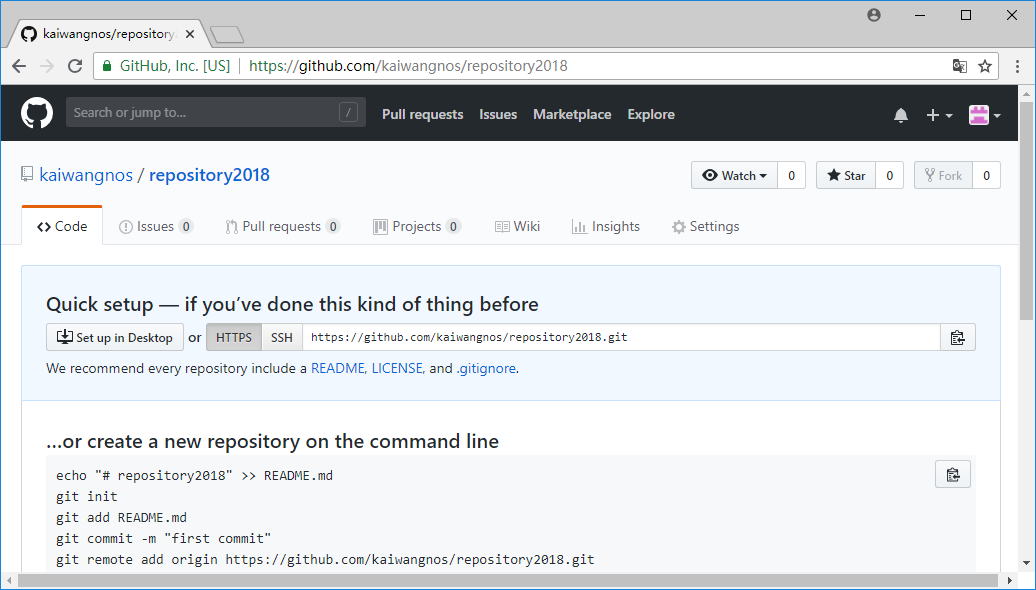
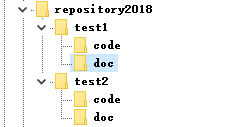
Github使用技巧：

有时我们不希望为每一个工程都创建一个一个远程仓库，或者说一个工程不仅仅只有代码，还有其他文档，资源文件等等，这些文件也需要协同开发。那么，怎么在一个远程仓库创建多个工程呢？

