

## MODUL 3

### SQL

#### A. TUJUAN

- Memahami SQL, komponen-komponen, dan penggunaannya.
- Mengenal operasi-operasi DML menggunakan phpmyadmin.
- Mampu melakukan operasi-operasi *retrieve* yang spesifik
- Mampu penguraikan dan mentranslasikan pernyataan-pernyataan SQL.

#### B. PETUNJUK

- Awali setiap aktivitas dengan do'a, semoga berkah dan mendapatkan kemudahan.
- Pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik dan benar.
- Kerjakan tugas-tugas praktikum dengan baik, sabar, dan jujur.
- Tanyakan kepada asisten/dosen apabila ada hal-hal yang kurang jelas.

#### C. DASAR TEORI

##### 1. SQL (Structured Query Language )

SQL (atau query language) adalah sebuah bahasa yang digunakan untuk me-request informasi dari basis data; bahasa ini sudah disediakan oleh DBMS. Konstruksi SQL terbentuk atas kombinasi *relational-algebra* dan *relational-calculus*.

SQL terdiri dari beberapa komponen, dua di antaranya yang sangat fundamental adalah DDL (Data Definition Language) dan DML (Data Manipulation Language).

##### 2. DML (Data Manipulation Language)

DML menyediakan perintah-perintah untuk menambah, memodifikasi, menghapus, dan mengambil data di dalam basis data.

Contoh perintah DML adalah : INSERT, UPDATE, DELETE, dan SELECT

##### Penambahan Data

Sintaks operasi penambahan data dapat dituliskan sebagai berikut:

```
INSERT INTO r (A1, A2, ..., An)
VALUES (V1,V2,..., An)
```

Dimana :

r adalah relasi (tabel)

A adalah atribut (*field* atau kolom)

V adalah value (nilai yang diisikan)

Misalkan tabel jurusan memiliki *field* id dan nama, maka pernyataan SQL untuk penambahan data dituliskan sebagai berikut :

```
INSERT INTO jurusan (id,)
VALUES (V1,V2,..., An)
```

### **Modifikasi Data**

Sintaks operasi modifikasi data dapat dituliskan sebagai berikut :

```
UPDATE r
SET A1-V1, A2-V2, ..., An-Vn
WHERE P
```

Di mana :

P adalah predikat

Misalkan nilai field nama untuk data dengan id 1 ingin diubah menjadi Teknik Informatika, berikut pernyataan SQL-nya:

```
UPDATE jurusan
SET nama = "Teknik Informatika"
WHERE id = 1
```



Perhatikan, operasi modifikasi tanpa menyertakan klausa WHERE akan mengakibatkan diubahnya seluruh data di table. Jadi, jika Anda hanya ingin memodifikasi data tertentu gunakan selalu klausa WHERE

### **Penghapusan Data**

Sintaks operasi penghapusan data dapat dituliskan sebagai berikut :

```
DELETE FROM r
WHERE P
```

Misalkan nilai data dengan id ingin dihapus, pernyataan SQL-nya dituliskan sebagai berikut :

```
DELETE FROM jurusan
WHERE id = 1
```

Perhatikan juga, operasi penghapusan tanpa menyertakan klausa WHERE akan mengakibatkan dihapusnya seluruh data di table. Jadi, jika Anda hanya ingin menghapus data tertentu gunakan selalu klausa WHERE.

### **Seleksi Data**

Sintaks operasi seleksi atau pengambilan data dapat dituliskan sebagai berikut :

```
SELECT A1, A2, ..., An
FROM r1, r2, r3, ..., rn
WHERE P
```

Misalkan ingin menampilkan data id dan nama jurusan, pernyataan SQL-nya dituliskan sebagai berikut :

```
SELECT id, nama  
FROM jurusan
```

Sebagai alternative, tersedia karakter asterik (\*) yang menyatakan seluruh field. Jadi, pernyataan di bawah ini ekuivalen dengan pernyataan di atas.

```
SELECT *  
FROM jurusan
```



Untuk efisiensi, karakter asterik sebaiknya digunakan jika ingin me-retrieve seluruh nilai field

## D. LATIHAN

### 1. Menciptakan Objek Query

1. Pada halaman awal PHPMyadmin, pilih menu **SQL**.
2. Maka akan muncul halaman editor SQL. Di Editor SQL inilah kita menuliskan pernyataan SQL. Sebagai contoh isikan pernyataan berikut:



3. Eksekusi pernyataan SQL dengan klik tombol “GO”



Untuk efisiensi penulisan SQL, gunakan editor teks (misalnya notepad). Selanjutnya, pernyataan SQL dapat di-copy ke editor SQL dan dimodifikasi untuk operasi lain.

