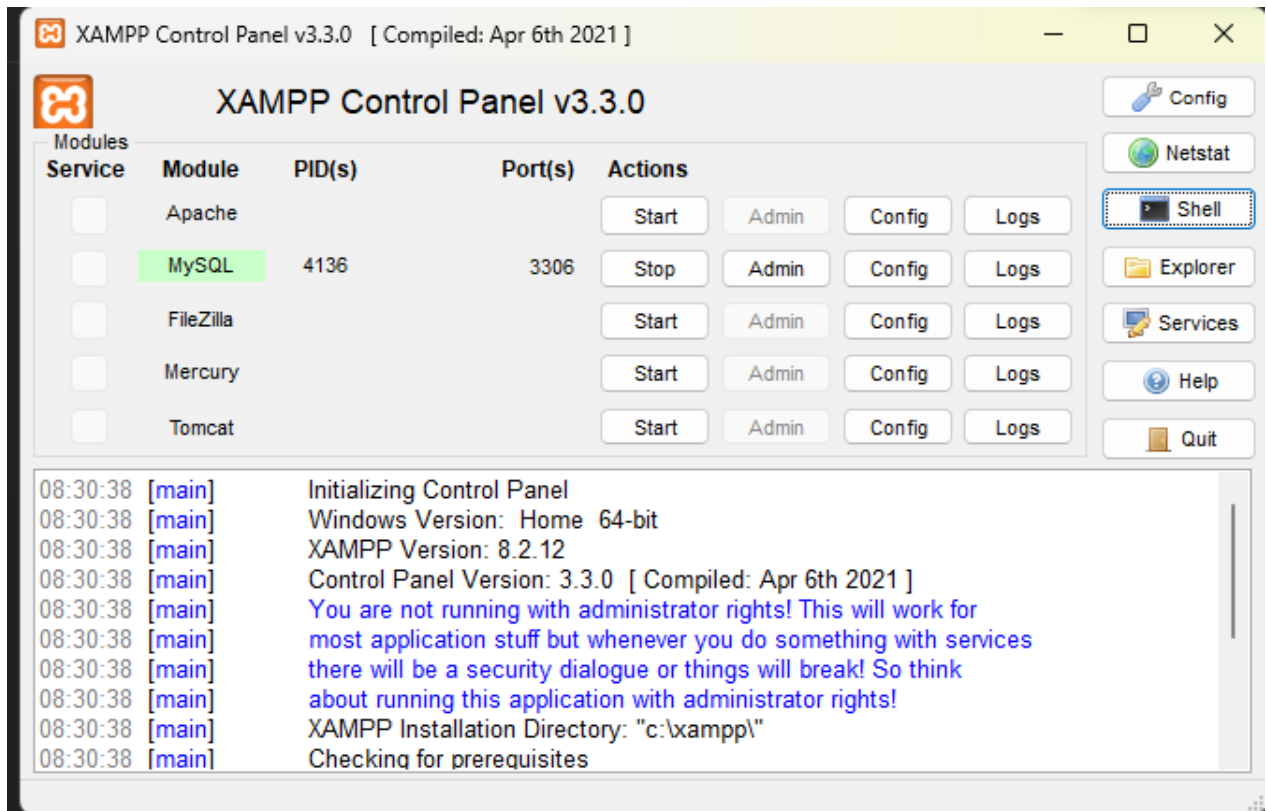


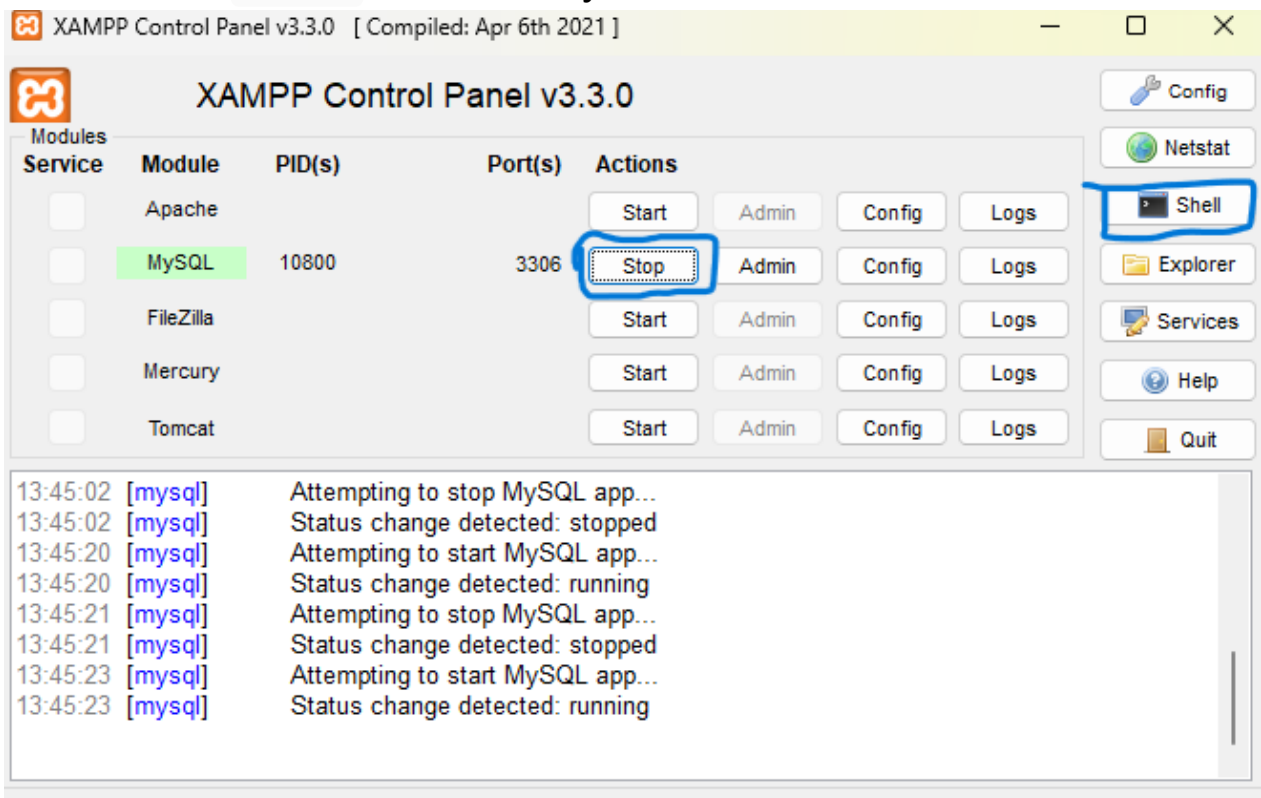
Instalasi MySQL

Menggunakan XAMPP

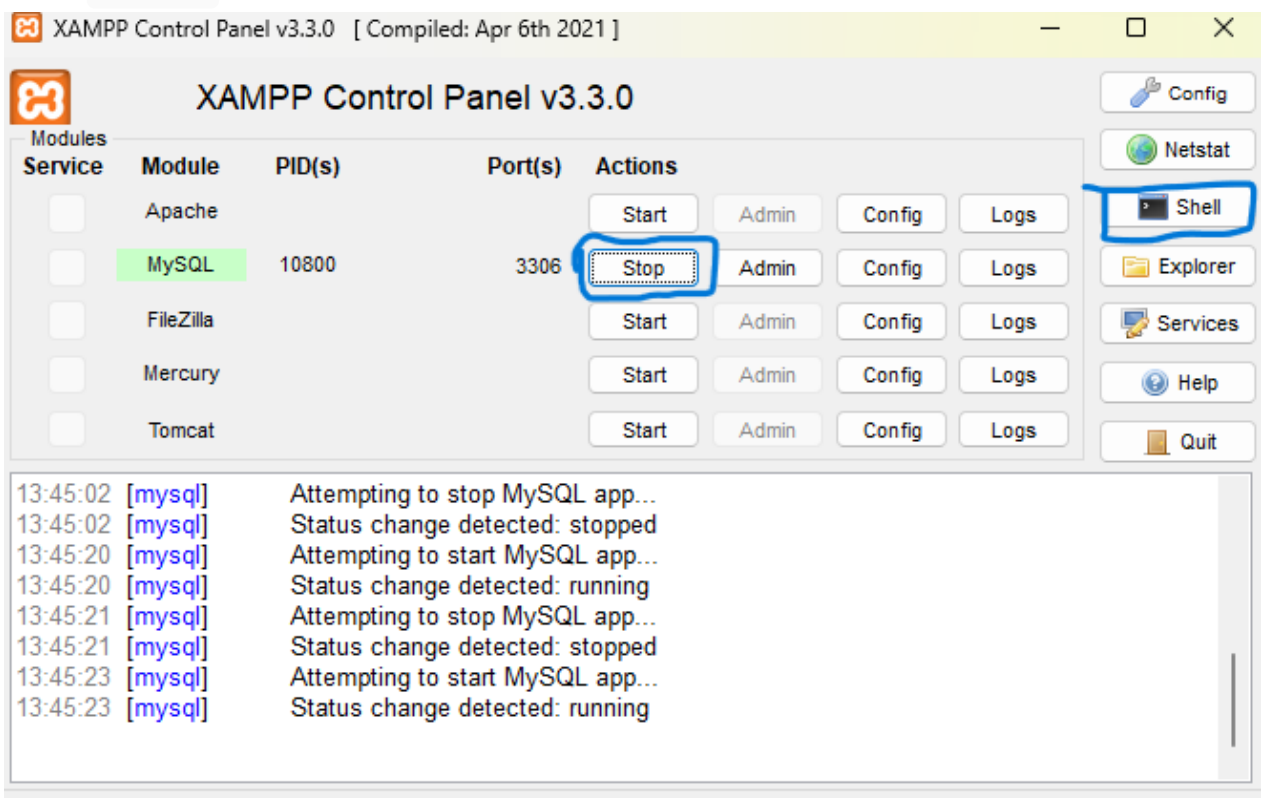
1. Buka Aplikasi XAMPP



2. Klik tombol `start` di menu `mysql`



3. Klik `shell`



4. Masuk ke database dengan akun administrator `mysql -u root -p`

Passwordnya kosong jadi silahkan langsung enter

