

自定义视图

要求：

- 1, 符合Android标准。
- 2, 提供可与Android XML布局配合使用的自定义样式属性。
- 3, 发送事件。
- 4, 兼容多个Android平台。

使用：

- 1, 继承自view或者继承自一个view的子类 (button) 。
提供构造器。
(context, attributeSet为必选参数) , 完成之后可以使布局创建和编辑。
- 2, 可以自定义属性。
自定义属性值使用有要求。
- 3, 使用自定义属性。
调用 invalidate()和requestLayout()方法。【方便让系统知道需要绘图, 如过属性更改会影响视图大小和形状需设置】

Custom Drawing:

- 1, 复写OnDraw()方法。
- 2, 需要Canvas。---->What
- 3, 需要Paint。---->How

处理布局事件

取决于视图在屏幕的大小。

不需要特殊的控制大小, 只需要复写 onSizeChanged()方法。

更精细的控制 onMeasure()

Draw:

- Draw text using `drawText()`. Specify the typeface by calling `setTypeface()`, and the text color by calling `setColor()`.
- Draw primitive shapes using `drawRect()`, `drawOval()`, and `drawArc()`. Change whether the shapes are filled, outlined, or both by calling `setStyle()`.
- Draw more complex shapes using the `Path` class. Define a shape by adding lines and curves to a `Path` object, then draw the shape using `drawPath()`. Just as with primitive shapes, paths can be outlined, filled, or both, depending on the `setStyle()`.
- Define gradient fills by creating `LinearGradient` objects. Call `setShader()` to use your `LinearGradient` on filled shapes.
- Draw bitmaps using `drawBitmap()`.