



**PROGRAM STUDI**  
**TEKNIK INFORMATIKA**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS DIAN NUSWANTORO**

MATA KULIAH  
*Sistem Basis Data*



# Data Manipulation Language

TEKNIK INFORMATIKA S1  
UNIVERSITAS DIAN NUSWANTORO

# Capaian Pembelajaran

Mahasiswa mampu:

- Menerapkan perintah select untuk menampilkan informasi
- Menerapkan perintah insert untuk memasukkan data
- Menerapkan perintah update untuk mengubah data
- Menerapkan perintah delete untuk menghapus data

# Pokok Bahasan

SELECT

INSERT

UPDATE

DELETE



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# SELECT

# Menampilkan Informasi

- Perintah untuk menampilkan query dari database adalah SELECT.
- Perintah SELECT bertujuan untuk mengambil dan menampilkan data dari satu atau lebih tabel database
- Bentuk umum penulisan SQL:

---

```
SELECT <attribute-list>  
FROM <table-list>  
WHERE <condition>
```

---

**Attribut-list:** daftar nama atribut yang nilainya diambilnya oleh query

**Table-list:** daftar nama relasi atau table yang dibutuhkan untuk memproses query

**Condition:** kondisi bersyarat, mengidentifikasi tuple yang akan diambil oleh query

## Menampilkan semua Informasi Tabel (1)

- Untuk menampilkan semua informasi pada tabel dapat menuliskan semua kolom tabel **tanpa** klause WHERE.
- Nama table: staf

nip	nama	sex	posisi	tgl_masuk	gaji
A001	Dani	L	Asisten	2014-02-01	3800000
M001	Slamet	L	Manajer	2004-01-01	7500000
S001	Siti	P	Supervisor	1997-07-10	5000000

- Penulisan Query:

---

```
SELECT nip,nama,sex,posisi,tgl_masuk,gaji FROM staf;
```

---

## Menampilkan semua Informasi Tabel (2)

- Untuk menampilkan semua informasi pada tabel dapat menuliskan menggunakan asterisk (\*) **tanpa** klause WHERE.
- Penulisan Query:

---

```
SELECT * FROM staf;
```

---

# Menampilkan Informasi tertentu

- Menampilkan informasi hanya 1 kolom:
  - Contoh: menampilkan kolom nama pada tabel staf

```
SELECT nama FROM staf;
```

- Menampilkan informasi lebih dari 1 kolom:
  - Contoh: menampilkan kolom nip, nama, dan sex

```
SELECT nip, nama, sex FROM staf;
```



## Menampilkan informasi dengan Alias Name (AS)

- Perintah **AS** digunakan untuk mengganti nama kolom dengan alias.
- Misal: Ekspresi SQL melibatkan operasi aritmatika seperti penjumlahan, pengurangan, perkalian dan pembagian. Contoh: gaji \* 0.2, maka hasil operasi akan ditempatkan pada kolom selanjutnya dengan nama **gaji \* 0.2**
- Dengan menggunakan alias name, nama gaji \* 0.2 dapat diatur dengan nama alias misalnya **Tunjangan**.

## Contoh – Penggunaan Alias Name (AS)

### Menampilkan Query Tanpa AS

```
MariaDB [db_usaha]> SELECT
-> nip,
-> nama,
-> posisi,
-> gaji,
-> gaji*0.2
-> FROM staf;
```

nip	nama	posisi	gaji	gaji*0.2
a01	wahyu	asisten	5000000	1000000.0
a02	devi	asisten	3500000	700000.0
m01	budi	manajer	10000000	2000000.0
s01	sari	supervisor	7500000	1500000.0

### Menampilkan Query Dengan AS

```
MariaDB [db_usaha]> SELECT
-> nip,
-> nama,
-> posisi,
-> gaji,
-> (gaji*0.2) as tunjangan
-> FROM staf;
```

nip	nama	posisi	gaji	tunjangan
a01	wahyu	asisten	5000000	1000000.0
a02	devi	asisten	3500000	700000.0
m01	budi	manajer	10000000	2000000.0
s01	sari	supervisor	7500000	1500000.0