```
ASUS@NAYAH c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 9
Server version: 10.4.32-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)]>
```

5. Buat Database

- `CREATE DATABASE [nama_database];`

6. Tampilkan Data Base

- `SHOW DATABASES ;`

7. Hentikan Database

- `DROP DATABASE [nama_database];`

8. Menggunakan Database

- `USE [nama_database];`

Referensi Video Youtube

PHP + MySQL Termux

<https://youtu.be/ez3nx3xH-y4?si=T4saycipqfBcqL1c>

Error MySQL Can't Run

<https://youtube.com/shorts/k46S7SrNgFQ?si=F3QXRQBTo5291DqA>

Error Can't Install Apps

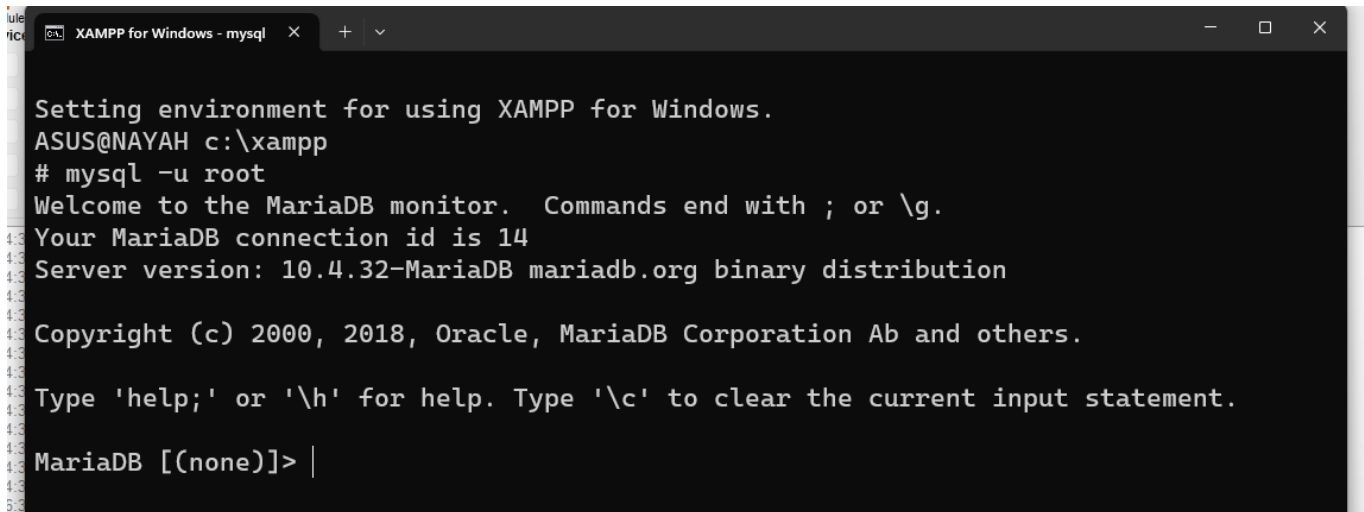
<https://youtu.be/xRvjfnJVqzM?si=pna1uARaWJTqj6mL>

