

# flutter第四天

## flutter第四天

### 一、Wrap组件的使用

- 1.1 什么是Wrap组件?
- 1.2 Wrap组件的属性与方法?
- 1.3 完成如下效果

### 二、GridView组件的使用

- 2.1 什么是GridView组件?
- 2.2 GridView组件的属性与方法?
- 2.3 使用GridView.count实现网格布局
- 2.4 使用GridView.builder实现网格布局

### 三、AspectRatio组件的使用

- 3.1 什么是AspectRatio组件?
- 3.2 AspectRatio组件的属性与方法?

### 四、Card组件的使用

- 4.1 什么是Card组件?
- 4.2 Card组件的属性与方法?
- 4.3 完成如下示例
- 4.4 使用Card组件实现图文列表?

### 五、Stack组件与Align组件和Positioned组件实现定位布局

- 5.1 什么是Stack组件?
- 5.2 Stach组件的属性与方法?
- 5.3 Align组件的使用?
- 5.4 Align组件的属性与方法
- 5.5 Positioned组件的使用?
- 5.6 Positioned组件的属性与方法?

### 六、项目实战

- 6.1 项目演示
- 6.2 创建项目
- 6.3 制作顶部标题栏
- 6.4 制作底部TabBar
- 6.5 制作底部TabBar页面切换效果
- 6.6 制作首页轮播图效果
- 6.7 制作猜你喜欢和推荐商品标题模块以及封装
- 6.8 制作猜你喜欢商品列表左右滑动效果
- 6.9 制作推荐商品列表模块

## 一、Wrap组件的使用

---

### 1.1 什么是Wrap组件?

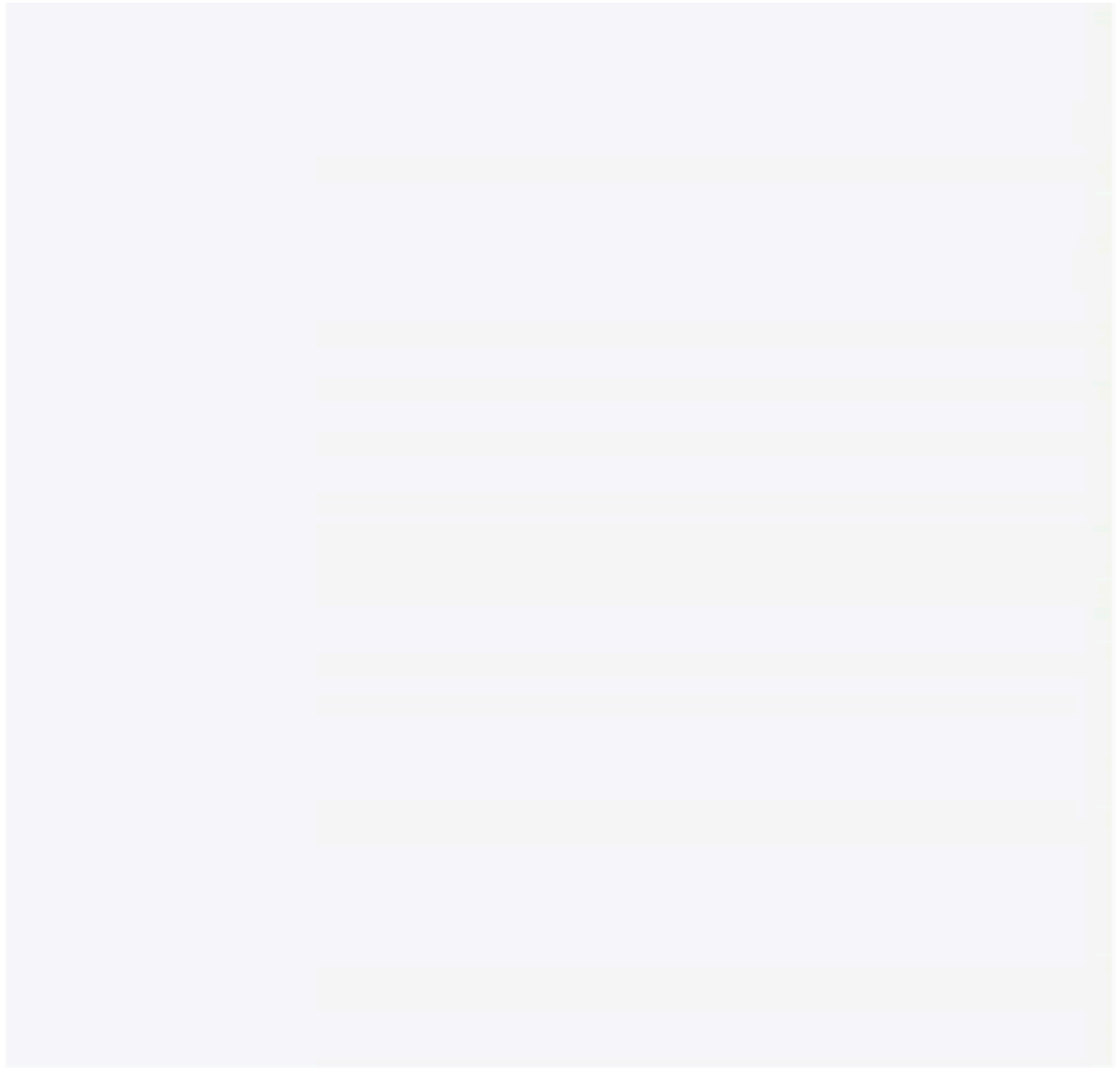
Wrap 可以实现流布局，单行的 Wrap 跟 Row 表现几乎一致，单列的 Wrap 则跟 Column 表现几乎一致。但 Row 与 Column 都是单行单列的，Wrap 则突破了这个限制，mainAxis 上空间不足时，则向 crossAxis 上去扩展显示

