

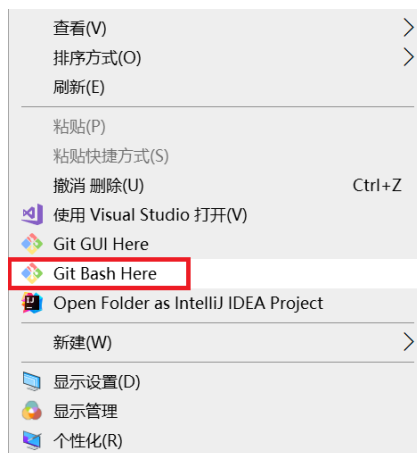
一、Gitee

1. <https://gitee.com> 上创建账号，必须绑定手机号，否则无法正常使用。

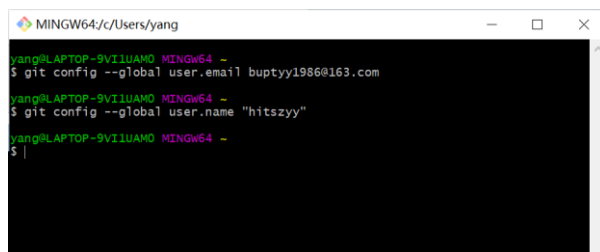


2. 登录官网下载 git: <https://git-scm.com/downloads>

下载安装后在桌面右键，能看到“Git Bash Here”则表示安装成功。

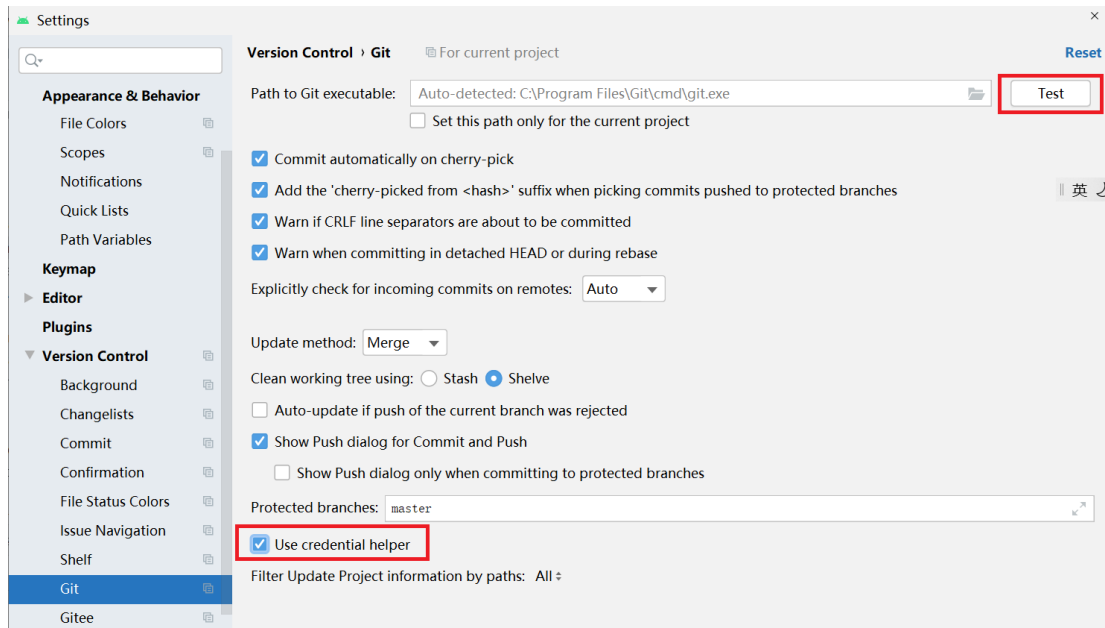


3. 配置本地 Git, 打开 Git Bash, 配置用户名和邮箱用于用户标识。

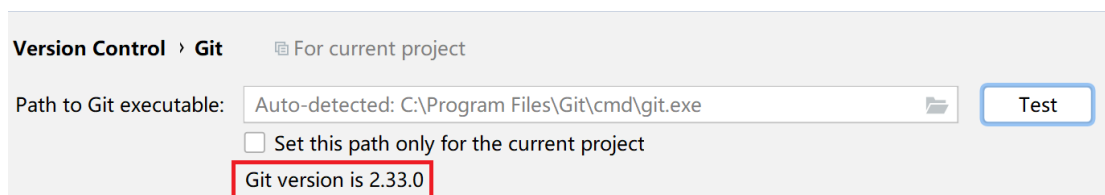


4. 在 AS 上设置 Git

File → Settings → Version Control → Git



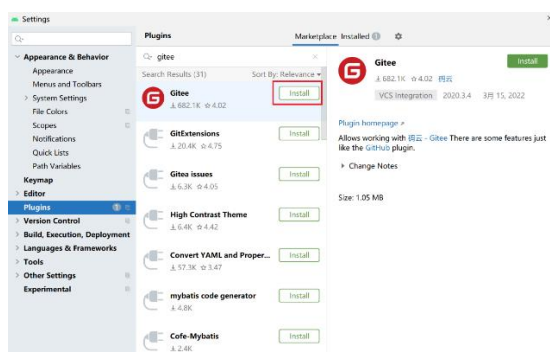
点击 Test 测试



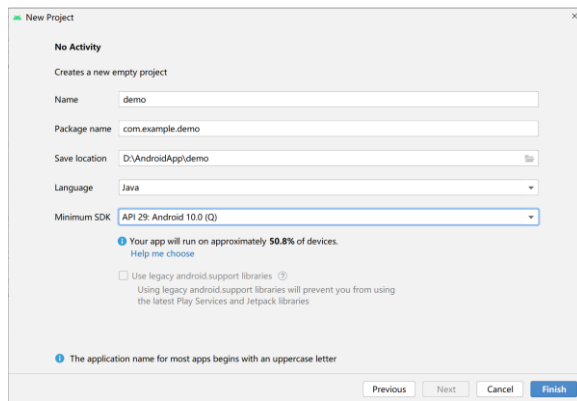
出现 Git 版本信息表示配置成功,同时勾选“Use credential helper”。

