

# MODUL PRAKTIKUM PEMOGRAMAN DAN BASIS DATA

Dosen Pengampu : Adri Arisena, S. Si. M.Stat



Asisten Laboratorium : Teni Deinarosa H

*-Modul ini diperuntukan bagi Mata Kuliah  
Pemograman dan Basis Data Program Studi S1  
Agribisnis-*

# PENGANTAR BASIS DATA, ATTRIBUT DAN TIPE DATA

## 1. PENDAHULUAN

### 1.1. Data dan Informasi

- **Data** adalah kumpulan fakta atau angka yang belum diolah. Contohnya: nama seseorang, tanggal lahir, nomor telepon.
- **Informasi** adalah data yang sudah diolah sehingga punya arti dan bisa digunakan untuk mengambil keputusan.

### 1.2. Sistem Informasi

Sistem Informasi adalah gabungan dari orang, alat, komputer, prosedur, dan data yang bekerja bersama untuk menyimpan, mengolah, dan menyajikan informasi.

Sistem Informasi Manajemen adalah sistem yang dipakai di dalam sebuah organisasi atau perusahaan untuk mengumpulkan, menyimpan, mengolah, dan menyajikan data menjadi informasi yang berguna, khususnya untuk mendukung pengambilan keputusan manajemen.

### 1.3. Komponen system informasi

- **Hardware** : alat fisik seperti komputer, printer.
- **Software** : program atau aplikasi.
- **Personal** : yang menggunakan sistem, seperti operator atau pengguna.
- **Data** : bahan mentah yang akan diolah.
- **Prosedur** : aturan atau langkah-langkah cara kerja sistem.

### 1.4. Basis Data

Basis data adalah kumpulan data yang saling berhubungan dan disimpan secara rapi dalam komputer agar mudah dicari, ditambah, diubah, dan dihapus.

### 1.5. Operasi Dasar Basis Data

- **Create Database**: Perintah yang digunakan untuk membuat basis data dengan nama yang diberikan atau membuat database baru.
- **Drop Database** : Perintah yang digunakan untuk menghapus basis data dengan nama yang diberikan atau menghapus database .
- **Create Table** : Perintah yang digunakan untuk menciptakan suatu tabel dalam basis data atau membuat tabel di dalam database.
- **Drop Table** : Perintah yang digunakan untuk menghapus suatu tabel dalam basis data atau menghapus tabel.
- **Insert** : Perintah yang digunakan untuk memasukkan data (record) ke dalam tabel atau menambah data ke tabel.
- **Update** : Perintah yang digunakan untuk memperbaharui data (record) pada table atau mengubah data.
- **Delete** : Perintah yang digunakan untuk menghapus data (record) pada table atau menghapus data.

### 1.6. Pemanfaatan Basis Data

- **Meningkatkan Efisiensi**: Data yang terorganisir memudahkan pengelolaan.
- **Mengurangi Redundansi Data**: Database memastikan data tidak duplikat.

- Meningkatkan Keamanan Data: Hanya pengguna tertentu yang dapat mengakses data sensitif.
- Mendukung Pengambilan Keputusan: Data yang akurat dan up-to-date membantu pengambilan keputusan yang tepat

### 1.7. Sistem Manajemen Basis Data (DBMS)

DBMS Merupakan perangkat lunak yang didesain untuk melakukan penyimpanan dan pengaturan basis data. Sistem Manajemen Basis Data (DBMS) juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan data, dll. Atau singkatnya DBMS adalah perangkat lunak yang digunakan untuk membuat dan mengelola database.



PostgreSQL

ORACLE

### 1.8. Peranan Basis data dalam pengembangan system

**Sistem Informasi** berperan sebagai sistem karena mempunyai ruang lingkup yang relatif lebih luas dan lebih kompleks. Sedangkan sistem basis data merupakan subsistem karena menjadi bagian dan berada di dalam Sistem Informasi.

**Sistem basis data** adalah sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lain dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

Keberadaan sistem basis data di dalam Sistem Informasi adalah mutlak. Sistem Informasi tidak akan terwujud tanpa melibatkan basis data.

## 2. SISTEM BASIS DATA

### 2.1. Pengertian sistem basis data

Sistem basis data adalah kumpulan data yang saling berhubungan dan dipakai bersama oleh banyak orang, menggunakan komputer dan perangkat lunak khusus untuk mengatur dan mengelola data tersebut.

