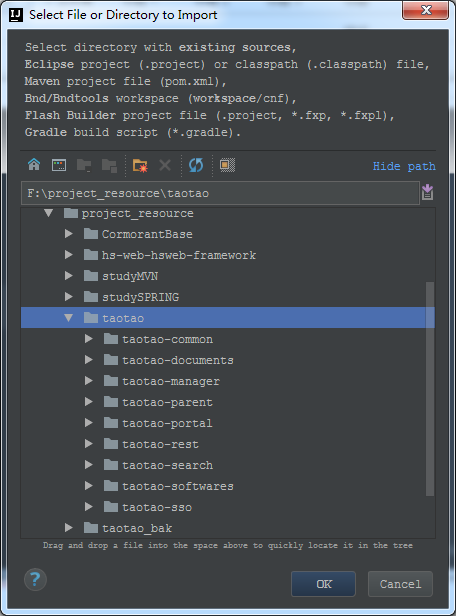
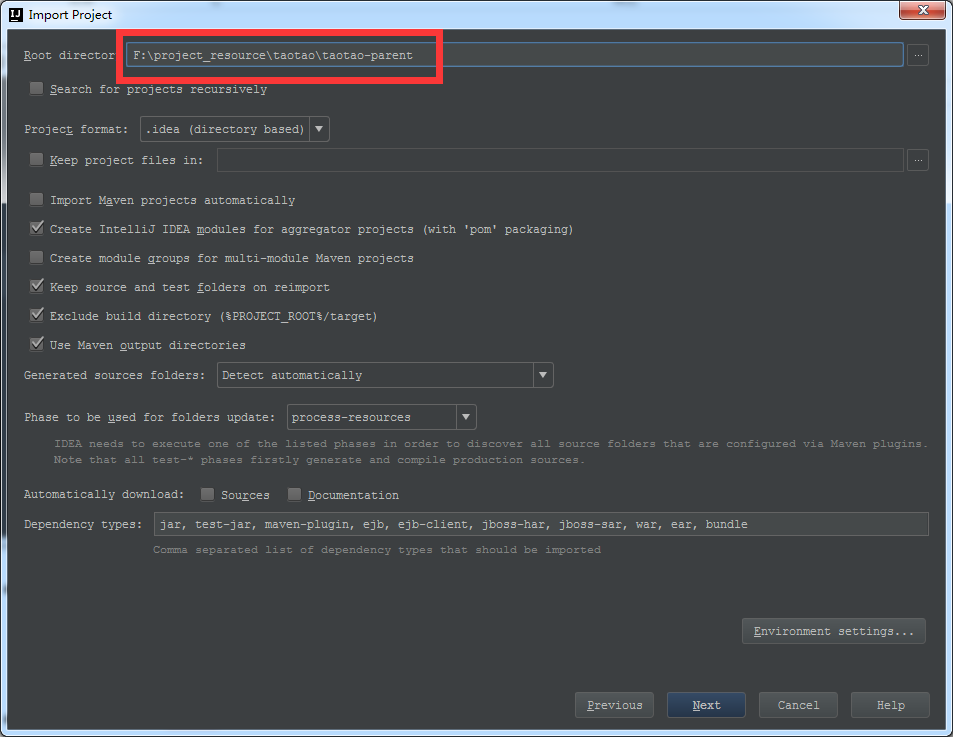
项目分享到GitHub上面

