



PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

MATA KULIAH
Sistem Basis Data



Data Definition Language (DDL)

TEKNIK INFORMATIKA S1
UNIVERSITAS DIAN NUSWANTORO

Capaian Pembelajaran

- Mahasiswa dapat menerapkan konsep DDL untuk menampilkan database, menghapus database, menampilkan struktur tabel, menghapus tabel, memodifikasi struktur tabel, me-rename nama tabel, serta menentukan kunci primer dan kunci komposit

Pokok Bahasan

- Penerapan Bahasa basis data DDL
- Menentukan Kunci Primer
- Menentukan Kunci Komposit

Penerapan Bahasa Basis Data DDL

- DDL merupakan perintah SQL yang berhubungan dengan pendefinisian suatu struktur database, dalam hal ini *database* dan *table*.
- Perintah yang termasuk DDL:
 - CREATE
 - ALTER
 - RENAME
 - DROP

Penerapan Bahasa Basis Data DDL

Membuat database:

```
CREATE DATABASE nama_database;
```

Menggunakan database:

```
USE nama_database;
```

Melihat database:

```
SHOW DATABASES;
```

Menghapus database:

```
DROP DATABASE nama_database;
```

Penerapan Bahasa Basis Data DDL

Menampilkan Tabel:

Show tables;

Menampilkan Struktur Tabel:

DESC nama_tabel;

Mengubah Nama Tabel:

RENAME TABLE nama_table TO nama_table_baru;

Menghapus Tabel:

Drop table nama_table;

Penerapan Bahasa Basis Data DDL

Membuat Tabel

```
CREATE TABLE nama_table (  
    field1 tipe_data(panjang), ..., fieldn tipe_data(panjang),  
    PRIMARY KEY(field_key)  
);
```

Melihat Tabel

```
SHOW TABLES;
```

Melihat Deskripsi Tabel

```
DESC nama_table;
```

Latihan 1

- Buatlah database dengan nama **olshop**
- Buatlah table **produk** jika diberikan contoh table seperti berikut:

| id_produk | nama | merk | satuan | harga |
|-----------|-------------------------|----------|--------|--------|
| ATK01 | Kertas Fotocopy A4 70gr | Sidu | Rim | 30000 |
| COM01 | Speaker N730 | Advance | Buah | 250000 |
| MEB01 | Rak 4 Pintu | Universe | Buah | 200000 |

- Tentukan tipe data dan primary key dari table diatas

Penerapan Bahasa Basis Data DDL

ALTER digunakan untuk mengubah struktur table

Jenis Perintah Alter:

- **ADD** : Menambah kolom
- **CHANGE** : Mengganti nama kolom dan type datanya
- **DROP** : Menghapus kolom atau indeks

ALTER TABLE - ADD

- Penggunaan klausa **ADD** pada **Alter Table** digunakan untuk menambah kolom pada suatu tabel

- **Format Penulisan**

ALTER TABLE *nama_table* **ADD** *nama_kolom tipe_data;*

- **Contoh**

- menambah kolom id_kategori pada table produk

ALTER TABLE produk **ADD** id_kategori int;

ALTER TABLE - CHANGE

- Penggunaan klausa **CHANGE** pada **ALTER TABLE** digunakan untuk mengubah nama kolom dan tipe datanya pada suatu tabel

- **Format Penulisan**

ALTER TABLE nama_table **CHANGE** nama_kolom nama_kolom_baru tipe_data;

- **Contoh**

- mengubah kolom nama dengan nama produk pada table produk

ALTER TABLE produk **CHANGE** nama nama_produk varchar (200)

Latihan 2

| id_produk | Nama_produk | merk | satuan | harga | Id_kategori |
|-----------|-------------------------|----------|--------|--------|-------------|
| ATK01 | Kertas Fotocopy A4 70gr | Sidu | Rim | 30000 | |
| COM01 | Speaker N730 | Advance | Buah | 250000 | |
| MEB01 | Rak 4 Pintu | Universe | Buah | 200000 | |

- Berdasarkan table produk di atas
- Tambahkan kolom **id_penjual** dengan tipe data int
- Ubahlah kolom harga menjadi harga satuan dengan tipe data yang sama
- Hapus kolom merk

ALTER TABLE - DROP

- Penggunaan klausa **DROP** pada **ALTER TABLE** digunakan untuk menghapus kolom pada suatu tabel

