一：Git是什么？  
 Git是目前世界上最先进的分布式版本控制系统。  
工作原理 / 流程：



Workspace：工作区  
Index / Stage：暂存区  
Repository：仓库区（本地仓库，存放在本机）  
Remote：远程仓库（外网服务器上）

#### 二：SVN与Git的最主要的区别

SVN是集中式版本控制系统，版本库是集中放在中央服务器的，而干活的时候，用的都是自己的电脑，所以首先要从中央服务器哪里得到最新的版本，然后干活，干完后，需要把自己做完的活推送到中央服务器。

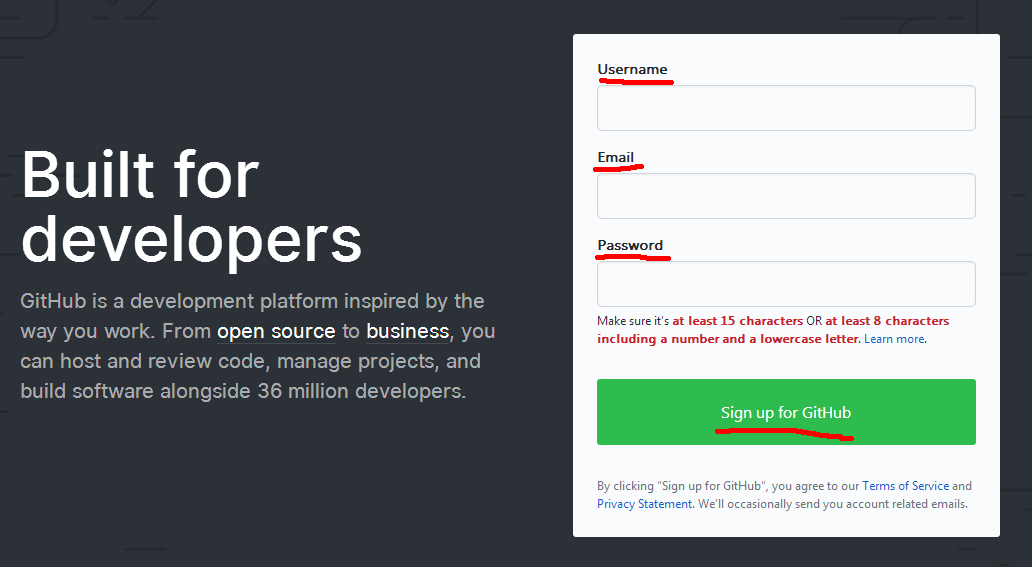
Git是分布式版本控制系统，那么它就没有中央服务器的，每个人的电脑就是一个完整的版本库，这样，工作的时候就不需要联网了，因为版本都是在自己的电脑上。既然每个人的电脑都有一个完整的版本库，那多个人如何协作呢？比如说自己在电脑上改了文件A，其他人也在电脑上改了文件A，这时，你们两之间只需把各自的修改推送给对方，就可以互相看到对方的修改了。

#### GitHub

最常用的Git代码托管平台，是一个面向开源及私有软件项目的托管平台，GitHub地址：<https://github.com/> 好多好的开源项目都来自GitHub，但是GitHub只能新建公开的Git仓库，私有仓库要收费！！！如果你做的是一个开源项目，可以首选GitHub。其它Git平台还有Gitlab 、Git @ OSC等

#### 注册GitHub

在浏览器上打开GitHub网址：<https://github.com/>



比如我的注册信息:

账号1：

Username：用户名：qin10086

Email：邮箱：[1017416274@qq.com](mailto:1017416274@qq.com)

Password：密码：qinjianji10086

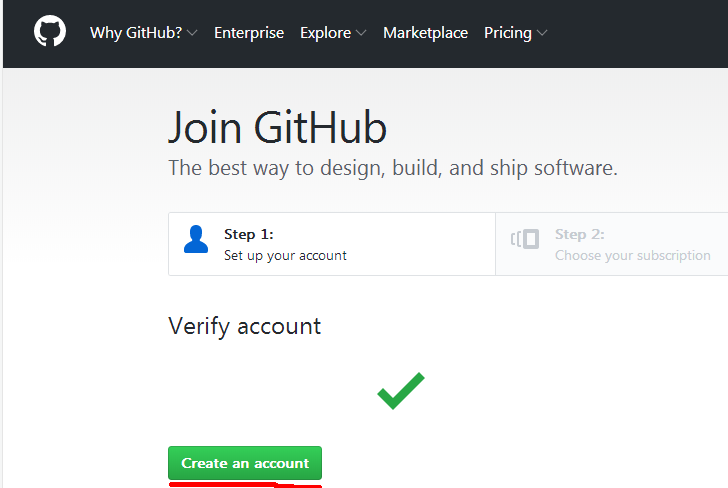
账号2：

Username：用户名：qin10010

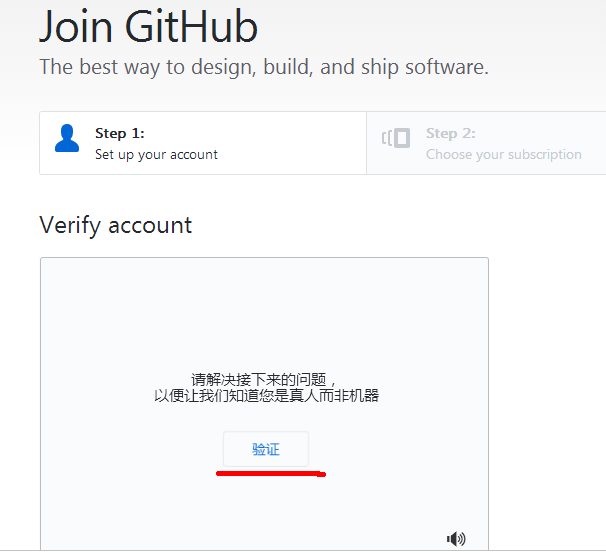
Email：邮箱：[506665186@qq.com](mailto:1017416274@qq.com)

Password：密码：qinjianji123456

Sign up for GitHub ：点击注册



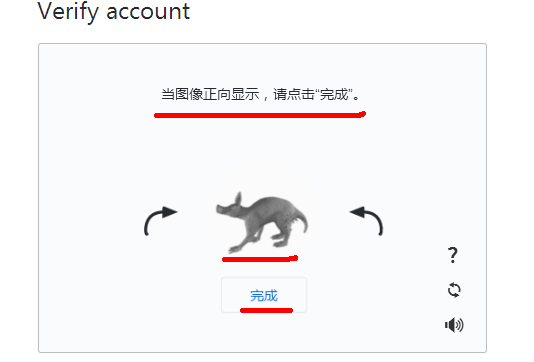
点击create an account ：建立一个账号，可能需要多点几次出现下面界面进行验证，通过验证才能注册



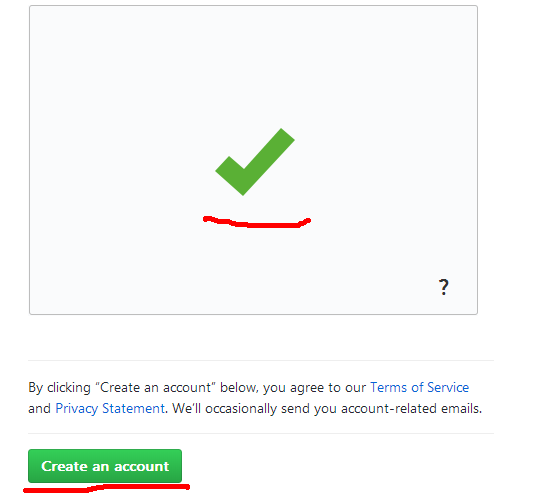
验证界面如下



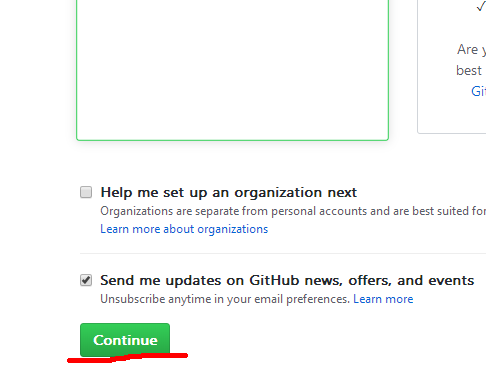
点击左右的箭头，移动动物的位置，移动为如下图正立状态点击完成，一共有3个动物，都要移动为正立状态点击完成才能通过。



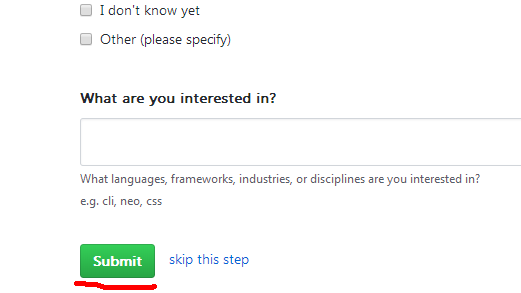
可能需要再输入一次用户名、密码、邮箱，输入成功后点击create an account



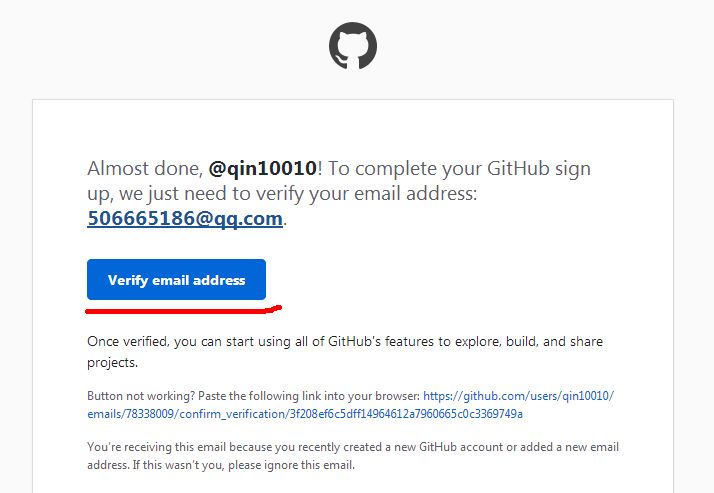
点击create an account后，会往注册邮箱发送一封邮件，往下移动点击continue



进入步骤三，无需设置，直接点击submit提交即可

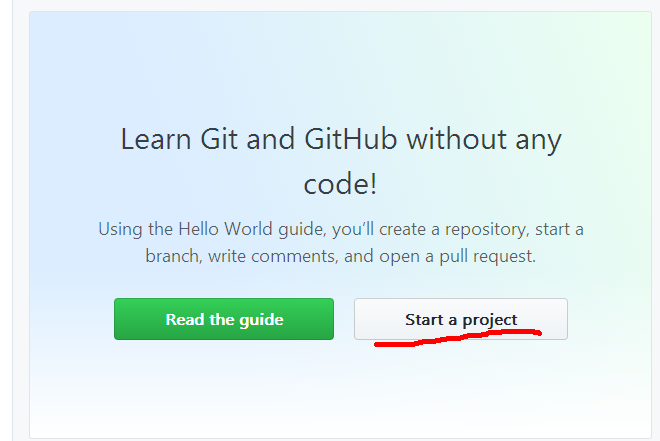


进入邮箱，点击激活账号即可使用

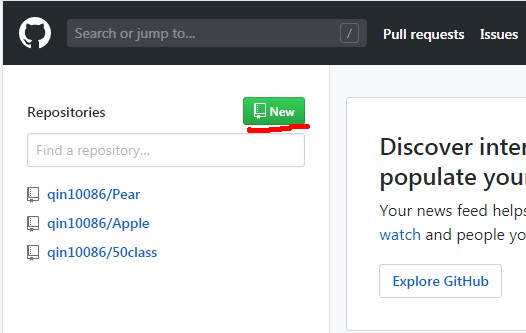


激活账号后，会直接进入git网页，

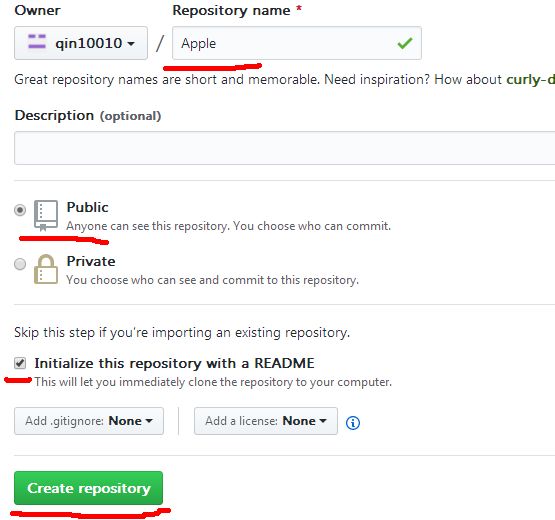
#### 新建仓库



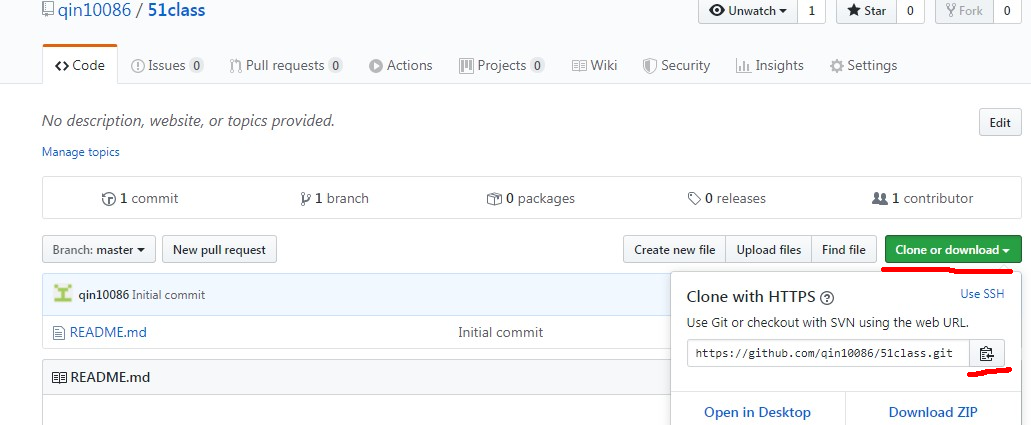
**Start a project 建立一个仓库,如果之前建过仓库,登陆git后,也可以通过new新建仓库**



**Repository name：输入仓库名，选择public（公共），注意勾上初始化创建readme文件，然后创建仓库**

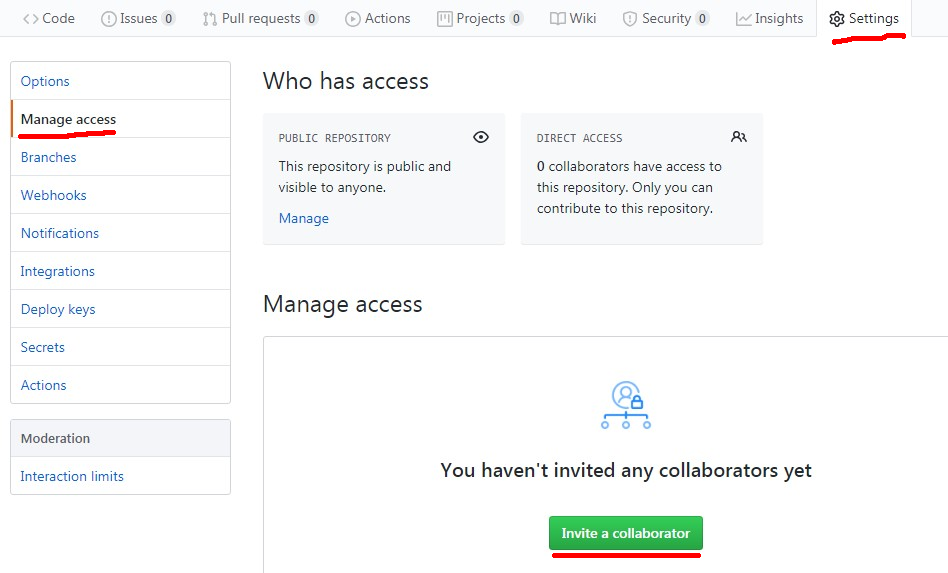


建库成功后可以看到远程库连接地址,注意每个人的地址不一样，比如我的连接地址：[https://github.com/qin10086/51class.git](https://github.com/qin10086/Apple.git) 查看后可以复制

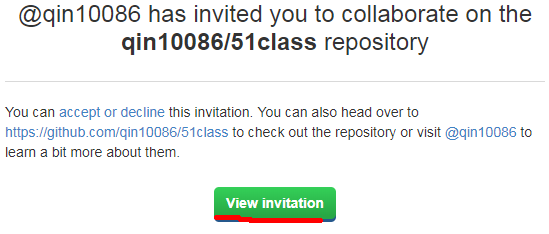


建库时没有创建redme文件，需要creating a new file，远程库必须有文件才能使用。

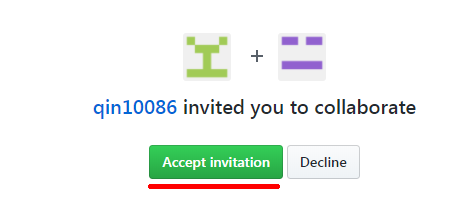
**邀请别人**参加你的项目（使用你的git），点击setting --->manage acess---> invite a collaborator.输入git用户名（pkzhbin1） ---->add collaborator



1. 如果邀请别人使用仓库，需要添加注册激活成功的账号，添加后会给账号的邮箱发送邀请邮件
2. 邀请人去注册绑定的邮箱查看邀请邮件， 点击绿色的 invitation，跳转到github网站，



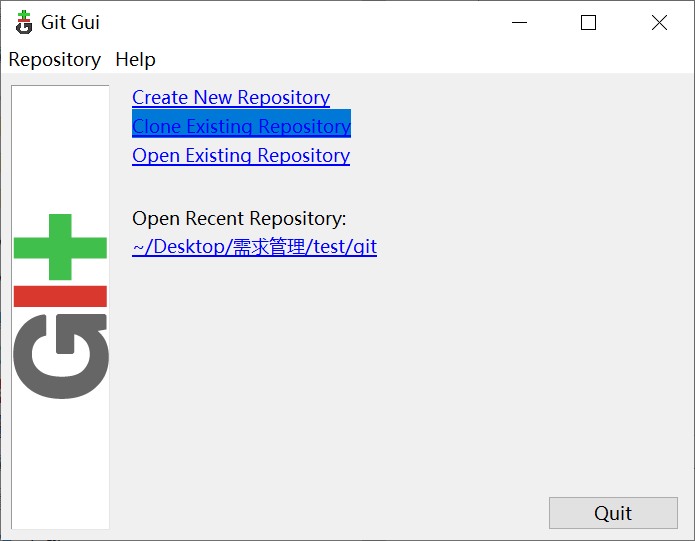
1. 没有登录可以登录一下， 点击绿色的 accept invitation 参加项目,这样就可以远程使用别人的git仓库



#### Git操作

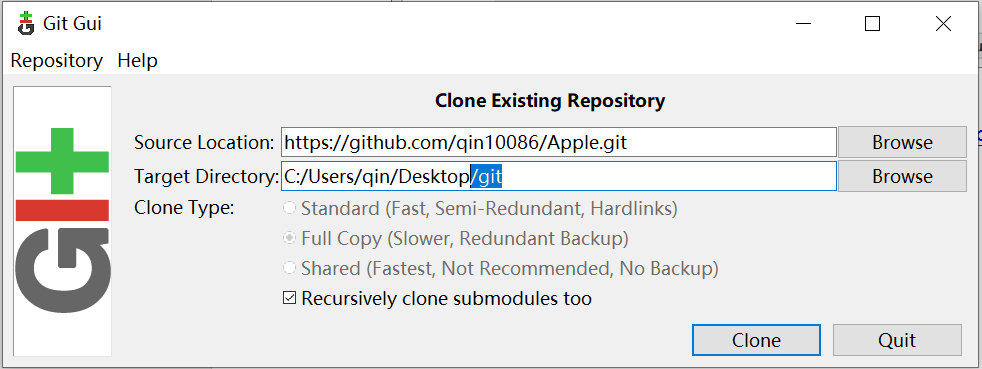
#### **Git操作：https://github.com/qin10086/Pear.git**

1. clone一个你需要参与的项目的仓库到你本地，进入目标目录，鼠标右键后选择Git Gui hear，打开git界面后选择第二个clone



1. 输入git的URL地址 source location：<https://github.com/qin10086/Apple.git>

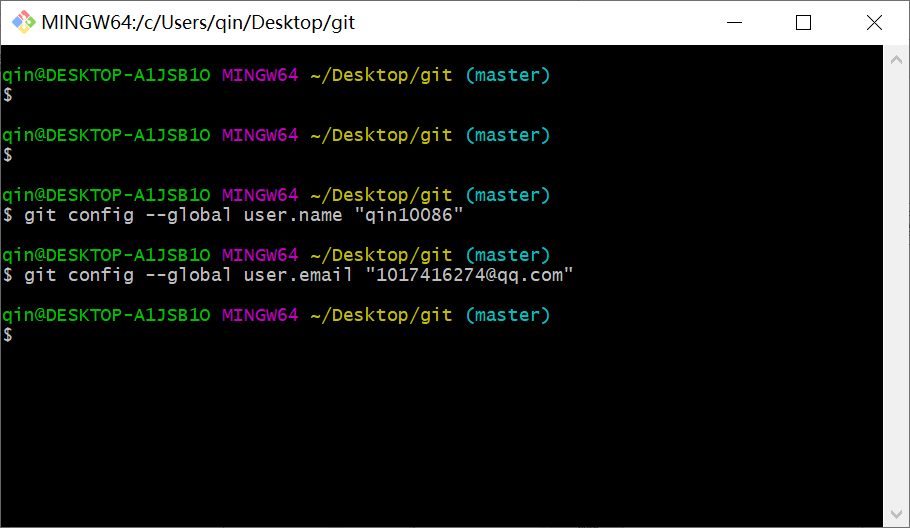
克隆到本地的目录 C:/Users/qin/Desktop/git 通过browse选中目录，需要在后面再加上clone的目录，目录明必须是不存在的， 比如我选中的目录是：C:/Users/qin/Desktop，手动在后面加上 /git 。 点击clone



1. Clone好后会打开 git 界面，这时需要输入本地的git账号，去clone的目录右键选择git bash here ，进入shell界面，可以输入git控制命令
2. 输入使用的账号和邮箱，进入命令行输入界面，分别在命令行执行下面2条语句：

git config --global user.name "qin10086" user.name处输入你的git账号，输入后回车执行

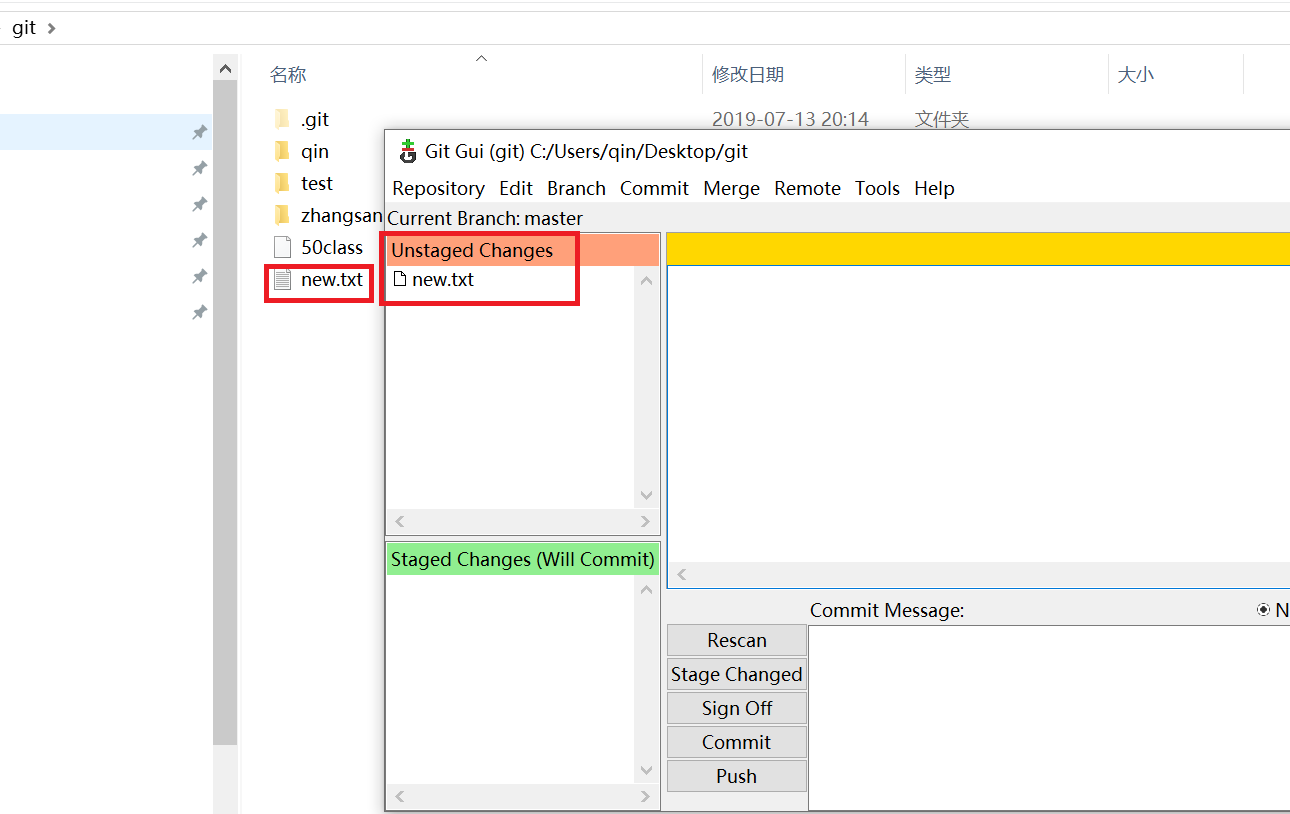
git config --global user.email "1017416274@qq.com" user.email处输入你的邮箱，输入后回车执行



**git使用**

1. 本地文件(增/删/改)到远程服务器：在本地仓库新建一个文件 new.txt，上传到远程服务器，按照rescan--stage –- sign off –- commit – - push 顺序执行即可

ReScan ：扫描，扫描文件更新、变化，会把有变化的文件扫描出来

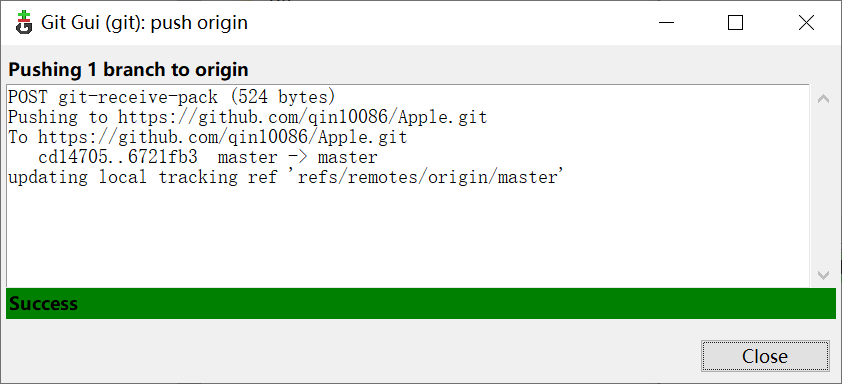


Stage：缓存：把修改文件缓存到本地的服务器（GIT），

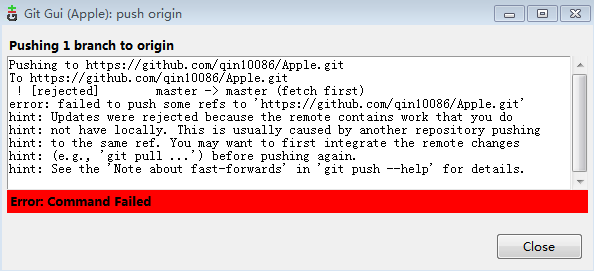
Sign off：签名，相当于svn的添加说明（git必须要写，svn可以不写）。

Commit：提交到本地服务器。

Push：推送到远程服务器，有限制，同一时间只有1个人能push成功（服务器版本更新呢！你必须先update：git pull origin master ，然后再 push）



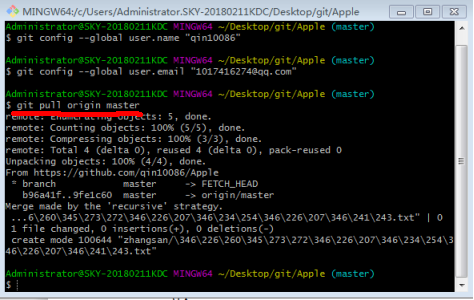
注意：如果本地git版本同git远程仓库版本比不是最新的版本，push会失败，提示如下信息，关掉即可，需要先更新本地的版本。