5. 在 Android Studio 中安装 gitee 插件

Android Studio→File→Settings→Plugins, 搜索 Gitee

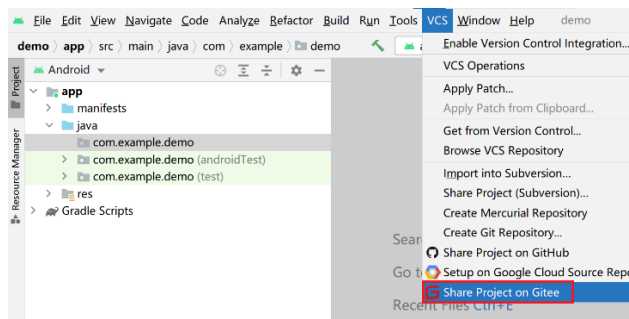


6. Android Studio 中新建一个项目 demo

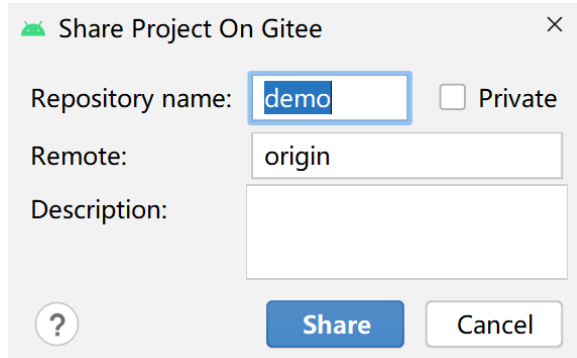
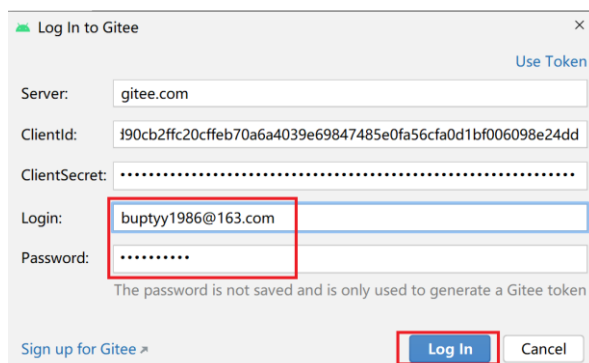


