

Pengolahan Database dengan dBase

dBase, adalah suatu program paket untuk pengolahan database, awalnya program ini diterapkan pada mikrokomputer 8 bit dengan menggunakan sistem operasi CP/M-80™ versi dBase pertama adalah dBase II program ini sudah digunakan oleh mikrokomputer seperti;

Pada tahun 1983 dBase II versi 2.4 diedarkan untuk komputer PC IBM dengan menggunakan sistem operasi PC DOS V1.1, 2.0 atau 2.1, kemudian berkembang dBase III, dBase III +, dBase IV dan dBase V.

Database, adalah sekumpulan informasi yang diorganisasikan dalam suatu bentuk spesifik dan terstruktur

Organisasi dari database

1. Database Hirarkis, mengorganisasikan elemen data dalam model hirarkis yang menyerupai pohon. Hirarkis tersebut tidak hanya mendefinisikan elemen data dalam database namun juga mendefinisikan hubungan diantara elemen-elemen data.
2. Database Relasional, mengorganisasikan elemen data dalam bentuk tabel dua dimensi yang terdiri dari baris dan kolom. Database Relasional pada mulanya dikembangkan untuk penggunaan komputer besar. Kemudian konsep ini dipakai oleh dBase dalam mengorganisasikan elemen data yang pada komputer PC.

Komponen-komponen Database Relasional.

1. File, kumpulan dari record-record yang sejenis
2. Record, merupakan kumpulan dari field-field data
3. Field, merupakan kumpulan dari item-item data (karakter)

Data atau Informasi pada dBase ditampung dalam

- A. Field, tempat penampungan Data Utama
- B. Variabel Memori, tempat penampung data Pembantu/Sementara

Jenis Elemen-elemen data pada Field :

- | | | |
|-------------------|--------------|---|
| 1. Karakter | (C) | Panjang maksimum : 254 karakter |
| 2. Numerik | (N) | Panjang maksimum : 20 digit |
| 3. Date (tanggal) | (D) | Panjang maksimum : 8 karakter |
| 4. Memo | (M) | Maksimum berisikan : 512 karakter |
| 5. Logika | (L) | Panjang maksimum : 1 Karakter berisikan .T. (true) atau .F. (false) |

Variabel memori dibedakan atas .

1. Public
2. Privat

Jenis elemen-elemen data pada variabel memori.

1. Karakter
2. Numerik

Jenis Operator Pada dBase

1. Operator Matematika, Aritmatika :

- Pangkat (^)
- Perkalian (*)
- Penjumlahan (+)
- Pengurangan (-)
- Pembagian (/)

Operator Relasi,

- Untuk Aritmatik : <, >, =, <=, >=, # atau <>
- Untuk string : \$

2. Operator Logika

- AND•
- OR•
- NOT•

dBase dapat membaca, menyimpan data atau informasi pada 13 file yang spesifik dengan extension file sudah ditetapkan oleh dBase yaitu :

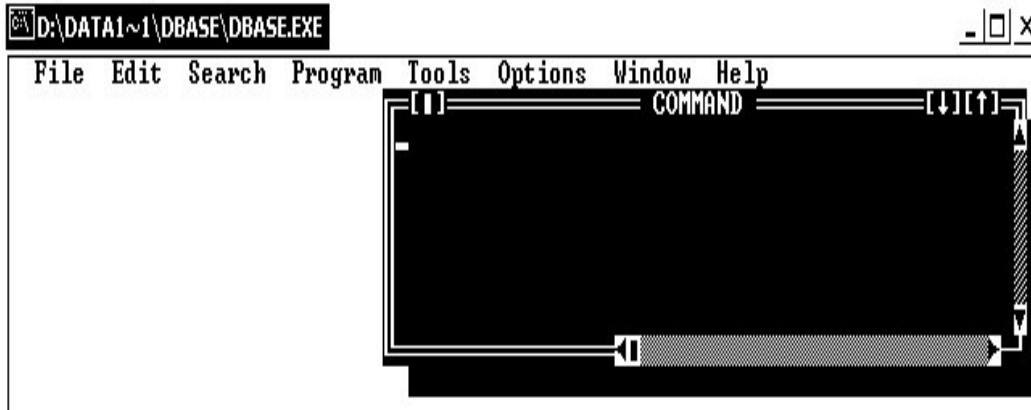
1. **File Database**, ekuivalen dengan tabel data relasional dimana file disusun berdasarkan record-record, dan setiap record terdiri dari field-field yang mengandung elemen-elemen data. (**Extension DBF**)
2. **File Memo Database**, file ini berifat sebagai file pendukung dari file Database, yang berfungsi untuk menyimpan isi dari **data field memo**. Artinya file ini muncul secara otomatis jika dalam struktur file database kita ada menggunakan field memo. (**Extension DBT**)
3. **File Index**, data dalam database dapat di index secara logic atau secara fisik pada satu atau beberapa field dari file yang bersangkutan. Hasil dari peng-index-an tersebut disimpan pada file ini. (**Extension NDX**)
4. **File Command atau Procedure**, adalah suatu file yang menyimpan sekumpulan perintah atau instruksi, dikenal dengan nama program. (**Extension PRG**)
5. **File Format**, adalah file yang menyimpan bentuk-bentuk format layar, pesanan yang digunakan dengan item-item data dalam sebuah database untuk memasukkan data dengan. (**Extension FMT**)