# 导入项目

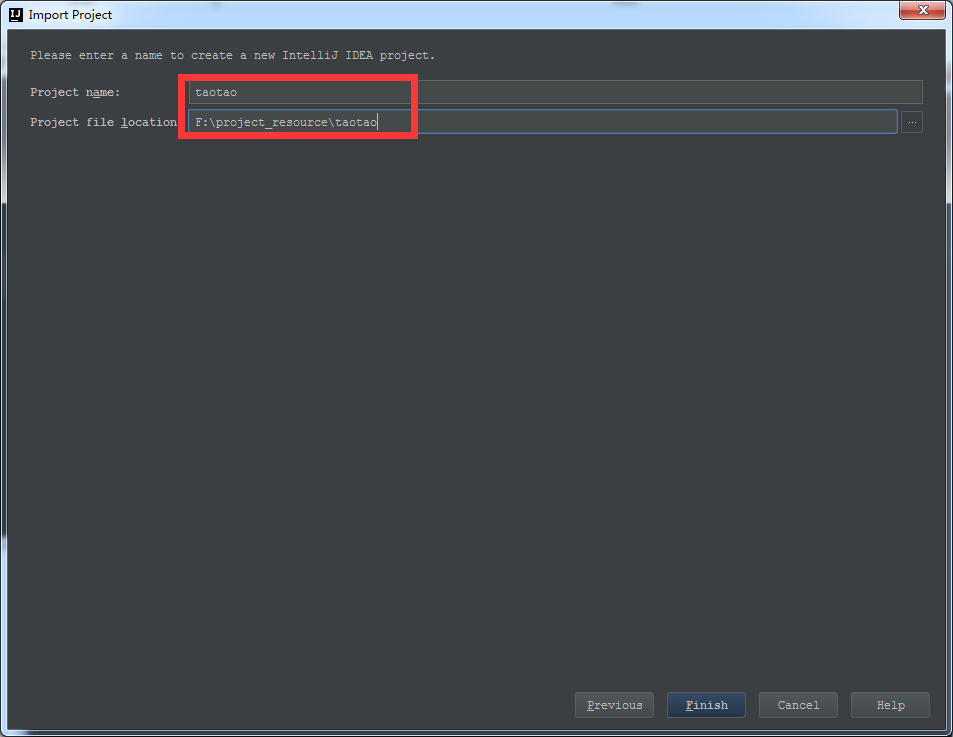
## 导入根目录



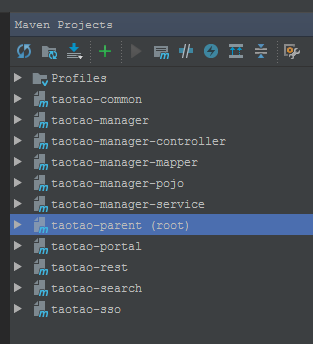
## 选择根目录



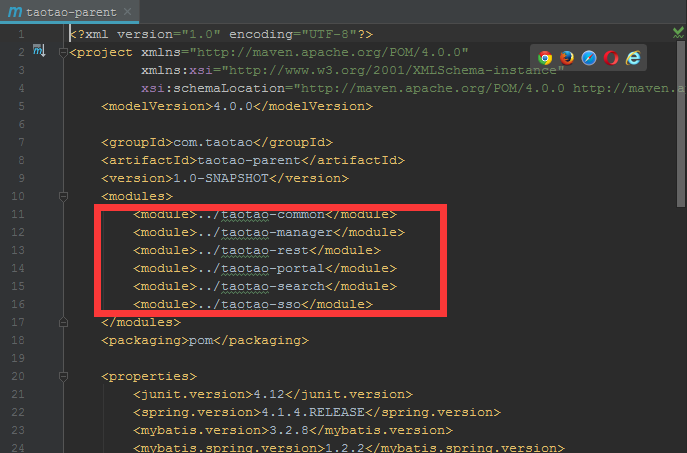
## 设置项目名称和项目文件位置



## 查看Maven

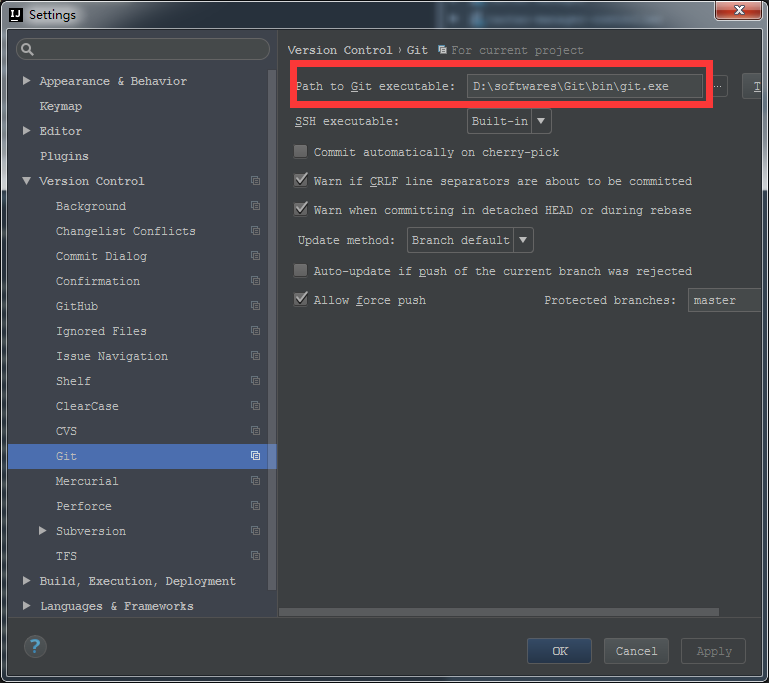


## 查看Pom文件

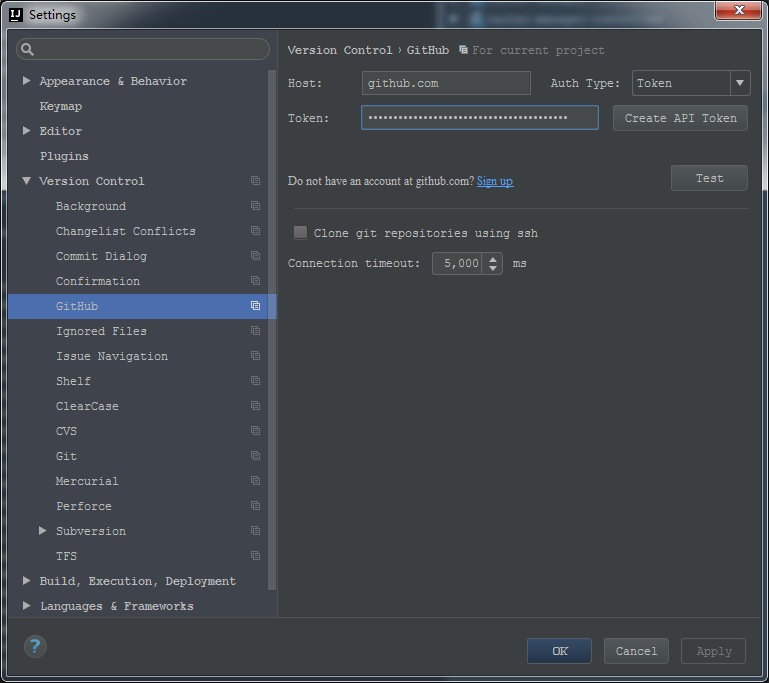


# 设置Intellij Idea

## 设置Git



## 设置GitHub



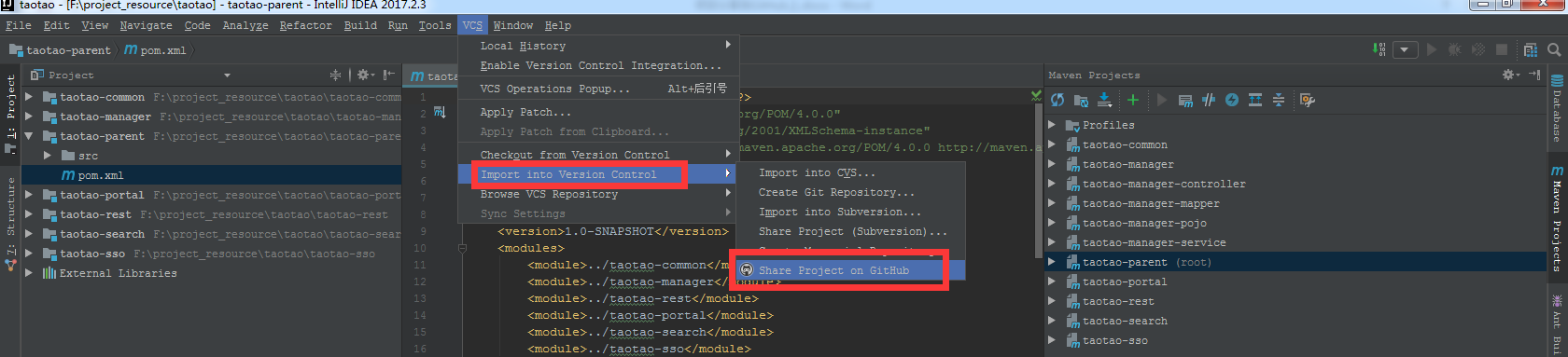
# 添加ignore文件

|  |
| --- |
| \*.class  # ignore Maven generated target folders #  target  # ignore idea files  \*\*/pom.xml.versionsBackup  \*\*/target/  \*\*/out/  \*\*/log/  \*.class  # Mobile Tools for Java (J2ME)  .mtj.tmp/  .idea/  /nbproject  \*.ipr  \*.iws  \*.iml  # Package Files #  \*.jar  \*.war  \*.ear  \*.log  # virtual machine crash logs, see http://www.java.com/en/download/help/error\_hotspot.xml  hs\_err\_pid\*  \*\*/transaction-logs/  # ignore eclipse files  .project  .classpath  .settings  .metadata |