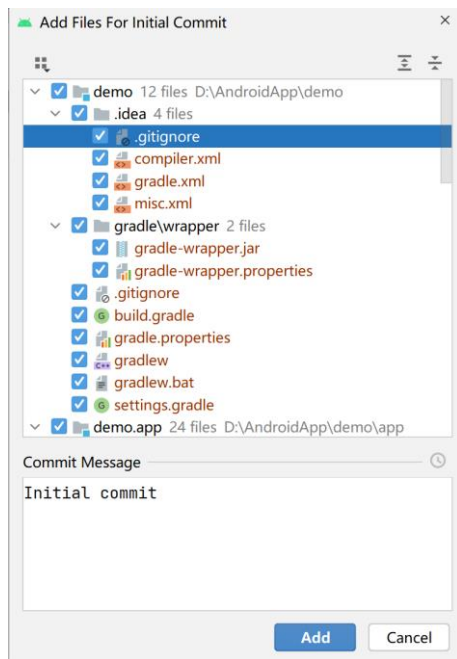
7. 将项目上传到 gitee 中

AS→VCS→Share Project on Gitee

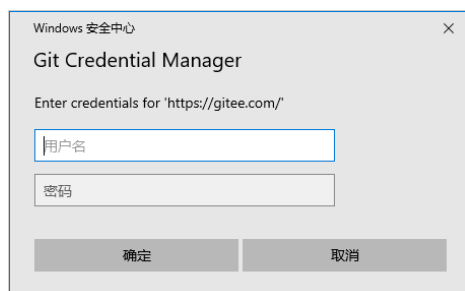


8. 输入 gitee 账户和密码

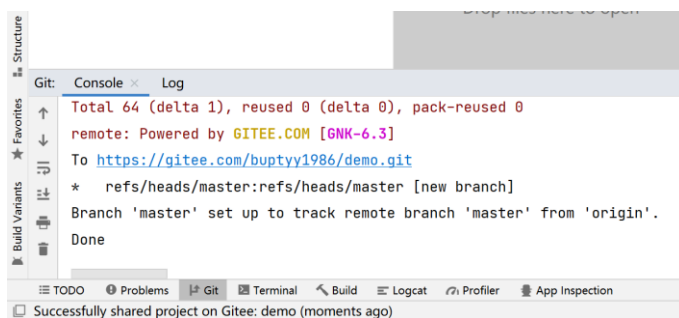




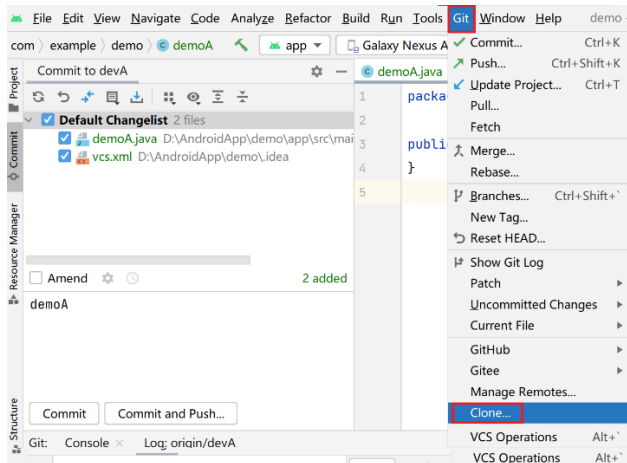
9. 输入 gitee 用户名和密码



10. 打开 git 面板，提示已经成功上传到 master 分支，远程仓库也能看到最新代码了。

