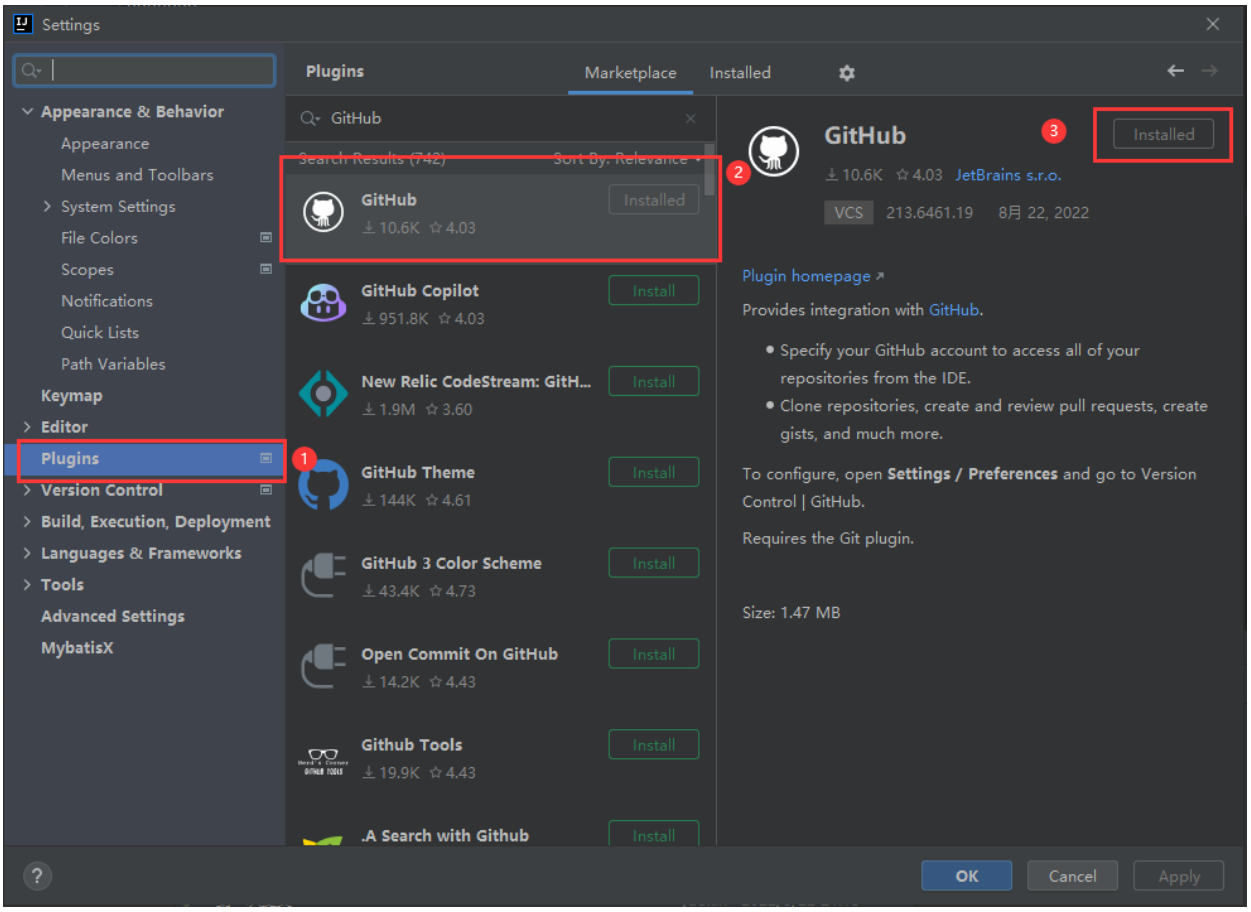
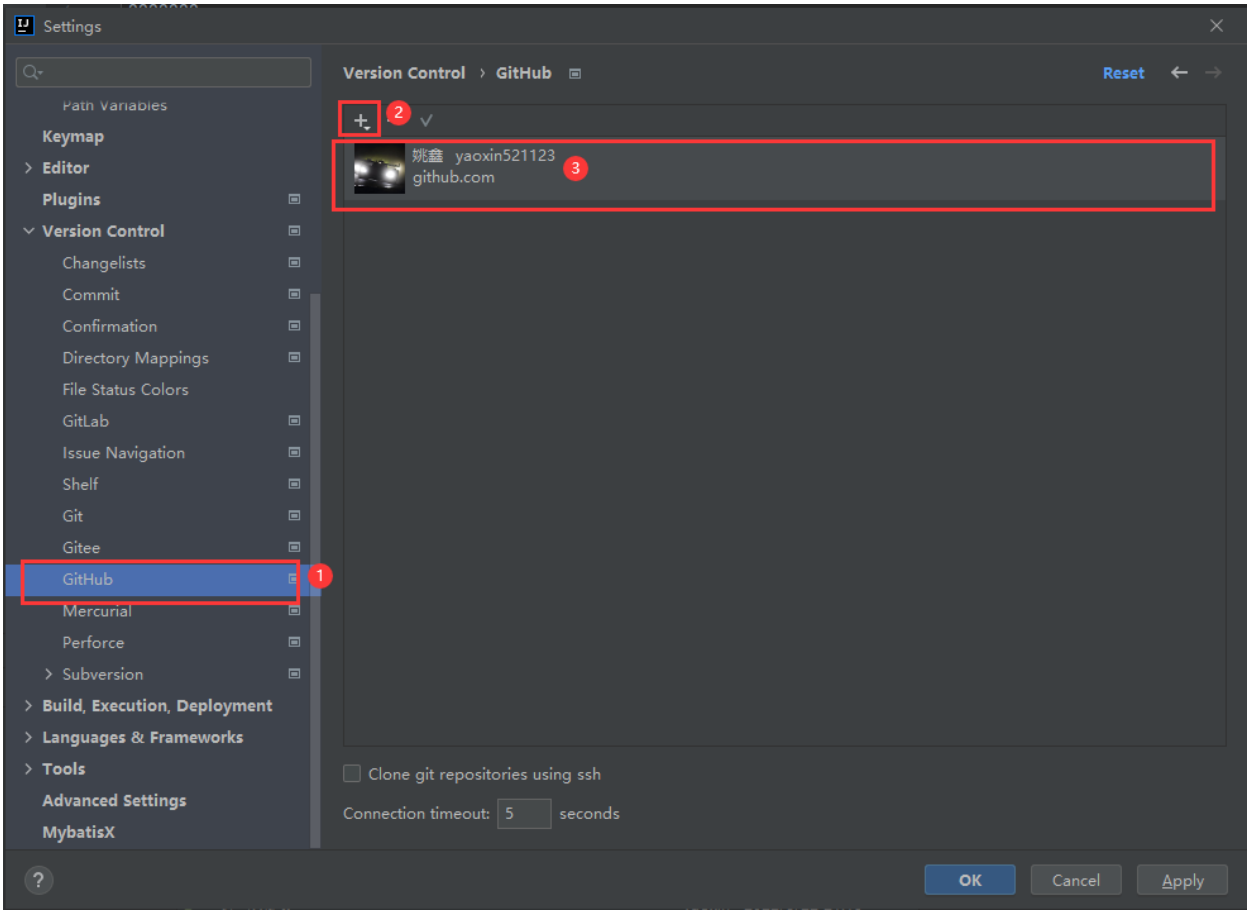
第11讲 Git IDEA VS GitHub

