

# Praktikum Pemrograman Clipper

Dwi Sakethi, [www.dwijim.wordpress.com](http://www.dwijim.wordpress.com)

17 Nopember 2010

## Daftar Isi

<b>1</b>	<b>Praktikum #1 - Membuat berkas basis data (.dbf)</b>	<b>1</b>
1.1	Program yang dibutuhkan . . . . .	1
1.2	Langkah-Langkah Pekerjaan . . . . .	1
<b>2</b>	<b>Praktikum #2 - Perintah <i>Dot Command</i></b>	<b>4</b>
2.1	Program yang dibutuhkan . . . . .	4
2.2	Masalah yang dipecahkan . . . . .	4
<b>3</b>	<b>Praktikum #3 - Membuat Program Clipper</b>	<b>6</b>
3.1	Program yang dibutuhkan . . . . .	6
3.2	Langkah-Langkah Pekerjaan . . . . .	6
3.3	Materi tambahan . . . . .	7
<b>4</b>	<b>Praktikum #4 - Membuat Program Pemasukan Data</b>	<b>8</b>
4.1	Bahan yang Dibutuhkan . . . . .	8
4.2	Materi Tambahan #1 . . . . .	8
4.3	Materi Tambahan #2 . . . . .	8
<b>5</b>	<b>Praktikum #5 - Program Pemasukan Data ke Basis Data</b>	<b>9</b>
5.1	Bahan yang Dibutuhkan . . . . .	9
5.2	Contoh Tampilan . . . . .	9
5.3	Materi Tambahan #1 . . . . .	9
<b>6</b>	<b>Praktikum #6 - Program Penghapusan Data di Basis Data</b>	<b>10</b>
6.1	Bahan yang Dibutuhkan . . . . .	10
6.2	Contoh Tampilan . . . . .	10
6.3	Materi Tambahan #1 . . . . .	10
<b>7</b>	<b>Praktikum #7 - Program Pemasukan Data dengan Validasi Biasa</b>	<b>11</b>
7.1	Bahan yang Dibutuhkan . . . . .	11

## Daftar Tabel

1	Atribut Tabel IPA2002.DBF . . . . .	2
2	Atribut Tabel PS.DBF . . . . .	2
3	Atribut Tabel SLTA.DBF . . . . .	2

## Daftar Gambar

1	Contoh Logo . . . . .	7
---	-----------------------	---

# 1 Praktikum #1 - Membuat berkas basis data (.dbf)

Praktikum #1 berisi materi tentang basis data dengan menggunakan xBase. Untuk melaksanakan materi ini mahasiswa sudah mendapat bahan kuliah tentang manajemen basis data dalam xBase khususnya tentang perintah-perintah *dot command* atau *dot prompt*.

## 1.1 Program yang dibutuhkan

Program yang dibutuhkan dalam praktikum ini tergantung sistem operasi yang digunakan. Jika sistem operasi menggunakan Windows XP maka program yang bisa digunakan adalah dBase III+. Jika menggunakan GNU Linux dengan *DOS Emulator*, maka bisa digunakan program FoxBase ataupun dBase III+. Program-program tersebut harus sudah ada di dalam sistem komputer dan mahasiswa mesti tahu persis, di mana letak program-program tersebut.

Jika di dalam sistem komputer yang dipakai belum ada perangkat-perangkat lunak yang dibutuhkan, maka perangkat lunak tersebut bisa diunduh di [www.dwijim.wordpress.com](http://www.dwijim.wordpress.com)

## 1.2 Langkah-Langkah Pekerjaan

Urut-urutan perintah yang dilakukan dalam pelaksanaan praktikum kali ini adalah sebagai berikut :

1. Jalankan program dBase III+ atau FoxBase. Cara menjalankan program ini berbeda-beda tergantung di mana meletakkan program dBase atau FoxBase. Untuk contoh perhatikan perintah berikut. Untuk menjalankan dBase, posisi direktori aktif harus berapa pada direktori dBase, jadi pindah direktori dulu, baru kemudian menjalankan dBase.

```
cd dbase [enter]    - pindah ke direktori dbase
dbase.exe [enter]   - menjalankan program dbase
                    Press the F1 key for HELP.
                    Type a command (or ASSIST) and press the ENTER key (?-+).
```

Kemudian dari posisi seperti ini, supaya lebih jelas tampilannya, berikan perintah :

```
set status on
.
Command Line      <D:>
                  Enter a dBASE III PLUS command.
```

Untuk penggunaan FoxBase caranya hampir sama, hanya mungkin berbeda direktori. Misalnya :

```
cd ca-52\bin      - pindah direktori tempat FoxBase
mfoxplus.exe     - menjalankan program FoxBase
```

Kemudian berikan perintah **set status on** juga.

