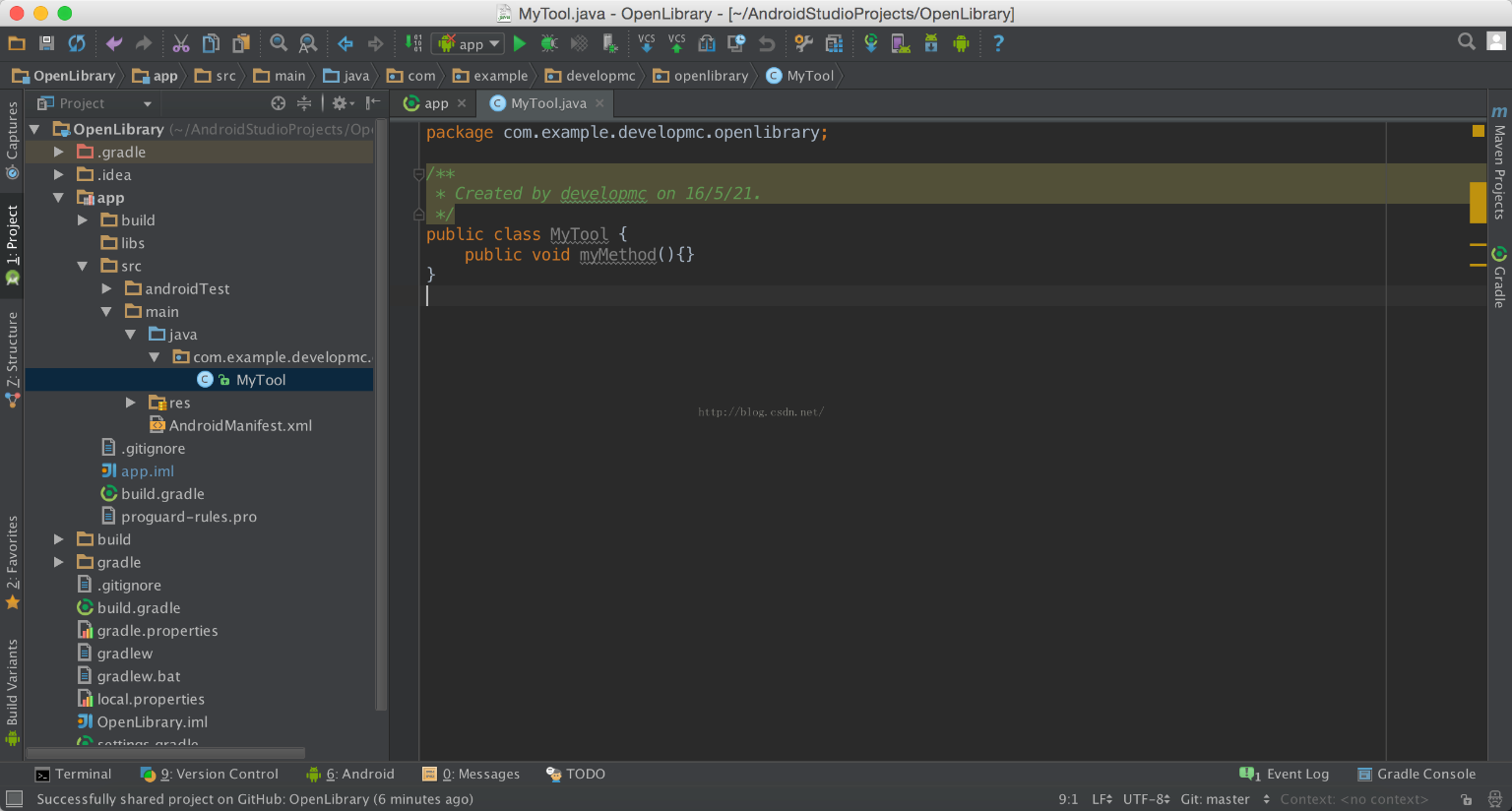
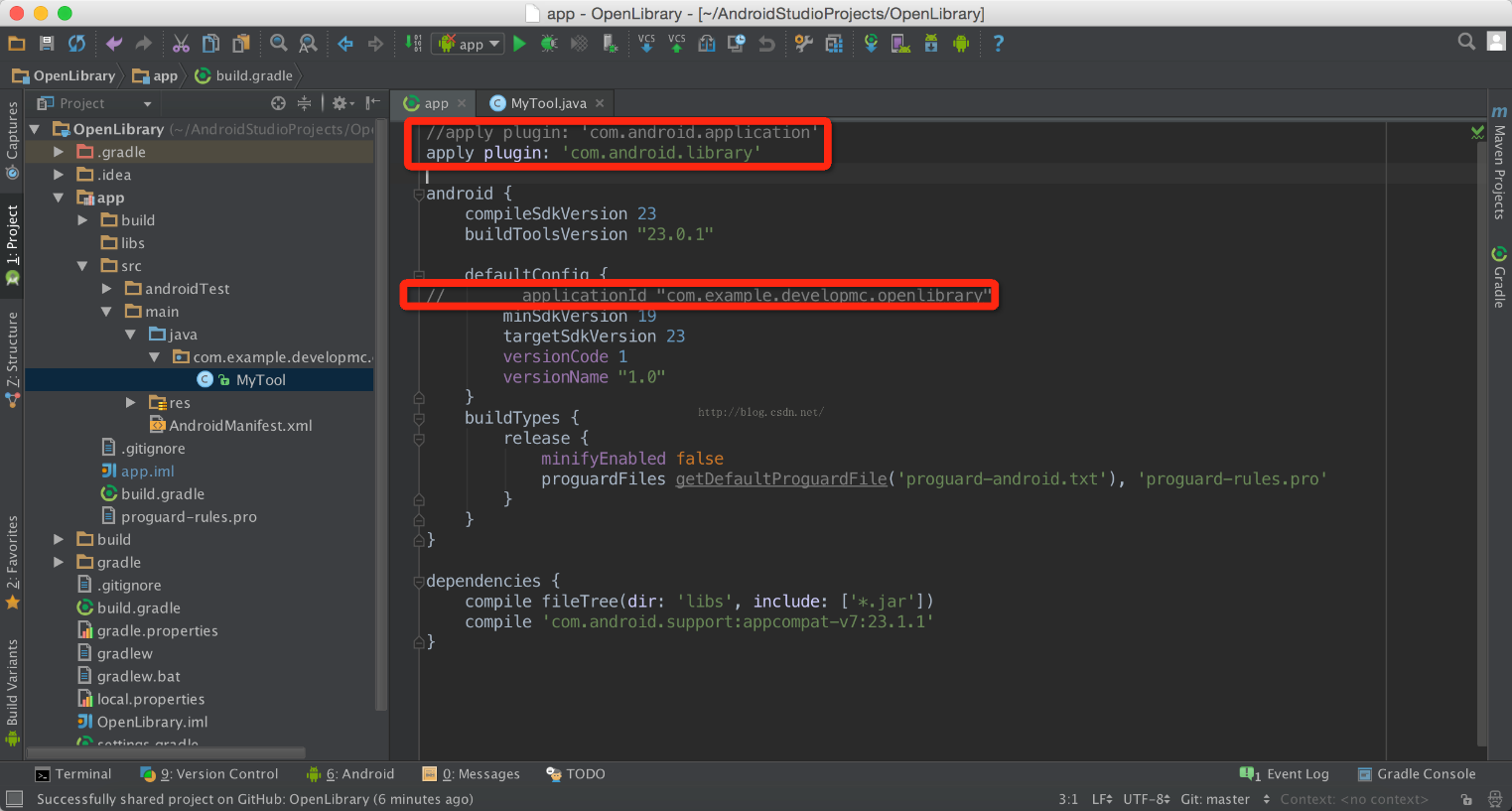
Jitpack实际上就是一个自定义的maven仓库，发布过程十