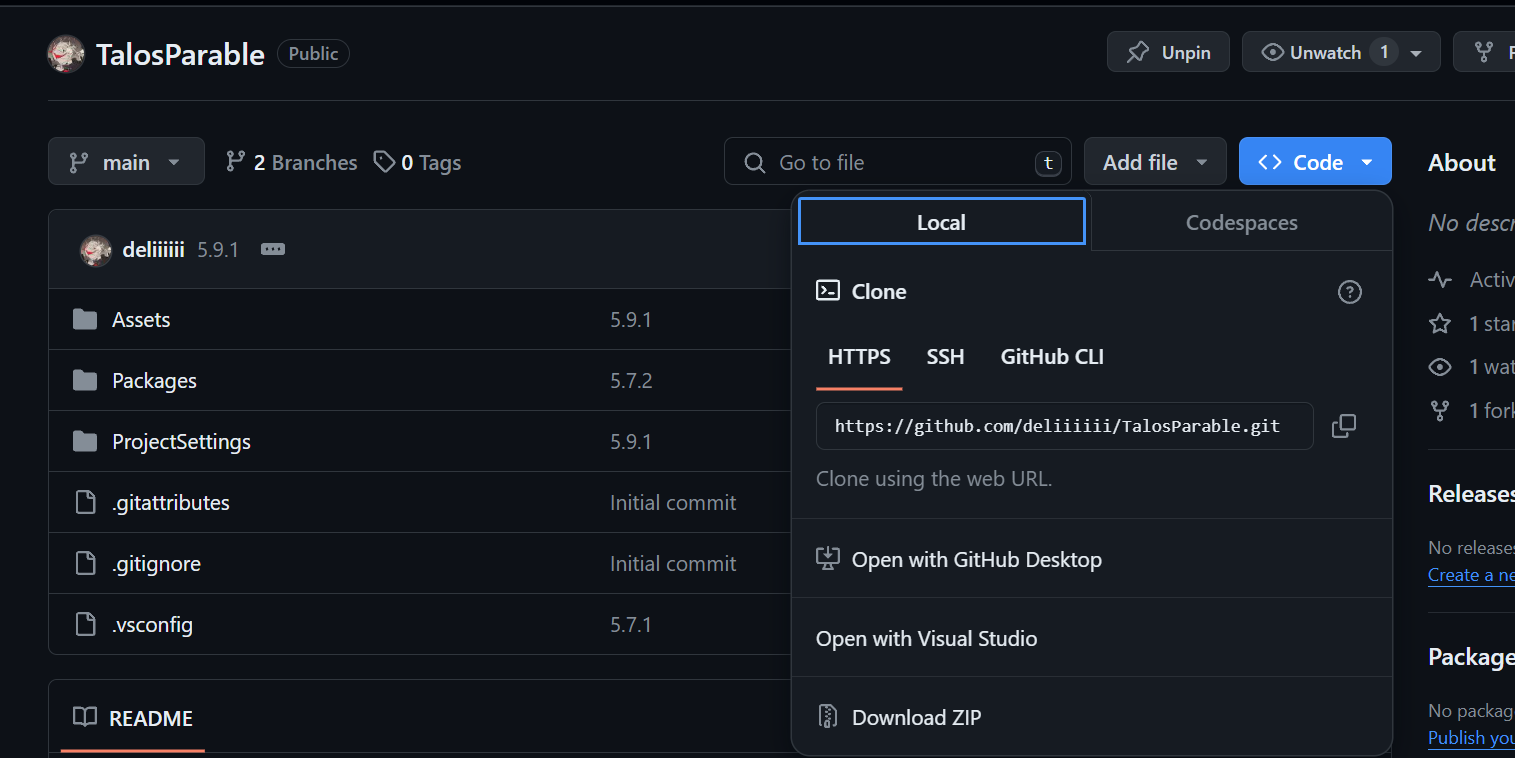
**GitHub仓库链接**

<https://github.com/deliiiiii/TalosParable>

**如何获取项目文件？**

**法一（不会用git的pull仓库）**：GitHub仓库的网页 - Code - Download ZIP - 解压

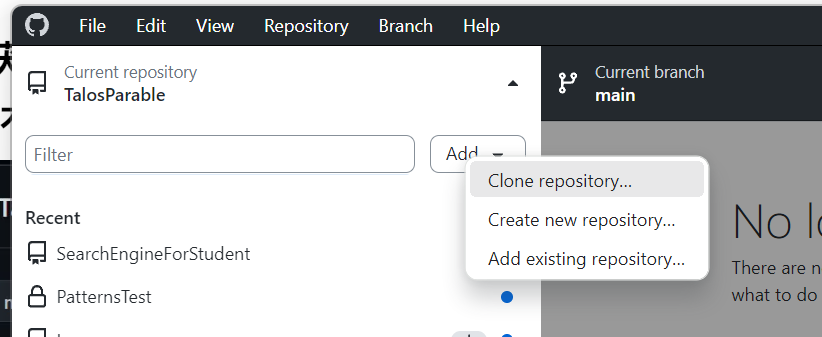
缺点：GitHub每次有更新时（会很频繁，且不会有什么提醒！）需要重新下载整个文件重新解压并覆盖原来的文件



