

Calendar Application with Add Event

1. Add dependency in pubspec.yaml

```
29 dependencies:
30   flutter:
31     sdk: flutter
32
33
34   # The following adds the Cupertino Icons font to your application.
35   # Use with the CupertinoIcons class for iOS style icons.
36   cupertino_icons: ^1.0.2
37   table_calendar: ^3.0.0
```

2. Create calender.dart file

```
1 import 'package:calendar_event/event.dart';
2 import 'package:flutter/material.dart';
3 import 'package:table_calendar/table_calendar.dart';
4
5 class Calendar extends StatefulWidget {
6   @override
7   _CalendarState createState() => _CalendarState();
8 }
9
10 class _CalendarState extends State<Calendar> {
11   Map<DateTime, List<Event>> selectedEvents;
12   CalendarFormat format = CalendarFormat.month;
13   DateTime selectedDay = DateTime.now();
14   DateTime focusedDay = DateTime.now();
15
16   TextEditingController _eventController = TextEditingController();
17
18   @override
19   void initState() {
20     selectedEvents = {};
21     super.initState();
22   }
23 }
```

```

24 List<Event> _getEventsfromDay(DateTime date) {
25     return selectedEvents[date] ?? [];
26 }
27
28 @override
29 void dispose() {
30     _eventController.dispose();
31     super.dispose();
32 }
33
34 @override
35 Widget build(BuildContext context) {
36     return Scaffold(
37         appBar: AppBar(
38             title: Text("Calendar Event"),
39             centerTitle: true,
40         ), // AppBar
41         body: Column(
42             children: [
43                 TableCalendar(
44                     focusedDay: selectedDay,
45                     firstDay: DateTime(1990),
46                     lastDay: DateTime(2050),
47                     calendarFormat: format,
48                     onFormatChanged: (CalendarFormat _format) {
49                         setState(() {
50                             format = _format;
51                         });
52                     },
53                     startingDayOfWeek: StartingDayOfWeek.sunday,
54                     daysOfWeekVisible: true,
55
56                     //Day Changed
57                     onDaySelected: (DateTime selectDay, DateTime focusDay) {
58                         setState(() {
59                             selectedDay = selectDay;
60                             focusedDay = focusDay;
61                         });
62                         print(focusedDay);
63                     },
64                     selectedDayPredicate: (DateTime date) {
65                         return isSameDay(selectedDay, date);
66                     },

```

lib > calendar.dart > ...

```
68     eventLoader: _getEventsfromDay,
69
70     //To style the Calendar
71     calendarStyle: CalendarStyle(
72       isTodayHighlighted: true,
73       selectedDecoration: BoxDecoration(
74         color: Colors.blue,
75         shape: BoxShape.rectangle,
76         borderRadius: BorderRadius.circular(5.0),
77       ), // BoxDecoration
78       selectedTextStyle: TextStyle(color: Colors.white),
79       todayDecoration: BoxDecoration(
80         color: Colors.purpleAccent,
81         shape: BoxShape.rectangle,
82         borderRadius: BorderRadius.circular(5.0),
83       ), // BoxDecoration
84       defaultDecoration: BoxDecoration(
85         shape: BoxShape.rectangle,
86         borderRadius: BorderRadius.circular(5.0),
87       ), // BoxDecoration
88       weekendDecoration: BoxDecoration(
89         shape: BoxShape.rectangle,
90         borderRadius: BorderRadius.circular(5.0),
```

```
94     formatButtonVisible: true,
95     titleCentered: true,
96     formatButtonShowsNext: false,
97     formatButtonDecoration: BoxDecoration(
98       color: Colors.blue,
99       borderRadius: BorderRadius.circular(5.0),
100     ), // BoxDecoration
101     formatButtonTextStyle: TextStyle(
102       color: Colors.white,
103     ), // TextStyle
104   ), // HeaderStyle
105 ), // TableCalendar
106 ..._getEventsfromDay(selectedDay).map(
107   (Event event) => ListTile(
108     title: Text(
109       event.title,
110     ), // Text
111   ), // ListTile
112 ),
113 ],
114 ), // Column
115 floatingActionButton: FloatingActionButton.extended(
116   onPressed: () => showDialog(
```

