

你好，现在开始学习 Git 分布式版本控制系统。

1. - `git init`, 初始化目录为一个 Git 仓库

- `git add`, 添加文件到 Git, 准备提交到仓库

- `git commit -m "wirte somthing about your changes"`, 提交文件到仓库

2. 修改你的文件

- `git status` 告诉你哪个文件被修改过

- `git diff` 告诉你文件哪里被修改成什么样子了

- `git add`, `git commit -m "xxx"` 把修改过的文件扔到仓库里

3. 版本退回

- `git log` 或者 `git log --pretty=oneline` 查看历史版本的 ID 号, 以及你的备注 (就是`commit -m ""`) 后面写的东西

- `git reset HEAD^^` 退回到上两个版本, 后面几个尖角号就是退回到前面第几个版本

- `git reflog` 现在退回到旧版本了, 但如果又想重返未来, 而log文件里面刚才最新的版本已经消失了, 此时, 要么在命令窗口里往上翻, 找到那个版本的 ID, 如果你把窗口关了, 重新打开之后, 也能在reflog文件里查看未来的版本记录

- `git reset --hard ID` 退回到 ID 指定的版本, ID 是你要退回的版本号, 在log或者reflog里面

4. 专有名词

工作区 就是文件存放的目录, 手动修改文件的地方

版本库 工作区同目录下的 `.git` 文件夹, 里面存放了分支、指向分支的指针 `HEAD`, 以及暂存区

暂存区 每次修改文件之后, 先把修改过的文件加入到暂存区之后, 再一起提交到分支

5. 撤销修改

- `git checkout -- file` 撤销工作区的修改, 注意, 两个短线左右都有空格, 或者暂存区还未提交的文件被修改了, 也可以撤销修改

- `git reset HEAD file` 如果修改过的文件被放到暂存区了, 那么用这个命令再拿回来, 然后再用上一条命令撤销修改

6. 删除文件与一键还原

- `git rm file` 从版本库里面删除文件。如果某个文件被删除了 (用的是 `rm` 或者是用资源管理器删除), 此时工作去和暂存区不一致, `git status` 会告诉那个文件被删了, 如果真的要删除它, 就用这个命令, 然后再提交

- `git checkout -- file` 如果发现是删错了, 则可以还原。这个命令真的是不管文件被修改了还是被删除了, 都可以一键还原