JOBSHEET

PRAKTIKUM BASIS DATA

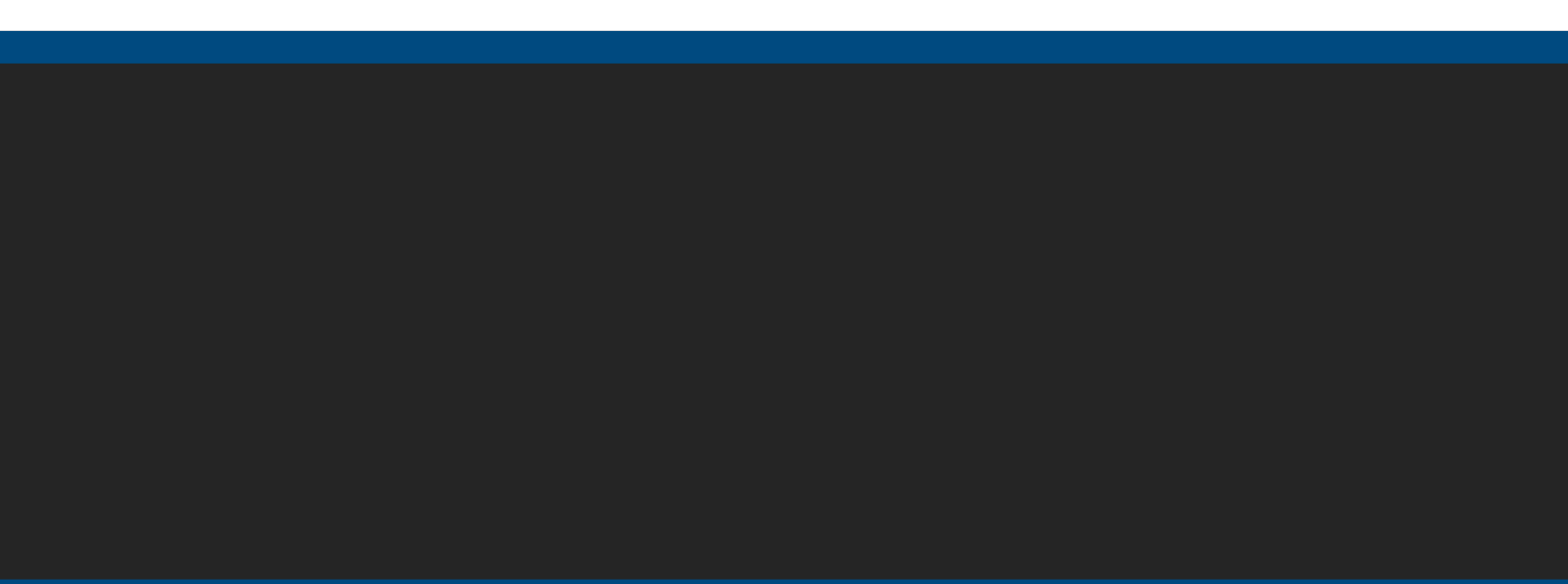
**Jurusan Teknologi Informasi**

**POLITEKNIK NEGERI MALANG**

**PERTEMUAN 10**

**Dasar MySQL**

**Team Teaching:**



Dwi Puspitasari, S.Kom., M.Kom.

Yan Watequlis Syaifudin, ST., MMT., PhD. Annisa Puspa Kirana, S. Kom, M.Kom Yoppy Yunhasnawa, S.ST., M.Sc.

Candra Bella Vista, S.Kom., MT

Elok Nur Hamdana, S.T., M.T Annisa Taufika Firdausi, ST., MT

Muhammad Shulhan Khairy, S.Kom, M.Kom Farid Angga Pribadi, S.Kom.,M.Kom

Retno Damayanti, S.Pd., M.T. **1**



# Topik

Jurusan Teknologi Informasi – Politeknik Negeri Malang

**Jobsheet-11: Dasar MySQL Mata Kuliah : Basis Data**

**Pengampu : Tim Ajar Basis Data**

**Nama : Fitri Cahyaniati(11)**

**April 2024**

Dasar MySQL dan *Data Definition Language (DDL)*

# Tujuan

Mahasiswa diharapkan dapat:

1. Memahami dasar-dasar MySQL
2. Membuat database dan tabel dengan menerapkan *data definition language (DDL)*
3. Memahami penggunaan perintah-perintah untuk mendefinisikan atribut-atribut database, tabel,

*field*, maupun batasan-batasan terhadap suatu atribut dan hubungan antar tabel

# Pendahuluan

**Basis Data :** Himpunan kelompok data (arsip) yang saling berhubungan, yang diorganisasi sedemikan rupa, sehingga kelak dapat dimanfaatkan kembali dengan cepat.