分简单暴力：编写项目（库）－>上传到Github－>发布－>在别的项目中引用。

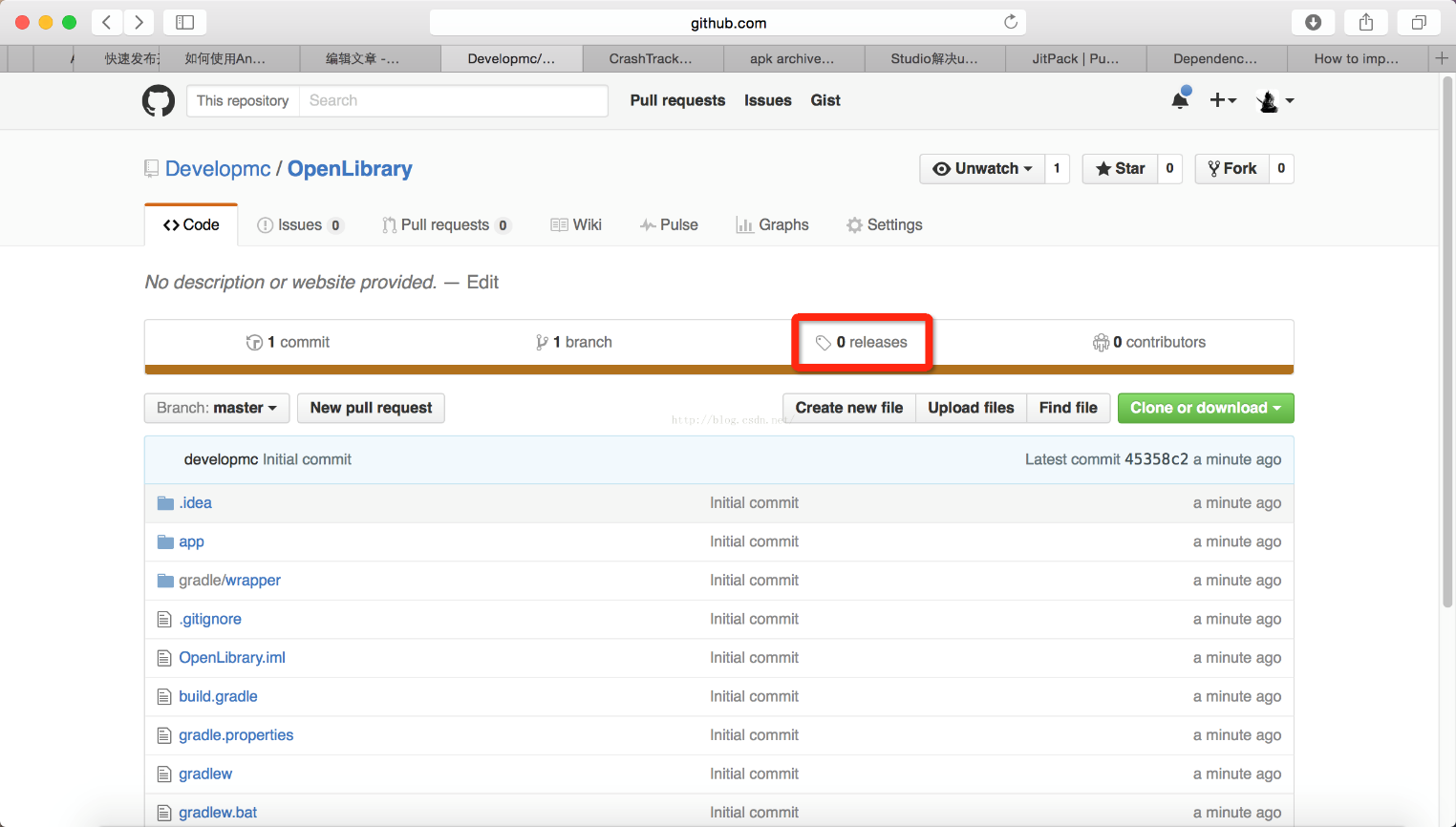
第一步：创建工程（库），如下图，我创建了名为“OpenLibrary”的工程，里边有个“MyTool”工具类（随便创建**[测试](http://lib.csdn.net/base/softwaretest" \o "软件测试知识库" \t "_blank)**的）；

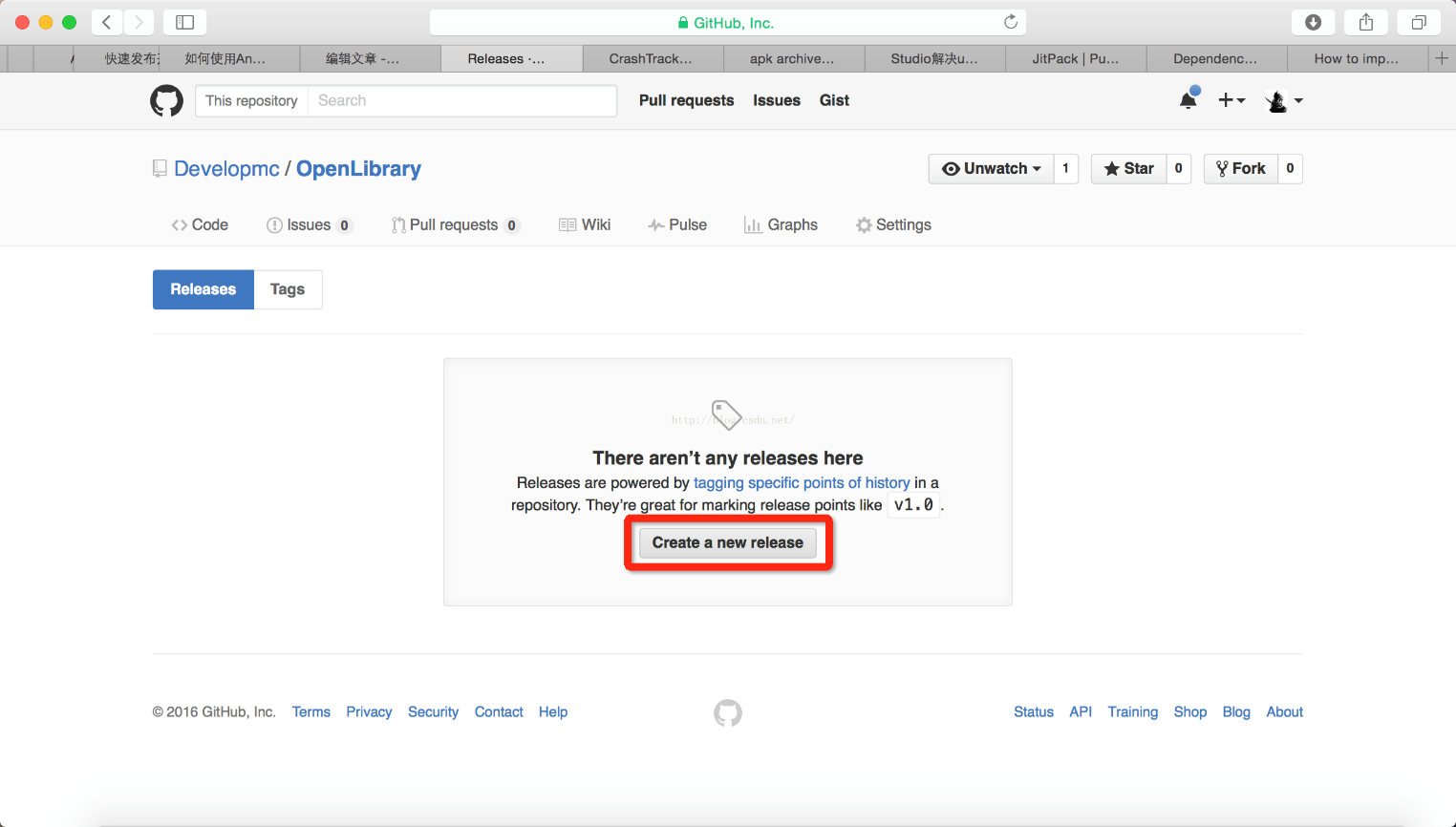


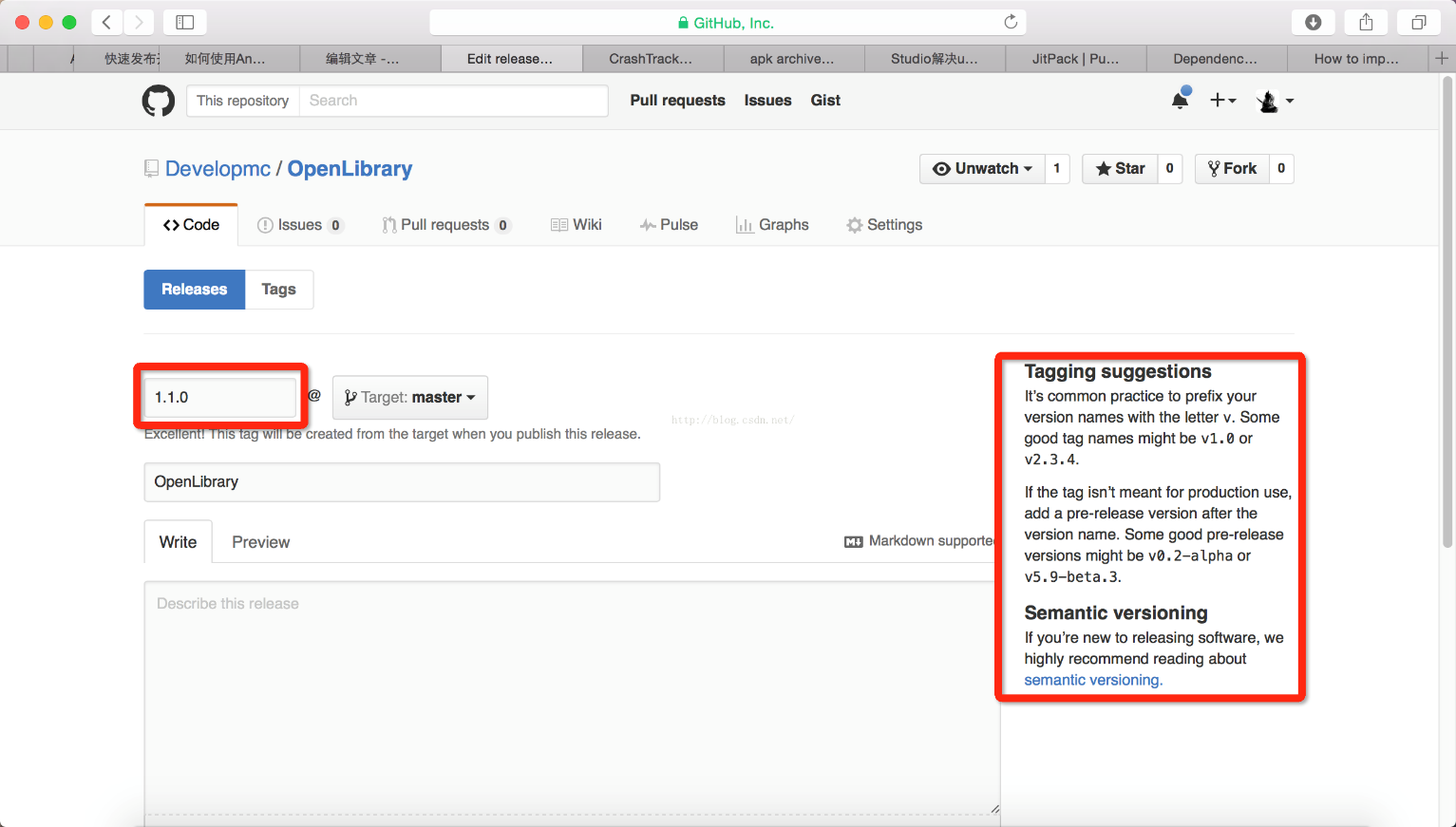
因为是library库，需要处理一下：如下，将module对应的build.gradle稍微修改。1.将apply plugin: 'com.android.application'改为apply plugin: 'com.android.library';2.注释applicationId.

