

TEKNOLOGI BASIS DATA

Praktikum MySQL pada CMD

Tugas Praktikum 1 Basis Data



Oleh:

Nova Heriyani

311710537

TI.17.D2

Program Studi Teknik Informatika

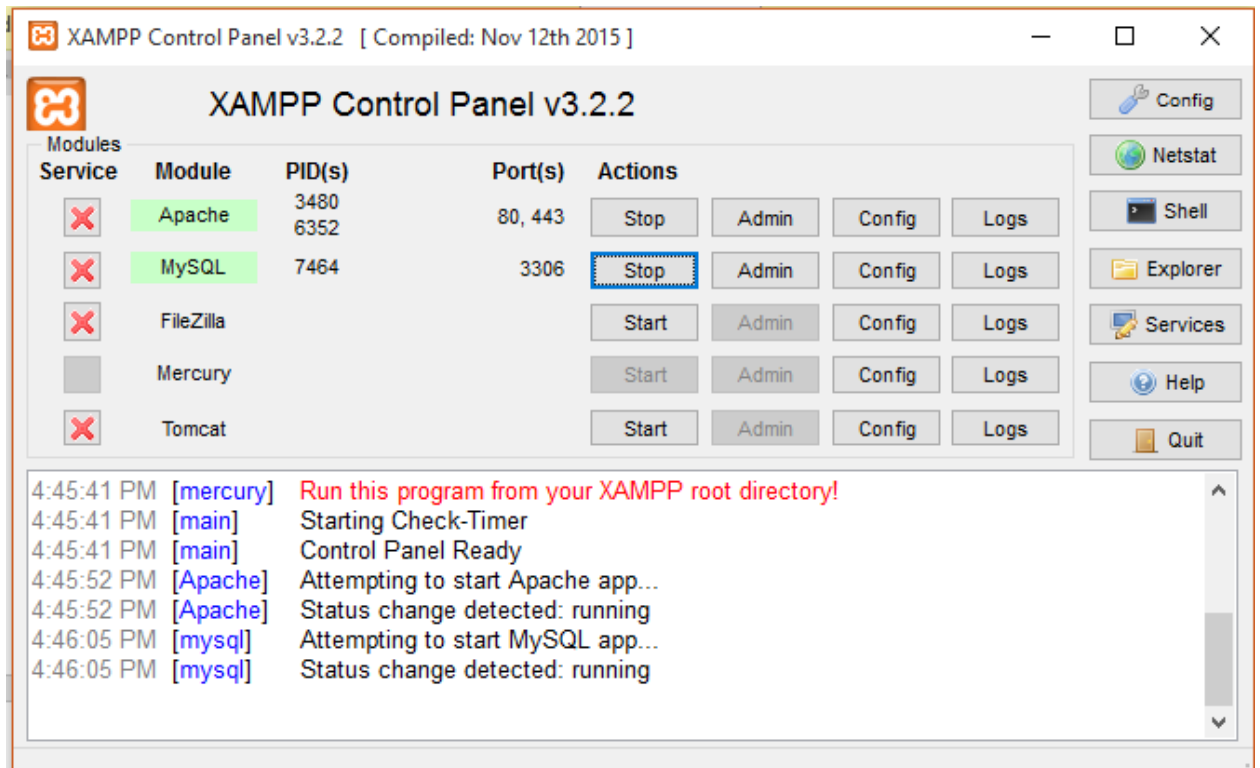
Sekolah Tinggi Teknik Pelita Bangsa

2019/2020

TUGAS PRAKTIKUM

1. Buatlah sebuah database dengan nama “latihan1”

Sebelum membuat database kita harus mengaktifkan fitur MySQL disini saya menggunakan aplikasi XAMPP. Aktifkan atau klik “Start” pada button APACHE dan MySQL seperti pada gambar di bawah ini.



