Git的使用：

-1:git与svn的区别：<https://www.cnblogs.com/Sungeek/p/9152223.html> 好

0：可参考该站：<https://blog.csdn.net/buknow/article/details/80325986> <https://blog.csdn.net/u011535541/article/details/83379151>这个更详细，更好

1：下载git软件 https://git-scm.com/download

2：安装git软件 下一步…安装完成

3：注册githup账户：<https://github.com/>

用户名：wxg-git

密码：r..x..\_099

[git@github.com:wxg-git/SPBoot201912.git](mailto:git@github.com:wxg-git/SPBoot201912.git)

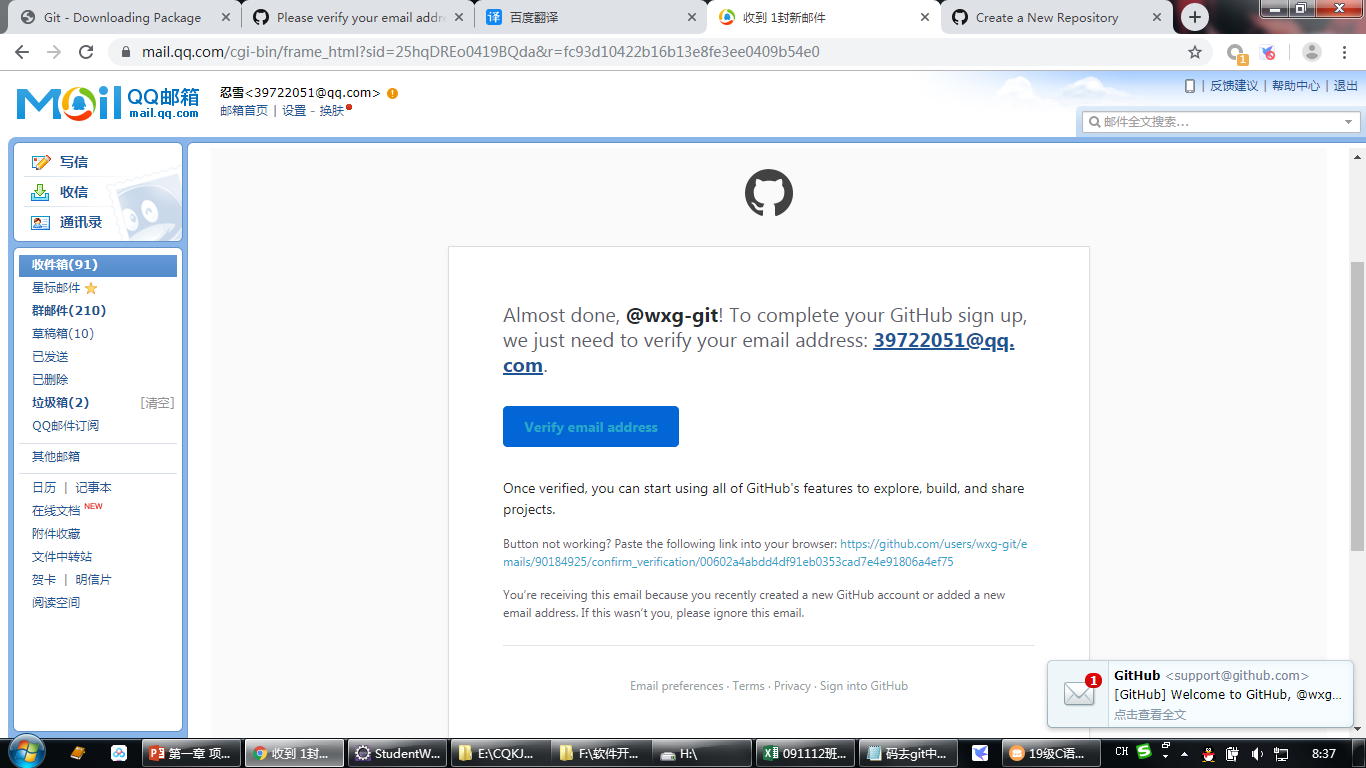
5: 多账户配置（多个repository）

<https://blog.csdn.net/Sophie_U/article/details/85259549>

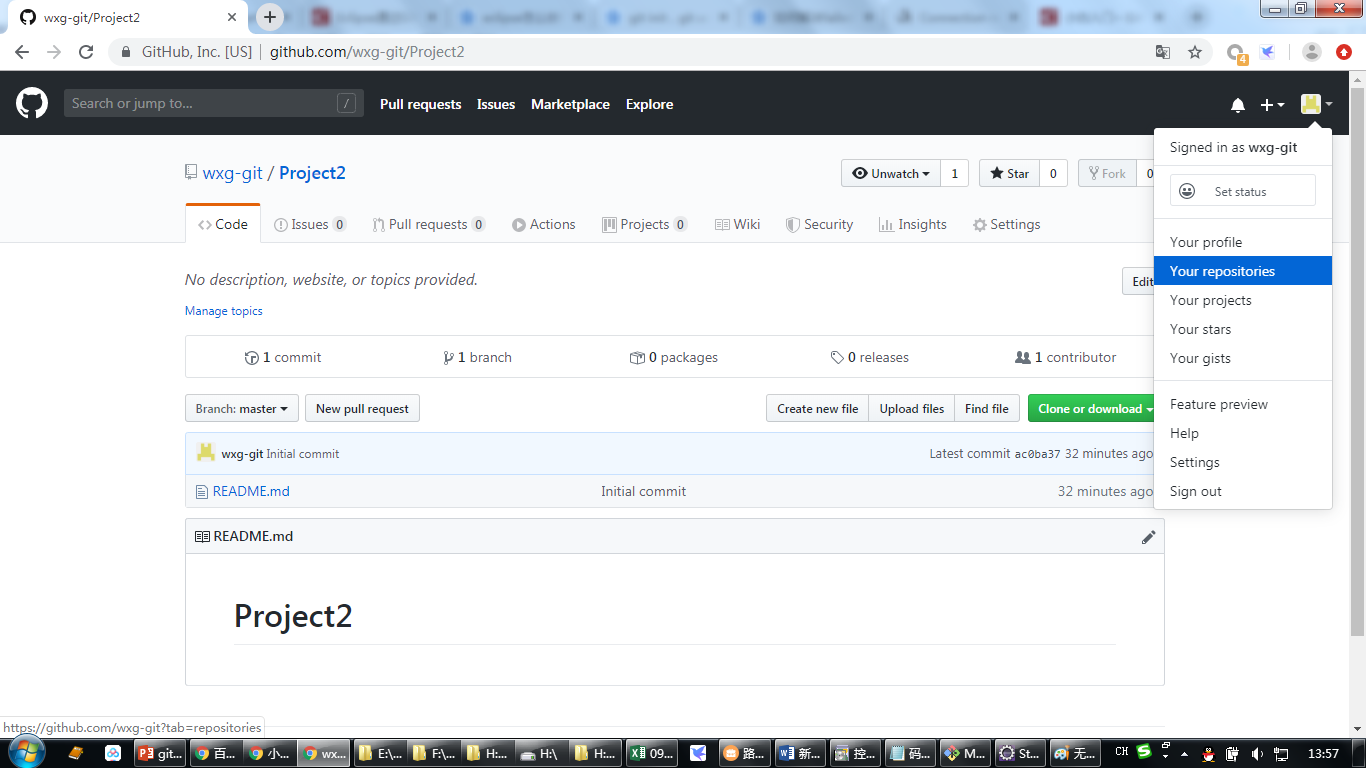
<https://www.cnblogs.com/logchen/p/10543808.html>