## Bahasa Basis Data

DBMS merupakan perantara antara user dengan database. Cara komunikasi diatur dalam suatu bahasa khusus yang ditetapkan oleh DBMS. Misalnya SQL, dBase, QUEL, dsb.

Ada beberapa bentuk bahasa SQL, namun yang biasa digunakan adalah 2 bentuk bahasa SQL yaitu DDL dan DML.

» ***Data Definition Language (DDL)***, digunakan dalam membuat tabel baru, indeks, mengubah tabel, menentukan struktur tabel, dsb

» ***Data Manipulation Language (DML)***, digunakan dalam memanipulasi dan pengambilan data pada database. Manipulasi data dapat mencakup :

* Pemanggilan data yang tersimpan dalam database (query)
* Penyisipan/penambahan data baru ke database
* Penghapusan data dari database
* Pengubahan data pada database

## Dasar-Dasar MySQL

Dalam bahasa SQL pada umumnya informasi tersimpan dalam tabel-tabel yang secara logik merupakan struktur dua dimensi terdiri dari baris (row atau record) dan kolom (column atau *field*). Sedangkan dalam sebuah database dapat terdiri dari beberapa table. Beberapa tipe data dalam MySQL yang sering dipakai:



***Data Definition Language (DDL)***

DDL merupakan sekumpulan set perintah yang bertujuan untuk mendefinisikan atribut-atribut database, tabel, atribut kolom (*field*), maupun batasan-batsan terhadap suatu atribut dan relasi/ hubungan antar tabel. Yang termasuk dalam kelompok perintah DDL adalah :

***CREATE***

*  merupakan perintah DDL yang digunakan untuk membuat database maupun tabel. Nama database maupun tabel tidak boleh mengandung spasi (space). Nama database tidak boleh sama antar database
* ***ALTER***  merupakan perintah DDL yang digunakan untuk mengubah nama/struktur tabel
* ***DROP***  merupakan perintah DDL yang digunakan untuk menghapus database ataupun tabel

Penerapan DDL pada :

### DATABASE

#### CREATE DATABASE nama\_database;

Merupakan perintah untuk membuat database baru, contoh:

### CREATE DATABASE db\_polinema;

#### SHOW DATABASES;

Merupakan perintah untuk menampilkan daftar nama database yang terdapat dalam database server.

#### USE nama\_database;

Merupakan perintah untuk masuk kedalam database yang akan digunakan. Sebelum membuat tabel yang digunakan untuk menyimpan data, terlebih dahulu harus memilih/mengaktifkan satu database sebagai database aktif. Contoh untuk dapat menggunakan/memanipulasi database *db\_polinema*, maka gunakan perintah berikut:

### USE db\_polinema;

#### DROP DATABASE namadatabase;

Merupakan perintah yang berfungsi untuk menghapus database. Contoh kita ingin menghapus database db\_polinema, maka perintahnya adalah:

### DROP DATABASE db\_polinema;

1. **TABEL**
   * Membuat Tabel

Nama tabel tidak boleh mengandung spasi (*space*). Ketika membuat tabel, ada beberapa yang harus dideklarasikan dalam pembuatannya yaitu antara lain meliputi : nama tabel, nama kolom (*field*), tipe data dari *field* dan panjang data. Perintah yang digunakan untuk membuat tabel secara umum adalah sebagai berikut :

### CREATE TABLE namatabel (field1 typedata1, field2 typedata2);

Contoh berikut ini adalah syntax untuk membuat tabel mahasiswa :

### CREATE TABLE mahasiswa (nim VARCHAR(20), nama\_mhs VARCHAR(50), umur INT);

* + Menampilkan tabel

Untuk menampilkan daftar nama-nama tabel yang terdapat dalam database yang sedang aktif/digunakan, menggunakan perintah :

### SHOW TABLES;

* + Menampilkan deskripsi atribut tabel

Untuk menampilkan deskripsi atribut-atribut yang terdapat pada suatu tabel dengan menggunakan perintah :

Contoh :

### DESC nama\_tabel; DESC mahasiswa;

* + Menghapus Tabel

Untuk menghapus Tabel perintahnya sama dengan untuk menghapus database yaitu dengan menggunakan perintah DROP. Perintah yang digunakan adalah :

### DROP TABLE namatabel;

Misalnya yang akan dihapus adalah tabel mahasiswa :

### DROP TABLE mahasiswa;

* + Mengisi Data / Input Data ke dalam tabel

Untuk memasukkan sebuah baris (*tuple*/*record*) kedalam tabel MySQL adalah sebagai berikut :

