Git基本说明： git 有本机 缓存区 远端

git commit -am "修改的记录注视"

这个命令只会把本机的修改提交到缓存区（stage）,但是不会提交到远端。注意本命令仅仅是将缓存所在的分支提交到远端，和本机无关，如果缓存stage不是本机最新，此命令不会提醒。

git push

这个。命令一般和git commit 一起使用git push 将当前最新的缓存上传到远端,

git log

多次使用git commit -am ""之后，可以通过git log 查看提交记录。

git checkout

多次使用git. Commit -am 但是没有git push ,如果想恢复到之前git commit 的版本。 可以使用git log查看版本号，然后

git checkout d6c287483827fa8be9879b915441502ad57e6ab2

这样就切回到之前的分支

注意：执行上述操作后，在使用git log 只能看到当前版本之前git commit 记录。 如果想看所有的需要git check master 切回到缓存区的主分支。

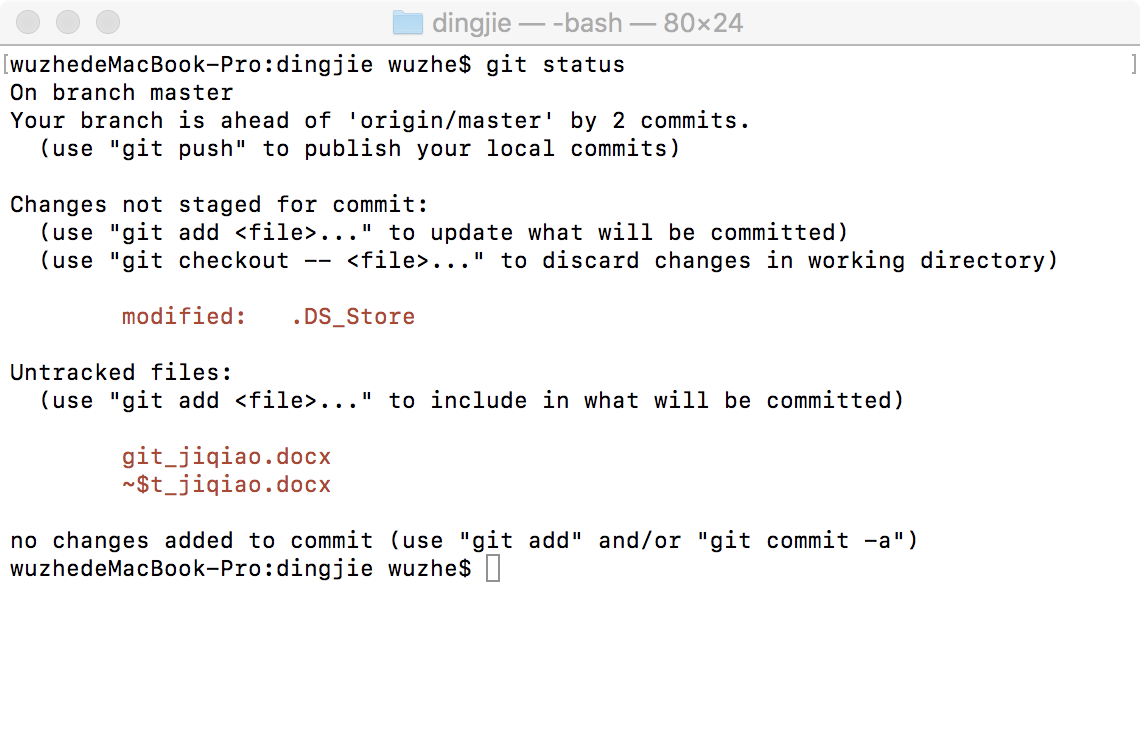
git status

如果想看当前git情况，可以使用git status.

On branch 当前的分支

Changes not staged for commit 当前本机的修改和当前分支的不同（注意此处的当前分支指的是缓存中的分支，并不是远端）

Untracked files. 本机新添加的文件，缓存上没有，需要添加到缓存进行跟踪



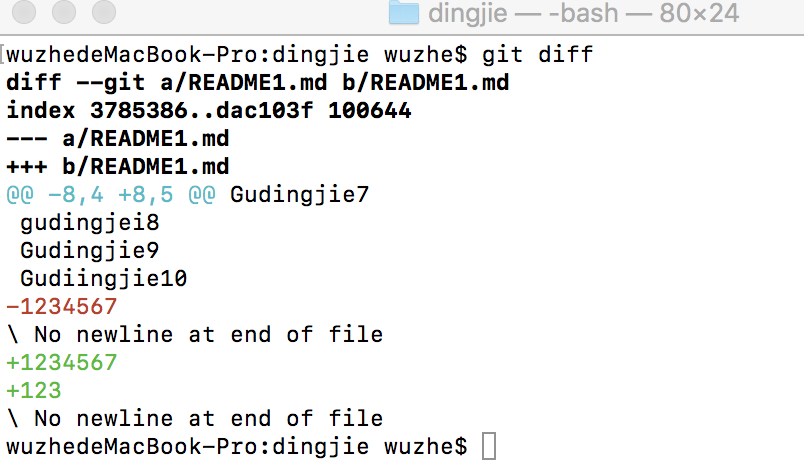
git add

git add git\_jiqiao.docx 本机git\_jiqiao.docx 会添加到缓存stage中。 然后便git. Commit 和git push了

git diff

本机修改了一些，git diff 可以看本机和所在分支的不同，所在分支既不是远端,是缓存上的文件，可能是master也可能是别的分支.

不同颜色的即是不同



Clear

终端内容很多是，可以clear. 清空终端

git stash

如果当前正在写代码，临时要切换到之前的版本，现在的正在谢的又不想上传，可以使用git stash 将当前的代码放在缓存区，之后可以在用git stash pop 回到这个版本