

## BAB 3. DATA DEFINITION LANGUAGE (DDL)

### 3.1 TUJUAN PRAKTIKUM

#### Tujuan Umum

Mahasiswa mampu memahami dan menerapkan Kueri SQL - Data Definition Language (DDL)

#### Tujuan Khusus

Mahasiswa mampu:

- Menerapkan Kueri SQL - DDL untuk Membuat Tabel
- Menerapkan Kueri SQL - DDL untuk Mengubah Struktur Tabel
- Menerapkan Kueri SQL - DDL untuk Menghapus Tabel

### 3.2 TEORI SINGKAT

#### a. Data Definition Language (DDL)

merupakan kumpulan perintah SQL yang digunakan untuk membuat (create), mengubah (alter dan rename) dan menghapus (drop) struktur dari objek-objek database seperti table, view, procedure, function dan trigger.

#### b. Tipe Data

Sebelum mempelajari perintah-perintah tersebut akan diperkenalkan beberapa tipe data yang umum dipakai dalam SQL.

Tabel 3.1 Tipe Data Umum pada SQL

Tipe Data	Keterangan
TINYINT	digunakan untuk menyimpan data bilangan bulat positif dan negatif. Jangkauan : -128 s/d 127 Ukuran : 1 byte (8 bit).
SMALLINT	digunakan untuk menyimpan data bilangan bulat positif dan negatif..

	Jangkauan : -8.388.608 s/d 8.388.607. Ukuran : 3 byte (24 bit).
INT	digunakan untuk menyimpan data bilangan bulat positif dan negatif. Jangkauan : -2.147.483.648 s/d 2.147.483.647 Ukuran : 4 byte (32 bit).
BIGINT	digunakan untuk menyimpan data bilangan bulat positif dan negatif. Jangkauan : $\pm 9,22 \times 10^{18}$ Ukuran : 8 byte (64 bit).
FLOAT	digunakan untuk menyimpan data bilangan pecahan positif dan negatif presisi tunggal. Jangkauan : -3.402823466E+38 s/d -1.175494351E 38, 0, dan 1.175494351E-38 s/d 3.402823466E+38. Ukuran : 4 byte (32 bit).
DOUBLE	digunakan untuk menyimpan data bilangan pecahan positif dan negatif presisi ganda. Jangkauan : -1.79...E+308 s/d -2.22...E-308, 0, dan 2.22...E-308 s/d 1.79...E+308. Ukuran : 8 byte (64 bit).

Tabel 3.2 Tipe Data Waktu pada SQL

Tipe Data	Keterangan
DATE	digunakan untuk menyimpan data tanggal. Jangkauan : 1000-01-01 s/d 9999-12-31 (YYYY-MMDD) Ukuran : 3 byte.
TIME	digunakan untuk menyimpan data waktu. Jangkauan : -838:59:59 s/d +838:59:59 (HH:MM:SS)

	Ukuran : 3 byte.
DATETIME	digunakan untuk menyimpan data tanggal dan waktu. Jangkauan : '1000-01-01 00:00:00' s/d '9999-12-31 23:59:59' Ukuran : 8 byte.
TIMESTAMP	Kombinasi tanggal dan jam saat tabel/data diakses dengan jangkauan '1970-01-01 00:00:00' s.d. '2037 '
YEAR	digunakan untuk menyimpan data tahun dari tanggal. Jangkauan : 1900 s/d 2155 Ukuran : 1 byte.

