

BAB 4. DATA MANIPULATION LANGUAGE (DML)

4.1 TUJUAN PRAKTIKUM

Tujuan Umum

Mahasiswa mampu memahami dan menerapkan Kueri SQL – Data Manipulation Language (DML)

Tujuan Khusus

Mahasiswa mampu:

- Menerapkan Kueri SQL - DML untuk Menyisipkan Data
- Menerapkan Kueri SQL - DML untuk Mengambil Data
- Menerapkan Kueri SQL - DML untuk Mengubah Data
- Menerapkan Kueri SQL - DML untuk Menghapus Data

4.2 TEORI SINGKAT

a. Data Manipulation Language (DDL)

DDL merupakan kumpulan perintah SQL yang digunakan untuk melakukan penyisipan informasi baru (Insert), pengambilan data (Select), mengubah data (Update), menghapus data (Delete).

b. Menyisipkan data (Insert)

Insert merupakan perintah yang berfungsi untuk menyisipkan data baru dari luar sistem ke dalam tabel pada suatu basis data.

- Menambahkan data baru pada semua kolom yang ada pada tabel

```
INSERT INTO nama_tabel  
VALUES (nilai1, nilai2, ...);
```

Contoh:

```
insert into mahasiswa(nim, nama, angkatan, umur, jurusan, email)  
values ("1001001", "DZAMAAR", 2020, 18, "Teknik Informatika",  
"dzamaar@mamamail.com");
```

- Menambahkan data baru hanya pada kolom tertentu pada tabel

```
INSERT INTO nama_tabel (kolom1, kolom2,...)  
VALUES (nilai1, nilai2,...);
```

Contoh:

```
insert into mahasiswa(nim, nama, angkatan, umur, jurusan, email)
values ("1001001", "DZAMAAR", 2020, 18, "Teknik Informatika",
"dzamaar@mamamail.com");
```

- Menambahkan data baru lebih dari satu data sekaligus

```
INSERT INTO namatabel (kolom1, kolom2,...)
VALUES (nilai11, nilai12,...), (nilai21, nilai22,...),
(nilai31, nilai32,...), ..... ;
```

Contoh:

```
insert into mahasiswa values
("1001003", "ZAFRAN", 2020, 18, "Teknik Informasi", "zafran@mail.com"),
("1001004", "ZASKIA", 2020, 18, "Teknik Mesin", "zaskia@mamamail.com"),
("1001005", "ZIDNI", 2020, 19, "Teknik Mesin", "zidni@mamamail.com");
```

c. Mengambil data (Select)

Select digunakan untuk melakukan pengambilan sejumlah data yang ada di dalam tabel untuk ditampilkan ataupun dimanipulasi.

- Menampilkan seluruh isi data pada tabel

```
SELECT * FROM nama_tabel
```

Contoh:

```
mysql> select * from mahasiswa;
```

| nim | nama | angkatan | umur | jurusan | alamat | email |
|---------|--------|----------|------|--------------------|--------|---------------------|
| 1001003 | ZAFRAN | 2020 | 18 | Teknik Informatika | NULL | zafran@mamamail.com |
| 1001004 | ZASKIA | 2020 | 18 | Teknik Mesin | NULL | zaskia@mamamail.com |
| 1001005 | ZIDNI | 2020 | 19 | Teknik Mesin | NULL | zidni@mamamail.com |

Gambar 4.1 Hasil tampilan data tampil mahasiswa

- Menampilkan isi data pada kolom tertentu

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
```

Contoh:

```
mysql> select nim, nama from mahasiswa;
+-----+-----+
| nim    | nama  |
+-----+-----+
| 1001003 | ZAFRAN |
| 1001004 | ZASKIA |
| 1001005 | ZIDNI  |
+-----+-----+
```

Gambar 4.2 Hasil tampilan kolom tertentu pada tabel

- Menampilkan isi data pada tabel dengan satu kondisi tertentu