安装 GitHub 插件



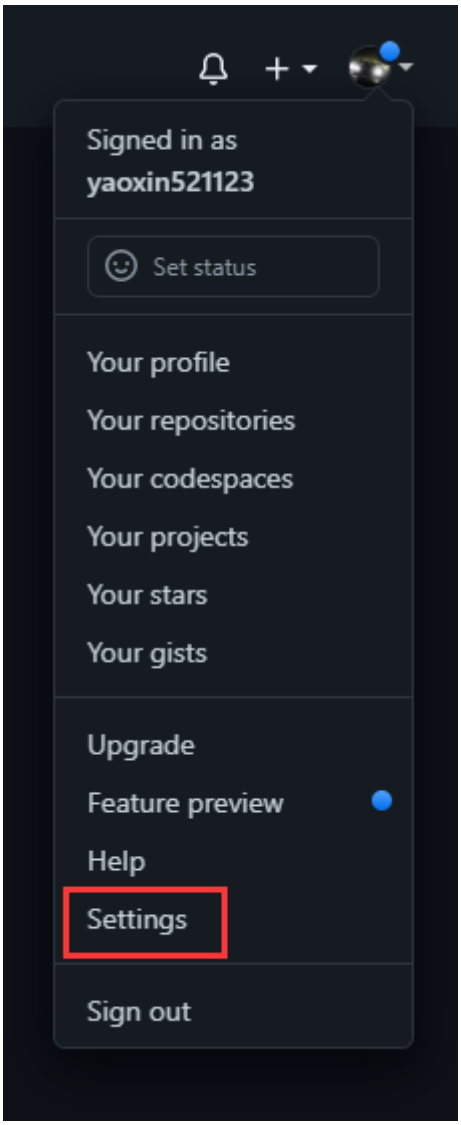
设置 GitHub 账号

- 直接登录帐号密码。

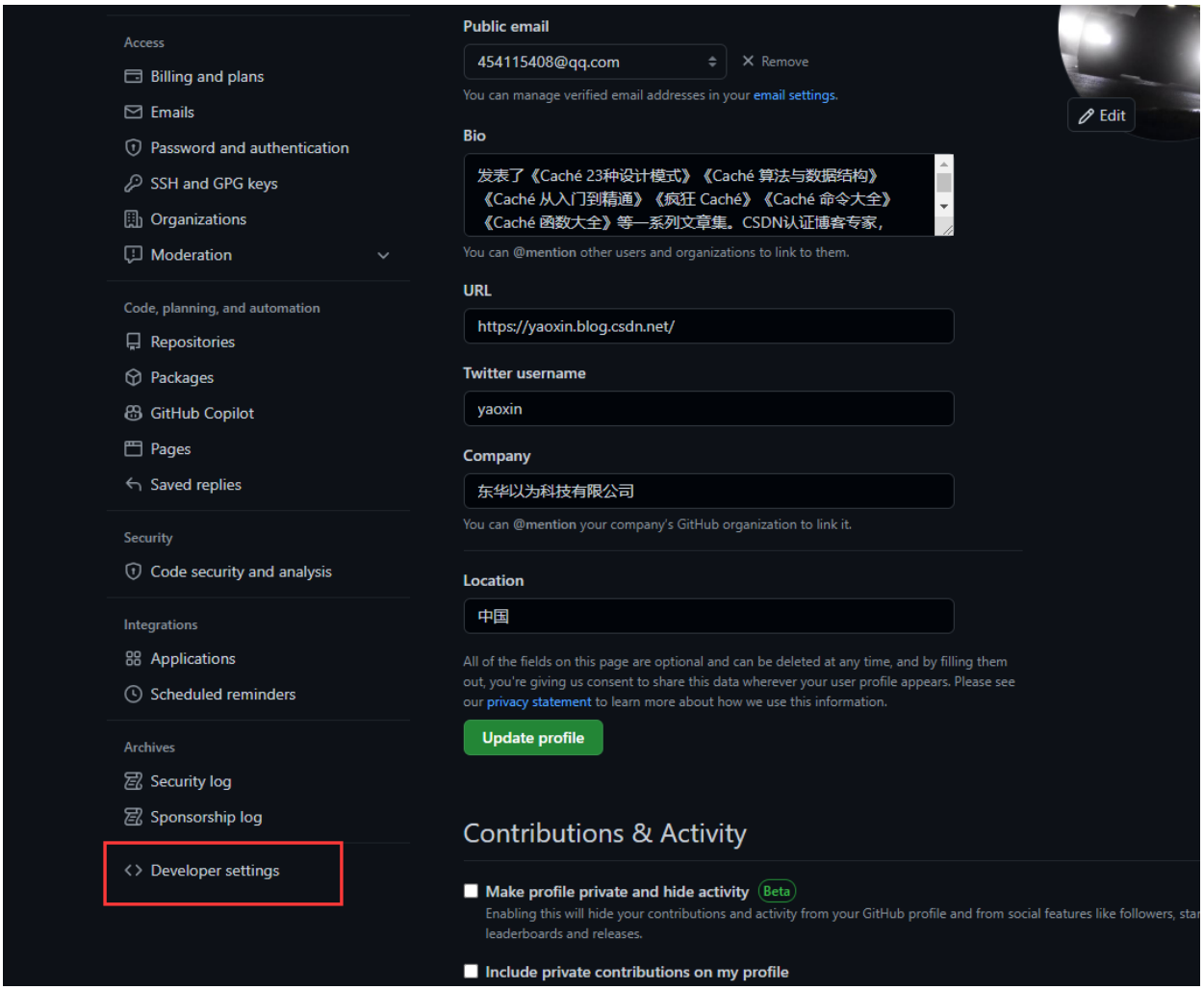


- 生成口令。

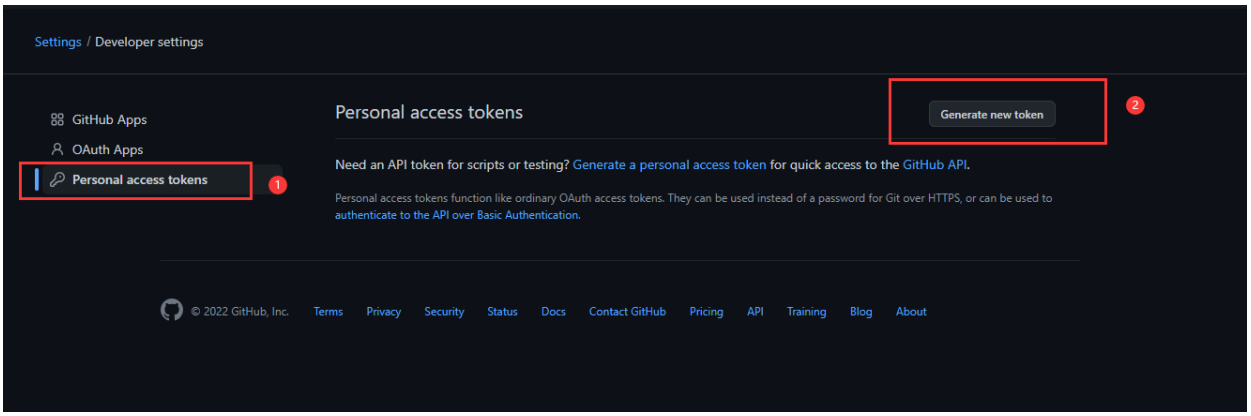
1. 选择Setttings。



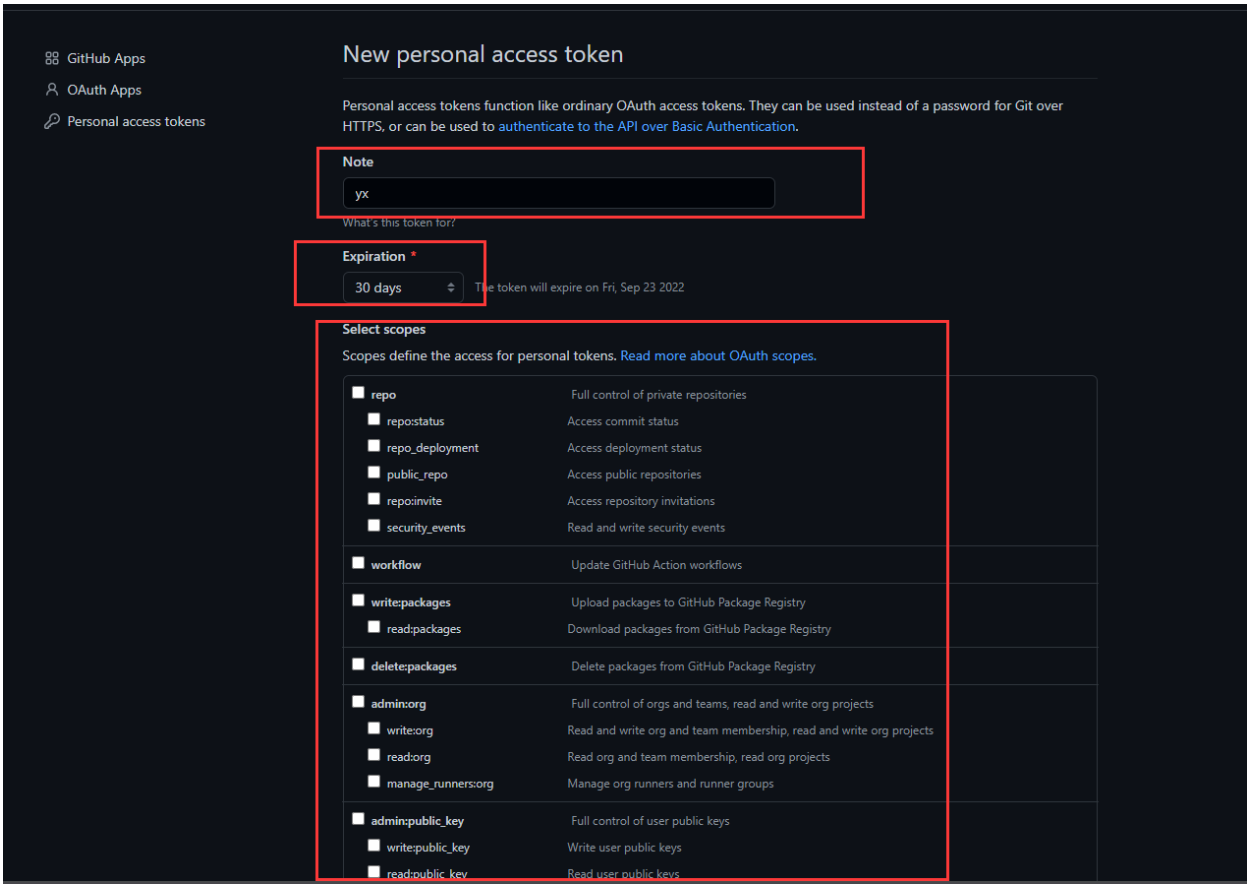
2. Developer settings。

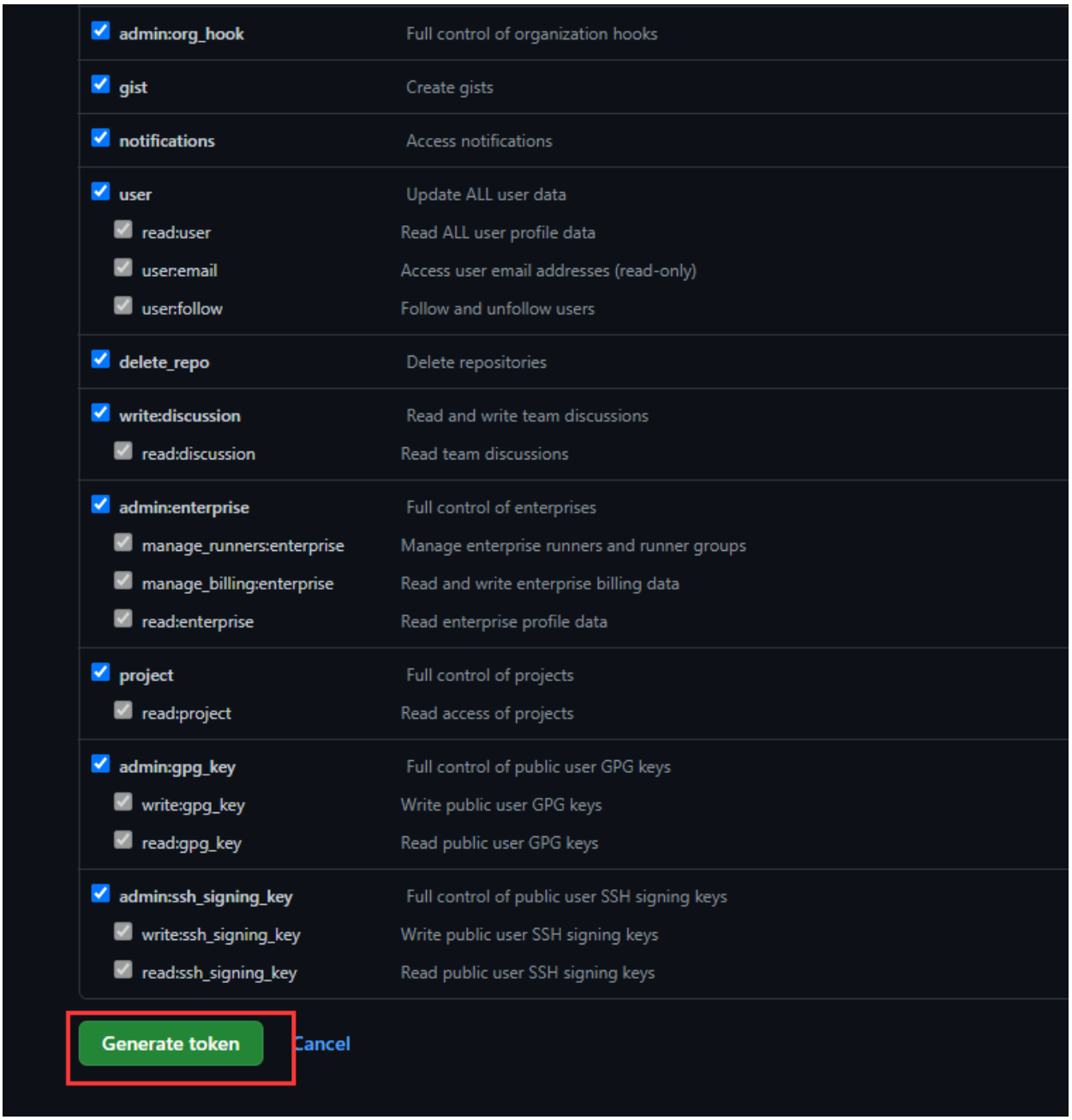


3. Personal access tokens > 生成新令牌。

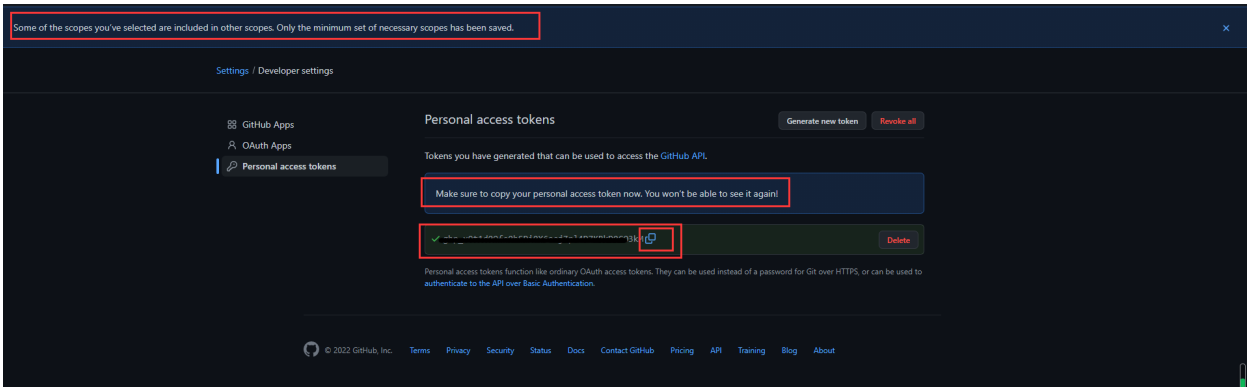


4. 填写名称 > 有效日期 > 功能全选。

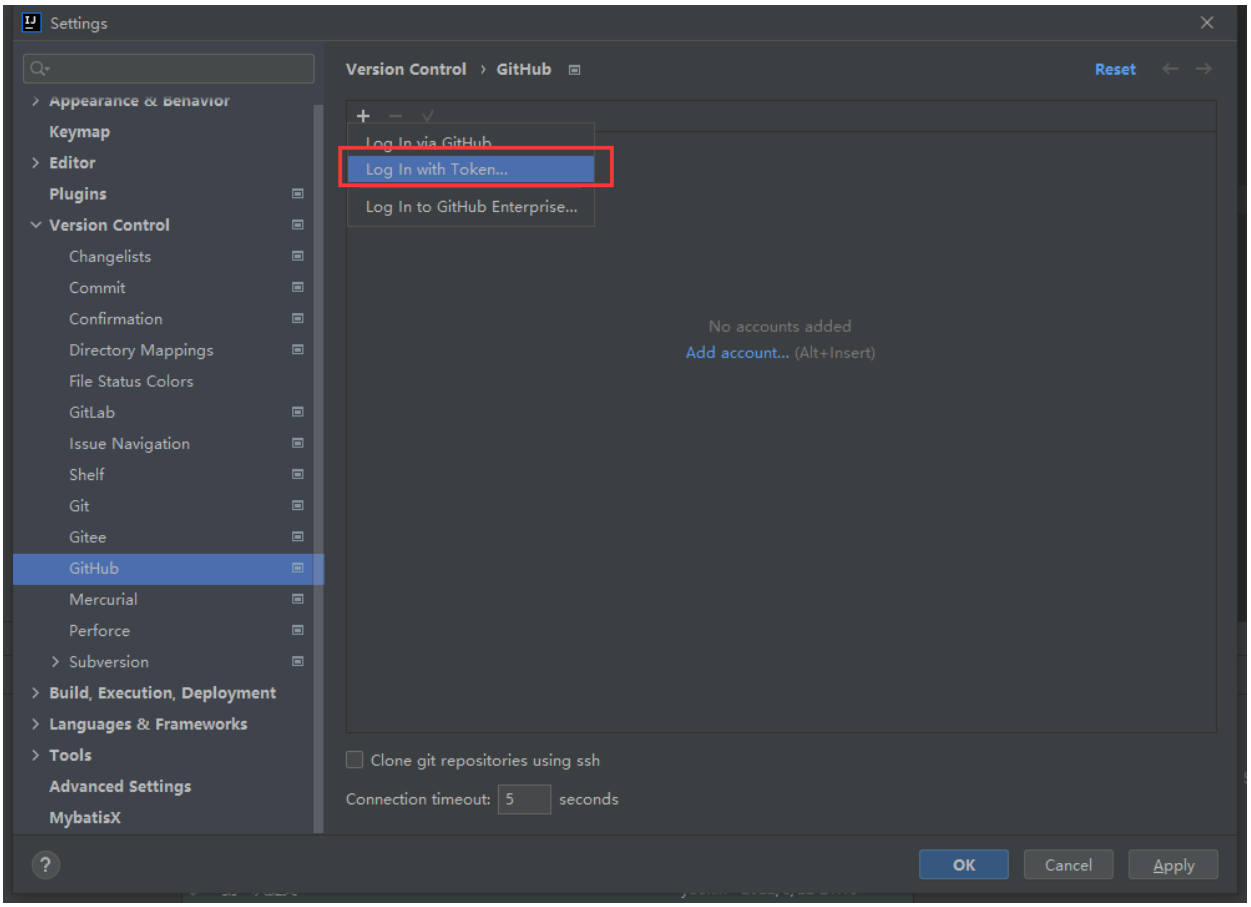

GitHub Apps
OAuth Apps
Personal access tokens
New personal access token
Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.
Note
yx
What's this token for?
Expiration *
30 days
The token will expire on Fri, Sep 23 2022
Select scopes
Scopes define the access for personal tokens. Read more about OAuth scopes.
repo
Full control of private repositories
repo:status
Access commit status
repo_deployment
Access deployment status
public_repo
Access public repositories
repo:invite
Access repository invitations
security_events
Read and write security events
workflow
Update GitHub Action workflows
write:packages
Upload packages to GitHub Package Registry
read:packages
Download packages from GitHub Package Registry
delete:packages
Delete packages from GitHub Package Registry
admin:org
Full control of orgs and teams, read and write org projects
write:org
Read and write org and team membership, read and write org projects
read:org
Read org and team membership, read org projects
manage_runners:org
Manage org runners and runner groups
admin:public_key
Full control of user public keys
write:public_key
Write user public keys
read:public_key
Read user public keys