<https://www.gwygd.com/NRqz3APx6.html>

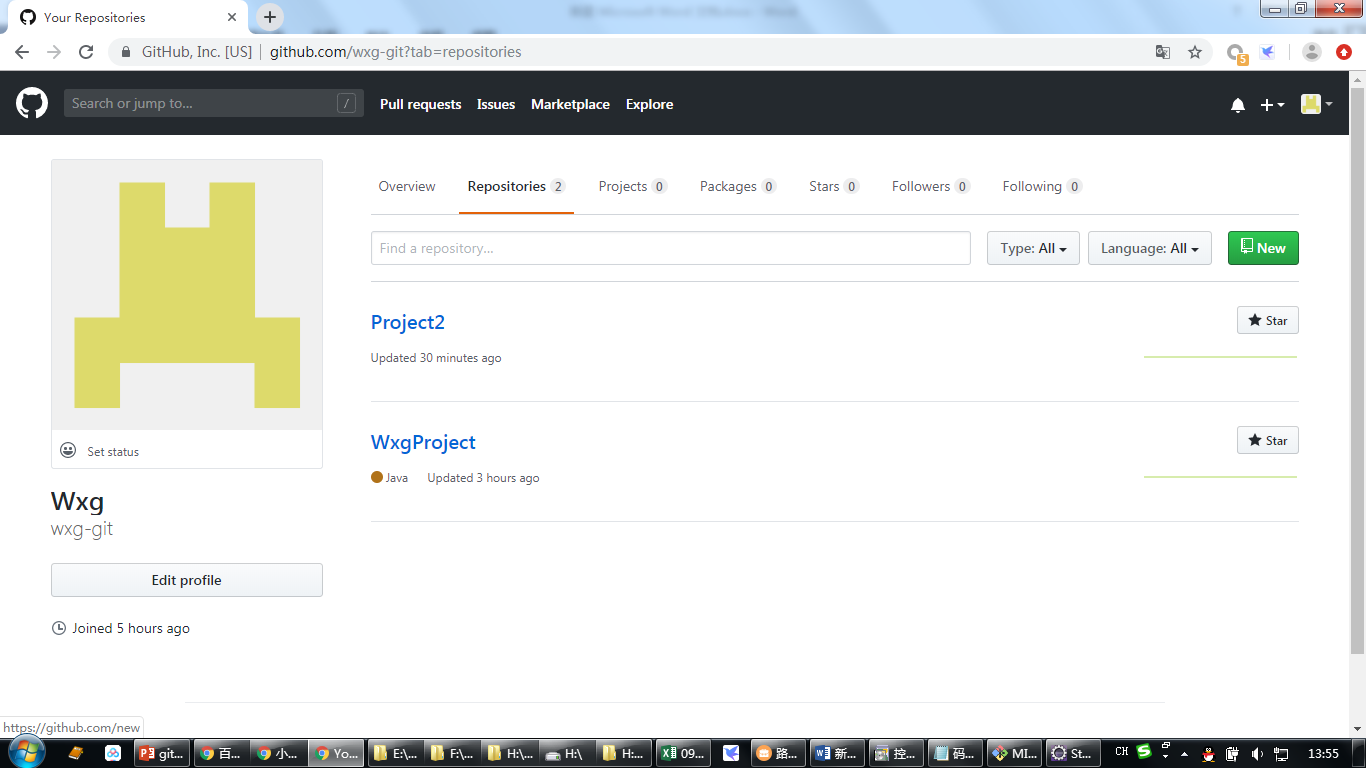
注：需要邮箱验证



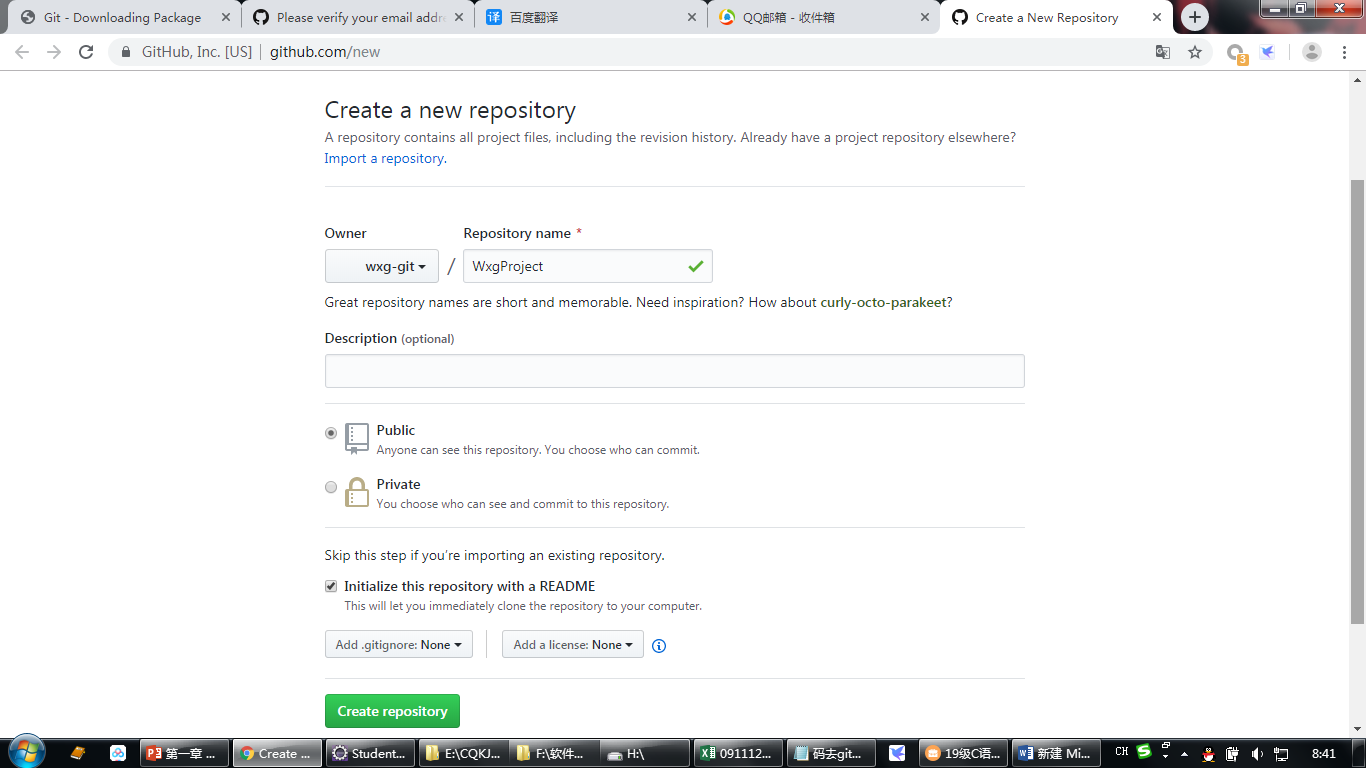
4:配置中央仓储（免费使用githup官网的空间）



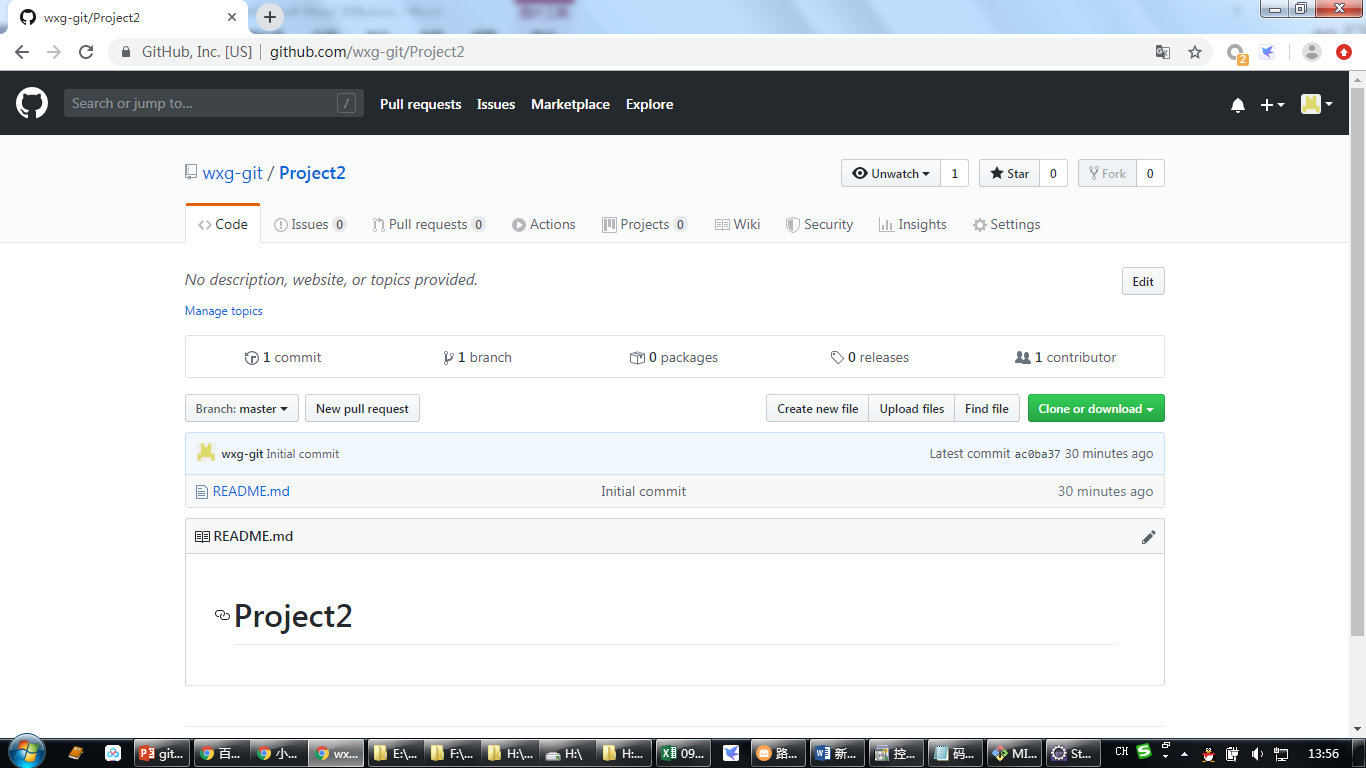
点击new



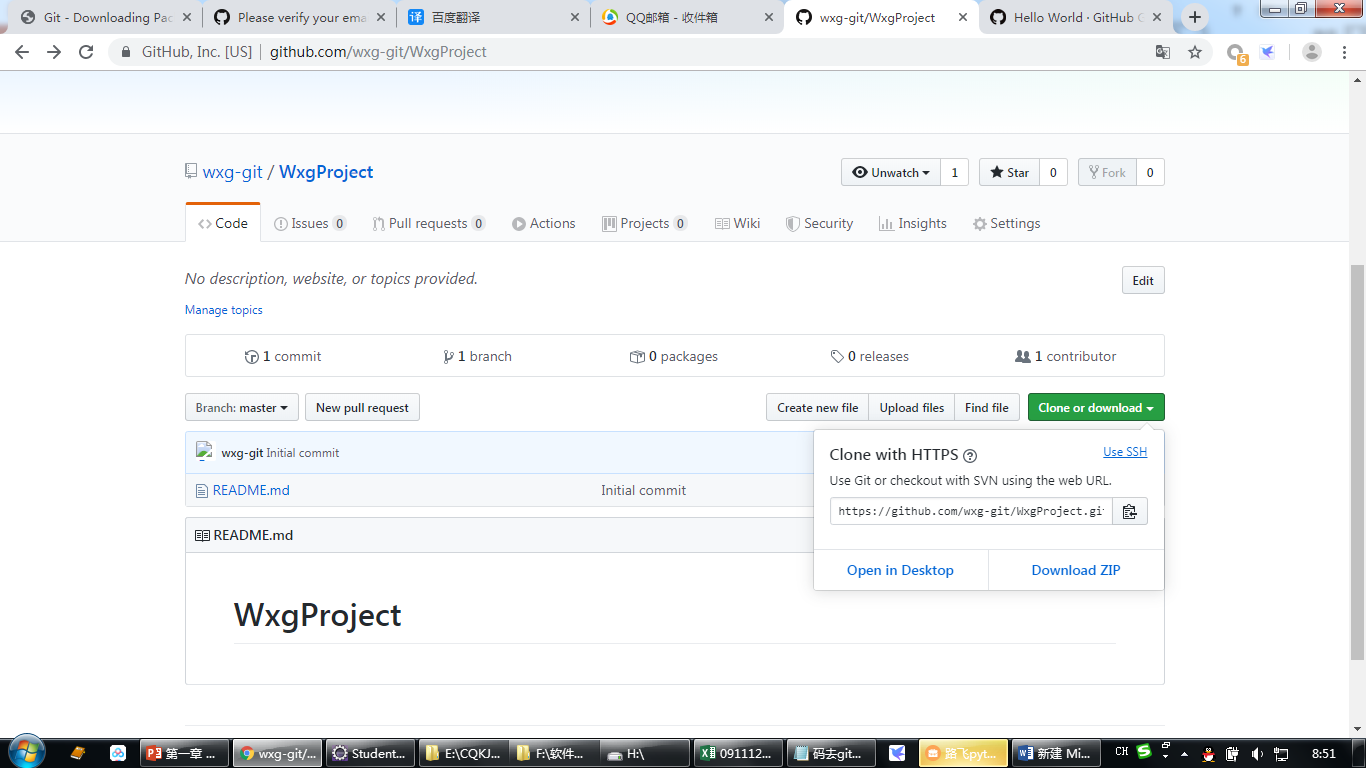
输入仓储名字，勾选initialize….



