//创建目录

$ mkdir learngit

$ cd learngit

//显示当前目录

$ pwd

//把这个目录变成Git可以管理的仓库：

$ git init

//创建readme文件添加到仓库

$ git add readme.txt

//把文件提交到仓库

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

//可一次提交多个文件

$ git add file1.txt

$ git add file2.txt file3.txt

$ git commit -m "add 3 files."

//查看历史记录

$ git add file1.txt

$ git add file2.txt file3.txt

$ git commit -m "add 3 files."

//减少输出内容

$ git log --pretty=oneline

//回退至上一版本

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

//查看文件内容

$ cat readme.txt

//直接写出指定版本号,不必写全

$ git re**set** --hard 1094a

//git reset命令既可以回退版本，也可以把暂存区的修改回退到工作区

//查看命令历史

$ git reflog

//查看状态

$ git status

//丢弃工作区的修改，让这个文件回到最近一次git commit或git add时的状态。

$ git checkout -- readme.txt

//删除文件

$ rm test.txt

//从版本库中删除该文件

$ git rm test.txt

$ git **commit** -m "remove test.txt"

**/**/拉取远程仓库

$ git remote add origin [git@github.com:github账户名/learngit.git](mailto:git@github.com:github账户名/learngit.git)

//把本地库的内容推送到远程库上

$ git push -u origin master

//从现在起，只要本地作了提交，就可以通过命令：

$ git push origin master

//从远程仓库克隆到本地

$ git clone [git@github.com:michaelliao/gitskills.git](mailto:git@github.com:michaelliao/gitskills.git)

//创建dev分支，然后切换到dev分支

$ git checkout -b dev

Git checkout 加上-b参数表示创建并切换，相当于

$ git branch dev//创建分支

$ git checkout dev//切换分支

//查看当前分支

$ git branch

//合并指定分支到当前分支

$ git merge dev

//删除指定分支

$ git branch -d dev

//查看解决冲突

$ git log --graph --pretty=oneline --abbrev-commit