CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF TECHNOLOGY AND ENGINEERING
CHANDUBHAI S. PATEL INSTITUTE OF TECHNOLOGY
DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

Subject : Mobile Application Development **Semester:** 5

Subject Code: AIML308 **Academic Year :**2025-26(ODD)

NAME : CHOKSI OM CHIRAGBHAI **ID: 23AIML010**

Practical 5

Problem Definition

You are building a mobile application where users navigate through multiple screens like login, dashboard, and profile. Build a Student Records App with CRUD operations using SQLite.

Supplementary Problems -	**Book management** or Expense tracker app

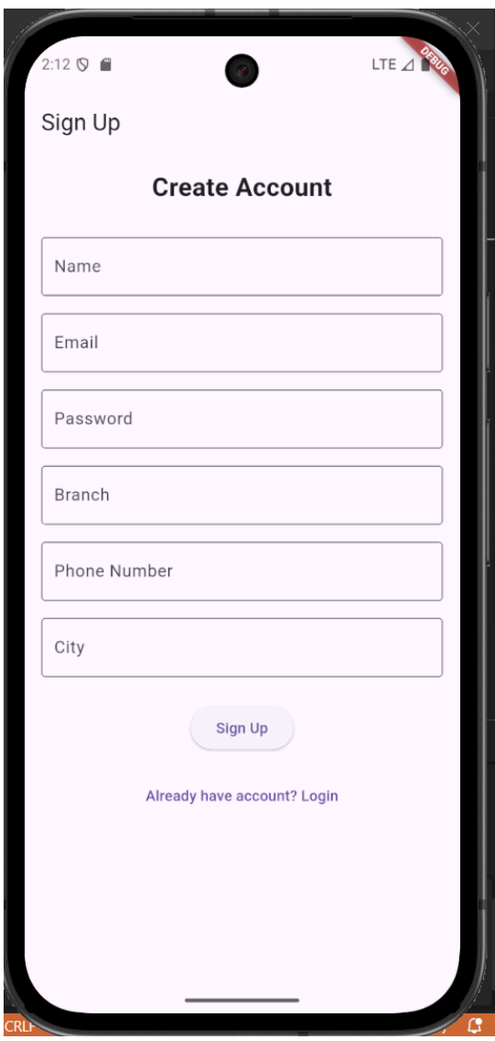
Technical Approach:

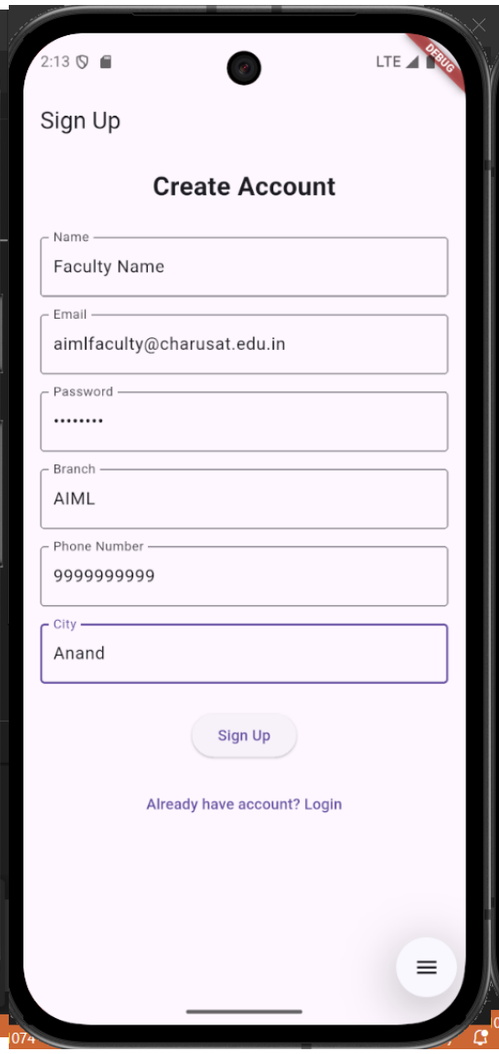
This Student Records App is built using Flutter and demonstrates local data persistence, CRUD operations, and basic authentication. The core logic uses the sqflite package for SQLite database management, shared_preferences for simple user data persistence, and state management via setState. Main widgets include StatefulWidget for dynamic screens, FutureBuilder for async database queries, ListView.builder for displaying student lists, and Form/TextFormField for input validation. The app's navigation is managed with named routes, and all CRUD/database logic is encapsulated in a singleton helper class.

