

**TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL X
DATA STORAGE (BAGIAN I)**



Disusun Oleh :

Yogi Hafidh Maulana / 2211104061

SE06-02

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

TUGAS PENDAHULUAN

SOAL

1. Jelaskan secara singkat fungsi SQLite dalam pengembangan aplikasi mobile!

SQLite adalah database relasional ringan yang sering digunakan dalam pengembangan aplikasi mobile untuk menyimpan dan mengelola data secara lokal di perangkat. Fungsi utamanya meliputi:

- Penyimpanan Data Lokal: SQLite memungkinkan aplikasi menyimpan data seperti pengaturan pengguna, cache, atau data aplikasi lainnya tanpa memerlukan koneksi internet.
- Relasional dan Berbasis SQL: SQLite mendukung operasi SQL seperti CRUD (Create, Read, Update, Delete), sehingga pengelolaan data menjadi lebih terstruktur dan mudah.
- Kinerja Cepat: Dengan ukurannya yang kecil dan pengoperasiannya yang efisien, SQLite cocok untuk aplikasi mobile dengan kebutuhan data yang tidak terlalu kompleks.
- Tanpa Server: SQLite tidak memerlukan server terpisah, sehingga memudahkan integrasi ke dalam aplikasi.
- Portabilitas Tinggi: Database SQLite disimpan sebagai file tunggal, sehingga mudah untuk dibawa atau dibagikan antar perangkat.

Fitur ini membuat SQLite menjadi pilihan populer untuk pengelolaan data lokal dalam aplikasi mobile.

2. Apa saja yang dimaksud dengan operasi CRUD? Berikan penjelasan singkat untuk masing-masing operasi!

Operasi CRUD adalah serangkaian operasi dasar yang digunakan untuk mengelola data dalam basis data. Berikut penjelasan masing-masing operasi:

- Create:
Operasi untuk menambahkan data baru ke dalam basis data. Misalnya, menyimpan data pengguna baru atau menambahkan produk baru ke dalam katalog.
- Read:
Operasi untuk membaca atau mengambil data dari basis data. Contohnya adalah menampilkan daftar produk atau informasi detail pengguna berdasarkan ID.
- Update:
Operasi untuk memperbarui data yang sudah ada di dalam basis data. Contohnya adalah mengubah informasi alamat pengguna atau memperbarui stok produk.

- Delete:
Operasi untuk menghapus data dari basis data. Misalnya, menghapus akun pengguna atau menghapus entri produk yang tidak lagi tersedia.

CRUD adalah konsep fundamental dalam pengelolaan data, baik untuk database relasional seperti SQLite maupun non-relasional.

3. Tuliskan kode SQL untuk membuat tabel bernama *users* dengan kolom berikut :

- id (integer, primary key, auto increment)
- name (text)
- email (text)
- createdAt (timestamp, default value adalah waktu sekarang)

```
CREATE TABLE users (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    name TEXT NOT NULL,  
    email TEXT NOT NULL,  
    createdAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

- id INTEGER PRIMARY KEY AUTOINCREMENT:
Kolom id digunakan sebagai kunci utama dengan tipe data INTEGER dan akan bertambah otomatis setiap kali data baru ditambahkan.
- name TEXT NOT NULL:
Kolom name bertipe data TEXT dan harus diisi (tidak boleh NULL).
- email TEXT NOT NULL:
Kolom email juga bertipe data TEXT dan bersifat wajib diisi.
- createdAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP:
Kolom createdAt akan secara otomatis menyimpan waktu saat data ditambahkan ke tabel. Tipe datanya adalah TIMESTAMP, dan nilai default-nya adalah waktu sekarang (CURRENT_TIMESTAMP).

4. Sebutkan langkah-langkah utama untuk menggunakan plugin sqflite di dalam Flutter!

- Tambahkan Dependensi

Tambahkan plugin sqflite dan path ke dalam file pubspec.yaml:

```
dependencies:  
  sqflite: ^2.0.0  
  path: ^1.8.0
```

Lalu jalankan perintah: `flutter pub get`

- Impor Plugin

Di file Dart tempat Anda ingin menggunakan SQLite, impor paket:

```
import 'package:sqflite/sqflite.dart';  
import 'package:path/path.dart';
```

- Buat dan Buka Database

Gunakan fungsi `openDatabase` untuk membuat atau membuka database:

```
Future<Database> initDatabase() async {  
  final databasePath = await getDatabasesPath();  
  final path = join(databasePath, 'my_database.db');  
  
  return openDatabase(  
    path,  
    onCreate: (db, version) {  
      return db.execute(  
        'CREATE TABLE users(id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT, email TEXT)',  
      );  
    },  
    version: 1,  
  );  
}
```

- Lakukan Operasi CRUD

Gunakan database yang sudah dibuat untuk melakukan operasi CRUD.

- Kelola Database dengan State

Pastikan pengelolaan database diintegrasikan dengan state management, seperti `setState` atau `Provider`, untuk memperbarui UI saat data berubah.

- Tutup Database (Opsional)

Tutup database saat tidak digunakan untuk menghindari kebocoran memori:

```
Future<void> closeDatabase(Database db) async {  
    await db.close();  
}
```

5. Lengkapi kode berikut untuk membaca semua data dari tabel *users* menggunakan sqlite.

```
static Future<List<Map<String, dynamic>>> getUsers() async {  
    final db = await SQLHelper.db();  
    return db.query(______);  
}
```

```
static Future<List<Map<String, dynamic>>> getUsers() async {  
    final db = await SQLHelper.db();  
    return db.query('users');  
}
```