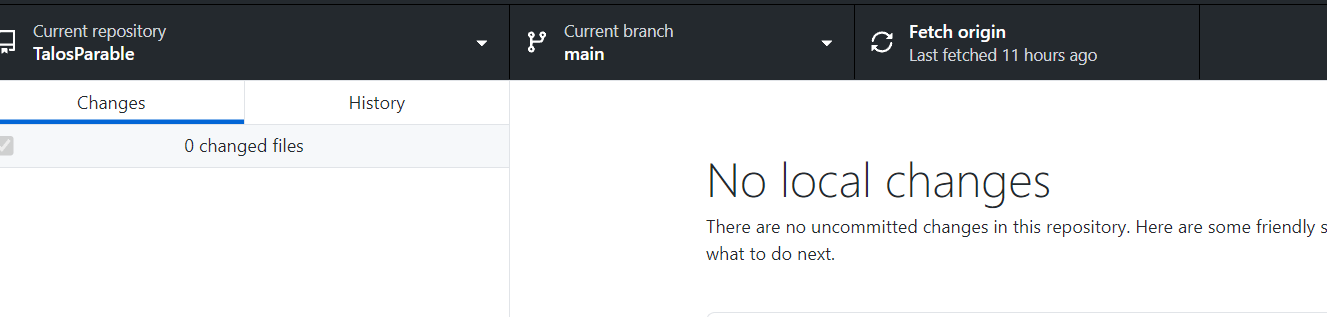
**法二（用命令行git或图形化界面GitHub Desktop）**：第一次需要克隆整个仓库到本地，之后每次打开项目前Fetch（同步）即可。个人用得多的是后者

第一次的克隆步骤：

下载GitHub Desktop - Current repository （不知道第一次用是不是显示这些词） - Add - Clone repository - URL - 输入仓库链接 - 选择Local path - Clone