# 分享到GitHub

## 分享

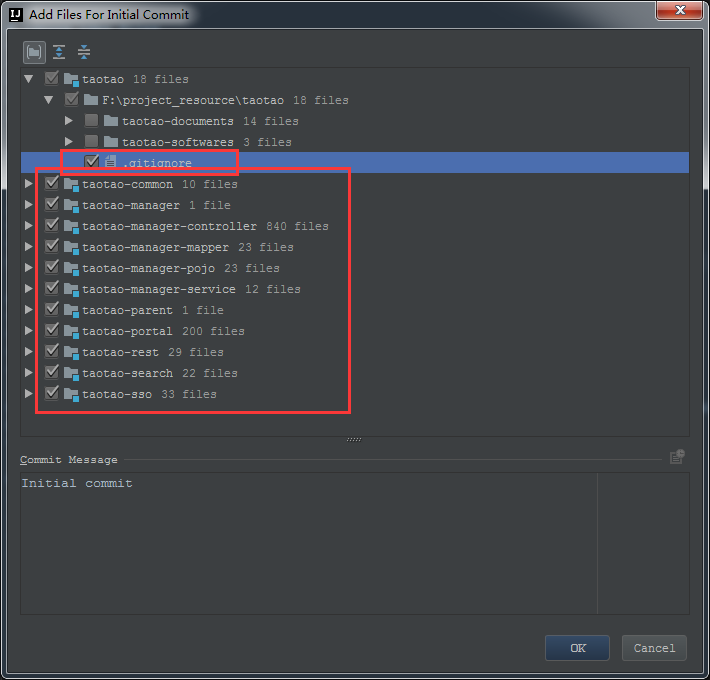
第一步：



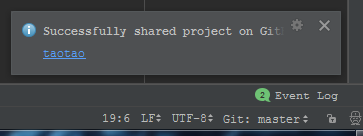
第二步：



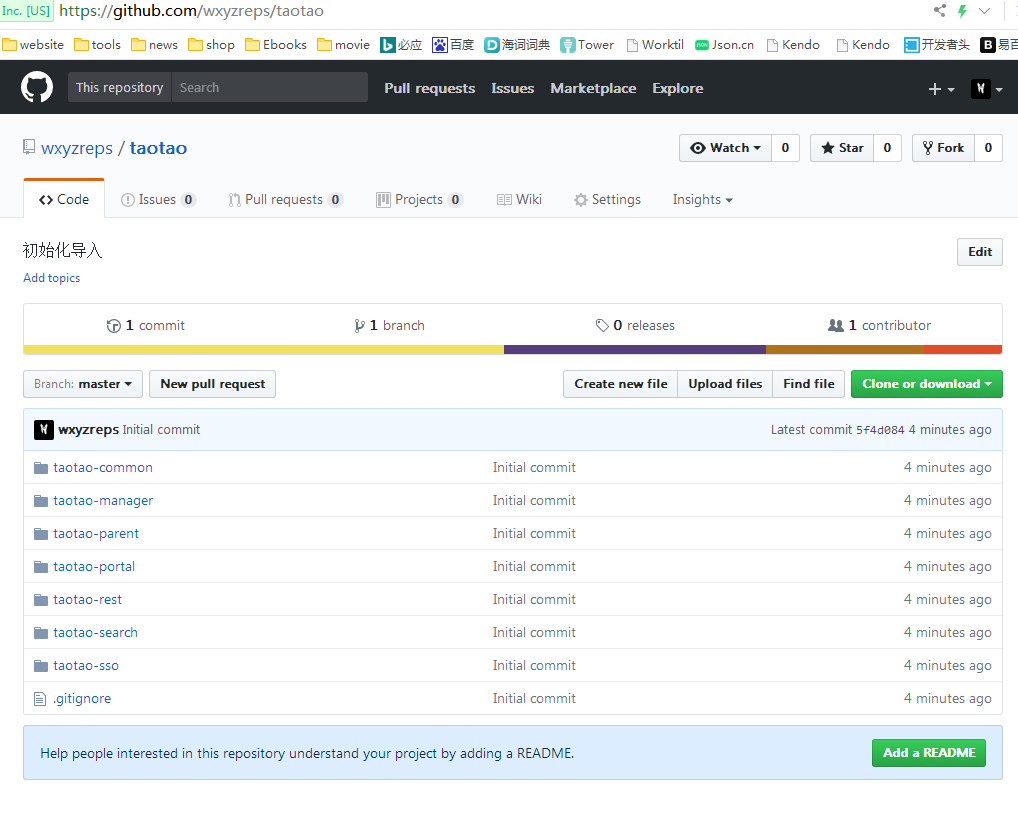
## 初始化提交



## 提示成功



# 测试

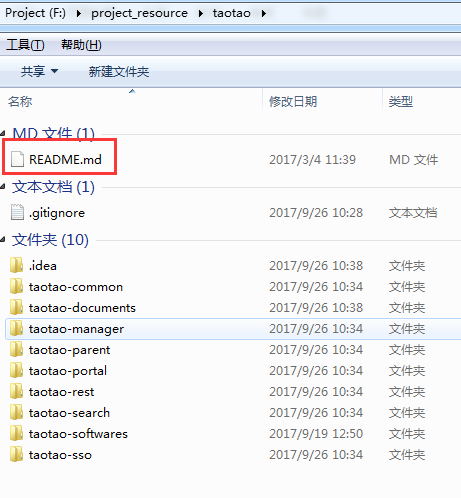


# IDEA与GitHub基本操作

## 添加新文件

### 拷贝

将文档拷贝到指定位置：



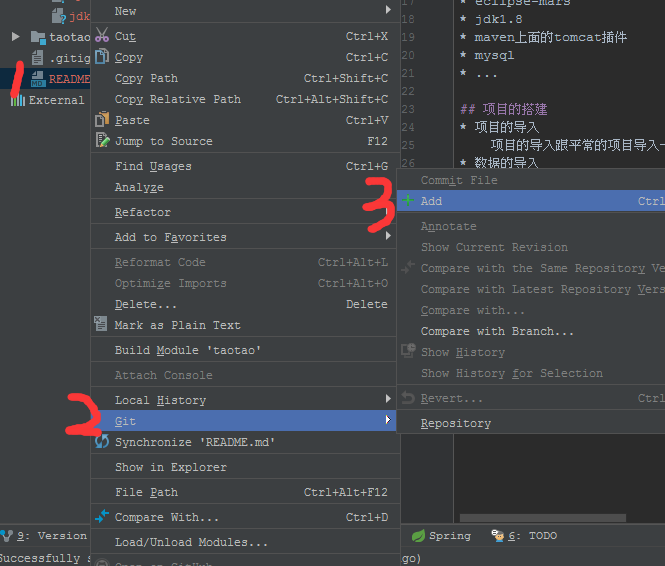
新文件在IDEA中是这个样子的：



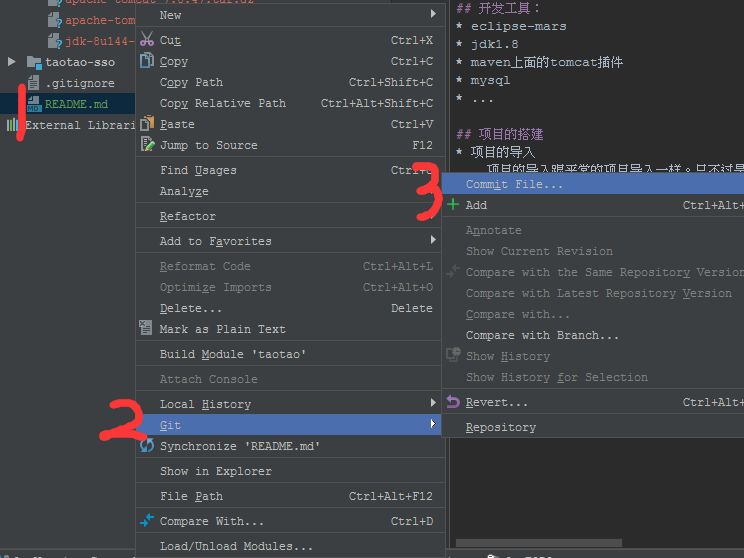
### 提交到GitHub上

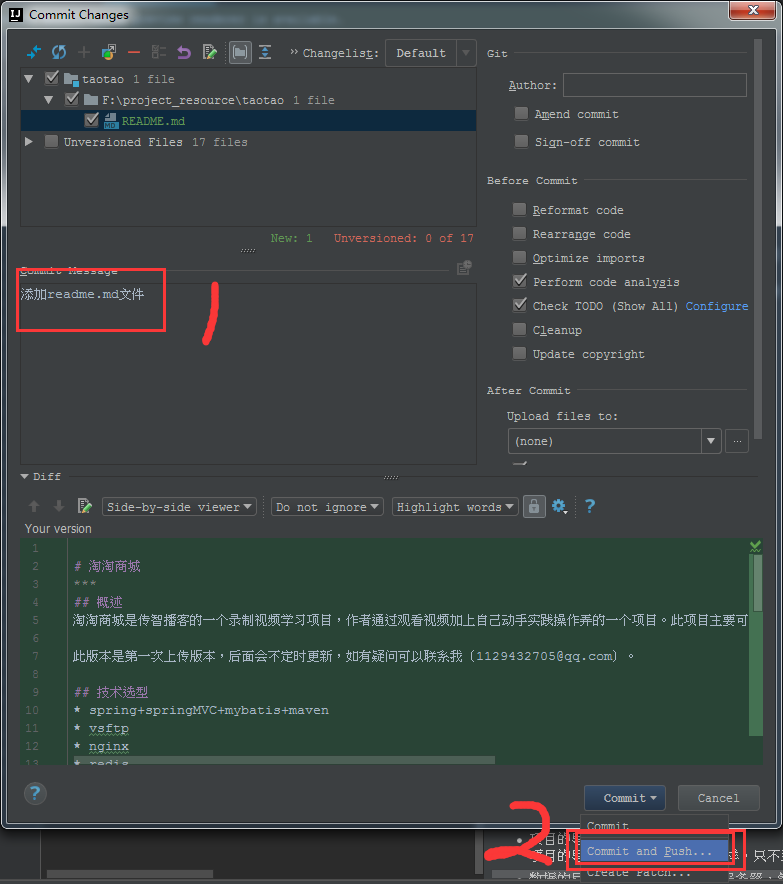
#### 添加

添加前后会发现文档颜色发生变化。

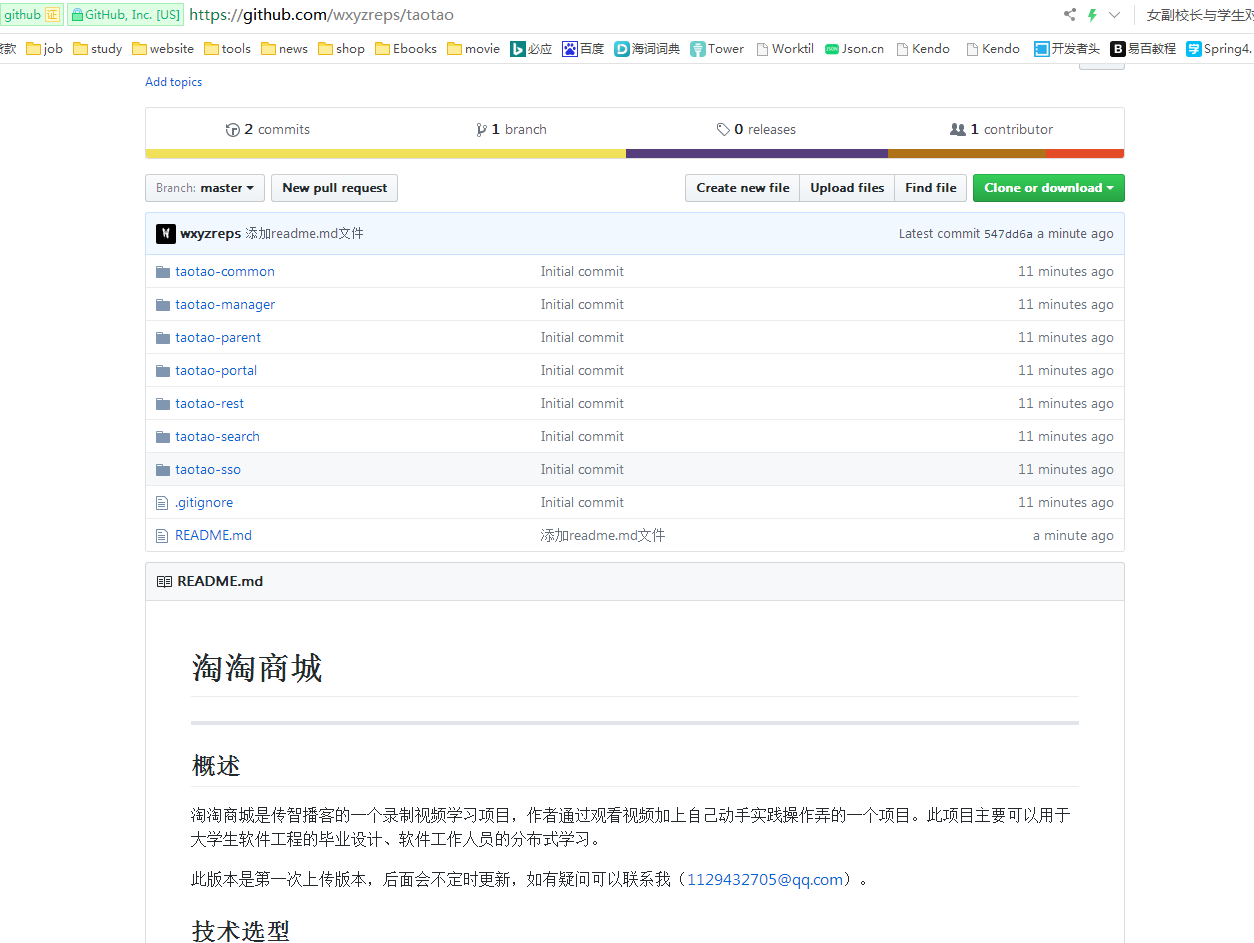


#### 提交并推送





### 测试



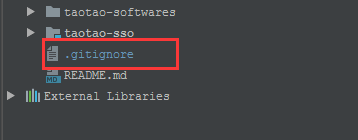
## 修改文件

### 修改文件内容

修改文件内容，添加对word备份文档的忽略：

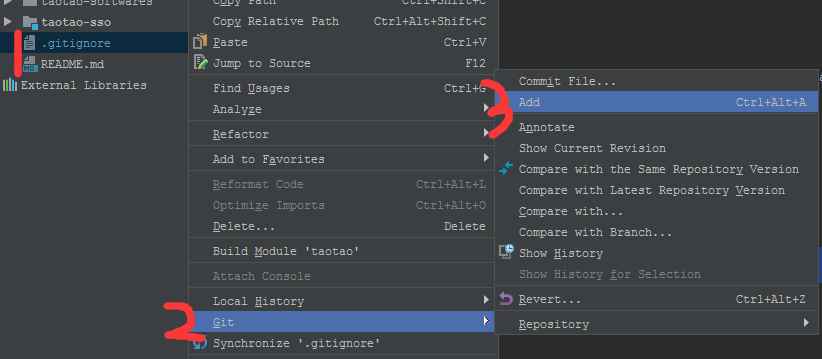
|  |
| --- |
| # ignore docx backup files ~$.\*.docx ~$.\*.doc |

