



## Basis Data 2

Sirojul Munir S.Si,M.Kom

[rojulman@nurulfikri.ac.id](mailto:rojulman@nurulfikri.ac.id) |  rojulman  
[dev.xbata.com](http://dev.xbata.com)

# SQL COMMAND

# SQL dan RDBMS

---

- ❑ **SQL (Structure Query Language)** - adalah suatu bahasa query yang dikembangkan oleh IBM pada proyek DBMS relasional system -R.
- ❑ Selama beberapa tahun SQL menjadi bahasa query yang digunakan untuk melakukan pendefinisian skema, manipulasi data dan query pada DBMS relasional.

# Standarisasi SQL (1)

---

- ❑ Banyak digunakan vendor pada produk RDBMS
- ❑ Dibutuhkan standar penulisan SQL.
- ❑ Produk RDBMS diharapkan mengacu pada suatu standar yang ditetapkan.
- ❑ Diharapkan tidak ada perbedaan perintah SQL pada produk yang dikeluarkan.

## Standarisasi SQL (2)

---

### ***American National Standards Institute (ANSI) mengeluarkan Standard SQL***

- ☐ 1986: SQL-86
- ☐ 1989: SQL-89 ; perubahan minor
- ☐ 1992: SQL-92 ; ANSI+ISO – perubahan besar
- ☐ 1999: SQL-99
- ☐ 2003: SQL-2003

# SQL: Structured Query Language

---

- ❑ **Data Definition Language (DDL)** –  
Perintah SQL untuk mendefinisikan skema database atau table : CREATE, DROP, ALTER
- ❑ **Data Manipulation Language (DML)** –  
Perintah SQL untuk manipulasi data dan pencarian data : INSERT, UPDATE, DELETE, SELECT
- ❑ **Data Control Language (DCL)** –  
Perintah SQL untuk mengontrol database atau table : GRANT, REVOKE

# DDL

## Mendefinisikan Database

---

- *Buat database*

- Perintah SQL :

***CREATE TABLE NAMA\_TABLE ;***

- Contoh :

```
mysql> CREATE DATABASE dbpenjualan ;
```

```
mysql> CREATE DATABASE dbtesting;
```

```
mysql> SHOW DATABASES;
```

- *Hapus database*

- Perintah SQL :

***DROP DATABASE NAMA\_DATABASE ;***

- Contoh :

```
mysql> DROP DATABASE dbtesting;
```

# DDL

## Mendefinisikan Table

---

- *Buat table*

- *Perintah SQL :*

- ```
CREATE TABLE nama_table (  
    nama_field tipe_data constraint,  
    ...  
    )Engine=Engine_Name;
```

- *Lihat table-table pada database*

- ```
SHOW TABLES;
```

- *Hapus table*

- *Perintah SQL :*

- ```
DROP TABLE nama_table;
```



# DDL

## Mendefinisikan Table

---

- Buat table Produk

```
CREATE TABLE produk (  
  idproduk integer auto_increment primary  
  key,  
  kode varchar(6) unique not null,  
  nama varchar(40) not null,  
  stok integer,  
  min_stok integer,  
  harga double  
)Engine=InnoDB;
```

# DDL:

## Mengubah Skema Table

---

### □ Tambah field baru

*ALTER TABLE NAMA\_TABLE ADD NAMA\_FIELD\_BARU TIPE\_DATA;*

*Contoh:*

*ALTER TABLE produk ADD keterangan VARCHAR(30);*

### □ Mengubah tipe data field

*ALTER TABLE NAMA\_TABLE CHANGE NAMA\_FIELD NAMA\_FIELD\_BARU TIPE\_DATA;*

*Contoh:*

*ALTER TABLE produk CHANGE keterangan ket VARCHAR(100);*

### □ Menghapus field

*ALTER TABLE NAMA\_TABLE DROP NAMA\_FIELD\_AKAN\_DIHAPUS;*

*Contoh:*

*ALTER TABLE produk ADD keterangan VARCHAR(30);*

# Skema Table Produk

---

## □ Melihat skema table

```
mysql> desc produk;
```

| Field    | Type        | Null | Key | Default | Extra          |
|----------|-------------|------|-----|---------|----------------|
| idproduk | int(11)     | NO   | PRI | NULL    | auto_increment |
| kode     | varchar(6)  | NO   | UNI | NULL    |                |
| nama     | varchar(40) | NO   |     | NULL    |                |
| stok     | int(11)     | YES  |     | NULL    |                |
| min_stok | int(11)     | YES  |     | NULL    |                |
| harga    | double      | YES  |     | NULL    |                |