### 2.2. Komponen Sistem Basis Data

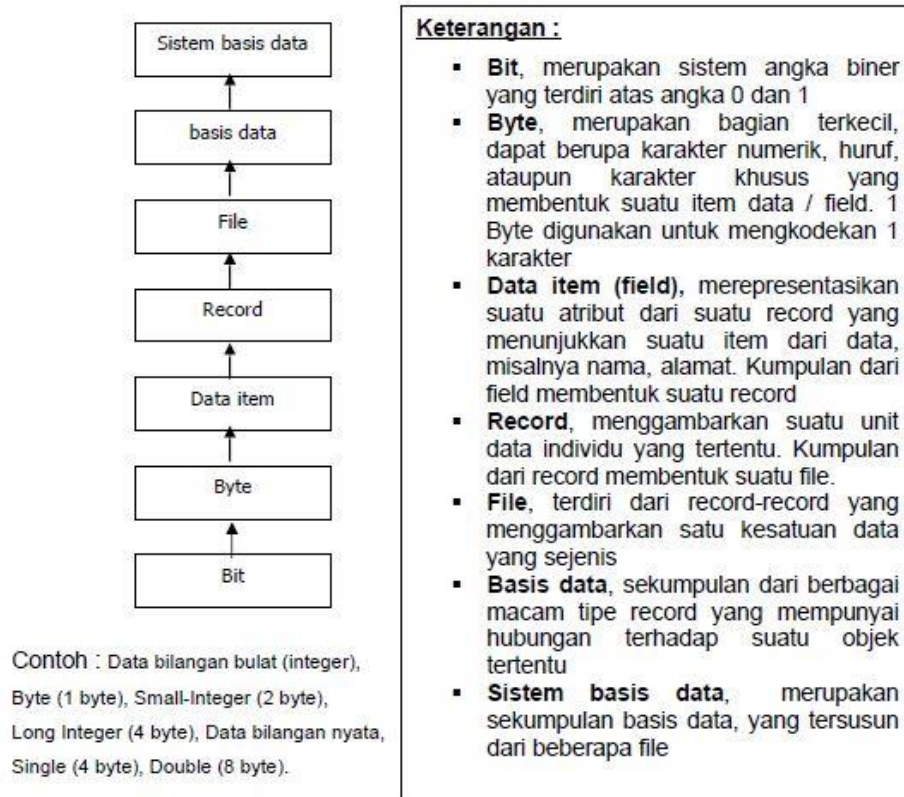
- Perangkat keras: komputer, server, harddisk.
- Sistem operasi: Windows, Linux, dan lainnya.
- Basis data: tempat penyimpanan data.
- DBMS: perangkat lunak pengatur basis data.

Pemakai:

- Programmer aplikasi: membuat program.
- Casual user: pengguna yang mahir query.
- End user: hanya memakai aplikasi.
- Specialized user: pengguna khusus untuk aplikasi tertentu.

### 2.3. Penyusun Sistem Basis Data

Sistem basis data mencakup seluruh komponen data dalam organisasi. Basis data adalah bagian inti, sedangkan sistem basis data adalah lingkungan lengkap yang terdiri dari perangkat keras, perangkat lunak, data, orang, dan prosedur.



## 2.4. Bahasa Basis Data

- **DDL (Data Definition Language):** Perintah untuk membuat, mengubah, dan menghapus struktur tabel dan basis data. Hasil dari kompulasi DLL menjadi kamus data, yaitu data yang menjelaskan data sesungguhnya.
- **DML (Data Manipulation Language):** Kumpulan perintah query yang digunakan untuk memanipulasi data pada database. DML Berguna untuk melakukan manipulasi dan pengambilan data pada suatu basis data, yang berupa insert, update, delete, dll. Ada 2 jenis, yaitu :
  - Prosedural: menyebutkan data yang diambil dan cara mendapatkannya.
  - Non-Prosedural: hanya menyebutkan data yang diinginkan.