### INSERT INTO table [(column1, column2,...) VALUES (value1, value2,...)];

Contoh :

### INSERT INTO mahasiswa (nim, nama, umur) values (‘001’, ’Imam’, 20);

* + Mendefinisikan ***Null/Not Null***

*Null* ataupun *Not Null* merupakan pernyataan yang digunakan untuk membuat kolom yang akan dibuat boleh kosong (*Null*) atau tidak boleh kosong (*Not Null*). Ketika pada kolom tabel tidak di set, maka secara default akan bernilai *Null* (boleh kosong). Untuk mendefinisikannya maka perintah yang akan digunakan adalah :

### CREATE TABLE mahasiswa (nim VARCHAR(20) NOT NULL, nama\_mhs VARCHAR(50) NOT NULL, umur INT);

* + Mendefinisikan Nilai Default

Nilai default merupakan nilai yang diberikan secara otomatis oleh sistem untuk suatu kolom ketika terjadi penambahan baris baru, sementara nilai pada kolom tersebut tidak diisi oleh pengguna.

Contoh :

### CREATE TABLE mahasiswa (nim VARCHAR(20) NOT NULL, nama\_mhs VARCHAR(50) NOT NULL, umur INT DEFAULT 0);

* + Mendefinisikan PRIMARY KEY pada Tabel

Suatu keharusan dalam suatu tabel adalah harus memiliki satu kolom yang dijadikan sebagai perwakilan dari tabel tersebut. Pembuatan perwakilan tabel ini berfungsi untuk melakukan hubungan / relasional dengan tabel lain. Bentuk perwakilan ini dalam database disebut sebagai **PRIMARY KEY** yang aturan pembuatannya adalah sebagai berikut :

* + - Satu tabel bisa memiliki 2 **primary key**
    - Nama kolom kunci tidak digunakan pada kolom lain satu tabel
    - Nama kolom kunci tidak boleh sama dengan kolom kunci yang ada pada tabel lain
    - Bentuk kolom kunci harus diset **NOT NULL**

Terdapat tiga cara untuk mendefinisikan *primary key*. Berikut ini perintah yang digunakan :

1. **Primary key** dibuat saat **create table**, dan didefinisikan di akhir

### CREATE TABLE mahasiswa (nim VARCHAR(20), nama\_mhs VARCHAR(50), umur INT, PRIMARY KEY (nim) );

1. **Primary key** dibuat saat **create table**, dan didefinisikan bersamaan dengan kolom/*field*

### CREATE TABLE mahasiswa (nim VARCHAR(20) NOT NULL PRIMARY KEY,

**nama\_mhs VARCHAR(50), umur INT);**

1. **Primary key** dibuat menggunakan **alter table**

### ALTER TABLE mahasiswa ADD CONSTRAINT namaconstraint PRIMARY KEY(namakolom);

**ALTER TABLE mahasiswa ADD CONSTRAINT PK\_Mahasiswa PRIMARY KEY(nim);**

* + Menghapus **PRIMARY KEY** pada tabel

» Cara 1: Jika **primary key** dibuat menggunakan **create table**: **ALTER TABLE nama\_table DROP PRIMARY KEY;**

Contoh

### ALTER TABLE mahasiswa DROP PRIMARY KEY;

» Cara 2: Jika **primary key** dibuat melalui **alter table**:

### ALTER TABLE namatabel DROP CONSRTRAINT namaconstraint;

Contoh:

### ALTER TABLE mahasiswa DROP CONSTRAINT PK\_Mahasiswa;

* + Menambah kolom baru pada tabel

Untuk menambah kolom baru pada suatu tabel bisa menggunakan perintah berikut :

### ALTER TABLE nama\_tabel ADD field\_baru typedata(lebar);

**nama\_tabel** merupakan nama tabel yang akan ditambahkan kolomnya. **field\_baru** merupakan nama kolom yang akan ditambahkan, **typedata(lebar)** merupakan type data dan lebar data yang akan ditambahkan. Contohnya menambahkan kolom telepon pada tabel mahasiswa setelah kolom umur:

### ALTER TABLE mahasiswa ADD COLUMN telepon VARCHAR(15) AFTER umur;

* + Mengubah Tipe Data atau Lebar Kolom pada Tabel Perintah yang digunakan adalah :

### ALTER TABLE nama\_tabel MODIFY COLUMN field type(lebar);

Contoh :

### ALTER TABLE mahasiswa MODIFY COLUMN telepon VARCHAR(12);

* + Mengubah Nama Kolom(*Field*) Perintah yang digunakan adalah :

### ALTER TABLE nama\_tabel CHANGE COLUMN nama\_kolom\_lama nama\_kolom\_baru typedatabaru(lebarbaru);

Contoh :