### 1.2 Wrap组件的属性与方法?

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

### 1.3 完成如下效果





## 二、GridView组件的使用

---

### 2.1 什么是GridView组件？

当数据量很大的时候用矩阵方式排列比较清晰。此时我们可以用网格列表组件 GridView 实现布局。

*\*GridView\* \*创建网格列表有多种方式，下面我们主要介绍两种。\**

- 1、可以通过 GridView.count 实现网格布局
- 2、通过 GridView.builder 实现网格布局

### 2.2 GridView组件的属性与方法？

名称	类型	说明
<u>scrollDirection</u>	Axis	滚动方法
padding	<u>EdgeInsetsGeometry</u>	内边距
resolve	<u>bool</u>	组件反向排序
<u>crossAxisSpacing</u>	double	水平子 Widget 之间间距
<u>mainAxisSpacing</u>	double	垂直子 Widget 之间间距
<u>crossAxisCount</u>	int	一行的 Widget 数量
<u>childAspectRatio</u>	double	子 Widget 宽高比例
children		<Widget>[ ]
<u>gridDelegate</u>	<u>SliverGridDelegateWithFixedCrossAxisCount</u> (常用) <u>SliverGridDelegateWithMaxCrossAxisExtent</u>	控制布局主要用在 <u>GridView.builder</u> 里面

## 2.3 使用GridView.count实现网格布局

## 2.4 使用GridView.builder实现网格布局

```
1 gridDelegate:SliverGridDelegateWithFixedCrossAxisCount
```

# 三、AspectRatio组件的使用

## 3.1 什么是AspectRatio组件?

AspectRatio 的作用是根据设置调整子元素 child 的宽高比

AspectRatio 首先会在布局限制条件允许的范围内尽可能的扩展，widget 的高度是由宽度和比率决定的，类似于 BoxFit 中的 contain，按照固定比率去尽量占满区域。

如果在满足所有限制条件过后无法找到一个可行的尺寸，AspectRatio 最终将会去优先适应布局限制条件，而忽略所设置的比率。

## 3.2 AspectRatio组件的属性与方法?

*属性*	*说明*
aspectRatio	宽高比，最终可能不会根据这个值去布局，具体则要看综合因素，外层是否允许按照这种比率进行布局，这只是一个参考值
child	子组件

