

mkdir blog 创建文件

cd blog 进入blog

ll -A 查看隐藏的.git文件内容

git init 初始化一个 Git 仓库

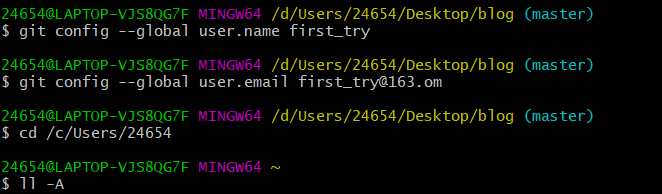
ll .git/ 表示查看.git 下的文件列表显示

cat .git/config 查看.git/config 内容

git config user.name zysheep

git config user.email zysheep@163.com

cat .git/config 查看设置结果



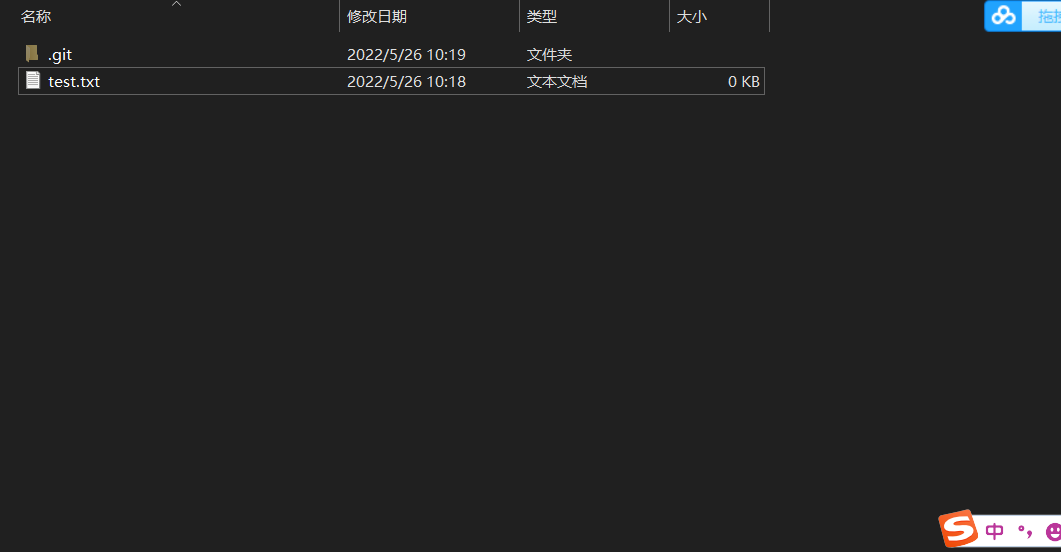
git config –global user.name zysheep

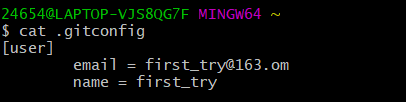
git config –global user.email zysheep@163.com