Jika sudah dijalankan seperti diatas, langkah selanjutnya kita login ke system MySQL untuk membuat database caranya seperti pada gambar di bawah ini:

```
Administrator: Command Prompt - mysql -u root
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd c:\xampp\mysql\bin

c:\xampp\mysql\bin>mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.1.10-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

Pada perintah **mysql -u root** berfungsi untuk login ke dalam system MySQL, maksud -u adalah username dengan root sebagai usernya . karena secara default MySQL pada XAMPP tidak memiliki password jadi kita tidak perlu mengisi passwordnya.

Setelah kita berhasil login kita dapat membuat Database latihan1 dengan perintah :

- CREATE DATABASE [nama_database]

```
MariaDB [(none)]> create database latihan1;  
Query OK, 1 row affected (0.03 sec)
```

2. Buat sebuah tabel dengan nama Biodata dengan field nama dan alamat pada database latihan1
Untuk membuat table pada database yang kita inginkan, kita terlebih dahulu harus menggunakan database tersebut dengan perintah:

- USE [nama_database]

```
MariaDB [(none)]> use latihan1;  
Database changed  
MariaDB [latihan1]>
```

Setelah kita menggunakan Database yang kita inginkan kita dapat membuat table dengan perintah seperti:

- CREATE TABLE nama_tabel (nama_field1 tipe_data(ukuran), nama_field2 tipe_data(ukuran);

```
Database changed  
MariaDB [latihan1]> create table biodata(nama varchar(50),alamat text);  
Query OK, 0 rows affected (0.42 sec)
```

```
MariaDB [latihan1]> desc biodata;
```

| Field | Type | Null | Key | Default | Extra |
|--------|-------------|------|-----|---------|-------|
| nama | varchar(50) | YES | | NULL | |
| alamat | text | YES | | NULL | |

```
2 rows in set (0.04 sec)
```

Saya menggunakan table dengan nama biodata dan jika ingin melihat hasilnya gunakan perintah “DESC nama_table ;”

3. Tambahkan sebuah Kolom **keterangan varchar(15)**, sebagai kolom terakhir.
- Untuk menambahkan kolom/field baru pada kolom terakhir kita gunakan perintah :
- **ALTER TABLE nama_table ADD COLUMN nama_field [type_data] (ukuran);**

```
MariaDB [latihan1]> alter table biodata add column keterangan varchar(15);
Query OK, 0 rows affected (0.75 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc biodata;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nama       | varchar(50)   | YES  |     | NULL    |       |
| alamat     | text          | YES  |     | NULL    |       |
| keterangan | varchar(15)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.03 sec)
```

Secara otomatis field tersebut akan menjadi kolom terakhir.

4. Tambahkan kolom **id INT(11)** di awal /sebagai kolom pertama.
- Sama seperti diatas, untuk menambahkan field baru pada urutan pertama, kita hanya perlu menambahkan kata **FIRST** pada kalimat terakhir diatas.
- **ALTER TABLE nama_table ADD COLUMN nama_field [type_data](ukuran);**

```
MariaDB [latihan1]> alter table biodata add column id int(11) first;
Query OK, 0 rows affected (0.71 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc biodata;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)       | YES  |     | NULL    |       |
| nama       | varchar(50)   | YES  |     | NULL    |       |
| alamat     | text          | YES  |     | NULL    |       |
| keterangan | varchar(15)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.04 sec)
```

5. Sisipkan sebuah kolom dengan nama phone VARCHAR(15) setelah kolom alamat.
- Kita gunakan perintah yang sama dengan sebelumnya hanya penambahan kata FIRST diganti menjadi AFTER nama_field
- **ALTER TABLE nama_table ADD COLUMN nama_field [type_data](ukuran) AFTER nama_field**

```
MariaDB [latihan1]> alter table biodata add column phone varchar(15) after alamat;  
Query OK, 0 rows affected (0.66 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [latihan1]> desc biodata;
```

| Field | Type | Null | Key | Default | Extra |
|------------|-------------|------|-----|---------|-------|
| id | int(11) | YES | | NULL | |
| nama | varchar(50) | YES | | NULL | |
| alamat | text | YES | | NULL | |
| phone | varchar(15) | YES | | NULL | |
| keterangan | varchar(15) | YES | | NULL | |

5 rows in set (0.04 sec)

6. Ubah type_data kolom **id** menjadi **CHAR(11)**

Untuk mengubah field yang kita inginkan kita dapat menggunakan perintah;

➤ ALTER TABLE nama_table MODIFY nama_field [type_data](ukuran);

```
MariaDB [latihan1]> alter table biodata modify id char(11);  
Query OK, 0 rows affected (0.93 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [latihan1]> desc biodata;
```

| Field | Type | Null | Key | Default | Extra |
|------------|-------------|------|-----|---------|-------|
| id | char(11) | YES | | NULL | |
| nama | varchar(50) | YES | | NULL | |
| alamat | text | YES | | NULL | |
| phone | varchar(15) | YES | | NULL | |
| keterangan | varchar(15) | YES | | NULL | |

5 rows in set (0.04 sec)

7. Ubah nama kolom **phone** menjadi **hp varchar (20)**

Seperti sebelumnya kita menggunakan **alter table** dengan **MODIFY** diganti dengan **CHANGE**

- ALTER TABLE nama_table CHANGE nama_field_lama nama_field_baru [type_data](ukuran);