# DML: Query INSERT

---

- Perintah SQL INSERT

***INSERT INTO nama\_table ([NAMA\_FIELD]) VALUES (DATA\_INPUTAN);***

- Input data ke table produk

***mysql> INSERT INTO produk  
(kode,nama,stok,min\_stok,harga) VALUES  
( 'TV01' , 'Televisi 21 inch' , 20 , 3 , 1200000 )***

- Melihat isi table produk :

***mysql> SELECT \* FROM produk;***

# DML : Query SELECT

- Inputlah data-data berikut ini ke table produk

```
mysql> select * from produk;
```

| <i>idproduk</i> | <i>kode</i> | <i>nama</i>      | <i>stok</i> | <i>min_stok</i> | <i>harga</i> |
|-----------------|-------------|------------------|-------------|-----------------|--------------|
| 1               | TV01        | Televisi 21 inch | 20          | 3               | 1200000      |
| 2               | K001        | Kulkas 2 Pintu   | 10          | 2               | 3150000      |
| 3               | K002        | Kulkas Mini      | 15          | 3               | 750000       |
| 4               | S001        | Sofa Tamu        | 8           | 1               | 3600000      |
| 5               | M001        | Meja Makan       | 1           | 3               | 850000       |
| 6               | MI01        | Mie Instan       | 100         | 20              | 1250         |
| 7               | TE01        | Teh Kotak        | 10          | 20              | 4000         |

# Latihan Query SELECT

---

- Tampilkan seluruh data produk diurutkan berdasarkan harga mulai dari terkecil

***SELECT \* FROM produk ORDER BY harga;***

- Tampilkan data kode,nama dan harga dari table produk

***SELECT kode,nama,harga FROM produk;***

- Tampilkan seluruh data customer diurutkan berdasarkan nama dari secara descending

***SELECT \* FROM customer ORDER BY nama DESC;***

- Tampilkan kode,nama dan email dari table customer

***SELECT kode,nama,email FROM customer;***

# OPERATOR PERBANDINGAN

---

## ***Operator perbandingan***

- ☐ = sama dengan
- ☐ < lebih kecil dari
- ☐ <= lebih kecil dari atau sama dengan
- ☐ > lebih besar dari
- ☐ >= lebih besar dari atau sama dengan
- ☐ <> tidak sama dengan

# Query SELECT dengan kondisi

---

- Tampilkan seluruh data produk yang memiliki kode “K001”

***SELECT \* FROM produk WHERE kode='K001'***

- Tampilkan data kode,nama dan harga dari table produk yang harganya diatas 3juta

***SELECT kode,nama,harga FROM produk WHERE harga > 3000000;***

- Tampilkan seluruh data customer perempuan

***SELECT \* FROM customer WHERE gender = 'P';***

- Tampilkan kode,nama dan tempat lahir dari table customer yang lahirnya bukan di Jakarta

***SELECT kode,nama,tmp\_lahir FROM customer WHERE tmp\_lahir <> 'Jakarta';***



# Operator AND dan OR

---

***Tampilkan seluruh data produk yang stoknya diatas 10 dan kurang dari 100***

☐ `SELECT * FROM produk WHERE stok > 10  
AND stok <100`

***Tampilkan data produk yang kode produknya 'K001' atau 'TE01'***

☐ `SELECT * FROM produk WHERE  
kode='K001' OR kode='TE01'`

# Fungsi Waktu

---

- ❑ `current_date` : mengambil tanggal saat ini pada komputer server
- ❑ `current_time` : mengambil jam saat ini
- ❑ `Year(kolom_tanggal)`: mengambil tahun dari kolom yang bertipe data tanggal (datetime)
- ❑ `Month(kolom_tanggal)`: mengambil bulan dari kolom yang bertipe data tanggal (datetime)
- ❑ `day(kolom_tanggal)`: mengambil tanggal dari kolom yang bertipe data tanggal (datetime)

