KRISHNA DINESH

RIT KOTTAYAM

INTERNSHIP REPORT

SOURCE CODE:

```
import 'dart:convert';

import 'package:flutter/material.dart';

import 'package:shared_preferences/shared_preferences.dart';


void main() {

  runApp(NoteApp());

}


class Note {

  final String title;

  final String content;


  Note({required this.title, required this.content});


  Map<String, String> toMap() => {

        'title': title,

        'content': content,

      };


  factory Note.fromMap(Map<String, dynamic> map) => Note(

        title: map['title'],

        content: map['content'],
      );
}

class NoteApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Notes App',
      home: NoteHomePage(),
      debugShowCheckedModeBanner: false,
    );
  }
}

class NoteHomePage extends StatefulWidget {
  @override
  _NoteHomePageState createState() => _NoteHomePageState();
}

class _NoteHomePageState extends State<NoteHomePage> {
  List<Note> _notes = [];
  late SharedPreferences _prefs;

  @override
  void initState() {
    super.initState();
    _loadNotes();
  }

  Future<void> _loadNotes() async {
    _prefs = await SharedPreferences.getInstance();
    final String? notesJson = _prefs.getString('notes');
    if (notesJson != null) {
      final List decoded = jsonDecode(notesJson);
      setState(() {
        _notes = decoded.map((note) => Note.fromMap(note)).toList();
      });
    }
  }

  Future<void> _saveNotes() async {
    final List<Map<String, String>> noteList =
        _notes.map((note) => note.toMap()).toList();
    await _prefs.setString('notes', jsonEncode(noteList));
  }

  void _addNote(String title, String content) {
    setState(() {
      _notes.add(Note(title: title, content: content));
    });
    _saveNotes();
  }
```

```
void _deleteNote(int index) {
  setState(() {
    _notes.removeAt(index);
  });
  _saveNotes();
}

void _showAddNoteDialog() {
  String title = '';
  String content = '';

  showDialog(
    context: context,
    builder: (context) => AlertDialog(
      title: Text('Add Note'),
      content: Column(
        mainAxisSize: MainAxisSize.min,
        children: [
          TextField(
            decoration: InputDecoration(labelText: 'Title'),
            onChanged: (val) => title = val,
          ),
          TextField(
            decoration: InputDecoration(labelText: 'Content'),
            onChanged: (val) => content = val,
     ),
    ],
   ),
   actions: [
    TextButton(
     onPressed: () {
      if (title.isNotEmpty || content.isNotEmpty) {
       _addNote(title, content);
      }
      Navigator.of(context).pop();
     },
     child: Text('Add'),
    ),
    TextButton(
     onPressed: () => Navigator.of(context).pop(),
     child: Text('Cancel'),
    ),
   ],
  ),
 );
}

@override
Widget build(BuildContext context) {
 return Scaffold(
  appBar: AppBar(title: Text('My Notes')),
      body: _notes.isEmpty
          ? Center(child: Text('No notes yet.'))
          : ListView.builder(
              itemCount: _notes.length,
              itemBuilder: (context, index) {
                final note = _notes[index];
                return ListTile(
                  title: Text(note.title),
                  subtitle: Text(note.content),
                  trailing: IconButton(
                    icon: Icon(Icons.delete),
                    onPressed: () => _deleteNote(index),
                  ),
                );
              },
            ),
      floatingActionButton: FloatingActionButton(
        onPressed: _showAddNoteDialog,
        child: Icon(Icons.add),
      ),
    );
  }
}
```

Dependency in pubspec.yaml

```
dependencies:
  flutter:
    sdk: flutter
  shared_preferences: ^2.2.2
```

The objective is to create an application that allows users to add, view and delete personal notes and persist data locally using shared_preferences. Through this project understood the basics of flutter including Textfield, Listview, FloatingActionButton. shared_preferences dependency is added on pubspec.yaml. Also understood state management using setState().

The application:

- Adds new notes with a title and content.
- View all saved notes in a scrollable list.
- Delete individual notes.

Textfield is used to take user input such as text, numbers, emails etc. Listview is a scrollable list of widgets arranged linearly so that a large number of notes can be made in a list. FloatingActionButton typically floats over the main content of the app. It is used to add a new note in the application.