**Github基本配置过程与问题解决**

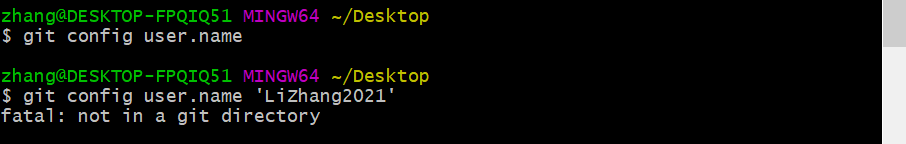
1、在Github上添加SSH密钥

1.1. 配置用户名和邮箱

1. 在开始菜单打开 Git Bash Here
2. 检测下自己之前有没有配置用户名和邮箱, 直接输入以下两个命令：

git config user.name

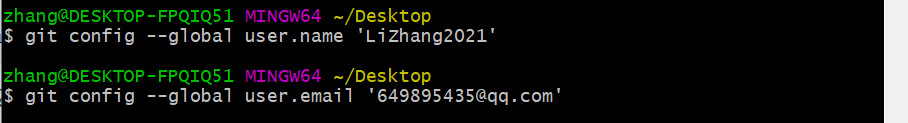
git config user.email



1. 如果之前没有创建，则执行以下命令：

git config –-global user.name "Zhangsan"

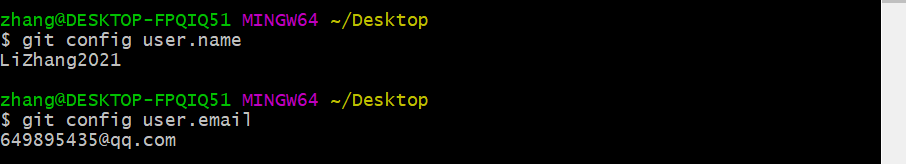
git config –-global user.email "Zhangsan@mail.com"



1. 检查用户名和邮箱是否配置成功输入以下两个命令

git config user.name

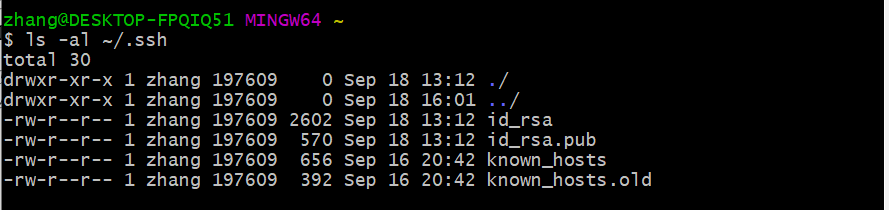
git config user.email



1.2生成SSH密钥对

1. 查看之前是否已经生成了密钥，输入命令

ls -al ~/.ssh



1. 如果出现id\_rsa(私钥)和id\_rsa.pub(公钥)两个文件则证明之前生成过SSH秘钥。删除这两个文件后重新生成SSH 密钥，输入命令：

ssh-keygen -t rsa -C " Zhangsan@mail.com "

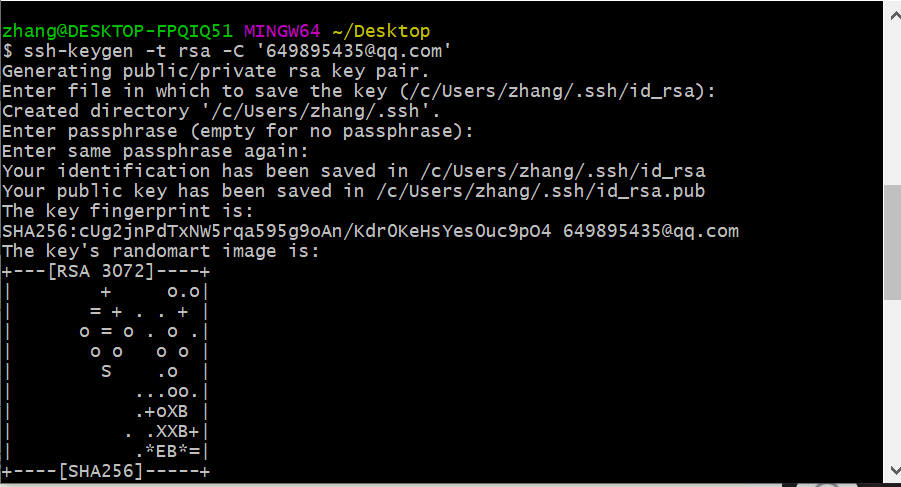
代码参数含义：

-t 指定密钥类型，默认是 rsa ，可以省略。

-C 设置注释文字，比如邮箱。

-f 指定密钥文件存储文件名。

接着按3个回车(也可以根据需要输入)



