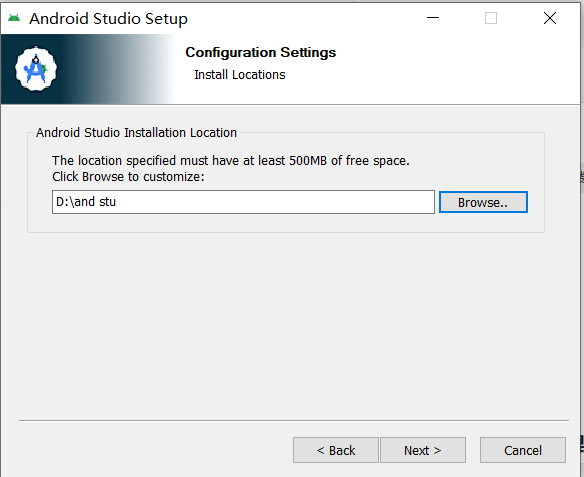
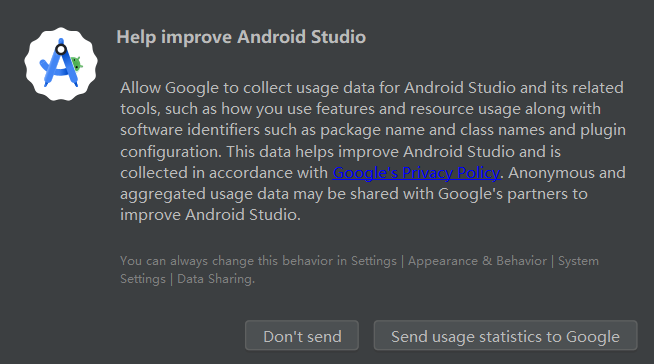
Android studio

在早期的Android Studio版本中并没有自带JDK，因此需要配置Java JDK的环境变量，才能够正常编译Android项目，而在新版本中， Android Studio中自带了一个JDK版本