Tabel 3.3 Tipe Data Lainnya pada SQL

Tipe Data	Keterangan
CHAR	digunakan untuk menyimpan data string ukuran tetap. Jangkauan : 0 s/d 255 karakter
VARCHAR	digunakan untuk menyimpan data string ukuran dinamis. Jangkauan : 0 s/d 255 karakter (versi 4.1), 0 s/d 65.535 (versi 5.0.3)
TINYBLOB, TINYTEXT	L+1 byte, dengan L<28. Tipe data TEXT atau BLOB dengan panjang maksimum 255 karakter.
MEDIUMBLOB, MEDIUMTEXT	L+2 byte, dengan L<224. Tipe TEXT atau BLOB dengan panjang maksimum 1677215 karakter

<p> LONGBLOB,  LONGTEXT </p>	<p> L+2 byte, dengan L&lt;32. Tipe TEXT atau BLOB dengan panjang maksimum 4294967295 karakter. </p>
<p> ENUM </p>	<p> Digunakan untuk enumerasi (kumpulan data).  Jangkauan : sampai dengan 65535 string. </p>
<p> SET </p>	<p> Digunakan untuk combination (himpunan data).  Jangkauan : sampai dengan 255 string anggota </p>

c. Aturan dan Saran Penamaan Objek Database

Dalam pembuatan database serta pembuatan objek dan struktur didalamnya terdapat beberapa aturan, secara umum bisa dikatakan sama dengan aturan penulisan variabel pada bahasa pemrograman. Agar penggunaan objek database dapat digunakan dengan baik, sebaiknya ikuti aturan dan saran berikut ini:

- Terdiri dari huruf, angka dan simbol tertentu ( \_ )
- Diawali dengan huruf
- Gunakan huruf kecil.
- Jika terdiri dari dua kata atau lebih gunakan garis bawah, contoh transaksi\_detail.
- Gunakan kata benda untuk tabel, jangan pernah menggunakan kata kerja untuk tabel.
- Beri nama tabel sesuai dengan data yang disimpan dalam tabel tersebut, jangan menggunakan kata umum seperti tabel1 , datatbl dan sejenisnya.
- Jangan menyingkat nama table maupun kolom, contoh tabel mhs atau tabel trans, padahal yang anda maksud adalah mahasiswa dan transaksi. Gunakan kata utuh dan jangan mempersingkat nama tabel yang tidak perlu.
- Untuk primary key auto increment gunakan pola idnamatable, contohnya tabel siswa nama kolom primary key nya dapat idkampus, id\_kampus atau kampusID.
- Pola penamaan tabel dan kolom harus konsisten
- Gunakan nama kolom yang sama untuk kolom relasi, yaitu kolom primary key dengan foreign key nya.

d. Membuat Tabel (Create)

**Create** digunakan untuk membuat basis data maupun objek-objek basis data.

Format SQL untuk membuat tabel yang umum digunakan, sebagai berikut:

```
CREATE TABLE [nama_tabel] (  
    nama_kolom1 tipe_data(panjang_data) constraint,  
    ..... );
```

Untuk melihat table yang terdapat dalam database dapat menggunakan kueri SQL berikut:

```
SHOW TABLES;
```

Untuk melihat struktur table dapat menggunakan salah satu kueri SQL berikut:

```
DESCRIBE [nama_tabel];  
DESC [nama_tabel];  
SHOW COLUMNS IN [nama_tabel];
```

e. Mengubah Struktur Tabel (Alter)

**Alter** digunakan untuk mengubah struktur pada tabel, SQL yang umum digunakan adalah:

```
ALTER TABLE [nama_tabel]  
    RENAME TO/ADD COLUMN/CHANGE COLUMN/DROP  
    COLUMN/MODIFY COLUMN/ADD CONSTRAINT [kondisi]
```

f. Menghapus Tabel (Drop)

**Drop** digunakan untuk menghapus sebuah tabel, SQL yang umum digunakan :

```
DROP TABLE [nama_table];
```

### 3.3 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 membuat tabel mahasiswa:

```
CREATE TABLE mahasiswa(  
    nim varchar(7) PRIMARY KEY,  
    nama varchar(25) NOT NULL,  
    angkatan year(4) NOT NULL,  
    mahasiswa_umur int(2),  
    jurusan varchar(25) NOT NULL,  
    alamat varchar(50) NOT NULL,  
    email varchar(30) NOT NULL,  
    CONSTRAINT email_unik UNIQUE KEY(email)  
);
```

6. Ketikkan perintah `show tables` untuk melihat tabel yang sudah anda buat seperti terlihat pada Gambar 3.1.

7. Ketikkan perintah `desc mahasiswa` untuk melihat struk tabel mahasiswa yang sudah anda buat seperti terlihat pada Gambar 3.2.