添加完毕之后ignore的状态

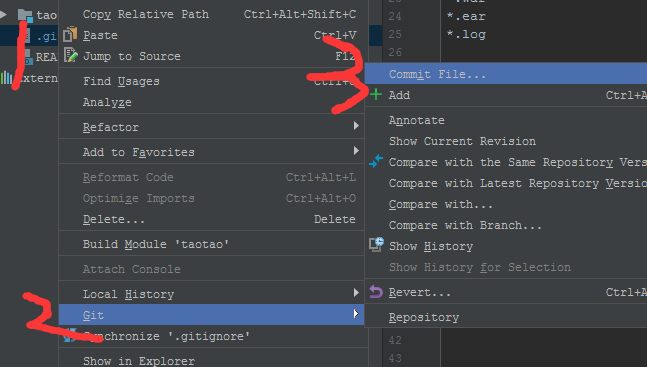


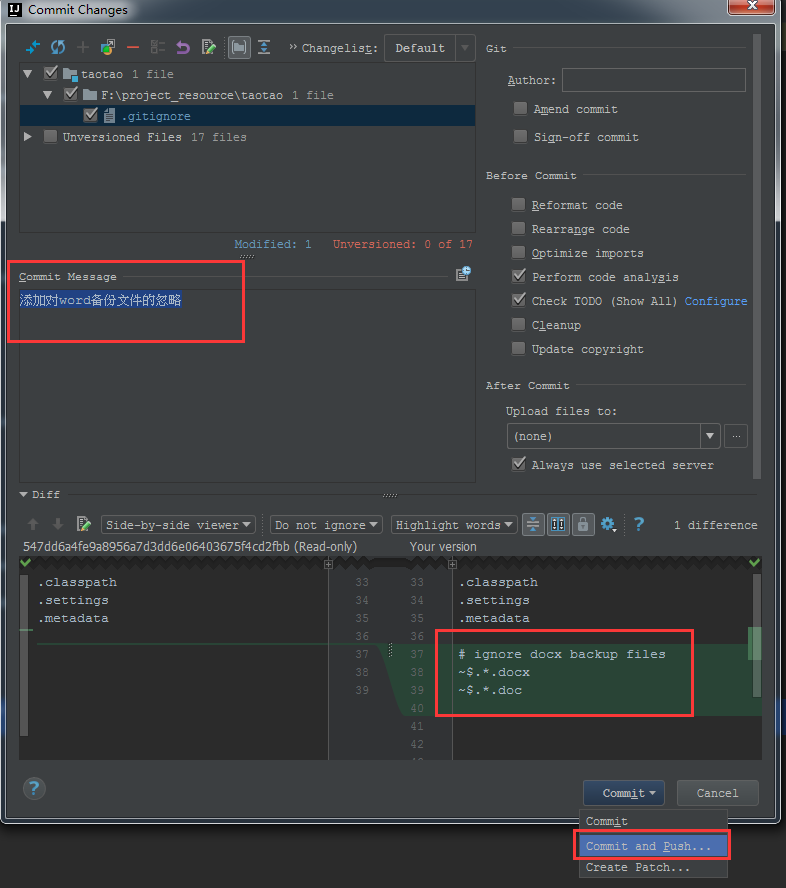
### 提交到GitHub上

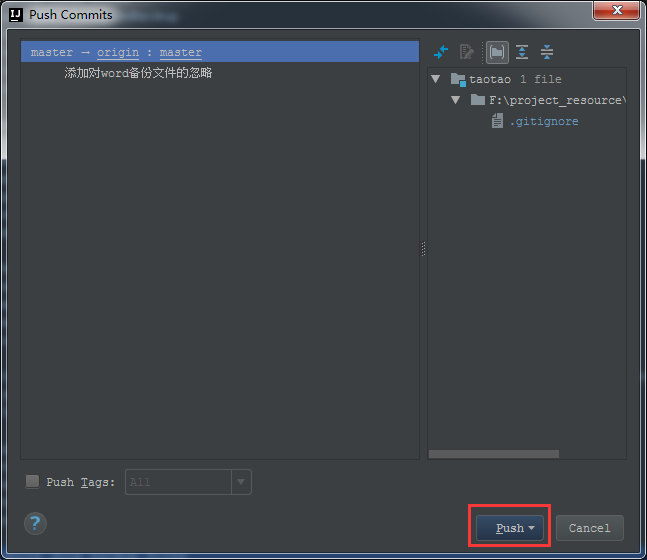
#### 添加



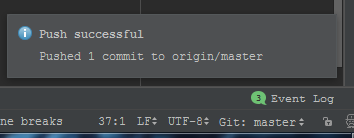
#### 提交并推送



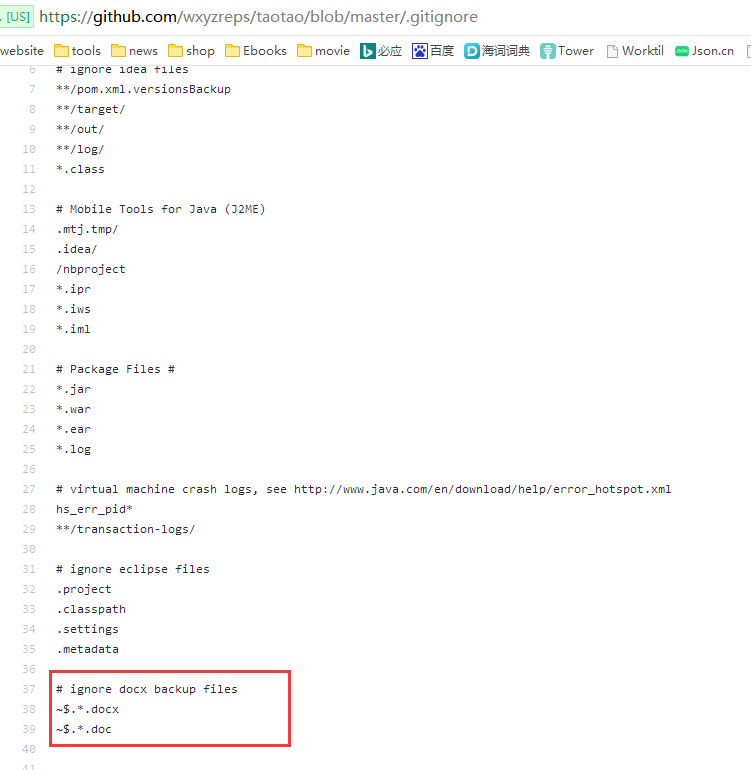




提交并推送成功



### 测试

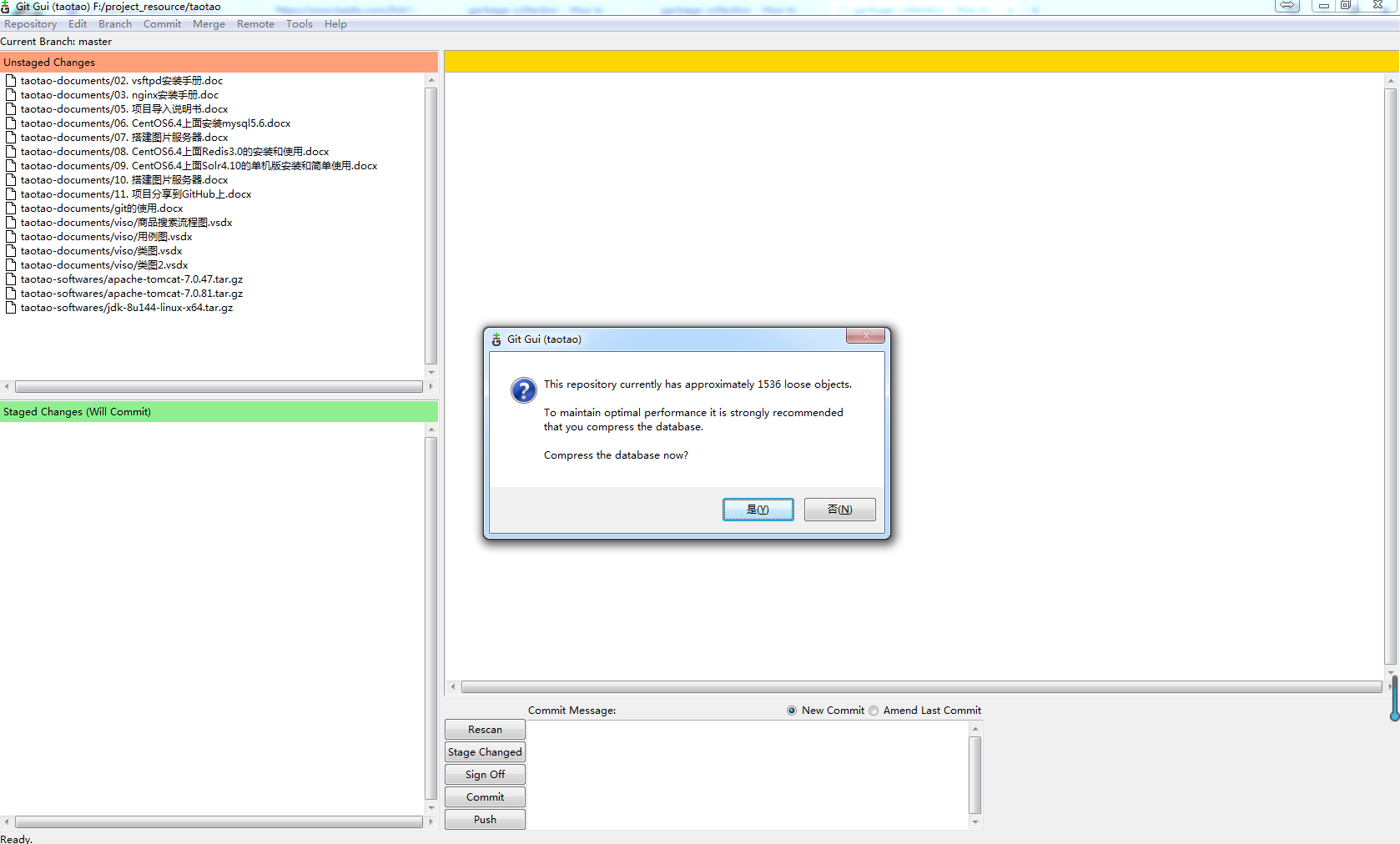


## 反追踪文件

假如不小心在IDEA中添加了不应该分享到项目中的文件，可以使用GitGui工具进行移除，前提是IDEA中添加了这些文件，换句话说就是IDEA中对这些文件执行过add操作。

