

Dosen Pengampu: Adri Arisena

# PRAKTIKUM 4

## Pengantar Basis Data, Attribut, dan Tipe Data

Asisten Laboratorium:  
Najlia Intani



*"Modul ini diperuntukkan bagi Mata Kuliah Pemrograman Dasar dan Basis Data Program Studi Agribisnis"*

## A. PENDAHULUAN

---

### A. Data dan Informasi

Data merupakan nilai yang merepresentasikan deskripsi dari suatu objek atau kejadian. Informasi merupakan hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian- kejadian nyata yang digunakan untuk pengambilan keputusan. Data lebih bersifat historis, sedangkan informasi mempunyai tingkatan yang lebih tinggi, lebih dinamis, serta mempunyai nilai yang sangat penting.

### B. Sistem Informasi

Sistem Informasi merupakan suatu sistem dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur dan pengendalian untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan.

Sistem Informasi Manajemen merupakan sekumpulan elemen yang saling berhubungan, saling berinteraksi dan bekerjasama antara berbagai bagian dengan cara-cara tertentu untuk melakukan fungsi pengolahan data, pemasukan data, dan menghasilkan keluaran berupa informasi yang berguna dan mempunyai nilai nyata, sebagai dasar pengambilan keputusan, mendukung kegiatan manajemen dan operasional dengan memanfaatkan berbagai sumber daya yang ada bagi proses tersebut guna mencapai tujuan organisasi.

### C. Komponen Sistem Informasi

1. Hardware : CPU, Disk, Terminal, Printer
2. Software : Sistem operasi, sistem basis data, program aplikasi
3. Personil : Operator sistem, Penyedia masukan, Pengguna, Keluaran
4. Data : Data yang tersimpan dalam jangka waktu tertentu
5. Prosedur : Instruksi dan kebijaksanaan untuk mengoperasikan sistem

### D. Basis Data

Basis Data merupakan suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan software untuk melakukan manipulasi untuk kegunaan tertentu. Basis data bisa diartikan juga sebagai sekumpulan data yang disusun dalam bentuk beberapa tabel yang saling memiliki relasi maupun berdiri sendiri.

### E. Operasi Dasar Basis Data

- Create database : Perintah yang digunakan untuk membuat basis data dengan nama yang diberikan
- Drop database : Perintah yang digunakan untuk menghapus basis data dengan nama yang diberikan

- Create table : Perintah yang digunakan untuk menciptakan suatu tabel dalam basis data
- Drop table : Perintah yang digunakan untuk menghapus suatu tabel dalam basis data
- Insert : Perintah yang digunakan untuk memasukkan data (record) ke dalam tabel
- Update : Perintah yang digunakan untuk memperbarui data (record) pada table
- Delete : Perintah yang digunakan untuk menghapus data (record) pada table

## F. Pemanfaatan Basis Data

1. Menentukan kualitas informasi : akurat, tepat waktu dan relevan.
2. Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi
3. Mengurangi duplikasi data (data redundancy)
4. Hubungan data dapat ditingkatkan
5. Manipulasi terhadap data dengan cepat dan mudah
6. Efisiensi penggunaan ruang penyimpanan

## G. Penerapan Basis Data

Tidak ada sistem informasi yang bisa dibangun tanpa adanya basis data, sehingga bisa dikatakan posisi basis data pada sebuah sistem informasi adalah sangat penting.

## H. Kriteria Basis Data

- Dapat digunakan oleh beberapa program aplikasi tanpa mengubah basis datanya
- Berorientasi pada data dan bukan berorientasi pada program
- Dapat berkembang dengan mudah, baik volume maupun strukturnya
- Dapat digunakan dengan cara berbeda-beda
- Kerangkapan data minimal