```
SELECT kolom1, kolom2, kolomN
FROM nama_tabel
WHERE kolom=nilai
```

Contoh:

```
mysql> select nim, nama, jurusan from mahasiswa where nim=1001003;
+-----+-----+-----+
| nim    | nama  | jurusan          |
+-----+-----+-----+
| 1001003 | ZAFRAN | Teknik Informatika |
+-----+-----+-----+
1 row in set (0.00 sec)
```

Gambar 4.3 Hasil tampilan kolom tertentu dengan kondisi tertentu pada tabel

- Menampilkan isi data pada tabel dengan satu kondisi tertentu (**NOT**)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE NOT kolom=nilai
```

Contoh:

```
mysql> select nim, nama, jurusan from mahasiswa where not jurusan="Teknik Informatika";
+-----+-----+-----+
| nim    | nama  | jurusan          |
+-----+-----+-----+
| 1001004 | ZASKIA | Teknik Mesin    |
| 1001005 | ZIDNI  | Teknik Mesin    |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Gambar 4.4 Hasil tampilan kolom tertentu dengan kondisi tertentu (NOT) pada tabel

- Menampilkan isi data pada tabel dengan satu kondisi NULL atau NOT NULL (IS)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE kolom IS NULL
```

Contoh:

```
mysql> select nim, nama, jurusan from mahasiswa where alamat is null;
+-----+-----+-----+
| nim    | nama  | jurusan          |
+-----+-----+-----+
| 1001003 | ZAFRAN | Teknik Informatika |
| 1001004 | ZASKIA | Teknik Mesin      |
| 1001005 | ZIDNI  | Teknik Mesin      |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Gambar 4.5 Hasil tampilan kolom tertentu dengan kondisi data pada kolom ada yang null pada tabel

- Menampilkan isi data pada tabel dengan lebih dari satu kondisi (OR)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE kolom1=nilai1 OR kolom2=nilai2
```

Contoh:

```
mysql> select nim, nama, jurusan from mahasiswa where nim=1001003 or nama="ZIDNI";
+-----+-----+-----+
| nim    | nama  | jurusan          |
+-----+-----+-----+
| 1001003 | ZAFRAN | Teknik Informatika |
| 1001005 | ZIDNI  | Teknik Mesin      |
+-----+-----+-----+
```

Gambar 4.6 Hasil tampilan kolom tertentu dengan lebih dari satu kondisi pada tabel (OR)

- Menampilkan isi data pada tabel dengan lebih dari satu kondisi (AND)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE kolom1=nilai1 AND kolom2=nilai2
```

Contoh:

```
mysql> select nim, nama, angkatan from mahasiswa where angkatan=2020 and umur=18;
+-----+-----+-----+
| nim    | nama  | angkatan |
+-----+-----+-----+
| 1001003 | ZAFRAN | 2020    |
| 1001004 | ZASKIA | 2020    |
+-----+-----+-----+
```

Gambar 4.7 Hasil tampilan kolom tertentu dengan lebih dari satu kondisi pada tabel (AND)

- Menampilkan isi data pada tabel dengan lebih dari dua kondisi gabungan (**OR**) dan (**AND**)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE (kolom1=nilai1 or kolom2=nilai2) AND
      kolom3=nilai3
```

Contoh:

```
mysql> select * from mahasiswa where (umur=18 or jurusan="Teknik Informatika") and angkatan=2020;
+-----+-----+-----+-----+-----+-----+-----+
| nim    | nama  | angkatan | umur | jurusan          | alamat | email                |
+-----+-----+-----+-----+-----+-----+-----+
| 1001003 | ZAFRAN | 2020    | 18   | Teknik Informatika | NULL   | zafran@mamamail.com |
| 1001004 | ZASKIA | 2020    | 18   | Teknik Mesin      | NULL   | zaskia@mamamail.com |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Gambar 4.6 Hasil tampilan kolom tertentu dengan lebih dari dua kondisi pada tabel OR dan AND

- Menampilkan isi data pada tabel hanya pada nilai yang berbeda (**DISTINCT**)