## SELECT dengan Klausa WHERE

- Pada contoh sebelumnya, perintah SELECT digunakan untuk menampilkan semua informasi dari tabel.
- Beberapa kondisi dapat menyebabkan ekspresi SQL membatasi query yang ditampilkan seperti:
  - Menampilkan data karyawan dengan gaji diatas 5.000.000
  - Menampilkan data karyawan dengan jenis kelamin laki - laki
  - Menampilkan data karyawan dengan posisi ='manajer' atau posisi = 'supervisor'
- Kondisi diatas dapat dicapai dengan penggunaan klausa WHERE diikuti kondisi pencarian yang menentukan baris yang diambil.

## Contoh SELECT dengan Klausa WHERE

- Menampilkan data karyawan dengan gaji diatas 5.000.000

```
MariaDB [db_usaha]> SELECT * FROM staf WHERE gaji > 5000000;
```

nip	nama	posisi	sex	tgl_masuk	gaji	id_cabang
m01	budi	manajer	L	1996-01-01	10000000	B001
s01	sari	supervisor	L	2000-02-05	7500000	B001

- Menampilkan data karyawan dengan posisi="manajer" atau "supervisor"

```
MariaDB [db_usaha]> SELECT * FROM staf WHERE posisi='manajer' OR posisi='supervisor';
```

nip	nama	posisi	sex	tgl_masuk	gaji	id_cabang
m01	budi	manajer	L	1996-01-01	10000000	B001
s01	sari	supervisor	L	2000-02-05	7500000	B001



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# INSERT

# Menambahkan Data ke Database

- Perintah untuk menambahkan data ke database adalah INSERT
- Untuk menambahkan data ke database, data yang ditambahkan ditentukan berdasarkan tupel atau menulis query yang hasilnya adalah tupel yang disisipkan.
- Bentuk umum penulisan SQL:

---

```
INSERT INTO TableName (columnList)  
VALUES (dataValueList) ;
```

---

- *tableName*, Nama tabel yang dibutuhkan untuk memproses query
- *columnList*, mewakili satu atau lebih nama kolom yang dipisahkan dengan koma
- *dataValueList*, jumlah nilai atribut harus disesuaikan dengan *columnList*

## Menambahkan Data ke Semua Kolom Table (1)

- Untuk menambahkan data ke semua kolom table dapat menuliskan semua kolom tabel dalam penulisan INSERT query.
- Misal memasukkan data pada table staf.

nip	nama	sex	posisi	tgl_masuk	gaji
...	...	...	...	...	...

- Penulisan Query:

---

```
INSERT INTO staf (nip,nama,sex,posisi,tgl_masuk,gaji)
VALUES ('A001','Dani','L','Asisten','2014-02-01',3800000);
```

---

## Menambahkan Data ke Semua Kolom Table (2)

- Untuk menambahkan data ke semua kolom tabel juga dapat dituliskan tanpa *columnList*.

---

```
INSERT INTO staf  
VALUES ('A001', 'Dani', 'L', 'Asisten', '2014-02-01', 3800000);
```

---



## Menambahkan Data ke Kolom Tertentu (1)

- Untuk memasukkan data pada kolom tertentu, kita harus menentukan nama kolom sebagai tempat untuk memasukkan data.
- Sebagai contoh, memasukkan data pada kolom nip, nama, dan posisi

---

```
INSERT INTO staf (nip,nama,posisi)  
VALUES ('A001', 'Dani', 'Asisten');
```

---

## Menambahkan Data ke Kolom Tertentu (2)

- Cara lain untuk menambahkan data ke kolom tertentu dilakukan dengan cara memasukkan nilai **NULL** ke kolom yang dikosongi.

