

TUGAS TAHAP I
MATA KULIAH PRAKTIKUM SISTEM BASIS DATA
PRODI INFORMATIKA

Nama : Ralfi Firmansyah

Npm : 201843500102

Kelas : S4C

1. Buatlah Database Tugas1

Syntax dibawah ini untuk membuat database dengan nama database tugas1 (atau sesuai dengan perintah)

- MULAI
- Fungsi untuk membuat database
- Telah selesai membuat sebuah database dengan nama Tugas1
- SELESAI

```
C:\Users\Ralfi_firmansyah>CD/
C:\>cd xampp
C:\xampp>cd mysql
C:\xampp\mysql>cd bin
C:\xampp\mysql\bin>mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.41 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database tugas1;
Query OK, 1 row affected (0.08 sec)
```

2. Aktifkan database

Syntax dibawah ini digunakan untuk mengaktifkan database tersebut, jika tidak menggunakan USE nama_database; maka akan muncul error “No database selected.

- MULAI
- Fungsi untuk mengaktifkan database
- Mengaktifkan database Tugas1
- SELESAI

```
mysql> use tugas1;  
Database changed
```

3. Buatlah 3 buah tabel diatas

Buatlah tabel :

- Mahasiswa (Npm, Nama, Alamat)!
- Matakuliah (KD_MK, Nama_MK, SKS,)!
- Nilai(NPM, KD_MK, UTS , UAS)!

Syntax dibawah ini digunakan untuk membuat sebuah table dengan 3 table berserta fieldnya.

- MULAI
- Fungsi menambah table
- Memasukan table (field1, field2, field3)
- Sistem memeriksa data table yang dimasukan ke dalam sistem
- Bila data benar, sistem akan menyimpan data table ke dalam database
- Bila data tidak benar, sistem akan menampilkan informasi kesalahan data
- SELESAI

```
mysql> CREATE TABLE mahasiswa(  
-> NPM varchar(30),  
-> NAMA varchar(40),  
-> ALAMAT varchar(200));  
Query OK, 0 rows affected (0.13 sec)  
  
mysql> CREATE TABLE Matakuliah(  
-> KD_MK varchar(30),  
-> NAMA_MK varchar(50),  
-> SKS integer(10));  
Query OK, 0 rows affected (0.38 sec)  
  
mysql> CREATE TABLE nilai(  
-> NPM varchar(20),  
-> KD_MK varchar(30),  
-> UTS integer(10),  
-> UAS integer(10));  
Query OK, 0 rows affected (0.42 sec)
```

4. Tampilkan seluruh database

Syntax dibawah ini digunakan untuk menampilkan seluruh database yaitu dengan cara SHOW DATABASES;

- MULAI
- Fungsi menampilkan Database
- Menampilkan Database
- SELESAI

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| cdcol      |
| mysql      |
| phpmyadmin |
| test       |
| tugas1     |
+-----+
6 rows in set (0.11 sec)
```

5. Tampilkan seluruh tabel

syntax dibawah ini digunakan untuk menampilkan seluruh table yang telah dibuat. Seperti ketika kita membuat banyak table kemudian kita ingin mengecek apakah semua table yang telah kita buat sudah selesai atau justru belum dan masih ada yang belum dibuat maka kita tinggal mengeceknya dengan cara **show tables;**

- MULAI
- Fungsi menampilkan semua table yang telah dibuat
- Menampilkan seluruh table
- SELESAI

```
mysql> SHOW TABLES;
+-----+
| Tables_in_tugas1 |
+-----+
| mahasiswa        |
| matakuliah        |
| nilai            |
+-----+
3 rows in set (0.05 sec)
```

6. Tampilkan struktur tabel Matakuliah

Syntax dibawah ini digunakan untuk menampilkan table yang ingin ditampilkan yaitu dengan cara **desc nama_table;**

- MULAI
- Fungsi menampilkan table sesuai yang diinginkan

- Menampilkan table tersebut
- SELESAI

```
mysql> DESC Matakuliah;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| KD_MK | varchar(30) | YES  |     | NULL    |       |
| NAMA_MK | varchar(50) | YES  |     | NULL    |       |
| SKS    | int(10)     | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.08 sec)
```

7. Hapus tabel Matakuliah

Syntax dibawah ini digunakan untuk menghapus table, yaitu dengan cara **drop table nama_tabel;** yang akan di hapus. Maka table tersebut akan otomatis terhapus

Setelah kita enter kemudian muncul query ok, maka kita telah berhasil menghapusnya. Tetapi, kita juga harus ingat nama table tersebut juga harus sesuai dengan besar kecilnya dari huruf nama table tersebut. Karena jika salah satu huruf saja maka itu tidak akan berhasil

- MULAI
- Fungsi menghapus table
- Table telah dihapus
- SELESAI

