



链滴

SimpleRecycler

作者: [xiaoqinghong](#)

原文链接: <https://ld246.com/article/1553059443091>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

SimpleRecycler

APK: [Demo下载](#)

项目地址: [转到GitHub](#)

step:

步骤1.将JitPack存储库添加到构建文件中

```
allprojects {  
    repositories {  
        maven { url 'https://jitpack.io' }  
    }  
}
```

步骤2.添加依赖项

```
dependencies {  
    implementation 'com.github.xiaoqinghong:SimpleRecycler:0.0.2'  
}
```

SimpleAdapter (只支持一种itemView)

```
public class TestAdapter extends SimpleRecyclerAdapter<String> {  
  
    public TestAdapter(List<String> list) {  
        super(R.layout.item_test, list);  
    }  
  
    @Override  
    protected void bind(SimpleViewHolder helper, String item) {  
        helper.setText(R.id.tv_test_item, item)  
            .addClickListener(R.id.tv_test_item) // 设置子项的点击事件监听  
            .addLongClickListener(R.id.tv_test_item) // 设置子项的长按事件监听  
            .getView(R.id.tv_test_item);  
    }  
}
```

MultipleAdapter (支持多种itemView)

```
/**  
 * bean类需要实现SimpleMultipleAdapter.TypeBind接口。每个bean类都可能属于不同的type  
 */  
public class MultipleBean implements SimpleMultipleAdapter.TypeBind {  
    @Override  
    public int type() {  
        return 0;  
    }  
}  
/**  
 * MultipleTestAdapter
```

```

*/
public class MultipleTestAdapter extends SimpleMultipleAdapter<MultipleBean> {
    public MultipleTestAdapter(List<MultipleBean> list) {
        super(list);
        // 添加多种item对应的layout。
        addLayout(R.layout.multiple_item_a, 0);
        addLayout(R.layout.multiple_item_b, 1);
        addLayout(R.layout.multiple_item_c, 2);
    }

    @Override
    protected void multipleBind(SimpleViewHolder helper, MultipleBean item, int itemType) {
        switch (itemType) {
            case 0:
                helper.setText(R.id.tvMultipleA, "this is type " + item.getName());
                break;
            case 1:
                helper.setText(R.id.tvMultipleB, "this is type " + item.getName());
                break;
            case 2:
                helper.setText(R.id.tvMultipleC, "this is type " + item.getName());
                break;
            default:
                break;
        }
    }
}

```

adapter暴露的接口

```

mAdapter.bindRecyclerView(recyclerView); // 内部默认使用LinearLayoutManager
mAdapter.addHeaderView(header);
mAdapter.addFooterView(footer);
// click
mAdapter.setOnItemClickListener(new SimpleRecyclerAdapter.OnItemClickListener() {
    @Override
    public void onClick(View v, int position) {
    });
});

mAdapter.setOnSubViewClickListener(new SimpleRecyclerAdapter.OnSubViewClickListener() {
    @Override
    public void onClick(View v, int position) {
    });
});

// long click
mAdapter.setOnItemLongClickListener(new SimpleRecyclerAdapter.OnItemLongClickListener()
{
    @Override
    public void onLongClick(View v, int position) {

```

```
        }
    });

mAdapter.setOnSubViewLongClickListener(new SimpleRecyclerAdapter.OnSubViewLongClickListener() {
    @Override
    public void onLongClick(View v, int position) {
        }
});
```