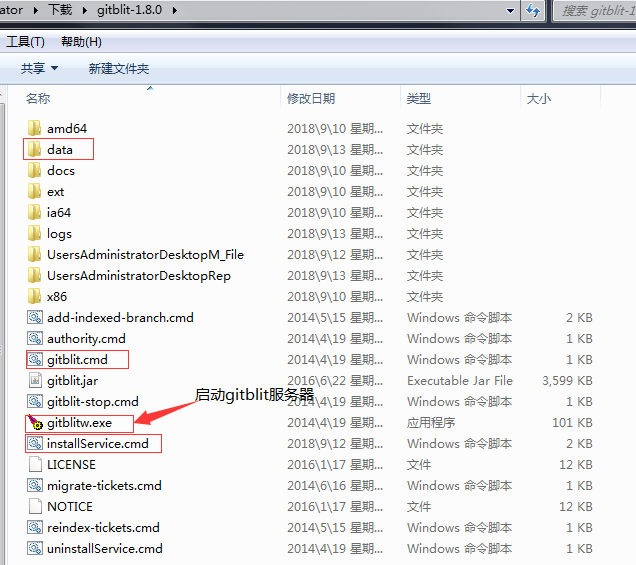
# Git

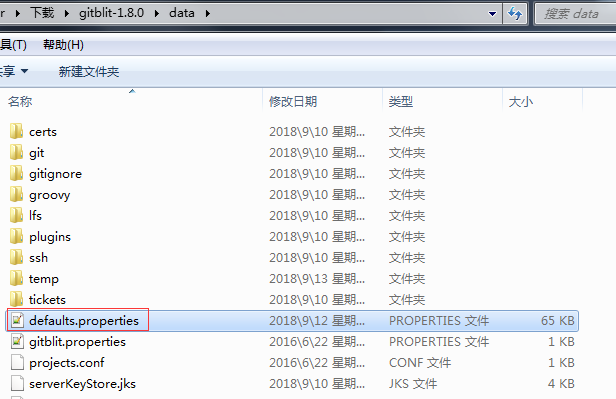
1. 项目和Gitblit服务器搭建
2. Gitblit需要安装Java JDK环境



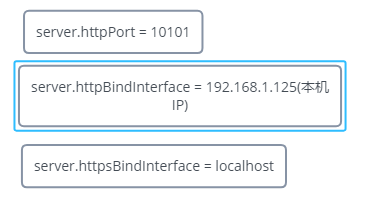
1. 解压gitblit-1.8.0.zip,不需要安装



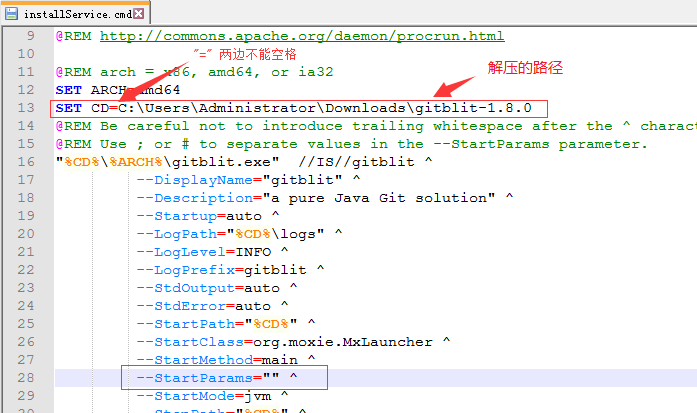
1. 对解压后的gitblit服务器文件进行配置



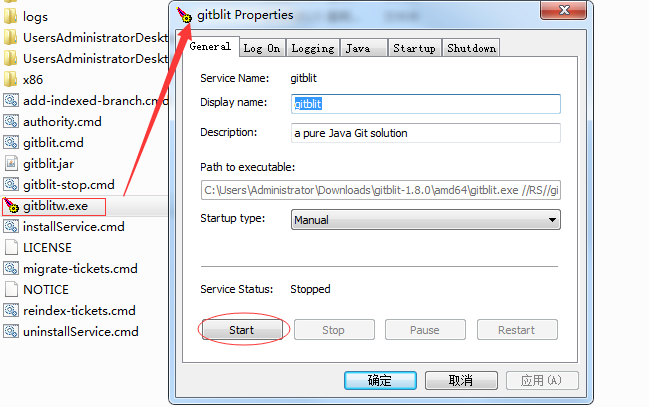
1. 找到Date中的defaults.properties文件,使用记事本或者Notepad++打开,Ctrl+F查找到并修改以下内容(10101为端口号)



