

Git & GitHub 入门作业讲义（中文版）

本讲义适合第一次接触 Git / GitHub 的同学。请一边在电脑上实际操作，一边回答问题。

所有问题都与这个示例仓库有关：

https://github.com/DaWenTi-sleep/git_learning_GitHub.git

一、Git 工作流程示意图（请在脑中想象这一张图）

工作区（你正在编辑的文件）

↓ git add

暂存区（准备好要提交的改动）

↓ git commit

本地版本库（记录在你电脑里的历史版本）

↓ git push

远程仓库 GitHub（云端的版本库，用来备份和协作）

二、基础概念

Q1：用自己的话回答下面四个问题（不要抄网上的定义）：

1) Git 是什么？

2) GitHub 是什么？

3) Git 和 GitHub 的关系是什么？

4) 你的本地电脑在这里面扮演什么角色？

三、克隆示例仓库（clone）

Q2：在命令行里执行：

https://github.com/DaWenTi-sleep/git_learning_GitHub.git

然后回答：

1) 克隆完成后，你的电脑上多了哪些东西？

2) 这个项目目录里面的 .git 文件夹是做什么用的？

四、提交一次改动（add + commit）

Q3：在仓库目录里，新建一个文件 helo.txt，写入任意一句英文或中文。

然后依次执行：

git add .

git commit -m "Add helo.txt"

请回答：

1) git add 做了什么？

2) git commit 做了什么？

3) 为什么不能只 commit 而不 add？

五、推送到 GitHub（push）

Q4：在命令行中执行：

git push origin main

然后用浏览器打开 GitHub 上对应的仓库页面，刷新。

请回答：

1) 推送之前和推送之后，GitHub 上有什么变化？

2) 你是怎么确认新的提交（commit）已经出现在 GitHub 上的？

六、创建分支并在分支上开发

Q5：在命令行中执行：

git checkout -b feature/add-json

然后新建一个文件 sample.json，内容示例：

```
{"name": "git learning", "author": "me"}
```

保存后执行：

```
git add .  
git commit -m "Add sample.json"  
git push -u origin feature/add-json
```

请回答：

- 1) 为什么我们要用分支（branch），而不是直接在 main 上改？
- 2) 分支能帮我们解决什么样的实际问题？（举一个你能理解的例子）

七、在 GitHub 上创建 Pull Request 并合并

Q6：打开 GitHub，对这个仓库：

- 1) 从 feature/add-json 分支向 main 分支发起一个 Pull Request
- 2) 自己把这个 PR 合并（merge）到 main。

然后回答：

- 1) “合并（merge）”这一步实际上做了什么？
 - 2) 合并之后，main 分支的代码和历史发生了什么变化？
- Request（简称 PR）。

八、制造一次简单的冲突并解决

Q7：请按以下步骤尝试制造一次冲突：

- 1) 在 main 分支上，把 helo.txt 内容改成：Hello from main 提交并推送。
- 2) 切换回 feature/add-json（或新建一个分支），把 helo.txt 内容改成：Hello from feature 提交并推送。
- 3) 尝试从这个分支合并到 main。

如果出现冲突，请观察 Git 给出的提示，然后手动修改 helo.txt，保留你认为正确的内容，最后重新提交并完成合并。

请回答：

- 1) 为什么会产生冲突？用自己的话解释。
- 2) 什么情况下，即使很多人同时改代码，也不会产生冲突？

九、pull 和 fetch 的区别

Q8：查资料或自己理解后，回答：

- 1) git fetch 做了什么？
- 2) git pull 做了什么？
- 3) 为什么通常认为 pull 比 fetch “更危险”一点？请结合“自动合并”来解释。

十、作业提交要求

请整理下面这些内容，一起提交：

- 1) 所有问题的文字回答（可以是一个 Markdown 文件，或一个 PDF / Word）。
- 2) 关键步骤的截图（例如：命令行运行结果、GitHub 上的 PR 页面截图等）。
- 3) 你的 GitHub 仓库链接，确保老师能在 GitHub 上看到：

- 你的 helo.txt 和 sample.json 文件；
- 你的分支和合并记录；
- 你的 Pull Request 记录。

完成这份作业后，你应该能说清楚：

- Git 和 GitHub 各自的作用；
- 工作区、暂存区、本地库、远程库之间的关系；

- 为什么要用分支，以及简单的冲突是什么。