```
MariaDB [latihan1]> alter table biodata change phone hp varchar(20);
Query OK, 0 rows affected (0.95 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc biodata;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | char(11)  | YES  |     | NULL    |       |
| nama  | varchar(50)| YES  |     | NULL    |       |
| alamat| text      | YES  |     | NULL    |       |
| hp    | varchar(20)| YES  |     | NULL    |       |
| keterangan| varchar(15)| YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.04 sec)
```

8. Hapus kolom keterangan dari table biodata

Untuk menghapus field yang kita inginkan, kita gunakan perintah:

- ALTER TABLE nama_table DROP field_yang_ingin_dihapus;

```
MariaDB [latihan1]> alter table biodata drop keterangan;
Query OK, 0 rows affected (0.85 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc biodata;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | char(11)  | YES  |     | NULL    |       |
| nama  | varchar(50)| YES  |     | NULL    |       |
| alamat| text      | YES  |     | NULL    |       |
| hp    | varchar(20)| YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.04 sec)
```

9. Ganti nama tabel menjadi **data_mahasiswa**

Untuk mengganti nama table kita langsung saja gunakan **RENAME** seperti di bawah ini:

- RENAME TABLE nama_table TO nama_table_baru;

```
MariaDB [latihan1]> rename table biodata to data_mahasiswa;
Query OK, 0 rows affected (0.23 sec)

MariaDB [latihan1]> des
```

10. Ganti nama field **id** menjadi **nim**

Kita gunakan perintah seperti kita mengganti nama phone menjadi hp hanya saja nama table yang kita masukkan adalah nama tabel yang baru.

➤ ALTER TABLE nama_table CHANGE field_lama field_baru [type_data](ukuran);

```
MariaDB [latihan1]> alter table data_mahasiswa change id nim char(11);
Query OK, 0 rows affected (0.10 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc data_mahasiswa;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nim   | char(11)      | YES  |     | NULL    |       |
| nama  | varchar(50)   | YES  |     | NULL    |       |
| alamat | text          | YES  |     | NULL    |       |
| hp    | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.03 sec)
```

11. Jadikan NIM sebagai PRIMARY KEY

Dengan menjadikan nim sebagai primary key berarti kita telah menyetujui bahwa nim adalah kunci utama atau field unik yang datanya tidak boleh ada yang sama. Caranya adalah sebagai berikut:

➤ ALTER TABLE nama_table ADD PRIMARY KEY (nama_field);

```
MariaDB [latihan1]> alter table data_mahasiswa add primary key (nim);
Query OK, 0 rows affected (0.91 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [latihan1]> desc data_mahasiswa;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nim   | char(11)      | NO   | PRI | NULL    |       |
| nama  | varchar(50)   | YES  |     | NULL    |       |
| alamat | text          | YES  |     | NULL    |       |
| hp    | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.04 sec)
```

EVALUASI DAN PERTANYAAN

1. Tuliskan semua perintah SQL percobaan di atas beserta outputnya!
 - Perintah-perintah SQL dan Outputnya sudah kita bahas di atas.
2. Apa maksud int(11)?
 - Maksud dari int(11) adalah sebuah type data yang digunakan untuk menyimpan data bilangan bulat positif dan negative, sedangkan angka 11 yang ada di dalam kurung menyatakan length atau ukuran maksimal data tersebut.
3. Ketika kita melihat struktur table dengan perintah desc, ada kolom NULL yang berisi YES dan NO. apa maksudnya?
 - Maksudnya adalah apabila kolom Null berisi Yes maka data yang ada di kolom tersebut atau field tersebut dapat kosong atau tidak ada isinya, sedangkan apabila berisi No maka data yang ada di field tersebut tidak boleh kosong.
4. Buat laporan praktikum beserta langkah-langkah yang dilakukan dalam bentuk tutorial beserta screenshot outputnya (point 1)
 - Data tersebut sudah saya buat dan upload di channel youtube dan github saya.
5. Upload hasilnya pada classroom