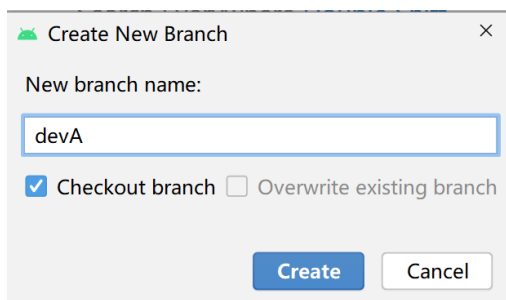
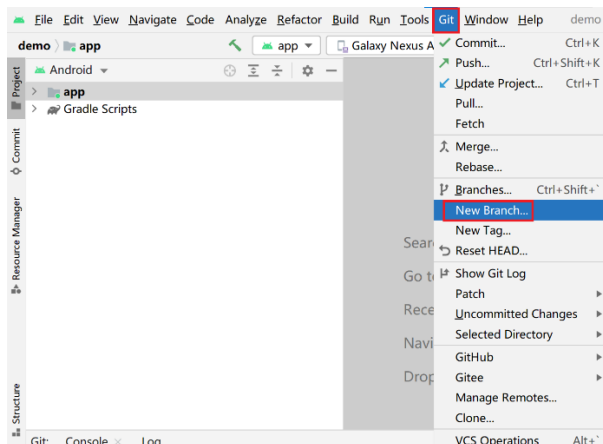


11. 如果是第一次下载远程仓库中的项目到本地，使用 git clone

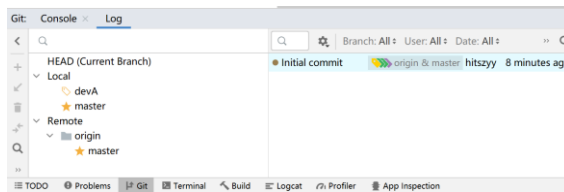


12. 假设有多个开发者同时进行开发，他们在开发前要创建自己的分支，在各自的分支上进行开发，开发完成后合并到 master 分支。

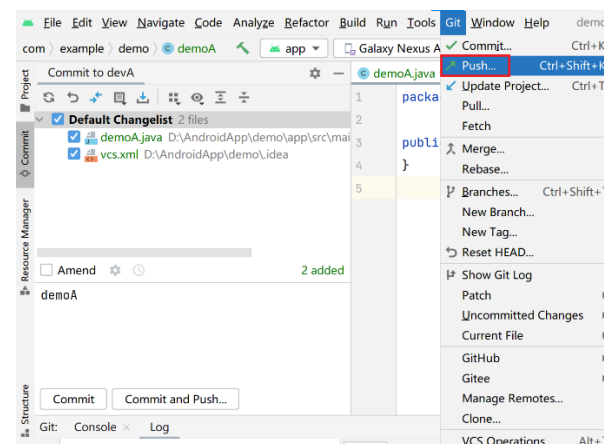
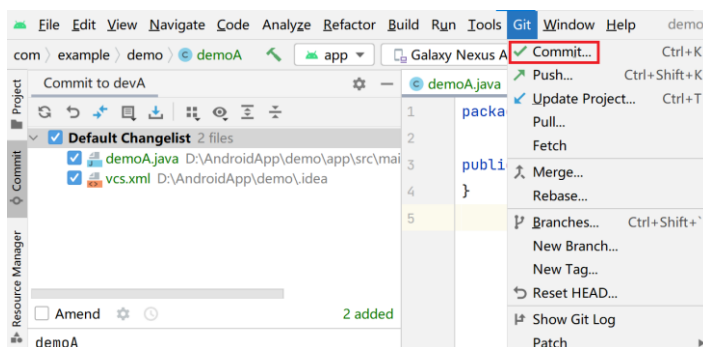
a) 创建分支 dev



