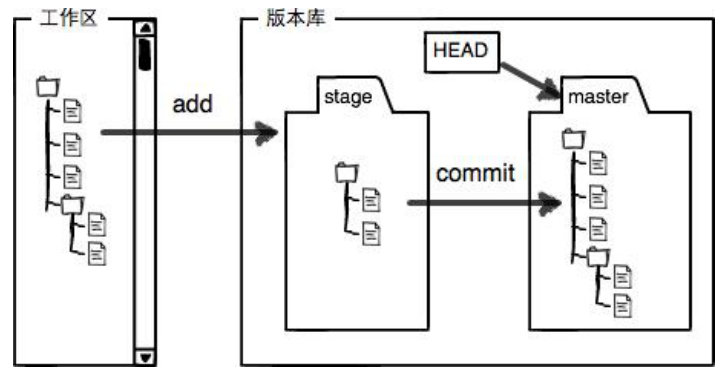
Git的执行流程如下:



版本库包括:暂存区(stage),head(指针)和分支(默认为master)

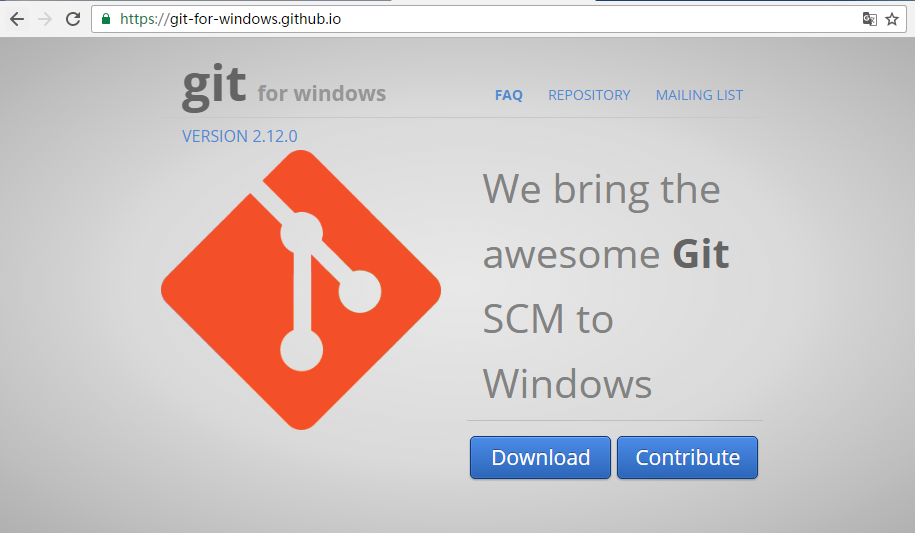
操作流程:

①在工作区编辑文件后,添加到暂存区(git add 文件名);

②将暂存区的文件提交到版本库(git commit -m “描述”);

Git 下载:

https://git-for-windows.github.io/



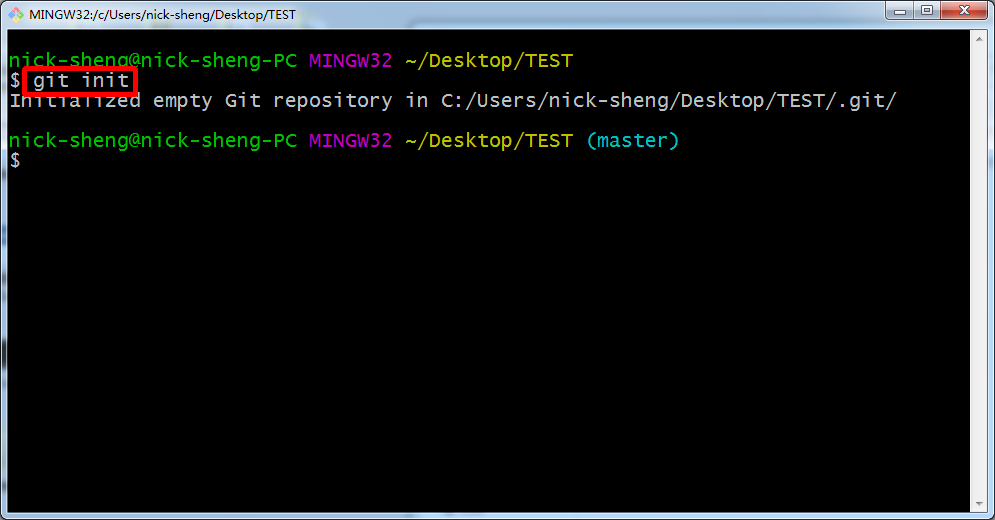
安装过程:略.(直接下一步即可)

安装完成,鼠标右击

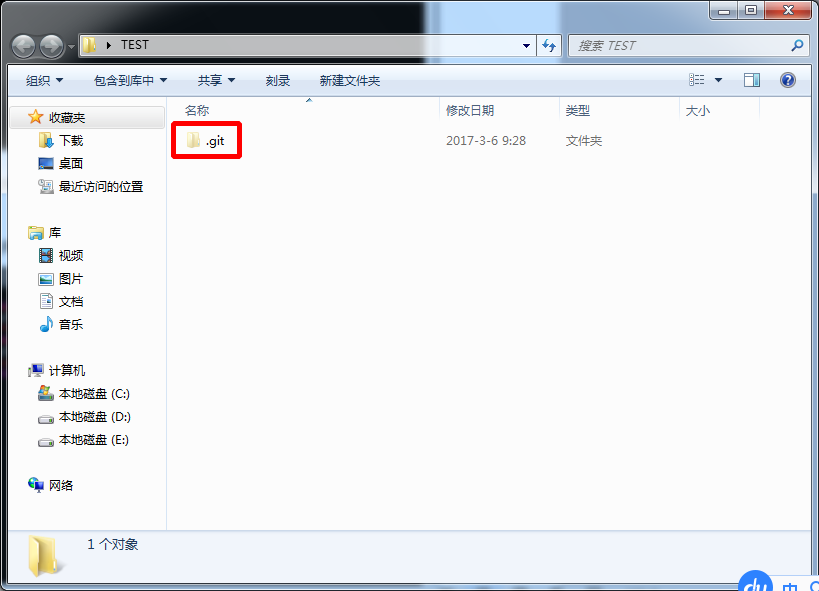


将一个目录生成版本库:

Git init



完成后,目录下会出现.git隐藏目录,即表示当前该目录就是版本库



配置用户名和邮箱(用于身份识别,即记录操作人的信息)

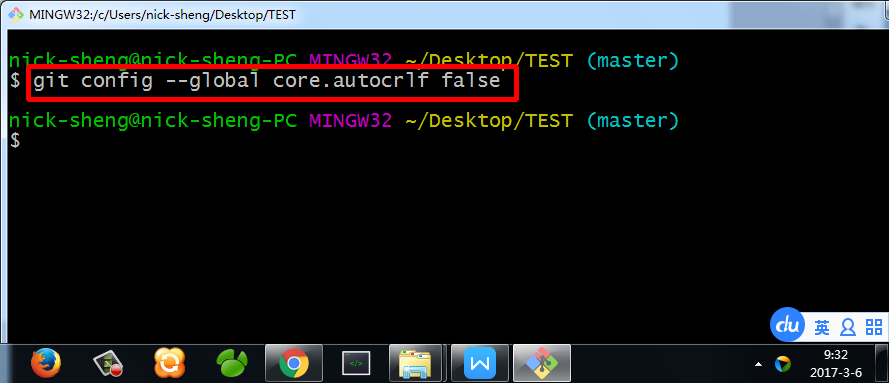
Git config --global user.name 用户名

Git config --global user.email 邮箱



Git 换行符的设置:

Git config --global core.autocrlf false



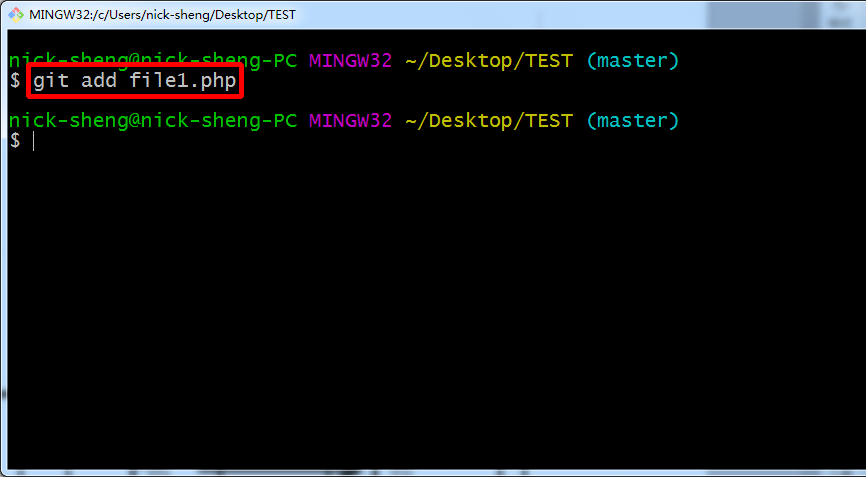
创建文件:

Vi 文件名 或 touch 文件名 i插入内容 先按Esc 再按:wq保存并退出



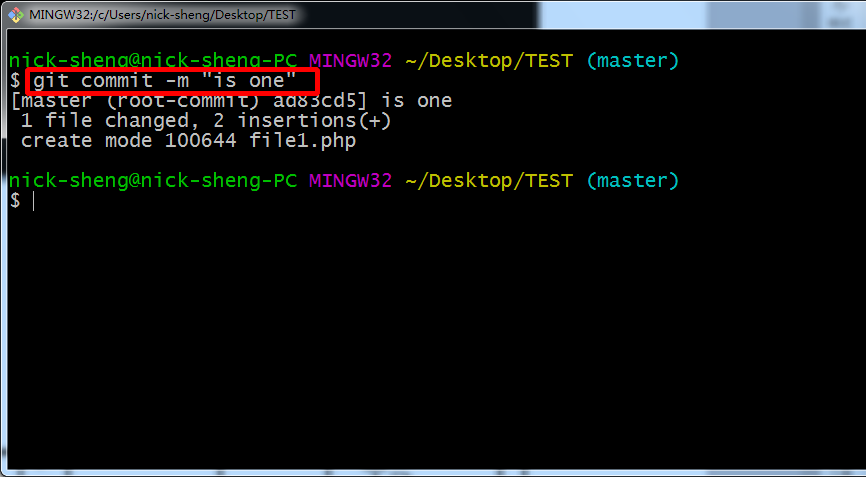
添加到暂存区

git add 文件名



添加到版本库

Git commit -m “描述”

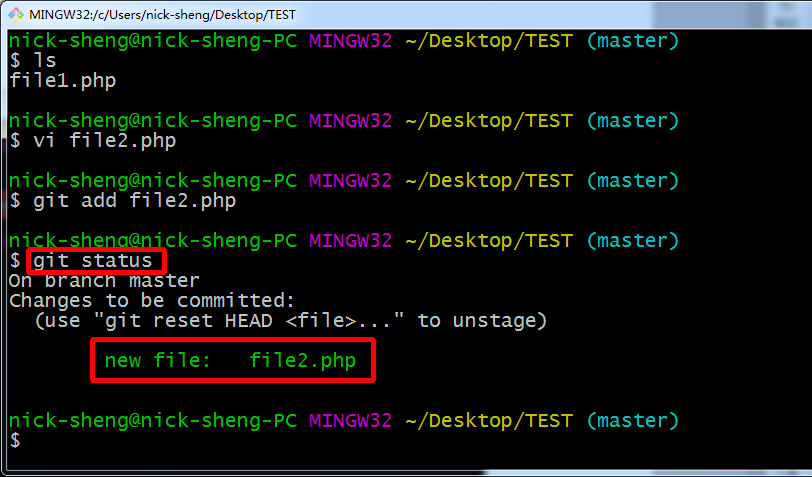


============== 以上,就是Git本地操作的流程 =============

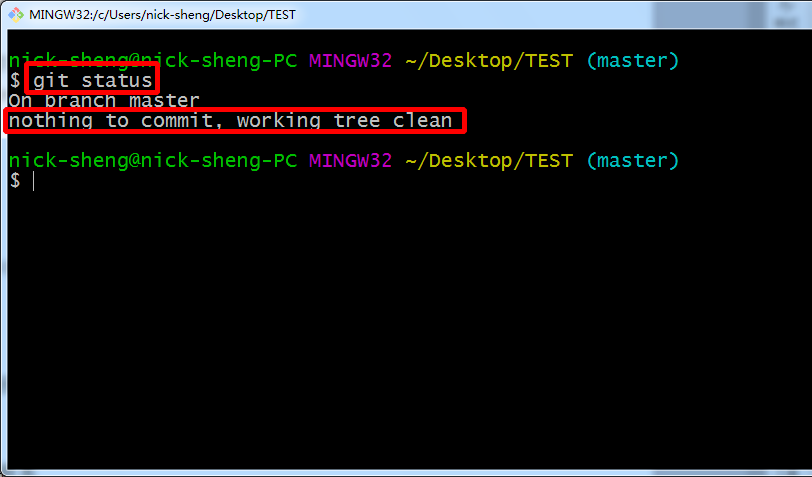
查看文件的状态(分为两种情况,1,文件已添加到暂存区;2,文件已提交到版本库)

