1. **引入Module(CustomDialog)**

**方法：**

1. **、导入module**

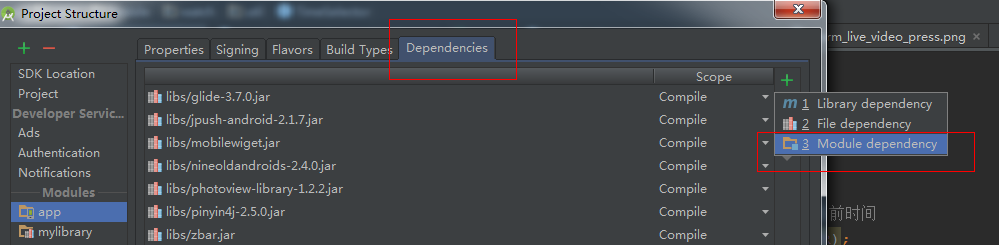
在Android studio中点击File->New->Import Module

选择module所在区域，最终点击finish。

1. **引入module**

File->Project Structure

选择Modules区域下的app,然后选择最后的Dependencies,在点击右方的加号，选择Module dependency,选择刚才导入的module即可。



1. **初始化**

CustomDialogs Dialog = new CustomDialogs(Context context);

Dialog .show

参数说明：

Context 上下文

1. **方法说明**
2. **、监听确定/取消点击事件：**

dialog.setListener(new CustomDialogs.OnCloseListener() {

@Override

public void onClickOK(Dialog dialog) {//确定}

@Override

public void onClickNo(Dialog dialog) {//取消}});

1. **设置是否可以点击空白区域取消弹框：**

dialog.setOnCloseListener(boolean truefalse);

默认不可以点击取消 设置true弹框变为可点击取消

1. **自定义弹框标题：**

dialog.setTitle(String title);

1. **自定义显示内容：**

dialog.setContent(String content);

1. **自定义确定按钮名字：**

dialog.setPositiveButton(String name);

1. **自定义取消按钮名字：**

dialog.setNegativeButton(String name);

1. **是否显示取消按钮：**

dialog.setCancleButton(boolean truefalse);

默认不显示 设置true显示

1. **显示为编辑框：**

dialog.setEdit(boolean truefalse);

默认不显示 设置true提示框内显示为编辑框

1. **设置编辑框hint内容：**

dialog.setEditHint(String hint);

1. **设置编辑输入类型:**

dialog.setEditType(int type);

1.普通文本 2.密码

1. **设置编辑框监听事件：**

dialog.setEditListener(new CustomDialogs.OnClickEditDialogListener() {

@Override

public void onClick(Dialog dialog, EditText editText) { //确定 }});

1. **设置自定义布局弹框：**

dialog.setCustomLayout(Int layoutID);

1. **设置底部弹出框：**

final CustomDialogs dialog = new CustomDialogs(Context context);

dialog.setBottomDialogList(String name, new View.OnClickListener() {

@Override

public void onClick(View v) {//点击}});

底部弹框的dialog 需设置final

name: 弹出行名字

1. **取消弹框:**

dialog.diadismiss();