```
mysql> DROP TABLE Matakuliah;
Query OK, 0 rows affected (0.00 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_tugas1 |
+-----+
| mahasiswa        |
| nilai            |
+-----+
2 rows in set (0.00 sec)
```

8. Buat tabel Matakuliah

Syntax dibawah ini untuk menunjukan cara membuat table , karena sebelumnya table matakuliah telah kita hapus maka untuk memunculkan lagi table matakuliah kita harus membuatnya seperti semula, yaitu

- MULAI
- Fungsi membuat table matakuliah
- Membuat field beserta dengan tipe data untuk table matakuliah sesuai dengan kebutuhan

- Bila data benar, sistem akan menyimpan Field ke dalam table matakuliah
- Bila data ada kesalahan, sistem akan menampilkan bacaan error
- SELESAI

```
mysql> CREATE TABLE Matakuliah(
-> KD_MK varchar(30),
-> NAMA_MK varchar(50),
-> SKS integer(10));
Query OK, 0 rows affected (0.09 sec)
```

9. Ganti tabel Matakuliah dengan MT_Kuliah

Syntax dibawah ini untuk menunjukan cara mengganti nama table. Yaitu dengan cara **Alter table nama_table_ sebelumnya rename nama_table_yang_Baru;**

Setelah berhasil kita dapat cek dengan cara show table; jika nama table berubah sesuai dengan keinginan itu berarti kita telah berhasil.

- MULAI
- Fungsi mengganti nama table
- Nama table telah diganti
- Bila data benar, sistem akan mengganti nama table dengan nama yang baru
- Bila data ada kesalahan, sistem akan menampilkan bacaan error
- SELESAI

```
mysql> ALTER TABLE Matakuliah rename MT_kuliah;
Query OK, 0 rows affected (0.08 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_tugas1 |
+-----+
| mahasiswa        |
| mt_kuliah         |
| nilai             |
+-----+
3 rows in set (0.00 sec)
```

10. Ganti nama field Nama_MK dengan NM_MK dan panjang field 30

Syntax dibawah ini digunakan untuk mengganti nama field dan panjang dari field tersebut. Jika tadi kita hanya merubah nama table sekarang kita justru harus merubah nama field yang ada pada table tersebut. tentu caranya pun beda dengan cara merubah nama table yaitu dengan cara: **alter table nama_tabel change nama_field_yang_akan_diubah enter Nama_field_baru(besertakan tipe data dan size nya);** seperti syntax dibawah ini

- MULAI

- Fungsi mengganti nama field berikut tipe data dan sizenya
- Nama field telah berubah
- SELESAI

```
mysql> ALTER TABLE mt_kuliah change NAMA_MK
-> NM_MK varchar(30);
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc mt_kuliah;
```

Field	Type	Null	Key	Default	Extra
KD_MK	varchar(30)	YES		NULL	
NM_MK	varchar(30)	YES		NULL	
SKS	int(10)	YES		NULL	

```
3 rows in set (0.02 sec)
```

11. Tambahkan kolom Tgl_lahir pada tabel Mahasiswa

Syntax dibawah ini ditunjukan untuk menambah field baru pada table mahasiswa. Jika kita sedang menjalankan sebuah database kemudian kita lupa untuk memasukan salah satu field ke dalam table kita tidak perlu khawatir. Karena ada cara menambahkan field baru ke dalam table yg diinginkan. Yaitu dengan cara **Alter Table Mahasiswa ADD column nama_field(beserta tipe data dan sizenya)**; jika berhasil maka akan muncul Query ok,

- Mulai
- Fungsi membuat(menambahkan) field baru
- Bila data benar, sistem akan menambahkan field baru ke dalam table mahasiswa
- Bila data ada kesalahan, sistem akan menampilkan bacaan error
- Field telah berhasil dimasukan ke dalam table
- SELESAI

```
mysql> ALTER TABLE Mahasiswa ADD column Tgl_lahir date;
Query OK, 0 rows affected (0.27 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESC Mahasiswa;
```

Field	Type	Null	Key	Default	Extra
NPM	varchar(30)	YES		NULL	
NAMA	varchar(40)	YES		NULL	
ALAMAT	varchar(200)	YES		NULL	
Tgl_lahir	date	YES		NULL	

