**GitHub文件管理Ⅱ--VSCode**

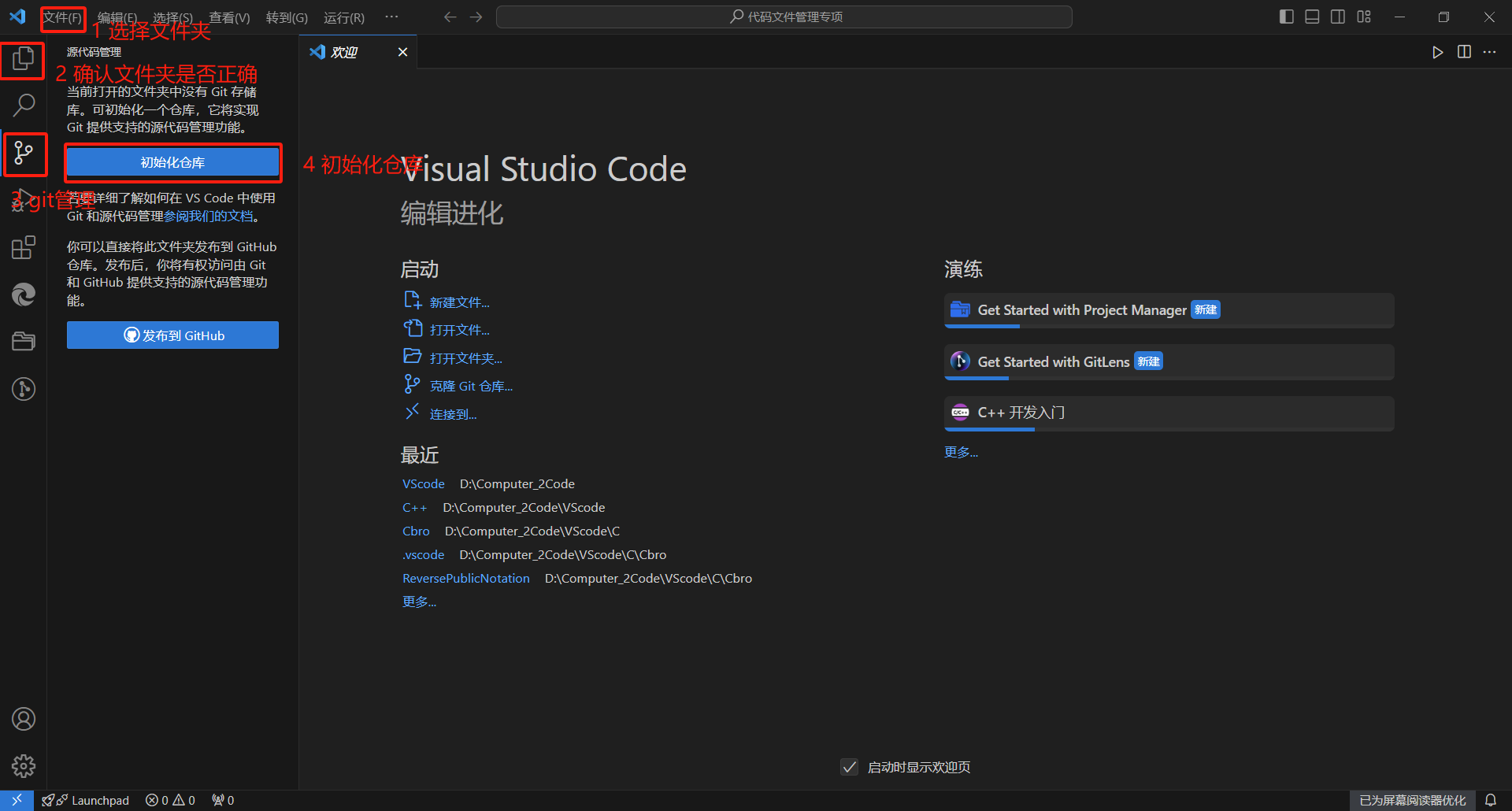
**王力萱**

**2024.7.12**

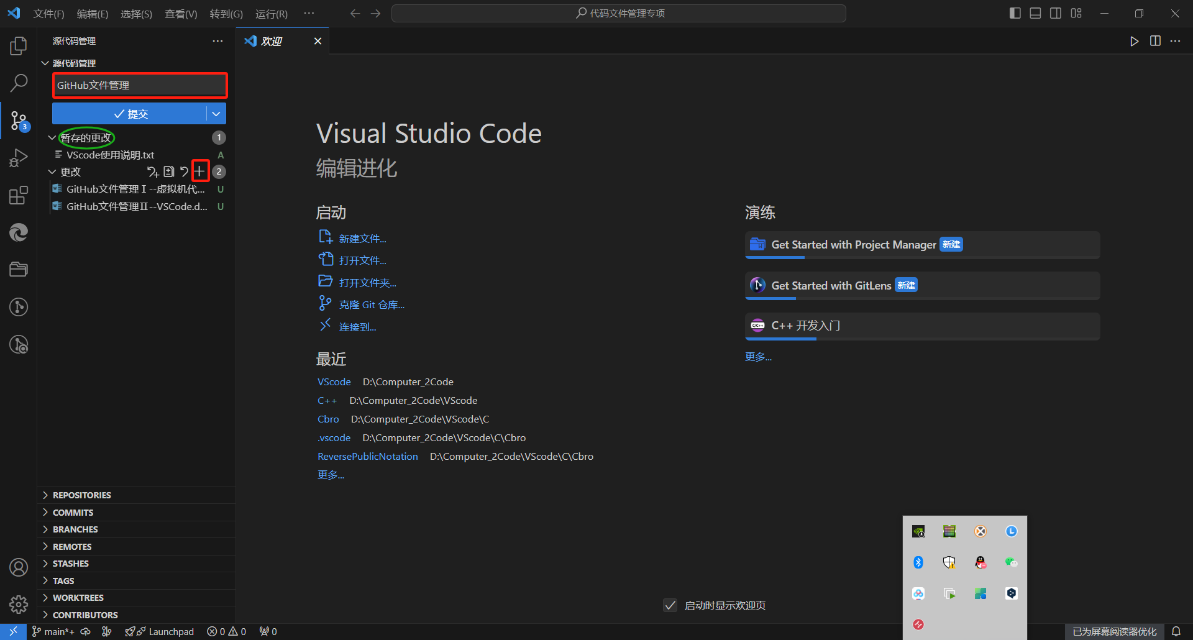
**在GitHub管理文件方面，Windows系统和Linux系统对VSCode的操作相同。以Windows的VSCode为例：**

