



AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极 关注

0.282 2019.04.25 14:58:28 字数 2,871 阅读 1,577

- 发布项目至JCenter(Bintray)报错或版本更新失败等问题请前往[发布项目到Maven\(JCenter\)](#)遇到的错误解决办法

写在前面：

之前在写项目时，喜欢将重复的功能或者自定义控件封装到公用的模块中，方便其他项目共用。当项目越来越多，很多项目都依赖了这个公用模块时，其中某一个项目中对公用模块中的代码进行了修改，此时其他依赖该公共模块的项目不得不做出修改或者同步代码，是不是相当麻烦？因此就不得不将公用模块打包成.aar文件(不清楚.aar文件的童鞋请自行Google或Baidu)，公开或放在私服提供给各个项目使用。这里我们就随着前辈们的脚步来学习如何将项目模块发布至Maven仓库。真正做到只需一处修改，四处升级即可。本文将介绍发布到Maven的三种方式。最终将项目发布远程Maven仓库后添加至JCenter，供其他人使用你的公共库。

1：Maven的定义：

简单来说，Maven是项目（Software Project）管理工具。对于maven来说，一个软件工程的唯一标识是由开发者（groupId）、生成物（artifactId）、版本（version）共同决定的。每个工程都有一个打包类型，可以是jar, aar, war, ear 或 pom，打包类型决定了工程最终产物的类型，其中pom类型用于构件多模块工程。其内部工作机制是通过解析pom.xml文件中的配置获取jar包或aar资源地址自动将资源download到Project中，省去了手动下载到本地的繁琐过程。

2：Maven的存储位置：

远程Maven仓库是存储服务器的物理空间上，本文介绍是将Maven发布到第三方OSS服务商Bintray的存储空间上。

发布代码到Maven的三种方式

一：发布到本地Maven

1：uploadArchives配置和Task生成

在要发布的模块build.gradle中配置uploadArchives属性，示例代码：

```
1 //添加这一行，告诉gradle应用到项目中
2 apply plugin: 'maven'
3 //配置模块未本地Maven仓库
4 uploadArchives{
5     repositories.mavenDeployer{
6         // 配置本地仓库路径，这里是项目的根目录下的maven目录中
7         repository(url: uri('../maven'))
8         // 唯一标识 一般为模块包名 也可其他
9         pom.groupId = "com.android.imusic.player.lib"
10        // 项目名称（一般为模块名称 也可其他
11        pom.artifactId = "music-player-lib"
12        // 发布的版本号
13        pom.version = "1.0.0"
14    }
15 }
```

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

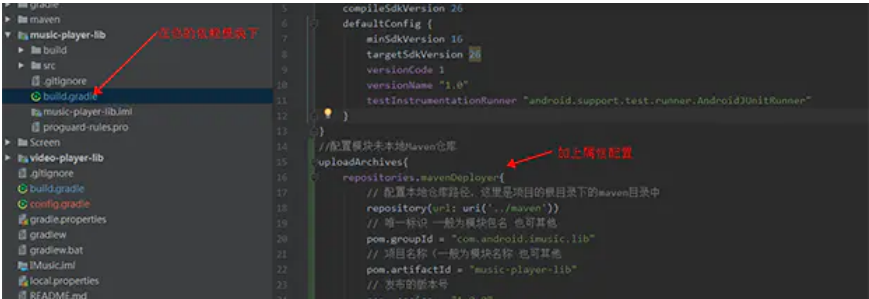
阅读 304

AndroidStudio发布项目到Maven(JCenter)

 撸代码的皇太极

[关注](#)

[赞赏支持](#)

