安装 git

$ git config --global user.name "Your Name"

$ git config --global user.email [email@example.com](mailto:email@example.com)

$ git config –list //查看设置

**版本库创建**

$ mkdir learngit

$ cd learngit

$ pwd

/Users/michael/learngit //目录

pwd命令用于显示当前目录。这个仓库位于/Users/michael/learngit。

git init命令把这个目录变成Git可以管理的仓库



.git目录不要修改

git add告诉Git，把文件添加到仓库



git commit告诉Git，把文件提交到仓库

git commit命令，-m后面输入的是本次提交的说明，可以输入任意内容，当然最好是有意义的，这样你就能从历史记录里方便地找到改动记录。

git commit命令执行成功后会告诉你，1 file changed：1个文件被改动（我们新添加的readme.txt文件）；2 insertions：插入了两行内容（readme.txt有两行内容）。

commit可以一次提交很多文件，所以你可以多次add不同的文件，比如：

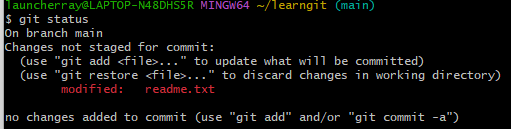
$ git add file1.txt

$ git add file2.txt file3.txt

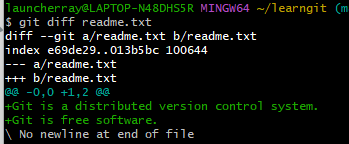
$ git commit -m "add 3 files."

**时光穿梭机**

git status命令可以让我们时刻掌握仓库当前的状态，上面的命令输出告诉我们，readme.txt被修改过了，但还没有准备提交的修改。

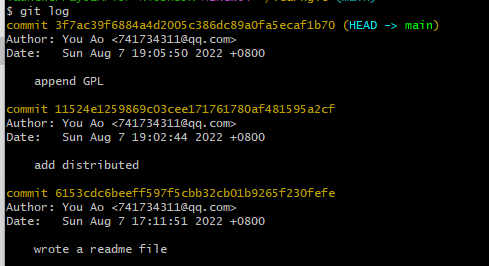


git diff查看具体修改了什么内容

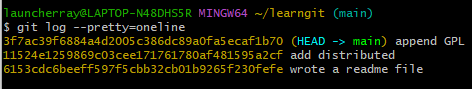


**版本回退**

Git log查看修改内容



git log –pretty=oneline （精简）



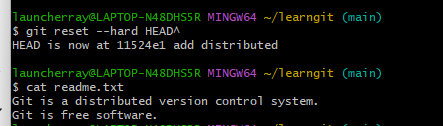
黄色数字为版本号（16进制）

Git reset

git re**set** --hard HEAD^



结果：

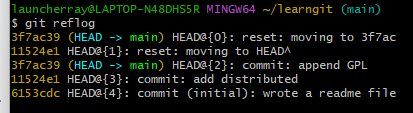


回到最新版本

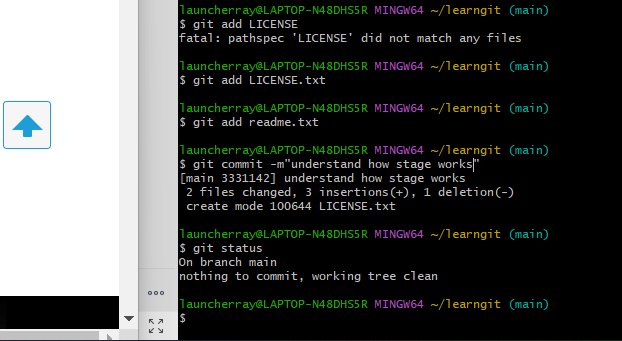


3f7ac是版本号前几位

Git reflog （记录命令）



**工作区暂存区**



问题



？

**撤销修改**

Git checkout – readme.txt （）丢弃工作区修改//git restore代替

就是让这个文件回到最近一次git commit或git add时的状态。

git reset HEAD <file>可以把暂存区的修改撤销掉//git restore –staged

**删除文件**

从版本库删除

Git rm xxx

Git commit -m“”

**远程仓库**

创建SSH key

$ ssh-keygen -t rsa -C "youremail@example.com"

把本地仓库推送到github

$ git remote add origin [git@github.com:launcherray/YouAo.git](mailto:git@github.com:launcherray/YouAo.git)

$ git push -u origin master