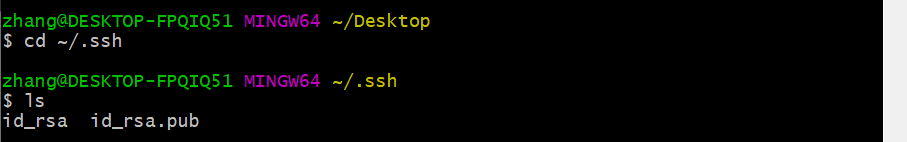
输入密钥锁码，或直接回车

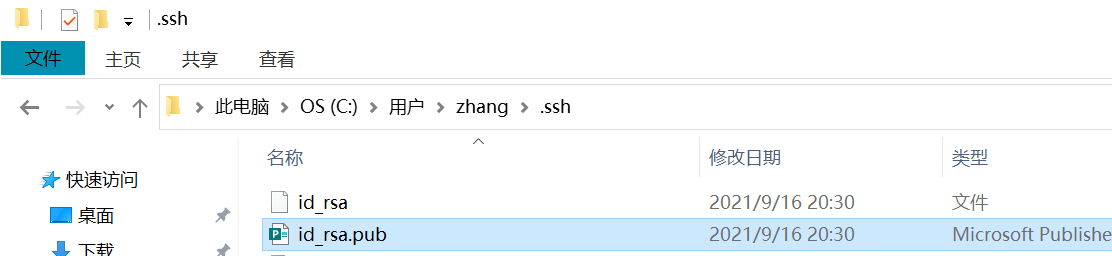
再次输入密钥锁码，或直接回车

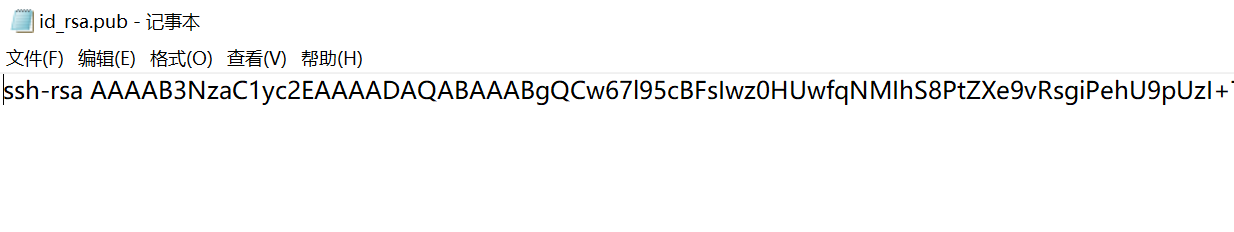
生成私钥

生成公钥

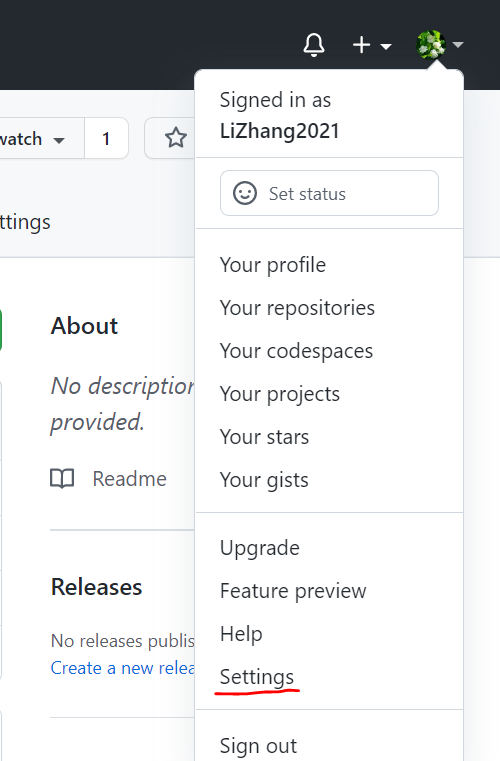
密钥文件默认存放的位置，回车即可

1. 进入.ssh文件中，发现在.ssh目录下(上图中绿色框)得到了两个文件：id\_rsa（私钥）和id\_rsa.pub（公钥），生成SSH密钥对成功
2. 
   1. 添加SSH密钥到Github
3. 首先进入.ssh文件夹中，找到id\_rsa\_pub文件，并复制内容