lib > calendar.dart > ...

```
115     floatingActionButton: FloatingActionButton.extended(
116       onPressed: () => showDialog(
117         context: context,
118         builder: (context) => AlertDialog(
119           title: Text("Add Event"),
120           content: TextFormField(
121             controller: _eventController,
122           ), // TextFormField
123           actions: [
124             TextButton(
125               child: Text("Cancel"),
126               onPressed: () => Navigator.pop(context),
127             ), // TextButton
128             TextButton(
129               child: Text("Ok"),
130               onPressed: () {
131                 if (_eventController.text.isEmpty) {
132                 } else {
133                   if (selectedEvents[selectedDay] != null) {
134                     selectedEvents[selectedDay].add(
135                       Event(title: _eventController.text),
136                     );
137                   } else {
138                     selectedEvents[selectedDay] = [
139                       Event(title: _eventController.text)
140                     ];
141                   }
142                 }
143                 Navigator.pop(context);
144                 _eventController.clear();
145                 setState(() {});
146                 return;
```

```
147             },
148           ), // TextButton
149         ],
150       ), // AlertDialog
151     ),
152     label: Text("Add Event"),
153     icon: Icon(Icons.add),
154   ), // FloatingActionButton.extended
155 ); // Scaffold
156 }
157 }
158
```

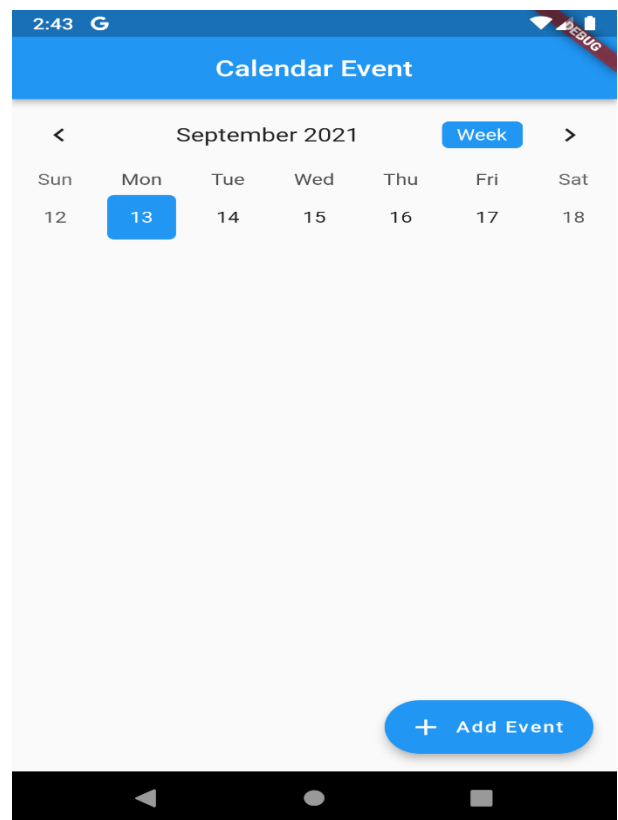
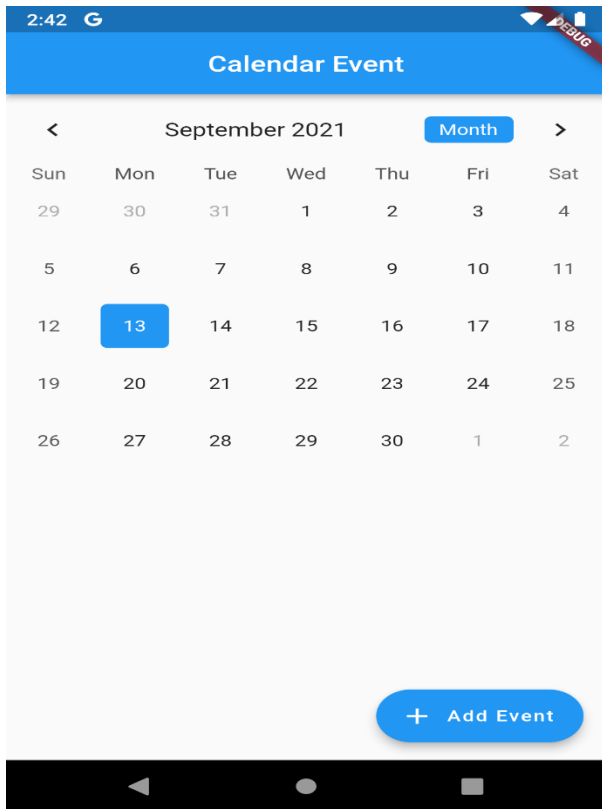
3. Create event.dart file

```
lib > event.dart > Event
1 import 'package:flutter/foundation.dart';
2
3 class Event {
4   final String title;
5   Event({@required this.title});
6   String toString() => this.title;
7 }
8
```