### 打开Git GUI

在文档目录右键打开Git GUI工具

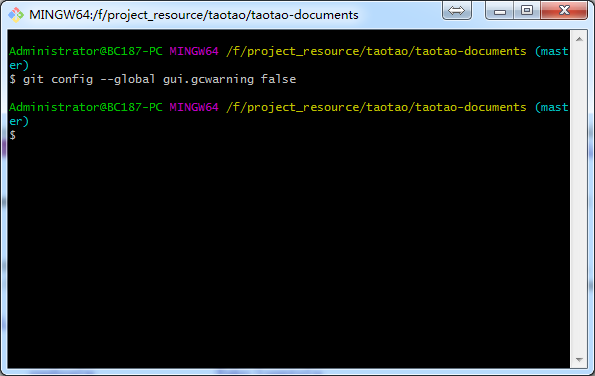


打开之后有一个提示，不知道为什么会有这个提示。选择“否”。

解决方式：

关闭GitGUI，然后在某一文件夹下面右键打开Git Bash，最后输入：

|  |
| --- |
| git config --global gui.gcwarning false |



再次打开GitGUI客户端就没有这个提示了。

### GitGUI工具简介

自己摸索。

### Untruck文件

自己摸索。

## 从GitHub上面拉去项目

### 设置Intellij IDEA

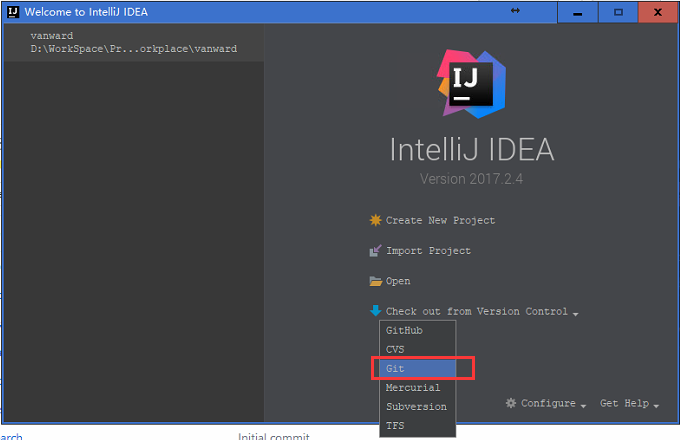
#### 设置Git工具

#### 设置GitHub

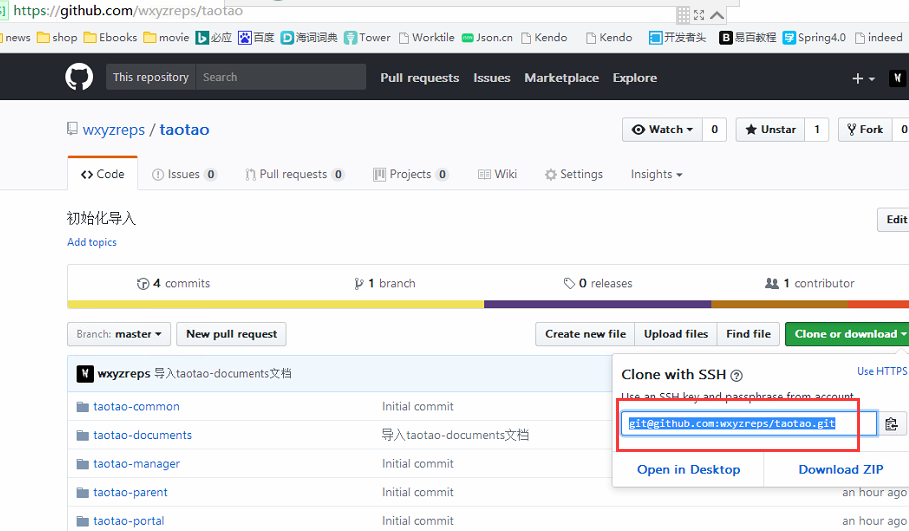
以上设置IDEA的方式和第二章节一样，这里不再赘述。

### CheckOutFromVCS

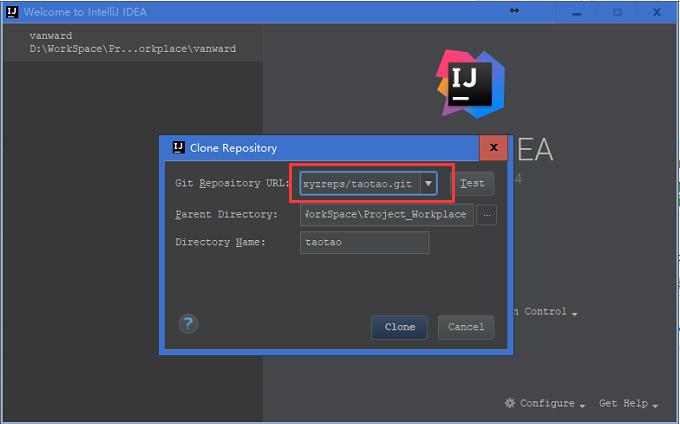
#### 打开IDEA



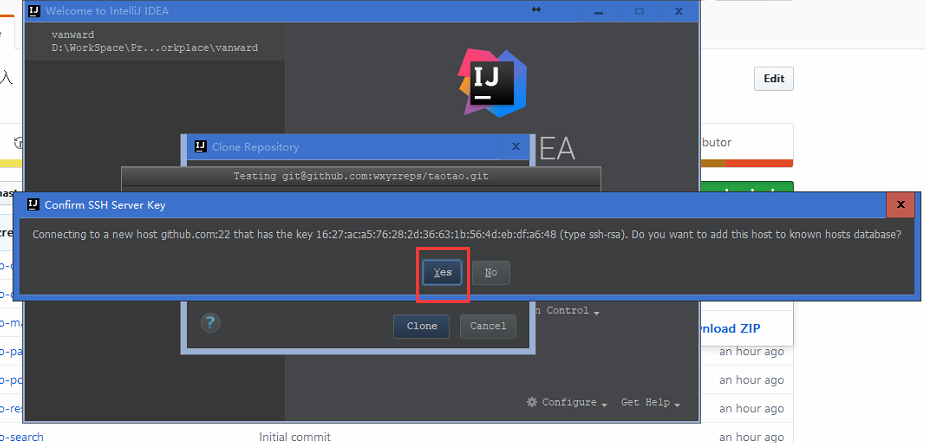
#### 复制项目地址



将复制的项目粘贴到URL中，并点击Test进行测试。



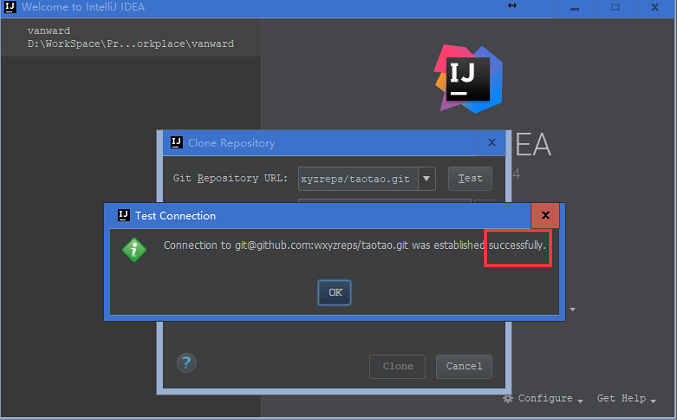
选择Yes，意思是连接到办公室里面的电脑。



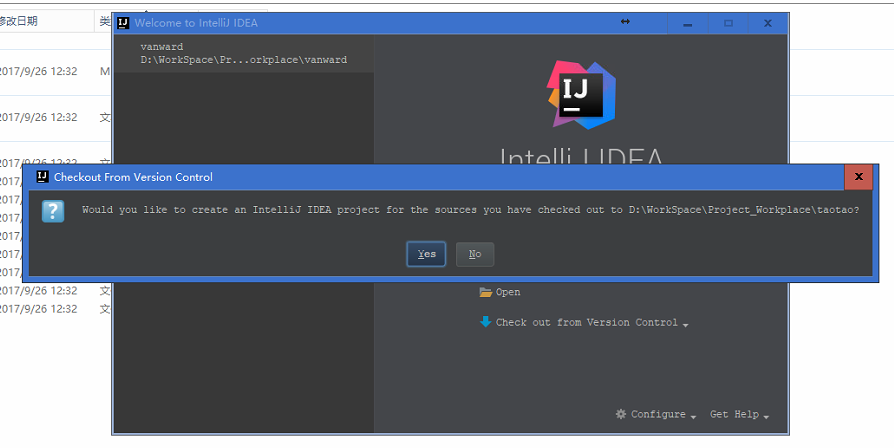
如果在这一过程中没有成功，去查看一下家目录下面是否有.ssh文件。没有的话重新生成

|  |
| --- |
| ssh-keygen -t rsa -C "438123371@qq.com" |

生成完成之后再去GitHub上面更新一下就行了。



#### 克隆

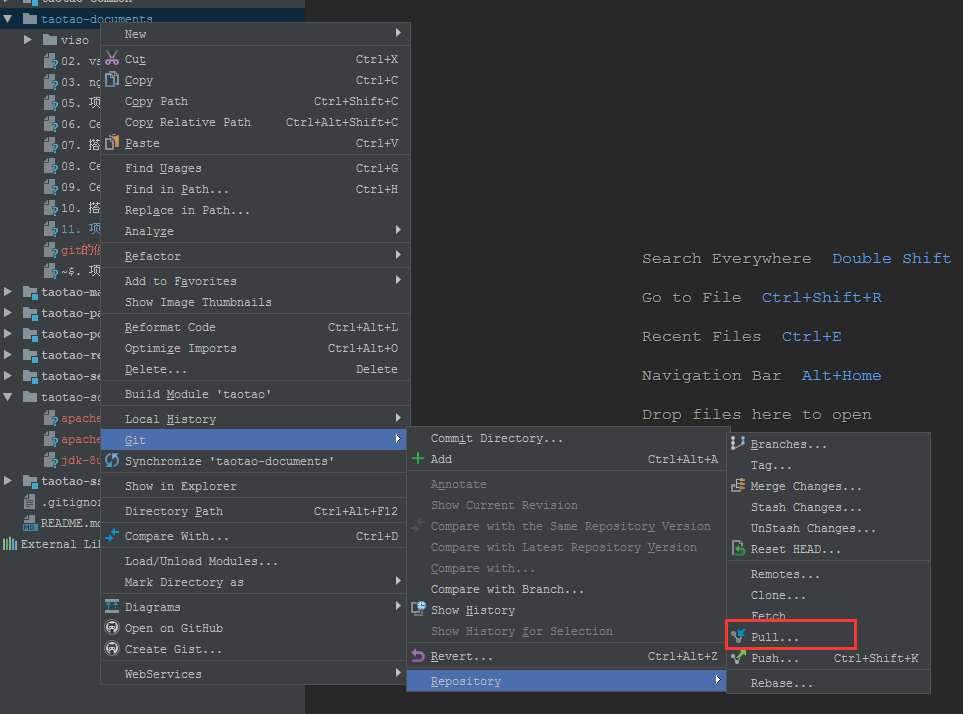


因为上传到GitHub上面的代码中没有IDEA的配置文件，只有源代码，所以，这一步选择Yes。

### 导入项目

这一步骤就和第一章节中的导入项目一样了。这里不再赘述。

## 从GitHub上面拉去文件



# Git操作问题及解决方案

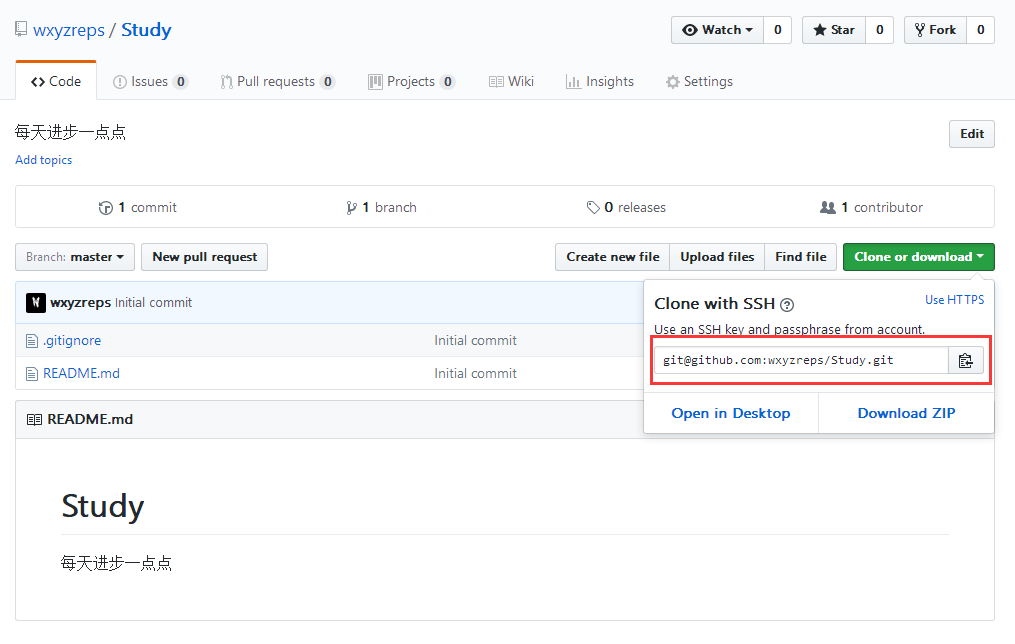
GitGUI乱码

# 克隆

## 建库

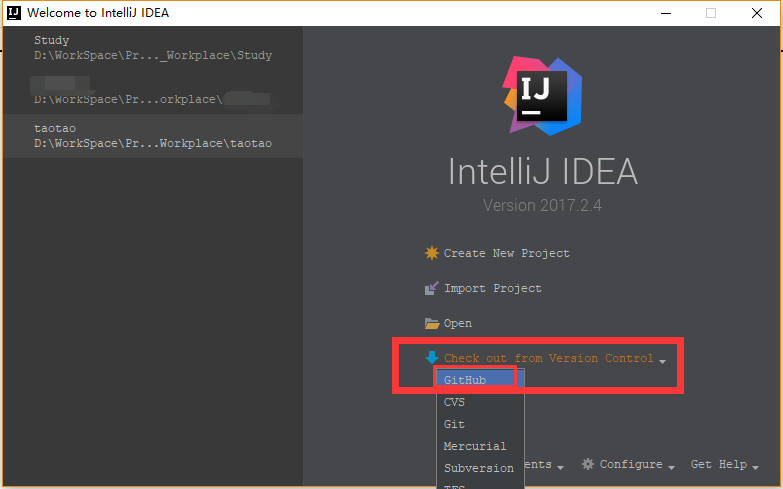
在gitHub上面创建带有.gitignore文件和README.md文件的仓库。并拷贝库的地址：

git@github.com:wxyzreps/Study.git

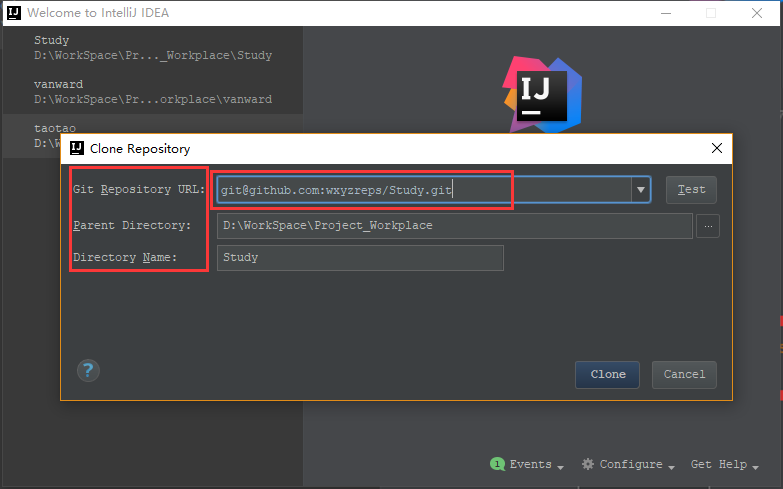


## 打开IDEA

打开IDEA，从VCS中检出

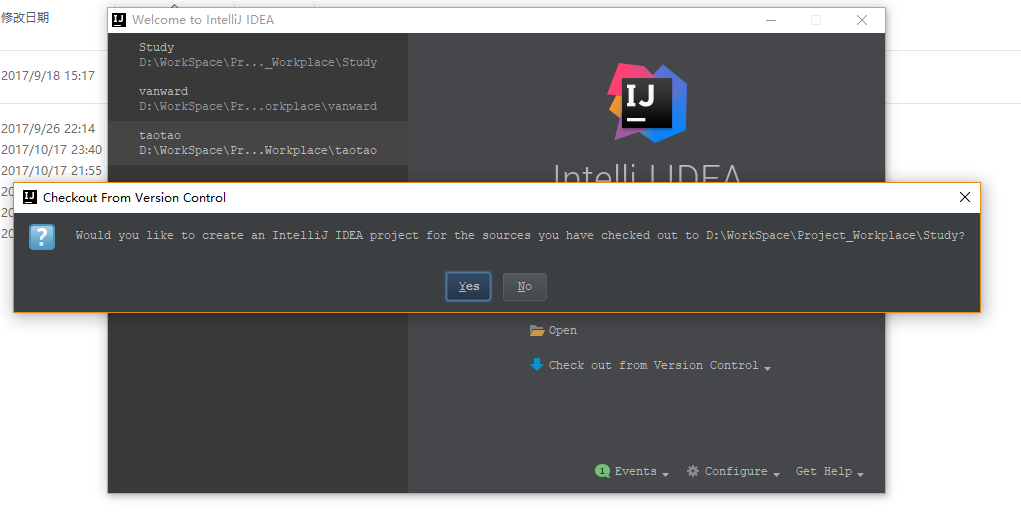


将第一步的地址粘贴到URL中，选定项目所在目录，点击克隆。

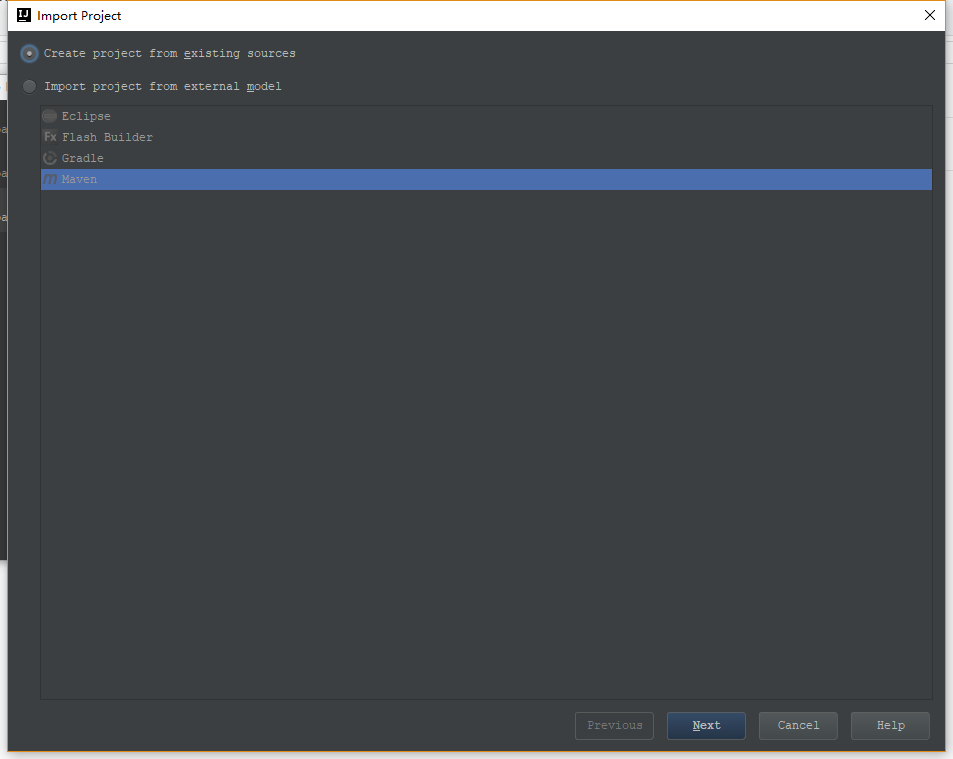


## 设置项目类型

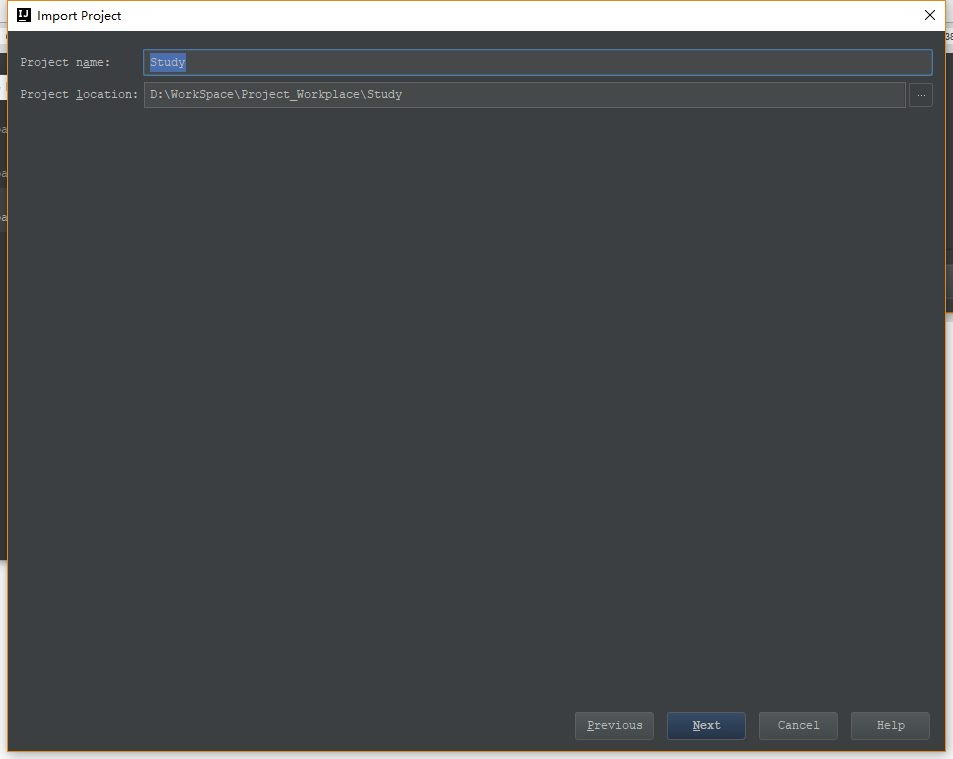
因为GitHub上面建立的仓库中只有两个文件：.gitignore文件和README.md文件，所以检出后，idea会询问是否需要建立一个IDEA项目。选是，



