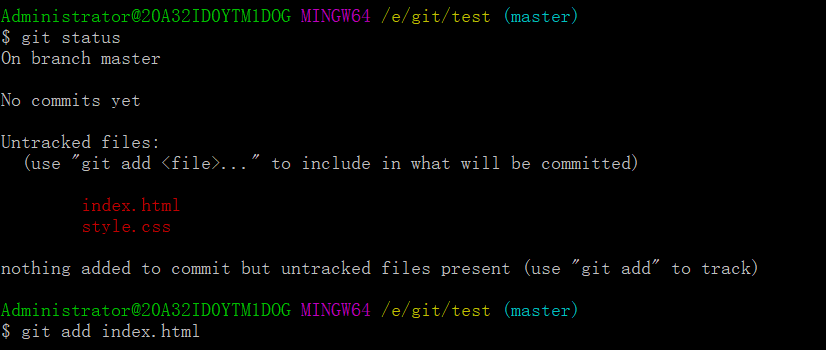
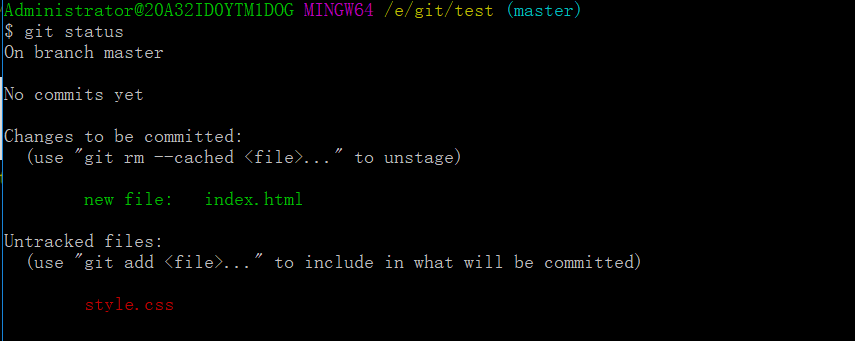
1.查看文件，初始化git仓库



