## 版本控制系统

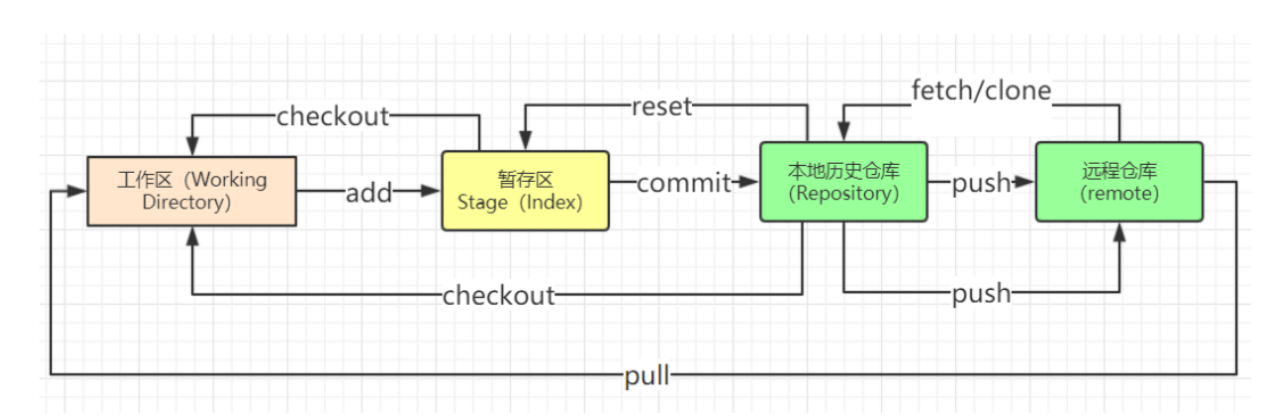
Git：分布式（多人协作）版本控制系统 （文件管理系统）

作用：记录文件的变化与修改，对文件进行溯源，对任何类型的文件都可控制

常见的版本控制系统：集中式多人版本控制系统：如SVN（统一的版本库） 分布式多人协作控制系统：如Git（每个人都有一套完整的版本库）

Git的工作模式：

四个功能分区：工作区 🡪 add暂存区🡪 commit 本地历史仓库🡪push 远程仓库



Git安装

使用命令：

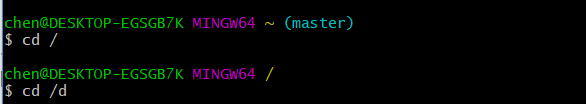
给本地仓库建用户名和邮箱

Git config --global user.name 陈颍

Git config --global user.email “[3376749473@qq.com](mailto:3376749473@qq.com)”

笔记：

进入磁盘



切换盘符



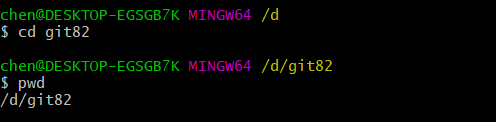
在当前目录创建文件夹



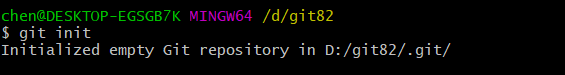
显示当前目录的文件或文件夹



打开文件夹 显示文件夹



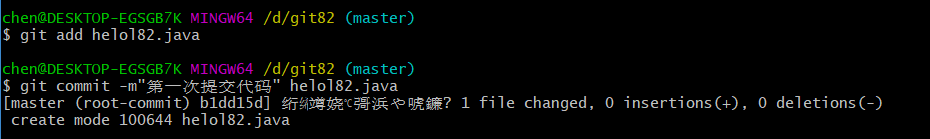
初始化



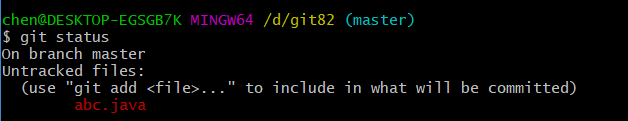
新建文件



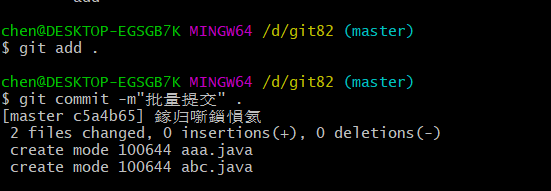
add到暂存区 commit提交



查看文件是否提交



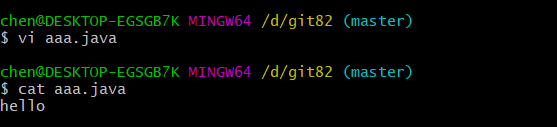
文件批量添加与提交



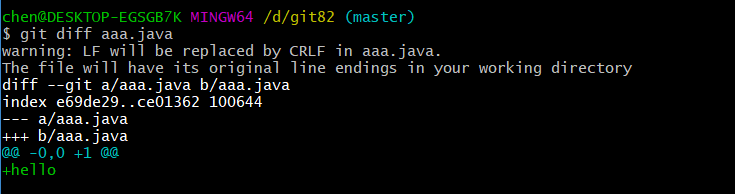
文件操作命令：vi命令

命令模式：按下i🡪插入模式🡪编写代码🡪ESC退出 🡪 shift+：🡪wq（保存退出）

cat 文件名：显示文件内容



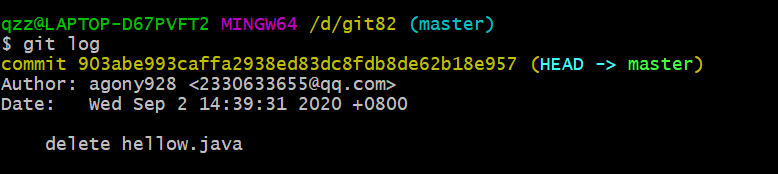
查看文件修改内容



强制递归删除 git rm -f 文件名



查看详情/简略(查看所有的版本号信息)