6. **File Label**, suatu fasilitas untuk keluaran /output dalam bentuk label. (**Extension LBL**)
7. **File Memori**, untuk menyimpan variable-variabel memori. (**Extension MEM**)
8. **File Report Form**, file untuk menghasilkan keluaran/ output dari database. (**ExtensionFMT**)
9. **FILE Output Text**, untuk menyimpan text yang dapat dibaca oleh perangkat lunak lainnya. (**Extension TXT**)
10. **File Catalog**, sejenis dengan file database yang bersifat special-pur-pose yang berisikan tentang informasi file-file guna memudahkan dalam penelusuran suatu lingkup aplikasi berisikan kumpulan dari file-file.DBF, .NDX, .FRM, .FMT, .LBL DAN .VIE. (**Extension CAT**)
11. **File Query**, berisikan informasi tentang data-data tertentu dari suatu file database yang sedang digunakan. Dapat juga digunakan penyaringan data dalam file database yang ada. (**Extension QRY**)
12. **File Screen**, berfungsi untuk menginformasikan tentang format (.FMT) file yang digunakan pada monitor sehingga kita dapat memberikan batasan-batasan tertentu. (**Extension SCR**)
13. **File View**, berfungsi untuk menyimpan informasi kondisi pengaturan atas dua buah file database dalam menampilkan unsur data pada masing-masing, dengan sistem key field. (**Extension VUE**)

Perintah Interaktif pada dBase

Masuk kedalam sistem dBase

C>dBase\dbase (enter)



Keluar dari sistem dBase

- QUIT (enter)

Perintah untuk pindah drive/directory aktif

- **SET DEFAULT TO** [drive], fungsi untuk memindahkan drive aktif
- **SET DIRECTORY TO** [directory], fungsi untuk memindahkan directory aktif
- **!CD** [directory], fungsi untuk memindahkan directory aktif

Perintah untuk Menghasilkan File

- **CREATE Sintak**
 - CREATE [drive/directory] [nama file]
 - CREATE LABEL [drive/directory] [nama file]
 - CREATE SCREEN [drive/directory] [nama file]
 - CREATE REPORT [drive/directory] [nama file]
 - CREATE QUERY [drive/directory] [nama file]

Membuat File Database

Sebagai contoh kita akan membuat satu file database (DBF) dengan nama file PEGAWAI, struktur filenya adalah sebagai berikut

<i>Field Name</i>	<i>Type</i>	<i>Width</i>	<i>Dec</i>
1. NIP	C	6	
2. Nama	C	30	
3. Alamat	C	30	
4. JK	C	1	
5. Tgl_Lahir	D	8	
6. Tmp_Lahir	C	15	
7. Agama	C	1	
8. Status	C	1	
9. Pendidikan	C	1	
10. Mulai_ker	D	8	
11. Jml_anak	N	2	0
12. Gapok	N	7	0

Pertama tuliskan perintah CREATE untuk membuat file pada DOT PROMT dBase, melalui cara seperti di bawah ini :

1. **Set directory to d:\data** [Enter]
 2. **Create d:\data\pegawai** [Enter]
- Create pegawai** [Enter]



Setelah melakukan perintah Create maka akan tersedia tempat untuk mulai mendefinisikan field yang diperlukan file tersebut.

