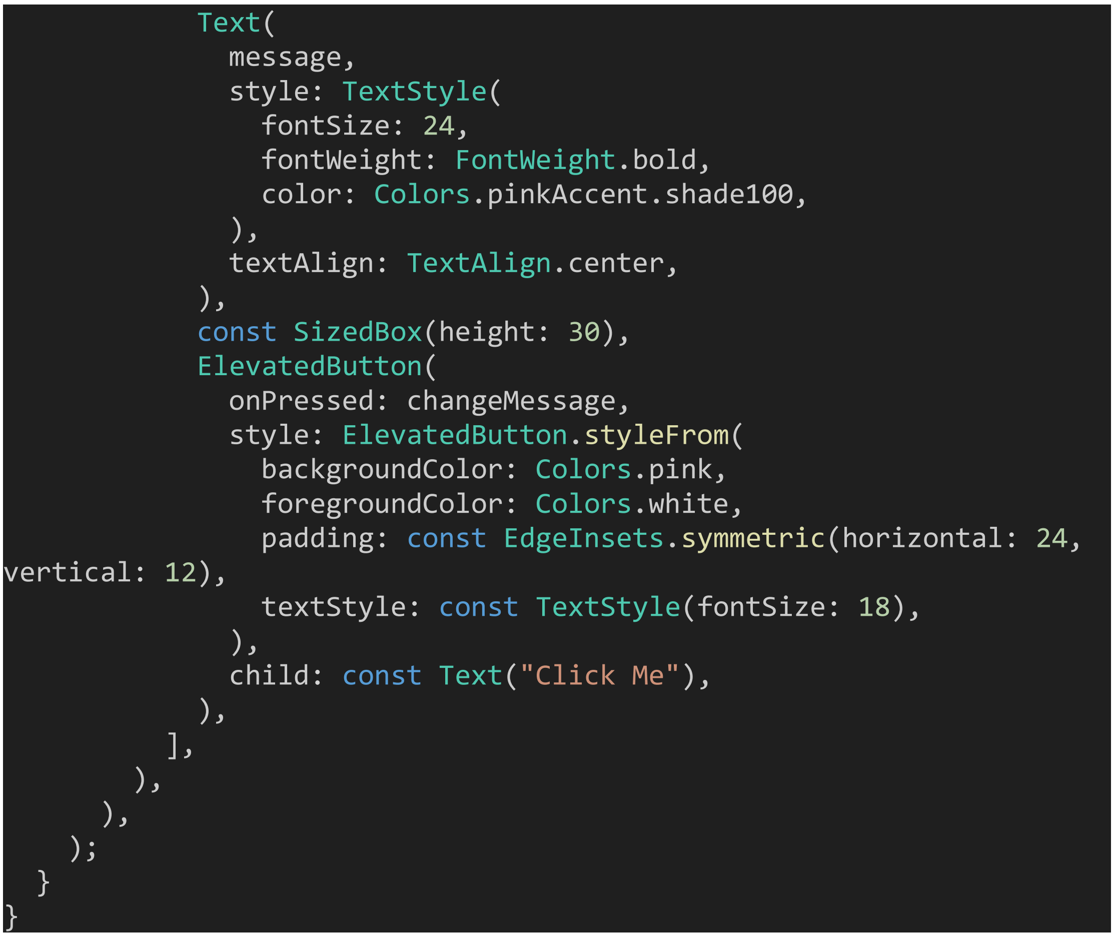
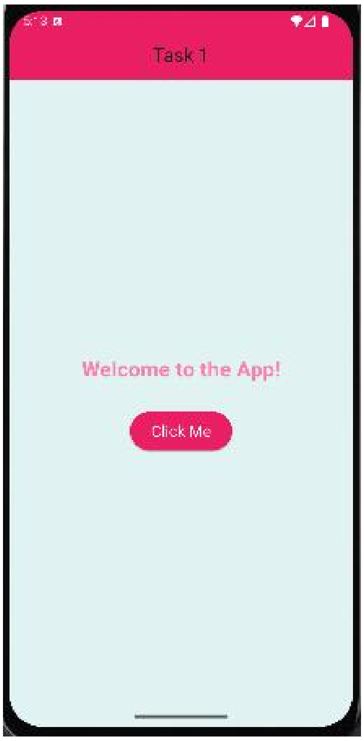
ID : 8c4e37f3a9f841868a14
Name: Amna Tanveer

TASK 1:

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

void main() {
  runApp(const Task1App());
}

class Task1App extends StatelessWidget {
  const Task1App({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Task 1',
      debugShowCheckedModeBanner: false,
      home: const WelcomeScreen(),
    );
  }
}

class WelcomeScreen extends StatefulWidget {
  const WelcomeScreen({Key? key}) : super(key: key);

  @override
  _WelcomeScreenState createState() => _WelcomeScreenState();
}

class _WelcomeScreenState extends State<WelcomeScreen> {
  String message = "Welcome to the App!";

  void changeMessage() {
    setState(() {
      message = "You clicked the button!";
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.teal.shade50,
      appBar: AppBar(
        title: const Text("Task 1"),
        centerTitle: true,
        backgroundColor: Colors.pink,
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
```

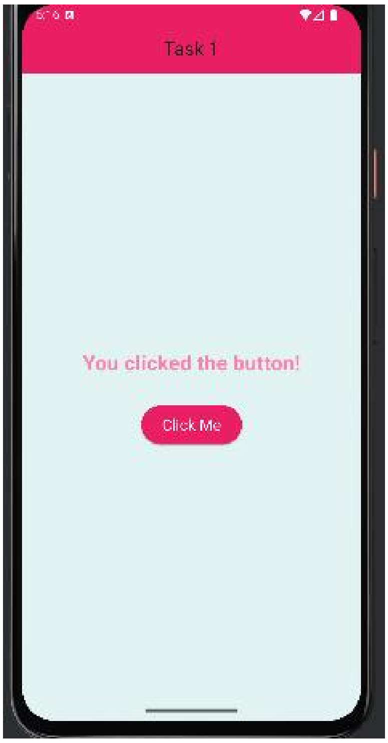


