



Pertemuan 11

PENYIMPANAN DATA

Kristina Annatasia Br Sitepu, M.Kom





PEMBAHASAN

Pengantar Materi

Shared Preference

SQLite

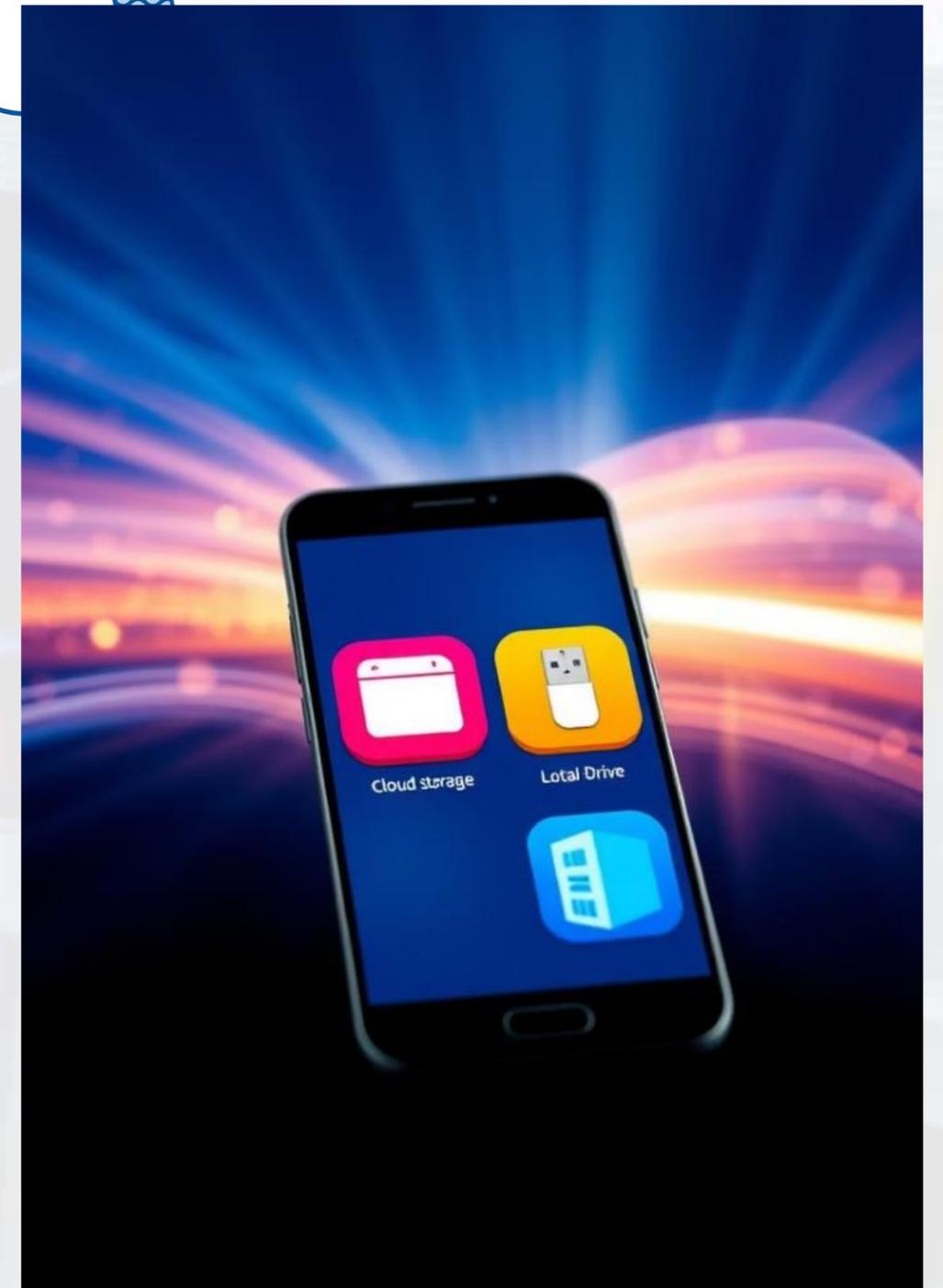
MySQL



Pendahuluan

Penyimpanan data sangat penting dalam aplikasi Android.

Ini memungkinkan aplikasi menyimpan informasi secara permanen atau sementara. Pemilihan media penyimpanan yang tepat mempengaruhi kinerja dan pengalaman pengguna.



Shared Preferences

Deskripsi

Menyimpan data dalam bentuk pasangan kunci-nilai. Cocok untuk preferensi pengguna.

Kelebihan

Mudah digunakan untuk data sederhana. Ringan dan cepat diakses.

Kekurangan

Hanya cocok untuk data sederhana. Tidak untuk data kompleks.

