Git学习笔记

创建空目录：mkdir 目录名

初始化：git init（前提：cd 到当前目录下）

添加：git add 文件名

上传：git commit -m “注释”

查看状态：git status

查看修改：git diff

查看版本：git log

**回退更新前查看版本历史!**

查看版本id：git log --pretty=oneline

回退：git reset –hard HEAD^/HEAD~n/commit id

查看命令历史：git reflog



**查看区别：**

git diff 工作区-暂存区

git diff HEAD ( -- filename) 工作区-版本库

git diff –cached 暂存区-版本库

**撤销修改：**

工作区撤销 改动只存在工作区，未add/commit：git checkout -- filename （恢复到版本库状态）

已经add：git reset HEAD filename (将暂存区的修改回退到工作区，之后使用上一步撤销指令彻底撤销)

已经commit：git reset –hard HEAD^/HEAD~n/commit id 版本回退指令，前提是并未推到远程库

**删除文件：**

在工作区直接删除：rm filename

找回：git checkout – filename(从版本库恢复上一个状态)

不找回，保存删除:add + commit (**git追踪更改，而不是文件，把本次删除更改保存到版本库，找回可使用git reset --hard commit id**)

**推送远程版本库：**

1. 先在GitHub上new repo，设置好名称等
2. git remote add origin [git@github.com:GrainGreen/learngit.git](mailto:git@github.com:GrainGreen/learngit.git)（关联一个远程库）
3. git push -u origin master（第一次推送master所有内容到远程库）
4. git push origin master(推送最新更改)

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s。

**克隆版本库：**

1. 获取远程版本库的具体位置；
2. git clone [git@github.com:GrainGreen/gitclone.git](mailto:git@github.com:GrainGreen/gitclone.git)
3. cd gitclone
4. ls (查看)

**分支：**

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>

**解决冲突：**

git merge <name>

修改冲突文件+ add + commit

git checkout -d <name>

**查看合并log：git log --graph --pretty=oneline --abbrev-commit**