1.打开待管理文件夹，初始化仓库，相当于git init。

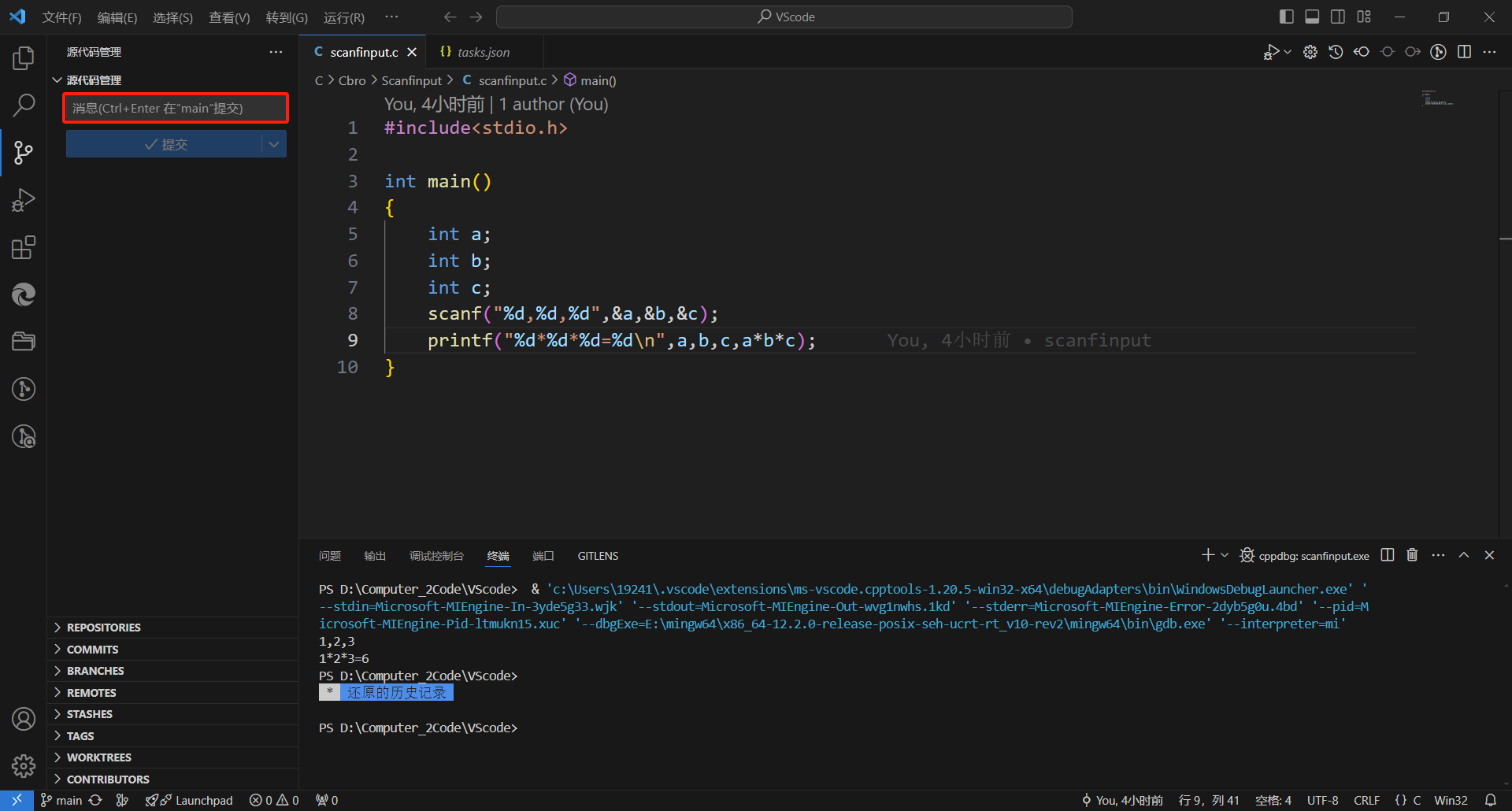
注意：Ⅰ中的代码操作需要先在GitHub网页上建立远端仓库，再git init，git clone，生成**本地工作区**(**远端仓库同名文件夹)**，待管理的文件/文件夹全部放入其中。

