

Git 多人协作注意事项

一、基础概念

二、协作流程步骤

三、避免冲突的核心原则

一、基础概念

- 本地仓库：自己电脑上的项目文件夹
- 远程仓库：GitHub 上的项目副本
- 分支：项目的 "平行宇宙"，可独立开发不干扰主分支
- 冲突：同一文件的同一部分被不同分支修改成不同内容时产生

二、协作流程步骤

1. 准备工作

- 所有人克隆远程仓库：`git clone 仓库地址`
- 各自创建功能分支：`git checkout -b feature/自己的功能名`

2. 日常开发

- 在自己分支独立开发，不直接修改 main 分支
- 定期保存修改：`git add .` + `git commit -m "备注"`

3. 关键同步操作（每天至少一次）bash

```
1  git checkout main          # 切到主分支
2  git pull origin main       # 拉取远程最新代码
3  git checkout 自己的分支名  # 切回自己分支
4  git merge main             # 合并主分支最新内容
```

4. 功能完成后

- 推送分支到远程: `git push origin 自己的分支名`
- 在 GitHub 发起 Pull Request (PR) 合并到 main
- 团队审查后合并代码

5. 冲突处理

- 冲突标记: `<<<<<<< HEAD` (主分支内容) 和 `=====` (自己分支内容) 之间
- 解决方式: 人工编辑保留正确内容, 删除冲突标记
- 解决后提交: `git add .` + `git commit -m "解决冲突"`

三、避免冲突的核心原则

1. 功能隔离: 尽量负责不同模块, 减少修改同一文件
2. 提前沟通: 若需修改同一文件, 明确分工 (如 A 改上半部分, B 改下半部分)
3. 频繁同步: 每天同步主分支代码, 避免分支差距过大
4. 小步提交: 经常提交小规模修改, 减少单次合并的代码量
5. 逐个合并: 按顺序合并分支 (A→B→C), 后合并者先同步前序代码