

Flutter 页面布局 Wrap 组件

主讲教师：（大地）

合作网站：www.itying.com （IT 营）

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

一、Flutter RaisedButton 定义一个按钮.....	1
二、Wrap 组件.....	1
三、Vscode 调试 Flutter 项目.....	4

一、Flutter RaisedButton 定义一个按钮

Flutter 中通过 RaisedButton 定义一个按钮。RaisedButton 里面有很多的参数，这一讲我们只是简单的进行使用。

```
return RaisedButton(  
  child: Text('女装'),  
  textColor: Theme.of(context).accentColor,  
  onPressed: ()  
  },  
);
```

二、Wrap 组件

Wrap 可以实现流布局，单行的 Wrap 跟 Row 表现几乎一致，单列的 Wrap 则跟 Row 表现几乎一致。但 Row 与 Column 都是单行单列的，Wrap 则突破了这个限制，mainAxis 上空间不足时，则向 crossAxis 上去扩展显示。

属性	说明
direction	主轴的方向，默认水平
alignment	主轴的对其方式
spacing	主轴方向上的间距
textDirection	文本方向
verticalDirection	定义了 children 摆放顺序，默认是 down，见 Flex 相关属性介绍。
runAlignment	run 的对齐方式。run 可以理解为新的行或者列，如果是水平方向布局的话，run 可以理解为新的一行
runSpacing	run 的间距



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

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    // TODO: implement build
  }
}
```

```
return MaterialApp(  
  home: Scaffold(  
    appBar: AppBar(title: Text('FlutterDemo')),  
    body: LayoutDemo(),  
  ));  
}  
}  
class LayoutDemo extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    // TODO: implement build  
    return Wrap(  
      spacing: 10,  
      runSpacing: 10,  
      alignment: WrapAlignment.spaceEvenly,  
  
      children: <Widget>[  
        MyButton("第 1 集"),  
        MyButton("第 2 集"),  
        MyButton("第 3 集"),  
        MyButton("第 4 集"),  
        MyButton("第 5 集"),  
        MyButton("第 6 集第 6 集"),  
        MyButton("第 7 集"),  
        MyButton("第 8 集第 6 集"),  
        MyButton("第 9 集"),  
        MyButton("第 10 集"),  
        MyButton("第 11 集"),  
      ],  
    );  
  }  
}  
  
class MyButton extends StatelessWidget{  
  
  final String text;  
  
  const MyButton(this.text,{Key key}) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    // TODO: implement build  
    return RaisedButton(  

```

```
child: Text(this.text),  
textColor: Theme.of(context).accentColor,  
onPressed: (){  
  },  
);  
}  
}
```

三、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。

3、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
```