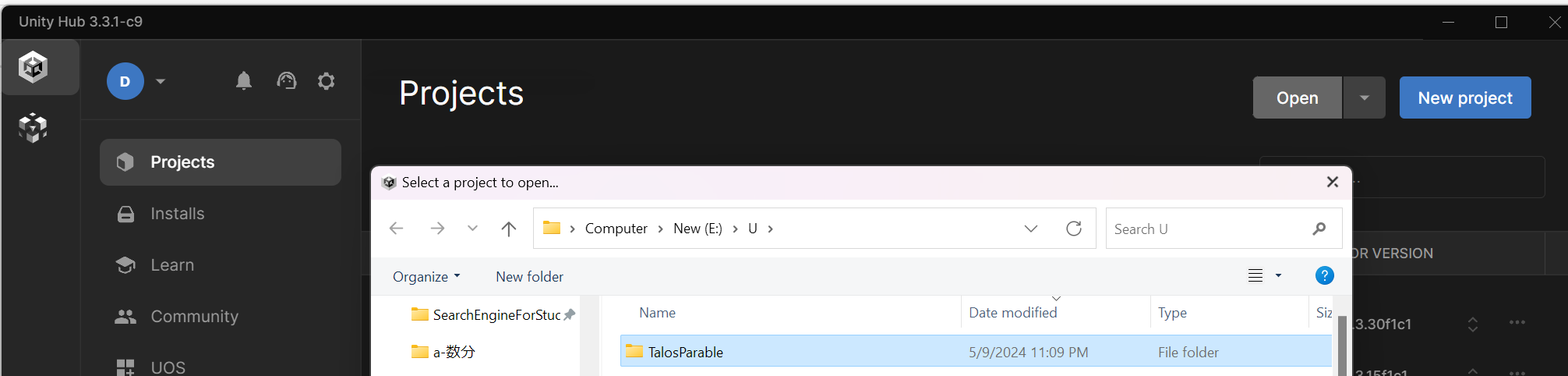


之后的Fetch（一般会出现Pull ↓）：



**如何用Unity Hub打开项目？**

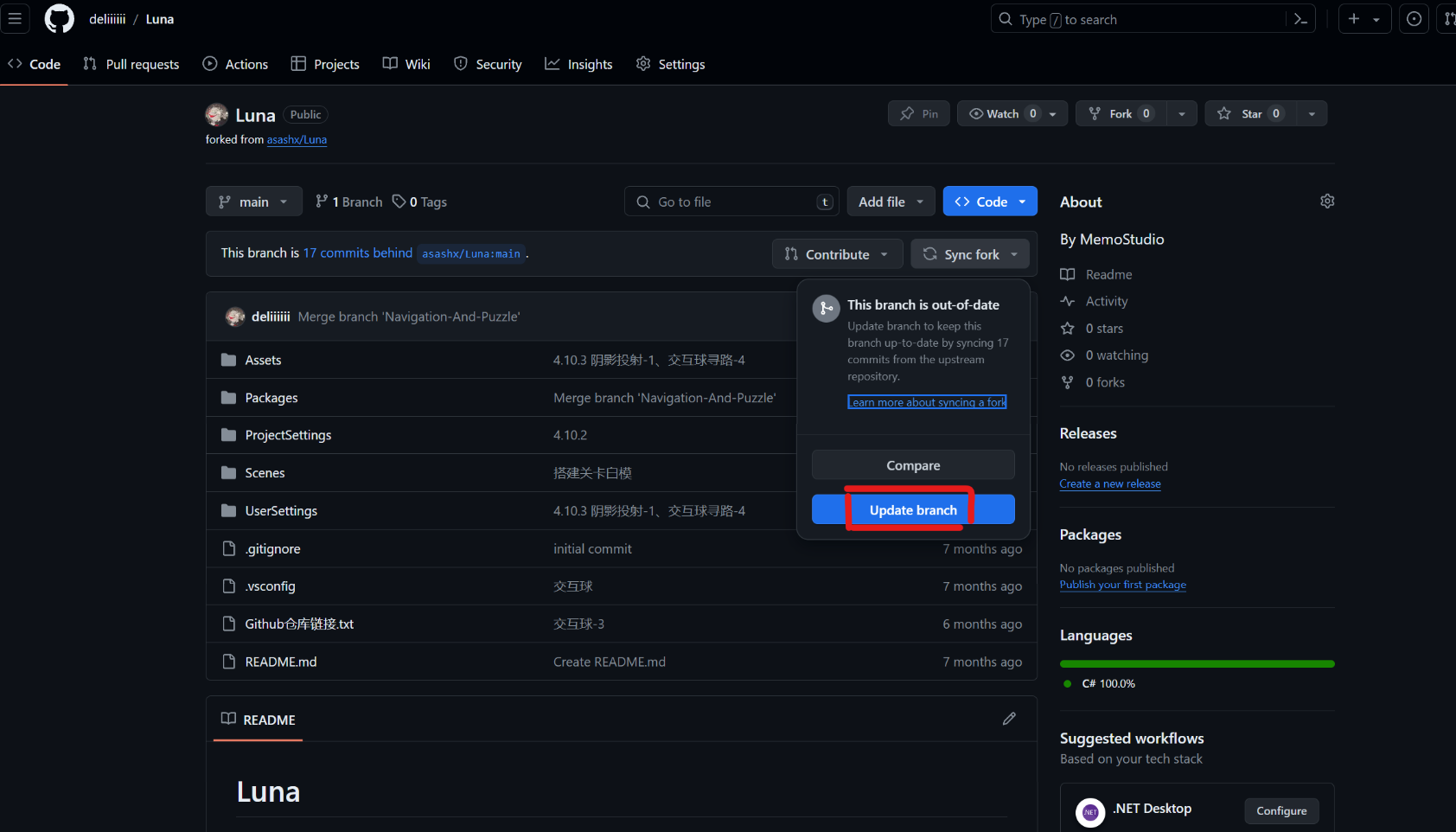
Open - 找到并单击选中TalosParable（不需要点进去）- 然后忘记要不要选择Unity Editor版本了，提前去官网下载2020.3.30f1c1就是（c1代表china，没有这两个字大概也没事）