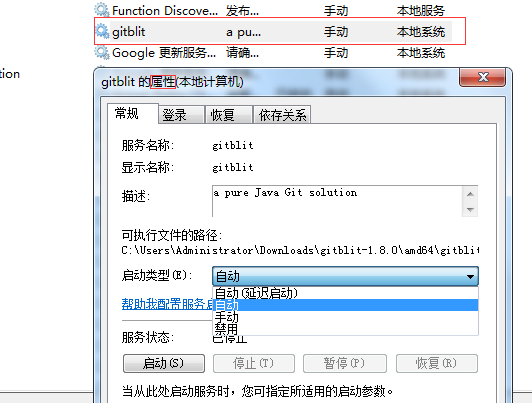
1. 找到gitblit-1.8.0下的installService.cmd,使用记事本打开修改以下内容



1. 运行gitblit-1.8.0下的gitblit.cmd进行配置
2. 运行gitblitw.cmd,弹出窗口点击start,开启服务器



1. 若要服务器开机自启,需要windows下搜索服务,进入设置gitblit属性中的启动类型为自动



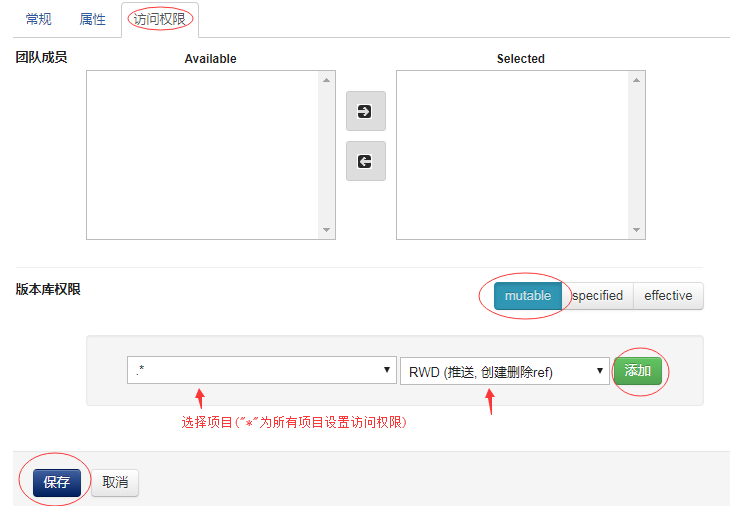
1. 开启gitblit服务器后,使用192.168.1.125:10101(设置好的本地Ip和端口)从浏览器登录gitblit,默认有一个用户名:admin,密码:admin的管理员用户.