Num	Field Name	Field Type	Width	Dec	Index
1	NIP	Character	6		N
2	NAMA	Character	30		N
3	ALAMAT	Character	30		N
4	JK	Character	1		N
5	TGLAHIR	Date	8		N
6	TMPLAHIR	Character	15		N
7	AGAMA	Character	1		N
8	STATUS	Character	1		N
9	PENDIDIKAN	Character	1		N
10	MULAIKER	Date	8		N
11	JMLANAK	Numeric	2	0	N
12	GAJIPOKOK	Numeric	8	0	N
13		Character			N

Untuk menyimpan Struktur file yang telah dibuat menggunakan tombol **CTRL-END** atau **CTRL- W** atau tekan **ENTER 2x**. Setelah itu akan muncul pertanyaan apakah data akan di-input sekarang atau nanti ?



catatan : Syarat untuk menulis nama field adalah sebagai berikut

- panjang nama field maksimum 10 Karakter
- tidak boleh ada spasi ,tanda baca kecuali tanda hubung bawah(_)
- tidak boleh diawali dengan angka

untuk mempermudah kita dalam memanipulasi data, kita buat kesepakatan secara bersama untuk pengisian data fieldnya, yaitu :

- Field JK diisi dengan : P = untuk Perempuan
: L = untuk Laki-laki
- Field Agama diisi dengan angka : 1. Untuk agama islam
2. untuk agama kristen katolik
3. untuk agama kristen protestan
4. untuk agama budha
5. untuk agama hindu
- Field Status diisi dengan : B = untuk belum kawin
K = untuk Kawin
D = untuk yang Duda
J = untuk yang Janda
- Field Pendidikan diisi dengan : 1 = untuk tamat SMP
2 = untuk tamat SMU
3 = untuk tamat Diploma
4 = untuk tamat Sarjana

Untuk menyimpan Data dipergunakan tombol CTRL-END atau CTRL-W.

Perintah-perintah untuk Memodifikasi dan Memanipulasi data

- **USE** : Fungsi untuk membuka atau menutup file database, sintaknya
 - USE, untuk menutup satu file dbf yang aktif
 - USE [drive/directory] <nama file dbf>, untuk membuka satu file dbf
 - USE [drive/directory] <nama file dbf> INDEX [drive/directory] <nama file index>, untuk membaca file dbf sekaligus dengan file indexnya

Cth. Use pegawai index pegawai

Contoh : aktifkan file database yang telah dibuat dimana data tersebut disimpan pada drive D dalam directory dBase.

Jawab : ada beberapa alternatif untuk melakukannya diantaranya:

1. *Set directory d:\dbase <enter>*
Use pegawai <enter>
2. *Use d:\dbase\pegawai <enter>*

- **CLEAR** berfungsi untuk membersihkan layar atau memori dari file-file dbase, sintak
 - CLEAR, untuk membersihkan layar monitor
 - CLEAR [all/database/memori/screens]
- **CLOSE** fungsi untuk membersihkan memori dan menutup file, sintak
 - CLOSE all / database / procedure / alternat / index
- **APPEND** : fungsi untuk menambah atau menggabung data, perintah ini akan berfungsi jika file database yang akan ditambah harus diaktifkan terlebih dahulu, sintak
 - APPEND, untuk menambah data
 - APPEND BLANK, untuk menambah / membuat satu record kosong setelah record terakhir

- APPEND FROM [drive / directory]<nama file database (dbf)>, untuk menggabung file (dbf) pada file (dbf) yang sedang aktif

● **LIST dan DISPLAY** fungsi untuk melihat atau menampilkan record-record data.

Perintah ini akan berfungsi jika file database (dbf) sudah diaktifkan. Sintak

- LIST [off] [scope] [fields] [**FOR**<kondisi>] [to print]
- DISPLAY [off] [scope] [fields] [**FOR**<kondisi>] [to print]
- LIST STRUCTURE untuk melihat struktur data yang aktif
- DISPLAY STRUCTURE sama dengan list structure

Jika **OFF** diikutkan setelah perintah **list atau display**, maka nomor record tidak ditampilkan, kalau tidak OFF nomor record ikut ditampilkan.