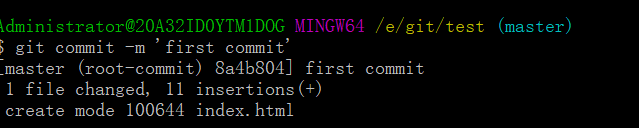
2.查看文件夹下的文件状态，添加到暂存区（追踪文件）



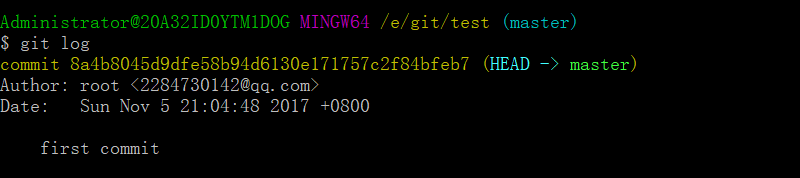
3.查看状态一个文件已到暂存区



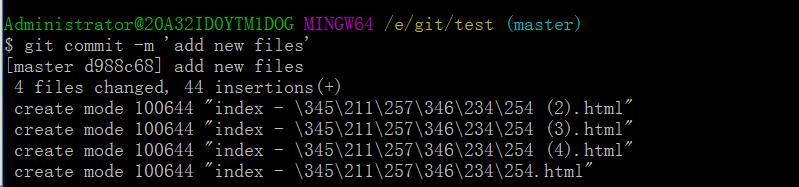
3.将暂存区文件放到版本库



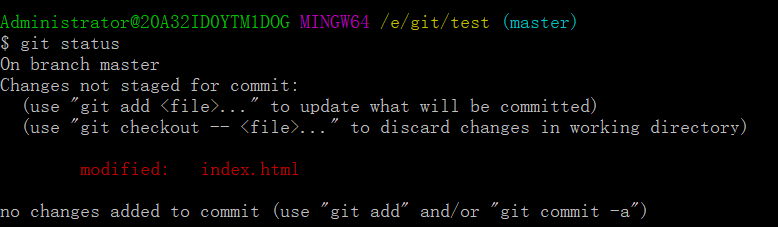
1. 查看提交的信息



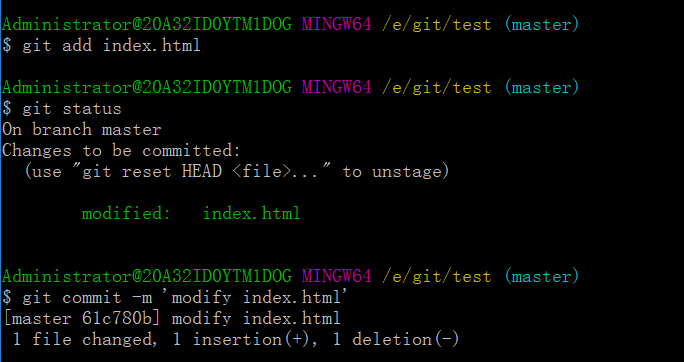
5.一次性添加很多文件



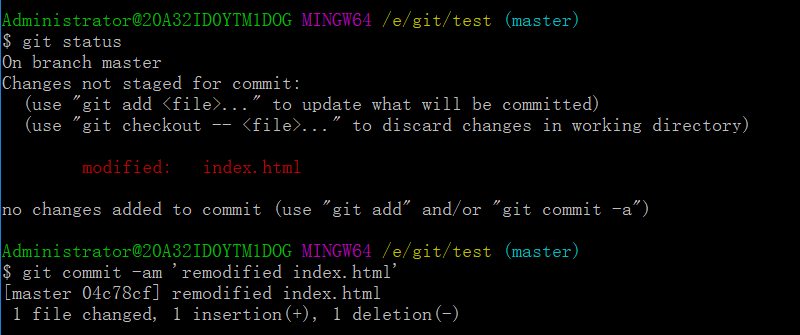