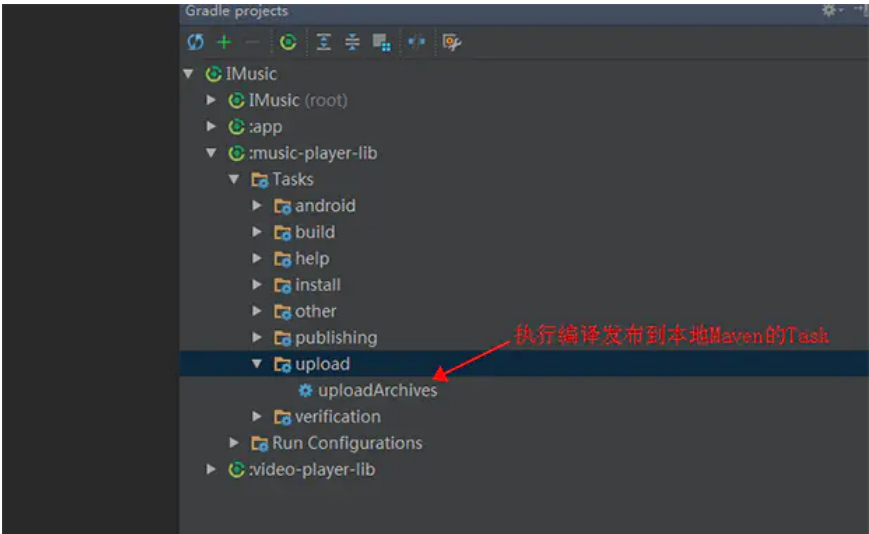


uploadArchives配置截图图示

配置后Sync Now一把，Studio编辑器右上角Gradle中你要发布的模块下会多出一个upload目录，目录中有个uploadArchives任务脚本，这个脚本就是将库发布到本地或私服Maven仓库的Task。

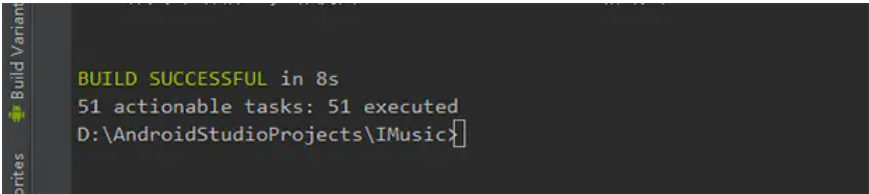
2：执行uploadArchives任务

点击如图所示的uploadArchives



uploadArchives在Studio中图示

uploadArchives有两种运行方法：1：双击运行。2：在AndroidStudio自带的Terminal中执行 gradlew uploadArchives 命令运行。根据喜好选取一种即可，开始任务后等待编译完成。



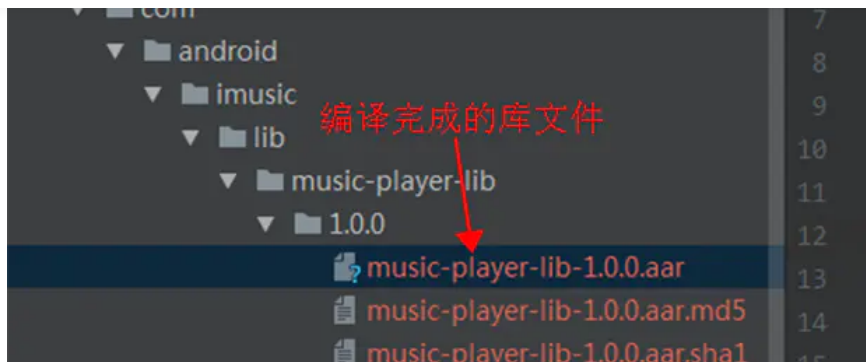
编译到Maven成功图示

编译完成后在你项目的根目录会生成一个maven目录，在目录maven->com.android.imusic.ib-

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

[关注](#)[赞赏支持](#)

模块编译到Maven完成图示

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304

3：应用music-player-lib.aar库文件到项目中

- 3.1：项目根目录build.gradle配置Maven

在你的项目中根目录下的build.gradle中配置如下代码：

```
1 allprojects {
2     repositories {
3         google()
4         jcenter()
5         //本地Maven仓库地址,取决于你的磁盘目录
6         maven {
7             url 'file:///D:/AndroidStudioProjects/IMusic/maven'
8         }
9     }
10 }
```

- 3.2：app中的build.gradle配置

在你想要依赖.aar的模块中的build.gradle中配置如下代码：

```
1 //应用这个依赖,这里填写刚才配置的包名+库名
2 implementation 'com.android.imusic.lib:music-player-lib:1.0.0'
```

- 或者你可以不配置本地仓库，直接将.aar文件复制到你的app模块中直接使用，配置代码如下：

```
1 //在你的app中的build.gradle中配置
2 android {
3     repositories {
4         flatDir {
5             dirs 'libs'
6         }
7     }
8 }
9 //添加依赖
10 implementation(name: 'music-player-lib-1.0.0', ext: 'aar')
```

