

2. DML (Data Manipulation Language)

Terdiri dari **SELECT, INSERT, UPDATE dan DELETE**

Memasukkan Data (*Insert*)

Ada dua cara yang dapat digunakan untuk memasukkan data kedalam tabel, yaitu:

1. *insert into nama_tabel values (isi_field_2,...,isi_field_n);*

Contoh :

```
insert into produk values ('SG 100',2500);
```

2. *insert into nama_tabel (nama_field_1, nama_field_2,...,nama_field_n) values (isi_field_1, isi_field_2,...,isi_field_n);*

Contoh :

```
Insert into produk (kode,harga) values ('SG 100',25000);
```

Menampilkan Data (*Select*)

Berikut ini perintah untuk menampilkan data :

Struktur Penulisan :

```
Select [fields] from [nama_tabel]
Where [kondisi]
Order by [nama_field]
Group by [nama_field] asc / desc
```

Contoh :

```
Select kode from produk;
Select kode,harga from produk;
Select * from produk;
```

Note :

Setelah penulisan select, [fields] dapat diganti dengan menyebutkan satu nama field saja atau beberapa field sekaligus yang dipisah dengan tanda koma (,). Penulisan field-field tersebut digunakan untuk memunculkan data dari kolom mana saja yang akan ditampilkan. Jika seluruh kolom akan ditampilkan dapat menggunakan tanda asterik (*) untuk mewakilinya.

Mengurutkan Tampilan

Klausa ORDER BY digunakan untuk mengurutkan hasil. Jika menginginkan data ditampilkanurut berdasarkan pada urutan terkecil ke besar, dapat menggunakan ASC (*ascending*). Sedangkan untuk mengurutkan data berdasarkan yang terbesar ke kecil, dapat menggunakan DESC (*descending*).

```
mysql> select * from siswa order by NIM desc;
+-----+-----+-----+-----+
| NIM      | Nama          | alamat      | Sex  |
+-----+-----+-----+-----+
| 01.05.2004 | Leon Auman    | Jl. depok   | P    |
| 01.05.2003 | Annisa Akhwat | Jl. Mawar   | W    |
| 01.05.2002 | Abu Bakar     | Jl. Tunjung | P    |
| 01.05.2001 | Laksmi dewi   | Jl. Balapan | W    |
| 01.05.2000 | Budi R        | Jl. Kaliurang | P    |
+-----+-----+-----+-----+
5 rows in set (0.11 sec)
```

Mengubah Data (Update)

Jika data sudah kadaluarsa, berikut perintah digunakan untuk merubah data.

Struktur penulisan :

```
Update nama_tabel set nama_field_1=isi_baru_1, nama_field_2=
isi_baru_2,..., nama_field_n=isi_baru_n
Where kriteria;
```

Contoh :

```
Update produk set harga=3000 where nama='Sikat Gigi';
```

OPERATOR RELASIONAL

Operator yang digunakan untuk perbandingan antara dua buah nilai. Jenis dari operator ini adalah =, >, <, >=, <=, <>

```
mysql> select * from siswa where sex='P';
+-----+-----+-----+-----+
| NIM      | Nama      | alamat      | Sex  |
+-----+-----+-----+-----+
| 01.05.2000 | Budi R    | Jl. Kaliurang | P    |
| 01.05.2002 | Abu Bakar | Jl. Tunjung  | P    |
| 01.05.2004 | Leon Auman | Jl. depok    | P    |
+-----+-----+-----+-----+

3 rows in set (0.06 sec)
```

Operator BETWEEN dan NOT BETWEEN

Operator between ini untuk menangani operasi “jangkauan”

```
mysql> select * from siswa where tgl_lhr between '1980-01-01' and '1982-12-29';
+-----+-----+-----+-----+-----+
| NIM      | Nama      | alamat      | Sex  | Tgl_lhr  |
+-----+-----+-----+-----+-----+
| 01.05.2000 | Budi R    | Jl. Kaliurang | P    | 1982-03-17 |
| 01.05.2001 | Laksmi dewi | Jl. Balapan  | W    | 1982-10-21 |
| 01.05.2003 | Annisa Akhwat | Jl. Mawar    | W    | 1980-01-12 |
+-----+-----+-----+-----+-----+
```

Operator IN dan NOT IN

Operator IN digunakan untuk mencocokkan dengan salah satu yang ada pada suatu daftar nilai. Sebagai contoh :

```
mysql> select nama from siswa where nim in ('01.05.2000','01.05.2004');
+-----+
| nama      |
+-----+
| Budi R    |
| Leon Auman |
+-----+

2 rows in set (0.00 sec)
```

Operator LIKE dan NOT LIKE

Operator LIKE atau NOT LIKE sangat bermanfaat dalam mencari suatu data. Operasi ini digunakan dengan menyebutkan tanda wildcard berupa garis bawah (_) atau (%)

<. Tanda garis bawah (_) berarti sebuah karakter apa saja. Contoh a_u cocok dengan anu,aku,alu,abu dan tidak cocok untuk andu,ambu ataupun allu

