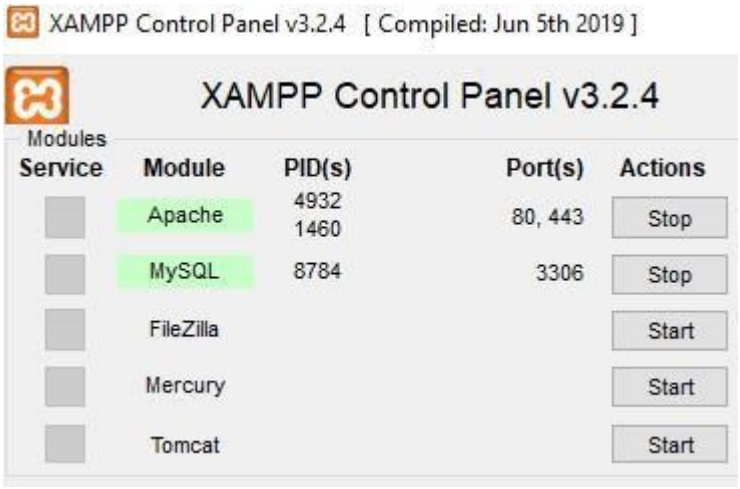
Ilyas Raihan Nadhif
L200180119
E

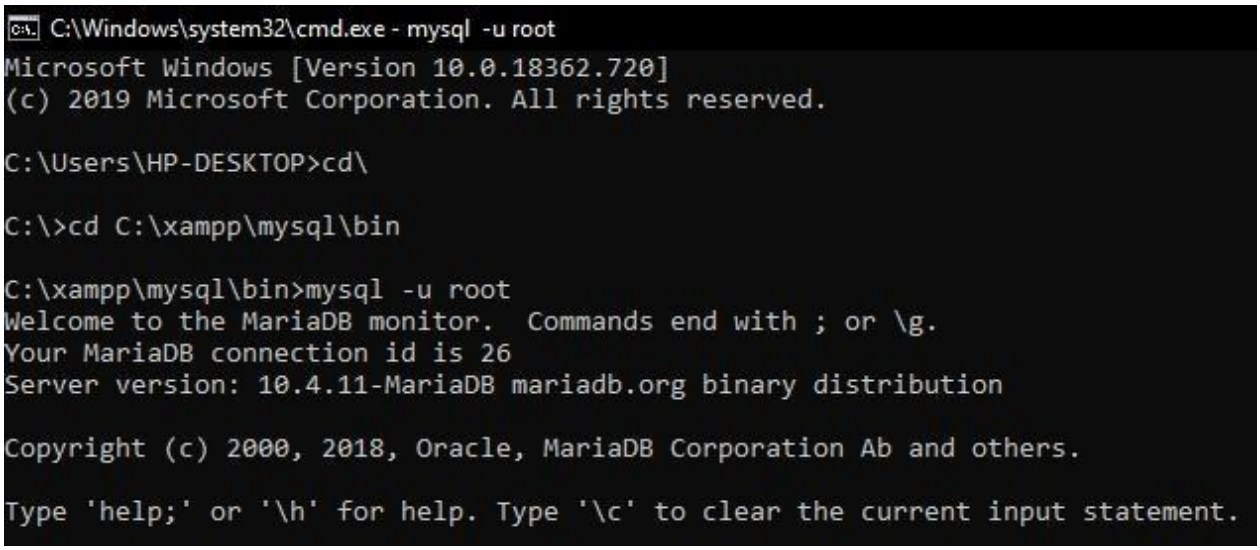
Modul 4

Kegiatan

1. Jalankan XAMPP Control Panel.
2. Jalankan server Apache dan MySQL.

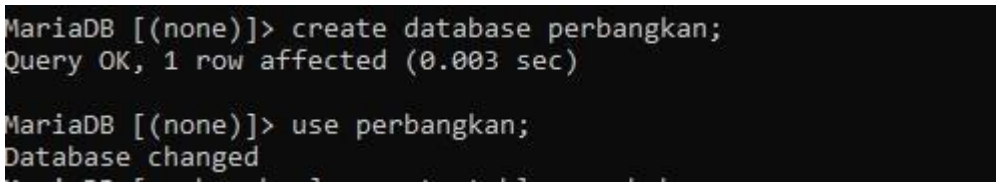


3. Buka Command Prompt dan login sebagai root ke MySQL seperti di langkah modul 1.



4. Buat database baru dengan perintah berikut. Create database perbankan;
5. hubungkan kedalam database yang telah dibuat dengan perintah berikut. Sehingga akan mmuncul pemberitahuan “database changed”.

   Use perbankan;



6. Membuat table nasabah dengan script berikut. CREATE TABLE nasabah (id_nasabah INTEGER PRIMARY KEY, nama_nasabah VARCHAR(45) NOT NULL, alamat_nasabah VARCHAR(255) NOT NULL,

   );
7. Membuat table cabang_bank dengan script berikut.

