# git命令详解

## 1、用git的命令行修改提交的用户名和邮箱：

git config --global user.name "Your Name"  
git config --global user.email youEmail@example.com

如果只是查看的话就不要加后面的用户名和邮箱了。

## 2、关于git ssh key密钥：

SSH key 的作用就是远程存放仓库的服务器要注册有你本地生成的ssh key ，这样你才能把你本地的仓库和远程的仓库进行关联，同步修改。不然你是没有权限把本地修改同步到远程的。

1、查看是否已经有了ssh密钥：cd ~/.ssh

存放密钥的文件夹：.ssh一般是自动生成在c盘的用户目录下的。如果上面的命令能定位到这个文件夹则表示已经生成好了。

还可以通过命令： ls 查看当前文件夹下的目录。

2、生成密钥：ssh-keygen –t rsa –C “623578381@@qq.com”

注意这里的C一定要是大写的。

之后会提示要你输入密码：直接3个回车就行了，表示密码为空。

3、在github上添加ssh密钥，找到github官网里的Settings -> SSH keys ,然后这里要添加的是“id\_rsa.pub”里面的公钥。Title 随便填。

4、github官网上是可以添加多个SSH key的，但是不同的电脑不能用同一个ssh key。都要重新独立生成。

## 3、远程仓库和本地仓库的关联同步：

有两种方式关联：

1. 使用添加远程仓库的方式：（注意远程仓库和本地仓库的根目录要同名）

首先，在github网站上建好一个仓库（当然也可以是任何远程主机），github上新建的仓库会提示你是否新建时建一个README.md文件，可以不用勾选，建一个空的仓库就好了。

------------------------------

然后在本地也建好一个仓库：步骤流程如下：首先，用git bash 命令行cd定位到要创建仓库的目录里。

然后，运行命令：git init ——初始化。

然后，运行命令：git add README.md ——把仓库目录下的已有文件一次添加到缓存。可以一多次添加多个文件，还有注意文件的路径。

然后， 运行命令： git commit -m “一些提交的说明文字” ——这个说明文字一般是强烈建议要求填写的。

然后，git remote add origin **repository URL。 ——这里就是添加远程仓库与本次仓库进行关联。示列：**git remote add origin [git@github.com:liuyaoao/LearnNotes.git](mailto:git@github.com:liuyaoao/LearnNotes.git)

最后， 运行命令：git push -u origin master ——把本地提交的内容一次性推送到远程仓库去同步。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。

1. **把一个远程仓库克隆到本地。**

**这个远程仓库一般是已经有内容了的。**

**然后运行命令：**git clone [git@github.com:liuyaoao/test.git](mailto:git@github.com:liuyaoao/test.git)

拉取远程仓库：**git** pull [remoteName] [localBranchName]

推送远程仓库：**git push** [remoteName] [localBranchName]

## 4、有关分支的命令：

git branch ——查看当前项目的所有分支。并且会在当前分支前加一个\*号。

git checkout branch\_name ——切换分支。branch\_name 为分支名。

git checkout -b branch\_name ——新建一个新分支，并且切换到这个新分支。

删除分支：**git** branch -d [name] ---- -d选项只能删除已经参与了合并的分支，对于未有合并的分支是无法删除的。如果想强制删除一个分支，可以使用-D选项

合并分支： **git** merge [name] ----将名称为[name]的分支与当前分支合并

创建远程分支(本地分支**push**到远程) **git push** origin [localBranchName:remoteBranchName] ——提交本地的一个分支localBranchName作为远程的remoteBranchName分支 //如果远程没有localBranchName，远程的github就会自动创建一个localBranchName分支作为新的远程分支。

删除远程分支：**git push** origin :remoteBranchName 如果”:”冒号左边的分支为空，那么将删除”:”冒号右边的远程分支。