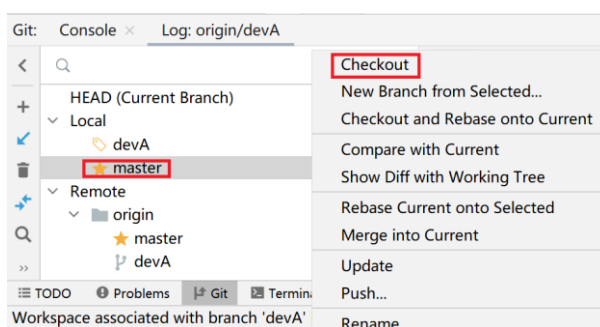
创建成功后打开 git 的 log，能看到本地有两个分支 devA 和 master，而远程目前只有一个分支 master



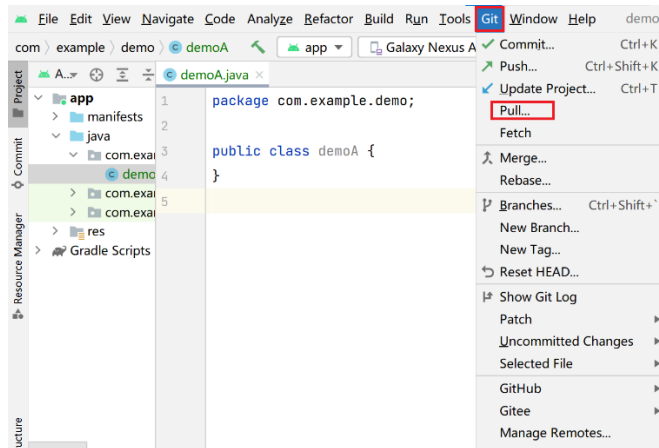
b) 开发者在每次完成自己的开发后，将改动 commit and push 到远程仓库的 dev 分支。此时其他开发者依然看不到这些改动，只有经过 c-f 步骤后将改动提交到远程 master 分支上后才可以。