```
SELECT DISTINCT kolom
FROM namatabel
```

Contoh:

```
mysql> select distinct jurusan from mahasiswa;
+-----+
| jurusan |
+-----+
| Teknik Informatika |
| Teknik Mesin      |
+-----+
2 rows in set (0.00 sec)
```

Gambar 4.6 Hasil tampilan kolom distinct dari suatu tabel

- Menampilkan isi data pada tabel dengan kondisi nilai mirip atau sebagian mirip (**LIKE**)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE kolom LIKE <OPERATOR>
```

Tabel 4.1 Operator LIKE dan Keterangannya

| Operator | Keterangan |
|----------|--|
| 'a%' | Nilai kolom dimulai huruf "a" |
| '%a' | Nilai kolom diakhiri huruf "a" |
| '%a%' | Nilai kolom mengandung huruf "a" |
| '_a%' | Nilai kolom mengandung huruf "a" pada posisi kedua |
| 'a_%' | Nilai kolom diawali huruf "a" dan memiliki minimal 2 karakter pada nilai kolom |
| 'a__%' | Nilai kolom diawali huruf "a" dan memiliki minimal 3 karakter pada nilai kolom |
| 'a%o' | Nilai kolom diawali huruf "a" dan diakhiri huruf "z" |

Contoh:

```
mysql> select nim, nama, umur from mahasiswa where nama like '%a';
+-----+-----+-----+
| nim    | nama   | umur |
+-----+-----+-----+
| 1001004 | ZASKIA | 18    |
+-----+-----+-----+
1 row in set (0.00 sec)
```

Gambar 4.7 Hasil tampilan data yang mengandung kata tertentu

- Menampilkan isi data pada tabel dengan kondisi diantara dua nilai (**BETWEEN**)

```
SELECT kolom1, kolom2, kolomN
FROM namatabel
WHERE kolom BETWEEN nilai1 AND nilai2
```

Contoh:

```
mysql> select nim, nama, umur from mahasiswa where umur between 18 and 20;
+-----+-----+-----+
| nim    | nama   | umur |
+-----+-----+-----+
| 1001003 | ZAFRAN | 18    |
| 1001004 | ZASKIA | 18    |
| 1001005 | ZIDNI  | 19    |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Gambar 4.8 Hasil tampilan data di dalam dua nilai tertentu

- Menampilkan isi data pada tabel dengan kondisi banyak nilai (**IN**)

```
SELECT kolom1, kolomN
FROM namatabel
WHERE umur IN (nilai1, nilai2, nilaiN)
```

Contoh:

```
mysql> select nim, nama, umur from mahasiswa where umur in (19, 20, 21);
+-----+-----+-----+
| nim    | nama   | umur |
+-----+-----+-----+
| 1001005 | ZIDNI  | 19    |
+-----+-----+-----+
1 row in set (0.00 sec)
```

Gambar 4.9 Hasil tampilan data yang termasuk salah satu nilai yang dinyatakan sebelumnya

- Menampilkan isi data pada tabel dengan batasan jumlah tertentu (**LIMIT**)

```
SELECT kolom1, kolomN
FROM namatabel LIMIT N
```

Contoh:

```
mysql> select nim, nama, umur from mahasiswa limit 2;
+-----+-----+-----+
| nim    | nama  | umur |
+-----+-----+-----+
| 1001003 | ZAFRAN | 18   |
| 1001004 | ZASKIA | 18   |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Gambar 4.10 Hasil tampilan data dengan pembatasan tertentu

d. Mengubah data (Update)

Update digunakan untuk mengubah data lama menjadi data baru. Perlu diperhatikan ketika melakukan proses update adalah bagian pemilihan data mana yang akan diubah, jika tidak ada kondisi tertentu yang ditambahkan maka proses update akan dilakukan pada semua data pada tabel yang dipilih.

```
UPDATE nama_tabel
SET kolom1=nilaibaru1, kolomN=nilaibaruN
WHERE <KONDISI>
```

Contoh tanpa kondisi:


```
mysql> update mahasiswa set alamat = "Indramayu";
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> select nim, nama, alamat from mahasiswa;
+-----+-----+-----+
| nim    | nama  | alamat  |
+-----+-----+-----+
| 1001003 | ZAFRAN | Indramayu |
| 1001004 | ZASKIA | Indramayu |
| 1001005 | ZIDNI  | Indramayu |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Gambar 4.11 Hasil tampilan update data

Contoh dengan kondisi:

```
mysql> update mahasiswa set alamat="Cirebon" where nim="1001003";
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select nim, nama, alamat from mahasiswa;
+-----+-----+-----+
| nim    | nama  | alamat  |
+-----+-----+-----+
| 1001003 | ZAFRAN | Cirebon  |
| 1001004 | ZASKIA | Indramayu |
| 1001005 | ZIDNI  | Indramayu |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Gambar 4.12 Hasil tampilan update data dengan kondisi tertentu

e. Menghapus data (Delete)

Delete digunakan untuk menghapus data pada suatu tabel. Sama seperti proses update, pada proses delete juga perlu memperhatikan bagian pemilihan data mana yang akan dihapus, jika tidak ada kondisi tertentu yang ditambahkan maka proses delete akan dilakukan pada semua data pada tabel yang dipilih.

```
DELETE FROM nama_tabel
```

| |
|-----------------|
| WHERE <KONDISI> |
|-----------------|

Contoh:

```
mysql> delete from mahasiswa where nim=1001003;
Query OK, 1 row affected (0.00 sec)

mysql> select nim, nama, alamat from mahasiswa;
+-----+-----+-----+
| nim   | nama  | alamat |
+-----+-----+-----+
| 1001004 | ZASKIA | Indramayu |
| 1001005 | ZIDNI  | Indramayu |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Gambar 4.11 Hasil tampilan delete data

4.3 PELAKSANAAN PRAKTIKUM

Langkah-langkah praktikum online dengan MySQL Terminal:

1. Buka aplikasi Browser Web (misal: Google Chrome atau Mozilla FireFox)
2. Ketikkan pada kolom url <http://elearning.ti.polindra.ac.id>, kemudian silahkan login ke akun elearning anda.
3. Pilih menu MySQL Terminal dan silahkan ketikkan “mysql login” lalu tekan tombol **Enter**.
4. Gunakan database `nimanda_kampus` yang sudah dibuat pada praktikum sebelumnya.

5. Ketikkan perintah dibawah ini untuk menyisipkan data ke tabel mahasiswa:

```
insert into mahasiswa values
("1001003", "ZAFRAN", 2020, 18, "Teknik Informasi", "zafran@mail.com"),
("1001004", "ZASKIA", 2020, 18, "Teknik Mesin", "zaskia@mamamail.com"),
("1001005", "ZIDNI", 2020, 19, "Teknik Mesin", "zidni@mamamail.com");
```

6. Ketikkan perintah `select * from mahasiswa` untuk melihat data pada tabel mahasiswa, seperti terlihat pada Gambar 4.1.
7. Ketikkan perintah `select nim, nama from mahasiswa` untuk melihat data kolom nim dan nama pada tabel mahasiswa, seperti terlihat pada Gambar 4.2.
8. Ketikkan perintah `update mahasiswa set alamat="indramayu" ` untuk mengubah data sehingga kolom alamat terisi dengan data “indramayu”.

9. Ketikkan perintah `select nim, nama, alamat from mahasiswa` untuk melihat data kolom nim, nama dan alamat yang telah diupdate pada tabel mahasiswa, seperti terlihat pada Gambar 4.3.
10. Silahkan buka menu log kueri dan pastikan perintah SQL yang sudah diketik di MySQL Terminal sudah ada.
11. Selesai.