点击仓储(显示对应仓储相关内容)

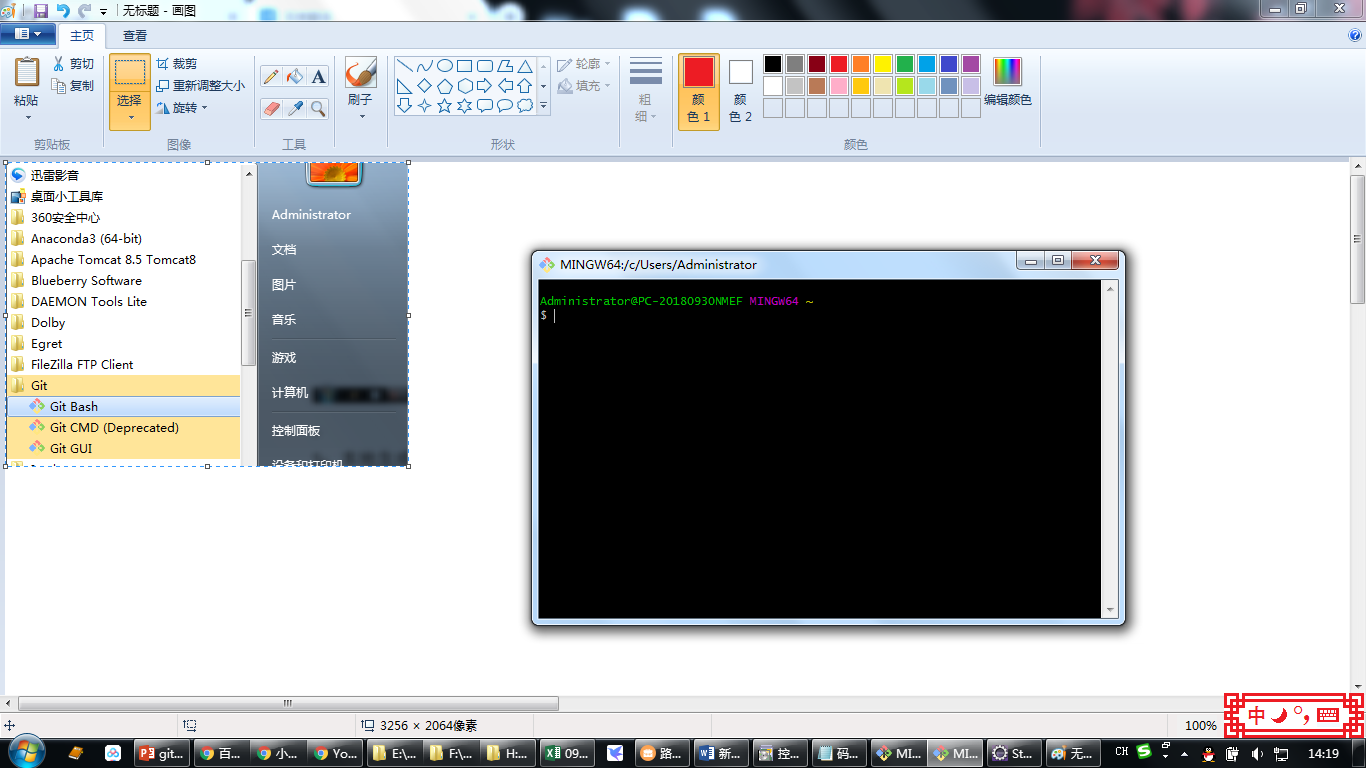


Download: 可以拿到仓储的地址(ssh或http)



5：本地生成ssh密匙， 配置到githup账号中

5.1: 安装完git后，打开Git Blush



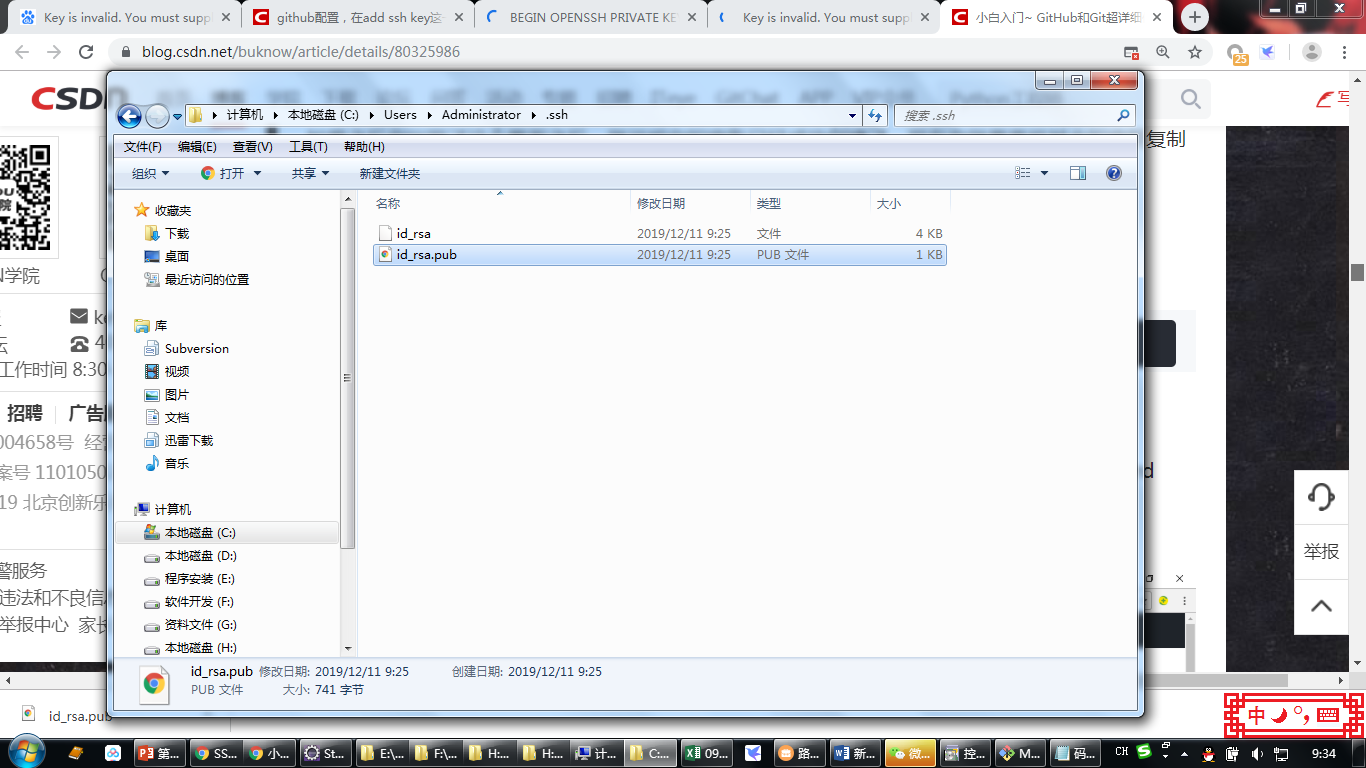
5.2:创建ssh key

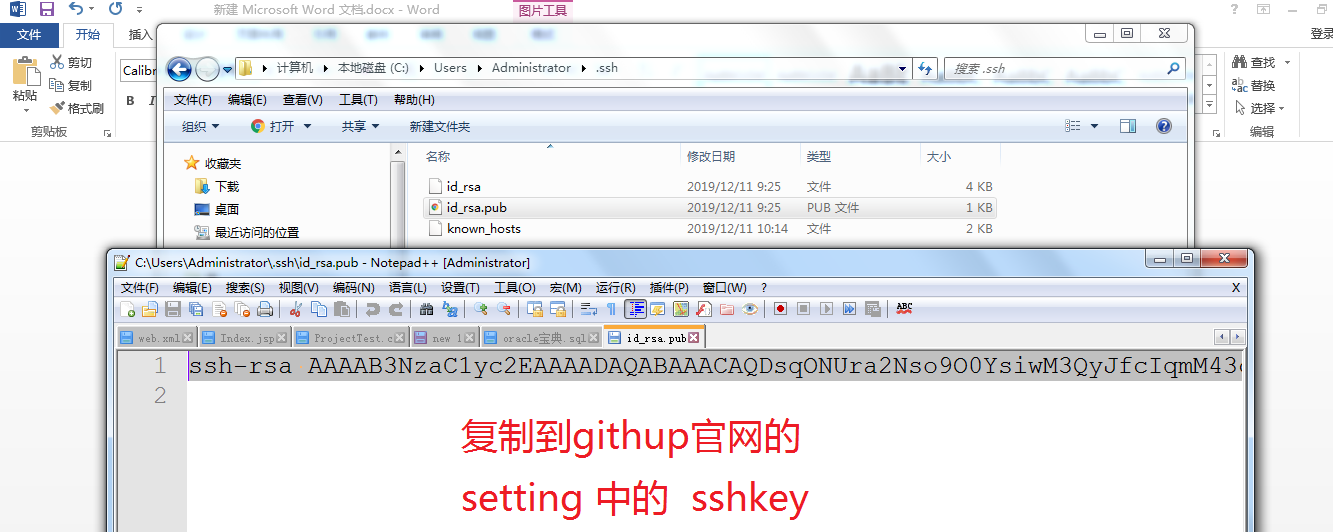
输入：$ ssh-keygen -t rsa -b 4096 -C [39722051@qq.com](mailto:39722051@qq.com)

$ ssh-keygen -t rsa -C "39722051@qq.com"

邮件地址：写自己注册githup的地址

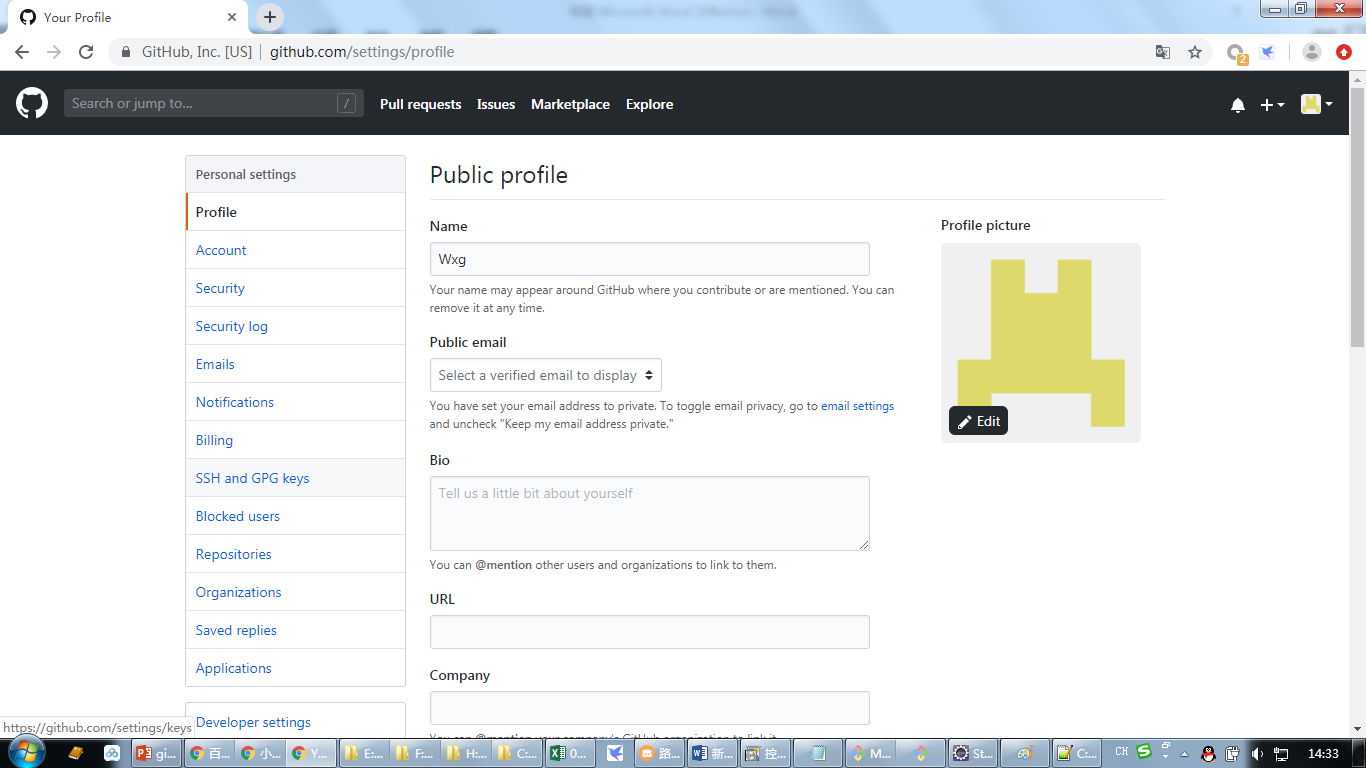
询问你Enter a file in which to save the key (/c/Users/you/.ssh/id\_rsa):[Press enter]直接按回车使用默认地址存放ssh key，也可以自己输入地址。