2. Membuat berkas basis data (.dbf). Setelah program DBMS (dBase III+ atau FoxBase siap digunakan, selanjutnya adalah membuat basis data (.dbf) dengan perintah **create IPA2002**. Isi dari berkas basis data IPA2002.dbf seperti berikut :

Field	Field Name	Type	Width	Dec
1	NO_PKAB	Character	5	
2	RESI	Character	1	
3	PILIHAN_1	Character	6	
4	PILIHAN_2	Character	6	
5	NAMA_SISWA	Character	30	
6	NO_INDUK	Character	10	
7	JURUSAN	Character	1	
8	KODE_SMA	Character	8	
9	NAMA_SMA	Character	35	
10	KELAMIN	Character	1	

Tabel 1: Atribut Tabel IPA2002.DBF

Setelah semua *field* dibuat dan ditentukan tipe serta ukurannya, kemudian tekan tombol Ctrl-End untuk merekam dan selesai dari proses pembuatan satu basis data. Jika ada pertanyaan, "Apakah akan memasukkan data sekarang?", jawab dengan menekan tombol 'N' yang artinya tidak akan memasukkan data. Selain itu, dibuat juga berkas basis data PS.DBF. Dari posisi *dot prompt* ketikkan perintah **create PS**. Sedangkan struktur basis data PS seperti berikut :

Field	Field Name	Type	Width	Dec
1	KODE	Character	6	
2	KETERANGAN	Character	37	

Tabel 2: Atribut Tabel PS.DBF

Kemudian juga berkas SLTA.DBF untuk menyimpan data-data nama sekolah SMA.

Field	Field Name	Type	Width	Dec
1	KODE_SMA	Character	8	
2	NEGE_SMA	Character	1	
3	NAMA_SMA	Character	35	
5	JALAN	Character	35	
6	KOTA	Character	25	

Tabel 3: Atribut Tabel SLTA.DBF

3. Isi masing-masing tabel dengan data-data paling tidak 5 *record* untuk masing-masing tabel. Untuk mengisi data SMA (misalnya), perintahnya adalah :

```
use SMA      - membuka file database SMA.DBF
```

append - menambah data  
use PS  
append

## 2 Praktikum #2 - Perintah *Dot Command*

Pada praktikum ini, mahasiswa diarahkan untuk memahami perintah-perintah dalam DBMS. Perintah-perintah ini mirip dengan fungsi perintah-perintah SQL di dalam basis data modern seperti MySQL, Oracle, MS SQL Server dan sebagainya.

### 2.1 Program yang dibutuhkan

Program yang dibutuhkan sama dengan kebutuhan pada Praktikum #1. Selain itu ditambah juga dengan basis data yang sudah dibuat pada Praktikum #1. Akan lebih baik lagi, jika data yang tersedia sudah banyak, misalnya minimal 20 data untuk masing-masing basis data. Untuk contoh basis data yang banyak, data IPA2002 bisa diunduh di [www.dwijim.wordpress.com](http://www.dwijim.wordpress.com) (mudah-mudahan).

