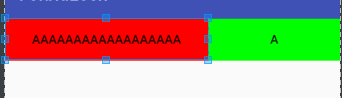
## 小技巧

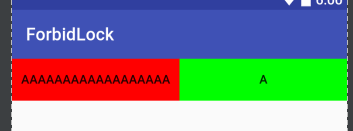
1，LinearLayout,平分区域：此方法当某个按钮字符比较长的时候，不能保证平分



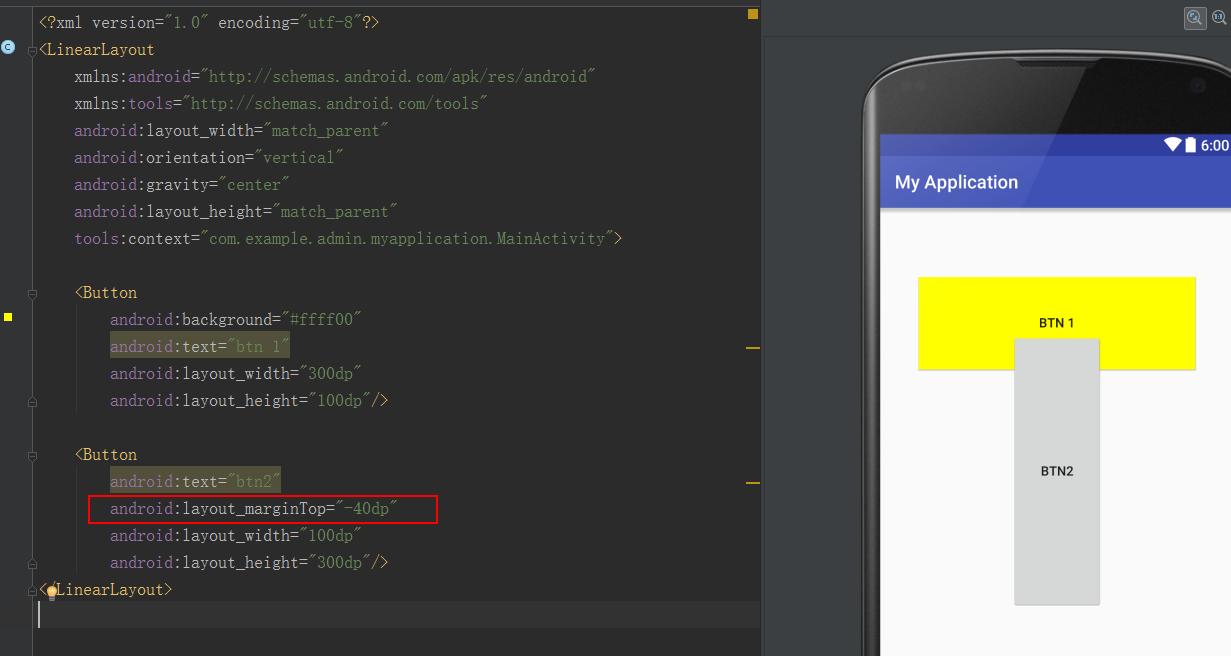


所以需要改变layout\_width = 0dp(竖方向同理设置layout\_height):