## 2.5. Penggunaan Basis Data

- **Database Administrator**  
Database administrator (DBA) merupakan pengguna yang memiliki kewenangan untuk melakukan pengawasan baik data maupun program. Fungsi DBA adalah
  - Mendefinisikan struktur basis data
  - Mendefinisikan struktur penyimpanan dan metode akses
  - Memodifikasi organisasi fisik

- Memberikan kewenangan pada user untuk mengakses data
- Menspesifikasikan keharusan integritas data

### **3. LINGKUNGAN BASIS DATA**

#### **3.1. Masalah yang Sering Muncul dalam Basis Data**

- Redundansi Data : Data yang sama tersimpan berulang-ulang.
- Inkonsistensi Data : Data berbeda untuk hal yang sama di beberapa tempat.
- Isolasi Data : Data tersebar di beberapa tempat sehingga sulit dicari.
- Banyak Pemakai : Data diakses bersamaan oleh banyak pengguna.
- Keamanan Data : Data harus aman, hanya bisa diakses oleh yang berhak.
- Integritas Data : Data harus valid dan sesuai aturan.
- Independensi Data : Perubahan data tidak memaksa perubahan program.

#### **3.2. Organisasi File Basis Data**

Tujuan: Agar data mudah dicari, ditambah, diperbarui, atau dihapus.

Jenis Media Penyimpanan:

- SASD (Sequential Access Storage Device): Data dibaca urut, seperti kaset.
- DASD (Direct Access Storage Device): Data bisa langsung diakses, seperti harddisk.

Metode Penyimpanan:

- Sequential : Data disimpan dan dibaca berurutan.
- Random : Data disimpan acak.
- Indexed Sequential : Gabungan urut dan acak dengan indeks.
- Indexed Random : Acak tapi ada indeks.

#### **3.3. Arsitektur system basis data**

Jenis arsitektur system basis data :

- Sistem tunggal (Standalone)  
DBMS, basis data, dan aplikasi basis data ditempatkan pada komputer yang sama. Hanya bisa dipakai oleh satu pemakai pada saat yang bersamaan.
- Sistem Terpusat (Centralized system)  
Terdiri dari sebuah server dan sejumlah terminal. Bagian yang terpusat adalah basis data, DBMS, dan aplikasi basis data.
- Sistem Client-server  
Ditujukan untuk mengatasi kelemahan yang terdapat pada sistem terpusat. Terdiri dari 2 komponen utama yaitu client dan server. Client berisi aplikasi basis data; server berisi DBMS dan basis data.

#### **3.4. Konsep DBMS**

DBMS (Data Base Management System) merupakan perangkat lunak yang memberikan fasilitas untuk melakukan fungsi pengaturan, pengawasan, pengendalian, pengolahan, dan koordinasi terhadap semua proses yang terjadi pada sistem basis data.

Komponen-komponen utama DBMS:

- Query language  
Digunakan oleh bagian lain dengan sedikit perintah sederhana  
Contoh : SQL (Structure Query Language), QBE (Query By Example)
- Report generator

Dirancang untuk membuat cetakan, yang memiliki perintah-perintah untuk membuat header, judul, kolom, summary, dll.

- **DML (Data Manipulation Language)**  
Terdiri dari perintah-perintah yang disediakan dalam program aplikasi untuk melakukan manipulasi data seperti append, list, atau update
- **DDL (Data Definition Language)**  
Dengan bahasa ini dapat membuat tabel baru, membuat indeks, mengubah tabel, menentukan struktur tabel, dll. Hasil dari kompilasi perintah DDL menjadi Kamus Data, yaitu data yang menjelaskan data sesungguhnya  
Contoh: Create, Modify report, Modify structure
- **Recovery**  
Merupakan kemampuan untuk mengembalikan data yang rusak atau hilang akibat operasi basis data (insert, update, delete, dll.)
- **Data dictionary**  
Digunakan untuk memelihara definisi-definisi standar seluruh rinci data dalam lingkup kecil pada sistem basis data
- **Database**  
Merupakan bagian dari DBMS yang menyediakan data dalam berbagai tipe dan format untuk memenuhi kebutuhan pemakai
- **Access routine**  
Suatu rutin yang dapat dipanggil dan dipergunakan oleh program lain untuk mengakses basis data

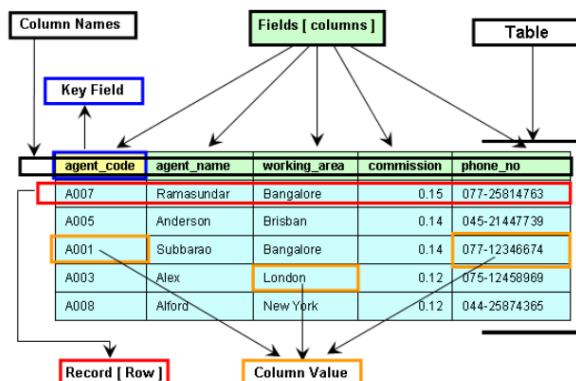
### 3.5. Kamus data

DBMS memberikan fasilitas data dictionary (kamus data) untuk mendefinisikan nama-nama rinci data dan format penyimpanannya. Kamus data digunakan untuk:

- Pada tahap analisis, sebagai alat komunikasi antara analis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem
- Pada tahap perancangan sistem, digunakan untuk merancang input, laporan-laporan dan database

Kamus data berisi: Nama arus data, alias, bentuk data, arus data, penjelasan atau keterangan - keterangan, periode terjadinya transaksi, volume arus data yang mengalir dalam periode tertentu, struktur data.