### 2.2 Masalah yang dipecahkan

Jawablah pertanyaan-pertanyaan berikut dan jawabannya dilakukan dengan perintah-perintah **dot command**, bukan dengan menjawab secara manual. Catatan : jika nama basis data berubah, maka harus diawali dengan perintah **use**. Misal : **use IPA2002**

1. Berapa jumlah data yang ada pada basis data IPA2002 ?  
**count to jumlah\_data**
2. Siapa yang ada pada data ke-5 dari basis data IPA2002 ?  
**go 5 - edit**
3. Tampilkan semua nama siswa yang ada  
**list nama\_siswa**
4. Menampilkan siswa-siswa yang berjenis kelamin perempuan ?  
**list nama\_siswa for kelamin=0**
5. Menampilkan siswa-siswa yang namanya diawali dengan huruf 'A'  
**list nama\_siswa for left(nama\_siswa,1)='A'**
6. Menampilkan siswa-siswa yang namanya mengandung kata 'WATI'  
**list nama\_siswa for 'WATI'\$upper(nama\_siswa)**
7. Menampilkan siswa-siswa yang namanya diakhiri kata 'WAHYUNI'  
**list nama\_siswa for right(rtrim(nama\_siswa),7)='WAHYUNI'**
8. Merubah nama siswa menjadi 'Dwi Sakethi' untuk peserta dengan No. PKAB-nya 101010  
**replace nama\_siswa with 'Dwi Sakethi' where no\_pkab='101010'**  
No. PKAB harus disesuaikan dengan data yang ada, jika data tidak ada yang sesuai maka tidak akan ada data yang diganti.
9. Menghapus data dengan nomor *record* 2-4  
**detele for recno()>=2 .and. recno()<=4**
10. Hitung berapa jumlah data yang ada pada basis data IPA2002 ? Perhatikan dan catat dengan seksama jumlahnya.  
**count to jumlah\_data**
11. Menghapus data-data yang sudah ditandai untuk dihapus pada perintah  
**delete**  
**pack**

12. Hitung berapa jumlah data yang ada pada basis data IPA2002 ? Perhatikan dan bandingkan jumlah data sebelum dan sesudah memberikan perintah **pack**. Ini untuk memperjelas perbedaan antara perintah **delete** tanpa dan yang diikuti dengan perintah **pack**.  
**count to jumlah\_data**
13. Tampilkan semua data yang ada dalam format seperti Excell  
**browse**
14. Tampilkan semua data yang ada dalam format seperti Excell untuk *field* nama siswa dan nama sekolah.  
**browse field nama\_siswa, nama\_sma**
15. Mengurutkan data berdasarkan urutan nama siswa.  
**index on nama\_siswa to nama**
16. Pergi ke data pertama.  
**go top**
17. Tampilkan semua data yang ada dalam format seperti Excell  
**browse**

## 3 Praktikum #3 - Membuat Program Clipper

Pada bagian ini, mahasiswa akan belajar dari menulis program, mengkompilasi dan kemudian me-linking supaya bisa menjadi program yang *executeable file*. Titik tekan materi ini adalah proses pembuatan *executeable file*. Artinya, program yang ditulis masih sederhana.

### 3.1 Program yang dibutuhkan

Untuk mengerjakan tugas pada praktikum ini, dibutuhkan perangkat mulai dari *text editor*, kemudian *compiler* dan *linker* beserta atribut tambahan seperti berkas-berkas pustaka (lib). Di sini, pemahaman tentang konsep direktori sangat diperlukan.

### 3.2 Langkah-Langkah Pekerjaan

Urut-urutan perintah yang dilakukan dalam pelaksanaan praktikum kali ini adalah sebagai berikut :

1. Dengan menggunakan *text editor* Qedit, suntinglah program berikut ini :

```
clear
@ 10,30 say "Ilmu
@ 11,30 say "Komputer "
@ 12,30 say "Unila"
@ 13, say "Lampung"
wait "tekan tombol apa saja !"
return
```

Menjalankan Qedit ada berbagai macam versi, tergantung direktori masing-masing. Perlu diingat bahwa direktori aktif (direktori kerja) sebaiknya di direktori masing-masing yang memang sudah disiapkan untuk xBase. Contoh cara menjalankan program Qedit :

```
D:\dwi\jim\kuliah\clipper>ca-52\bin\q - menjalankan program Qedit
```