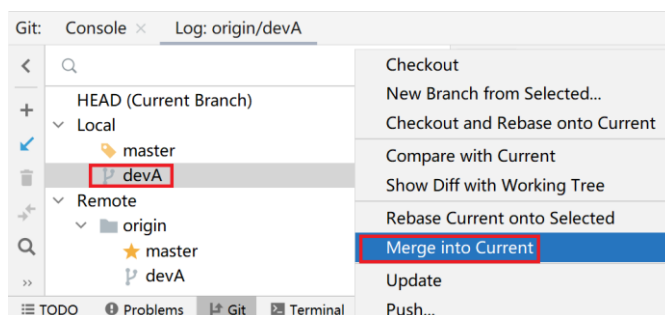
c) Checkout master 分支



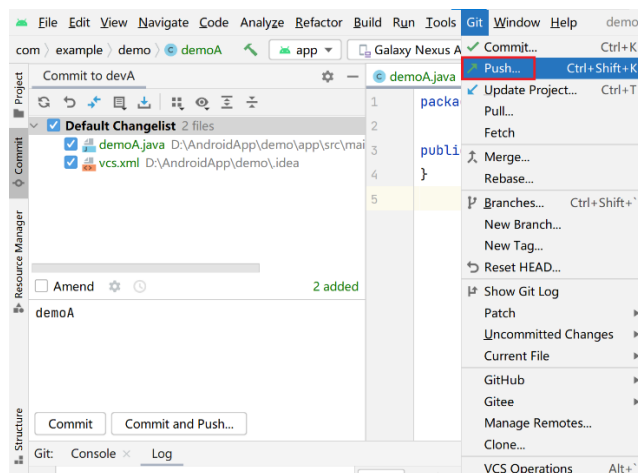
d) 从远程仓库的 master 分支 pull 最新代码



e) 将 dev 分支合并到本地 master，此时有可能出现冲突，要先根据提示处理冲突后才能继续合并。



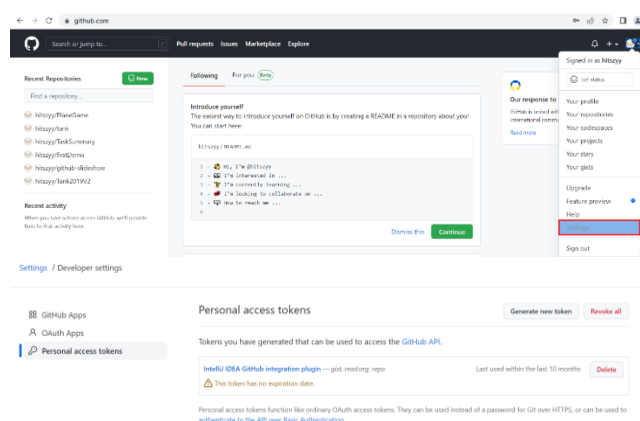
f) 将本地 master 分支 push 到远程仓库 master 分支，此时代码的改动才能被其他开发人员看到。



二、GitHub

GitHub 的使用和 Gitee 类似，这里只说明不同处

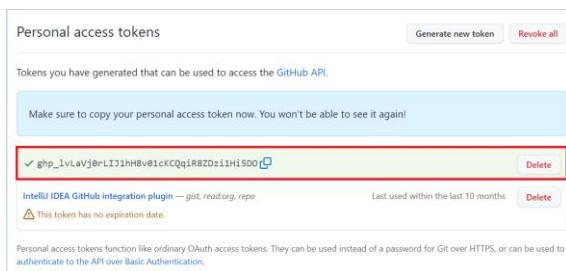
1. 在 GitHub 网站 <https://github.com/>上注册自己的账号
2. 打开 GitHub 网页，个人 settings→Developer settings→Personal access tokens→Generate new token



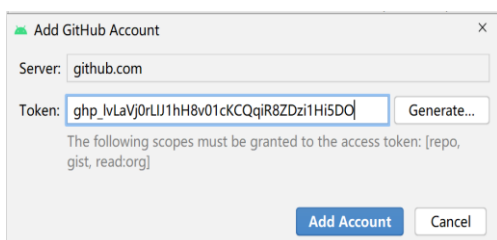
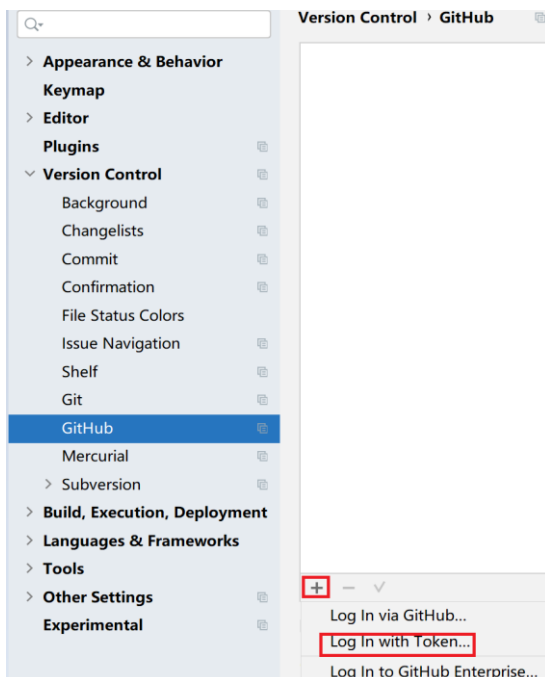
勾选上如下权限

