# 1.创建git仓库

$ mkdir learngit

$ cd learngit

$ pwd

/Users/michael/learngit

$ git init

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

# 2.将文件添加进本地仓库暂存区

touch hello.c

git add hello.c

# 3.将本地仓库暂存区提交到当前分支

$ git commit -m "wrote a readme file 这个是说明"

[master (root-commit) cb926e7] wrote a readme file

1 file changed, 2 insertions(+) //这里指的是一个文件被改动，添加了两行

create mode 100644 readme.txt

# 4.查看本地仓库状态

$ git status

# On branch master

nothing to commit (working directory clean)

# 5.查看版本历史

## 1.查看详细信息

$ git log

commit 3628164fb26d48395383f8f31179f24e0882e1e0

Author: Michael Liao <askxuefeng@gmail.com>

Date: Tue Aug 20 15:11:49 2013 +0800

append GPL

commit ea34578d5496d7dd233c827ed32a8cd576c5ee85

Author: Michael Liao <askxuefeng@gmail.com>

Date: Tue Aug 20 14:53:12 2013 +0800

add distributed

## 2.查看简单信息

$ git log --pretty=oneline

3628164fb26d48395383f8f31179f24e0882e1e0 append GPL

ea34578d5496d7dd233c827ed32a8cd576c5ee85 add distributed

cb926e7ea50ad11b8f9e909c05226233bf755030 wrote a readme file

# 6.回退版本

## 1.回退上一个版本

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

HEAD **is** now **at** ea34578 **add** distributed

## 2.回退到指定版本号

$ git re**set** --hard 3628164

HEAD **is** now **at** 3628164 append GPL

# 7.查看每次提交命令历史

$ git reflog

ea34578 HEAD@{0}: reset: moving to HEAD^

3628164 HEAD@{1}: commit: append GPL

ea34578 HEAD@{2}: commit: add distributed

cb926e7 HEAD@{3}: commit (initial): wrote a readme file

# 8.在本地仓库添加一个远程仓库

$ git remote add origin git@github.com:michaelliao/learngit.git

# 9.将本地分支推送至服务器

$ git push origin master //origin是服务器的默认名字，master是本地分支名字

# 10.将服务器工程拷贝到本地

$ git clone [git@github.com:michaelliao/gitskills.git](mailto:git@github.com:michaelliao/gitskills.git) //clone后面的直接就是网址

# 10.将远程最新程序更新至本地

$ git pull origin //如果已经将服务器工程拷贝到本地以后，服务器程序又更新过了，则通过这个命令将最新的更新到本地，origin是服务器的默认名字

# 11.创建分支

$ git branch dev //创建了一个dev分支，默认分支为master

# 12.切换分支

$ git checkout dev //切换到dev分支下

Switched to branch 'dev'

# 13.查看当前分支

$ git branch

\* dev //当前所在分支

master //剩下的其他的分支

# 14.合并分支

$ git merge dev //将分支dev合并到当前的分支来

Updating d17efd8..fec145a

Fast-forward

readme.txt | 1 +

1 file changed, 1 insertion(+)

# 15.删除分支

$ git branch -d dev

Deleted branch dev (was fec145a).

# 16.获取服务器分支并与本地合并

$ git pull

# 17.打一个标签

$ git tag v1 //将当前commit后的程序打上标签v1

# 18.查看标签

$ git tag //查看当前程序打了那些标签

# 18.切换到指定tag的位置

$ git checkout tag //这里的tag是本地的版本号