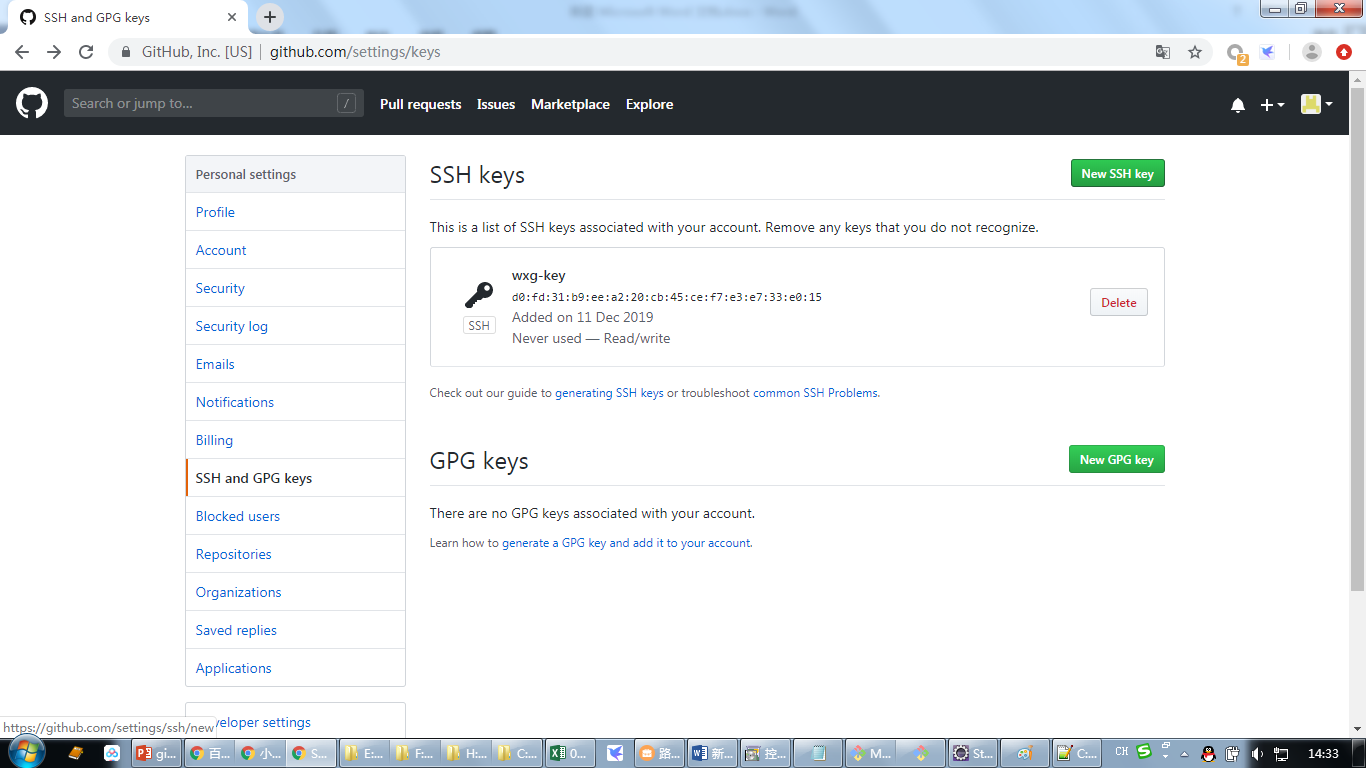




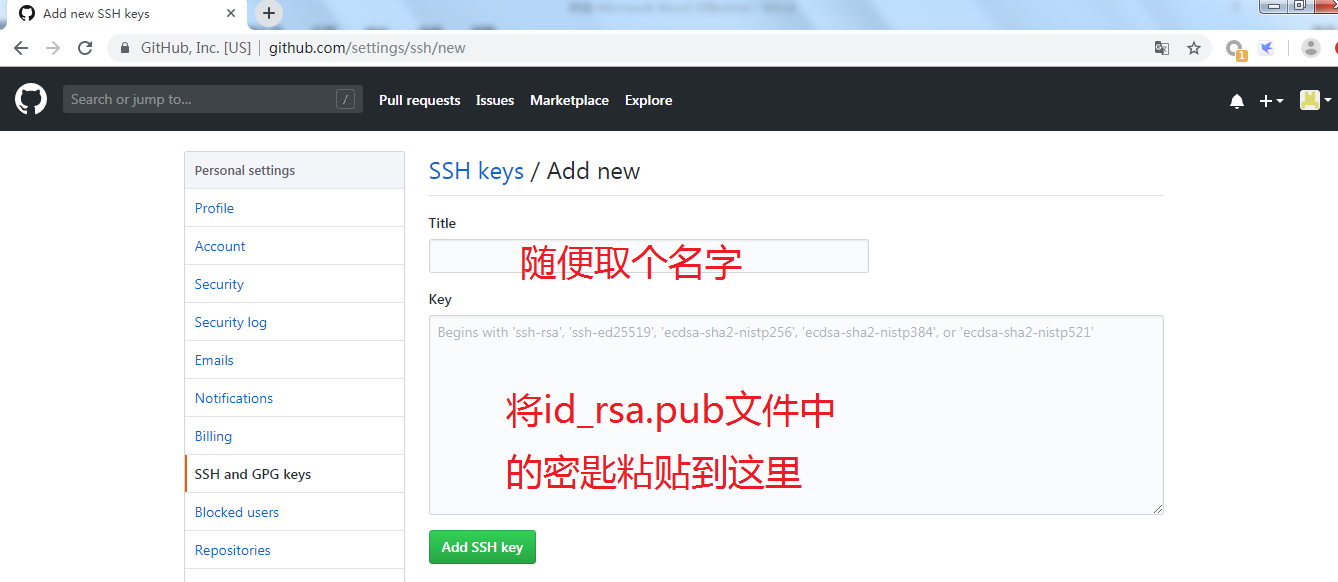
点击：settings



点击：SSH and GPG keys



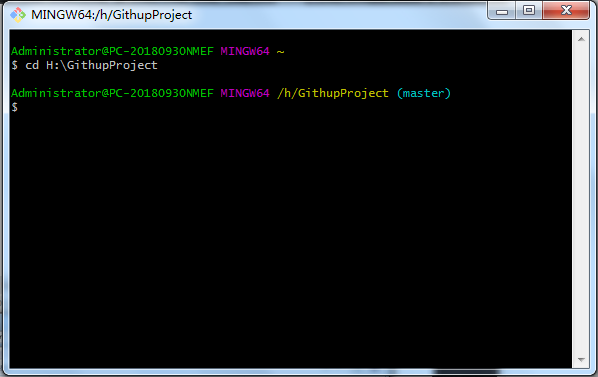
点击 New SSHkey



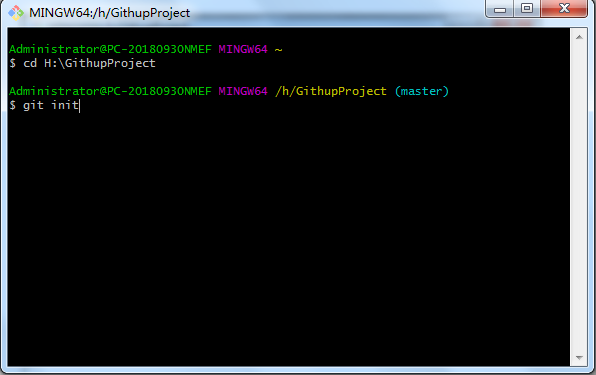
搞定！

5.3: 定位到一个项目目录： cd 项目目录

如：cd H:\GithupProject



5.4:首先初始化git ，使用git先要在本地创建一个项目，用于存放代码



输入后：会在当前文件夹生成一个.git隐藏文件



5.5：测试一下本地仓储与远程仓储是否能建立通讯

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

如果出现了：You've successfully authenticated, but GitHub does not provide shell access 。那就说明，已经成功

5.6:重点来了，自报家门

$ git config --global user.name "wxg-git"

$ git config --global user.email "[39722051@qq.com](mailto:39722051@qq.com)"

输入上边的代码，name最好和GitHub上边的一样，这里的your\_email@youremail.com要写成你自己github上绑定的邮箱

这两个的顺序可以颠倒，没有固定的顺序。

通过该命令可以查看刚才配置的信息：git config --global  --list

6：接下来可以克隆代码或上传代码了。

6.1：上传代码：

将项目代码放入：.git文件夹中。 上面用的是：H:\GithupProject

1. 上传代码之前：关联一下远程仓储

**git remote add 远程仓库名 git@github.com:用户名/仓库名.git**

**git remote add origin** [**git@github.com:wxg-git/WxgProject.git**](mailto:git@github.com:wxg-git/WxgProject.git)

**(一般 远程仓库名使用 origin)**

**git remote add origin git@github.com:wxg-git/TestGit.git**

**git remote add origin git@github.com:wxg-git/SPBoot201912.git**

另：**git remote -v //查看关联信息**

**git remote rm 远程库的名字 //解除关联**

1. 输入命令(添加到暂存区)：

git add 文件名 或 git add . (.表示所有文件)