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo:deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input checked="" type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input checked="" type="checkbox"/> read:org	Read org and team membership, read org projects
<input type="checkbox"/> admin:public_key	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input type="checkbox"/> read:repo_hook	Read repository hooks
<input type="checkbox"/> admin:org_hook	Full control of organization hooks
<input checked="" type="checkbox"/> gist	Create gists

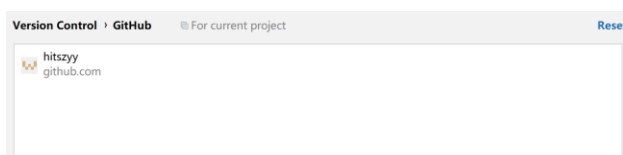
复制生成的 token



3. AS 中 File → Settings→Version Control→GitHub, 不要使用直接添加 GitHub 账号的方式, 会失败



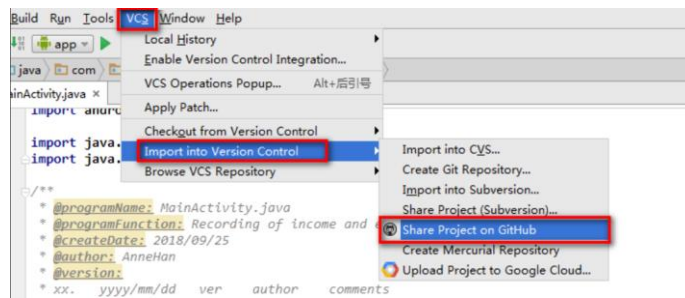
账户添加成功后



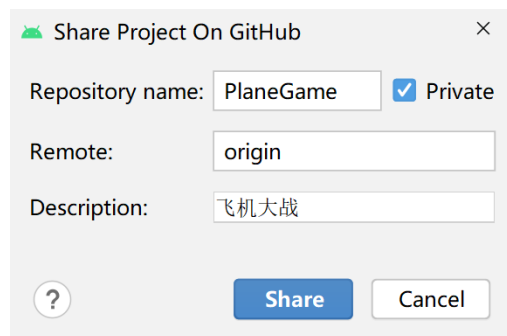
4. 上传项目到 GitHub

Android Studio→VCS→Import into Version Control→Share

Project on GitHub



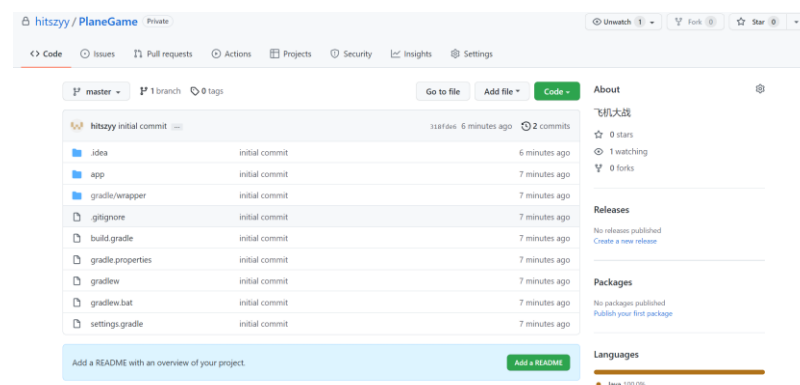
输入项目信息



AS 中显示 push 成功

```
15:51:40.406: [PlaneGame] git -c ... push --progress --porcelain origin ref
Writing objects: 100% (79/79), 399.74 KiB | 15.99 MiB/s, done.
Total 79 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/hitszyy/PlaneGame.git
 * refs/heads/master:refs/heads/master [new branch]
Branch 'master' set up to track remote branch 'master' from 'origin'
```

Github 页面上出现代码



注意：github 不稳定，建议大家优先使用 gitee。