Tampilkan nama dan tahun lahir customer

- ❑ `SELECT nama, year(tgl_lahir) FROM customer`

Tampilkan bulan saat ini

- ❑ `SELECT month(current_date)`

# Operator BETWEEN

---

***Tampilkan data produk yang mempunyai stok antara 10 sampai 100***

- ❑ `SELECT * FROM produk WHERE stok >= 10  
AND stok <=100`
- ❑ `SELECT * FROM produk WHERE stok  
BETWEEN 10 AND 100`

# Operator LIKE dan NOT LIKE

---

***Operator LIKE digunakan untuk pencarian data yang biasanya untuk tipe data character***

Cari data produk yang field nama berawalan huruf K

❑ `SELECT * FROM produk WHERE nama like 'K%'`

Cari data produk yang field nama mengandung huruf UL

❑ `SELECT * FROM produk WHERE nama like '%UL%'`

Cari data produk yang field nama berakhiran huruf AN

❑ `SELECT * FROM produk WHERE nama like '%AN'`

# Operator LIMIT dan OFFSET

---

***Operator LIMIT digunakan untuk menampilkan sejumlah baris data dari table, sedangkan OFFSET untuk menentukan baris awal data yang akan ditampilkan***

Tampilkan 3 data pertama dari table produk

❑ `SELECT * FROM produk LIMIT 3`

Tampilkan 3 data produk dimulai dari data ke 3

❑ `SELECT * FROM produk LIMIT 3 OFFSET 2`

# Fungsi Aggregate

---

***SQL telah mempunyai fungsi aggregate :***

- ☐ COUNT (hitung jumlah data)
  - ☐ SUM (total nilai suatu kolom)
  - ☐ AVG (hitung rata-rata suatu kolom)
  - ☐ MAX / MIN (mendapatkan nilai maksimum atau minimum suatu kolom)
- 

Ada berapa jumlah data customer :

- ☐ `SELECT COUNT(*) AS jumlah_customer FROM customer`

Berapa rata-rata stok produk

- ☐ `SELECT AVG(stok) AS rata2 FROM produk`

# Keyword DISTINCT

---

***Keyword DISTINCT digunakan untuk menampilkan informasi data suatu kolom yang bernilai berbeda ( data yang sama akan dimunculkan satu kali saja)***

Tampilkan kota-kota tempat kelahiran

```
❑ SELECT DISTINCT(tmp_lahir) as kota_kelahiran  
FROM customer
```

# Keyword GROUP BY

---

***Keyword GROUP BY digunakan untuk mengelompokkan Data hasil fungsi aggregate yang dikombinasikan dengan KEYWORD DISTINCT***

Tampilkan statistik jumlah customer berdasarkan gender

❑ `SELECT DISTINCT(gender),COUNT(*)`

`FROM customer`

`GROUP BY gender`

❑ SOAL :

- 1) Tampilkan data statistik jumlah tempat lahir customer



# Operasi Aritmatika

---

***SQL telah men-support operasi aritmatika***

- ☐ + (penjumlahan)
- ☐ \* (perkalian)
- ☐ / (pembagian)
- ☐ - (pengurangan)

Tampilkan hasil perkalian  $5 * 5$  :

☐ `SELECT 5 + 5 AS '5 +5 = '`

Tampilkan total asset masing2 produk

☐ `SELECT kode,nama,stok* harga as asset from produk`

# HAPUS DATA

---

*DELETE FROM nama\_table  
WHERE  
field\_constraint=nilai\_constraint;*

- ❑ Ketika ingin menghapus data pastikan anda mengetahui data yang akan dihapus dengan mengidentifikasi field kunci (primary key)
- ❑ **Hapus data customer yang memiliki idcustomer=6**  
**DELETE FROM customer WHERE idcustomer=6**
- ❑ **Hapus data customer yang tinggal lahirnya di Jakarta**  
**DELETE FROM customer WHERE tmp\_lahir='Jakarta'**