<. Tanda % berarti cocok dengan kata apa saja dan berapapun panjangnya

contoh:

```
mysql> select nim,nama,alamat from siswa where nama like 'a%';
+-----+-----+-----+
| nim      | nama      | alamat  |
+-----+-----+-----+
| 01.05.2002 | Abu Bakar  | Jl. Tunjung |
| 01.05.2003 | Annisa Akhwat | Jl. Mawar  |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

FUNGSI AGREGAT

Fungsi Agregat	Keterangan
AVG	Memperoleh nilai rata-rata
COUNT	Menghitung cacah data
MAX	Menghasilkan nilai terbesar
MIN	Menghasilkan nilai terkecil
SUM	Memperoleh penjumlahan data

Contoh :

```
mysql> select avg(nilai) from nilai;
+-----+
| avg(nilai) |
+-----+
|      63.0000 |
+-----+
1 row in set (0.05 sec)
```

```
mysql> select count(nama) from siswa;
+-----+
| count(nama) |
+-----+
|           5 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select max(nilai),min(nilai) from nilai;
+-----+-----+
| max(nilai) | min(nilai) |
+-----+-----+
|          100 |          10 |
+-----+-----+
1 row in set (0.05 sec)
```

```
mysql> select sum(nilai) from nilai;
```

```
+-----+
| sum(nilai) |
+-----+
|          315 |
+-----+
1 row in set (0.05 sec)
```

Ekspresi Pada Query

Ekspresi pada query memungkinkan untuk mengubah judul kolom keluaran, melakukan perhitungan pada kolom keluaran, dan mengubah tipe data keluaran.

- Memberi nama lain terhadap kolom keluaran

Untuk memberi nama lain terhadap suatu kolom keluaran dengan klausa AS.

```
mysql> select nim,nilai*10 as Nilai_MHS from nilai;
+-----+-----+
| nim      | Nilai_MHS |
+-----+-----+
| 01.05.2000 |          900 |
| 01.05.2001 |          400 |
| 01.05.2002 |          750 |
| 01.05.2003 |          100 |
| 01.05.2004 |         1000 |
+-----+-----+
5 rows in set (0.00 sec)
```

- Menambah teks dalam baris keluaran

```
mysql> select 'No Mahasiswa ',nim,' Nilainya ',nilai from nilai;
+-----+-----+-----+-----+
| No Mahasiswa | nim      | Nilainya | nilai |
+-----+-----+-----+-----+
| No Mahasiswa | 01.05.2000 | Nilainya | 90 |
| No Mahasiswa | 01.05.2001 | Nilainya | 40 |
| No Mahasiswa | 01.05.2002 | Nilainya | 75 |
| No Mahasiswa | 01.05.2003 | Nilainya | 10 |
| No Mahasiswa | 01.05.2004 | Nilainya | 100 |
+-----+-----+-----+-----+
```

- Ekpresi berkondisi

BU : CASE nilai_ekspresi WHEN nilai_ekspresi THEN nilai_ekspresi

ELSE nilai_ekspresi END

```
mysql> select nim,nama,case sex when 'P' then 'Pria' else 'Wanita'
end as sex from siswa;
```

nim	nama	sex
01.05.2000	Budi R	Pria
01.05.2001	Laksmi dewi	Wanita
01.05.2002	Abu Bakar	Pria
01.05.2003	Annisa Akhwat	Wanita
01.05.2004	Leon Auman	Pria

```
5 rows in set (0.06 sec)
```

- Ekspresi tentang Waktu

Pernyataan berikut akan menampilkan identitas siswa yang tanggal lahirnya lebih dari tanggal 29 desember 1980

```
mysql> select nim,nama,tgl_lhr from siswa where tgl_lhr>date '1980-
12-29';
```

nim	nama	tgl_lhr
01.05.2000	Budi R	1982-03-17
01.05.2001	Laksmi dewi	1982-10-21
01.05.2004	Leon Auman	1983-01-19

```
3 rows in set (0.00 sec)
```

```
mysql> select time '23:12:00'-time '05:00:00';
```

time '23:12:00'-time '05:00:00'
18

```
1 row in set (0.06 sec)
```

- FUNGSI NILAI

CURRENT_DATE : Menghasilkan tanggal sekarang

CURRENT_TIME : Menghasilkan Waktu sekarang

