**Git Instructions**

**--By Justin hu**

cd 打开 D:

ls 显示当前文件下的子文件

ctr + l 命令行置顶

Git操作指令

git init 初始化

**本地仓库操作：**

git add xxx.txt 提交xxx文件暂存到缓存区

git add \* 提交全部文件到暂存到缓存区

git rm --cached 清除add添加的文件

git commit -m "wrote a readme file" xxx.txt 缓存区的xxx.txt文件提交到本地仓库

git commit -m "wrote a readme file" 缓存区的文件全部提交到本地仓库

git log 显示所有日志 =>空格向下翻页 =>b向上翻页 =>q退出

git log --pretty=oneline 以一行的形式显示日志 -hash值全部显示

git log --oneline 以一行的形式显示日志 -hash值部分显示

git reflog 以一行的形式显示日志 并显示出本版下标HEAD@{移动到当前版本需要的step}

git reset --hard xxx 回退到某个版本(xxx)是对应的hash值

git reset --hard HEAD 重置到HEAD指针指向的版本

git reset --hard HEAD^ 后退到上一个版本

git reset --hard HEAD^^^ 后退到上三个版本

git reset --hard HEAD~n 后退n个退后n步

git reset --soft 仅仅在本地库移动HEAD指针

git reset --mixed 在本地库移动HEAD指针；重置暂存区

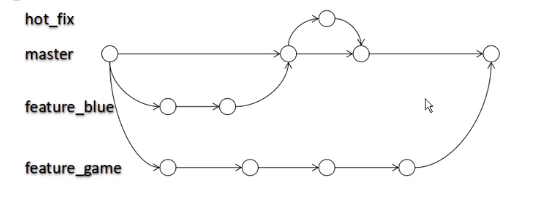
git reset --hard在本地库移动HEAD指针；重置暂存区;重置工作区

git diff xx.txt 将工作区的xxx文件和暂存区进行比较

git diff HEAD xx.txt 将工作区的xxx文件和本地库历史记录比较

不带文件名比较所有文件

分支：



分支好的好处 ：

1. 同时并行推进多个功能开发，提高开发效率，
2. 各个分支在开发过程中，如果某一个分支开发失败，也不会对其他分支有任何的影响，失败的分支删除重新开始即可

git branch -v 查看所有分支

git breach xxx 创建一个xxx分支

git checkout xxx 切换到xxx分支

合并分支

1. 第一步切换到接受修改的分支（被合并，增加新内容）；

git checkout xxx0 切换到要被合并的分支名xxx0

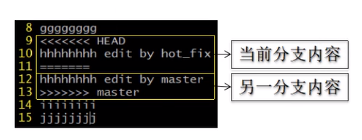
1. 第二步执行merge命令

git merge xxx1 把要合并的分支xxx1合并到当前分支xxx0

要想把B分支的内容合并到A分支上，先切换到git checkout A分支，然后在git merge B ，就把B的内容合并到A里面

合并的冲突解决

合并分支，修改同一行内容产生的冲突的表现：



冲突的解决

1. 第一步：编辑文件，删除特殊符号
2. 第二步：把文件修改到满意的程度，保存退出
3. 第三步：git add xxx.txt
4. 第四步：git commit -m “日志信息”