### ALTER TABLE mahasiswa CHANGE COLUMN telepon phone VARCHAR(25);

* + Menghapus Kolom pada Tabel Perintah yang digunakan adalah :

Contoh :

### ALTER TABLE nama\_tabel DROP COLUMN nama\_kolom; ALTER TABLE mahasiswa DROP COLUMN phone;

* + Mendefinisikan *Foreign Key*

*Foreign Key* adalah salah satu jenis *constraint* yang digunakan untuk merelasikan antar dua tabel atau lebih. *Foreign Key* digunakan pada tabel kedua (detail) yang mereferensi ke tabel utama yang mempunyai *constraint primary key*.



Pada gambar tersebut kolom **Cust\_ID** pada tabel **Orders** berperan sebagai *foreign key*

yang mereferensi ke tabel **Customers** pada kolom **ID**.

**ALTER TABLE nama\_tabel ADD FOREIGN KEY nama\_kolom REFERENCES nama\_tabel\_referensi (nama\_kolom\_referensi);**

|  |  |
| --- | --- |
| ***Field*** | **Type Data** |
| kode\_prodi | VARCHAR (6) PRIMARY KEY |
| nama\_prodi | VARCHAR (30) |

# PRAKTIKUM

|  |  |
| --- | --- |
| **No** | **Keterangan** |
| **1** | Buka prompt jalankan perintah berikut ini :  ***C:\>Program Files\xampp\mysql\bin>mysql –u root –p*** (enter) |
|  |
| **2** | Buatlah sebuah database dengan nama ***db\_polinema*** |
|  |
|  | Sebelum menlanjutkan langkah No. 3, jalankan perintah “use db\_polinema” |
| **3** | a. Tabel ***prodi*** |
|  |

|  |  |
| --- | --- |
| ***Field*** | **Type Data** |
| nim | INT (8) PRIMARY KEY |
| nama\_mhs | VARCHAR (50) |
| jenis\_kelamin | ENUM (‘L’,’P’) DEFAULT ‘L’ |
| alamat | VARCHAR (50) |
| kota | VARCHAR (20) DEFAULT ‘MALANG’ |
| asal\_sma | VARCHAR (30) |
| no\_hp | VARCHAR (12) |
| umur | INT |
| kode\_prodi | VARCHAR (6) *FOREIGN KEY* fk0 (kode\_prodi) REFERENSCES prodi (kode\_prodi) |

|  |  |
| --- | --- |
| ***Field*** | **Type Data** |
| mk\_id | VARCHAR (10) PRIMARY KEY |
| nama\_mk | VARCHAR (50) |
| jumlah\_jam | FLOAT (4,2) |
| sks | INTEGER |

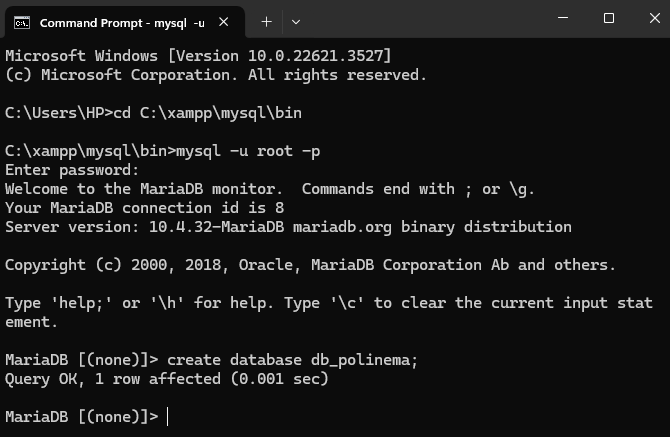
|  |  |
| --- | --- |
| **4** | Buatlah beberapa tabel dalam database tersebut sesuai dengan kriteria berikut :  b. Tabel ***mahasiswa*** |
| Untuk DBMS MySQL    Untuk DBMS MariaDB |
| **5** | c. Tabel ***mata\_kuliah*** |
|  |

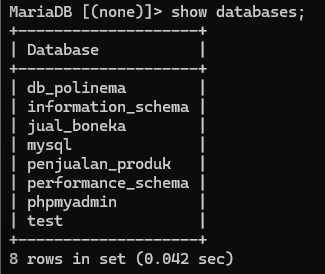
|  |  |
| --- | --- |
| ***Field*** | **Type Data** |
| ruang\_id | VARCHAR (3) PRIMARY KEY |
| nama\_ruang | VARCHAR (20) |
| Kapasitas | INTEGER |

