**Android的ListView复用EditText数据混乱的问题**

在企业的互生业务办理的工具申购列表有这样一个需求，就是在ListView中，每一个item项里面有一个输入框EditText，需要把每个EdiText输入的数据保存到对应的java bean中。

往往多数人在getView方法写的代码（部分核心代码）如下：

final ToolsInfo mTool = mToolsList.get(position);

// 监听输入框EditText改变

holder.etInput.addTextChangedListener(new TextWatcher() {

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {

}

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void afterTextChanged(Editable s) {

String newCount = s.toString().trim();

if(StringUtils.isEmpty(newCount)){

newCount = "0";

}

mTool .setQuanilty(newCount);

}

});

holder.etInput.setText(mTool .getQuanilty());

运行之后，发现想要的结果和想象的差距太大，数据都乱套了，EditText输入框的数据随着ListView滑动都复用了。最后猜想应该是文本改变监听里面设置数据出了问题，设置的数据不是当前控件所在的position里面的java bean，而是复用了其他的java bean。于是考虑使用setTag()方法，即把java bean绑定在当前的EditText身上，其中的核心代码如下所示：

ToolsInfo mTool = mToolsList.get(position);

holder.etInput.setTag(mTool); //把Bean与输入框进行绑定,解决输入框的数字复用问题

// 监听输入框EditText改变

holder.etInput.addTextChangedListener(new TextWatcher() {

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {}

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {}

@Override

public void afterTextChanged(Editable s) {

String newCount = s.toString().trim();

f(StringUtils.isEmpty(newCount)){

newCount = "0";

}

ToolsInfo tool = (ToolsInfo) holder.etInput.getTag();

tool.setQuanilty(newCount);

}

});

运行发现终于解决了显示数据混乱的问题了。但是还是存在一些小问题，就是EditText的焦点出现了复用的item也跟着获取了焦点，于是需要加清除焦点的属性EditText.clearFoucus()，这样就解决了。

最后运行发现程序中还是不够完美，在真机测试的时候点击EditText输入框虽然弹出了输入法，但是还得点击EditText才能输入字符，这对于用户体验来说极度不好。经过排查，其实是在点击EditText时，系统会默认的刷新adapter，重新绘制当前的屏幕，导致第一次点击EditText输入框时清除了EditText的焦点，所以导致需要二次点击才能输入字符。所以，我们需要在清单文件中的当前Activity里面配置弹出输入法禁止绘制当前屏幕的属性

<activity

android:name=".hsxt.ui.business.ToolPurchaseActivity"

android:screenOrientation="portrait"

android:windowSoftInputMode="stateAlwaysHidden|adjustPan"/>

最后再次运行，点击输入框就弹出软键盘而当前的输入框也能直接输入字符了。

如果有这样的用户交互，即当滑动ListView的时候，软键盘会自动关闭，无需手动点击关闭软键盘。实现的原理是给ListView设置滚动监听，当当前状态为滚动时，隐藏软键盘，代码如下：

listView.setOnScrollListener(new AbsListView.OnScrollListener() {

@Override

public void onScrollStateChanged(AbsListView view, int scrollState) {

switch (scrollState) {

case AbsListView.OnScrollListener.SCROLL\_STATE\_IDLE://当停止滚动时

break;

case AbsListView.OnScrollListener.SCROLL\_STATE\_TOUCH\_SCROLL://滚动时

//没错，下面这一坨就是隐藏软键盘的代码

((InputMethodManager)getSystemService(INPUT\_METHOD\_SERVICE)).hideSoftInputFromWindow(ToolPurchaseActivity.this.getCurrentFocus().getWindowToken(), InputMethodManager.HIDE\_NOT\_ALWAYS);

break;

case AbsListView.OnScrollListener.SCROLL\_STATE\_FLING: //手指抬起，但是屏幕还在滚动状态

break;

}

}

@Override

public void onScroll(AbsListView view, int firstVisibleItem, int visibleItemCount, int totalItemCount) {

}

});

至此，完美解决了有关ListView含有EditText输入框出现了各种情况的问题。