



# Hari 3 (Rabu) – SQL Lanjutan: UPDATE, DELETE, RELASI

---



## Tujuan Pembelajaran

- Memahami konsep relasi antar tabel (foreign key)
  - Menguasai perintah UPDATE dan DELETE untuk mengubah dan menghapus data
  - Menggunakan JOIN untuk menggabungkan data dari beberapa tabel
- 

## 1. Relasi Antar Tabel (Foreign Key)

Relasi antar tabel memungkinkan data di satu tabel terhubung dengan data di tabel lain. Foreign key adalah kolom yang mengacu ke primary key di tabel lain.

Analogi:

Bayangkan tabel **orders** seperti nota pembelian, dan tabel **users** seperti daftar pelanggan. Foreign key di **orders** adalah "siapa" yang melakukan pembelian (mengacu ke user).

Contoh:

```
CREATE TABLE users (  
  id SERIAL PRIMARY KEY,  
  nama VARCHAR(100)  
);  
  
CREATE TABLE orders (  
  id SERIAL PRIMARY KEY,  
  user_id INT REFERENCES users(id),  
  tanggal TIMESTAMP  
);
```

---

## 2. Perintah UPDATE dan DELETE

### a. UPDATE

Digunakan untuk mengubah data pada tabel.

```
UPDATE products SET harga = 15000 WHERE id = 1;
```

### b. DELETE

Digunakan untuk menghapus data dari tabel.

```
DELETE FROM products WHERE id = 2;
```

---

### 3. JOIN Antar Tabel

JOIN digunakan untuk mengambil data dari beberapa tabel sekaligus berdasarkan relasi.

#### a. INNER JOIN

Mengambil data yang cocok di kedua tabel.

```
SELECT orders.id, users.nama FROM orders  
INNER JOIN users ON orders.user_id = users.id;
```

#### b. LEFT JOIN

Mengambil semua data dari tabel kiri, meski tidak ada pasangan di tabel kanan.

```
SELECT orders.id, users.nama FROM orders  
LEFT JOIN users ON orders.user_id = users.id;
```

---

### 4. Studi Kasus: Update & Relasi Data

1. Buat tabel `orders` dan `order_items` dengan relasi ke `users` dan `products`
2. Update harga produk tertentu
3. Hapus produk tertentu
4. JOIN untuk menampilkan pesanan beserta nama user dan produk

---

### 5. Praktik Mandiri (8 Jam)

- Buat tabel `orders` dan `order_items` dengan foreign key yang tepat
- Praktikkan relasi antar tabel dengan data dummy
- Coba JOIN untuk mengambil data pesanan dengan nama user dan produk
- Simulasi soft delete dengan kolom `is_deleted` (bukan benar-benar menghapus data, tapi menandai data tidak aktif)

#### Contoh Soft Delete

```
ALTER TABLE products ADD COLUMN is_deleted BOOLEAN DEFAULT FALSE;  
UPDATE products SET is_deleted = TRUE WHERE id = 3;  
SELECT * FROM products WHERE is_deleted = FALSE;
```



## Tips untuk Pemula

- Selalu backup data sebelum melakukan DELETE
- Gunakan soft delete untuk menjaga histori data
- Visualisasikan relasi tabel dengan diagram agar lebih mudah dipahami