6.修改文件过后再次查看状态



7.重新添加文件到版本库并查看



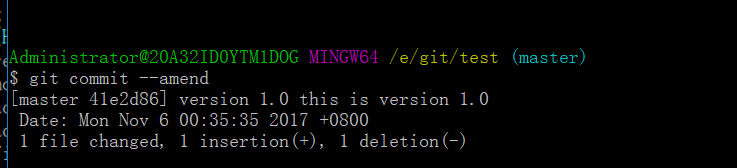
8.修改文件后一次性直接将文件从工作区提交到版本库

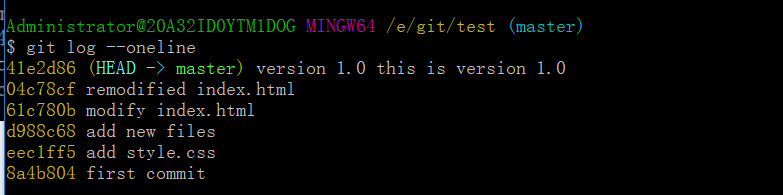


9.查看信息的简略版

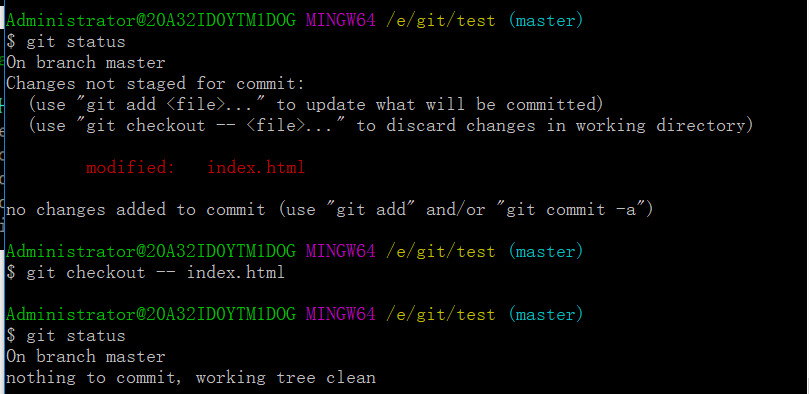


10.修改文件的版本信息

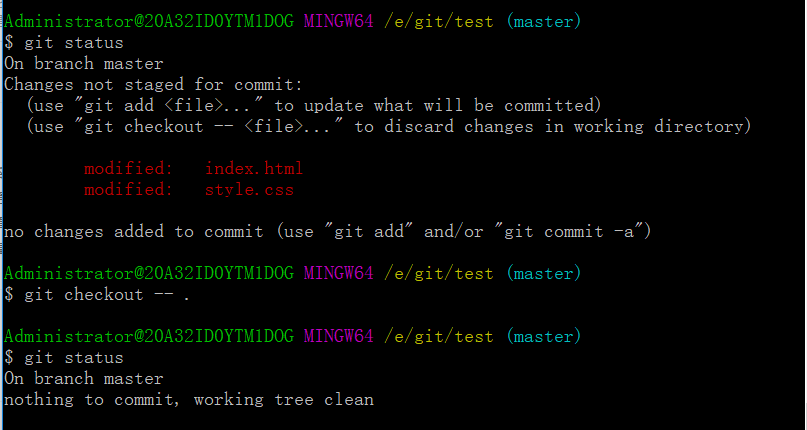




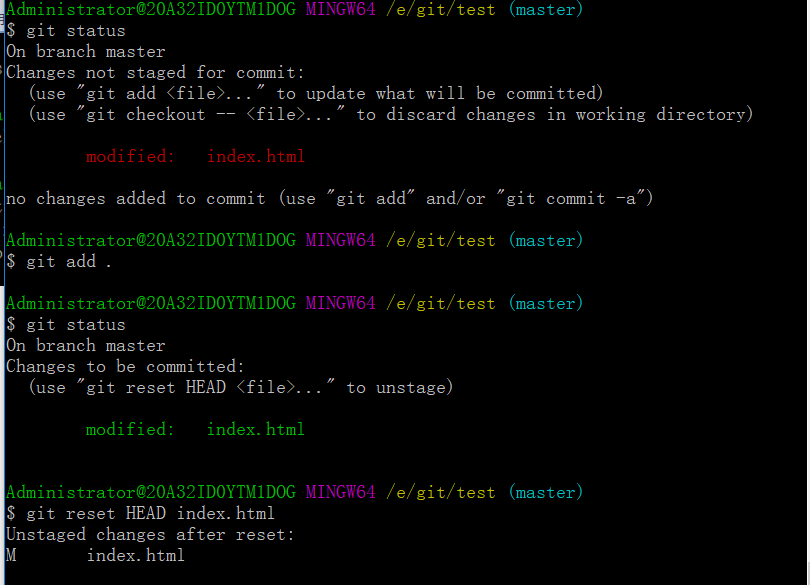
11.回退修改的工作区中的错误文件



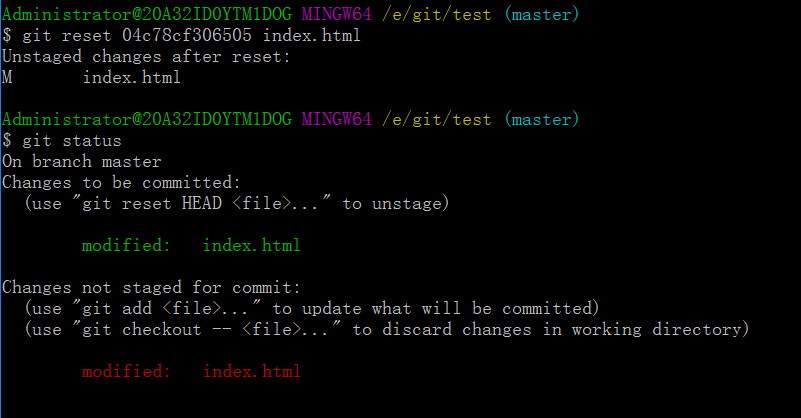
