

# SQL

Memahami Dasar-dasar Bahasa Query SQL

# SQL

- SQL (Structured Query Language) adalah bahasa yang digunakan untuk mengelola dan mengakses basis data.
- SQL memiliki berbagai jenis perintah yang digunakan untuk berbagai tujuan.
- Perintah SQL dapat dibagi menjadi lima kategori utama:
  - Data Definition Language (DDL)
  - Data Query Language (DQL)
  - Data Manipulation Language (DML)
  - Data Control Language (DCL)
  - Transaction Control Language (TCL)

# SQL: Data Definition Language (DDL)

- DDL (Data Definition Language) digunakan untuk mendefinisikan struktur basis data.
- Contoh perintah DDL:
  - `CREATE TABLE` : Membuat tabel baru.
  - `ALTER TABLE` : Mengubah struktur tabel.
  - `DROP TABLE` : Menghapus tabel.
- Digunakan untuk mengatur struktur basis data.

# Contoh DDL

```
-- Membuat Tabel
CREATE TABLE users (
  id INT PRIMARY KEY,
  name VARCHAR(50),
  email VARCHAR(50),
  password VARCHAR(50)
);

-- Mengubah Tabel
ALTER TABLE users ADD registered_at DATE;

-- Menghapus Tabel
DROP TABLE users;
```

# SQL: Data Query Language (DQL)

- DQL (Data Query Language) digunakan untuk mengambil data dari basis data.
- Contoh perintah DQL:
  - `SELECT` : Mengambil data dari tabel.
  - `FROM` : Menentukan sumber data.
  - `WHERE` : Mengatur kondisi untuk pemilihan data.
- Digunakan untuk pengambilan data.

# Contoh DQL

```
-- Mengambil Data
SELECT name, email
FROM users;
WHERE email = 'john@example.com';

-- Menggabungkan Data
SELECT users.name, users.email
FROM users
JOIN orders ON users.id = orders.user_id;
```

# SQL: Data Manipulation Language (DML)

- DML (Data Manipulation Language) digunakan untuk mengelola data dalam tabel.
- Contoh perintah DML:
  - `INSERT INTO` : Menambahkan data ke tabel.
  - `UPDATE` : Memperbarui data dalam tabel.
  - `DELETE` : Menghapus data dari tabel.
- Digunakan untuk pengelolaan data dalam tabel.

# Contoh DML

-- Menambahkan Data

```
INSERT INTO users (name, email, password)
VALUES ('John Doe', 'john@example.com', 'password');
```

-- Memperbarui Data

```
UPDATE users
SET name = 'John Doe', email = 'john@example.net'
WHERE id = 1;
```

-- Menghapus Data

```
DELETE FROM users
WHERE id = 1;
```



# SQL: Data Control Language (DCL)

- DCL (Data Control Language) digunakan untuk mengendalikan hak akses pengguna terhadap basis data.
- Contoh perintah DCL:
  - **GRANT** : Memberikan hak akses.
  - **REVOKE** : Mencabut hak akses.

```
-- Memberikan Hak Akses
GRANT SELECT, INSERT ON users TO jane;

-- Mencabut Hak Akses
REVOKE SELECT, INSERT ON users FROM jane;
```

# SQL: Transaction Control Language (TCL)

- TCL (Transaction Control Language) digunakan untuk mengontrol transaction dalam basis data.
- Contoh perintah TCL:
  - `COMMIT` : Menyimpan perubahan dalam transaction.
  - `ROLLBACK` : Membatalkan transaction.
- Digunakan untuk pengendalian transaction.

# Database Transaction

- Transaction adalah sekumpulan perintah SQL yang dijalankan sebagai satu kesatuan.
- Penting untuk memastikan bahwa semua perintah dalam transaction berhasil atau tidak ada yang berhasil sama sekali.

## COMMIT

- Perintah `COMMIT` digunakan untuk menyimpan transaction yang telah berhasil.
- Semua perubahan yang dilakukan dalam transaction disimpan secara persisten.

```
BEGIN; -- Mulai transaction  
  
-- Perintah-perintah dalam transaction  
  
COMMIT; -- Simpan perubahan
```

## ROLLBACK

- Perintah `ROLLBACK` digunakan untuk membatalkan transaction jika terjadi kesalahan.
- Semua perubahan yang dilakukan dalam transaction dibatalkan.

```
BEGIN; -- Mulai transaction  
  
-- Perintah-perintah dalam transaction  
  
ROLLBACK; -- Batalkan transaction
```

## Contoh TCL

```
BEGIN; -- Mulai transaction

UPDATE balances SET balance = balance - 100 WHERE user_id = 1;
INSERT INTO balance_logs (user_id, amount) VALUES ('John', -100);

COMMIT; -- Simpan perubahan jika berhasil
-- atau ROLLBACK; -- Batalkan transaction jika terjadi kesalahan
```

# Latihan

## Latihan 1: Kueri Dasar

Tulis kueri SQL untuk mengambil nama-nama provinsi di Indonesia dari tabel 'provinces'.

## Latihan 2: Menggabungkan Tabel

Tulis kueri SQL untuk mengambil nama-nama kabupaten/kota di dalam provinsi tertentu, beserta nama provinsi tersebut. Gunakan operasi `join` untuk melakukannya.

## Latihan 3: Memfilter dan Menghitung

Tulis kueri SQL untuk menghitung jumlah kabupaten/kota dari tiap-tiap provinsi. Gunakan operasi `join` dan `count` untuk melakukannya.

# Latihan

## Latihan 4: Membuat Database

Buat basis data baru dengan nama `bookstore` dengan storage engine `InnoDB`

## Latihan 5: Membuat Tabel

Buat dua tabel, `authors` dan `books` dengan kolom-kolom yang sesuai untuk masing-masing tabel. Pastikan bahwa tabel `authors` memiliki kunci utama (primary key), dan tabel `books` memiliki hubungan kunci asing (foreign key) dengan tabel `authors`.

## Latihan 6: Memasukkan Data

Masukkan setidaknya tiga catatan ke dalam tabel `authors` dan lima catatan ke dalam tabel `books`. Pastikan bahwa buku-buku terkait dengan penulis-penulis di tabel `authors`.



# Latihan

## Latihan 7: Memperbarui Data

Ubah nama salah satu penulis di tabel `authors` dan perbarui judul salah satu buku di tabel `books`.

## Latihan 8: Menghapus Data

Hapus salah satu penulis dari tabel `authors` dan salah satu buku dari tabel `books`. Pastikan bahwa penghapusan berjalan dengan benar jika diperlukan.

## Latihan 9: Mengambil Data

Tulis kueri untuk mengambil nama semua penulis dan judul-judul buku mereka.