

Modul 4

Data Manipulation Language

Data Manipulation Language (DML) merupakan query yang digunakan untuk memanipulasi data seperti menampilkan data, mengubah data, atau mengisi data.

4.1 Memasukkan/Mengisi Data

Perintah DML INSERT digunakan untuk memasukkan data ke dalam table
Sintaks yang dipakai adalah :

```
INSERT INTO Nama_Tabel (Daftar_Kolom) VALUES (Daftar Nilai)
```

Yang harus diperhatikan dalam insert sebagai berikut :

1. Jika type data string (varchar, char, text) gunakan tanda petik tunggal untuk mengapit data yang akan di-insertkan.
2. Jika type data numeric (int, numeric, decimal) tidak boleh menggunakan petik tunggal.
3. Jika type data datetime gunakan petik tunggal jika settingan system adalah tanggal dengan format inggris atau amerika maka insert data adalah 'mm-dd- yyyy'
4. Jika data autonumber sebaiknya data tidak diisi.

Ada beberapa cara INSERT data,yaitu :

1. Tanpa menyebutkan daftar kolom, dengan ketentuan tidak ada kolom autonumber dan autocompute dalam table itu dan data yang diisikan harus terurut sesuai dengan urutan kolom dalam table.
2. Menyebutkan daftar kolom

Aktifkan database anda dengan perintah :

USE DATABASE pbd

Contoh :

Mengisi data Dokter

```
insert into dokter values('D0001','Ema Utami','P','Yogyakarta',2500000)
```

4.1.1 Praktikum INSERT

Gunakan perintah SQL untuk mengisi semua data berikut ini ke tabel masing-masing,

Dokter

Kd-dokter	Nama	Gender	Alamat	Gaji
D0001	Ema Utami	P	Yogyakarta	2500000
D0002	Suwanto Raharjo	L	Jakarta	2000000
D0003	Emi Susanti	P	Semarang	1500000
D0004	Andi Sunyoto	L	Bandung	2000000
D0005	Emha Taufik Luthfi	L	Yogyakarta	3000000

Pasien

Kd_pasien	Nama	Tgl_Lahir	Gender	Alamat	Periksa
P0001	Risa	1997/12/10	P	Jakarta	1
P0002	Janu	1996/10/02	L	Surabaya	2
P0003	Reva	2000/11/30	P	Yogyakarta	4
P0004	Didi	1989/01/20	L	Surakarta	3
P0005	Lativa	1999/04/14	P	Purwokerto	1
P0006	Fanie	1980/09/20	L	Cilacap	2
P0007	Putri	2001/06/06	P	Yogyakarta	5
P0008	Randi	2010/07/12	L	Jakarta	3
P0009	Daffa	1995/05/05	L	Semarang	2
P0010	Vika	1996/06/06	P	Bandung	4

Penyakit

Kd_penyakit	Penyakit
K0001	Batuk
K0002	Panu
K0003	Kadas
K0004	Kurap
K0005	Konstipasi
K0006	Migrain
K0007	Tipus
K0008	Hepatitis
K0009	Diabetes
K0010	Hipertensi

Obat

Kd_obat	Obat
M0001	Antalgin
M0002	Parasetamol
M0003	CTM
M0004	Antibiotik
M0005	Salep
M0006	Dulcolax
M0007	Pacepil
M0008	Imunolax
M0009	Hiposetamol
M0010	Metromex

Jadwal_Dokter

Kd_jadwal	Hari	Shift	Kd_dokter
A0001	Senin	Pagi	D0001
A0002	Senin	Sore	D0002
B0001	Selasa	Pagi	D0004
B0002	Selasa	Sore	D0003
C0001	Rabu	Pagi	D0001
C0002	Rabu	Sore	D0005
D0001	Kamis	Pagi	D0003
D0002	Kamis	Sore	D0002
E0001	Jum'at	Pagi	D0004
E0002	Jum'at	Sore	D0001
F0001	Sabtu	Pagi	D0003

