

EXPERIMENT – 5

AIM:

- Learn about stateful and stateless widgets.
- Implement state management using `setState` and `Provider`.

DESCRIPTION:

What are Widgets?

Each element on the screen of the Flutter app is a widget. The view of the screen completely depends upon the choice and sequence of the widgets used to build the apps. The structure of the code of apps is a tree of widgets.

Types of Widgets

There are broadly two types of widgets in the flutter:

- Stateless Widget
- Stateful Widget

1. Stateless Widget

Stateless Widget is a type of widget which once built, then it's properties and state can't be changed. These widgets are immutable, once created can't be modified.

Note: These are used for static content or UI content that don't need a change after time.

Key Characteristics of Stateless Widgets are: Immutable, No State and Lightweight.

Examples: Display Text, Icons, Images, etc.

2. Stateful Widget

Stateful Widgets is a type of widget that can change state. It can maintain and update the appearance in the response to change in state.

Note: These are used for dynamic change in the properties and appearance over the time.

Key Characteristics of Stateful Widgets are: Mutable State, State Lifecycle and Dynamic Updates.

Examples: Buttons, Sliders, Text Fields, etc.

Experiment - 5(a)

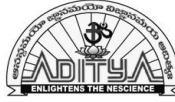
AIM: Learn about stateful and stateless widgets.

SOLUTION:

Stateless Multi-Card Widget

```
import 'package:flutter/material.dart';
void main() {
  runApp(const MyApp());
}
```

```
class MyApp extends StatelessWidget {
  const MyApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
```

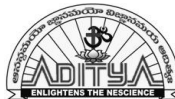


```

home: Scaffold(
  appBar: AppBar(title: const Text('Stateless Multi-Card Example')),
  body: CardListScreen(),
),
);
}
}

class CardListScreen extends StatelessWidget {
  CardListScreen({super.key});
  final List<bool> _cardSelections = List.generate(5, (index) => false);
  @override
  Widget build(BuildContext context) {
    return ListView.builder(
      itemCount: 5,
      itemBuilder: (context, index) {
        return Card(
          margin: const EdgeInsets.all(8.0),
          color: _cardSelections[index] ? Colors.blueAccent.shade100 : Colors.white,
          elevation: _cardSelections[index] ? 8.0 : 2.0,
          child: Padding(
            padding: const EdgeInsets.all(16.0),
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                Text(
                  'Card ${index + 1}',
                  style: const TextStyle(
                    fontSize: 20,
                    fontWeight: FontWeight.bold,
                  ),
                ),
                const SizedBox(height: 8.0),
                Text(
                  _cardSelections[index]
                    ? 'This card is selected!'
                    : 'Tap to select this card.',
                  style: TextStyle(
                    color: _cardSelections[index] ? Colors.white : Colors.black87,
                  ),
                ),
              ],
            ),
          ),
        );
      },
    );
  },
}

```


$$\left. \begin{array}{l} \{ \\ \} \end{array} \right\} \right);$$

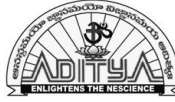
Output:



Stateful Widgets:

Stateful button-widget

```
import 'package:flutter/material.dart';
void main()
{
  runApp(MyApp());
}
class MyApp extends StatelessWidget
{
  @override
  Widget build(BuildContext context)
  {
    return MaterialApp(
      home: CounterApp(),
    );
  }
}
class CounterApp extends StatefulWidget
{
  @override
  _CounterAppState createState() => _CounterAppState();
}
class _CounterAppState extends State<CounterApp>
{
  int _counter = 0;
  void _incrementCounter()
  {
```



```

    setState() {
      _counter++;});
  }
  @override
  Widget build(BuildContext context)
  {
    return
    Scaffold (
      appBar: AppBar(
        title: Text('Counter App'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            Text( 'Counter:',
              style: TextStyle(fontSize: 24),
            ),
            Text( '$_counter',
              style: TextStyle(fontSize: 36, fontWeight: FontWeight.bold),
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed:
          _incrementCounter, tooltip: 'Increment',
        child: Icon(Icons.add),
      ),
    );
  }
}

```

OUTPUT:

