**base-adapter**

https://github.com/hongyangAndroid/baseAdapter

Android 万能的Adapter for ListView,RecyclerView,GridView等，支持多种Item类型的情况。

**引入**

**ForRecyclerView**

compile 'com.zhy:base-rvadapter:3.0.2'

**ForListView**

compile 'com.zhy:base-adapter:3.0.2'

**使用**

**（1）简单的数据绑定(ListView与其使用方式一致)**

首先看我们最常用的单种Item的书写方式：

mRecyclerView.setAdapter(new CommonAdapter<String>(this, R.layout.item\_list, mDatas)

{

@Override

public void convert(ViewHolder holder, String s)

{

holder.setText(R.id.id\_item\_list\_title, s);

}

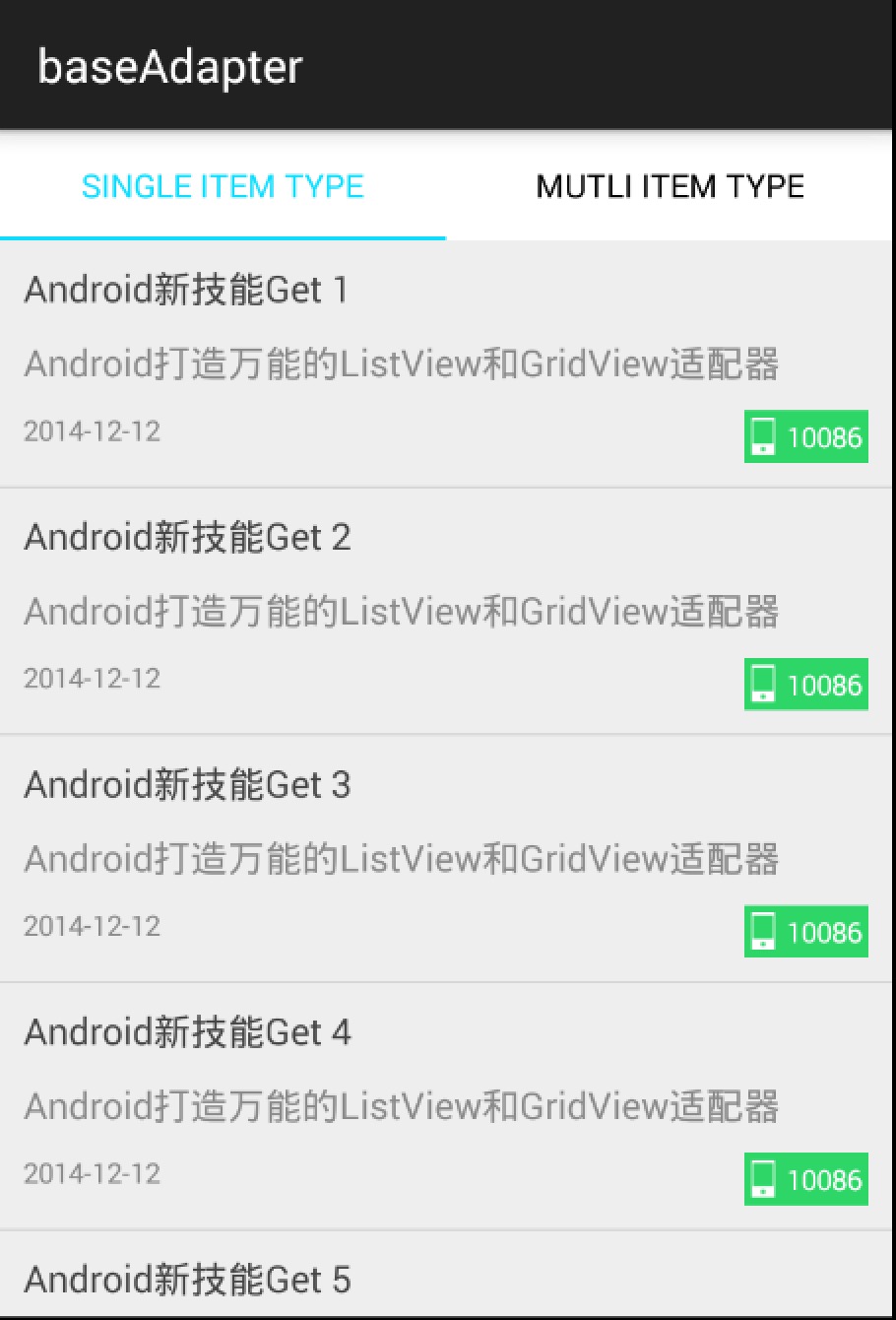
});