- **Format Penulisan**

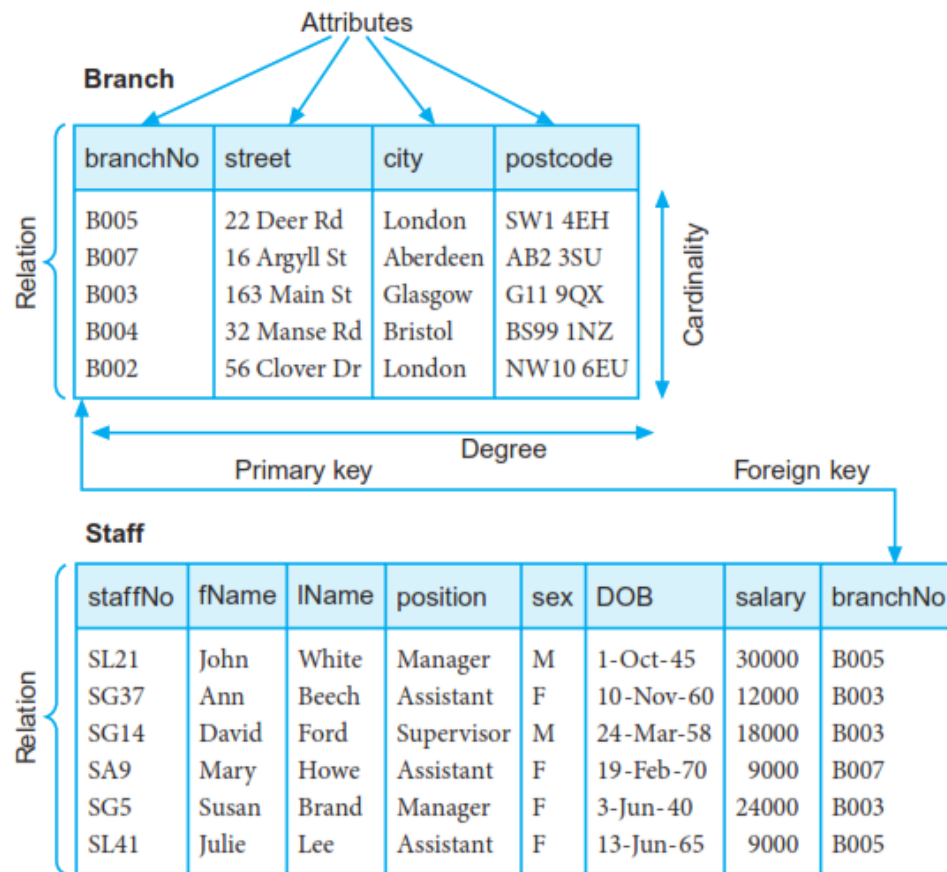
ALTER TABLE *nama_table* **DROP** *nama_kolom*;

- **Contoh**

- Menghapus kolom satuan pada tabel produk

ALTER TABLE produk **DROP** satuan;

KUNCI PRIMER (PRIMARY KEY)



- Suatu table **Branch** memiliki **Primary Key** yaitu **branchNo**, dimana **branchNo** akan bertindak sebagai **Foreign Key** pada table **Staff**

KUNCI PRIMER (PRIMARY KEY)

- Maka dalam pembuatan kunci primer kita dapat lakukan dengan menulis

Create table Branch(
 branchNo varchar(5) NOT NULL **primary key**,

 street varchar(200) NOT NULL,

 city varchar(25) NOT NULL,

 postcode varchar(10)

);

KUNCI PRIMER (PRIMARY KEY)

- Secara umum, suatu skema relasi memungkinkan memiliki lebih dari satu kunci. Dalam hal ini, setiap kunci disebut sebagai kunci kandidat (candidate key).
- Sebagai contoh, relasi CAR pada Figure 5.4 memiliki dua kunci kandidat yaitu **License_number** dan **Engine_serial_number**.
- Hal yang umum untuk menunjuk salah satu kunci kandidat sebagai kunci utama dari relasi.
- Bagaimana jika kita ingin membuat primary key lebih dari satu pada suatu table?

KUNCI KOMPOSIT (COMPOSITE KEY)

CAR

| <u>License_number</u> | <u>Engine_serial_number</u> | Make | Model | Year |
|-----------------------|-----------------------------|------------|---------|------|
| Texas ABC-739 | A69352 | Ford | Mustang | 02 |
| Florida TVP-347 | B43696 | Oldsmobile | Cutlass | 05 |
| New York MPO-22 | X83554 | Oldsmobile | Delta | 01 |
| California 432-TFY | C43742 | Mercedes | 190-D | 99 |
| California RSK-629 | Y82935 | Toyota | Camry | 04 |
| Texas RSK-629 | U028365 | Jaguar | XJS | 04 |

Figure 5.4

The CAR relation, with two candidate keys: License_number and Engine_serial_number.

- Pada table di atas kita akan membuat lebih dari satu primary key, yaitu **License_number** dengan **Engine_serial_number**, maka dapat kita tuliskan dengan cara:

Primary key(field1, field2, ...)

Latihan 3

CAR

| <u>License_number</u> | Engine_serial_number | Make | Model | Year |
|-----------------------|----------------------|------------|---------|------|
| Texas ABC-739 | A69352 | Ford | Mustang | 02 |
| Florida TVP-347 | B43696 | Oldsmobile | Cutlass | 05 |
| New York MPO-22 | X83554 | Oldsmobile | Delta | 01 |
| California 432-TFY | C43742 | Mercedes | 190-D | 99 |
| California RSK-629 | Y82935 | Toyota | Camry | 04 |
| Texas RSK-629 | U028365 | Jaguar | XJS | 04 |

- Buatlah tabel Car menyesuaikan struktur table di atas
- Terapkan kolom License_number dan Engine_serial_number sebagai primary key
- Isilah dengan tiga data

KUNCI KOMPOSIT (COMPOSITE KEY)

Create table car(
License_number varchar(100) NOT NULL,
Engine_serial_number varchar(8) NOT NULL,
Make varchar(15),
Model varchar(25),
Year char(2),
primary key(License_number, Engine_serial_number));

Referensi

UTAMA

1. Silberschatz, A., Korth, H. F. & Sudarshan, S., 2022. Database System Concepts. 7th ed. New York: McGraw-Hill Education
2. Connolly, T. & Begg, C., 2015. Database Systems A practical Approach to Design, Implementation, and Management. Sixth Edition ed. s.l.:Pearson.
3. Elmasri, R. & Navathe, S. B., 2016. Fundamentals of Database Systems. 7th ed. s.l.:Pearson

PENDUKUNG

Aripin., 2005. *Praktikum Basis Data Dengan Database Server MySQL*. Semarang: Fakultas Ilmu Komputer



TERIMA KASIH

ANY QUESTIONS?