```
4 rows in set (0.01 sec)
```

12. Hapus kolom Tgl_lahir

Syntax dibawah ini digunakan untuk menghapus field, yaitu dengan cara **Alter table nama_tabel drop nama_field;** yang akan di hapus. Maka field tersebut akan otomatis terhapus

Setelah kita enter kemudian muncul query ok, maka kita telah berhasil menghapusnya. Tetapi, kita juga harus ingat nama field tersebut juga harus sesuai dengan besar kecilnya dari huruf nama fieldn tersebut. Karena jika salah satu huruf saja beda maka itu tidak akan berhasil

- MULAI
- Fungsi menghapus field
- Field telah dihapus
- SELESAI

```
mysql> ALTER TABLE Mahasiswa DROP Tgl_lahir;
Query OK, 0 rows affected (0.45 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESC Mahasiswa;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| NPM   | varchar(30)   | YES  |     | NULL    |       |
| NAMA  | varchar(40)   | YES  |     | NULL    |       |
| ALAMAT | varchar(200)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

13. Tambahkan kolom Tgl_Lahir pada tabel Mahasiswa pada awal field

Syntax dibawah ini ditunjukan untuk menambah field baru pada table mahasiswa seperti sebelumnya. Namun bedanya sekarang ditentukan letak field baru tersebut . Yaitu dengan cara **Alter Table Mahasiswa ADD column nama_field(beserta tipe data dan sizenya) first;** jika berhasil maka akan muncul Query ok,

- Mulai
- Fungsi membuat(menambahkan) field baru dengan posisi yang diinginkan
- Bila data benar, sistem akan menambahkan field baru ke dalam table mahasiswa
- Bila data ada kesalahan, sistem akan menampilkan bacaan error
- Field telah berhasil dimasukan ke dalam table mahasiswa
- SELESAI

```
mysql> ALTER TABLE Mahasiswa ADD column Tgl_lahir date first;
Query OK, 0 rows affected (0.16 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESC Mahasiswa;
```

Field	Type	Null	Key	Default	Extra
Tgl_lahir	date	YES		NULL	
NPM	varchar(30)	YES		NULL	
NAMA	varchar(40)	YES		NULL	
ALAMAT	varchar(200)	YES		NULL	

```
4 rows in set (0.02 sec)
```

14. Hapus kolom Tgl_Lahir

Syntax dibawah ini digunakan untuk menghapus field, yaitu dengan cara **Alter table nama_tabel drop nama_field;** yang akan di hapus. Maka field tersebut akan otomatis terhapus

Setelah kita enter kemudian muncul query ok, maka kita telah berhasil menghapusnya. Tetapi, kita juga harus ingat nama field tersebut juga harus sesuai dengan besar kecilnya dari huruf nama field tersebut. Karena jika salah satu huruf saja beda maka itu tidak akan berhasil

- MULAI
- Fungsi menghapus field
- Field telah dihapus
- SELESAI

```
mysql> ALTER TABLE Mahasiswa DROP Tgl_lahir;
Query OK, 0 rows affected (0.17 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESC Mahasiswa;
```

Field	Type	Null	Key	Default	Extra
NPM	varchar(30)	YES		NULL	
NAMA	varchar(40)	YES		NULL	
ALAMAT	varchar(200)	YES		NULL	

```
3 rows in set (0.02 sec)
```

15. Tambahkan kolom Tgl_Lahir pada tabel Mahasiswa setelah kolom nama

Syntax dibawah ini ditunjukkan untuk menambah field baru pada table mahasiswa seperti sebelumnya. Namun bedanya sekarang ditentukan letak field baru tersebut . Yaitu dengan cara **Alter Table Mahasiswa ADD column nama_field(beserta tipe data dan sizenya) after nama_field_sebelumnya;** jika berhasil maka akan muncul Query ok. Setelah itu kita dapat mengecek dengan cara **Desc Mahasiswa;** jika telah berubah maka bisa dikatakan kita telah berhasil.

- Mulai
- Fungsi membuat(menambahkan) field baru dengan posisi yang diinginkan
- Bila data benar, sistem akan menambahkan field baru ke dalam table mahasiswa
- Bila data ada kesalahan, sistem akan menampilkan bacaan error
- Field telah berhasil dimasukan ke dalam table mahasiswa
- SELESAI

```
mysql> ALTER TABLE Mahasiswa ADD column Tgl_lahir date after Nama;
Query OK, 0 rows affected (0.53 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESC Mahasiswa;
```

Field	Type	Null	Key	Default	Extra
NPM	varchar(30)	YES		NULL	
NAMA	varchar(40)	YES		NULL	
Tgl_lahir	date	YES		NULL	
ALAMAT	varchar(200)	YES		NULL	

```
4 rows in set (0.02 sec)
```