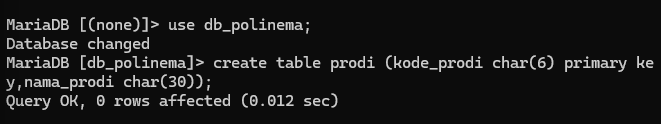
|  |  |
| --- | --- |
| ***Field*** | **Type Data** |
| nidn | INTEGER (20) PRIMARY KEY |
| nama\_dosen | VARCHAR (50) |
| status | ENUM (‘PNS’,’KONTRAK’) DEFAULT ‘PNS’ |
| jenis\_kelamin | ENUM (‘L’,’P’) DEFAULT ‘L’ |
| no\_hp | VARCHAR (15) |

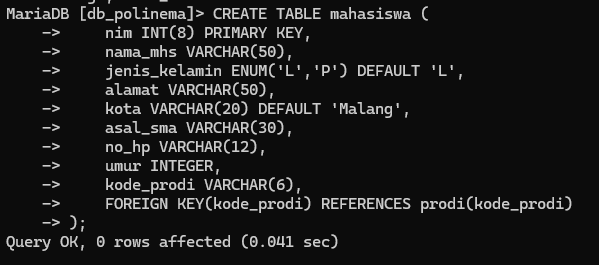
|  |  |
| --- | --- |
| **6** | d. Tabel ***ruang*** |
|  |
| **7** | e. Tabel ***dosen*** |
|  |
| **8** | **<Soal>**  Tambahkan sebuah kolom ***agama (varchar(10))*** pada tabel mahasiswa sebagai kolom terakhir Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **9** | **<Soal>**  Tambahkan kolom ***alamat(varchar(50))*** pada tabel dosen sebagai kolom terakhir Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **10** | **<Soal>**  Lakukan insert data ke dalam tabel-tabel yang ada pada pada database ***db\_polinema*** sesuai dengan *field*, tipe data dan panjang datanya  Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **11** | **<Soal>**  Tampilkan semua tabel yang ada didalam database ***db\_polinema***  Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **12** | **<Soal>**  Tampilkan semua isi tabel yang ada didalam tabel mahasiswa Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **13** | **<Soal>**  Tampilkan struktur(metadata) tabel mahasiswa |

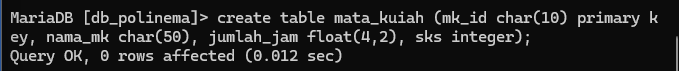
|  |  |
| --- | --- |
|  | Catat : Buat Screenshot dari perintah yang anda ketikkan |
| **14** | **<Soal>**  hilangkan kolom asal\_sma yang terdapat didalam tabel mahasiswa Catat : Buat Screenshot dari perintah yang anda ketikkan |

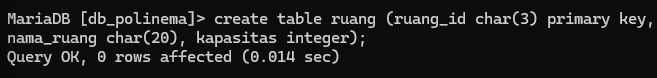


****

****

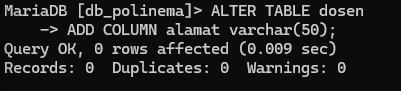
****

****

****

****

1. Tambahkan sebuah kolom ***agama (varchar(10))*** pada tabel mahasiswa sebagai kolom terakhir Catat : Buat Screenshot dari perintah yang anda ketikkan

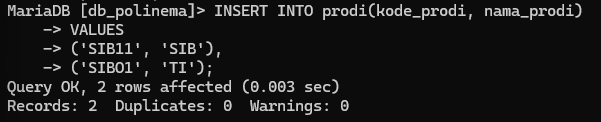
****

1. Tambahkan kolom ***alamat(varchar(50))*** pada tabel dosen sebagai kolom terakhir Catat : Buat Screenshot dari perintah yang anda ketikkan

****

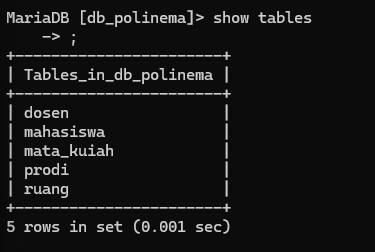


Lakukan insert data ke dalam tabel-tabel yang ada pada pada database ***db\_polinema*** sesuai dengan *field*, tipe data dan panjang datanya