```
            Text(
              message,
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
                color: Colors.pinkAccent.shade100,
              ),
              textAlign: TextAlign.center,
            ),
            const SizedBox(height: 30),
            ElevatedButton(
              onPressed: changeMessage,
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.pink,
                foregroundColor: Colors.white,
                padding: const EdgeInsets.symmetric(horizontal: 24,
vertical: 12),
                textStyle: const TextStyle(fontSize: 18),
              ),
              child: const Text("Click Me"),
            ),
          ],
        ),
      ),
    );
  }
}
```



Task 1

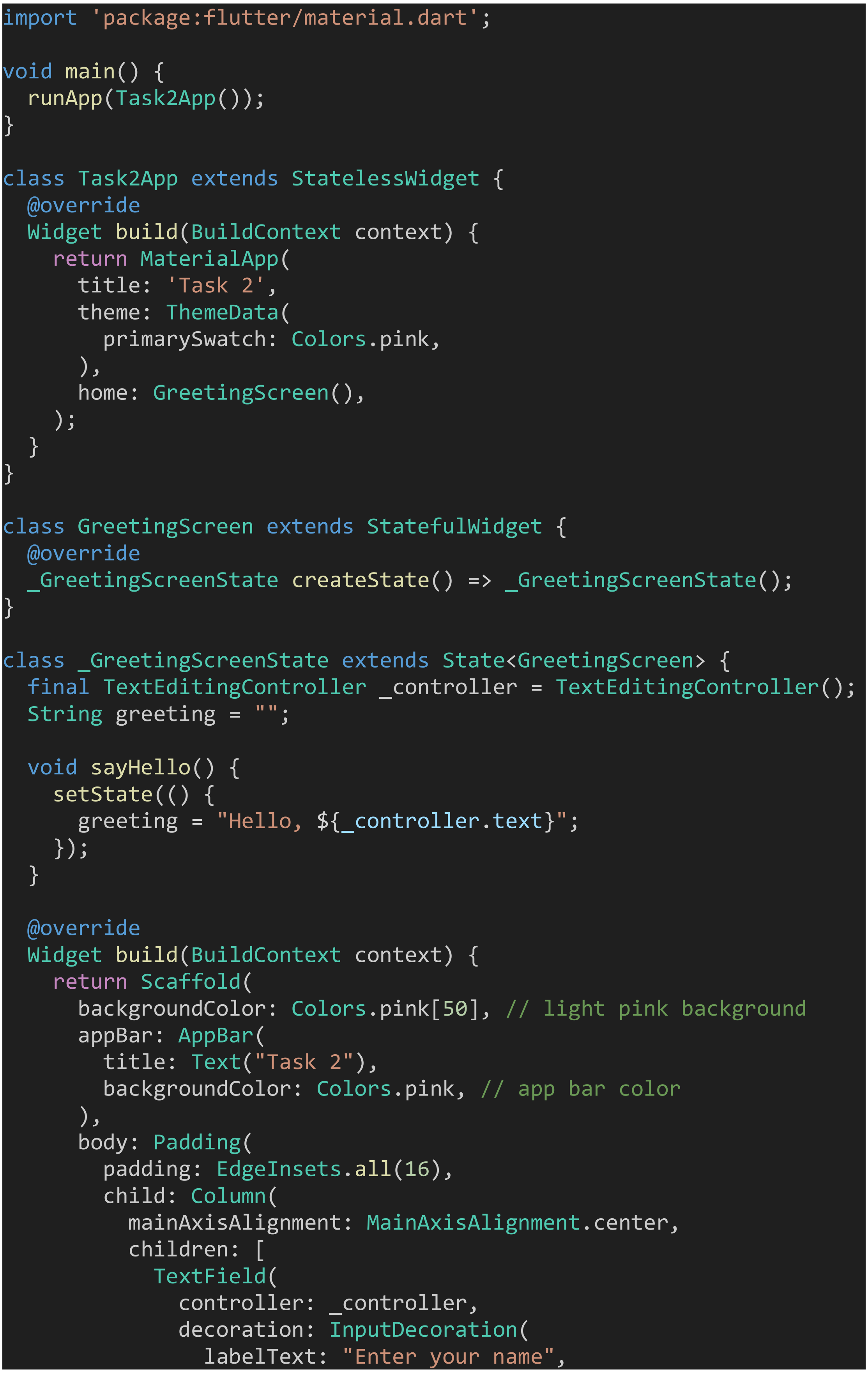
Welcome to the App!



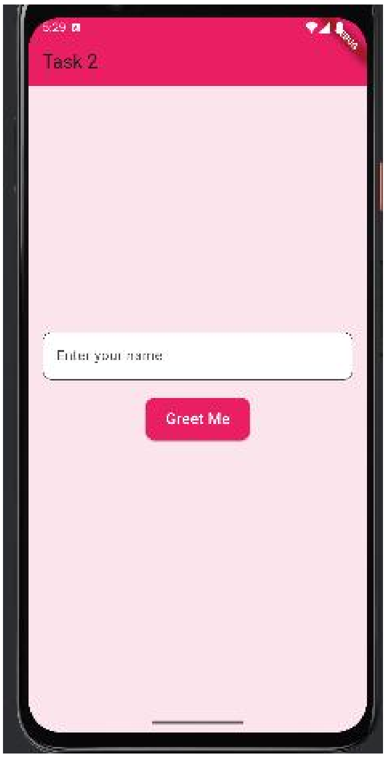
Task 1

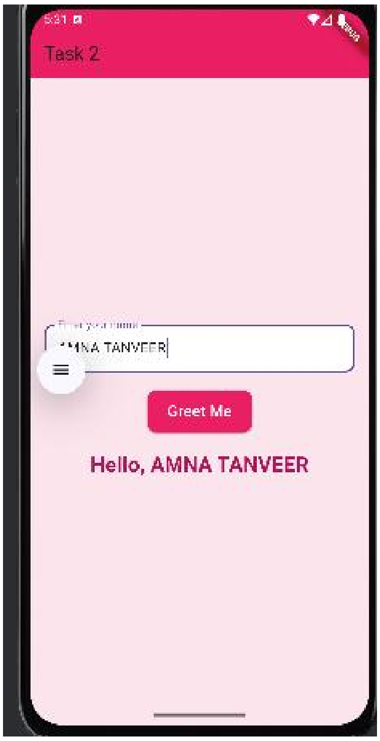
You clicked the button!

TASK 2:

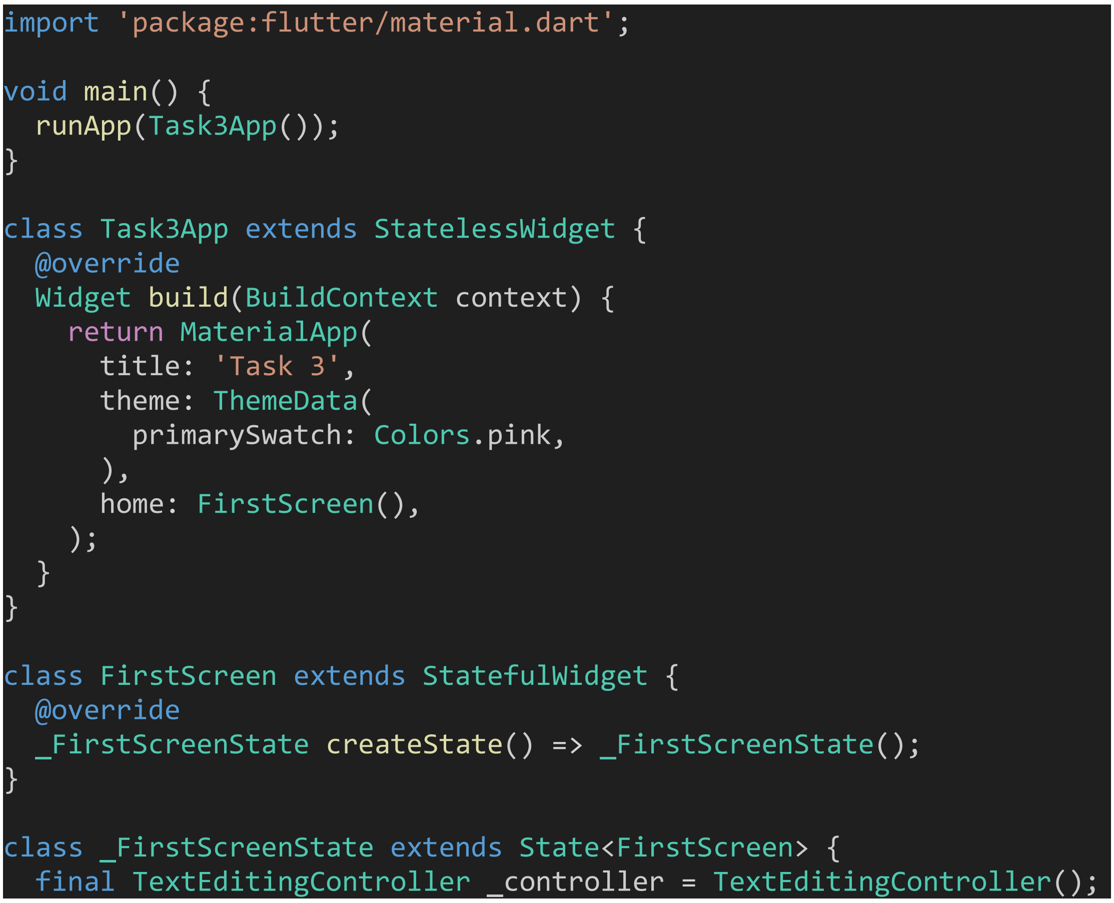