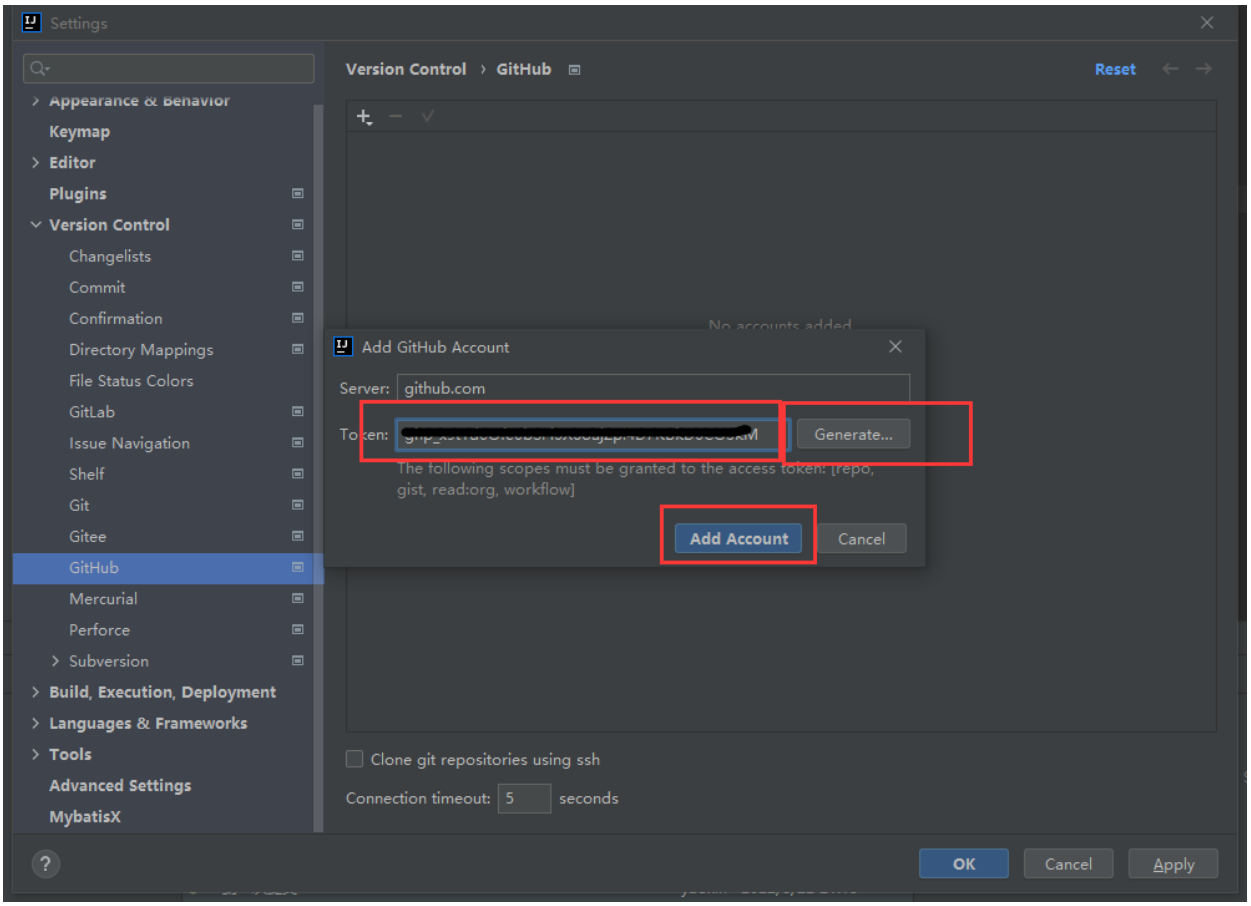
5. 点击生成口令。



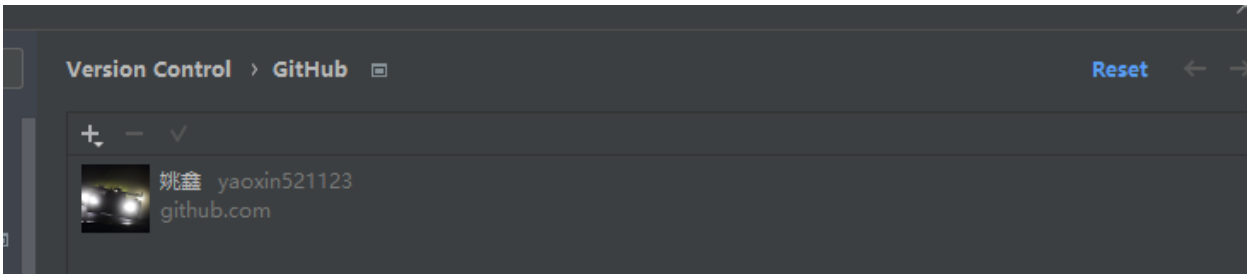
6. 选择Token。



7. Add Account。

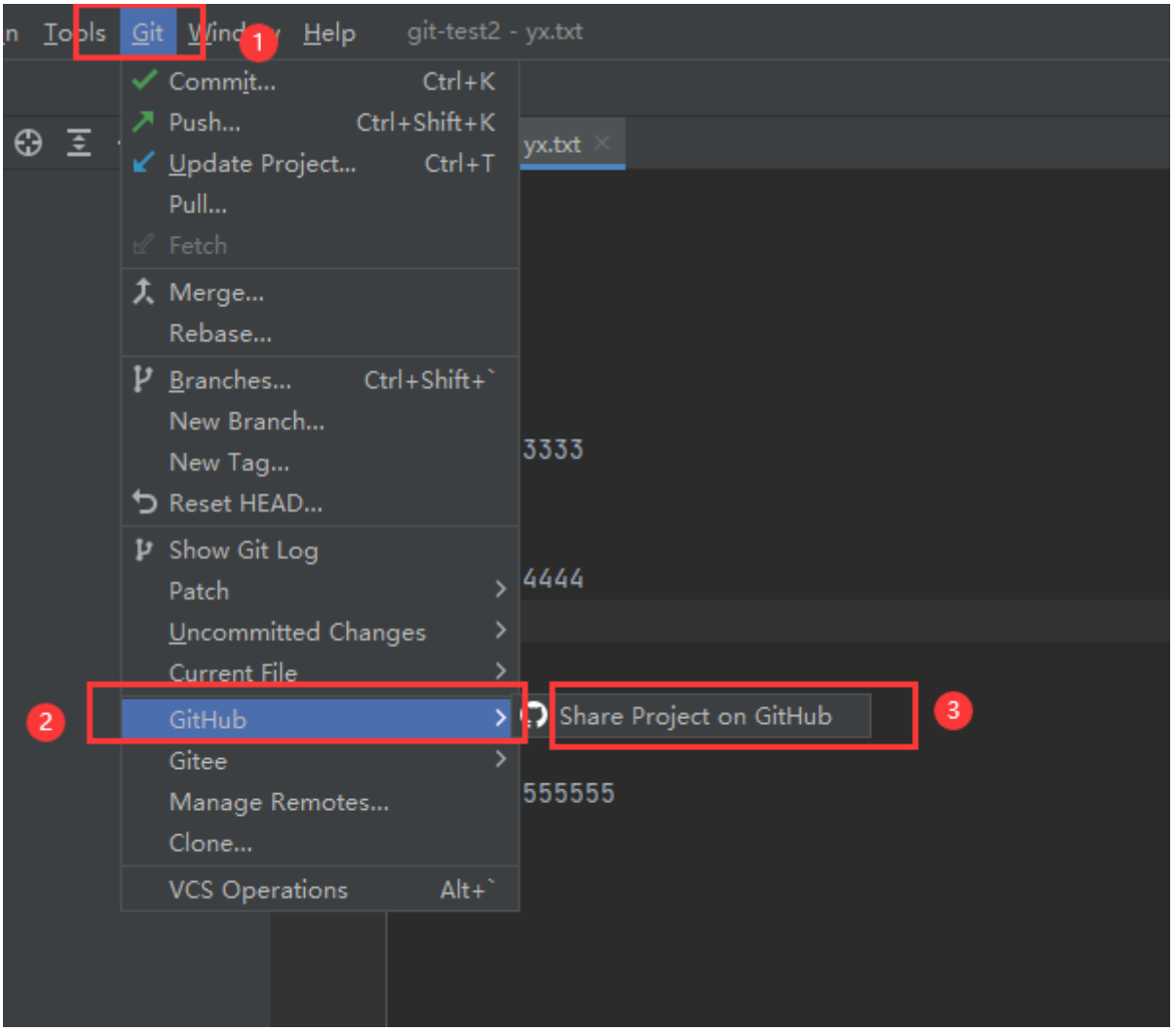


8. 成功。

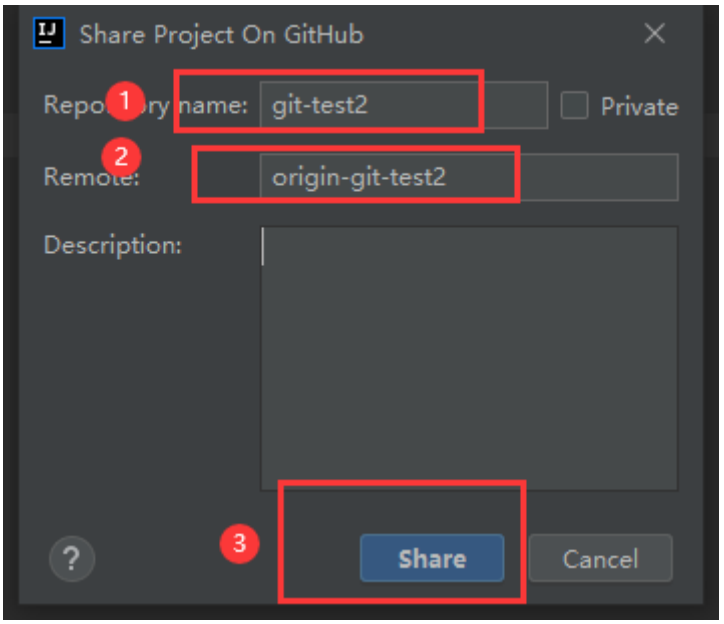


分享工程到 GitHub

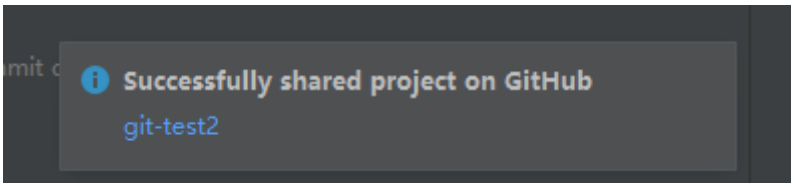
1. Git > GitHub > Share Projuect on GitHub。