## I. Sistem Manajemen Basis Data (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.

## J. Peranan Basis Data dalam Pengembangan Sistem

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

## B. SISTEM BASIS DATA

### A. Pengertian Sistem Basis Data

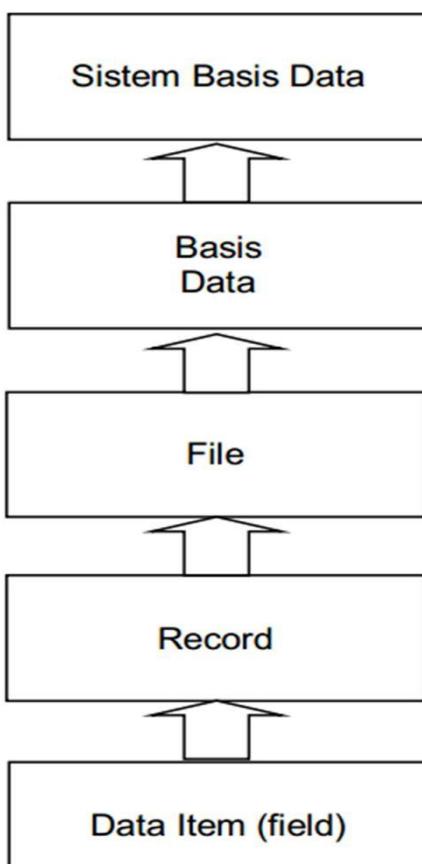
Sistem Basis Data merupakan sekumpulan basis data dengan para pemakai yang menggunakan basis data secara bersama-sama, personil yang merancang dan mengelola basis data, teknik-teknik untuk merancang dan mengelola basis data, serta sistem komputer yang mendukungnya.

### B. Komponen Sistem Basis Data

1. Perangkat keras
2. Sistem operasi
3. Basis data
4. Sistem pengelola basis data (DBMS)
5. Pemakai (Programmer, User mahir, user umum, user khusus)

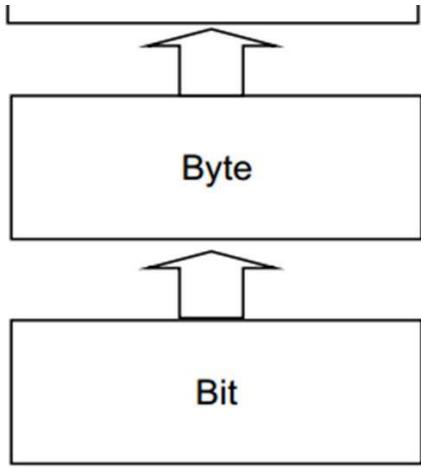
### C. Penyusun Sistem Basis Data

Sistem basis data merupakan lingkup terbesar dalam organisasi data. Sistem basis data mencakup semua bentuk komponen data yang ada dalam suatu sistem. Sedangkan basis data merupakan komponen utama yang menyusun sistem basis data.



#### Keterangan :

- **Bit**, merupakan sistem angka biner yang terdiri atas angka 0 dan 1
- **Byte**, merupakan bagian terkecil, dapat berupa karakter numerik, huruf, ataupun karakter khusus yang membentuk suatu item data / field. 1 Byte digunakan untuk mengkodekan 1 karakter
- **Data item (field)**, merepresentasikan suatu atribut dari suatu record yang menunjukkan suatu item dari data, misalnya nama, alamat. Kumpulan dari field membentuk suatu record
- **Record**, menggambarkan suatu unit data individu yang tertentu. Kumpulan dari record membentuk suatu file.
- **File**, terdiri dari record-record yang menggambarkan satu kesatuan



- data yang sejenis
- **Basis data**, sekumpulan dari berbagai macam tipe record yang mempunyai hubungan terhadap suatu objek tertentu
  - **Sistem basis data**, merupakan sekumpulan basis data, yang tersusun dari beberapa file

## D. Bahasa Basis Data

Bahasa basis data merupakan perantara bagi pemakai dengan basis data dalam berinteraksi, yang telah ditetapkan oleh pembuat DBMS. Bahasa basis data dapat dibedakan menjadi 2, yaitu:

### 1) Data Definition Language (DDL)

DDL merupakan kumpulan perintah SQL yang digunakan untuk membuat (create), mengubah (alter) dan menghapus (drop) struktur dan definisi tipe data dari objek-objek database. 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.