Git status

文件已添加到暂存区

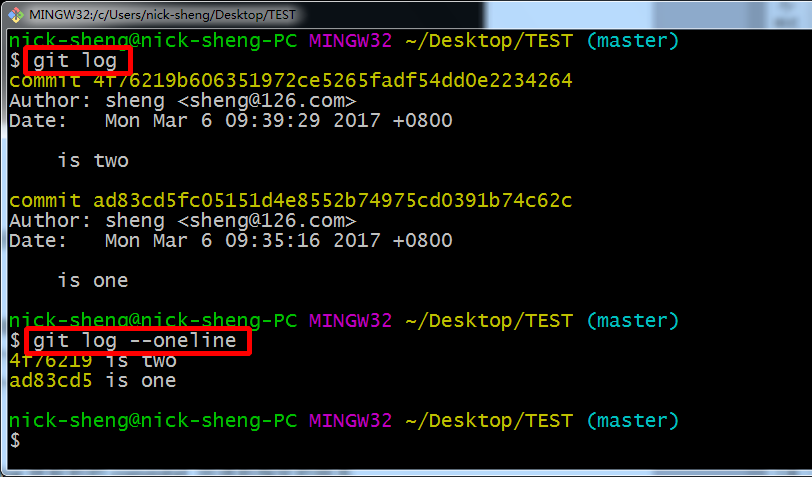


2.已提交到版本库



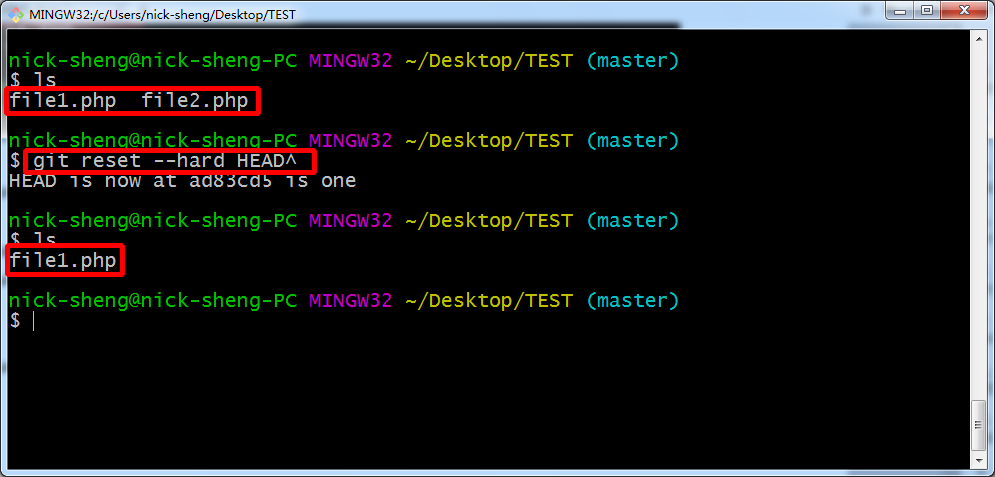
查看git日志

Git log (可加参数--oneline,即一行显示)



版本回退

Git reset --hard HEAD^ 回退上一个版本



Git reset --hard HEAD^^ 回退上两个版本

Git reset --hard HEAD~n 回退上n个版本

Git reset --hard commitid 根据git log获取到的commitid 回退到指定的版本

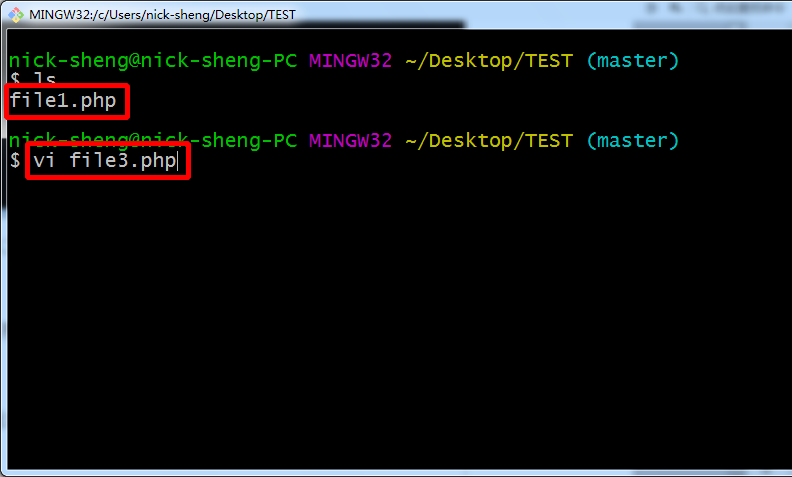
行为日志,显示所做的操作以及commitid

Git reflog



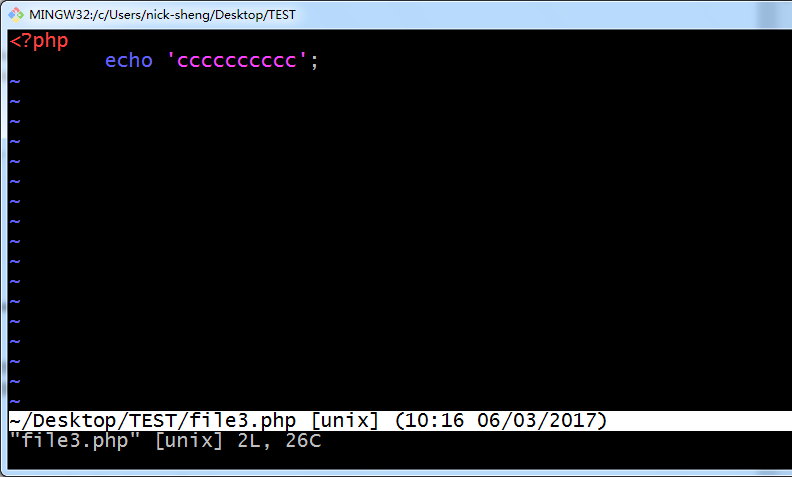
撤销修改(分为两种情况:1.文件尚未添加到暂存区;2.文件添加到暂存区)

文件尚未添加到暂存区(详解:共7步)

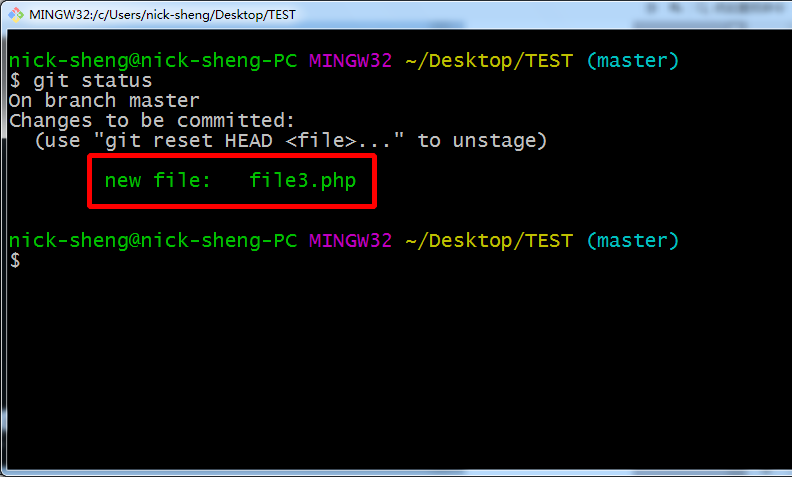


git checkout --文件名

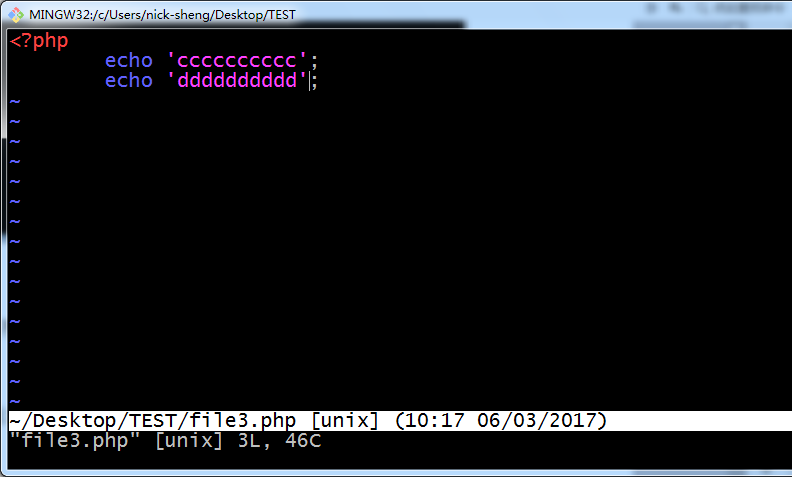
ls查看当前目录下只有一个file1.php,再创建一个file3.php)