```
import 'package:flutter/material.dart';

void main() {
  runApp(Task2App());
}

class Task2App extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Task 2',
      theme: ThemeData(
        primarySwatch: Colors.pink,
      ),
      home: GreetingScreen(),
    );
  }
}

class GreetingScreen extends StatefulWidget {
  @override
  _GreetingScreenState createState() => _GreetingScreenState();
}

class _GreetingScreenState extends State<GreetingScreen> {
  final TextEditingController _controller = TextEditingController();
  String greeting = "";

  void sayHello() {
    setState(() {
      greeting = "Hello, ${_controller.text}";
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.pink[50], // light pink background
      appBar: AppBar(
        title: Text("Task 2"),
        backgroundColor: Colors.pink, // app bar color
      ),
      body: Padding(
        padding: EdgeInsets.all(16),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            TextField(
              controller: _controller,
              decoration: InputDecoration(
                labelText: "Enter your name",
                filled: true,
                fillColor: Colors.white,
                border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
              ),
            ),
            SizedBox(height: 20),
            ElevatedButton(
              onPressed: sayHello,
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.pink, // button background
                foregroundColor: Colors.white, // text color
                padding: EdgeInsets.symmetric(horizontal: 24,
vertical: 12),
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
              ),
              child: Text("Greet Me", style: TextStyle(fontSize:
18)),
            ),
            SizedBox(height: 20),
            Text(
              greeting,
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
                color: Colors.pink[800],
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