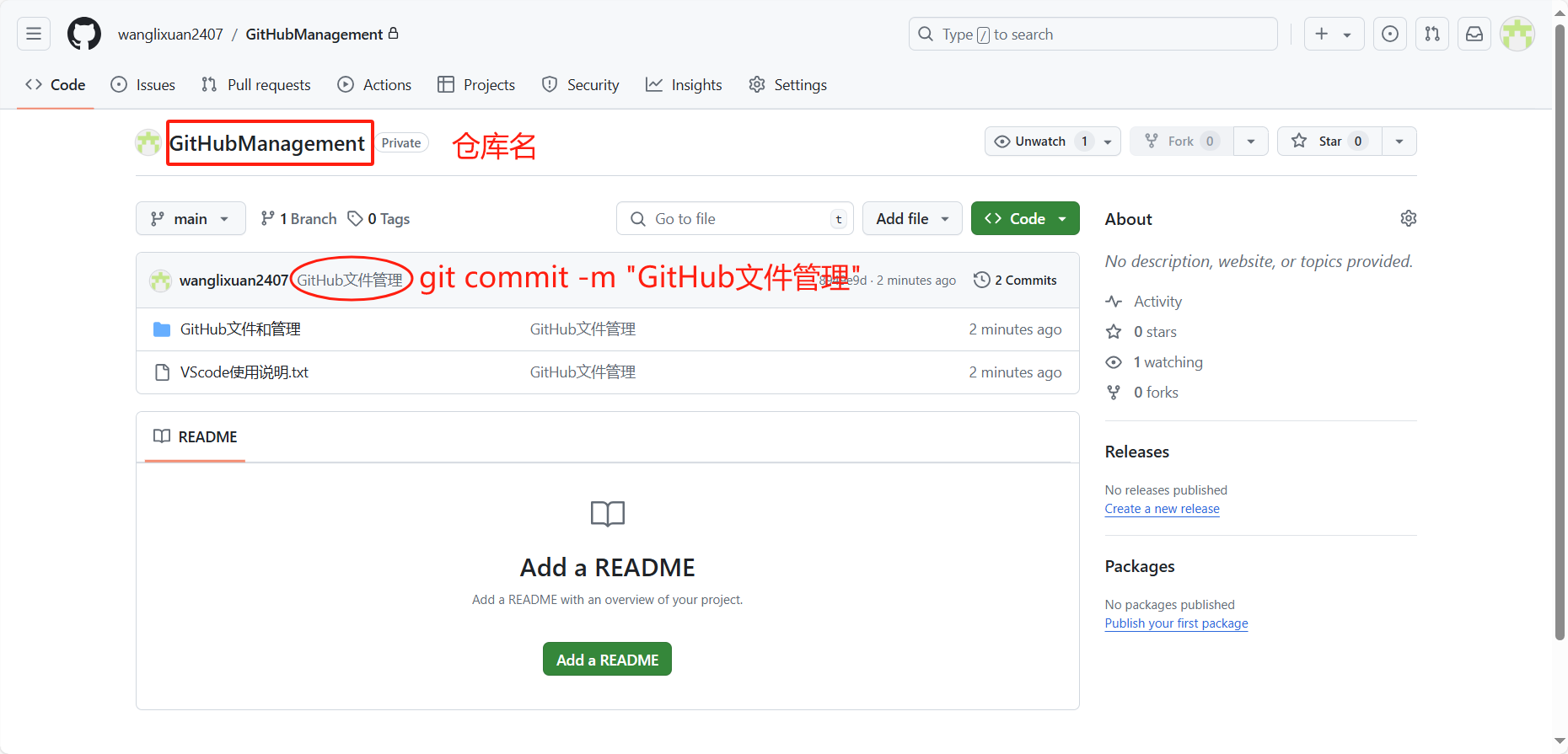


2.提交到本地仓库的是“暂存的更改”区域的文件，而非“更改”区域的文件。如果一次性提交，可以点“更改”右侧的加号再提交；如果要逐个注释(commit),则点每个文件右边的加号，上方红框处填上注释，逐个提交。

