

奥特曼超人的博客专栏

<http://www.immqy.com>; 偶尔才逛CSDN->有问题请关注GitHub的Q群

[目录视图](#)[摘要视图](#)[RSS 订阅](#)

图灵赠书——程序员11月书单 【思考】Python这么厉害的原因竟然是！ 感恩节赠书：《深度学习》等另
作译者评选启动！ 每周荐书：京东架构、Linux内核、Python全栈

Android studio 中引用jar的其实是Maven ? (一)

标签：[android](#) [maven](#) [gradle](#) [jar](#) [android studio lib](#)

2016-03-21 20:11

1881人阅读

评论(0)

收藏

分类：

[Android 技术汇总 \(148\)](#)

版权声明：本文为博主杜锦阳原创文章，未经博主允许不得转载，如有侵权将依法追究其法律责任。

由于Studio比eclipse多了一步对工程构建的步骤，即为build.gradle这个文件运行，因此其引入第三方开发jar包与lib工程对比Eclipse已完成不同，引入第三方jar与lib工程显得并没有那么友好了，因为过程中会涉及更多内容，而且你必须认清楚studio的目录结构。

举个例子：

下图是我在git上面下载的一个工程，工程中包括了一个lib工程和第三方jar包。

Wechat：sheep0704

Author：KARL-Dujinyang

Offer QQ：1875125470

BAT QQ：309933706

游戏移动技术精英Q群：Q群主邀请

老司机Q群：Q群主邀请

深度学习Q群：325296553

破解逆向Q群：暂不开放

安卓移动技术精英Q群：246231638

个人资料



奥特曼超人Dujinyang

关注

发私信



访问：1183444次

积分：11405

等级：BLOG > ?