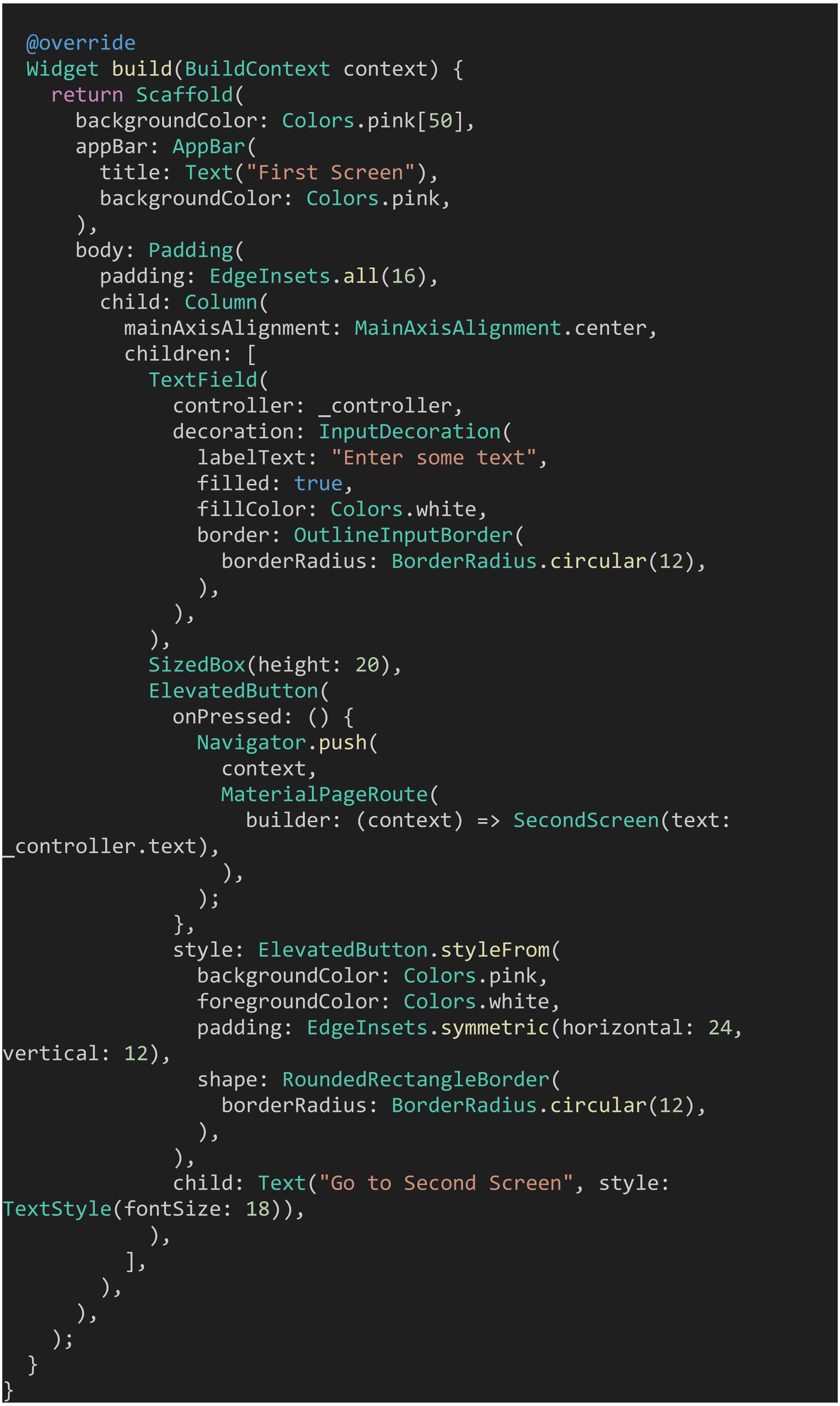




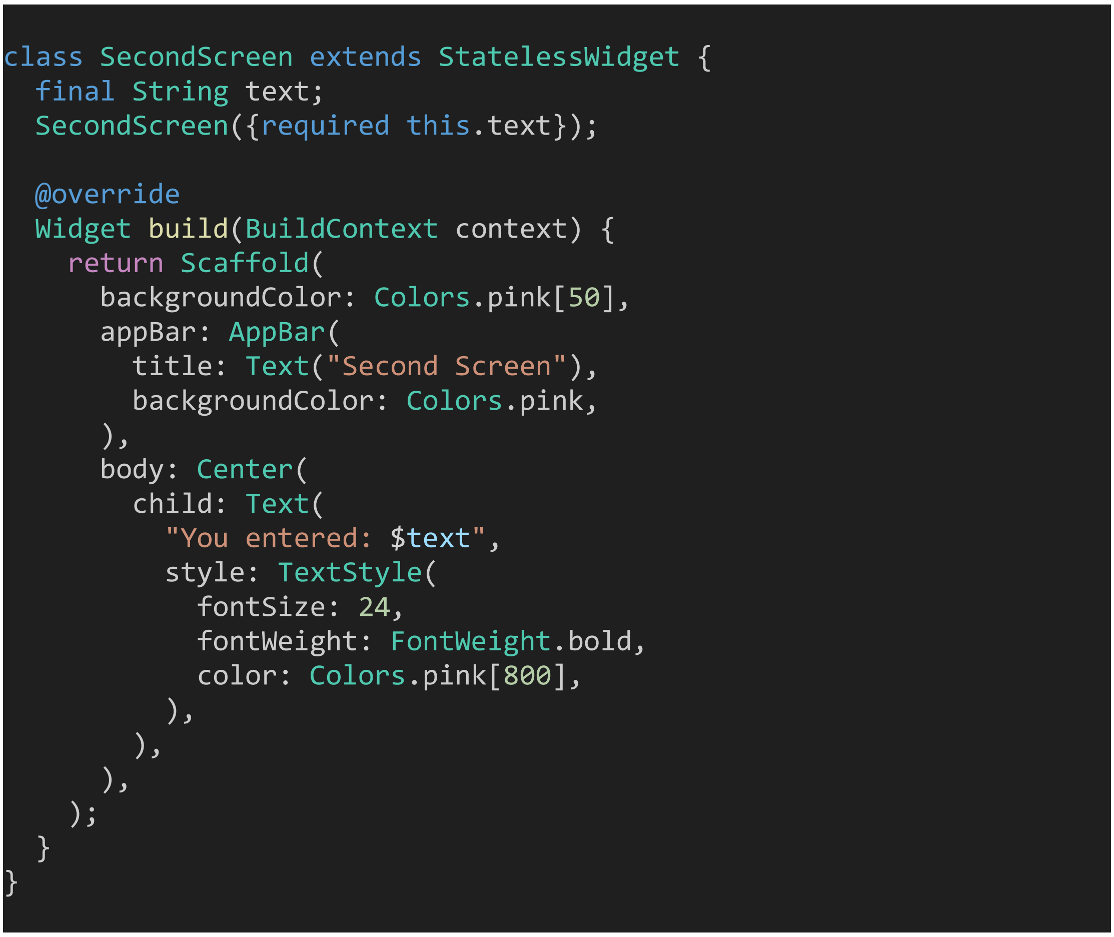
TASK 3:



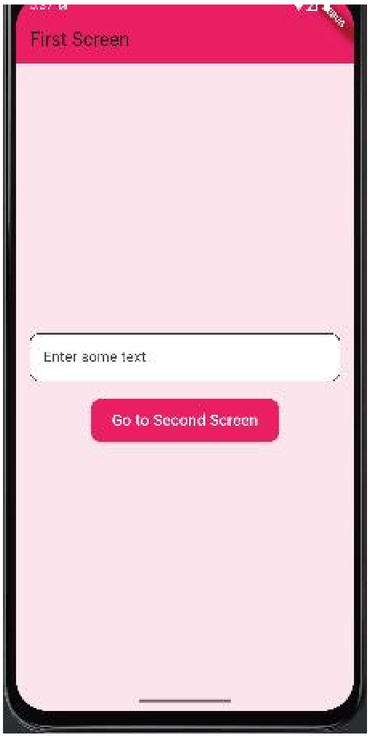
```
import 'package:flutter/material.dart';

void main() {
  runApp(Task3App());
}

class Task3App extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Task 3',
      theme: ThemeData(
        primarySwatch: Colors.pink,
      ),
      home: FirstScreen(),
    );
  }
}

class FirstScreen extends StatefulWidget {
  @override
  _FirstScreenState createState() => _FirstScreenState();
}

class _FirstScreenState extends State<FirstScreen> {
  final TextEditingController _controller = TextEditingController();
```

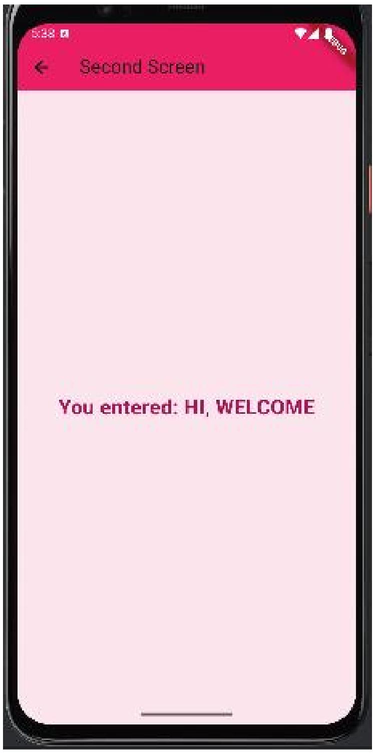


```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.pink[50],
      appBar: AppBar(
        title: Text("First Screen"),
        backgroundColor: Colors.pink,
      ),
      body: Padding(
        padding: EdgeInsets.all(16),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            TextField(
              controller: _controller,
              decoration: InputDecoration(
                labelText: "Enter some text",
                filled: true,
                fillColor: Colors.white,
                border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
              ),
            ),
            SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(
                    builder: (context) => SecondScreen(text:
_controller.text),
                  ),
                );
              },
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.pink,
                foregroundColor: Colors.white,
                padding: EdgeInsets.symmetric(horizontal: 24,
vertical: 12),
                shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(12),
                ),
              ),
              child: Text("Go to Second Screen", style:
TextStyle(fontSize: 18)),
            ),
          ],
        ),
      ),
    );
  }
}
```



```
class SecondScreen extends StatelessWidget {
  final String text;
  SecondScreen({required this.text});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.pink[50],
      appBar: AppBar(
        title: Text("Second Screen"),
        backgroundColor: Colors.pink,
      ),
      body: Center(
        child: Text(
          "You entered: $text",
          style: TextStyle(
            fontSize: 24,
            fontWeight: FontWeight.bold,
            color: Colors.pink[800],
          ),
        ),
      ),
    );
  }
}
```