点击Sync Now等待完成后发布依赖到本地Maven就成功啦！在发布到本地Maven前你应该要校验混淆开启的情况噢~免得他人依赖你的项目时掉坑里去了。发布到本地Maven固然方便，但是公司项目团队人数较多时，将模块打包成.aar文件后复制到项目多少有点不方便，比如所A程序员的本地Maven路径是在D盘，而B程序员的Maven路径又在E盘，这样同步代码改来改去着实难受，所以接下来就介绍适应公司内部团队多人开发的场景，即将代码发布至局域网仓库。

二：发布到局域网Maven

局域网部署需要用到私服部署，这里跟随前辈脚步介绍使用Nexus来搭建部署局域网Maven仓

写下你的评论...

评论1

赞4

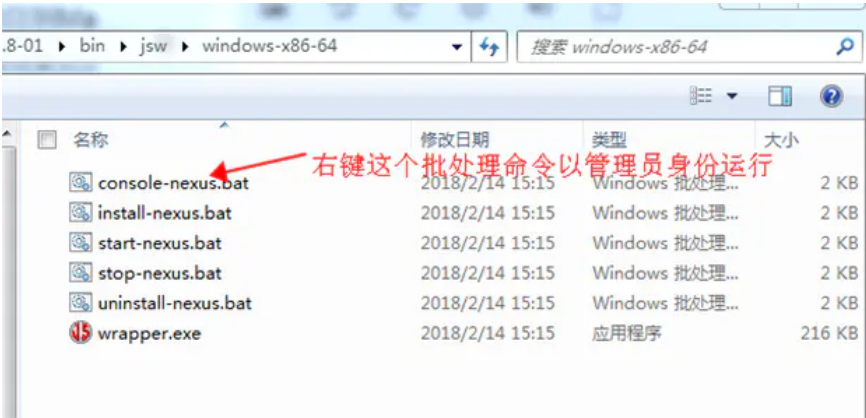
AndroidStudio发布项目到Maven(JCenter)

撸代码的皇太极

[关注](#)

[赞赏支持](#)

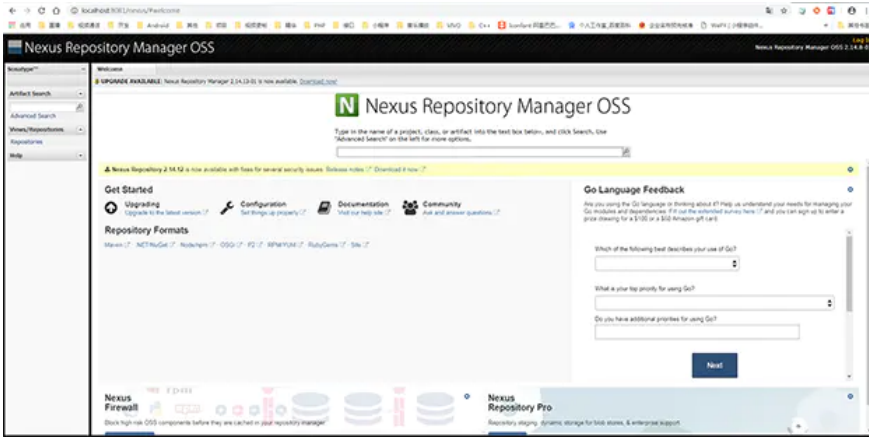
点此前往[下载nexus-2.14.2-01](#)
下载解压后会看到目录下下面几个文件夹



nexus目录文件预览图示

目录下批处理脚本释义：
console-nexus.bat--->常规安装Nexus服务
install-nexus--->启动开机自启动Nexus服务
start-nexus.bat--->启动Nexus服务
stop-nexus.bat--->停止Nexus服务
uninstall-nexus.bat--->卸载Nexus组件
wrapper.exe--->桌面程序

- 鼠标右键"console-nexus.bat"以管理员身份运行，等待CMD创建安装完成后在浏览器输入：<http://localhost:8081/nexus>。
安装和成功启动后是这样的



