

Image

一个显示图片的widget



Image

一个显示图片的widget

图像类

一个小部件显示一个图像。

几个构造函数提供了一个图像的各种方法 指定：

- [新形象](#),从一个获取图像[ImageProvider](#)。
- [新Image.asset](#),从一个获取图像[AssetBundle](#)使用一个密钥。
- [新Image.network](#),从一个URL获取图像。
- [新Image.file](#),获取图像[文件](#)。
- [新Image.memory](#),获取图像[Uint8List](#)。

下面的图像格式支持:JPEG、PNG、GIF动画GIF, WebP, 动画WebP, BMP, WBMP

自动执行pixel-density-aware资产决议, 指定 使用一个形象[AssetImage](#)并确保[MaterialApp](#), [WidgetsApp](#), 或[MediaQuery](#)小部件存在以上[图像](#)部件在树小部件。

用画的图像[paintImage](#), 这描述的含义 各领域的更详细地对这门课。

参见：

- [图标](#)从字体,显示一个图像。

- [新Ink.image](#)显示一个图像,这是首选的 材料(特别是图像的应用[材料](#)并将有一个[墨水池](#)在上面)。

继承

- [对象](#)
- [Diagnosticable](#)
- [DiagnosticableTree](#)
- [小部件](#)
- [StatefulWidget](#)
- [图像](#)

构造函数

[图像](#)({[关键](#) 关键, @ required [ImageProvider](#) 图像, [双](#) 宽度, [双](#) 高度, [颜色](#) 颜色, [BlendMode](#) colorBlendMode, [BoxFit](#) 适合, [AlignmentGeometry](#) 对齐:Alignment.center, [ImageRepeat](#) 重复:ImageRepeat.noRepeat, [矩形](#) centerSlice, [bool](#) matchTextDirection:假, [bool](#) gaplessPlayback:假})

创建一个小部件显示一个图像。[...]

常量

[Image.asset](#)([字符串](#) 的名字, { [关键](#) 关键, [AssetBundle](#) 包, [双](#) 规模, [双](#) 宽度, [双](#) 高度, [颜色](#) 颜

色, [BlendMode](#) colorBlendMode, [BoxFit](#) 适合, [AlignmentGeometry](#) 对齐:Alignment.center, [ImageRepeat](#) 重复:ImageRepeat.noRepeat, [矩形](#) centerSlice, [bool](#) matchTextDirection:假, [bool](#) gaplessPlayback:假, [字符串](#) 包})

创建一个小部件显示[ImageStream](#)从一个资产包。的图像的关键name论点。[...]

[Image.file](#)([文件](#) 文件, { [关键](#) 关键, [双](#) 规模:1.0, [双](#) 宽度, [双](#) 高度, [颜色](#) 颜色, [BlendMode](#) colorBlendMode, [BoxFit](#) 适合, [AlignmentGeometry](#) 对齐:Alignment.center, [ImageRepeat](#) 重复:ImageRepeat.noRepeat, [矩形](#) centerSlice, [bool](#) matchTextDirection:假, [bool](#) gaplessPlayback:假})

创建一个小部件显示[ImageStream](#)获得一个[文件](#)。[...]

[Image.memory](#)([Uint8List](#) 字节, { [关键](#) 关键, [双](#) 规模:1.0, [双](#) 宽度, [双](#) 高度, [颜色](#) 颜色, [BlendMode](#) colorBlendMode, [BoxFit](#) 适合, [AlignmentGeometry](#) 对齐:Alignment.center, [ImageRepeat](#) 重复:ImageRepeat.noRepeat, [矩形](#) centerSlice, [bool](#) matchTextDirection:假, [bool](#) gaplessPlayback:假})

创建一个小部件显示[ImageStream](#)获得一个[Uint8List](#)。[...]