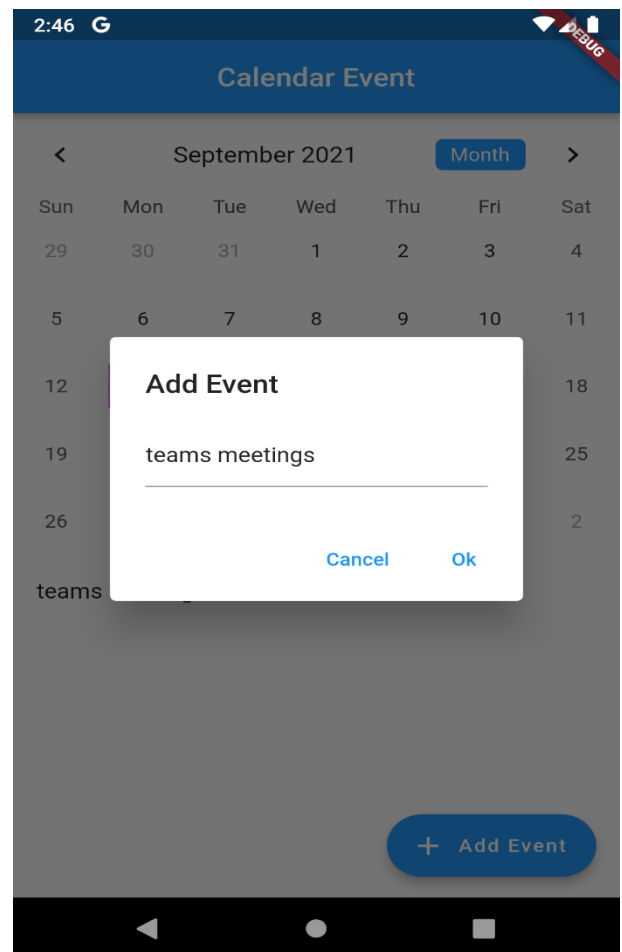
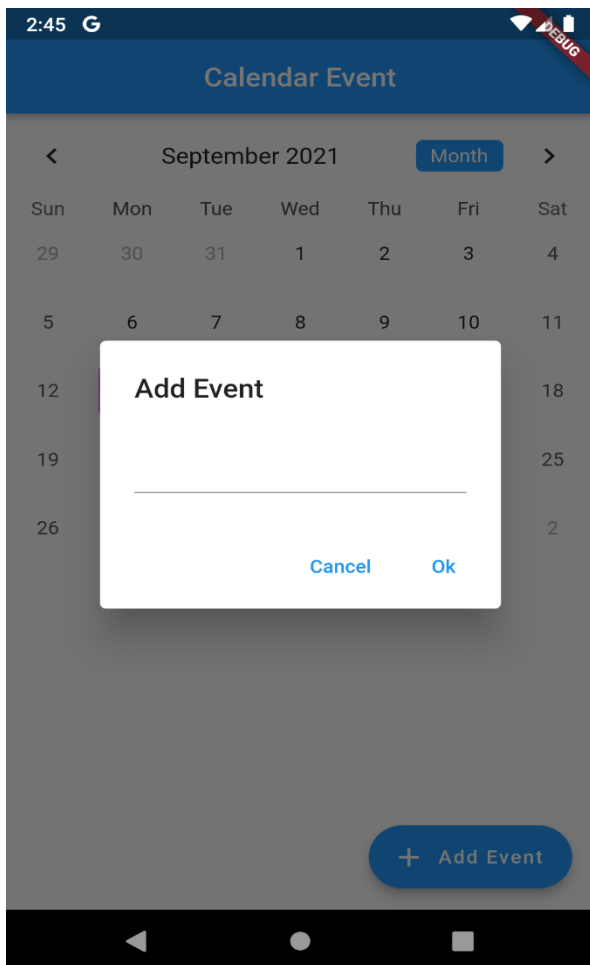
4. In main.dart file

```
lib > main.dart > MyApp > build
1 import 'package:flutter/material.dart';
2 import 'package:calendar_event/calendar.dart';
3
4 void main() {
5   runApp(MyApp());
6 }
7
8 class MyApp extends StatelessWidget {
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(
12      title: 'Calendar Event',
13      home: Calendar(),
14    ); // MaterialApp
15  }
16 }
17
```

Output:



With Events



Calendar Event

<	September 2021					Month	>
Sun	Mon	Tue	Wed	Thu	Fri	Sat	
29	30	31	1	2	3	4	
5	6	7	8	9	10	11	
12	13	14	15	16	17	18	
19	20	21	22	23	24	25	
26	27	28	29	30	1	2	

teams meetings

+ Add Event