Resep

Hari	Tanggal	Kd_Pasien	Kd_Penyakit	Kd_Obat	Kd_Dokter
Senin	2015/11/2	P0001	K0001	M0001	D0001
Senin	2015/11/2	P0002	K0001	M0001	D0002
Senin	2015/11/2	P0002	K0001	M0001	D0002

Lanjutkan isi sendiri sebanyak 10 record

4.2 Retrieving Data / Menampilkan Data

SELECT merupakan query yang digunakan untuk mengambil data atau menampilkan data. Data yang telah diambil atau dipilih boleh saja ditampilkan dalam aplikasi yang kita buat atau digunakan untuk kontrol program. Sintak umum dari SELECT adalah sebagai berikut :

```
SELECT Daftar_Kolom FROM Nama_Tabel WHERE Kondisi ORDER BY Kolom
```

Contoh :

Menampilkan data-data dokter

```
SELECT * FROM Dokter
```

4.2.1 Praktikum SELECT

Sebagai latihan untuk perintah SELECT, kerjakan soal berikut ini :

1. Tampilkan semua data pasien
2. Tampilkan nama dan gaji dari tabel dokter

4.3 SELECT Dengan Kondisi

4.3.1 Klausula WHERE

WHERE Digunakan untuk menyatakan kondisi yang harus dipenuhi oleh sebuah record data yang akan dipilih oleh perintah SELECT.

Sintaks UMUM :

```
SELECT [Daftar_Kolom] FROM Tabel WHERE Kondisi
```

Operator yang digunakan untuk klausa WHERE adalah sebagai berikut :

Operator	Pengertian
=	Sama Dengan
<>	Tidak Sama Dengan
>	Lebih Besar
<	Lebih Kecil
>=	Lebih Besar atau Sama Dengan
<=	Lebih Kecil atau Sama Dengan
BETWEEN	Diantara Dua Nilai
LIKE	Mencari Bagian yang sama
AND	Menggabungkan Dua Kondisi
NOT	Menolak Kondisi
OR	Syarat Kondisi Untuk Setiap Pilihan

Operator logika terdiri dari AND, OR, dan NOT. Operator AND dan OR digunakan membandingkan kondisi pada klausa WHERE. Operator NOT digunakan untuk mencari nilai kebalikan dari kondisi yang dicari.

4.3.2 DISTINCT

Apabila kita ingin menampilkan data tetapi data yang sama tidak ditampilkan berulang-ulang maka kita bisa menggunakan kata kunci DISTINCT. Perintah ini dilakukan jika terdapat banyak data yang sama dalam satu kolom.

Sintaks UMUM :

```
SELECT DISTINCT [Kolom] FROM Tabel
```

4.3.3 BETWEEN

Perintah ini jika kita ingin mengambil beberapa data yang terletak antara nilai dalam klausa BETWEEN. Hanya berlaku pada data yang bertipe numeric dan string.

Sintaks UMUM :

```
SELECT [Daftar_Kolom] FROM Tabel WHERE [Kondisi]  
BETWEEN [Batas_Nilai_1] AND [Batas_Nilai_2]
```

4.3.4 TOP

Perintah ini digunakan untuk menampilkan beberapa data paling atas dari hasil SELECT. Misalnya kita ingin menampilkan 3 data paling atas saja.

Sintaks UMUM :

```
SELECT TOP n [Daftar_Kolom] FROM Tabel [WHERE Kondisi]
```

Contoh :

Menampilkan 3 data saja dari urutan paling atas :

```
SELECT TOP 3 * FROM Dokter
```

4.3.5 IS NULL

NULL adalah nilai dari sebuah field atau kolom yang tidak memiliki data apapun. NULL *tidak sama* dengan sebuah karakter spasi dan atau nilai 0 pada data numeric.

4.3.6 Sorting

Untuk mengurutkan data-data dari hasil SELECT, klausa yang digunakan ialah ORDER BY. Proses pengurutan dapat dilakukan secara ascending maupun descending. Secara default pengurutan dilakukan secara ascending.