

Modul 3

DataBase Dan Tabel

Tujuan :

1. Membuat database dan tabel dengan data definition language
2. Mampu memodifikasi tabel

Dasar teori

DDL (Data Definition Language) , DDL merupakan kelompok perintah yang berfungsi untuk mendefinisikan atribut-atribut basis data, tabel, atribut(kolom), batasan-batasan terhadap suatu atribut, serta hubungan antar tabel. Yang termasuk dalam kelompok DDL ini adalah CREATE, ALTER, dan DROP.

a. Syntax Membuat Database

```
CREATE DATABASE namadatabase;
```

Namadatabase tidak boleh mengandung spasi dan tidak boleh memiliki nama yang sama antar database.

Syntax tambahan untuk menampilkan daftar nama database yang ada pada mysql menggunakan perintah

```
SHOW DATABASES;
```

b. Syntax Memilih Database

```
USE namadatabase;
```

Sebelum membuat suatu tabel, terlebih dahulu harus memilih salah satu database sebagai database aktif yang akan digunakan untuk menyimpan tabel-tabel.

c. Syntax Menghapus Database

```
DROP DATABASE namadatabase;
```

Database yang akan dihapus sesuai dengan namadatabase.

d. Syntax Membuat Tabel

```
CREATE TABLE namatabel2 ( Field1 TipeData1,Field2 TipeData2);
```

Namatabel tidak boleh mengandung spasi (space). Field1 dan TipeData1 merupakan nama kolom pertama dan tipe data untuk kolom pertama. Jika ingin membuat tabel dengan kolom lebih dari satu, maka setelah pendefinisian tipe data sebelumnya diberikan tanda koma (,).

e. Syntax Menampilkan Tabel

```
SHOW TABLES;
```

Untuk menampilkan daftar nama tabel yang ada pada database yang sedang aktif/digunakan

f. Syntax Menampilkan Atribut Tabel

```
DESC namatabel;
```

Untuk menampilkan deskripsi tabel

g. Syntax Menghapus Tabel

```
DROP TABLE namatabel;
```

Tabel yang akan dihapus sesuai dengan namatabel,

h. Syntax Mendefinisikan Null/Not Null

```
CREATE TABLE namatabel ( Field1 TipeData1 NOT NULL,Field2 TipeData2);
```

i. Syntax Mendefinisikan Nilai Default

```
CREATE TABLE namatabel (Field1 TipeData1 DEFAULT nilaidefault,Field2  
TipeData2 DEFAULT nilaidefault);
```

Nilai default adalah nilai yang otomatis diberikan oleh sistem untuk suatu kolom ketika ada penambahan baris baru, sementara nilai pada kolom tersebut tidak diisi oleh pengguna.

j. Syntax Mendefinisikan Primary Key Pada Tabel

Terdapat tiga cara untuk mendefinisikan primary key.

Berikut ini adalah Syntax mendefinisikan primary key untuk Field1

```
CREATE TABLE namatabel(Field1 TipeData1 NOT NULL PRIMARY KEY, Field2  
TipeData2);
```

Atau

```
CREATE TABLE namatabel ( Field1 TipeData1, Field2 TipeData2,  
PRIMARY KEY(Field1));
```

Atau

```
ALTER TABLE namatabel ADD CONSTRAINT namaconstraint PRIMARY KEY  
(namakolom);
```

k. Syntax Menghapus Primary Key Pada Tabel

Cara 1 : Jika primary key dibuat dengan menggunakan alter table :

```
ALTER TABLE namatabel DROP CONSTRAINT namaconstraint;
```

Cara 2 : Jika primary key dibuat melalui create table :

```
ALTER TABLE namatabel DROP PRIMARY KEY;
```

l. Syntax Menambah Kolom Baru Pada Tabel

Namatabel adalah nama tabel yang akan ditambah fieldnya. Fieldbaru adalah nama kolom yang akan ditambahkan, tipe adalah tipe data dari kolom yang akan ditambahkan. Berikut ini perintah untuk menambah kolom keterangan dengan tipe data varchar(25) :

```
ALTER TABLE JENISFILM ADD KETERANGAN VARCHAR(25);
```

Untuk meletakkan field diawal, tambahkan sintaks first :

```
ALTER TABLE PELANGAN ADD COLUMN KODE CHAR(5) FIRST;
```

Untuk menyisipkan field setelah field tertentu, tambahkan sintaks after :

```
ALTER TABLE PELANGAN ADD COLUMN PHONE CHAR(5) AFTER ALAMAT;
```

m. Syntax Mengubah Tipe Data atau Lebar Kolom Pada Tabel

```
ALTER TABLE NAMATABEL MODIFY COLUMN FIELD TIPE
```

Namatabel adalah nama tabel yang akan diubah tipe data atau lebar kolomnya. Field adalah kolom yang akan diubah tipe data atau lebarnya. Tipe adalah tipe data baru atau tipe data lama dengan lebar kolom yang berbeda.

n. Syntax Mengubah Nama Kolom

```
ALTER TABLE namatabel CHANGE COLUMN namalamakolom namabarukolom  
tipedatabaru;
```

Namatabel adalah nama tabel yang akan diubah nama kolomnya, namalamakolom adalah kolom yang akan diganti namanya, namabarukolom adalah nama baru kolom, typedatanya adalah tipe data dari kolom tersebut.

o. Syntax Menghapus Kolom Pada Tabel

```
ALTER TABLE namatabel DROP COLUMN namakolom;
```

p. Syntax Membuat dan Menghapus Index

Index berfungsi mempercepat proses pencarian data dalam suatu tabel. Adanya index pada suatu field tabel menyebabkan proses pencarian otomatis akan dilakukan terlebih dahulu ke dalam index, apabila ditemukan baru akan diambilkan data sesungguhnya dari tabel, apabila tidak ditemukan dalam index, sudah dipastikan bahwa data tersebut memang tidak ada dalam tabel. Index juga dapat dibuat untuk setiap kolom yang akan dijadikan kriteria tertentu untuk pencarian data sehingga proses pencariannya akan lebih cepat. Ada perintah untuk membuat dan menghapus index, tapi tidak ada perintah untuk merubah index.

Perhatikan contoh berikut :

```
CREATE INDEX IDXNOMOR ON MAHASISWA (NIM) ;
```

Atau

```
ALTER TABLE MAHASISWA ADD INDEX IDXNIM (NIM) ;
```

Untuk menghapus index :

```
DROP INDEX IDXNIM ON MAHASISWA;
```

Atau

```
ALTER TABLE MAHASISWA DROP INDEX IDXNIM;
```

Praktik

1. Buat Database dengan nama tes
2. Buat sebuah tabel dengan nama mahasiswa di dalam database tes !

Field	Type	Null	Key	Default	Extra
Nama	varchar(15)	NO			
Nim	int(10)	YES		NULL	
Alamat	varchar(30)	YES		NULL	
Email	varchar(30)	YES		NULL	

4 rows in set (0.20 sec)

3. Tambahkan sebuah kolom : Keterangan (varchar 15), sebagai kolom terakhir !
4. Tambahkan kolom Kelas (varchar 10) di awal (sebagai kolom pertama) !
5. Sisipkan sebuah kolom dengan nama Telepon (varchar 15) setelah kolom alamat !
6. Ubah kolom nim menjadi char(15) !
7. Ubah nama kolom Telepon menjadi Phone (varchar 20) !
8. Hapus kolom Kelas dari tabel !
9. Ganti nama tabel menjadi student!
10. Jadikan index untuk NIM !
11. Jadikan nim sebagai primary key !