



微信交流群



点击购买

Flexible

Expanded、Flexible和Spacer都是具有权重属性的组件，可以控制Row、Column、Flex的子控件如何布局的控件。

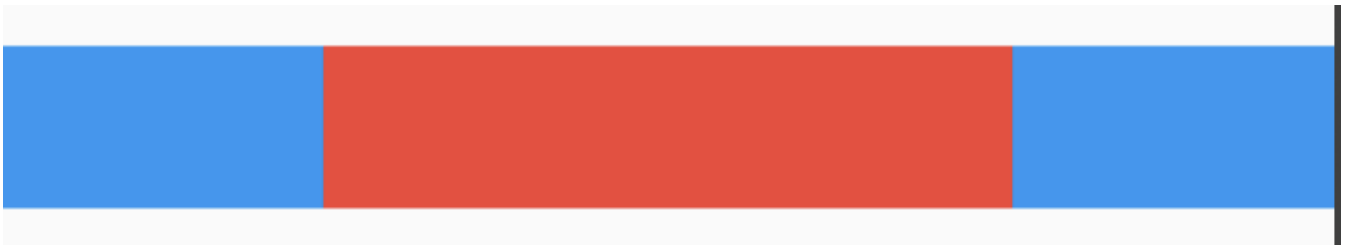
Flexible

Flexible组件可以控制Row、Column、Flex的子控件占满父控件，比如，Row中有3个子控件，2边的固定宽，中间的占满剩余的空间，代码如下：

```
1 Row(  
2     children: <Widget>[  
3         Container(  
4             color: Colors.blue,  
5             height: 50,  
6             width: 100,  
7         ),
```

```
8      Flexible(  
9        child: Container(  
10          color: Colors.red,  
11          height: 50,  
12        )  
13      ),  
14      Container(  
15        color: Colors.blue,  
16        height: 50,  
17        width: 100,  
18      ),  
19    ],  
20  )
```

效果如图：



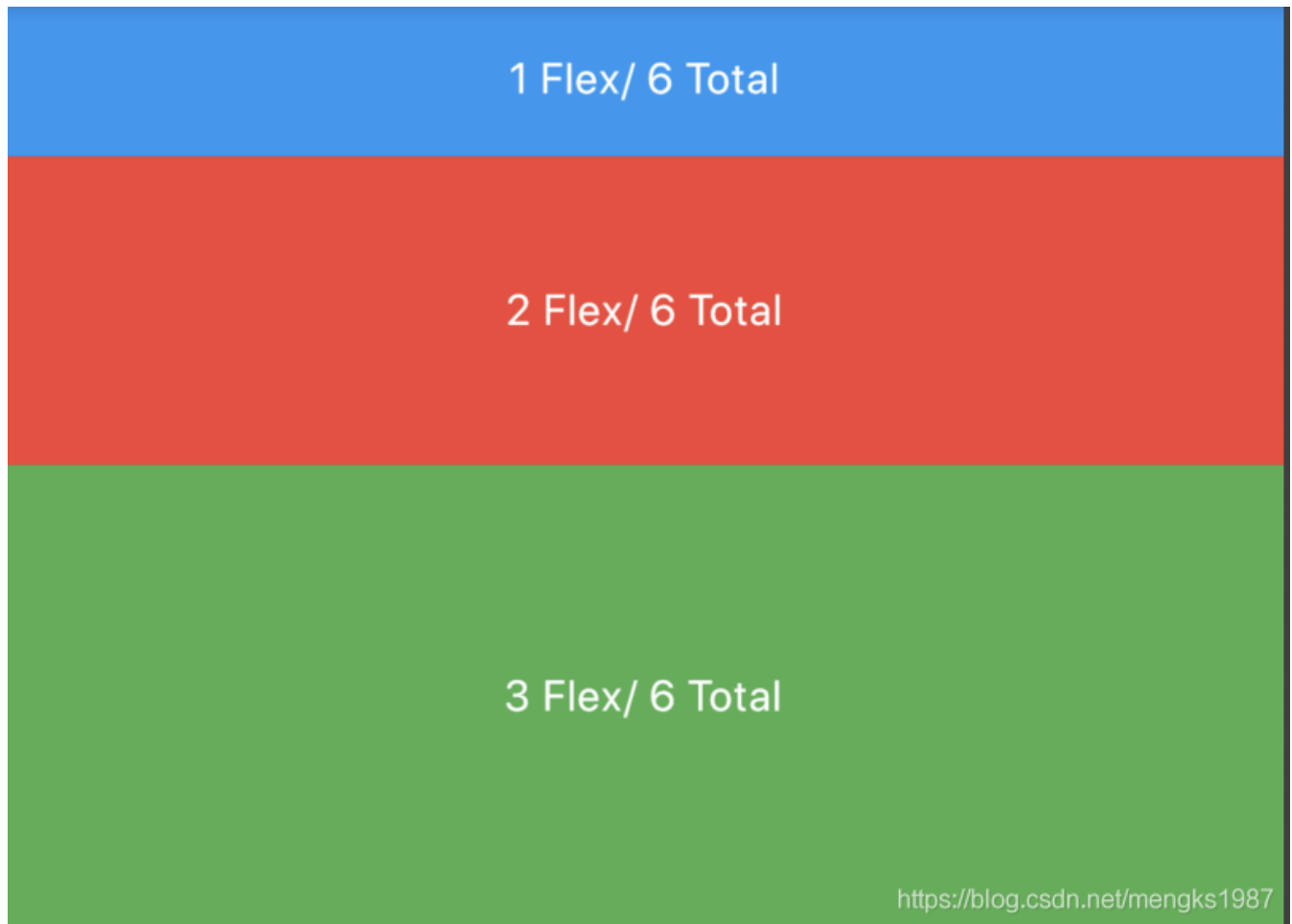
还是有3个子控件，希望第一个占1/6，第二个占2/6，第三个占3/6，代码如下：

```
1  Column(  
2    children: <Widget>[  
3      Flexible(  
4        flex: 1,  
5        child: Container(  
6          color: Colors.blue,  
7          alignment: Alignment.center,  
8          child: Text('1 Flex/ 6 Total', style: TextStyle(color: Color  
9        ),  
10     ),  
11     Flexible(  
12       flex: 2,  
13       child: Container(  
14         color: Colors.red,  
15         alignment: Alignment.center,  
16         child: Text('2 Flex/ 6 Total', style: TextStyle(color: Color  
17       ),  
18     ),  
19     Flexible(  
20       flex: 3,  
21       child: Container(  