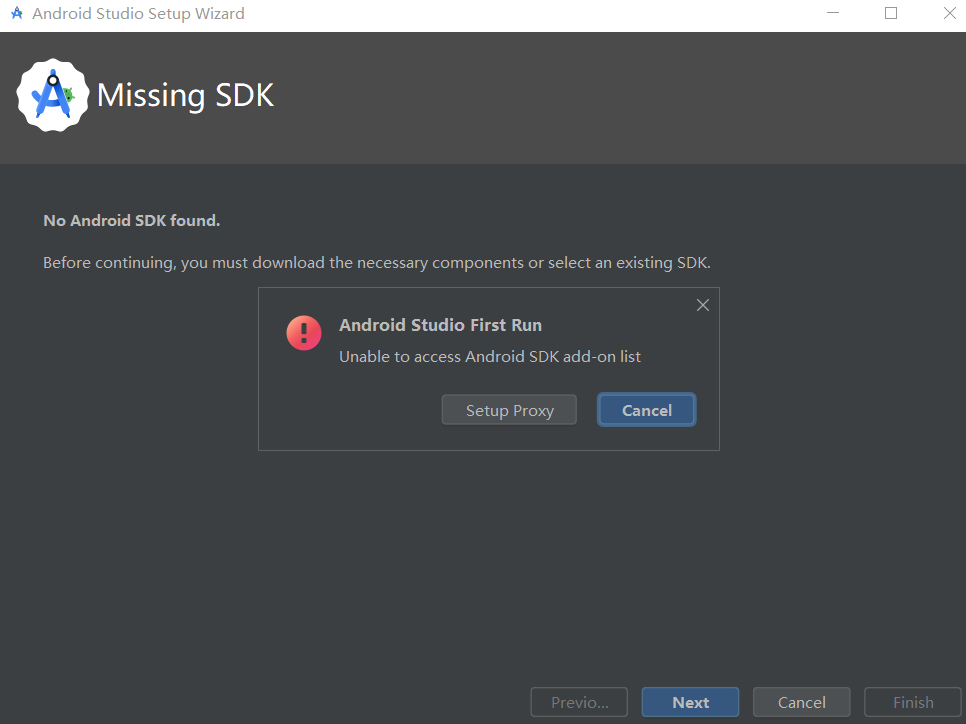
1. 官网下载Android studio
2. 改路径



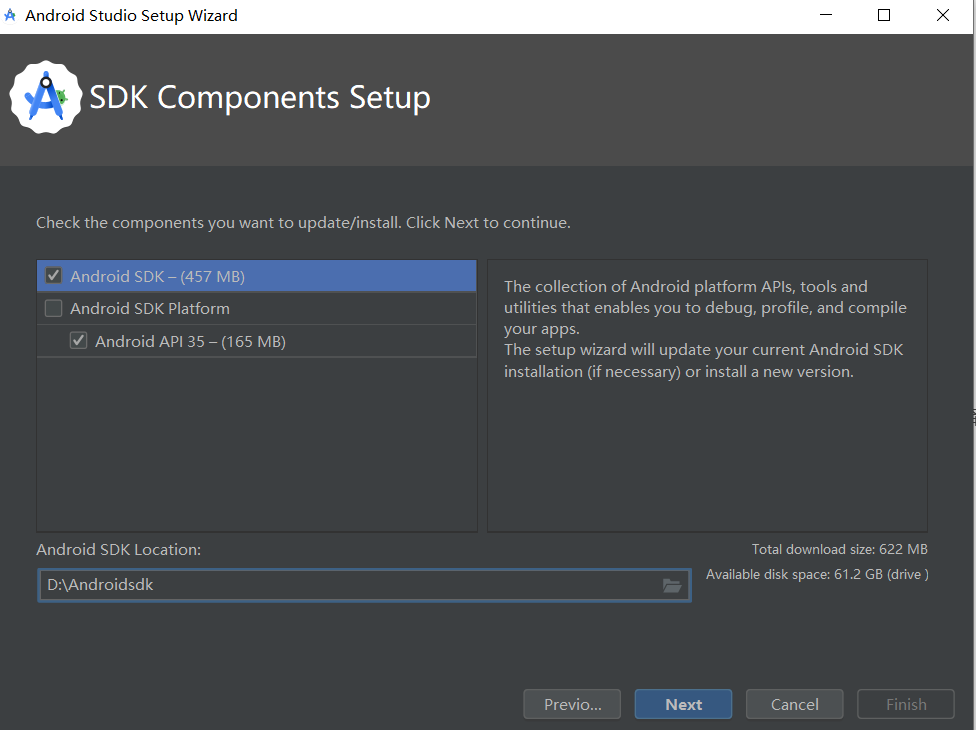
1. 其余都是next
2. Don’t send，不帮助goole收集信息