是不是相当方便，在convert方法中完成数据、事件绑定即可。

只需要简单的将Adapter继承CommonAdapter，复写convert方法即可。省去了自己编写ViewHolder等大量的重复的代码。

* 可以通过holder.getView(id)拿到任何控件。
* ViewHolder中封装了大量的常用的方法，比如holder.setText(id,text)，holder.setOnClickListener(id,listener)等，可以支持使用。

效果图：

[](https://github.com/hongyangAndroid/baseAdapter/blob/master/screenshot/single.png)

**（2）多种ItemViewType(ListView与其使用方式一致)**

对于多中itemviewtype的处理参考：<https://github.com/sockeqwe/AdapterDelegates> ，具有极高的扩展性。

MultiItemTypeAdapter adapter = new MultiItemTypeAdapter(this,mDatas);

adapter.addItemViewDelegate(new MsgSendItemDelagate());

adapter.addItemViewDelegate(new MsgComingItemDelagate());

每种Item类型对应一个ItemViewDelegete，例如：

public class MsgComingItemDelagate implements ItemViewDelegate<ChatMessage>

{

@Override

public int getItemViewLayoutId()

{

return R.layout.main\_chat\_from\_msg;

}

@Override

public boolean isForViewType(ChatMessage item, int position)

{

return item.isComMeg();

}

@Override

public void convert(ViewHolder holder, ChatMessage chatMessage, int position)

{

holder.setText(R.id.chat\_from\_content, chatMessage.getContent());

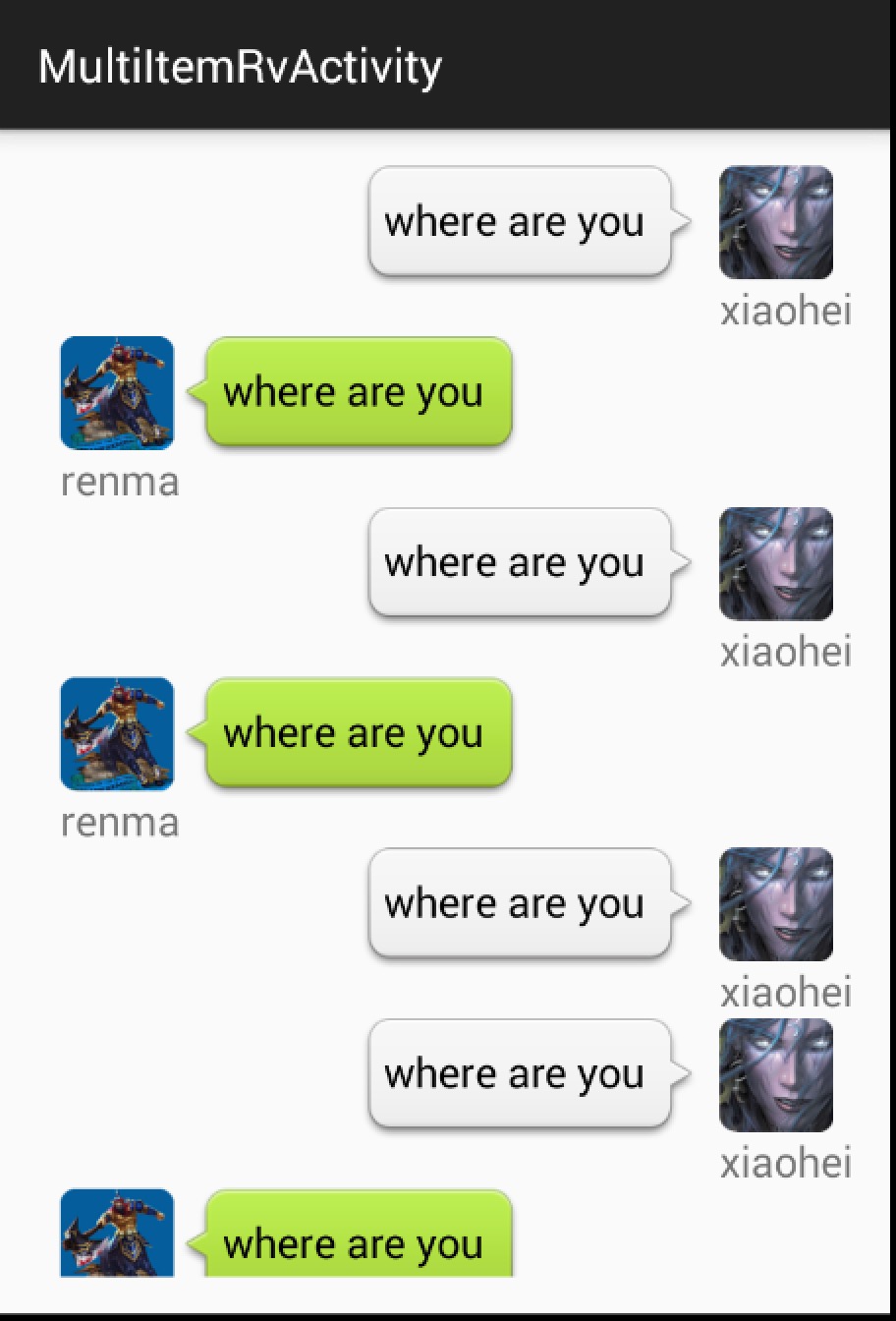
holder.setText(R.id.chat\_from\_name, chatMessage.getName());