- **SCOPE**, berisikan perintah sebagai berikut (**All,Record n,Next n**):
 - jika mengikutkan perintah **ALL** setelah perintah display, maka akan menampilkan semua record mulai dari record pertama sampai record terakhir
 - jika mengikutkan perintah **Record n** setelah perintah **list atau display**, maka akan menampilkan nomor record yang disebutkan (record ke n)
 - jika mengikutkan perintah **Next n**, setelah perintah **list atau display**, maka akan menampilkan record sebanyak n record mulai dari posisi record aktif.
- **FIELDS**, untuk menampilkan field-field tertentu saja dengan cara menulis nama-nama field yang akan ditampilkan dimana setiap nama field dipisah dengan tanda koma (,)
- **FOR** <kondisi>, untuk menampilkan record-record tertentu saja, sebagai filter dipergunakan satu atau lebih field. Jika lebih dari satu field sebagai filter maka setiap field dipisah dengan menggunakan operan dari operator logika.

- **TO PRINT**, untuk mencetak tampilan ke printer. Jika perintah *To Printer* diikuti setelah perintah list atau display maka tampilan juga akan dicetak ke printer.

Dibawah ini adalah contoh penggunaan perintah LIST.

- *List <enter>*

Record#	NIP	NAMA	TGLAHIR	TMPLAHIR	AGAMA	STATUS	PENDIDIKAN	MULAIKER	JMLANAK	GAJIPOKOK
1	108501	Dira Suprima	06/06/66	Semarang	1	k	4	05/15/85	2	10000000
2	308902	Rini	07/07/77	Solo	2	b	3	02/01/89	0	4000000
3	409404	Rossi Sofia	05/05/73	Cirebon	1	b	2	10/09/94	0	2500000
4	208401	Indra	08/19/70	Cianjur	1	k	1	01/01/84	1	8000000
5	309002	Suhardiyanto	12/01/75	Jogja	1	b	3	06/11/90	0	4500000
6	409004	Sondang Tiurma Julianti S	04/07/76	Medan	2	k	2	11/11/90	0	2000000
7	309005	Meilani	09/09/76	Jakarta	2	b	3	08/11/90	0	3000000
8	408807	Prabu Jayadiningrat	10/08/73	Magelang	1	b	2	03/02/88	0	1500000
9	208501	Arinithia khristiyantie	07/05/69	Madiun	2	k	6	09/08/85	1	8000000

- *List off nip,nama,mulaiker <enter>*

nip	nama	mulaiker
108501	Dira Suprima	05/15/85
308902	Rini	02/01/89
409404	Rossi Sofia	10/09/94
208401	Indra	01/01/84
309002	Suhardiyanto	06/11/90
409004	Sondang Tiurma Julianti S	11/11/90
309005	Meilani	08/11/90
408807	Prabu Jayadiningrat	03/02/88
208501	Arinithia khristiyantie	09/08/85

- List nip,nama,mulaiker for year(mulaiker)=1990

```

D:\DATA1~1\DBASE\DBASE.EXE
File Edit Search Program Tools Options Window Help
Record# nip nama mulaiker
5 309002 Suhardiyanto 06/11/1990
6 409004 Sondang Tiurma Julianti S 11/11/1990
7 309005 Meilani 08/11/1990

```

Perbedaan perintah list dan display adalah :

LIST	DISPLAY
<ul style="list-style-type: none"> - Dari record awal sampai record akhir - Secara terus menerus 	<ul style="list-style-type: none"> - record per record - perhalaman

Perintah lainnya yang berfungsi hampir sama dengan List adalah BROWS. Perintah Brows tidak mempunyai nilai kondisi. Dengan Brows kita juga dapat melakukan editing terhadap data kita.

- BROWS <enter>

```

D:\DATA1~1\DBASE\DBASE.EXE
Records Organize Fields Go To Exit
NIP NAMA ALAMAT JK TGLAHI
108501 Dira Suprma Jakarta p 06/06/
308902 Rini Jakarta Selatan p 07/07/
409404 Rossi Sofia Jakarta Pusat P 05/05/
208401 Indra Jakarta Selatan l 08/19/
309002 Suhardiyanto Depok l 12/01/
409004 Sondang Tiurma Julianti S Jakarta Utara p 04/07/
309005 Meilani Jakarta Barat p 09/09/
408807 Prabu Jayadiningrat Jakarta Pusat L 10/08/
208501 Arinithia khristiyantie Jakarta Barat p 07/05/

Browse ||D:\...\dbase\PEGAWAI ||Rec 1/9 ||File ||ExclLock|| Num

```

Perintah untuk menghapus Record :

● **DELETE ON/OFF.**

ON record-record yang dihapus tidak ditampilkan di layar.