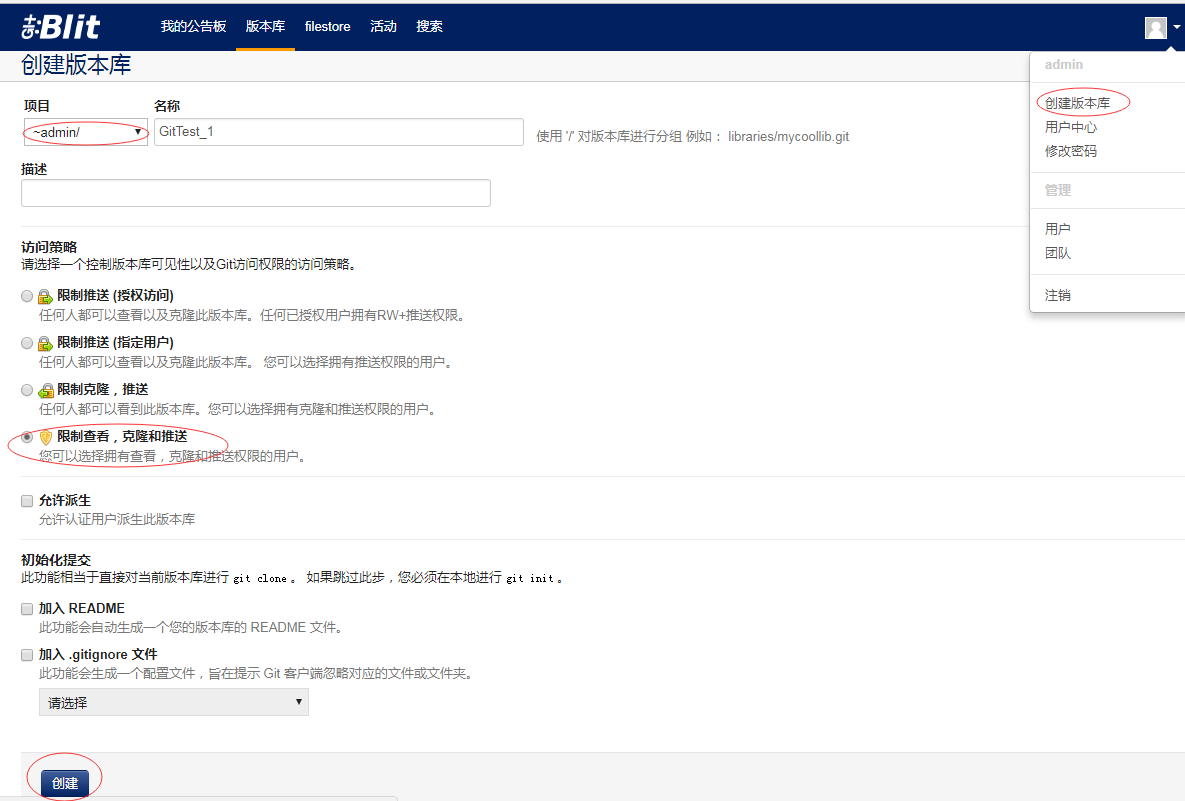
1. 点击用户创建团队成员账号







1. 使用管理员账户在Gitblit创建空项目



1. 创建成功后,登录普通用户(pt1),在活动中可以查看到刚刚创建的GitTest\_1项目



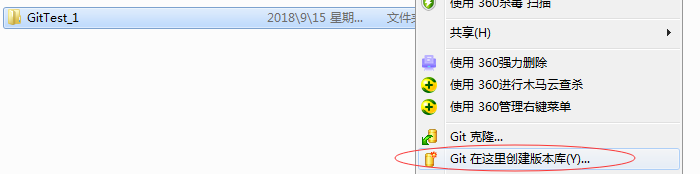
点击GitTest\_1可查看GitTest\_1项目状态



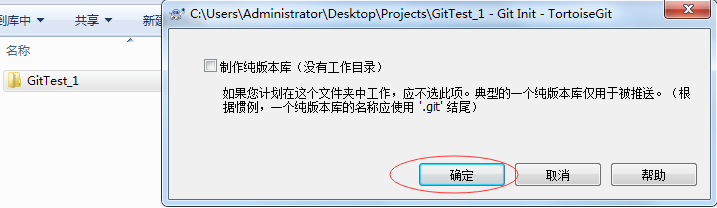
1. 将本地的Unity项目关联到Gitblit



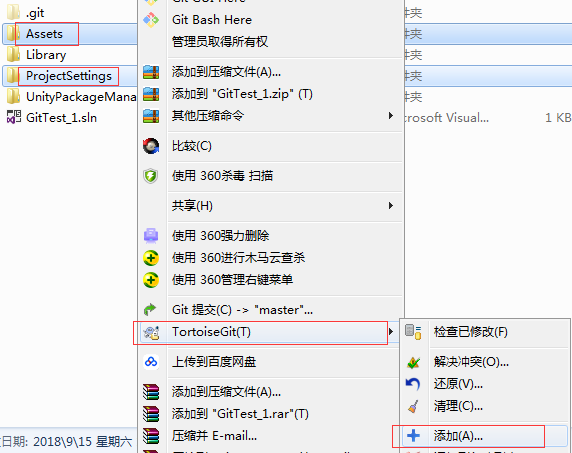
依次安装以上三个文件.