cd ~ 家目录（就是用户目录）

ll –A | grep –i git 查看叫git的文件名称

cat .gitconfig 查看文件内容

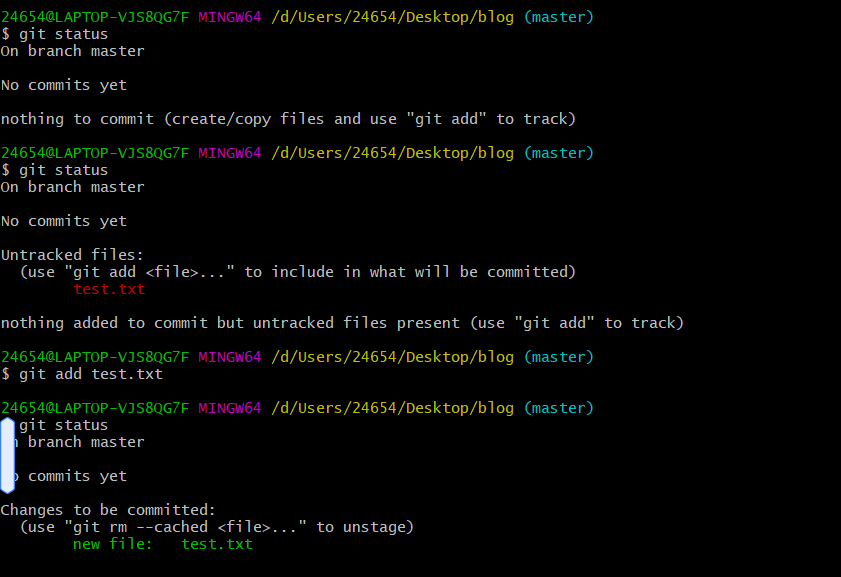




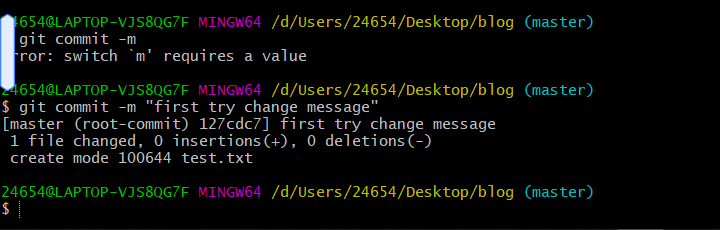
Git add +文件名之后才会变成绿码

Git add. 加入所有文件

Git ad



Git commit -m +描述

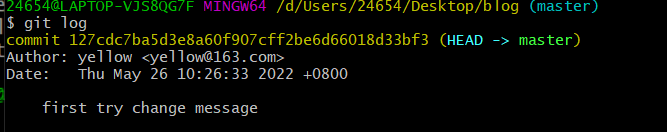


