

Basis Data II

00. Pengantar Basis Data II

Teguh Pribadi
<https://github.com/pribbadi> || 082 337 475 885

Pengantar Basis Data II

- a. Pendahuluan
- b. Normalisasi Data / ERD
- c. DDL-DML
- d. DDL
- e. DML

a) Pendahuluan

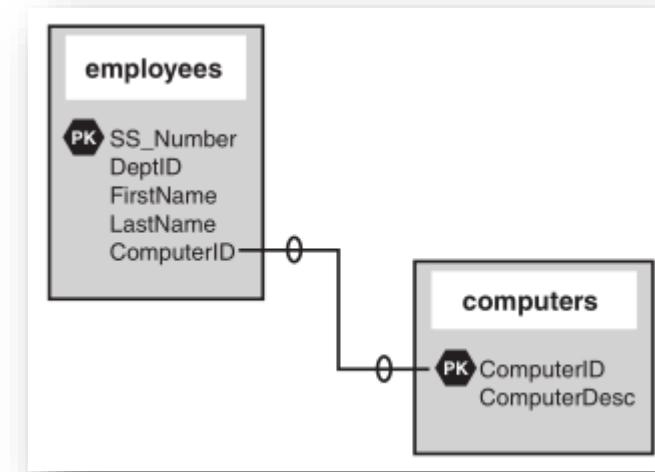
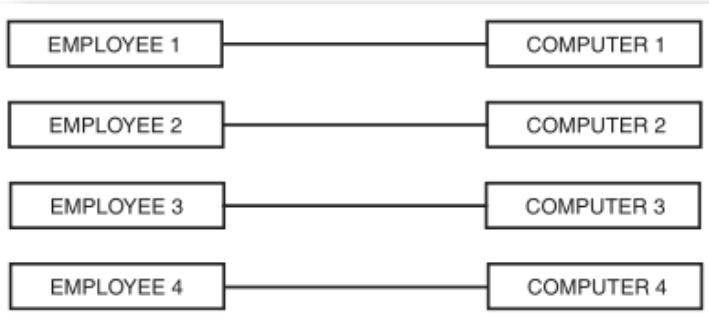
- A database is an **organized collection** of data. In MySQL you often create **separate databases** for each of your projects. (Andrea Tar, 2012)
- The purpose of a **database** is to help people keep **track of things**, and the most commonly used type of database is the relational database. Data are recorded **facts** and numbers. (Kroenke David, 2013)
- A database is a **collection of data**. The term database usually indicates that the collection of data is **stored on a computer**. (Suehering steve, 2009)
- Basis atau **kumpulan data lengkap yang terelasi**.

b) ERD

- Adalah **hubungan** antar tabel dalam database yang menciptakan keutuhan data.
- Ada 3 hubungan antar table: (Melani Julie C., 2004)
 1. One to one
 2. Many to one
 3. Many to many

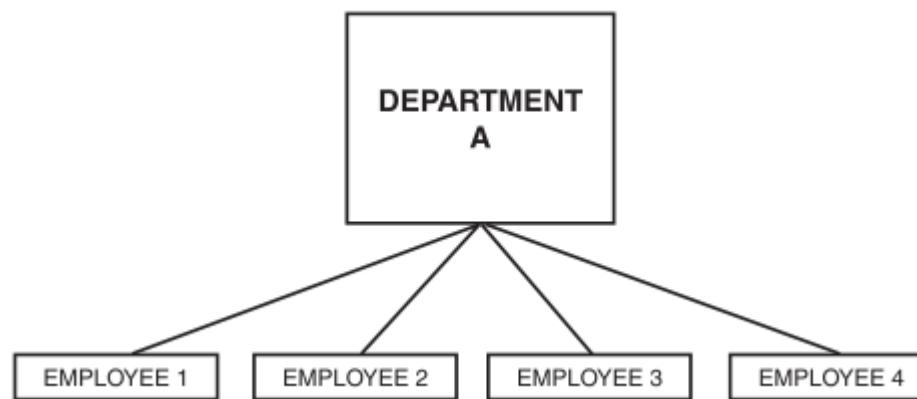
b) ERD - one to one

- ❑ A key appears only once in a related table. (Melani Julie C., 2004)



