

Minahil Kashif

Minahilkashif10@gmail.com

Minahil K

CODE

main.dart

```
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import 'viewmodels/task_viewmodel.dart';
import 'views/task_list_screen.dart';

void main() {
  runApp(
    MultiProvider(
      providers: [
        ChangeNotifierProvider(create: (_) => TaskViewModel()),
      ],
      child: MyApp(),
    ),
  );
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Simple To-Do App',
```

```
        theme: ThemeData(primarySwatch: Colors.brown),  
        home: TaskListScreen(),  
    );  
}  
}
```

task.dart

```
class Task {  
  
    final String title;  
  
    final String description;  
  
    Task({required this.title, required this.description});  
}
```

Task_viewmodel.dart (this has my provider class)

```
import 'package:flutter/foundation.dart';  
import '../models/task.dart';  
  
class TaskViewModel with ChangeNotifier {  
  
    final List<Task> _tasks = [];  
  
    List<Task> get tasks => _tasks;  
  
    void addTask(String title, String description) {  
        if (title.isNotEmpty && description.isNotEmpty) {  
            _tasks.add(Task(title: title, description: description));  
            notifyListeners();  
        }  
    }  
}
```

```
}  
}
```

Task_list_screen.dart

```
import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
import '../viewmodels/task_viewmodel.dart';  
import 'add_task_screen.dart';  
  
class TaskListScreen extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,  
        title: Text('To-Do List')  
      ),  
      body: Consumer<TaskViewModel>(  
        builder: (context, taskViewModel, child) {  
          final tasks = taskViewModel.tasks;  
          return tasks.isEmpty  
            ? Center(child: Text('No tasks added yet'))  
            : ListView.builder(  
              itemCount: tasks.length,  
              itemBuilder: (context, index) {  
                final task = tasks[index];  
                return ListTile(  
                  title: Text(task.title),
```

```

        subtitle: Text(task.description),

      );

    },

  );

},

),

floatingActionButton: FloatingActionButton(

  onPressed: () {

    Navigator.push(

      context,

      MaterialPageRoute(builder: (context) => AddTaskScreen()),

    );

  },

  child: Icon(Icons.add),

),

);

}

}

```

add_task_screen.dart

```

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import '../viewmodels/task_viewmodel.dart';

class AddTaskScreen extends StatelessWidget {

  final TextEditingController _titleController = TextEditingController();

  final TextEditingController _descriptionController = TextEditingController();

```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: Text('Add New Task')),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        children: [
          TextField(
            controller: _titleController,
            decoration: InputDecoration(labelText: 'Task Title'),
          ),
          TextField(
            controller: _descriptionController,
            decoration: InputDecoration(labelText: 'Task Description'),
          ),
          SizedBox(height: 20),
          ElevatedButton(
            onPressed: () {
              final title = _titleController.text.trim();
              final description = _descriptionController.text.trim();
              if (title.isNotEmpty && description.isNotEmpty) {
                Provider.of<TaskViewModel>(context, listen: false)
                  .addTask(title, description);
                Navigator.pop(context);
              } else {
                ScaffoldMessenger.of(context).showSnackBar(
```

```
        Snackbar(content: Text('Please fill in all fields')),  
      );  
    }  
  },  
  child: Text('Add Task'),  
),  
],  
),  
),  
);  
}  
}
```