排名：第1517名

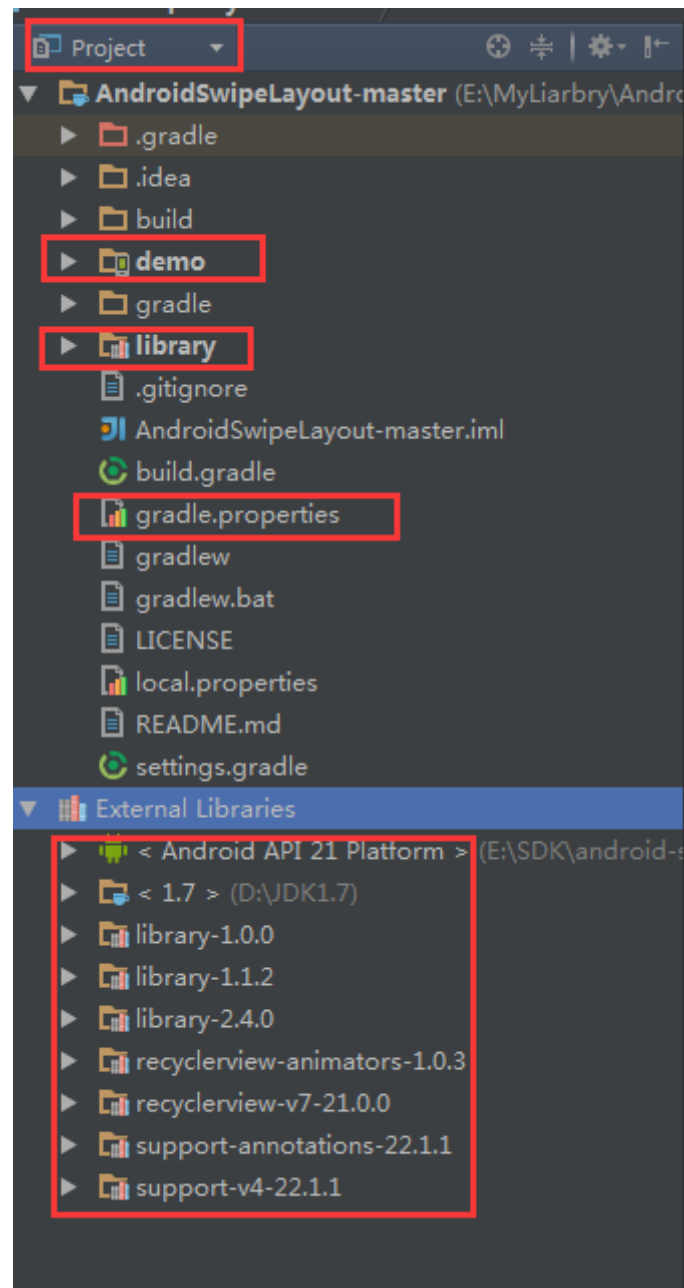
原创：128篇

转载：96篇

译文：25篇

评论：116条

博客专栏



以上为一个工程的project视图,由这个视图可以看出,其中有两个代码模块,一个是demo,一个是library,这两个



大数据

文章：10篇

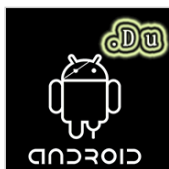
阅读：167021



JAVA设计模式

文章：20篇

阅读：97955



Android-黑客的世界

文章：10篇

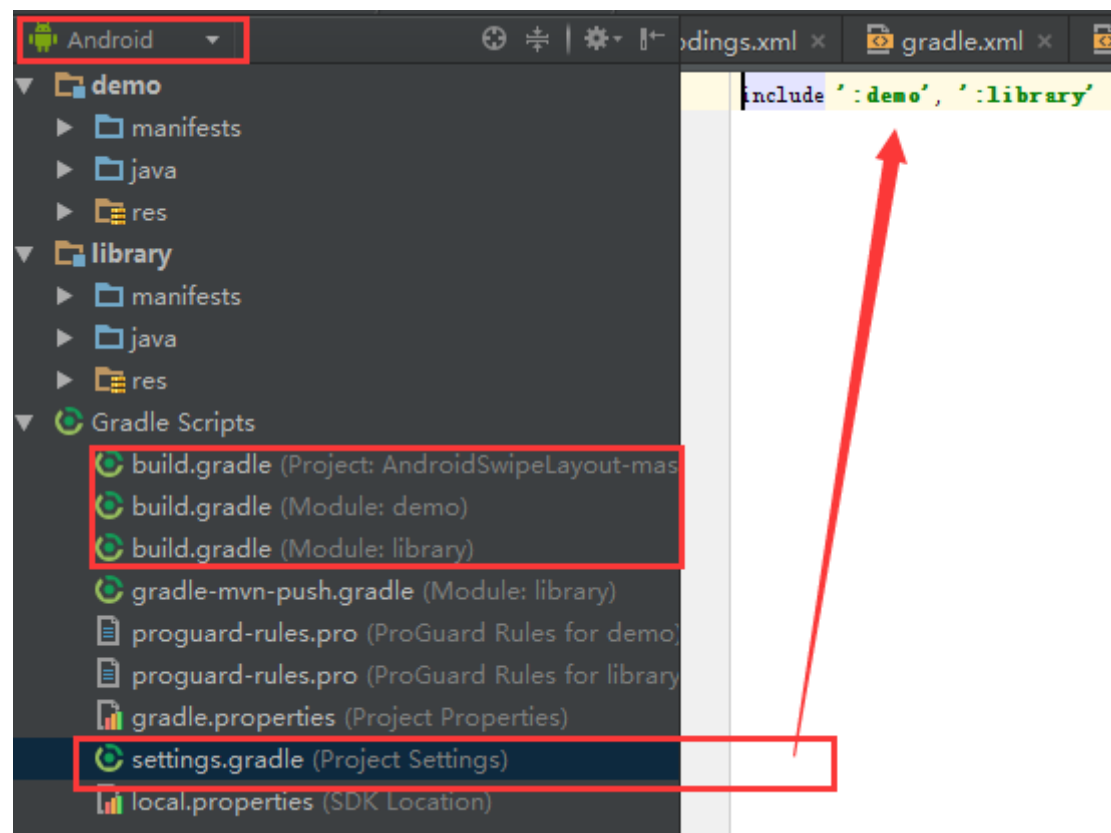
阅读：4926159

文章分类

[联系方式](#) (1)[Bug 疑难杂症 | 不分语言](#) (7)[-](#) (0)[JAVA 技术汇总](#) (20)[JAVA 入坑教程 | 基础大全](#) (12)[-](#) (0)[Android 技术汇总](#) (149)[Android 黑客的世界](#) (12)[-](#) (0)[Python 技术汇总](#) (4)[IOS 技术汇总](#) (22)[C 技术汇总](#) (2)[C++ 技术汇总](#) (1)[-](#) (0)