Penggunaan Awal MySQL

Query

```
mysql -u root p;
```

Hasil

A screenshot of a terminal window titled "XAMPP for Windows - mysql". The terminal shows the following text: "Setting environment for using XAMPP for Windows.", "ASUS@NAYAH c:\xampp", "# mysql -u root", "Welcome to the MariaDB monitor. Commands end with ; or \g.", "Your MariaDB connection id is 14", "Server version: 10.4.32-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.", and "MariaDB [(none)]> |".

```
Setting environment for using XAMPP for Windows.
ASUS@NAYAH c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 14
Server version: 10.4.32-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)]> |
```

Analisis Kesimpulan

Kode MySQL tersebut adalah argumen yang digunakan saat menjalankan perintah melalui baris perintah (command line) untuk mengakses server MySQL. Berikut penjelasan masing-masing argumen:

1. `"mysql"` : Ini adalah perintah untuk memanggil program MySQL dari baris perintah.
2. `"-u"` : Ini diikuti oleh nama pengguna (username) yang akan digunakan untuk masuk ke server MySQL. Dalam contoh ini, `"root"` adalah nama pengguna yang digunakan.

3. `"root"` : Merupakan nilai dari argumen `-u`, yaitu nama pengguna yang digunakan untuk masuk ke server MySQL.
4. `"-p"` : Ini menunjukkan bahwa setelahnya akan diminta password pengguna untuk masuk. Jika tidak disertakan secara langsung di baris perintah, MySQL akan meminta password secara interaktif setelah Anda menekan Enter.

Jadi, keseluruhan kode tersebut digunakan untuk menjalankan perintah `MySQL` dengan nama pengguna `"root"` dan meminta password untuk masuk ke server MySQL.

Data Base

Database adalah **koleksi data yang sistematis dan sistematis yang disimpan secara elektronik**. Ini dapat berisi semua jenis data, termasuk kata, angka, gambar, video, dan file. Anda dapat menggunakan perangkat lunak yang disebut sistem manajemen database (DBMS) untuk menyimpan, mengambil, dan mengedit data.

Buat Database

QUERY:

```
CREATE DATABASE [NAMA DATA_BASE]
```

```
create database nayah_comel;
```

HASIL:

```
XAMPP for Windows - mysql X + v

Setting environment for using XAMPP for Windows.
ASUS@NAYAH c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.32-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)]> create database nayah_comel;
Query OK, 1 row affected (0.004 sec)

MariaDB [(none)]> show databases;
```

ANALISIS KESIMPULAN:

CREATE DATABASE digunakan untuk membuat basis data yang baru.

Tampilkan Database

****QUERY**

```
SHOW DATABASES;
```

HASIL:

```
XAMPP for Windows - mysql X + v

Setting environment for using XAMPP for Windows.
ASUS@NAYAH c:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.32-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)]> create database nayah_comel;
Query OK, 1 row affected (0.004 sec)

MariaDB [(none)]> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| nayah_comel             |
| performance_schema      |
| phpmyadmin              |
| test                    |
| xi_rpl_1                |
+-----+
7 rows in set (0.016 sec)

MariaDB [(none)]>
```

ANALISIS KESIMPULAN:

`SHOW DATABASES` digunakan untuk menampilkan data basenya.

Hapus Database

QUERY:

```
DROP DATABASE [NAMA DATA_BASE];
```

```
drop database nayah_comel;
```

HASIL:

```
XAMPP for Windows - mysql
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| nayah_comel |
| performance_schema |
| phpmyadmin |
| test |
| xi_rpl_1 |
+-----+
7 rows in set (0.003 sec)

MariaDB [(none)]> drop database nayah_comel;
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
| xi_rpl_1 |
+-----+
6 rows in set (0.003 sec)

MariaDB [(none)]>
```

ANALISIS KESIMPULAN:

`DROP DATABASE` adalah untuk menghapus databasenya**.