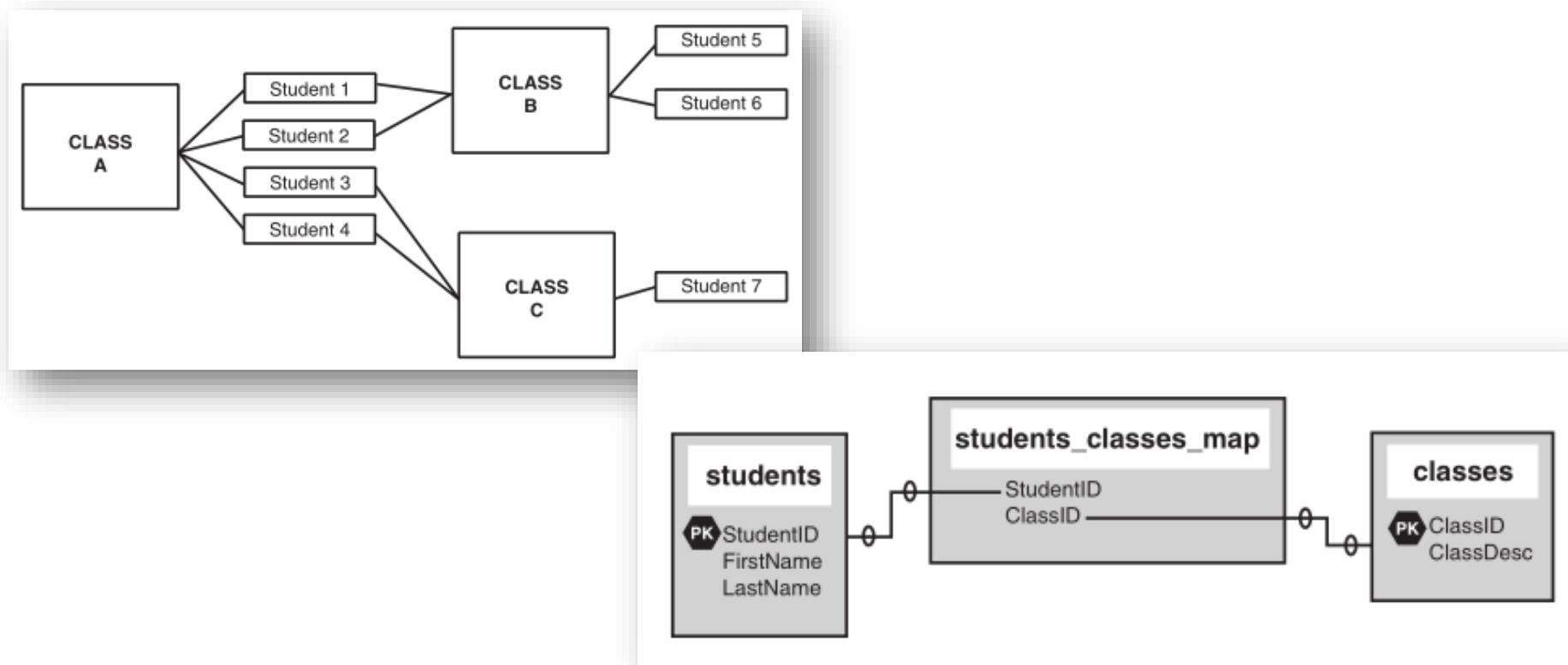
b) ERD - one to many

- ❑ Keys from one table appear multiple times in a related table. (Melani Julie C., 2004)



b) ERD - many to many

- ❑ Keys from one table appear multiple times in a related table. (Melani Julie C., 2004)



b) ERD - Mengapa dibutuhkan?

A good database design is crucial for a high performance application, just like an aerodynamic body is important to a race car. If the car doesn't have smooth lines, it will produce drag and go slower. The same holds true for databases. If a database doesn't have optimized relationships (normalization) it won't be able to perform as efficiently as possible. (Melani Julie C., 2004)

b) ERD - Mengapa dibutuhkan?

Beyond performance is the issue of maintenance. Your database should be easy to maintain. This includes storing a limited amount (if any) of repetitive data. If you have a lot of repetitive data and one instance of that data undergoes a change (such as a name change), that change has to be made for all occurrences of the data. (Melani Julie C., 2004)

b) ERD - Alur menentukan relasi tabel

□ Alur menentukan Relasi table:

1. Pahami system yang akan dibuat.
2. Berdasar point 1, ekstrak satu per satu tabelnya.
3. Relasikan tabel-tabel dengan terlebih dahulu menentukan hubungan antar tabelnya.

b) ERD - Studi Kasus 1

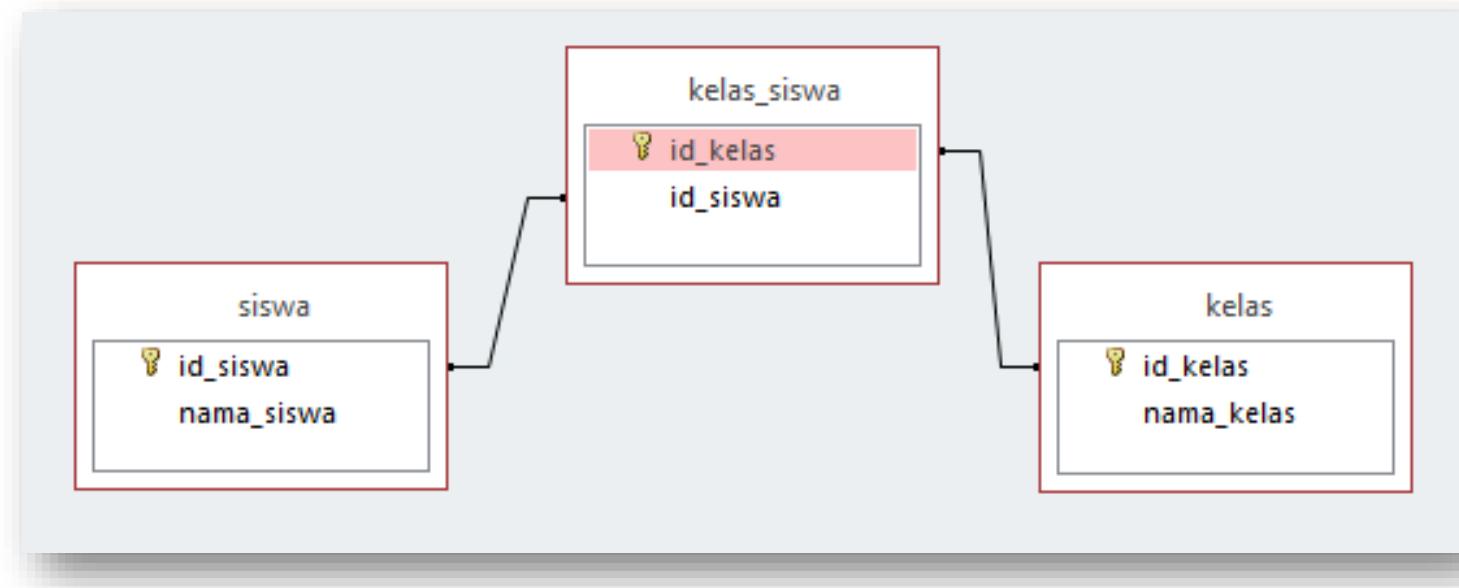
Sebuah sekolah dengan ketentuan:

1. Setiap siswa akan masuk dalam sebuah kelas berjenjang.
2. Setiap kelas akan memiliki pelajaran yang telah ditetapkan.
3. Setiap pelajaran diajar oleh seorang guru.

b) ERD - Studi Kasus 1 (proses 1)

Sebuah sekolah dengan ketentuan:

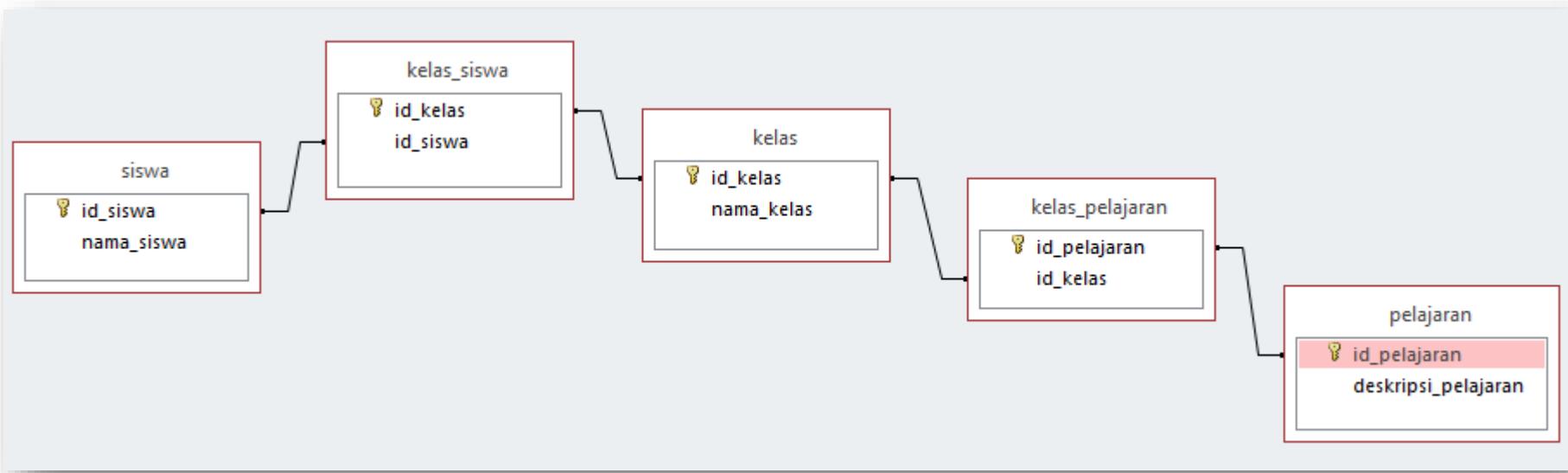
1. Setiap siswa akan masuk dalam sebuah kelas berjenjang.



b) ERD - Studi Kasus 1 (proses 2)

Sebuah sekolah dengan ketentuan:

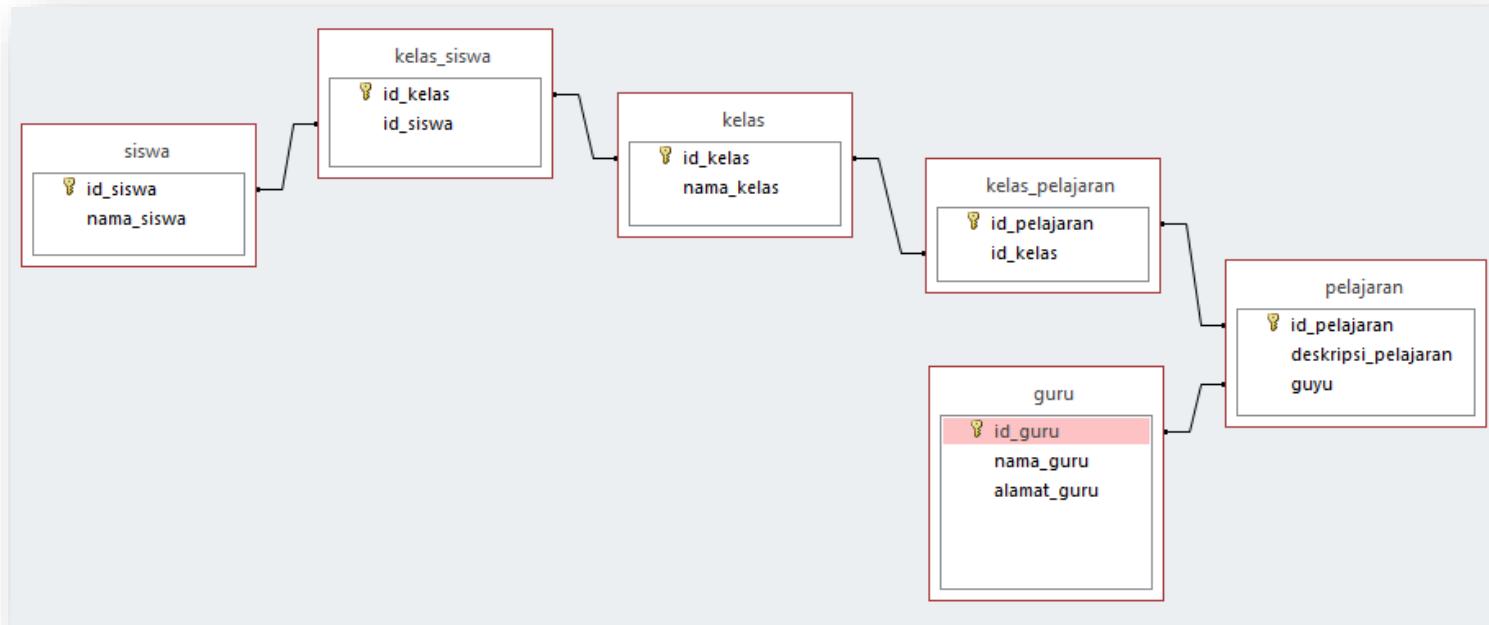
1. Setiap siswa akan masuk dalam sebuah kelas berjenjang.
2. Setiap kelas akan memiliki pelajaran yang telah ditetapkan.



b) ERD - Studi Kasus 1 (proses 3)

Sebuah sekolah dengan ketentuan:

1. Setiap siswa akan masuk dalam sebuah kelas berjenjang.
2. Setiap kelas akan memiliki pelajaran yang telah ditetapkan.
3. **Setiap pelajaran diajar oleh seorang guru.**



b) ERD - Studi Kasus 2

Perpustakaan X akan membuat sebuah aplikasi dengan ketentuan:

- Anggota perpustakaan adalah mahasiswa, dosen, dan staff X.
- Lama waktu peminjaman buku untuk setiap anggota adalah 1 minggu.
- Jumlah maksimal peminjaman buku dalam satu waktu adalah 3 buah.
- Peminjaman dilayani oleh staff perpustakaan.
- Denda akan dikenakan jika peminjam melewati waktu pengembalian.
- Gambarkan relasi tabelnya!**

b) ERD - Studi Kasus 3

Seorang dosen akan membuatkan aplikasi bagi seluruh dosen di kampus ABC untuk mengorganisir pengumpulan tugas dari kelas mahasiswanya, dimana ketentuannya adalah sbb:

- Dibutuhkan akses login untuk memasuki aplikasi tersebut, dengan menginputkan username dan password.
- Tugas harus dikumpulkan tepat waktu.
- Pengumpulan tugas dapat dalam bentuk pdf, teks, atau gambar.
- Gambarkan relasi tabelnya!

b) ERD - Studi Kasus 4

- Sebuah supermarket memiliki pelayanan prima terhadap pelanggannya, salah satu wujudnya adalah dengan menyediakan kasir minimal 10.
- Setiap kasir bertugas melayani pelanggan yang akan melakukan pembelian barang-barang.
- Dalam satu shift juga terdapat 1 orang supervisor yang akan mengawasi 5 kasir dan membantu dalam penganan khusus dari setiap transaksi yang terjadi.
- Pelayanan prima juga dilakukan dengan menerapkan 2 shift per harinya.
- Supermarket tersebut akan di SI-kan, maka gambarkan relasi tabelnya!