恢复

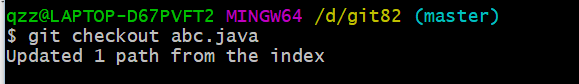
1. 回退到上一个版本(一个^就是回退一个版本，^^就是回退两个)

Git reset head^^

1. 根据版本号回退版本

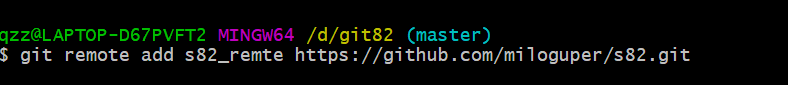
git reset –hard 版本号(eb35a67)

放弃工作区的修改，从暂存区拷贝一份最新的 (放弃本次修改的内容)

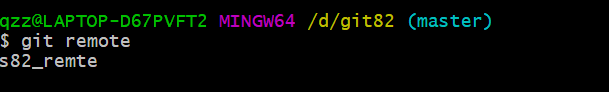


远程仓库：

在本地仓库建立远程仓库 s82\_remte自定义名



查看本地有哪些远程仓库



上传到远程仓库

先git pull 仓库名 master --allow-unrelated-histories 先更新

再 git push 仓库名 master 上传本地仓库的文件

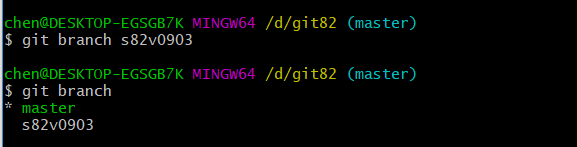
## Git 分支

作用：

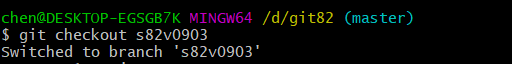
如何创建一个分支：

Git branch 分支名

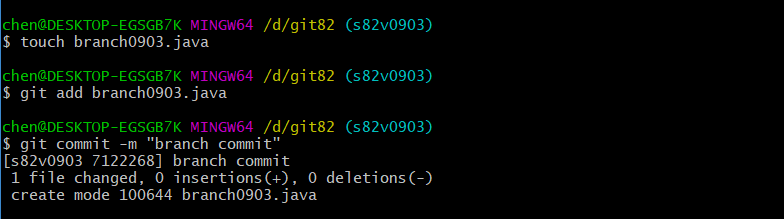
Git branch 查看所有的分支



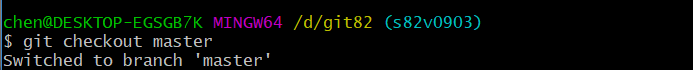
工作区切换为分支：



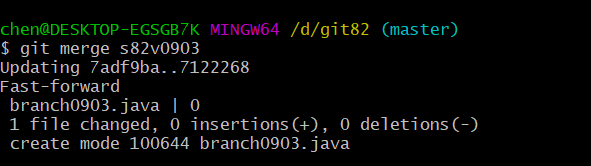
提交文件到分支



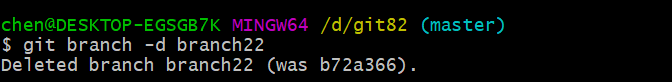
切换分支：



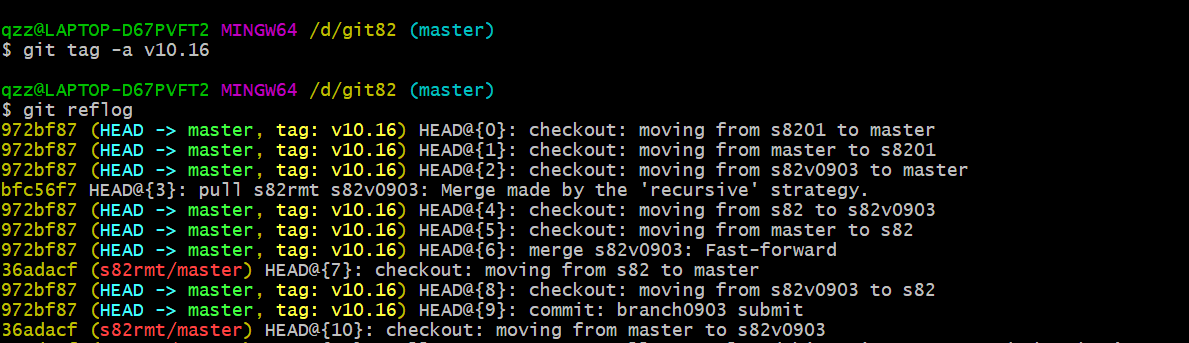
分支合并到主干



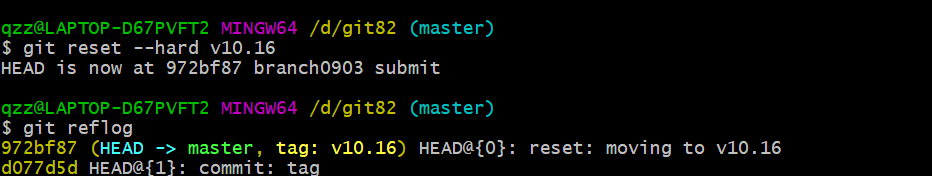
删除分支：



创建标签



给新文件添加标签



克隆clone：从远程仓库克隆