1 文件变更，0 insertions代码增加 0 deletions 代码减少

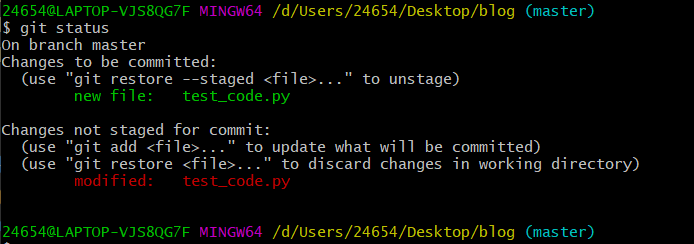
Git log 查看修改日志，看得到Author Date 和描述

Git lg

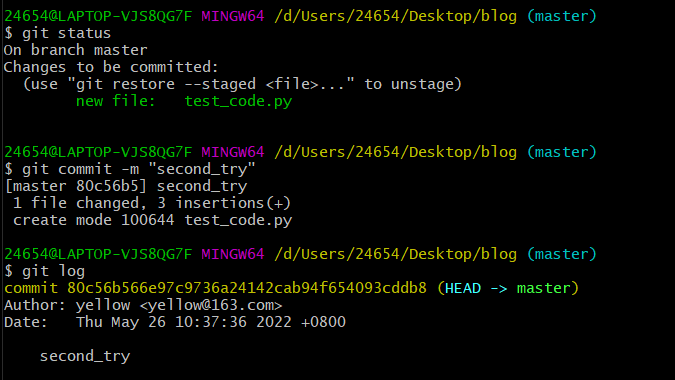
Commit 后面生成的是哈希值 唯一的，差不多用来标识的作用



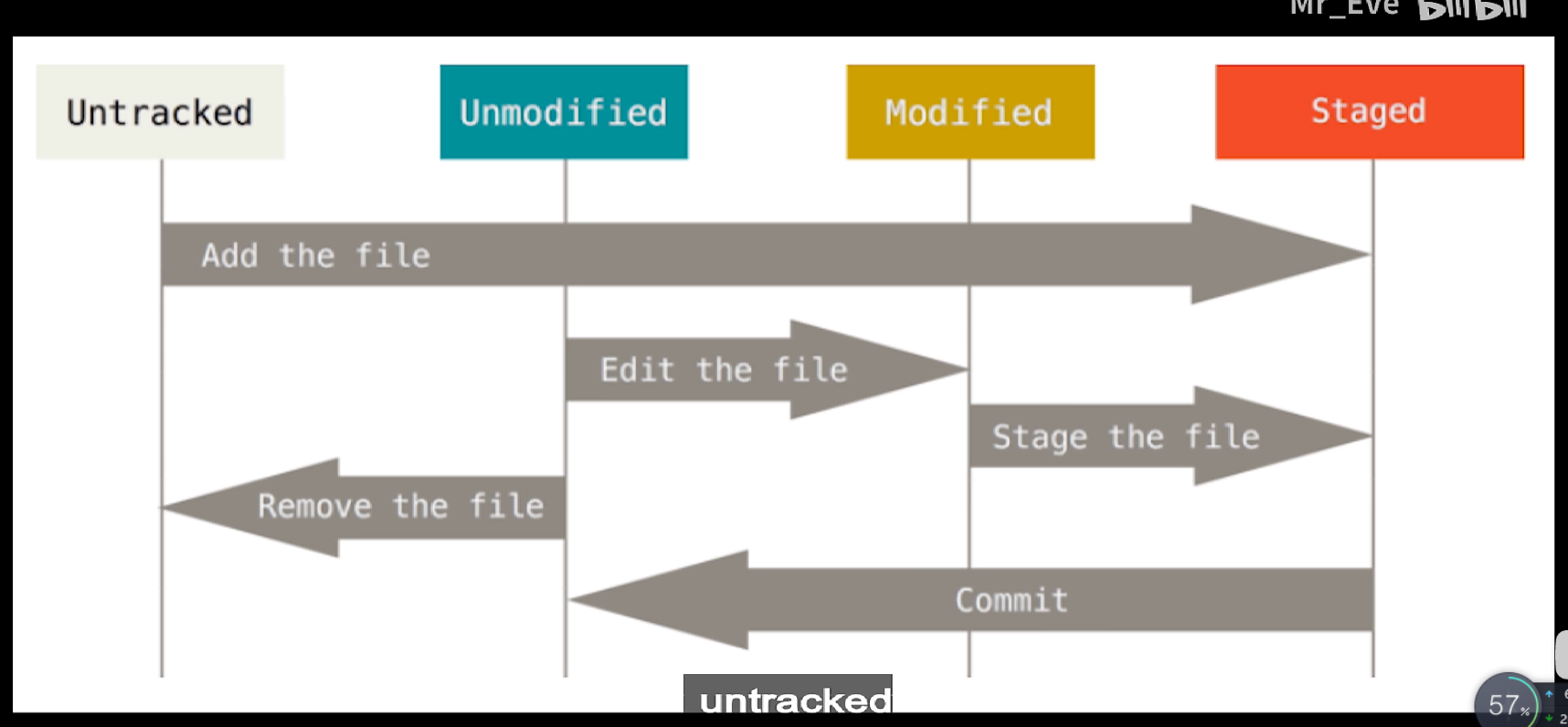
修改了一下之后再次查询又显示出现了修改



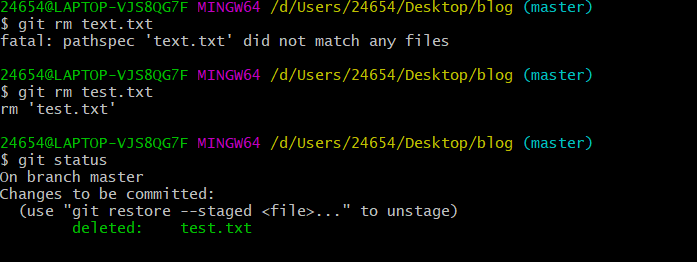