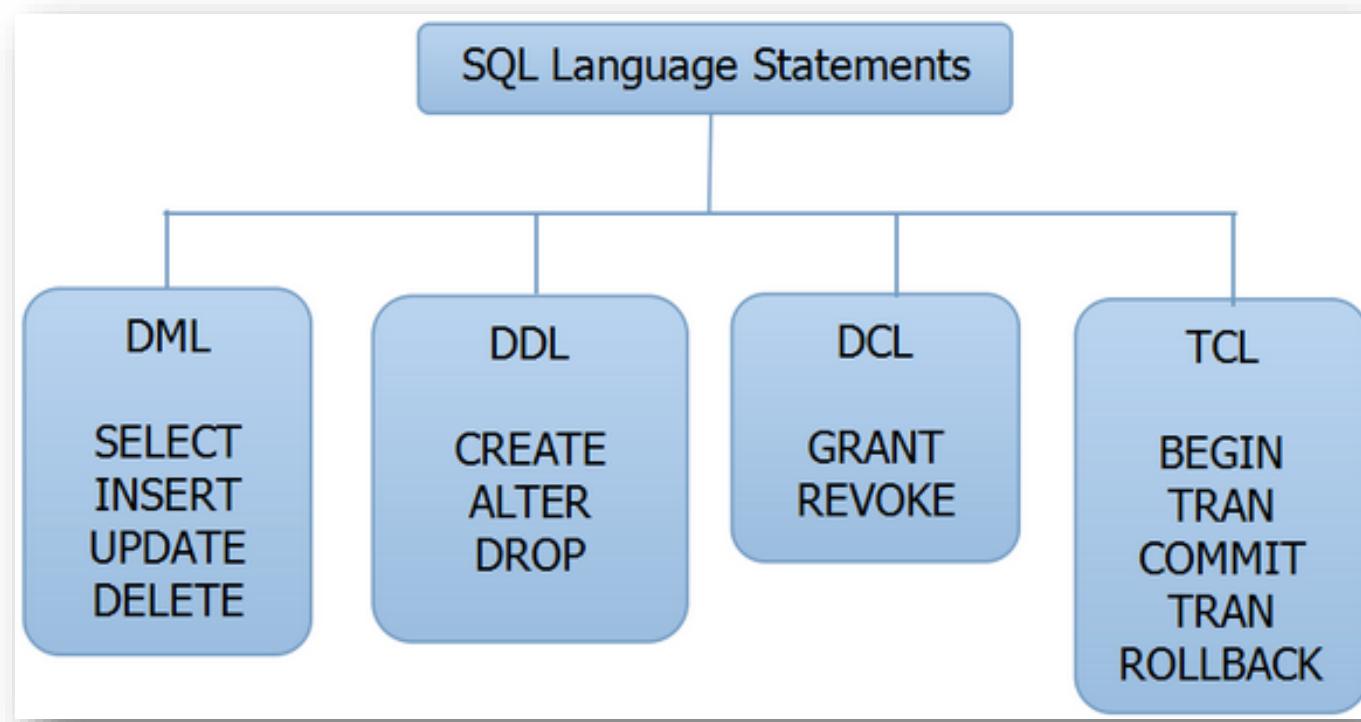
b) ERD - Studi Kasus 5

- Instansi dengan jumlah pengolahan data pegawai banyak membutuhkan bantuan SI.
- Pegawai yang didata mencakup PNS, Pensiun (meninggal, pemberhentian dengan tidak hormat, pemberhentian dengan hormat), dan Non-PNS.
- Setiap pegawai akan memiliki riwayat terkait dengan pendidikan, keluarga, pangkat, dan lokasi kerjanya.
- Seluruh kegiatan dalam sistem akan ada log-nya, sehingga superadmin dapat mengetahui kegiatan di dalam Si tersebut.
- Orang yang akan menggunakan SI dengan username dan passwordnya adalah superadmin (pengatur sistem), admin (setiap skpd), dan pegawai.

b) ERD - Studi Kasus 6

- Sekolah dengan keterbukaan dan ketertiban administrasi yang tinggi akan men-SI-kan pembayarannya.
- Setiap siswa yg membayar SPP akan dilayani oleh bendahara.
- Jumlah pembayaran siswa dalam tiap bulannya berkemungkinan berbeda antara kelas X, XI, dan XII.
- Pembayaran siswa boleh dilakukan secara cicil dalam tiap bulannya.
- Dalam pembayaran SPP terdapat beberapa rincian sub yang telah ditetapkan oleh sekolah.