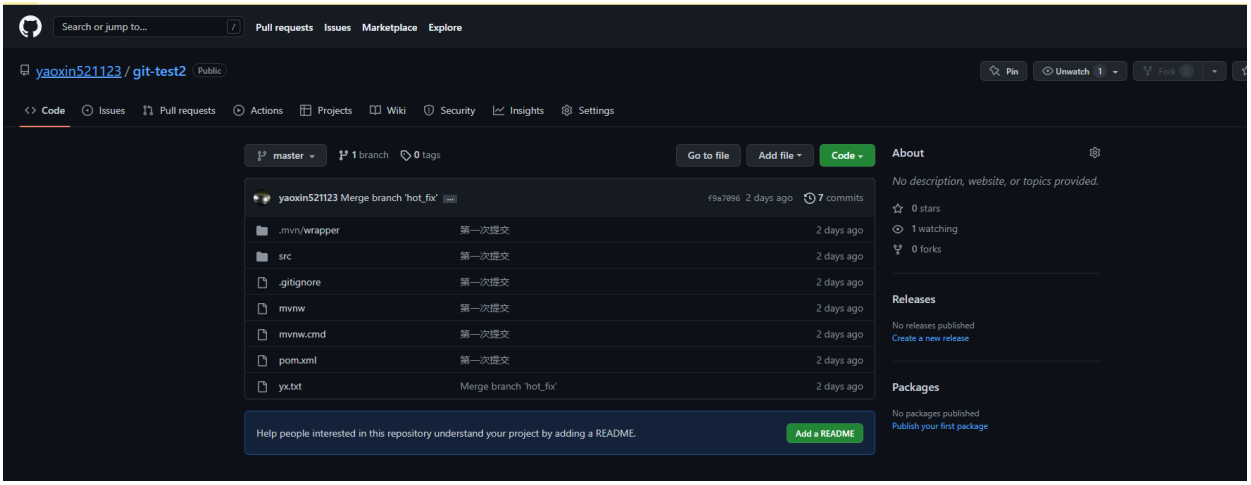
2. 输入别名 > Share。



3. 提示成功。



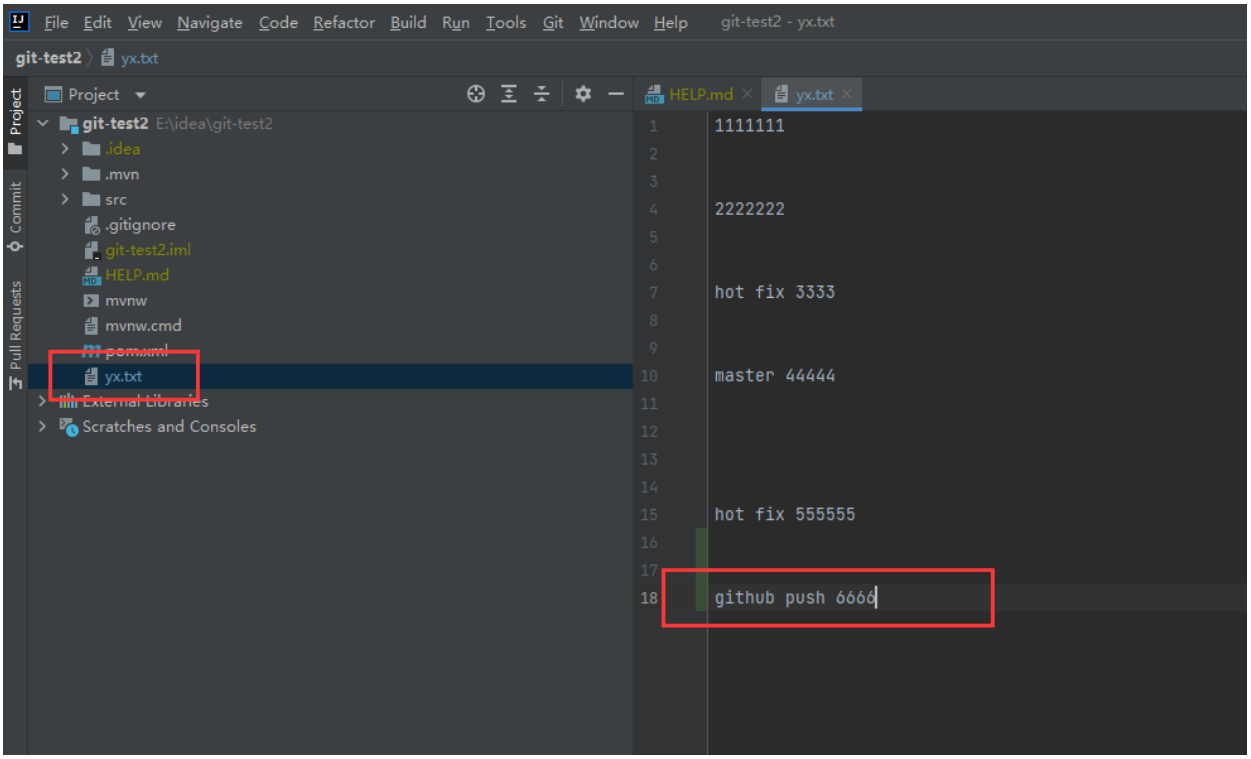
- GitHub 自动创建远程仓库。



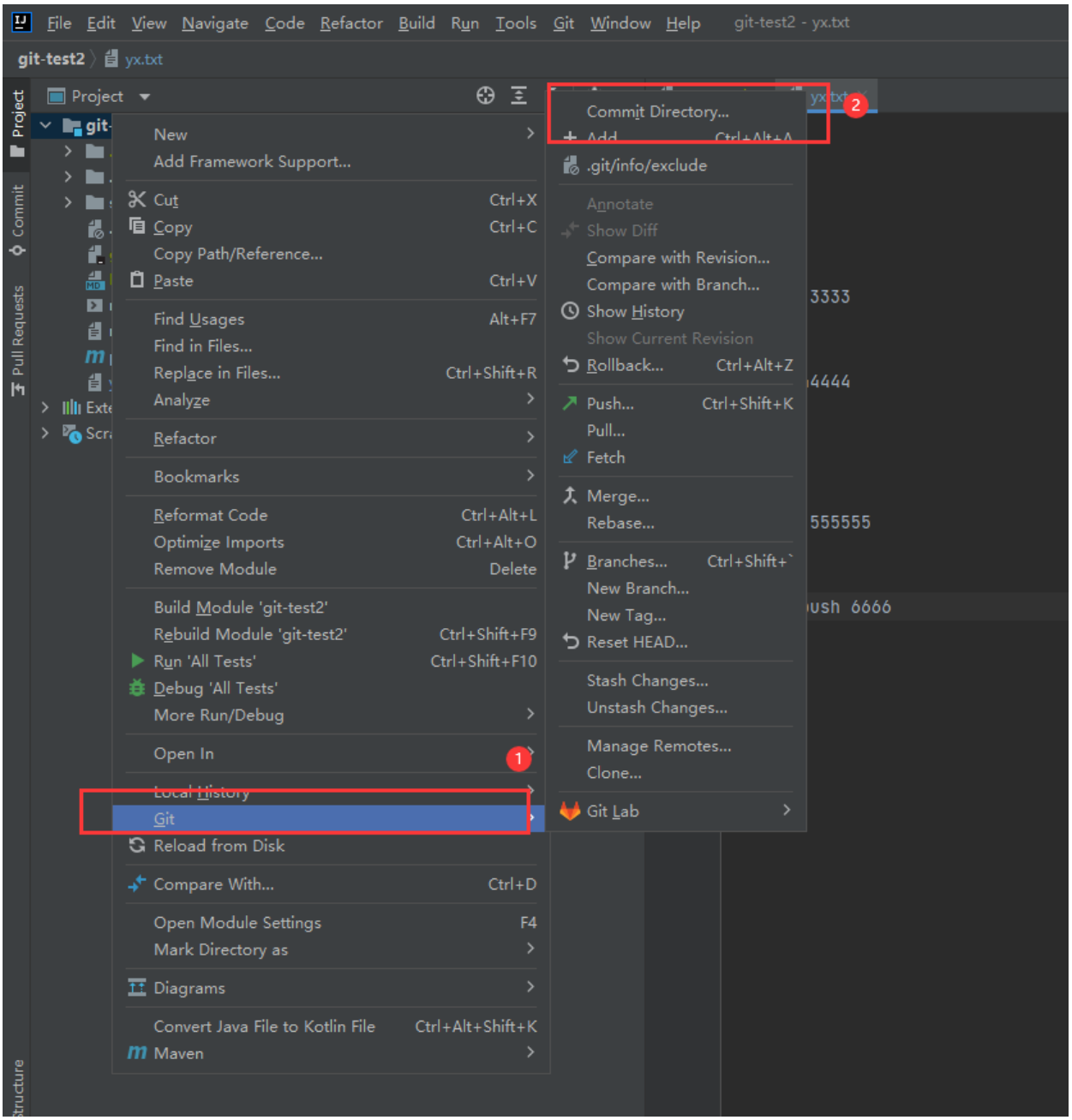
push 推送本地库到远程库

- HTTPS 链接。

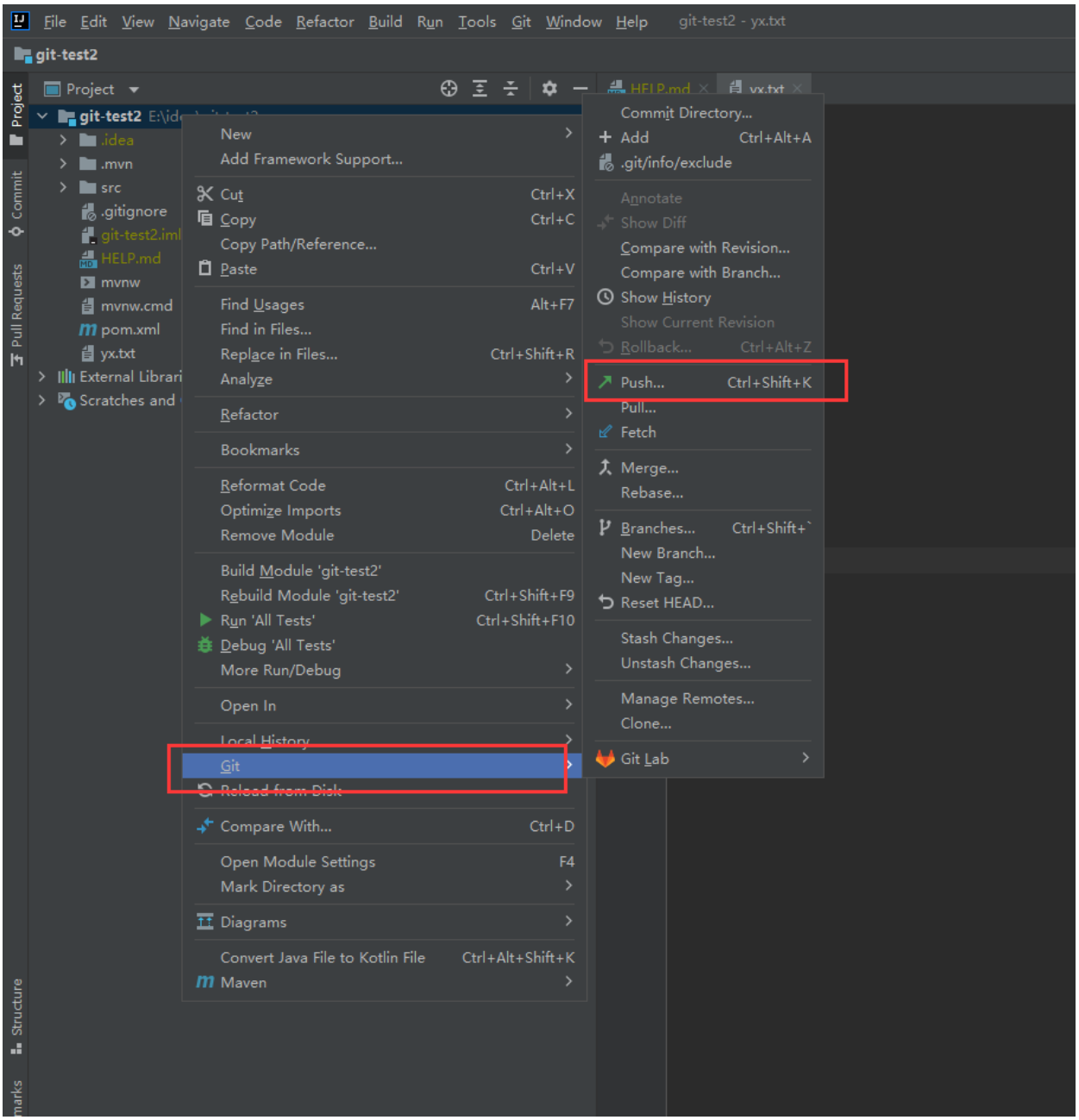
1. 修改本地代码。

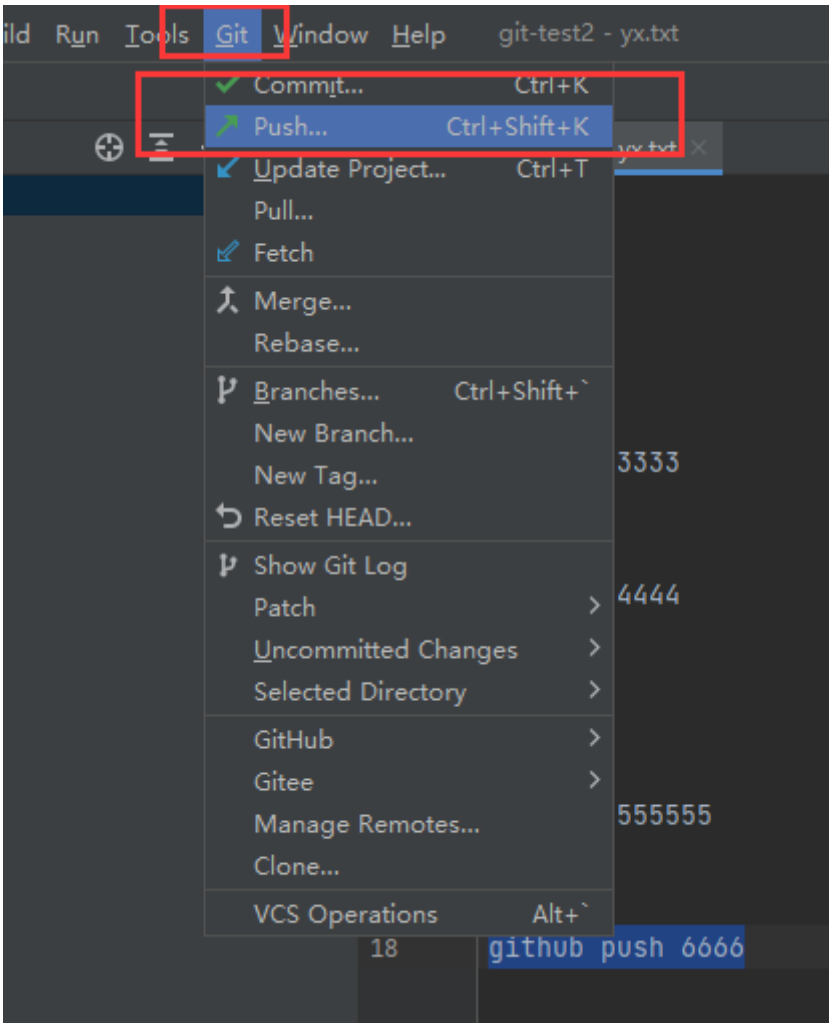

File Edit View Navigate Code Refactor Build Run Tools Git Window Help
git-test2 - yx.txt
git-test2
yx.txt
Project
Commit
Pull Requests
git-test2 E:\idea\git-test2
.idea
.mvn
src
.gitignore
git-test2.iml
HELP.md
mvnw
mvnw.cmd
yx.txt
Scratches and Consoles
HELP.md
yx.txt
1111111
2222222
hot fix 3333
master 44444
hot fix 555555
github push 6666

2. 提交本地库。

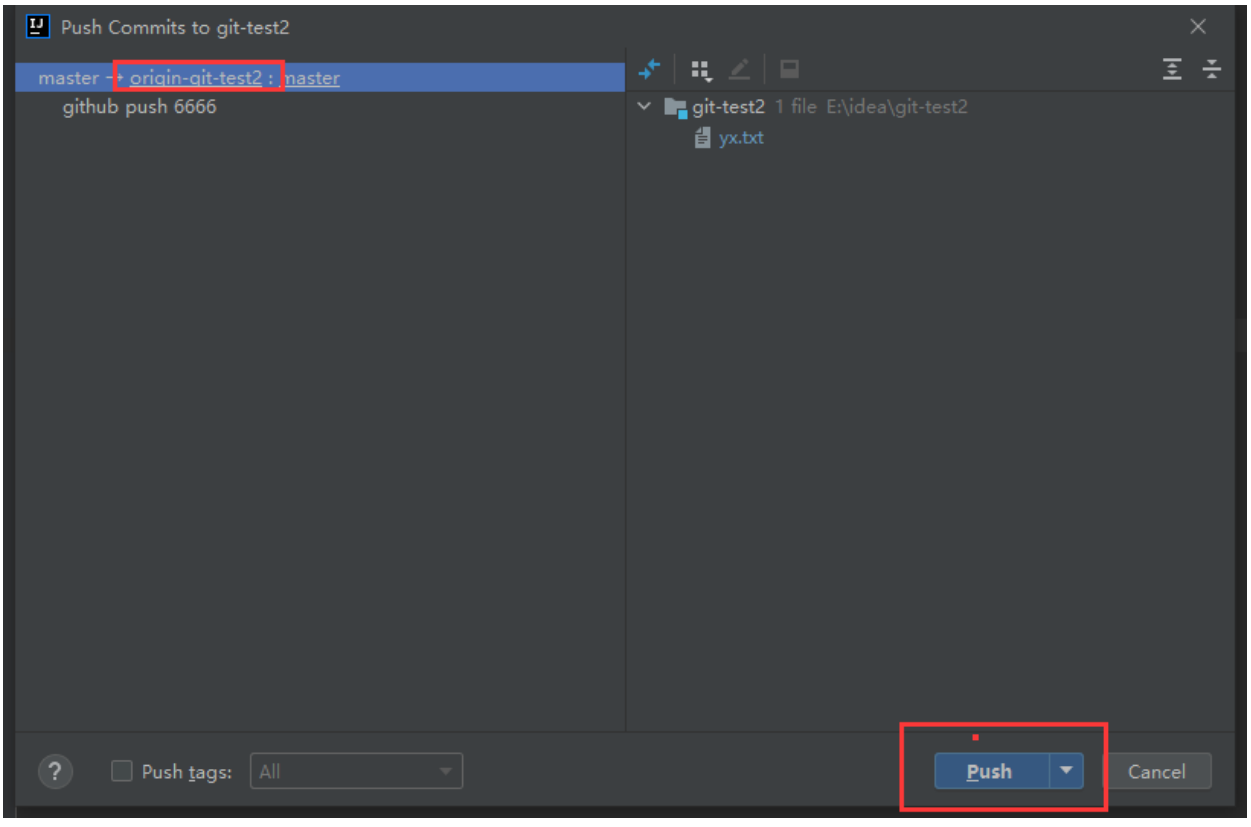


3. Push远程库GitHub。

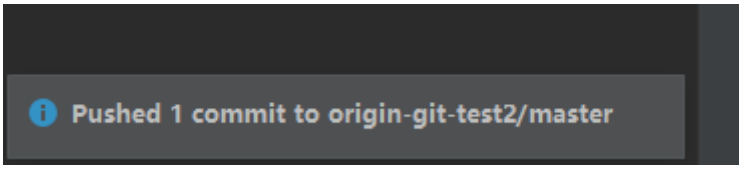

File Edit View Navigate Code Refactor Build Run Tools Git Window Help
git-test2 - yx.txt
git-test2
Project
git-test2
.idea
.mvn
src
.gitignore
git-test2.iml
HELP.md
mvnw
mvnw.cmd
pom.xml
yx.txt
External Librari
Scratches and
HELP.md
yx.txt
New
Add Framework Support...
Cut Ctrl+X
Copy Ctrl+C
Copy Path/Reference...
Paste Ctrl+V
Find Usages Alt+F7
Find in Files...
Replace in Files... Ctrl+Shift+R
Analyze
Refactor
Bookmarks
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Remove Module Delete
Build Module 'git-test2'
Rebuild Module 'git-test2' Ctrl+Shift+F9
Run 'All Tests' Ctrl+Shift+F10
Debug 'All Tests'
More Run/Debug
Open In
Local History
Git
Reload from Disk
Compare With... Ctrl+D
Open Module Settings F4
Mark Directory as
Diagrams
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Maven
Commit Directory...
Add Ctrl+Alt+A
.git/info/exclude
Annotate
Show Diff
Compare with Revision...
Compare with Branch...
Show History
Show Current Revision
Rollback... Ctrl+Alt+Z
Push... Ctrl+Shift+K
Pull...
Fetch
Merge...
Rebase...
Branches... Ctrl+Shift+`
New Branch...
New Tag...
Reset HEAD...
Stash Changes...
Unstash Changes...
Manage Remotes...
Clone...
Git Lab
Project
Commit
Pull Requests
Structure
marks

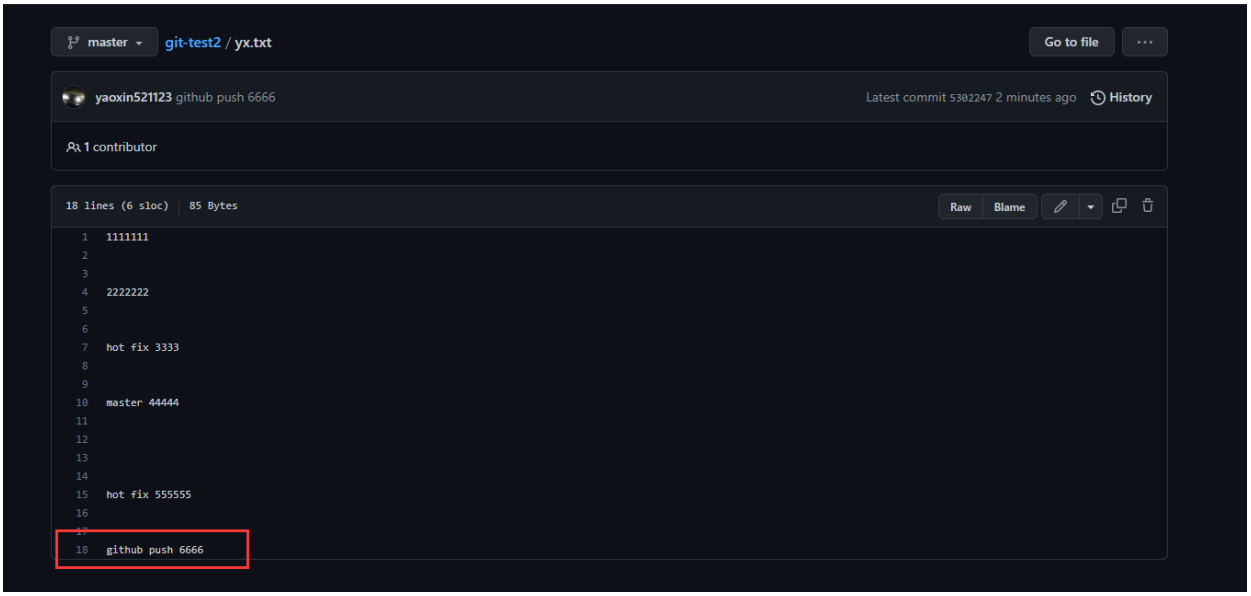

ild Run Tools Git Window Help git-test2 - yx.txt
Commit... Ctrl+K
Push... Ctrl+Shift+K
Update Project... Ctrl+T
Pull...
Fetch
Merge...
Rebase...
Branches... Ctrl+Shift+`
New Branch...
New Tag...
Reset HEAD...
Show Git Log
Patch
Uncommitted Changes
Selected Directory
GitHub
Gitee
Manage Remotes...
Clone...
VCS Operations Alt+`
3333
4444
555555
18
github push 6666

4. 点击Push即可。


Push Commits to git-test2
master → origin-git-test2 : master
github push 6666
git-test2 1 file E:\idea\git-test2
yx.txt
Push tags:
All
Push
Cancel