Nexus启动默认界面图示

2：Nexus仓库配置

- 2.1： 点击右上角Log In， 登录默认账号,默认账号：admin 密码：admin123

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！
阅读 4,690

若依框架springBoot-搭建和部署
阅读 402

Android 检测是否通知栏权限(显示通知)
阅读 211

【Small Tools】Spring Cloud
Alibaba + Vue 前后端分离项目
阅读 309

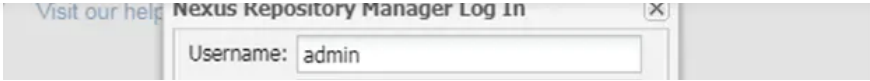
Android Bugly上报过程源码分析
阅读 304

AndroidStudio发布项目到Maven(JCenter)

撸代码的皇太极

关注

赞赏支持



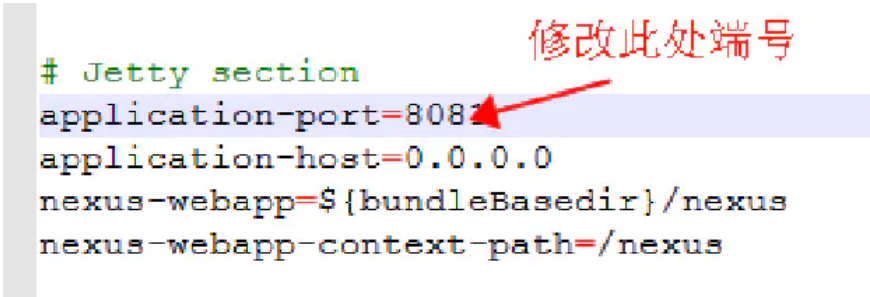
Maven登录图示

- 2.2：修改默认密码



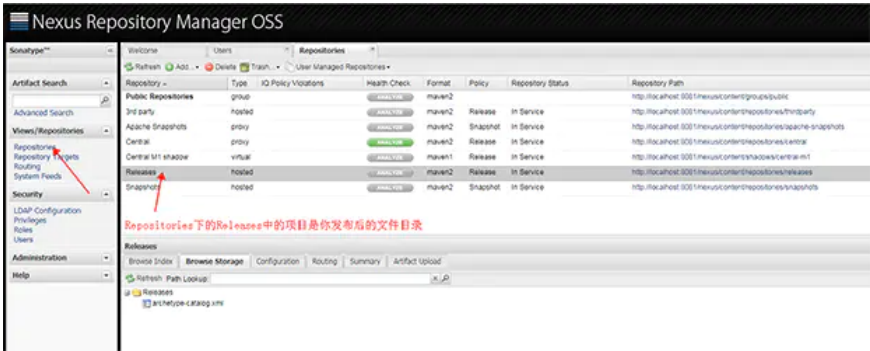
Maven账号默认密码修改图示

按照步骤修改密码即可,如果需要修改默认的端口号，修根目录下的conf-nexus.properties文件中的application-port=8081一栏，将8081替换你指定的端口号：、



Nexus端口号修改图示

配置就绪后，准备发布，发布后你的项目路径如图所示：



AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

[关注](#)[赞赏支持](#)

• 3.1: 模块的build.gradle配置

```
1  apply plugin: 'com.android.library'
2  //应用到Maven
3  apply plugin: 'maven'
4
5  android {
6      compileSdkVersion 26
7      defaultConfig {
8          minSdkVersion 16
9          targetSdkVersion 26
10         versionCode 1
11         versionName "1.0"
12         testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
13     }
14 }
15
16 //打包main目录下代码和资源的 task
17 task androidSourcesJar(type: Jar) {
18     classifier = 'sources'
19     from android.sourceSets.main.java.srcDirs
20 }
21 //配置需要上传到maven仓库的文件
22 artifacts {
23     archives androidSourcesJar
24 }
25 //上传到Maven仓库的task
26 uploadArchives {
27     repositories {
28         mavenDeployer {
29             //指定maven的仓库url, IP+端口+目录
30             repository(url: "http://localhost:8081/nexus/content/repositories/releases.
31                 //填写你的Nexus的账号密码
32                 authentication(userName: "admin", password: "123456")
33             }
34             // 唯一标识 一般为模块包名 也可其他
35             pom.groupId = "com.android.imusic.lib"
36             // 项目名称 (一般为模块名称 也可其他
37             pom.artifactId = "music-player-lib"
38             // 发布的版本号
39             pom.version = "1.0.0"
40         }
41     }
42 }
```

• 3.2: 执行uploadArchives, 将项目发布到Nexus

执行uploadArchives或者使用Studio自带的Terminal中执行 gradlew uploadArchives 命令开始构建运行。等待结束后, 浏览器打开[Nexus](#), 查看Repositories->Releases 目录。成功发布后的代码库模块目录如图所示:

推荐阅读

再见命令行! K8S傻瓜式安装, 图形化管理真香!

