# 实验一：Git和Github的使用

## 实验目的：

1.掌握github的使用,

2.掌握在windows环境下git的安装

3.学会通过git创建本地仓库，并在本地仓库创建文件夹，创建文件，然后推送到github。

4.学会通过git拉取github的文件，修改并推送至github仓库

扩展内容：

1. 了解基于master主分支上新建develop分支，并在上面进行文件的增删改操作并推送到github。

2.了解学习通过Tag标记版本号，并提交推送到分支.

3.学会在IntelliJ IDEA 开发工具中使用git和github

## 实验要求：

1. 掌握github的使用
2. 注册github个人账户
3. 在github个人账户中创建仓库
4. 掌握git的安装
5. 在git官网上下载相应的安装包
6. 通过安装包进行安装，并检查是否安装成功
7. 环境配置，设置用户名，邮箱。
8. Git创建本地化仓库，并推送到github
9. 初始化本地仓库
10. 添加文件到版本库
11. 把添加的文件提交到版本库，并填写提交备注
12. 把本地库与远程库关联
13. 推送本地文件到远程库
14. （扩展内容）创建一个tag标签，推送到github仓库

四 通过git拉取github的文件，修改并推送至github仓库

1. 获取github仓库的地址，通过gitl拉取该地址的文件到本地
2. 新增或者修改本地文件
3. 推送到github远程仓库，到github仓库中检查是否推送成功