13.一次性撤销



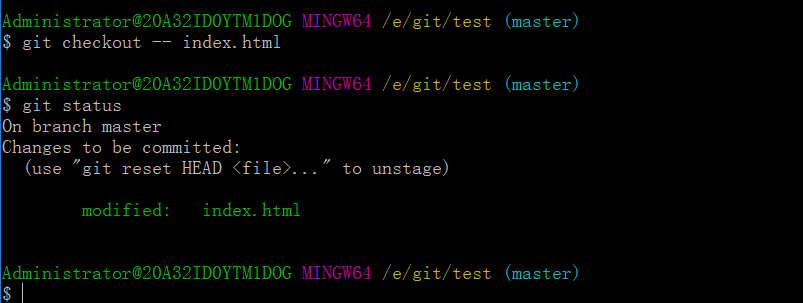
14.撤销添加到暂存区的东西



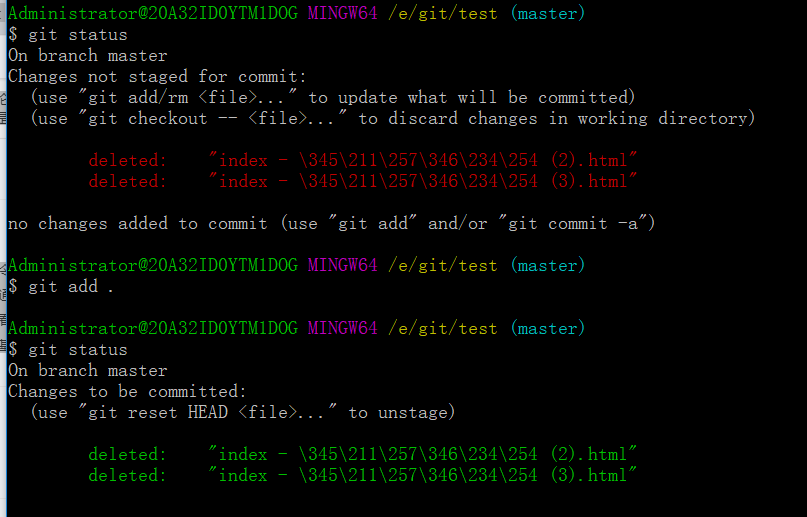
15.拉回某个版本的方法(到暂存区)以及状态



16.从暂存区拉到工作区



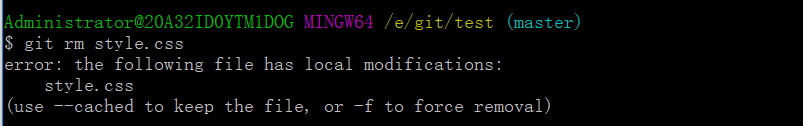
17.文件删除操作（要再向暂存提交一次）最常用



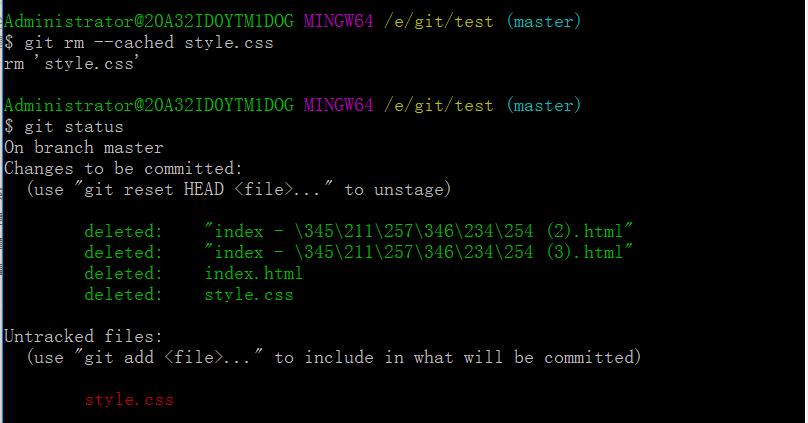
18.直接将工作区和暂存区文件删除 ./方法



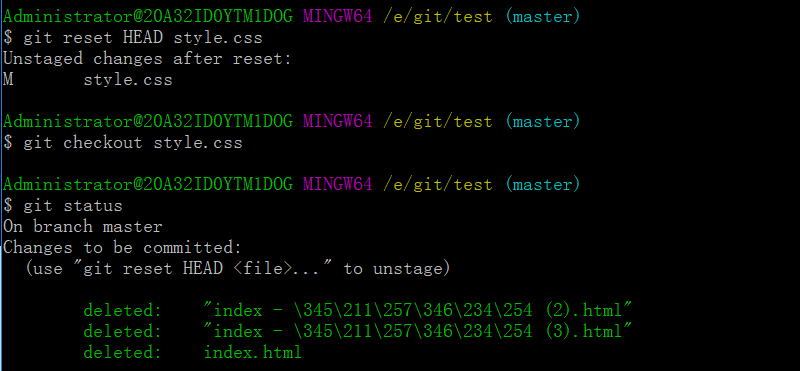