← Settings

ON/OFF -OFF

ON/OFF

JV/SEFF

ONSOFF

SQLite Database

1

Deskripsi

Database lokal untuk menyimpan data terstruktur dalam bentuk tabel.

2

Kelebihan

Mendukung penyimpanan data besar dan terstruktur.
Mendukung query SQL kompleks.

3

Kekurangan

Pengelolaan lebih rumit dibanding Shared Preferences. Tidak cocok untuk data tidak terstruktur.



MySQL Database

Deskripsi

Database server berbasis cloud untuk penyimpanan data terpusat.

Kelebihan

Cocok untuk data besar dan terdistribusi. Mendukung akses dari banyak perangkat.

Kekurangan

Memerlukan koneksi internet. Tidak cocok untuk aplikasi offline.

Internal dan External Storage

Internal Storage

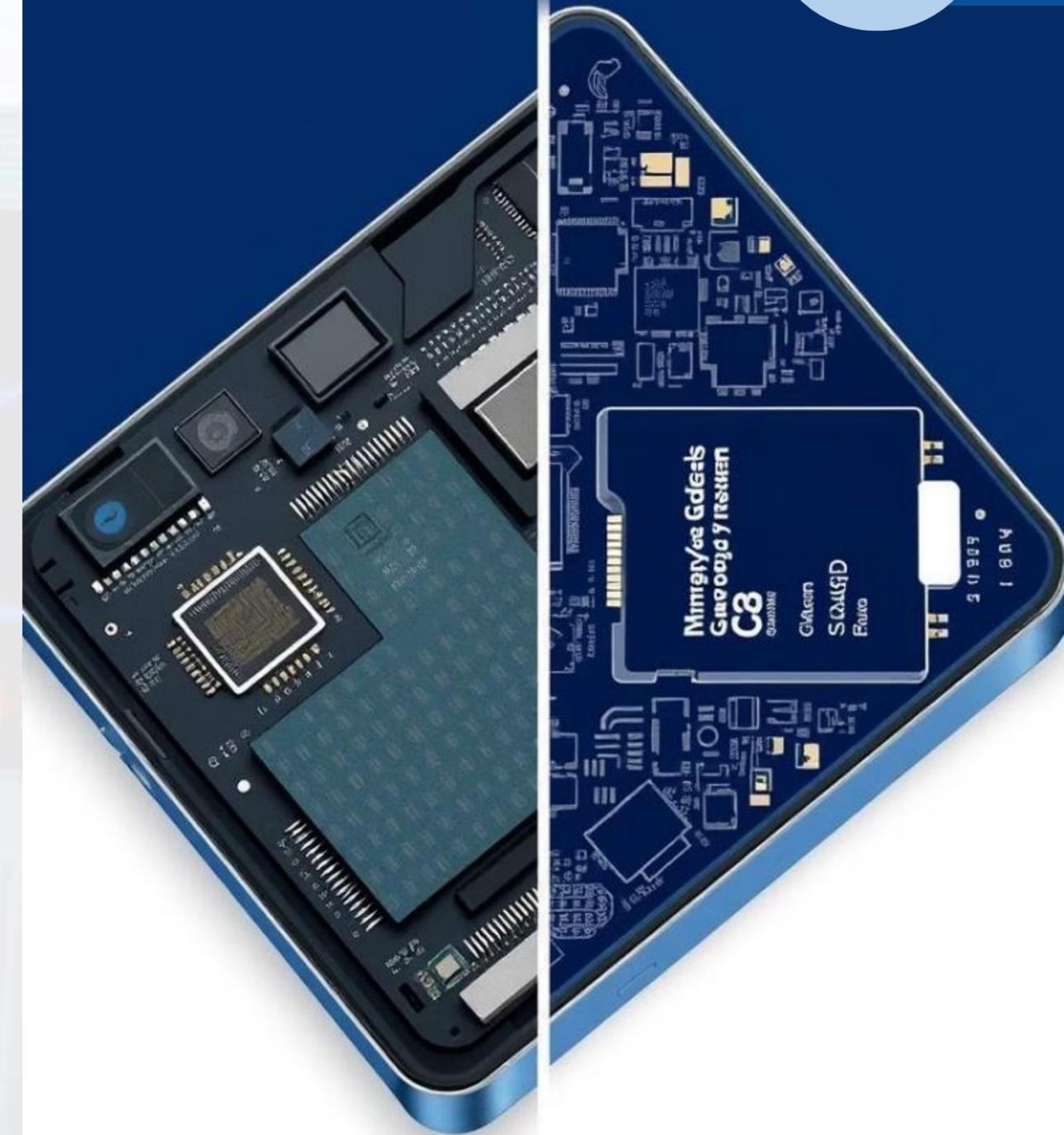
Data aman, tidak bisa diakses aplikasi lain