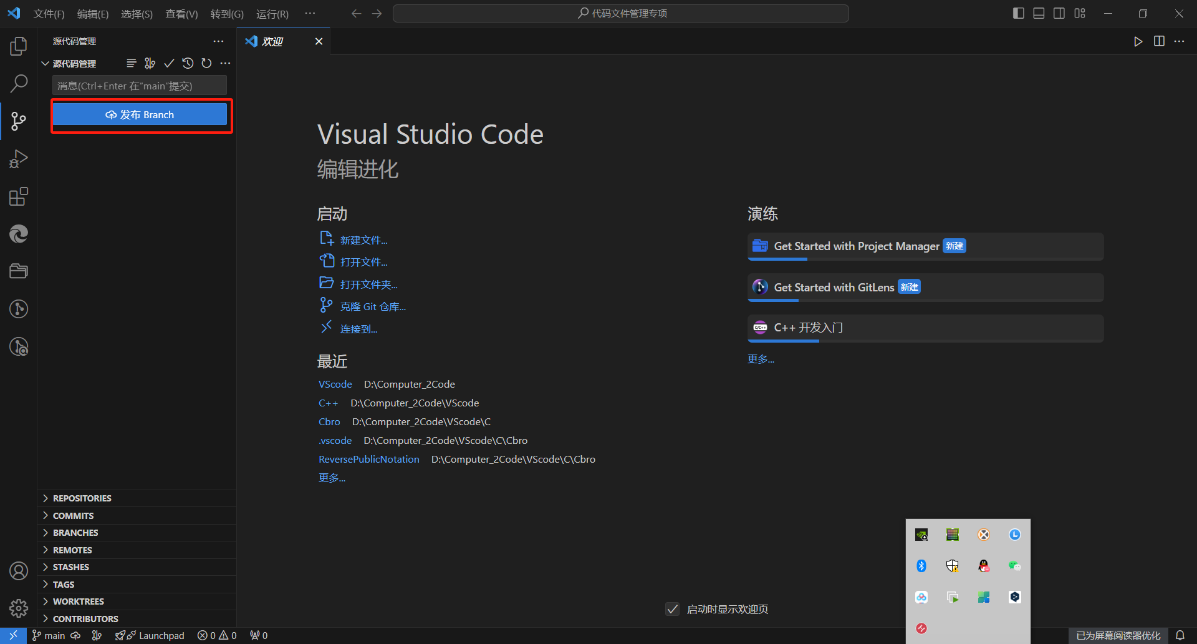
****

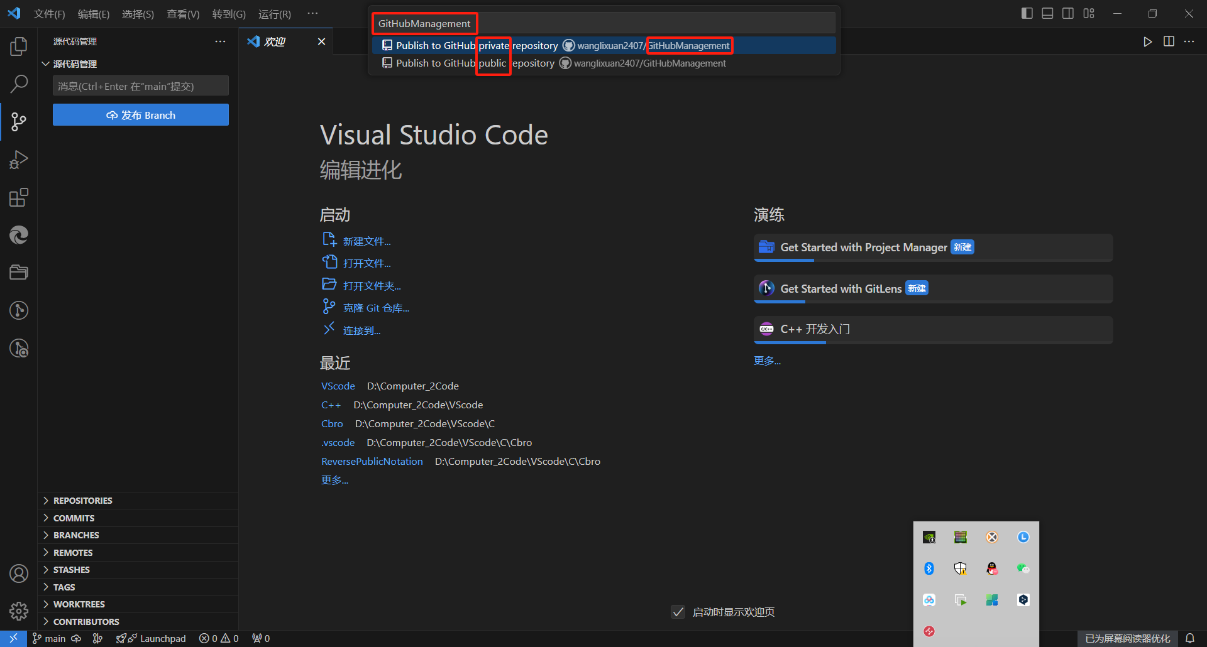
3.GitHub上提交时**必须**在红框处写信息，相当于Ubantu代码中git commit -m “A”中的“A”注释。最近一次提交，无论改了什么，GitHub网站相应仓库的红圈处也是“A”注释，此处以GitHub文件管理为例。

