# Git

Git四大对象的笼统理解：

在git的世界里， 有四种类型的object, 分别是tree、 blob 、commit、tag。

tree代表的是目录结构， 或者简单地理解为代表一个目录。

blob用来存储文件内容， 或者说表示一个文件。

commit 存储一次提交的信息，包括所在的tree，parent是谁，作者及message等信息。

tag 就是标签的意思， 实际就是commit的别名。

配置信息：

git config --global user.name “sean” 修改用户名为sean

git config --global user.email [“1319587163@qq.com”](mailto:\“1319587163@qq.com\”) 改为此邮箱

git config --global alias.olog “log --oneline” 配置log --oneline的别名为olog,则git olog = git log --oneline

/C:/Users/ff/.gitconfig 本用户全局配置文件

/C:/Users/ff/Desktop/GitTest/.git/config 本仓库配置文件

常用命令汇总：

git add ./filename 修改提交至缓存区

git commit -m “record of changes” 修改提交至仓库

git commit -a -m “record of changes” 一次性将修改提交至仓库

git status 查看改修状态

git log 查看提交记录（最近的）

git reflog 查看所有提交记录

git show commit哈希值 查看某commit情况

git rm filename 删除某文件（然后需要commit）

git rm --cached filename 从暂存区中删除，工作目录中保留

git reset --soft head^ commit回到前一个，这次commit修改内容退回到暂存区，后退了0.5步

git reset --mixed head^ commit回到前一个，这次commit修改内容退回到工作区，后退了0.75步

git reset --hard head^ commit回到前一个，这次commit修改内容被删除，后退了1步

git reset --hard 哈希值 回到特定哈希值的版本

git blame filename 查看该文件的修改记录（缉凶利器）

git branch new 创建一个新分支new

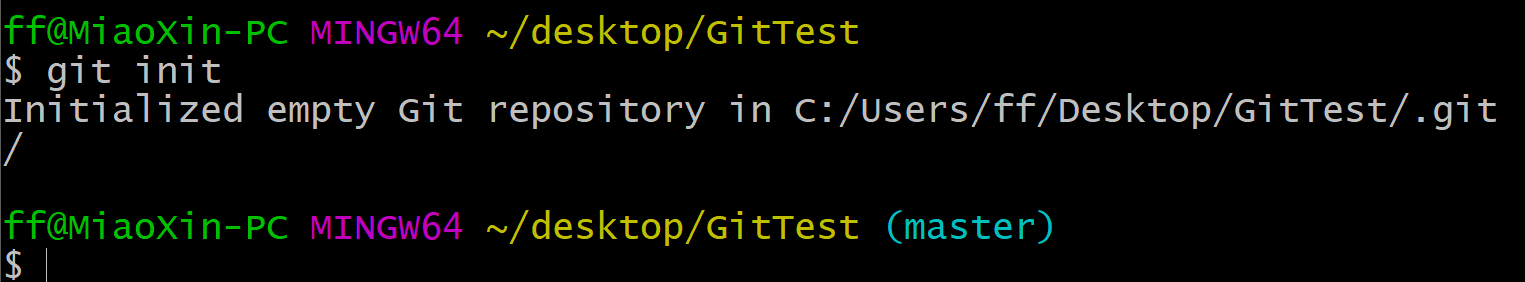
git checkout new 切换到新分支new

git checkout -b new 创建然后切换到新分支new

git merge new 将new分支合并到当前分支

操作过程(windows)：

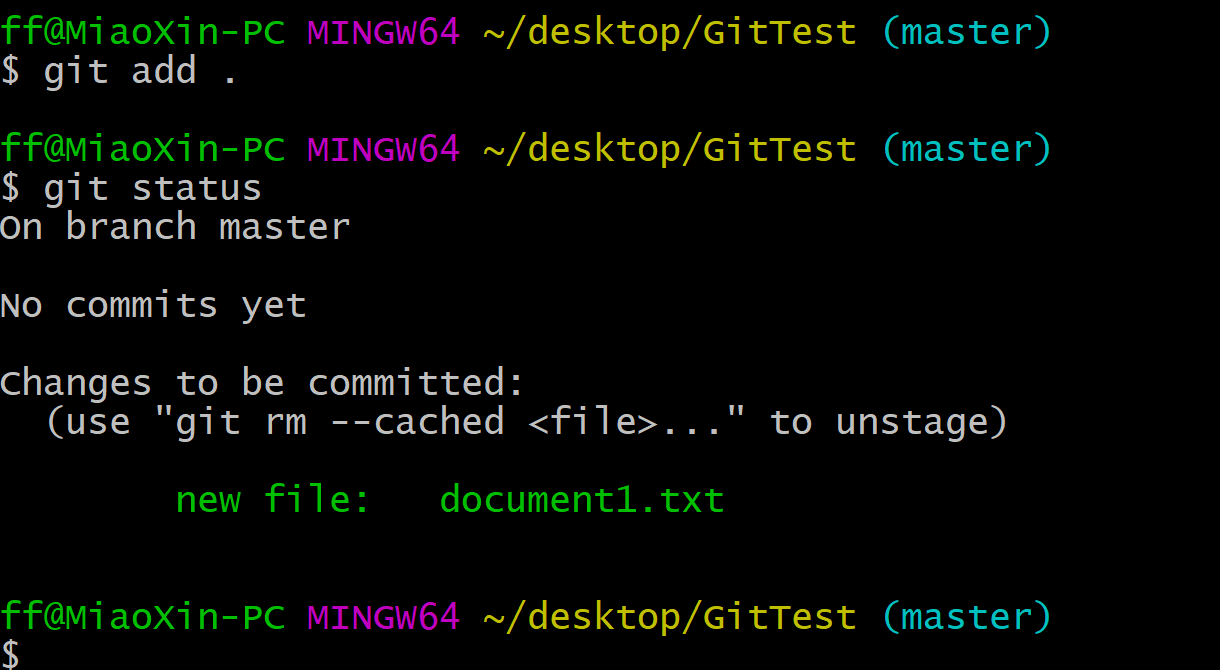
1.打开git-bash，cd命令进入要初始化成库的文件夹里，用git init命令进行初始化。



2.初始化后的文件夹下面多了一个.git文件夹



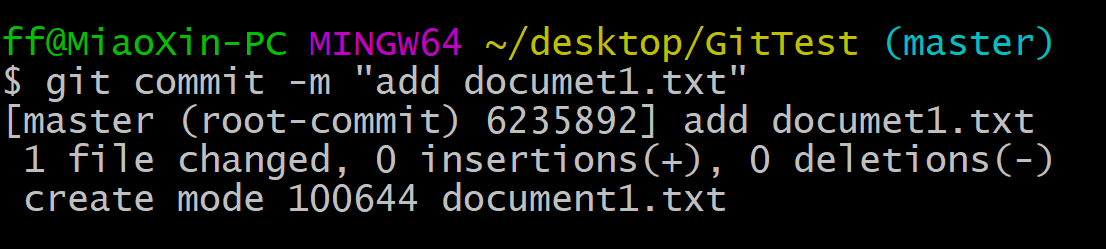
3.在GitTest文件夹中添加document1.txt文件，然后用git add .将所有文件修改提交到缓存区（也可以用git add document1.txt只提交该文件的修改到缓存区）。



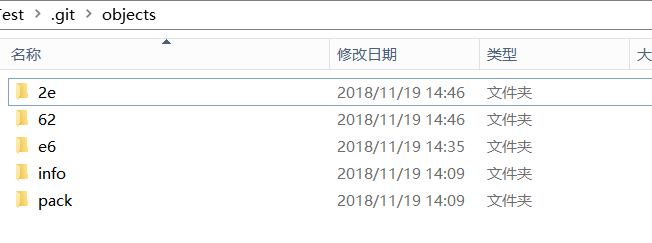
4.再查看.git/objects文件夹，发现已经对document1.txt文件生成了一个blob对象。



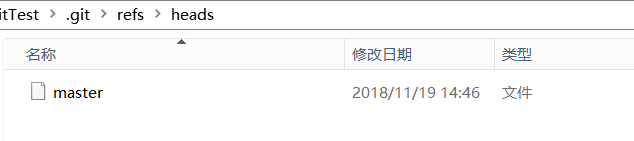
5.使用命令git commit -m “add document1.txt”将缓存区中的内容提交。



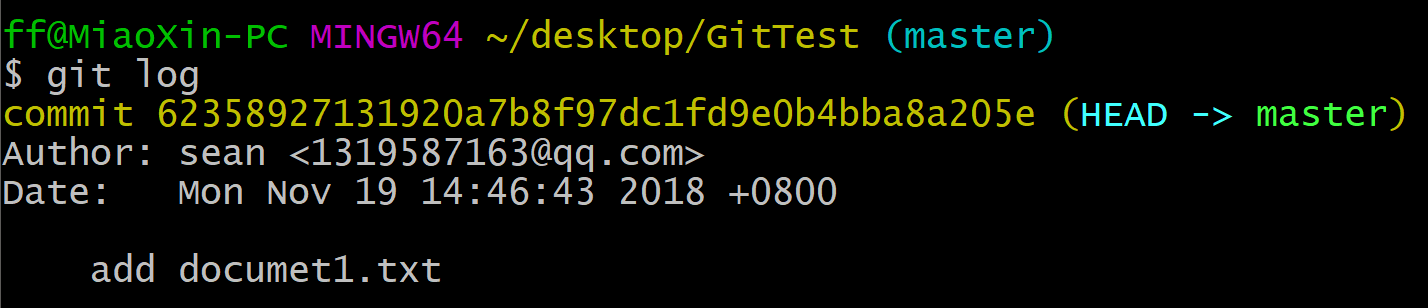
6.再次查看.git/objects文件夹，发现又多生成了两个新的对象，即分别为改修的文件夹对应的tree和整个仓库对应的新状态的commit。



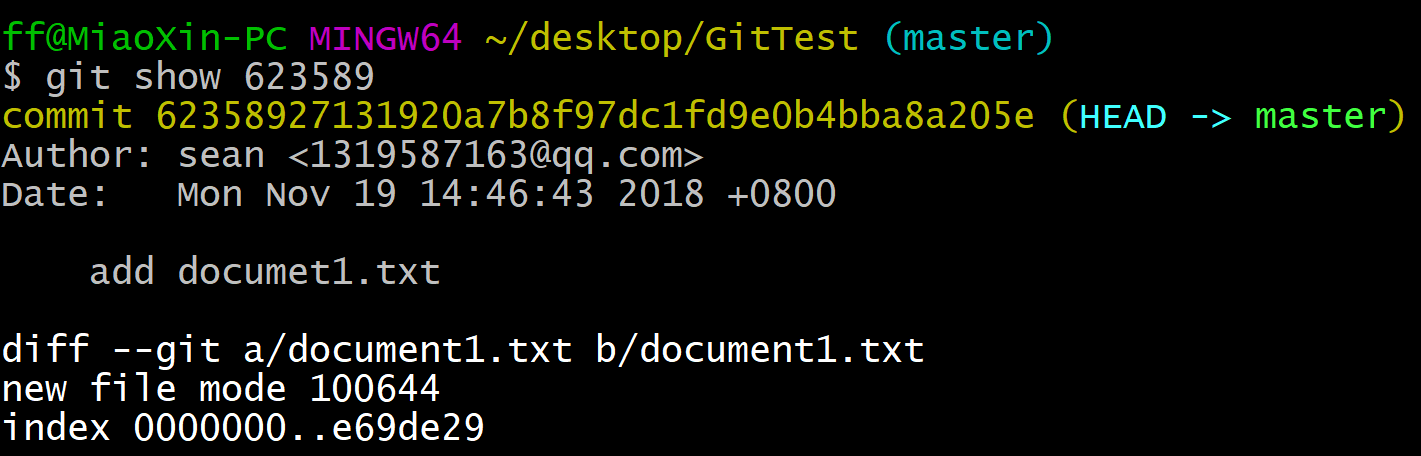
7.同时查看.git/refs/heads文件夹，可以看到由于产生了commit则head有东西可指向，里面就生成了一个master文件，它里面存储的信息就是commit对应的hash值。



8.可以通过git log命令查看提交历史记录



9.使用命令git show commit哈希值 可以查询某个特定commit的情况。



远程仓库：

基本命令：

Git remote add origin [git@github.com:SeanMiao95/GitTest](mailto:git@github.com:SeanMiao95/GitTest)

建立当前仓库的远程仓库（连接），origin为远程仓库名，可以改为其它名字

git push origin master（本地）

将本地的master分支推送到远程仓库，可其它分支完成该操作

git pull origin master（远程）

将远程仓库master分支合并到当前分支

git clone [git@github.com:SeanMiao95/GitTest](mailto:git@github.com:SeanMiao95/GitTest)

把远程仓库GitTest克隆到本地的当前目录下

第一次使用SSH通信需要创建SSH Key:

打开gitbash输入：

$ ssh-keygen -t rsa -C "1319587163@qq.com"

一路回车顺利的话可以在用户目录下找到.ssh目录，里面有id\_rsa和id\_rsa.pub两个文件，这两个就是SSH Key的秘钥对，id\_rsa是私钥，不能泄露出去，id\_rsa.pub是公钥，可以放心地告诉任何人。

在github中添加上id\_rsa.pub中的内容就可以在本地推送修改到github远程仓库了。