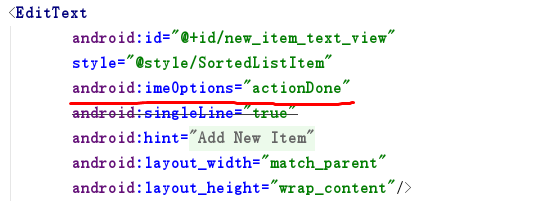
# 修改输入法中的按钮：



上面的属性是把输入法中回车键变成了一个勾，如下：

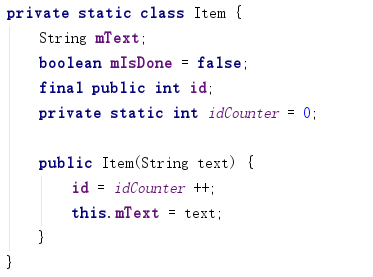


可以修改那个属性的值，可以变成一些其它的图标和对应的功能。

在代码中怎么知道按下了这个按键呢，如下：

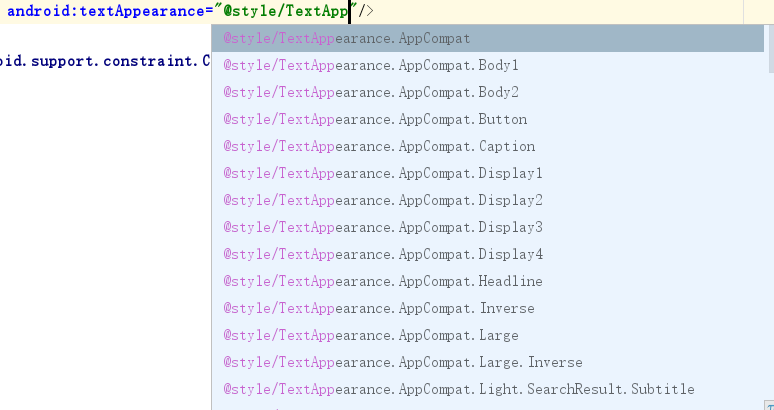
mEditText.setOnEditorActionListener(**new** TextView.OnEditorActionListener() {  
 @Override  
 **public boolean** onEditorAction(TextView textView, **int** id, KeyEvent keyEvent) {  
 **if** (id == EditorInfo.***IME\_ACTION\_DONE*** && (keyEvent == **null** || keyEvent.getAction() == KeyEvent.***ACTION\_DOWN***)) {  
 *// 如果按下了输入法中的完成按键  
 // 这里的参数TextView其实就是EditText，所以可以直接从textView.getText取出输入框的文本* **return true**;  
 }  
 **return false**;  
 }  
});

# 实现一个id自增：



每new一个Item对象，都不需要给id赋值，因为构造方法内部自动赋值了，而且是自增的，秒啊！

# 看到Android源生Demo中使用的主题

* 在布局中直接给容器设置的主题：android:theme="@style/ThemeOverlay.AppCompat.Dark"，用了这个主题后，虽然布局背景没有变成黑色，但是它的子孩子的颜色都会变成亮色调。
* 给TextView设置的样式：android:textAppearance="@style/TextAppearance.AppCompat.Body1"
* 还有很多关于字体外观的样式，如下：  
  
* 在AppbarLayout上使用了这个主题：android:theme="@style/AppTheme.AppBarOverlay"
* 在Toolbar上使用这些属性：android:layout\_height="?attr/actionBarSize"、 android:background="?attr/colorPrimary" 、app:popupTheme="@style/AppTheme.PopupOverlay"
* 上面AppbarLayout和Toolbar上使用的主题是在样式文件中定义的，如下：

<**resources**><**style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar"**><**item name="colorPrimary"**>@color/colorPrimary</**item**>  
 <**item name="colorPrimaryDark"**>@color/colorPrimaryDark</**item**>  
 <**item name="colorAccent"**>@color/colorAccent</**item**>  
 </**style**>  
  
 <**style name="AppTheme.AppBarOverlay" parent="ThemeOverlay.AppCompat.Dark.ActionBar"** />  
 <**style name="AppTheme.PopupOverlay" parent="ThemeOverlay.AppCompat.Light"** />  
</**resources**>

# 一个线性渐变的矩形

<**shape xmlns:android="http://schemas.android.com/apk/res/android" android:shape="rectangle"**>  
 <**gradient android:angle="135" android:centerColor="#4CAF50" android:endColor="#2E7D32"  
 android:startColor="#81C784" android:type="linear"** />  
</**shape**>

# ImageView也可使用app:srcCompat设置图片

# 可以把Android的一些drawable起个别名

在values目录下创建drawables.xml，内容如下：

<**resources xmlns:android="http://schemas.android.com/apk/res/android"**>  
 <**item name="ic\_menu\_camera" type="drawable"**>@android:drawable/ic\_menu\_camera</**item**>  
 <**item name="ic\_menu\_gallery" type="drawable"**>@android:drawable/ic\_menu\_gallery</**item**>  
 <**item name="ic\_menu\_slideshow" type="drawable"**>@android:drawable/ic\_menu\_slideshow</**item**>  
 <**item name="ic\_menu\_manage" type="drawable"**>@android:drawable/ic\_menu\_manage</**item**>  
 <**item name="ic\_menu\_share" type="drawable"**>@android:drawable/ic\_menu\_share</**item**>  
 <**item name="ic\_menu\_send" type="drawable"**>@android:drawable/ic\_menu\_send</**item**>  
</**resources**>

然后我们需要使用这些图片的时候，只需要background=”@drawable/ic\_menu\_camera”，这样代码比较短一点

# 一个主题继承两个主题