****

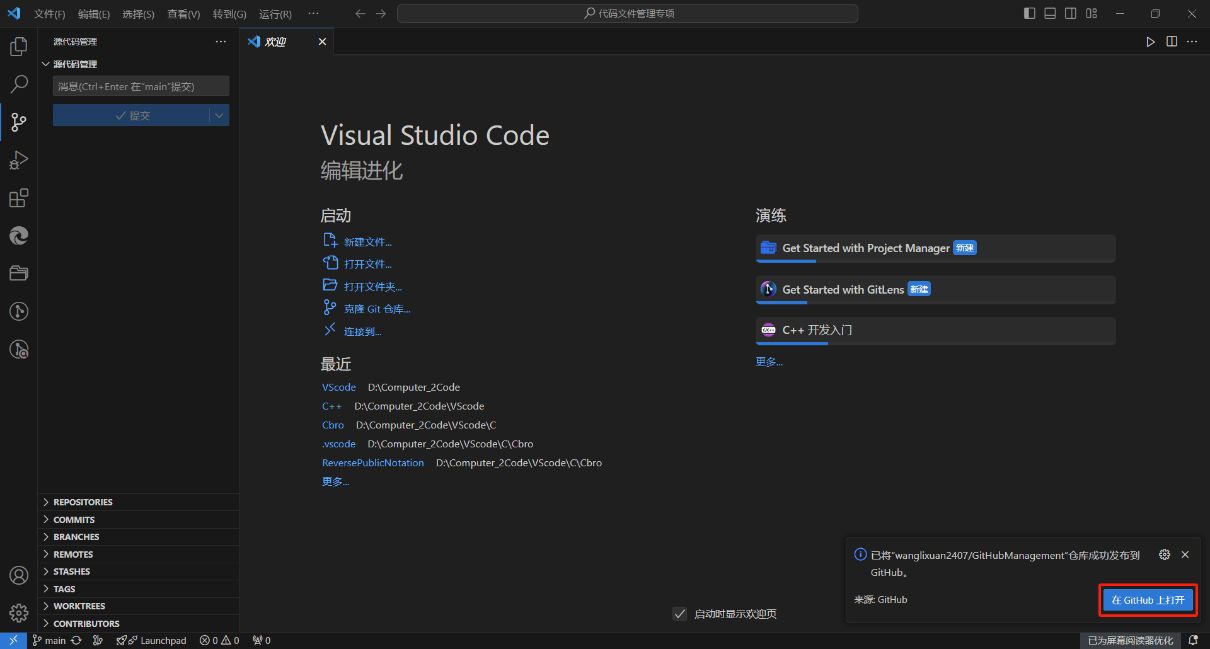


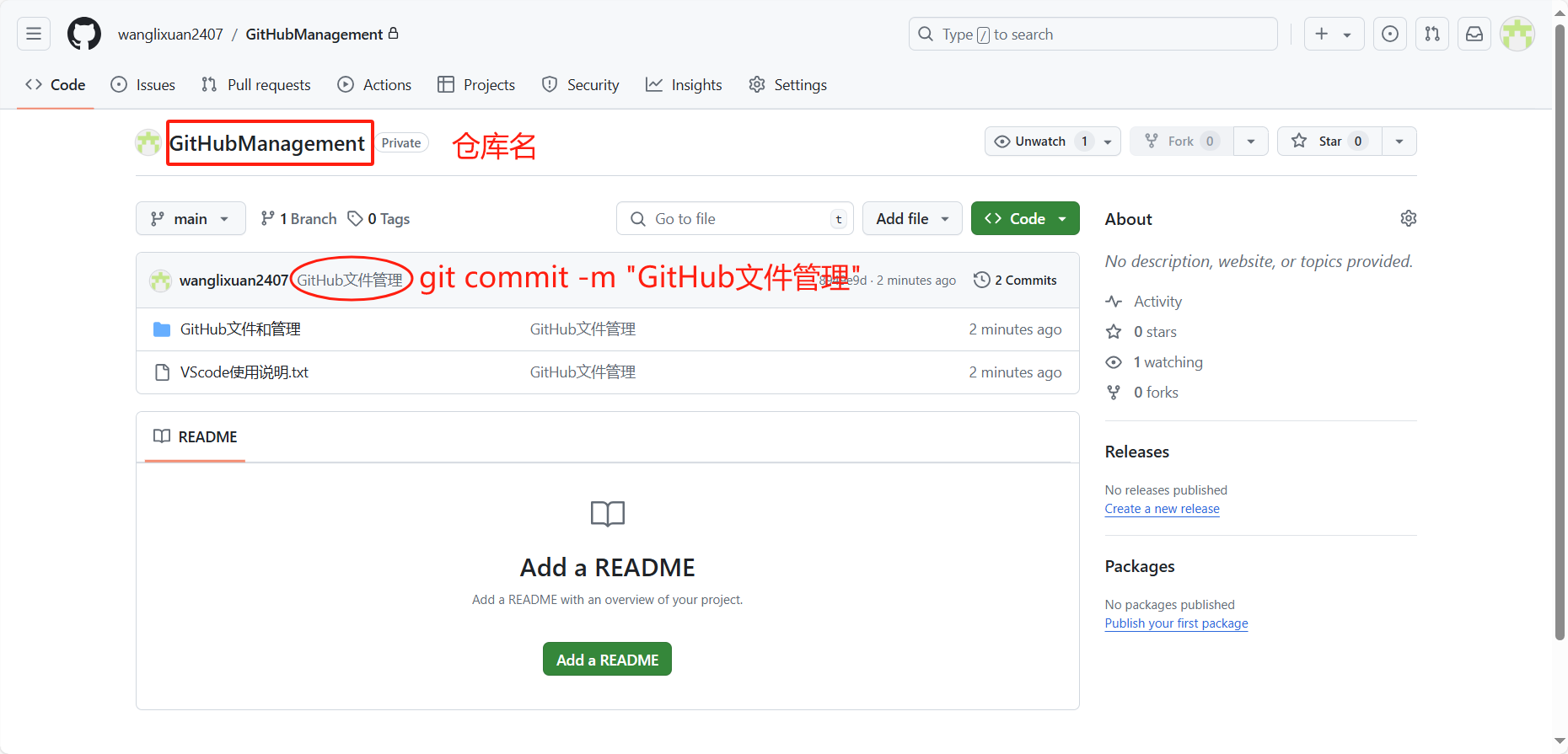
4.点击“发布branch”，输入远端仓库名(此处以GitHubManagement为例)，选择公开Public或私人可见Private。