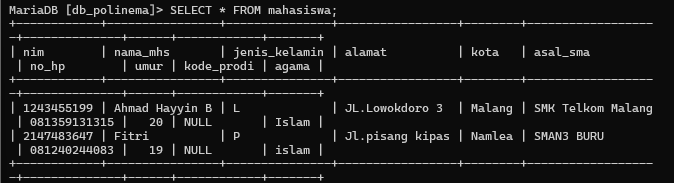
****

****

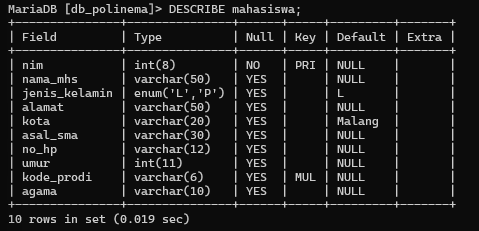
1. Tampilkan semua isi tabel yang ada didalam tabel mahasiswa Catat : Buat Screenshot dari perintah yang anda ketikkan

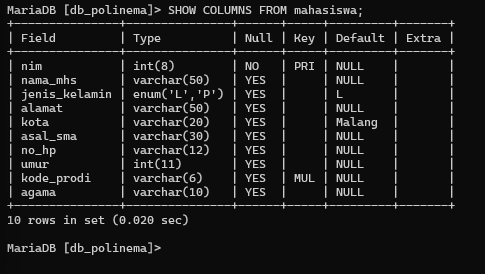
****

1. Tampilkan semua isi tabel yang ada didalam tabel mahasiswa Catat : Buat Screenshot dari perintah yang anda ketikkan

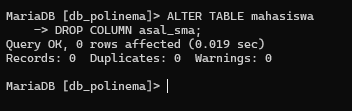
****

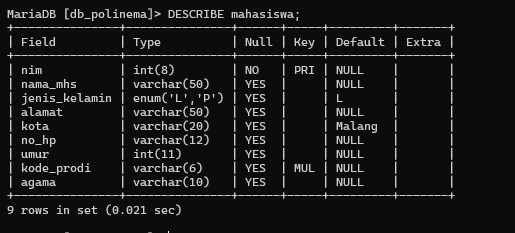
1. Tampilkan struktur(metadata) tabel mahasiswa

****

****

1. hilangkan kolom asal\_sma yang terdapat didalam tabel mahasiswa Catat : Buat Screenshot dari perintah yang anda ketikkan

****

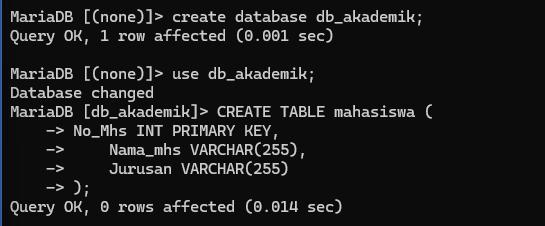
****

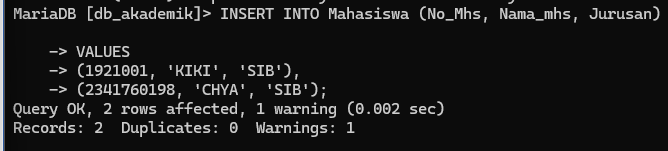
**TUGAS**

### Buatlah basis data Akademik dengan data sebagai berikut :

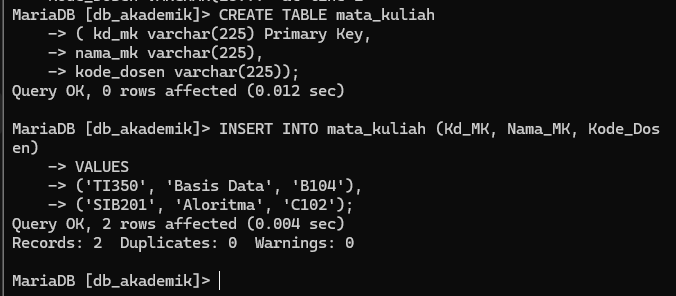
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No\_Mhs | Nama\_mhs | Jurusan | Kd\_MK | Nama\_mk | Kd\_Dosen | Nm\_Dosen | nilai |
| 1921001 | Aminah | MI | MI350 | Basis Data | B104 | Ati | 85 |
| 1921001 | Budiman | MI | MI465 | Pemrograman | B105 | Dita | 87 |
| 1921002 | Carina | MI | MI465 | Pemrograman | B105 | Dita | 85 |
| 1921003 | Della | TI | TI201 | Mobile | C102 | Leo | 78 |
| 1921004 | Firda | TI | TI201 | Mobile | C102 | Leo | 80 |

* 1. deskripsikan struktur data dari table-tabel berikut serta isikan datanya:
* Tabel Mahasiswa {No\_Mhs, Nama\_mhs}



