**Flutter中的普通路由、普通路由传值、命名路由、命名路由传值**

**主讲教师：（大地）**

**合作网站：[www.itying.com](http://www.itying.com) （IT营）**

**我的专栏：https://www.itying.com/category-79-b0.html （IT营）**

[一、 Flutter 中的普通路由 1](#_Toc29495)

[二、 Flutter 中的基本路由使用 1](#_Toc26291)

[三、 Flutter 中的基本路由跳转传值 2](#_Toc15827)

[四、 Flutter 中的命名路由 3](#_Toc32297)

[五、 Flutter 中的命名路由跳转传值 4](#_Toc7351)

[六、 Vscode调试Flutter项目 6](#_Toc20549)

# **Flutter 中的路由**

Flutter中的路由通俗的讲就是页面跳转。在Flutter中通过Navigator组件管理路由导航。

并提供了管理堆栈的方法。如：Navigator.push和Navigator.pop

Flutter中给我们提供了两种配置路由跳转的方式：1、基本路由 2、命名路由

# **Flutter 中的基本路由使用**

比如我们现在想从HomePage组件跳转到SearchPage组件。

1. **需要在HomPage中引入SearchPage.dart**

|  |
| --- |
| import '../SearchPage.dart'; |

1. **在HomePage中通过下面方法跳转**

|  |
| --- |
| RaisedButton(  child: Text("跳转到搜索页面"),  onPressed: (){  Navigator.of(context).push(  MaterialPageRoute(  builder: (BuildContext context){  return SearchPage();  }  )  );    },  color: Theme.of(context).accentColor,  textTheme: ButtonTextTheme.primary  ) |

# **Flutter 中的基本路由跳转传值**

比如我们现在想从HomePage组件跳转到SearchPage组件传值。

**1、需要在HomPage中引入SearchPage.dart**

|  |
| --- |
| import '../SearchPage.dart'; |

**2、在HomePage中通过下面方法跳转**

|  |
| --- |
| RaisedButton(  child: Text("跳转到搜索页面"),  onPressed: (){  Navigator.of(context).push(  MaterialPageRoute(  builder: (BuildContext context){  return SearchPage(title:"表单"); //传值  }  )  );    },  color: Theme.of(context).accentColor,  textTheme: ButtonTextTheme.primary  ) |

# **Flutter 中的命名路由**

1. **配置路由**

|  |
| --- |
| import 'package:flutter/material.dart';  import 'pages/Tabs.dart';  import 'pages/Search.dart';  import 'pages/Form.dart';  void main() => runApp(MyApp());  class MyApp extends StatelessWidget {  @override  Widget build(BuildContext context) {  return MaterialApp(  // home:Tabs(),  initialRoute: '/',  routes: {  '/':(contxt)=>Tabs(),  '/search':(contxt) =>SearchPage(),  '/form': (context) => FormPage(),  },  );  }  } |

1. **路由跳转**

|  |
| --- |
| RaisedButton(  child: Text("跳转到搜索页面"),  onPressed: (){  Navigator.pushNamed(context, '/search');  },  color: Theme.of(context).accentColor,  textTheme: ButtonTextTheme.primary  ) |

# **Flutter 中的命名路由跳转传值**

官方文档：<https://flutter.dev/docs/cookbook/navigation/navigate-with-arguments>

1. **配置路由：**

|  |
| --- |
| import 'package:flutter/material.dart';  import 'pages/Tabs.dart';  import 'pages/Search.dart';  import 'pages/Form.dart';  void main() => runApp(MyApp());  class MyApp extends StatelessWidget {  final routes = {  '/': (contxt) => Tabs(),  '/search': (contxt) => SearchPage(),  '/form': (context, {arguments}) => FormPage(arguments: arguments),  };  @override  Widget build(BuildContext context) {  return MaterialApp(  home: Tabs(),  onGenerateRoute: (RouteSettings settings) {  // 统一处理  final String name = settings.name;  final Function pageContentBuilder = this.routes[name];  if (pageContentBuilder != null) {  if (settings.arguments != null) {  final Route route = MaterialPageRoute(  builder: (context) => pageContentBuilder(context,  arguments: settings.arguments));  return route;  } else {  final Route route = MaterialPageRoute(  builder: (context) => pageContentBuilder(context));  return route;  }  }  });  }  } |

1. **跳转传值**

|  |
| --- |
| RaisedButton(  child: Text("跳转到表单演示页面"),  onPressed: (){  Navigator.pushNamed(context, '/form',arguments: {  "id":20  });  },  color: Theme.of(context).accentColor,  textTheme: ButtonTextTheme.primary  ) |

1. **接收参数**

|  |
| --- |
| import 'package:flutter/material.dart';  class FormPage extends StatelessWidget {  final Map arguments;  FormPage({this.arguments});    @override  Widget build(BuildContext context) {  return Scaffold(  appBar: AppBar(  title: Text("搜索"),  ),  body:Text("我是一个表单页面 ${arguments != null ? arguments['id'] : '0'}")  );  }  } |

# **Flutter 中的命名路单独抽离到一个文件**

|  |
| --- |
| import 'package:flutter/material.dart';  import '../pages/Tabs.dart';  import '../pages/Search.dart';  import '../pages/Form.dart';  final Map<String, Function> routes = {  '/':(contxt,{arguments})=>Tabs(),  '/search':(contxt,{arguments}) =>SearchPage(arguments: arguments),  '/form': (context,{arguments}) =>FormPage(arguments: arguments),  };  var onGenerateRoute=(RouteSettings settings) {  // 统一处理  final String name = settings.name;  final Function pageContentBuilder = routes[name];  if (pageContentBuilder != null) {  final Route route = MaterialPageRoute(  builder: (context) =>  pageContentBuilder(context, arguments: settings.arguments));  return route;  }  }; |

|  |
| --- |
| import 'package:flutter/material.dart';  import 'routes/Routes.dart';  void main() => runApp(MyApp());  class MyApp extends StatelessWidget {  @override  Widget build(BuildContext context) {  return MaterialApp(  // home:Tabs(),  initialRoute: '/',  onGenerateRoute: onGenerateRoute  );  }  } |

# Vscode调试Flutter项目

1. **Vscode中打开flutter项目进行开发**
2. **运行Flutter项目**

|  |
| --- |
| flutter run |

**r 键**：点击后热加载，也就算是重新加载吧。

**p 键**：显示网格，这个可以很好的掌握布局情况，工作中很有用。

**o 键**：切换android和ios的预览模式。

**q 键**：退出调试预览模式。

1. **Vscode默认连不上第三方模拟器解决方案**

cd到对应夜神模拟器D:\Program Files\Nox\bin目录 然后运行

|  |
| --- |
| nox\_adb.exe connect 127.0.0.1:62001 |