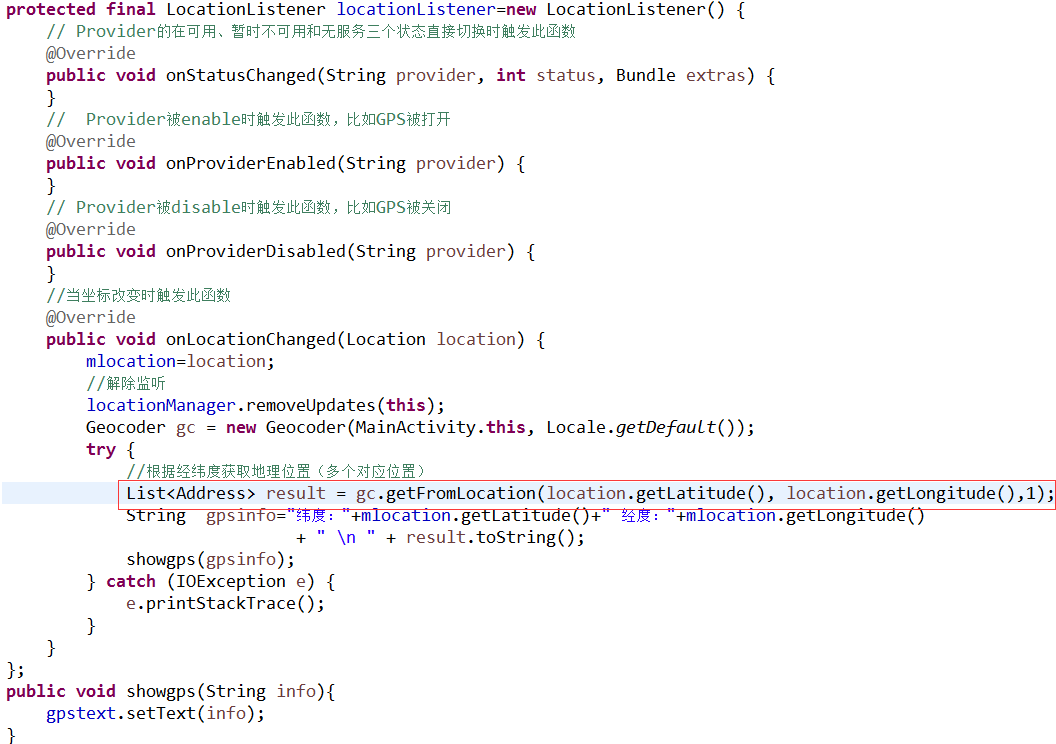


1. 组件嵌套，巧妙设置margin负值

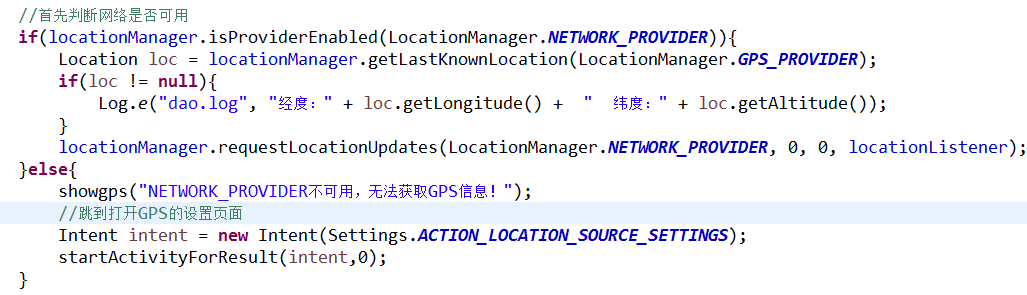


## 2,读取地理位置：前提代开了GPS:

地理位置监听：



请求地理位置：

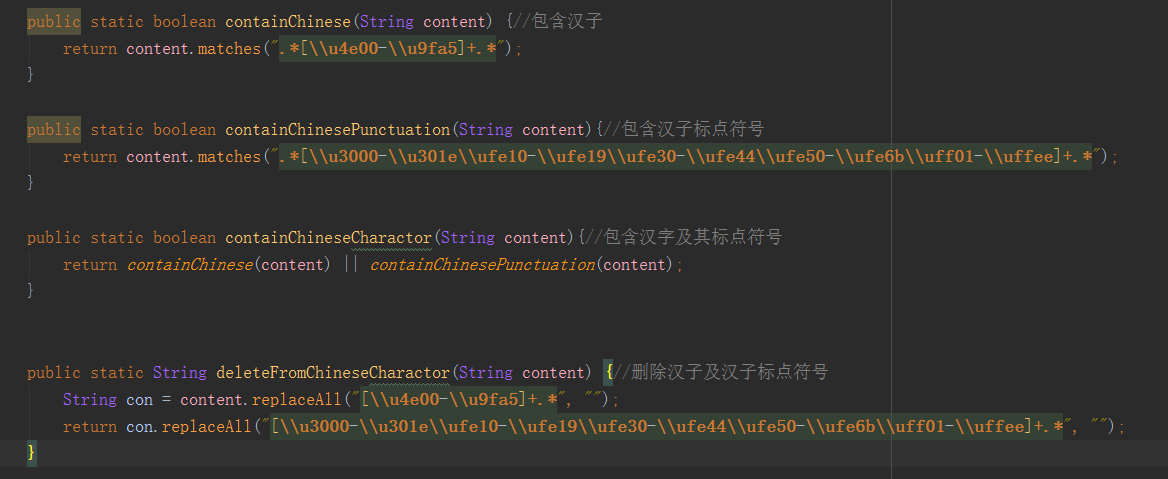




GPS没打开，则跳到打开GPS的界面：



## 3，中文检测：中文和标点符号



## 4，键盘

键盘确认键设置：

1、输入法Enter键图标的设置：EditText.setImeOptions

软件盘的界面替换只有一个属性android:imeOptions，这个属性的可以取的值有normal，actionUnspecified，actionNone，actionGo，actionSearch，actionSend，actionNext，actionDone，例如当值为actionNext时enter键外观变成一个向下箭头，而值为actionDone时enter键外观则变成了“完成”两个字。   
我们也可以重写enter的事件

软键盘的Enter键默认显示的是“完成”文本，通过设置android:imeOptions来改变默认的“完成”文本。这里举几个常用的常量值：

actionUnspecified  未指定，对应常量EditorInfo.IME\_ACTION\_UNSPECIFIED.    
actionNone 没有动作,对应常量EditorInfo.IME\_ACTION\_NONE   
actionGo 去往，对应常量EditorInfo.IME\_ACTION\_GO  
actionSearch 搜索，对应常量EditorInfo.IME\_ACTION\_SEARCH      
actionSend 发送，对应常量EditorInfo.IME\_ACTION\_SEND     
actionNext 下一个，对应常量EditorInfo.IME\_ACTION\_NEXT     
actionDone 完成，对应常量EditorInfo.IME\_ACTION\_DONE

editText.setImeOptions注意事项

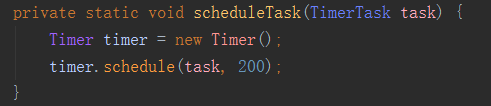
EditorInfo.IME\_ACTION\_DONE可以和其他的标志一起组合使用来设置软键盘,比如：