1. 输入命令（将暂存区文件提交到本地仓储）：

git commit –m ‘注释

1. 输入命令(将本地仓储文件推送到中央服务区)

git push origin master

注：如果以上命令有问题，则通过：

-- 通过以下代码把仓储的readme.md拉到本地

git pull --rebase origin master

6.2：克隆远程仓储代码

**git clone git@github.com:用户名/仓库名.git**

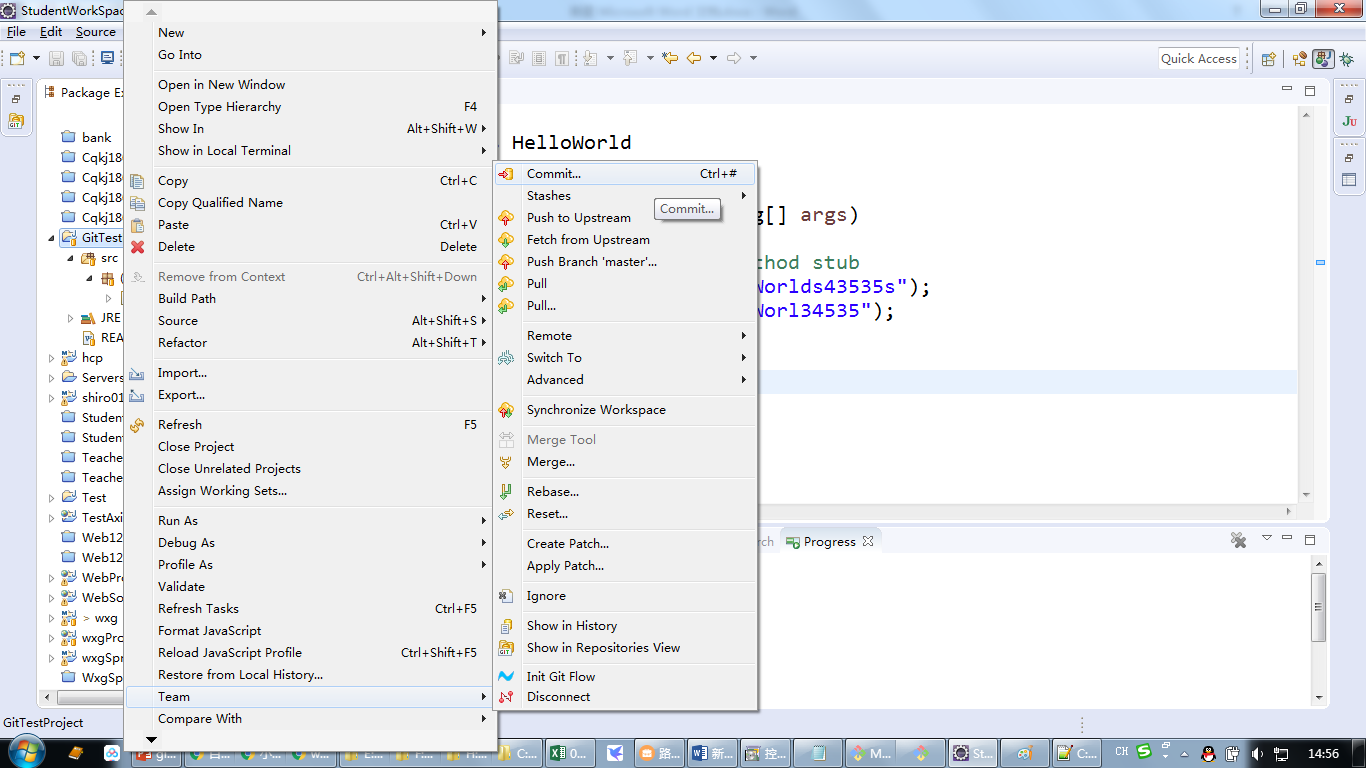
7:Eclipse使用git

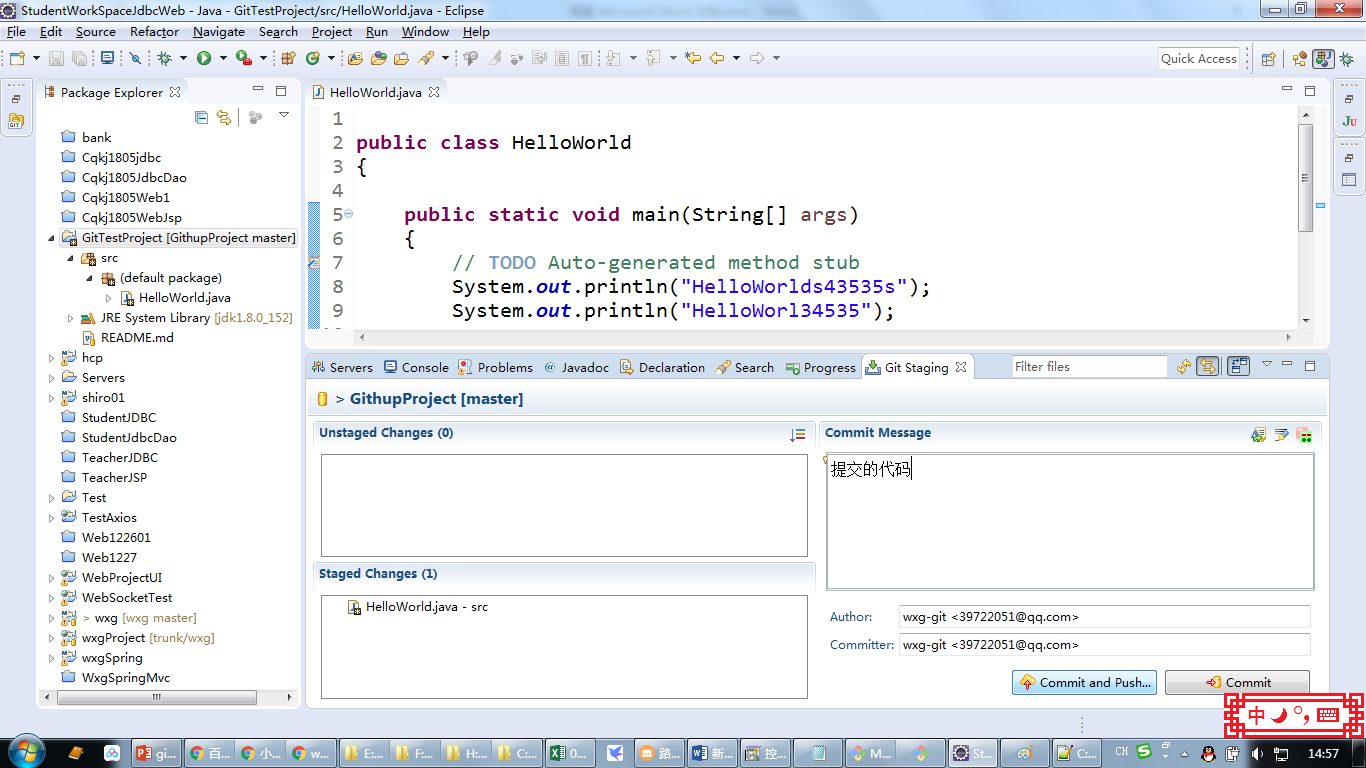
安装git插件