五 （扩展内容）基于master主分支上新建develop分支

1.master分支上新建develop分支

2.检出develop分支

3.进行文件的操作，开发等，然后推送到github的origin/develop

六 （补充内容）在IntelliJ IDEA 开发工具中使用 git的常见的应用场景

假设小组中有两个人，组长小张，组员小袁

场景一：小张创建项目并提交到远程GitHub仓库

场景二：小袁从远程[Git](http://lib.csdn.net/base/git)仓库上获取项目源码

场景三：小袁修改了部分源码，提交到远程仓库

场景四：小张从远程仓库获取小袁的提交

场景五：小袁接受了一个新功能的任务，创建了一个分支并在分支上开发

场景六：小袁把分支提交到远程Git仓库

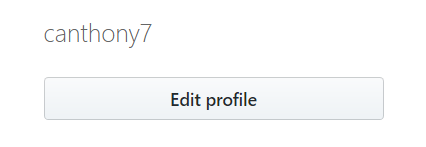
场景七：小张获取小袁提交的分支

场景八：小张把分支合并到主干

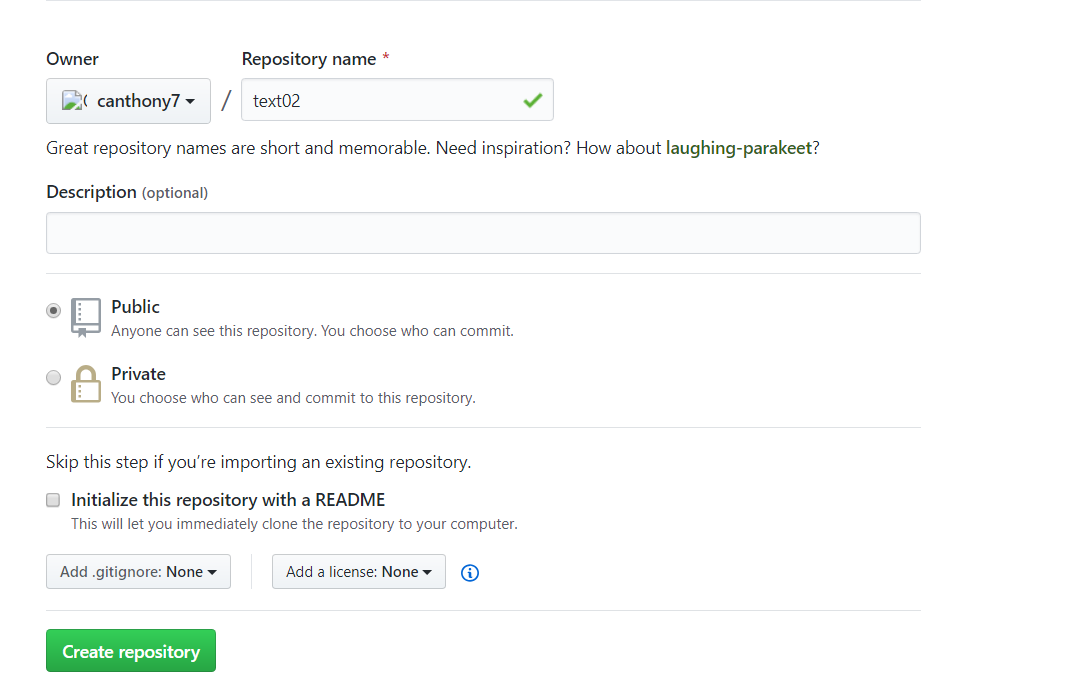
## 实验过程及结果

一：掌握github的使用

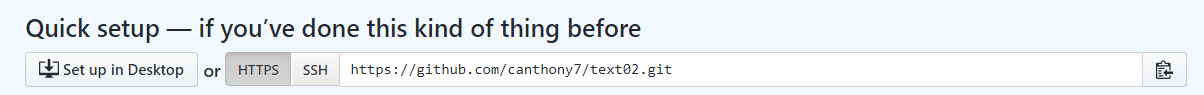
1. 之前账户已经创建好



1. 创建仓库



仓库地址：

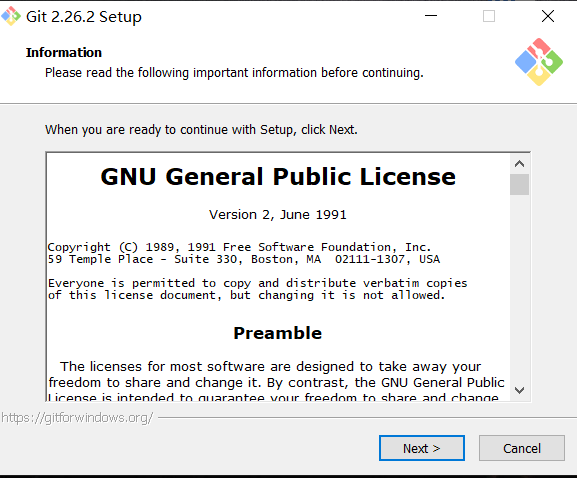


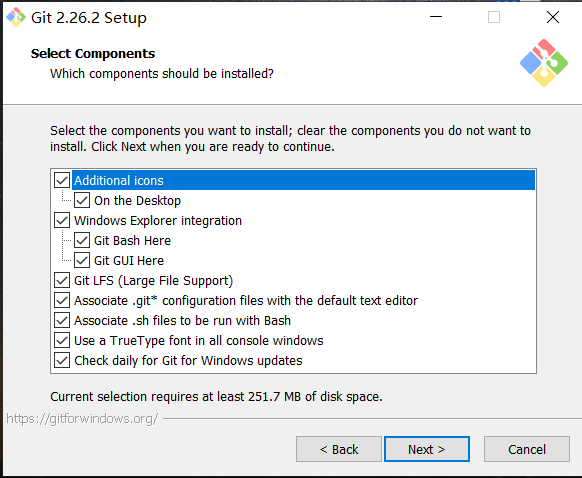
二：掌握git的安装

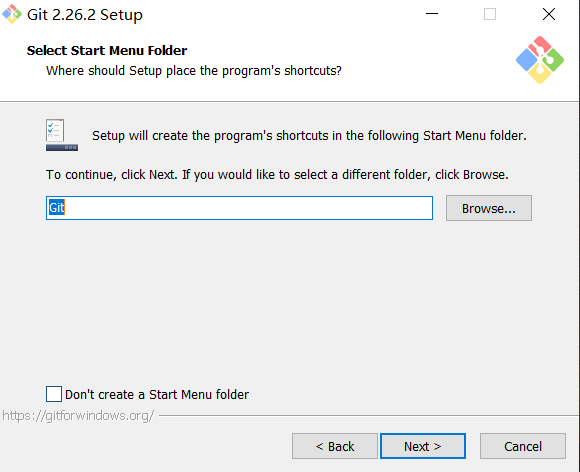
1. 安装包