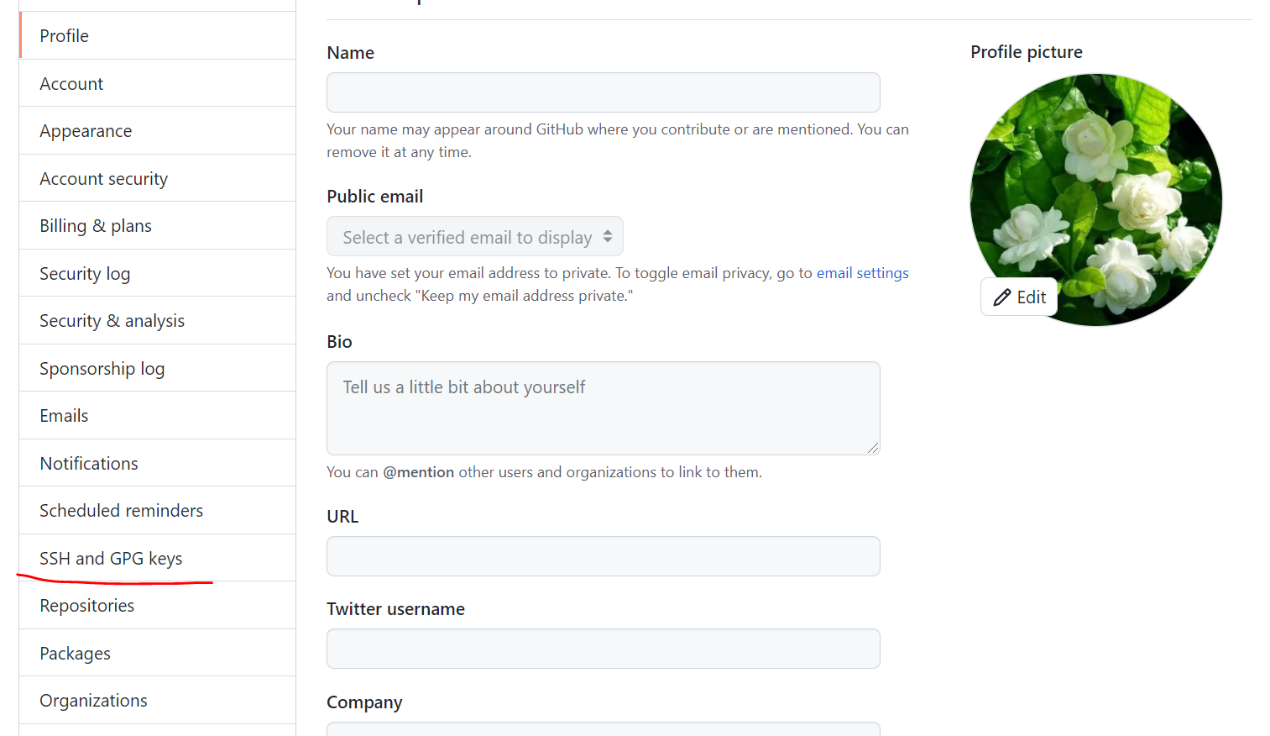




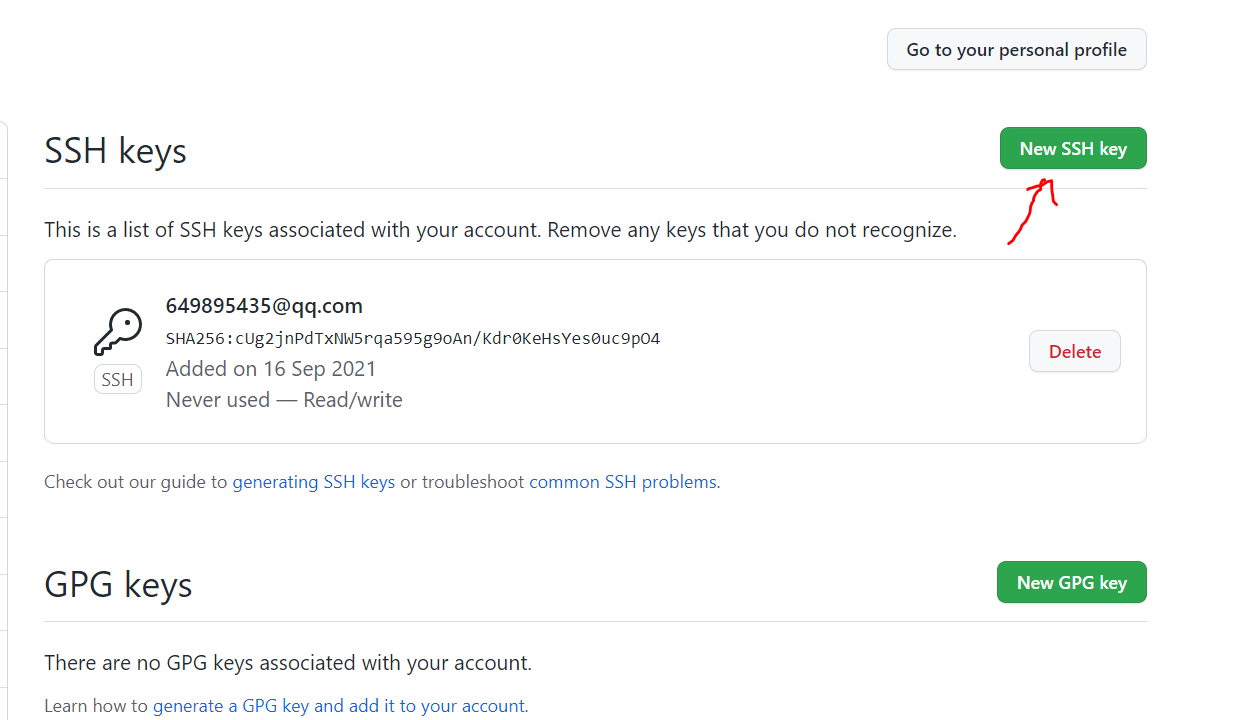
1. 登录Github账户，进入Settings



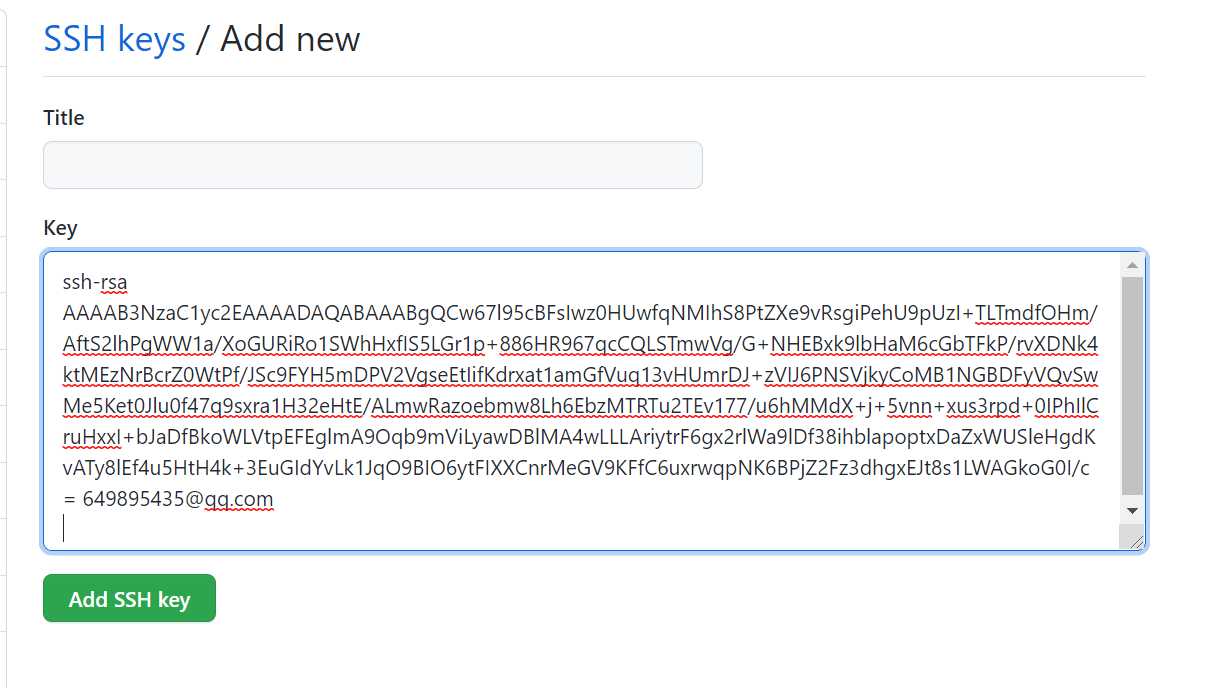
1. 在左边的目录中点击进入SSH and GPG Keys



1. 创建New SSH Key



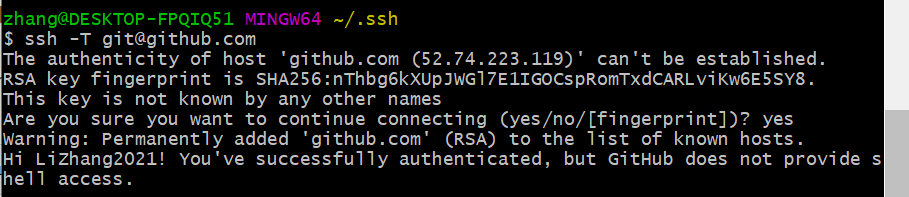
1. 将之前复制的id\_rsa\_pub中的内容粘贴进key中，title可根据需要填写，最后点击Add SSH Key



1.4 将本地计算机与远程Github建立SSH连接

在命令窗口输入，确定连接后，将看到如下信息，就成功了。

ssh -T [git@github.com](mailto:git@github.com)

