+LTRIM(STR(urut))
    case urut<100000000
        vnoreg=LTRIM(STR(urut))
    ENDcase
ENDIF
thisform.text3.Value=vnoreg
```

- 2) Ketik kode program untuk prosedur hidupdata

```
Object: Form1 Procedure: hidupdata

thisform.text3.Enabled=.t.
thisform.text4.Enabled=.t.
thisform.text5.Enabled=.t.
thisform.combo1.Enabled=.t.
thisform.combo2.Enabled=.t.
|
```

- 3) Ketik kode program untuk prosedur hiduptombol

```
Object: Form1 Procedure: hiduptombol

thisform.tblbaru.Enabled=.t.
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.tblubah.Enabled=.t.
thisform.tblhapus.Enabled=.t.
thisform.tblkeluar.Enabled=.t.
```

- 4) Ketik kode program untuk prosedur matidata

```
Object: Form1 Procedure: matidata

thisform.text3.Enabled=.f.
thisform.text4.Enabled=.f.
thisform.text5.Enabled=.f.
thisform.combo1.Enabled=.f.
thisform.combo2.Enabled=.f.
```

- 5) Ketik kode program untuk prosedur matitombol

```
Object: Form1 Procedure: matitombol

thisform.tblbaru.Enabled=.f.
thisform.tblsimpan.Enabled=.f.
thisform.tblbatal.Enabled=.f.
thisform.tblubah.Enabled=.f.
thisform.tblhapus.Enabled=.f.
thisform.tblkeluar.Enabled=.f.
```

- 6) Ketik kode program untuk prosedur nilaiawal

```
Object: Form1 Procedure: nilaiawal View Parent Code

thisform.text1.Value="P01" && nilai P01 hanya sebagai contoh
thisform.text2.Value=""
thisform.text3.Value=""
thisform.text4.Value=DATE()
thisform.text5.Value=DATETIME()
thisform.text7.Value=""
thisform.text8.Value=""
thisform.text9.Value=""
thisform.text10.Value=0
thisform.combo1.Value=""
thisform.combo2.Value=""
```

- 7) Ketik kode program untuk prosedur simpandata

```
Object: Form1 Procedure: simpandata

replace kdpet with thisform.text1.Value
replace noreg with thisform.text3.Value
replace tglmasuk with thisform.text4.Value
replace jammasuk with thisform.text5.Value
replace kdtamu with thisform.combo1.Value
replace nokamar with thisform.combo2.Value
replace ket WITH "I"
```

- 8) Ketik kode program untuk prosedur tampildata

```
Object: Form1 Procedure: tampildata

thisform.text1.Value=inap.kdpet
thisform.text2.Value=petugas.namapet
thisform.text3.Value=inap.noreg
thisform.text4.Value=inap.tglmasuk
thisform.text5.Value=inap.jammasuk
thisform.combo1.Value=inap.kdtamu
thisform.text6.Value=tamu.nmtamu
thisform.text7.Value=tamu.alamat
thisform.text8.Value=tamu.telp
thisform.combo2.Value=inap.nokamar
thisform.text9.Value=jeniskm.namajk
thisform.text10.Value=jeniskm.tarifds
```

I. Buatlah kode program untuk masing-masing tombol

1) Ketik kode program untuk tombol Baru

```
Object: tblbaru Procedure: Click

thisform.nilaiawal()
thisform.hidupdata()
SELECT inap
thisform.buatkode()
thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.combo1.SetFocus()
```

2) Ketik kode program untuk tombol Simpan

```
Object: tblsimpan Procedure: Click View Parent Code

mnoreg=thisform.text3.Value
SELECT inap
SET ORDER TO noreg
GOTO top
SEEK mnoreg
IF FOUND()
    kom=MESSAGEBOX("Data sudah ada, mau diperbarui?",4+32,"Konfirmasi")
    IF kom=6
        thisform.simpandata()
    endif
ELSE
    APPEND BLANK
    thisform.simpandata()
    &&merubah status kamar menjadi terisi (I)
    SELECT kamar
    GOTO top
    vnokamar=thisform.combo2.Value
    SEEK vnokamar
    IF FOUND()
        replace statuskm WITH "I"
    endif
ENDIF
GOTO bottom
thisform.grid1.Refresh()
thisform.hiduptombol()
thisform.tblsimpan.Enabled=.f.
thisform.tblbatal.Enabled=.f.
thisform.matidata()
```

- 3) Ketik kode program untuk tombol Batal

```
Object: tblbatal Procedure: Click

SELECT INAP
SET ORDER TO noreg
COUNT TO jmldata
IF jmldata>0 THEN &&jika data ada
    GOTO top
    thisform.tampildata()
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tblbatal.Enabled=.f.
ELSE && jika data tidak ada
    thisform.nilaiawal()
    thisform.matitombol()
    thisform.tblbaru.Enabled=.t.
    thisform.tblkeluar.Enabled=.t.
ENDIF
thisform.matidata()
```

- 4) Ketik kode program untuk tombol Ubah

```
Object: tblubah Procedure: Click

thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.hidupdata()
thisform.text4.SetFocus()
```

- 5) Ketik kode program untuk tombol Hapus

```
Object: tblhapus Procedure: Click View Parent Code

cek=MESSAGEBOX("Yakin data ini mau dihapus?",4+32,"Konfirmasi")
if cek=6
    SELECT INAP
    DELETE
    IF EOF()
        GOTO top
    ELSE
        SKIP
    ENDIF
    thisform.tampildata()
    thisform.grid1.Refresh()
ENDIF
```

- 6) Ketik kode program untuk tombol Keluar

```
Object: tblkeluar Procedure: Click

thisform.Release()
```



- m. Buatlah kode program pada Form Active

```
SET DELETED ON
SELECT INAP
SET ORDER TO noreg
COUNT TO jmldata
IF jmldata>0 THEN &&jika data ada
    GOTO top
    thisform.tampildata()
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tblbatal.Enabled=.f.
ELSE && jika data tidak ada
    thisform.nilaiawal()
    thisform.matitombol()
    thisform.tblbaru.Enabled=.t.
    thisform.tblkeluar.Enabled=.t.
ENDIF
thisform.matidata()
```

- n. Buatlah kode program pada saat Grid interactiveChange

```
LPARAMETERS nColIndex
thisform.tampildata()
```

- o. Lakukan modifikasi untuk Grid supaya tampilan data lebih informatif

Jika diperhatikan bentuk rancangan form seperti terlihat pada Gambar 9.1, secara khusus kolom-kolom yang ada pada Grid, masih kurang informatif.

Coba perhatikan kolom kedua yaitu kolom kdtamu (orang akan kesulitan memahami informasi kode tamu, tentu akan lebih jelas apabila pada kolom itu yang muncul adalah nama tamu). Bagaimana caranya? Padahal tabel yang digunakan untuk grid adalah tabel inap yang tidak ada field nmtamu (nama tamu). Field nmtamu hanya ada di tabel dtamu.

Perhatikan juga kolom kelima yaitu kolom kdpet (orang akan kesulitan memahami informasi kode petugas, tentu akan lebih jelas apabila pada kolom itu yang muncul adalah nama petugas). Bagaimana caranya? Padahal tabel yang digunakan untuk grid adalah tabel inap yang tidak ada field namapet (nama petugas). Field namapet hanya ada di tabel petugas.

Pekerjaan yang harus dilakukan adalah melakukan modifikasi pada objek Grid (hasil jadi bisa dilihat pada Gambar 9.3) dengan langkah sebagai berikut:

- Klik kanan pada Grid
- Pilih Edit
- Lakukan perubahan untuk judul kolom dengan mengklik pada bagian judul grid, sehingga pada properties akan aktif bagian header. Lakukan perubahan judul dengan mengetik pada bagian Caption.
- Lakukan perubahan untuk field (khusus untuk kolom kedua (kdtamu) dan kolom kelima (kdpet)). Lakukan perubahan untuk field dengan melakukan perubahan isian properties pada bagian ControlSource, dan pilih (cari dengan mengklik combo) untuk field yang diinginkan.

Sistem Administrasi Perhotelan

## Hotel Purnama

Jl. Adisucipto KM 6.3 Depok Sleman Yogyakarta

### PENGOLAHAN DATA INAP TAMU

Kode Petugas

Nama Petugas

No. Register   
 Kode Tamu   
 Nama Tamu   
 Alamat   
 Telepon

Tanggal Masuk   
 Jam Masuk   
 No. Kamar   
 Jenis Kamar   
 Tarif

No. Registrasi	Nama Tamu	Tgl Masuk	Jam Masuk	Nama Petugas	No. Kamar	Keterangan
abl	abl	abl	abl	abl	abl	abl

Klik disin untuk  
merubah field

Klik disin untuk  
merubah judul

Properties - finap.scx

Column2

All Data Methc Layout Other Favori

☒ ☐ ☐ ☐ ☐ tamu.nmtamu

BaseClass Column

Bound T. - True (Default)

Class Column

ClassLibrary (None)

ColumnOrder 2

Comment (None)

ControlCount 2

ControlSource tamu.nmtamu

Controls 0

CurrentContro Text1

Destroy (Default)

Specifies the source of data to which an object is bound.

Properties - finap.scx

Header1

All Data Methc Layout Other Favori

☒ ☐ ☐ ☐ ☐ AddProperty [Default]

Alignment 0 - Middle Left (D

BackColor 240,240,240

BaseClass Header

Caption Nama Tamu

Class Header

ClassLibrary (None)

Click [Default]

Comment (None)

DblClick [Default]

Destroy [Default]

Adds a new property to an object.

Gambar 9.3 Desain Form Inap Setelah Dimodifikasi

Catatan:

Supaya lebih mudah dan cepat dalam melakukan modifikasi terhadap grid, maka kerjakan terlebih dahulu semua yang terkait judul (header). Setelah semua judul selesai, baru melakukan perubahan terhadap field (ControlSource).

2. Membuat Form transaksi untuk data tamu keluar

Untuk pengolahan data inap membutuhkan enam tabel yaitu tabel keluar, tabel inap, tabel dtamu, tabel kamar, tabel jenis kamar, dan tabel petugas. Lakukan langkah-langkah sebagai berikut:

- Untuk itu rancanglah form seperti terlihat pada Gambar 9.4. Rancangan form bisa dibuat dari awal atau cukup melakukan Save As dari form inap agar tidak terlalu lama dalam prosesnya.

**Sistem Administrasi Perhotelan**

**Hotel Purnama**  
 Jl. Adisucipto KM 6.3 Depok Sleman Yogyakarta  
**PENGOLAHAN DATA TAMU KELUAR**

Kode Petugas:  Nama Petugas:

No. Register:  Tanggal Keluar:

Tanggal Masuk:  Jam Keluar:

Jam Masuk:

Nama Tamu:  Biaya Inap:

Alamat:  Biaya Lain:

No. Kamar:  Pajak:

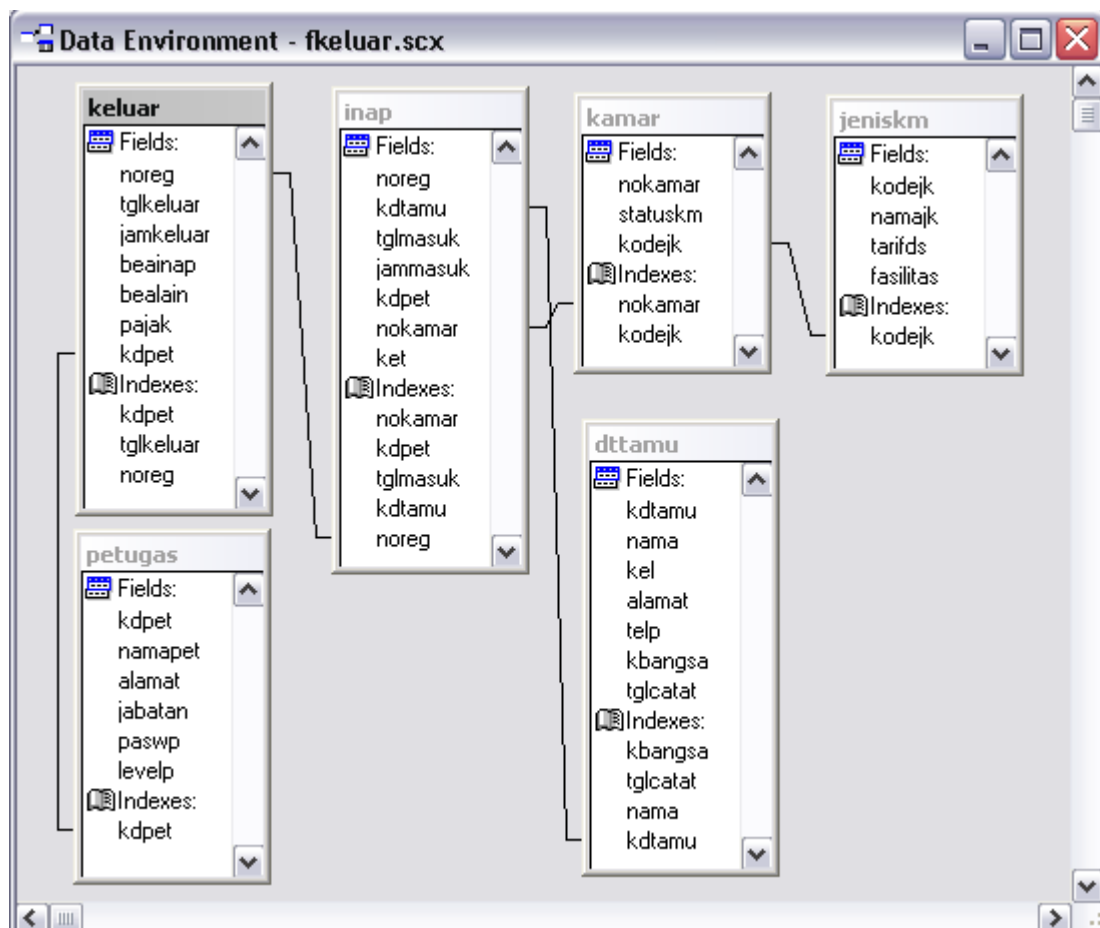
Tarif:  Jumlah Bayar:

Noreg	Tgkeluar	Jamkeluar	Beainap	Bealain	Pajak	Kdpet
abl	abl	abl	abl	abl	abl	abl

**Baru Simpan Batal Ubah Hapus Keluar**

Gambar 9.4 Tampilan desain form tamu keluar

- Aturlah properties Inputmask dan properties Format untuk objek Text8, Text11, Text12, Text13 dan Text14, mengisikan 99,999,999 agar tampilan angka bisa disajikan dengan format keuangan.
- Isilah Grid1 dengan klik kanan lalu pilih Builder, dimana tabel yang digunakan adalah tabel **KELUAR**. Susunlah field seperti Gambar 9.4.
- Aturlah DataEnvironment, sehingga semua tabel yang dibutuhkan sudah masuk. Jangan lupa untuk membuat relasi diantara masing-masing tabel seperti Gambar 9.5.
- Buatlah daftar pilihan (Combo1) untuk No. Register yang datanya diambil dari tabel **INAP**. Klik kanan Combo1, lalu pilih Builder, kemudian pilih tabel INAP sebagai sumber datanya, dimana field yang dipakai cukup field noreg.



Gambar 9.5 Tampilan DataEnvironment Form Tamu Keluar

- f. Cek prosedur-prosedur yang ada (karena melakukan Save As dari form finap, maka prosedurnya akan sama). Untuk prosedur pembuatan kode otomatis (prosedur buatkode) tidak dipergunakan, maka dihapus saja.
- g. Sesuaikan kode program untuk masing-masing prosedur yang disediakan.
  - 1) Kode program untuk prosedur hidupdata

Object: Form1 Procedure: hidupdata