### 2) Data Manipulation Language (DML)

Data Manipulation Language merupakan 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 (ditentukan data yang diinginkan dan cara mendapatkannya) dan non-prosedural (tanpa menyebutkan cara mendapatkannya)

## E. Pengguna Basis Data

### 1) 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

## 2) Database User

Ada 4 pemakai basis data, yaitu :

1. Programmer aplikasi: Merupakan pembuat program aplikasi
2. Casual user / Naive User: Pemakai yang sudah mahir, berinteraksi dengan sistem tanpa menulis program, tapi menggunakan query
3. End user: Pemakai yang belum mahir, hanya menjalankan aplikasi yang sudah dibuat oleh programmer aplikasi
4. Specialized user: Pemakai khusus yang menuliskan aplikasi database tidak dalam kerangka pemrosesan data, namun untuk keperluan khusus seperti CAD, AI, ES, dll

## C. LINGKUNGAN BASIS DATA

---

### A. Kekangan Dalam Basis Data

Penyusunan basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data, yaitu:

#### 1) Redundansi Data

Redudansi data merupakan munculnya data-data yang sama secara berulang-ulang pada beberapa file basis data yang semestinya tidak diperlukan. Akan mengakibatkan proses updating lebih lama dan memungkinkan terjadinya inconsistency data.

#### 2) Inkonsistensi Data

Inkonsistensi data merupakan munculnya data yang tidak konsisten pada field yang sama untuk beberapa file dengan kunci yang sama. Terjadi akibat kesalahan dalam pemasukan data atau update data. Akan mengakibatkan kesalahan pada hasil pengolahan basis data yang tidak sesuai dengan fakta

#### 3) Isolasi Data untuk Standarisasi

Disebabkan oleh pemakaian beberapa file basis data yang tersebar dalam beberapa file, hal ini menyulitkan program-mer untuk mengambil dan menyimpan data.

#### 4) Banyak Pemakai (Multi User)

Basis data dapat diakses oleh beberapa pemakai secara simultan, karena data yang diolah tidak bergantung dan menyatu dalam program tapi terlepas dalam satu kelompok data.

#### 5) Masalah Keamanan (Security)

Pada prinsipnya file basis data hanya boleh diakses oleh pemakai tertentu yang mempunyai wewenang. Pembatasan dapat dilakukan melalui DBMS atau program aplikasi.

## 6) Masalah Integritas (Integrity)

Untuk menjaga agar unjuk kerja sistem tetap dalam pengendalian penuh. Secara teknis maka ada kunci primer yang menghubungkan beberapa file yang saling berkaitan.

## 7) Masalah Kebebasan Data (Independence)

Basis data yang dirancang hendaknya tidak bergantung pada program aplikasi yang dibangun. Sehingga apabila ada perubahan terhadap field, tidak perlu merubah struktur programnya

## B. Organisasi File Basis Data

Tujuan organisasi file dalam sistem basis data :

- Menyediakan sarana pencarian record bagi pengolahan, seleksi, atau penyaringan
- Memudahkan pembuatan atau pemeliharaan file

Ada 2 jenis media penyimpanan file :

1. SASD (Sequential Access Storage Device)
  - Proses pembacaan record harus berurutan
  - Tidak ada pengalaman
  - Data disimpan dalam bentuk blok
  - Proses penulisan hanya bisa dilakukan sekali
- Contoh : magnetic tape
2. DASD (Direct Access Storage Device)
  - Pembacaan record tidak harus urut
  - Mempunyai alamat
  - Data dapat disimpan dalam karakter atau blok
  - Proses penulisan dapat dilakukan beberapa kali
- Contoh : harddisk, floppy disk

Metode susunan file:

1. Sequential (urut)
  - Record disimpan berdasarkan suatu kunci
  - Pencarian record tertentu dilakukan record demi record berdasarkan kuncinya
2. Random (Acak)
  - Kunci record ditransformasikan ke alamat penyimpanan dalam media fisik secara acak
3. Indexed Sequential
  - Merupakan gabungan antara metode urut dan acak
  - Record disimpan secara berurutan dengan menggunakan kunci
  - Masing-masing record memiliki indeks
  - Pengalaman dilakukan secara acak
4. Indexed Random
  - Record disimpan secara acak
  - Masing-masing record memiliki indeks

## C. Arsitektur sistem basis data

Pertimbangan dalam memilih arsitektur sistem basis data:

- Keunggulan teknologi
- Biaya pengembangan
- Sesuai dengan kebutuhan pengguna

Jenis arsitektur sistem basis data:

- a. 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.
- b. Sistem Terpusat (Centralized system)  
Terdiri dari sebuah server dan sejumlah terminal. Bagian yang terpusat adalah basis data, DBMS, dan aplikasi basis data.
- c. 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

## D. 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:

- a. Query language  
Digunakan oleh bagian lain dengan sedikit perintah sederhana  
Contoh : SQL (Structure Query Language), QBE (Query By Example)
- b. Report generator  
Dirancang untuk membuat cetakan, yang memiliki perintah-perintah untuk membuat header, judul, kolom, summary, dll.
- c. DML (Data Manipulation Language)  
Terdiri dari perintah-perintah yang disediakan dalam program aplikasi untuk melakukan manipulasi data seperti append, list, atau update
- d. 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
- e. Recovery  
Merupakan kemampuan untuk mengembalikan data yang rusak atau hilang akibat operasi basis data (insert, update, delete, dll.)

- f. Data dictionary  
Digunakan untuk memelihara definisi-definisi standar seluruh rinci data dalam lingkup kecil pada sistem basis data
- g. Database  
Merupakan bagian dari DBMS yang menyediakan data dalam berbagai tipe dan format untuk memenuhi kebutuhan pemakai
- h. Access routine  
Suatu rutin yang dapat dipanggil dan dipergunakan oleh program lain untuk mengakses basis data