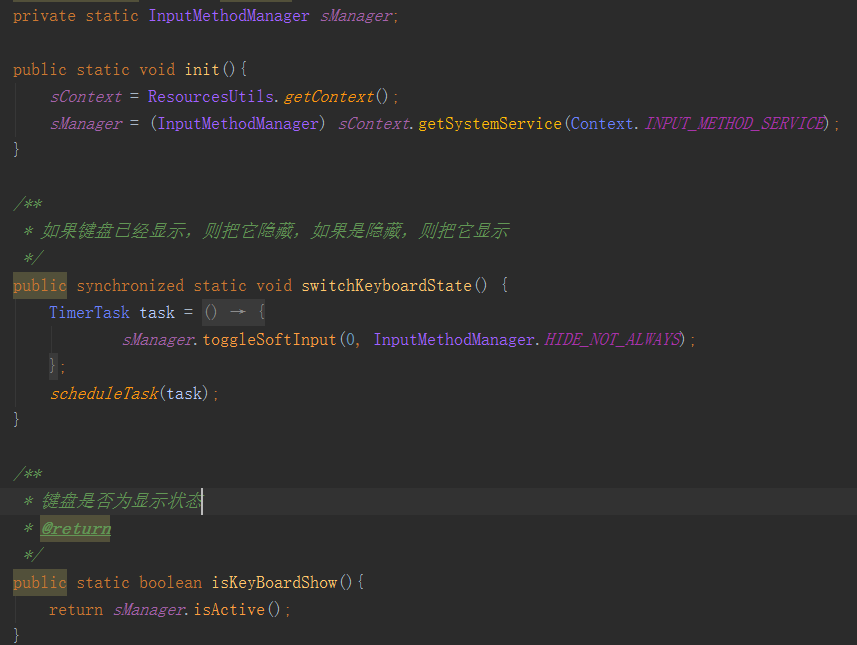
editText.setImeOptions(EditorInfo.IME\_FLAG\_NO\_EXTRACT\_UI|EditorInfo.IME\_ACTION\_DONE);