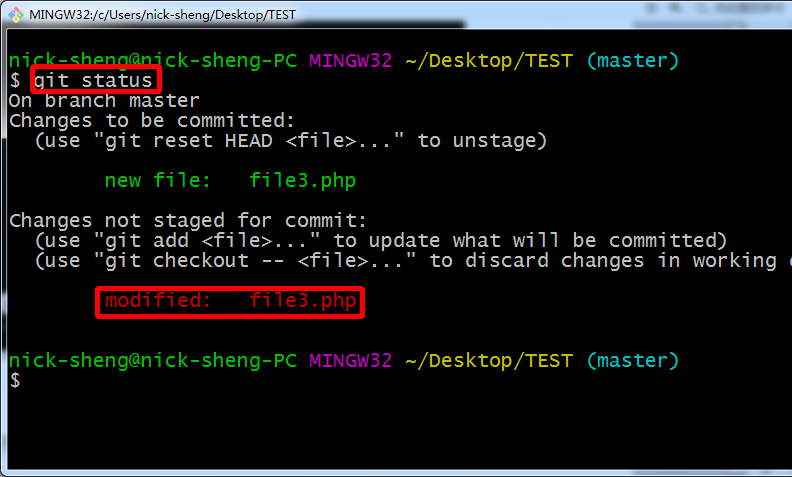
在file3.php中写入一句话)



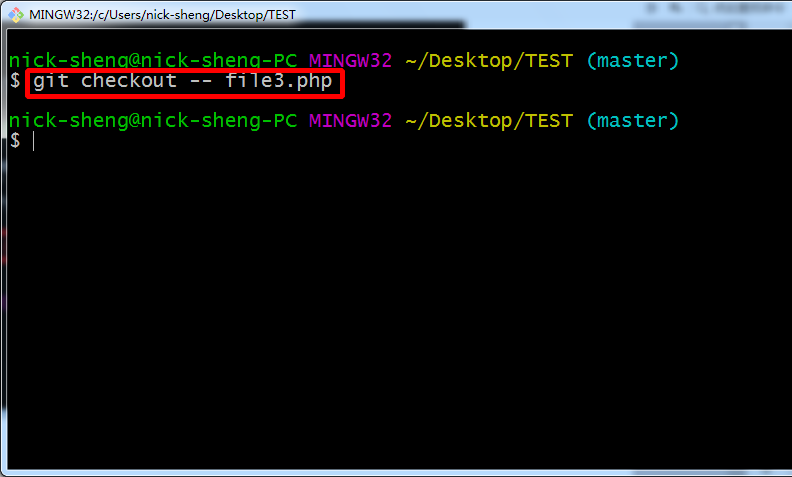
(3,查看此时file3.php的状态,new file即表示该文件还未添加到版本库)



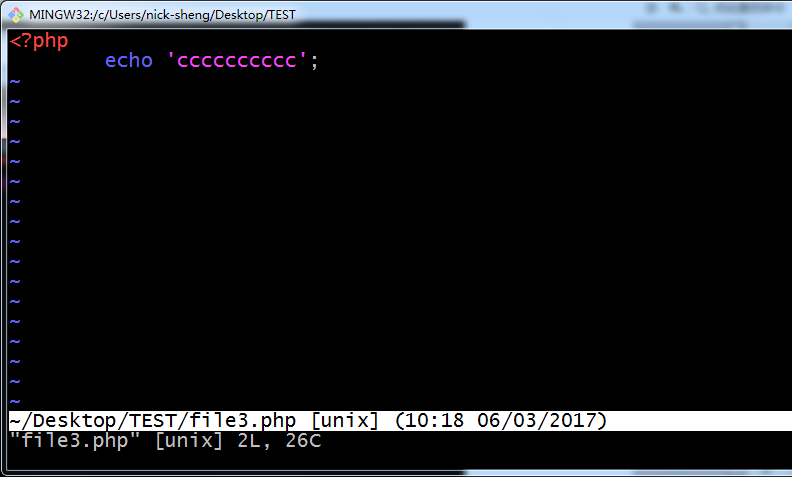
(4,此时打开file3.php,再写入一句话)



此时再查看file3.php的状态,显示modified,即表示该文件被修改了)



通过命令 git checkout -- file3.php撤销修改)



(7.打开file3.php,此时只有一开始写的cccccc了,后面写上去的ddddd已经没有了,被撤销回来了)

文件添加到暂存区

Git checkout HEAD [文件名] 如果不指定文件名,则撤回所有(撤回到工作区,如果撤回到暂存区,同上一条命令)

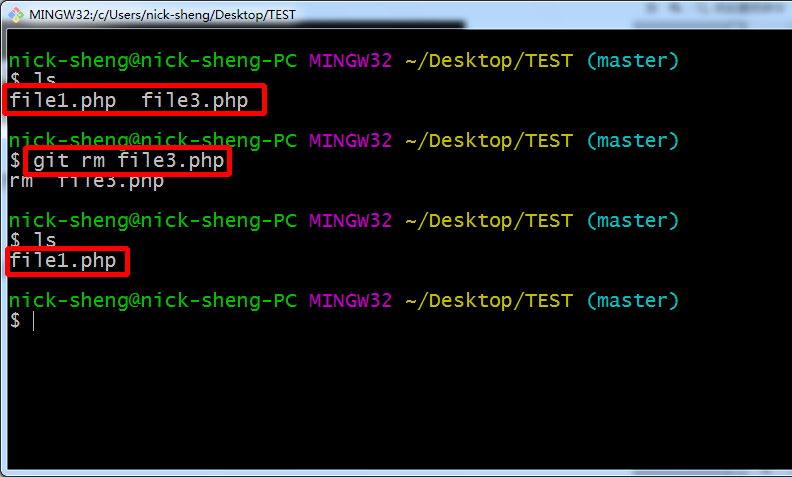
删除文件(1,删除工作区文件;2.删除版本库文件)

1,删除工作区文件

rm 文件名

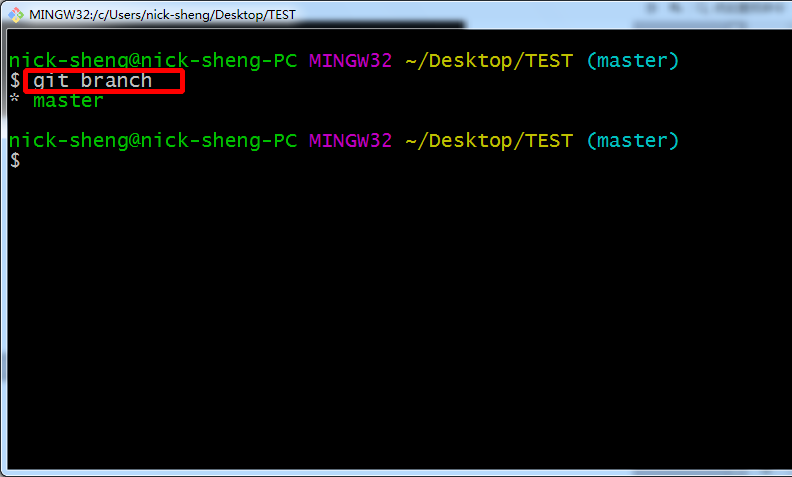
2,删除版本库文件

git rm 文件名(**删除成功后,要add和commit**)