代码模块会被粗体文字标示出来。还有一个大的文件分支：External Libraries,由字面上的意思就是，第三方依赖的库文件。然后，下面一推的jar包。

还有其他的一些文件，一些gradle脚本相关的文件，如果第一次使用studio你会很惊讶，究竟这些文件又啥用呢？其实除了demo和library之外的文件都是这个工程的构建文件，这个工程的建立依赖了这些文件。因此如果还不够明白，我们可以切换到android工程视图来看。



上图是一个工程的android视图，由视图可以看出，该工程下面包含了三个部分：demo，library,Gradle Scripts。

demo是主体工程，library是依赖库，那么Gradle Scripts就是这个工程的构建脚本了。Gradle Scripts下面可以到脚

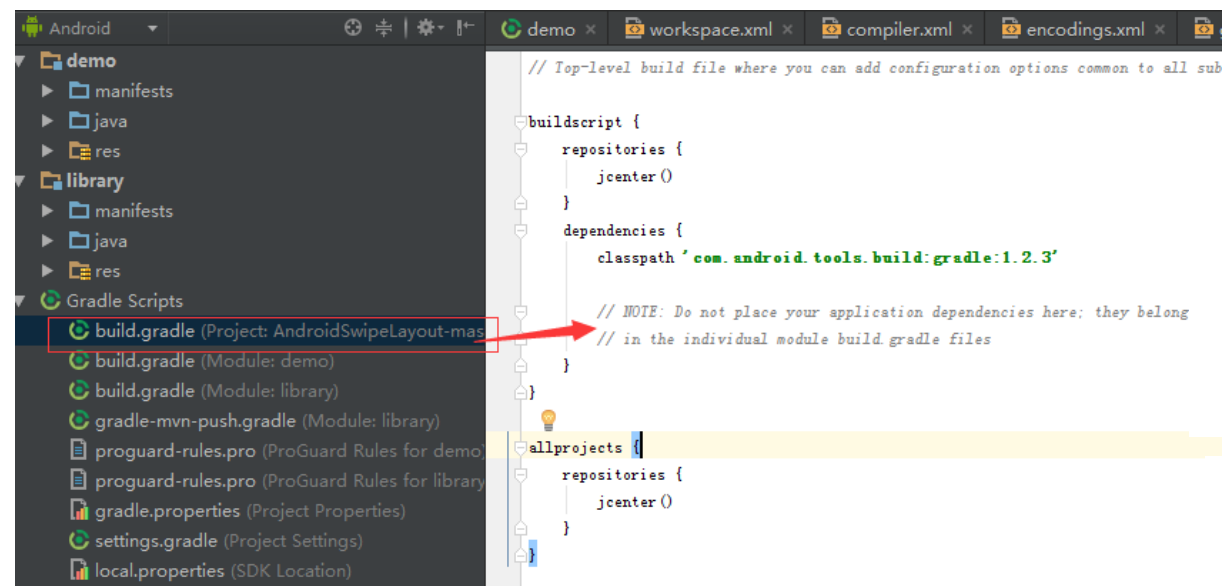
[区块链](#) (1)[机器学习](#) (1)[人工智能](#) (3)[-](#) (0)[Other 非技术类型文章](#) (29)[————— 联系方式 —————](#) (1)

阅读排行

[Android Bluetooth 蓝牙开发资...](#) (75971)[移动开发之【微信小程序】的...](#) (33895)[android MultiDex multidex原理...](#) (26733)[链路层的双链路--大型服务器的..](#) (25229)[android webview 底层实现的逻辑](#) (24416)[Twitter 架构优化之路--Twitter是..](#) (23993)[facebook的Fresco \(FaceBook...](#) (21978)[Fresco源码解析 - DataSource怎...](#) (21966)[android 引导界面的实现](#) (21947)[Fresco源码解析 - 创建一个Ima...](#) (21858)