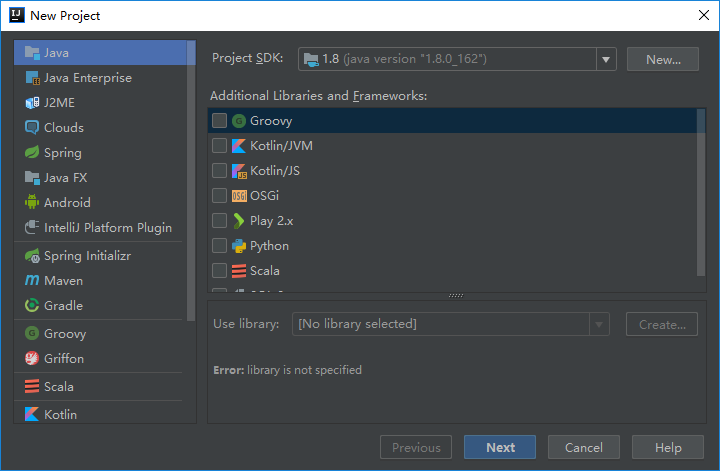
一、创建远程仓库

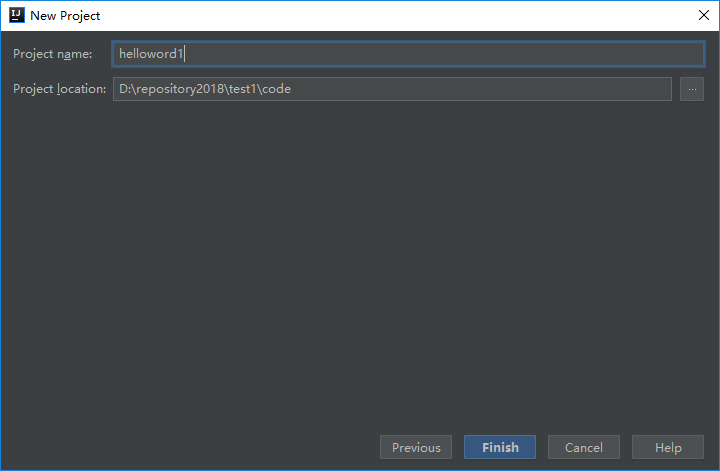


1. 在本地磁盘建立文件夹，作为本地仓库。同时建立目录结构。如下图所示：



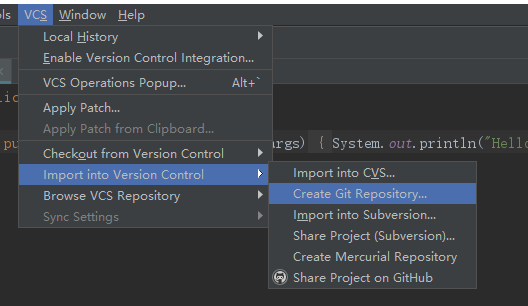
1. 用IDEA创建本地仓库和第一个工程，步骤二中的***D:\repository2018***目录作为本地仓库
2. 打开IDEA创建新工程



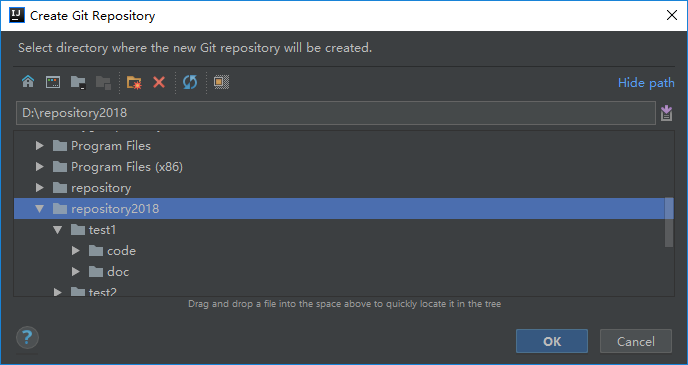


注意Project location的位置，***D:\repository2018\test1\code***

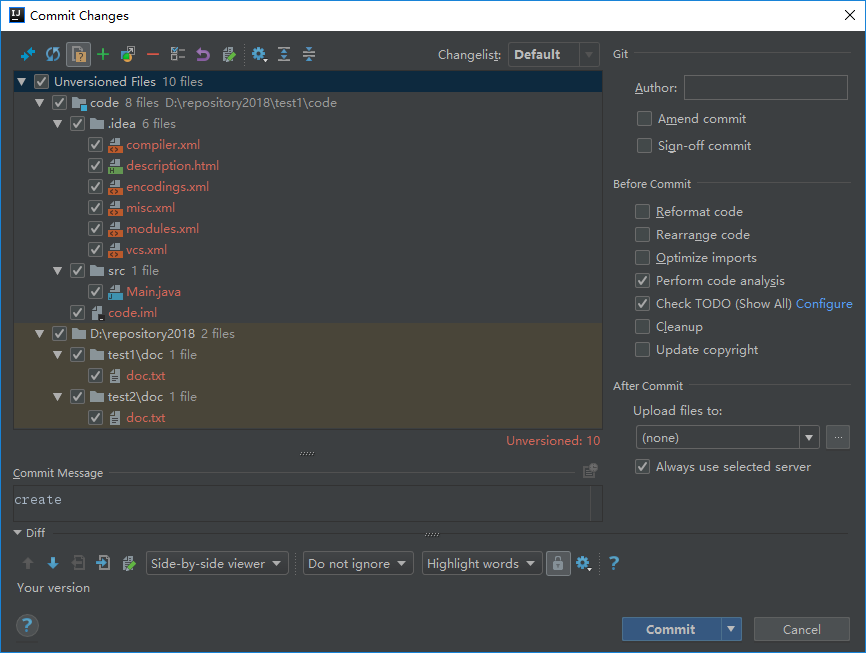
1. 依次点击VCS→Import into Verson Control→Create Git Repository



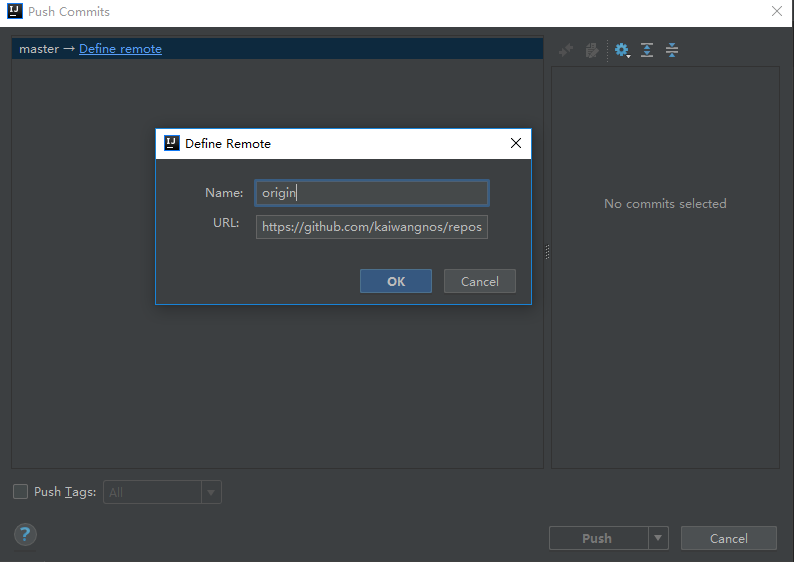
1. 在窗口中选择要创建本地仓库的目录（步骤三开头提到的***D:\repository2018***）



