设置用户名和地址

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

$ git config --global user.email "email@example.com"

设置Git管理仓库（英文名**repository）**

$ mkdir learngit

$ cd learngit

$ pwd

/Users/michael/learngit

 通过git init命令把这个目录变成Git可以管理的仓库

$ git init

用命令git add告诉Git，把文件添加到仓库

 $ git add readme.txt

用命令git commit告诉Git，把文件提交到仓库

-m后面输入的是本次提交的说明

 $ git **commit** -m "wrote a readme file"

运行git status命令看看结果

$ git status

用git diff这个命令看具体修改了什么内容

$ git diff readme.txt

用git log命令查看历史记录

$ git log

输出信息太多可以试试--pretty=oneline参数

$ git log --pretty=oneline

在Git中，用HEAD表示当前版本，上一个版本就是HEAD^，上上一个版本就是HEAD^^，当然往上100个版本写100个^比较容易数不过来，所以写成HEAD~100

我们要把当前版本回退到上一个版本，就可以使用git reset命令

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

Git提供了一个命令git reflog用来记录你的每一次命令

$ git reflog

git checkout -- file可以丢弃工作区的修改

命令git checkout -- readme.txt意思就是，把readme.txt文件在工作区的修改全部撤销，这里有两种情况：

一种是readme.txt自修改后还没有被放到暂存区，现在，撤销修改就回到和版本库一模一样的状态；

一种是readme.txt已经添加到暂存区后，又作了修改，现在，撤销修改就回到添加到暂存区后的状态。

总之，就是让这个文件回到最近一次git commit或git add时的状态

$ git checkout -- readme.txt

用命令git reset HEAD <file>可以把暂存区的修改撤销掉（unstage），重新放回工作区

$ git re**set** HEAD readme.txt

用rm直接在文件管理器中把没用的文件删除

$ rm test.txt