## 创建版本库

Cd到一个文件夹后，用git init命令即可把该文件夹变成git可管理的仓库(repository)  
$ git init

Initialized empty Git repository **in** /Users/michael/learngit/.git/

Git只能跟踪文本文件的改动，不能跟踪二进制文件的内容的变化，但是依然可以把二进制文件放入git仓库中进行管理，只是不知道每次二进制文件做了哪些改动，仅此而已。但是，经过实践，我惊奇地发现，git 已经可以支持word文件内容变动显示了！Fantasy!

## 文件修改与提交

git add <files>：将files添加到暂存区中，为commit命令做准备，这是commit命令的准备动作；

git commit -m “msg”：前之前add进暂存区中的files往master分支上提交更改；

## 仓库状态

git status:显示工作区的状态

git diff:查看文件修改内容

## 时光穿梭机

HEAD指针指向当前版本，上个版本是HEAD^，上上个版本是HEAD^^，再往上100个版本写做HEAD~100

git log:查看版本commit的历史记录，最关系的是commit id版本号；

git reset –-hard HEAD^: 回退到上一个版本

git reset –hard 1094a: 回到1094a对应的版本

当不记得版本号时，可以用：

git reflog：显示每一次commit命令。

## 管理修改

Git的先进之处在于，它不像其他版本管理工具一样管理的是文件，它跟踪并管理的是修改。

如果不用git add添加到暂存区，它就不会被commit命令提交。

## 撤销修改

场景1：git checkout – file：修改了工作区（文件夹）中的某个文件，想直接丢弃修改时，用此命令即可

场景2：git reset HEAD <file>: 修改了工作区（文件夹）中的某个文件并且add到了暂存区，想丢弃修改，可使用此命令回到场景1，然后按场景1中的命令操作；

场景3：已经commit到了仓库，想撤销本次提交，可以使用版本回退，但前提是没有推送到远程库

## 删除文件

git rm “test.txt”: 当删除工作区中的某个文件之后，如果想把仓库里的文件也删掉，可以用此命令；如果是误删文件，可以用git checkout – test.txt将版本库里的文件拉回工作区中