

一、Image

1、构造方法

2、fit参数

3、alignment

3、repeat

4、filterQuality

5、范例代码

二、Icon

1、构造方法

2、代码范例

一、Image

1、构造方法

```
1 class Image extends StatefulWidget {  
2  
3   const Image({  
4     @required this.image,  
5     ...  
6   })  
7  
8   Image.network(String src,{  
9     ...  
10    })  
11  
12   Image.file(File file, {  
13     ...  
14    })  
15  
16   Image.asset(String name, {  
17     ...
```

```

18  })
19
20  Image.memory(Uint8List bytes, {
21    ...
22  })
23
24  }

```

Image一共有五个构造方法，我们常用的则是四个。

- Image.network(url):从网络加载显示图片
- Image.file(file):从本地文件加载显示图片
- Image.asset(name):从Flutter App的资源文件里加载显示图片
- Image.memory(bytes):从内存中加载显示图片

而这些构造方法的参数各有不同，这里我们选择公共的参数来讲解：

参数名字	参数类型	意义	必选 or 可选
key	Key	Widget 的标识	可选
scale	double	图形显示的比例	可选
semantics Label	String	给 Image 加上一个语义标签 没有实际用处	可选
width	double	图片的宽 如果为null的话，则图像将选择最佳大小显示，而且会保留其固有宽高比的大小	可选
height	double	图片的高 如果为null的话，则图像将选择最佳大小显示，而且会保留其固有宽高比的大小	可选
color	Color	图片的混合色值	可选
colorBlendMode	BlendMode	图片与颜色的混合模式	可选
fit	BoxFit	用于在图片的显示空间和图片本身大小不同时指定图片的适应模式	可选
alignment	Alignment	图片的对齐方式	可选
repeat	ImageRepeat	当图片本身大小小于显示空间时，指定图片的重复规则	可选
centerSlice	Rect	在这个矩形范围内的图片会被当成.9的图片	可选
matchTextDirection	bool	图片的绘制方向 true:从左往右 false:从右往左	可选
gaplessPlayback	bool	当图像提供者更改时 true: 继续显示旧图像 false: 简单地显示任何内容	可选

filterQuality	FilterQuality	设置图片的过滤质量	可选
---------------	---------------	-----------	----

2、fit参数

BoxFit，用于在图片显示空间和图片本身大小不同时指定图片的适应模式

BoxFit的值	含义
BoxFit.fill	会拉伸填充满显示空间，图片本身长宽比会发生变化，图片会变形。
BoxFit.contain	会按图片的长宽比放大后居中填满显示空间，图片不会变形，超出显示空间部分会被剪裁。
BoxFit.cover	这是图片的默认适应规则，图片会在保证图片本身长宽比不变的情况下缩放以适应当前显示空间，图片不会变形。
BoxFit.fitWidth	图片的宽度会缩放到显示空间的宽度，高度会按比例缩放，然后居中显示，图片不会变形，超出显示空间部分会被剪裁。
BoxFit.fitHeight	图片的高度会缩放到显示空间的高度，宽度会按比例缩放，然后居中显示，图片不会变形，超出显示空间部分会被剪裁。
BoxFit.scaleDown	对齐目标框内的源（默认情况下，居中），并在必要时缩小源以确保源适合框内。这与contains相同，如果这会缩小图像，否则它与none相同。
BoxFit.none	图片没有适应策略，会在显示空间内显示图片，如果图片比显示空间大，则显示空间只会显示图片中间部分

3、alignment

图片的对齐方式

Alignment的值	含义
Alignment.topLeft	左上对齐
Alignment.topCenter	上部居中对齐
Alignment.topRight	右上对齐
Alignment.centerLeft	中间居左对齐
Alignment.center	中间对齐
Alignment.centerRight	中间居右对齐
Alignment.bottomLeft	左下对齐
Alignment.bottomCenter	底部居中对齐
Alignment.bottomRight	右下对齐

3、repeat

ImageRepeat，当图片本身大小小于显示空间时，指定图片的重复规则

ImageRepeat的值	含义
---------------	----

ImageRepeat.repeat	在x和y轴上重复图像，直到填满空间。
ImageRepeat.repeatX	在x轴上重复图像，直到填满空间。
ImageRepeat.repeatY	在y轴上重复图像，直到填满空间。
ImageRepeat.noRepeat	不重复

4、filterQuality

设置图

FilterQuality的值	含义
FilterQuality.none	最快的过滤。
FilterQuality.low	比none的过滤质量好，但是比none的时间要长。
FilterQuality.medium	比low的过滤质量好，但是也比low的时间要长
FilterQuality.high	过滤质量最高，但也最慢

5、范例代码

```

1 class ImageWidget extends StatelessWidget {
2   @override
3   Widget build(BuildContext context) {
4     // TODO: implement build
5     return MaterialApp(
6       title: "Flutter Demo",
7       theme: ThemeData(primaryColor: Colors.blue),
8       home: Scaffold(
9         appBar: AppBar(title: Text("Flutter学习手册之图片组件")),
10        // body: Image.asset("images/flutter.png", fit: BoxFit.cover),
11        body: Image.network(
12          "https://timgsa.baidu.com/timg?image&quality=80&size=b9999_10000&sec=1557781801455&di=17f9f2fc3ded4efb7214d2d8314e8426&imgtype=0&src=http%3A%2F%2Fimg2.mukewang.com%2F5b4c075b000198c216000586.jpg",
13          width: 120,
14          height: 30,
15          fit: BoxFit.fill,
16        ),
17      ),
18    );
19  }

```

二、Icon

Flutter中可以像Web中一样使用iconfont，即将图标作为文字，然后通过不同的字符显示不同的图片。

当然我们用image一样可以实现，那么它的优势是：

- 体积小：减少安装包大小
- 矢量的：放大不会影响清晰度
- 可以应用文本样式：即可以像文本一样改变字体图标的颜色，大小对齐等
- 通过TextSpan与文本混用

1、构造方法

```
1 class Icon extends StatelessWidget {
2
3   const Icon(this.icon, {
4     Key key,
5     this.size,
6     this.color,
7     this.semanticLabel,
8     this.textDirection,
9   }) : super(key: key);
10
11   ...
12 }
```

参数名字	参数类型	意义	必选 or 可选
key	Key	Widget 的标识	可选
size	double	icon的大小	可选
color	Color	icon的颜色	可选
semantics Label	String	给 Image 加上一个语义标签 没有实际用处	可选
textDirect ion	TextDirection	Icon的绘制方向	可选

2、代码范例

```
1 class IconWidget extends StatelessWidget {
2   @override
3   Widget build(BuildContext context) {
4     // TODO: implement build
```

```
5  return MaterialApp(  
6    title: "Flutter Demo",  
7    theme: ThemeData(primaryColor: Colors.blue),  
8    home: Scaffold(  
9      appBar: AppBar(title: Text("Flutter学习笔记之Icon组件")),  
10     body: Icon(  
11       Icons.android,  
12       size: 50.0,  
13       color: Colors.green,  
14     ),  
15   ),  
16 );  
17 }  
18 }
```

无服务 中



100% 晚上9:16

DEBUG

Flutter学习笔记之Icon组件