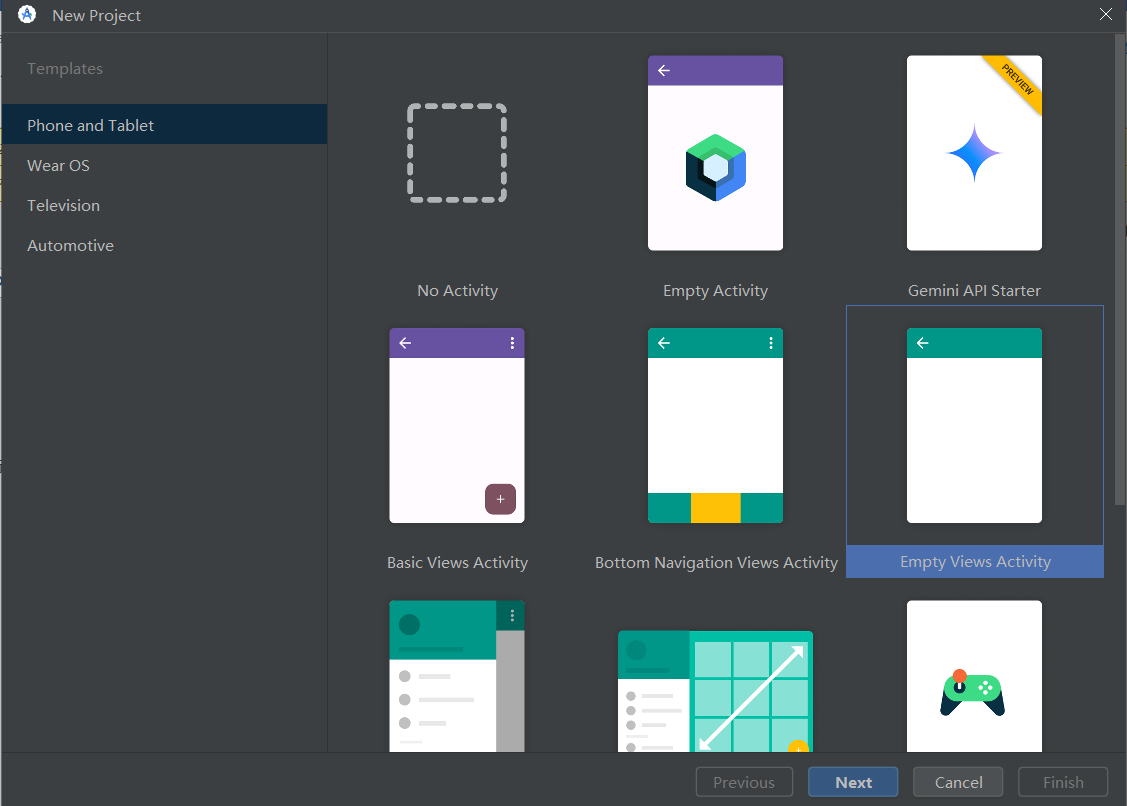
1. 说没有SDK，让你设置代码，这里不设置，点击 **Cancel**



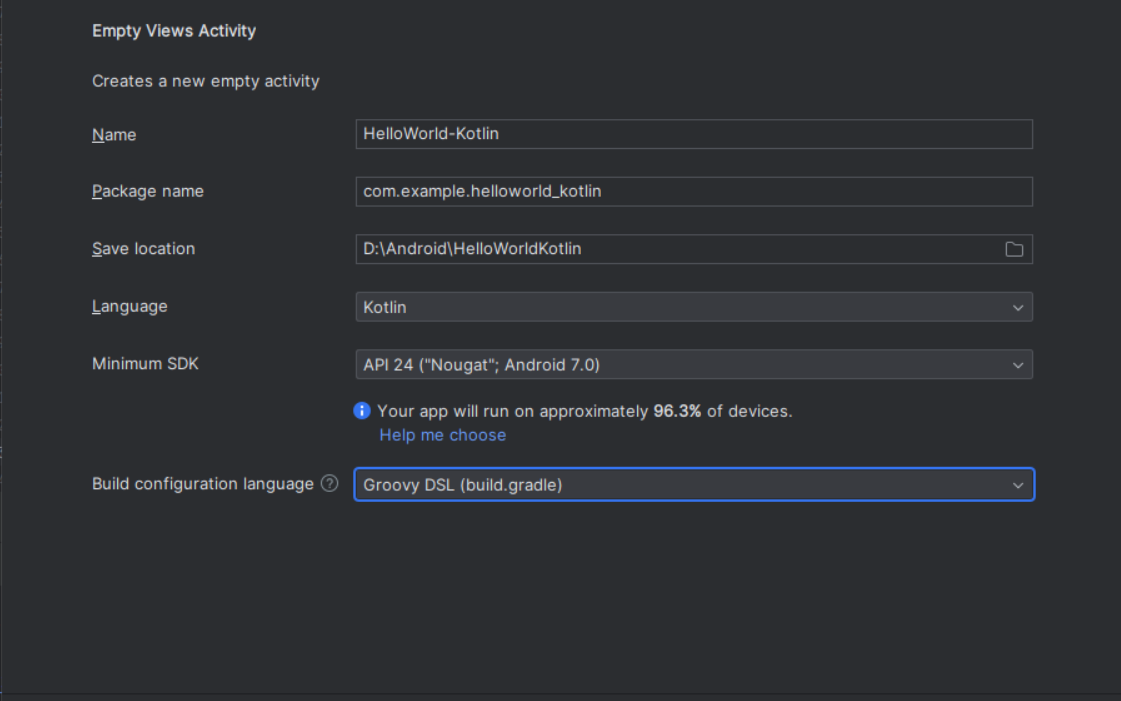
1. 下载sdk改路径



