

```

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

void main() async {
  await Hive.initFlutter();
  await Hive.openBox('tasksBox');
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: ToDoListScreen(),
    );
  }
}

class ToDoListScreen extends StatefulWidget {
  @override
  _ToDoListScreenState createState() => _ToDoListScreenState();
}

class _ToDoListScreenState extends State<ToDoListScreen> {
  final _taskBox = Hive.box('tasksBox');
  final TextEditingController _controller = TextEditingController();

  void _addTask(String task) {
    if (task.isNotEmpty) {
      _taskBox.add({'task': task, 'completed': false});
      _controller.clear();
      setState(() {});
    }
  }

  void _toggleTaskCompletion(int index) {
    var task = _taskBox.getAt(index);
    _taskBox.putAt(index, {'task': task['task'], 'completed': !task['completed']});
    setState(() {});
  }
}

```

```
}
```

```
void _deleteTask(int index) {  
  _taskBox.deleteAt(index);  
  setState(() {});  
}
```

```
@override
```

```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(title: Text('To-Do List')),  
    body: Column(  
      children: [  
        Padding(  
          padding: const EdgeInsets.all(8.0),  
          child: Row(  
            children: [  
              Expanded(  
                child: TextField(  
                  controller: _controller,  
                  decoration: InputDecoration(labelText: 'Enter Task'),  
                ),  
              ),  
              IconButton(  
                icon: Icon(Icons.add),  
                onPressed: () => _addTask(_controller.text),  
              ),  
            ],  
          ),  
        ),  
        Expanded(  
          child: ValueListenableBuilder(  
            valueListenable: _taskBox.listenable(),  
            builder: (context, Box box, _) {  
              if (box.isEmpty) {  
                return Center(child: Text('No tasks yet!'));  
              }  
              return ListView.builder(  
                itemCount: box.length,  
                itemBuilder: (context, index) {
```

```

var task = box.getAt(index);
return ListTile(
  title: Text(
    task['task'],
    style: TextStyle(
      decoration: task['completed']
        ? TextDecoration.lineThrough
        : TextDecoration.none,
    ),
  ),
  leading: Checkbox(
    value: task['completed'],
    onChanged: (value) => _toggleTaskCompletion(index),
  ),
  trailing: IconButton(
    icon: Icon(Icons.delete, color: Colors.red),
    onPressed: () => _deleteTask(index),
  ),
);
},
);
},
),
),
],
),
);
}
}

```