You entered: HI, WELCOME

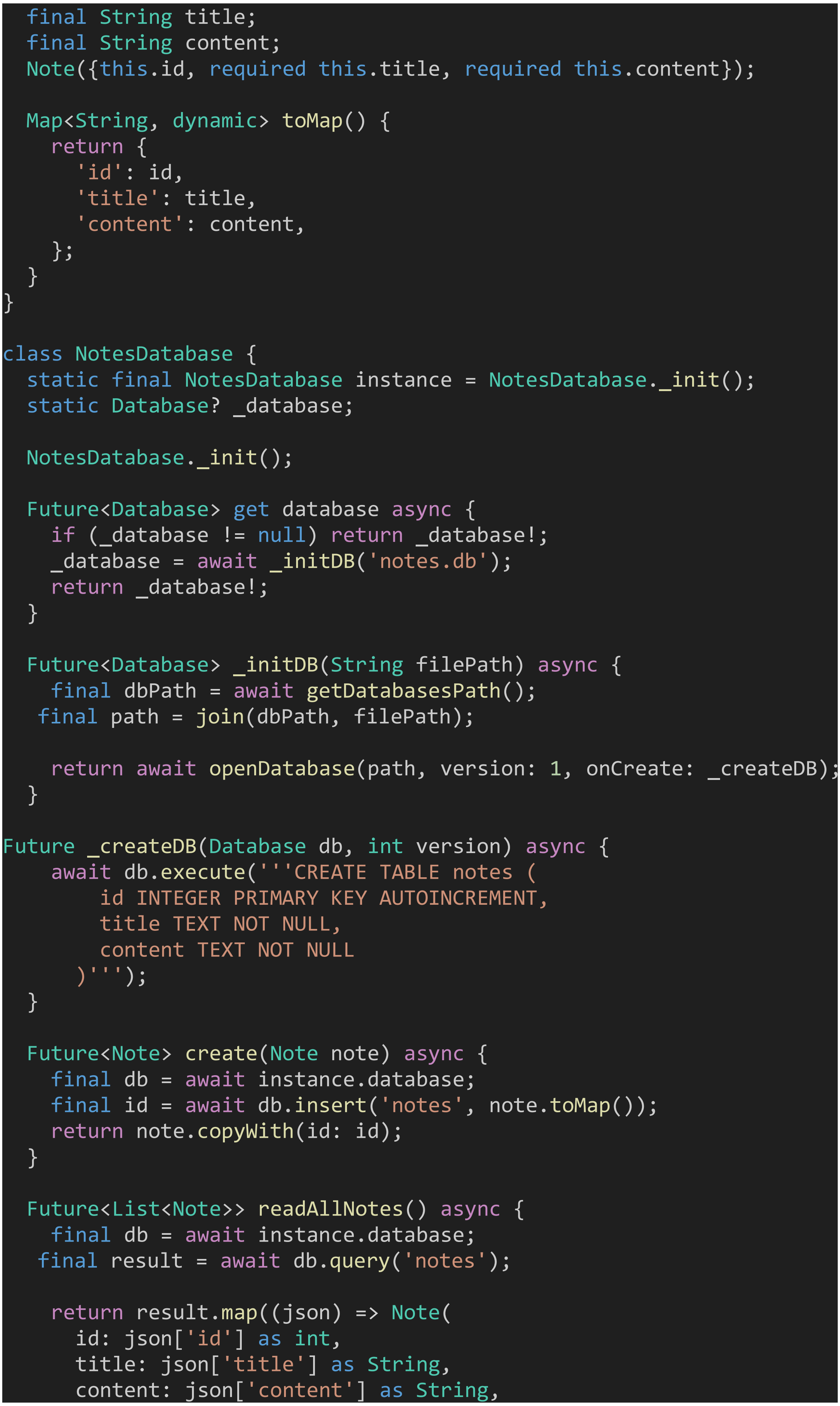
TASK 4:

```
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';

class Post {
  final int id;
  final String title;
  final String body;

  Post({required this.id, required this.title, required this.body});

  factory Post.fromJson(Map<String, dynamic> json) {
    return Post(
      id: json['id'],
      title: json['title'],
      body: json['body'],
    );
  }
}

Future<List<Post>> fetchPosts() async {
  final response =
      await
http.get(Uri.parse('https://jsonplaceholder.typicode.com/posts'));

  if (response.statusCode == 200) {
    List jsonData = json.decode(response.body);
    return jsonData.map((post) => Post.fromJson(post)).toList();
  } else {
    throw Exception('Failed to load posts');
  }
}

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'API Fetch Example',
      theme: ThemeData(primarySwatch: Colors.blue),
      home: const PostsPage(),
    );
  }
}
```