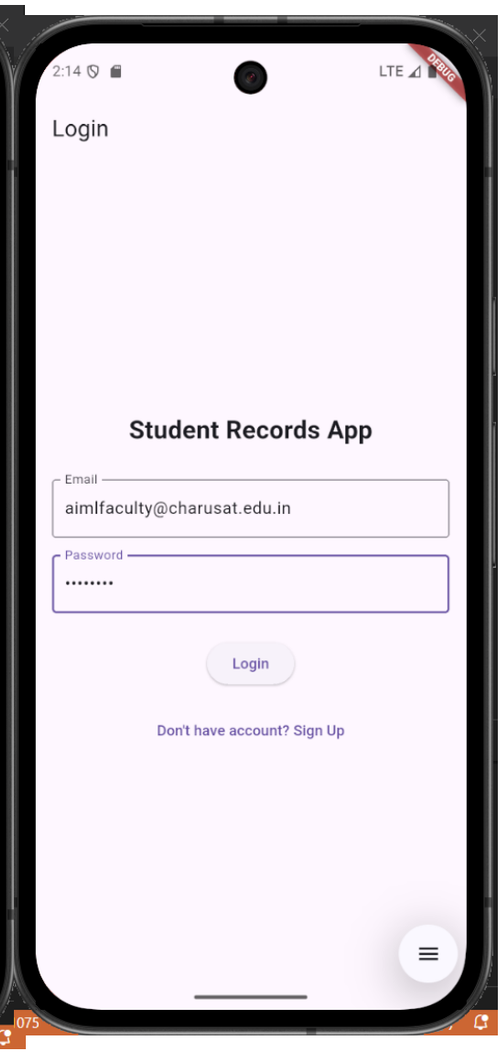
File Structure:

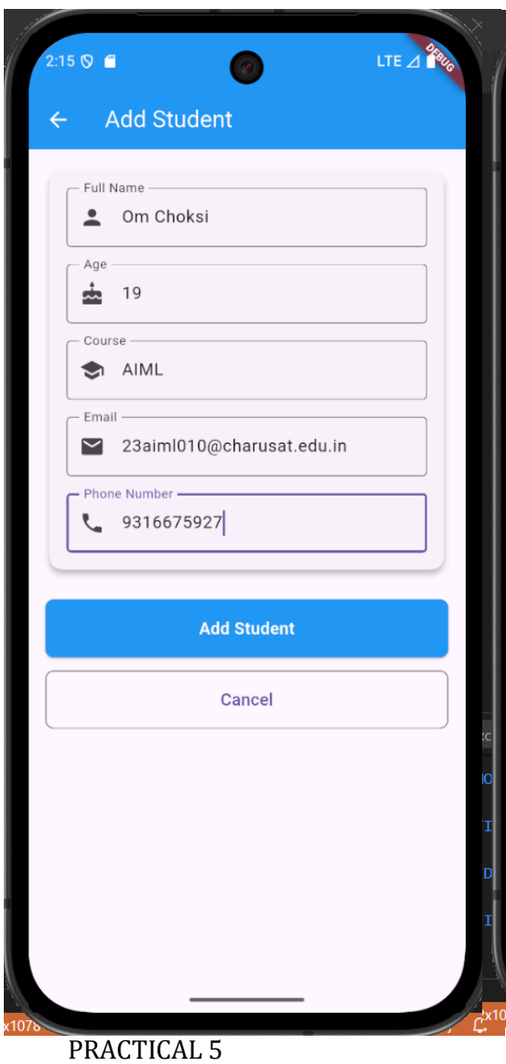
```
lib/
├─── main.dart - Entry point for the Flutter app, initializes the widget tree.
├─── database/
│   └─── database_helper.dart - Singleton class for SQLite CRUD operations on students and users.
├─── models/
│   ├─── student.dart - Student data model with properties and serialization.
│   └─── user_data.dart - User data model for authentication and profile info.
└─── screens/
    ├─── add_edit_student_screen.dart - Form screen for adding/editing students.
    ├─── login_screen.dart - Login screen with authentication.
    ├─── profile_screen.dart - User profile display/edit screen.
    ├─── signup_screen.dart - User registration screen.
    └─── student_list_screen.dart - List screen for viewing/editing/deleting students.
```