注意1：EditorInfo.IME\_ACTION\_DONE只有对android:singleLine="true"的EditText有效。

注意2：对于EditorInfo.IME\_ACTION\_DONE，有些输入法并不支持它，比如搜狗拼音输入法。

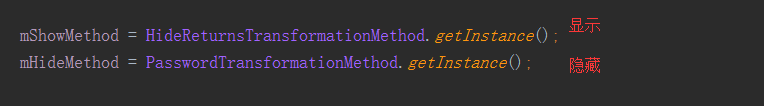
2,键盘显隐控制：





## 5，EditText

密码显示和隐藏：setTransformationMethod



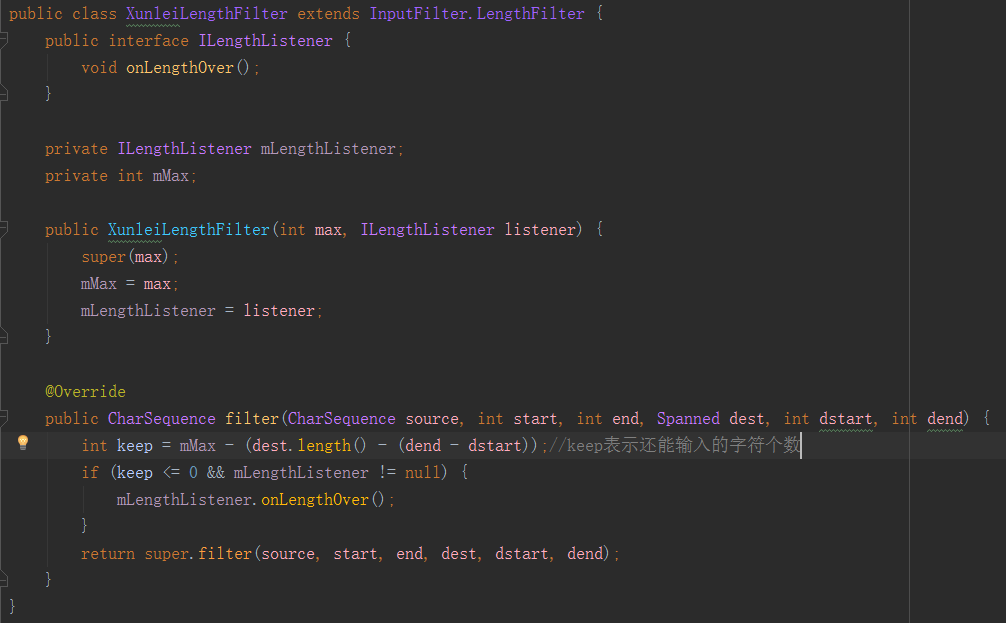


防止全屏：

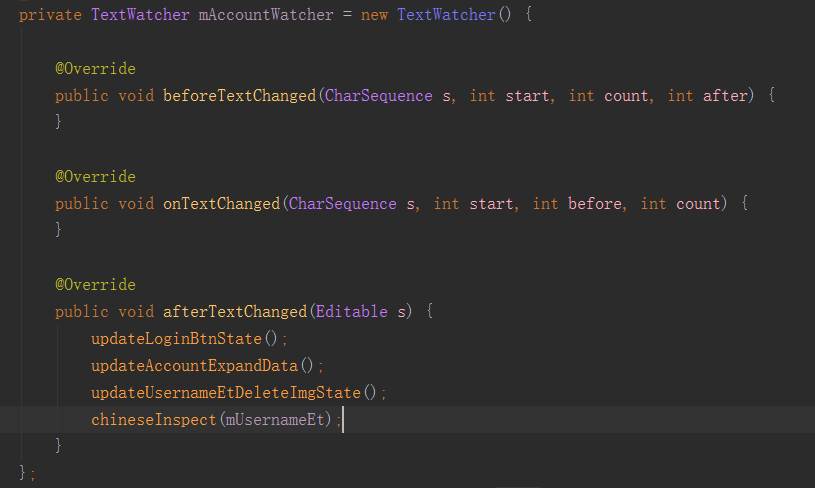


长度：





输入时的内容监听：



## 6,控件自定义

1,组合控件：XunleiItemView

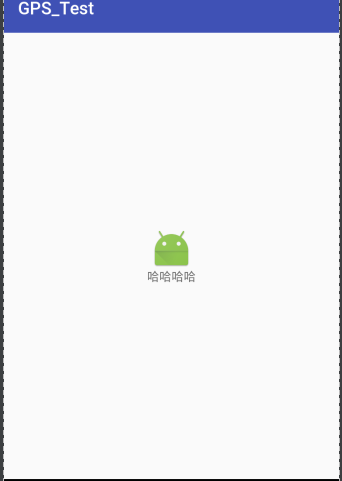
2,继承控件：XunleiAccountExpandView

3,自绘控件：CircleTextView:

## 7,TextView

图片+文字：





## 8，休眠控制

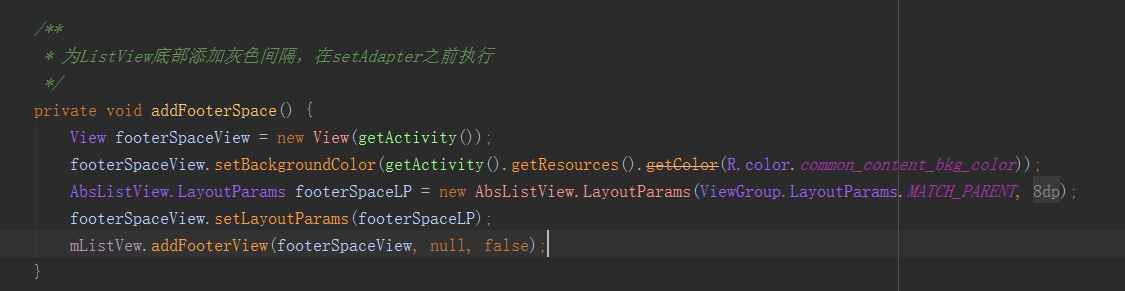
在SeContentView之前加入此句，则可以在此界面在最前端的时候不休眠：

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_KEEP\_SCREEN\_ON, WindowManager.LayoutParams.FLAG\_KEEP\_SCREEN\_ON);

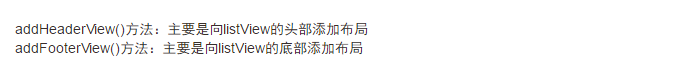
、

## 9,ListView

增加底部间隔：设置ListView高度为match\_parent，最后一个元素



添加页脚页头：



页尾页头：

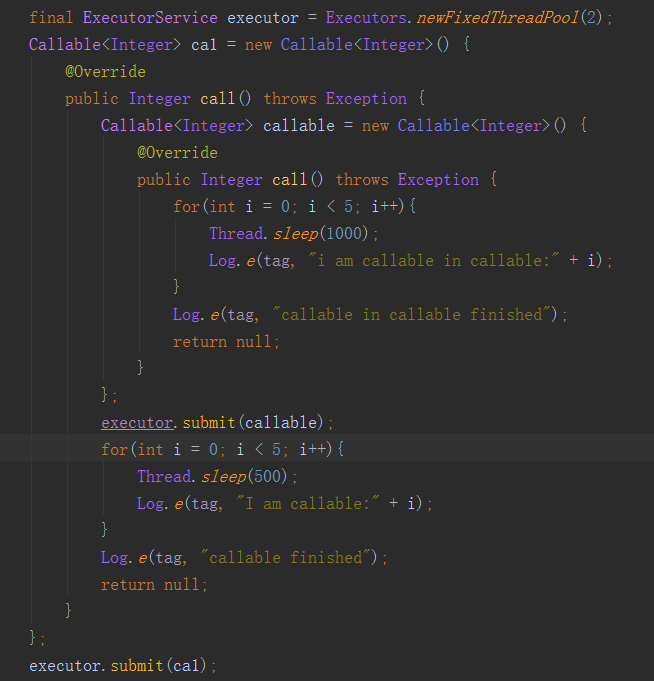
<https://software.intel.com/zh-cn/blogs/2014/12/15/android-listview-addheaderview-addfooterview>

