

add的使用

- `git add *.c`
- `git add .`
- `git add filename`

提交

- `git commit -m '提交说明'`

查看缓存区

- `git status` #告诉你 `git add` 和 `git commit` 的进展。status 信息还包括了添加缓存和移除缓存的相关指令。

项目历史

- `git log` #使用默认格式显示完整地项目历史。如果输出超过一屏，你可以用 空格键 来滚动，按 q 退出。
- `git log -n` #用 限制提交的数量。比如 `git log -n 3` 只会显示 3 个提交。
- `git log --oneline` #将每个提交压缩到一行。当你需要查看项目历史的上层情况时这会很有用。
- `git log --stat` #除了 `git log` 信息之外，包含哪些文件被更改了，以及每个文件相对的增删行数。
- `git log -p` #显示代表每个提交的一堆信息。显示每个提交全部的差异 (diff)，这也是项目历史中最详细的视图。
- `git log --author=""` #搜索特定作者的提交。可以是字符串或正则表达式。
- `git log --grep=""` #搜索提交信息匹配特定 的提交。可以是字符串或正则表达式。
- `git log ..` #只显示发生在 和 之间的提交。两个参数可以是提交 ID、分支名、HEAD 或是任何一种引用。
- `git log` #只显示包含特定文件的提交。查找特定文件的历史这样做会很方便。
- `git log --graph --decorate --oneline`
#还有一些有用的选项。`--graph` 标记会绘制一幅字符组成的图形，左边是提交，右边是提交信息。`--decorate` 标记会加上提交所在的分支名称和标签。`--oneline` 标记将提交信息显示在同一行，一目了然。