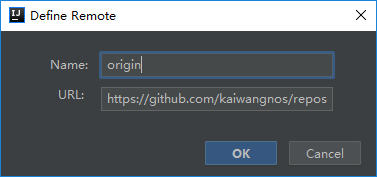
1. 将本地仓库push到远程仓库。
2. 依次点击VCS→Git→Commit Directory，勾选所有文件，填写Commit Message，点击右下角的Commit按钮，等待Commit成功。



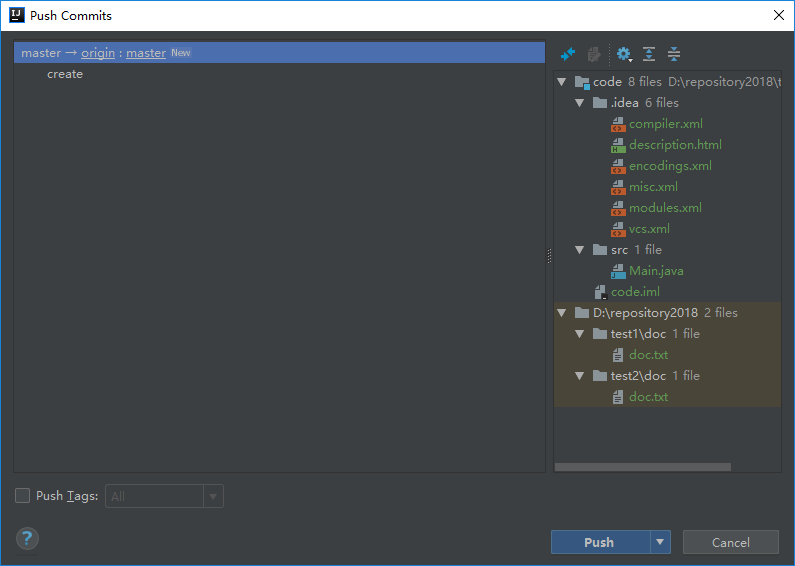
1. 依次点击VCS→Git→Push，点击Define remote。



1. 填写GitHub远程仓库URL，点击OK。



1. 点击Push按钮



Push成功后我们已经完成了本地仓库和远程仓库的关联，在test1目录创建了第一个工程，也包含了code和doc两个文件夹。在IDEA里只关联code下的代码。接下来尝试修改一个文件并push到远程仓库。