```
mysql> select current_date as Tgl_Sekarang;
```

Tgl_Sekarang
2003-10-08

```
1 row in set (0.06 sec)
```

```
mysql> select current_time as Waktu_Sekarang;
+-----+
| Waktu_Sekarang |
+-----+
| 06:45:14      |
+-----+
1 row in set (0.06 sec)
```

QUERY ANTAR TABEL

Query yang akan dibahas pada bagian ini melibatkan lebih dari satu tabel

Field	Type	Null	Key	Default	Extra
NIM	varchar(10)		PRI		
Nama	varchar(25)	YES		NULL	
alamat	varchar(35)	YES		NULL	
Sex	char(1)	YES		P	
Tgl_lhr	date	YES		NULL	

Struktur dari table Siswa

NIM	Nama	alamat	Sex	Tgl_lhr
01.05.2000	Budi R	Jl. Kaliurang	P	1982-03-17
01.05.2001	Laksmi dewi	Jl. Balapan	W	1982-10-21
01.05.2002	Abu Bakar	Jl. Tunjung	P	1977-12-29
01.05.2003	Annisa Akhwat	Jl. Mawar	W	1980-01-12
01.05.2004	Leon Auman	Jl. depok	P	1983-01-19

Isi dari table Siswa

Field	Type	Null	Key	Default	Extra
NIM	varchar(10)		PRI		
Nilai	int(11)	YES		0	

Struktur dari table Siswa

NIM	Nilai
01.05.2000	90
01.05.2001	40
01.05.2002	75
01.05.2003	10
01.05.2004	100

Isi dari table Siswa

MySQL mempunyai kemampuan untuk menggabungkan dua tabel atau lebih guna mendapatkan informasi yang diinginkan. Proses yang dilakukan dengan nama JOIN.

Dalam penggabungan dari beberapa tabel (join) ada beberapa hal yang perlu diperhatikan, antara lain :

1. Setiap kolom disebutkan dengan bentuk,
2. tabel-tabel yang dilibatkan dalam query perlu disebutkan dalam Klausa FROM dengan antar tabel dipisah oleh koma.
3. Kondisi dalam WHERE menentukan macam join yang terbentuk

Contoh:

```
mysql> select siswa.nim,nama,nilai from nilai,siswa where
nilai.nim=siswa.nim;
+-----+-----+-----+
| nim      | nama      | nilai |
+-----+-----+-----+
| 01.05.2000 | Budi R    | 90    |
| 01.05.2001 | Laksmi dewi | 40    |
| 01.05.2002 | Abu Bakar | 75    |
| 01.05.2003 | Annisa Akhwat | 10    |
| 01.05.2004 | Leon Auman | 100   |
+-----+-----+-----+
```

Macam-macam bentuk Penggabungan (Join)

1. CROSS JOIN

Cross Join merupakan bentuk penggabungan yang paling sederhana, tanpa ada kondisi.

BU : SELECT field1,field2 FROM Tabel1 CROSS JOIN tabel2;

2. INNER JOIN

Hampir sama dengan cross join tetapi diikuti dengan kondisi

BU : SELECT Field FROM tabel1 INNER JOIN tabel 2 ON kondisi


```
mysql> select * from siswa inner join nilai on siswa.nim=nilai.nim;
```

NIM	Nama	alamat	Sex	Tgl_lhr	NIM	Nilai
01.05.2000	Budi R	Jl. Kaliurang	P	1982-03-17	01.05.2000	90
01.05.2001	Laksmi dewi	Jl. Balapan	W	1982-10-21	01.05.2001	40
01.05.2002	Abu Bakar	Jl. Tunjung	P	1977-12-29	01.05.2002	75
01.05.2003	Annisa Akhwat	Jl. Mawar	W	1980-01-12	01.05.2003	10
01.05.2004	Leon Auman	Jl. depok	P	1983-01-19	01.05.2004	100

3. STRAIGHT JOIN

Straight Join identik dengan inner join tetapi tidak mengenal klausa where

BU : SELECT field FROM Tabel1 SATRIGHT JOIN tabel2

4. LEFT (OUTER) JOIN

Akan menampilkan tabel disebelah kanannya (dalam hal ini tabel nilai) dengan

NULL jika tidak terdapat hubungan antara tabel disebelah kiri (dalam hal ini tabel siswa)

BU : SELECT field FROM tabel1 LEFT JOIN tabel2 ON kondisi

Contoh

```
mysql> select siswa.nim,nama,nilai from siswa left join nilai on
siswa.nim=nilai
.nim;
```

nim	nama	nilai
01.05.2000	Budi R	90
01.05.2001	Laksmi dewi	40
01.05.2002	Abu Bakar	75
01.05.2003	Annisa Akhwat	10
01.05.2004	Leon Auman	100

5. RIGHT/OUTER JOIN

Kebalikan dari LEFT JOIN, jika idak menemukan hubungan dengan tabel disebelah kiri maka akan ditampilkan NULL

BU : SELECT field FROM tabel1 RIGHT JOIN tabel2 ON kondisi

Contoh

```
mysql> select siswa.nim,nama,nilai from siswa right join nilai  
on siswa.nim=nilai.nim;
```

nim	nama	nilai
01.05.2000	Budi R	90
01.05.2001	Laksmi dewi	40
01.05.2002	Abu Bakar	75
01.05.2003	Annisa Akhwat	10
01.05.2004	Leon Auman	100
NULL	NULL	68

```
6 rows in set (0.06 sec)
```

Menghapus Data (*Delete*)

Jika terdapat kesalahan data atau data sudah tidak digunakan lagi, hapus dengan perintah *delete*.

Struktur penulisan :

Delete from nama_tabel where kriteria;

Contoh :

```
Delete from produk where nama='Sikat Gigi',
```