```

```
22         color: Colors.green,  
23         alignment: Alignment.center,  
24         child: Text('3 Flex/ 6 Total',style: TextStyle(color: Color  
25             ),  
26     ),  
27 ],  
28 )
```

效果如图：



子控件占比 = 当前子控件flex/所有子控件flex之和。

Flexible中 **fit** 参数表示填满剩余空间的方式，说明如下：

- tight：必须（强制）填满剩余空间。
- loose：尽可能大的填满剩余空间，但是可以不填满。

这2个看上去不是很好理解啊，什么叫尽可能大的填满剩余空间？什么时候填满？看下面的例子：

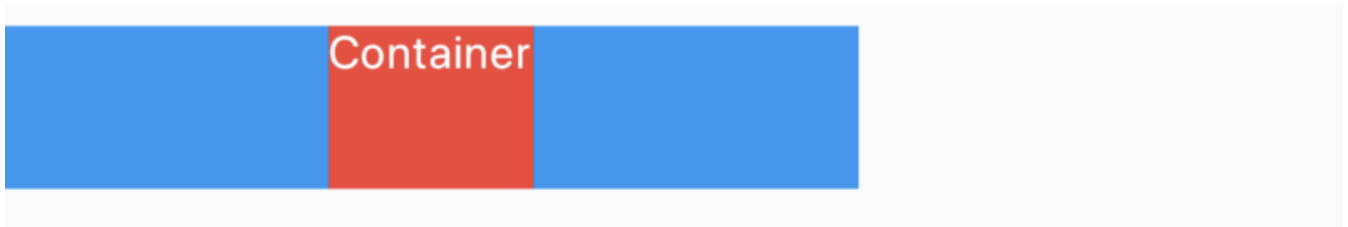
```
1 Row(  
2     children: <Widget>[  
3         Container(  
4             child: Text('3 Flex/ 6 Total',style: TextStyle(color: Color  
5                 ),  
6         ),  
7     ],  
8 )
```

```

4         color: Colors.blue,
5         height: 50,
6         width: 100,
7     ),
8     Flexible(
9         child: Container(
10            color: Colors.red,
11            height: 50,
12            child: Text('Container',style: TextStyle(color: Colors.wh
13        )
14    ),
15    Container(
16        color: Colors.blue,
17        height: 50,
18        width: 100,
19    ),
20 ],
21 )