## 2. Menciptakan dan Menghapus Tabel

Sebagai contoh, buat table jurusan dengan struktur sebagai berikut:

No	Nama Field	Tipe data	Keterangan
1	Kd_jurusan	Integer	Primary Key
2	Nama	Varchar	Nama Jurusan

Langkah-langkah :

1. Klik Database “modul2” yang telah dibuat pada latihan 1.
2. Tuliskan Pernyataan SQL berikut :

```
1 CREATE TABLE jurusan(  
2     kd_jurusan INT,  
3     nama_jurusan VARCHAR(20),  
4     PRIMARY KEY(kd_jurusan)  
5 );
```

3. Eksekusi pernyataan SQL.
4. Periksa hasil pembuatan tabel pada sidebar.

Dalam contoh di atas sengaja digunakan tipe data INT dan VARCHAR karena lebih general. Sebenarnya penggunaan tipe data NUMBER dan TEXT juga diterima, namun kurang standar.

## 3. Menciptakan dan Menghapus Tabel

Untuk menghapus tabel, ikuti langkah-langkah berikut:

1. Klik Database “modul2” yang telah dibuat pada latihan 1.
2. Buat Pernyataan SQL seperti berikut

```
1 DROP TABLE jurusan;
```

3. Eksekusi pernyataan SQL.

## 4. Penambahan Data

1. Klik Database “modul2” yang telah dibuat pada latihan 1.
2. Buat Pernyataan SQL seperti berikut :

```
1 INSERT into jurusan  
2     (kd_jurusan, nama_jurusan)  
3     VALUES (1, "Teknik Elektro");  
4
```

3. Eksekusi pernyataan SQL

## 5. Modifikasi Data

1. Klik Database “modul2” yang telah dibuat pada latihan 1.
2. Buat Pernyataan SQL seperti berikut :

```
1 UPDATE jurusan  
2     SET nama_jurusan = "Teknik Informatika"  
3     WHERE kd_jurusan = 1;
```

3. Eksekusi pernyataan SQL.

## 6. Seleksi Data

1. Klik Database “modul2” yang telah dibuat pada latihan 1.
2. Buat Pernyataan SQL seperti berikut :

```
1 SELECT kd_jurusan, nama_jurusan
2     FROM jurusan;
3
```

3. Eksekusi pernyataan SQL.

## 7. Seleksi dengan Kriteria

Manfaatkan kembali tabel matakuliah yang sudah dibuat sebelumnya pada pertemuan pertama. Selanjutnya, isikan data sebagai berikut:

kode_mk	nama_mk	sks	semester
PTI447	Praktikum Basis Data	1	3
TIK342	Praktikum Basis Data	1	3
PTI333	Basis Data Terdistribusi	3	5
TIK123	Jaringan Komputer	2	5
TIK333	Sistem Operasi	3	5
PTI123	Grafika Multimedia	3	5
PTI777	Sistem Informasi	2	3

1. Dapatkan kode\_mk, nama\_mk, dan semester yang sksnya 3.

```
1 SELECT * FROM `matakuliah` WHERE sks=3
2
```

Hasil :

kode_mk	nama_mk	sks	semester
PTI123	Grafika Multimedia	3	5
PTI333	Basis Data Terdistribusi	3	5
TIK333	Sistem Operasi	3	5

2. Dapatkan kode\_mk, nama\_mk, sks, dan semester yang 2 dan semester 3.

```
1 SELECT * FROM `matakuliah` WHERE sks=2 AND semester=3
2
```

Hasil:

kode_mk	nama_mk	sks	semester
PTI777	Sistem Informatika	2	3

3. Dapatkan kode\_mk, nama\_mk, sks, dan semester yang sksnya kurang dari sama dengan 2.

```
1 SELECT * FROM `matakuliah` WHERE sks <= 2
2
```

Hasil :

kode_mk	nama_mk	sks	semester
PTI447	Praktikum Basis Data	1	3
PTI777	Sistem Informatika	2	3
TIK123	Jaringan Komputer	2	5
TIK342	Praktikum Basis Data	1	3

## 8. Seleksi dengan Operator LIKE

Operator LIK memungkinkan pengambilan data yang mirip atau mendekati kata kunci.

