

# Git 原理详解及实用指南 - 扔物线 - 掘金小册

## 进阶 5：关于 add

前面说过了可以通过 `add` 来把改动的内容放进暂存区。这一节关于 `add` 再多说两点。都是基础的东西，但对新学 Git 的人可能有些用处。

### 1. add 后面加个点 ".": 全部暂存

---

`add` 指令除了 `git add 文件名` 这种用法外，还可以使用 `add .` 来直接把工作目录下的所有改动全部放进暂存区：

```
git status
```

```
➔ git-practice git:(branch1) ✕ git status
On branch branch1
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   b.txt
```

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   a.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        c.txt
```

```
git add .
git status
```

```
→ git-practice git:(branch1) X git add .
→ git-practice git:(branch1) X git status
On branch branch1
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

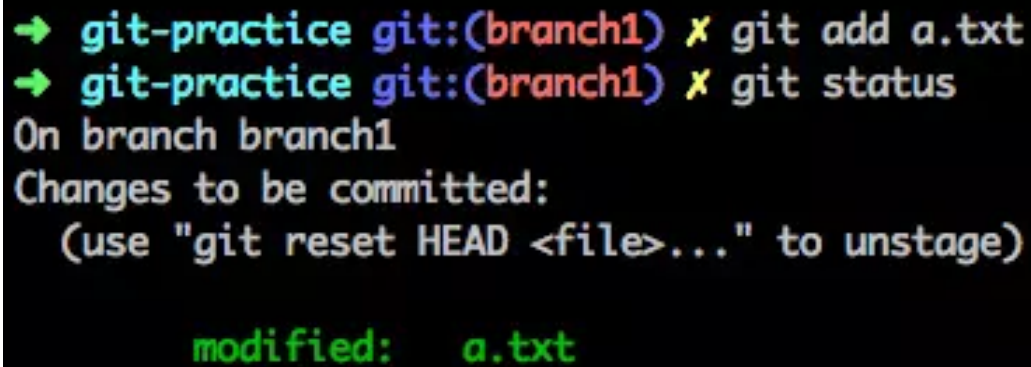
        modified:   a.txt
        new file:   b.txt
        new file:   c.txt
```

这个用法没什么特别的好处，但就一个字：方便（咦？）。你在用的时候会更加深刻地体会到。

## 2. add 添加的是文件改动，而不是文件名

假如你修改了文件 `a.txt`，然后把它 `add` 进了暂存区：

```
git add a.txt
git status
```

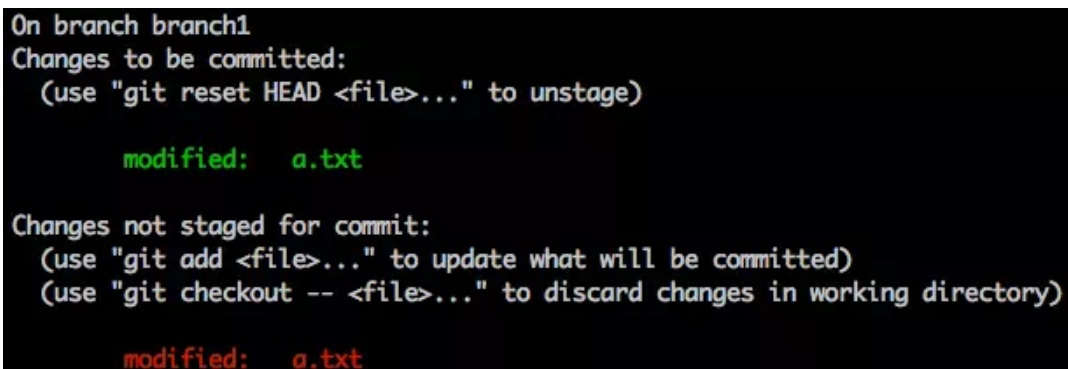


```
→ git-practice git:(branch1) ✗ git add a.txt
→ git-practice git:(branch1) ✗ git status
On branch branch1
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   a.txt
```

然后你又往 `a.txt` 里写了几行东西。这时候你再 `status` 一下的话：

```
git status
```



```
On branch branch1
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   a.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   a.txt
```

你会发现你的 `a.txt` 既在 "Changes to be committed" 的暂存区，又在 "Changes not staged for commit"。不用觉得奇怪，这是因为通

过 `add` 添加进暂存区的不是文件名，而是具体的文件改动内容。你在 `add` 时的改动都被添加进了暂存区，但在 `add` 之后的新改动并不会自动被添加进暂存区。在这时如果你提交：

```
git commit
```

那么你那些新的改动是不会被提交的。

这种逻辑也许会让新学 Git 的人有点困惑和感到麻烦，但当你使用 Git 一段时间后，你会发现这种设计很巧妙，而且不但不麻烦，还很方便。具体的原因我就不长篇大论地说了，你以后慢慢体会吧。

## 留言

评论将在后台进行审核，审核通过后对所有人可见

●

何佳羲

`add` 也支持匹配表达式，如 `add *.java`

add 也支持匹配表达式, 如 add \*.java

▲0 评论 1 月前

●

碧海银剑

神奇的 add, 点个赞!

▲0 评论 3 月前

●

xqgdmg

如果我一直都是用 add ., 也就是第二次也不指定文件名, 不是就可以了。

▲0 收起评论 3 月前

●

BeeShark iOS

执行完命令 git add . 之后, 再次修改文件 a.txt, 所做的改动并没有添加到暂存区, 必须再次执行命令 git add . 才可以。

3 月前

oliver 君

bingo

2 月前

评论审核通过后显示

评论

● 明果 Android 开发

初次将一个文件 add 到暂存区了，但是后来发现又希望忽略这个文件，还怎么办呢

▲0

收起评论 7 月前

● 今天 [互联网](#)

git checkout 文件名

7 月前

● 揽雀尾

use "git reset  
HEAD <file>..." to  
unstage

6 月前

• 自述文件

use "git checkout -  
- <file>..." to  
discard changes in  
working directory  
这个是把工作区中的  
改动删除

4 月前

---

• BeeShark iOS

git reset HEAD  
filename 将当前文  
件从暂存区回退到工  
作区

3 月前

---

• zachaxy

git rm filename

2 月前

• 1 • 2 • 下一  
页

评论审核通过后显示

评论

子豪伊诺

签到 打卡

▲0

评论 9 月前

---

宁攀□

终于知道每次新增或修改都需要重新 add 的原因了，非常感谢。

▲0

评论 9 月前

---

小葱□ 前端

留名

▲0

评论 9 月前

---

Jima

git add -A stages All

git add . stages new and modified, without  
deleted

git add -u stages modified and deleted,  
without new



▲0

收起评论 9 月前

- chenxiaochun

我试了一下，第二条  
不对吧？ git version  
2.8.4 (Apple Git-  
73)

9 月前

评论审核通过后显示

评论

- 

**Jima**

用发外 @光头哥哥 你有新的 bug 拜托 fix 一下

▲0

收起评论 9 月前

- 扔物线  
Android 布道师  
(假装) @  
HenCoder

已改，感谢！

9 月前

评论审核通过后显示

评论

●

**Ashley\_T** Android 应用开发工程师

原来 add 是这样的。。。学习了。。

▲0

评论 9 月前

●

**RTFSC**

666

▲0

评论 9 月前

●

**祺在掘金**

感谢

▲0

评论 9 月前

全文完