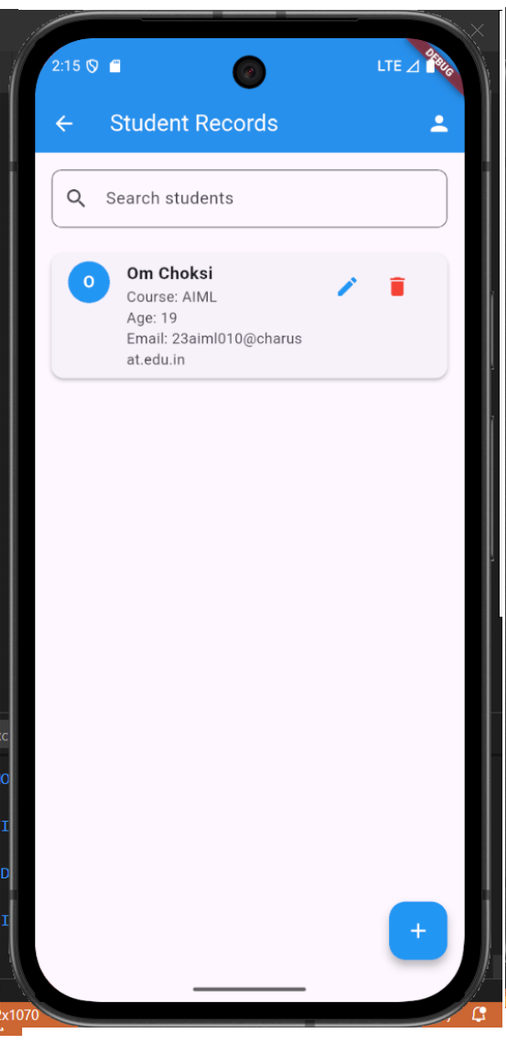
Screenshots:

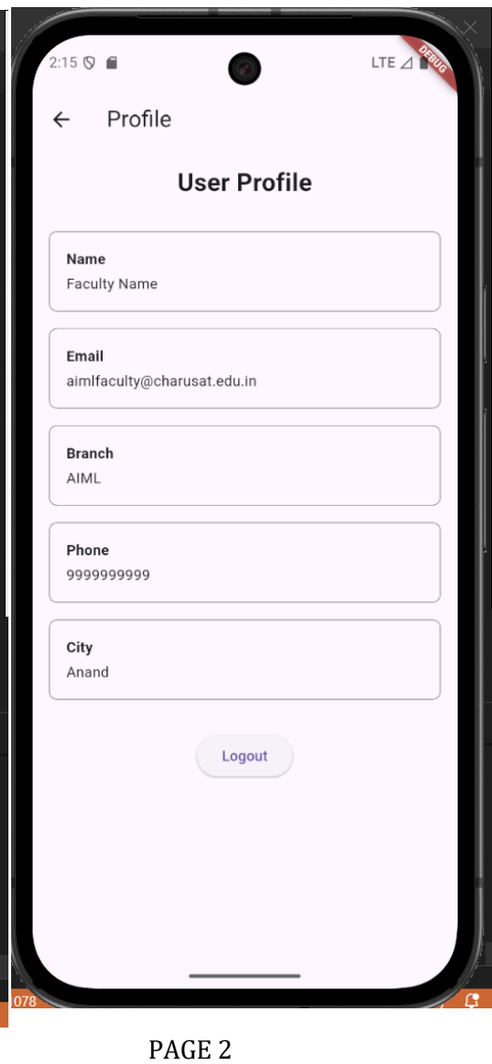


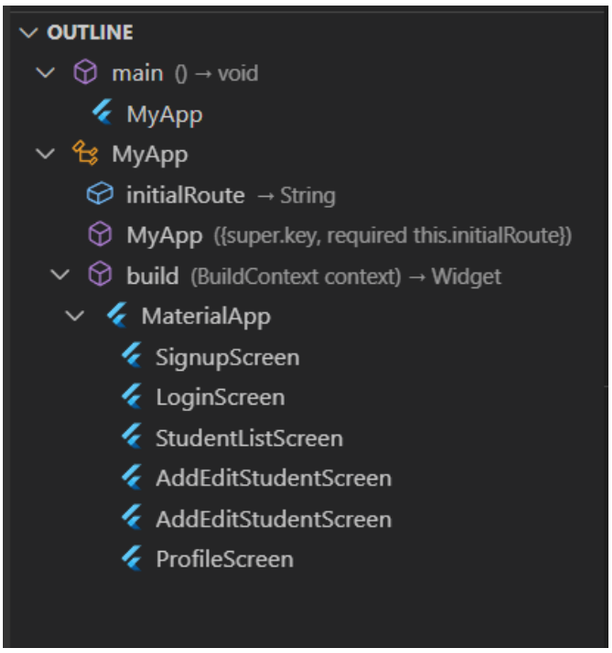


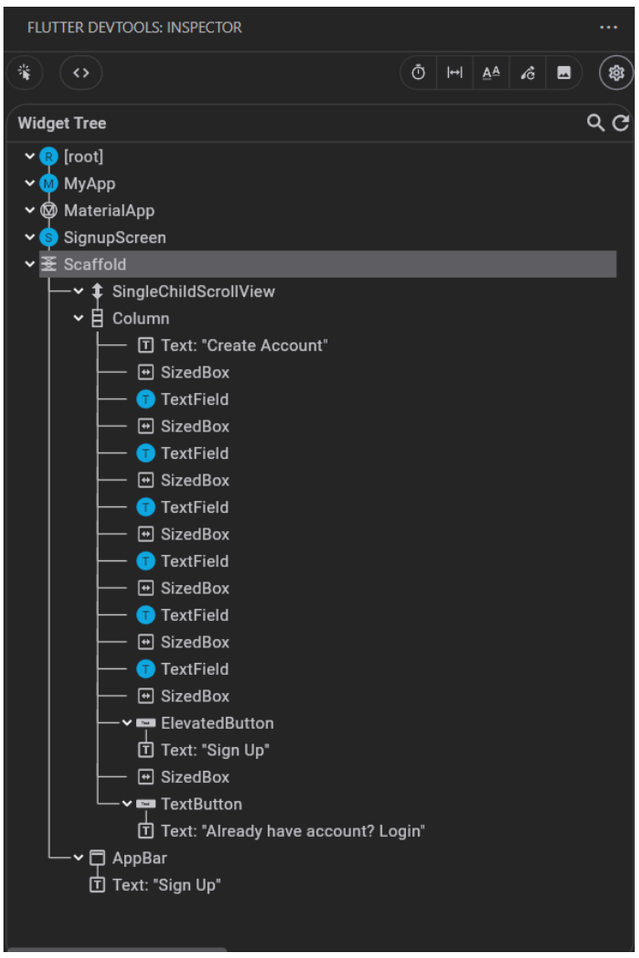












Key Questions:

1. How to store, update, retrieve data locally?
Ans: Data is stored, updated, and retrieved using the sqflite package, with all CRUD operations implemented in DatabaseHelper. Student records are inserted, updated, deleted, and queried from a local SQLite database table (students).

2. What is the role of FutureBuilder?
Ans: FutureBuilder is used to asynchronously fetch and display student records from the database, ensuring the UI updates automatically when data is loaded or changed.

3. How to connect to SQLite?
Ans: The app connects to SQLite using the sqflite package. The database is initialized in DatabaseHelper with openDatabase, and all queries are performed using async methods (insert, query, update, delete).

Key Skills to be Understand:

Database operations, CRUD logic

Implement local DB solutions