[Image.network](#)([字符串](#) src, { [关键](#) 关键, [双](#) 规模:1.0, [双](#) 宽度, [双](#) 高度, [颜色](#) 颜色, [BlendMode](#) colorBlendMode, [BoxFit](#) 适合, [AlignmentGeometry](#) 对齐:Alignment.center, [ImageRepeat](#) 重复:ImageRepeat.noRepeat, [矩形](#) centerSlice, [bool](#) matchTextDirection:假, [bool](#) gaplessPlayback:假, [地图](#)<[字符串](#), [字符串](#)> 头})

创建一个小部件显示[ImageStream](#)从网络获得。[...]

属性

[对齐](#) → [AlignmentGeometry](#)

如何使图像在其范围内。[...]

最后

[centerSlice](#) → [矩形](#)

中心nine-patch形象片。[...]

最后

[颜色](#) → [颜色](#)

如果非空, 这颜色是与每个图像像素混合使用colorBlendMode。

最后

[colorBlendMode](#) → [BlendMode](#)

结合使用color这张图片。 [\[...\]](#)

最后

[适合](#) → [BoxFit](#)

如何将图像写入期间分配的空间布局。 [\[...\]](#)

最后

[gaplessPlayback](#) → [bool](#)

是否继续显示旧图片(真正的),或短暂的显示(假),当图片提供者的变化。

最后

[高度](#) → [双](#)

如果非空,要求形象有这个高度。 [\[...\]](#)

最后

[图像](#) → [ImageProvider](#)

图像显示。

最后

[matchTextDirection](#) → [bool](#)

是否要画图像的方向TextDirection。 [\[...\]](#)

最后

[重复](#) → [ImageRepeat](#)

如何画的任何部分布局范围不受图像。

最后

[宽度](#) → [双](#)

如果非空,要求图像宽度。 [\[...\]](#)

最后

[hashCode](#) → [int](#)

这个对象的哈希码。 [\[...\]](#)

只读的,遗传的

[关键](#) → [关键](#)

控制一个小部件替换另一个小部件在树上。 [\[...\]](#)

最后,继承了

[runtimeType](#) → [类型](#)

一个对象的运行时类型的代表。

只读的, 遗传的

方法

[`createState\(\)`](#) → [`_ImageState`](#)

为这个小部件创建可变状态给定树中的位置。 [\[...\]](#)

[`debugFillProperties`](#) ([`DiagnosticPropertiesBuilder`](#) 属性) → 无效

[`createElement\(\)`](#) → [`StatefulElement`](#)

创建一个[`StatefulElement`](#)在树上来管理这个小部件的位置。 [\[...\]](#)

继承了

[`debugDescribeChildren\(\)`](#) → [列表<`DiagnosticsNode`>](#)

返回一个列表[`DiagnosticsNode`](#)描述该节点的对象 的孩子。 [\[...\]](#)

@protected, 继承了

[`noSuchMethod\(\)`](#) (调用 调用) → 动态

当用户访问一个不存在的方法或属性调用。 [\[...\]](#)

继承了

[`toDiagnosticsNode`](#) ({[字符串](#) 的名字, [`DiagnosticsTreeStyle`](#) 风格}) → [`DiagnosticsNode`](#)

返回一个对象被调试的调试表示 工具和[`toStringDeep`](#)。 [\[...\]](#)

继承了

[`toString`](#) ({[`DiagnosticLevel`](#) minLevel:[`DiagnosticLevel.debug`](#)}) → [字符串](#)

返回该对象的字符串表示。

继承了

[`toStringDeep`](#) ({[字符串](#) prefixLineOne:", [字符串](#) prefixOtherLines, [`DiagnosticLevel`](#) minLevel:[`DiagnosticLevel.debug`](#)}) → [字符串](#)

返回一个字符串表示该节点及其后代。 [\[...\]](#)

继承了

[`toStringShallow`](#) ({[字符串](#) 乔伊纳:", " , [`DiagnosticLevel`](#) minLevel:[`DiagnosticLevel.debug`](#)}) → [字符串](#)

返回一行详细描述的对象。 [\[...\]](#)

继承了

[`toStringShort`](#) () → [字符串](#)

短, 这个小部件的文本描述。

继承了

操作

运算符==(动态 其他)→bool

相等操作符。[...]

继承了