****

****

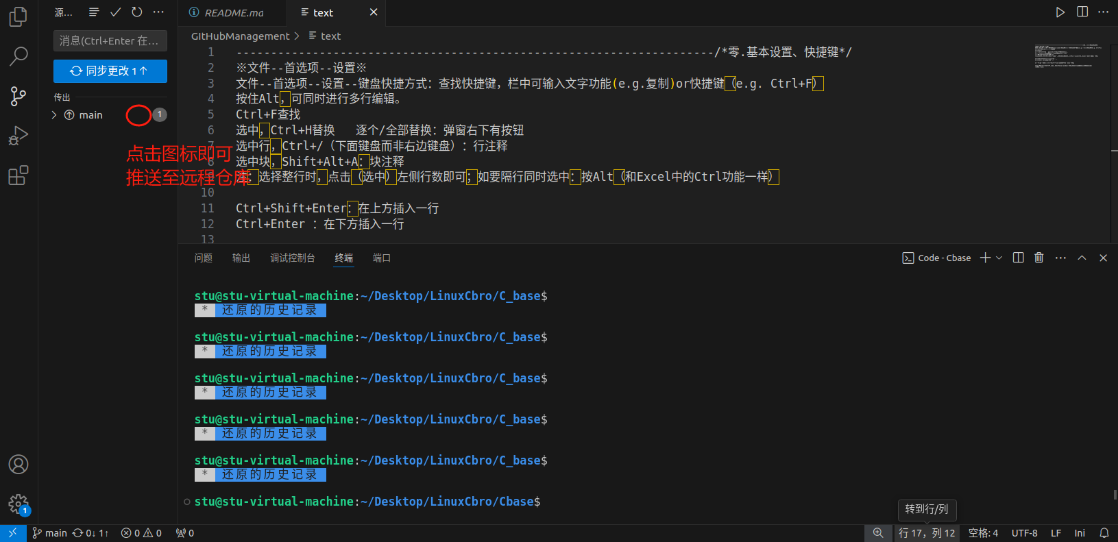
5.点击在GitHub打开，转到GitHub网页，看到远端仓库与文件。

****



6.建库完成，修改/添加/删除文件:更改/暂存的更改操作见2.

传到本地仓库之后，还需要传到远端仓库(等于Ⅰ中的git push)



代码在Windows系统编写完成后，复制到虚拟机上，传到GitHub备份仓库

**但是：由于虚拟机上编译其它.h文件较为麻烦，且有些函数执行会出错，目前已弃用虚拟机的代码编写、编译功能，只将其作为查看代码、上传GitHub备份的平台。**