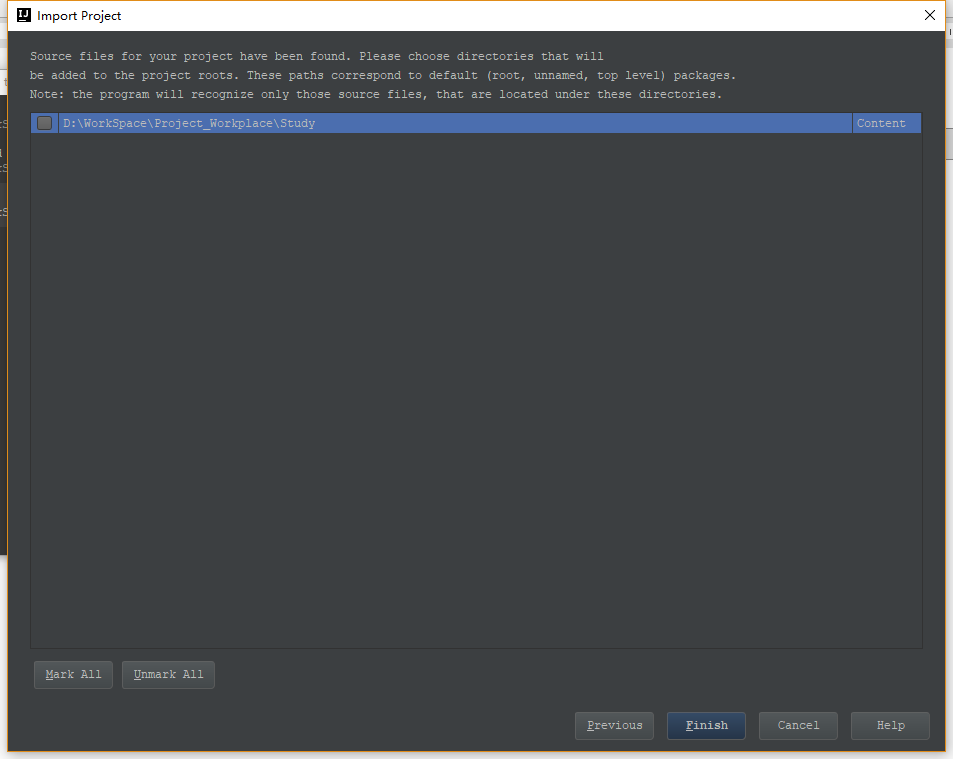
因为我这里是空项目，后期也不打算用Maven，所以选择第一项，然后下一步：



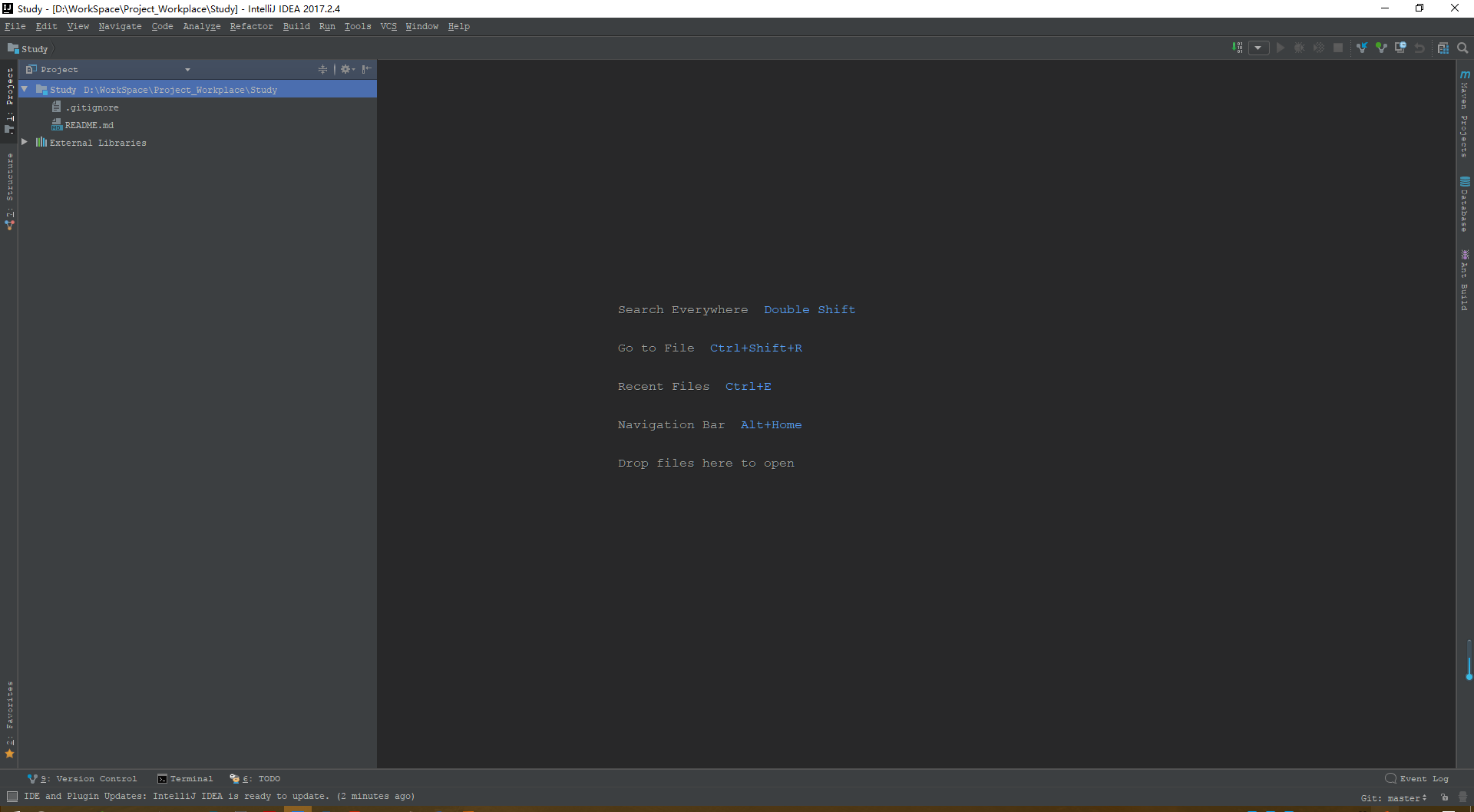
选定项目所在位置：



由于从Github上面检下来的没有src文件，所以不要勾选，后面自己再进行添加即可。点击finish完成。

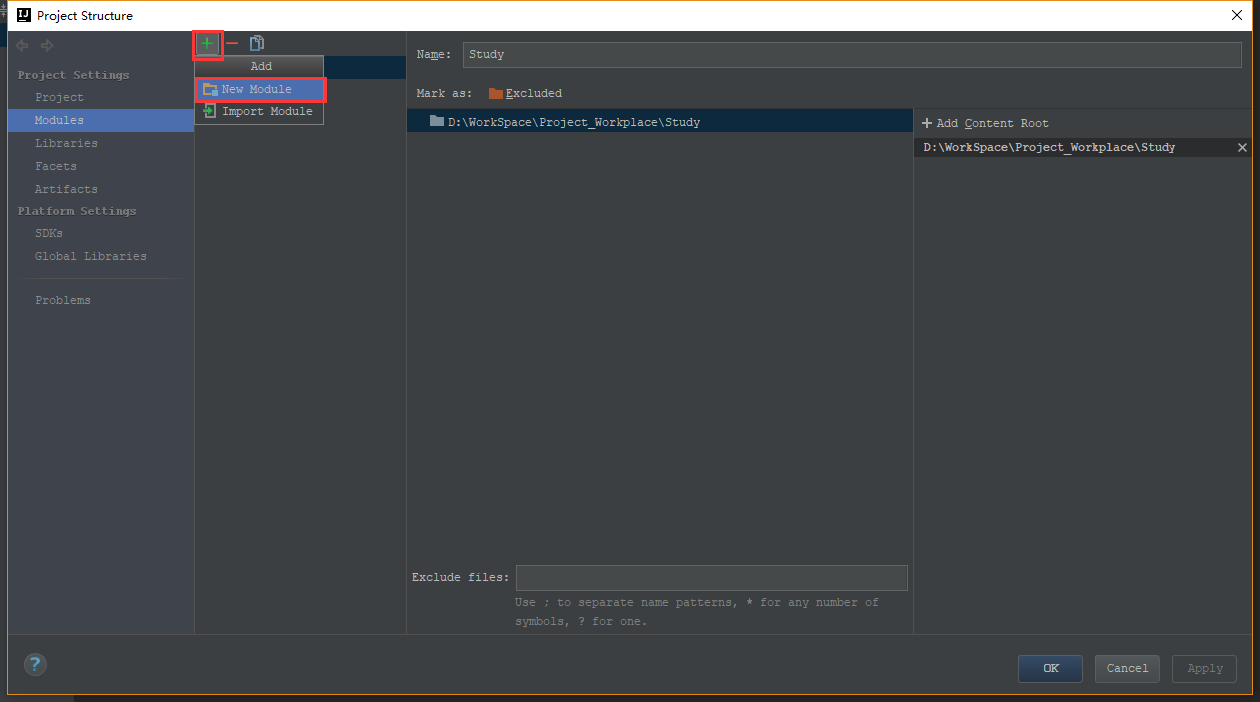


完成：

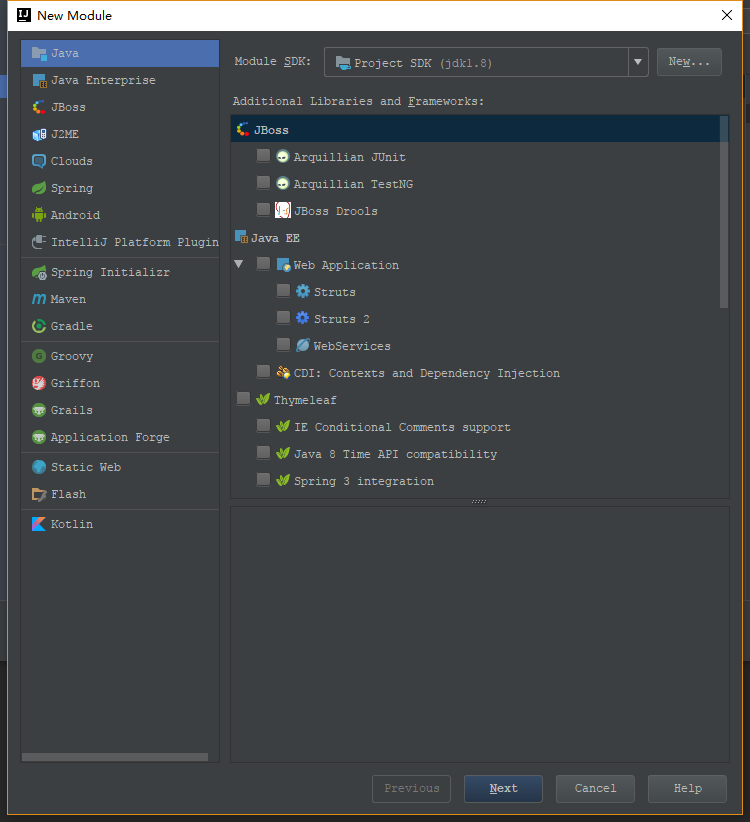