---

```
INSERT INTO staf  
VALUES ('A001', 'Dani', NULL, 'Asisten', NULL, NULL) ;
```

---



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# UPDATE

# Mengubah Data pada Database

- Perintah untuk modifikasi atau mengubah data dalam tabel menggunakan UPDATE
- Dalam kondisi tertentu, kita dapat mengubah nilai dalam tupel tertentu tanpa harus mengubah semua nilai tupel dalam tabel.
- Penulisan Query:

---

**UPDATE** *TableName*

**SET** *columnName1* = *dataValue1*, *columnName2* = *dataValue2*...

**WHERE** *Condition*

---

- Klausa SET menentukan list nama kolom (bisa 1 atau lebih) yang akan diubah
- Klausa WHERE bersifat **opsional**, jika diperlukan maka update nilai harus memenuhi kondisi dari baris yang akan diubah.

## Mengubah semua Nilai (Baris)

nip	nama	sex	posisi	tgl_masuk	gaji
A001	Dani	L	Asisten	2014-02-01	3800000
M001	Slamet	L	Manajer	2004-01-01	7500000
S001	Siti	P	Supervisor	1997-07-10	5000000

- Misal, pada tabel **staf** diatas, semua gaji staf ditambahkan 500.000
- Penulisan UPDATE query dapat dilakukan **tanpa** menyertakan klausa WHERE

---

```
UPDATE staf  
SET gaji = gaji + 500000;
```

---

## Mengubah Nilai (baris) tertentu

nip	nama	sex	posisi	tgl_masuk	gaji
A001	Dani	L	Asisten	2014-02-01	3800000
M001	Slamet	L	Manajer	2004-01-01	7500000
S001	Siti	P	Supervisor	1997-07-10	5000000

- Misal, menambahkan gaji sebesar 1.000.000 khusus kepada staf dengan posisi Manajer
- penulisan UPDATE query adalah sebagai berikut

---

```
UPDATE staf
SET gaji = gaji + 1000000;
WHERE posisi = 'Manajer';
```

---

## Mengubah Nilai pada Beberapa Kolom (*multiple column*)

nip	nama	sex	posisi	tgl_masuk	gaji
A001	Dani	L	Asisten	2014-02-01	3800000
M001	Slamet	L	Manajer	2004-01-01	7500000
S001	Siti	P	Supervisor	1997-07-10	5000000

- Misal, Dani promosi ke posisi='Supervisor' dan gajinya menjadi 4.500.000
- penulisan UPDATE query adalah sebagai berikut:

---

```
UPDATE staf
SET posisi='Supervisor', gaji = 4500000;
WHERE nip = 'A001';
```

---



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# DELETE



# Menghapus dari Database

- Perintah menghapus data dari database menggunakan perintah DELETE
- Perintah DELETE memungkinkan baris untuk dihapus dari table.
- Penulisan Query:

---

```
DELETE FROM TableName  
WHERE Condition
```

---

- Kondisi bersifat opsional, jika dihilangkan maka semua baris dari tabel akan dihapus
- Jika digunakan, maka hanya baris tertentu saja yang memenuhi kondisi yang akan dihapus.

## Menghapus semua Nilai Tabel

- Misal, menghapus semua nilai pada table staf.
- Penulisan DELETE query sebagai berikut:

```
DELETE FROM staf;
```

- Untuk menghapus semua nilai pada tabel staf **tanpa** menyertakan klausa WHERE.
- Pada kasus ini, perintah DELETE **hanya** menghapus baris tabel saja dan hanya meninggalkan definisi tabel. Jadi tabel masih bisa diisi data nantinya.

## Menghapus Baris Tabel tertentu

nip	nama	sex	posisi	tgl_masuk	gaji
A001	Dani	L	Asisten	2014-02-01	3800000
M001	Slamet	L	Manajer	2004-01-01	7500000
S001	Siti	P	Supervisor	1997-07-10	5000000

- Menghapus baris tabel tertentu disesuaikan dengan baris yang akan dihapus.
- Misalnya, menghapus nama Dani dari tabel. Perintah DELETE query menyertakan klausa WHERE untuk menyesuaikan baris yang sesuai dengan nama Dani, misalnya dari NIP.

---

```
DELETE FROM staf  
WHERE nip='A001';
```

---

# 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?