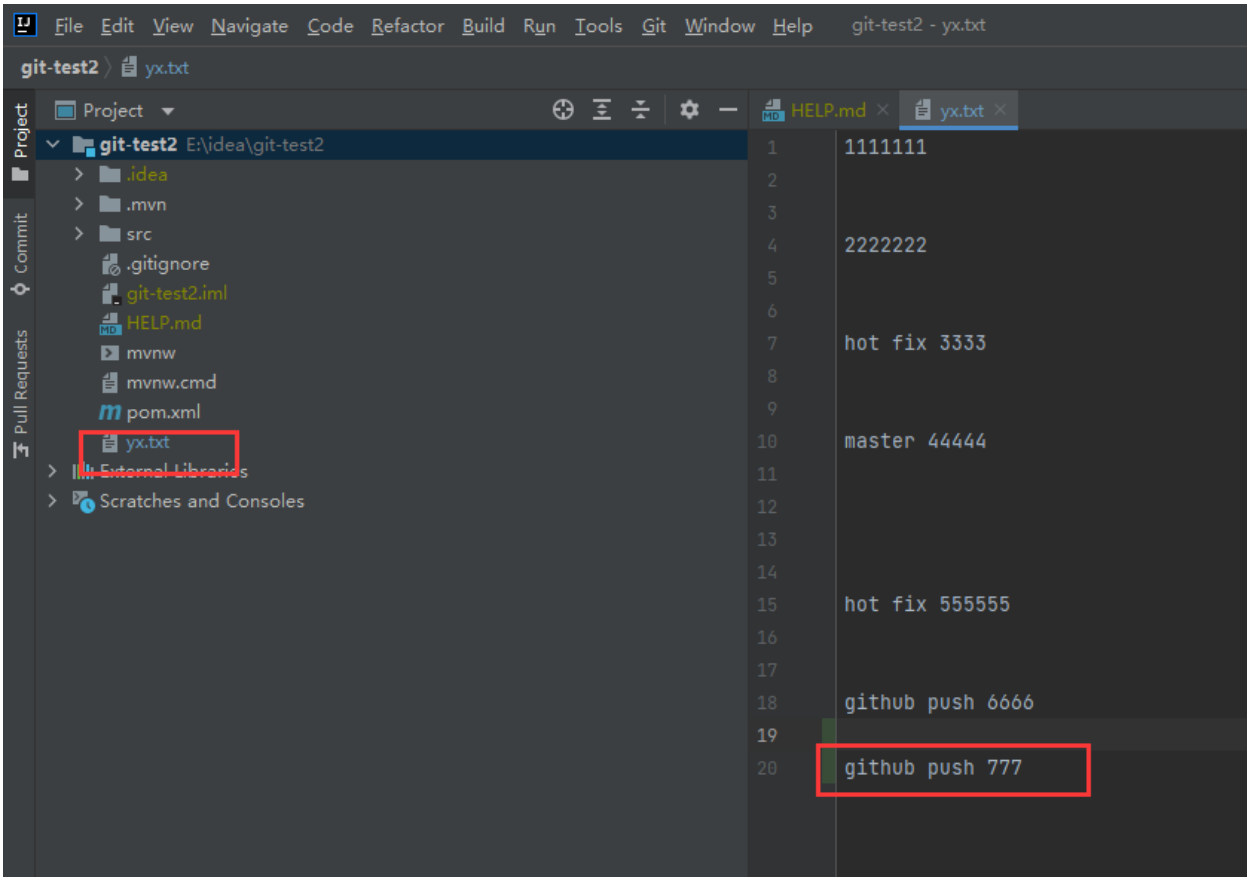
5. 查看GitHub远程库。



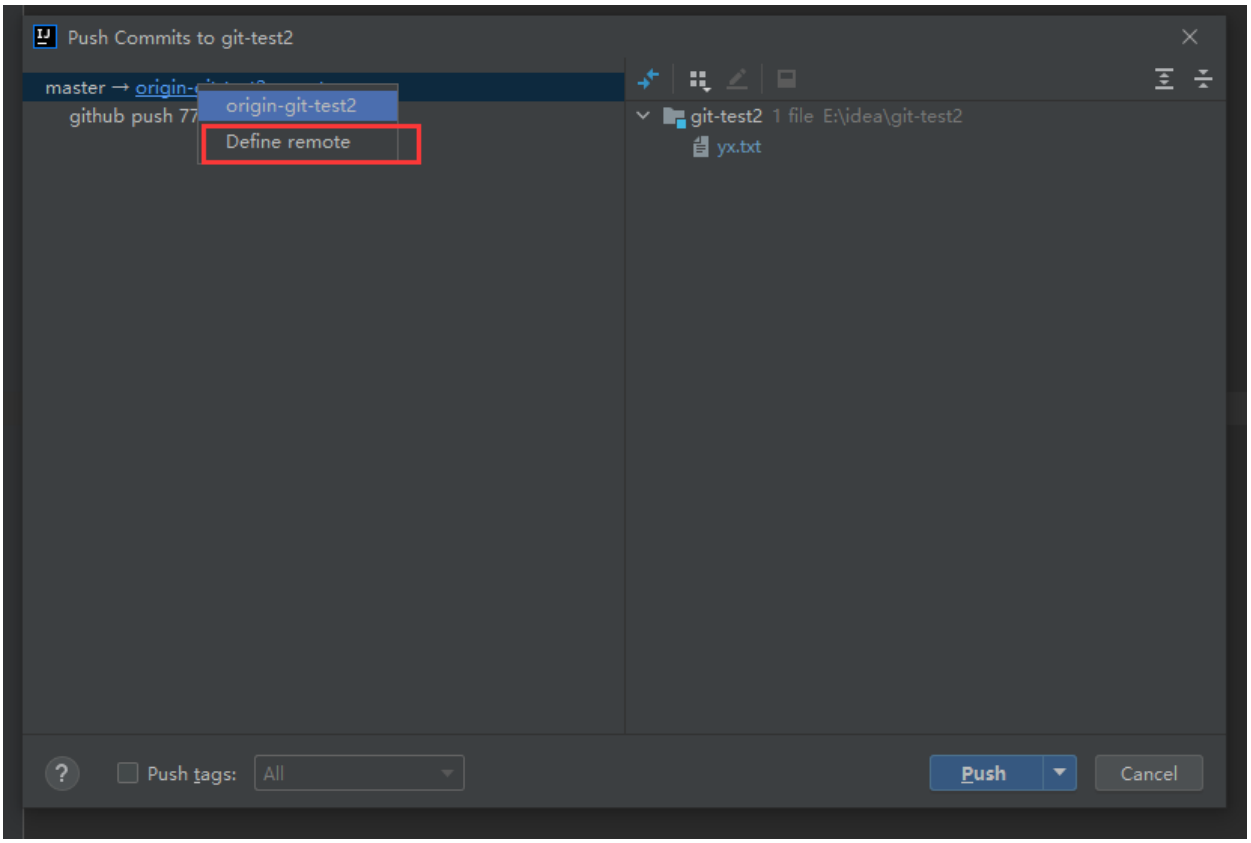


- SSH Push远程库。

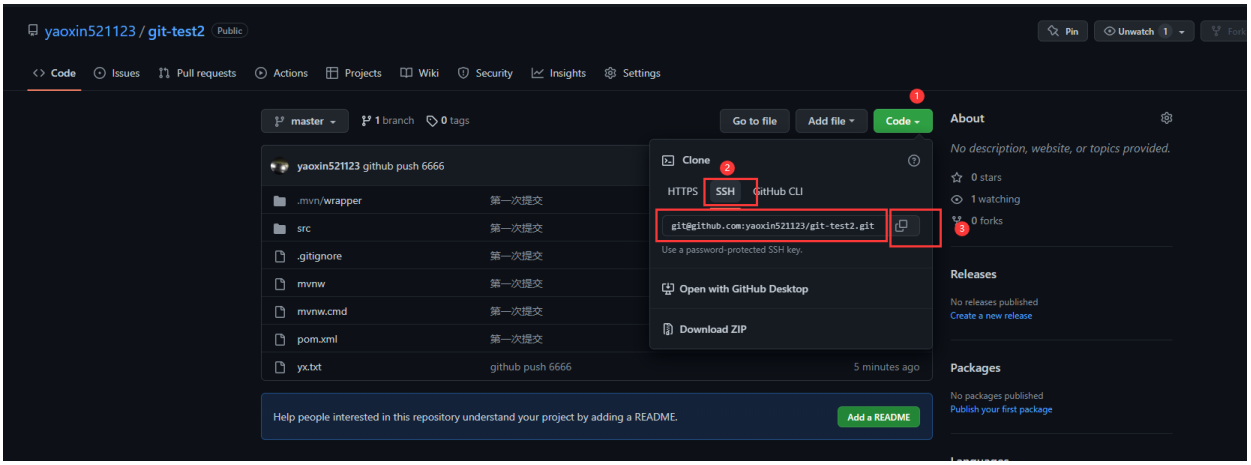
1. 修改代码，提交本地库。



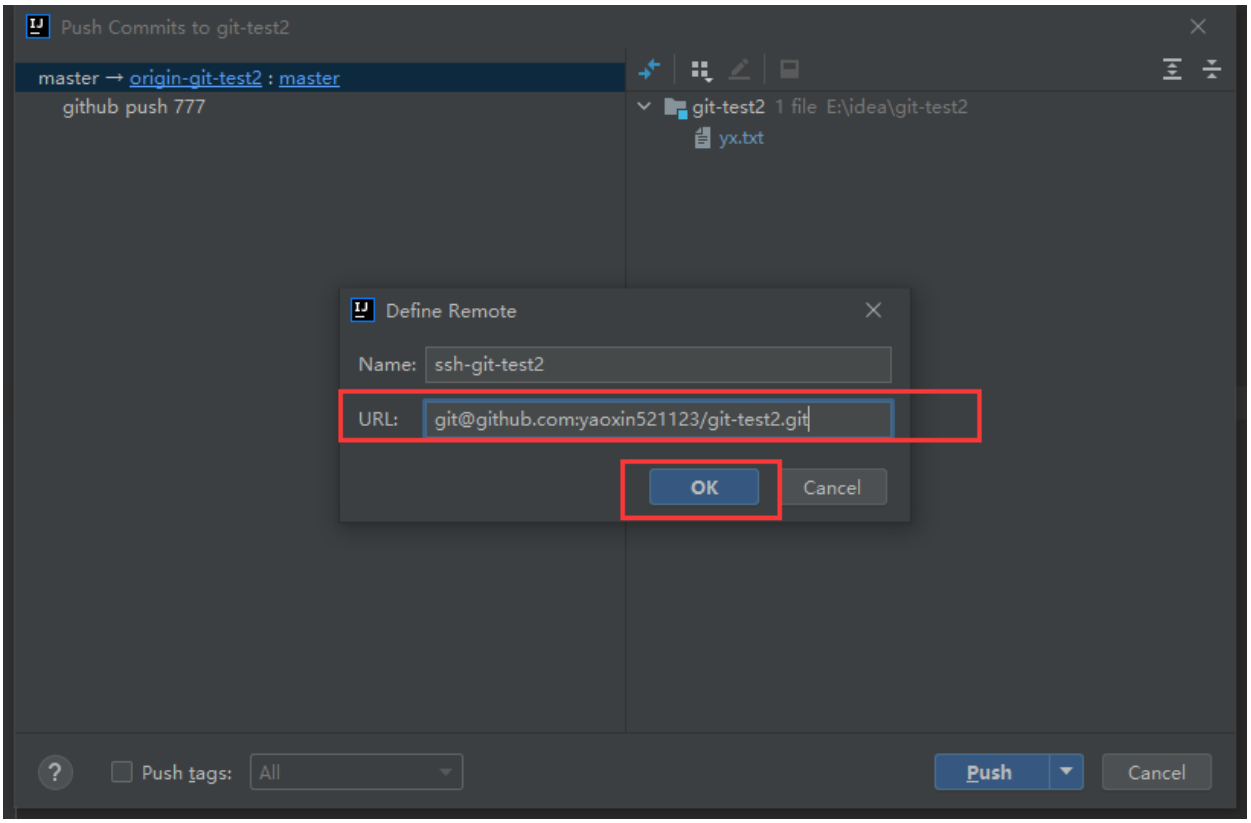
2. 创建新的连接。



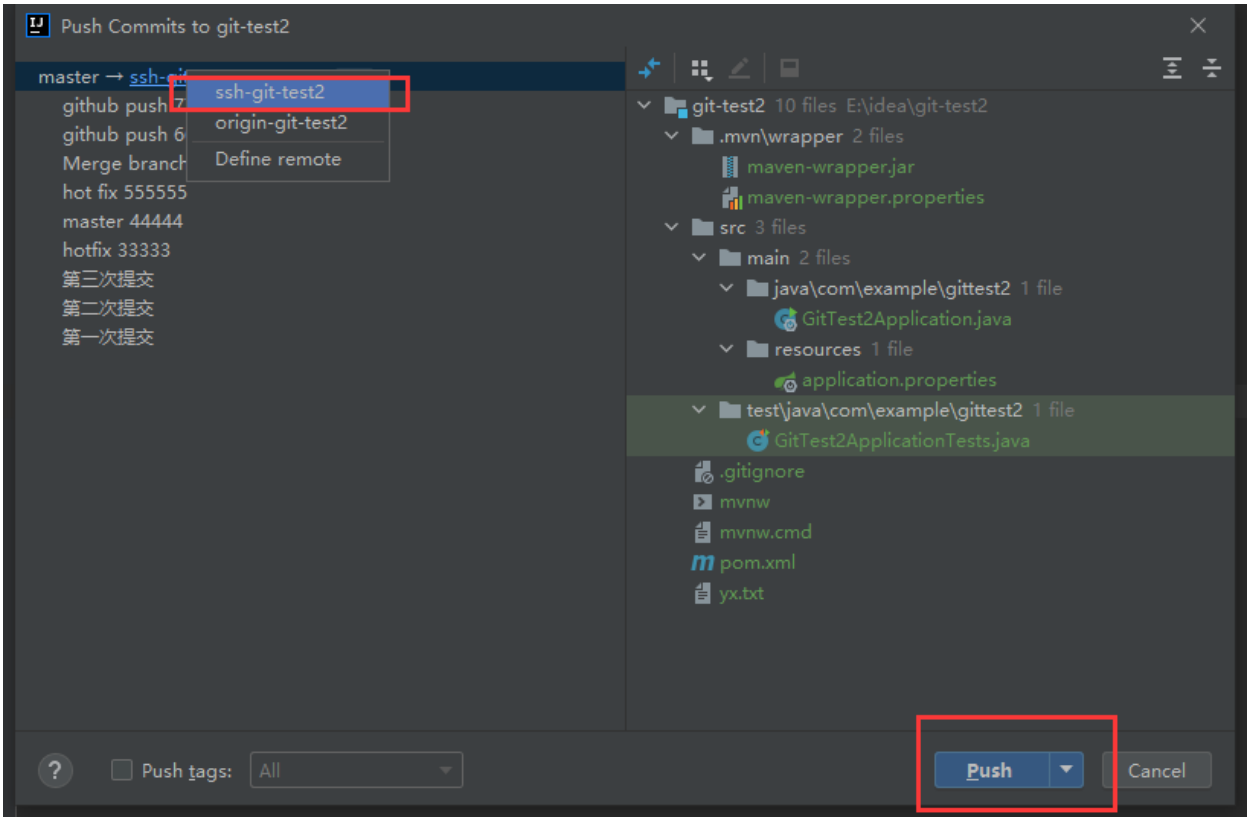
3. 复制SSH连接。



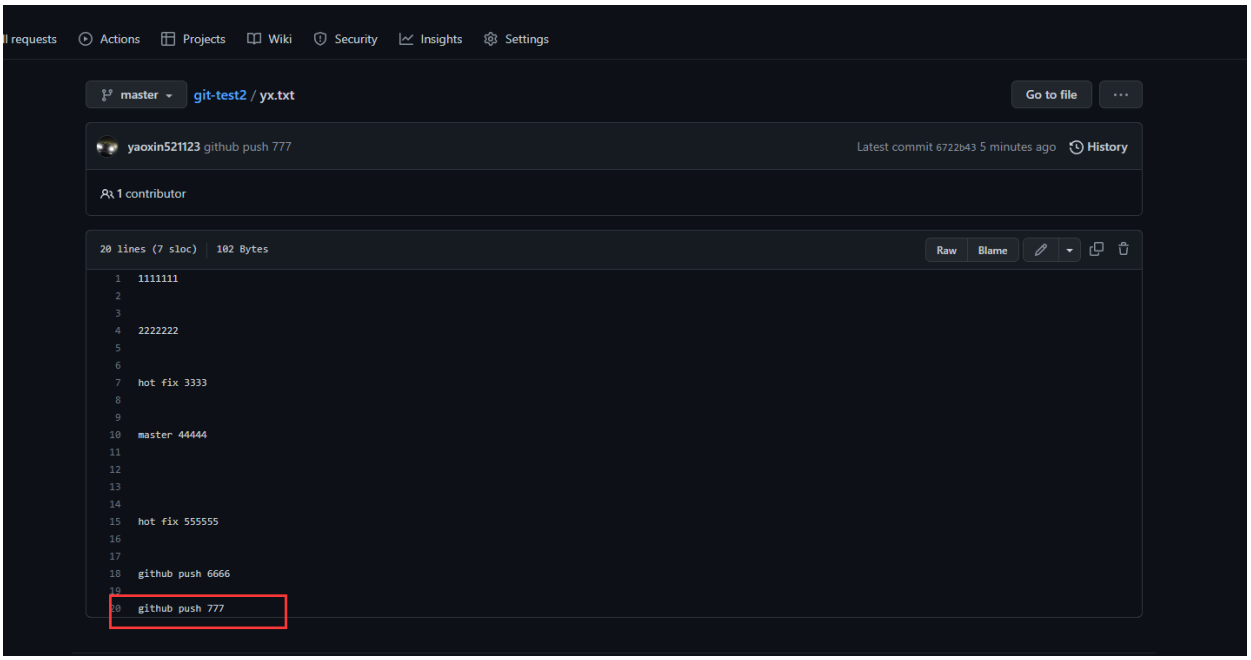
4. 修改连接名称，添加连接。



5. Push即可。



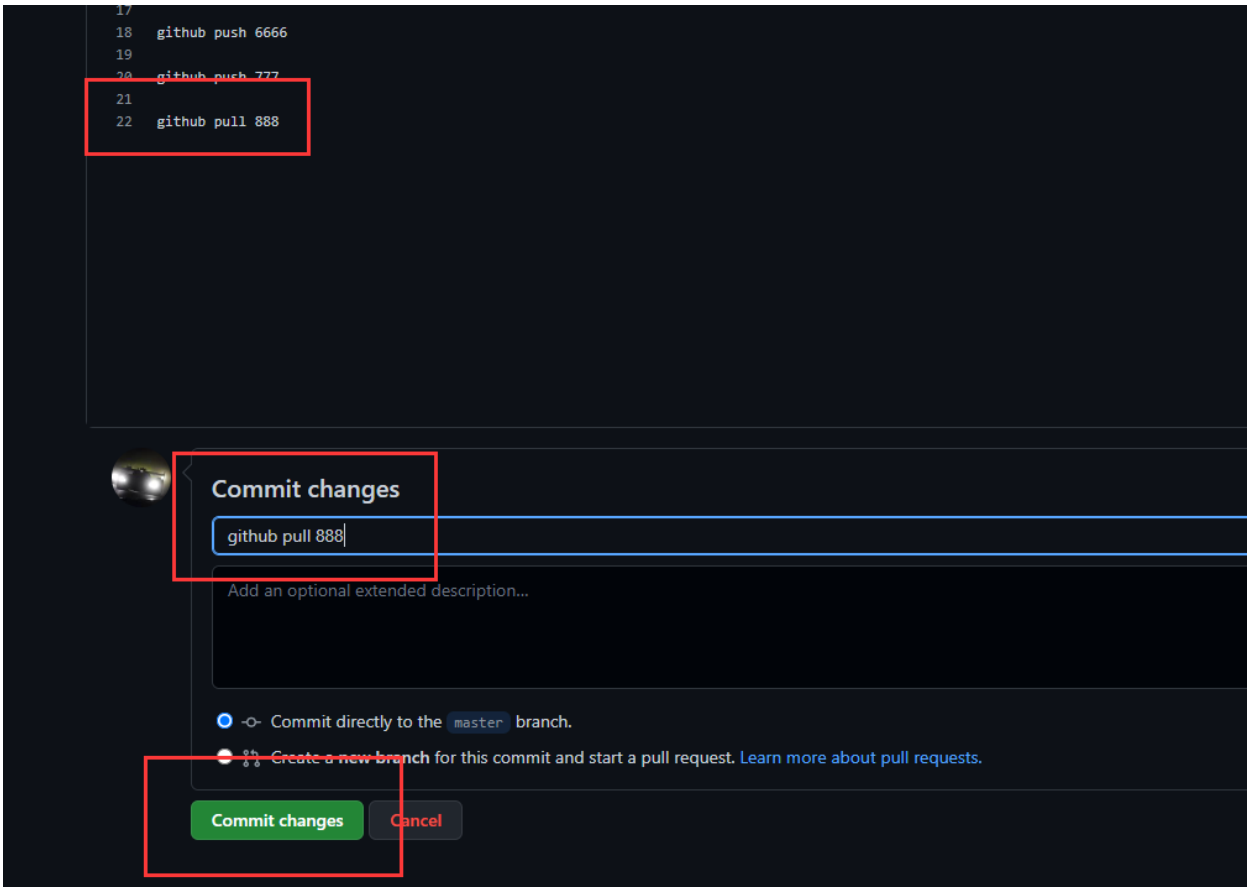
6. 查看远程库代码。



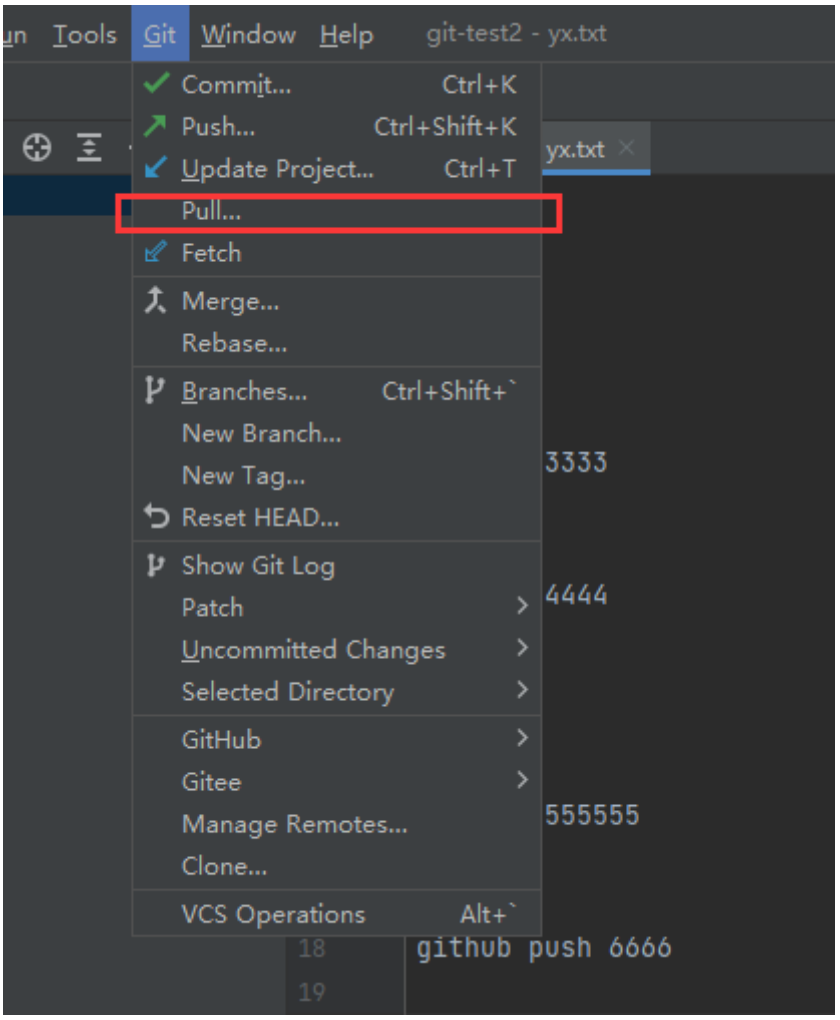
pull 拉取远程库到本地库