19.删除修改后的文件的方法



20.从暂存区删除文件（只会提示修改）不影响工作区



21.从版本库拉回暂存区再拉回工作区



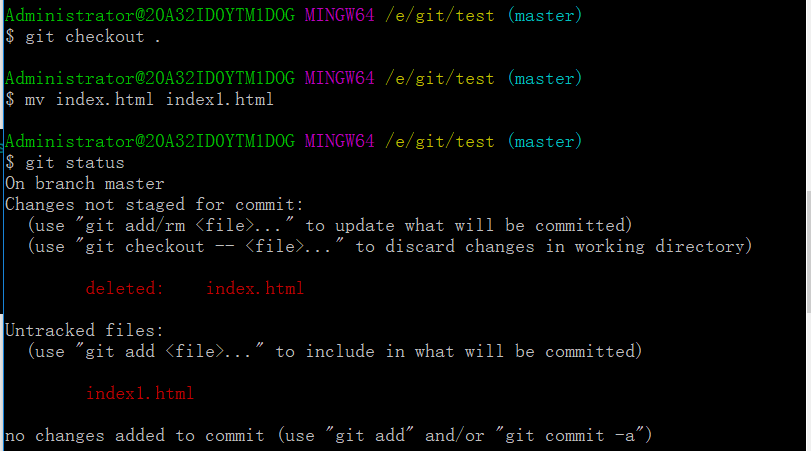
22.从工作区和暂存区一起删除



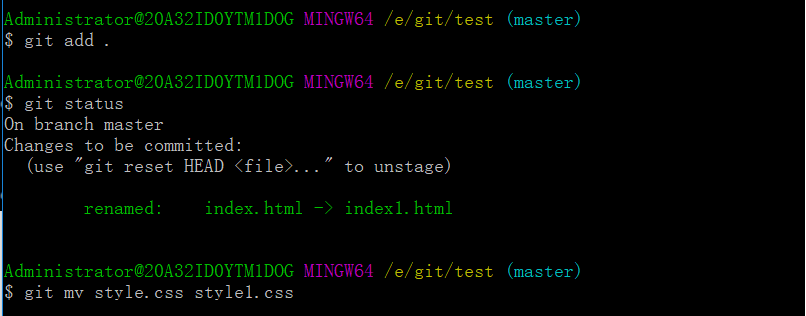
23.返回文件到工作区



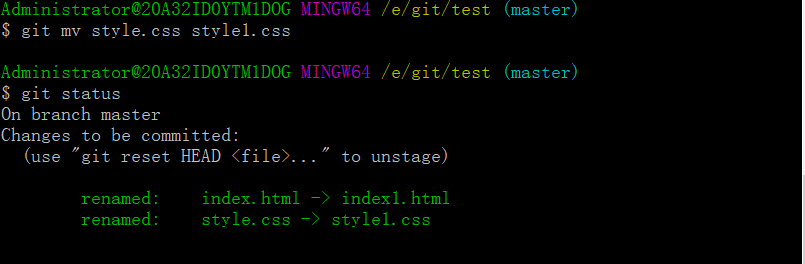
24.修改文件名字



25.将修改提交到暂存区



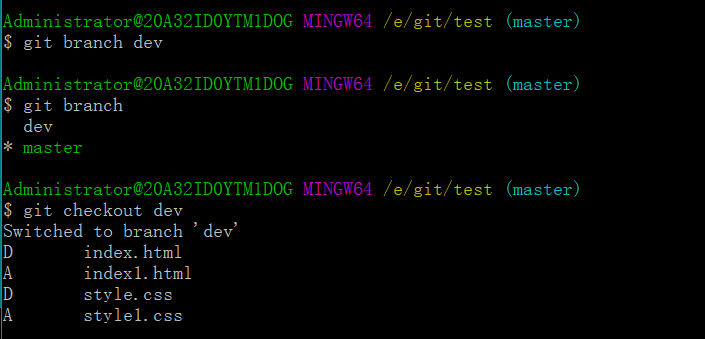
26.直接将工作区修改的提交到了暂存区

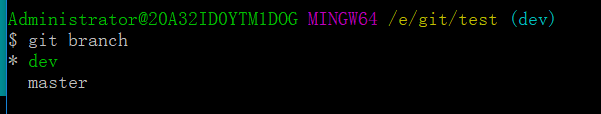


