# Git简易使用指南

使用原则：先有本地库，后有远程库，再关联远程库，使用push推送到远程库，可使用pull或fetch从远程库更新到本地库

## 一、创建版本库

初始化一个Git仓库，使git init命令。

添加文件到Git仓库，分两步：

第一步，使用命令git add ，注意，可反复多次使用，添加多个文件；

第二步，使用命令git commit，完成。

## 二、版本回退

随时掌握工作区的状态，使用git status命令。

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

HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，

版本回退前，使用git log可以查看提交历史，以便确定要回退到哪个版本。

使用命令git reset --hard commit\_id 返回到指定提交id的版本

git reset --hard HEAD^ 返回上一个版本

git reset --hard HEAD~100 返回上100个版本

要重返未来，使用git reflog查看命令历史，以便确定要回到未来的哪个版本。

## 三、Git撤销修改和删除文件操作

### 1.撤销修改：

场景1：当你改乱了工作区某个文件的内容，想直接丢弃工作区的修改时，使用命令 git checkout -- filename。

场景2：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即使用了add命令，想丢弃修改，分两步，第一步使用命令git reset HEAD 文件名，就回到了场景1，第二步按场景1操作。

场景3：已经提交了不合适的修改到版本库时，想要撤销本次提交，使用版本回退，不过前提是没有推送到远程库。

### 2.删除文件：

可以直接在文件目录中把文件删了，或者使用如上git rm命令：git rm filename

## 四、远程仓库

### **1.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，如果有的话，直接跳过此如下命令，如果没有的话，打开命令行，输入如下命令生成公钥 ssh-keygen -t rsa -C 852465964@qq.com

邮箱改成自己的邮箱。设置用户名，密码。

可使用命令ssh -T git@github.com 测试能否链接

### 2.添加公钥

将生成的公钥添加到远程仓库的ssh公钥；登录github,打开”settings”中的SSH Keys页面，然后点击“Add SSH Key”,填上任意title，在Key文本框里黏贴id\_rsa.pub文件的内容。

### **3.关联远程库**：

git remote add qcrs git@github.com:dk0o0o0/qcrs.git

### 4.将本地库的内容推送到远程

使用命令git push –u [alias] [branch]第一次推送master分纸支的所有内容；

推送本地分支dev到远程分支 dev并建立关联关系

a.远程已有dev分支并且已经关联本地分支dev且本地已经切换到dev 命令：git push origin dev（可配置git 命令 git config --global push.default simple使命令简化为git push)

b.远程已有dev分支但未关联本地分支dev且本地已经切换到dev 命令：git push -u qcrs/dev

c.远程没有dev分支并本地已经切换到dev 命令：git push qcrs dev:dev

### 5.从远程库克隆：

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。

如：git clone git@github.com:michaelliao/gitskills.git

### 6.从远程获取最新版本到本地

git fetch origin master:tmp 相当于是从远程获取最新版本到本地，不会自动merge

或者git pull origin master 相当于是从远程获取最新版本并merge到本地

## 五、分支管理

### 1.常用命令

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

### 2.合并分支冲突：

当Git无法自动合并分支时，就必须手动解决冲突。

解决冲突后，再提交，合并完成。用git log --graph命令可以看到分支合并图。

### 3.分支管理策略。

通常合并分支时，git一般使用”Fast forward”模式，在这种模式下，删除分支后，会丢掉分支信息，使用带参数 –no-ff来禁用”Fast forward”模式。：如命令 git merge –no-ff -m “注释” dev

## 六、bug分支

修复bug时，我们会通过创建新的bug分支进行修复，然后合并，最后删除；

如果修复bug是手头工作没有完成时，先把工作现场git stash一下，然后去修复bug，修复后，

有两种方式可以回到工作现场：

一种方式git stash apply恢复，恢复后，stash内容并不删除，你需要使用命令git stash drop来删除。

另一种方式是使用git stash pop,恢复的同时把stash内容也删除了。

## 七、Feature分支

开发一个新feature，最好新建一个分支；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。

## 八、多人合作

工作模式通常是这样：

1. 首先，可以试图用git push origin branch-name推送自己的修改；

2. 如果推送失败，则因为远程分支比你的本地更新，需要先用git pull试图合并；

3. 如果合并有冲突，则解决冲突，并在本地提交；

4. 没有冲突或者解决掉冲突后，再用git push origin branch-name推送就能成功！

如果git pull提示“no tracking information”，则说明本地分支和远程分支的链接

关系没有创建，用命令git branch --set-upstream branch-name origin/branch-name。

## 九、标签（或里程牌）管理

命令git tag name用于新建一个标签，默认为HEAD，也可以指定一个commit id；

如：git tag name commit id

-a tagname -m "blablabla..."可以指定标签信息；

-s tagname -m "blablabla..."可以用PGP签名标签；

命令git tag可以查看所有标签；

命令git push origin tagname可以推送一个本地标签；

命令git push origin --tags可以推送全部未推送过的本地标签；

命令git tag -d tagname可以删除一个本地标签；

命令git push origin :refs/tags/tagname可以删除一个远程标签。

## 十、Git常用命令

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/tugenhua0707/testgit 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/tugenhua0707/testgit 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