注意：push 是将本地库代码推送到远程库，如果本地库代码跟远程库代码版本不一致，push 的操作是会被拒绝的。也就是说，要想 push成功，一定要保证本地库的版本要比远程库的版本高！因此一个成熟的程序员在动手改本地代码之前，一定会先检查下远程库跟本地代码的区别！如果本地的代码版本已经落后，切记要先 pull拉取一下远程库的代码，将本地代码更新到最新以后，然后再修改，提交，推送！

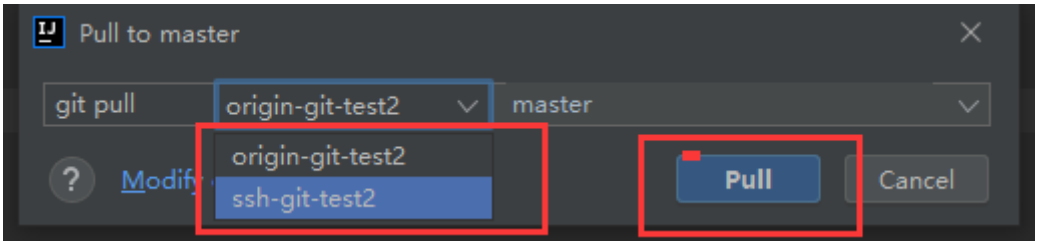
1. 修改远程库代码。


17
18 github push 6666
19
20 github push 777
21
22 github pull 888
Commit changes
github pull 888
Add an optional extended description...
Commit directly to the master branch.
Create a new branch for this commit and start a pull request. Learn more about pull requests.
Commit changes
Cancel

