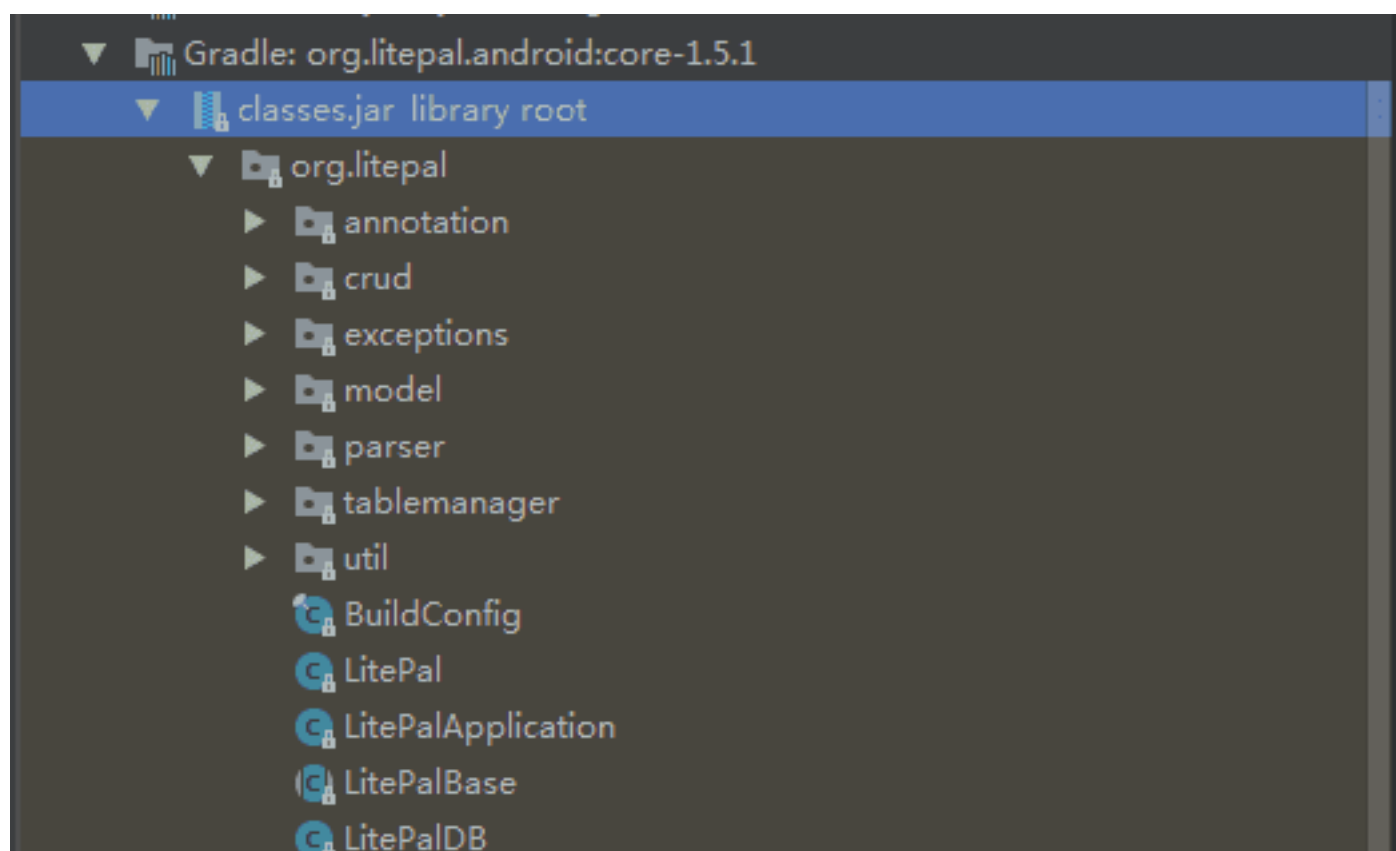


android studio 升级3.1版本导包爆红问题

先上两张图：

```
import org.litepal.crud.DataSupport;  
import java.io.Serializable;  
  
public class DownloadVideo extends DataSupport implements Serializable {
```