27.查看分支，有且仅有一个分支master（\*代表选中的分支）

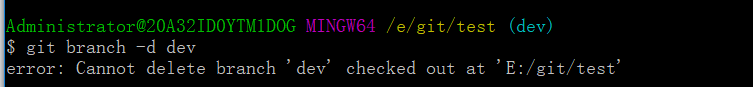


28.创建、更换分支

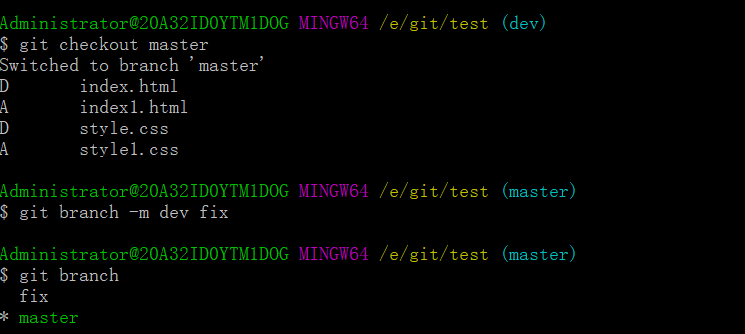




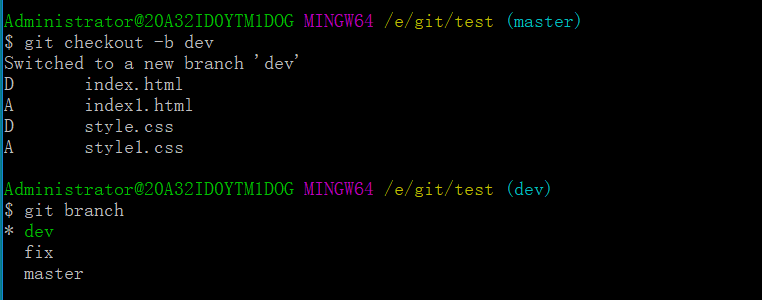
29.删除分支（不能在被选中的分支上删除那个分支，会报错，要先切换）



30.修改分支名字



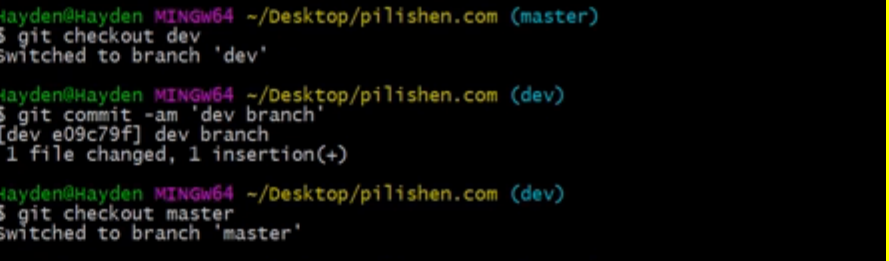
31.创建分支并切换



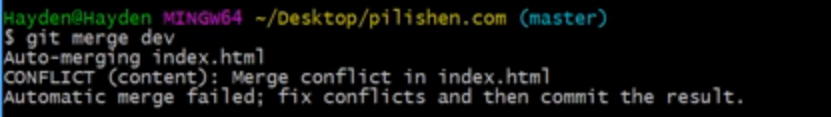