问题:部分人会出现每次push都需要输入账号密码，需要进行设置，在文件夹（包含隐藏文件夹.git文件夹的）内右键 git bash here ，在git bash交互环境输入命令 git config  credential.helper store   。下次push输入正确账号密码即可

1. 更新本地git版本为远程服务器最新版本，在shell里输入：

git pull origin master 相当于svn下 update 更新为最新版本， 会有如下提示更新的信息：不需要关注，只是提示。相当于进入了Linux系统里的vi编辑模式



退出方法： ESC （退出编辑模式） -- > shift+： --> q！ 回车退出

Git使用特点：开发比较分散，不需要集中起来干活，国外推崇的自由工作时间。 (push 一次只能一个人来)

1. 删除文件找回

git reset HEAD ceshi/new.txt　 标记文件 ceshi/new.txt

git checkout ceshi/new.txt 找回文件ceshi/new.txt

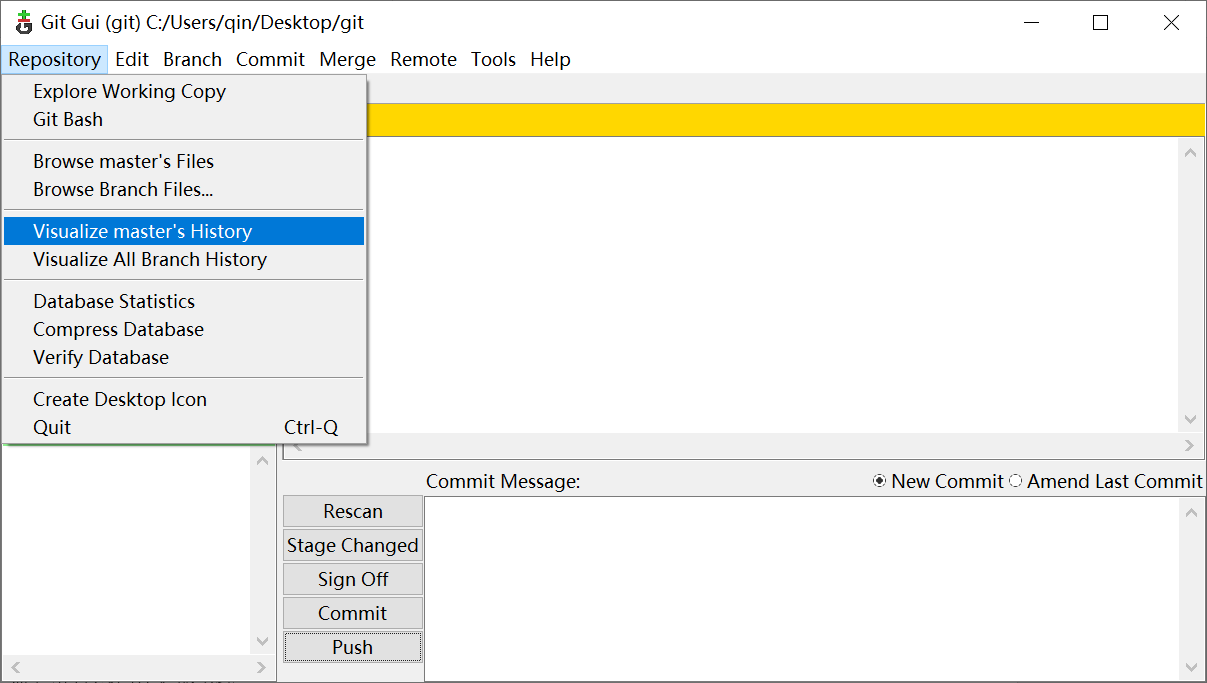
如果我们不知道之前删除了哪些文件或则文件夹，可以用如下命令查看到：

git status 查看历史信息 如果是中文不能直接用，需要用转义符

4. 查询版本历史信息：（show log）

1.我们想了解本地目录的一个历史信息。

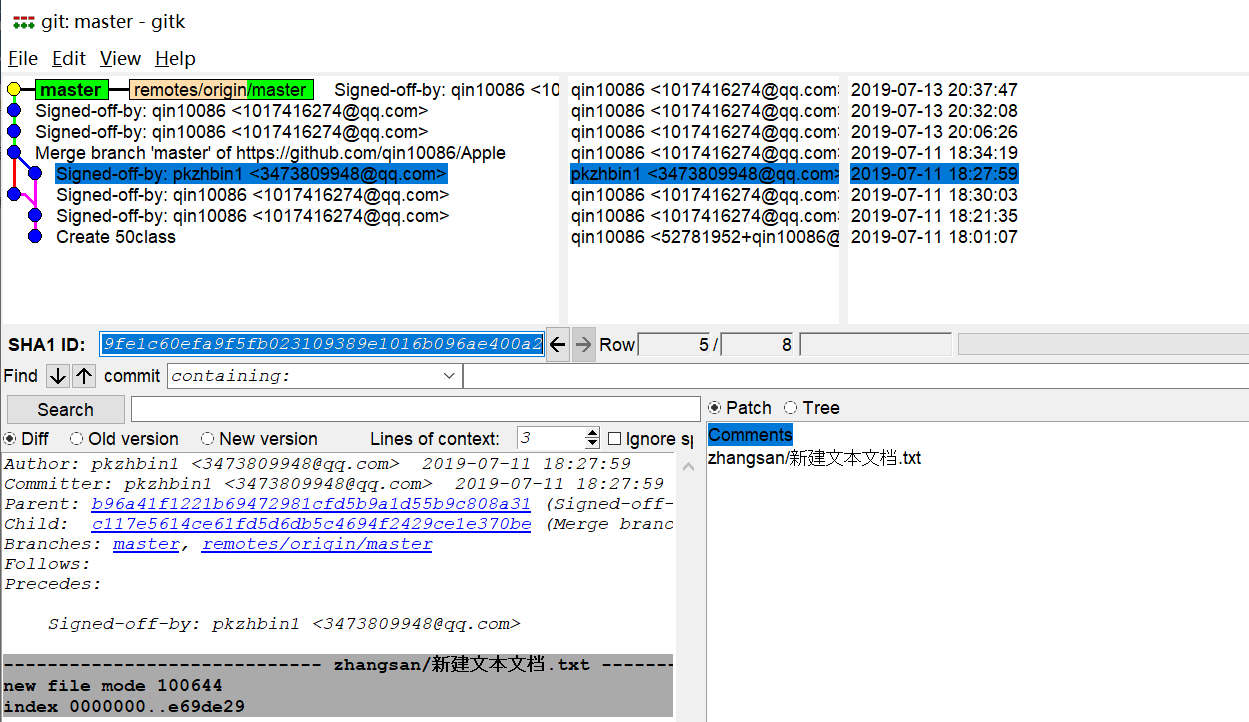
2.可以在git gui->repository->visualize master's history