```

效果图：



这段代码是在最上面代码的基础上给中间红色Container添加了Text子控件，此时红色Container就不在充满空间，再给Container添加对齐方式，代码如下：

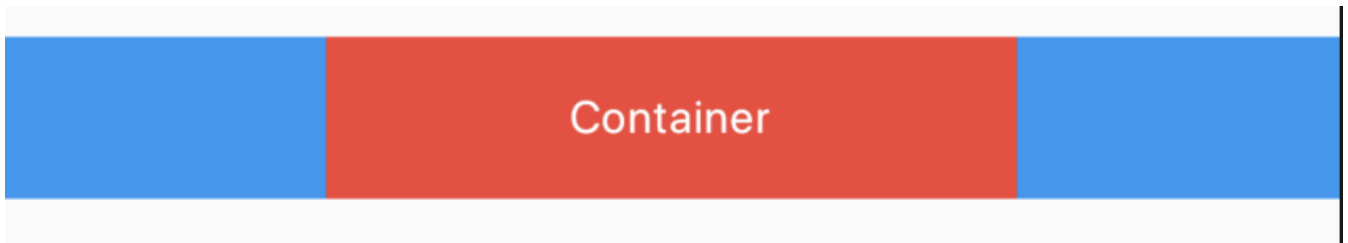
```

1 Row(
2     children: <Widget>[
3         Container(
4             color: Colors.blue,
5             height: 50,
6             width: 100,
7         ),
8         Flexible(
9             child: Container(
10                color: Colors.red,
11                height: 50,
12                alignment: Alignment.center,
13                child: Text('Container',style: TextStyle(color: Colors.whi
14            )

```

```
15     ),
16     Container(
17       color: Colors.blue,
18       height: 50,
19       width: 100,
20     ),
21   ],
22 )
```

效果图：



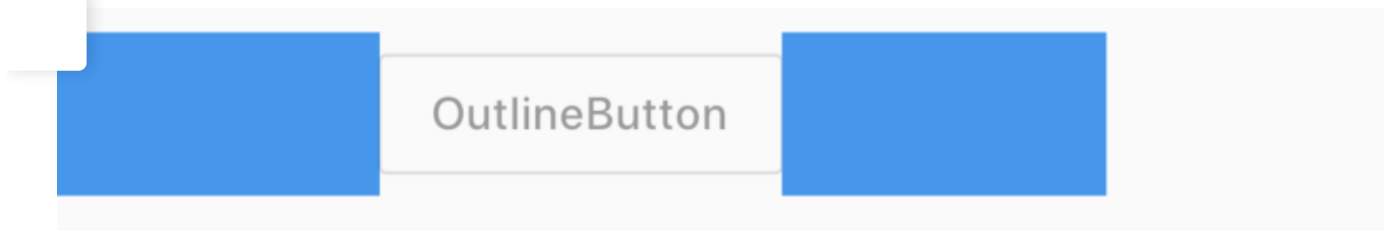
此时又填满剩余空间。

大家是否还记得Container控件的大小是调整的吗？Container默认是适配子控件大小的，但当设置对齐方式时Container将会填满父控件，在[Flutter Widgets 之 Container](#)中已经详细介绍，因此是否填满剩余空间取决于子控件是否需要填满父控件。

如果把Flexible中子控件由Container改为OutlineButton，代码如下：

```
1   Row(
2     children: <Widget>[
3       Container(
4         color: Colors.blue,
5         height: 50,
6         width: 100,
7       ),
8       Flexible(
9         child: OutlineButton(
10          child: Text('OutlineButton'),
11        ),
12      ),
13      Container(
14        color: Colors.blue,
15        height: 50,
16        width: 100,
17      ),
18    ],
19  )
```