   CREATE TABLE(

   Kode_cabang VARCHAR(20) PRIMARY KEY,

   Nama_cabang VARCHAR(45) UNIQUE NOT NULL,

   Alamat_cabang VARCHAR(255) NOT NULL,
8. Membuat table rekening dengan script berikut. CREATE TABLE rekening(no_rekening integer primary key,

kode_cabangfk varchar(20) references cabang_bank(kode_cabang)
on delete cascade on update cascade, pin varchar(20) default „1234“
not null, saldo integer default 0 not null
);

9. Membuat table transaksi dengan script berikut ini.
   CREATE TABLE rekening(
   No_transaksi SERIAL PRIMARY KEY,
   Id_nasabahFK INTEGER REFERENCES nasabah(id_nasabah)
   ON DELETE SET NULL ON UPDATE CASCADE,
   No_rekening INTEGER REFERENCES rekening(no_rekening)
   ON DELETE SET NULL ON UPDATE CASCADE,
   Jenis_transaksi VARCHAR(20) DEFAULT „debit“ NOT NULL,
   Tanggal DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
   Jumlah INTEGER NOT NULL CHECK(jumlah>=20000));
10. Membuat table nasabah_has_rekening dengan script berikut ini.
    CREATE TABLE nasabah_has_rekening(
    Id_nasabahFK INTEGER REFERENCES nasabah(id_nasabah)
    ON DELETE CASHCADE ON UPDATE CASCADE,
    No_rekeningFK INTEGER REFERENCES rekening(no_rekening)
    ON DELETE CASHCADE ON UPDATE CASCADE,
    PRIMARY KEY(id_nasabahFK, no_rekeningFK));

```
MariaDB [perbangkan]> create table nasabah (
    -> id_nasabah INTEGER PRIMARY KEY,
    -> nama_nasabah VARCHAR(45) NOT NULL,
    -> alamat_nasabah VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.438 sec)

MariaDB [perbangkan]> create table cabang_bank (
    -> kode_cabang VARCHAR(20) PRIMARY KEY,
    -> nama_cabang VARCHAR(45) UNIQUE NOT NULL,
    -> alamat_cabang VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.320 sec)

MariaDB [perbangkan]> create table rekening (
    -> no_rekening INTEGER PRIMARY KEY,
    -> kode_cabangFK VARCHAR(20) REFERENCES cabang_bank(kode_cabang)
    -> ON DELETE CASCADE ON UPDATE CASCADE,
    -> pin VARCHAR(20) DEFAULT '1234' NOT NULL,
    -> saldo INTEGER DEFAULT 0 NOT NULL
    -> );
MariaDB [perbangkan]> create table transaksi (
    -> no_transaksi SERIAL PRIMARY KEY,
    -> id_nasabahFK INTEGER REFERENCES nasabah(id_nasabah)
    -> ON DELETE CASCADE ON UPDATE CASCADE,
    -> no_rekeningFK INTEGER REFERENCES rekening(no_rekening)
    -> ON DELETE CASCADE ON UPDATE CASCADE,
    -> jenis_transaksi VARCHAR(20) DEFAULT 'debit' NOT NULL,
    -> tanggal DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
    -> jumlah INTEGER NOT NULL CHECK (jumlah>20000)
    -> );
Query OK, 0 rows affected (0.560 sec)

MariaDB [perbangkan]> create table nasabah_has_rekening(
    -> id_nasabahFK INTEGER REFERENCES nasabah(id_nasabah)
    -> ON DELETE CASCADE ON UPDATE CASCADE,
    -> no_rekeningFK INTEGER REFERENCES rekening(no_rekening)
    -> ON DELETE CASCADE ON UPDATE CASCADE,
    -> PRIMARY KEY(id_nasabahFK, no_rekeningFK)
    -> );
Query OK, 0 rows affected (0.432 sec)
```

11. Untuk mengecek hasil pembuatan database gunakan perintah show tables;

```
MariaDB [perbangkan]> show tables;
+----------------------+
| Tables_in_perbangkan |
+----------------------+
| cabang_bank          |
| nasabah              |
| nasabah_has_rekening |
| rekening             |
| transaksi            |
+----------------------+
5 rows in set (0.137 sec)
```

12. Kemudian untuk melihat struktur tiap table dapat dilakukan dengan perintah scribe. Misalkan untuk melihat struktur table nasabah dapat dilakukan dengan perintah describe nasabah;

```
MariaDB [perbangkan]> describe nasabah;
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| id_nasabah     | int(11)      | NO   | PRI | NULL    |       |
| nama_nasabah   | varchar(45)  | NO   |     | NULL    |       |
| alamat_nasabah | varchar(255) | NO   |     | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
3 rows in set (0.257 sec)
```

Tugas

Implementasikan hasil rancangan database yang menangani data kuliah pada tugas modul 2 ke dalam program mysql.

```
MariaDB [(none)]> create database mahasiswa;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> use mahasiswa
Database changed
MariaDB [mahasiswa]> create table mahasiswa;
ERROR 1113 (42000): A table must have at least 1 column
MariaDB [mahasiswa]> create table mahasiswa(
    -> nim varchar(15) primary key,
    -> nama_mhs varchar(75) not null,
    -> alamat_mhs varchar(200) not null,
    -> tgllahir_mhs varchar(30) not null
    -> );
Query OK, 0 rows affected (0.11 sec)

MariaDB [mahasiswa]> describe mahasiswa;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| nim         | varchar(15)  | NO   | PRI | NULL    |       |
| nama_mhs    | varchar(75)  | NO   |     | NULL    |       |
| alamat_mhs  | varchar(200) | NO   |     | NULL    |       |
| tgllahir_mhs| varchar(30)  | NO   |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
4 rows in set (0.01 sec)

MariaDB [mahasiswa]> create table dosen(
    -> nip_dosen varchar(15) primary key,
    -> nama_dosen varchar(75) not null,
    -> alamat_dosen varchar(200) not null,
    -> kontak_dosen varchar(12) not null
    -> );
Query OK, 0 rows affected (0.12 sec)

MariaDB [mahasiswa]> describe dosen;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| nip_dosen   | varchar(15)  | NO   | PRI | NULL    |       |
| nama_dosen  | varchar(75)  | NO   |     | NULL    |       |
| alamat_dosen| varchar(200) | NO   |     | NULL    |       |
| kontak_dosen| varchar(12)  | NO   |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
```