[×](https://blog.csdn.net/fomin_zhu/article/details/50397560)

# PopupWindow返回参数至Activity两种方式：接口和Rxjava

# 使用Rxjava回调参数至Activity

定义一个继承PopupWindow的类

public class PopupWindows extends PopupWindow implements View.OnClickListener {

public PopupWindows (View parent, final Context context) {

super(context);

View view = View.inflate(context, R.layout.ppw\_groups\_join, null);

//设置PopupWindows 显示动画

setWidth(WindowManager.LayoutParams.MATCH\_PARENT);

setHeight(WindowManager.LayoutParams.MATCH\_PARENT);

setFocusable(true);

setOutsideTouchable(true);

setContentView(view);

showAtLocation(parent, Gravity.BOTTOM, 0, 0);

update();

view.findViewById(R.id.btn1).setOnClickListener(this);

}

@Override

public void onClick(View v) {

switch (v.getId()){

case R.id.btn1:

Observable.just(1).subscribe(MainActivity.action1);

break;

}

dismiss();

}

}

在MainActivity定义Action1方法，需要先实例化

public class MainActivity extends Activity {

private Button btn1;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btn1=(Button) findViewById(R.id.btn1);

btn1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

new PopupWindows(btn1, MainActivity.this);

}

});

}

public static Action1<int> action1 = new Action1<int>() {

@Override

public void call(int i) {

System.out.println(i);

}

};

}

# 使用接口回调参数至Activity

定义继承PopupWindow类

public class JoinPopupWindows extends PopupWindow implements View.OnClickListener{

public PopupWindows (View parent, final Context context) {

super(context);

View view = View.inflate(context, R.layout.ppw\_groups\_join, null);

//设置PopupWindows 显示动画

setWidth(WindowManager.LayoutParams.MATCH\_PARENT);

setHeight(WindowManager.LayoutParams.MATCH\_PARENT);

setFocusable(true);

setOutsideTouchable(true);

setContentView(view);

showAtLocation(parent, Gravity.BOTTOM, 0, 0);

update();

view.findViewById(R.id.btn1).setOnClickListener(this);

}

@Override

public void onClick(View v) {

switch (v.getId()){

case R.id.btn1:

test.getTest(1);

break;

}

dismiss();

}

private ITest test;

public void setTest(ITest test) {

this.test = test;

}

public interface ITest {

void getTest(int i);

}

}

在MainActivity继承ITest接口

public class MainActivity extends Activity implements ITest {

private Button btn1;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btn1=(Button) findViewById(R.id.btn1);

btn1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

new PopupWindows(btn1, MainActivity.this).setTest(MainActivity.this);

}

});

}

@Override

public void getTest(int i) {

System.out.println(i);

}

}

## 不良信息举报

窗体顶端

举报内容：

PopupWindow返回参数至Activity两种方式：接口和Rxjava

举报原因：

色情 政治 抄袭 广告 招聘 骂人 其他

原文地址：



原因补充：



最多只允许输入30个字



窗体底端