**如何同步项目进度？**

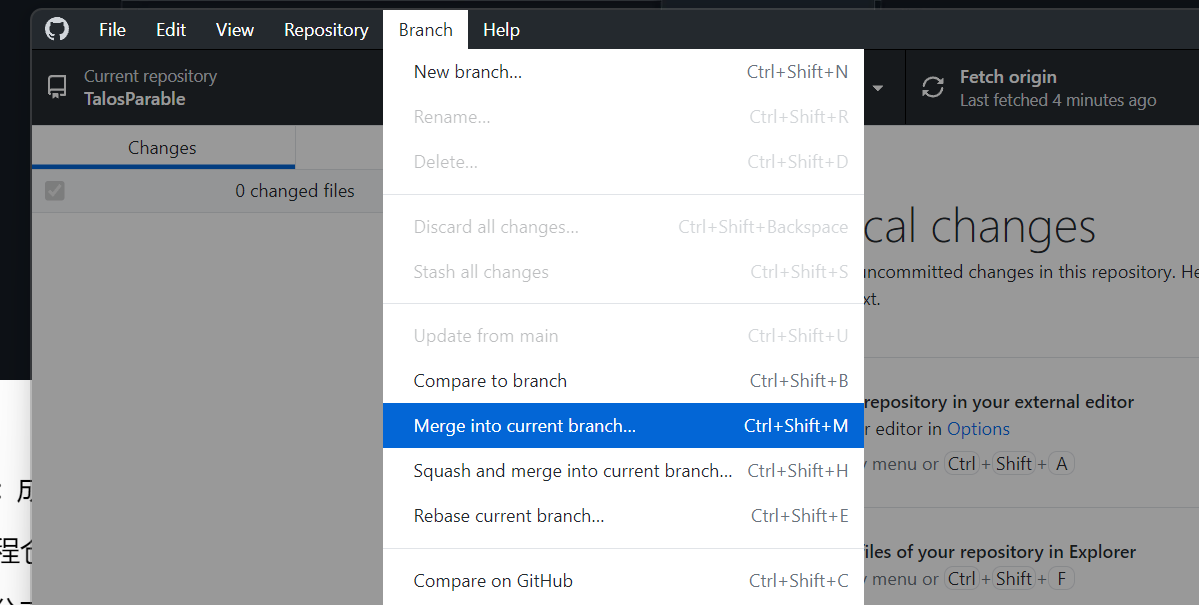
**法一：在GitHub上fork后，成为另一个仓库（与主程仓库不是同一个，对它只能clone不能直接push）**：有修改后以交pull request的方式把你的仓库的主分支合并到主程仓库的主分支，要等主程审核，你也要经常在GitHub上update主程仓库的进度，把进度同步你自己的仓库（没有必要在你自己的仓库建其他分支）。

