

Mobile Application Development Lab Report

Department of Computer Science

NUML

Student Name: Muhammad Awais

Roll No: 2883

Report: Lab 01

Semester: Fall 2025

Instructor: Sir Shahid Malik

Date of Submission: November 26, 2025

Lab Task 1: state management using the Provider

Code Snippet Attendance Model

```
1 import 'package:flutter/foundation.dart';
2 import 'student.dart';
3
4 class AttendanceModel extends ChangeNotifier {
5
6     final List<Student> _students = [
7         Student(id: 1, name: 'Alice Johnson', isPresent: false),
8         Student(id: 2, name: 'Bob Smith', isPresent: false),
9         Student(id: 3, name: 'Charlie Brown', isPresent: false),
10        Student(id: 4, name: 'Diana Prince', isPresent: false),
11        Student(id: 5, name: 'Ethan Hunt', isPresent: false),
12        Student(id: 6, name: 'Fiona Green', isPresent: false),
13        Student(id: 7, name: 'George Miller', isPresent: false),
14        Student(id: 8, name: 'Hannah Montana', isPresent: false),
15        Student(id: 9, name: 'Isaac Newton', isPresent: false),
16        Student(id: 10, name: 'Julia Roberts', isPresent: false),
17    ];
18
19
20     List<Student> get students => List.unmodifiable(_students);
21
22
23     void toggleAttendance(int studentId) {
24         final index = _students.indexWhere((student) => student.id ==
25             studentId);
26         if (index != -1) {
27             _students[index] = _students[index].copyWith(
28                 isPresent: !_students[index].isPresent,
29             );
30             notifyListeners();
31         }
32     }
33
34     void markAllAbsent() {
35         for (int i = 0; i < _students.length; i++) {
36             _students[i] = _students[i].copyWith(isPresent: false);
37         }
38         notifyListeners();
39     }
40
41
42     int getPresentCount() {
43         return _students.where((student) => student.isPresent).length;
44     }
45
46
47     Student? getStudentById(int studentId) {
48         try {
49             return _students.firstWhere((student) => student.id == studentId)
50                 ;
51         } catch (e) {
52             return null;
53         }
54     }
55 }
```

```

52     }
53   }
54
55   bool getStudentAttendance(int studentId) {
56     return getStudentById(studentId)?.isPresent ?? false;
57   }
58 }
```

Listing 1: attendance-model.dart

Code Snippet Student Model

```

1
2 class Student {
3   final int id;
4   final String name;
5   bool isPresent;
6
7   Student({
8     required this.id,
9     required this.name,
10    this.isPresent = false,
11  });
12
13
14   Student copyWith({
15     int? id,
16     String? name,
17     bool? isPresent,
18   }) {
19     return Student(
20       id: id ?? this.id,
21       name: name ?? this.name,
22       isPresent: isPresent ?? this.isPresent,
23     );
24   }
25
26   @override
27   String toString() => 'Student(id: $id, name: $name, isPresent:
28     $isPresent)';
}
```

Listing 2: Student.dart

Code Snippet Widget (Student)

```

1 import 'package:flutter/material.dart';
2 import 'package:provider/provider.dart';
3 import '../models/attendance_model.dart';
4 import '../models/student.dart';
5
6 class StudentRow extends StatelessWidget {
7   final Student student;
8
9   const StudentRow({
10     Key? key,
```

```

11     required this.student,
12   ) : super(key: key);
13
14   @override
15   Widget build(BuildContext context) {
16
17     final isPresent = context.select<AttendanceModel, bool>(
18       (model) => model.getStudentAttendance(student.id),
19     );
20
21     return Container(
22       margin: const EdgeInsets.symmetric(horizontal: 8, vertical: 4),
23       decoration: BoxDecoration(
24         borderRadius: BorderRadius.circular(8),
25         color: isPresent ? Colors.green.withOpacity(0.1) : Colors.red.
26           withOpacity(0.1),
27       ),
28       child: ListTile(
29         title: Text(
30           student.name,
31           style: TextStyle(
32             fontWeight: FontWeight.w500,
33             color: isPresent ? Colors.green.shade700 : Colors.grey.
34               shade700,
35           ),
36         ),
37         trailing: Switch(
38           value: isPresent,
39           onChanged: (value) {
40
41             context.read<AttendanceModel>().toggleAttendance(student.id
42               );
43           },
44           activeColor: Colors.green,
45         ),
46       );
47     }