```
mysql> select * from mahasiswa;
```

| nim | nama | angkatan | umur | jurusan | alamat | email |
|---------|--------|----------|------|--------------------|--------|---------------------|
| 1001003 | ZAFRAN | 2020 | 18 | Teknik Informatika | NULL | zafran@mamamail.com |
| 1001004 | ZASKIA | 2020 | 18 | Teknik Mesin | NULL | zaskia@mamamail.com |
| 1001005 | ZIDNI | 2020 | 19 | Teknik Mesin | NULL | zidni@mamamail.com |

Gambar 4.1 Hasil tampilan isi data tabel mahasiswa

```
mysql> select nim, nama from mahasiswa;
```

| nim | nama |
|---------|--------|
| 1001003 | ZAFRAN |
| 1001004 | ZASKIA |
| 1001005 | ZIDNI |

Gambar 4.2 Hasil tampilan kolom nim dan nama pada tabel mahasiswa

```
mysql> update mahasiswa set alamat = "Indramayu";
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> select nim, nama, alamat from mahasiswa;
```

| nim | nama | alamat |
|---------|--------|-----------|
| 1001003 | ZAFRAN | Indramayu |
| 1001004 | ZASKIA | Indramayu |
| 1001005 | ZIDNI | Indramayu |

3 rows in set (0.00 sec)

Gambar 4.3 Hasil tampilan setelah update data mahasiswa

4.4 LATIHAN

1. Berdasarkan tabel barang yang sudah dibuat pada latihan sebelumnya.

Ketikkan perintah kueri SQL berikut:

```
insert into barang values
(null, 'Beras', 10000, 'B01'),
(null, 'Gula', 10000, 'B02'),
(null, 'Garam', 10000, 'B03'),
(null, 'Mie', 10000, 'B04'),
(null, 'Sabun', 10000, 'B05'),
(null, 'Deterjen', 10000, 'B06'),
(null, 'Minyak Tanah', 10000, 'B07'),
(null, 'Minyak Goreng', 10000, 'B08'),
(null, 'Roti', 10000, 'B09'),
(null, 'Sirup', 10000, 'B10');

select * from barang;
```

Tunjukkan hasilnya dengan cara ambil gambar atau screenshot hasil kueri tersebut!

2. Ketikkan perintah kueri SQL berikut:

```
update barang set barang_nama='Roti A' where
barang_kode='B09';
update barang set barang_harga=5000 where
barang_kode='B04';
update barang set barang_kode='B11' where
barang_kode='B03';
update barang set barang_harga=6000 where
barang_kode='B06';
update barang set barang_harga=8000 where
barang_kode='B02' OR barang_kode='B01';

select * from barang;
```

Tunjukkan hasilnya dengan cara ambil gambar atau screenshot hasil kueri tersebut!

3. Ketikkan perintah kueri SQL berikut:

```
select barang_kode, barang_nama from barang
where barang_nama LIKE "%i%";

select * from barang;
```

Tunjukkan hasilnya dengan cara ambil gambar atau screenshot hasil kueri tersebut!

4.5 TUGAS

1. Berdasarkan tabel karyawan yang sudah dibuat pada tugas sebelumnya yang terdiri dari 5 kolom. Buatlah kueri SQL untuk melakukan penyisipan data ke dalam tabel karyawan tersebut sebanyak 10 data!
2. Buatlah 5 kueri SQL untuk mengambil data pada tabel karyawan dengan variasi nilai WHERE (kondisi) yang berbeda!
3. Buatlah 5 kueri SQL untuk mengubah data pada tabel karyawan dengan variasi nilai SET dan WHERE (kondisi) yang berbeda!
4. Buatlah 5 kueri SQL untuk menghapus data pada tabel karyawan dengan variasi nilai WHERE (kondisi) yang berbeda!