黄色的点代表的是本机服务器的当前版本，相当于svn里show log的加黑加粗那个版本

蓝色的代表历史版本，点击黄点或蓝色的点可以查看版本号

Sha1 id == 加密的版本号



选中版本的版本号Sha1 id：9fe1c60efa9f5fb023109389e1016b096ae400a2

5. 回到之前的版本：git reset --hard sha1 id

1.比如我们想回到之前刚提交new.txt文件的版本（文件内容是空的时候）。

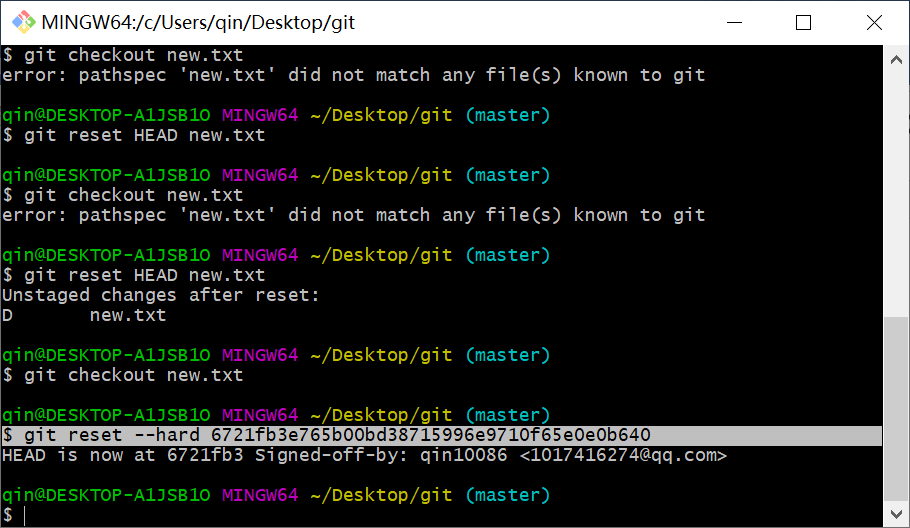
2.首先查找那个时候的版本SHA值,查看历史记录：

版本号Sha1 id：6721fb3e765b00bd38715996e9710f65e0e0b640

拷贝这个SHA值，然后再git base里输入

git reset --hard 6721fb3e765b00bd38715996e9710f65e0e0b640

回到git目录，查看new.txt文件，变成空文件



1. 比较二个版本之间的变化：git diff sha1 id1 sha1 id2

比较sha1 id1 版本同 sha1 id2 之间的变化

git diff 6721fb3e765b00bd38715996e9710f65e0e0b640 55475a929668699ee322cf03c063d7d6743043ae

帮助手册：

https://git-scm.com/book/zh/v2/%E8%B5%B7%E6%AD%A5-%E5%85%B3%E4%BA%8E%E7%89%88%E6%9C%AC%E6%8E%A7%E5%88%B6