本的一些属性设置项，settings决定了脚本目标模块是谁，而三个不同的build.gradle则针对不同的模块，针对其后的括号内容。

下图是这个工程的的build.gradle(project)的内容：



由上图可知，负责构建project的gradle脚本，仅仅定义了gradle的版本，并且注释着：

```
<code class="hljs lua has-numbering" style="display: block; padding: 0px; color: inherit; box-sizing: border-box; font-family: monospace; font-size: 1em; white-space: pre; margin: 0;">// NOTE: Do not place your application dependencies here; they belong
// in the individual module build.gradle files
```

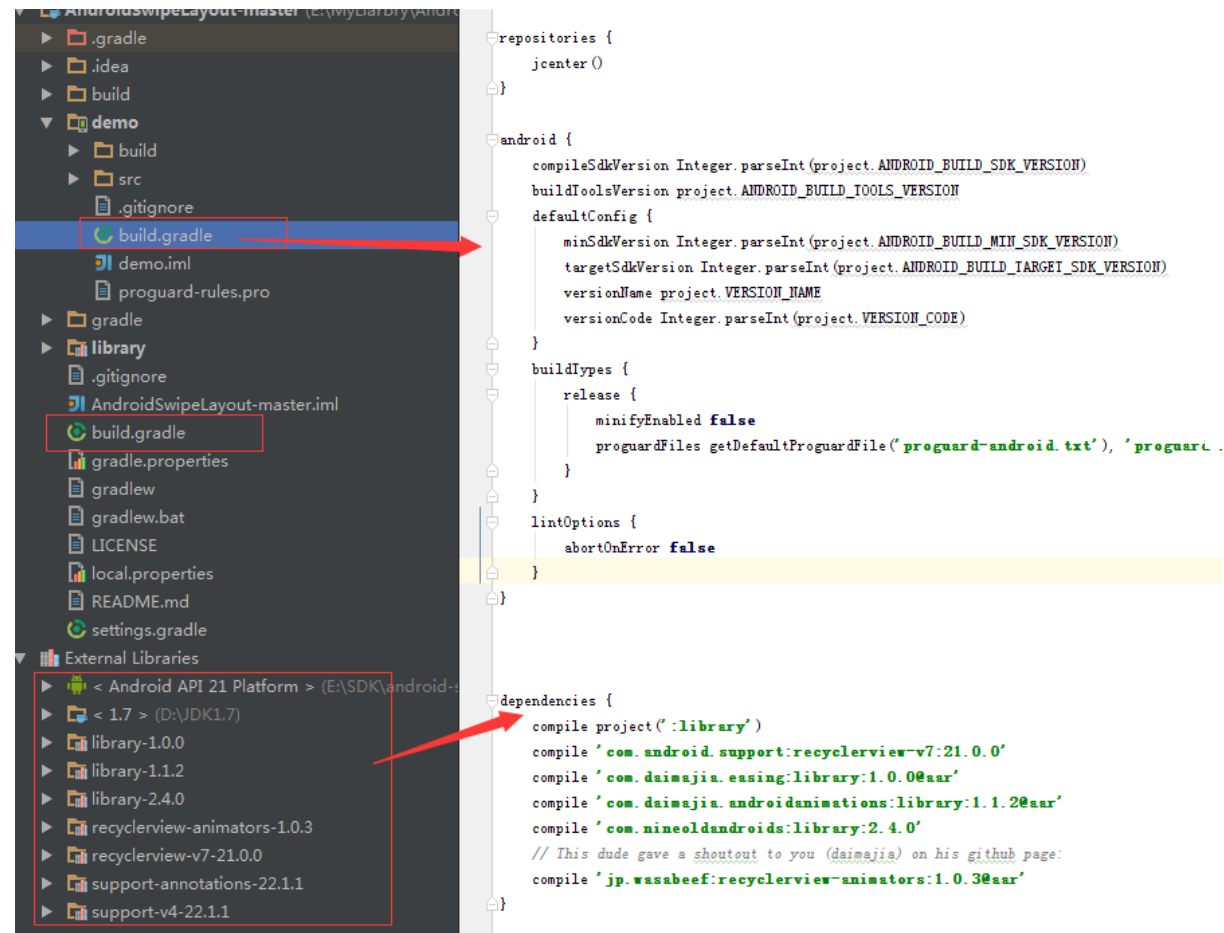
貌似这样看来瞬间对studio的工程结构清晰了很多，可是问题又来了：在android视图下面，第三方android的jar包去哪里了，即External Libraries去哪里？而在project视图下，这些jar又是在哪里下载的？