2. Lakukan proses kompilasi dengan perintah seperti pada contoh. Perintah ini perlu dimodifikasi sesuai dengan posisi letak direktori masing-masing. Demikian juga dengan nama *file* yang dikompilasi. Jika terdapat kesalahan maka perbaiki *error* yang ada. *Error* yang mungkin muncul :

```
C:\dwi\clipper>\ca-52\bin\clipper test
Clipper (R) Version 5.2
Copyright (c) 1985-1993, Computer Associates International, Inc.
Microsoft C Floating Point Support Routines
Copyright (c) Microsoft Corp 1984-1987. All Rights Reserved.
336K available
Compiling TEST.PRG
TEST.PRG(2) Error C2007 Unterminated string: "Ilmu"
TEST.PRG(6) Error C2005 Statement not recognized,
match failed at: "<end of line>"
2 errors

No code generated
```



Lakukan perbaikan sesuai dengan pesan kesalahan yang diberikan.

```
C:\dwi\clipper>\ca-52\bin\clipper test
Clipper (R) Version 5.2
Copyright (c) 1985-1993, Computer Associates International, Inc.
Microsoft C Floating Point Support Routines
Copyright (c) Microsoft Corp 1984-1987. All Rights Reserved.
336K available
Compiling TEST.PRG
Code size 144, Symbols 112, Constants 80
```

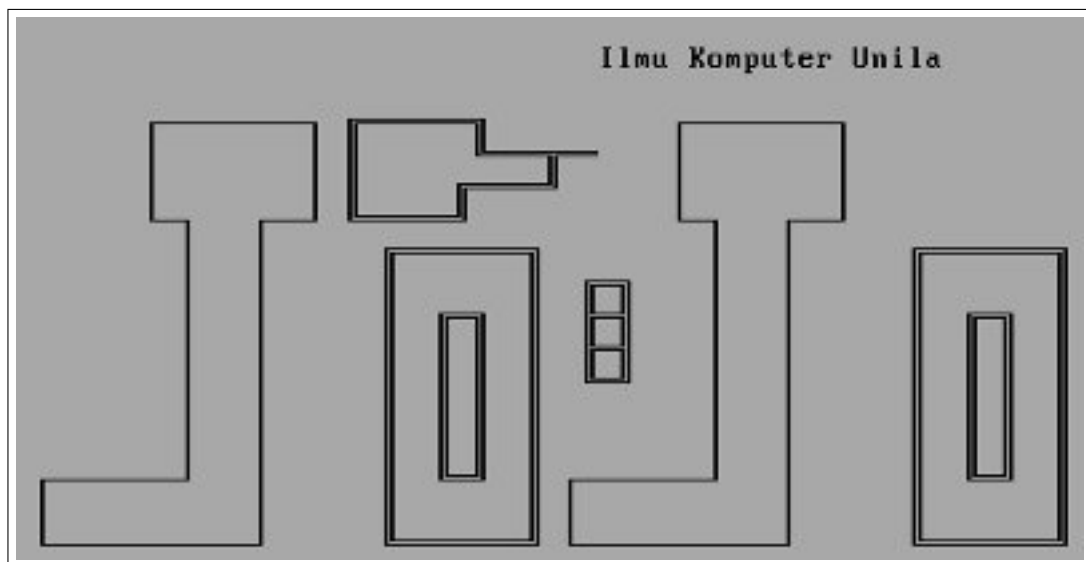
3. Setelah proses kompilasi sukses, lanjutkan dengan proses *linking*. Proses *linking* dilakukan dengan perintah :

```
C:\dwi\clipper>\ca-52\bin\blinker file test
```

4. Proses *linking* yang sukses akan menghasilkan *file* yang bisa dieksekusi *executable file*. Untuk menjalankan *file* tersebut, cukup dengan mengetikkan nama *file* tersebut.

### 3.3 Materi tambahan

Jika waktu yang tersedia masih banyak, buatlah program yang menghasilkan tampilan seperti pada contoh berikut. Tampilan bisa juga digubah sesuai dengan selera atau sesuai dengan identitas masing-masing.