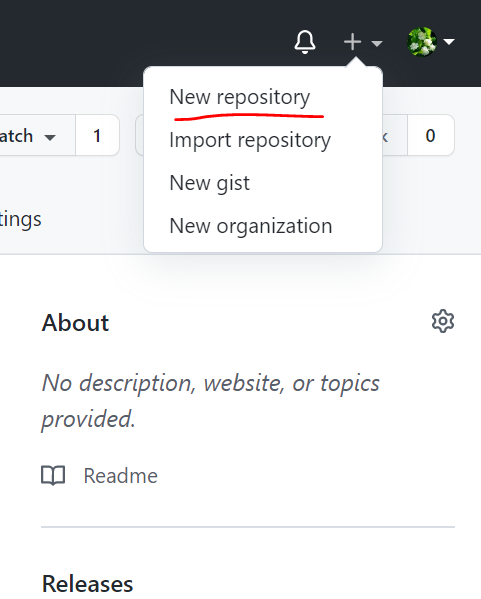


输入 yes

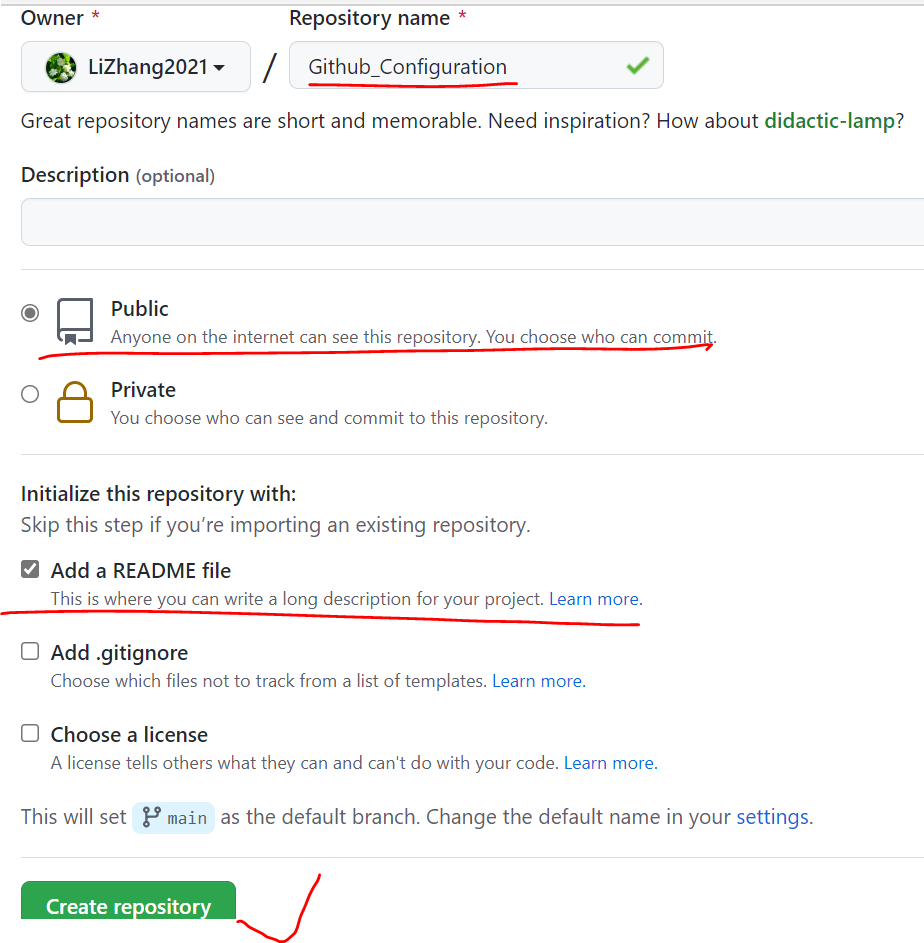
2. 将文件从本地仓库上传到Github仓库中

2.1 在Github上建立一个仓库

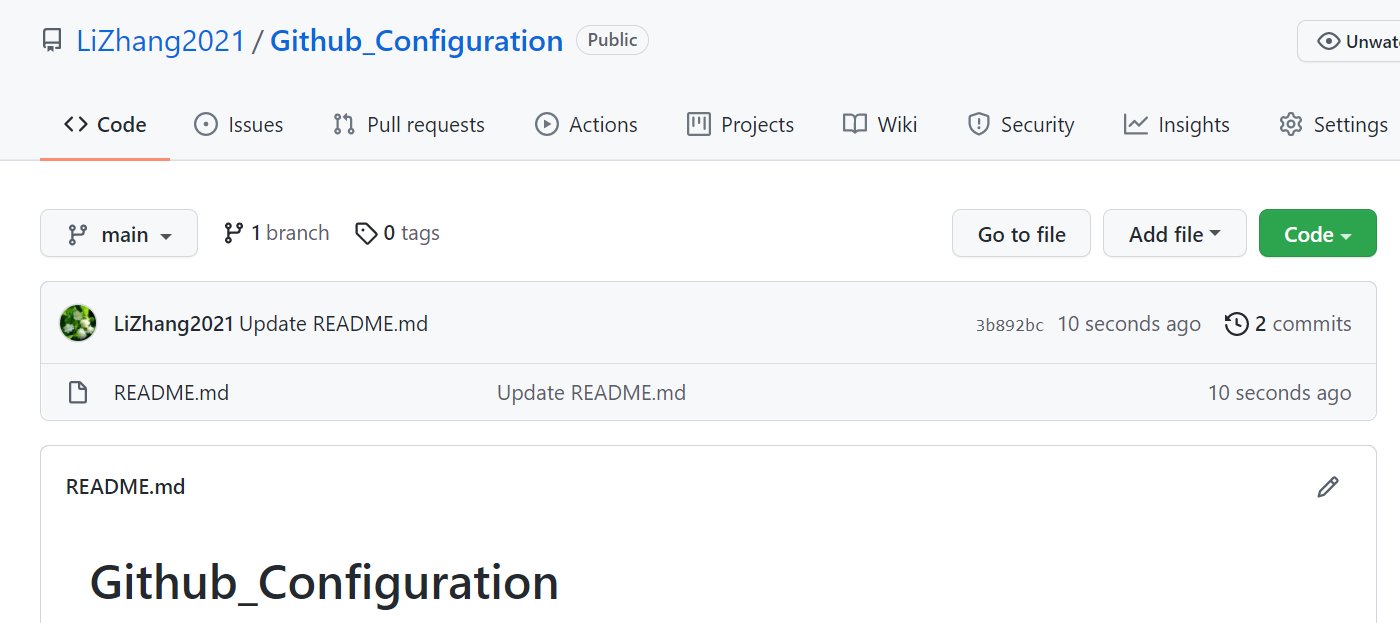
1. 登陆Github，点击右上角的"+"创建一个新的远端仓库



1. 填写好仓库名，并选中添加README.md文件，最后点击创建仓库



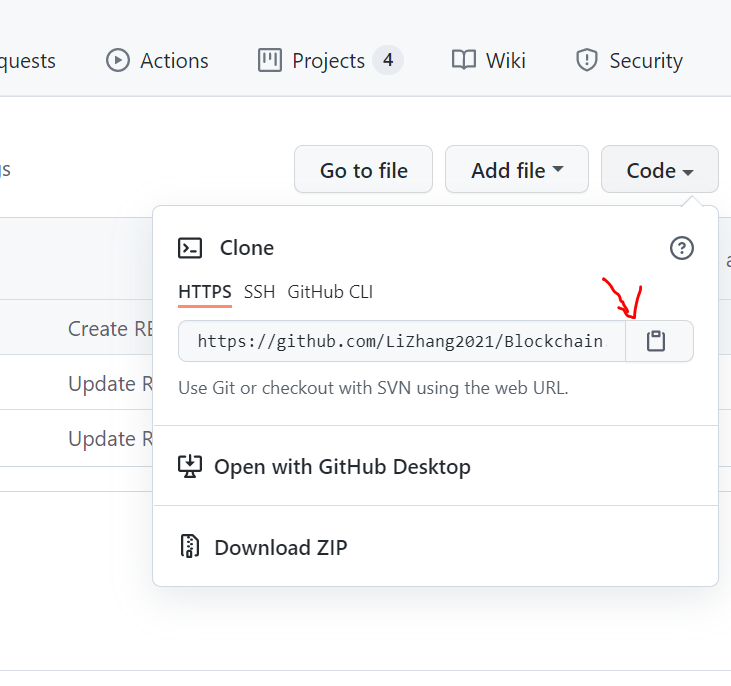
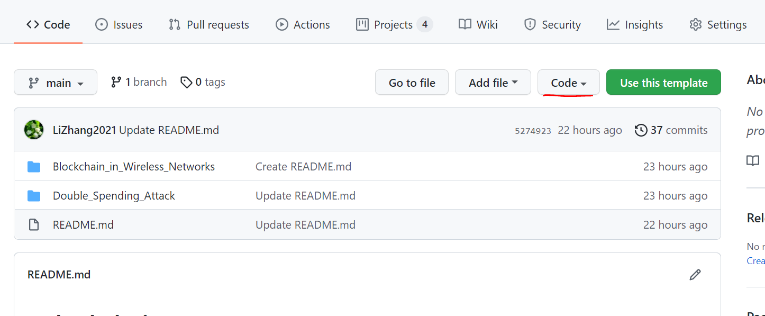
1. 新建仓库成功，如下图所示：