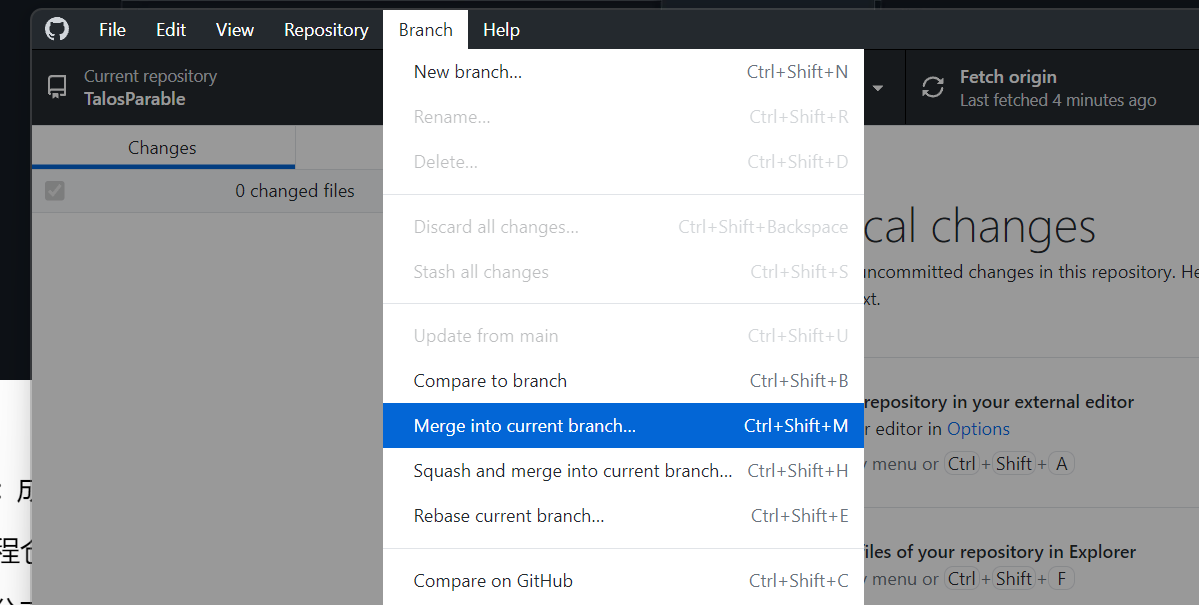
示例：GitHub上update主程仓库（fork的主程仓库是asashx/Luna，我fork的仓库落后了17次提交）：



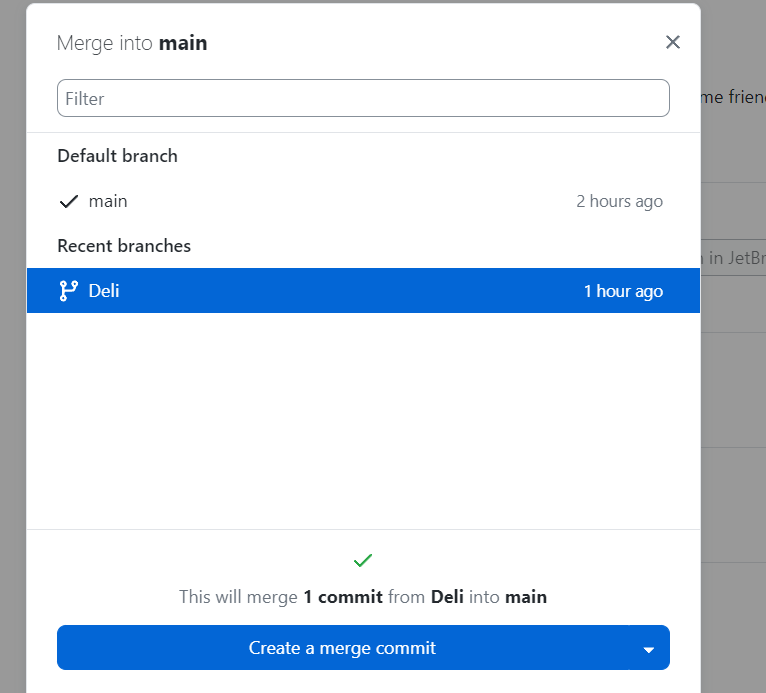
**法二：成为主程仓库的合作者**：把GitHub昵称发给主程后，等待一份邀请邮件，点同意加入。然后一定要在主程仓库创建子分支，有修改后在Github Desktop上把这个子分支合并到主分支，反过来（主分支到子分支）合并一次就可以同步其他进度，不需要等主程审核

示例：用Github Desktop把子分支Deli合并到主分支main

切换到main分支，Branch - Merge into current branch...