holder.setImageResource(R.id.chat\_from\_icon, chatMessage.getIcon());

}

}

贴个效果图：

[](https://github.com/hongyangAndroid/baseAdapter/blob/master/screenshot/rvadapter_01.png)

**(3) 添加HeaderView、FooterView**

mHeaderAndFooterWrapper = new HeaderAndFooterWrapper(mAdapter);

TextView t1 = new TextView(this);

t1.setText("Header 1");

TextView t2 = new TextView(this);

t2.setText("Header 2");

mHeaderAndFooterWrapper.addHeaderView(t1);

mHeaderAndFooterWrapper.addHeaderView(t2);

mRecyclerView.setAdapter(mHeaderAndFooterWrapper);

mHeaderAndFooterWrapper.notifyDataSetChanged();

类似装饰者模式，直接将原本的adapter传入，初始化一个HeaderAndFooterWrapper对象，然后调用相关API添加。

**(4) 添加LoadMore**

mLoadMoreWrapper = new LoadMoreWrapper(mOriginAdapter);

mLoadMoreWrapper.setLoadMoreView(R.layout.default\_loading);

mLoadMoreWrapper.setOnLoadMoreListener(new LoadMoreWrapper.OnLoadMoreListener()

{

@Override

public void onLoadMoreRequested()

{

}

});

mRecyclerView.setAdapter(mLoadMoreWrapper);

直接将原本的adapter传入，初始化一个LoadMoreWrapper对象，然后调用相关API即可。

**(5)添加EmptyView**

mEmptyWrapper = new EmptyWrapper(mAdapter);

mEmptyWrapper.setEmptyView(R.layout.empty\_view);

mRecyclerView.setAdapter(mEmptyWrapper );

直接将原本的adapter传入，初始化一个EmptyWrapper对象，然后调用相关API即可。

支持链式添加多种功能，示例代码：

mAdapter = new EmptyViewWrapper(

new LoadMoreWrapper(

new HeaderAndFooterWrapper(mOriginAdapter)));

**感谢**

* <https://github.com/JoanZapata/base-adapter-helper>

应该是我所知道的最早的对listview,gridview封装的adapter了，参考了其中很多，比如在ViewHolder.setXXX类的辅助方法。

* <https://github.com/sockeqwe/AdapterDelegates>

参考该库多种ItemType绑定数据方式

* <https://github.com/ragunathjawahar/simple-section-adapter>

SectionAdapter对其进行了参考。