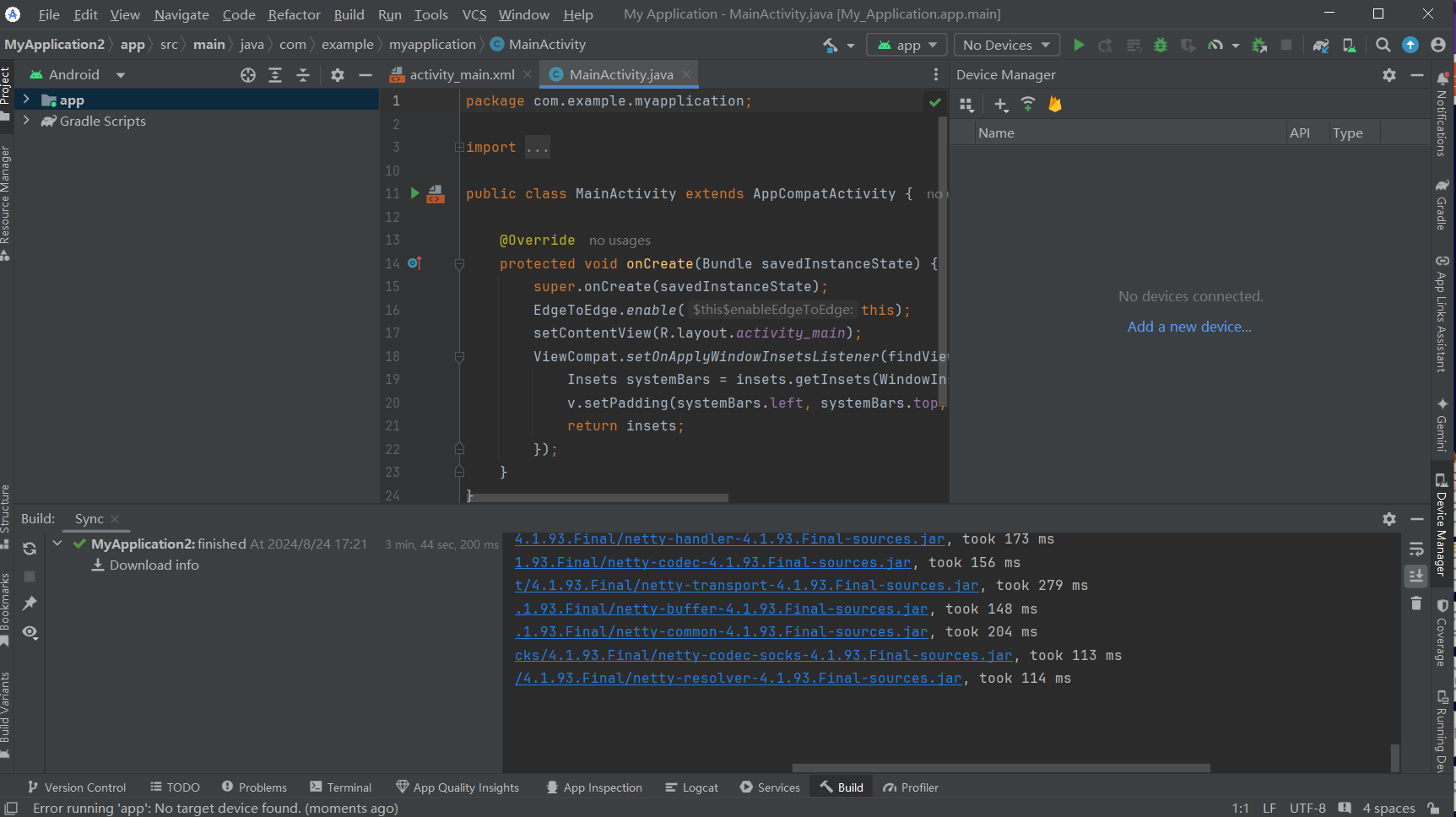
1. 其余next
2. 在新版本的Android Studio中Empty Activity模板默认的就是Compose模板，如果你希望创建Java或者Kotlin语言模板的项目，那么就选择Empty Views Activity