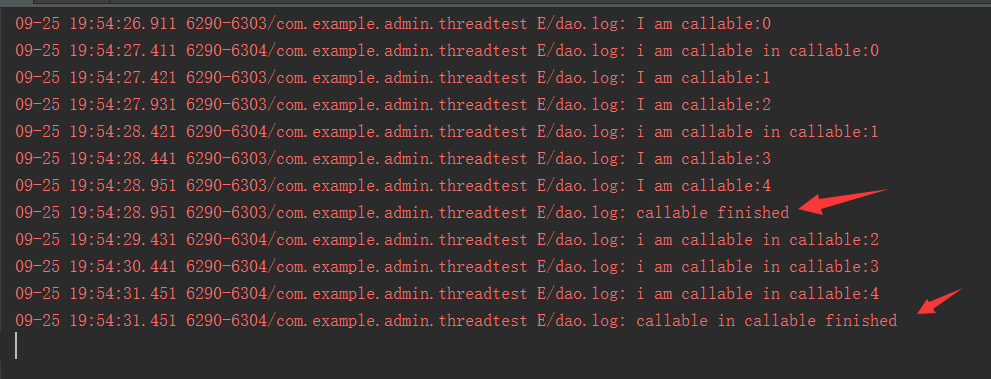
去除阴影：



## 10 多线程

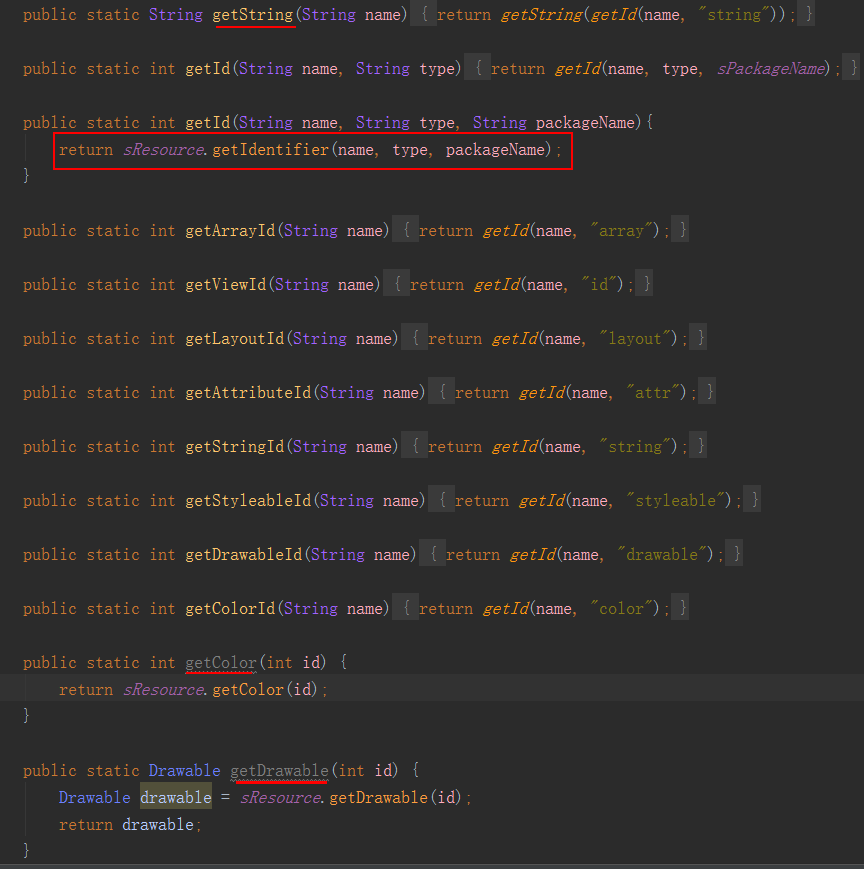
1，线程中可以创建线程，且不会相互影响，线程退出，但不会影响他创建的线程



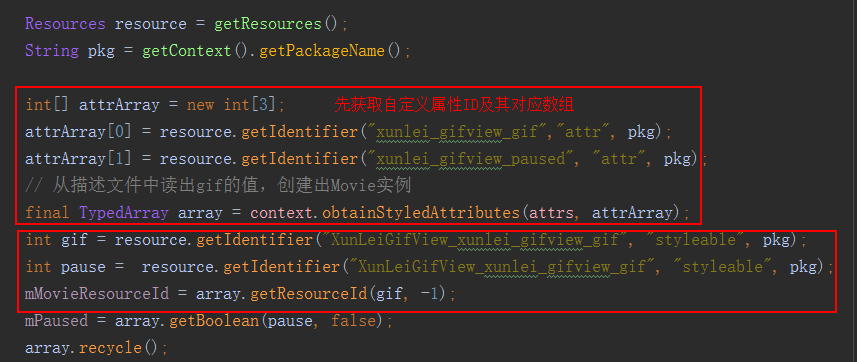


## 11 反射

1. 通过反射获取各种资源的资源ID，然后再根据获取的资源ID获取资源