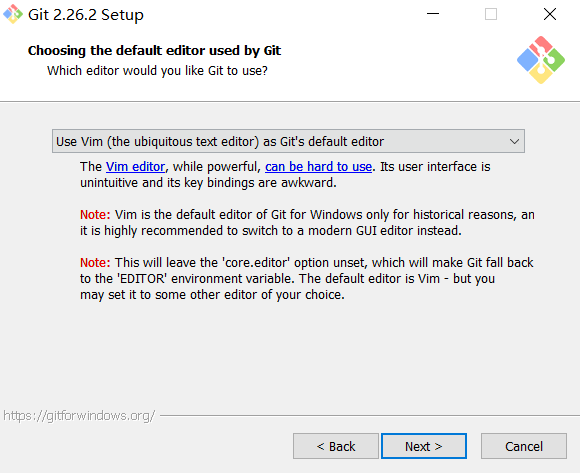


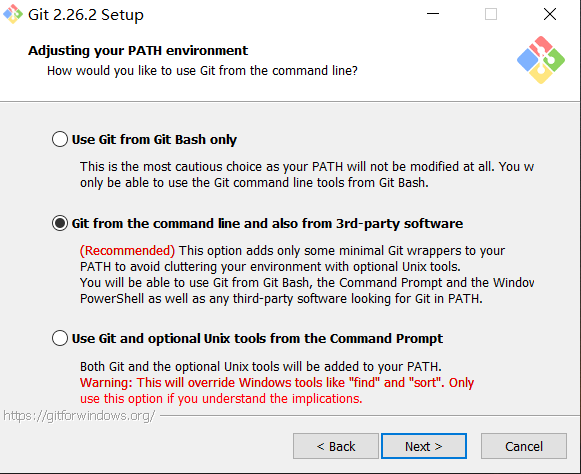
1. 安装

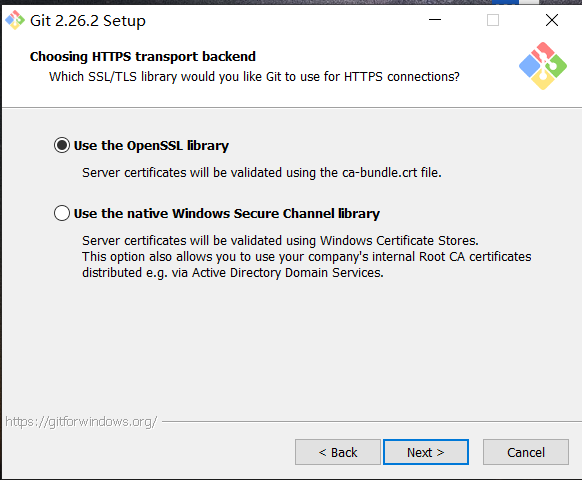


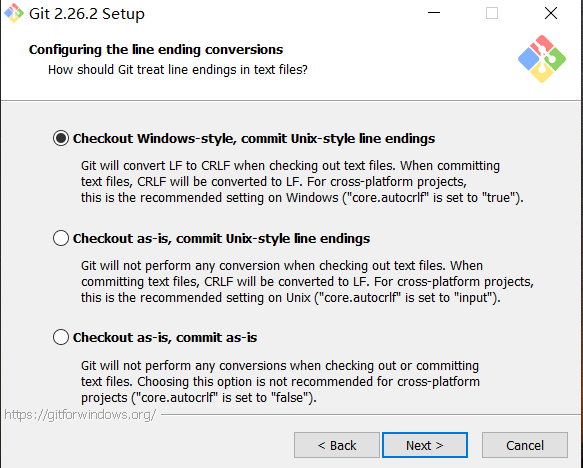


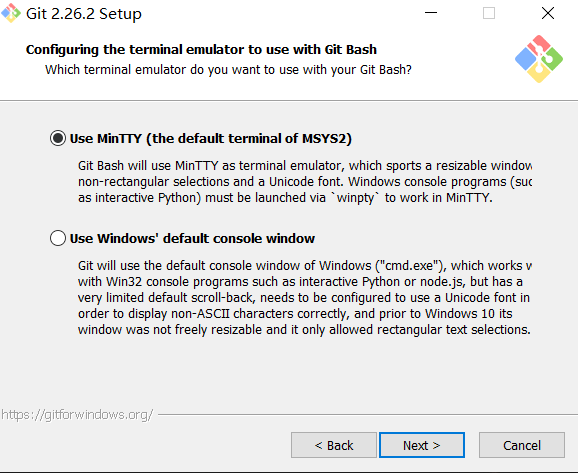


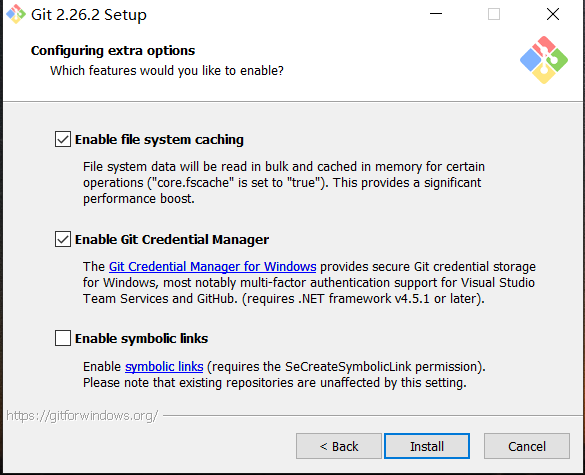






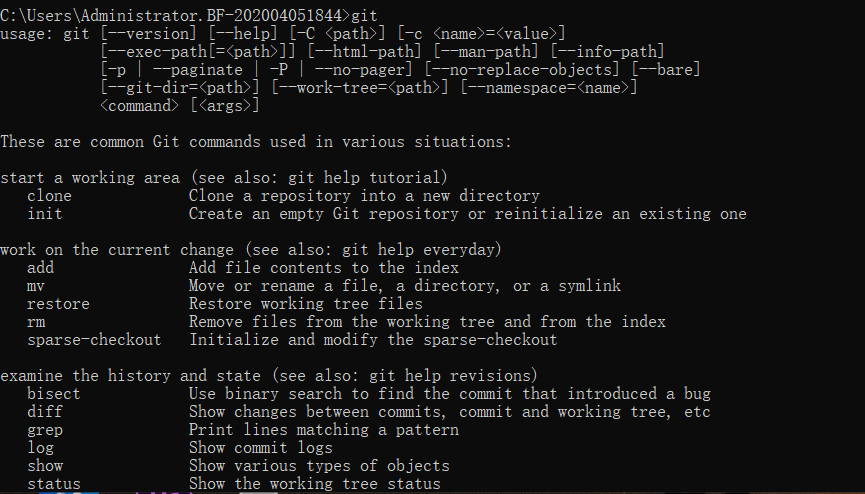




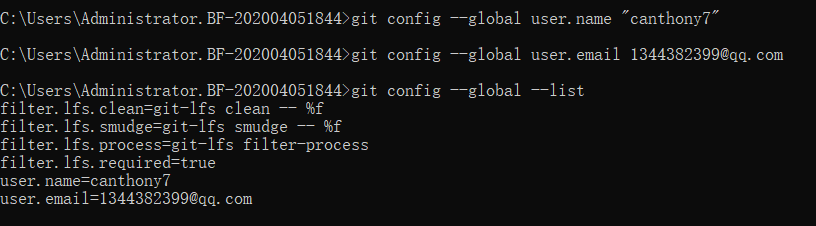