```

thisform.combo1.Enabled=.t.
thisform.text9.Enabled=.t.
thisform.text10.Enabled=.t.
thisform.text12.Enabled=.t.

```

- 2) Kode program untuk prosedur hiduptombol

Object: Form1 Procedure: hiduptombol

```

thisform.tblbaru.Enabled=.t.
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.tblubah.Enabled=.t.
thisform.tblhapus.Enabled=.t.
thisform.tblkeluar.Enabled=.t.

```

3) Kode program untuk prosedur matidata

```
Object: Form1 Procedure: matidata

thisform.combo1.Enabled=.f.
thisform.text9.Enabled=.f.
thisform.text10.Enabled=.f.
thisform.text12.Enabled=.f.
```

4) Kode program untuk prosedur matitombol

```
Object: Form1 Procedure: matitombol

thisform.tblbaru.Enabled=.f.
thisform.tblsimpan.Enabled=.f.
thisform.tblbatal.Enabled=.f.
thisform.tblubah.Enabled=.f.
thisform.tblhapus.Enabled=.f.
thisform.tblkeluar.Enabled=.f.
```

5) Kode program untuk prosedur nilaiawal

```
Object: Form1 Procedure: nilaiawal View Parent Code

thisform.text1.Value="P01" && nilai P01 hanya sebagai contoh
thisform.text2.Value=""
thisform.text3.Value=DATE()
thisform.text4.Value=DATETIME()
thisform.text5.Value=""
thisform.text6.Value=""
thisform.text7.Value=""
thisform.text8.Value=0
thisform.text9.Value=DATE()
thisform.text10.Value=DATETIME()
thisform.text11.Value=0
thisform.text12.Value=0
thisform.text13.Value=0
thisform.text14.Value=0
thisform.combo1.Value=""
```

6) Kode program untuk prosedur simpandata

```
Object: Form1 Procedure: simpandata

replace kdpet with thisform.text1.Value
replace noreg with thisform.combo1.Value
replace tglkeluar with thisform.text9.Value
replace jamkeluar with thisform.text10.Value
replace beainap with thisform.text11.Value
replace bealain with thisform.text12.Value
replace pajak with thisform.text13.Value
```

7) Kode program untuk prosedur tampildata



Object:  Form1 Procedure:  tampildata

```

thisform.text1.Value=inap.kdpet
thisform.text2.Value=petugas.namapet
thisform.combo1.Value=keluar.noreg
thisform.text3.Value=inap.tglmasuk
thisform.text4.Value=inap.jammasuk
thisform.text5.Value=dtamu.nmtamu
thisform.text6.Value=dtamu.alamat
thisform.text7.Value=inap.nokamar
thisform.text8.Value=jeniskm.tarifds
thisform.text9.Value=keluar.tglkeluar
thisform.text10.Value=keluar.jamkeluar
thisform.text11.Value=keluar.beainap
thisform.text12.Value=keluar.bealain
thisform.text13.Value=keluar.pajak
a=thisform.text11.value
b=thisform.text12.value
c=thisform.text13.value
thisform.text14.Value=a+b+c

```

h. Sesuaikan kode program pada Form Active

Object:  Form1 Procedure:  Activate

```

SET DELETED ON
SELECT KELUAR
SET ORDER TO noreg
COUNT TO jmldata
IF jmldata>0 THEN &&jika data ada
    GOTO top
    thisform.tampildata()
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tblbatal.Enabled=.f.
ELSE && jika data tidak ada
    thisform.nilaiawal()
    thisform.matitombol()
    thisform.tblbaru.Enabled=.t.
    thisform.tblkeluar.Enabled=.t.
ENDIF
thisform.matidata()

```

- i. Buatlah kode program untuk event Combo1 dipilih (Click)

Object: Combo1 Procedure: Click

```
cek=thisform.combo1.Value
SELECT inap
GOTO top
SEEK cek
IF FOUND()
    thisform.text3.Value=inap.tglmasuk
    thisform.text4.Value=inap.jammasuk
    thisform.text5.Value=dt tamu.nmtamu
    thisform.text6.Value=dt tamu.alamat
    thisform.text7.Value=inap.nokamar
    thisform.text8.Value=jeniskm.tarifds
ENDIF
```

- j. Buatlah kode program untuk event Text9 LostFocus

Object: Text9 Procedure: LostFocus [View Parent Code](#)

```
IF thisform.text9.Value<thisform.text3.Value
p=MESSAGEBOX("Tanggal keluar < tanggal masuk, silahkan dicek..",0+64,"Infoormasi")
thisform.text9.SetFocus()
ELSE
lamainap=thisform.text9.Value-thisform.text3.Value
thisform.text11.Value=lamainap*thisform.text8.value
ENDIF
```

- k. Buatlah kode program untuk event Text12 InteractiveChange

Object: Text12 Procedure: InteractiveChange [View Parent Code](#)

```
&&mengisi variabel a dengan nilai yang ada di text11
a=thisform.text11.Value
&&mengisi variabel b dengan nilai yang ada di text12
b=thisform.text12.value
&&mengisi variabel c dengan nilai yang ada di text11+text12 dikali 10%
c=(thisform.text11.Value+thisform.text12.value)*0.1
&&mengisi text14 dengan penjumlahan masing-masing elemen pembayaran
thisform.text14.Value=a+b+c
```

- l. Buatlah kode program untuk masing-masing tombol

- 1) Kode program untuk tombol Baru

Object: tblbaru Procedure: Click

```
thisform.nilaiawal()
thisform.hidupdata()
SELECT KELUAR
thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tblbatal.Enabled=.t.
thisform.combo1.SetFocus()
```

2) Kode program untuk tombol Simpan

Object:  Procedure:  [View Parent Code](#)

```
mnoreg=thisform.combol.Value
SELECT KELUAR
SET ORDER TO noreg
GOTO top
SEEK mnoreg
IF FOUND()
    kom=MESSAGEBOX("Data sudah ada, mau diperbarui?",4+32,"Konfirmasi")
    IF kom=6
        thisform.simpendata()
    endif
ELSE
    APPEND BLANK
    thisform.simpendata()
    &&merubah keterangan inap menjadi keluar (K)
    SELECT inap
    GOTO top
    vnoreg=thisform.combol.Value
    SEEK vnoreg
    IF FOUND()
        replace ket WITH "K"
    endif
    &&merubah status kamar menjadi kosong (K)
    SELECT kamar
    GOTO top
    vnokamar=thisform.text7.Value
    SEEK vnokamar
    IF FOUND()
        replace statuskm WITH "K"
    endif
ENDIF

GOTO bottom
thisform.grid1.Refresh()
thisform.hiduptombol()
thisform.tblsimpan.Enabled=.f.
thisform.tblbatal.Enabled=.f.
thisform.matidata()
```



3) Kode program untuk tombol Batal

```
Object: tlbatal Procedure: Click

SELECT KELUAR
SET ORDER TO noreg
COUNT TO jmldata
IF jmldata>0 THEN &&jika data ada
    GOTO top
    thisform.tampildata()
    thisform.hiduptombol()
    thisform.tblsimpan.Enabled=.f.
    thisform.tlbatal.Enabled=.f.
ELSE && jika data tidak ada
    thisform.nilaiawal()
    thisform.matitombol()
    thisform.tblbaru.Enabled=.t.
    thisform.tblkeluar.Enabled=.t.
ENDIF
thisform.matidata()
```

4) Kode program untuk tombol Ubah

```
Object: tlubah Procedure: Click

thisform.matitombol()
thisform.tblsimpan.Enabled=.t.
thisform.tlbatal.Enabled=.t.
thisform.hidupdata()
thisform.text9.SetFocus()
```

5) Kode program untuk tombol Hapus

```
Object: tlhapus Procedure: Click View Parent Code

cek=MESSAGEBOX("Yakin data ini mau dihapus?",4+32,"Konfirmasi")
if cek=6
    SELECT KELUAR
    DELETE
    IF EOF()
        GOTO top
    ELSE
        SKIP
    ENDIF
    thisform.tampildata()
    thisform.grid1.Refresh()
ENDIF
```

6) Kode program untuk tombol Keluar