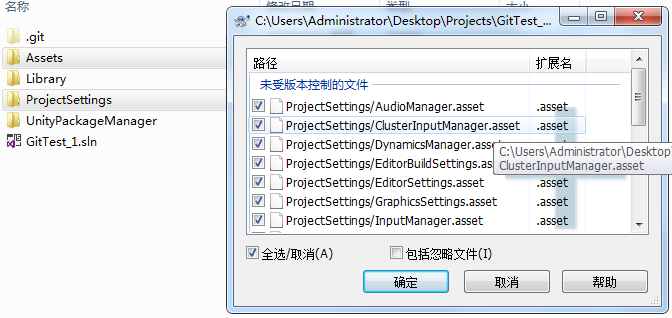


选中项目(GitTest\_1)右键创建版本库,项目中生成.git文件夹

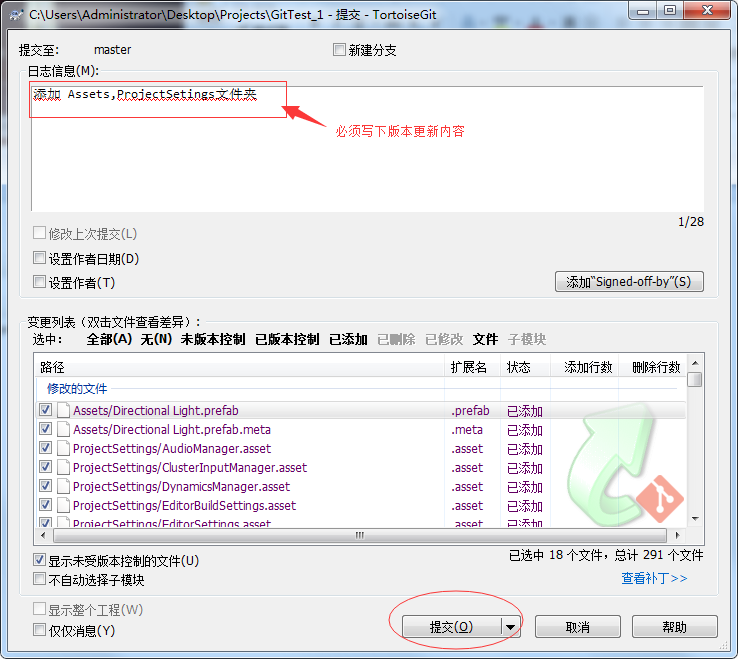


一般只要Git管理Unity项目中的Assets文件和ProjectSettings文件,选中两个文件夹使用添加(此步骤为Git追踪两个文件,详细内容第二章说明)

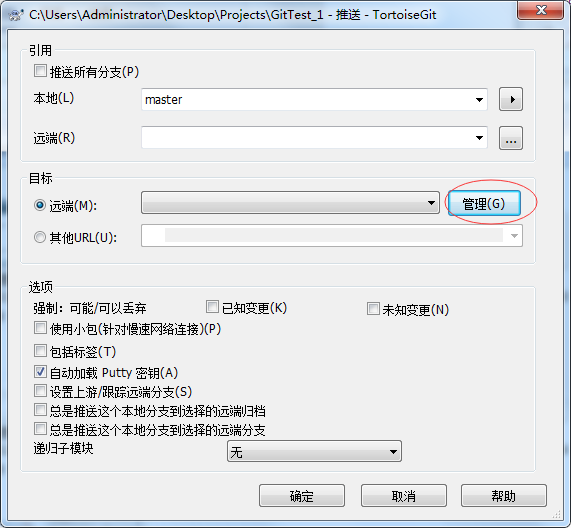




写下版本更新的有意义的内容并提交(在本地的版本库进行一次更新)



点击推送(将本地版本同步到服务器版本库)



点击管理设置远端服务器路径,由于是管理员将项目托管到Gitblit,所以先使用管理员登录Gitblit,查看到GitTest\_1的路径,填入URL中,点击”添加/保存” 🡪确认,推送到Gitblit服务器中





在Gitblit中即可刷新出刚刚推送的Assets和ProjectSettings



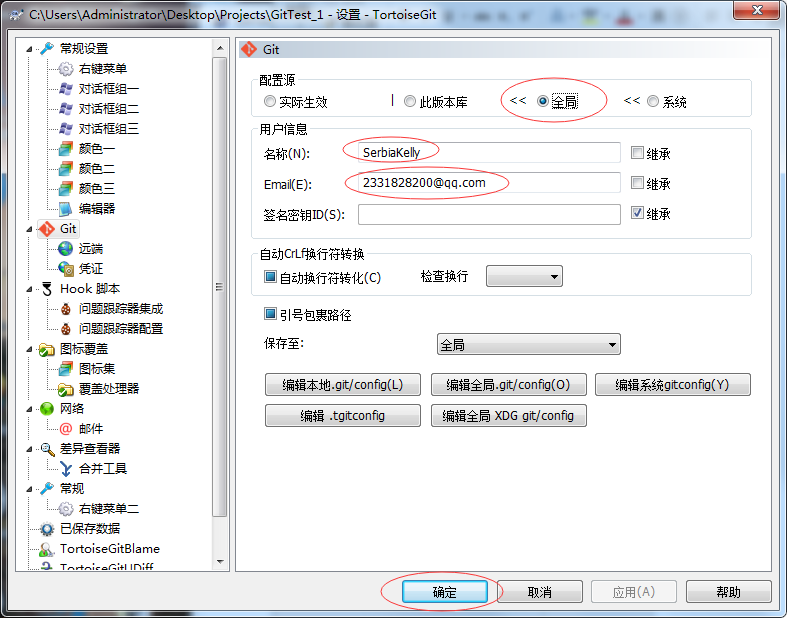
1. 团队成员使用Git
2. 依次安装以下三个文件.

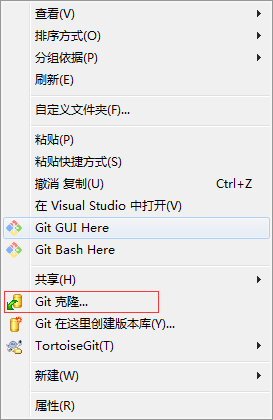


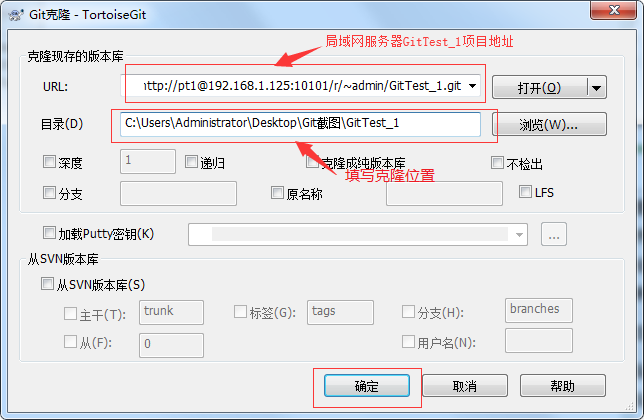
1. 团队成员可以通过登录管理员注册好的普通账号在各自客户端登录Gitblit查看到项目在本地服务器地址,通过地址克隆项目到本地(要保证服务器开启)



