



USM

PEMROGRAMAN WEB

Perintah Dasar SQL

Abstrak

Materi dasar-dasar pengenalan syntak SQL, sehingga saat sudah mulai diterapkan dengan menggunakan bahasa pemrograman lain semisal PHP. Diharapkan mahasiswa sudah paham dan tidak bingung saat akan menampilkan data dengan PHP dan MySQL.

Materi tentang database SQL

Dalam mempelajari MySQL dengan mudah dapat menggunakan :

1. Menggunakan OS windows
2. Install Xampp di drive C:\ (versi baru menggunakan Mariadb)

Belajar perintah dasar dengan menggunakan CMD

1. Untuk bisa mengakses SQL,

- Silahkan pindah ke drive **C:**
- Pindah ke folder **/xampp/mysql/bin** dengan mengetikan **cd /xampp/mysql/bin** kemudian enter
- Untuk bisa masuk ke MySQL, maka perintahnya **mysql -u root** enter(ket : -u root adalah user root), apabila menggunakan password maka perintahnya **mysql -u root -p** enter(ket: user root password). Password diketikan setelah di enter, nanti akan ada perintah untuk menginputkan password.

```
C:\>cd /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 15
Server version: 10.1.38-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> _
```

Berikut ini tampilan untuk menggunakan password

```
c:\xampp\mysql\bin>mysql -u root -p
Enter password: _
```

- Sedangkan untuk keluar dengan perintah **quit;** atau **exit;**.
2. Perintah untuk melihat database yang ada di mysql yaitu **SHOW DATABASES;** , setiap menuliskan perintah sql diakhiri dengan tanya titik koma (;)
 3. Perintah untuk membuat database, **create database nama_database**. Misal mau membuat database dengan nama db_kampus, maka perintahnya **create database db_kampus;**.

```
MariaDB [(none)]> create database db_kampus;  
Query OK, 1 row affected (0.01 sec)
```

4. Karena didalam mysql tadi ada banyak database, maka untuk mengaktifkan atau menggunakan database yang akan kita gunakan dalam contoh kali ini. Kita akan menggunakan database yang sudah kita buat tadi dengan mengetikan perintah **use nama_database**, maka perintahnya **use db_kampus;** .

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

5. Untuk selanjutnya kita akan belajar membuat tabel sederhana dan simpel karena buat latihan, maka kita akan membuat tabel dengan nama tbl_mhs dengan ketentuan sebagai berikut :

Nama Field	Type Filed	Keterangan
id_mhs	Int(11)	Auto_increment, primary_key
nim	Varchar(15)	Not null
nama	Varchar(50)	Not null

Perintah sqlnya :

CREATE TABLE nama_tabel (field_name field_type field_size);

Maka caranya sebagai berikut :

**create table tbl_mhs
(id_mhs int(11) auto_increment,
nim varchar(15) not null,
nama varchar(50) not null,
primary key(id_mhs));**

```
MariaDB [db_kampus]> create table tbl_mhs  
-> (id_mhs int(11) auto_increment,  
-> nim varchar(15) not null,  
-> nama varchar(50) not null,  
-> primary key(id_mhs));  
Query OK, 0 rows affected (0.20 sec)  
  
MariaDB [db_kampus]>
```

6. Untuk mengecek apakah tabel ada didalam database db_kampus, maka perintahnya **SHOW TABLES;**.

```
MariaDB [db_kampus]> show tables;
+-----+
| Tables_in_db_kampus |
+-----+
| tbl_mhs              |
+-----+
1 row in set (0.00 sec)

MariaDB [db_kampus]>
```

7. Perintah **DESCRIBE** atau **DESC** adalah perintah untuk menampilkan diskripsi tabel, dengan perintah **desc nama_tabel;**. Sebagai contoh kita akan menampilkan diskripsi tabel `tbl_mhs`, maka perintahnya **desc tbl_mhs;**.

```
MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)   | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15) | NO   |     | NULL    |                |
| nama   | varchar(50) | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.04 sec)
```

8. Perintah untuk mengecek data yang ada didalam tabel yaitu dengan perintah **SELECT * FROM nama_tabel;**. Untuk contoh kita akan mengecek `tbl_mhs` dengan perintah sebagai berikut **select * from tbl_mhs;**, silahkan perhatikan gambar dibawah ini. Tertulis **“Empty set (0.00 sec)”**, yang artinya tidak ada data dan membutuhkan waktu proses 0.00 sec.

```
MariaDB [db_kampus]> select * from tbl_mhs;
Empty set (0.00 sec)

MariaDB [db_kampus]>
```

