

Lập trình Mobile



Tuần 4

Giảng viên: Trần Đức Minh

Nội dung bài giảng



- Chương trình đầu tiên
- Stateless Widget

Chương trình đầu tiên



```
import 'package:flutter/material.dart';
```

```
void main() => runApp(  
  Center(  
    child: Text(  
      'Lập trình Mobile !!!',  
      textDirection: TextDirection.ltr  
    ),  
  ),  
);
```

Chương trình đầu tiên



- **Material Design** là một hệ thống thiết kế nguồn mở được xây dựng và hỗ trợ bởi các nhà thiết kế và phát triển của Google.
 - Tài liệu hỗ trợ Material
 - <https://m3.material.io/develop/flutter>
- Hàm **runApp()** có nhiệm vụ hiển thị Widget tree lên màn hình.
 - Cụ thể ta chỉ cần gửi nút gốc của widget tree vào làm tham số của runApp()

Stateless Widget



- Stateless widget (widget có dạng stateless) không thể thay đổi trạng thái của nó trong khi chạy ứng dụng. Tức là không thể vẽ lại giao diện.
- Stateless widget hữu ích đối với phần giao diện người dùng không phụ thuộc vào widget nào khác.

Stateless Widget



- Để tạo một Widget có dạng Stateless ta cho nó kế thừa từ lớp **StatelessWidget**
- Ví dụ:

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Container(  
      decoration: const BoxDecoration(color: Colors.red),  
      child: const Center(  
        child: Text(  
          'Mobile Programming',  
          textDirection: TextDirection.ltr,  
          style: TextStyle(  
            fontSize: 48,  
            color: Colors.black, ), ), ), );  
    }  
  }  
}
```

Stateless Widget



- Phương thức build
 - Được dùng để sinh ra giao diện đồ họa.
 - Mỗi Widget đều có phương thức build của riêng mình.
 - Đối số của phương thức build có kiểu là **BuildContext**
 - Đối số này cho biết widget hiện tại đang nằm ở đâu (tham chiếu đến vị trí) trên widget tree.
 - Phương thức build trả về một widget.

Stateless Widget



- **MaterialApp** class
 - Tạo ra ứng dụng sử dụng thiết kế Material
- Ví dụ:

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(home : Text('Hello'),);  
  }  
}
```


Stateless Widget



- **Scaffold** là một widget được sử dụng để triển khai cấu trúc bố cục hình ảnh material design cơ bản.
- Scaffold cung cấp nhiều widget hoặc API cho ứng dụng như Drawer, SnackBar, BottomNavigationBar, AppBar, FloatingActionButton, v.v.

- Ví dụ:

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Material App',  
      home: Scaffold(  
        appBar: AppBar(  
          title: const Text('Scaffold title'),  
        ),  
        body: const Center(  
          child: Text('Lập trình mobile'),  
        ), ), );  
  }  
}
```

Stateless Widget



- Scaffold
 - appBar
 - Là một thanh ngang được hiển thị ở trên cùng của widget Scaffold. Ta có thể tùy chỉnh các thuộc tính khác nhau của appBar như độ cao, tiêu đề, độ sáng, v.v.
 - body
 - Là thuộc tính chính và bắt buộc của Scaffold, body sẽ hiển thị nội dung chính trong Scaffold.
 - body hiển thị bên dưới appBar và phía sau floatActionButton & drawer.

Stateless Widget

- Scaffold

- Drawer: là một bảng điều khiển được hiển thị ở bên cạnh body. Thông thường, nó bị ẩn, nhưng người dùng có thể mở nó bằng cách vuốt hoặc nhấn biểu tượng.

```
drawer: Drawer(  
  child: ListView(  
    children: const <Widget>[  
      DrawerHeader(  
        decoration: BoxDecoration(  
          color: Colors.blue,  
        ), // BoxDecoration  
        child: Text(  
          'Day la Drawer',  
          style: TextStyle(  
            color: Colors.yellow,  
            fontSize: 26,  
          ), // TextStyle  
        ), // Text  
      ), // DrawerHeader  
      ListTile(  
        title: Text('Trang chủ'),  
      ), // ListTile  
      ListTile(  
        title: Text("All Mail Inboxes"),  
        leading: Icon(Icons.mail),  
      ), // ListTile  
      Divider(  
        height: 3,  
      ), // Divider  
      ListTile(  
        title: Text("Dòng thứ 1"),  
      ), // ListTile  
    ], // <Widget>[]  
  ), // ListView  
) // Drawer
```

Stateless Widget



- Scaffold
 - persistentFooterButtons: Đó là danh sách các nút được hiển thị ở cuối Scaffold. Nó luôn được bao bọc trong một widget ButtonBar.

```
persistentFooterButtons: <Widget>[  
  ElevatedButton(  
    onPressed: () {},  
    style: ElevatedButton.styleFrom(  
      primary: Colors.green, // background  
      onPrimary: Colors.white, // foreground  
    ),  
    child: const Icon(  
      Icons.add,  
      color: Colors.white,  
    ), // Icon  
  ), // ElevatedButton  
  ElevatedButton(  
    onPressed: () {},  
    style: ElevatedButton.styleFrom(  
      primary: Colors.green, // background  
      onPrimary: Colors.white, // foreground  
    ),  
    child: const Icon(  
      Icons.clear,  
      color: Colors.white,  
    ), // Icon  
  ), // ElevatedButton  
], // <Widget>[]
```

Stateless Widget



- Scaffold

- FloatingActionButton: Là nút hiển thị ở góc dưới cùng bên phải và nổi phía trên phần thân.

- Ví dụ:

```
floatingActionButton: FloatingActionButton(  
    elevation: 8.0,  
    child: Icon(Icons.add),  
    onPressed: () {  
        print('Floating Button vừa được nhấn');  
    },  
),
```

Stateless Widget



- Scaffold
 - BottomNavigationBar:
 - Menu hiển thị thanh điều hướng ở cuối Scaffold.
 - Được hiển thị bên dưới phần thân và persistentFooterButtons.

```
bottomNavigationBar: BottomNavigationBar(  
  currentIndex: 0,  
  fixedColor: Colors.grey,  
  items: const [  
    BottomNavigationBarItem(  
      label: "Home",  
      icon: Icon(Icons.home),  
    ), // BottomNavigationBarItem  
    BottomNavigationBarItem(  
      label: "Search",  
      icon: Icon(Icons.search),  
    ), // BottomNavigationBarItem  
    BottomNavigationBarItem(  
      label: "User Profile",  
      icon: Icon(Icons.account_circle),  
    ), // BottomNavigationBarItem  
  ],  
), // BottomNavigationBar
```

Hết Tuần 4



Cảm ơn các bạn đã chú ý lắng nghe !!!