Gunakan Database

QUERY:

```
`USE [NAMA DATA_BASE]`
```



```
use nayah_comel;
```

HASIL:

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| nayah_comel |
| performance_schema |
| phpmyadmin |
| test |
| xi_rpl_1 |
+-----+
7 rows in set (0.002 sec)

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

ANALISI KESIMPULAN:

`USE` , statement ini **digunakan untuk berkomunikasi dengan MySQL untuk menggunakan suatu database sebagai database default untuk statement selanjutnya.**

Tipe Data

Data type atau tipe data adalah **klasifikasi value suatu variabel yang membantu operasi komputer untuk mengenali nilai data tanpa menghasilkan output error.** Tipe data memudahkan program dalam mendeteksi berbagai jenis informasi.

Angka

- `INT` : Menyimpan bilangan bulat dalam rentang tertentu.

- **TINYINT** : Menyimpan bilangan bulat kecil.
- **BIGINT** : Menyimpan bilangan bulat besar.
- **DECIMAL**: Menyimpan angka desimal dengan presisi yang ditentukan.
- **FLOAT** : Menyimpan angka desimal dengan presisi yang lebih rendah daripada DECIMAL.
- **DOUBLE** : Menyimpan angka desimal dengan presisi ganda daripada FLOAT.

Teks

- **CHAR** : Menyimpan string dengan panjang tetap.
- **VARCHAR** : Menyimpan string dengan panjang variabel.
- **TEXT** : Menyimpan teks panjang yang lebih besar daripada VARCHAR.

Tanggal

- **DATE** : Menyimpan tanggal (YYYY-MM-DD).
- **TIME** : Menyimpan waktu (HH:MM:SS).
- **DATETIME** : Menyimpan tanggal dan waktu (YYYY-MM-DD HH:MM:SS)

Boolean

- **BOOLEAN** : Menyimpan nilai benar (TRUE) atau salah (FALSE).

Binary

- **BLOB** : Menyimpan data biner seperti gambar atau file.

Lainnya

- **ENUM** : Menyimpan satu nilai dari daftar nilai yang ditentukan.
- **SET** : Menyimpan beberapa nilai dari daftar nilai yang ditentukan.

Tabel

Buat Tabel

Query

```
create table mobil;
```

Hasil

```
ERROR 1113 (42000): A table must have at least 1 column  
MariaDB [rental_nayah]> create table mobil  
-> |
```

Analisis Kesimpulan

Perintah **CREATE TABLE** digunakan dalam sql untuk membuat sebuah tabel baru dalam basis data.

Berikut adalah penjelasan bagaimana menggunakan perintah **CREATE TABLE**

- Kata kunci **CREATE TABLE** diikuti oleh nama tabel yang ingin dibuat.

Tampilkan Struktur Table

Query

```
create table [nama_table] ;
```

```
create table mobil (  
  nama_mobil varchar(15) primary key not null,  
    plat_mobil char(10) not null unique,  
    warna_mobil varchar(10) not null unique);
```

Hasil

```
type help; 01 (M) 101 help; type (e) to read the callen  
MariaDB [(none)]> use rental_nayah;  
Database changed  
MariaDB [rental_nayah]> create table mobil (  
  -> nama_mobil varchar(15) primary key not null,  
  -> plat_mobil char(10) not null unique,  
  -> warna_mobil varchar(10) not null unique);  
Query OK, 0 rows affected (0.025 sec)
```

Analisis Kesimpulan

1. Kolom "nama mobil" didefinisikan sebagai `VARCHAR(15)`, yang berarti itu akan menyimpan data string dengan panjang maksimal 15 karakter. Kemudian, `PRIMARY KEY` menandakan bahwa kolom ini akan menjadi kunci utama untuk tabel, yang berarti nilainya harus unik dan tidak boleh kosong (`NOT NULL`).
2. Kolom "plat mobil" didefinisikan sebagai `CHAR(10)`, yang berarti itu akan menyimpan data karakter dengan panjang tetap sebanyak 10 karakter. Selanjutnya, `NOT NULL` menunjukkan bahwa kolom ini harus memiliki nilai (tidak boleh kosong), dan

`UNIQUE` menandakan bahwa nilai-nilai dalam kolom ini harus unik (tidak ada duplikat).