通过反射获取自定义属性：

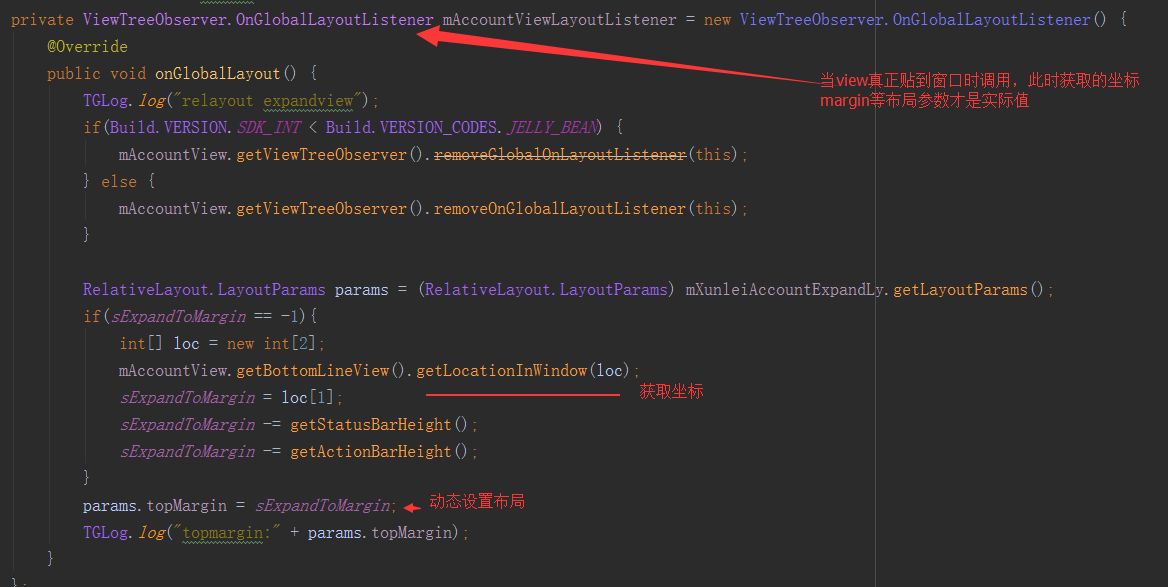


****

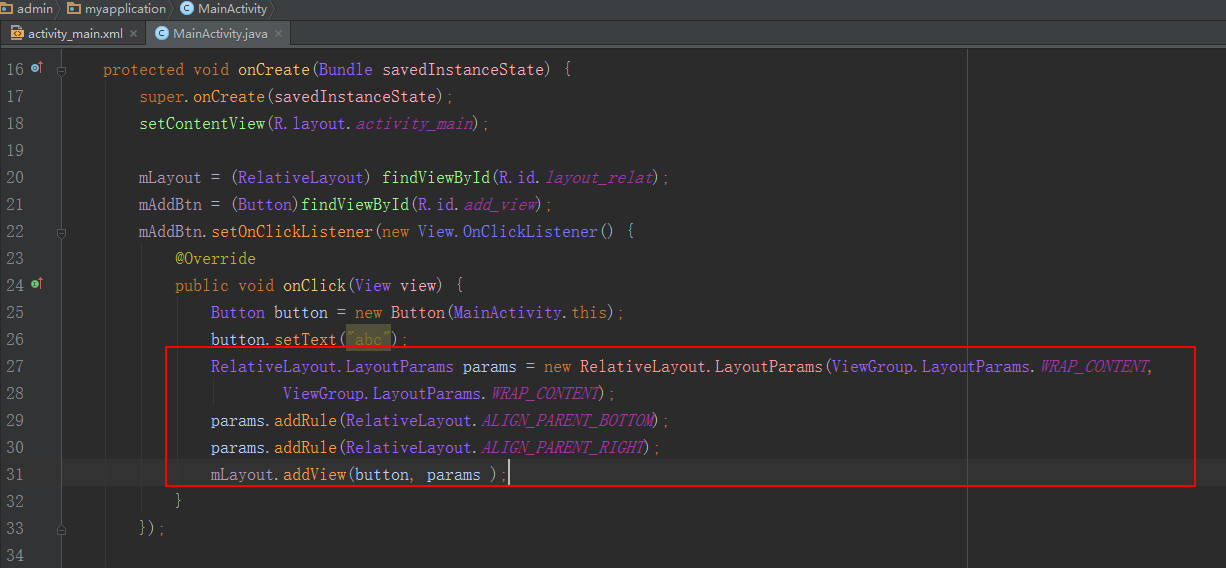
## 12，动态布局

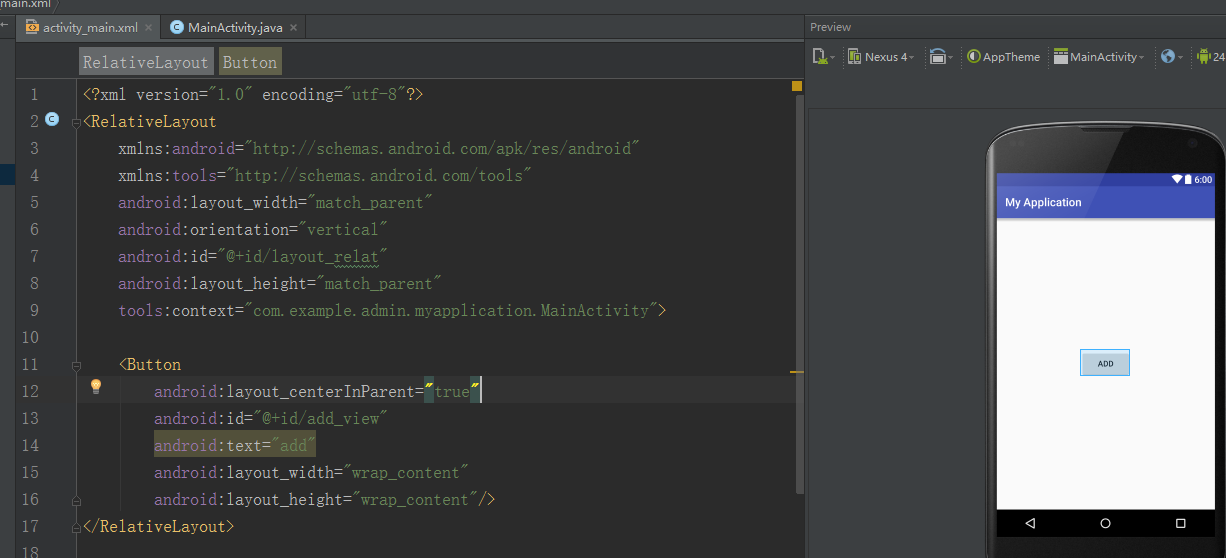
1. 获取坐标，margin等，直接在fragment或者activity中的生命周期中获取坐标或者margin的布局数据，往往获取的是0，此时*可*以试用如下方法：