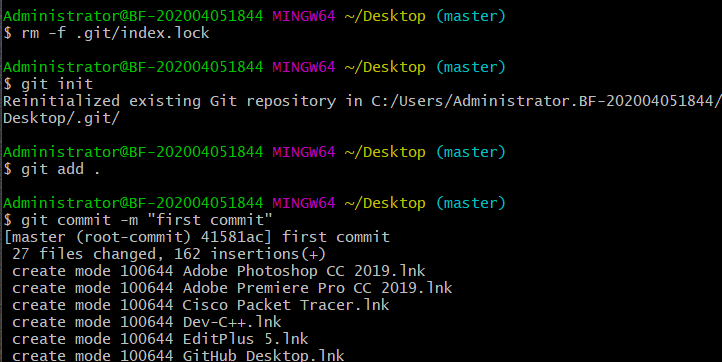
检查：



1. 配置用户名邮箱



三：Git创建本地化仓库，并推送到github

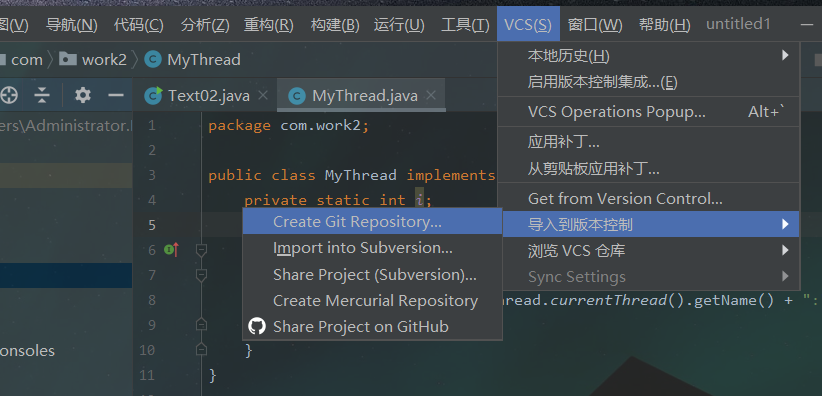


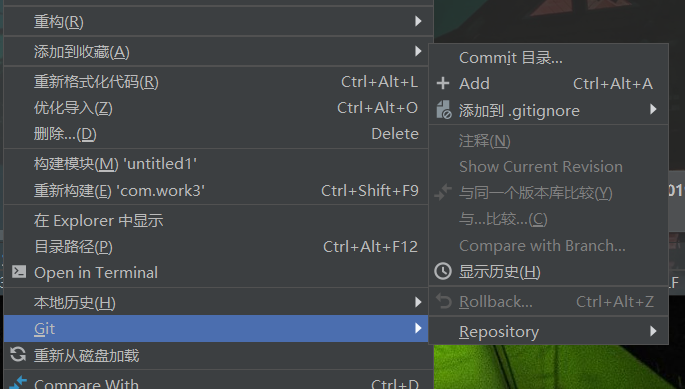
四：通过git拉取github的文件，修改并推送至github仓库

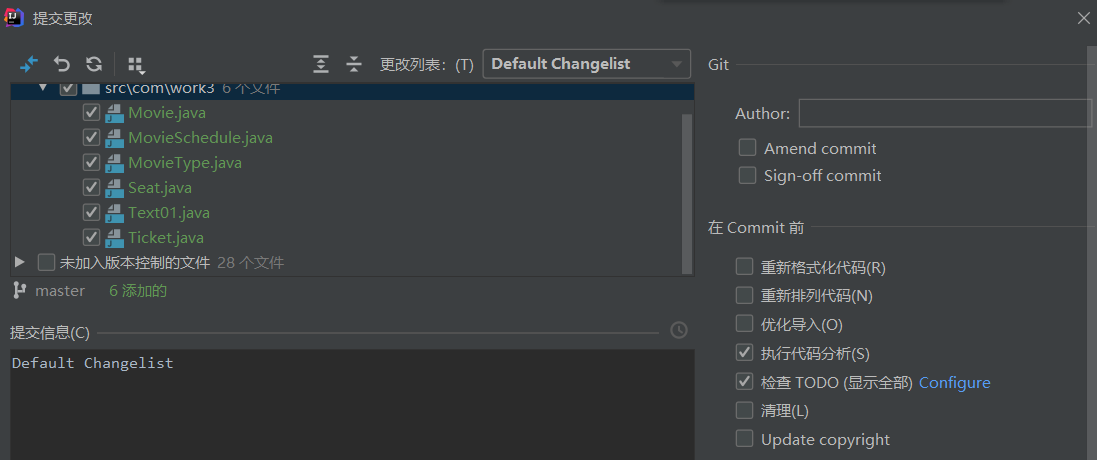
五：（扩展内容）基于master主分支上新建develop分支

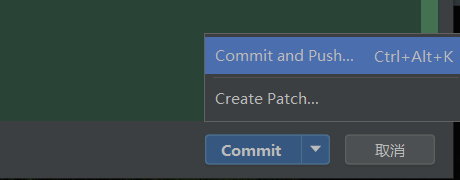
六：（补充内容）在IntelliJ IDEA 开发工具中使用 git的常见的应用场景