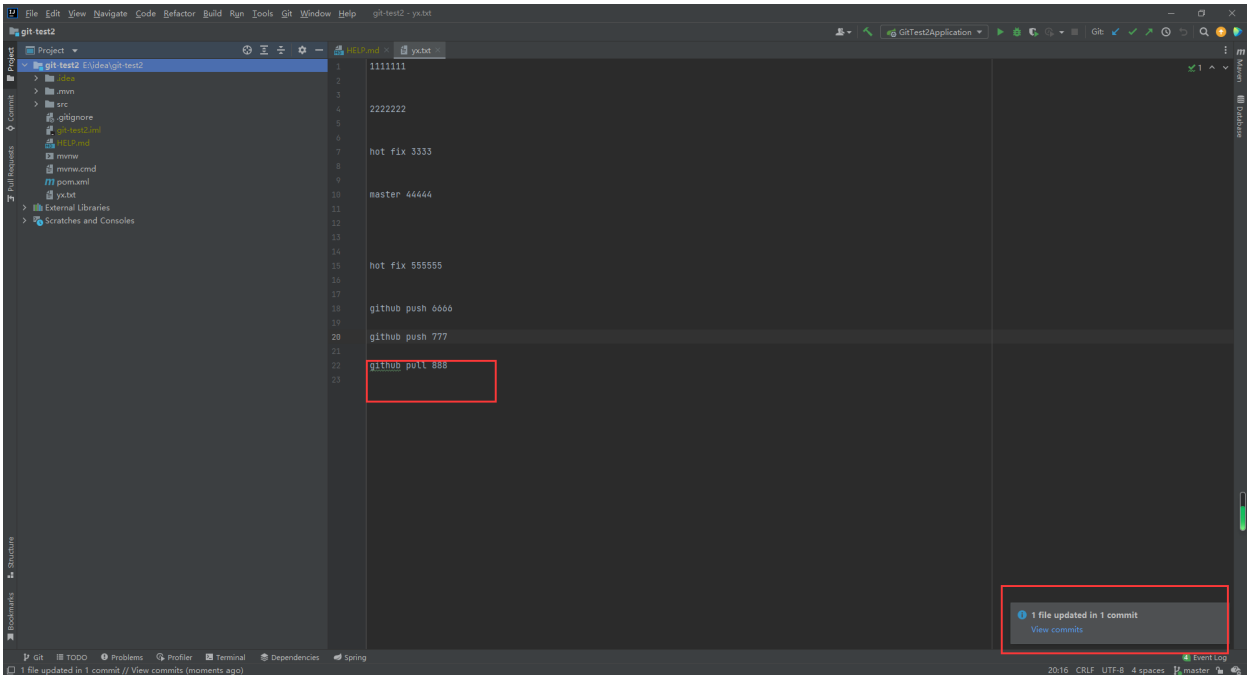
2. 选择Git > Pull。



3. 选择Pull 连接。



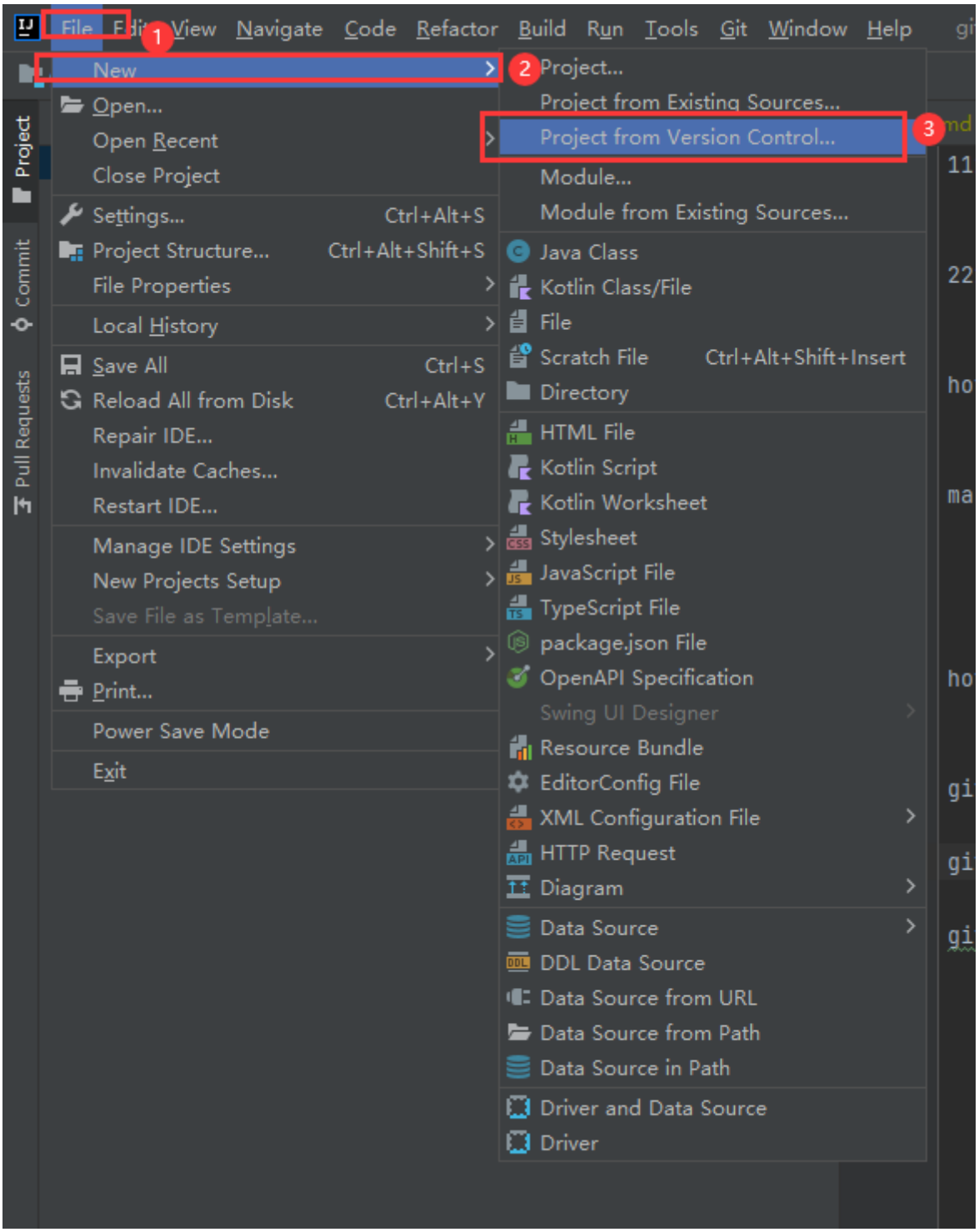
4. 查看结果。



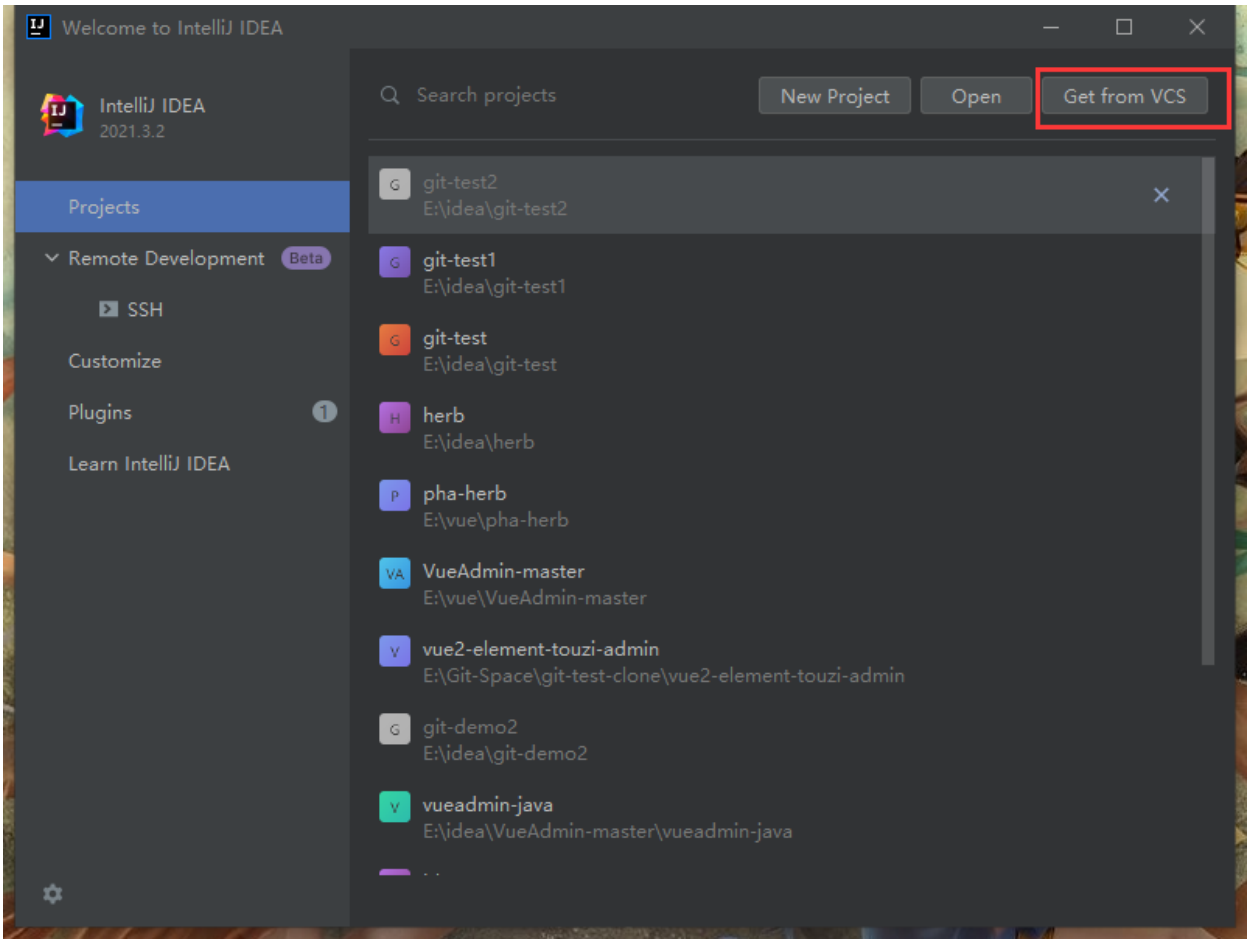
注意：pull 是拉取远端仓库代码到本地，如果远程库代码和本地库代码不一致，会自动合并，如果自动合并失败，还会涉及到手动解决冲突的问题。

clone 克隆远程库到本地

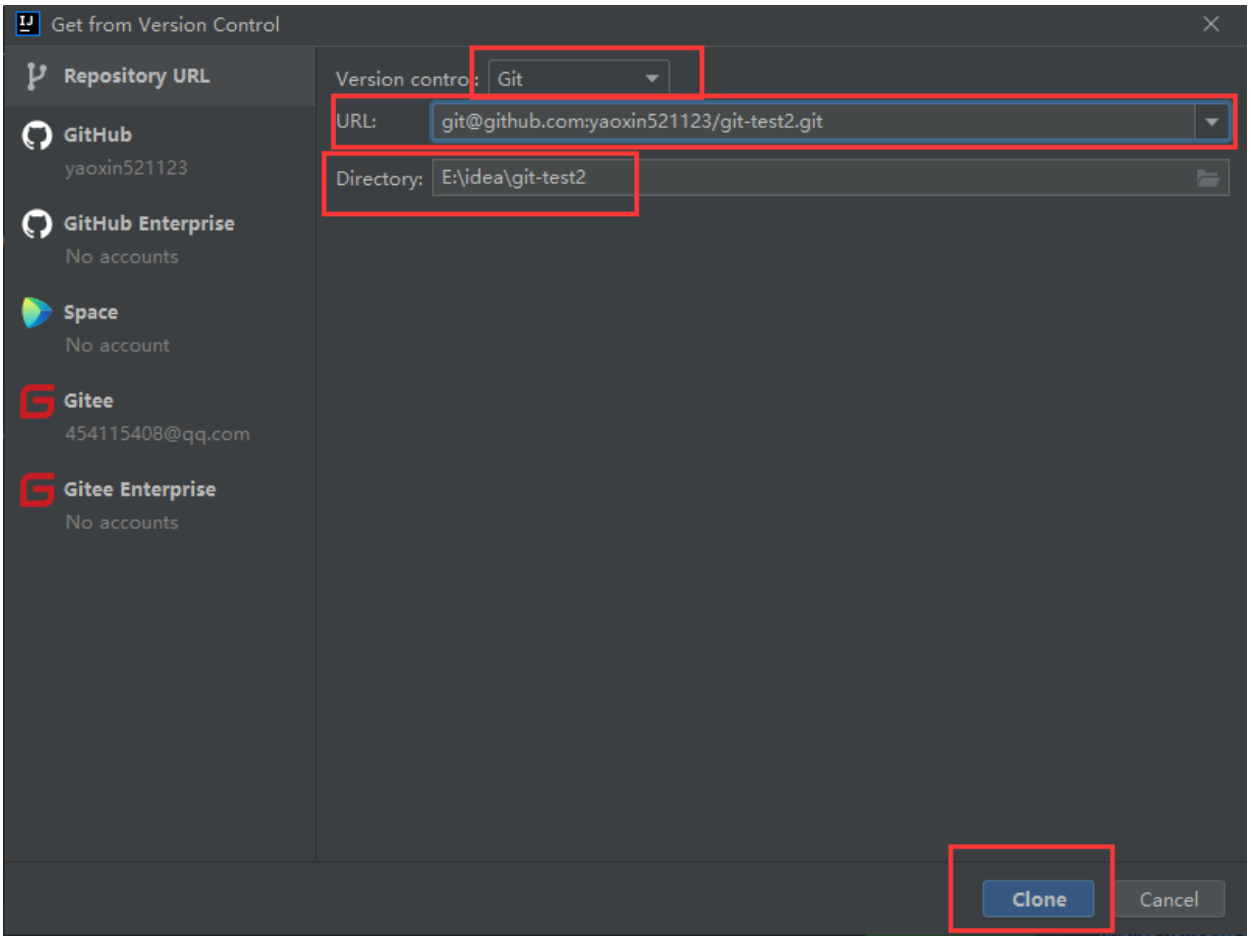
1. File > New > Project From Version Control。



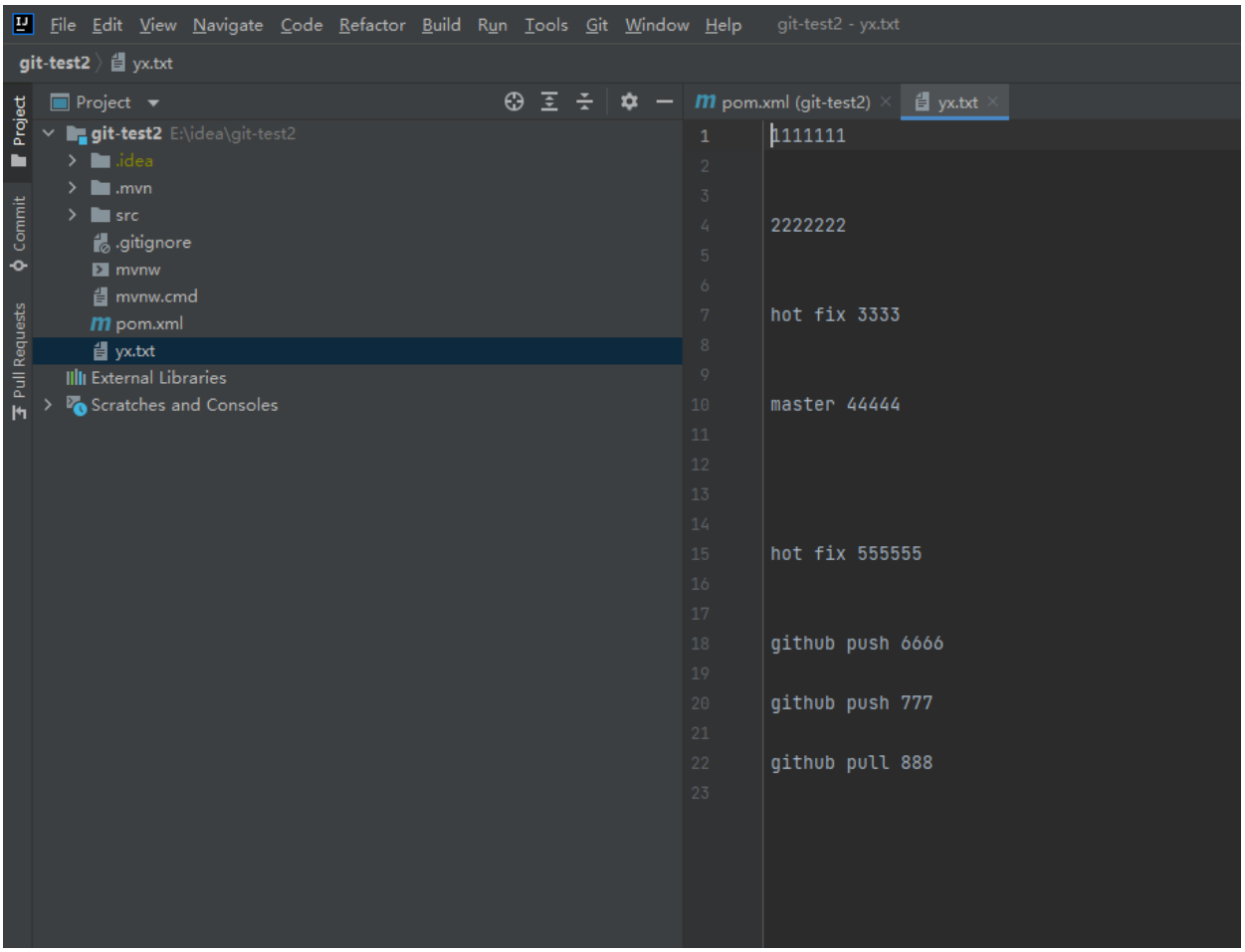
- 删除本地库，在打开IDEA。



2. 输入GitHub远程连接，点击Clone。



3. 克隆成功。


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
git-test2 - yx.txt
git-test2 yx.txt
Project
git-test2 E:\idea\git-test2
.idea
.mvn
src
.gitignore
mvnw
mvnw.cmd
pom.xml
yx.txt
External Libraries
Scratches and Consoles
pom.xml (git-test2)
yx.txt
1111111
2222222
hot fix 3333
master 44444
hot fix 555555
github push 6666
github push 777
github pull 888