StatefulWidget e StatelessWidget

import 'package:flutter/material.dart';  
  
// Função principal que inicia o app Flutter  
void main() {  
 runApp(const MyApp());  
}  
  
// MyApp é um widget sem estado (Stateless) que serve como o widget raiz da aplicação  
class MyApp extends StatelessWidget {  
 const MyApp({super.key});  
  
 // O método build é responsável por construir o widget e retorna um MaterialApp  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 title: 'Flutter Stateful vs Stateless',  
 theme: ThemeData(  
 // Define o tema do aplicativo com um esquema de cores baseado em uma cor semente  
 colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),  
 useMaterial3: true, // Habilita o uso do Material 3  
 ),  
 home: const MyHomePage(), // Define a página inicial da aplicação  
 );  
 }  
}  
  
// MyHomePage é um widget sem estado (Stateless) que exibe a interface da página inicial  
class MyHomePage extends StatelessWidget {  
 const MyHomePage({super.key});  
  
 // O método build constrói a interface da página, que inclui um AppBar e um corpo central  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 title: const Text('Stateful vs Stateless'), // Título da barra superior (AppBar)  
 ),

body: const Center(  
 // O corpo da página é um Column contendo dois widgets: StatelessWidgetExample e StatefulWidgetExample  
 child: Column(  
 mainAxisAlignment: MainAxisAlignment.center, // Centraliza verticalmente  
 children: <Widget>[  
 StatelessWidgetExample(), // Exibe um widget Stateless  
 StatefulWidgetExample(), // Exibe um widget Stateful  
 ],  
 ),  
 ),  
 );  
 }  
}  
  
// StatelessWidgetExample é um widget sem estado que apenas exibe um texto simples  
class StatelessWidgetExample extends StatelessWidget {  
 const StatelessWidgetExample({super.key});  
  
 // O método build retorna um widget de texto  
 @override  
 Widget build(BuildContext context) {  
 return const Text('Este é um Stateless Widget'); // Exibe a mensagem para o usuário  
 }  
}  
  
// StatefulWidgetExample é um widget com estado, que pode ter sua interface atualizada dinamicamente  
class StatefulWidgetExample extends StatefulWidget {  
 const StatefulWidgetExample({super.key});  
  
 // O método createState retorna a instância do estado associado ao widget  
 @override  
 State<StatefulWidgetExample> createState() => \_StatefulWidgetExampleState();  
}  
  
// \_StatefulWidgetExampleState gerencia o estado do StatefulWidgetExample  
class \_StatefulWidgetExampleState extends State<StatefulWidgetExample> {  
 int \_count = 0; // Variável privada para armazenar o valor do contador

// Método que incrementa o valor do contador e atualiza o estado  
 void \_incrementCount() {  
 setState(() {  
 \_count++; // Atualiza o valor de \_count e notifica o Flutter para redesenhar o widget  
 });  
 }  
  
 // O método build constrói a interface do widget, que inclui um texto e um botão  
 @override  
 Widget build(BuildContext context) {  
 return Column(  
 children: [  
 Text('Este é um Stateful Widget: $\_count'), // Exibe o valor atual do contador  
 ElevatedButton(  
 onPressed: \_incrementCount, // Chama o método para incrementar o contador  
 child: const Text('Incrementar'), // Texto do botão  
 ),  
 ],  
 );  
 }  
}