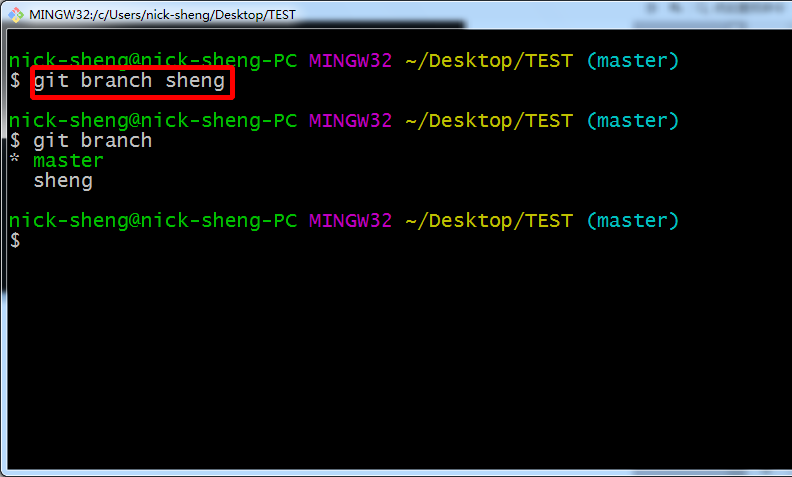
查看分支

Git branch



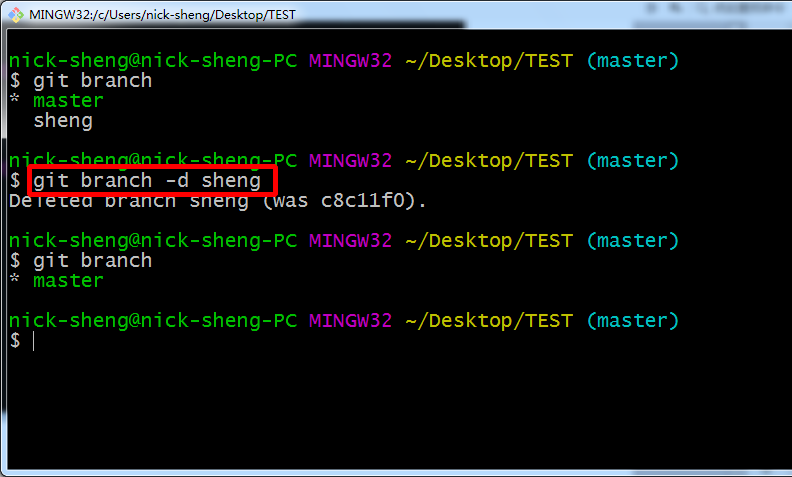
创建分支

Git branch 分支名



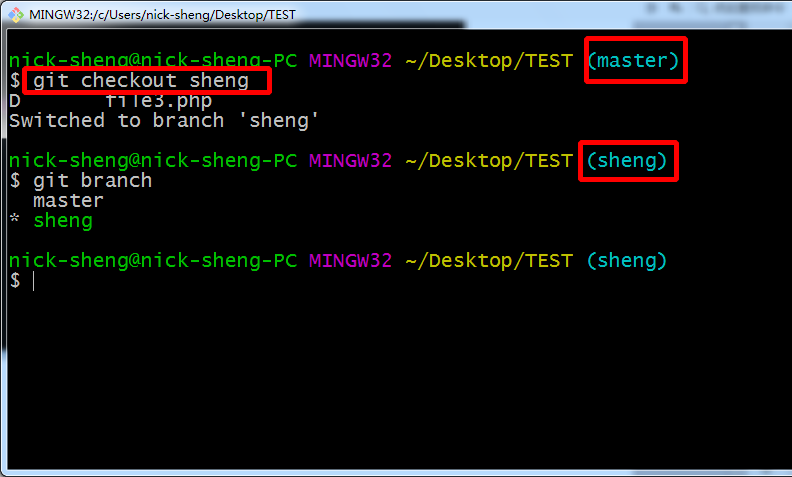
删除本地分支

Git branch -d 分支名



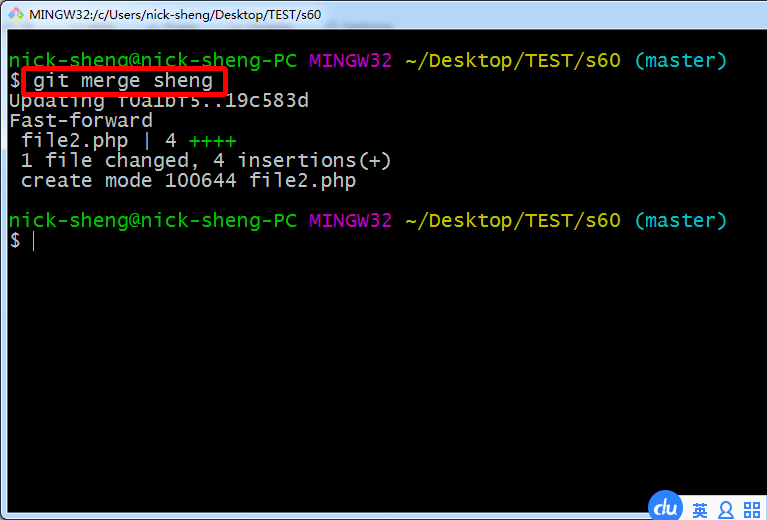
切换分支

Git checkout 分支名



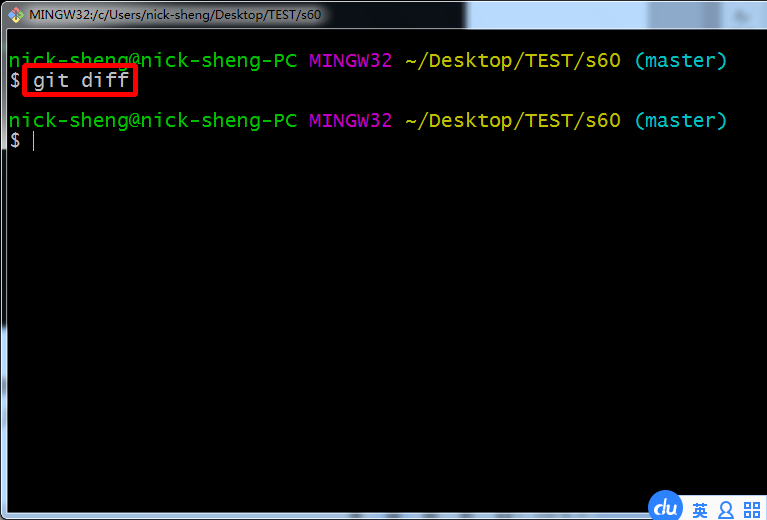
分支合并

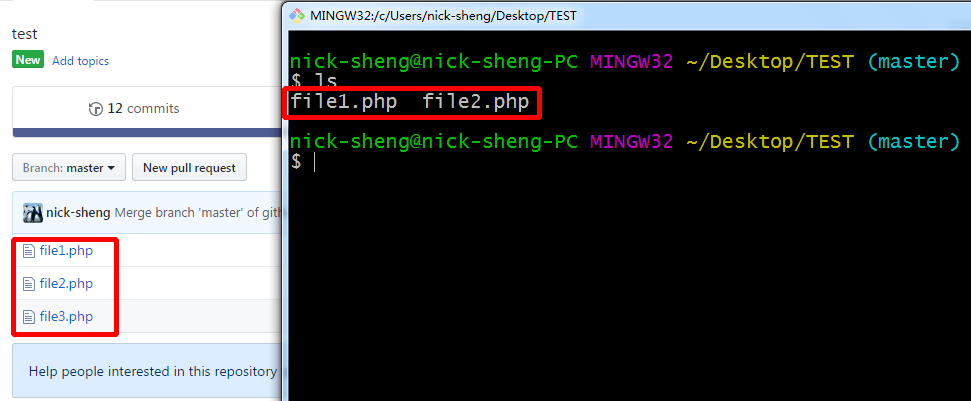
Git merge 要合并的分支名