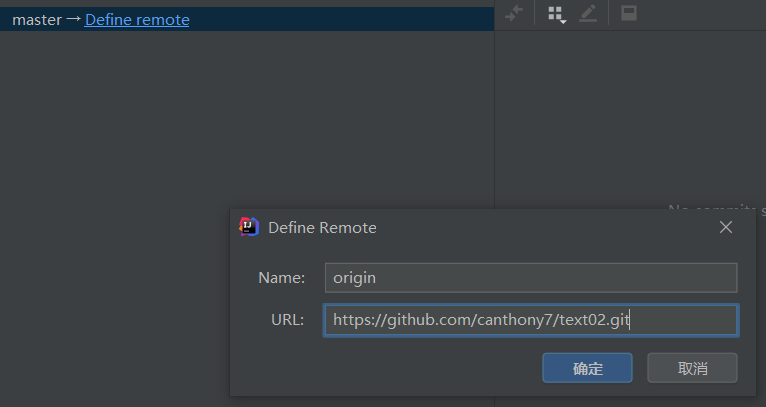
场景一：

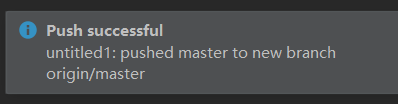




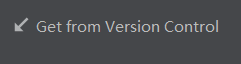


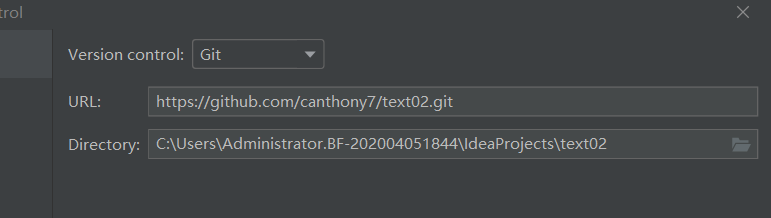


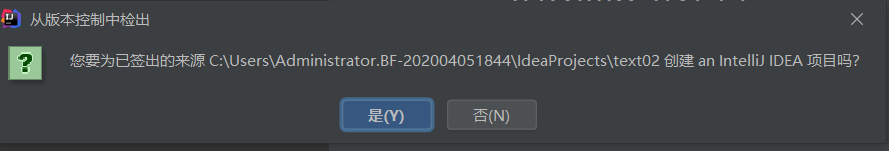




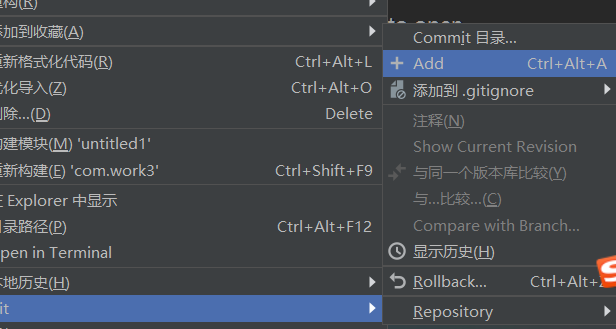
场景二：

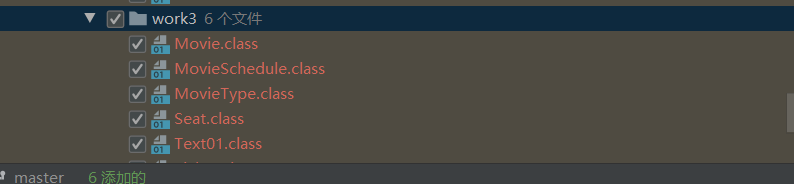




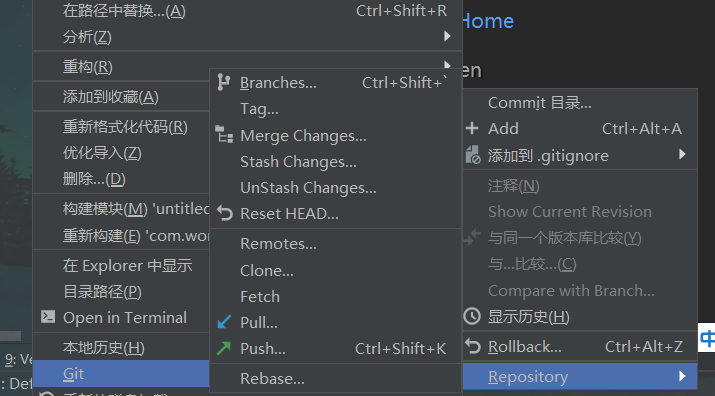