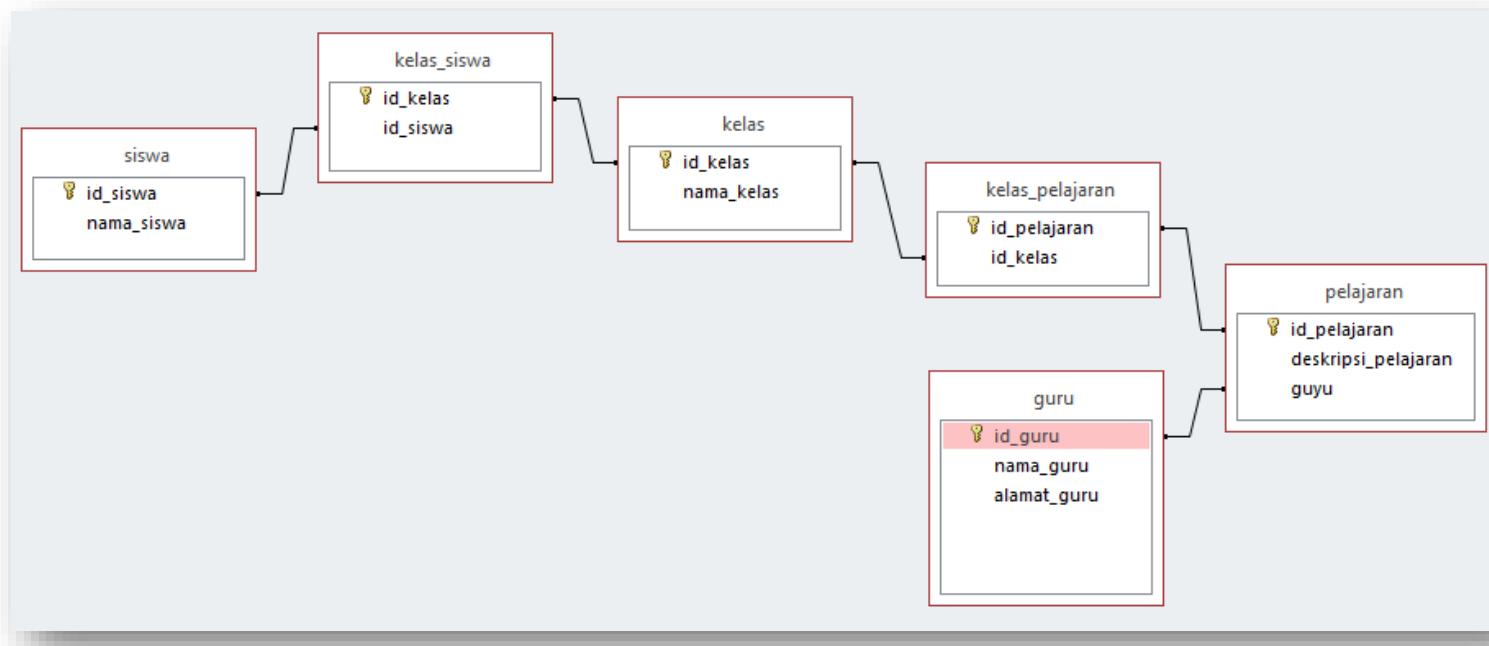
c) DDL DML



- Data Definition Language (DDL)
- Data Manipulation Language (DML)

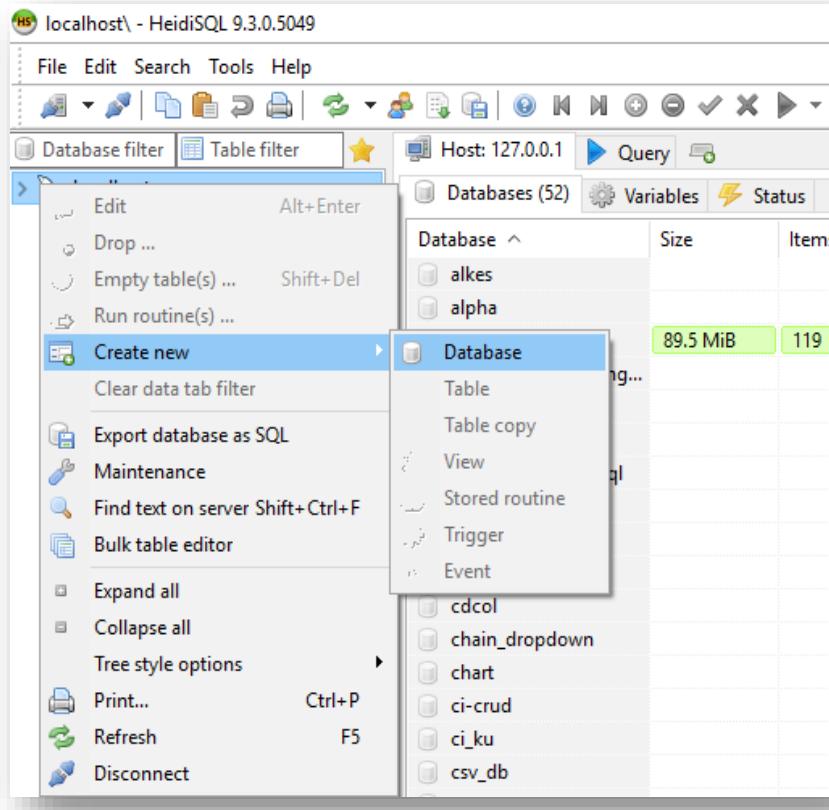
d) DDL

- ❑ Pengelolaan pembuatan database dan tabel.
- ❑ Dengan berdasar pada relasi table seperti gambar berikut, (nama database = **sekolah**)



