



Faculty of Technology and Engineering

Chandubhai S. Patel Institute of Technology

Department of Computer Science & Engineering

Date: / /

Laboratory Manual Performa

Academic Year	:	2024-25	Semester	:	5 TH
Course code	:	CSE309	Course name	:	Mobile Application and Development

Practical - 4

Create a to-do Application using Text field Button. On the button click event display data in List.

Introduction of relevant concepts used:

Code (Screenshot)

Main.dart

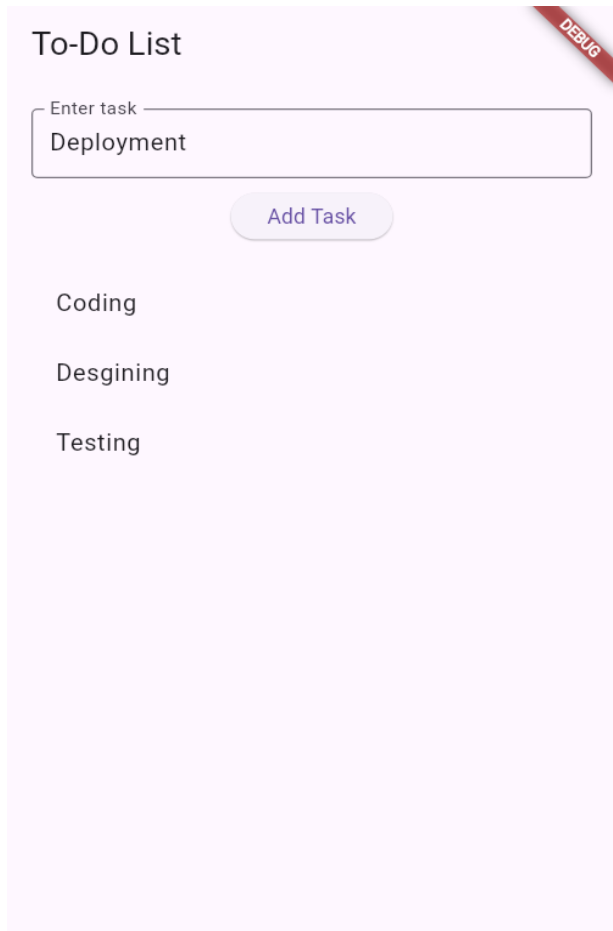
```
main.dart ×
lib > main.dart > _TodoAppState > build
1  import 'package:flutter/material.dart';
2
3  Run | Debug | Profile
4  void main() {
5    runApp(const MyApp());
6  }
7
8  class MyApp extends StatelessWidget {
9    const MyApp({Key? key}) : super(key: key);
10
11    @override
12    Widget build(BuildContext context) {
13      return MaterialApp(
14        title: 'To-Do App',
15        theme: ThemeData(
16          primarySwatch: Colors.blue,
17        ), // ThemeData
18        home: const TodoApp(),
19      ); // MaterialApp
20    }
21  }
22
23  class TodoApp extends StatefulWidget {
24    const TodoApp({Key? key}) : super(key: key);
25
26    @override
27    _TodoAppState createState() => _TodoAppState();
28  }
```

```

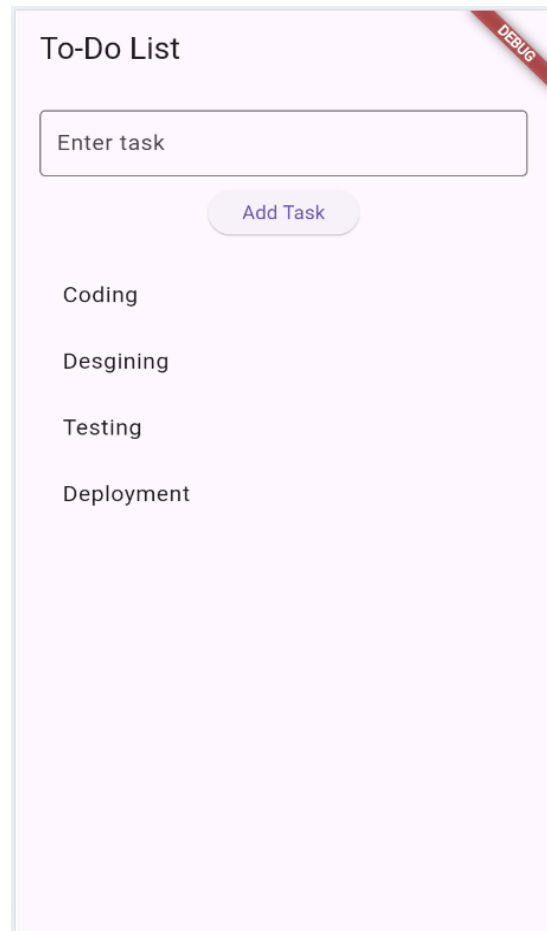
28
29 class _TodoAppState extends State<TodoApp> {
30   final _taskController = TextEditingController();
31   final List<String> _tasks = [];
32
33   void _addTask() {
34     String task = _taskController.text;
35     if (task.isNotEmpty) {
36       setState(() {
37         _tasks.add(task);
38         _taskController.clear();
39       });
40     }
41   }
42
43   @override
44   Widget build(BuildContext context) {
45     return Scaffold(
46       appBar: AppBar(
47         title: const Text('To-Do List'),
48       ), // AppBar
49       body: Padding(
50         padding: const EdgeInsets.all(16.0),
51         child: Column(
52           children: [
53             TextField(
54               controller: _taskController,
55               decoration: const InputDecoration(
56                 labelText: 'Enter task',
57                 border: OutlineInputBorder(),
58               ), // InputDecoration
59             ), // TextField
60             const SizedBox(height: 10),
61             ElevatedButton(
62               onPressed: _addTask,
63               child: const Text('Add Task'),
64             ), // ElevatedButton
65             const SizedBox(height: 20),
66             Expanded(
67               child: ListView.builder(
68                 itemCount: _tasks.length,
69                 itemBuilder: (context, index) {
70                   return ListTile(
71                     title: Text(_tasks[index]),
72                   ); // ListTile
73                 },
74               ), // ListView.builder
75             ), // Expanded
76           ],
77         ), // Column
78       ), // Padding
79     ); // Scaffold
80   }
81 }
82

```

Output (Screenshot)



The screenshot shows a web application titled "To-Do List" with a light purple background. In the top right corner, there is a red ribbon with the word "DEBUG" in white. Below the title, there is a text input field with the placeholder text "Enter task". The word "Deployment" has been typed into this field. To the right of the input field is a rounded rectangular button with the text "Add Task" in purple. Below the input field, there is a list of tasks: "Coding", "Desgining", and "Testing".



This screenshot shows the same "To-Do List" application after the "Add Task" button has been clicked. The "Enter task" input field is now empty. The task list now includes four items: "Coding", "Desgining", "Testing", and "Deployment". The "DEBUG" ribbon remains in the top right corner.

Grade/Marks
(____ / 10)

Sign of Lab Teacher with Date