9. Perintah untuk menginputkan data ke dalam tabel `tbl_mhs` yaitu **INSERT INTO nama_tabel (nama_field) VALUES ('inputan_data');**. Sebagai contoh kita inputkan data di `tbl_mhs` sebagai berikut: **insert into tbl_mhs (nim,nama) values ('G.211.99.0009','Alfian');**. Perhatikan gambar dibawah ini **“Query OK, 1 row affected (0.07 sec)”**, query ok yang berarti perintah sql benar dan 1 row yang berarti 1 data telah berhasil ditambahkan dengan waktu 0.07 sec.

```
MariaDB [db_kampus]> insert into tbl_mhs
-> (nim,nama) values
-> ('G.211.99.0009','Alfian');
Query OK, 1 row affected (0.07 sec)

MariaDB [db_kampus]> _
```

NB : Disini saya tidak menggunakan `id_mhs`, karena `id_mhs` bersifat auto increment dan primary key. Secara otomatis akan terisi dengan nomor urut.

Kemudian kita mengecek apakah data sudah masuk ke tabel `tbl_mhs`, dengan mengetikkan **`select * from tbl_mhs;`** .

```
MariaDB [db_kampus]> select * from tbl_mhs;
+-----+-----+-----+
| id_mhs | nim          | nama   |
+-----+-----+-----+
|      1 | G.211.99.0009 | Alfian |
+-----+-----+-----+
1 row in set (0.00 sec)

MariaDB [db_kampus]> _
```

10. Berikut ini cara untuk menginputkan multi data atau lebih dari satu inputan data, dengan cara sebagai berikut :

`insert into tbl_mhs (nim, nama) values`
`('G.211.98.0021','Alpha'),`
`('G.211.97.0002','Omega');`

```
MariaDB [db_kampus]> insert into tbl_mhs
-> (nim,nama)values
-> ('G.211.98.0021','Alpha'),
-> ('G.211.97.0002','Omega');
Query OK, 2 rows affected (0.10 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

Kemudian kita cek dengan **`select * from tbl_mhs;`** , seperti gambar dibawah ini :

```
MariaDB [db_kampus]> select * from tbl_mhs;
+-----+-----+-----+
| id_mhs | nim          | nama   |
+-----+-----+-----+
|      1 | G.211.99.0009 | Alfian |
|      2 | G.211.98.0021 | Alpha  |
|      3 | G.211.97.0002 | Omega  |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

11. Mengubah data yang ada didalam tabel `tbl_mhs`, dengan perintah **UPDATE nama_tabel SET nama_field="data_baru" where id_field=id'**;. Sebagai contoh kita akan mengubah nama Omega menjadi Kalkulus dengan `id_mhs=3`, maka perintahnya **update tbl_mhs set nama='Kalkulus' where id_mhs='3'**;

```
MariaDB [db_kampus]> update tbl_mhs set nama="Kalkulus" where id_mhs='3';
Query OK, 1 row affected (0.07 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Kita cek apakah berhasil di ubah dengan menggunakan perintah **select * from tbl_mhs;**, seperti gambar dibawah ini.

```
MariaDB [db_kampus]> select * from tbl_mhs;
+-----+-----+-----+
| id_mhs | nim           | nama    |
+-----+-----+-----+
|      1 | G.211.99.0009 | Alfian  |
|      2 | G.211.98.0021 | Alpha   |
|      3 | G.211.97.0002 | Kalkulus|
+-----+-----+-----+
3 rows in set (0.00 sec)
```

12. Sedangkan untuk perintah menghapus yaitu dengan mengetikan **DELETE FROM nama_tabel WHERE id_field=id'** ; , untuk contohnya kita akan menghapus data ditabel `tbl_mhs` tepatnya di `id_mhs='3'`. Untuk perintahnya **delete from tbl_mhs where id_mhs='3'**;

```
MariaDB [db_kampus]> delete from tbl_mhs where id_mhs='3';
Query OK, 1 row affected (0.12 sec)
```

Apakah data berkurang, mari kita cek dengan melihat gambar dibawah ini:

```
MariaDB [db_kampus]> select * from tbl_mhs;
+-----+-----+-----+
| id_mhs | nim           | nama    |
+-----+-----+-----+
|      1 | G.211.99.0009 | Alfian  |
|      2 | G.211.98.0021 | Alpha   |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

