**git常用命令：**

* git init //初始化本地git环境
* git clone XXX//克隆一份代码到本地仓库
* git pull //把远程库的代码更新到工作台
* git pull --rebase origin master //强制把远程库的代码跟新到当前分支上面
* git fetch //把远程库的代码更新到本地库
* git add . //把本地的修改加到stage中
* git commit -m 'comments here' //把stage中的修改提交到本地库
* git push //把本地库的修改提交到远程库中
* git branch -r/-a //查看远程分支/全部分支
* git checkout master/branch //切换到某个分支
* git checkout -b test //新建test分支
* git checkout -d test //删除test分支
* git merge master //假设当前在test分支上面，把master分支上的修改同步到test分支上
* git merge tool //调用merge工具
* git stash //把未完成的修改缓存到栈容器中
* git stash list //查看所有的缓存
* git stash pop //恢复本地分支到缓存状态
* git blame someFile //查看某个文件的每一行的修改记录（）谁在什么时候修改的）
* git status //查看当前分支有哪些修改
* git log //查看当前分支上面的日志信息
* git diff //查看当前没有add的内容
* git diff --cache //查看已经add但是没有commit的内容
* git diff HEAD //上面两个内容的合并
* git reset --hard HEAD //撤销本地修改
* echo $HOME //查看git config的HOME路径
* export $HOME=/c/gitconfig //配置git config的HOME路径

**团队协作git操作流程：**

* 克隆一个全新的项目，完成新功能并且提交：

1. git clone XXX //克隆代码库
2. git checkout -b test //新建分支
3. modify some files //完成修改
4. git add . //把修改加入stage中
5. git commit -m '' //提交修改到test分支
6. review代码
7. git checkout master //切换到master分支
8. git pull //更新代码
9. git checkout test //切换到test分支
10. git meger master //把master分支的代码merge到test分支
11. git push origin 分支名//把test分支的代码push到远程库

* 目前正在test分支上面开发某个功能，但是没有完成。突然一个紧急的bug需要处理

1. git add .
2. git stash
3. git checkout bugFixBranch
4. git pull --rebase origin master
5. fix the bug
6. git add .
7. git commit -m ''
8. git push
9. git checkout test
10. git stash pop
11. continue new feature's development

* git工作流

