Git

版本控制系统，是一种文件管理系统，被它关联的文件内容发生任何变化都会记录下来；

常见的版本控制软件有SVN,Git

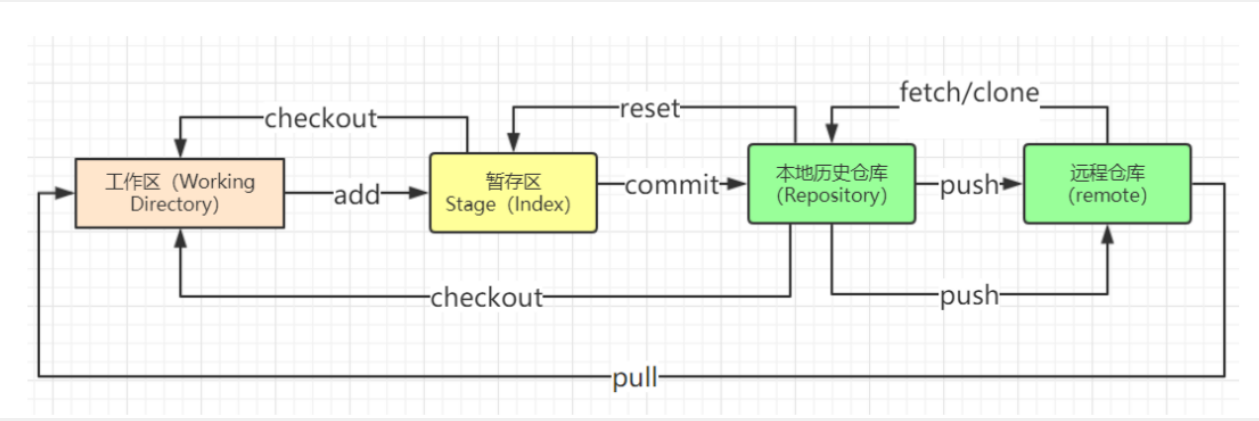
Git:分布式多人协助版本控制系统

每个人都拥有一套完整的版本库

SVN:集中式多人协助版本控制系统

统一的版本库，用户去版本库上update或checkout

Git的工作模式四个区：工作区，暂存区，本地历史仓库，远程仓库



mkdir 创建文件夹

touch 创建文件

Vi 文件编辑

i 从命令模式切换到输入模式

ESC 退出输入模式，进入命令模式

【d:删除当行，u:撤销命令，】

“：” 进入底线模式

【w:保存，q:退出，q!不保存退出】

操作：

Git init ; 初始化仓库

Git add 文件名 ; 添加文件到暂存区

Git commit -m “备注内容” 文件名 ; 提交文件到本地仓库

Git status ; 查看提交状态

Git add . ; 批量添加当前目录的文件到暂存区

Git add \* ; 批量添加所有目录的文件到暂存区

Git commit -m “备注内容” . ; 批量提交当前目录未提交的文件

Git commit -m “备注内容” \* ; 批量提交所有目录未提交的所有文件

cat 文件名 ; 查看文件内容

Git diff 文件名 ; 查看该文件修改的内容

Git config --global user.name “用户名” ; 设置一个用户名

Git config --global user.email “邮箱” ; 设置一个邮箱

Git rm -f 文件名 删除



Git commit -m “注释” . 显示删除一个文件

Git log 查看所有提交记录

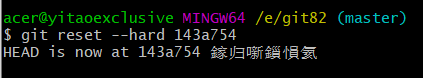
Git reflog 查看版本记录



Git reset head^ 回滚上一个版本

Git reset head^^ 回滚 n个^ 版本

git reset --hard （版本号） 到指定版本



Git checkout 文件名 放弃工作区的修改，从暂存区拿一份新的

Git remote add 文件名 链接 添加一个远程仓库

git pull s82\_remote master 拉取远程仓库数据

git push 文件名 master 提交到远程仓库

git clone 远程地址 克隆远程仓库代码到本地仓库