OutlineButton正常情况下是不充满父控件的，因此最终的效果应该是不填满剩余空间，效果如图：



Expanded

看一下Expanded的源代码：

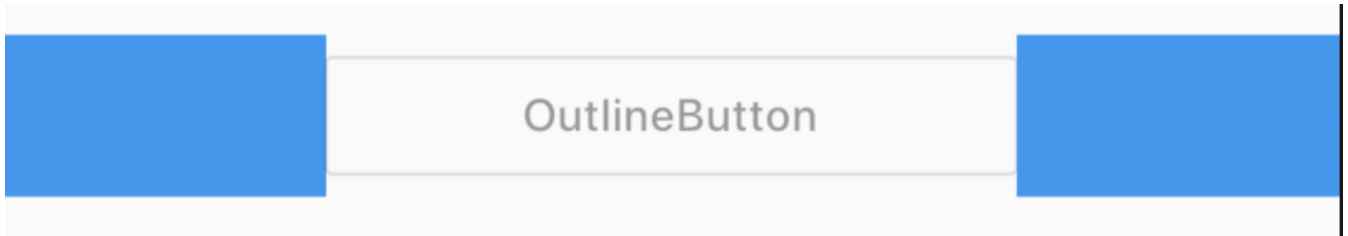
```
1 class Expanded extends Flexible {
2   /// Creates a widget that expands a child of a [Row], [Column], or [Fl
3   /// so that the child fills the available space along the flex widget'
4   /// main axis.
5   const Expanded({
6     Key key,
7     int flex = 1,
8     @required Widget child,
9   }) : super(key: key, flex: flex, fit: FlexFit.tight, child: child);
10 }
```

Expanded继承自Flexible，`fit` 参数固定为 `FlexFit.tight`，也就是说Expanded必须（强制）填满剩余空间。上面的OutlineButton想要充满剩余空间可以使用Expanded：

```
1 Row(
2   children: <Widget>[
3     Container(
4       color: Colors.blue,
5       height: 50,
6       width: 100,
7     ),
8     Expanded(
9       child: OutlineButton(
10        child: Text('OutlineButton'),
11      ),
12    ),
13    Container(
14      color: Colors.blue,
```

```
15         height: 50,  
16         width: 100,  
17       ),  
18     ],  
19   )
```

效果如图：



Spacer

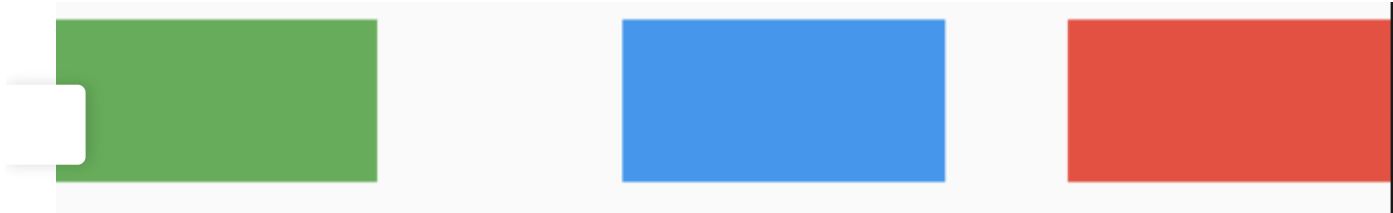
看下Spacer的build源代码：

```
1  @override  
2  Widget build(BuildContext context) {  
3    return Expanded(  
4      flex: flex,  
5      child: const SizedBox.shrink(),  
6    );  
7  }
```

Spacer是通过Expanded的实现的，和Expanded的区别是：Expanded可以设置子控件，而Spacer的子控件尺寸是0，因此Spacer适用于撑开Row、Column、Flex的子控件的空隙，用法如下：

```
1  Row(  
2    children: <Widget>[  
3      Container(width: 100,height: 50,color: Colors.green,),  
4      Spacer(flex: 2,),  
5      Container(width: 100,height: 50,color: Colors.blue,),  
6      Spacer(),  
7      Container(width: 100,height: 50,color: Colors.red,),  
8    ],  
9  )
```

效果如下：



总结

总结如下：

- Spacer是通过Expanded来实现的，Expanded继承自Flexible。
- 填满剩余空间直接使用Expanded更方便。
- Spacer用于撑开Row、Column、Flex的子控件的空隙。

喜欢作者

版权所有，禁止私自转发、克隆网站。

[< ExpandIcon](#)

[ExpansionPanelList >](#)