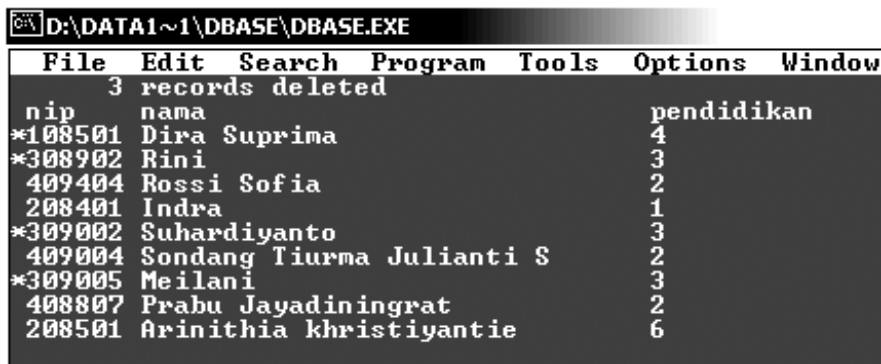
OFF record-record yang dihapus ikut ditampilkan tetapi ada *tanda bintang(*)*.

Record-record yang dihapus tersebut, juga bisa kita lakukan pembatalan penghapusan atau benar-benar kita ingin menghilangkan record-record tersebut dari memori dengan perintah sebagai berikut:

- **RECALL** untuk pembatalan penghapusan record-record, dengan sintak **RECALL [SCOPE][FOR<kondisi>]**
- **PACK** untuk penegas penghapusan record

Contoh : *delete off for pendidikan='3' <enter>*

List off nip,nama,pendidikan <enter>



File	Edit	Search	Program	Tools	Options	Window
3 records deleted						
nip	nama					pendidikan
*108501	Dira Suprima					4
*308902	Rini					3
409404	Rossi Sofia					2
208401	Indra					1
*309002	Suhardiyanto					3
409004	Sondang Tiurma Julianti S					2
*309005	Meilani					3
408807	Prabu Jayadiningrat					2
208501	Arinithia khristiyantie					6

Untuk membatalkan penghapusan yang telah dilakukan gunakan perintah : *recall for pendidikan='3'*

Untuk penghapusan File :

- **DELETE FILE** [drive/directory]<nama file>
- **ERASE**, fungsi untuk menghapus file-file dalam media penyimpanan sama dengan perintah **DELETE FILE**
Sintak **ERASE** [drive/directory]<nama file>
Contoh : *delete file pegawai <enter>*
erase pegawai <enter>

Untuk merubah isi data/field dengan data baru.

- **REPLACE**, fungsi untuk merubah atau mengisi field dari dotprompt secara otomatis.

Sintak **REPLACE** [scope]<nama field> **WITH** <isi field yang baru>
[for<kondisi>]

Contoh :

```
.replace all gajipokok with gajipokok*1.1
```

menaikkan seluruh gaji pokok sebesar 10 %

```
.replace all gajipokok with gajipokok*1.1 for pendidikan='4'
```

menaikkan seluruh gaji pokok sebesar 10 % untuk pegawai yang pendidikannya sarjana

```
.replace alamat with 'Jakarta Selatan' for lower(nama)='dira  
suprina'
```

merubah data alamat Dira Suprina dengan alamat baru yaitu Jakarta Selatan

- Perintah untuk membawa atau memindahkan petunjuk record

- **GO RECORD #**, untuk memindah petunjuk ke record nomor tertentu

```
.Go reco 5
```

- **GO TOP**, untuk memindahkan penunjuk record ke record pertama
- **GO BOTTOM**, untuk memindahkan penunjuk record ke record terakhir
- **SKIP n/-n**, untuk memindah penunjuk record maju sebanyak n record atau mundur sebanyak -n record.

```
.skip 2
```

```
.skip -2
```