Gambar 1: Contoh Logo

## 4 Praktikum #4 - Membuat Program Pemasukan Data

Praktikum ke-4 ini mengajak pemakai untuk belajar membuat program untuk memasukkan data melalui *keyboard*. Konsep tentang jenis-jenis variabel, cara memberi harga awal kepada masing-masing jenis variabel harus sudah dikenalkan. Pada praktikum ini masalah yang diangkat adalah membuat simulasi program untuk kasir. Akan tetapi simulasi di sini sangat disederhanakan. Program digunakan untuk menghitung harga pembelian suatu item barang. Contoh pemasukan dan hasil penghitungan dapat dilihat pada tampilan berikut:

```
Nama Barang   :  
Jumlah        :    0  
Harga Satuan  :      0  
Tgl. Beli     : 11/13/10  
Harga Total   :  
Bayar         :    0.000  
Kembali       :
```

Informasi yang dihasilkan berupa besarnya uang kembalian jika memang ada sisa dari uang belanja yang dibayarkan.

### 4.1 Bahan yang Dibutuhkan

Karena praktikum ini berkaitan dengan proses pembuatan program maka di sini dibutuhkan *compiler* dan *linker* Clipper dengan berbagai pelengkapannya seperti berkas *library* dan *include*. Termasuk juga di dalamnya adalah pengolah teks (text editor).

### 4.2 Materi Tambahan #1

Jika materi pada bagian pertama sudah selesai, maka praktikum dapat dilanjutkan ke materi tambahan berikut ini. Karena pada contoh ini, materi berhubungan dengan tanggal, maka pemberian harga awal kepada variabel bertipe tanggal harus sudah diberikan juga. Materi tambahan ini berisi materi berupa pengembangan sistem untuk menghitung umur seseorang. Data yang dimasukkan paling tidak berupa nama, tanggal lahir. Sedangkan tanggal sekarang bisa diisi, bisa langsung diambil dari sistem. Keluaran berupa tahun dan bulan. Misalnya, "Dwi Sakethi umur 18 tahun 7 bulan". Keterangan ini berbeda dengan hasil 18.7, untuk itu perlu dicek dengan sebenar-benarnya.

### 4.3 Materi Tambahan #2

Dari materi praktikum sebelumnya, paling tidak akan dimiliki dua buah berkas program. Maka selanjutnya pada praktikum ini, buatlah menu untuk menggabungkan kedua buah berkas tersebut. Dengan adanya menu ini, maka untuk menjalankan program-program tersebut akan menjadi lebih mudah.

## 5 Praktikum #5 - Program Pemasukan Data ke Basis Data

Pada pemasukan data di program-program sebelumnya, data yang diolah hanya disimpan ke dalam variabel memori. Akibatnya setelah selesai proses dan program kembali ke DOS PROMPT, maka data juga hilang. Karena data tidak disimpan ke dalam media penyimpanan. Oleh karena itu, perlu dikembangkan lagi proses pemasukan data. Data akan disimpan ke dalam media penyimpanan (ke dalam berkas .dbf).

### 5.1 Bahan yang Dibutuhkan

Untuk pemasukan data, yang mana data akan direkam ke dalam berkas .dbf, maka diperlukan berkas .dbf untuk menyimpan data-data tersebut. Oleh karena itu, pada praktikum ini, dibutuhkan berkas-berkas :

1. ipa2002.dbf, berkas ini digunakan untuk menyimpan data-data siswa.
2. sma.dbf, berkas ini digunakan untuk menyimpan data-data sekolah.
3. ps.dbf, berkas ini digunakan untuk menyimpan data-data program studi.

Struktur dari berkas basis data tersebut dapat dilihat pada Praktikum #1.

Karena praktikum ini berkaitan dengan proses pembuatan program maka di sini dibutuhkan *compiler* dan *linker* Clipper dengan berbagai pelengkapannya seperti berkas *library* dan *include*. Termasuk juga di dalamnya adalah pengolah teks (text editor).

### 5.2 Contoh Tampilan

Buatlah program untuk memasukkan data-data pada berkas ipa2002.dbf dengan item data seperti berikut :

Pemasukan/Editing Peserta PKAB

```
No. PKAB      :
Kode SMA      :
Kelamin       :          [1:Laki-laki 2:Perempuan]
No. Induk     :
Nama Siswa    :
B. Indonesia:
B. Inggris   :
Matematika    :
Fisika        :
Kimia         :
Biologi       :
```

### 5.3 Materi Tambahan #1

Jika praktikum bagian pertama sudah selesai, maka proses dapat diteruskan dengan membuat pemasukan data tambahan untuk data SMA dan data program studi. Item-item data yang dimasukkan sesuai dengan *field-field* yang ada pada kedua berkas basis data tersebut.

## 6 Praktikum #6 - Program Penghapusan Data di Basis Data

Pada program yang dibuat di praktikum sebelumnya, program digunakan untuk memasukkan data baru atau untuk memperbaiki data lama (data yang sudah ada). Pada praktikum kali ini, akan dibuat program untuk menghapus data. Suatu sistem yang di dalamnya ada pemasukkan data, maka setidaknya mesti dilengkapi dengan kemampuan untuk memperbaiki data yang sudah ada dan menghapus data yang ada.

### 6.1 Bahan yang Dibutuhkan

Sama seperti pada praktikum sebelumnya, bahan-bahan yang dibutuhkan meliputi berkas-berkas :

1. ipa2002.dbf, berkas ini digunakan untuk menyimpan data-data siswa.
2. sma.dbf, berkas ini digunakan untuk menyimpan data-data sekolah.
3. ps.dbf, berkas ini digunakan untuk menyimpan data-data program studi.

Karena praktikum ini berkaitan dengan proses pembuatan program maka di sini dibutuhkan *compiler* dan *linker* Clipper dengan berbagai pelengkapannya seperti berkas *library* dan *include*. Termasuk juga di dalamnya adalah pengolah teks (text editor).

### 6.2 Contoh Tampilan

Buatlah program untuk menghapus data-data di ipa2002.dbf dengan contoh tampilan seperti berikut ini :

```
-----  
-----  
--  
-- Penghapusan Peserta PKAB  
--  
-- No. PKAB      : 212  
-- Kode SMA      :  
-- Kelamin       : 1  
-- No. Induk     :  
-- Nama Siswa    : wiro sableng  
--  
---Mau dihapus [y/t] : -T-----  
-----  
-----
```

### 6.3 Materi Tambahan #1

Jika praktikum bagian pertama sudah selesai, maka proses dapat diteruskan dengan membuat penghapusan data untuk data SMA dan data program studi.

## 7 Praktikum #7 - Program Pemasukan Data dengan Validasi Biasa

Validasi adalah suatu proses untuk menentukan apakah data yang diisi sesuai dengan kriteria yang dibolehkan atau tidak. Jika kriteria sesuai dengan yang diperbolehkan maka data dianggap benar dan proses pemasukan data bisa dilanjutkan dengan data berikutnya. Jika data yang diisi tidak berada di dalam kriteria yang diperbolehkan maka proses pemasukan data tetap berada pada item tersebut, sampai item tersebut diisi dengan benar.

Buatlah program untuk pemasukan data seperti pada praktikum sebelumnya, yaitu untuk pemasukan data ipa2002. Pada saat memasukkan item **Jenis Kelamin** lakukan validasi dengan memberi batasan bahwa data hanya boleh diisi dengan 1 untuk laki-laki atau 2 untuk perempuan. Kemudian untuk nilai-nilai mata pelajaran hanya boleh diisi dengan nilai antara 0 sampai dengan 100. Jadi  $0 \leq \text{nilai} \leq 100$ . Jika data-data yang dimasukan salah maka berikan keterangan di layar sesuai dengan kriteria yang diperbolehkan.

Pemasukan/Editing Peserta PKAB

No. PKAB :  
Kode SMA :  
Kelamin : [1:Laki-laki 2:Perempuan]  
No. Induk :  
Nama Siswa :  
B. Indonesia:  
B. Inggris :  
Matematika :  
Fisika :  
Kimia :  
Biologi :

### 7.1 Bahan yang Dibutuhkan

Sama seperti pada praktikum sebelumnya, bahan-bahan yang dibutuhkan meliputi berkas-berkas :

1. ipa2002.dbf, berkas ini digunakan untuk menyimpan data-data siswa.

Karena praktikum ini berkaitan dengan proses pembuatan program maka di sini dibutuhkan *compiler* dan *linker* Clipper dengan berbagai pelengkapya seperti berkas *library* dan *include*. Termasuk juga di dalamnya adalah pengolah teks (text editor).