1. Dapatkan kode\_mk, nama\_mk, sks, dan semester yang matakuliahnya berawalan “Sistem”.

```
1 SELECT * FROM `matakuliah` WHERE nama_mk like "Sistem%"
2
```

Hasil :

kode_mk	nama_mk	sks	semester
PTI777	Sistem Informatika	2	3
TIK333	Sistem Operasi	3	5

2. Dapatkan kode\_mk, nama\_mk, sks, dan semester yang matakuliahnya berakhiran “Data”.

```
1 SELECT * FROM `matakuliah` WHERE nama_mk LIKE "%Data"  
2
```

Hasil :

kode_mk	nama_mk	sks	semester
PTI447	Praktikum Basis Data	1	3
TIK342	Praktikum Basis Data	1	3

3. Dapatkan kode\_mk, nama\_mk, sks, dan semester yang nama matakuliahnya mengandung kata “Basis”.

```
1 SELECT * FROM `matakuliah` WHERE nama_mk LIKE "%Basis%"  
2
```

Hasil :

kode_mk	nama_mk	sks	semester
PTI333	Basis Data Terdistribusi	3	5
PTI447	Praktikum Basis Data	1	3
TIK342	Praktikum Basis Data	1	3

## 9. Pengurutan Data

Untuk mengurutkan data, kita memanfaatkan klausa ORDER BY, dimana nilai defaultnya adalah ASC (*ascending* atau pengurutan menaik).

1. Urutkan data matakuliah berdasarkan nama matakuliah dengan urutan menaik (*ascending*)

```
1 SELECT * FROM `matakuliah` ORDER BY nama_mk  
2
```

2. Urutkan data matakuliah berdasarkan nama matakuliah dengan urutan menaik (*descending*)

```
1 SELECT * FROM `matakuliah` ORDER BY sks DESC  
2
```

### E. TUGAS PRAKTIKUM

1. Tuliskan pernyataan SQL untuk menciptakan tabel mahasiswa2 dengan struktur seperti tabel mahasiswa di modul pertama. Masukkan 10 data pada tabel mahasiswa2.

No	Nama Field	Tipe Data	Keterangan
1	Nim	Varchar	Primary Key
2	Nama	Varchar	Nama Mahasiswa
3	Jenis_kelamin	Varchar	Jenis Kelamin
4	Alamat	Varchar	Alamat Mahasiswa

2. Tuliskan pernyataan SQL untuk mendapatkan data matakuliah yang sksnya **1 atau 2**.  
Hasil :

Kode_mk	Nama_mk	sks	Semester
PTI447	Praktikum Basis Data	1	3
TIK342	Praktikum Basis Data	1	3
TIK123	Jaringan Komputer	2	5
TIK342	Praktikum Basis Data	1	3

3. Tuliskan Pernyataan SQL untuk mendapatkan data matakuliah yang nama depannya **TIDAK SAMA DENGAN** “praktikum”

Hasil :

Kode_mk	Nama_mk	sks	Semester
PTI333	Basis Data Terdistribusi	3	5
TIK123	Jaringan Komputer	2	5
TIK333	Sistem Operasi	3	5
PTI123	Grafika Multimedia	3	5
PTI777	Sistem Informasi	2	3

### F. TUGAS RUMAH

1. Buat dan tuliskan pernyataan SQL untuk membuat tabel dengan nama **mahasiswa\_nim** (NIM anda masing-masing) dengan field sebagai berikut:

Nama Field	Tipe Data	Keterangan
<b>Nim</b>	Varchar	Primary Key; NIM mahasiswa UM
<b>Nama</b>	Varchar	Nama Mahasiswa UM
<b>Jenis_kelamin</b>	Varchar	Jenis kelamin mahasiswa UM
<b>Tempat_lahir</b>	Varchar	Tempat lahir mahasiswa UM
<b>Tanggal_lahir</b>	Date/time	Tanggal lahir mahasiswa UM
<b>alamat</b>	Varchar	Alamat mahasiswa UM



Note : Tentukan tipe data yang sesuai dengan field yang ada, beserta dengan ukuran fieldnya.



## Praktikum Basis Data 2017 – TE UM

Isikanlah Record pada tabel yang anda buat diatas dengan data sebagai berikut:

nim	nama	jenis_kelamin	tempat_lahir	tanggal_lahir	alamat
140533601613	Cintya	Perempuan	Batam	1998-09-07	Jalan Apel
140533606464	Lugas	Laki-Laki	Batu	1995-04-19	Jalan Mangga
160533608100	Dera	laki-laki	Surabaya	1997-05-17	Jalan Melon
160533608101	Budi	Laki-Laki	Medan	1998-10-29	Jalan Apel
160533608135	Fahmi	Laki-laki	Jombang	1999-06-01	Jalan Sirsat
160533608158	Dona	Perempuan	Bandung	1998-05-16	Jalan Jeruk
160533608189	Erni	Perempuan	Lampung	1997-04-10	Jalan Anggur
160533608203	Dewi	Perempuan	Jogjakarta	1998-04-18	Jalan Markisa

2. Jika diketahui NIM mahasiswa adalah 130533608145, maka:

- 2 digit NIM pertama adalah kode tahun angkatan
- 2 digit NIM berikutnya adalah kode fakultas
- 1 digit NIM berikutnya adalah kode jurusan
- 1 digit NIM berikutnya adalah kode prodi
- 1 digit NIM berikutnya adalah kode jenjang
- 5 digit NIM terakhir adalah nomor urut mahasiswa

Tuliskan pernyataan SQL untuk mendapatkan data mahasiswa dengan kode tahun angkatan = 16, kemudian urutkan secara ascending!

Sehingga akan menampilkan data sebagai berikut:

nim ▲ 1	nama	jenis_kelamin	tempat_lahir	tanggal_lahir	alamat
160533608100	Dera	laki-laki	Surabaya	1997-05-17	Jalan Melon
160533608101	Budi	Laki-Laki	Medan	1998-10-29	Jalan Apel
160533608135	Fahmi	Laki-laki	Jombang	1999-06-01	Jalan Sirsat
160533608158	Dona	Perempuan	Bandung	1998-05-16	Jalan Jeruk
160533608189	Erni	Perempuan	Lampung	1997-04-10	Jalan Anggur
160533608203	Dewi	Perempuan	Jogjakarta	1998-04-18	Jalan Markisa

**Praktikum Basis Data 2017 – TE UM**

3. Tuliskan pernyataan SQL untuk mendapatkan data mahasiswa dengan kode fakultas = 05 **DAN** jenis kelamin = laki-laki. Urutan berdasarkan nim secara descending

nim	nama	jenis_kelamin	tempat_lahir	tanggal_lahir	alamat
140533606464	Lugas	Laki-Laki	Batu	1995-04-19	Jalan Mangga
160533608100	Dera	laki-laki	Surabaya	1997-05-17	Jalan Melon
160533608101	Budi	Laki-Laki	Medan	1998-10-29	Jalan Apel
160533608135	Fahmi	Laki-laki	Jombang	1999-06-01	Jalan Sirsat

4. Tuliskan pernyataan SQL untuk mendapatkan data nim, nama, jenis kelamin, dan alamat pada tabel mahasiswa yang memiliki nim dengan kode tahun angkatan = 14 dan alamat <> “jalan mangga”.

nim	nama	jenis_kelamin	alamat
140533601613	Cintya	Perempuan	Jalan Apel

5. Tuliskan pernyataan SQL untuk mendapatkan data nim, nama, tanggal lahir, dan umur pada tabel mahasiswa yang mana umur mahasiswa di dapatkan dari tanggal lahir yang terdapat dalam table mahasiswa seperti pada contoh di bawah ini :

nim	nama	tanggal_lahir	umur
140533606464	lugas	1995-04-19	21
160533608100	Budi	1998-10-29	18

6. Analisislah query yang terdapat pada soal nomor 2,3, dan 4, 5!