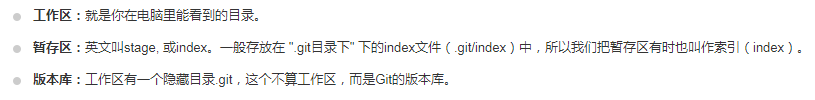
设置用户信息







1. Git基本概念



1. 项目克隆下来后设置忽略文件(使Git不管理忽略的文件)

使用Unity/VS打开克隆下来的项目,会生成其他不需要使用Git管理的文件(如Library和Temp等),选中忽略这些文件,之后会生成.gitignore的文本,将该文本也忽略

常用操作

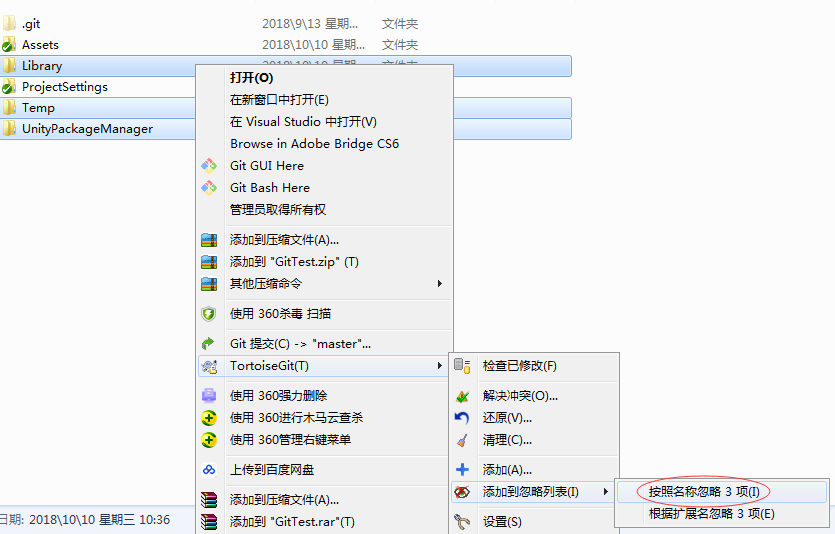
克隆:从远程库克隆项目到本地库

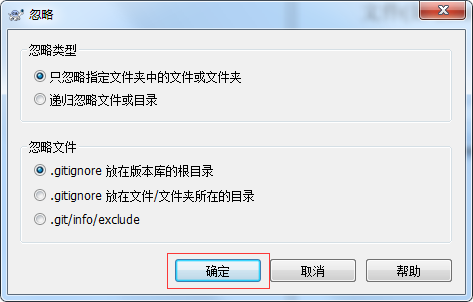
添加:使Git管理/追踪该文件(添加到暂存区)

提交:将修改或新添加的文件提交到本地库,并产生一个新版本

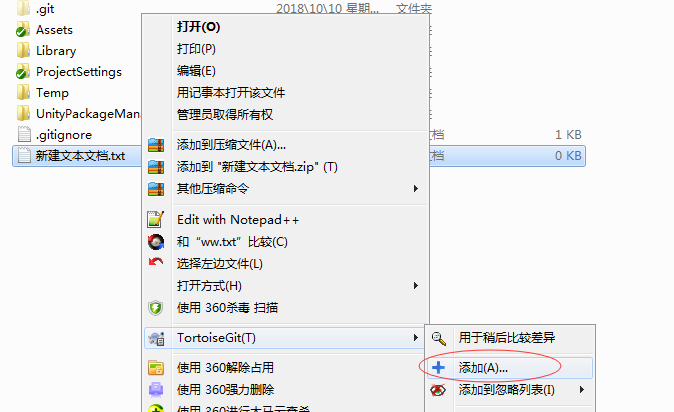
推送:将本地库版本更新到远程库

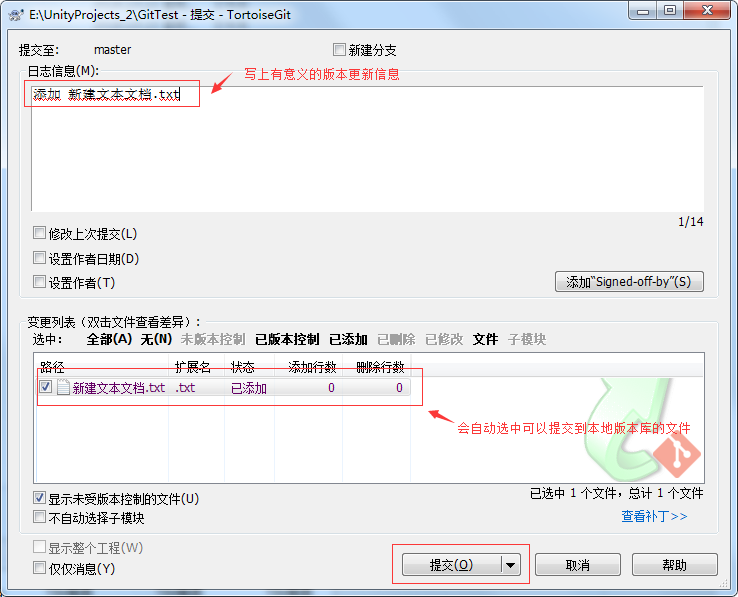
拉取:将其他成员推送到远程库的版本更新到自己的本地库

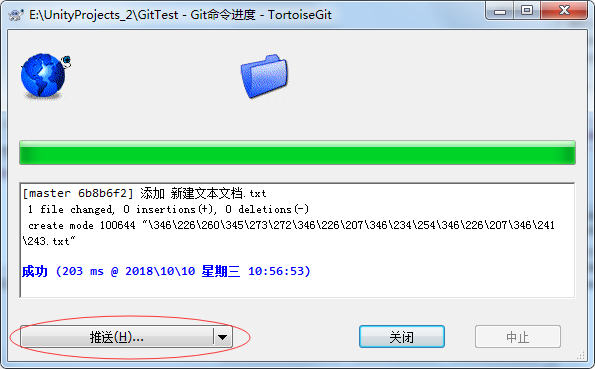




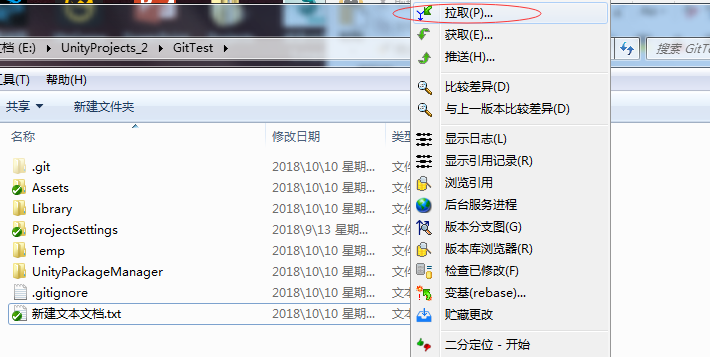
1. 新添加一个文件(下图中的txt),需要先右键添加(添加到暂存区,使Git追踪该文件),添加之后再提交(将文件添加到本地库,会升级一个版本),添加之后再推送(将本地库的版本更新提交到远程库,其他成员可以通过拉取更新到最新版)