1. 创建



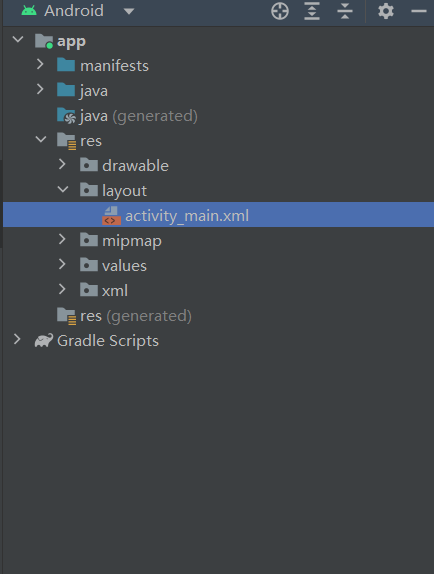
1. 创建完后会下载一堆插件，使用vpn，不然有几率失败
2. 创建虚拟机

****

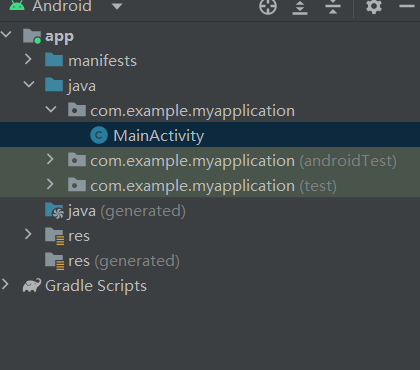
**右上角手机+安卓图标**

**或者连接真机，打开开发者模式与usb调试**

1. **在下图里改变布局**

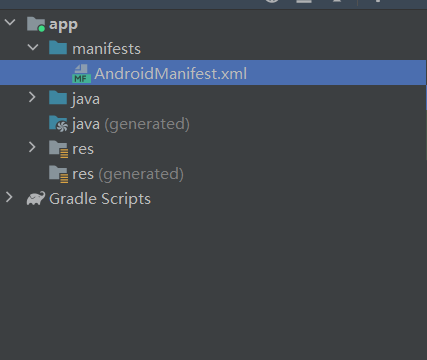
****

1. **在下图里写交互逻辑等**

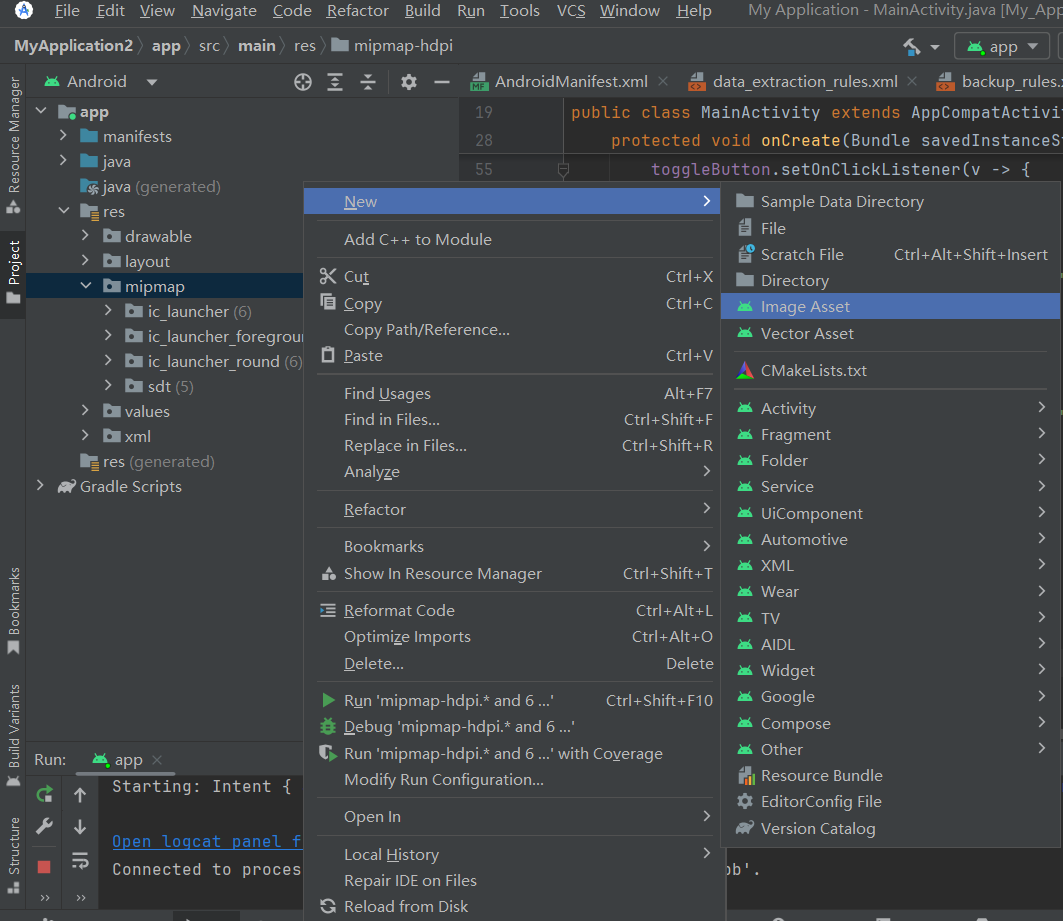
****

1. **在下图里提供权限，修改app名称图标等**

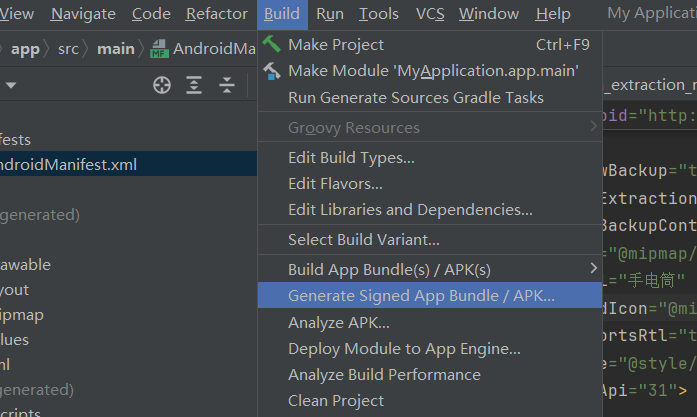
