import 'package:flutter/material.dart';  
  
void main() {  
 runApp(const MyApp());  
}  
  
class MyApp extends StatelessWidget {  
 const MyApp({super.key});  
  
 // This widget is the root of your application.  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 title: 'Flutter Demo',  
 theme: ThemeData(  
 // This is the theme of your application.  
 //  
 // TRY THIS: Try running your application with "flutter run". You'll see  
 // the application has a purple toolbar. Then, without quitting the app,  
 // try changing the seedColor in the colorScheme below to Colors.green  
 // and then invoke "hot reload" (save your changes or press the "hot  
 // reload" button in a Flutter-supported IDE, or press "r" if you used  
 // the command line to start the app).  
 //  
 // Notice that the counter didn't reset back to zero; the application  
 // state is not lost during the reload. To reset the state, use hot  
 // restart instead.  
 //  
 // This works for code too, not just values: Most code changes can be  
 // tested with just a hot reload.  
 colorScheme: ColorScheme.fromSeed(seedColor: Colors.*deepPurple*),  
 useMaterial3: true,  
 ),  
 home: const MyHomePage(title: 'Flutter Demo Home Page'),  
 );  
 }  
}  
  
class MyHomePage extends StatefulWidget {  
 const MyHomePage({super.key, required this.title});  
  
 // This widget is the home page of your application. It is stateful, meaning  
 // that it has a State object (defined below) that contains fields that affect  
 // how it looks.  
  
 // This class is the configuration for the state. It holds the values (in this  
 // case the title) provided by the parent (in this case the App widget) and  
 // used by the build method of the State. Fields in a Widget subclass are  
 // always marked "final".  
  
 final String title;  
  
 @override  
 State<MyHomePage> createState() => \_MyHomePageState();  
}  
  
class \_MyHomePageState extends State<MyHomePage> {  
 int \_counter = 0;  
  
 void \_incrementCounter() {  
 setState(() {  
 // This call to setState tells the Flutter framework that something has  
 // changed in this State, which causes it to rerun the build method below  
 // so that the display can reflect the updated values. If we changed  
 // \_counter without calling setState(), then the build method would not be  
 // called again, and so nothing would appear to happen.  
 \_counter++;  
 });  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return SafeArea(  
 child: Scaffold(  
 appBar: AppBar(  
 elevation: 200,  
 backgroundColor: Colors.*greenAccent*,  
 title: Text(  
 '🛍️ List of Fruits',  
 style: TextStyle(  
 color: Colors.*white*,  
 fontSize: 30,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 centerTitle: true,  
 ),  
 body: Center(  
 child: Padding(  
 padding: const EdgeInsets.all(50.0),  
 child: Column(  
 // mainAxisAlignment: MainAxisAlignment.start,  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 children: [  
 Row(  
 children: [  
 Text(  
 '🍎',  
 // textAlign: TextAlign.start,  
 style: TextStyle(  
 fontSize: 35,  
 color: Colors.*red*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Apple',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*red*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 ],  
 ),  
  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍇',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Greps',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*purpleAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍒',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*purpleAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Cherry',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*purpleAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍓',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*pink*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Strawberry',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*pink*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🥭',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*yellow*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Mango',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*yellow*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍍',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*greenAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Pineapple',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*greenAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍋',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*yellowAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Lemon',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*yellowAccent*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🍉',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*lightGreen*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
 Text(  
 'Watermelon',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*lightGreen*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
  
 ],  
 ),  
 Row(  
 // crossAxisAlignment: CrossAxisAlignment.center,  
 // mainAxisAlignment: MainAxisAlignment.start,  
 children: [  
 Text(  
 '🥥',  
 // textAlign: TextAlign.center,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*brown*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
 SizedBox(width: 20),  
  
 Text(  
 'Coconut',  
 // textAlign: TextAlign.start,  
 style: TextStyle(  
 fontSize: 35,  
 letterSpacing: 3,  
 color: Colors.*brown*,  
 fontWeight: FontWeight.*bold* ),  
 ),  
  
 ],  
 ),  
 ],  
 ),  
 ),  
 ),  
 ),  
 );  
 }  
}