3. 上传本地文件到Github仓库

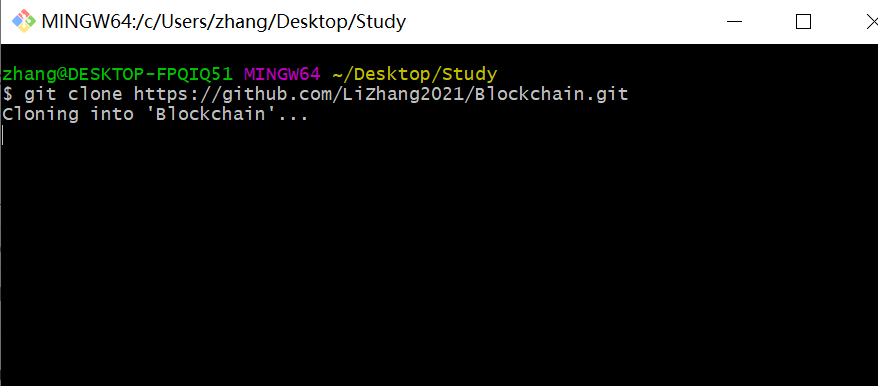
3.1 克隆Github中的仓库到本地

1. 进入Github的仓库获取仓库地址：

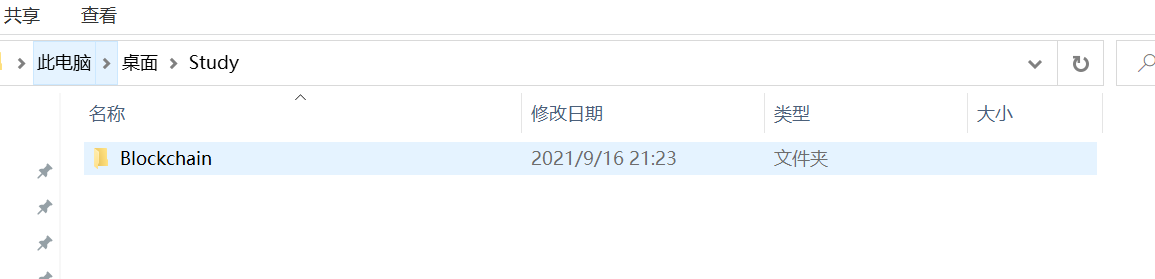


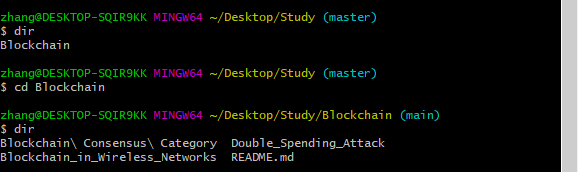
1. 从Github上复制仓库地址，并在终端输入命令克隆下来一个仓库

git clone 仓库的地址



1. 克隆成功后进入仓库，可以看到仓库下的文件





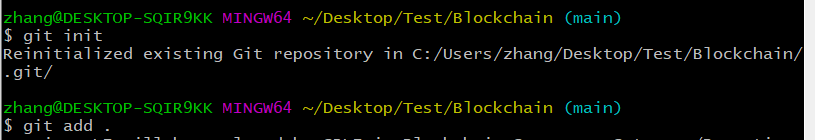
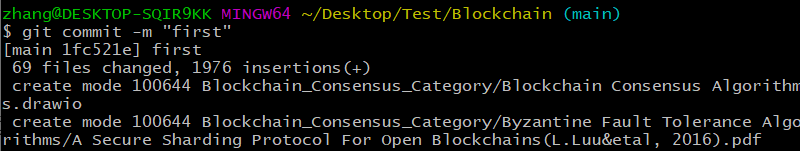
3.2 上传/更新本地仓库中的文件到Github仓库

1. 初始化仓库，输入以下命令

git init

git add .

git commit -m "文件说明"