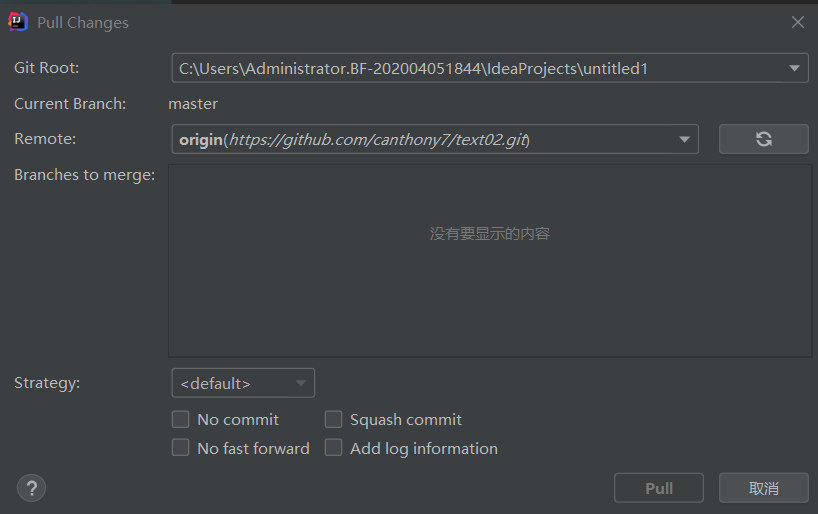
场景三：



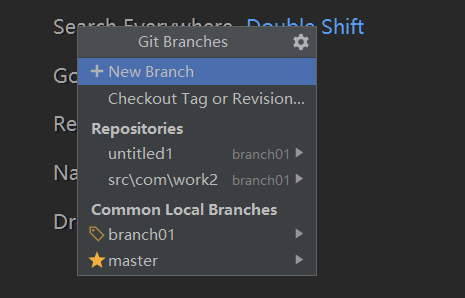


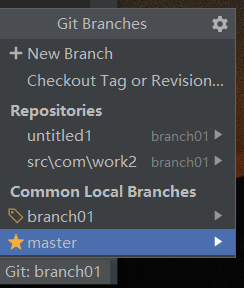
场景四：

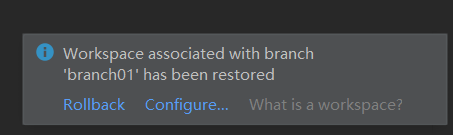




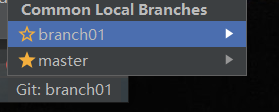
场景五：

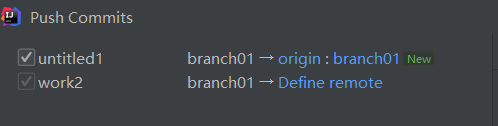




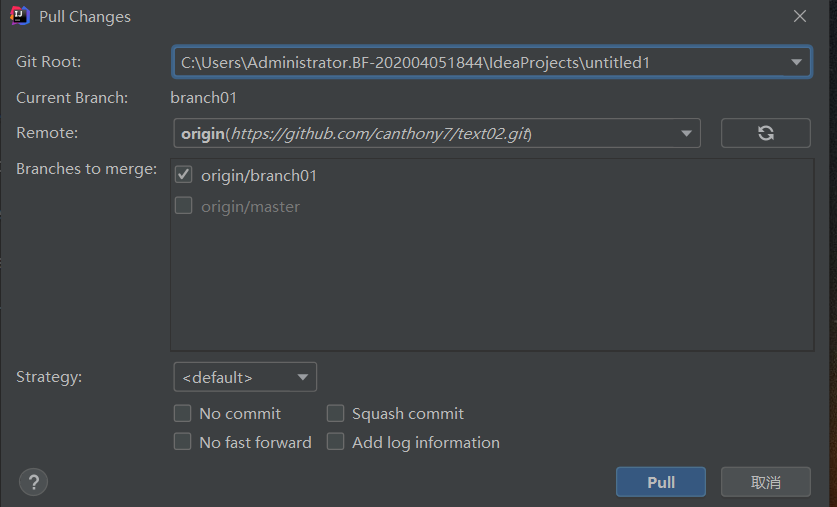
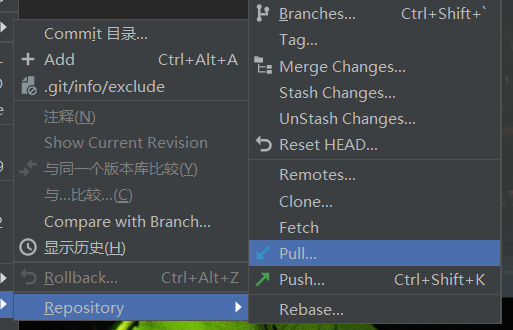


场景六：





场景七：



场景八：