- **INSERT**, fungsi untuk menyisipkan record pada posisi tertentu
 Sintak : Insert [blank] [before]
 Contoh :
 penunjuk record sekarang berada pada posisi 4 bila dilakukan operasi :

insert blank → menyediakan record kosong pada posisi record ke 5
 insert before → menyediakan record kosong pada posisi record 4

- **COPY**, fungsi untuk menyalin file, record-record atau struktur field-field tertentu ke dalam suatu file baru, dengan sintak sebagai berikut :

 - **COPY TO** <nama file> [FOR<kondisi>], fungsi untuk menyalin record-record dari file yang aktif ke file baru dengan hasil file (dbf)
 Contoh : *copy to wanita for lower(jk)='p'*
 - **COPY FIELD** <field 1,field2,...> TO [drive/directory] <nama file> [FOR<kondisi>], untuk menyalin field-field tertentu dan record-record dari file yang aktif ke file baru dengan hasil file (dbf)
 Contoh :
copy field nip,nama,pendidikan,gajipokok for pendidikan='2'
 - **COPY STRUCTURE TO** [drive/directory] <nama file>, hanya untuk menyalin struktur field dari file aktif, hasil file (dbf) tanpa isi recordnya
 Contoh : *copy stru to a:staff*
 - **COPY FILE** [drive/directory] <nama file> TO [drive/directory] <nama file>, untuk menyalin suatu file ke dalam file lain beserta isi recordnya
 Contoh : *copy file pegawai to a:*

- **RENAME**, fungsi untuk mengganti nama file, sintak
RENAME [drive/directory] < file lama> TO [drive/directory] <file baru>
 Contoh : *rename pegawai to karyawan*

- **SORT**, fungsi untuk mengurutkan data secara fisik berdasarkan field-field yang diinginkan. Perintah ini akan menghasilkan file dbf juga.
 Sintak **SORT ON** <field 1,field2,...> TO [drive/directory] <nama file>[A/D]

Contoh : *sort on nip,nama to nipsort A*

Data sebelum dilakukan sort : *list nip,nama*

```
Record#  nip  nama
1  108501  Dira Suprima
2  308902  Rini
3  409404  Rossi Sofia
4  208401  Indra
5  309002  Suhardiyanto
6  409004  Sondang Tiurma Julianti S
7  309005  Meilani
8  408807  Prabu Jayadiningrat
9  208501  Arinithia khristiyantie
```

Data setelah dilakukan sort dengan nama file beda.

. *Use nipsort*

. *List nip,nama*

```
Record#  nip  nama
1  108501  Dira Suprima
2  208401  Indra
3  208501  Arinithia khristiyantie
4  308902  Rini
5  309002  Suhardiyanto
6  309005  Meilani
7  408807  Prabu Jayadiningrat
8  409004  Sondang Tiurma Julianti S
9  409404  Rossi Sofia
```

- **INDEX**, fungsi untuk mengurutkan data secara logic dan fisik dengan menjadikan satu atau lebih field sebagai key. Perintah ini akan menghasilkan file dengan extension NDX untuk dbase, IDX untuk jenis fox dan NTX untuk jenis clipper. Sintaknya adalah :

INDEX ON [+/-] <field+Field+...> **TO** [drive/directory] <nama file>

Contoh : *index on nip to nipind*

Untuk membuka file index perintahnya adalah :

Use pegawai inde nipin

```
Record#  nip  nama
1  108501  Dira Suprima
4  208401  Indra
9  208501  Arinithia khristiyantie
2  308902  Rini
5  309002  Suhardiyanto
7  309005  Meilani
8  408807  Prabu Jayadiningrat
6  409004  Sondang Tiurma Julianti S
3  409404  Rossi Sofia
```

Perintah untuk mencari record tertentu.