## 四、Card组件的使用

---

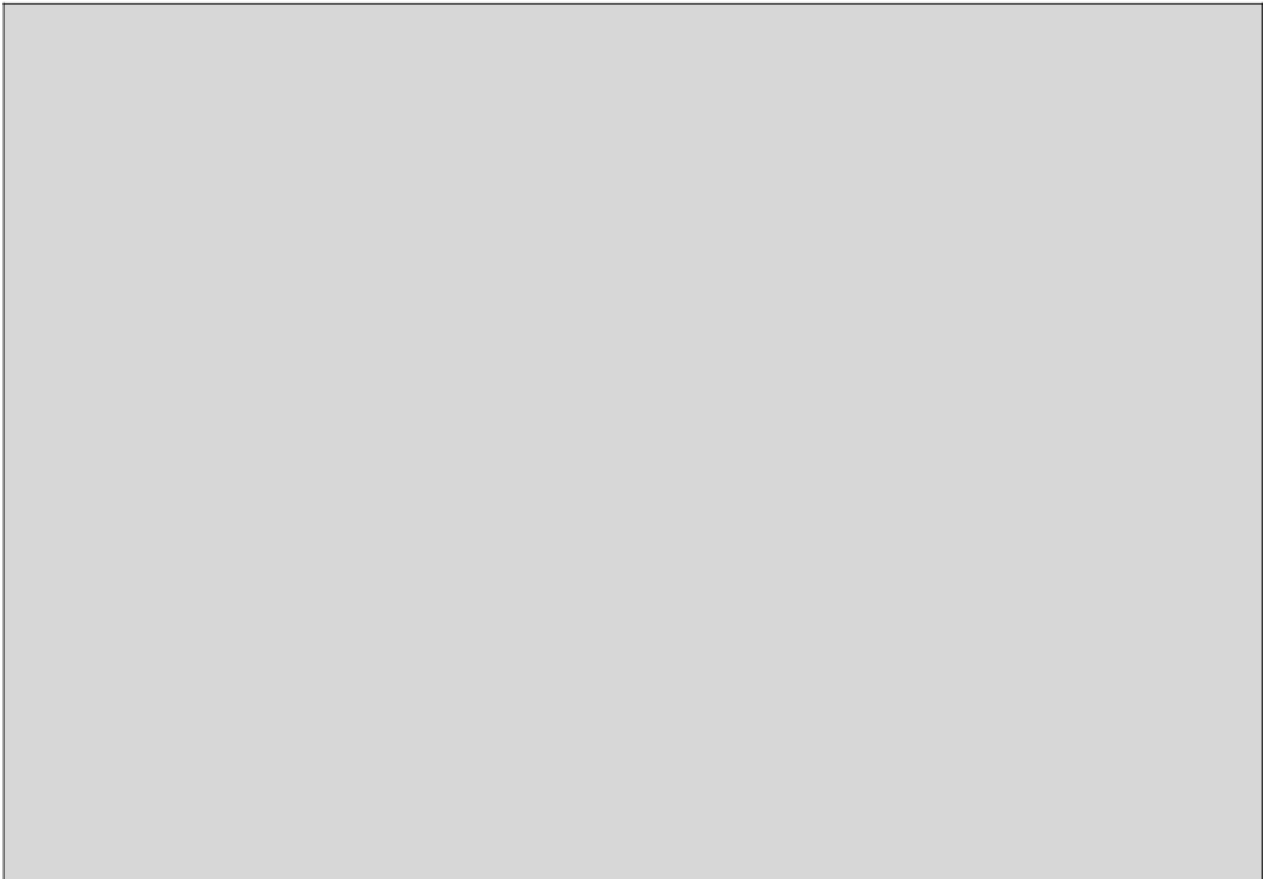
### 4.1 什么是Card组件？

Card 是卡片组件块，内容可以由大多数类型的 Widget 构成，Card 具有圆角和阴影，这让它看起来有立体感

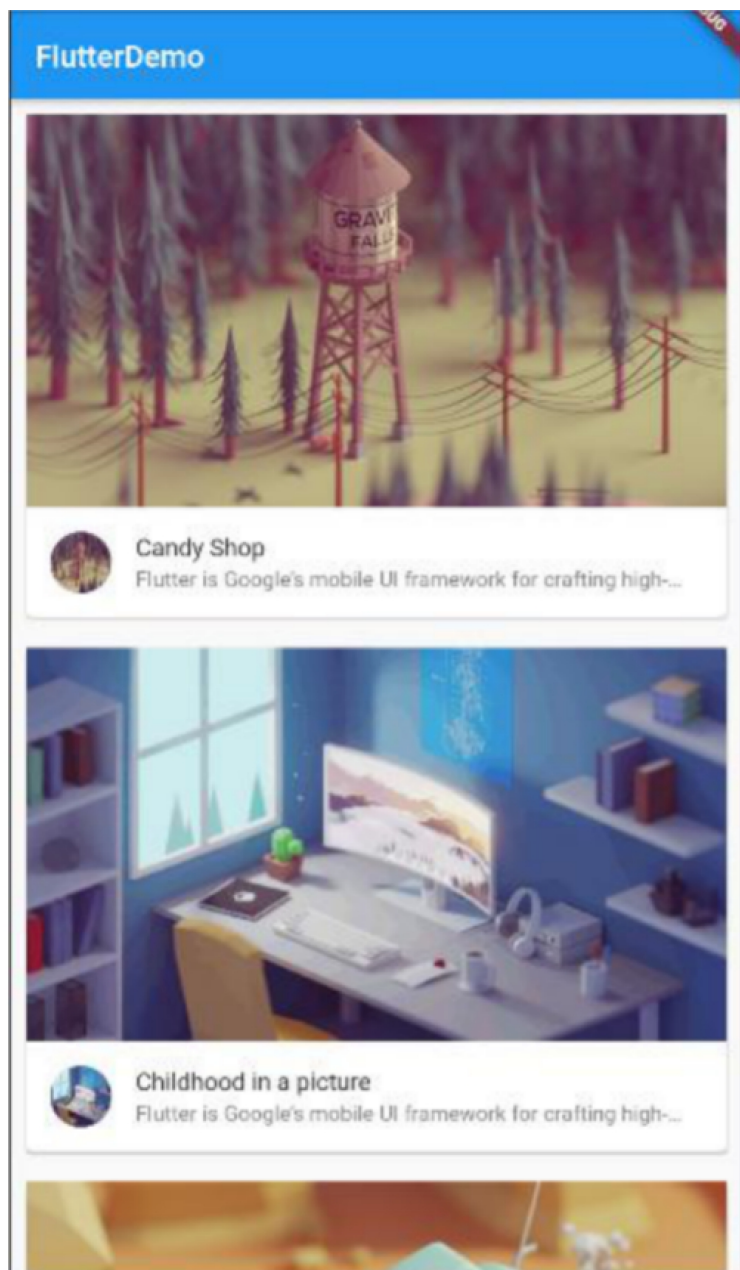
### 4.2 Card组件的属性与方法？

*属性*	*说明*
margin	外边距
child	子组件
Shape	Card 的阴影效果，默认的阴影效果为圆角的长方形边。

### 4.3 完成如下示例



#### 4.4 使用Card组件实现图文列表?



#### 五、Stack组件与Align组件和Positioned组件实现定位布局

---

## 5.1 什么是Stack组件?

Stack 表示堆的意思，我们可以用 Stack 或者 Stack 结合 Align 或者 Stack 结合 Positiond 来实现页面的定位布局

## 5.2 Stach组件的属性与方法?

*属性*	*说明*
alignment	配置所有子元素的显示位置
children	子组件

## 5.3 Align组件的使用?

Stack 组件中结合 Align 组件可以控制每个子元素的显示位置.

## 5.4 Align组件的属性与方法

*属性*	*说明*
alignment	配置所有子元素的显示位置
child	子组件

## 5.5 Positioned组件的使用?

Stack 组件中结合 Positioned 组件也可以控制每个子元素的显示位置

## 5.6 Positioned组件的属性与方法?

top	子元素距离顶部的距离
bottom	子元素距离底部的距离
left	子元素距离左侧距离
right	子元素距离右侧距离
child	子组件

# 六、项目实战

---

## 6.1 项目演示

## 6.2 创建项目



6.3 制作顶部标题栏

6.4 制作底部TabBar

6.5 制作底部TabBar页面切换效果

6.6 制作首页轮播图效果

6.7 制作猜你喜欢和推荐商品标题模块以及封装

6.8 制作猜你喜欢商品列表左右滑动效果

6.9 制作推荐商品列表模块