[导包爆红.png]



external libraries中已加载.png

这是 AS 升级 3.1 后出现的问题，很奇怪明明 lib 已经加载了为什么项目中导入却报错。出现问题了就要解决。

1. 经验告诉我们遇到爆红首先 clean 或 rebuild。结果：失败。

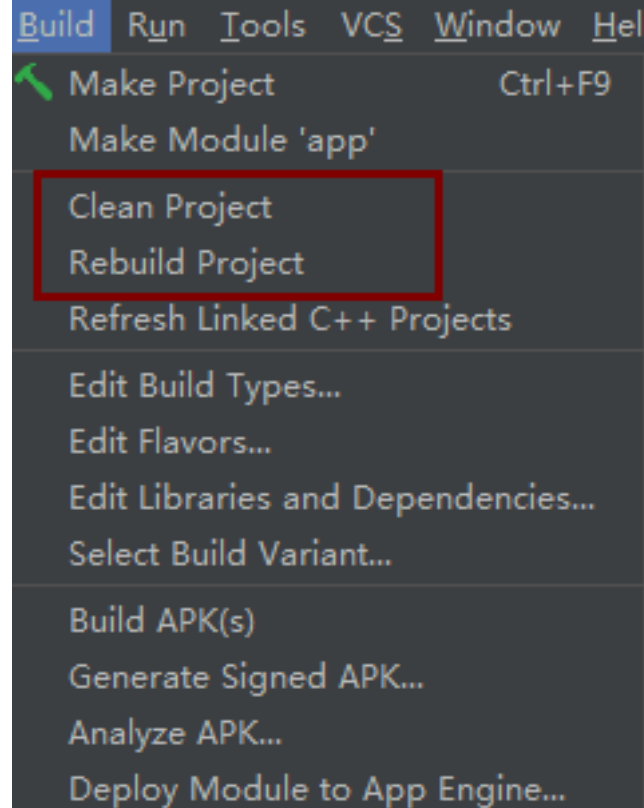
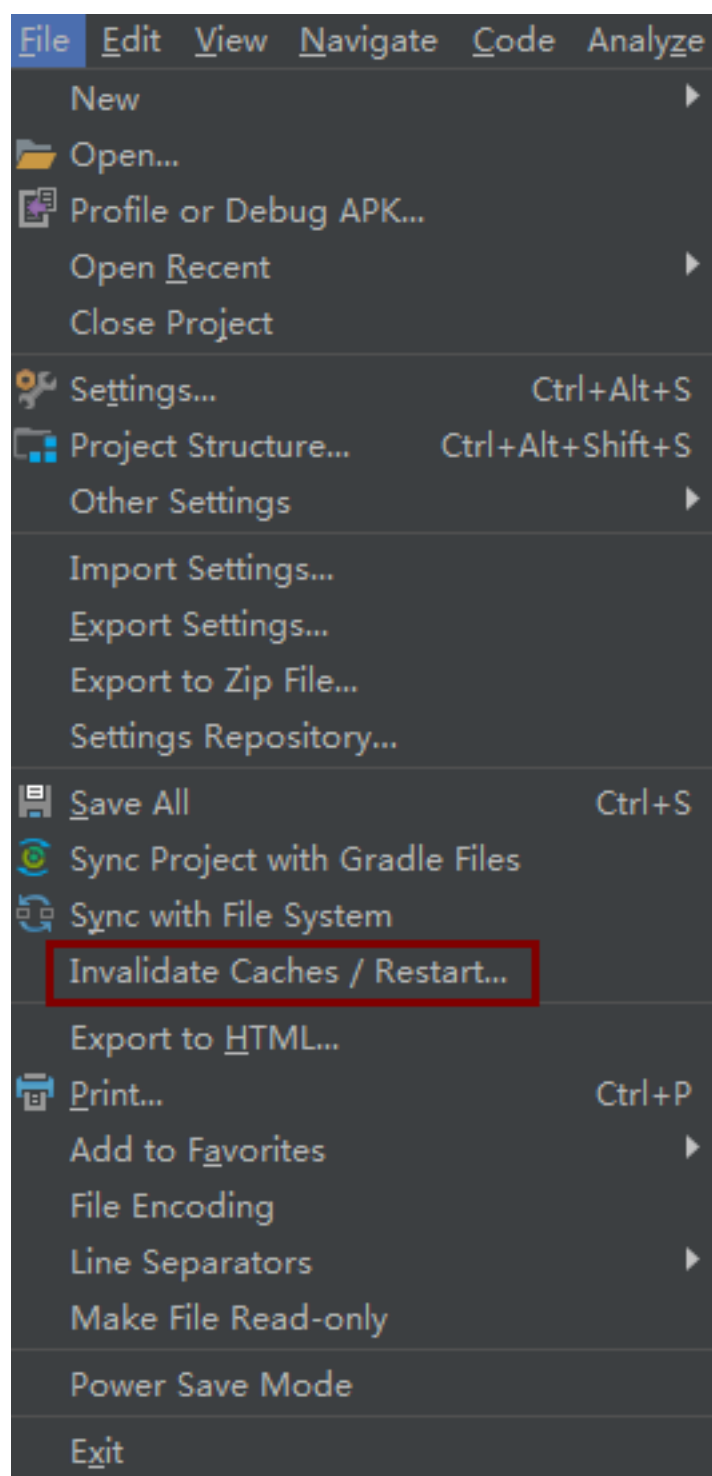