## E. 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

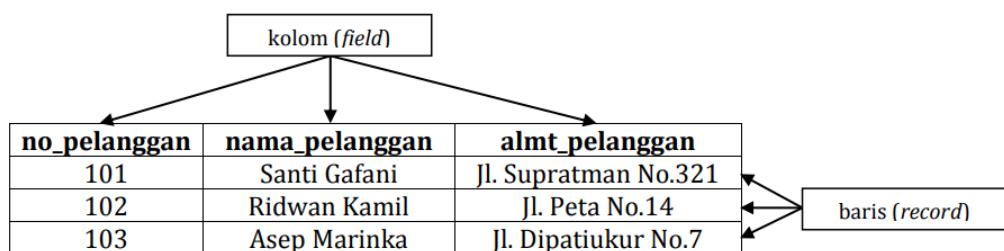
## D. ISTILAH, ATTRIBUT, DAN TIPE DATA

---

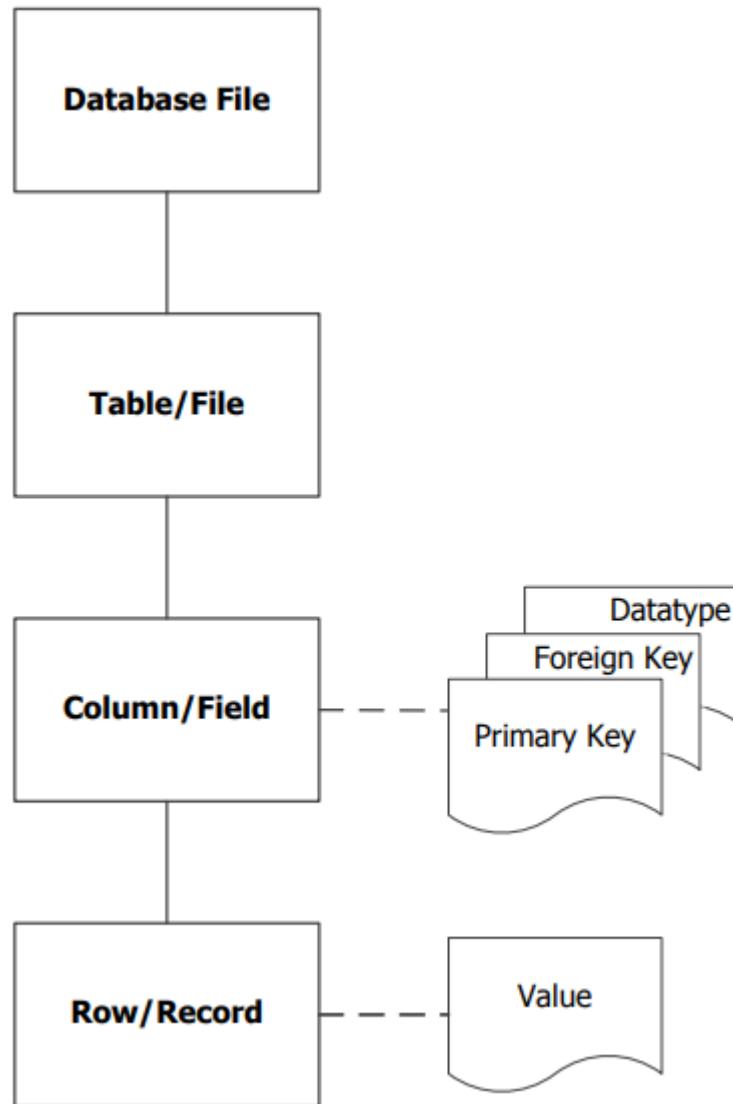
### A. Istilah

Tabel merupakan kumpulan informasi secara logis yang terkait dan diperlakukan sebagai unit. Setiap tabel selalu terdiri atas lajur vertikal yang biasa disebut dengan kolom atribut (column/field) dan lajur horizontal yang biasa disebut dengan baris data (row/record).

Disetiap pertemuan kolom atribut dan baris data ditempatkan item-item data (satuan data terkecil). Contoh tabel ditunjukkan pada gambar di bawah ini.



Hirarki dari sebuah elemen basis data dapat dilihat di bawah ini



- 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

## B. Atribut

### 1. Atribut Kunci

Merupakan atribut yang digunakan untuk menentukan data yang bersifat unik. Pada umumnya, data dari atribut key berbentuk angka. Contohnya NIM (Nomor Induk Mahasiswa), No. KTP, SIM, NPWP, dan lain sebagainya.

### 2. Atribut Simpel

Yaitu atribut yang tidak dapat dipecah lagi atau atomic dan bernilai tunggal. Contohnya adalah alamat rumah, kantor, nama penerbit, tahun terbit jurnal, dan lain – lain.

3. Atribut Multinilai (Multivalue)

Merupakan atribut yang memiliki sekelompok nilai untuk setiap entitas -nya. Contoh dari atribut multivalue adalah kumpulan nama pengarang dalam sebuah novel.

4. Atribut Gabungan (Composite)

Yaitu atribut yang berasal dari susunan atribut yang lebih kecil dalam artian tertentu.

Contohnya adalah data terkait nama lengkap, yang terdiri dari nama depan, tengah, dan belakang.

5. Atribut Derivatif

Merupakan atribut yang berasal dari atribut lain dan tidak bersifat wajib untuk ditulis pada ERD. Contohnya adalah usia, selisih waktu, kelas atau ruang, dan lain sebagainya.

6. Atribut Kunci (Key)

Atribut unik yang digunakan untuk mengidentifikasi entitas.

| Jenis Key            | Pengertian  |
|----------------------|---|
| <b>Primary Key</b>   | Kunci utama untuk mengidentifikasi setiap baris secara unik; tidak boleh NULL atau duplikat.                                      |
| <b>Secondary Key</b> | Atribut yang digunakan untuk pencarian atau pengurutan, bukan untuk identifikasi unik.  |
| <b>Candidate Key</b> | Atribut atau kombinasi atribut yang berpotensi menjadi primary key karena bersifat unik.  |
| <b>Alternate Key</b> | Candidate key yang tidak dipilih sebagai primary key.   |
| <b>Composite Key</b> | Primary key yang terdiri dari dua atau lebih atribut untuk menciptakan nilai unik.  |
| <b>Foreign Key</b>   | Atribut yang merujuk ke primary key di tabel lain untuk membentuk relasi antar tabel.   |
| <b>Super Key</b>     | Kumpulan satu atau lebih atribut yang dapat mengidentifikasi baris secara unik, termasuk candidate key dan tambahan atribut lain. |

