

NUC Course Sharing Program
github 教学

命令行与 Git 操作速查手册

——新手保姆级指南 · 拒绝死记硬背

适用对象：Git 初学者 / 课程贡献者

核心内容：文件管理 · 代码提交 · LFS

版本：v1.0 (2025 版)

目 录

1 ➜ 第一部分：终端基础（文件操作）	1
2 git 第二部分：Git 基础流程	2
3 📁 第三部分：Git LFS（大文件管理）	3
4 🔥 第四部分：进阶与救火	3

1 >_ 第一部分：终端基础 (文件操作)

在使用 Git 之前，你需要先学会如何在黑框框（终端）里“走路”和“拿东西”。

⌚ 查看与定位

› <code>pwd</code>	告诉我当前在哪里 (Print Working Directory)
› <code>ls</code>	看看当前文件夹里有什么 (List)
› <code>ls -l</code>	查看详细信息 (大小、日期、权限)
› <code>ls -a</code>	查看所有文件 (包括隐藏文件.git)

📂 切换目录 (走路)

› <code>cd < 目录名 ></code>	进入指定的文件夹
› <code>cd ..</code>	返回上一级目录 (退一步)
› <code>cd~</code>	回家 (切换到用户主目录)
› <code>cd -</code>	返回刚才呆的目录 (类似遥控器回看键)

文件管理 (增删改查)

› <code>mkdir < 目录名 ></code>	新建一个文件夹 (Make Directory)
› <code>touch < 文件名 ></code>	新建一个空文件
› <code>cp < 源 > < 目标 ></code>	复制文件 (Copy)
› <code>mv < 旧 > < 新 ></code>	重命名或移动文件 (Move)
› <code>rm < 文件名 ></code>	删除文件 (* 慎用！通常不进回收站 *)

2 git 第二部分：Git 基础流程

这是日常开发最常用的“三板斧”，请务必熟练掌握。

👤 0. 首次配置（你的名片）

```
# 配置用户名  
git config --global user.name "你的昵称"  
  
# 配置邮箱  
git config --global user.email "你的邮箱@example.com"
```

▶ 1. 开始项目

➤ <code>git init</code>	在当前目录初始化仓库 (生成.git)
➤ <code>git clone <URL></code>	下载远程仓库到本地

⬆ 2. 提交三部曲（核心）

```
# Step 1: 把文件放入暂存区 (打包快递)  
git add .  
  
# Step 2: 提交到本地仓库 (封箱贴单)  
git commit -m "这里写备注"  
  
# Step 3: 推送到远程 (发货)  
git push origin main
```

* 注：第一次推送可能需要用 `git push -u origin main`

⌚ 3. 同步与分支

➤ <code>git status</code>	查看状态 (* 最常用 *, 看看改了啥)
➤ <code>git pull</code>	拉取远程代码并合并 (更新)
➤ <code>git branch <名></code>	新建分支
➤ <code>git checkout <名></code>	切换分支

3 📁 第三部分：Git LFS (大文件管理)

普通 Git 传不了超过 100MB 的文件 (如 PDF, 视频), 必须用 LFS。

☒ LFS 配置流程

```
# 1. 初始化 LFS (每个仓库做一次)
git lfs init

# 2. 追踪大文件类型 (告诉 Git 哪些是大的)
git lfs track "*.pdf"
git lfs track "*.zip"
git lfs track "*.mp4"

# 3. 提交配置文件 (*非常重要, 否则无效*)
git add .gitattributes
git commit -m "配置 LFS"

# 4. 之后正常提交即可
git add .
git commit -m "添加大文件"
git push origin main
```

4 🔥 第四部分：进阶与救火

☒ 花式克隆 (省流/省时)

› 只克隆最近一次提交 (省流王):

```
git clone --depth 1 https://github.com/xxx.git
```

› 克隆但不下载大文件 (只拉代码):

```
GIT_LFS_SKIP_SMUDGE=1 git clone https://github.com/xxx.git
```

⌚ 后悔药 (慎用)

⚠ `git checkout HEAD`

丢弃工作区修改 (恢复到上次提交)

⚠ `git reset --hard <ID>`

强制回退到指定版本 (* 修改会丢失 *)

⚡ 常见报错速查

- ❗ refusing to merge unrelated histories:

```
git pull origin main --allow-unrelated-histories
```

- ❗ HTTP 413 / 推送失败: 缓存不够大

```
git config --global http.postBuffer 524288000
```