3. Kolom "warna mobil" didefinisikan sebagai `VARCHAR(10)` , yang berarti itu akan menyimpan data string dengan panjang maksimal 10 karakter Seperti kolom sebelumnya, `NOT NULL` menunjukkan bahwa kolom ini harus memiliki nilai (tidak boleh kosong), dan `UNIQUE` menandakan bahwa nilai-nilai dalam kolom ini harus unik (tidak ada duplikat)

Struktur Table

Query

```
describe [nama_tabel]
```

```
describe mobil;
```

Hasil

```
MariaDB [rental_nayah]> describe mobil;
```

| Field | Type | Null | Key | Default | Extra |
|-------------|-------------|------|-----|---------|-------|
| nama_mobil | varchar(15) | NO | PRI | NULL | |
| plat_mobil | char(10) | NO | UNI | NULL | |
| warna_mobil | varchar(10) | NO | UNI | NULL | |

3 rows in set (0.023 sec)

Analisis Kesimpulan

Perintah `DESCRIBE` digunakan dalam `SQL` untuk mendapatkan informasi tentang struktur kolom dalam sebuah tabel.

Ini berguna untuk mengetahui nama kolom, tipe data, dan batasan yang diterapkan pada kolom tersebut. Berikut adalah penjelasan mengenai penggunaan perintah `DESCRIBE`

1. Kata kunci `DESCRIBE` diikuti oleh nama tabel yang ingin Anda periksa. Misalnya, jika Anda ingin mengetahui informasi mengenai tabel dengan nama mobil
2. Setelah menjalankan perintah `DESCRIBE` , Anda akan mendapatkan hasil yang menampilkan informasi mengenai kolom-kolom dalam tabel tersebut. Informasi yang biasanya ditampilkan antara lain:
 - Nama kolom (column name)
 - Tipe data kolom (data type)
 - Panjang maksimal kolom (maximum length)
 - Apakah kolom dapat memiliki nilai NULL atau tidak (nullable)
 - Apakah kolom memiliki batasan unik (unique)

Menampilkan daftar tabel

Query

```
show tables;
```

Hasil

```
MariaDB [rental_nayah]> show tables;
+-----+
| Tables_in_rental_nayah |
+-----+
| mobil                   |
| pelanggan               |
+-----+
2 rows in set (0.002 sec)
```

Analisis Kesimpulan

Perintah `SHOW TABLES` digunakan dalam `SQL` untuk menampilkan daftar tabel yang ada dalam basis data.

Ini memungkinkan Anda melihat tabel-tabel yang telah dibuat sebelumnya. Setelah menjalankan perintah `SHOW TABLES`, Anda akan mendapatkan hasil yang menampilkan daftar tabel yang ada dalam basis data tersebut. Hasilnya dapat berupa daftar tabel dalam bentuk kolom tunggal atau kolom ganda, tergantung pada implementasi sistem manajemen basis data (DBMS) yang digunakan

QnA

❓ mengapa hanya kolom `id_pelanggan` yang menggunakan constraint `PRIMARY KEY`?

karena `id_pelanggan` ini adalah cara untuk mengidentifikasi setiap baris dalam tabel, dan juga `primary key` digunakan untuk memberikan hak istimewa dari sebuah kolom dan tidak dapat diduplikat oleh setiap baris

❓ mengapa pada kolom no_telp yang menggunakan tipe data char bukan VARCHAR?

karena tipe data char menyimpan `string` karakter tetap dan no_telepon bersifat karakter tetap

❓ mengapa hanya kolom no_telp yang menggunakan constraint UNIQUE?

karena unique berfungsi untuk memastikan bahwa setiap baris data yg terdapat dalam suatu tabel bersifat unik (tidak sama) dan no_tlp pasti tidak akan sama

❓ mengapa kolom no_telp tidak memakai constraint NOT_NULL, sementara kolom lainnya, menggunakan constraint tersebut?

karena no_tlp dianggap opsional, nomor telepon hanya menjadi wajib isi saat pengguna melakukan langkah-langkah tertentu, tetapi mungkin tidak mewajibkan pengisian telepon pada awal pengisian data.