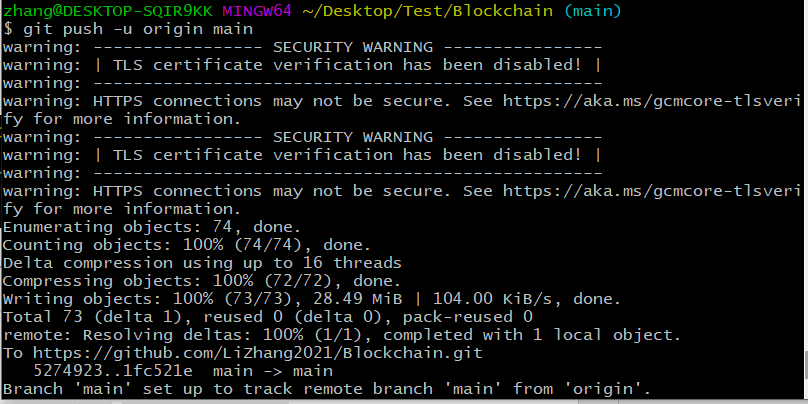
1. 在准备好提交文件后，我们将本地仓库与Github仓库建立连接，输入命令

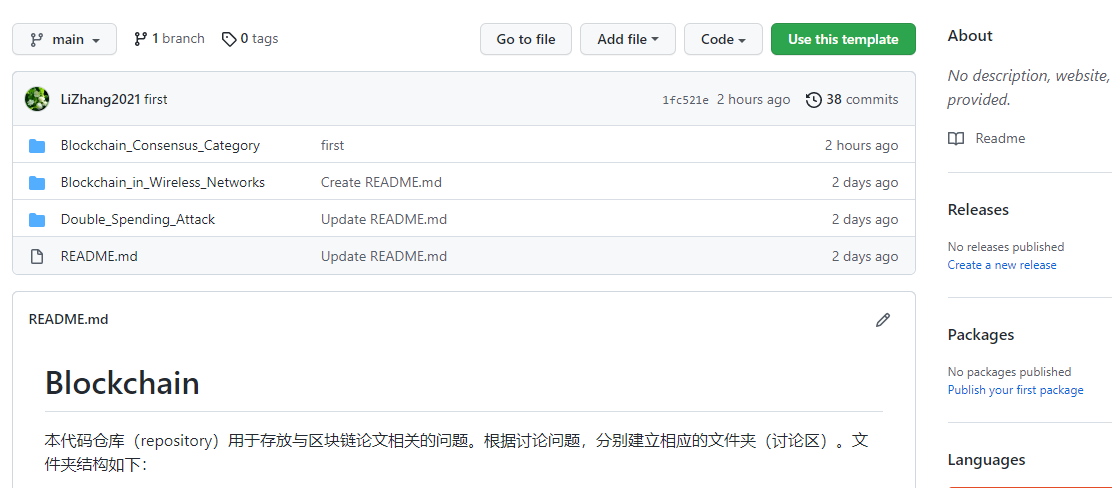
git remote add origin Git仓库地址



1. 我们就将文件push到Git仓库中，输入命令后，文件会上传到远端Github仓库

git push -u origin main



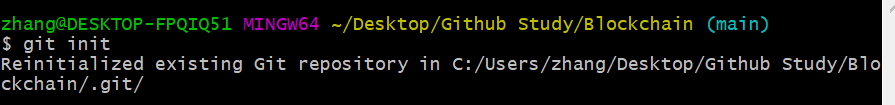


4. 更新本地仓库

4.1 本地仓库更新最新的Github仓库

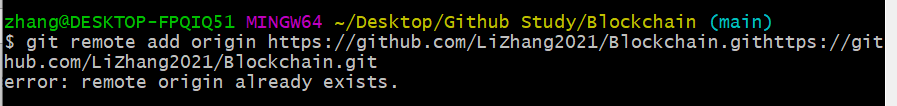
1. 初始化本地仓库，输入命令

git init



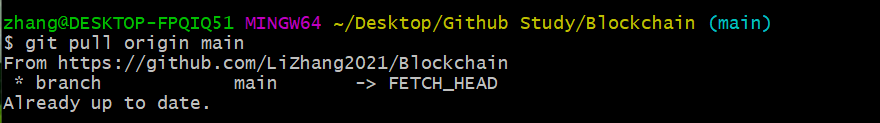
1. 将本地仓库关联到Github仓库

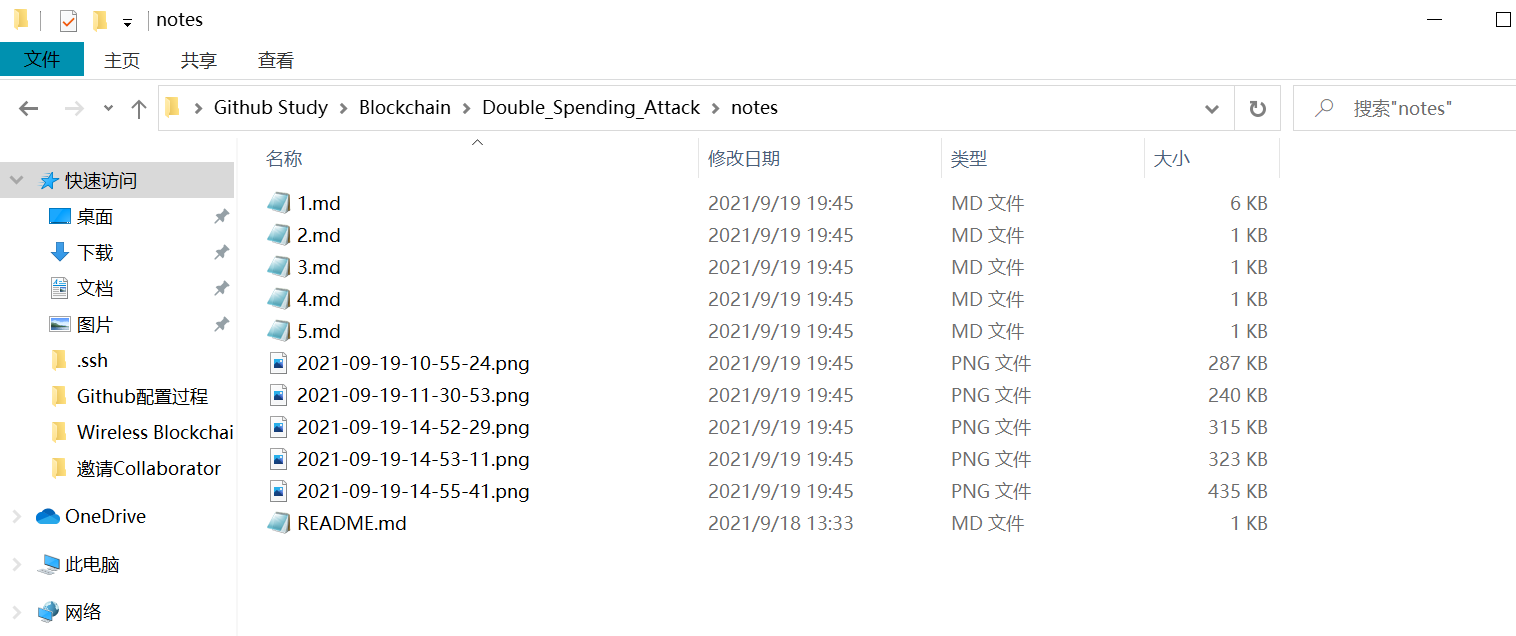
git remote add origin 仓库地址



1. 将Github仓库中更新后的文件拉到本地仓库中

git pull (git pull origin main)



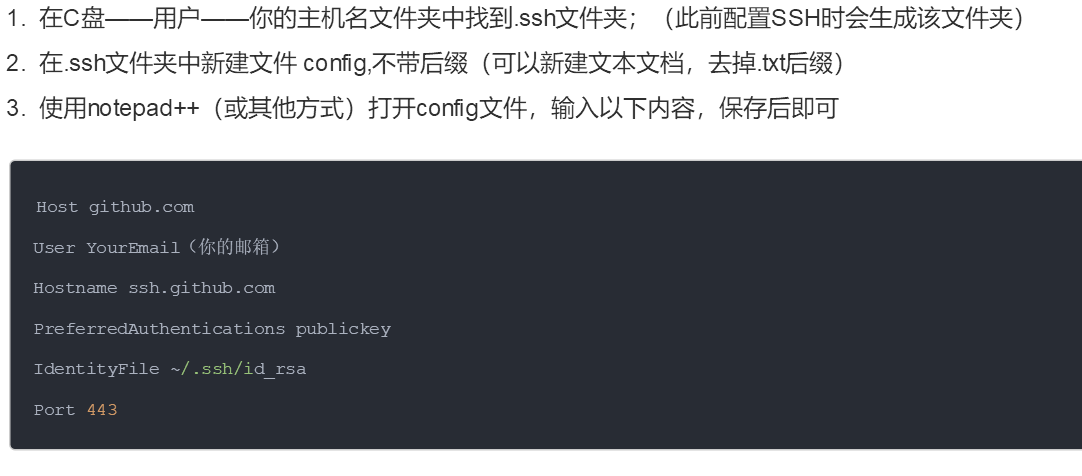


5. 解决问题

在配置过程中出现各种不同的问题，导致不能访问远端Github仓库，下面就给出总结。

5.1本地与Github无法建立SSH连接错误

1. 出现" ssh: connect to host github.com port 22: Connection timed out"错误时，通过下图方式解决：



1. 随后成功建立连接



5.2 Github报错10054和443

1. 当在克隆或上传文档出现10054和443故障时，通常时因为电脑代理的原因，要么就是服务器的SSL证书没有经过第三方机构的签署，所以报错。通过以下方法解决：

方法一：取消代理命令

git config --global --unset http.proxy

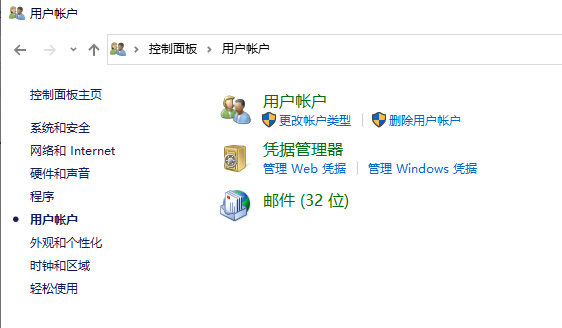
git config --global --unset https.proxy

方法二：解除SSL验证

git config --global http.sslVerify "false"

1. 如果通过上述方法还不能解决，则需要查看是否缺少Github的Windows凭据，如果缺少则需要手动添加。进入控制面板->用户账户->凭据管理器->Windows凭据->删除与自己Github账号不匹配的凭据/添加自己Github账号的凭据。之后会让我们输入新的github用户名和密码