必须要再次提交才能实现更改



文件的四种状态

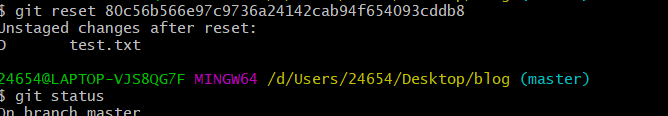


可以用rm命令移除

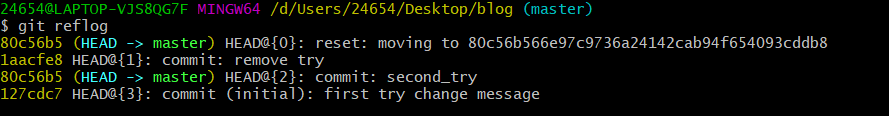


Git reset 回到某次修改的时候的形式（填入某次commit时产生的代码）输入前7个数字也可以，没啥问题

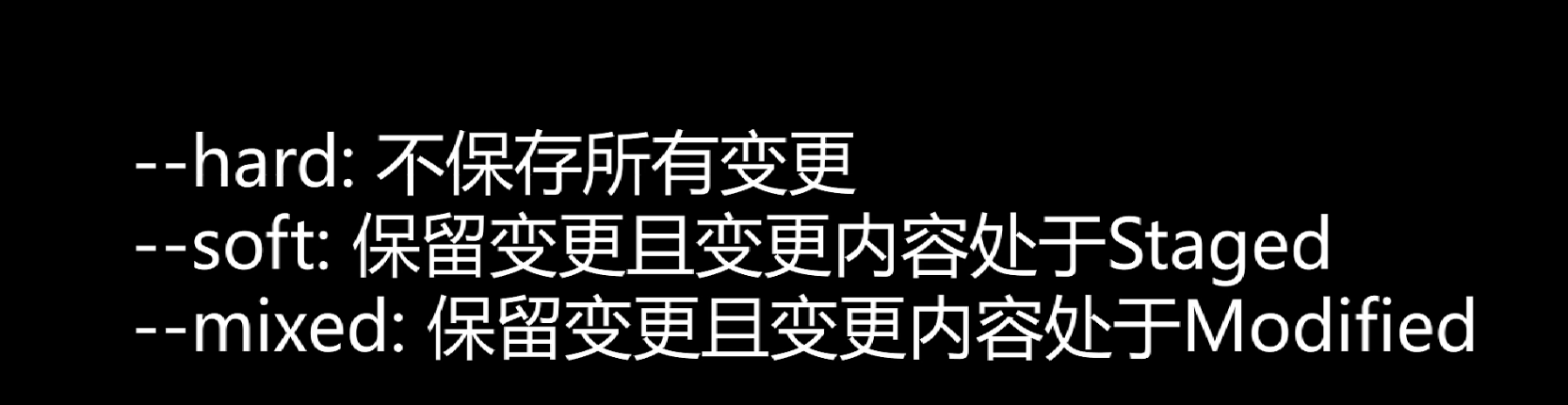
然后那一次的commit记录就会被消除

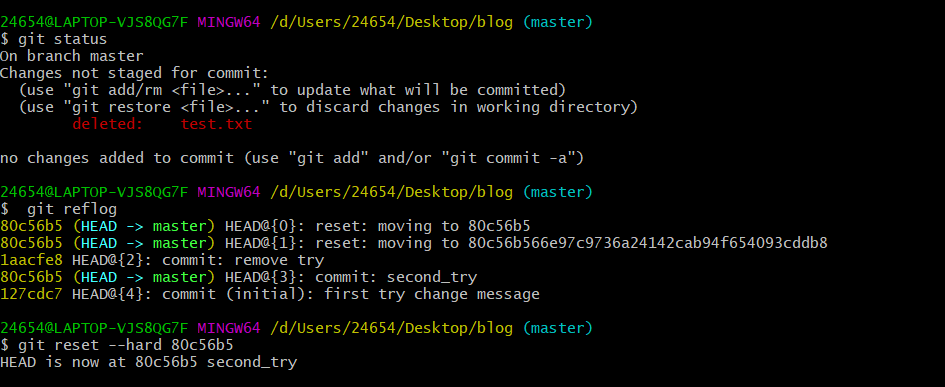


显示所有之前的命令，前面七位就是哈希码，也可以用于reset



Reset命令可以增加的后缀



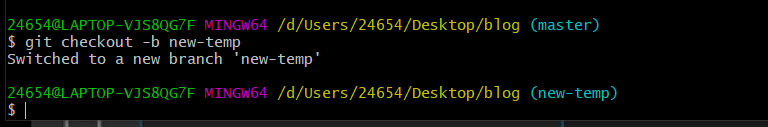
删除后找回的正确方式

分支

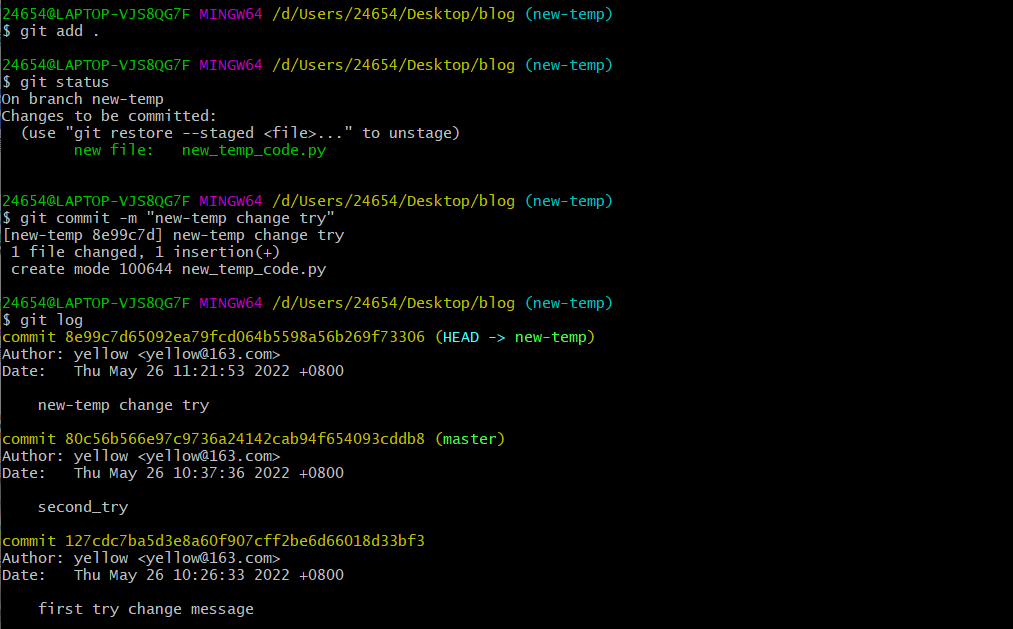
Git checkout -b+name+template（模板）

不填template就会以当前分支为模板

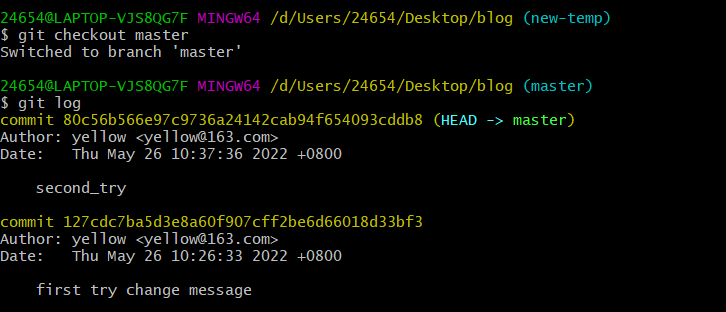
切入一个新的分支 分支相当于模板的影分身，一开始的时候和模板一模一样，但是修改分支不影响模板