❓ apa perbedaan PRIMARY KEY dengan UNIQUE?

kalau `PRIMARY KEY` untuk membedakan data yang sama dan hanya boleh 1 dan tidak boleh tidak ada, sedangkan unique boleh 1,2,3, dan seterusnya dan boleh tidak ada

Insert

Insert 1 Data

Struktur Query

```
insert into [nama_tabel]
values (nilai1,nilai2,nilai3,...)
```

```
insert into pelanggan_rental
values (1,"nayah","merah","08123456789");
```

Hasil

```
MariaDB [rental_nayah]> select * from pelanggan_rental;
+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
+-----+-----+-----+-----+
| 1            | nayah          | merah       | 08123456789 |
+-----+-----+-----+-----+
1 row in set (0.002 sec)
```

Analisis Kesimpulan

- `insert into` adalah query yang digunakan untuk menginput isi tabel
- `pelanggan_rental` adalah nama tabelnya
- `values` adalah query yang digunakan untuk memasukkan nilai kekolom
- `(1,"nayah","merah","08123456789"` nilai yang ingin dimasukkan kekolom
- `;` penutup query

Insert >1 data

Struktur Query

```
insert into [nama_tabel]
values (2,"nilai2","nilai3","nilai4")
      (3"nilai2","nilai3","nilai4")
      (4,"nilai2","nilai3","nilai4")
```

```
insert into pelanggan_rental
values (2,"alya","biru","08546838374")
      (3,"adel","kuning","0867363833")
      (4,"bombom","hijau","0876733628")
```

Hasil

```
MariaDB [rental_nayah]> select * from pelanggan_rental;
+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
+-----+-----+-----+-----+
| 3            | adel           | kuning      | 0867363833 |
| 2            | alya           | biru        | 08546838374 |
| 4            | bombom         | hijau       | 0876733628  |
| 1            | nayah          | merah       | 08123456789 |
+-----+-----+-----+-----+
4 rows in set (0.002 sec)
```

Analisis Kesimpulan

- `insert into` query yang di gunakan untuk menginput isi table
- `pelanggan_rental` adalah nama table yang ingin di isi
- `values` query yang di gunakan untuk memasukkan nilai ke kolom

Select

Menyebut kolom

Struktur Query

```
insert into [nama_tabel]
(nama_kolom,nama_kolom) values
(nilai_1,nilai_2);
```

```
insert into pelanggan_rental
(nama_pelanggan,id_pelanggan) values
("reno",5);
```

Hasil

```
MariaDB [rental_nayah]> select * from pelanggan_rental;
+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
+-----+-----+-----+-----+
| 3            | adel           | kuning      | 0867363833 |
| 2            | alya          | biru        | 08546838374 |
| 4            | bombom        | hijau       | 0876733628  |
| 1            | nayah         | merah       | 08123456789 |
| 5            | reno          |             |             |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

Analisis Kesimpulan

- `insert into` query yg digunakan untuk menginput isi tabel
- `pelanggan_rental` adalah nama tabel yg ingin di isi
- `nama_pelanggan,id_pelanggan` nama kolom yg hanya akan diisi
- `values` query yg digunakan untu memasukkan nilai kekolom
- `reno,5` merupakan nilai yg akan dimasukkan

Seluruh Data

Struktur Query

```
select * from [nama_tabel];
```

```
select * from pelanggan_rental;
```

Hasil

```
MariaDB [rental_nayah]> select * from pelanggan_rental;
```

| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
|--------------|----------------|-------------|-------------|
| 3 | adel | kuning | 0867363833 |
| 2 | alya | biru | 08546838374 |
| 4 | bombom | hijau | 0876733628 |
| 1 | nayah | merah | 08123456789 |
| 5 | reno | | |

5 rows in set (0.001 sec)

Analisis Kesimpulan

- **Select** merupakan query yang digunakan untuk menampilkan hasil insert
- ***** artinya semua kolom yang ada di table akan di tampilkan
- **from** query yang digunakan untuk memberikan penanda bahwa table mana yang akan di tampilkan
- **pelanggan_rental** merupakan nama table yang isi nya akan di tampilkan

Data kolom Tertentu

Struktur Query

```
select [nama_kolom1],[nama_kolom2],  
        ..., [nama_kolom_n]  
from [nama_tabel];
```