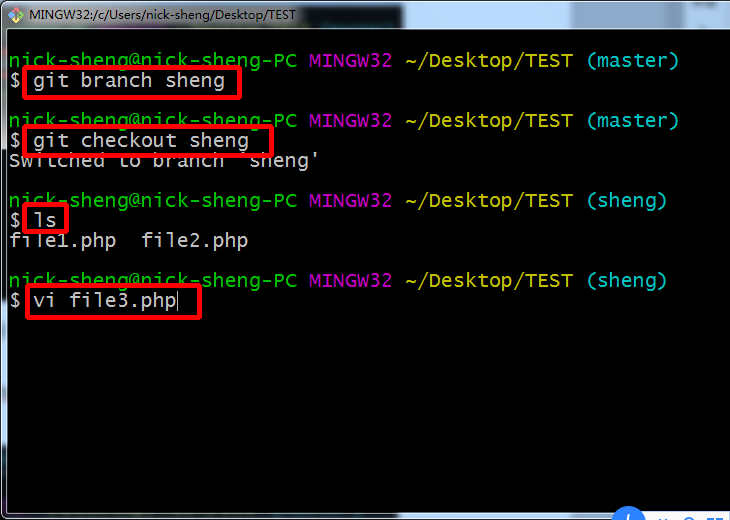
查看有无冲突(如果有,则会以列表的形式展现,通过以下15步,演示冲突及其解决办法)

Git diff

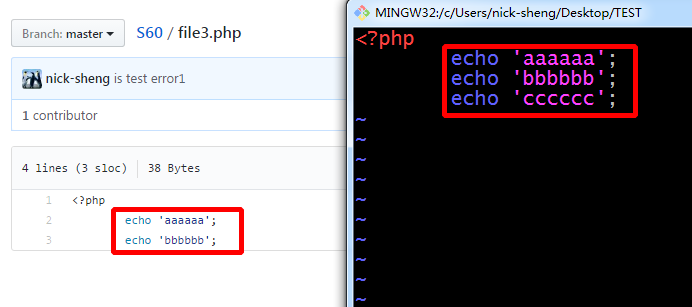




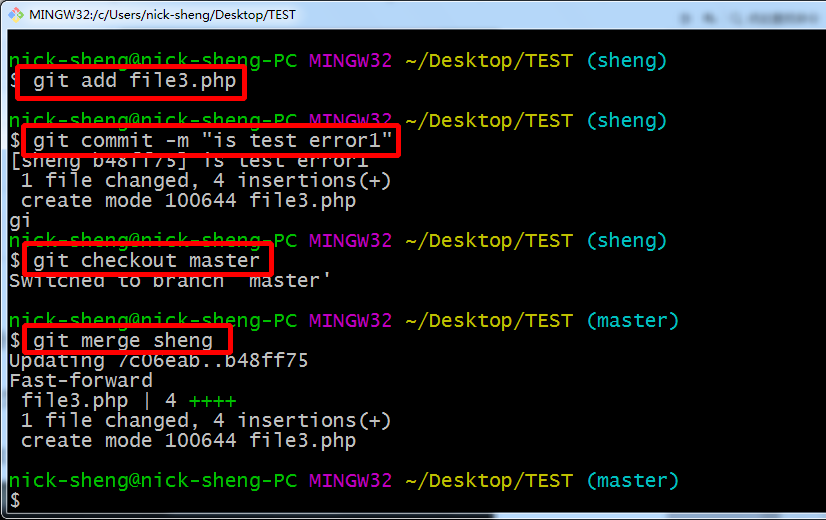
远程仓库有三个文件,而本地的仓库只有两个文件)



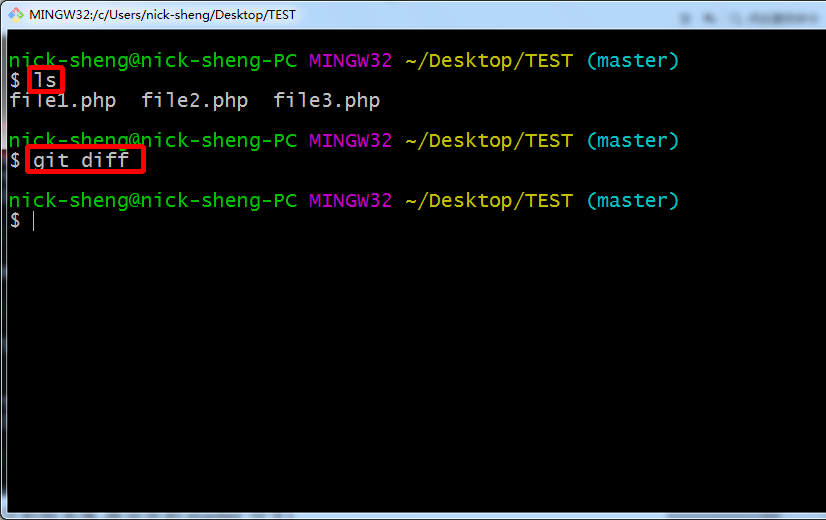
在本地仓库sheng分支下创建一个file3.php,与远程仓库文件同名)



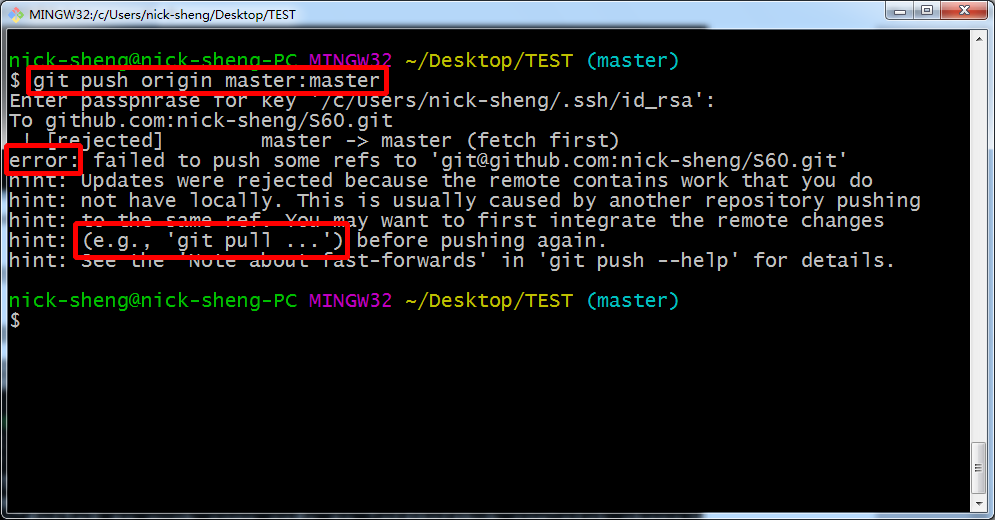
在本地file3.php中,写上与远程仓库同样的代码)