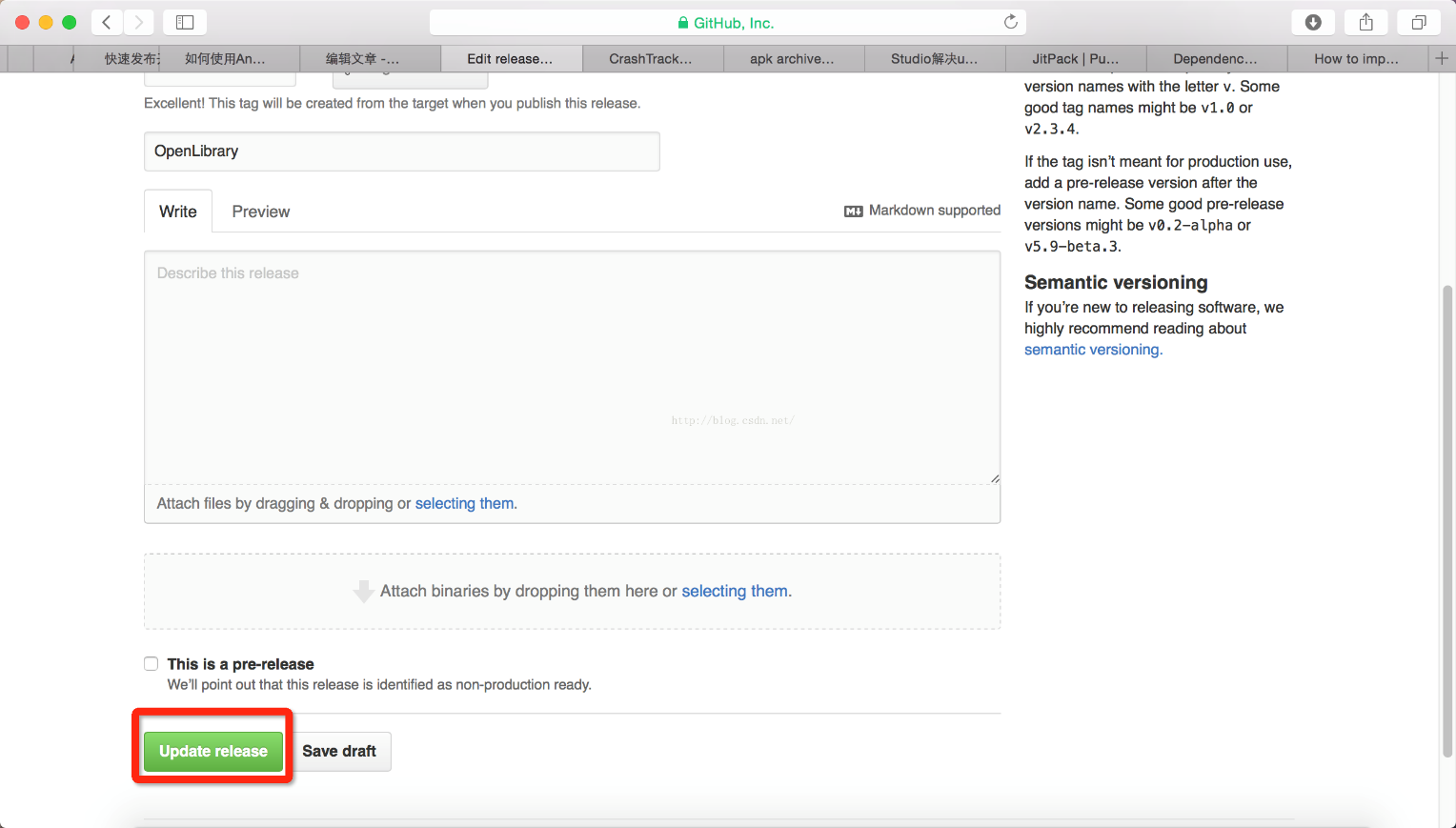
第二步：上传项目给Github。

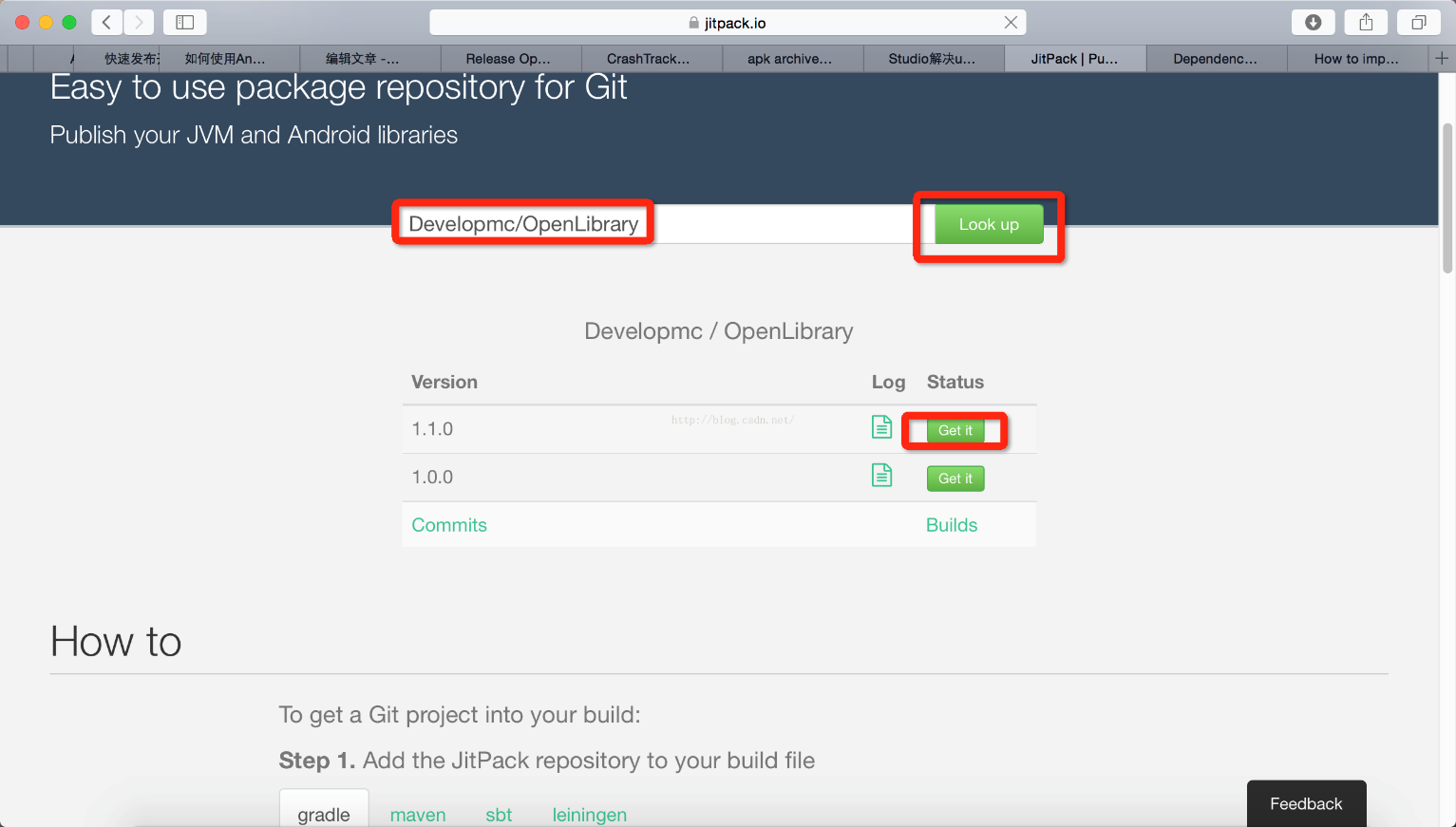
第三步：发布该库，在Github上打开该库，点击“release”,

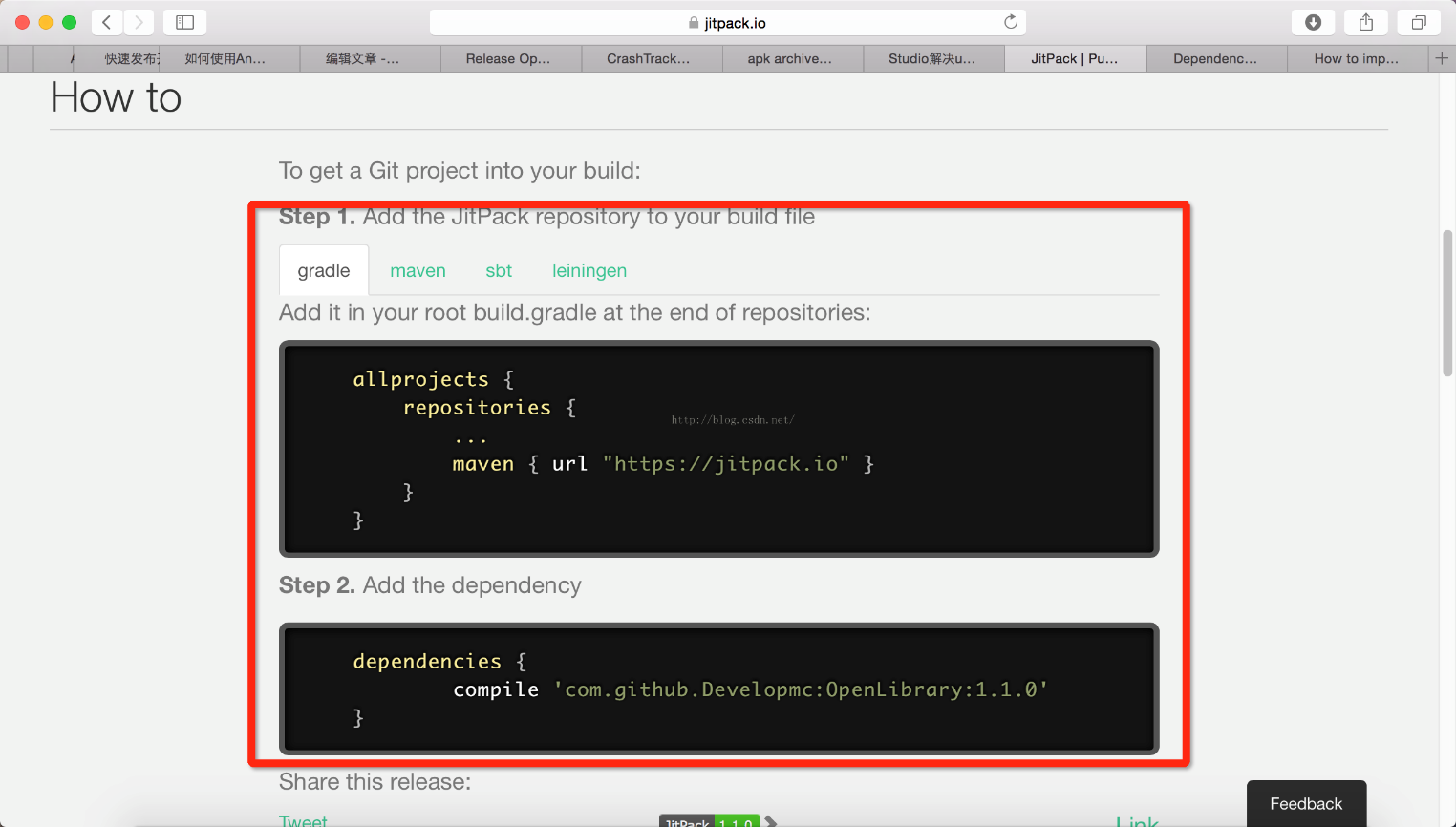


接着点击“create a new release”,

填写相关信息（右边红色框是对Tag的说明），最后点击“uodate release”.z



到这里，OpenLibrary库就发布完成了，是不是很简捷！发布之后，由于在Github上无法直接看到该库的引用信息，那么我们如何该如何获知？打开[https://jitpack.io](https://jitpack.io/" \t "_blank)，在搜索输入OpenLibrary库的地址，点击“Look Up”之后，下方出现版本信息，点击对应版本的“Get it”，即可得到该OpenLibrary库的引用信息。



这里对OpenLibrary库的引用:

compile 'com.github.Developmc:OpenLibrary:1.1.0'

其中：com.github.\*\*\*\*\*\*是你的Github地址，OpenLibrary是库名称，1.1.0是版本号，无需使用[https://jitpack.io](https://jitpack.io/)，我们也能把它拼凑出来。到这里你的库就可以被其他开发者引用了！