d) DDL Script - buat database

GUI



CLI

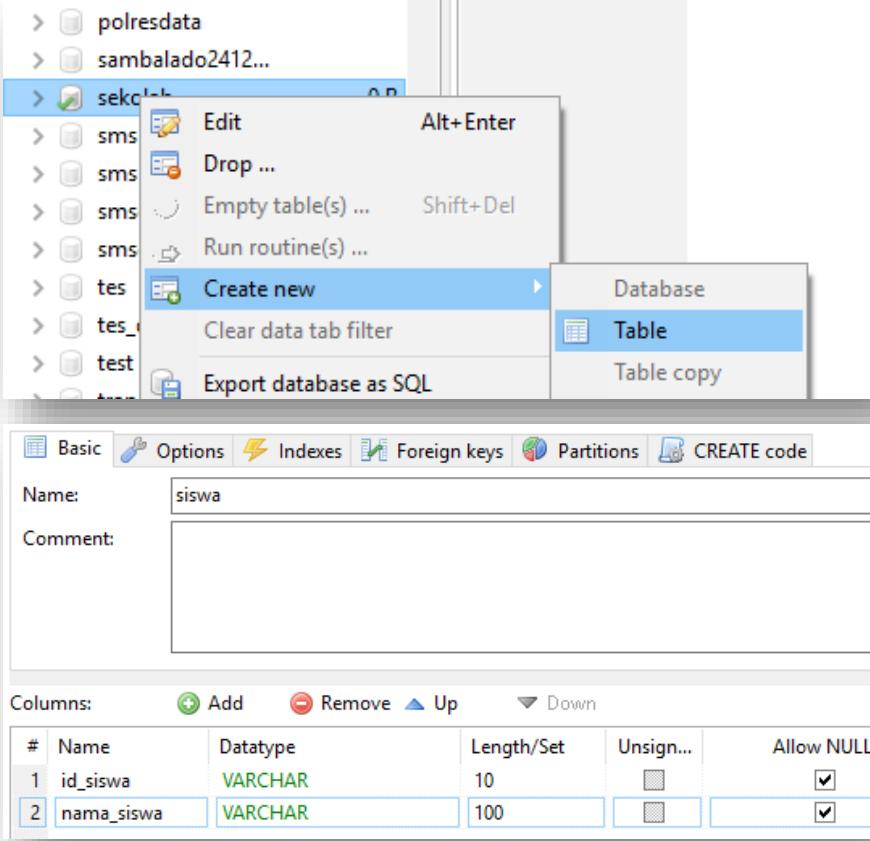
```
CREATE DATABASE  
`sekolah`;
```

```
USE `sekolah`;
```

```
SHOW TABLES;
```

d) DDL Script - buat tabel

GUI



CLI

```
CREATE TABLE `siswa` (
  `id_siswa` VARCHAR(10)
NULL,
  `nama_siswa` VARCHAR(100)
NULL
);
```

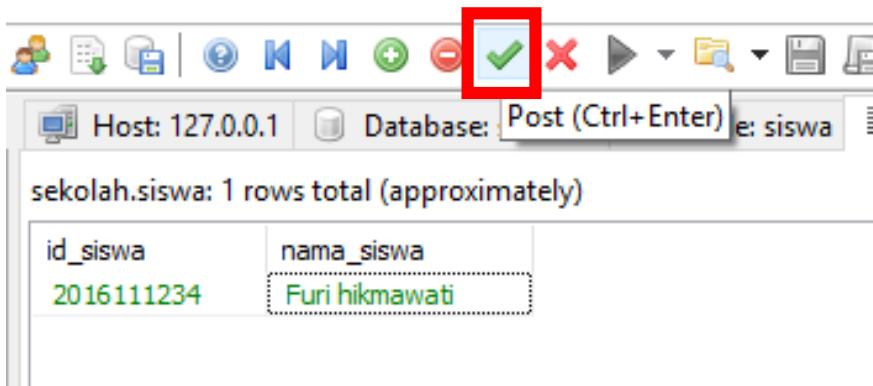
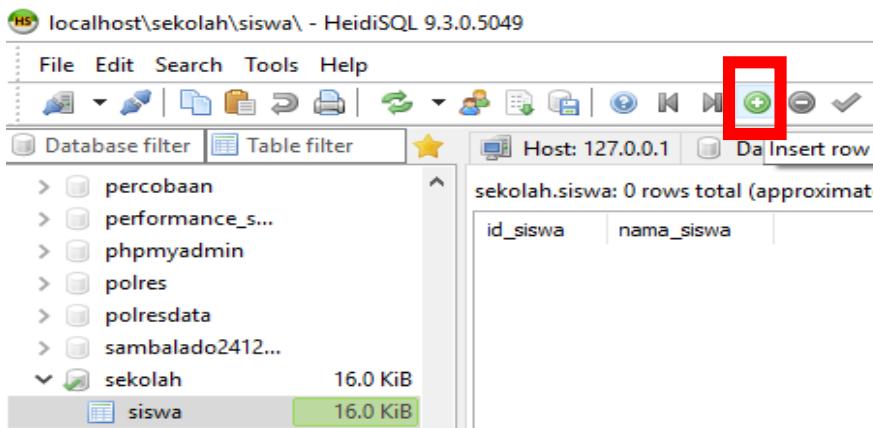
```
SHOW TABLES;
```

e) DML

- Pengelolaan **data** dalam tabel.
- Bentuk **CRUD**:
 1. Create
 2. Read (Max, Min, Sum, dll)
 3. Update
 4. Delete

e) DML Script - Create

GUI



CLI

INSERT INTO

```
`sekolah`.`siswa`  
(`id_siswa`,  
 `nama_siswa`) VALUES  
(`2016111234`, 'Furi  
Hikmawati');
```

e) DML Script - Read

GUI

The screenshot shows the MySQL Workbench interface. At the top, there are tabs for Host: 127.0.0.1, Database: sekolah, Table: siswa, and a redboxed Data tab. Below the tabs, a message says 'sekolah.siswa: 1 rows total (approximately)'. A table below contains one row with columns id_siswa and nama_siswa. The data is: id_siswa 2016111234 and nama_siswa Furi hikmawati.

