

`git rm` 命令用于从工作区和索引中删除文件。

简介

```
git rm [-f | --force] [-n] [-r] [--cached] [--ignore-unmatch] [--quiet] [--] <file>...
```

Shell

描述

从索引中删除文件，或从工作树和索引中删除文件。`git rm` 不会从您的工作目录中删除文件。(没有任何选项只能从工作树中删除文件，并将其保留在索引中;)要删除的文件必须与分支的提示相同，并且在索引中不能对其内容进行更新，尽管可以使用`-f` 选项覆盖(默认行为)。当给出`--cached` 时，暂存区内容必须与分支的提示或磁盘上的文件相匹配，从而仅将文件从索引中删除。

使用 `git rm` 来删除文件，同时还会将这个删除操作记录下来；而使用 `rm` 来删除文件，仅仅是删除了物理文件，没有将其从 `git` 的记录中剔除。

直观的来讲，`git rm` 删除过的文件，执行 `git commit -m "commit message or mark"` 提交时，会自动将删除该文件的操作提交上去。

而对于用 `rm` 命令直接删除的文件，执行 `git commit -m "commit message or mark"` 提交时，则不会将删除该文件的操作提交上去。不过不要紧，即使你已经通过 `rm` 将某个文件删除掉了，也可以再通过 `git rm` 命令重新将该文件从 `git` 的记录中删除掉，这样的话，在执行 `git commit -m "commit message or mark"` 以后，也能将这个删除操作提交上去。

如果之前不小心用 `rm` 命令删除了一大批文件呢？如此时用 `git rm` 逐个地再删除一次就显得相当卵痛了。可如下的方式做提交：`git commit -am "commit message or mark"`

示例

以下是一些示例 -

在 `git` 中我们可以通过 `git rm` 命令把一个文件删除，并把它从 `git` 的仓库管理系统中移除。但是注意最后要执行 `git commit` 才真正提交到 `git` 仓库。

示例 1

删除 `text1.txt` 文件，并把它从 `git` 的仓库管理系统中移除。

```
git rm text1.txt
```

Shell

示例 2

删除文件夹：`mydir`，并把它从 `git` 的仓库管理系统中移除。

```
git rm -r mydir
```

Shell

示例 3

```
$ git add 10.txt
$ git add -i

        staged      unstaged path
1:      +0/-0       nothing 10.txt
2:      +0/-0       nothing branch/t.txt
3:      +0/-0       nothing branch/t2.txt

*** Commands ***
1: [s]tatus      2: [u]pdate      3: [r]evert      4: [a]dd untracked
5: [p]atch       6: [d]iff       7: [q]uit       8: [h]elp
What now> 7
Bye.
$ git rm --cached 10.txt
rm '10.txt'
$ ls
10.txt  2  3.txt  5.txt  readme.txt
$ git add -i

        staged      unstaged path
1:      +0/-0       nothing branch/t.txt
2:      +0/-0       nothing branch/t2.txt

*** Commands ***
1: [s]tatus      2: [u]pdate      3: [r]evert      4: [a]dd untracked
5: [p]atch       6: [d]iff       7: [q]uit       8: [h]elp
```

Shell

在通过 `git add 10.txt` 命令把文件 `10.txt` 添加到索引库中后，又通过 `git rm --cached 10.txt` 把文件 `10.txt` 从 `git` 的索引库中移除，但是对文件 `10.txt` 本身并不进行任何操作。

另外对于已经被 `git rm` 删除掉(还没被提交)的文件或目录，如果想取消其操作的话，可以首先通过 `git add -i` 的子命令 `revert` 从索引库中把它们剔除，然后用 `git checkout <文件>` 命令来达到取消的目的。

示例 4

```
$ git rm Documentation/*.txt
```

Shell

从 **Documentation** 目录及其任何子目录下的索引中删除所有 **.txt** 文件。

示例 5

```
git rm -f git-*.sh
```

Shell

因为这个例子让 **shell** 扩展星号(即显式列出文件), 它不会删除子目录中的文件, 如:
subdir/git-foo.sh 文件不会被删除。