# PEMROGRAMAN PERANGKAT BERGERAK

# UNGUIDED MODUL X

**Disusun Oleh :**

**Ilham Lii Assidaq**

**2311104068**

**Asisten Praktikum :**

**Yoga Eka Pratama**

**Zulfa Mustafa Akhyar Iswahyudi**

**Dosen Pengampu :**

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

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2025**

## TUGAS PENDAHULUAN

1. SOAL

(Soal) Buatlah sebuah project aplikasi Flutter dengan SQLite untuk menyimpan data biodata mahasiswa yang terdiri dari nama, NIM, domisili, dan hobi. Data yang dimasukkan melalui form akan ditampilkan dalam daftar di halaman utama.

Alur Aplikasi:

Form Input: Buat form input untuk menambahkan biodata mahasiswa, dengan kolom:

Nama

Nim

Alamat

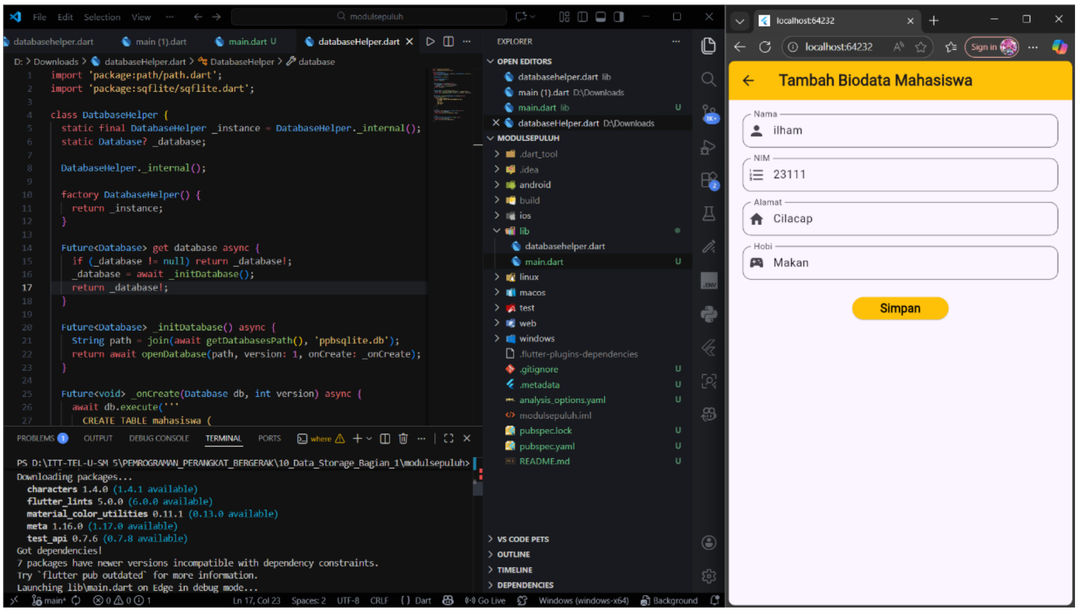
Hobi

Tampilkan Daftar Mahasiswa: Setelah data berhasil ditambahkan, tampilkan daftar semua data mahasiswa yang sudah disimpan di halaman utama.

Implementasikan fitur Create (untuk menyimpan data mahasiswa) dan Read (untuk menampilkan daftar mahasiswa yang sudah disimpan).

Contoh output:

2. SOURCE CODE



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

class DatabaseHelper {
  static final DatabaseHelper _instance = DatabaseHelper._internal();
  static Database? _database;

  DatabaseHelper._internal();

  factory DatabaseHelper() {
    return _instance;
  }

  Future<Database> get database async {
    if (_database != null) return _database!;
    _database = await _initDatabase();
    return _database!;
  }

  Future<Database> _initDatabase() async {
    String path = join(await getDatabasesPath(), 'ppbsqlite.db');
    return await openDatabase(path, version: 1, onCreate: _onCreate);
  }

  Future<void> _onCreate(Database db, int version) async {
    await db.execute('''
      CREATE TABLE mahasiswa (
```

```
PS D:\ITT-TEL-U-SM 5\PEMROGRAMAN_PERANGKAT_BERGERAK\10_Data_Storage_Bagian_1\modulsepuluh>
Downloading packages...
  characters 1.4.0 (1.4.1 available)
  flutter_lints 5.0.0 (6.0.0 available)
  material_color_utilities 0.11.1 (0.13.0 available)
  meta 1.16.0 (1.17.0 available)
  test_api 0.7.6 (0.7.8 available)
Got dependencies!
7 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
Launching lib\main.dart on Edge in debug mode...
```

Tambah Biodata Mahasiswa

Nama: ilham

NIM: 23111

Alamat: Cilacap

Hobi: Makan

Databasehelper.dart

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

class DatabaseHelper {
  static final DatabaseHelper _instance = DatabaseHelper._internal();
  static Database? _database;

  DatabaseHelper._internal();

  factory DatabaseHelper() {
    return _instance;
  }

  Future<Database> get database async {
    if (_database != null) return _database!;
    _database = await _initDatabase();
    return _database!;
  }

  Future<Database> _initDatabase() async {
    String path = join(await getDatabasesPath(), 'ppbsqlite.db');
    return await openDatabase(path, version: 1, onCreate: _onCreate);
  }

  Future<void> _onCreate(Database db, int version) async {
    await db.execute('''
      CREATE TABLE mahasiswa (
        id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
        nama TEXT,
        nim TEXT,
        alamat TEXT,
        hobi TEXT
      )
    ''');
  }

  // CREATE
  Future<int> create(Map<String, dynamic> data) async {
    final db = await database;
    return await db.insert('mahasiswa', data);
  }

  // READ
  Future<List<Map<String, dynamic>>> read() async {
    final db = await database;
    return await db.query('mahasiswa');
  }
}
```