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

4：使用局域网Nexus存储库代码库

• 4.1：根build.gradle配置修改

在你的项目中的根build.gradle中添加如下代码

```
1 allprojects {
2     repositories {
3         google()
4         jcenter()
5         //本地Maven仓库地址,取决于你的磁盘目录
6         maven {
7             //这里本地不再使用,改用下面的局域网的路径
8             //url 'file://D://AndroidStudioProjects//IMusic//maven',
9             //添加本地仓库URL
10            url 'http://localhost:8081/nexus/content/repositories/releases/'
11        }
12    }
13 }
```

• 4.2：app模块中build.gradle配置修改

在你的项目中要使用代码库的模块中的build.gradle做如下修改：

```
1 dependencies {
2     implementation fileTree(include: ['*.jar'], dir: 'libs')
3     //和刚才使用本地Maven仓库一致: 包名+库名
4     implementation 'com.android.imusic.lib:music-player-lib:1.0.0'
```

到此，发布代码到本地仓库到此结束了。但是随之又来了一个问题，局域网仓库对公司团队来说固然是好，但是你开发的模块很牛逼的时候，别人要用你的模块时怎么办？这个时候就需要将模块发布至外网仓库，即发布代码至JCenter。

三：发布到外网Maven

发布到外网Maven，本问介绍发布到JCenter上。在早期的AndroidStudio版本中，Google默认使用的仓库是mavenCentral，但由于发布流程太复杂(前辈们是这样分析的，不知道对不对)，后来改用默认仓库是JCenter,说了这么久还没介绍JCenter是什么？简单来说JCenter是Bintray其下的一个分区仓库，代码托管的存储空间。那既然JCenter是属于Bintray旗下的，那就首先得从Bintray开始。

1：Bintray账号准备

Bintray是一家OSS服务商，类似国内的阿里云OSS。要将代码发布之JCenter，必须先有Bintray账号和分区目录。感兴趣可以去看下AndroidStudio的Bintray库在Github中的项目：[Bintray-Github](#)。注册账号我们注册个人账号即可。[前往Bintray官网个人账号注册](#)。

- 注意：1.注册账号最好注册个人的，不然企业账号没有add To Jcenter功能，需要收费。2：注册的邮箱一定要使用国外的邮箱，不然注册没反应。

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

Bintray账号注册图示

2：为Bintray账号添加组织

添加组织有两种方式：1：创建Bintray组织。2：导入第三方托管平台组织。

2.1：创建Bintray组织

在bintray主页鼠标放至头像弹窗的Menu框中点击createOrganization->界面跳转后点击->Create new organization创建新的组织。

创建组织入口图示

按照提示填写组织基本信息提交即可。

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！
阅读 4,690

若依框架springBoot-搭建和部署
阅读 402

Android 检测是否通知栏权限(显示通知)
阅读 211

【Small Tools】Spring Cloud
Alibaba + Vue 前后端分离项目
阅读 309

Android Bugly上报过程源码分析
阅读 304



写下你的评论...

评论1

赞4

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

Bintray支持第三方托管平台的组织导入，但目前为止只支持从Github导入组织到Bintray。

- 2.2.1：导入Github组织
在bintray主页鼠标放至头像弹窗的Menu框中点击createOrganization->再点击Import from GitHub导入Github账号组织。第一次导入官方会给出操作流程图，如下：

Bintray导入Github组织流程图示

- 2.2.1.1 前往Github设置中心
点击图中标记的Account Setting page，前往Github设置中心界面。跳转至Github的设置中心后依次点击左侧的Applications，展开界面后点击右侧的Authorized OAuth Apps，展开列表就可看到支持列表中有bintray了。

授权设置及操作图示

- 2.2.1.2 授权Bintray访问Github账户的权限
点击Bintray一栏中的Grant，授予Bintray访问Github账户的权限。

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304



写下你的评论...