## 4. ISTILAH, ATTRIBUT, DAN TIPE DATA.



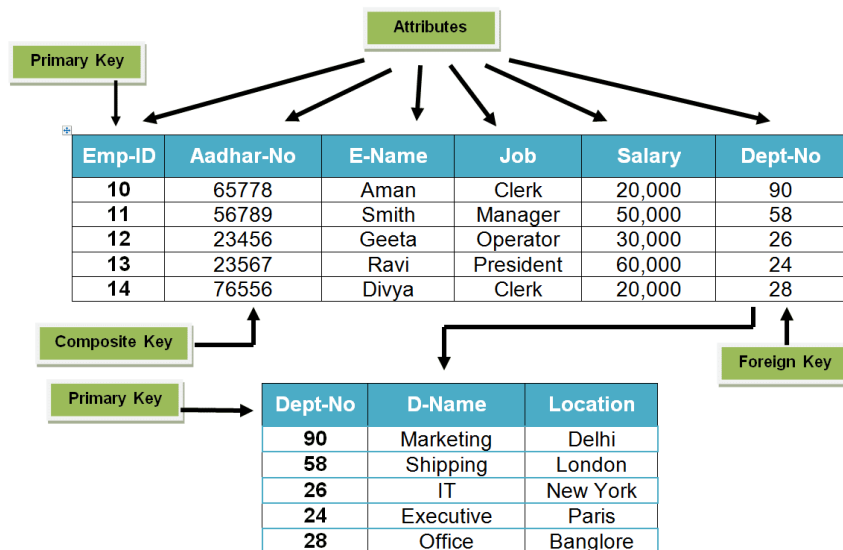
#### 4.1. Istilah

- Tabel: Struktur utama dalam database, terdiri dari baris dan kolom.
- Baris (Row/Record): Mewakili satu data individu dalam tabel.
- Kolom (Field): Mewakili atribut atau karakteristik data.
- Key Field: Kolom unik yang mengidentifikasi setiap baris dalam tabel.
  - Primary Key (PK): Kolom atau kombinasi kolom yang secara unik mengidentifikasi setiap baris dalam tabel.
  - Foreign Key (FK): Kolom yang digunakan untuk membuat hubungan dengan kolom kunci utama di tabel lain.
- Record/Tuple : Sebuah baris dalam suatu relasi. Disebut juga tuple
- Cardinality : Banyaknya record dalam sebuah relasi
- Atribut : Suatu kolom dalam sebuah relasi
- Domain : Batasan nilai dalam atribut dan tipe datanya
- Derajat / degree: Banyaknya kolom dalam relasi

#### 4.2. Jenis Atribut

- Atribut Kunci (Key Attribute): Kolom yang isinya harus unik dan bisa dipakai untuk membedakan setiap data. Contoh: NIM.
- Atribut Sempel (Simple Attribute): Data yang tidak bisa dipecah lagi. Contoh: Nama.
- Atribut Multinilai (Multivalued Attribute): Kolom yang bisa berisi lebih dari satu nilai. Contoh: Nomor telepon.
- Atribut Gabungan (Composite Attribute): Kolom yang bisa dipecah jadi beberapa bagian. Contoh: Nama lengkap jadi nama depan, tengah, belakang.
- Atribut Turunan (Derived Attribute): Data yang dihasilkan dari atribut lain. Contoh: Umur dihitung dari tanggal lahir.

#### 4.3. Jenis Key



- Primary Key : Kunci utama, harus unik dan tidak boleh kosong.
- Secondary Key : Digunakan untuk pencarian, tidak wajib unik.
- Candidate Key : Calon primary key.
- Alternate Key : Candidate key yang tidak dipakai sebagai primary key.
- Composite Key : Gabungan dua atau lebih kolom sebagai primary key.