13. Untuk menampilkan data tertentu saja maka perintah nya sebagai berikut, **SELECT nama_field FROM nama_tabel;**. Sebagai contoh kita akan menampilkan data nama saja di tabel `tbl_mhs`, maka perintahnya **select nam from tbl_mhs;** .

```
MariaDB [db_kampus]> select nama from tbl_mhs;
+-----+
| nama  |
+-----+
| Alfian |
| Alpha  |
+-----+
2 rows in set (0.00 sec)
```

14. Menambahkan field baru didalam tabel dengan mengetikan **ALTER TABLE nama_tabel ADD nama_field type_field (field_size);**, maka mengetikannya didalam tabel tbl_mhs sebagai berikut : **alter table tbl_mhs add jurusan varchar(20);** .

```
MariaDB [db_kampus]> alter table tbl_mhs add jurusan varchar(20);
Query OK, 0 rows affected (0.52 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Mari kita cek apakah sudah bertambah field baru dengan cara **DESC nama_tabel**, dengan mengetikan desc tbl_mhs;.

```
MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)       | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15)   | NO   |     | NULL    |                |
| nama   | varchar(50)   | NO   |     | NULL    |                |
| jurusan | varchar(20)  | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.07 sec)
```

15. Selanjutnya yaitu bagaimana menghapus field didalam tabel, namun sebelumnya untuk contoh silahkan ditambahkan field baru dengan nama salah varchar(2). Untuk menambah kita gunakan **alter table tbl_mhs add salah varchar(2);**, kemudian kita cek dengan **desc tbl_mhs** seperti gambar dibawah ini.

```
MariaDB [db_kampus]> alter table tbl_mhs add salah varchar(2);
Query OK, 0 rows affected (0.53 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)       | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15)   | NO   |     | NULL    |                |
| nama   | varchar(50)   | NO   |     | NULL    |                |
| jurusan | varchar(20)  | YES  |     | NULL    |                |
| salah  | varchar(2)    | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)
```

Kemudian kita menghapus field salah tadi dengan cara **ALTER TABLE nama_tabel DROP nama_field;**, sehingga kalau kita ketik perintahnya **alter table tbl_mhs drop salah;**.

```
MariaDB [db_kampus]> alter table tbl_mhs drop salah;
Query OK, 0 rows affected (0.55 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Kita cek dengan desc tbl_mhs, seperti gambar dibawah ini:

```
MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)       | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15)   | NO   |     | NULL    |                |
| nama   | varchar(50)   | NO   |     | NULL    |                |
| jurusan | varchar(20)  | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.07 sec)
```

16. Untuk mengubah nama field kita gunakan perintah sebagai berikut :

**ALTER TABLE nama_tabel
CHANGE COLUMN nama_field_lama nama_field_baru
type_field_data_lama;**

untuk studi kasus kita tambahkan dulu field salah, dengan cara **alter table tbl_mhs add salah varchar(2);**. Kemudian baru kita ubah field salah menjadi field benar, dengan cara sebagai berikut :

**alter table tbl_mhs
change column salah baru
varchar(2);**

seperti gambar dibawah ini dan kita cek sekaligus dengan desc tbl_mhs.

```
MariaDB [db_kampus]> alter table tbl_mhs
-> change column salah baru
-> varchar(2);
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)       | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15)   | NO   |     | NULL    |                |
| nama   | varchar(50)   | NO   |     | NULL    |                |
| jurusan | varchar(20)  | YES  |     | NULL    |                |
| baru   | varchar(2)    | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)
```


Sedangkan untuk mengubah type field atau size dengan cara hampir sama yaitu :

ALTER TABLE nama_tabel

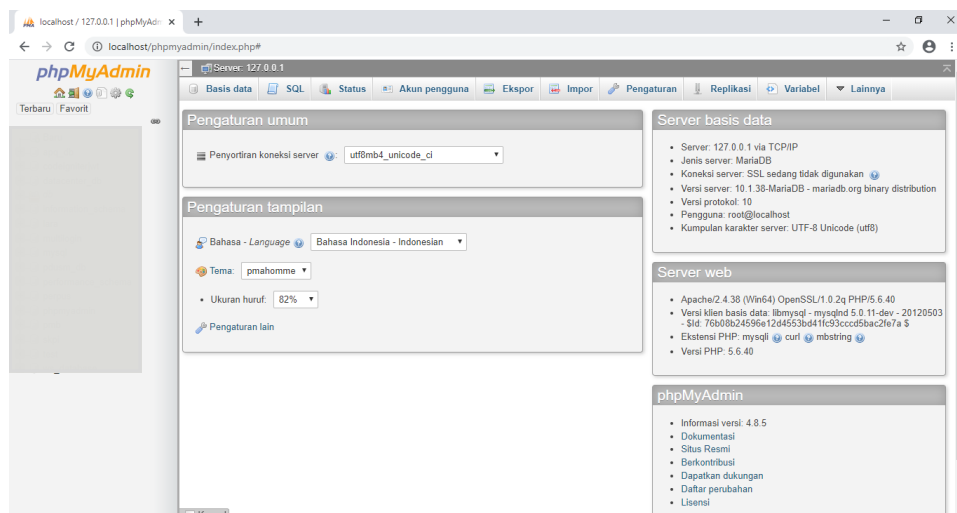
***CHANGE COLUMN nama_field_lama nama_field_lama
type_field_baru field_size_baru null/not null;***