1. 创建第二个工程

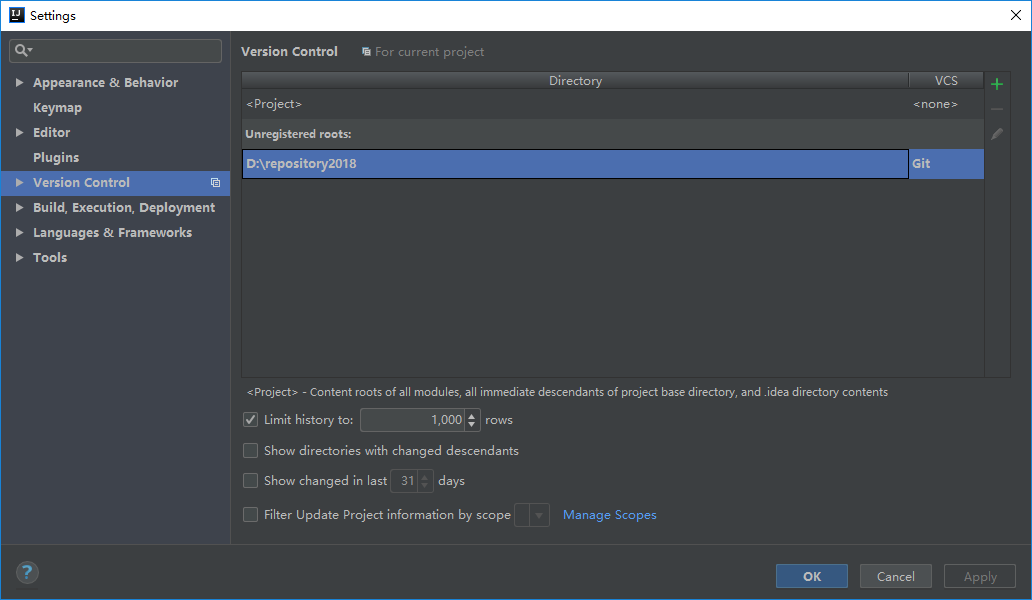
创建工程的方法与创建第一个工程相同，这里直接跳过创建过程。

***D:\repository2018\test2\code***

1. 将第二个工程PUSH到远程仓库。

1、打开第二个工程，依次点击File→Settings

选择Version Control

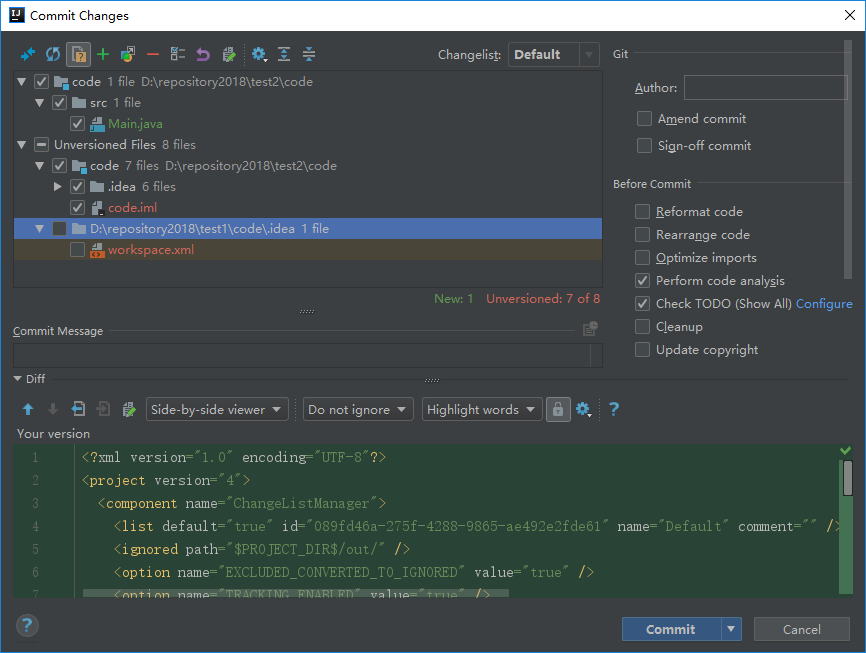


在右侧Directory点击选中步骤三中的本地仓库目录***D:\repository2018，***点击右侧边栏的绿色加号。点击OK，本地仓库配置完成。

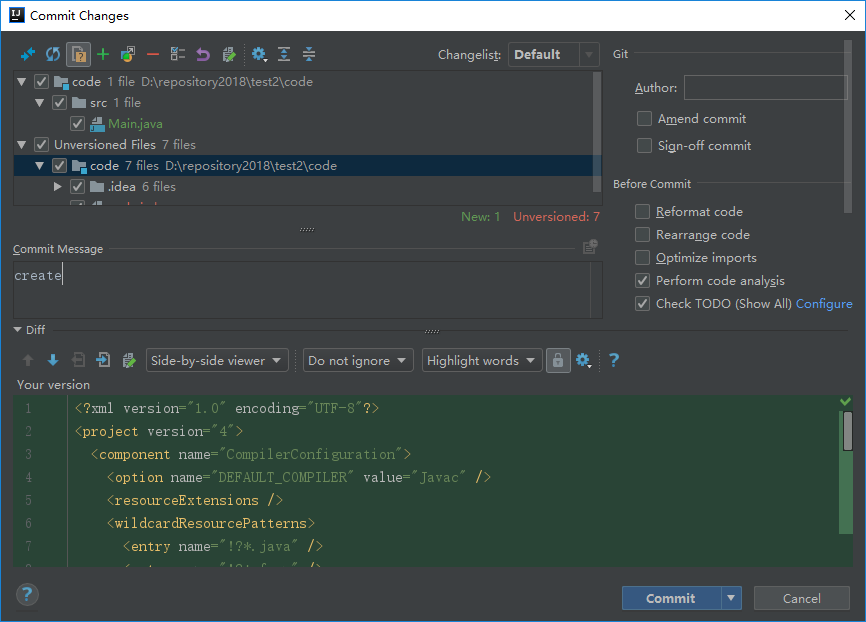
2、依次点击VCS→Git→Add再依次点击VCS→Git→Commit

移除其他无关的目录





点击Commit按钮



3、push即可。