- **LOCATE**, fungsi untuk mencari data mulai dari record awal sampai record terakhir dengan berpedoman pada satu atau lebih field dari record, dengan sintak **LOCATE FOR <nama field dan kondisinya>**

Contoh : *loca for lower(nama)='Rini'*

loca for tglahir=ctod('05/05/73')

loca for nip='3049005'

Jika ingin melanjutkan pencarian penggunaan perintah **CONTINUE**

- **SEEK dan FIND**, fungsi untuk mencari data pada data index dengan menggunakan key index dari data tersebut, sintak sbb

SEEK <field key> berlaku untuk jenis type data C, N, dan D

FIND <field key> hanya berlaku untuk jenis type data Karakter

Contoh : *seek '3049005'*

find 3049005

Perintah untuk modifikasi/editing struktur file adalah :

- **MODIFY STRUCTURE** untuk merubah struktur data yang aktif
MODI STRU [nama file]

FUNGSI ARITMATIKA PADA dBase

Operasi matematika dari jenis data numerik dapat dilakukan baik untuk field, variabel memori, atau secara langsung seperti kalkulator.

- ? 3 + 5
8
- ? 3/4
0.75
- SQRT(4)
2
- X = 15
- Y = 20
- ? X * Y
300
- use pegawai
- go reco 6
- repla gajipokok with gajipokok+500000

Fungsi absolut **ABS()** memberikan nilai absolut pada data numerik

- tanggal1 = {5/12/05}
25/12/88
- tanggal2 = {01/04/05}
01/04/05
- ? abs(tanggal1-tanggal2)
268.00
- ? abs(tanggal2-tanggal1)
268.00

Funfsi **INT()** menentukan nilai integer data numerik

- ? int(10.23)
10

Fungsi **ROUND()** untuk membulatkan angka dengan angka desimalnya.

- ? round(-5.556,0)
-6.00
- ? round(14.764,2)
14.75
- ? round(14.764,0)
15

Fungsi **VAL()** untuk merubah/konversi dari data karakter ke format data numerik

- ? val("234")
234
- ? val("abc")
0

Fungsi **STR()** untuk merubah/konversi dari data numerik ke format data karakter

- ? str(234)
234

Fungsi **MOD()** untuk menghitung sisa suatu pembagian

- ? mod(14,12)
2
- ? mod(0,3)
0

AVERAGE menghitung rata-rata

- average gajipokok
9 records average
gajipokok
4833333.33

COUNT berfungsi untuk menghitung jumlah record tertentu

- count for lower(alamat)='jakarta selatan'
3 records
- count for year(tglahir)<1975
5 records

MAX menghitung nilai terbesar dari data numerik

- satu = 12.32
- dua = 34.12
- tiga = 26.21
- ? max(satu,dua,tiga)
34.12

MIN menghitung nilai terkecil dari data numerik

- ? min(satu,dua,tiga)
12.32

SUM berfungsi untuk menjumlahkan seluruh data numerik pada sebuah field atau variabel memori

- sum gajipokok
9 records summed
gajipokok
43500000

FUNGSI STRING PADA dBase

AT() untuk menentukan posisi awal dari suatu karakter pada suatu string

- ? AT('a','Jakarta')
- 2
- ? AT('b','Jakarta')
- 0
- ? AT('lima','dua puluh lima')
- 11

LEFT() untuk mengambil sebagian karakter dari suatu string mulai dari posisi ujung **kiri** sampai jumlah yang ditentukan.

- ? LEFT('Jakarta',3)
- Jak

RIGHT() untuk mengambil sebagian karakter dari suatu string mulai dari posisi ujung **kanan** sampai jumlah yang ditentukan.

- ? RIGHT('Jakarta',3)
- rta

STUFF() menggantikan atau menyisipkan suatu karakter ke dalam suatu string yang ditentukan posisi awalnya serta jumlahnya.

- ? STUFF('filandere',3,5,"")
- fire
- ? STUFF('finder',3,1,"lan")
- filander

REPLICATE perintah untuk mengulangi penulisan suatu string yang sama.

- ? REPL('abc-',3)
- abc-abc-abc-

LOWER() berfungsi merubah seluruh huruf besar menjadi huruf kecil

- ? LOWER('Guna Darma')
- guna darma

UPPER() berfungsi merubah seluruh huruf kecil menjadi huruf besar

- ? UPPER('Guna Darma')
- GUNA DARMA

LEN() menentukan panjang suatu string

- ? LEN('Universitas Gunadarma')
- 21

SPACE() memberikan spasi kosong (maksimum=254 karakter)

- ? npm+space(2)+nama
10405001 Steven Gerrad

STR() untuk merubah type data numerik menjadi karakter

- ? STR(11.14)
11

VAL() untuk merubah type data karakter menjadi numerik

- ? VAL("abc")
0
- ? VAL('123.45')
123.5

SUBSTR() berfungsi untuk mengambil string dari suatu string

- SUBSTR('1958 1959 1960',8,2)
59
- STORE "juventus milan" to ITALY
- ? SUBSTR(ITALY,7,5)
usmi

LTRIM() berfungsi untuk menghapus spasi kosong yang berada di bagian kiri dari suatu string.