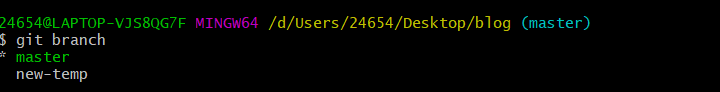
分支的变化不影响模板



切回master后，master（模板）是不受影响的

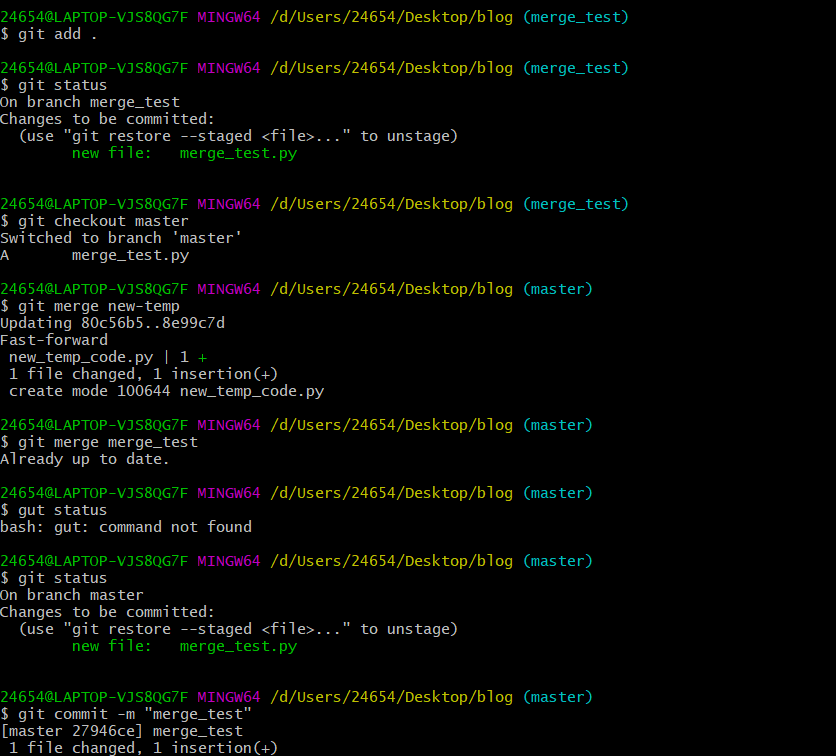


Git branch查看所有分支



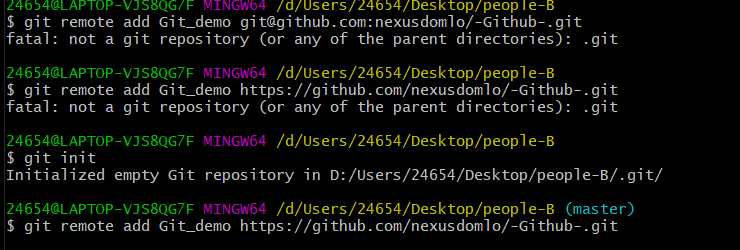
合并功能merge

合并操作



如果合并的两个版本中，有某些一样名字的文件但是内容不同，会发生冲突，可以自己手动更改以解决冲突

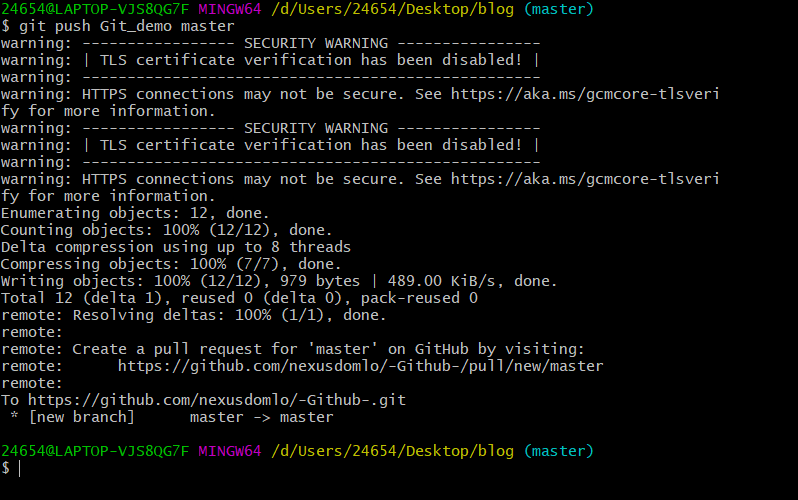
将某一个在github的库创建别名，方便之后push或者pull操作



远程仓库

方便多人在不同的电脑上操作

远程push



远程pull（拉）拉取Git\_demo中的master分支（这个分支必须是原来的已经被推送过去的分支，相当与就是改版本）