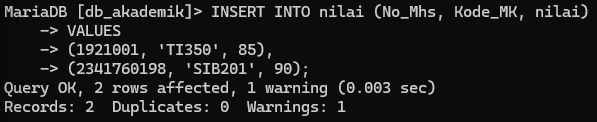


* Tabel Mata\_Kuliah {Kd\_MK, Nama\_MK}

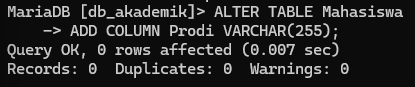


* Tabel nilai {No\_Mhs, Kode\_MK}

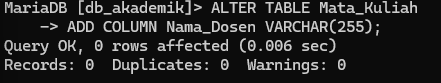




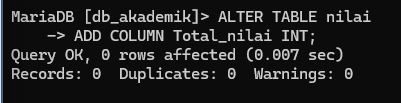
* **tambahkan kolom Jurusan pada tabel Mahasiswa di kolom terakhir**



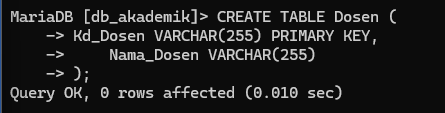
* 1. tambahkan kolom Kode Dosen pada tabel Mata\_Kuliah

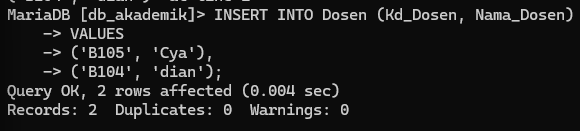


* 1. tambahkan kolom nilai pada tabel nilai serta berikanlah kunci *foreign key*

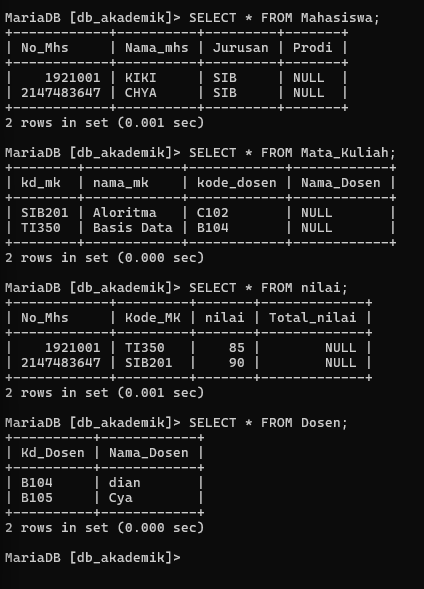
**

* 1. tambahkan Tabel Dosen dengan atributnya Kd\_Dosen dan Nama Dosen





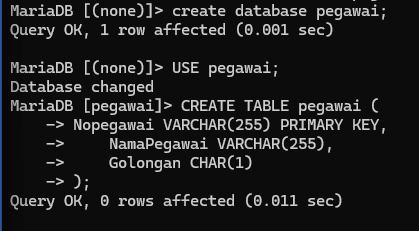
* 1. tampilkan semua data yang ada pada tiap tabel

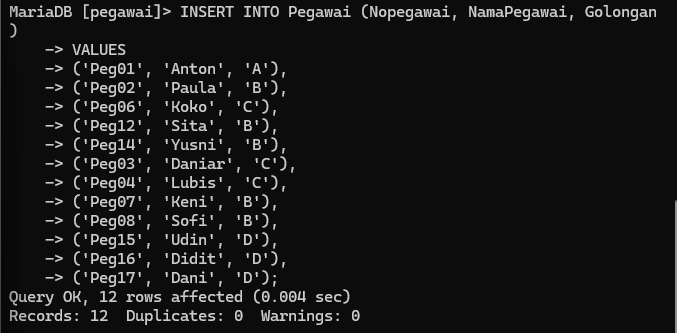


### Buatlah basis data Pegawai yang terdiri dari tabel sebagai berikut :

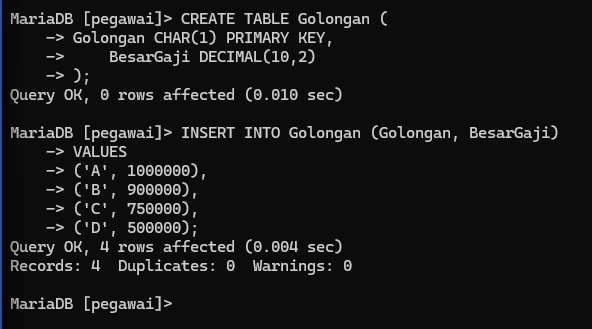
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Noproyek | NamaProyek | Nopegawai | NamaPegawai | Golongan | BesarGaji |
| NP001 | BRR | Peg01 | Anton | A | 1.000.000 |
| NP001 | BRR | Peg02 | Paula | B | 900.000 |
| NP001 | BRR | Peg06 | Koko | C | 750.000 |
| NP002 | PEMDA | Peg01 | Anton | A | 1.000.000 |
| NP002 | PEMDA | Peg12 | Sita | B | 900.000 |
| NP002 | PEMDA | Peg14 | Yusni | B | 900.000 |
| NP003 | CBR | Peg02 | Paula | B | 900.000 |
| NP003 | CBR | Peg03 | Daniar | C | 750.000 |
| NP003 | CBR | Peg04 | Lubis | C | 750.000 |
| NP004 | ASK | Peg07 | Keni | B | 900.000 |
| NP004 | ASK | Peg08 | Sofi | B | 900.000 |
| NP004 | ASK | Peg06 | Yuni | C | 750.000 |
| NP005 | OB | Peg15 | Udin | D | 500.000 |
| NP005 | OB | Peg16 | Didit | D | 500.000 |
| NP005 | OB | Peg17 | Dani | D | 500.000 |