```
Object: tblkeluar Procedure: Click

thisform.Release()
```

m. Lakukan modifikasi untuk Grid supaya tampilan lebih informatif

Jika diperhatikan Gambar 9.4, tampilan Grid terdiri dari 7 kolom, hal ini karena memang pada tabel KELUAR sebagai sumber datanya hanya terdiri dari 7 field. Apabila dicermati secara seksama, ada informasi penting yang belum ada pada grid tersebut, misalnya kolom untuk nama tamu, kolom untuk tanggal masuk, kolom untuk lama inap, dan kolom untuk jumlah bayar. Pertanyaan yang muncul adalah bagaimana untuk menambahkan jumlah kolom dari 7 kolom menjadi 11 kolom. Pertanyaan selanjutnya adalah bagaimana caranya jika kolom yang akan ditampilkan tidak ada fieldnya. Jika kolom nama tamu dan kolom tanggal masuk tidak jadi masalah karena fieldnya ada pada tabel dtamu (nmtamu) dan tabel inap (tglmasuk). Untuk kolom lama inap, kolom jumlah bayar bisa dihasilkan dengan membuat rumus atau fungsi.

Untuk melakukan hal itu semua, lakukan perubahan untuk properties grid. Langkah pertama adalah menambahkan jumlah kolom dengan cara

- Klik Grid
- Lakukan perubahan pada ColumnCount, yang nilainya 7 dirubah menjadi 11.
- Klik kanan pada grid, lalu pilih Edit
- Lakukan perubahan untuk judul kolom (header) dan field (column), sehingga tampilan grid bisa terlihat seperti disajikan pada Gambar 9.6.

**Sistem Administrasi Perhotelan**

**Hotel Purnama**  
 Jl. Adisucipto KM 6.3 Depok Sleman Yogyakarta  
**PENGOLAHAN DATA TAMU KELUAR**

Kode Petugas: Text1      Nama Petugas: Text2

No. Register: Combo1      Tanggal Keluar: Text9

Tanggal Masuk: Text3      Jam Keluar: Text10

Jam Masuk: Text4

Nama Tamu: Text5

Alamat: Text6

No. Kamar: Text7

Tarif: Text8

Biaya Inap: Text11

Biaya Lain: Text12

Pajak: Text13

Jumlah Bayar: Text14

No. Register	Nama Tamu	Tgl Masuk	Tgl Keluar	Jam Keluar	Lama Inap	Bea Inap	Bea Lain	Pajak	Jml Bayar	Nama Petugas
abl	abl	abl	abl	abl	abl	abl	abl	abl	abl	abl

Baru   Simpan   Batal   Ubah   Hapus   Keluar

Gambar 9.6 Tampilan desain form tamu keluar yang sudah dimodifikasi

# Bab 10

## Membuat Laporan

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami dan menerapkan proses pembuatan laporan sebagai hasil dari pengolahan data.

Pada pertemuan ini akan dibahas mengenai proses pembuatan laporan dengan berbagai variasinya. Laporan yang dibuat masih bersifat sederhana, maksudnya baru menggunakan satu tabel.

### 10.1 Mengetahui dan Membuat Laporan

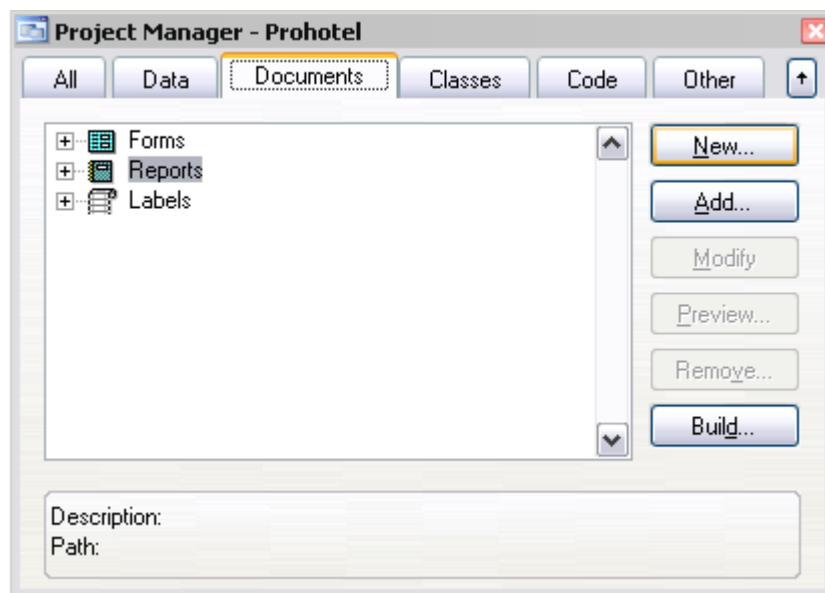
Laporan merupakan suatu hasil yang didapatkan dari proses pengolahan data. Untuk membuat laporan dapat memanfaatkan fasilitas yang disediakan oleh Visual FoxPro. Pilihan fasilitas untuk menghasilkan laporan ada dua yaitu Reports dan Labels. Dengan memanfaatkan Reports hasil yang didapat adalah informasi yang memanfaatkan utuh satu halaman. Sedangkan Labels pada satu halaman bisa dibentuk menjadi kolom-kolom kecil seperti Kartu Tanda Penduduk (KTP), Surat Izin Mengemudi (SIM), Kartu Mahasiswa, dan lain-lain.

Untuk membuat Reports maupun Labels caranya hampir sama dengan proses pembuatan form yaitu dapat dilakukan dengan cara otomatis (*wizard*) atau dengan cara manual.

#### 1. Membuat Reports

Untuk membuat reports dapat dilakukan dengan langkah-langkah sebagai berikut:

- Dari Project Manager, klik tab **Documents**
- Klik pilihan **Reports** (lihat Gambar 10.1)



Gambar 10.1 Langkah Awal Pembuatan Reports

- c. Klik tombol **New**
- d. Saat ditampilkan kotak dialog untuk pilihan proses pembuatan reports, pilih salah satu. Untuk pemula disarankan memilih **Wizard**, yang dilanjutkan untuk menentukan jenis reports yang akan dibuat seperti Gambar 10.2





Gambar 10.2 Pilihan Jenis Report

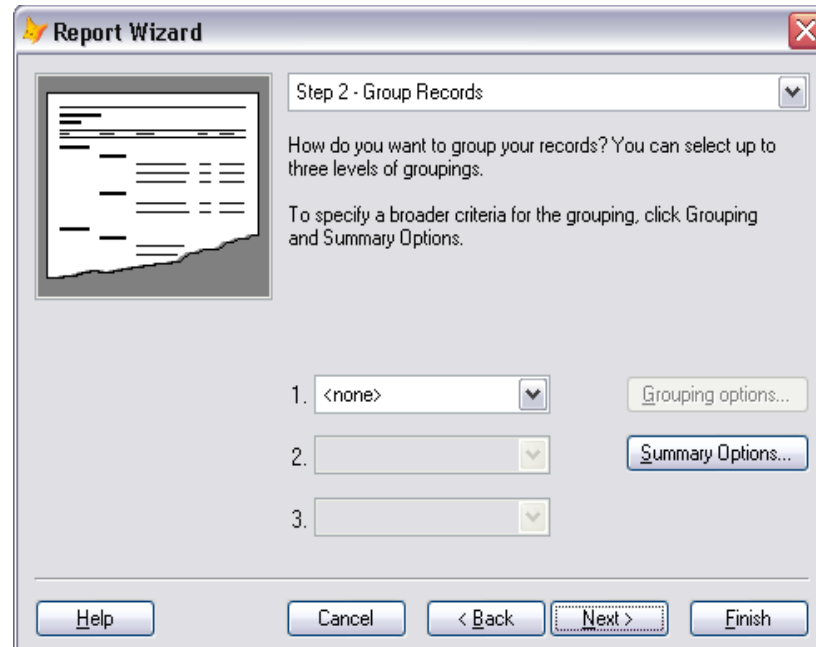
Pilihan pertama (*One-to-Many Report Wizard*) maksudnya adalah membuat report dengan lebih dari satu tabel yang relasinya adalah satu ke banyak. Sedangkan pilihan kedua (*Report Wizard*) maksudnya adalah membuat report standar yaitu hanya satu tabel (tabel yang lain dapat ditambahkan pada saat desain report).

- e. Pilih/klik salah satu, sesuai kebutuhan (untuk tahap awal pilih yang kedua dulu), lanjutkan dengan mengklik tombol **OK**, sehingga akan ditampilkan kotak dialog seperti Gambar 10.3.



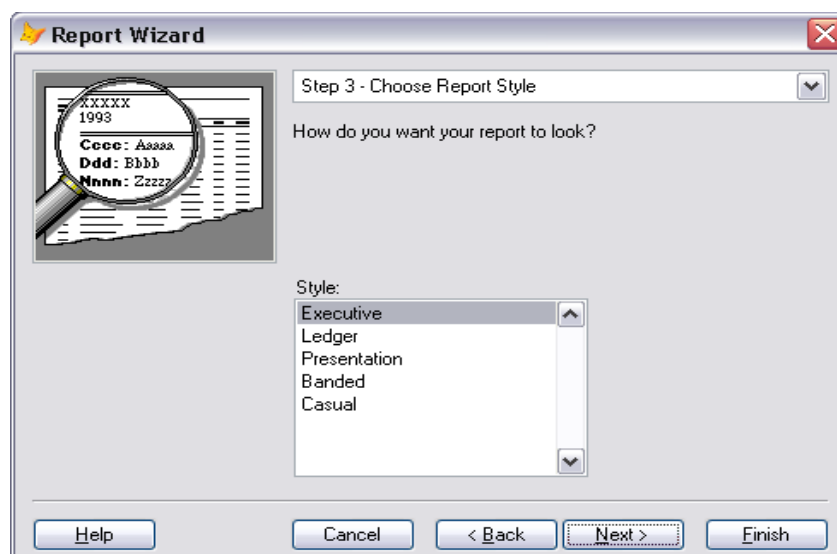
Gambar 10.3 Proses Menentukan Tabel dan Field untuk Report

- f. Tentukan tabel yang akan dibuatkan reportnya (klik kolom sebelah kiri). Jika nama tabel belum muncul, klik tombol titik tiga (...) untuk mencari nama tabel. Saat mengklik tabel, maka pada kolom bagian tengah akan ditampilkan field-fieldnya.
- g. Tentukan/pilih field yang akan dicantumkan pada laporan. Jika ingin mengambil semua field, klik tombol panah dua (  ). Jika memilih satu-persatu gunakan tombol panah satu (  ).
- h. Setelah memilih field, klik tombol Next, sehingga akan ditampilkan kotak dialog seperti Gambar 10.4



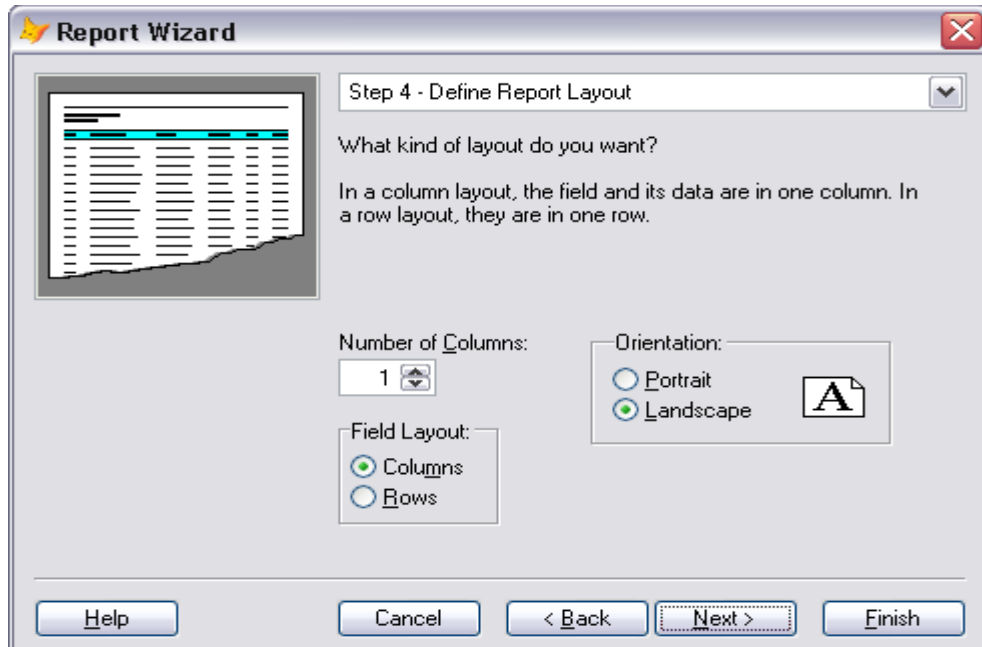
Gambar 10.4 Kotak Dialog Group Report

- i. Tentukan apakah report yang dibuat akan dikelompokkan atau tidak. Jika ingin dikelompokkan, pilih field yang menjadi dasar pengelompokkan (klik combo, pilih field). Jika tidak ingin dikelompokkan langsung saja mengklik tombol Next (catatan: pengelompokkan dapat juga dilakukan pada desain report). Saat setelah mengklik tombol Next akan ditampilkan kotak dialog seperti Gambar 10.5



Gambar 10.5 Kotak Dialog Jenis Report

- j. Tentukan jenis report (pilih salah satu dari yang ditampilkan pada kolom style), lanjutkan dengan mengklik tombol Next sehingga akan ditampilkan kotak dialog seperti Gambar 10.6



Gambar 10.6 Kotak Dialog Orientasi Report

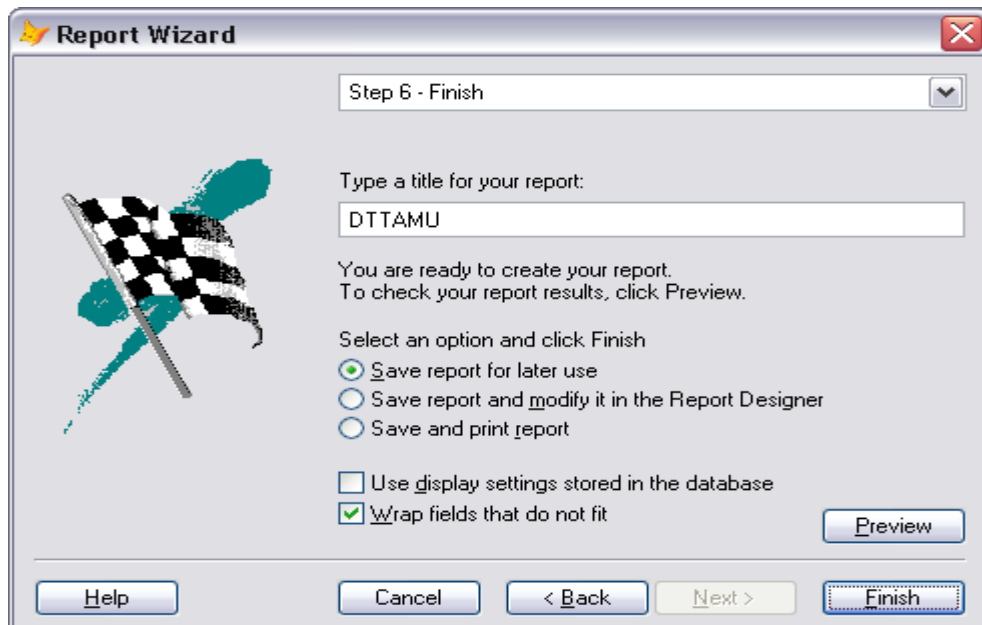
- k. Tentukan orientasi laporan (Potrait=tegak, Landscape=melebar), lanjutkan mengklik tombol Next dan akan ditampilkan kotak dialog seperti Gambar 10.7



Gambar 10.7 Kotak Dialog Pengurutan Report

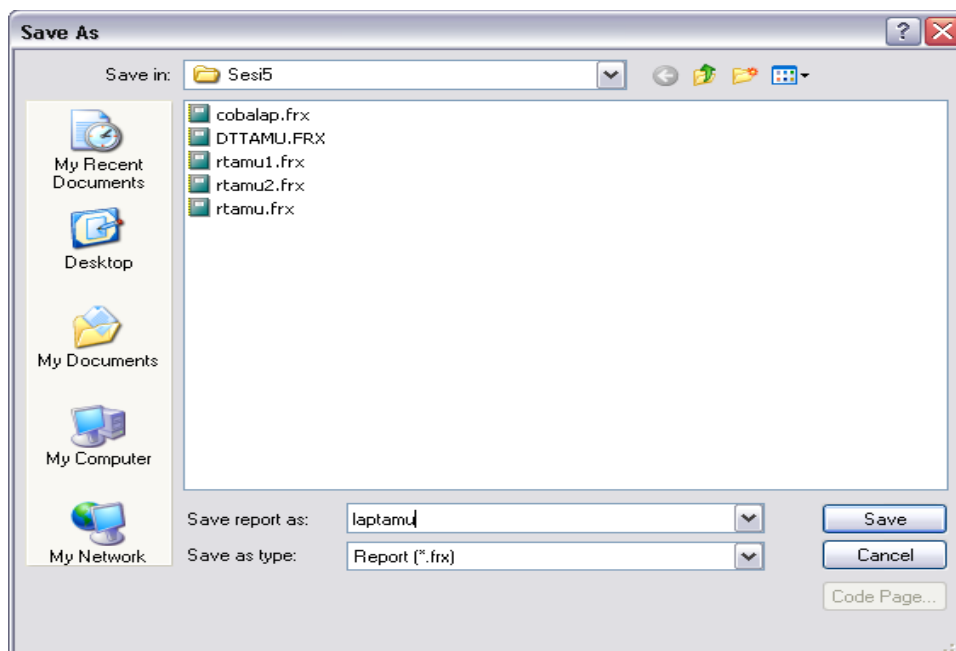
- l. Tetapkan apakah data akan diurutkan menurut field tertentu. Jika diinginkan, maka klik field yang akan dijadikan acuan pengurutan lalu klik tombol Add, lalu tetapkan

juga metode pengurutan (Ascending atau Descending) kemudian klik tombol Next. Jika tidak diurutkan maka lanjutkan saja mengklik tombol Next, sehingga akan ditampilkan kotak dialog seperti Gambar 10.8



Gambar 10.8 Kotak Dialog Akhir Pembuatan Report

- m. Klik tombol Finish, sebagai tanda berakhirnya pembuatan report sehingga akan ditampilkan kotak dialog penyimpanan seperti terlihat pada Gambar 10.9



Gambar 10.9 Kotak Dialog Penyimpanan Report

- n. Ketikkan nama report yang dibuat (misal LAPTAMU) dan akhiri dengan mengklik tombol **Save**.  
o. Report sudah selesai dan siap untuk dilakukan modifikasi sesuai kebutuhan

## 2. Membuat Labels

Proses pembuatan label hampir sama dengan proses pembuatan report. Ada beberapa langkah yang berbeda. Untuk itu ikuti langkah sebagai berikut:

- a. Dari Project Manajer, klik tab Documents
- b. Klik pilihan Labels
- c. Klik tombol New
- d. Klik pilihan Label Wizard
- e. Jika tabel belum ditampilkan, maka cari tabel dengan cara mengklik tombol titik tiga
- f. Setelah mengklik/menentukan tabel lanjutkan dengan mengklik tombol Next
- g. Pilih jenis Label yang akan dibuat, lanjutkan dengan mengklik tombol Next
- h. Definisikan layout Label dengan menyeleksi field dan menambahkan Text. Jika sudah selesai klik tombol Next
- i. Tetapkan apakah data akan diurutkan menurut field tertentu. Jika diinginkan, maka klik field yang akan dijadikan acuan pengurutan lalu klik tombol Add, lalu tetapkan juga metode pengurutan (Ascending atau Descending) kemudian klik tombol Next.
- j. Klik tombol Finish sebagai tanda proses pembuatan Label sudah selesai
- k. Berikan nama untuk menyimpan Label yang sudah dihasilkan (misal LblTamU), lanjutkan mengklik tombol Save
- l. Label sudah selesai dan siap untuk dimodifikasi sesuai dengan selera masing-masing.

## 10.2 Memanggil Laporan

### 1. Memanggil Report

Bentuk umum perintah untuk memanggil report adalah sebagai berikut:

**REPORT FORM <namareport> [FOR <Ekspresi>] [PREVIEW/TO PRINT]**

REPORT FORM adalah perintah dasar yang harus ditulis seperti itu

<namareport> adalah nama report yang akan ditampilkan

[FOR <Ekspresi>] adalah batasan untuk menampilkan report

PREVIEW adalah memberitahu ke sistem bahwa report akan dilihat di layar

TO PRINT adalah memberitahu ke sistem bahwa report akan dilihat di kertas melalui printer

Contoh 1:

**REPORT FORM RTAMU PREVIEW**

Perintah di atas adalah memanggil report yang namanya RTAMU yang ditampilkan ke layar.

Contoh 2:

**REPORT FORM RTAMU TO PRINT**

Perintah di atas adalah memanggil report yang namanya RTAMU yang ditampilkan ke kertas melalui printer.



Contoh 3:

**REPORT FORM RTAMU FOR KEL="L" PREVIEW**

Perintah di atas adalah memanggil report yang namanya RTAMU di mana data yang ditampilkan adalah semua tamu dengan jenis kelamin L (Laki-laki) yang ditampilkan ke layar.

Contoh 4:

**REPORT FORM RTAMU FOR Year(Tglcatat)=Thisform.text2.value PREVIEW**

Perintah di atas adalah memanggil report yang namanya RTAMU di mana data yang ditampilkan adalah semua tamu dengan tahun pencatatan sama dengan data yang diisikan pada Text2, yang ditampilkan ke layar.

## 2. Memanggil Label

Bentuk umum perintah untuk memanggil label adalah sebagai berikut:

**LABEL FORM <namalabel> [FOR <Ekspresi>] [PREVIEW/TO PRINT]**

LABEL FORM adalah perintah dasar yang harus ditulis seperti itu

<namalabel> adalah nama report yang akan ditampilkan

[FOR <Ekspresi>] adalah batasan untuk menampilkan label

PREVIEW adalah memberitahu ke sistem bahwa label akan dilihat di layar

TO PRINT adalah memberitahu ke sistem bahwa label akan dilihat di kertas melalui printer

Contoh 1:

**LABEL FORM LBTAMU PREVIEW**

Perintah di atas adalah memanggil report yang namanya LBTAMU yang ditampilkan ke layar.

Contoh 2:

**LABEL FORM LBTAMU TO PRINT**

Perintah di atas adalah memanggil label yang namanya LBTAMU yang ditampilkan ke kertas melalui printer.

Contoh 3:

**LABEL FORM LBTAMU FOR KBANGSA="WNA" PREVIEW**

Perintah di atas adalah memanggil label yang namanya LBTAMU di mana data yang ditampilkan adalah semua tamu dengan kebangsaan WNA yang ditampilkan ke layar.

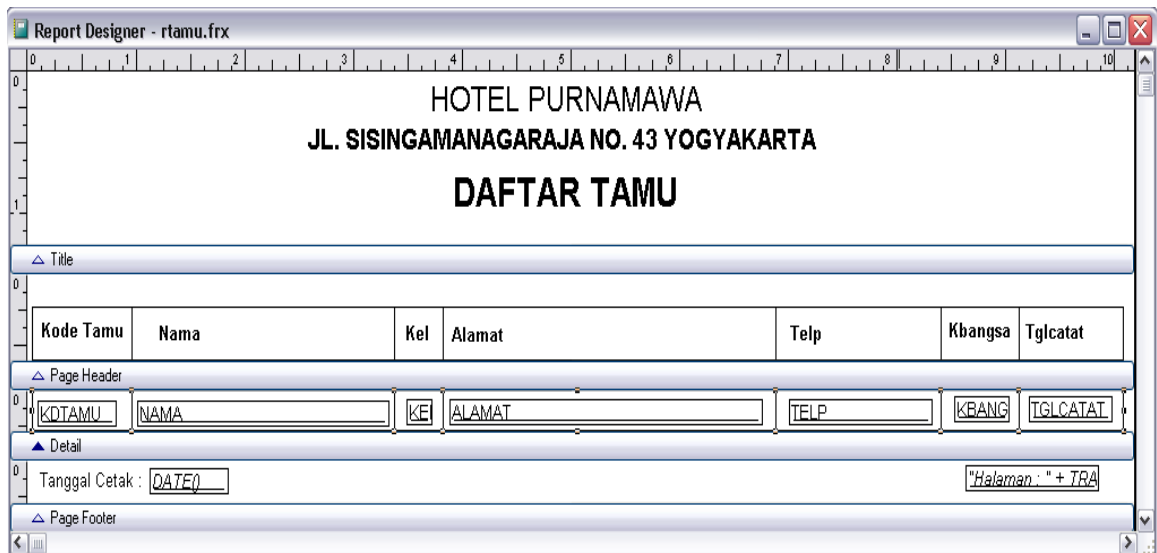
### 10.3 Praktek

Kegiatan praktek pada kesempatan ini adalah untuk membuat laporan secara standar, laporan mengelompok, dan laporan untuk analisis. Untuk bahan pembuatan laporan pada kesempatan ini akan digunakan tabel tamu, yang akan disajikan dengan berbagai variasi informasi.

#### 2. Membuat Laporan Standar (Laporan Tamu)

Laporan standar disini maksudnya adalah bentuk laporan adalah biasa (apa adanya), di mana informasi yang disajikan sama dengan data yang masuk

- a. Dari Project Manajer, klik tab **Documents**
- b. Klik pilihan **Reports**
- c. Klik tombol **New**
- d. Klik tombol Report **Wizard**
- e. Klik pilihan Report Wizard, lanjutkan mengklik tombol **OK**
- f. Saat ditampilkan kotak dialog, jika tabel-tabel belum ada, maka cari dengan mengklik tombol titik tiga. Tetapi jika tabel-tabel sudah ditampilkan, maka pilih/klik tabel tamu (dtamu).
- g. Ambil semua field (klik tombol panah dua), lanjutkan dengan mengklik tombol **Next**.
- h. Abaikan saja proses untuk Group Record dengan mengklik tombol **Next**.
- i. Tentukan jenis report (style), pilih yang Executive saja dan lanjutkan mengklik tombol **Next**.
- j. Tentukan orientasi report (karena data cukup banyak pilih landscape), dan lanjutkan mengklik tombol **Next**.
- k. Tetapkan apakah data akan diurutkan menurut field tertentu. Jika diinginkan, maka klik field yang akan dijadikan acuan pengurutan lalu klik tombol Add, lalu tetapkan juga metode pengurutan (Ascending atau Descending) kemudian klik tombol Next. Jika tidak diurutkan maka lanjutkan saja mengklik tombol **Next**.
- l. Klik tombol **Finish**.
- m. Ketikkan nama report yang dibuat (misal RTAMU) dan akhiri dengan mengklik tombol **Save**.
- n. Saat berada di Project Manajer, cari report yang sudah dihasilkan (diklik), lalu klik tombol **Modify** untuk melakukan penyuntingan.
- o. Sebelum melakukan perbaikan terhadap report yang dibuat, atur dulu lembar desain report dengan cara tampilkan menu Format, pilih menu Set Grid Scale, dan rubahlah nilai Harizontal spacing dan Verical spacing dari 12 pixel menjadi 1 pixel. Hal ini dilakukan agar saat mengatur objek atau komponen lebih leluasa.
- p. Ubahlah report sedemikian rupa sehingga susunannya tampak seperti Gambar 10.10. Dalam hal ini dapat ditambahkan komponen Label untuk membuat judul-judul atau keterangan. Selain itu juga dapat ditambahkan komponen **Rectangle** atau **Line**, dan lain sebagainya. Kunci untuk menghasilkan report yang baik adalah kesabaran dan nilai seni.



Gambar 10.10 Tampilan Desain Laporan Standar

3. Membuat Laporan Mengelompok (Laporan Tamu)  
Laporan mengelompok di sini maksudnya adalah bentuk laporan dikelompokkan berdasarkan data tertentu. Misalkan data tamu akan ditampilkan dengan mengelompok berdasarkan jenis kelamin, berdasarkan kebangsaan, berdasarkan tanggal pencatatan, berdasarkan tahun pencatatan, dan lain sebagainya.
  - a. Contoh berikut adalah laporan yang dikelompokkan berdasarkan kebangsaan
    - 1) Dari Project Manager, klik tab **Documents**
    - 2) Klik pilihan **Reports**
    - 3) Klik tombol **New**
    - 4) Klik tombol **Report Wizard**
    - 5) Klik pilihan Report Wizard, lanjutkan mengklik tombol **OK**
    - 6) Saat ditampilkan kotak dialog, jika tabel-tabel belum ada, maka cari dengan mengklik tombol titik tiga. Tetapi jika tabel-tabel sudah ditampilkan, maka pilih/klik tabel tamu (dtamu).
    - 7) Ambil semua field, kecuali field kbangsa, lanjutkan dengan mengklik tombol Next
    - 8) Pilih field kbangsa pada kotak dialog proses untuk Group Record, lanjutkan dengan mengklik tombol Next
    - 9) Tentukan jenis report (style), pilih yang Executive saja dan lanjutkan mengklik tombol Next.
    - 10) Tentukan orientasi report (karena data cukup banyak pilih landscape), dan lanjutkan mengklik tombol Next.
    - 11) Tetapkan field kbangsa sebagai dasar pengurutan dengan cara klik field kbangsa lalu klik tombol Add, metode pengurutan yang dipilih adalah Ascending kemudian klik tombol Next.
    - 12) Klik tombol Finish
    - 13) Ketikkan nama report yang dibuat (misal RTAMU1) dan akhiri dengan mengklik tombol Save.
    - 14) Saat berada di Project Manager, cari report yang sudah dihasilkan (diklik), lalu klik tombol Modify untuk melakukan penyuntingan.
    - 15) Sebelum melakukan perbaikan terhadap report yang dibuat, atur dulu lembar desain report dengan cara tampilkan menu Format, pilih menu Set Grid Scale, dan rubahlah

nilai Horizontal spacing dan Vertical spacing dari 12 pixel menjadi 1 pixel. Hal ini dilakukan agar saat mengatur objek atau komponen lebih leluasa.

- 16) Ubahlah report sedemikian rupa sehingga susunannya tampak seperti Gambar 10.11. Dalam hal ini dapat ditambahkan komponen Label untuk membuat judul-judul atau keterangan. Selain itu juga dapat ditambahkan komponen Rectangle atau Line, dan lain sebagainya. Kunci untuk menghasilkan report yang baik adalah kesabaran dan nilai seni.

Report Designer - rtamu1.frx

HOTEL PURNAMAWA  
JL. SISINGAMANAGARAJA NO. 43 YOGYAKARTA  
DAFTAR TAMU

△ Title

△ Page Header

Kebangsaan : kbangsa

Kode Tamu	Nama	Kel	Alamat	Telp	Tglcatat
<u>KODETAMU</u>	<u>NAMA</u>	<u>KE</u>	<u>ALAMAT</u>	<u>TELP</u>	<u>TGLCATAT</u>

△ Group Header 1:kbangsa

△ Detail

△ Group Footer 1:kbangsa

Tanggal Cetak : DATE() "Halaman : " + TRA

△ Page Footer

Gambar 10.11 Tampilan Desain Laporan Dikelompokkan Menurut Kebangsaan

b. Contoh berikut adalah laporan yang dikelompokkan berdasarkan tahun pencatatan

- 1) Dari Project Manager, klik tab **Documents**
- 2) Klik pilihan **Reports**
- 3) Klik tombol **New**
- 4) Klik tombol Report **Wizard**
- 5) Klik pilihan Report Wizard, lanjutkan mengklik tombol **OK**
- 6) Saat ditampilkan kotak dialog, jika tabel-tabel belum ada, maka cari dengan mengklik tombol titik tiga. Tetapi jika tabel-tabel sudah ditampilkan, maka pilih/klik tabel tamu (dttamu).
- 7) Ambil semua field dengan cara mengklik tombol panah dua, lanjutkan dengan mengklik tombol **Next**.
- 8) Abaikan proses pengelompokkan pada kotak dialog proses untuk Group Record (karena field tahun tidak ada, yang ada hanya tanggalcatat), lanjutkan saja dengan mengklik tombol **Next**.
- 9) Tentukan jenis report (style), pilih yang **Executive** saja dan lanjutkan mengklik tombol **Next**.
- 10) Tentukan orientasi report (karena data cukup banyak pilih landscape), dan lanjutkan mengklik tombol **Next**.
- 11) Tetapkan field tglcatat sebagai dasar pengurutan dengan cara klik field tglcatat lalu klik tombol Add, metode pengurutan yang dipilih adalah Ascending kemudian klik tombol **Next**.
- 12) Klik tombol **Finish**.

- 13) Ketikkan nama report yang dibuat (misal RTAMU2) dan akhiri dengan mengklik tombol **Save**.
- 14) Saat berada di Project Manager, cari report yang sudah dihasilkan (diklik), lalu klik tombol Modify untuk melakukan penyuntingan.
- 15) Sebelum melakukan perbaikan terhadap report yang dibuat, atur dulu lembar desain report dengan cara tampilkan menu Format, pilih menu Set Grid Scale, dan rubahlah nilai Horizontal spacing dan Vertical spacing dari 12 pixel menjadi 1 pixel. Hal ini dilakukan agar saat mengatur objek atau komponen lebih leluasa.
- 16) Buatlah pengelompokan data dengan cara menampilkan menu Report, pilih Data Grouping, klik tombol Add, tulis ekspresi pengelompokan : Year(tglcatat). Setelah selesai klik tombol OK. Untuk kembali ke desain report klik tombol OK.
- 17) Atur dan rancanglah judul untuk kelompok (Group Header)
- 18) Ubahlah report sedemikian rupa sehingga susunannya tampak seperti Gambar 10.12. Dalam hal ini dapat ditambahkan komponen Label untuk membuat judul-judul atau keterangan. Selain itu juga dapat ditambahkan komponen Rectangle atau Line, dan lain sebagainya. Kunci untuk menghasilkan report yang baik adalah kesabaran dan nilai seni.

Report Designer - rtamu3.frx

HOTEL PURNAMAWA  
JL. SISINGAMANAGARAJA NO. 43 YOGYAKARTA  
**DAFTAR TAMU**

△ Title

△ Page Header

Tahun : Year(Tglcatat)

Kode Tamu	Nama	Kel	Alamat	Telp	Kbangsa	Tglcatat
KDTAMU	NAMA	KEI	ALAMAT	TELP	KBANG	TGLCATAT

△ Group Header 1:year(tglcatat)

△ Detail

△ Group Footer 1:year(tglcatat)

Tanggal Cetak : DATE()

Page Footer

"Halaman : " + TRA

Gambar 10.12 Tampilan Desain Laporan Dikelompokkan Menurut Tahun

#### 4. Membuat Laporan untuk Analisis (Laporan Tamu)

Laporan untuk analisis disini maksudnya adalah bentuk laporan yang disajikan sedemikian rupa dengan suatu fungsi tertentu sehingga informasi tampilkan berdasarkan cara melihat informasi. Biasanya laporan jenis ini berupa laporan dalam bentuk rekapitulasi atau dalam bentuk statistik.

Misalkan bentuk laporan yang akan dibuat seperti ilustrasi berikut ini:

Tahun	Jumlah Menurut Kebangsaan		Jumlah Semua
	WNI	WNA	

Apa bisa menghasilkan laporan seperti di atas? Sementara field tahun tidak ada, lalu bagaimana cara menghitung cacah masing-masing data?

Dengan menggunakan proses pengelompokan data dan fungsi-fungsi laporan seperti di atas dapat dihasilkan walaupun ada kendala seperti yang diungkapkan.

- 1) Dari Project Manajer, klik tab **Documents**
- 2) Klik pilihan **Reports**
- 3) Klik tombol **New**
- 4) Klik tombol Report **Wizard**
- 5) Klik pilihan Report **Wizard**, lanjutkan mengklik tombol **OK**
- 6) Saat ditampilkan kotak dialog, jika tabel-tabel belum ada, maka cari dengan mengklik tombol titik tiga. Tetapi jika tabel-tabel sudah ditampilkan, maka pilih/klik tabel tamu (dttamu).
- 7) Ambil field tglcatat, field kbangsa, field kdtamu, lanjutkan dengan mengklik tombol **Next**.
- 8) Abaikan proses pengelompokkan pada kotak dialog proses untuk Group Record (karena field tahun tidak ada, yang ada hanya tanggalcatat), lanjutkan saja dengan mengklik tombol **Next**.
- 9) Tentukan jenis report (*style*), pilih yang Executive saja dan lanjutkan mengklik tombol **Next**.
- 10) Tentukan orientasi report (karena data hanya sedikit pilih potrait), dan lanjutkan mengklik tombol **Next**.
- 11) Tetapkan field tglcatat sebagai dasar pengurutan dengan cara klik field tglcatat lalu klik tombol Add, metode pengurutan yang dipilih adalah Ascending kemudian klik tombol **Next**.
- 12) Klik tombol **Finish**.
- 13) Ketikkan nama report yang dibuat (misal RTAMU3) dan akhiri dengan mengklik tombol **Save**.
- 14) Saat berada di Project Manajer, cari report yang sudah dihasilkan (diklik), lalu klik tombol **Modify** untuk melakukan penyuntingan.
- 15) Sebelum melakukan perbaikan terhadap report yang dibuat, atur dulu lembar desain report dengan cara tampilkan menu Format, pilih menu Set Grid Scale, dan rubahlah nilai Harizontal spacing dan Verical spacing dari 12 pixel menjadi 1 pixel. Hal ini dilakukan agar saat mengatur objek atau komponen lebih leluasa.
- 16) Buatlah pengelompokkan data dengan cara menampilkan menu Report, pilih Data Grouping, klik tombol Add, tulis ekspresi pengelompokkan : Year(tglcatat). Setelah selesai klik tombol **OK**. Untuk kembali ke desain report klik tombol **OK**.
- 17) Atur dan rancanglah judul untuk kelompok (Group Header)
- 18) Pindahkan field-field ke bagian Group Footer.
- 19) Buatlah fungsi **IIF** pada field kbangsa dan berikan proses perhitungan statistik (**SUM**)
- 20) Lakukan proses perhitungan statistik untuk field kdtamu (**COUNT**)
- 21) Ubahlah report sedemikian rupa sehingga susunannya tampak seperti Gambar 10.13. Dalam hal ini dapat ditambahkan komponen Label untuk membuat judul-judul atau keterangan. Selain itu juga dapat ditambahkan komponen Rectangle atau Line, dan lain sebagainya. Kunci untuk menghasilkan report yang baik adalah kesabaran dan nilai seni.



Gambar 10.13 Tampilan Desain Laporan Untuk Analisis (Rekapitulasi)

# Bab 11

## Laporan Tingkat Lanjut dan Form Laporan

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami dan menerapkan proses pembuatan laporan lebih lanjut dan pembuatan form pengendalian untuk laporan.

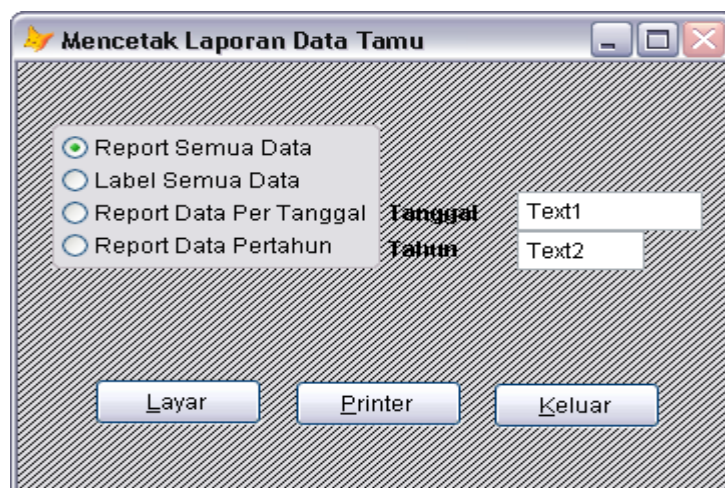
Pada pertemuan ini akan dibahas mengenai proses pembuatan laporan dengan banyak tabel, dan membuat *form* perantara sebelum laporan ditampilkan, dan cara memanggil laporan.

### 11.1 Laporan Dengan Banyak Tabel (lebih dari satu tabel)

Pada pembahasan bab sebelumnya telah dibahas dan dicobakan proses pembuatan laporan, namun laporan yang dihasilkan masih menggunakan satu tabel. Pertanyaan yang muncul adalah, **bagaimana kalau laporan yang akan dihasilkan membutuhkan sumber data lebih dari satu tabel?** Hal ini terjadi untuk menghasilkan laporan (informasi) untuk file-file transaksi. Secara prinsip proses pembuatannya tetap sama. Gunakan tabel utama yang diperlukan dalam laporan tersebut, kemudian hubungkan dengan tabel-tabel lain yang datanya terkait dengan tabel utama. Pada pemrograman basis data, hal ini sangat banyak ditemui dan memang proses seperti inilah yang dibutuhkan untuk menghasilkan informasi yang bermanfaat.

### 11.2 Fom Laporan (*Report Form*)

Form laporan diperlukan untuk memberi kesempatan kepada pemakai agar dapat memilih laporan yang akan ditampilkan (yang diinginkan), dari beberapa alternatif yang disediakan. Misalkan *report* mengenai tamu akan dibuat laporan seperti Gambar 11.1.



Mencetak Laporan Data Tamu

☒ Report Semua Data  
☐ Label Semua Data  
☐ Report Data Per Tanggal  
☐ Report Data Per Tahun

Tanggal: Text1  
Tahun: Text2

Layar Printer Keluar

Gambar 11.1 Tampilan Form Laporan



## 11.3 Praktek

Kegiatan praktek pada kesempatan ini adalah untuk membuat laporan yang menggunakan lebih dari satu tabel. Selain itu pada kegiatan praktek ini juga akan dicobakan untuk membuat form laporan, sehingga laporan bisa disajikan dengan berbagai pilihan.

### 5. Membuat Laporan Lebih dari Satu Tabel

Laporan model ini adalah laporan yang kompleks, di mana data yang digunakan bisa dua atau lebih. Sebagai contoh kasus 1: akan dibuat laporan mengenai daftar kamar. Jika dilihat dari kebutuhan tabel untuk pengolahan datanya, akan ada dua tabel yang diperlukan yaitu tabel kamar dan tabel jeniskm. Contoh kasus 2: akan dibuat laporan mengenai daftar tamu yang menginap. Jika memperhatikan kebutuhan tabel untuk pengolahan datanya, akan ada sebanyak lima tabel yaitu tabel inap, tabel dtamu, tabel petugas, tabel kamar, dan tabel jeniskm. Bagaiman langkahnya dan bagaimana mengelola tabelnya?

Langkah untuk membuat hampir sama dengan proses pembuatan laporan yang lain. Hal yang perlu diperhatikan adalah tabel mana yang akan dijadikan acuan dalam pembuatannya. Untuk kasus pertama, gunakan tabel kamar sebagai acuan untuk membuatnya karena tabel kamar sebagai tabel anak. Untuk kasus kedua, gunakan tabel inap sebagai acuan untuk membuatnya karena tabel inap merupakan tabel anak atau tabel transaksi.

Berikut ini adalah langkah untuk mewujudkan kasus kedua dengan catatan untuk lebih mudah sebagai contoh. Sedangkan kasus kedua dapat dijadikan sebagai latihan jika contoh kasus pertama dipahami.

- a. Dari Project Manajer, klik tab **Documents**
- b. Klik pilihan **Reports**
- c. Klik tombol **New**
- d. Klik tombol **Report Wizard**
- e. Klik pilihan **Report Wizard**, lanjutkan mengklik tombol **OK**
- f. Saat ditampilkan kotak dialog, jika tabel-tabel belum ada, maka cari dengan mengklik tombol titik tiga. Tetapi jika tabel-tabel sudah ditampilkan, maka pilih/klik tabel kamar.
- g. Ambil semua field dengan cara mengklik tombol panah dua, lanjutkan dengan mengklik tombol **Next**.
- h. Abaikan proses pengelompokkan pada kotak dialog proses untuk Group Record, lanjutkan saja dengan mengklik tombol **Next**.
- i. Tentukan jenis report (style), pilih yang **Executive** saja dan lanjutkan mengklik tombol **Next**.
- j. Tentukan orientasi report (karena data tidak banyak pilih **potrait**), dan lanjutkan mengklik tombol **Next**.
- k. Tetapkan field kodejk sebagai dasar pengurutan dengan cara klik field kodejk lalu klik tombol **Add**, metode pengurutan yang dipilih adalah **Ascending** kemudian klik tombol **Next**.
- l. Klik tombol **Finish**.
- m. Ketikkan nama report yang dibuat (misal RKAMAR) dan akhiri dengan mengklik tombol **Save**.
- n. Saat berada di Project Manajer, cari report yang sudah dihasilkan (diklik), lalu klik tombol **Modify** untuk melakukan penyuntingan.
- o. Sebelum melakukan perbaikan terhadap report yang dibuat, atur dulu lembar desain report dengan cara tampilkan menu Format, pilih menu **Set Grid Scale**, dan rubahlah nilai Harizontal spacing dan Verical spacing dari 12 pixel menjadi 1 pixel. Hal ini dilakukan agar saat mengatur objek atau komponen lebih leluasa.
- p. Tampilkan DataEnvironment dengan cara klik kanan di tempat yang kosong pada desain report, lalu pilih **DataEnvironment**. Klik kanan di dalam area

DataEnvironment lalu pilih **Add**. Pilih tabel jeniskm dan lanjutkan mengklik tombol Add.

- q. Atur relasi antar tabel, jika secara otomatis sudah ditampilkan, maka dapat dihapus dulu untuk menghindari kesalahan dalam relasi. Untuk melakukan relasi caranya adalah **field** pada tabel anak yang sama dengan tabel induk diarahkan ke bagian **index** dari tabel induk.
- r. Ubahlah report sedemikian rupa sehingga susunannya tampak seperti Gambar 11.2. Dalam hal ini dapat ditambahkan komponen Label untuk membuat judul-judul atau keterangan. Selain itu, juga dapat ditambahkan komponen **Rectangle** atau **Line**, dan lain sebagainya. Kunci untuk menghasilkan report yang baik adalah kesabaran dan nilai seni.

The screenshot shows the 'Report Designer - rkamar.frx' window. The report design is as follows:

- Title Section:** Contains the text 'HOTEL PURNAMA WA', 'JL. SISINGAMANAGARAJA NO. 43 YOGYAKARTA', and 'DAFTAR KAMAR'.
- Table:** A table with 5 columns: 'No. Kamar', 'Status Kamar', 'Kode', 'Nama Jenis Kamar', and 'Tarif Dasar'.
- Page Header:** Contains input fields for 'NOKAM', 'STA', 'KODEJK', 'namajk', and 'tarifds'.
- Detail:** An empty section for data rows.
- Page Footer:** An empty section for footer information.

Gambar 11.2 Tampilan Desain Laporan Kamar

**Catatan:**

Proses pembuatan laporan dengan sumber data lebih dari satu tabel ternyata juga tidak susah untuk membuatnya. Hanya dibutuhkan ketekunan dan jam terbang untuk memperlancar mekanisme pembuatan laporan.

Sebagai bahan latihan, silahkan membuat laporan untuk menghasilkan informasi mengenai tamu yang menginap. Sebagai panduan, bahwa tabel utamanya adalah tabel Inap, kemudian hubungkan dengan tabel-tabel lain yang terkait pada saat mengaturnya di **DataEnvironment**.

**6. Membuat Form Laporan**

Dari beberapa praktek yang sudah dicobakan, terciptalah beberapa report. Laporan-laporan yang sejenis dapat disajikan menjadi satu kesatuan. Misalkan laporan yang berhubungan dengan data tamu, dapat disajikan dalam satu form yang dapat dipilih oleh pengguna sesuai kebutuhan. Dari pekerjaan praktek untuk membuat *report* yang berhubungan dengan tamu ada 4 variasi. Untuk itu buatlah *form* untuk menawarkan laporan sebagai berikut:

- a. Dari Project Manajer, klik tab **Documents**
- b. Klik pilihan **Forms**
- c. Klik tombol **New**
- d. Klik pilihan **New Form**
- e. Ubahlah **Caption** dari Form menjadi : Sistem Informasi Perhotelan.
- f. Gunakanlah komponen-komponen sehingga tampilan form menjadi seperti pada Gambar 11.13.

- Ambil komponen Label, lalu letakkan di bagian atas area form. Atur properties Caption dan ketik : LAPORAN TAMU. Atur juga Aligment menjadi Center. Selain itu diatur juga FontBold mejadi True, FontSize menjadi 18
- Ambil komponen Optiongroup, lalu letakkan dibagian kiri area form. Atur pilihan yang akan ditampilkan dengan cara klik kanan, lalu pilih Builder. Isian Number of Column dirubah menjad 4 karena pilihan yang akan ditampilkan sebanyak 4 pilihan. Lalu ketikkan tulisan masing-maing pilihan pada bagian bawah kolom Caption. Setelah itu atur juga layoutnya dengan merubah isian pada Spacing between button menjadi 20.
- Ambil komponen Label, dan letakkan di samping kanan dari Optiongroup1 sejajar pilihan kedua. Ubah Caption menjadi Kebangsaan.
- Ambil komponen Combo1, dan letakkan di samping kanan Label kedua.
- Ambil komponen Label, dan letakkan di samping kanan Optiongroup1 sejajar pilihan ketiga. Ubahlah Caption menjadi Tahun.
- Ambil komponen Text, dan letakkan di samping kanan dari Label ketiga
- Ambil tiga buah komponen Button berturut-turut dari kiri ke kanan. Untuk Button1 ubahlah Caption dengan \<Layar dan Name menjadi TblLayar. Untuk Button2 ubahlah Caption dengan \<Cetak dan Name menjadi TblCetak. Untuk Button3 ubahlah Caption dengan \<Keluar dan Name menjadi TblKeluar



Gambar 11.3 Tampilan Desain Form Cetak Laporan

- Masukkan tabel DTTAMU ke **DataEnvironment**
- Tulislah kode pada kondisi **Form Active**

Kode baris 1 sampai bari 3 adalah proses untuk mengisi komponen Combo1, dengan dua pilihan yaitu WNI dan WNA.



Kode baris 4 adalah mengisi Text1 dengan nilai tahun sekarang (diambil dari tanggal sistem komputer).

Kode mulai baris 5 dan seterusnya adalah proses untuk menguji pilihan yang ditetapkan oleh useer. Jika pilihan pertama (menampilkan laporan semua tamu), maka tampilan kebangsaan dan tampilan tahun akan dihilangkan (Visible=.f.).

Object:  form1 Procedure:  Activate

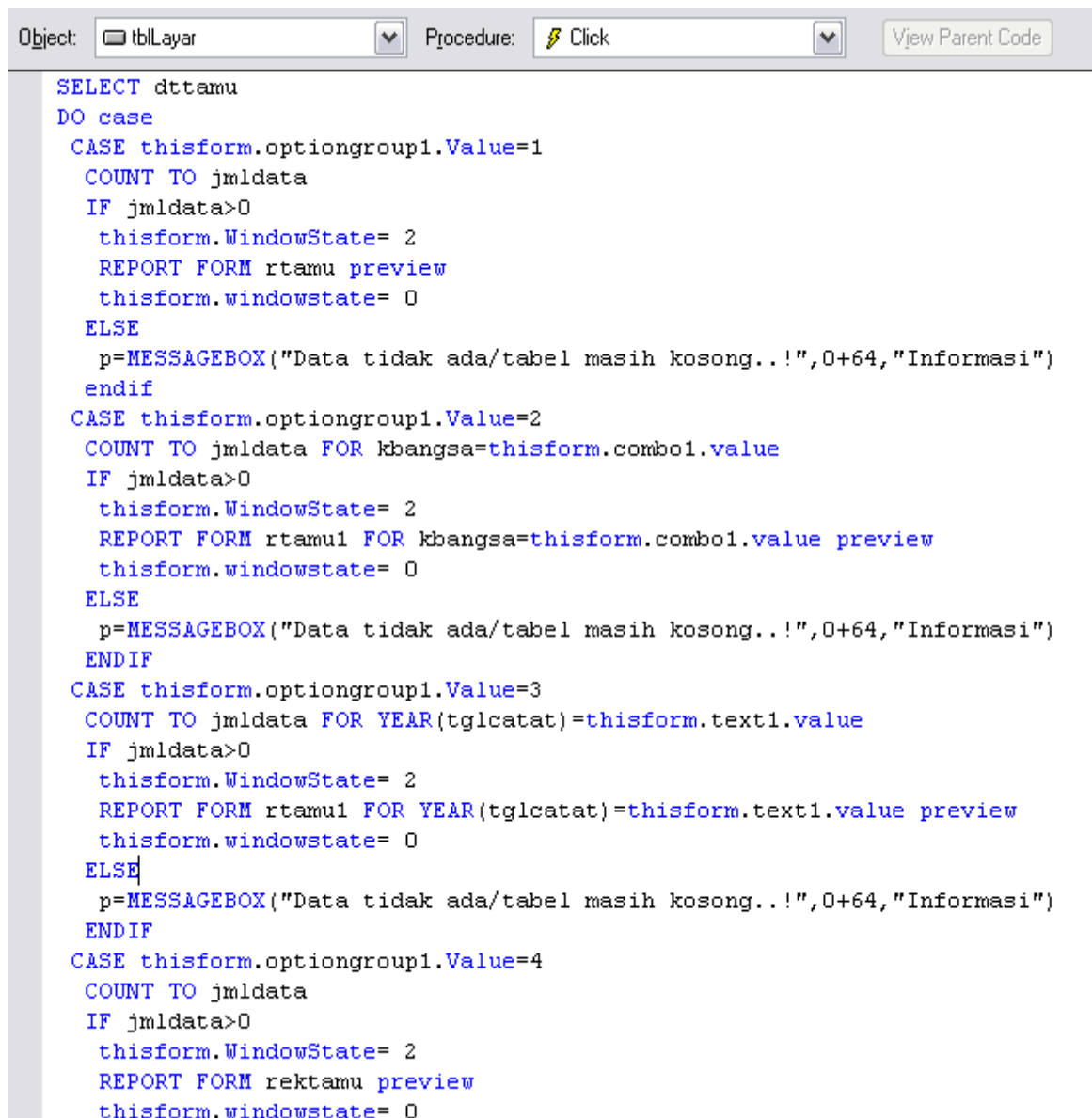
```
thisform.combo1.Clear
thisform.combo1.AddItem("WNI")
thisform.combo1.AddItem("WNA")
thisform.text1.Value=YEAR( DATE() )
DO case
CASE thisform.optiongroup1.Value=1
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
CASE thisform.optiongroup1.Value=2
    thisform.label2.Visible=.t.
    thisform.combo1.Visible=.t.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
CASE thisform.optiongroup1.Value=3
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.t.
    thisform.text1.Visible=.t.
CASE thisform.optiongroup1.Value=4
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
ENDCASE
```

- i. Tulislah kode pada saat Optiongroup1 di pilih (Click)

Object:  Optiongroup1 Procedure:  Click

```
DO case
CASE thisform.optiongroup1.Value=1
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
CASE thisform.optiongroup1.Value=2
    thisform.label2.Visible=.t.
    thisform.combo1.Visible=.t.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
CASE thisform.optiongroup1.Value=3
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.t.
    thisform.text1.Visible=.t.
CASE thisform.optiongroup1.Value=4
    thisform.label2.Visible=.f.
    thisform.combo1.Visible=.f.
    thisform.label3.Visible=.f.
    thisform.text1.Visible=.f.
ENDCASE
```

- j. Tulislah kode pada tombol Layar

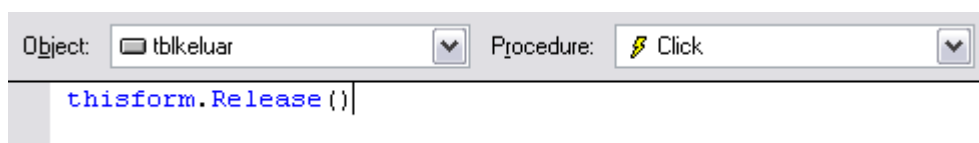


```
SELECT dttamu
DO case
CASE thisform.optiongroup1.Value=1
COUNT TO jmldata
IF jmldata>0
    thisform.WindowState= 2
    REPORT FORM rtamu preview
    thisform.windowstate= 0
ELSE
    p=MESSAGEBOX("Data tidak ada/tabel masih kosong...!",0+64,"Informasi")
endif
CASE thisform.optiongroup1.Value=2
COUNT TO jmldata FOR kbangsa=thisform.combo1.value
IF jmldata>0
    thisform.WindowState= 2
    REPORT FORM rtamu1 FOR kbangsa=thisform.combo1.value preview
    thisform.windowstate= 0
ELSE
    p=MESSAGEBOX("Data tidak ada/tabel masih kosong...!",0+64,"Informasi")
ENDIF
CASE thisform.optiongroup1.Value=3
COUNT TO jmldata FOR YEAR(tglcatat)=thisform.text1.value
IF jmldata>0
    thisform.WindowState= 2
    REPORT FORM rtamu1 FOR YEAR(tglcatat)=thisform.text1.value preview
    thisform.windowstate= 0
ELSE
    p=MESSAGEBOX("Data tidak ada/tabel masih kosong...!",0+64,"Informasi")
ENDIF
CASE thisform.optiongroup1.Value=4
COUNT TO jmldata
IF jmldata>0
    thisform.WindowState= 2
    REPORT FORM rektamu preview
    thisform.windowstate= 0
```

- k. Tulislah kode pada tombol Cetak

Kode untuk tombol cetak hampir sama dengan kode pada tombol Layar. Perbedaannya adalah instruksi **Preview** pada tombol Layar diganti dengan **TO PRINT** pada tombol Cetak. Untuk itu copy saja lalu dilakukan perubahan.

- l. Tulislah kode pada tombol Keluar



```
thisform.Release()
```

# Bab 12

## Menu, Form Menu, Form Login, dan Program Utama

### Tujuan Instruksional Khusus:

Setelah mempelajari bab ini, mahasiswa diharapkan dapat memahami dan menerapkan proses pembuatan pengendalian aplikasi supaya dapat bekerja secara sistematis.

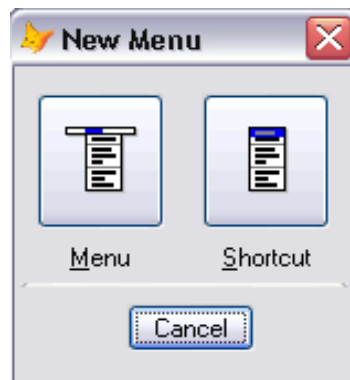
Pada pertemuan ini akan dibahas mengenai proses pembuatan desain menu, membuat *form* menu, *form login*, dan membuat program utama.

### 12.1 Membuat Desain Menu

Desain menu yang akan dibuat adalah desain menu untuk *pull down*. Bentuk dan cara kerja menu *pull down* adalah seperti menu bar pada aplikasi-aplikasi yang sudah banyak beredar. Saat pilihan menu diklik atau diaktifkan, akan ditampilkan sub menu atau daftar perintah yang ada pada pilihan tersebut.

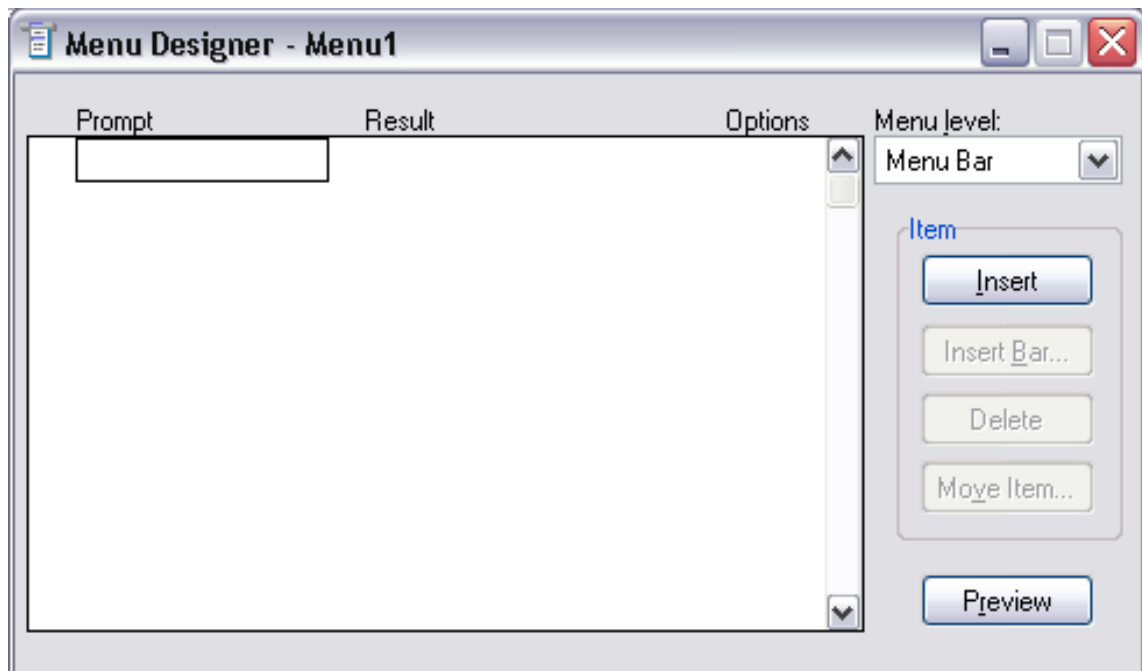
Untuk membuat desain menu, ikutilah langkah-langkah sebagai berikut:

1. Dari Project Manager, klik tab **Other**
2. Klik pilihan **Menus**
3. Klik tombol **New**, sehingga akan ditampilkan pilihan seperti Gambar 12.1



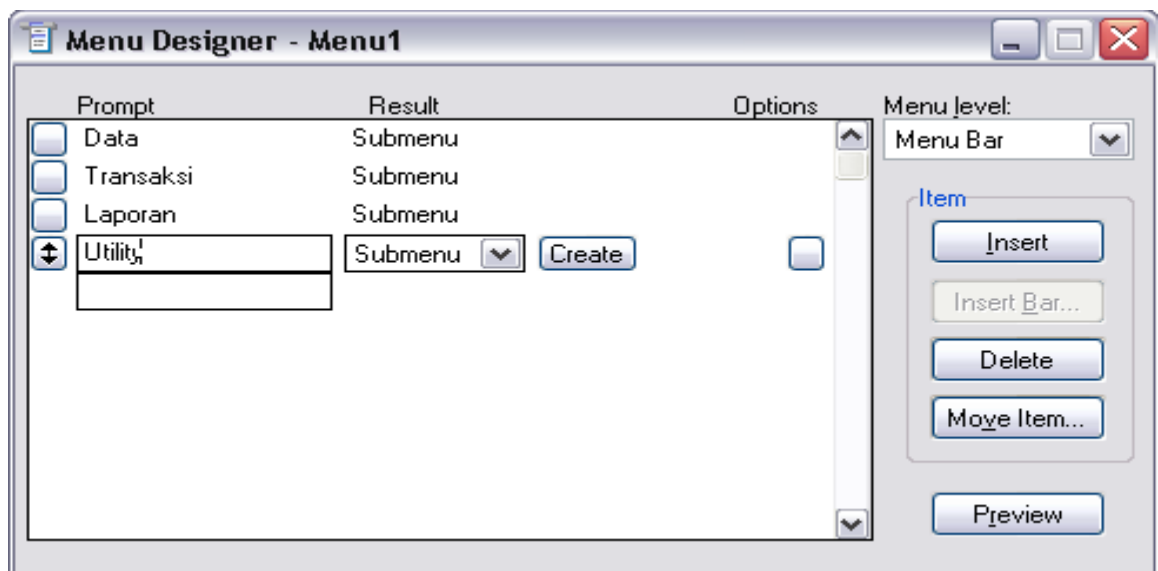
Gambar 12.1 Pilihan pembuatan desain menu

4. Klik pilihan **Menu** (pilihan sebelah kiri), sehingga akan ditampilkan kotak dialog untuk desain menu seperti terlihat pada Gambar 12.2



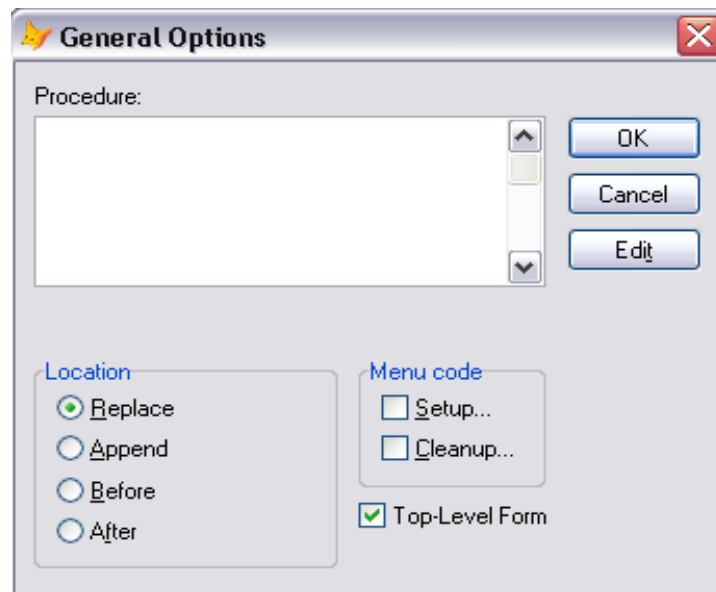
Gambar 12.2 Kotak dialog desain menu

- Ketuklah pilihan menu pada kolom Prompt, sehingga bentuk kotak dialog akan berubah seperti Gambar 12.3



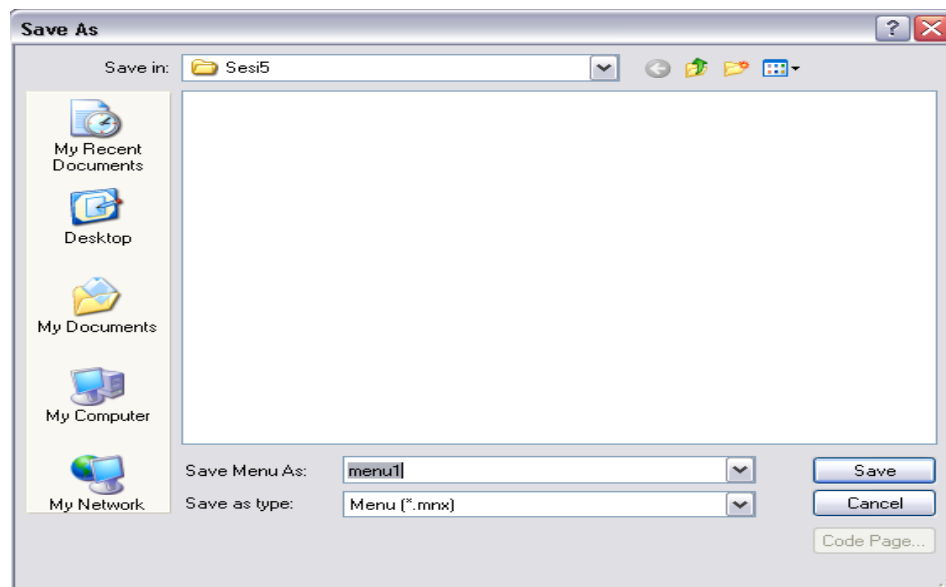
Gambar 12.3 Kotak dialog desain menu yang telah diisi

- Jika semua pilihan menu sudah selesai dimasukkan, maka berikutnya tampilkan menu **View**, lalu pilih **General Option**, kemudian tandai **checkbox Top Level Form**, dan akhiri dengan mengklik tombol **OK**, seperti terlihat pada Gambar 12.4



Gambar 12.4 Pengaturan general option

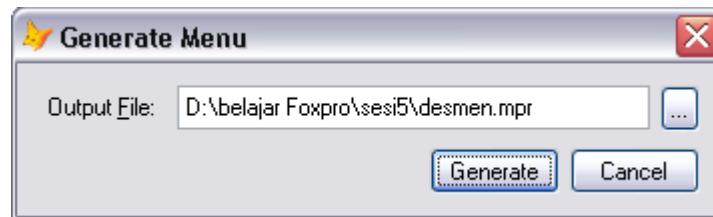
7. Simpan hasil rancangan menu dengan mengklik icon **Save** (gambar disket), sehingga akan ditampilkan kotak dialog penyimpanan *file* seperti Gambar 12.5.



Gambar 12.5 Kotak dialog penyimpanan menu

8. Isikan atau ketik nama *file* untuk menyimpan desain menu yang dibuat, misal desmen (boleh menggunakan nama yang ditawarkan yaitu menu1), dan akhiri dengan mengklik tombol **Save**.
9. Tampilkan menu yang namanya **Menu**
10. Pilih **Generate**, sehingga akan ditampilkan kotak dialog seperti Gambar 12.6





Gambar 12.6 Kotak dialog proses generate menu

11. Klik tombol **Generate**, jika sebelumnya telah melakukan generate, akan ditampilkan kotak dialog untuk konfirmasi. Jika baru pertama kali melakukan proses generate, maka proses akan dilakukan segera. Tunggu sampai proses selesai, dengan demikian desain menu telah berhasil dibuat.
12. Tutup kotak dialog desain menu untuk kembali ke **Project Manager**.

## 12.2 Membuat Form Menu

*Form* menu dibutuhkan untuk mengaitkan desain menu, sehingga menu bisa tampil lebih baik jika dibandingkan tanpa menggunakan *form*. Untuk mebuat form menu, langkah-langkahnya sama seperti membuat form biasa, sebagai berikut:

12. Klik Tab Page **Documents**
13. Klik pilihan **Form**
14. Klik tombol **New**
15. Pilihan **New Form**
16. Aturlah properties form antara lain :
  - **AutoCenter** dirubah menjadi **True**
  - **MDIForm** dirubah menjadi **True**
  - **MaxButton** dirubah menjadi **False**
  - **MinButton** dirubah menjadi **False**
  - **ShowTips** dirubah menjadi **True**
  - **ShowWindow** dirubah menjadi pilihan **2-As-Top-Level Form**
  - **Windowstate** dirubah menjadi pilihan **2-Maximized**
17. Silahkan tambah komponen-komponen lain pada form supaya tampilan lebih menarik, misalnya menampilkan tanggal, jam, gambar dan lain sebagainya.
18. Tulislah kode program yang akan mengaitkan desain menu dengan form menu. Perintah tersebut bisa ditempatkan pada kejadian Form-Active atau Form-Init.  
Bentuk umum perintah:  
**Do <Namamenu> with this,.T.**  
Penulisan nama menu harus lengkap dengan tipe filenya. Sebagai contoh, misal nama desain menu adalah desmen, maka untuk mengitkannya dengan form ditulis perintah sebagai berikut:  
**Do desmen.mpx with this, .t.**
19. Simpan Form
20. Coba jalankan

## 12.3 Membuat Form Login

*Form* login dibutuhkan untuk memberikan keamanan bagi sistem, sehingga tidak semua orang dapat menggunakan sistem. Tampilan form login akan disajikan sebelum menu utama ditampilkan. Metode yang digunakan dapat digunakan dua macam, yaitu dengan

metode form login yang statis atau form login dinamis. Form login statis adalah nama user dan password tetap atau sudah pasti. Sedangkan form login dinamis adalah **nama user** dan **password** terkait dengan *database* pengguna, sehingga akan berbeda untuk setiap pemakai program.

## 12.4 Membuat Program Utama

Program utama adalah sebuah rangkaian perintah yang digunakan untuk melakukan pengaturan agar aplikasi bisa menjadi sistem yang terintegrasi dan dapat menjadi aplikasi yang mandiri. Jika program utama tidak dibuat, maka aplikasi hanya bisa dijalankan bersama dengan perangkat lunak Visual FoxPro saja.

Beberapa perintah yang sering dicantumkan pada program utama antara lain:

- 1      **clear all**
- 2      **close all**
- 3      **set talk off**
- 4      **set bell off**
- 5      **set status off**
- 6      **set date italian**
- 7      **set century on**
- 8      **set delete on**
- 9      **clear**
- 10     **set sysmenu OFF**
- 11     **\_screen.visible=.f.**
- 12     **do form kunci**
- 13     **read events**
- 14     **quit**

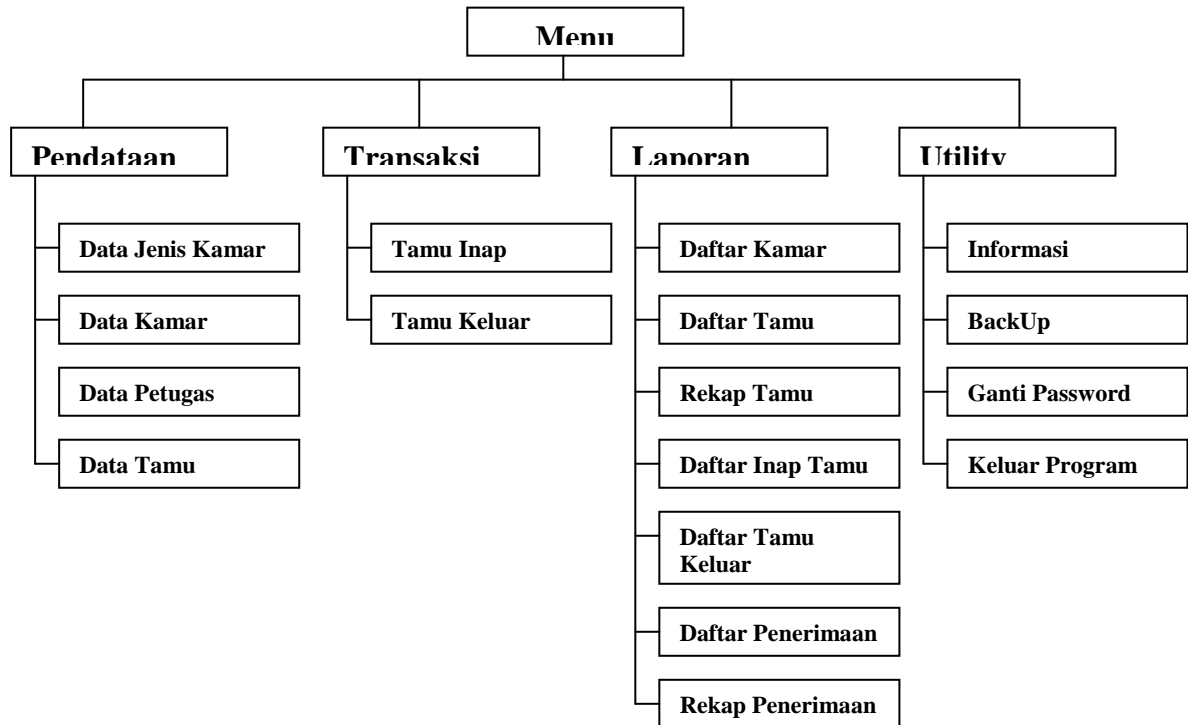
Perintah baris 1 berguna untuk membersihkan seluruh nilai termasuk nilai variabel memori. Perintah baris 2 berguna untuk menutup semua komponen (database, tabel, index). Perintah baris 3 untuk mematikan sifat interaktif sistem. Perintah baris 4 digunakan untuk mematikan bunyi speaker (bell). Perintah baris 5 digunakan untuk mematikan baris status. Perintah baris 6 digunakan untuk menetapkan format penulisan tanggal dengan metode Italia (sama seperti Indonesia yaitu tgl-bln-thn). Perintah baris 7 digunakan untuk menampilkan tahun dengan empat digit. Perintah baris 8 digunakan untuk mengaktifkan modus penghapusan (data yang sudah dihapus dengan perintah delete tidak diakses lagi). Perintah baris 9 digunakan untuk membersihkan layar. Perintah baris 10 digunakan untuk mematikan sistem menu default. Perintah baris 11 digunakan untuk mematikan layar default Visual FoxPro. Perintah baris 12 digunakan untuk menjalankan form login yang namanya kunci. Perintah baris 13 digunakan untuk menunggu sampai pemakai memberikan respon kepada program seperti mengklik mouse atau menentukan tombol lainnya. Perintah baris 14 digunakan untuk keluar dari sistem visual FoxPro.

## 12.5 Praktek

Kegiatan praktek pada kesempatan ini adalah untuk membuat menu utama dengan fasilitas yang telah disediakan yaitu dalam bentuk menu pulldown.

### 1. Membuat Desain Menu

Untuk mengimplementasikan pembuatan desain menu, berikut ini diberikan ilustrasi struktur menu yang akan dibuat sebagai berikut:



Gambar 12.7 Ilustrasi struktur menu

Keterangan:

Pilihan Pendataan, Transaksi, Laporan, dan Utility adalah pilihan utama. Sedangkan uraian dari masing-masing pilihan disebut sub menu.

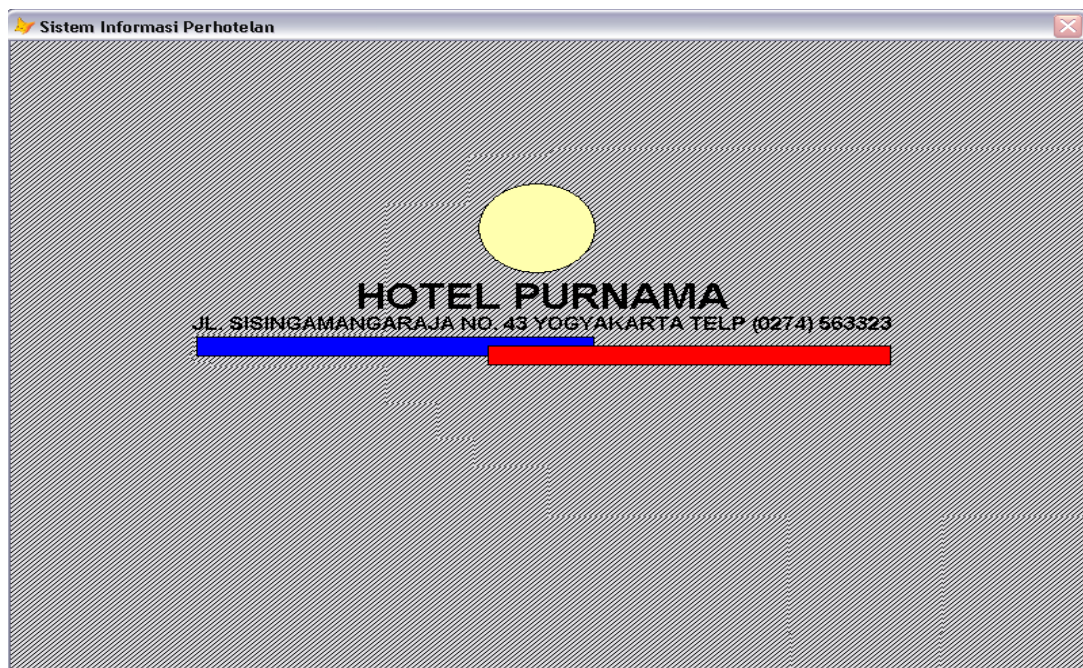
Lakukan langkah sebagai berikut:

- Dari posisi Project Manager, klik tab **Other**
- Klik pilihan **Menus**
- Klik tombol **New**
- Pilih **Menu**
- Ketik pilihan-pilihan utama yang sudah dijelaskan di atas.
- Ciptakan sub menu untuk masing-masing pilihan utama dengan mengklik tombol **Create**
- Ketik pilihan untuk sub menu yang bersangkutan. Untuk memberi garis pembatas diantara pilihan dapat diketik \- pada suatu baris.
- Ketik operasi atau perintah yang akan dilaksanakan saat memilih sub menu tersebut dengan cara mengetik atau mengisi pada kolom **Result** dengan **Command**, kemudian ditulis perintah pada baris bagian sebelah kanan. Misalkan perintah yang akan dikerjakan adalah menjalankan form yang namanya fkamar, maka perintah yang diketikkan adalah: **Do Form fkamar**
- Kembali ke pilihan utama dengan mengklik bagian Menu Level dan diarahkan ke pilihan Menu Bar
- Ulangi langkah h sampai langkah i, untuk membuat pilihan yang lain sampai selesai.
- Tampilkan menu View, lalu dicentang fasilitas **Top-Level Form**, lalu klik tombol **OK**.
- Simpan desain meni ini dengan mengklik icon Save, saat ditampilkan kotak dialog, ketik nama untuk menyimpan desain menu, misal diberi nama **PILMEN**.
- Tampilkan menu yang nama **Menu**, pilih **Generate**. Jika ada pertanyaan jawab dengan **Yes**. Lanjutkan dengan mengklik tombol **Generate**. Jika muncul pertanyaan jawab **Yes**.
- Tutup kotak dialog pembuatan desain menu

## 2. Membuat Form Menu

Untuk membuat form menu, langkah-langkahnya sama seperti membuat *form* biasa, sebagai berikut:

1. Klik Tab Page **Documents**
2. Klik pilihan **Form**
3. Klik tombol **New**
4. Pilihan **New Form**
5. Aturlah properties form antara lain :
  - **AutoCenter** dirubah menjadi **True**
  - **MDIForm** dirubah menjadi **True**
  - **MaxButton** dirubah menjadi **False**
  - **MinButton** dirubah menjadi **False**
  - **ShowTips** dirubah menjadi **True**
  - **ShowWindow** dirubah menjadi pilihan **2-As-Top-Level Form**
  - **WindowState** dirubah menjadi pilihan **2-Maximized**
6. Silahkan rancang form seperti Gambar 12.8, jika perlu tambah komponen-komponen lain pada form supaya tampilan lebih menarik, misalnya menampilkan tanggal, jam, gambar dan lain sebagainya.



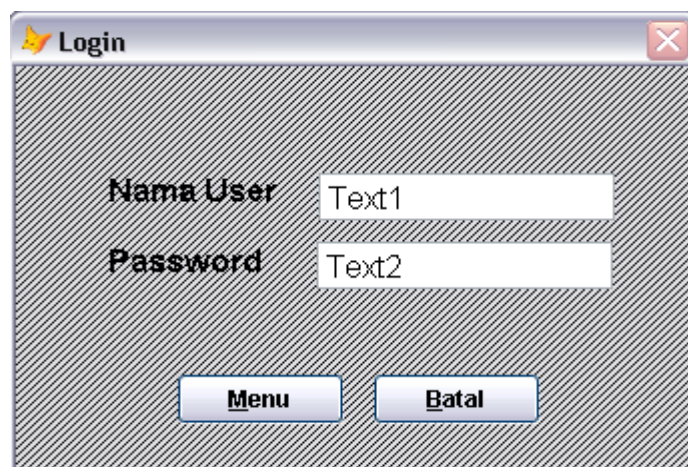
Gambar 12.8 Rancangan form menu

7. Tulislah kode program yang akan mengaitkan desain menu dengan form menu. Perintah tersebut bisa ditempatkan pada kejadian **Form-Active** atau Form-Init. Bentuk umum perintah:  
Do <Namamenu> with this,.T.  
Penulisan nama menu harus lengkap dengan tipe filenya. Sebagai contoh, misal nama desain menu adalah desmen, maka untuk mengitkannya dengan form ditulis perintah sebagai berikut:  
**Do desmen.mpx with this, .t.**
8. Simpan *Form*, dan beri nama **FMENU**

### 3. Membuat Form Login

Untuk membuat *form* login, langkah-langkahnya sama seperti membuat *form* biasa, sebagai berikut:

1. Klik Tab Page **Documents**
2. Klik pilihan **Form**
3. Klik tombol **New**
4. Pilihan **New Form**
5. Aturlah properties *form* antara lain :
  - **AutoCenter** dirubah menjadi **True**
  - **MDIForm** dirubah menjadi **True**
  - **MaxButton** dirubah menjadi **False**
  - **MinButton** dirubah menjadi **False**
  - **ShowTips** dirubah menjadi **True**
  - **ShowWindow** dirubah menjadi pilihan **2-As-Top-Level Form**
6. Silahkan rancang form seperti Gambar 12.9, jika perlu tambah komponen-komponen lain pada form supaya tampilan lebih menarik, misalnya gambar dan lain sebagainya.



Gambar 12.9 Rancangan form login

- Untuk label1 dan label2 yang perlu diganti :
- BackStyle diubah menjadi Transparent
  - Caption diubah menjadi Nama User
  - FontBold diubah menjadi True
  - FontSize diubah menjadi 12
- Untuk Text1 yang perlu diganti
- FontSize diubah menjadi 12
  - MaxLength diubah menjadi 30
- Untuk Text2 yang perlu diganti
- FontSize diubah menjadi 12
  - MaxLength diubah menjadi 6
  - PasswordChar diubah menjadi \*
- Untuk Command1 yang perlu diganti
- Caption diubah menjadi \<Menu
  - FontBold diubah menjadi True
  - Name diubah tblmenu
- Untuk Command2 yang perlu diganti
- Caption diubah menjadi \<Batal
  - FontBold diubah menjadi True
  - Name diubah tblbatal

7. Tulis kode pada tombol **Menu**, klik dua kali pada tombol **Menu** dan ketik perintah sebagai berikut:

Object:  Procedure:  [View Parent Code](#)