untuk melihat contoh silahkan lihat gambar dibawah ini:

```
MariaDB [db_kampus]> alter table tbl_mhs
-> change column baru baru
-> int(2)not null;
Query OK, 2 rows affected, 2 warnings (0.79 sec)
Records: 2 Duplicates: 0 Warnings: 2

MariaDB [db_kampus]> desc tbl_mhs;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id_mhs | int(11)       | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(15)   | NO   |     | NULL    |                |
| nama   | varchar(50)   | NO   |     | NULL    |                |
| jurusan | varchar(20)  | YES  |     | NULL    |                |
| baru   | int(2)        | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

Cukup disini untuk pengenalan syntax sql, saya rasa untuk dasar-dasar sudah cukup. Silahkan dipraktekan dirumah atau pada Pelajaran Praktikum di lab-lab yang sudah disediakan. Untuk langkah-langkah selanjutnya karena sudah paham akan query sql dapat menggunakan tools, untuk standart bisa menggunakan phpMyAdmin. Untuk bisa mengkases bisa menggunakan browser, kemudian ketikan di url <http://localhost/phpmyadmin>.



Rangkuman:

1. Perintah masuk ke MySQL
`"mysql -u root -p"`
2. Membuat database di MySQL
`"CREATE DATABASE nama_database;"`
3. Melihat database yang ada di MySQL
`"SHOW DATABASES;"`
4. Mengaktifkan database yang akan digunakan
`"USE nama_database;"`
5. Melihat tabel di database yang aktif
`"SHOW TABLES;"`
6. Membuat tabel
`"CREATE TABLE nama_tabel (field_name field_type field_size);"`
7. Melihat deskripsi tabel
`"DESC nama_tabel;"`
8. Menginputkan data di dalam tabel
`"INSERT INTO nama_tabel (nama_field)VALUES('data_inputan');"`
9. Melihat isi data didalam tabel
`"SELECT * FROM nama_tabel;"`
10. Mengubah data yang ada di dalam tabel
`"UPDATE nama_tabel SET nama_field='data_baru' WHERE id_field='id';"`
11. Menghapus data didalam tabel
`"DELETE FROM nama_tabel WHERE id_field='id';"`
12. Menampilkan data tertentu saja
`"SELECT nama_field FROM nama_tabel;"`
13. Menambahkan field baru ditabel
`"ALTER TABLE nama_tabel ADD nama_field field_type (field_size);"`
14. Menghapus field yang ditabel
`"ALTER TABLE nama_tabel DROP nama_field;"`
15. Mengubah nama field dan mengubah type field
 - Mengubah nama field
`ALTER TABLE nama_tabel`

```
CHANGE COLUMN nama_field_lama nama_field_baru  
type_field_data_lama;
```

- Mengubah type field

```
ALTER TABLE nama_tabel  
CHANGE COLUMN nama_field_lama nama_field_lama  
type_field_baru field_size_baru null/not null;
```

PENUTUP

Maaf apabila ada kesalahan kata dalam penulisan atau pun salah dalam penyusunan modul ini. Untuk modul selanjutnya akan diterangkan tentang dasar-dasar php, dan modul-modul yang berkaitan dengan matakuliah pemrograman web. Yang nantinya akan dibuat juga modul CRUD menggunakan php dan mysql.

DAFTAR PUSTAKA

Nugroho, B. (2008). *Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL*. yogyakarta: Gava Media.

tutorialspoint. (n.d.). *MySQL Tutorial*. Retrieved from tutorialspoint.com:
<https://www.tutorialspoint.com/mysql/index.htm>

w3schools. (n.d.). *SQL Tutorial*. Retrieved from w3schools: <https://www.w3schools.com/sql/default.asp>