点击子分支Deli，一般要解决冲突(Conflicts) - Create a merge commit



**法三：石墨文档**

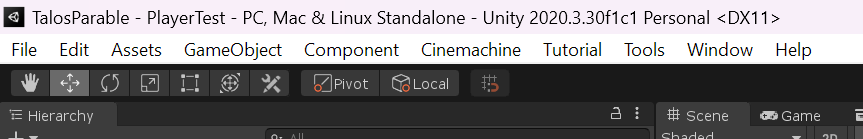
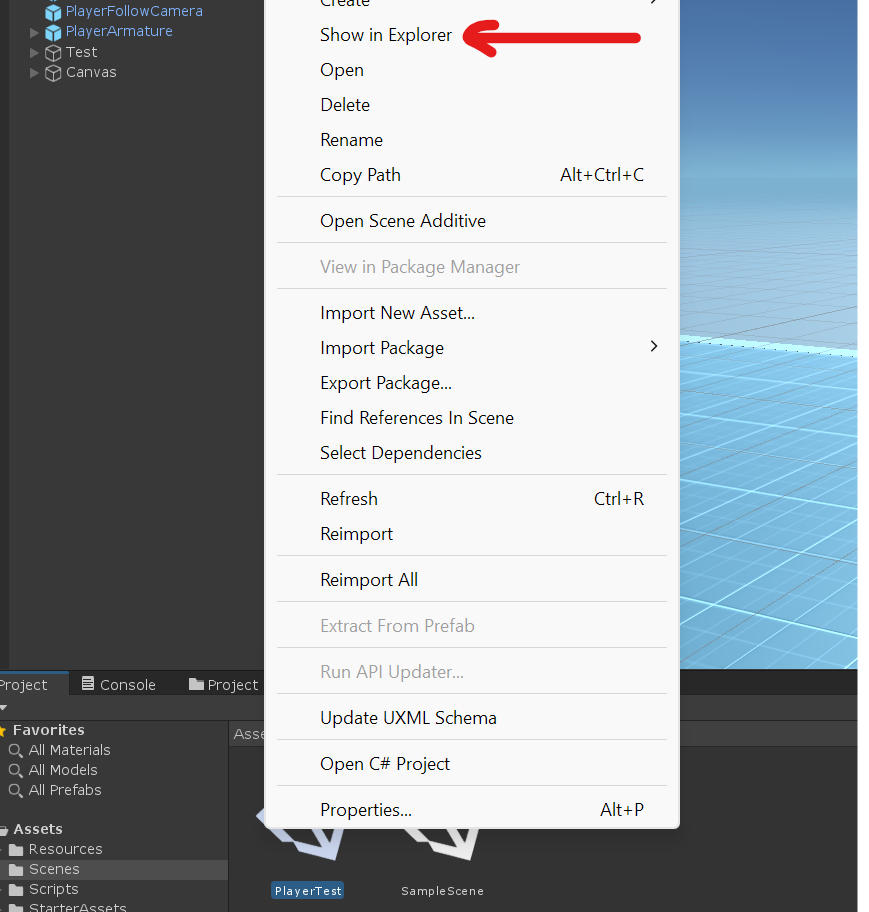
批量上传和批量下载很麻烦，不如GitHub直接在本地文件夹读写，但方便看修改时间和修改人、改某个需求表格

法四：群文件（每次小改动都要重发）

法五：私信发（不一定回）

**能动的机器人主场景(打开Unity后，找到并双击即可)：**TalosParable/Assets/Scenes/PlayerTest.unity

随后在文件夹中(打开所在文件夹的快捷方法：Show In Explorer)复制一个副本，双击进入那个场景进行你的修改（Unity窗口标题会显示项目名 - 场景名 - 平台 - Unity版本）



一些没用的闲扯（会用Github Desktop且是合作者，与主程操作同一个仓库）：

养成经常点Fetch（同步，有时候会显示Pull和Push证明有落后和超前的进度了）的好习惯（bushi）；

点Current Branch - Add创建分支时是在当前分支的最新commit生成一个副本，在左侧的commit上右键也可以创建分支，新分支push前Fetch那里会变成Publish branch；

切换分支前最好commit一次，最好不要带着修改去其他分支；

切换分支真的很自动！本地文件会自动修改（甚至不用关闭VS和Unity）；

向主分支合并有冲突时迅速私戳那次commit的人。