关于第一个问题，为何在android视图下面没有显示第三方jar包我还不清楚，是studio犯抽了还是故意隐藏掉，不得而知。

关于在哪里下载，这个必须好好找找原因。

在第一次打开使用各个模块的build脚本之后，你会发现，studio开始下载一些工具库，包括在project视图下面的第三方jar包，而这些jar在哪里定义？哪里下载呢？

切换回project视图，你会发现这些第三方jar包都是被定义在各个模块中的脚本中，而这些脚本都引用了仓库，maven仓库，又脚本上段可以看出这个仓库名称”jcenter()“。



既然是一个代码仓库，那么总会有下载地址的吧，所以在脚本里面，按住ctrl,选中”jcenter ()“这个方法，跳到这个方法实现类里面,如图：

```

~ ✓ view
*
* @return the added resolver
* @see #jcenter (Action)
*/
MavenArtifactRepository jcenter ():

/**
 * Adds a repository which looks in the Maven central repository for dependencies. The URL used to access
 * @value org.gradle.api.artifacts.ArtifactRepositoryContainer#MAVEN_CENTRAL_URL}.
 *
 * <p>The following parameter are accepted as keys for the map:
 *
 * <table summary="Shows property keys and associated values">
 * <tr><th>Key</th>

```

找到这个方法定义之后，在往上找，找到该类：

```

/**
 * A @code RepositoryHandler manages a set of repositories, allowing repositories to be d
 */
public interface RepositoryHandler extends ArtifactRepositoryContainer {

/**
 * Adds a resolver that looks into a number of directories for artifacts. The artifacts
 * root of the specified directories. The resolver ignores any group/organization inform
 * dependency section of your build script. If you only use this kind of resolver you mi
 * dependencies like <code>"junit:4.4" instead of <code>"junit:junit:4.4"
 *
 * The following parameter are accepted as keys for the map:
 *
 * <table summary="Shows property keys and associated values">

```

ctrl+点击该类，你会发现惊喜。没错，仓库的地址定义在这个类ArtifactRepositoryContainer里面：

```
*/  
public interface ArtifactRepositoryContainer extends NamedDomainObjectList<ArtifactRepository>, Configurable<ArtifactRepositoryContainer>  
{  
    String DEFAULT_MAVEN_CENTRAL_REPO_NAME = "MavenRepo";  
    String DEFAULT_MAVEN_LOCAL_REPO_NAME = "MavenLocal";  
    String MAVEN_CENTRAL_URL = "https://repo1.maven.org/maven2/";  
}
```

然后打开以上网址，你会惊讶地发现，这个仓库实在有太多代码块了，按着jar包名称，可以慢慢找到之前定义好的jar包。



至于他们如何对这些jar包进行下载的，有待进一步研究maven仓库。

下一篇：Android studio 中引用jar的其实是Maven ? (二)

顶 踩
1 0

- 上一篇 Android Studio 如何打JAR包

- [下一篇](#) [Android studio 中引用jar的其实是Maven ? \(二\)](#)

相关文章推荐

- [android studio 远程导入jar包 maven](#)
- [MySQL在微信支付下的高可用运营--莫晓东](#)
- [Android Studio 中的maven仓库使用](#)
- [容器技术在58同城的实践--姚远](#)
- [Android studio Maven仓库使用](#)
- [SDCC 2017之容器技术实战线上峰会](#)
- [maven管理Android项目依赖和jar包](#)
- [SDCC 2017之数据库技术实战线上峰会](#)
- [看来真不能直接将.so文件直接添加到Android的JAR...](#)
- [腾讯云容器服务架构实现介绍--董晓杰](#)
- [linux下搭建SVN遇到authentication failed问题的解决...](#)
- [微博热点事件背后的数据库运维心得--张...](#)
- [Android studio 中引用jar的其实是Maven ?](#)
- [Android学习开发之Android Studio-史上最](#)
- [集合对象-“块数据”操作--其实是同一对象](#)

查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)

网站客服

杂志客服

微博客服

webmaster@csdn.net

400-660-0108

| 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved