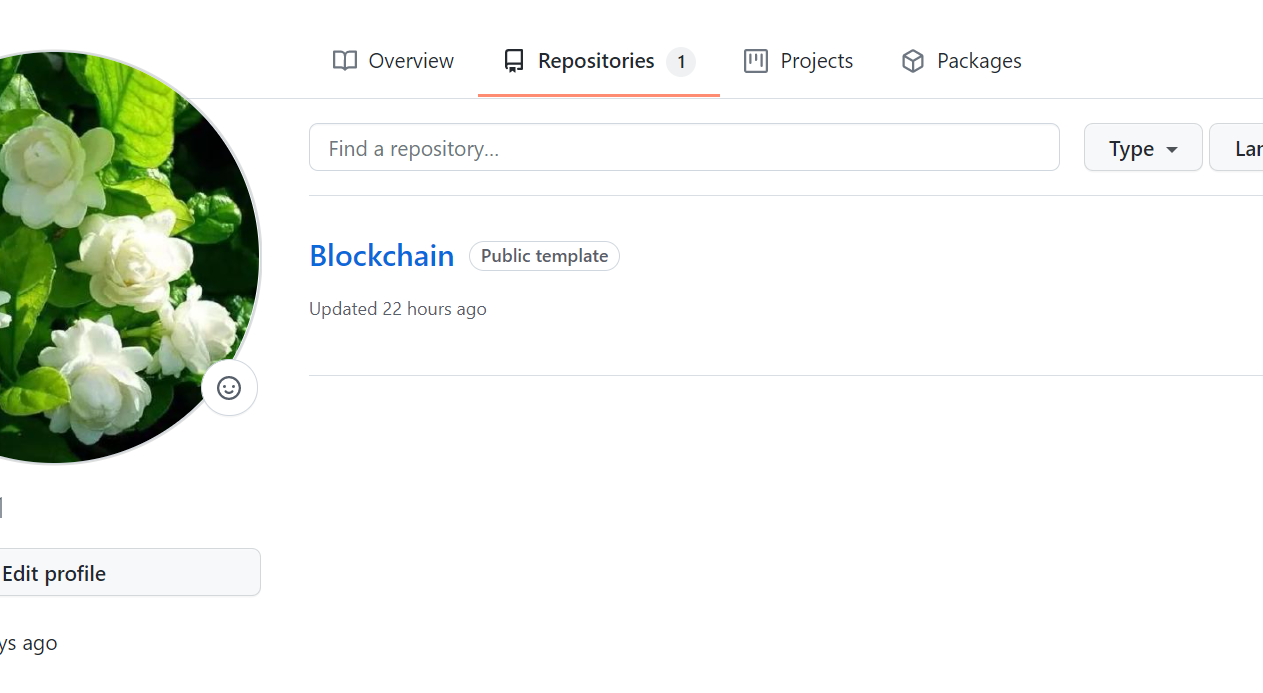




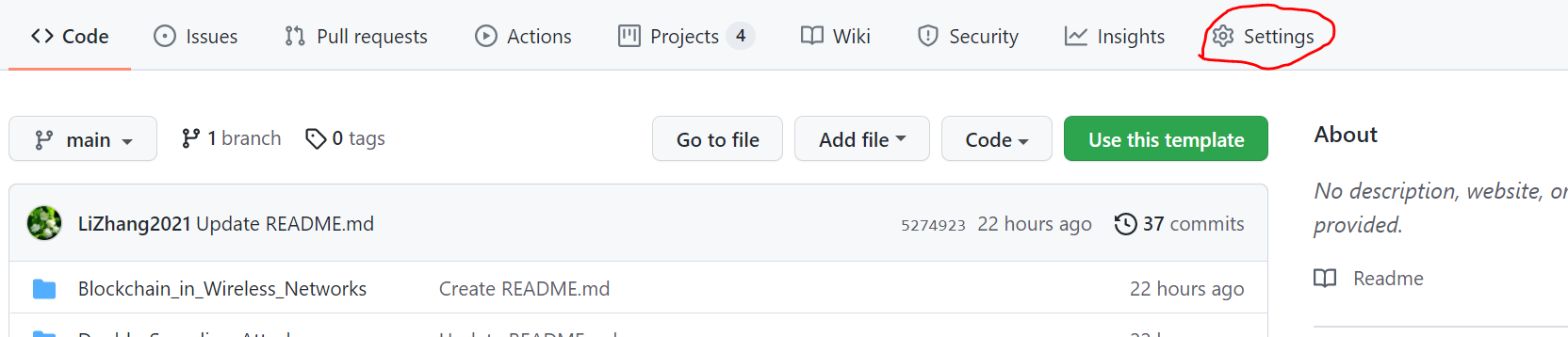


6. 添加仓库合作者

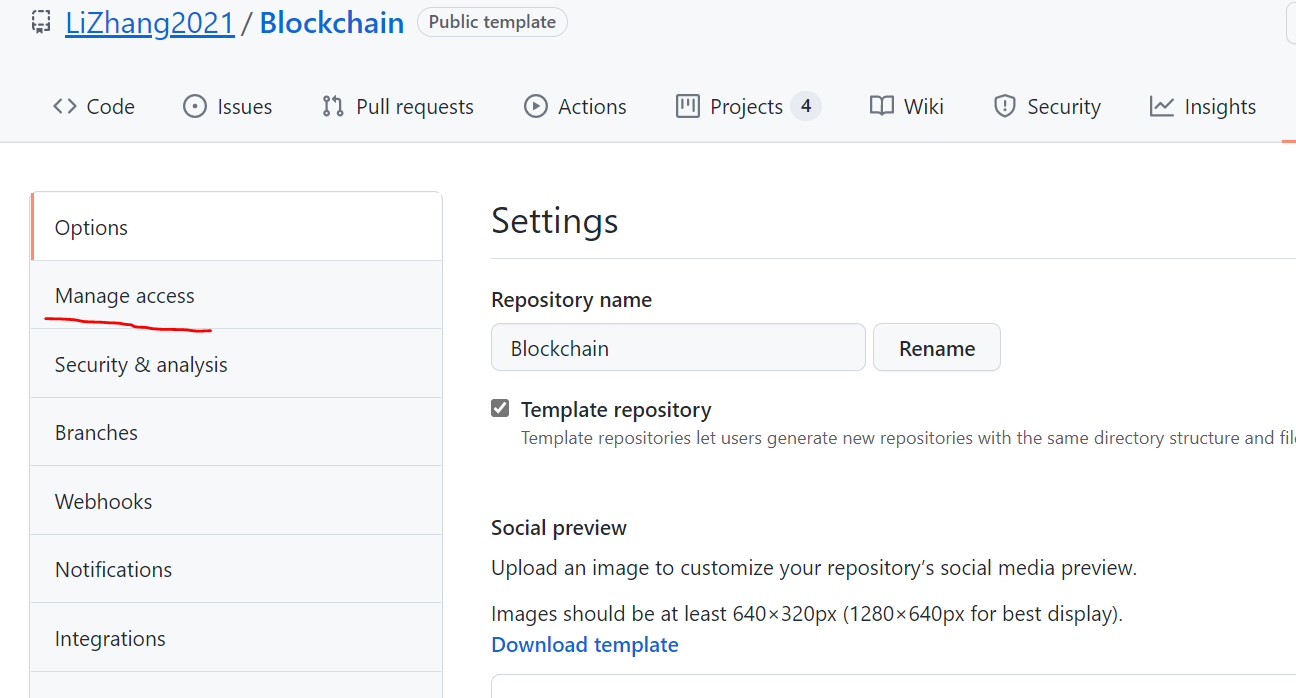
1. 首先进入要与他人共同协作管理的项目



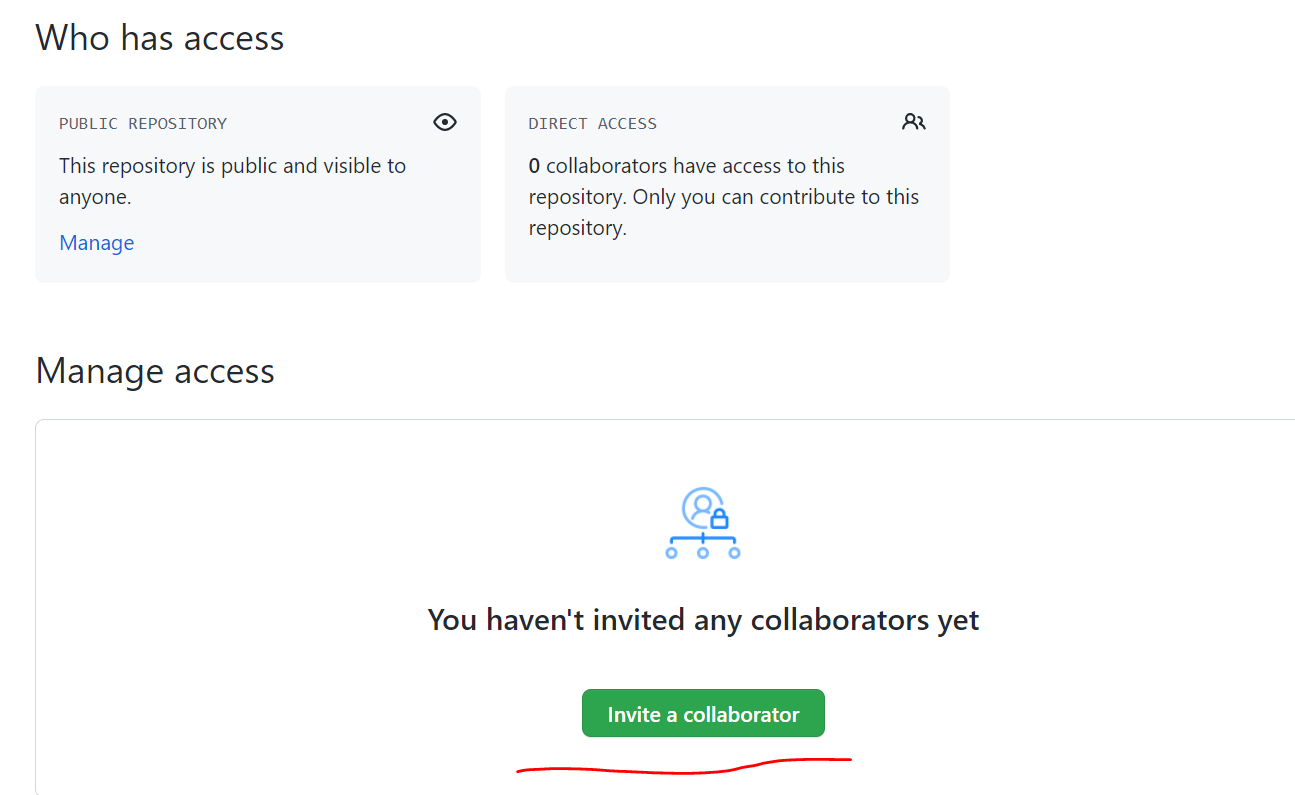
1. 随后进入仓库的设置