```

Listing 3: Student-row.dart

Code Snippet Main

```

1 import 'package:flutter/material.dart';
2 import 'package:provider/provider.dart';
3 import '../models/attendance_model.dart';
4 import '../widgets/student_row.dart';
5
6 void main() {
7   runApp(const MyApp());
8 }
9
10 class MyApp extends StatelessWidget {
11   const MyApp({Key? key}) : super(key: key);
12
13   @override

```

```

14     Widget build(BuildContext context) {
15       return MaterialApp(
16         debugShowCheckedModeBanner: false,
17         title: 'Attendance Tracker',
18         theme: ThemeData(
19           primarySwatch: Colors.blue,
20           useMaterial3: true,
21         ),
22         home: ChangeNotifierProvider(
23             create: (context) => AttendanceModel(),
24             child: const AttendanceScreen(),
25         ),
26     );
27   }
28 }
29
30 class AttendanceScreen extends StatelessWidget {
31   const AttendanceScreen({Key? key}) : super(key: key);
32
33   @override
34   Widget build(BuildContext context) {
35     return Scaffold(
36       appBar: _buildAppBar(context),
37       body: _buildStudentList(context),
38       floatingActionButton: _buildFloatingActionButton(context),
39     );
40   }
41
42 PreferredSizeWidget _buildAppBar(BuildContext context) {
43   return AppBar(
44     title: Consumer<AttendanceModel>(
45       builder: (context, attendanceModel, _) {
46         final presentCount = attendanceModel.getPresentCount();
47         final totalCount = attendanceModel.students.length;
48         return Text(
49           'Attendance: $presentCount/$totalCount Present',
50           style: const TextStyle(fontSize: 18, fontWeight: FontWeight
51             .bold),
52         );
53       },
54       elevation: 4,
55       backgroundColor: Colors.blue.shade700,
56     );
57   }
58
59
60   Widget _buildStudentList(BuildContext context) {
61     return Consumer<AttendanceModel>(
62       builder: (context, attendanceModel, _) {
63         final students = attendanceModel.students;
64
65         if (students.isEmpty) {
66           return const Center(
67             child: Text('No students available'),
68         );
69       }
70     }
71   }

```

```
71     return ListView.builder(
72         itemCount: students.length,
73         itemBuilder: (context, index) {
74             return StudentRow(student: students[index]);
75         },
76     );
77 },
78 );
79 }
80
81
82 Widget _buildFloatingActionButton(BuildContext context) {
83     return FloatingActionButton.extended(
84         onPressed: () {
85             context.read<AttendanceModel>().markAllAbsent();
86             ScaffoldMessenger.of(context).showSnackBar(
87                 const SnackBar(
88                     content: Text('All students marked as absent'),
89                     duration: Duration(seconds: 2),
90                 ),
91             );
92         },
93         icon: const Icon(Icons.clear_all),
94         label: const Text('Mark All Absent'),
95         backgroundColor: Colors.red.shade600,
96     );
97 }
98 }
```

Listing 4: main.dart

Output Screenshot

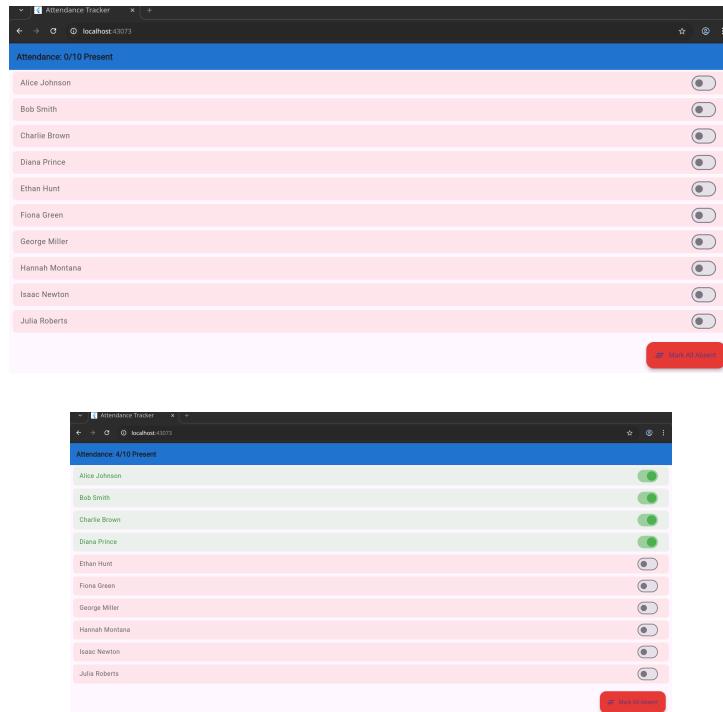


Figure 1: Main APP