评论1

赞4

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

Github授权Bintray访问权限图示

- 2.2.2：填写第三方组织在Bintray中的基本信息
授予访问权限后，再次走导入Github组织流程，选择你的组织后，根据界面提示填写基本信息提交即可。

3：创建Bintray存储分区

点击Add New Repository创建存储分区

创建存储分区入口图示

按图中所示，填写分区基本信息

推荐阅读

- 再见命令行！K8S傻瓜式安装，图形化管理真香！
阅读 4,690
- 若依框架springBoot-搭建和部署
阅读 402
- Android 检测是否通知栏权限(显示通知)
阅读 211
- 【Small Tools】Spring Cloud Alibaba + Vue 前后端分离项目
阅读 309
- Android Bugly上报过程源码分析
阅读 304



AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

[关注](#)[赞赏支持](#)

填写分区基本信息图示

创建成功后是这样的

存储分区建成功

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304

4：模块build.gradle配置

4.1：在你项目的根build.gradle中配置bintray环境依赖

```
1 buildscript {
2     repositories {
3         google()
4         jcenter()
5     }
6     dependencies {
7         classpath 'com.android.tools.build:gradle:3.0.1'
8         //添加bintray环境
9         classpath 'com.novoda:bintray-release:0.8.1'
10    }
11 }
```

4.2：在你要发布的模块build.gradle中配置如下配置

```
1 apply plugin: 'com.novoda.bintray-release'
2
3 //推送到Bintray配置
4 publish {
5     //你的账户bintray下某个组织id
6     userOrg = 'xxxxxx'
7     //Maven仓库下库的包名,一本与包名相同
8     groupId = 'xxx.xxx.xxx'
9     //项目名称
10    artifactId = 'xxx'
11    //版本号
12    publishVersion = '1.0.0'
13    //项目介绍,可以不写
14    desc = 'xxx'
15    //项目主页,可以不写
16    website = 'xxx'
17 }
```

最终的依赖地址是compile '仓库包名:项目名称:版本号'

5：编译并发布到Bintray

5.1：获取Bintray API Key



写下你的评论...

评论1

赞4

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！
阅读 4,690

若依框架springBoot-搭建和部署
阅读 402

Android 检测是否通知栏权限(显示通知)
阅读 211

【Small Tools】Spring Cloud Alibaba + Vue 前后端分离项目
阅读 309

Android Bugly上报过程源码分析
阅读 304

Bintray获取API KEY图示

5.2：运行发布命令

在AndroidStudio编译器自带的Terminal中输入命令：

```
1 | gradlew clean build bintrayUpload -PbintrayUser=BINTRAY_USERNAME -PbintrayKey=BINTRAY_KEY
2 | //BINTRAY_USERNAME替换为你的Bintray用户名，BINTRAY_KEY替换为你刚才获取的Bintray API KEY
3 | //PdryRun释义：false：编译且将你的项目上传至Bintray，true：只是构建你的项目，不会上传
```

如果你在编译中遇到问题，请前往[AndroidStudio发布项目到Bintray遇到的错误及解决](#)开始构建后耐心等待，可能会因为socket出现多次上传失败，重试即可。直到出现这个表示上传成功。

发布到Bintray成功

上传完成后可在你的主页或者组织下查看



AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

6：添加至JCenter

点击右下角add to JCenter，确认后点击send提交，等待审核通过。(白天提交会到晚上才审核通过，因为有时差，在他们上班时间提交，会立即审核通过)

添加项目至JCenter图示

send后，等待审核通过，通过后即可在外网使用你的库作为依赖了。

发送代码到JCenter信息填写图示

这个是Bintray回复的站内通知

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304



写下你的评论...