- STORE ' Liverpool' to KOTA
Liverpool
- ? LTRIM(STR(KOTA))
Liverpool

RTRIM berfungsi untuk menghapus spasi kosong yang berada di bagian kiri dari suatu string.

- ? RTRIM('Manchester ') + 'United'
Manchester United

FUNGSI DATE PADA dBase

CTOD() Singkatan dari Character To Date berfungsi mengubah isian tanggal yang dimaksudkan sebagai karakter menjadi type data Date

- STORE '07/03/05' TO tgl
- ? TYPE('tgl')

C

- STORE CTOD(tgl) TO tgl_baru
- ? TYPE('tgl_baru')

D

DATE() merupakan fungsi untuk menampilkan tanggal dari sistem

- ? DATE()
07/03/05

SET CENTURY ON fungsi ini untuk menampilkan tulisan tahun secara lengkap

- ? DATE()
07/03/2005

DAY() fungsi menampilkan tanggal dari date, hasilnya nilai numerik

- ? DAY(tgl)
7

DOW() Day Of the WEEK menampilkan angka yang menggambarkan hari dalam satu minggu dari tanggal, yang dimulai dengan hari Minggu yang ditetapkan sebagai angka 1. Hasilnya nilai numerik

- ? DOW(tgl)
2

CDOW() Calender Day Of the Week menampilkan nama hari dari suatu tanggal

- ?CDOW(tgl)
Monday

MONTH() menampilkan angka yang menggambarkan bulan dari tanggal. Hasilnya nilai numerik

- ? MONTH(tgl)
3

CMONTH() Calender Month menampilkan nama bulan dari data tanggal

- ? CMONTH(tgl)
MARCH

DTOC Date To Character mengubah data dari jenis tanggal menjadi string

- ?DTOC(tgl)

07/03/05

- ?TYPE(tgl)

C

YEAR() fungsi untuk menampilkan tahun dari data tanggal

- ? YEAR(tgl)

2005

Perhitungan tanggal :

- tgl1={07/03/05}

- tgl2={01/01/05}

- ?tgl1-tgl2

65

VARIABEL MEMORI

Jenis Variabel Memori

1. Variabel memori **Public** nilai suatu variabel dapat dipakai secara bersama
2. Variabel memori **Privat** nilai suatu variabel hanya dipakai sendiri-sendiri

Perintah untuk menipulasi variabel memori

STORE atau =, fungsi untuk memasukan harga pada suatu variabel memori

DISPLAY MEMORI, fungsi untuk melihat variabel memori yang aktif

SAVE TO [D/D] <nama file>, fungsi untuk menyimpan variabel memori yang aktif pada suatu file memori (mem))

RELEASE [ALL]<nama variabel>, fungsi untuk menghapus variabel memori yang aktif

RESTORE [D/D] <nama file mem>, fungsi untuk mengaktifkan file memori (mem)

Contoh membuat

Set talk off <enter>

SEKOLAH='STMIK' <enter>

ALAMAT='BRI RADIO DALAM JAKARTA SELATAN' <enter>

B=1035 <enter>

A=5 <enter>

C=6 <enter>

D=A*C <enter>

DISP MEMORI <enter>

User memori variables

D pub N 30 (30.000000000000000000)

C pub N 6 (6.000000000000000000)

B pub N 1035 (1035.0000000000000000)

A pub N 5 (5.000000000000000000)

ALAMAT pub C "BRI RADIO DALAM JAKARTA SELATAN"

SEKOLAH pub C "STMIK"

6 out of 500 memvar defined (and 0 array elements)
user MEMVAR/RTSYM Memori Usage

2800 bytes used for 1 memvar blocks (max=10)

850 bytes used for 1 rtsym blocks (max=10)

0 bytes used for 0 array element memvars

36 bytes used for 2 memvar character strings

3686 bytes total

150016 bytes available memory