```
cek1=UPPER(thisform.text1.Value)
cek2=UPPER(thisform.text2.Value)
SELECT petugas
LOCATE FOR UPPER(namapet)=cek1 .and. UPPER(paswp)=cek2
IF FOUND()
    DO FORM fmenu
ELSE
    p=MESSAGEBOX("Nama user dan pasword tidak cocok, ulangi...",0+64,"Informasi")
    thisform.text1.SetFocus()
ENDIF
```

8. Tulis kode pada tombol **Batal**, klik dua kali pada tombol **Batal** dan ketik perintah sebagai berikut:

Object:  Procedure:

```
thisform.Release()
```

9. Simpan *form* dengan nama **FKUNCI**

#### 4. Membuat Program Utama

Untuk membuat program utama ikuti langkah sebagai berikut:

- Dari Project Manager, klik tab **Code**
- Klik pilihan **Programs**
- Klik tombol **New**
- Ketik kode program sebagai berikut :

```
clear all
set talk off
set date italian
set century on
set delete on
clear
set sysmenu OFF
__screen.visible=.f.
do form fkunci
read events
quit
```

Disesuaikan  
dengan nama form  
yang akan diakses  
pertamakali



- Simpan program dengan mengklik icon **Save**
- Ketik nama file untuk menyimpan, misal **putama**, klik tombol **Save**
- Tutup layar penulisan kode dengan mengklik tanda silang.

## DAFTAR PUSTAKA

- Antonovich, M., 1996, Special Edition Using Visual FoxPro 5, QUE.
- Choliludin, 2006, Aplikasi Laporan Penjualan dan Kartun MS Agent Dengan FoxPro 9.0, Elex Media Komputindo, Jakarta.
- Chayo, Y., 2007, Microsoft Visual FoxPro 9.0, Elex Media Komputindo, Jakarta.
- Hadi, M., 2006, Visual FoomPro 9.0 Untuk Orang Awam, Maxikoom, Palembang
- Indriyawan, E., 006, Pemrograman Database, Andi Offset, Yogyakarta.
- Supardi, Y., 2009, Sistem Populer Dengan Microsoft Visual FoxPro 9.0, Elex Media Komputindo, Jakarta.
- Supardi, Y., 2011, Semua Bisa Menjadi Programmer Visual FoxPro 9.0, Elex Media Komputindo, Jakarta.