评论1

赞4

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

7: 项目引用JCenter库

7.1未添加至JCenter引用:

- 7.1.1: 在根build.gradle中配置如下代码

```
1 | allprojects {
2 |     repositories {
3 |         //bintray环境
4 |         maven { url 'https://dl.bintray.com/novoda-oss/snapshots/' }
5 |         //你的maven路径
6 |         maven { url 'https://dl.bintray.com/你的id/maven' }
7 |     }
8 | }
```

- 7.1.2: 在app模块中的build.gradle中配置如下代码

```
1 | dependencies {
2 |     //添加你的项目依赖地址,如下规则
3 |     implementation '仓库包名:项目名称:版本号'
4 |     //例如我的:
5 |     //implementation 'com.imusic.player:music-player:1.0.0'
6 | }
```

7.2: 已添加至JCenter引用:

- 7.2.1: 在根build.gradle中配置如下代码

```
1 | allprojects {
2 |     repositories {
3 |         //bintray环境
4 |         maven { url 'https://dl.bintray.com/novoda-oss/snapshots/' }
5 |         //添加支持JCenter即可
6 |         jcenter()
7 |     }
8 | }
```

- 7.2.2: 在app模块中的build.gradle中配置如下代码

```
1 | dependencies {
2 |     //添加你的项目依赖地址,如下规则
3 |     implementation '仓库包名:项目名称:版本号'
4 |     //例如我的:
5 |     //implementation 'com.imusic.player:music-player:1.0.0'
6 | }
```

至此,全部Maven及JCenter发布流程完毕,版本更新时只需要更改配置版本号重新发布即可。祝君要的愉快~~



4人点赞 >

日记本

...

更多精彩内容下载简书APP



写下你的评论...

评论1

赞4

推荐阅读

再见命令行! K8S傻瓜式安装, 图形化管理真香!

阅读 4,690

若依框架springBoot-搭建和部署

阅读 402

Android 检测是否通知栏权限(显示通知)

阅读 211

【Small Tools】Spring Cloud

Alibaba + Vue 前后端分离项目

阅读 309

Android Bugly上报过程源码分析

阅读 304

AndroidStudio发布项目到Maven(JCenter)



撸代码的皇太极

关注

赞赏支持

赞赏支持

还没有人赞赏，支持一下



撸代码的皇太极 Android、小程序、Node.js 程序猿
总资产1 (约0.11元) 共写了5572字 获得17个赞 共13个粉丝

关注

推荐阅读

再见命令行！K8S傻瓜式安装，图形化管理真香！
阅读 4,690

若依框架springBoot-搭建和部署
阅读 402

Android 检测是否通知栏权限(显示通知)
阅读 211

【Small Tools】Spring Cloud Alibaba + Vue 前后端分离项目
阅读 309

Android Bugly上报过程源码分析
阅读 304

- 包拯可以不割吗
- 个人创业项目
- web网站
- 养殖小项目
- 发布项目
- python试题
- web培训学习
- 三国志9下载
- 6000度近视
- web培训机构
- linux编程
- 电子评标系统
- web组态
- 创业好项目排行
- 实训人工智能
- web开发培训
- 5万创业好项目
- 虫草怎么喝

被以下专题收入，发现更多相似内容



技术干货



Android...



Android开发



Android...

推荐阅读

更多精彩内容>

[Gradle系列]Gradle发布module（maven）到jcenter, 并构...

Tamic : <http://www.jianshu.com/users/3bbb1ddf4fd5/> 前言 And...



Tamic 阅读 4,063 评论 4 赞 28

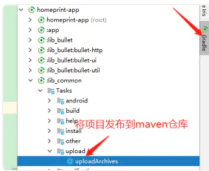


Android：超详细的本地搭建maven私服以及使用Nexus3.x...

前言 在Android开发中，会经常遇到需要将写好的某个lib用于各个不同的项目中。一般的做法，就是将lib直接导...



李益99 阅读 2,533 评论 4 赞 13

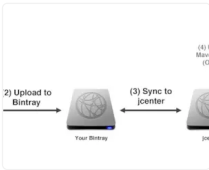


如何使用Android Studio把自己的Android library分享到jCen...

转载自 <http://www.jcodecraeer.com/a/anzhuokaifa/androidkaifa...>



猿ape 阅读 1,169 评论 5 赞 5



致自己

亲爱的我，不知不觉你已成年。在过去的很多日子里，你一直是那个胆小怕见光的女孩，像微风的轻飘，像细雨的短暂，在这万...



远方南巢 阅读 25 评论 0 赞 0

《自律更自由》

偶遇到小班长奈奈发的圈，大概看了一下通过一百天锻炼大家的自律性。大概了解每天只需要几分钟就可以达到自律的效果...



童心相伴 阅读 117 评论 0 赞 0



写下你的评论...

评论1

赞4