| Jenis Key        | Unik<br>(Unique) | Boleh<br>NULL | Fungsi Utama   | Jumlah per<br>Tabel                 | Merujuk<br>ke Tabel<br>Lain? |
|------------------|------------------|---------------|--|-------------------------------------|------------------------------|
| Primary<br>Key   | ✓                | ✗             | Identifikasi utama<br>baris  | 1                                   | ✗                            |
| Secondary<br>Key | ✗ / ✓            | ✓             | Digunakan untuk<br>pencarian atau<br>pengurutan                    | $\geq 0$                            | ✗                            |
| Candidate<br>Key | ✓                | ✗             | Calon yang valid<br>untuk dijadikan<br>primary key                 | $\geq 1$                            | ✗                            |
| Alternate<br>Key | ✓                | ✗             | Candidate key<br>yang tidak dipilih<br>sebagai primary<br>key      | $\geq 0$                            | ✗                            |
| Composite<br>Key | ✓                | ✗             | Kunci utama dari<br>kombinasi dua atau<br>lebih atribut            | 1 (jika<br>digunakan<br>sebagai PK) | ✗                            |
| Foreign<br>Key   | ✗                | ✓             | Menunjukkan<br>hubungan antar<br>tabel                             | $\geq 0$                            | ✓                            |
| Super Key        | ✓                | Tergantung    | Kunci unik<br>termasuk candidate<br>key & tambahan<br>atribut lain | Banyak<br>(termasuk<br>PK & CK)     | ✗                            |

## 7. 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

## C. 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,<br>TINYTEXT     | Ukuran 255 byte. Mampu menangani data sampai $2^{8-1}$ data.  |
| BLOB, TEXT                | Ukuran 65535 byte. Type string yang mampu menangani data hingga $2^{161}$ (16M-1) data.                                     |
| MEDIUMBLOB,<br>MEDIUMTEXT | Ukuran 16777215 byte. Mampu menyimpan data hingga $2^{24-1}$ (16M-1) data.  |

|                                      |  |
|--------------------------------------|--|
| LONGBLOB,<br>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('nilai','nilai2',...,'nilaiN')   | 1,2,3,4 atau 8 byte, tergantung jumlah anggota himpunan (maksimum 64 anggota)  |

## E. 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,<br>09.00 | A     |
| P001       | K002        | Andi | <a href="mailto:andi@gmail.com">andi@gmail.com</a> | 081234567890 | Rabu,<br>10.00  | B+    |
| P002       | K001        | Budi | <a href="mailto:budi@gmail.com">budi@gmail.com</a> | 081298765432 | Senin,<br>09.00 | B     |

| Konsep               | Penjelasan Singkat  |
|----------------------|---|
| <b>Primary Key</b>   | 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 → dual-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>  |

## F. 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:

| <b>Konsep Basis Data</b> | <b>Pertanyaan</b>   |
|--------------------------|---|
| <b>Primary Key</b>       | Apa primary key dalam tabelmu? Mengapa itu dipilih sebagai primary key?     |
| <b>Composite Key</b>     | Apakah ada kombinasi dua kolom yang bisa menjadi composite key? Sebutkan.   |
| <b>Candidate Key</b>     | Apa saja kombinasi kolom yang <i>mungkin</i> dijadikan primary key?         |
| <b>Alternate Key</b>     | Jika ada lebih dari satu candidate key, mana yang jadi alternate key?       |
| <b>Foreign Key</b>       | Jika tabelmu bisa berelasi ke tabel lain, kolom mana yang jadi foreign key? |

|                      |  |
|----------------------|--|
| <b>Secondary Key</b> | Kolom apa yang sering dicari/dipakai dalam pencarian data?               |
| <b>Super Key</b>     | Kombinasi kolom apa saja yang dapat secara unik mengidentifikasi record? |
| <b>Record</b>        | Berapa banyak baris (record) dalam tabelmu?                              |
| <b>Cardinality</b>   | Berapa jumlah total baris dalam tabelmu?                                 |
| <b>Atribut</b>       | Sebutkan semua nama kolom (atribut) pada tabelmu.                        |
| <b>Domain</b>        | Jelaskan domain dari minimal 2 atribut dalam tabelmu.                    |
| <b>Degree</b>        | Berapa jumlah kolom (atribut) pada tabelmu?                              |