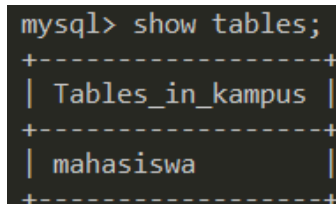
8. Ketikkan perintah seperti dibawah ini untuk mengubah struktur tabel mahasiswa.

```
alter table tabel_mahasiswa rename to mahasiswa;  
alter table mahasiswa add column telepon varchar(15);  
alter table mahasiswa change column umur usia int(2);  
alter table mahasiswa change column usia umur int(2) not null;  
alter table mahasiswa drop column telepon;  
alter table mahasiswa modify column alamat text;  
alter table mahasiswa add constraint alamat_unik UNIQUE KEY(email);  
alter table mahasiswa drop KEY alamat_unik;
```

9. Ketikkan perintah `desc mahasiswa` untuk melihat struktur tabel mahasiswa yang sudah anda ubah pada langkah sebelumnya.

10. Silahkan buka menu log kueri dan pastikan perintah SQL yang sudah diketik di MySQL Terminal sudah ada.

11. Selesai.



```
mysql> show tables;  
+-----+  
| Tables_in_kampus |  
+-----+  
| mahasiswa        |  
+-----+
```

Gambar 3.1 Hasil tampilan isi tabel

```
mysql> desc mahasiswa;
```

Field	Type	Null	Key	Default	Extra
nim	varchar(7)	NO	PRI	NULL	
nama	varchar(25)	NO		NULL	
angkatan	year(4)	NO		NULL	
mahasiswa_umur	int(2)	YES		NULL	
jurusan	varchar(25)	NO		NULL	
alamat	varchar(50)	NO		NULL	
email	varchar(30)	NO	UNI	NULL	

Gambar 3.2 Hasil tampilan struktur tabel

```
mysql> desc mahasiswa;
```

Field	Type	Null	Key	Default	Extra
nim	varchar(7)	NO	PRI	NULL	
nama	varchar(25)	NO		NULL	
angkatan	year(4)	NO		NULL	
mahasiswa_umur	int(2)	YES		NULL	
jurusan	varchar(25)	NO		NULL	
alamat	text	YES		NULL	
email	varchar(30)	NO	UNI	NULL	

Gambar 3.3 Hasil tampilan struktur tabel setelah perubahan

### 3.4 LATIHAN

1. Ketikkan perintah kueri SQL berikut pada MySQL Terminal:

```
create table barang (  
  barang_id int(5) primary key auto_increment,  
  barang_nama varchar(25) null,  
  barang_harga int(11) null  
);  
  
desc barang;
```

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

2. Ketikkan perintah kueri SQL berikut pada MySQL Terminal:

```
alter barang add column barang_kode varchar(10);  
alter barang modify column barang_nama varchar(50) not null;  
  
desc barang;
```

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

### 3.5 TUGAS

1. Buatlah kueri SQL untuk membuat tabel karyawan yang terdiri dari 5 kolom dengan minimal mengimplementasikan penggunaan 3 tipe data yang berbeda, contoh: kolom pertama menggunakan tipe data int, kolom kedua menggunakan varchar dan kolom ketiga menggunakan date, kolom sisanya bebas menggunakan tipe data apa saja sesuai kebutuhan!
2. Berdasarkan soal tugas no.1 buatlah kueri SQL yang dapat mengubah salah satu nama kolom dan tambahkan 2 kolom baru serta tambahkan juga constraint primary key dan atau unique key ke dalam struktur tabel tersebut!
3. Buatlah kueri SQL untuk menghapus tabel karyawan yang telah dibuat!