首先打开Eclipse，然后点击Help>Install New Software>Add

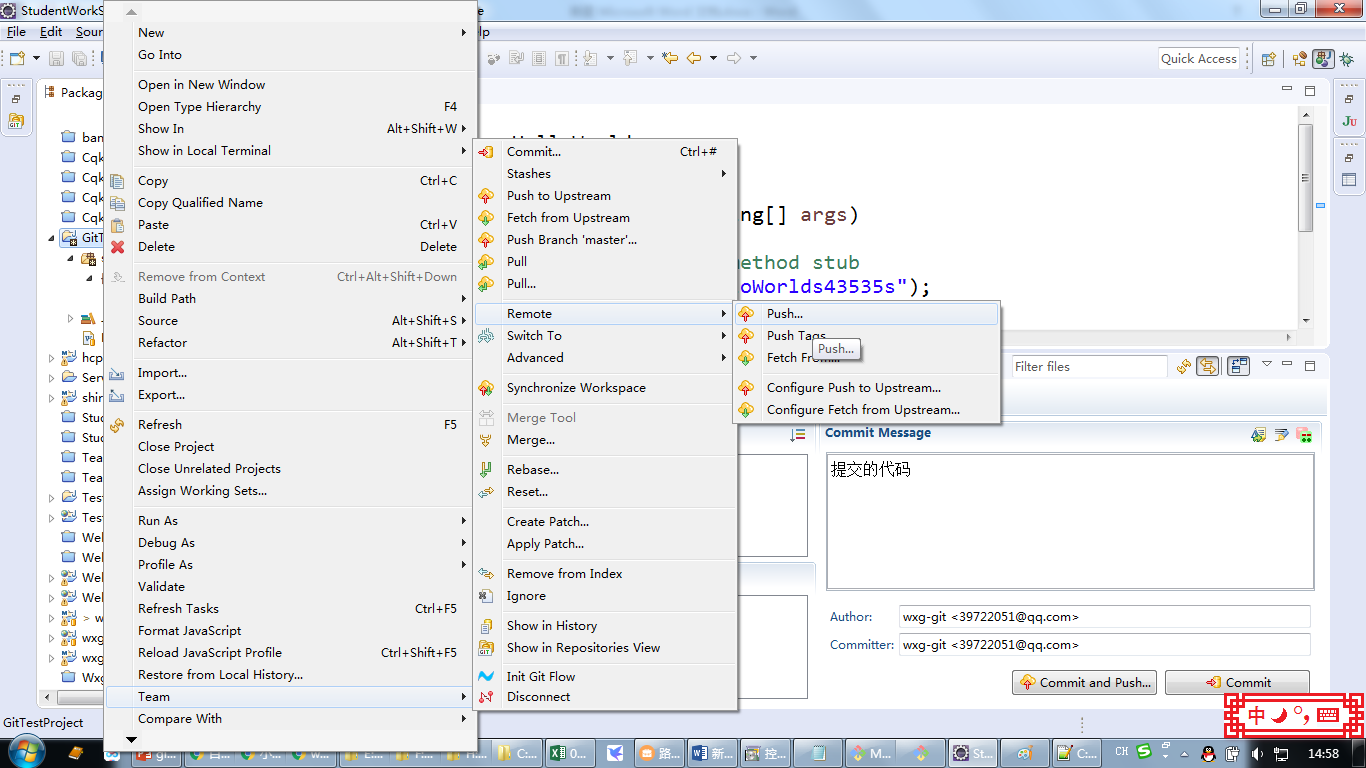
Name：EGit

Location： http://download.eclipse.org/egit/updates

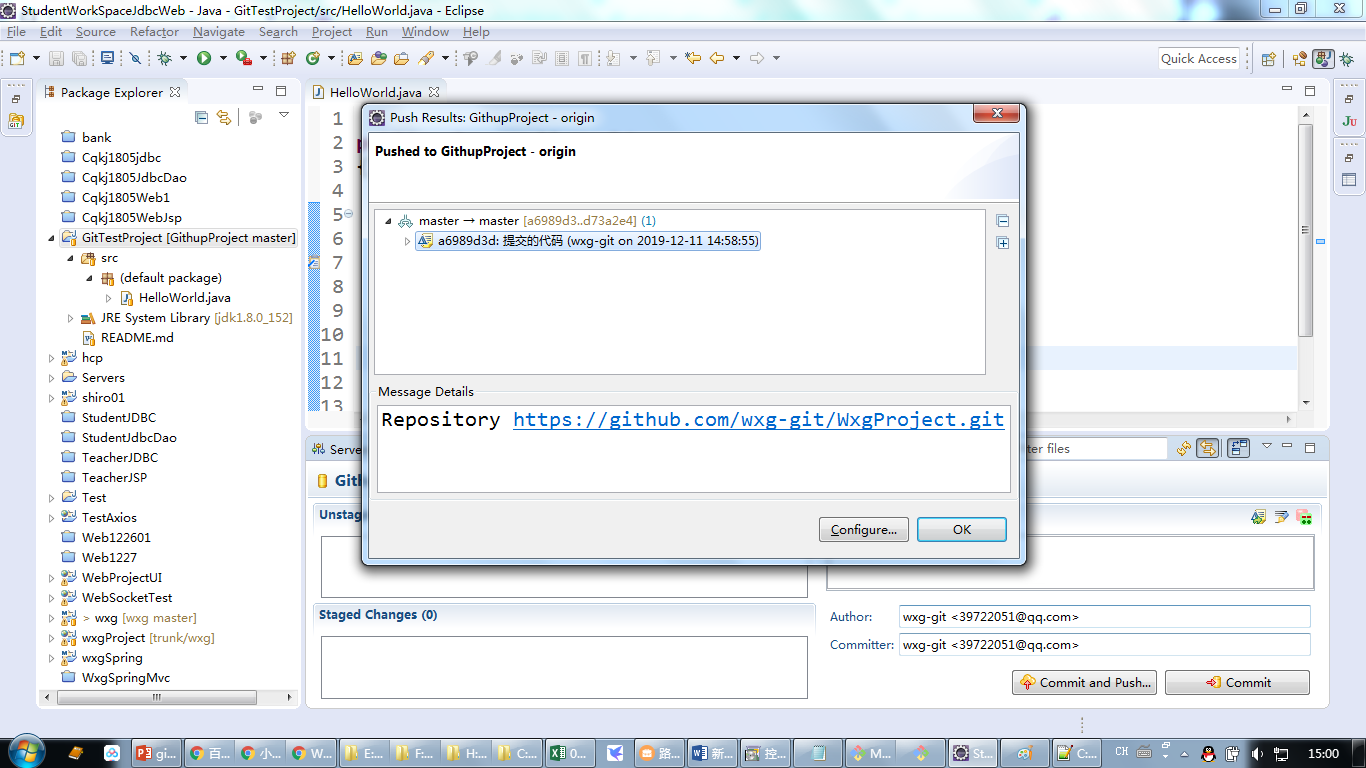




可以点击 commit, 只提交本地，然后再



通过： remote 推送

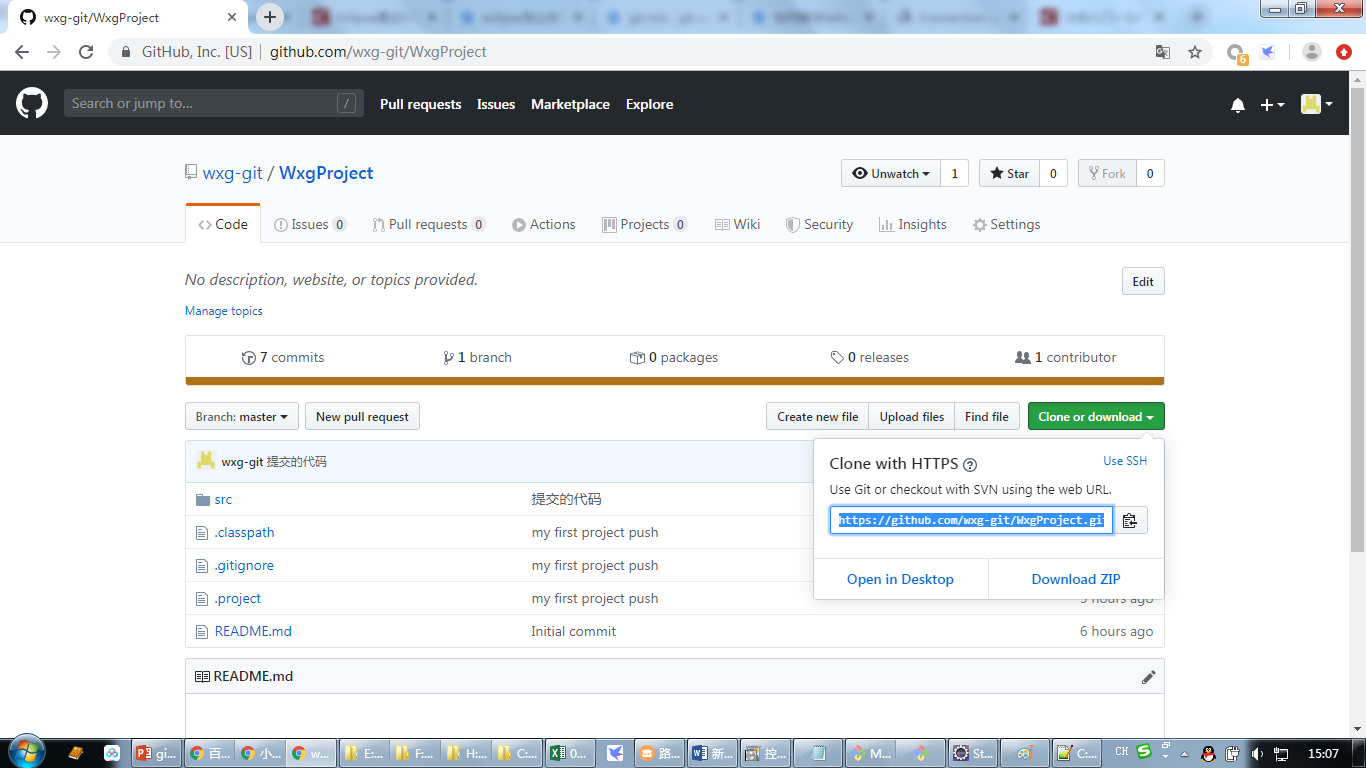


成功

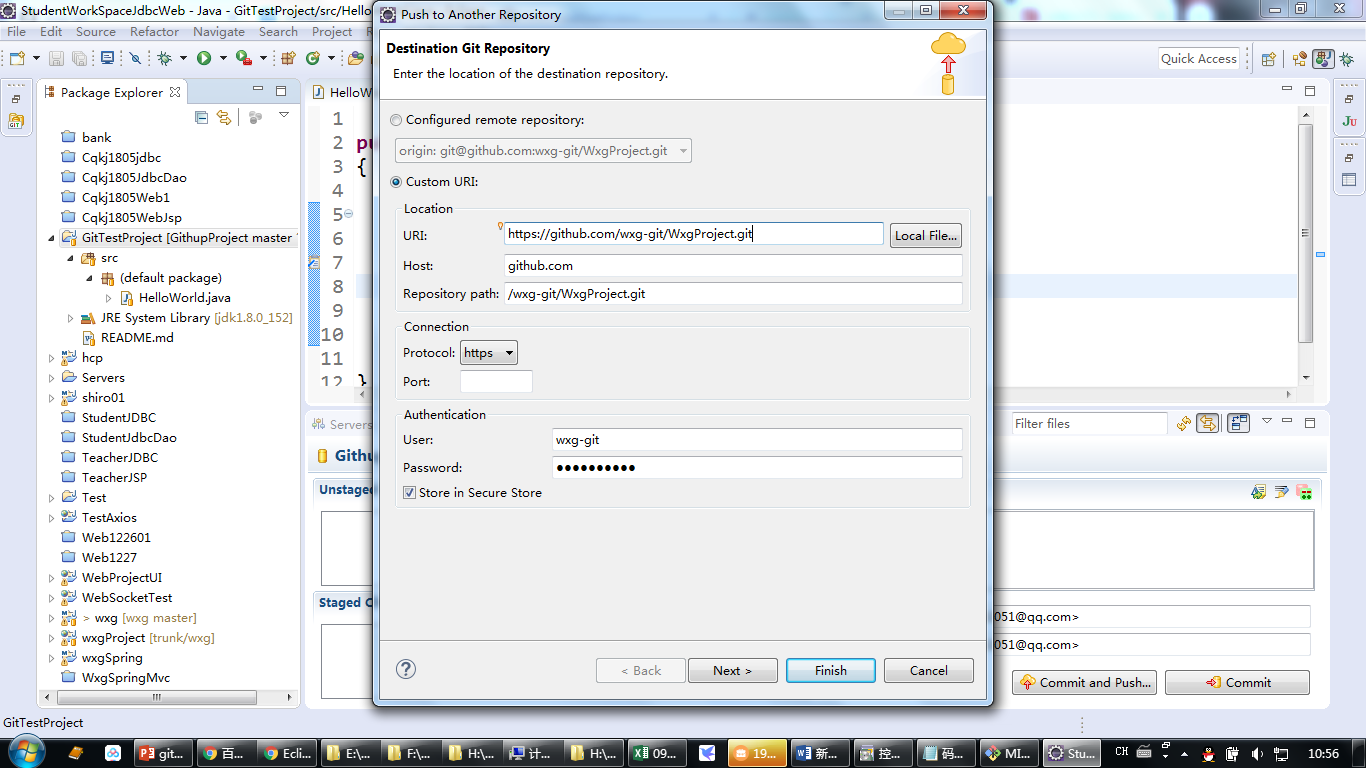
也可以点击commit and push …. (提交推送一起)