**Label改名，icon改图标**

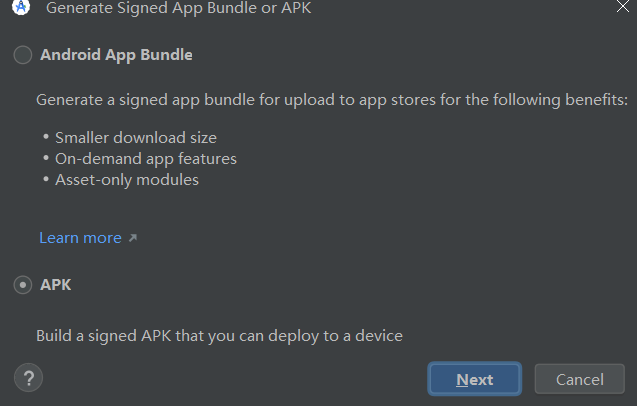
****

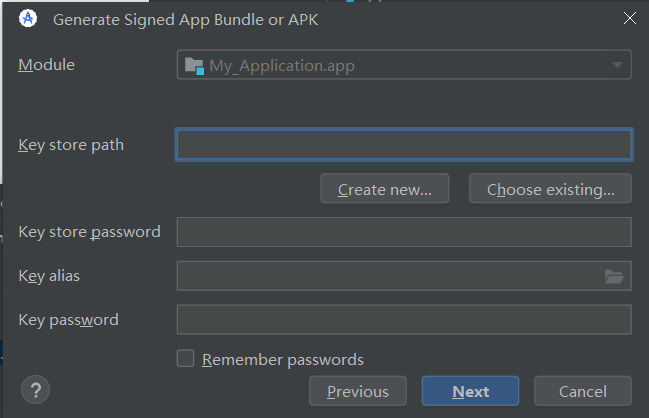
1. **在下图里新增app图标**

****

1. **签名打包apk，目的是便于更新，安全，防止篡改。**

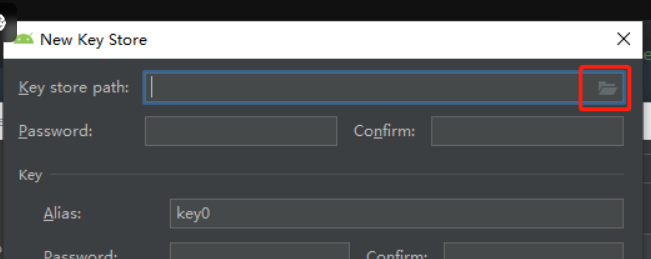
****

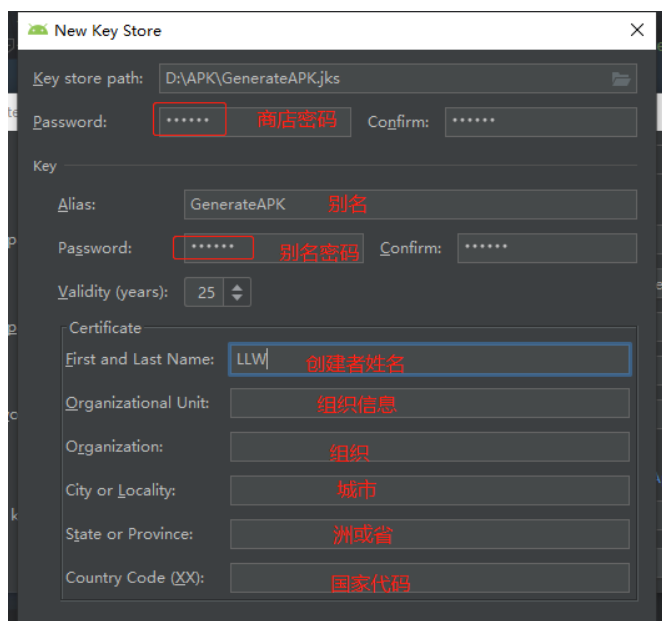
****

****

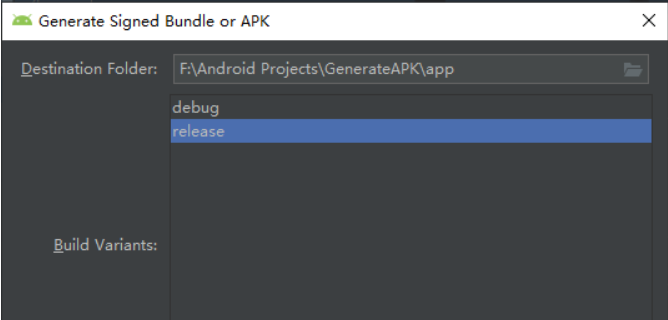
**这里需要一个商店的key，因为没有，所以创建一个新的，点击Create new**