## 创建Model

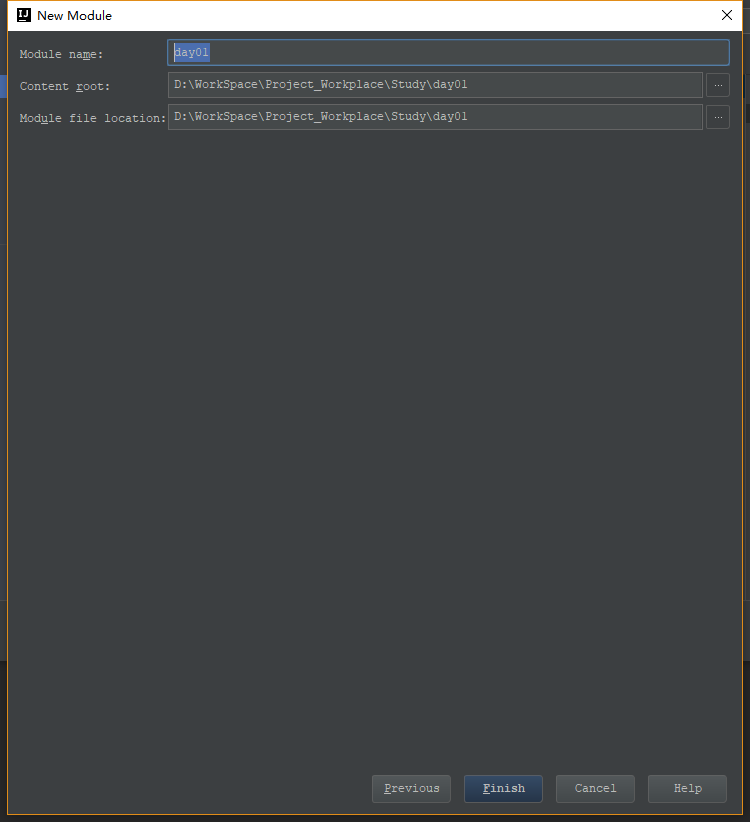
Ctrl+Alt+Shift+S快捷键打开ProjectStructure



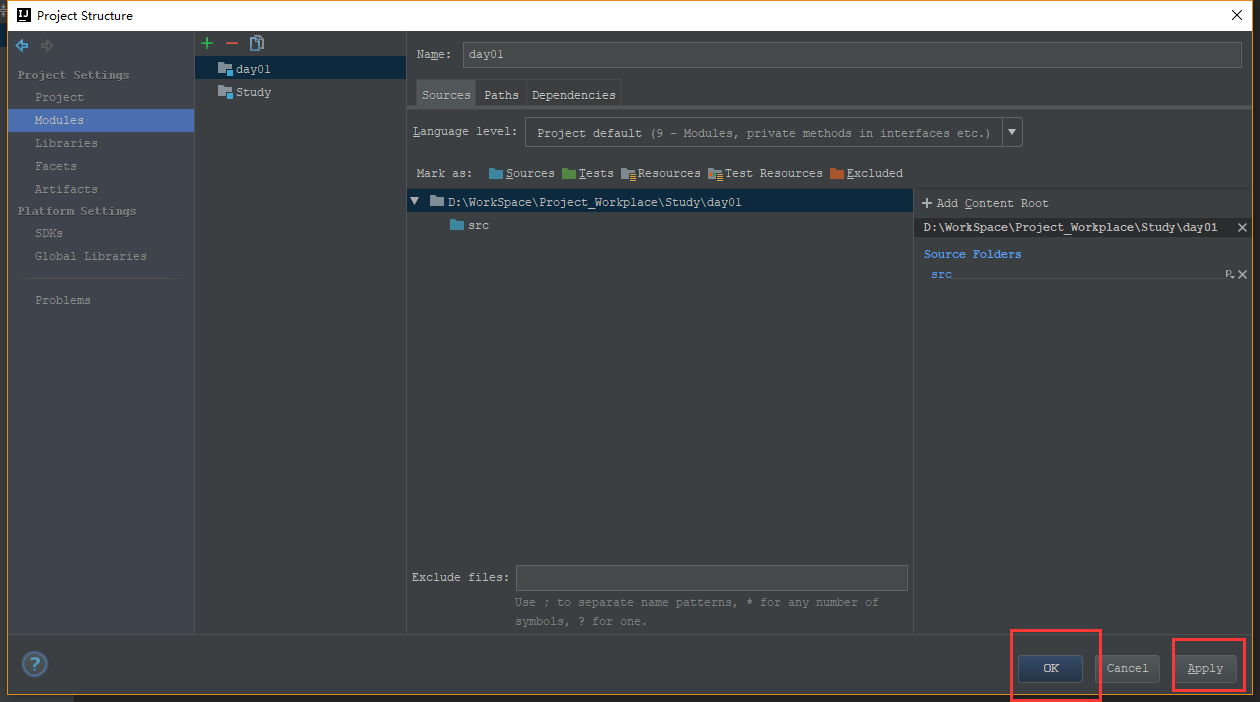
选择Module的类型：选择Java，然后下一步



选择文档所在位置，然后finish。

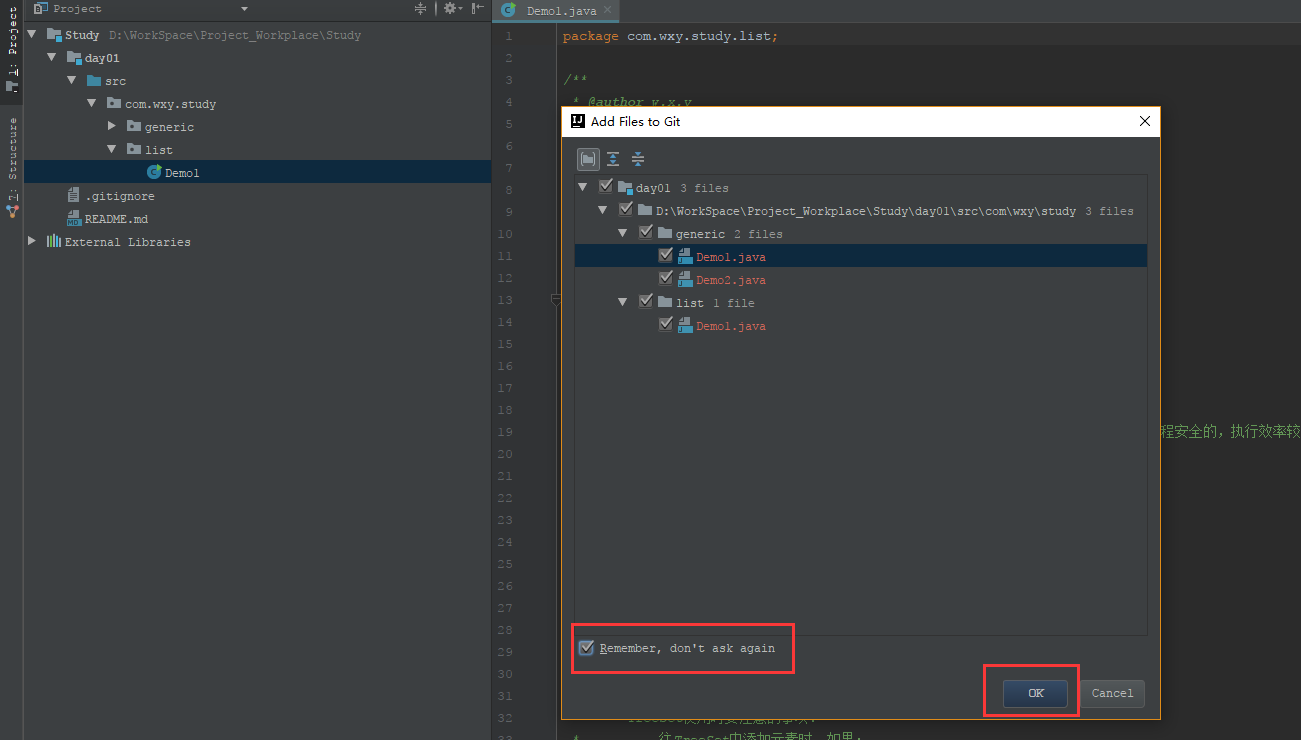


Apply，OK，完成：



## 编写代码

向项目中添加代码时，会自动跳出添加到GitHub的对话框。选择不再询问，点击OK



## 提交代码

提交操作就跟上面的一样了