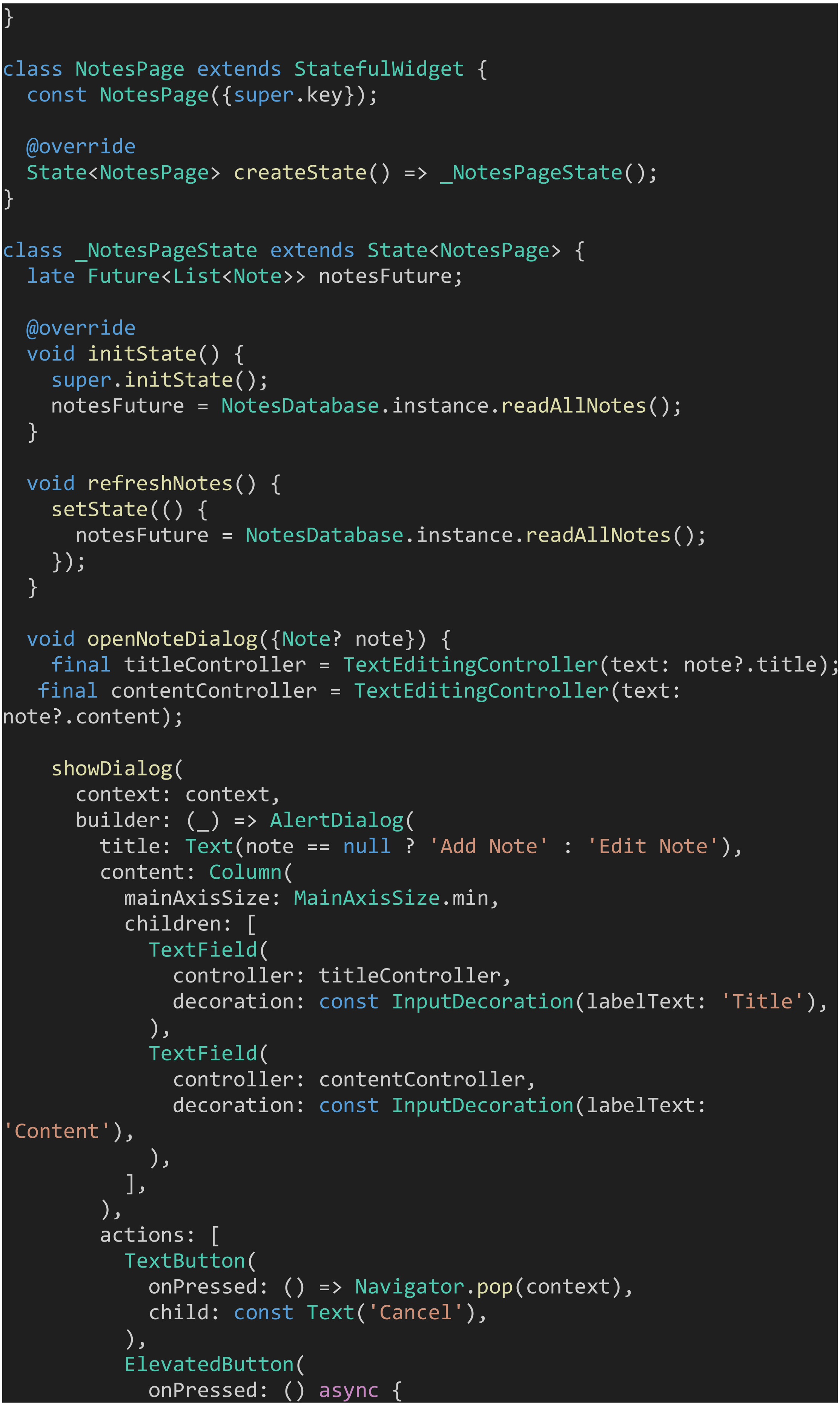
```
class PostsPage extends StatelessWidget {
  const PostsPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Posts from API')),
      body: FutureBuilder<List<Post>>(
        future: fetchPosts(),
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return const Center(child: CircularProgressIndicator());
          } else if (snapshot.hasError) {
            return Center(child: Text('Error: ${snapshot.error}'));
          } else if (!snapshot.hasData || snapshot.data!.isEmpty) {
            return const Center(child: Text('No posts available'));
          } else {
            final posts = snapshot.data!;
            return ListView.builder(
              itemCount: posts.length,
              itemBuilder: (context, index) {
                final post = posts[index];
                return Card(
                  margin: const EdgeInsets.symmetric(horizontal: 12,
vertical: 6),
                  elevation: 2,
                  child: ListTile(
                    title: Text(post.title,
                        style: const TextStyle(fontWeight:
FontWeight.bold)),
                    subtitle: Text(post.body),
                  ),
                );
              },
            );
          }
        },
      ),
    );
  }
}
```

TASK 5:

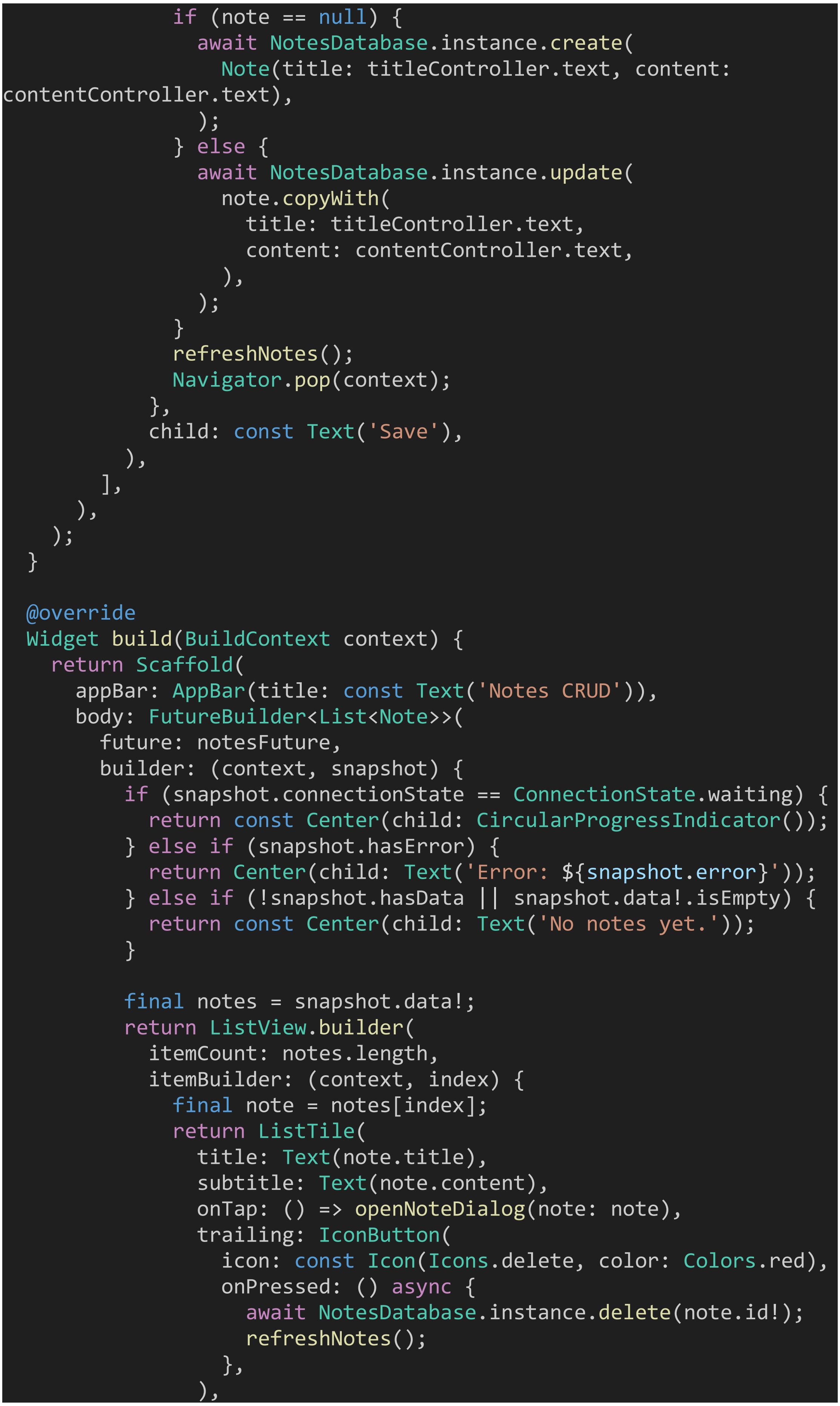
```
import 'package:flutter/material.dart';
import 'package:path/path.dart';
import 'package:sqflite/sqflite.dart';

class Note {
  final int? id;
```



```
  final String title;
  final String content;
  Note({this.id, required this.title, required this.content});

  Map<String, dynamic> toMap() {
    return {
      'id': id,
      'title': title,
      'content': content,
    };
  }
}

class NotesDatabase {
  static final NotesDatabase instance = NotesDatabase._init();
  static Database? _database;

  NotesDatabase._init();

  Future<Database> get database async {
    if (_database != null) return _database!;
    _database = await _initDB('notes.db');
    return _database!;
  }

  Future<Database> _initDB(String filePath) async {
    final dbPath = await getDatabasesPath();
   final path = join(dbPath, filePath);

    return await openDatabase(path, version: 1, onCreate: _createDB);
  }

Future _createDB(Database db, int version) async {
    await db.execute('''CREATE TABLE notes (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        title TEXT NOT NULL,
        content TEXT NOT NULL
      )''');
  }

  Future<Note> create(Note note) async {
    final db = await instance.database;
    final id = await db.insert('notes', note.toMap());
    return note.copyWith(id: id);
  }

  Future<List<Note>> readAllNotes() async {
    final db = await instance.database;
   final result = await db.query('notes');

    return result.map((json) => Note(
      id: json['id'] as int,
      title: json['title'] as String,
      content: json['content'] as String,
    )).toList();
  }

  Future<int> update(Note note) async {
    final db = await instance.database;
    return db.update(
      'notes',
      note.toMap(),
      where: 'id = ?',
      whereArgs: [note.id],
    );
  }

  Future<int> delete(int id) async {
    final db = await instance.database;
    return db.delete(
      'notes',
      where: 'id = ?',
      whereArgs: [id],
    );
  }

  Future close() async {
    final db = await instance.database;
    db.close();
  }
}

extension on Note {
  Note copyWith({int? id, String? title, String? content}) {
    return Note(
      id: id ?? this.id,
      title: title ?? this.title,
      content: content ?? this.content,
    );
  }
}

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Notes CRUD',
      theme: ThemeData(primarySwatch: Colors.blue),
      home: const NotesPage(),
    );
  }
```



```
}

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

  @override
  State<NotesPage> createState() => _NotesPageState();
}

class _NotesPageState extends State<NotesPage> {
  late Future<List<Note>> notesFuture;

  @override
  void initState() {
    super.initState();
    notesFuture = NotesDatabase.instance.readAllNotes();
  }

  void refreshNotes() {
    setState(() {
      notesFuture = NotesDatabase.instance.readAllNotes();
    });
  }

  void openNoteDialog({Note? note}) {
    final titleController = TextEditingController(text: note?.title);
   final contentController = TextEditingController(text:
note?.content);

    showDialog(
      context: context,
      builder: (_) => AlertDialog(
        title: Text(note == null ? 'Add Note' : 'Edit Note'),
        content: Column(
          mainAxisSize: MainAxisSize.min,
          children: [
            TextField(
              controller: titleController,
              decoration: const InputDecoration(labelText: 'Title'),
            ),
            TextField(
              controller: contentController,
              decoration: const InputDecoration(labelText:
'Content'),
            ),
          ],
        ),
        actions: [
          TextButton(
            onPressed: () => Navigator.pop(context),
            child: const Text('Cancel'),
          ),
          ElevatedButton(
            onPressed: () async {
```



```
              if (note == null) {
                await NotesDatabase.instance.create(
                  Note(title: titleController.text, content:
contentController.text),
                );
              } else {
                await NotesDatabase.instance.update(
                  note.copyWith(
                    title: titleController.text,
                    content: contentController.text,
                  ),
                );
              }
              refreshNotes();
              Navigator.pop(context);
            },
            child: const Text('Save'),
          ),
        ],
      ),
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Notes CRUD')),
      body: FutureBuilder<List<Note>>(
        future: notesFuture,
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return const Center(child: CircularProgressIndicator());
          } else if (snapshot.hasError) {
            return Center(child: Text('Error: ${snapshot.error}'));
          } else if (!snapshot.hasData || snapshot.data!.isEmpty) {
            return const Center(child: Text('No notes yet.'));
          }

          final notes = snapshot.data!;
          return ListView.builder(
            itemCount: notes.length,
            itemBuilder: (context, index) {
              final note = notes[index];
              return ListTile(
                title: Text(note.title),
                subtitle: Text(note.content),
                onTap: () => openNoteDialog(note: note),
                trailing: IconButton(
                  icon: const Icon(Icons.delete, color: Colors.red),
                  onPressed: () async {
                    await NotesDatabase.instance.delete(note.id!);
                    refreshNotes();
                  },
                ),
              );
            },
          );
        },
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () => openNoteDialog(),
        child: const Icon(Icons.add),
      ),
    );
  }
}
```