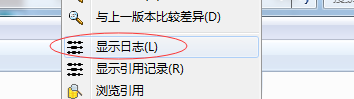


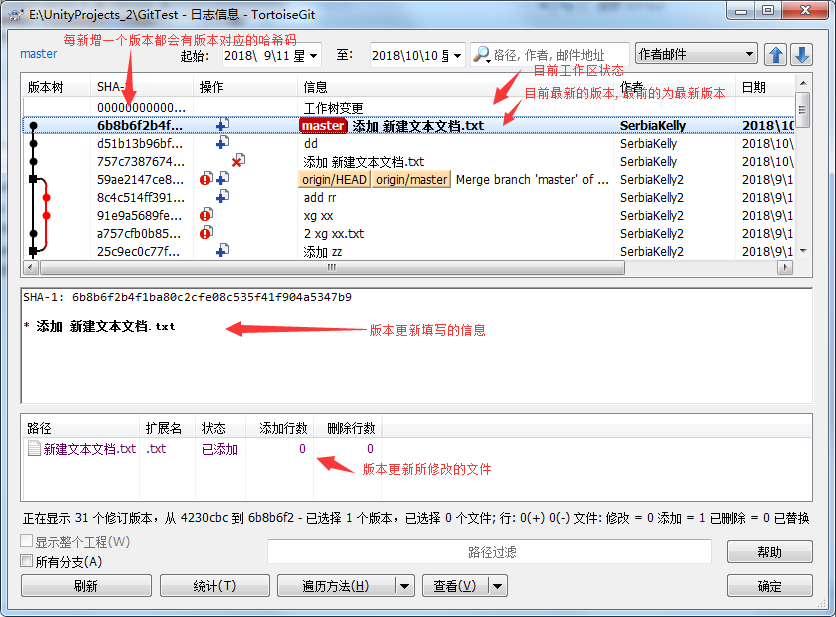


1. 其他成员通过拉取更新到最新版

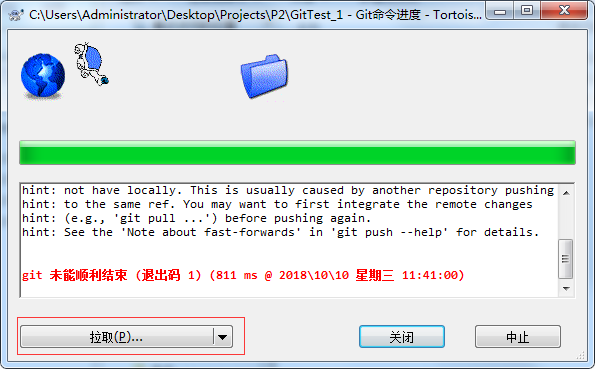


1. 查看本地库各版本





1. 多人同时开发推送时难免遇到冲突

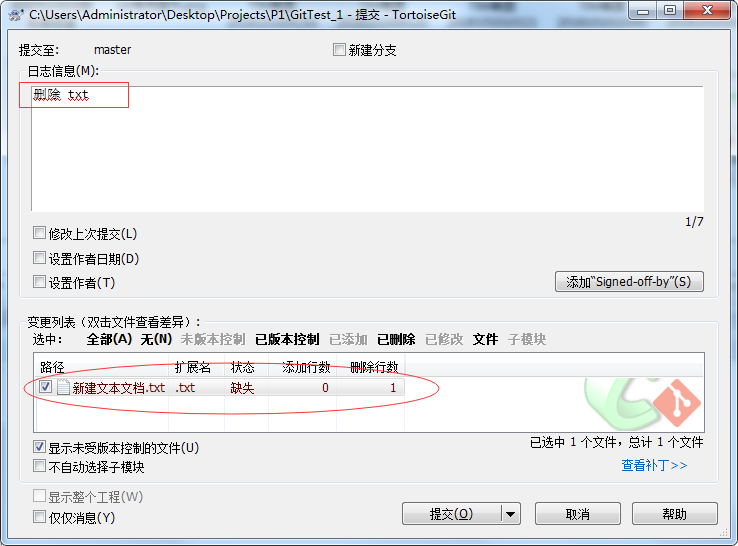


上图推送失败提示要先拉取远程库最新版本更新到本地,然后再推送自己完成的模块更新到远程库

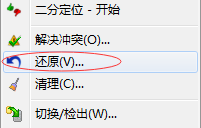
1. 删除文件

a. 没有被Git追踪的文件可以直接在文件夹中删除

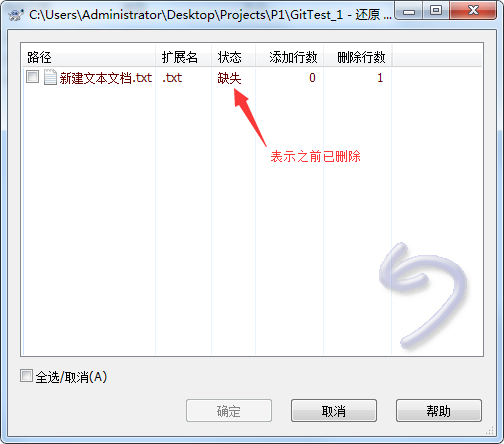
b. 被Git追踪的文件可以直接在文件夹中删除,之后需要进行一次提交



1. 还原文件,删除或修改某个文件,需要还原该文件时可以右键再点击还原弹出可还原文件的列表,进行还原

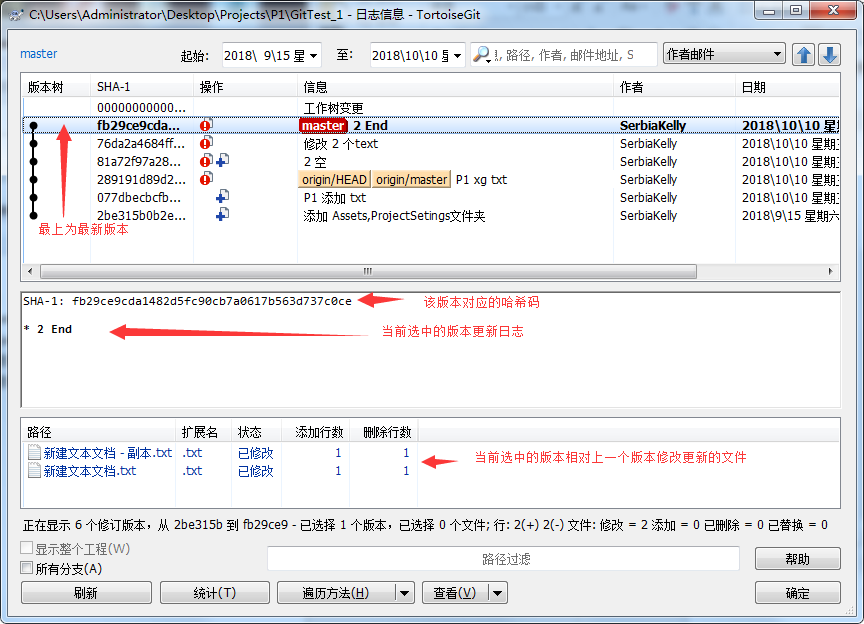


