## 一、Git常用命令

初始化一个Git仓库，使用git init命令。

添加文件到Git仓库，分两步：

使用命令git add <file>，注意，可反复多次使用，添加多个文件；

使用命令git commit -m <message>，完成。

要随时掌握工作区的状态，使用git status命令。

git status告诉你有文件被修改过，用git diff可以查看修改内容。

使用命令git log可以查看提交历史

使用命令git log –pretty=oneline可以查看提交记录变一行显示

使用命令git reflog查看命令历史

HEAD指向的版本就是当前版本，使用命令git reset --hard commit\_id。历史之间穿梭

想直接丢弃工作区的修改时，用命令git checkout – file

当你不但改乱了工作区某个文件的内容，还添加到了暂存区时，想丢弃修改，分两步，第一步用命令git reset HEAD <file>，第二步使用git checkout – file

已经提交了不合适的修改到版本库时，想要撤销本次提交，则采用回退命令 git reset – hard commitId

命令git rm用于删除一个文件

## 二、远程厂库

第1步：创建SSH Key。在用户主目录下，看看有没有.ssh目录，如果有，再看看这个目录下有没有id\_rsa和id\_rsa.pub这两个文件，如果已经有了，可直接跳到下一步。如果没有，打开Shell（Windows下打开Git Bash），创建SSH Key：

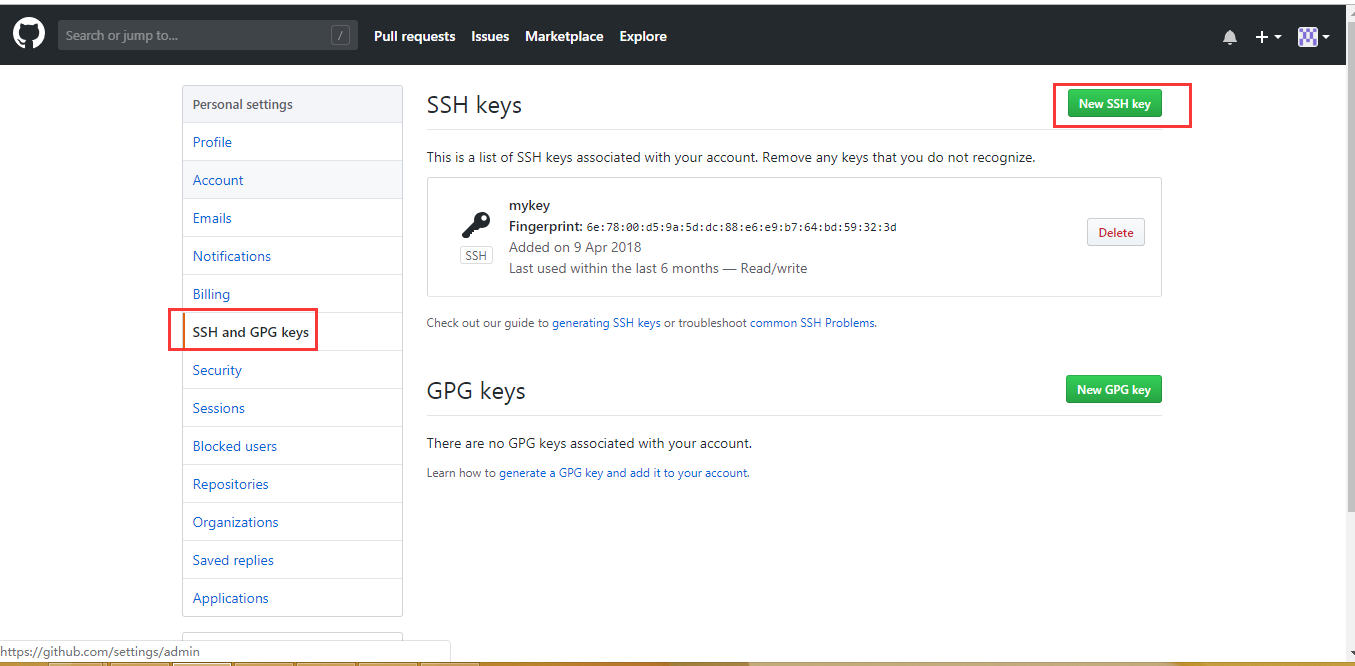
|  |
| --- |
| ssh-keygen -t rsa -C "youremail@example.com"ssh-keygen -t rsa -C "921119098@qq.com" |

你需要把邮件地址换成你自己的邮件地址，然后一路回车，使用默认值即可，由于这个Key也不是用于军事目的，所以也无需设置密码。

如果一切顺利的话，可以在用户主目录里找到.ssh目录，里面有id\_rsa和id\_rsa.pub两个文件，这两个就是SSH Key的秘钥对，id\_rsa是私钥，不能泄露出去，id\_rsa.pub是公钥，可以放心地告诉任何人。

第2步：登陆GitHub，打开“Account settings”，“SSH Keys”页面：

然后，点“Add SSH Key”，填上任意Title，在Key文本框里粘贴id\_rsa.pub文件的内容：



常用命令：

要克隆一个仓库，首先必须知道仓库的地址，然后使用git clone命令克隆。

要关联一个远程库，使用命令git remote add origin

把本地库的内容推送到远程，用git push命令，实际上是把当前分支master推送到远程。

由于远程库是空的，我们第一次推送master分支时，加上了-u参数，Git不但会把本地的master分支内容推送的远程新的master分支，还会把本地的master分支和远程的master分支关联起来，在以后的推送或者拉取时就可以简化命令。

要查看远程库的信息，用git remote

用git remote -v显示更详细的信息

从本地推送分支，使用git push origin branch-name，如果推送失败，先用git pull抓取远程的新提交；

在本地创建和远程分支对应的分支，使用git checkout -b branch-name origin/branch-name，本地和远程分支的名称最好一致；

建立本地分支和远程分支的关联，使用git branch --set-upstream branch-name origin/branch-name；

从远程抓取分支，使用git pull，如果有冲突，要先处理冲突。

## 三、分支操作

查看分支：git branch

创建分支：git branch <name>

切换分支：git checkout <name>

创建+切换分支：git checkout -b <name>

合并某分支到当前分支：git merge <name>

删除分支：git branch -d <name>

用git log --graph命令可以看到分支合并图

git merge --no-ff -m"merge with no-ff" dev合并分支时，加上--no-ff参数就可以用普通模式合并，合并后的历史有分支，能看出来曾经做过合并，而fast forward合并就看不出来曾经做过合并

git stash 可以把当前工作现场“储藏”起来，等以后恢复现场后继续工作

你可以多次stash，恢复的时候，先用git stash list查看，然后恢复指定的stash，用命令git stash apply恢复，但是恢复后，stash内容并不删除，你需要用git stash drop来删除

git stash pop，恢复的同时把stash内容也删了

git stash apply stash@{0}恢复指定版本

如果要丢弃一个没有被合并过的分支，可以通过git branch -D <name>强行删除