**Git后悔药 - 版本回退操作（三）**

**（3）git reset --hard命令**

git reset --hardcommit-id命令：回退到指定版本。（hard：强硬，严格的回退）

**该命令不仅移动了分支中HEAD指针的位置，还将工作区和暂存区中数据也回退到了指定的版本。**

（提示：该命令慎用。）

示例开始：

首先在版本库中的readme.txt文件中添加一行内容，并提交该内容。我们的目的就是要再回退到该版本。

**1）查看本地版本库日志**

|  |
| --- |
| Bash # 1.使用git log查看历史版本记录 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git log --oneline 529ad74 (HEAD -> master) 第3次提交，新增内容：readme.txt file v3 1b23cae 第2次提交，新增内容：readme.txt file v2 2612adf 第1次提交，创建readme.txt文件  # 2.使用git reflog查看历史版本记录 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reflog 529ad74 (HEAD -> master) HEAD@{0}: commit: 第3次提交，新增内容：readme.txt file v3 1b23cae HEAD@{1}: commit: 第2次提交，新增内容：readme.txt file v2 2612adf HEAD@{2}: commit (initial): 第1次提交，创建readme.txt文件  # 3.查看readme.txt文件的内容 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ cat readme.txt readme.txt file v1 readme.txt file v2 readme.txt file v3 |

**2）向readme.txt文件中新增一行数据，并提交到本地版本库**

|  |
| --- |
| Bash # 1.新增数据 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ echo "readme.txt file v4" >> readme.txt  # 2.查看readme.txt文件内容 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ cat readme.txt readme.txt file v1 readme.txt file v2 readme.txt file v3 readme.txt file v4  # 3.提交到本地版本库 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git commit -a -m '第4次提交，新增内容：readme.txt file v4' warning: LF will be replaced by CRLF in readme.txt. The file will have its original line endings in your working directory [master e72b30f] 第4次提交，新增内容：readme.txt file v4  1 file changed, 1 insertion(+)  # 4.现在查看此时本地版本库日志 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git log --oneline e72b30f (HEAD -> master) 第4次提交，新增内容：readme.txt file v4 529ad74 第3次提交，新增内容：readme.txt file v3 1b23cae 第2次提交，新增内容：readme.txt file v2 2612adf 第1次提交，创建readme.txt文件 |

**3）现在比对工作区与暂存区、暂存区与本地版本库的差异**

|  |
| --- |
| Bash # 1.比对工作区与暂存区中文件的差异 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git diff readme.txt  # 2.比对暂存区与本地版本库中文件的差异 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git diff --cached readme.txt |

我们可以看到此时，工作区、暂存区与本地版本库中的readme.txt文件状态无差异。

**4）开始回退操作，退回到V3版本**

使用git reset --hard HEAD^命令，退回到前一个版本。

|  |
| --- |
| Bash # 回退一个提交版本 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reset --hard HEAD^ HEAD is now at 529ad74 第3次提交，新增内容：readme.txt file v3 |

说明：

HEAD is now at 529ad74 第3次提交，新增内容：readme.txt file v3

意思：HEAD现在位于commit-id为529ad74的提交，提交说明为：第3次提交，新增内容：readme.txt file v3。

**5）回退后，对比工作区、暂存区与本地库中版本中文件的差异**

|  |
| --- |
| Bash # 1.比对工作区与暂存区中文件的差异 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git diff readme.txt  # 2.比对暂存区与本地版本库中文件的差异 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git diff --cached readme.txt  # 3.比对工作区与本地版本库中文件的差异 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git diff HEAD readme.txt |

回退后，我们再次对比了工作区、暂存区与本地库中版本中文件的差异：没有内容差异，完全一样。

说明：工作区、暂存区和本地库中的内容，都回退到了之前的版本。（重要）

**6）查看本地版本库的提交日志信息**

|  |
| --- |
| Bash # 1.使用git log查看历史版本记录 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git log --oneline 529ad74 (HEAD -> master) 第3次提交，新增内容：readme.txt file v3 1b23cae 第2次提交，新增内容：readme.txt file v2 2612adf 第1次提交，创建readme.txt文件  # 2.使用git reflog查看历史版本记录 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reflog 529ad74 (HEAD -> master) HEAD@{0}: reset: moving to HEAD^ e72b30f HEAD@{1}: commit: 第4次提交，新增内容：readme.txt file v4 529ad74 (HEAD -> master) HEAD@{2}: commit: 第3次提交，新增内容：readme.txt file v3 1b23cae HEAD@{3}: commit: 第2次提交，新增内容：readme.txt file v2 2612adf HEAD@{4}: commit (initial): 第1次提交，创建readme.txt文件 |

从上我们可以看到：（重点）

* 使用git log命令查看历史版本记录，发现已经看不到第4次提交了。
* 使用git reflog命令查看历史版本记录，第四次提交这个版本仍然存在的。

**7）恢复到回退前版本**

上面说了，使用git reset --hard命令回退，做了三个操作：

1. 移动HEAD指针（版本库的回退）。
2. 暂存区的内容回退到HEAD指针指向的版本。
3. 工作区的内容也回退到HEAD指针指向的版本。

我们现在来查看一下工作目录中的文件状态。

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master nothing to commit, working tree clean |

可以看到工作目录中是非常干净的。

所以若要恢复到回退之前的版本，只能使用git reset --hard命令回退。

|  |
| --- |
| Bash # 1.查看可回退的历史版本 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reflog 529ad74 (HEAD -> master) HEAD@{0}: reset: moving to HEAD^ e72b30f HEAD@{1}: commit: 第4次提交，新增内容：readme.txt file v4 529ad74 (HEAD -> master) HEAD@{2}: commit: 第3次提交，新增内容：readme.txt file v3 1b23cae HEAD@{3}: commit: 第2次提交，新增内容：readme.txt file v2 2612adf HEAD@{4}: commit (initial): 第1次提交，创建readme.txt文件  # 2.回退到第四次提交 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reset --hard e72b30f HEAD is now at e72b30f 第4次提交，新增内容：readme.txt file v4  # 3.查看工作目录中文件状态 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master nothing to commit, working tree clean  # 4.查看版本历史 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git log --oneline e72b30f (HEAD -> master) 第4次提交，新增内容：readme.txt file v4 529ad74 第3次提交，新增内容：readme.txt file v3 1b23cae 第2次提交，新增内容：readme.txt file v2 2612adf 第1次提交，创建readme.txt文件  # 5.查看可回退的历史版本 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git reflog e72b30f (HEAD -> master) HEAD@{0}: reset: moving to e72b30f 529ad74 HEAD@{1}: reset: moving to HEAD^ e72b30f (HEAD -> master) HEAD@{2}: commit: 第4次提交，新增内容：readme.txt file v4 529ad74 HEAD@{3}: commit: 第3次提交，新增内容：readme.txt file v3 1b23cae HEAD@{4}: commit: 第2次提交，新增内容：readme.txt file v2 2612adf HEAD@{5}: commit (initial): 第1次提交，创建readme.txt文件  # 6.查看readme.txt内容 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ cat readme.txt readme.txt file v1 readme.txt file v2 readme.txt file v3 readme.txt file v4 |

我们查看到，readme.txt文件的内容已经完全恢复。

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/3da3b2998bb7 来源：简书 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。