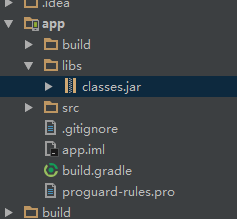
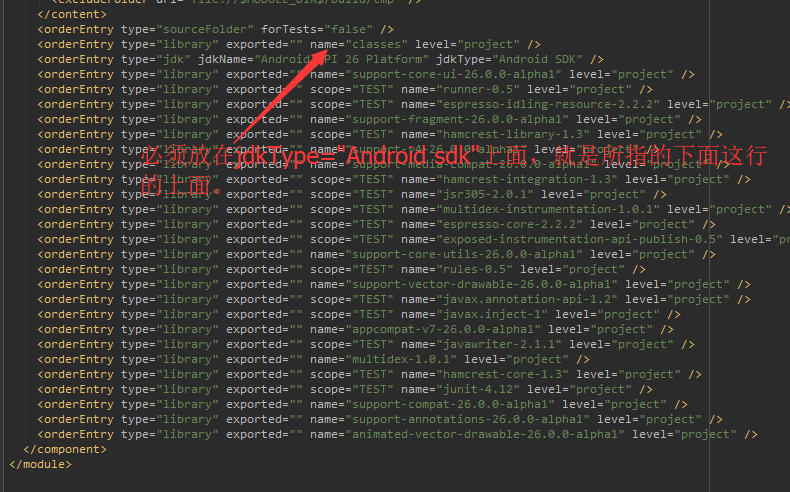
1,开发环境：AndroidStudio

2,导入classes.jar到libs目录下



3,同步jar包以后，打开上图中的app.iml



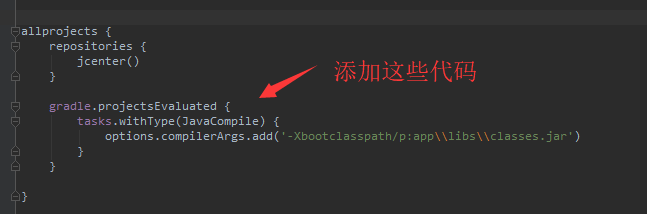
本来这行应该是在中间位置的，现在把它放在

<orderEntry type="jdk" jdkName="Android API 26 Platform" jdkType="Android SDK" />

的前面，然后保存。

注意：当你下次打开这个项目的或者修改了某一些地方后需要Sypc Now的时候又要手动的把它放在上面去(因为你会发现，它又会跑到中间去)。不然代码会报错。

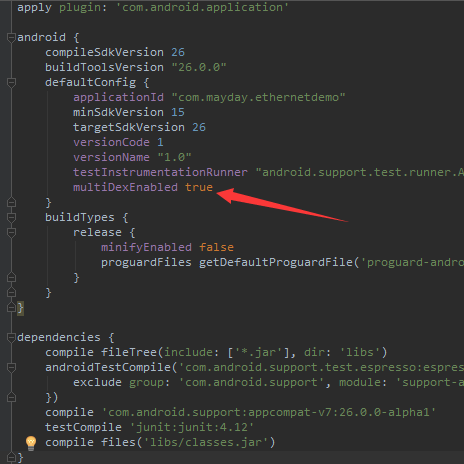
4,build.gradle修改(整个项目的)



直接copy过去就ok(注意位置)

gradle.projectsEvaluated {  
 tasks.withType(JavaCompile) {  
 options.compilerArgs.add('-Xbootclasspath/p:app\\libs\\classes.jar')  
 }  
}

5,build.gradle(当前model下的)



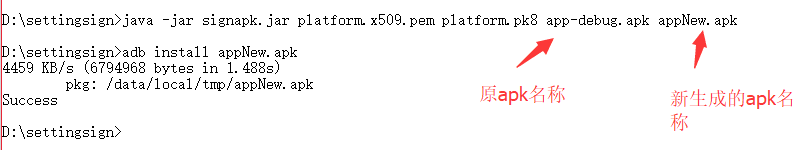
添加改行: multiDexEnabled true

6,添加相应的权限

<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />  
<uses-permission android:name="android.permission.CHANGE\_WIFI\_STATE" />  
<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />  
<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.CHANGE\_NETWORK\_STATE" />  
<uses-permission android:name="android.permission.BROADCAST\_STICKY" />  
<uses-permission android:name="android.permission.READ\_SETTINGS" />  
<uses-permission android:name="android.permission.BLUETOOTH" />  
<uses-permission android:name="android.permission.BLUETOOTH\_ADMIN" />  
<uses-permission android:name="android.permission.MODIFY\_PHONE\_STATE" />  
<uses-permission android:name="android.permission.CONNECTIVITY\_INTERNAL" />  
  
<uses-permission android:name="android.permission.WRITE\_SETTINGS" />  
<uses-permission android:name="android.permission.WRITE\_SECURE\_SETTINGS" />  
<uses-permission android:name="android.permission.INTERACT\_ACROSS\_USERS"/>  
<uses-permission android:name="android.permission.INTERACT\_ACROSS\_USERS\_FULL"/>  
  
<uses-permission android:name="android.permission.READ\_PHONE\_STATE"/>

都加上吧！这是之前项目需要的权限，有些可能用不到。

7，系统签名（必须通过系统签名后才能正常使用！）



8,jar包以及demo会在文档中给出。

9,调用方式：

EthernetMain ethernetMain=new EthernetMain(Context context); //参数：上下文

ethernetMain.openEth();

ethernetMain.closeEth();

ethernetMain.dhcpEth();

//参数: IP 子网掩码 DNS1 DNS2 默认网关

ethernetMain.staticEth("192.168.1.222","255.255.255.0","192.168.1.1","192.168.1.1","192.168.1.1"

);

详细参考Demo项目!