# git常用命令

1. 下载并安装git
2. 下载地址

<https://git-scm.com/downloads>

1. 安装并查看版本

git --version

1. 全局初始化用户名、邮箱、颜色
2. 初始化用户名

git config --global user.name '你的姓名'

1. 初始化邮箱

git config --global user.email '你的邮箱'

1. 查看初始化设置是否正确(用户名&邮箱)

git config --global --list

1. 初始化颜色

git config --global color.ui true 开启

git config --global color.ui false 关闭

1. 查git的配置列表

git config --list

1. 切换项目路径、查看文件
2. 切换盘符及项目路径

切换盘符cd E: 切换项目cd projectName

1. 查看当前所在路径

pwd

1. 查看当前路径中的所有文件

ls或dir

1. 查看某一个文件的内容

cat xxx(文件名称)

1. 初始化版本库，
2. 项目根目录会创建一个.git文件夹

git init

1. 设置忽略文件
2. 设置每个人都想要忽略的文件
3. 在根目录创建一个名为.gitignore的文本文件，在命令 行 执行 echo \*.jpg>.gitignore   
    注:>左右不要有空格
4. 将.gtignore文件加入版本库并提交
5. 设置只有自己需要忽略的文件

修改.gitignore/info/exclude文件

可使用正则表达式 例如\*.[oa]等价于\*.o和\*.a

1. 文件提交到暂存区
2. 提交暂存区

git add somefile.txt(文件名) 添加单个文件夹

git add \*.txt 添加所有文件

git add . 添加所有文件 包含子目录，但不包含空目录

1. 告诉git把文件提交到暂存区，记录日志

git commit -m '描述信息(改动什么)'

1. 查看仓库状态(文件有没有新增、编辑、删除)

git status

1. 查看文件具体修改什么内容

git diff

1. 查看提交历史记录

git log 或 git log --pretty=oneline

1. 查看每次记录，commit id，确定要回到未来的哪个版本

git reflog

1. 回退版本
2. 指定版本号

git reset --hard xxx(版本号=commit id)

1. 回退到上一版本

git reset --hard HEAD^

注：在git中，用HEAD表示当前版本，就是最新提交 的1094adb...(commit id)，上一个版本就是HEAD^， 上上一个版本就是HEAD^^，往上100个版本就是 写100个^比较容易数不过来，所以就写HEAD~100。

1. 删除文件
2. 直接删除

git rm xxx(文件名称) 之后执行git commit 命令

2.误删除，恢复文件

1. 创建、切换、合并、删除分支
2. 查看分支

git branch

1. 创建分支并切换

git checkout -b dev(分支名称)

git switch -c dev(分支名称)

注：git checkout命令加上-b参数表示创建并切换，相 当于以下两条命令

git branch 分支名称(创建新分支)

git checkout 分支名称(切换到新分支)

1. 合并分支(删除分支前，先切换到主分支在合并)

git merge dev(分支名称)

1. 删除分支(需要切换到其他分支)

git branch -d dev(分支名称)

1. 提交到远程仓库github
2. 创建远程仓库origin与本地仓库相对应

git remote add origin xxx(仓库地址)

1. 提交代码

git push -u origin master  
 注：-u origin master将本地仓库的提交到origin仓库的 master 分支里。这三个参数只有在第一次push的 时候才需要添加，后续提交直接用git push即可。

1. 克隆项目
2. 直接克隆

git clone xxx(仓库地址)

1. 指定分支克隆

git clone -b master(分支名称) xxx(仓库地址)