```
select nama_pelanggan from pelanggan_rental;
```

Hasil

```
MariaDB [rental_nayah]> select nama_pelanggan from pelanggan_rental;  
+-----+  
| nama_pelanggan |  
+-----+  
| nayah          |  
| alya           |  
| adel           |  
| bombom         |  
| reno           |  
+-----+  
5 rows in set (0.001 sec)
```

Analisis Kesimpulan

- `Select` merupakan query yang digunakan untuk menampilkan hasil insert
- `nama_pelanggan` adalah nama kolom yang akan di tampilkan
- `from` query yang digunakan untuk memberikan penanda bahwa table mana yang akan di tampilkan
- `pelanggan_rental` merupakan nama table yang isi nya akan di tampilkan

Klausula Where

Struktur Query

```
select [nama_kolom / * ] from [nama_tabel]
where [kondisi];
```

```
select nama_pelanggan from pelanggan_rental
where id_pelanggan=2;
```

Hasil

```
MariaDB [rental_nayah]> select nama_pelanggan from pelanggan_rental
-> where id_pelanggan=2;
+-----+
| nama_pelanggan |
+-----+
| alya           |
+-----+
1 row in set (0.009 sec)
```

Analisis Kesimpulan

- **Select** merupakan query yang digunakan untuk menampilkan hasil insert
- **nama_pelanggan** adalah nama kolom yang akan di tampilkan
- **from** query yang digunakan untuk memberikan penanda bahwa table mana yang akan di tampilkan
- **pelanggan_rental** merupakan nama table yang isi nya akan di tampilkan

Update

Struktur query

```
update nama_tabel set nama_kolom where kondisi;
```

```
update pelanggan_rental set no_telepon="085340567087"
where id_pelanggan="1";
```

Hasil

```
MariaDB [rental_nayah]> update pelanggan_rental set no_telepon="085340567087" where id_pelanggan="1";
Query OK, 1 row affected (0.023 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [rental_nayah]> select * from pelanggan_rental;
+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
+-----+-----+-----+-----+
| 3           | adel           | kuning      | 0867363833 |
| 2           | alya           | biru        | 08546838374 |
| 4           | bombom         | hijau       | 0876733628 |
| 1           | nayah          | merah       | 085340567087 |
| 5           | reno           |             |             |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [rental_nayah]> |
```

Analisis Kesimpulan

- `update` adalah query yang digunakan untuk memperbarui nilai dari kolom
- `pelanggan_rental` nama table yang akan di perbarui nilai kolomnya
- `set` query yang digunakan untuk memberikan penanda bahwa yang nilainya akan di rubah
- `where` query yang digunakan untuk memberikan sebuah kondisi

Delete (hapus data)

Struktur Query

```
delete from nama_tabel where kondisi;
```

```
delete from pelanggan_rental where id="1";
```

Hasil

```
MariaDB [rental_nayah]> delete from pelanggan_rental where id_pelanggan="1";  
Query OK, 1 row affected (0.007 sec)
```

```
MariaDB [rental_nayah]> select * from pelanggan_rental;
```

| id_pelanggan | nama_pelanggan | warna_mobil | no_telepon |
|--------------|----------------|-------------|-------------|
| 3 | adel | kuning | 0867363833 |
| 2 | alya | biru | 08546838374 |
| 4 | bombom | hijau | 0876733628 |
| 5 | reno | | |

4 rows in set (0.001 sec)

Analisis Kesimpulan

- `delete` query yang digunakan untuk menghapus baris kolom
- `from` query yang digunakan untuk memberikan penanda bahwa table mana yang akan di hapus baris nya
- `pelanggan_rental` nama table yang akan di hapus baris nya
- `. where` query yang digunakan untuk memberikan sebuah kondisi
- `id _pelanggan` nama kolom nya
- `=` operatornya
- `1` nilai nya

Delete (Hapus Table)

Struktur Query

```
drop table [nama_table]
```



```
drop table pelanggan_rental;
```

Hasil

```
MariaDB [rental_nayah]> drop table pelanggan_rental;
Query OK, 0 rows affected (0.020 sec)

MariaDB [rental_nayah]> show tables;
+-----+
| Tables_in_rental_nayah |
+-----+
| mobil                  |
| pelanggan              |
+-----+
2 rows in set (0.003 sec)

MariaDB [rental_nayah]> |
```

Analisis Kesimpulan

- `drop table` query yang digunakan untuk menghapus sebuah table
- `pelanggan_rental` merupakan nama table yang akan di hapus