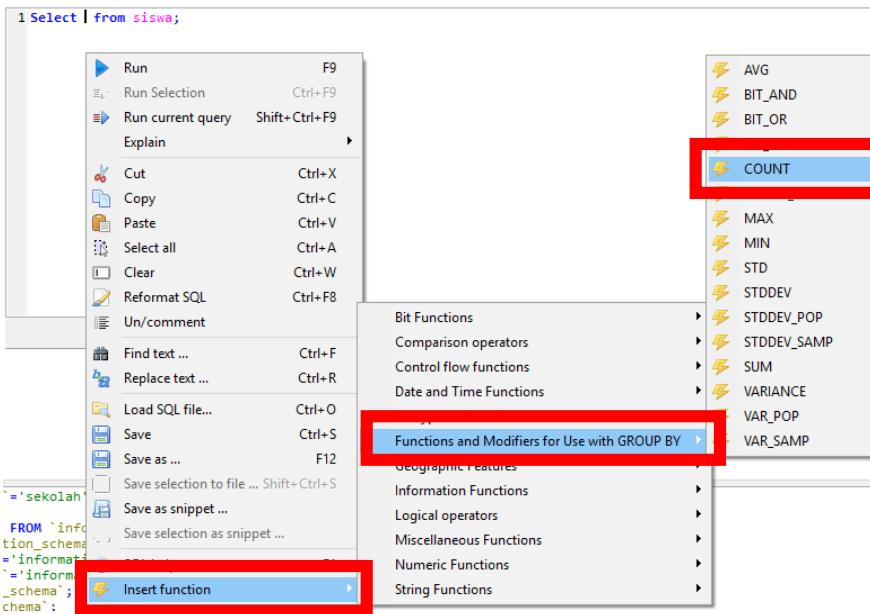
id_siswa	nama_siswa
2016111234	Furi hikmawati

CLI

```
SELECT * FROM  
`sekolah`.`siswa`;
```

e) DML Script - Read (count)

GUI

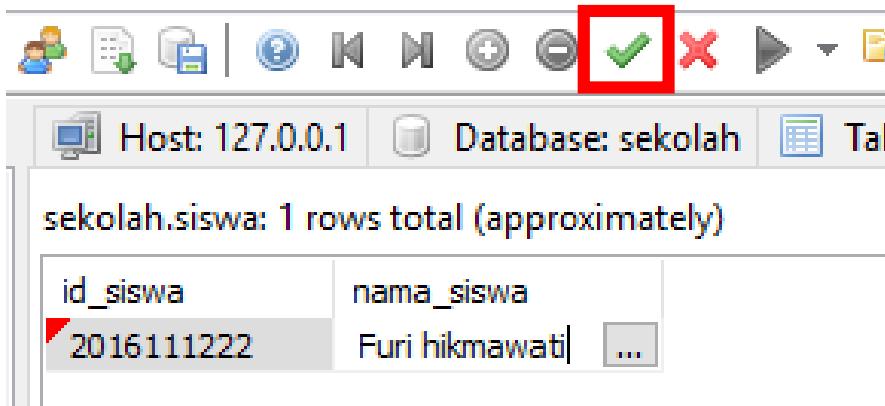


CLI

```
Select COUNT(*) from siswa;
```

e) DML Script - Update

GUI

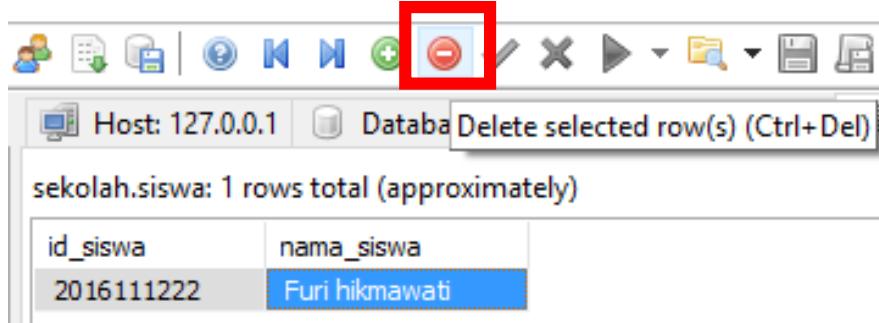


CLI

```
UPDATE `sekolah`.`siswa`  
SET  
`id_siswa`='2016111222'  
WHERE  
`id_siswa`='2016111234'  
AND `nama_siswa`='Furi  
hikmawati' LIMIT 1;
```

e) DML Script – Delete

GUI



CLI

DELETE FROM

```
`sekolah`.`siswa` WHERE
`id_siswa`='2016111222'
AND `nama_siswa`='Furi
hikmawati' LIMIT 1;
```