1. 本地库初始化

* 命令：git init
* 作用：生成 ./git隐藏文件夹

1. 设置签名

* 形式：
  + 用户名：tom
  + Email地址：[goodMorning@xx.com](mailto:goodMorning@xx.com)
* 作用：区分不同开发人员的身份
* 辨析：这里的签名与登录远程库的账号密码没有关系
* 命令
* 级别优先级
  + 项目级别/仓库级别：仅在当前本地库范围
    - git config user.name tom\_pro
    - git config user.email [goodMorning@xx.com](mailto:goodMorning@xx.com)
    - 信息保存的位置：.git/config文件中
  + 系统用户级别：登录当前操作系统的用户范围
    - git config --global user.name tom\_pro
    - git config --global user.email [goodMorning@xx.com](mailto:goodMorning@xx.com)
    - 信息保存的位置：~/.gitconfig文件中
  + 优先级
    - 就近原则：项目级别优于系统用户级别，二者都有时采用项目级别
    - 不允许两者都没有

1. 基本操作
   1. git status 查看工作区、缓存区的状态
   2. git add [file name] 将工作区“新建/修改”添加到缓存区
   3. git commit -m “message” [file name] 将缓存区的内容提交到本地库
   4. 1)git log完整的日志，多屏显示时，空格向下翻页，b向下翻页，q；2)git log --pretty=oneline 显示一行; 3)git log --oneline 显示一行； git reflog 显示移动到当前版本的步数
   5. 前进后退

* 本质，HEAD指针
* 基于索引值（推荐）git reset --hard [局部索引值]
* 使用^符号：只能后退。git reset --hard HEAD^^(一个^退一个版本)
* 使用~符号：只能后退。git reset --hard HEAD~3(回退3个版本)
  1. a

1. H
2. 设置别名

查看别名：git remote -v

添加别名：git remote add origin https://github.com/xx/xx.git，用origin作为网址的别名。

1. 推送

git push origin master,需要登录

1. 解决冲突

* 要点
  + 如果不是基于github远程库的最新版所做的修改，不能推送，必须先拉取
  + 拉取下来后如果进入冲突状态，安装“分支冲突解决”操作