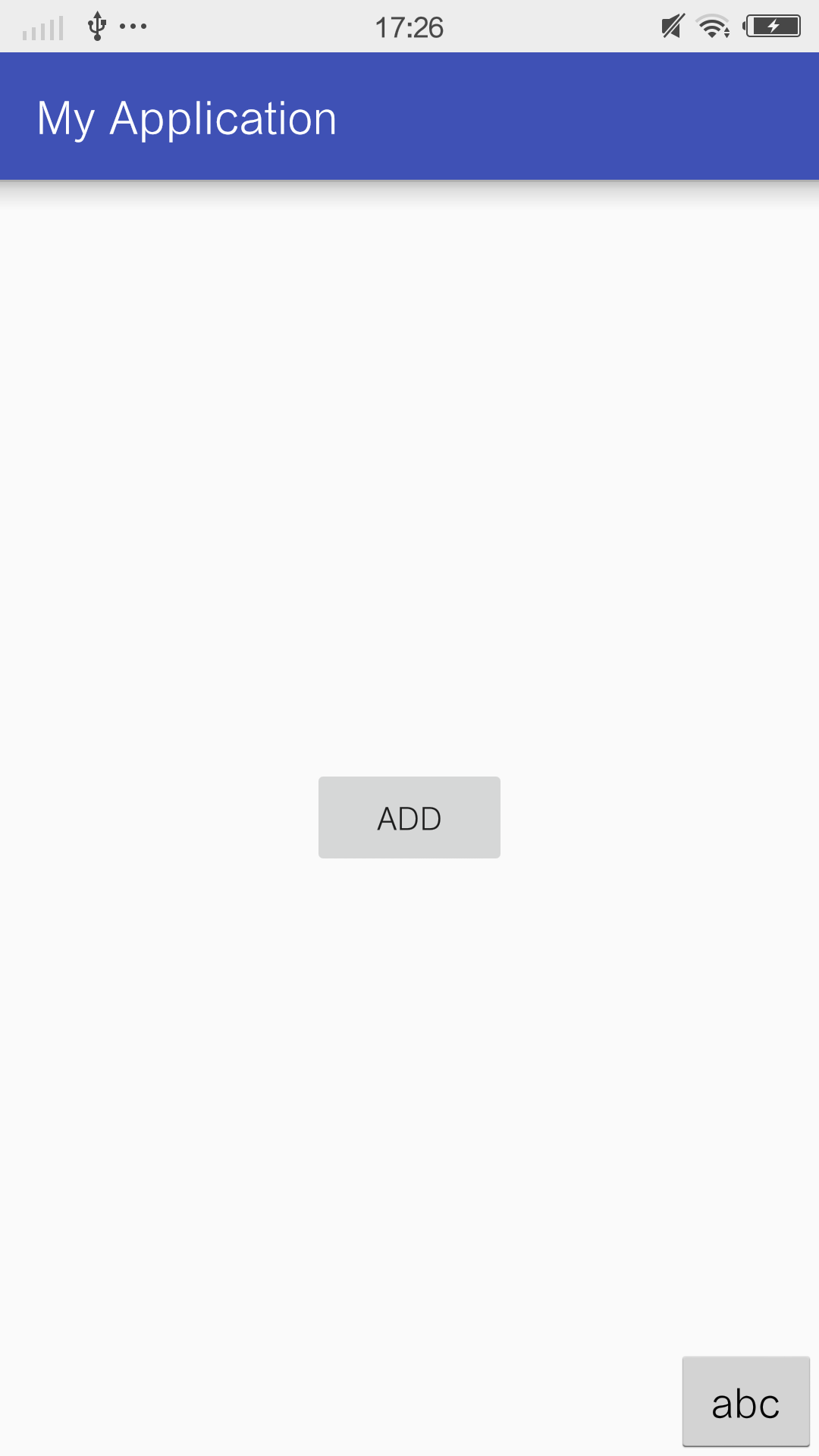




1. 动态布局，要注意父layout的gravity属性：

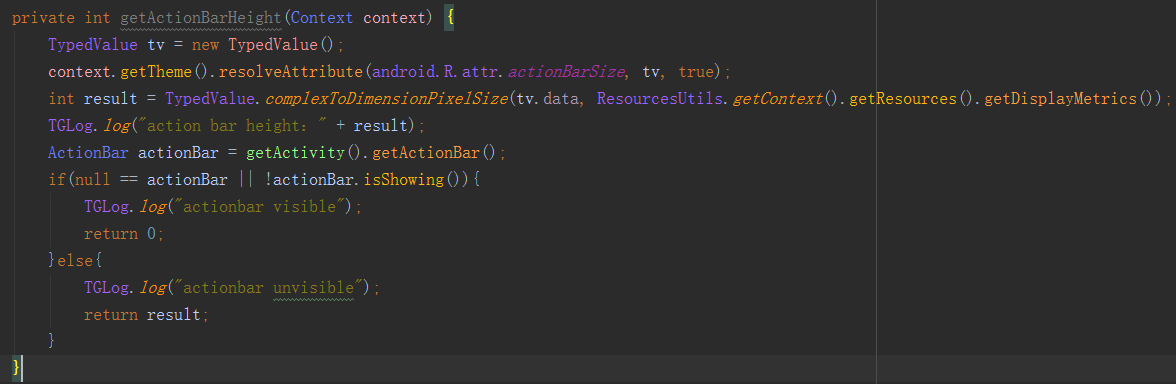






## 13位置获取

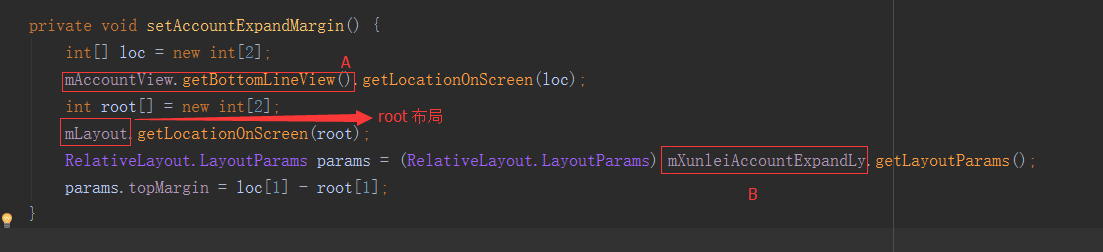
获取标题栏高度：



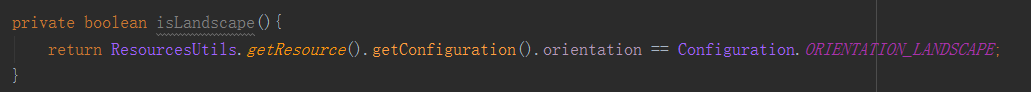
获取状态栏高度；



将viewB 放在viewA下面：用布局的间距作为margin



判断横竖屏：



## 14，各种栏

在Activity中动态设置notitlebar

  requestWindowFeature(Window.FEATURE\_NO\_TITLE);

## 15，Scrollview

一个Scrollview中不能包含多个子view，否则会报错：ScrollView can host only one direct child

如：

<Scrollview>

<Textview> </Textview>

<Textview> </Textview>

….

</Scrollview>

解决方案：将多个子view放到一个子view中

<Scrollview>

<LinearLayout>

<Textview> </Textview>

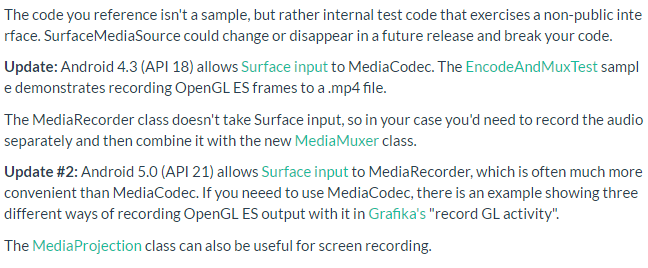
<Textview> </Textview>

….

</LinearLayout>

</Scrollview>

## 打开网址，过滤浏览器



Intent.createChooser//TODO

PopupWindow

Scope:compile和providedd:

http://supercharles888.blog.51cto.com/609344/981316