注： 如果第一次使用，可能还没有关联远程服务器，则需要自己配置一下。

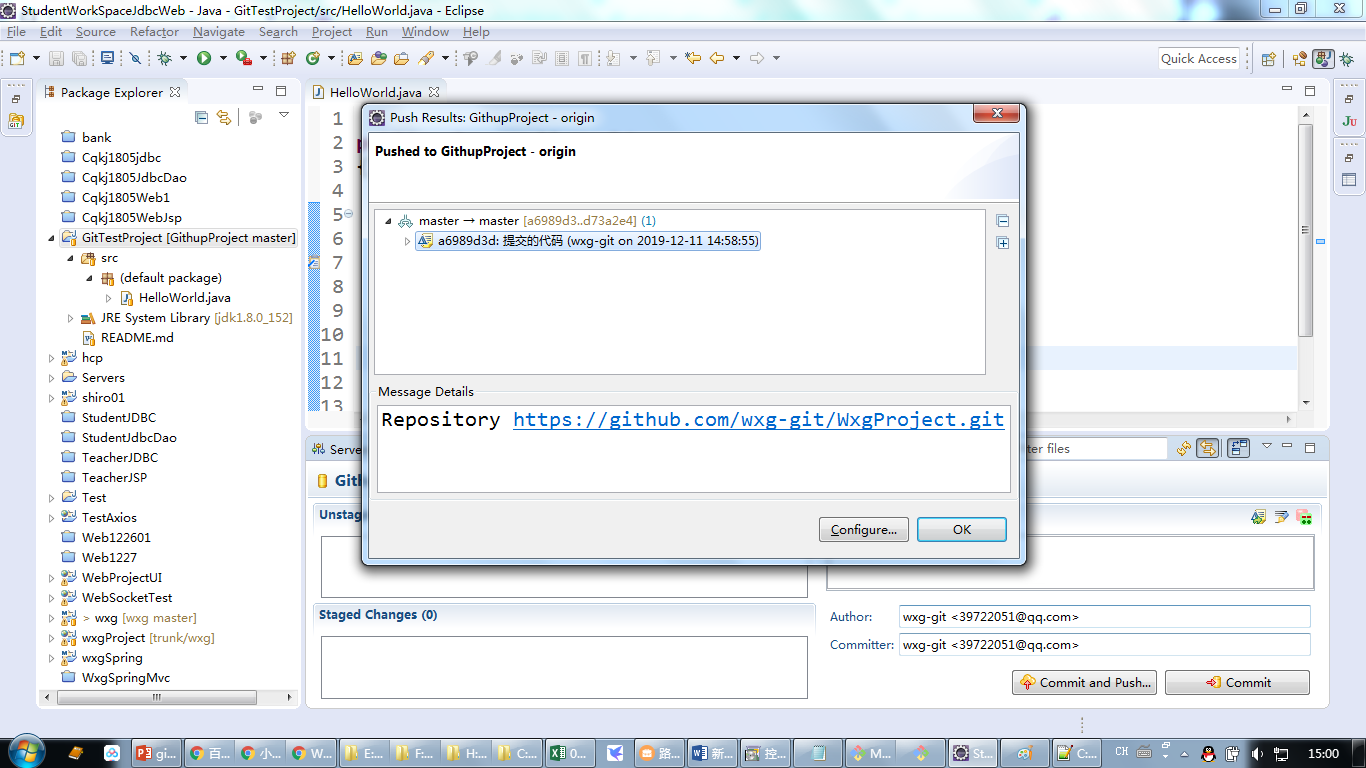
如：URL输入远程仓库的地址，可以通过以下图示提示去拿。



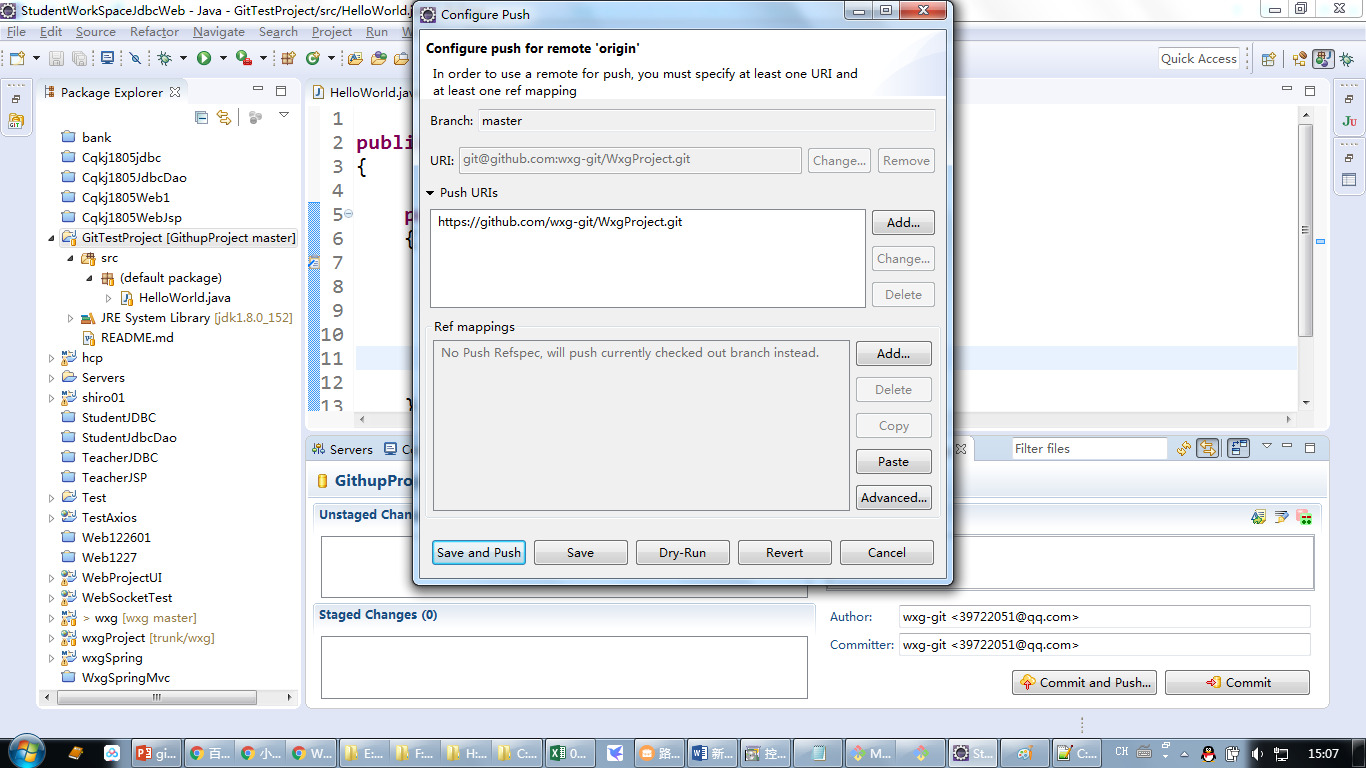
用户名密码就是githup 账户



第一次推送代码成功后：可以将配置信息保存，不然每次都可能要你填写。

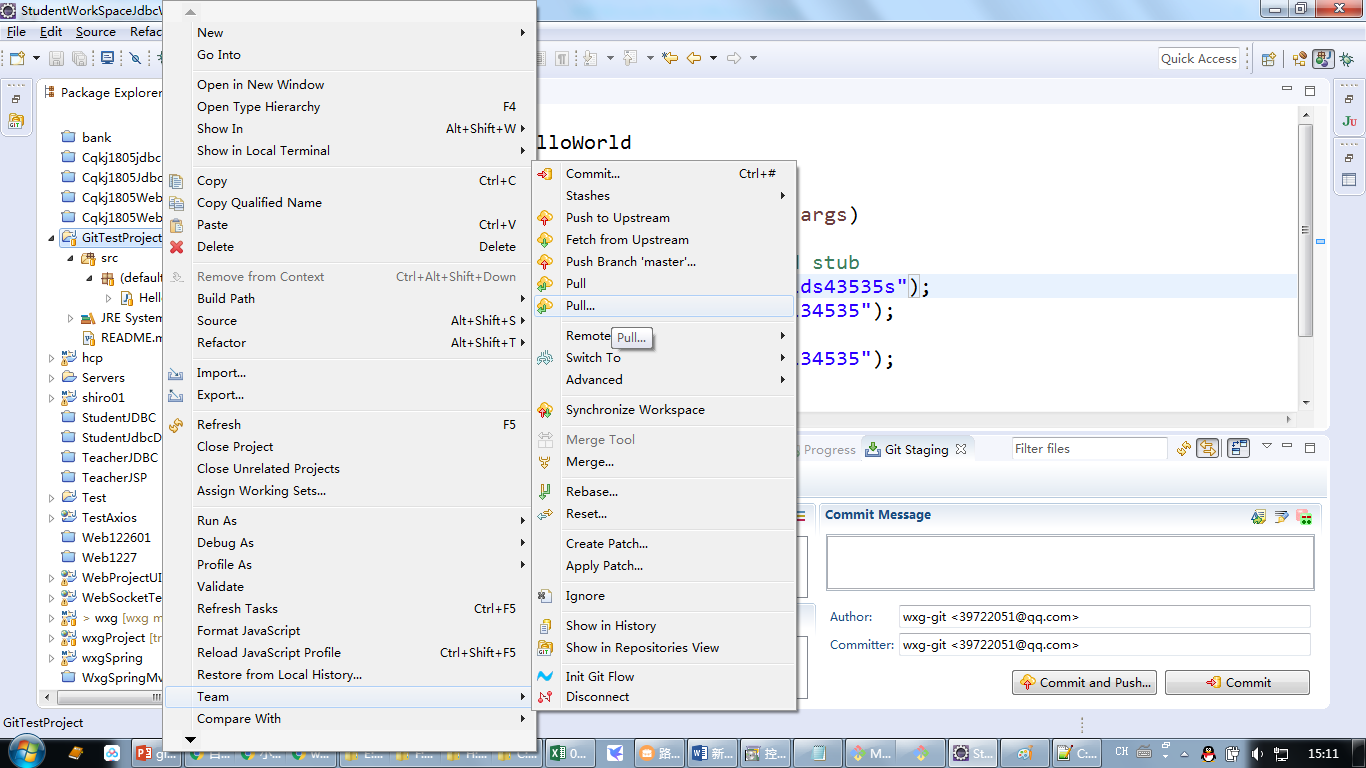


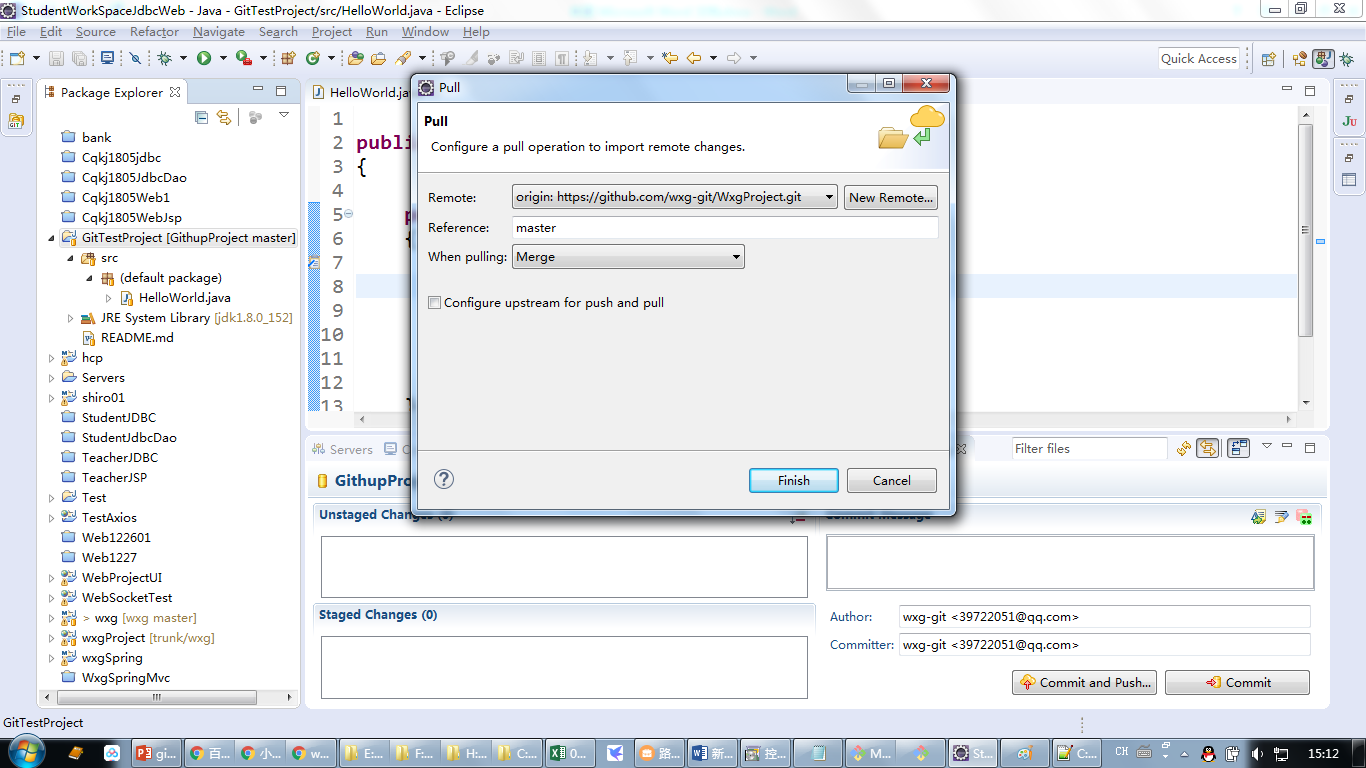
点击Configure…



在push URLs点击 Add…

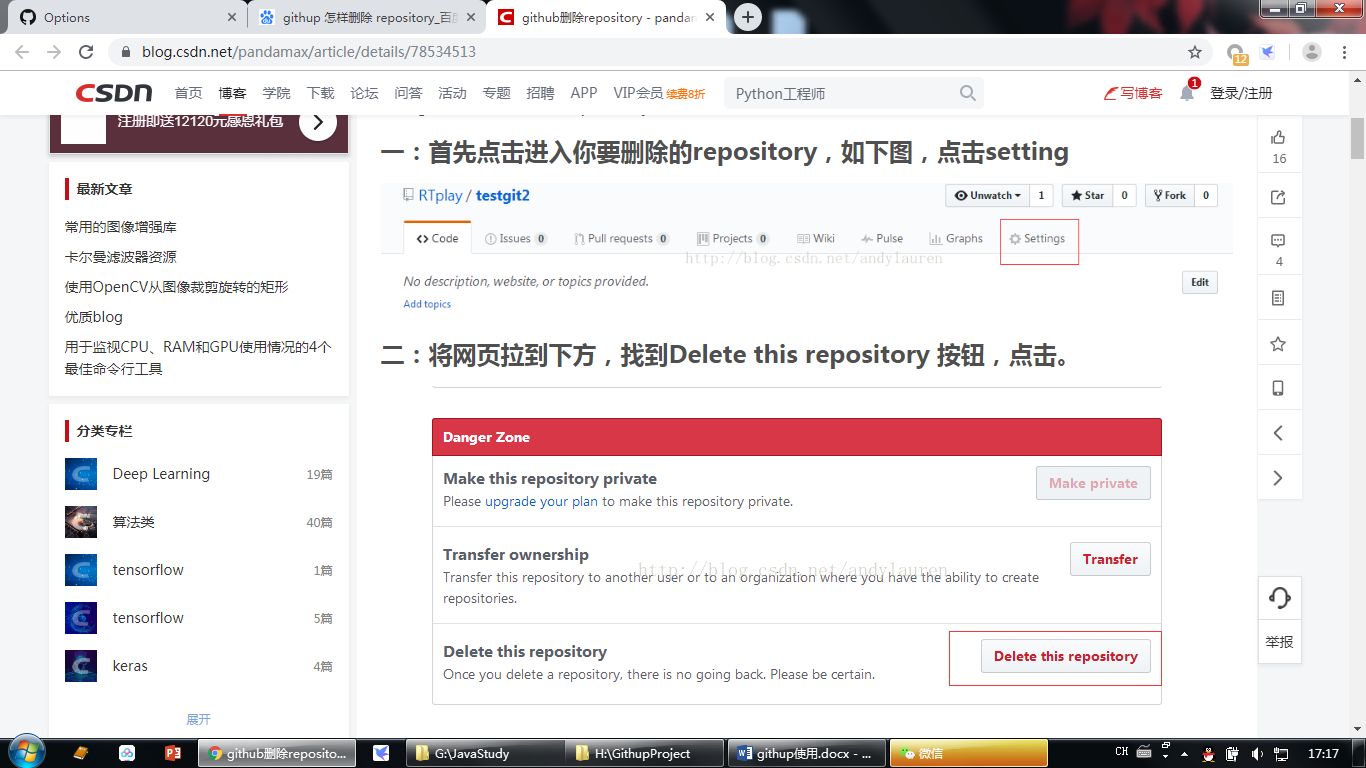
9:获取最新代码：右键项目 pull…

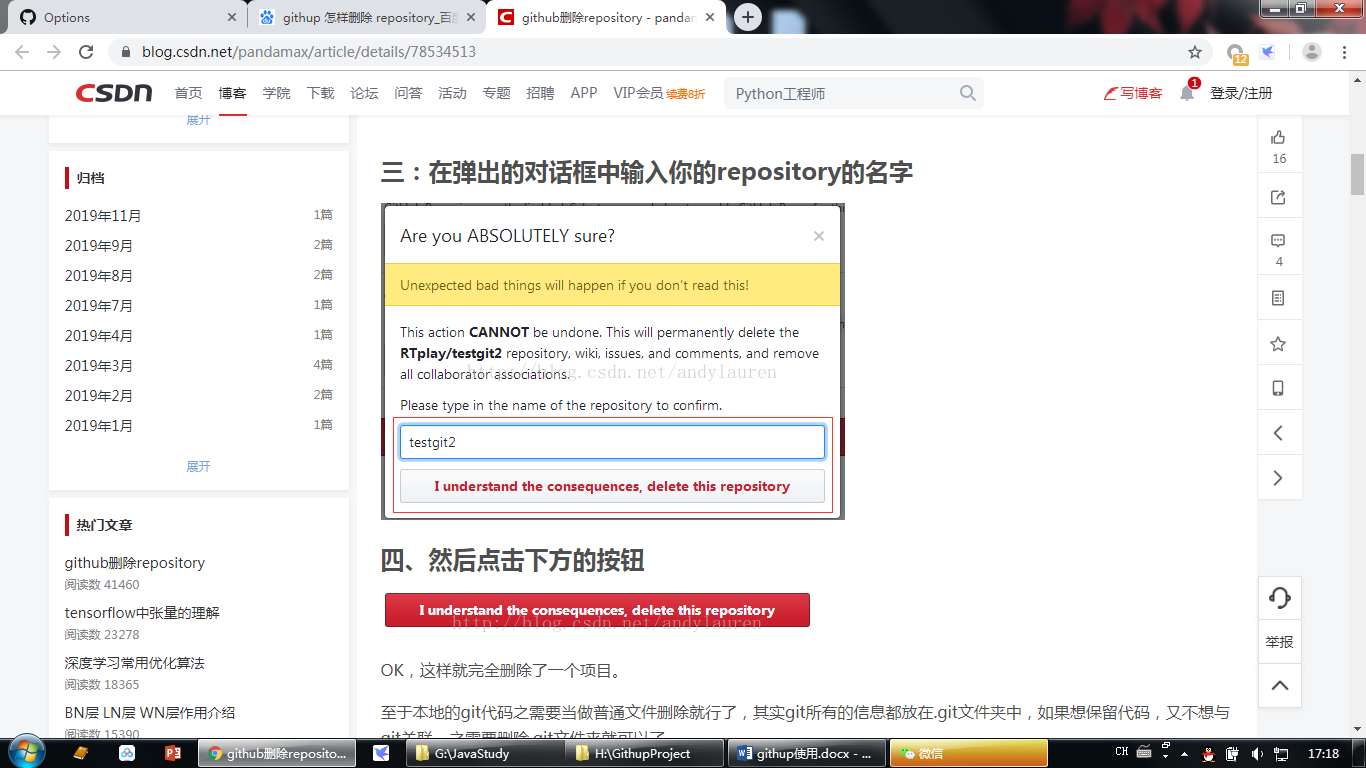




输入地址： New Remote 添加新的。。。

Githup:删除repository：





**Git基本常用命令如下：**

***mkdir***：         XX (创建一个空目录 XX指目录名)  
   ***pwd***：          显示当前目录的路径。  
   ***git init*** ：         把当前的目录变成可以管理的git仓库，生成隐藏.git文件。  
   ***git add XX*** ：      把xx文件添加到暂存区去。  
   ***git commit -m “XX”*** ： 提交文件 –m 后面的是注释。  
   ***git status***：        查看仓库状态  
   ***git diff  XX***：     查看XX文件修改了那些内容  
   ***git log*** ：         查看历史记录  
   ***git reset  --hard HEAD^***：或者 ***git reset  --hard HEAD~*** 回退到上一个版本  
                        (如果想回退到100个版本，使用***git reset --hard HEAD~100*** )  
   ***cat XX***   ：      查看XX文件内容  
   ***git reflog***  ：     查看历史记录的版本号id  
   ***git checkout -- XX*** ： 把XX文件在工作区的修改全部撤销。  
   ***git rm XX*** ：        删除XX文件  
   ***git remote add origin https://github.com/RTplay/testgit.git***： 关联一个远程库  
   ***git push -u(第一次要用-u 以后不需要) origin master***：把当前master分支推送到远程库  
   ***git clone https://github.com/RTplay/testgit.git*** ： 从远程库中克隆  
   ***git checkout -b dev*** ： 创建dev分支 并切换到dev分支上  
   ***git branch***  ：查看当前所有的分支  
   ***git checkout master*** ：切换回master分支  
   ***git merge dev***    ：在当前的分支上合并dev分支  
   ***git branch -d dev*** ：删除dev分支  
   ***git branch name***  ：创建分支  
   ***git stash*** ：把当前的工作隐藏起来 等以后恢复现场后继续工作  
   ***git stash list*** ：查看所有被隐藏的文件列表  
   ***git stash apply*** ：恢复被隐藏的文件，但是内容不删除  
   ***git stash drop***： 删除文件  
   ***git stash pop***： 恢复文件的同时 也删除文件  
   ***git remote***： 查看远程库的信息  
   ***git remote -v*** ：查看远程库的详细信息  
   ***git push origin master***  ：Git会把master分支推送到远程库对应的远程分支上

config 配置有system级别 global（用户级别） 和local（当前仓库）三个 设置先从system-》global-》local  底层配置会覆盖顶层配置 分别使用--system/global/local 可以定位到配置文件

查看系统config

|  |  |
| --- | --- |
| 1 | git config --system --list |

查看当前用户（global）配置

|  |  |
| --- | --- |
| 1 | git config --global  --list |

查看当前仓库配置信息

|  |  |
| --- | --- |
| 1 | git config --local  --list |

# 错误 You've successfully authenticated, but Gitee.com does not provide she access.

git pull origin master –allow-

unrelated-history

然后提交