* 1. Deskripsikan struktur data dari table-tabel berikut serta isikan datanya:
* Table Pegawai {Nopegawai, NamaPegawai}

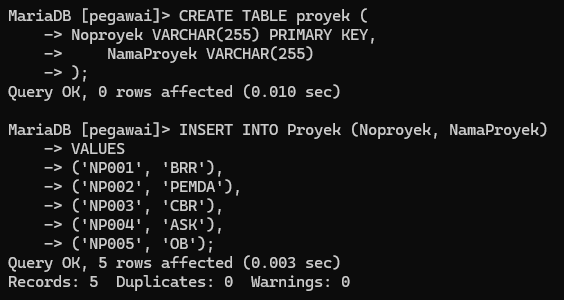




* Tabel Golongan {Golongan}



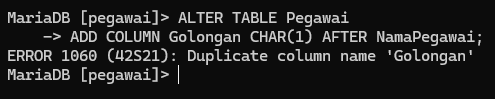
* Tabel Proyek {Noproyek]



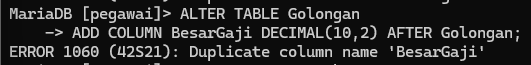
* Tabel Proyekpegawai {Noproyek}



* 1. Tambahkan kolom Golongan pada tabel Pegawai di kolom terakhir



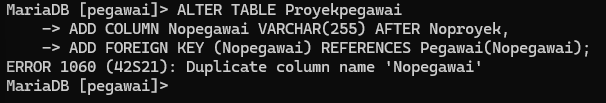
* 1. Tambahkan kolom Besar Gaji pada tabel Golongan di kolom terakhir



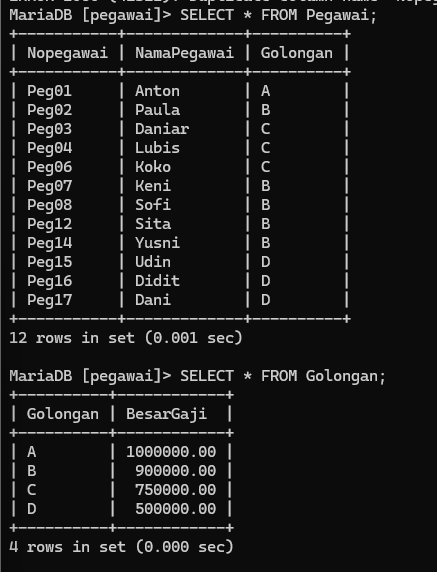
* 1. Tambahkan kolom NamaProyek pada table Proyek

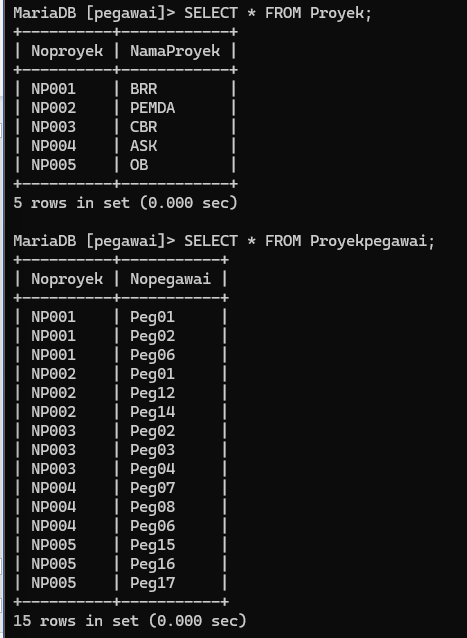


* 1. Tambahkan kolom NoPegawai pada table Proyekpegawai serta berikanlah kunci *foreign key*

**

* 1. Tampilkan semua data yang ada pada tiap tabel





#### ~~ Selamat Mengerjakan ~~