Main.dart

```
import 'package:flutter/material.dart';
import 'databaseHelper.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      debugShowCheckedModeBanner: false,
      home: HomePage(),
    );
  }
```

```
}

class HomePage extends StatefulWidget {
  const HomePage({super.key});

  @override
  State<HomePage> createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  late Future<List<Map<String, dynamic>>> datamahasiswa;

  @override
  void initState() {
    super.initState();
    datamahasiswa = DatabaseHelper().read();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const Color(0xfffaf5ff),

      appBar: AppBar(
        backgroundColor: Colors.amber,
        title: const Text(
          "SQLite Biodata Mahasiswa",
          style: TextStyle(fontWeight: FontWeight.bold),
        ),
        centerTitle: true,
      ),

      floatingActionButton: FloatingActionButton(
        backgroundColor: Colors.amber,
        shape: const CircleBorder(),
        child: const Icon(Icons.add, color: Colors.black),
        onPressed: () async {
          await Navigator.push(
            context,
            MaterialPageRoute(builder: (_) => const InputPage()),
          );
          setState(() {
            datamahasiswa = DatabaseHelper().read();
          });
        },
      ),

      body: FutureBuilder(
        future: datamahasiswa,
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return const Center(child: CircularProgressIndicator());
          }

          if (!snapshot.hasData || snapshot.data!.isEmpty) {
            return const Center(child: Text("Belum ada data"));
          }

          final data = snapshot.data!;

          return ListView.builder(
            padding: const EdgeInsets.all(15),
            itemCount: data.length,
            itemBuilder: (context, index) {
              final m = data[index];

              return Card(
                elevation: 3,
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
                child: Padding(
                  padding: const EdgeInsets.all(15),
                  child: Column(
                    crossAxisAlignment: CrossAxisAlignment.start,
                    children: [
                      Text(
                        m['nama'],
                        style: const TextStyle(
                          fontSize: 18,
                          fontWeight: FontWeight.bold,
                        ),
                      ),
                      const SizedBox(height: 5),
                      Text("NIM : ${m['nim']}"),
                      Text("Alamat : ${m['alamat']}"),
                      Text("Hobi : ${m['hobi']}"),
                    ],
                  ),
                ),
              );
            },
          );
        },
      ),
    );
  }
}

class InputPage extends StatefulWidget {
  const InputPage({super.key});
  @override
  State<InputPage> createState() => _InputPageState();
}

class _InputPageState extends State<InputPage> {
  final namaC = TextEditingController();
  final nimC = TextEditingController();
  final alamatC = TextEditingController();
  final hobiC = TextEditingController();

  InputDecoration inputStyle(String label, IconData icon) {
    return InputDecoration(
      labelText: label,
      prefixIcon: Icon(icon),
      border: OutlineInputBorder(borderRadius: BorderRadius.circular(12)),
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const Color(0xfffaf5ff),

      appBar: AppBar(
        backgroundColor: Colors.amber,
        title: const Text(
          "Tambah Biodata Mahasiswa",
          style: TextStyle(fontWeight: FontWeight.bold),
        ),
      ),

      body: Padding(
        padding: const EdgeInsets.all(20),
child: Column(
  children: [
    TextField(
      controller: namaC,
      decoration: inputStyle("Nama", Icons.person),
    ),
    const SizedBox(height: 15),

    TextField(
      controller: nimC,
      decoration: inputStyle("NIM", Icons.format_list_numbered),
    ),
    const SizedBox(height: 15),

    TextField(
      controller: alamatC,
      decoration: inputStyle("Alamat", Icons.home),
    ),
    const SizedBox(height: 15),

    TextField(
      controller: hobiC,
      decoration: inputStyle("Hobi", Icons.sports_esports),
    ),
    const SizedBox(height: 25),

    ElevatedButton(
      onPressed: () async {
        await DatabaseHelper().create({
          "nama": namaC.text,
          "nim": nimC.text,
          "alamat": alamatC.text,
          "hobi": hobiC.text,
        });
```

```
                Navigator.pop(context);
              },
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.amber,
                padding: const EdgeInsets.symmetric(
                  horizontal: 40,
                  vertical: 12,
                ),
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(30),
                ),
              ),
              child: const Text(
                "Simpan",
                style: TextStyle(
                  fontSize: 17,
                  color: Colors.black,
                  fontWeight: FontWeight.bold,
                ),
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```