# UPDATE DATA

---

```
UPDATE nama_table  
SET nama_field=data_baru, ...  
WHERE  
field_constraint=nilai_constraint;
```

- ❑ Ganti kode produk yang mempunyai idproduk 3

**UPDATE produk SET kode='K020' WHERE idproduk=3**

- ❑ Coba ganti data customer yang mempunyai idcustomer 3 dengan nama baru 'Defghi Arsy Muhammad'
- ❑ Coba update stok produk dengan jumlah stok lama + 20 yang stoknya sudah kurang dari min\_stok
- ❑ Naikan semua harga produk sebesar 20% dari harga semula

# Soal 1

---

- 1) Tampilkan data produk yang stoknya 10 dan 20
- 2) Tampilkan data produk yang harganya kurang dari 1jt tetapi lebih dari 500rb
- 3) Tampilkan data produk yang harus segera ditambah stoknya
- 4) Tampilkan data customer mulai dari yang paling muda
- 5) Tampilkan customer yang lahirnya di Jakarta dan gendernya perempuan
- 6) Tampilkan customer yang dicustomernya 2,4 dan 6
- 7) Tampilkan data customer yang lahirnya antara tahun 2003 sampai 2006
- 8) Tampilkan data customer yang lahirnya tahun 2005
- 9) Tampilkan data customer yang bulan ini berulang tahun
- 10) Tampilkan data customer yang tempat lahirnya di medan dan di semarang

# Soal 2

---

- 1) Tampilkan produk yang kode awalnya huruf K dan huruf M
- 2) Tampilkan produk yang kode awalnya bukan huruf M
- 3) Tampilkan produk-produk kulkas
- 4) Tampilkan customer mengandung huruf 'SA'
- 5) Tampilkan customer yang emailnya yahoo
- 6) Tampilkan customer yang emailnya bukan gmail
- 7) Tampilkan customer yang lahirnya bukan di Jakarta dan mengandung huruf 'fa'

# Soal 3

---

- 1) Tampilkan 2 data produk termahal
- 2) Tampilkan produk yang paling murah
- 3) Tampilkan produk yang stoknya paling banyak
- 4) Tampilkan dua produk yang stoknya paling sedikit
- 5) Tampilkan data customer diurutkan berdasarkan gender
- 6) Tampilkan data customer diurutkan berdasarkan tempat lahir customer
- 7) Tampilkan customer yang paling muda
- 8) Tampilkan customer yang paling tua

# Soal 4

---

- 1) Berapa jumlah customer yang tahun lahirnya 2005
- 2) Berapa jumlah data customer perempuan yang tempat lahirnya di Jakarta
- 3) Berapa jumlah total stok semua produk yang harganya dibawah 10rb
- 4) Ada berapa produk yang harganya diatas 1jt
- 5) Ada berapa produk yang mempunyai kode awal K
- 6) Berapa harga rata-rata produk yang diatas 1jt
- 7) Tampilkan jumlah stok yang paling besar
- 8) Ada berapa produk yang stoknya kurang
- 9) Berapa total asset dari keseluruhan produk
- 10) berapa rata-rata stok yang harganya diantara 1jt sampai 4 jt

# Soal 5

---

- 1) Berapa jumlah data customer perempuan yang tempat lahirnya di Jakarta
- 2) Berapa jumlah total stok semua produk yang harganya dibawah 10rb
- 3) Ada berapa produk yang harganya diatas 1jt
- 4) Ada berapa produk yang mempunyai kode awal K
- 5) Berapa harga rata-rata produk yang diatas 1jt
- 6) Tampilkan jumlah stok yang paling besar
- 7) Ada berapa produk yang stoknya kurang
- 8) Berapa total asset dari keseluruhan produk
- 9) berapa rata-rata stok yang harganya diantara 1jt sampai 4 jt



# Soal 6

---

- 1) Tampilkan produk yang assetnya diatas 20jt
- 2) Tampilkan data produk beserta selisih stok dengan minimal stok
- 3) Tampilkan usia masing-masing customer
- 4) Tampilkan rata-rata usia dari customer
- 5) Tampilkan total asset produk secara keseluruhan