- Foreign Key : Kolom yang merujuk ke primary key di tabel lain.
- Super Key : Gabungan satu atau lebih kolom yang bisa membedakan data.

#### 4.4. Relasi

- Unary relation : Suatu relasi yang hanya mempunyai satu kolom
- Binary relation : Suatu relasi yang hanya mempunyai dua kolom
- Ternary relation : Suatu relasi yang mempunyai tiga kolom

#### 5. Tipe Data

Tipe data adalah suatu bentuk pemodelan data yang dideklarasikan pada saat melakukan pembuatan tabel. Tipe data ini akan mempengaruhi setiap data yang akan dimasukkan ke dalam sebuah tabel. Data yang akan dimasukkan harus sesuai dengan tipe data yang dideklarasikan.

Berbagai type data pada MySQL dapat dilihat pada tabel berikut:

Type Data	Keterangan
TINYINT	Ukuran 1 byte. Bilangan bulat terkecil, dengan jangkauan untuk bilangan bertanda: -128 sampai dengan 127 dan untuk yang tidak bertanda : 0 s/d 255. Bilangan tak bertanda dengan kata UNSIGNED
SMALLINT	Ukuran 2 Byte. Bilangan bulat dengan jangkauan untuk bilangan bertanda : 32768 s/d 32767 dan untuk yang tidak bertanda : 0 s/d 65535
MEDIUMINT	Ukuran 3 byte. Bilangan bulat dengan jangkauan untuk bilangan bertanda : - 8388608 s/ d 8388607 dan untuk yang tidak bertanda : 0 s/d 16777215
INT	Ukuran 4 byte. Bilangan bulat dengan jangkauan untuk bilangan bertanda : - 2147483648 s/d 2147483647 dan untuk yang tidak bertanda : 0 s/d 4294967295
INTEGER	Ukuran 4 byte. Sinonim dari int
BIGINT	Ukuran 8 byte. Bilangan bulat terbesar dengan jangkauan untuk bilangan bertanda : - 9223372036854775808 s/d 9223372036854775807 dan untuk yang tidak bertanda : 0 s/d 1844674473709551615
FLOAT	Ukuran 4 byte. Bilangan pecahan
DOUBLE	Ukuran 8 byte. Bilangan pecahan



DOUBLEPRECISION	Ukuran 8 byte. Bilangan pecahan
REAL	Ukuran 8 byte. Sinonim dari DOUBLE
DECIMAL (M,D)	Ukuran M byte. Bilangan pecahan, misalnya DECIMAL(5,2) dapat digunakan untuk menyimpan bilangan -99,99 s/d 99,99
NUMERIC (M,D)	Ukuran M byte. Sinonim dari DECIMAL, misalnya NUMERIC(5,2) dapat digunakan untuk menyimpan bilangan -99,99 s/d 99,99

Type Data untuk Bilangan (Number)

Type Data	Keterangan
DATETIME	Ukuran 8 byte. Kombinasi tanggal dan jam, dengan jangkauan dari „1000-0101 00:00:00“ s/d „9999-12-31 23:59:59“
DATE	Ukuran 3 Byte. Tanggal dengan jangkauan dari ‘1000- 01-01’ s/d ‘9999-12-31’
TIMESTAMP	Ukuran 4 byte. Kombinasi tanggal dan jam, dengan jangkauan dari ‘1970-0101 00:00:00’ s/d ‘2037’
TIME	Ukuran 3 byte. Waktu dengan jangkauan dari ‘839:59:59’ s/d ‘838:59:59’
YEAR	Ukuran 1 byte. Data tahun antara 1901 s/d 2155

Type Data untuk Tanggal dan Jam

Type Data	Keterangan
CHAR	Mampu menangani data hingga 255 karakter. Tipe data CHAR mengharuskan untuk memasukkan data yang telah ditentukan.

VARCHAR	Mampu menangani data hingga 255 karakter. Tipe data VARCHAR tidak mengharuskan untuk memasukkan data yang telah ditentukan.
TINYBLOB, TINYTEXT	Ukuran 255 byte. Mampu menangani data sampai $2^8-1$ data.
BLOB, TEXT	Ukuran 65535 byte. Tipe string yang mampu menangani data hingga $2^{16}-1$ (16M-1) data.
MEDIUMBLOB, MEDIUMTEXT	Ukuran 16777215 byte. Mampu menyimpan data hingga $2^{24}-1$ (16M-1) data.
LOB, LONGTEXT	Ukuran 4294967295 byte. Mampu menyimpan data hingga berukuran GIGA BYTE. Tipe data ini memiliki batas penyimpanan hingga $2^{32}-1$ (4G-1) data.
ENUM('nilai1','nilai2',..., 'nilaiN')	Ukuran 1 atau 2 byte. Tergantung jumlah nilai enumerasinya (maksimum 65535 nilai)
SET('nilai1','nilai2',..., 'nilaiN')	1,2,3,4 atau 8 byte, tergantung jumlah anggota himpunan (maksimum 64 anggota)