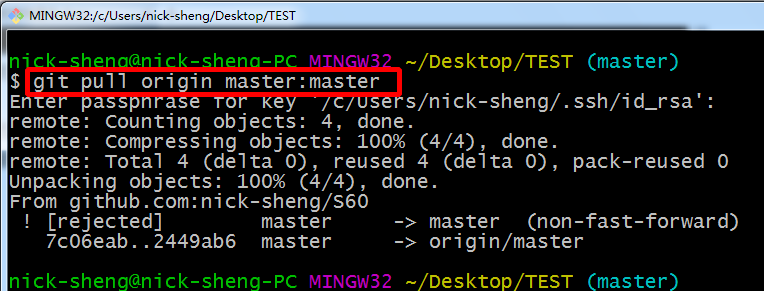
将file3.php添加到暂存区并提交到版本库,再合并到master分支)



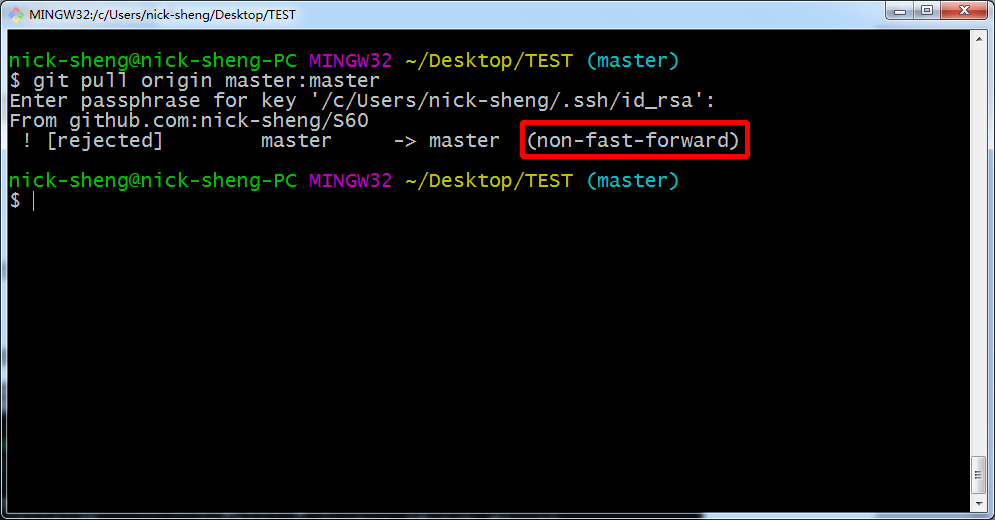
在master分支下,可以看到合并过来的file3.php;git diff,并没有冲突)



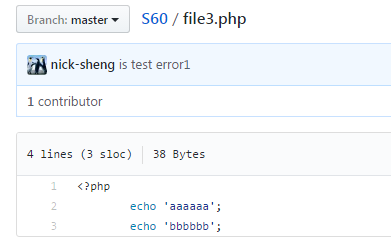
(6.提交到远程仓库,出现上图的报错信息,即有error,看e.g,需要先拉取,再提交)



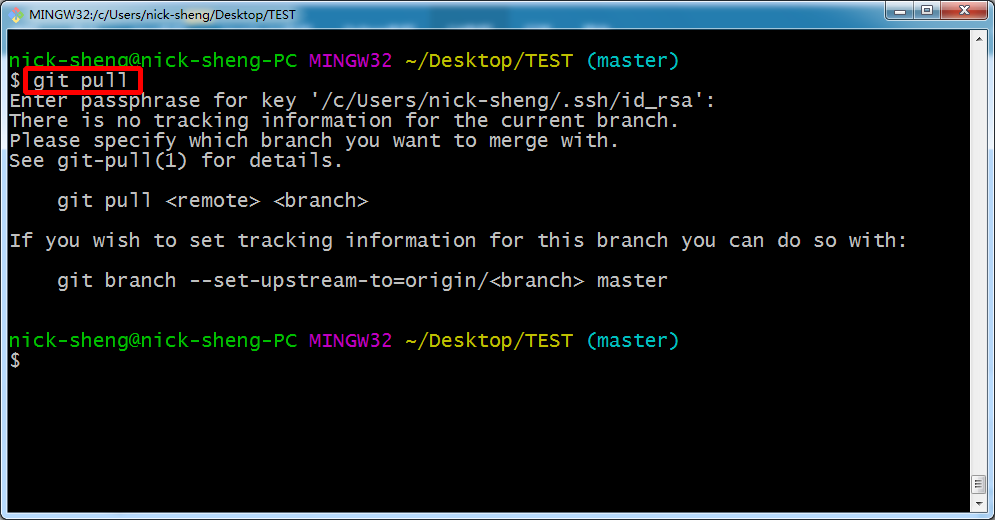
拉取远程仓库的内容到本地)



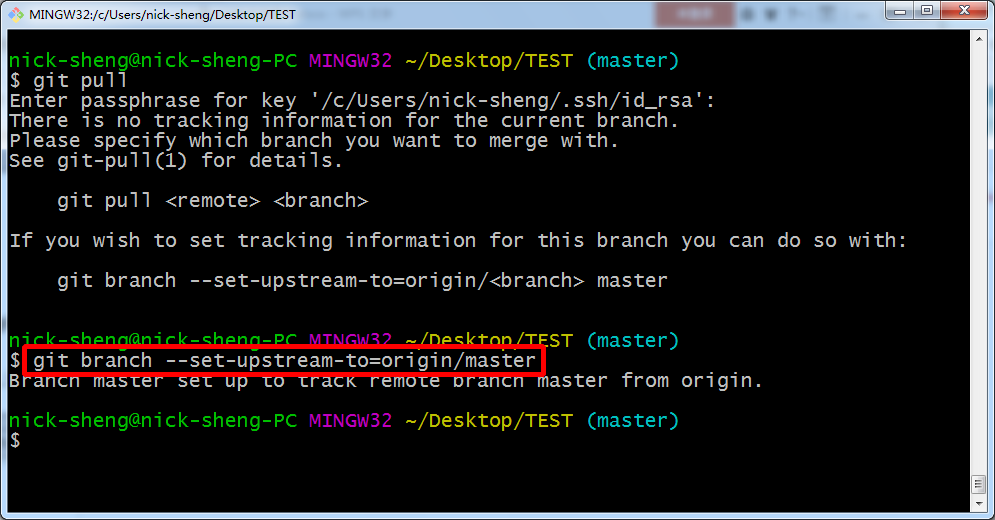
此时,提交后出现该提示,即提交不成功)



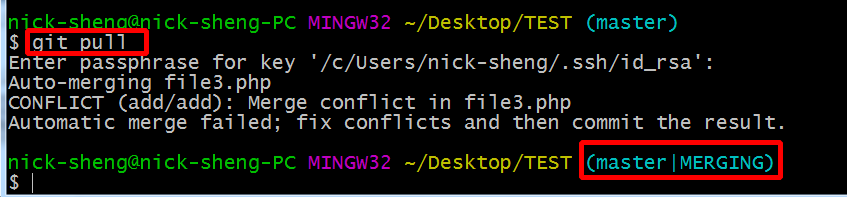
查看远程仓库上的file3.php,并没有刚刚写的ccccc这一行)