1. **指明生成的jks文件的路径**

****

****

1. **选择release**

****

1. **可在app-release里找到app**

**参考：[Android Studio 打包APK（详细版）\_as打包apk-CSDN博客](https://blog.csdn.net/qq_38436214/article/details/112288954)**

Github上传

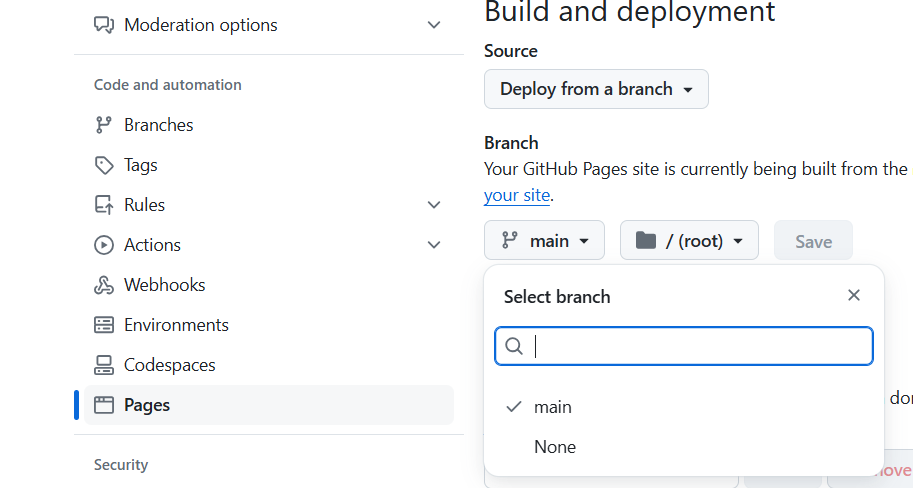
1. 在github上创建仓库

2）设置访问页面链接

项目的链接，可以直接访问，看到项目里的内容，静态网页可以浏览器看效果。

进入刚新建的仓库，点击Settings。

往下滑动，找到Pages。

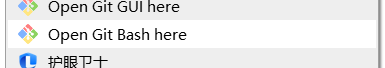
****

选择master，save。跳转到这个页面，此处链接直接访问到仓库里的项目。

3.本地库和GitHub建立链接，上传

在自己电脑上新建一个文件夹用来放你想要上传至GitHub的文件。

进入文件夹，右击选择Git Bash Here。

****

进入git，输入git init ，文件夹下生成.git文件夹。

git init

把想要上传至GitHub的文件放入文件夹下。

输入 git add .(注：add空格后有个点）

git add .

出现warning: in the working copy of '.gitignore', LF will be replaced by CRLF the ne

xt time Git touches it

Git 可以在你提交时自动地把回车（CR）和换行（LF）转换成换行（LF），而在检出代码时把换行（LF）转换成回车（CR）和换行（LF）。 你可以用 git config --global core.autocrlf true 来打开此项功能。 如果是在 Windows 系统上，把它设置成 true，这样在检出代码时，换行会被转换成回车和换行：