32.修改文件后提交一个新版本

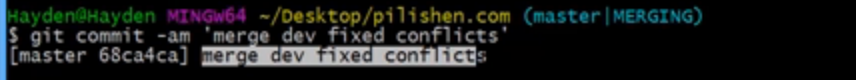


33.切换到dev分支之后（并且修改文件之后）再提交给dev分支，然后切回master

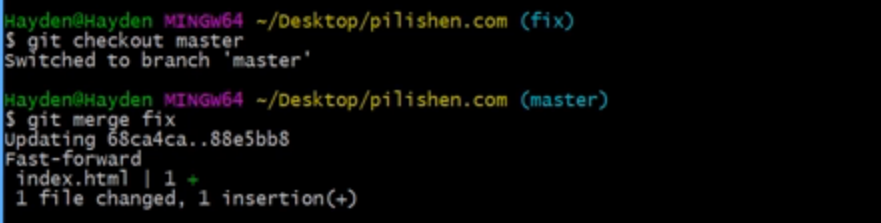


34.合并文件，修改完文件后再次提交并且写上注释

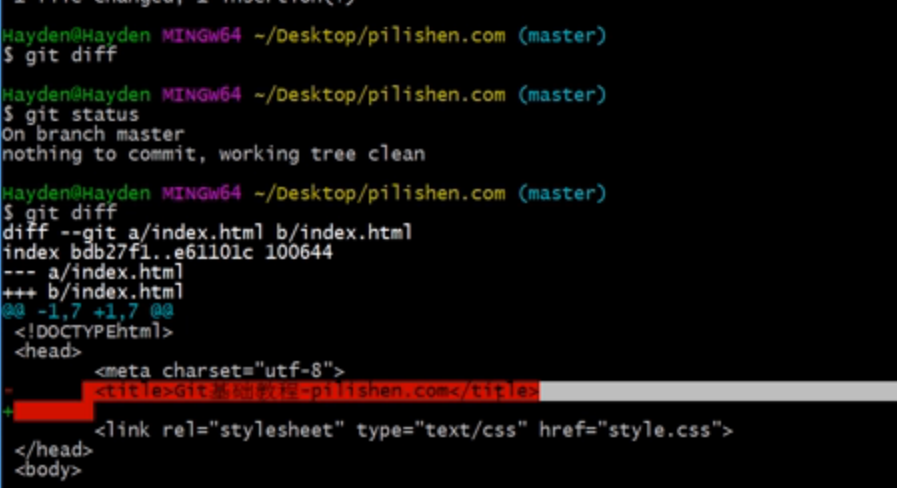




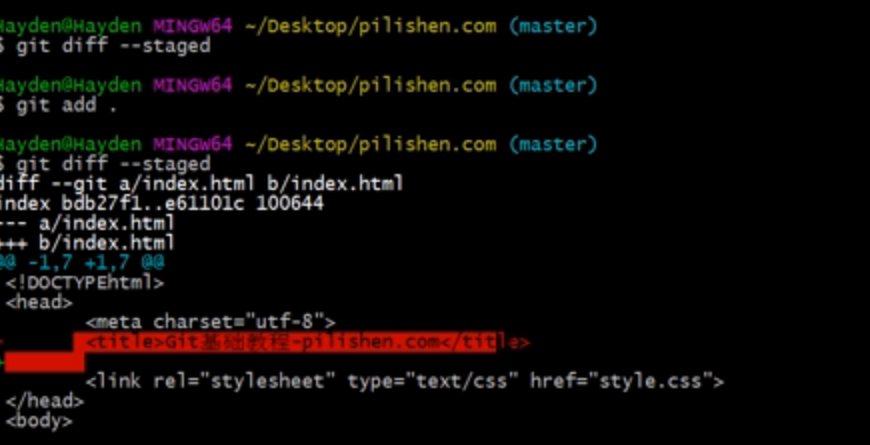
35.快进式合并（没有冲突的合并）



36.查看差异（git diff 工作区和暂存区差异）



37.工作区和版本库差异



38.比较两个版本之间的差异（两个分支也可以）