Tidak perlu izin akses

External Storage

Kapasitas lebih besar, dapat diakses aplikasi lain

Membutuhkan izin akses dari pengguna



Memilih Media Penyimpanan yang Tepat

1

Shared Preferences

Untuk pengaturan pengguna atau data sederhana

2

SQLite Database

Untuk data terstruktur dan query kompleks

3

MySQL Database

Untuk data besar dan terdistribusi

4

External Storage

Untuk data besar seperti gambar atau video

Implementasi Shared Preferences

- 1** — Menyimpan Data
Gunakan `SharedPreferences.Editor` untuk menyimpan data kunci-nilai.
- 2** — Mengambil Data
Gunakan metode `get` untuk mengambil nilai berdasarkan kunci.
- 3** — Menghapus Data
Gunakan metode `remove` atau `clear` untuk menghapus data.

Implementasi SQLite Database

```

1  SQLiteConnection;
2  }
3  onCreate() {
4      super.onCreate();
5      createDatabase();
6      createTable();
7      insertData();
8      getData();
9  }
10 private void createDatabase() {
11     SQLiteDatabase db = getWritableDatabase();
12     db.execSQL("CREATE DATABASE IF NOT EXISTS mydb");
13 }
14 private void createTable() {
15     SQLiteDatabase db = getWritableDatabase();
16     db.execSQL("CREATE TABLE IF NOT EXISTS users (id INTEGER PRIMARY KEY, name TEXT, age INTEGER)");
17 }
18 private void insertData() {
19     SQLiteDatabase db = getWritableDatabase();
20     ContentValues values = new ContentValues();
21     values.put("name", "John");
22     values.put("age", 25);
23     db.insert("users", null, values);
24 }
25 private void getData() {
26     SQLiteDatabase db = getWritableDatabase();
27     Cursor cursor = db.query("users", null, null, null, null, null, null);
28     while (cursor.moveToNext()) {
29         int id = cursor.getInt(0);
30         String name = cursor.getString(1);
31         int age = cursor.getInt(2);
32     }
33 }

```

Table Description		
Interlagier	Pengage	
Archiery	Froy	
Dieer blays	Faalty	
Buis	Color andperctions	
Emriality	Syepring	
Proficrations	Start Lasral	Verd
Prrojoy	Statemancs	
Eactlars	Tegeriare agecuiry	
Audirabes		
Fabt	Tor pigge	
Projject		



Membuat Database
Gunakan SQLiteOpenHelper untuk membuat dan mengelola database.



Menambah Data
Gunakan metode insert untuk menambahkan data ke tabel.

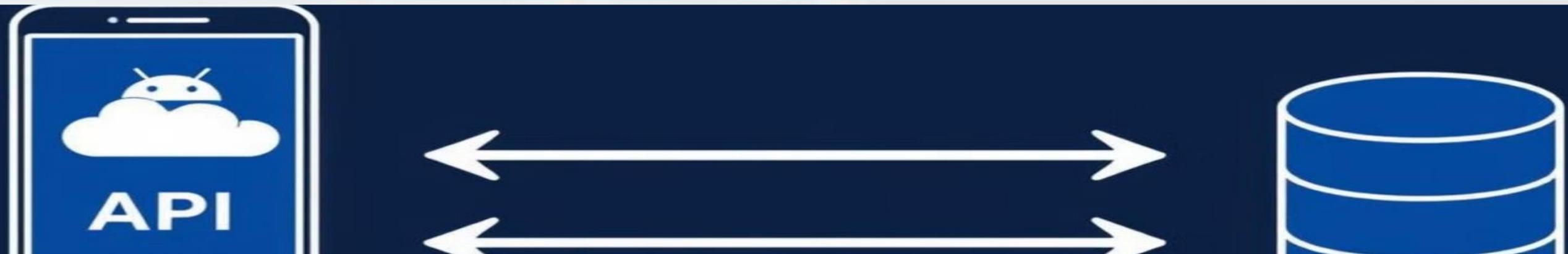


Mengambil Data
Gunakan query SQL untuk mengambil data dari database.



Memperbarui Data
Gunakan metode update untuk memperbarui data yang ada.

Implementasi MySQL Database



- 1** Persiapan Server
Siapkan server backend untuk mengelola query SQL.
- 2** API Web
Buat API web untuk komunikasi antara aplikasi Android dan server.
- 3** Integrasi dengan Android
Gunakan library seperti Retrofit untuk melakukan HTTP request ke server.

KESIMPULAN

Shared Preferences

Ideal untuk data sederhana seperti pengaturan pengguna.

SQLite

Cocok untuk data terstruktur dengan query kompleks.

MySQL

Pilihan terbaik untuk data terpusat dan dapat diakses secara jaringan.

Oleh : Kristina Annatasia Br Sitepu, M.Kom



se k i a n

TERIMA KASIH



Oleh : Kristina Annatasia Br Sitepu, M.Kom

