**Git 使用**

git中文版：http://git.oschina.net/progit/

git命令：

git init //初始化仓库

git status //查看状态

git add <filename> //加到暂存区

git commit //查看信息

git commit -m "提交信息"

git log //提交记录

git clone [url] //从github上获取项目

git config --global user.name "mogulili"

git config --global user.email "1244076248@qq.com"

git config --list //查看所有的配置项

git对比:

git diff //工作区和暂存区的区别

git diff --cached(--staged) //暂存区和版本库的区别

git diff master //工作区和版本库之间的差异对比，master是主分支

git撤销:

git reset HEAD <file.name> //暂存区撤消回工作区

git checkout -- <file.name> //工作区撤消回版本区

git commit -m <file.name and file.name> --amend //错误提交，重新提交，撤回之前的

git删除：

git rm <file.name> //工作区删除文件，暂存区还有，删除暂存区的文件

git.rm -f <file.name> //工作区和暂存区同时删除

git.rm --cached <file.name> //删除暂存区文件，工作区保留

git恢复:

git checkout id <file.name> //恢复工作区删除的文件

git reset --hard id //还原版本 恢复到指定的版本状态

git reset --hard HEAD^ //还原前一个版本

git reset --hard HEAD~2 //还原到前2个版本

git reflog //回到现在，找到reset的记录，回到当前版本

同步到远程仓库：

git remote 查看远程仓库名字

git remote add //改变远程仓库名字

git remote -v //远程仓库地址

git push 远程仓库名字 master //向github上数据推送，远程仓库默认名字origin

多人协作解决冲突：

git fetch 更新版本，不进行合并操作，

>> Git diff master origin/master //查看不同

>> Git merge origin/master //手动合并 ,commit再push即可解决冲突

git pull 直接合并

&&-------

开源项目协作：

fork //对不开源的项目克隆一下，相当于一个分支

pull request

git分支：

git branch //查看分支

git branch <分支名> //创建分支

>> -d:git branch -d 分支名 //删除分支，被合并的分支。把d改成大写的D，会删除没有合并的分支

>> --merged:git branch --merged //查看合并的分支

>> --no-merged //查看没有合并的分支

git checkout <分支名> //切换到某分支

>> -b: git checkout -b <分支名> //-b代表branch 创建分支并切换到分支

git merge <要合并的分支> //合并分支