#提交时转换为LF，检出时转换为CRLF

$ git config --global core.autocrlf true

**可以输入git status，看看当前状态**

**git status**

**开始提交，git commit -m “提交时的注释”。**

**git commit -m "first commit"**

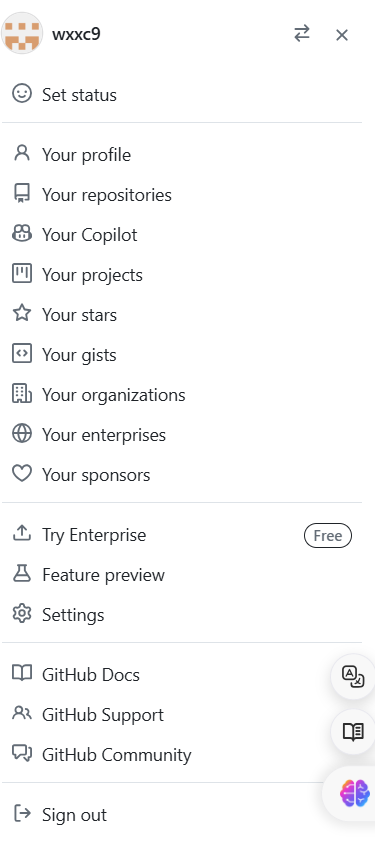
**开始链接仓库，传输有密钥，要进行设置。**

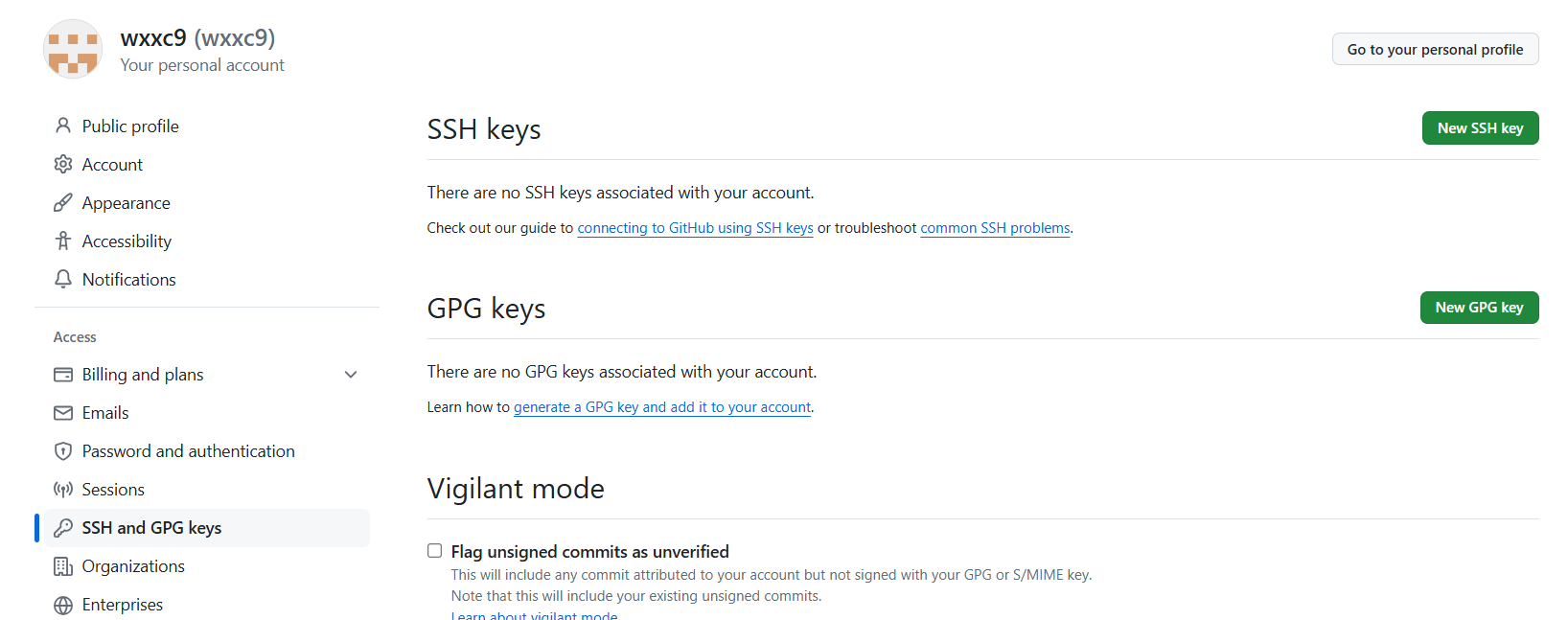
**输入代码，一直回车，在用户下找到.ssh文件夹。**

**ssh-keygen -t rsa -C "你的注册邮箱"**

**找到 id\_rsa.pub。**

**进入你的GitHub账户，找到Settings（头像旁边三角，往下滑）。**

****

****

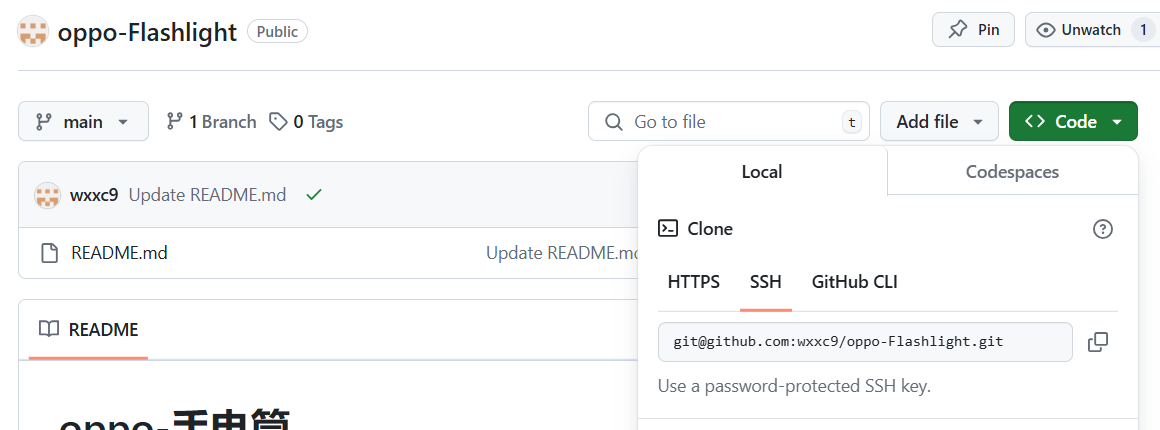
**点new ssh key**

**把刚才id\_rsa.pub文件下的内容复制到key里。**

****

**GitHub仓库和本地库进行链接。**

**复制ssh**

****

**输入**

**git remote add origin** [**https://github.com/wxxc9/oppo-Flashlight.git**](https://github.com/wxxc9/oppo-Flashlight.git)

**第一次会出现The authenticity of host 'github.com (20.205.243.166)' can't be established.**

**ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.**

**This key is not known by any other names.**

**Are you sure you want to continue connecting (yes/no/[fingerprint])?**

**输入yes就行**

**拉取**

**git pull --rebase origin master**

**为空时上传**

**git push -u origin master**

**不为空时上传**

**git push origin master**

**注意：默认是master分支上传**

**修改默认分支**

**接下来我们把所有默认分支都改为main**

**3.1 初始化时修改**

**在初始化时指定默认分支为main**

**git init -b main**

**3.2 初始化之后修改**

**如果已经初始化，依然可以修改本地分支**

**如原来本地初始化时默认分支为master**

**现在我想让他改为和远程仓库一致变成main**

**git branch -m master main**

**3.3 全局修改**

**全局方式修改默认分支**

**git config --global init.defaultBranch main**

**3.4 Github设置修改**

**还可以在Github中设置创建仓库时默认的分支名称**

**Settins—Repositories**

**将分支修改后Update即可**