**CONTOH:**

ID_Peserta	Kode_Kursus	Nama	Email	No_HP	Jadwal	Nilai
P001	K001	Andi	<a href="mailto:andi@gmail.com">andi@gmail.com</a>	081234567890	Senin, 09.00	A
P001	K002	Andi	<a href="mailto:andi@gmail.com">andi@gmail.com</a>	081234567890	Rabu, 10.00	B+
P002	K001	Budi	<a href="mailto:budi@gmail.com">budi@gmail.com</a>	081298765432	Senin, 09.00	B

Konsep	Penjelasan Singkat
Primary Key	Kombinasi ID_Peserta + Kode_Kursus → unik untuk setiap peserta di tiap kursus

<b>Composite Key</b>	ID_Peserta + Kode_Kursus karena dua kolom digabung jadi kunci utama
<b>Candidate Key</b>	Bisa pakai ID_Peserta + Kode_Kursus atau Email + Kode_Kursus → dua-duanya unik
<b>Alternate Key</b>	Email + Kode_Kursus tidak dipilih sebagai primary key → jadi alternate key
<b>Foreign Key</b>	Kode_Kursus bisa merujuk ke tabel lain: Daftar_Kursus(Kode_Kursus)
<b>Secondary Key</b>	Nama atau Jadwal → bisa digunakan untuk pencarian, tapi tidak unik
<b>Super Key</b>	Kombinasi kolom yang bisa mengidentifikasi baris, misalnya: ID_Peserta + Kode_Kursus + No_HP
<b>Record (Tuple)</b>	Baris data. Contoh: baris pertama adalah satu record
<b>Cardinality</b>	Jumlah baris dalam tabel = <b>3 record</b>
<b>Atribut (Kolom)</b>	Setiap nama kolom seperti ID_Peserta, Email, Nilai → total <b>7 atribut</b>
<b>Domain</b>	Nilai yang mungkin untuk sebuah kolom. Misalnya: Nilai = {A, B+, B, C, D, E}
<b>Degree</b>	Jumlah kolom dalam tabel = <b>7</b>

### LATIHAN!!

Jawab pertanyaan dibawah ini! (Boleh di Buku Catatan atau di Microsoft Word)

1. Jelaskan konsep basis data secara singkat dengan bahasamu sendiri!
2. Sebutkan minimal 3 keuntungan menggunakan basis data dibanding menyimpan data di file biasa!
3. Carilah atau buatlah sendiri satu tabel sederhana (misalnya tentang data mahasiswa, pelanggan toko, pendaftaran kursus, karyawan, dll.) yang terdiri dari minimal 6 kolom dan 5 baris data.
  - Sebutkan tipe data apa saja yang dimuat dalam tiap field/kolom tabelnya.
  - Berdasarkan tabel tersebut, identifikasikan dan jelaskan masing-masing elemen berikut:
    - Primary Key: Kolom apa yang dipakai sebagai primary key? Kenapa?
    - Composite Key: Apakah ada kombinasi dua kolom yang bisa dijadikan composite key? Sebutkan!
    - Candidate Key: Kolom atau kombinasi kolom apa yang bisa jadi candidate key?

- Alternate Key: Jika ada lebih dari satu candidate key, mana yang jadi alternate key?
- Foreign Key: Jika tabel kamu bisa terhubung ke tabel lain, kolom mana yang jadi foreign key?
- Secondary Key: Kolom mana yang sering dipakai untuk pencarian data?
- Super Key: Kombinasi kolom apa saja yang dapat secara unik mengidentifikasi data?
- Record: Berapa banyak baris (record) dalam tabel kamu?
- Cardinality: Berapa jumlah total baris dalam tabel?
- Atribut: Sebutkan semua nama kolom (atribut) pada tabel kamu.
- Domain: Jelaskan domain dari minimal 2 atribut di tabel kamu.
- Degree: Berapa jumlah kolom (atribut) pada tabel kamu?