image.png

2. 继续清楚缓存，重启 AS。结果：失败。



3. 求助 搜索引擎，有说删除 .gradle 文件，重新加载，我还真试了，花了半天时间证明不行。结果：失败。
4. 绝望之际想想或许从官网能找到答案。还真找到了答案：

使用新依赖项配置

Gradle 3.4 引入了新的 [Java 库插件配置](#)，允许您控制到编译和运行时类路径的发布（适用于模块间依赖项）。Android 插件 3.0.0 正在迁移到这些新依赖项配置。要迁移您的项目，只需更新您的依赖项以使用新配置，而非已弃用配置，如下表中所列。

新配置	已弃用配置	行为
implementation	compile	依赖项在编译时对模块可用，并且仅在运行时对模块的消费者可用。对于大型多项目构建，使用 implementation 而不是 api/compile 可以显著缩短构建时间，因为它可以减少构建系统需要重新编译的项目量。大多数应用和测试模块都应使用此配置。
api	compile	依赖项在编译时对模块可用，并且在编译时和运行时还对模块的消费者可用。此配置的行为类似于 compile（现在已弃用），一般情况下，您应当仅在库模块中使用它。应用模块应使用 implementation，除非您想要将其 API 公开给单独的测试模块。
compileOnly	provided	依赖项仅在编译时对模块可用，并且在编译或运行时对其消费者不可用。此配置的行为类似于 provided（现在已弃用）。
runtimeOnly	apk	依赖项仅在运行时对模块及其消费者可用。此配置的行为类似于 apk（现在已弃用）。

就像当前稳定版本的 Android 插件一样，上面的配置对风味或构建类型特定的依赖项可用。例如，您可以使用 api 让依赖项对所有变体可用，也可以使用 redApi 让其仅对模块的 red 变体可用。

注：compile、provided 和 apk 目前仍然可用。不过，它们将在下一个主要版本的 Android 插件中消失。

从图中一下可以看出来：

- implementation 依赖项在编译时对模块可用，并且仅在运行时对模块的消费者可用。
- api 依赖项在编译时对模块可用，并且在编译时和运行时还对模块的消费者可用。

问题找到了，我们项目 app 中引入本地 library，而 library 中用 implementation 方式引用了很多第三方库，如官方描述那样，其实 implementation 只对改模块可用，所以我们的 app 导入不了 library 加载的第三方库。那么只要把 library 中的 implementation 依赖方式改成 api 依赖方式即可。

后来从 万能的 [stackoverflow](#) 中也找到了答案，第二个回答详细说明了 implementation 与 api 两种依赖方式的不同。

5. 总结，本来一个很小的问题却花费了我差不多一天的时间，究其原因还是

没有认真阅读官方文档。太凭感觉干活，这个要不得，以后切记先阅读官方文档。

参考：<https://stackoverflow.com/questions/44493378/whats-the-difference-between-implementation-and-compile-in-gradle>
https://developer.android.google.cn/studio/build/gradle-plugin-3-0-0-migration.html#new_configurations