注意：此时commit一定不能带具体文件名

**本地加远程仓库操作**

git remote add origin <https://github.com/xxxxx> 链接到远端仓库

git remote -v 查看当前仓库连接地址

git push -u origin master 提交master分支到的远端仓库

git push -u origin xxx 提交xxx分支到的远端仓库

**git fetch操作流程：（分支有大的改动-建议想fetch 再merge）**

git fetch origin master 拉取远程仓库master分支的数据 并没有修改本地仓库内容，只是下载，

git fetch origin xxx拉取远程仓库xxx分支的数据 并没有修改本地仓库内容，只是下载，

git checkout origin/master 切换到master-branch下的fetch拉取下来的版本

git checkout origin/xxx 切换xxx-branch状态下的fetch拉取下来的版本

git checkout master 切换到master-branch版本

git checkout xxx 切换到xxx-branch版本

git merge origin/master 把fetch拉取下来的origin/master 合并到当前分支

git merge origin/xxx 把fetch拉取下来的origin/xxx合并到当前分支

**git pull 操作流程：（分支小的改动直接pull，pull = fetch+ merge）**

git pull origin master 直接拉取远程仓库master分支内容并合并到当前分支

git pull origin xxx 直接拉取远程仓库xxx分支内容并合并到当前分支

解决冲突

1. 如果不是基于GitHub远程库的最新版本所做的修改，不能推送，必须先拉取pull下来
2. 拉取下来进去冲突状态，按照分支解决冲突的方式解决即可

**跨团队协做**

团队2， fork 一份仓库在GitHub上

团队2 clone 自己fork过来的仓库连接

团队2 修改内容 push到自己的仓库

团队2然后请求合并到团队1 ，GitHub上点击Pull requests => New pull request

=> Create pull request

团队1确认合并

团队1 GitHub点击Pull request =>merge pull request

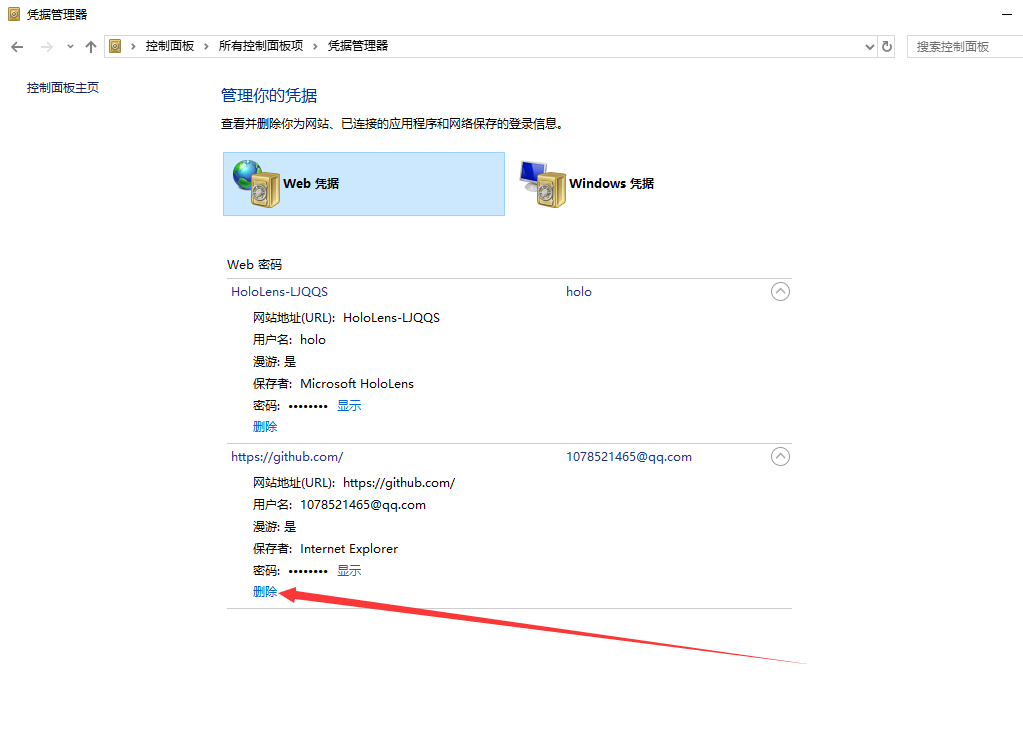
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

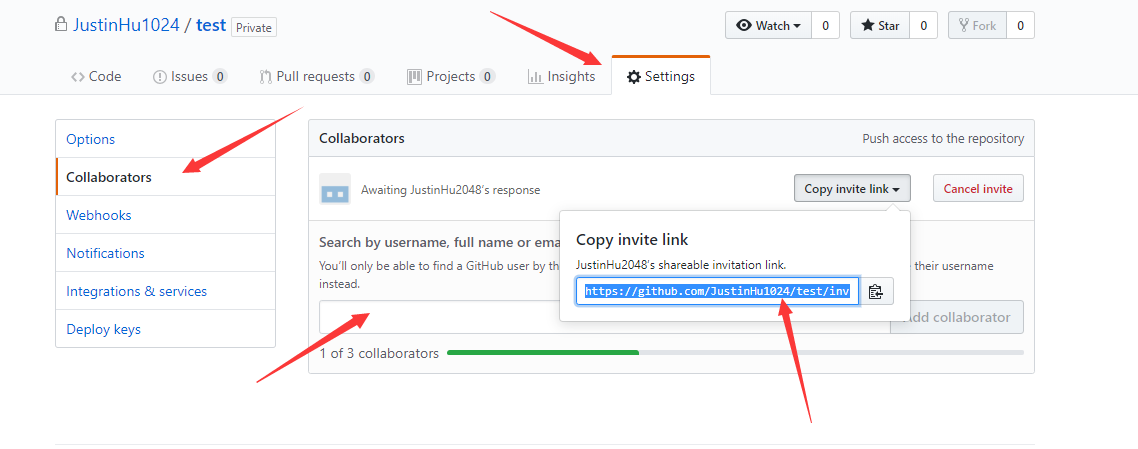
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

同一台电脑切换git账号先删除当前登陆账号

控制面板 => 凭据管理器 => https//github.com/ =>删除



**github添加其他成员合作开发流程：**



**邀请链接发给合作人**

**GitHub忽略push上传文件**

[**https://github.com/github/gitignore 各种开发语言忽略初步配置.gitignore**](https://github.com/github/gitignore%20各种开发语言忽略初步配置.gitignore)