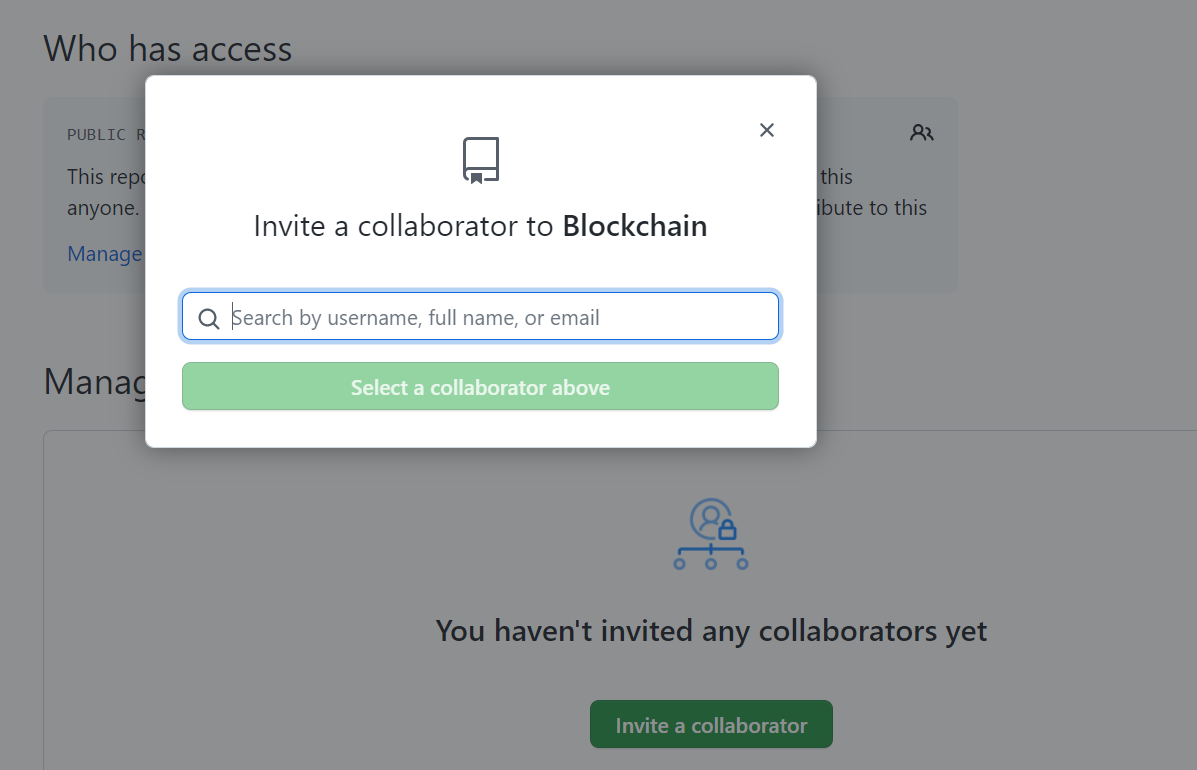
1. 在左边的目录找到"Manage access"，点击进入

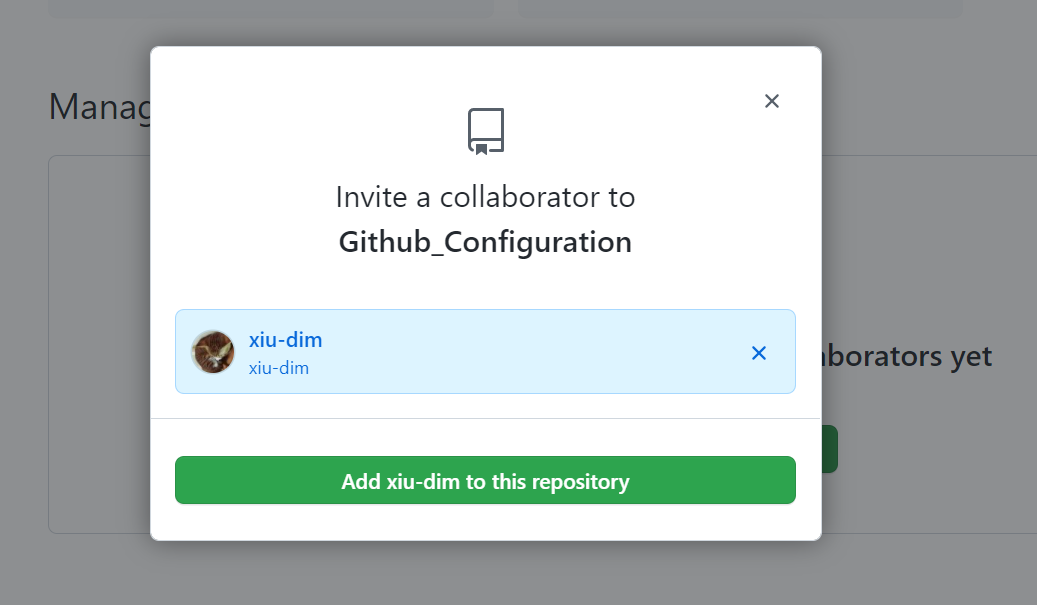


1. 拉到最下面，有一个邀请合作者的绿色按钮"Invite a collaborator"



1. 输入合作者的用户名或者邮箱后，点击添加





1. 最终合作者添加成功