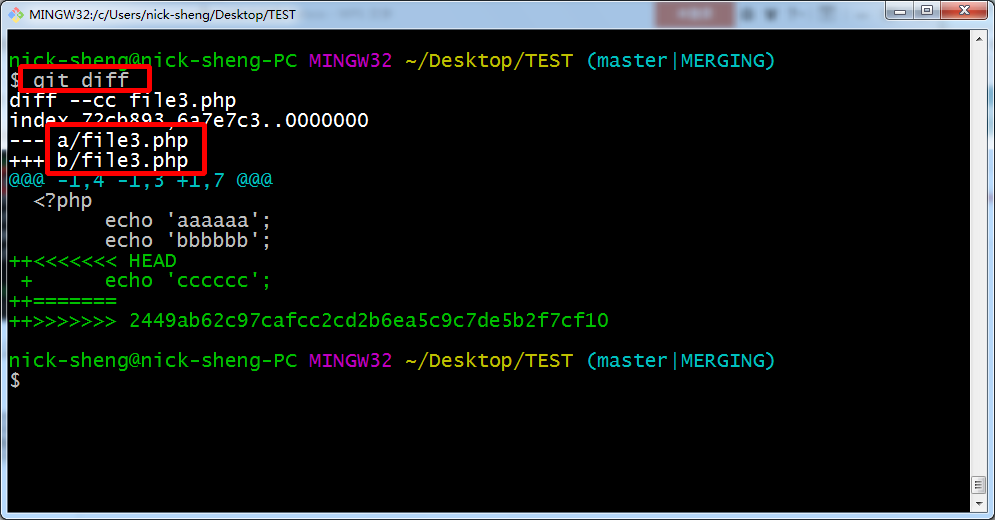
使用git pull从远程获取最新版本到本地,拉取不成功,原因:本地master分支与远程origin/master分支没有链接)



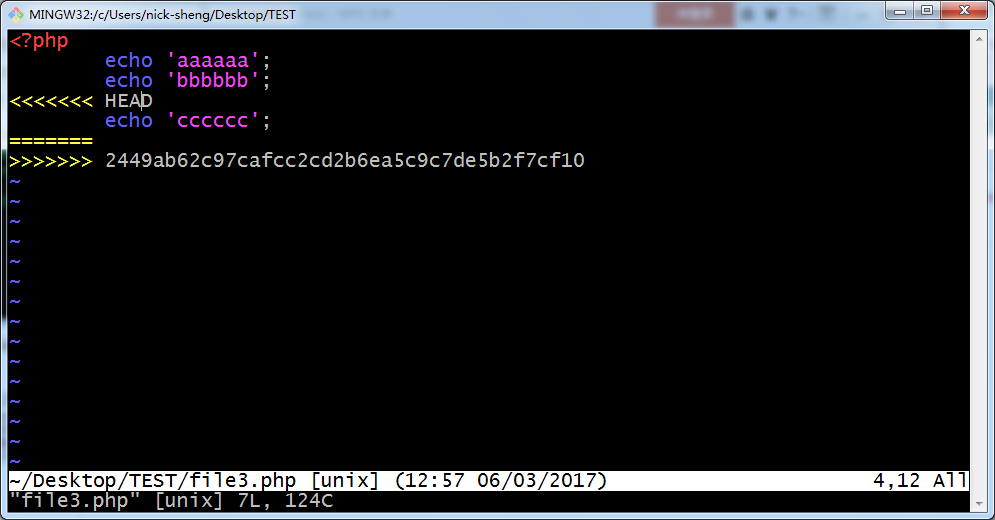
设置本地master和远程origin/master的链接)

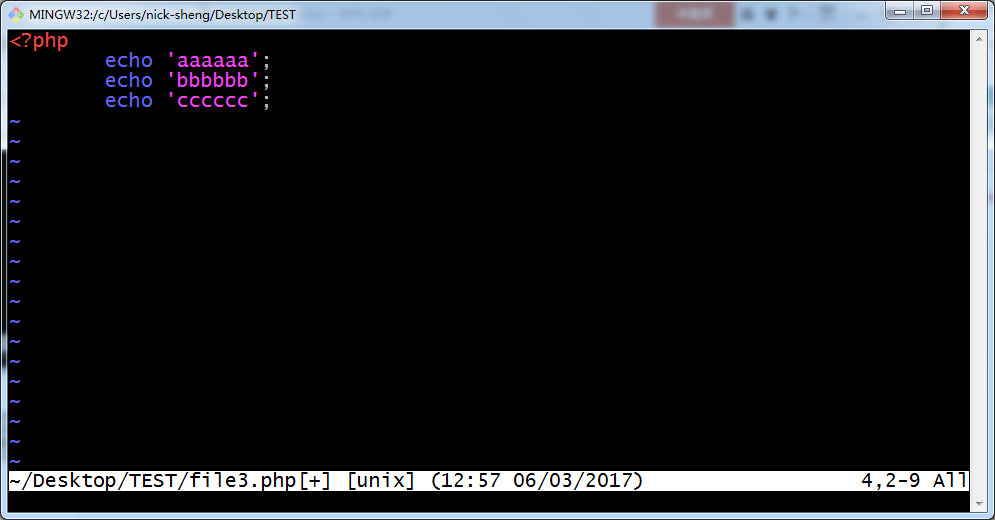


再pull,已经拉取成功,但有冲突)

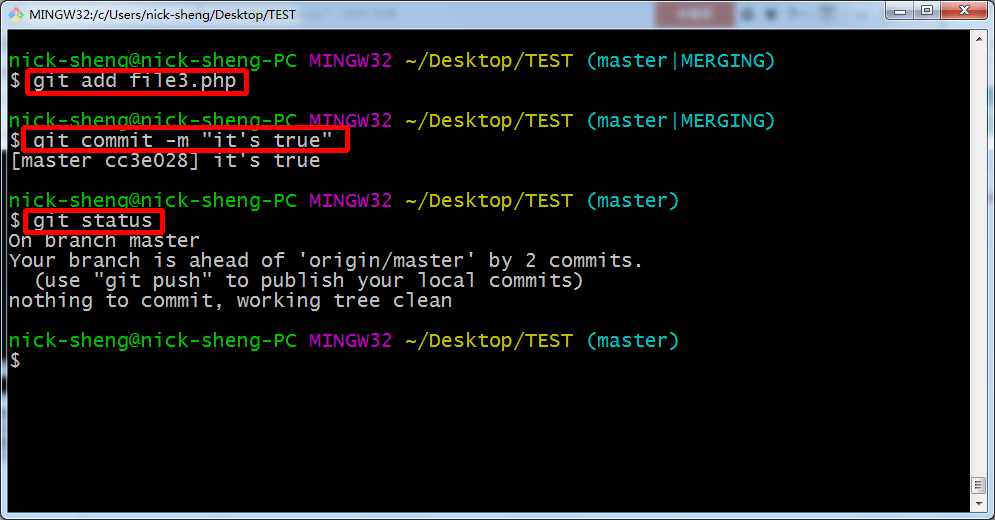


git diff查看冲突的文件)





解决冲突,打开file3.php,将不需要保留的代码删掉)



(15.保存好冲突文件,再一次add,添加到暂存区,再commit,提交到版本库,即可正常推送了)

**如果有冲突,解决完冲突,再commit提交到版本库,最后推送到远程仓库.**

**解决冲突务必要小心,切不可随意删除他人代码.**