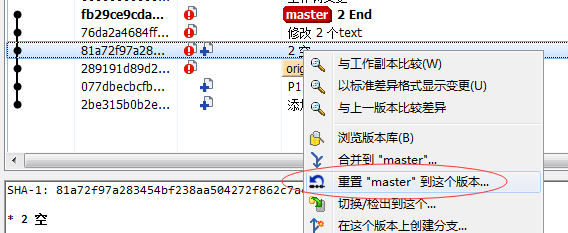
此还原为还原该文件到本地库最新版本的原文件状态,若要还原到指定版本可右键再点击”显示日志”,选中版本->文件->还原到此版本.



1. 回退版本

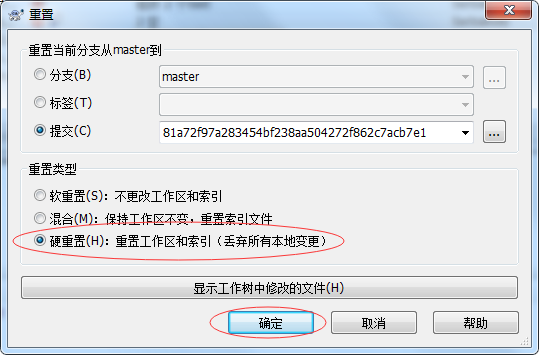
进入显示日志界面,显示本地库所有版本列表,



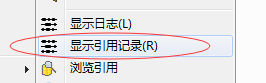


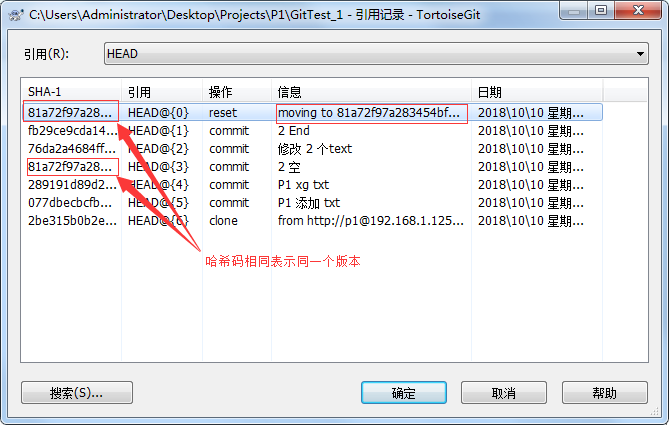
右键要回退到的版本点击”重置master…”

一般选中硬重置(整个项目都会重置到这个版本)

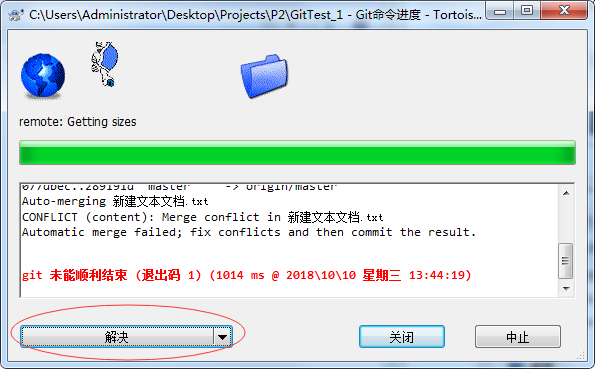


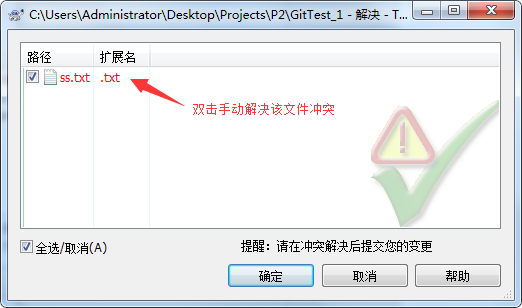
若又要回退到上一步那个版本,右键点击”显示应用记录”,可以查看到所有回退版本记录,同样是右键版本选择硬重置即可回退版本





1. 双方修改同一个文件时解决冲突





合并时有冲突一定要手动解决冲突,若使用默认合并会导致文本或资源混乱