**Git操作 - 撤销操作（一）**

**1、撤销操作说明**

我们在使用Git版本管理时，往往需要撤销某些操作。比如说我们想将某个修改后的文件撤销到上一个版本，或者是想撤销某次多余的提交，都要用到Git的撤销操作，因此撤销操作在平时使用中还是比较多的。

本文介绍几种，对于已修改过的文件，需要进行撤销操作，根据修改文件出现的位置，可以分为三种情况：

1. 仅仅是工作区中内容进行了修改，还未添加到暂存区。
2. 文件已经添加到暂存区，但是还未提交到本地版本库。
3. 文件已经提交到本地版本库。

前两种可以叫撤销操作，后面一种叫回退版本，不同的情况具有不同的撤销方式。

下面就撤销操作给出详细的解释。

**2、撤销工作区中文件的修改**

如果工作区的某个文件被改乱了，但还没有提交，可以用git restore或者git checkout命令找回本次修改之前的文件。

前提：test.txt文件已在本地版本库中。

**1）修改test.txt文件中的内容**

|  |
| --- |
| Bash # 1.查看test.txt文件内容 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ cat test.txt hello git  # 2.进行修改内容 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ echo "hello git v2" >> test.txt  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ cat test.txt hello git hello git v2 |

**2）查看工作目录中文件的状态**

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Changes not staged for commit:  (use "git add <file>..." to update what will be committed)  (use "git restore <file>..." to discard changes in working directory)  modified: test.txt  no changes added to commit (use "git add" and/or "git commit -a") |

好好的说明一下这些提示信息

* On branch master：当前工作在主分支上。
* Changes not staged for commit:：暂存区中没有可提交的变化。
* git add <file>...：可使用该命令把更新添加到暂存区。
* git restore <file>...：丢弃工作区的修改。同git checkout -- <file>...命令一样。
* no changes added to commit (use "git add" and/or "git commit -a")：可以使用git add命令添加到暂存区，也可以使用git commit -a命令直接提交到本地版本库。

Git的命令提示是非常详细的，我们按照提示信息操作就可以。

**说明：**

老版本Git会提示：(use "git checkout -- <file>..." to discard changes in working directory)

新版本Git会提示：(use "git restore <file>..." to discard changes in working directory)

git restore <file>...和git checkout -- <file>...命令：

* git checkout这个命令承担了太多职责，既被用来切换分支，又被用来恢复工作区文件，对用户造成了很大的认知负担。
* Git社区发布了Git的2.23版本中，有一个特性非常引人瞩目，就是引入了两个新命令 git switch和git restore，用以替代现在的git checkout。
* 即：Git2.23版本开始，git restore <file>...命令可代替git checkout -- <file>...命令文件恢复的工作。

**3）撤销工作区中test.txt文件的修改**

|  |
| --- |
| Bash # 1.使用git restore 命令撤销工作区中的操作 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git restore test.txt  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master nothing to commit, working tree clean  # 2.再次修改test.txt文件 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ echo "hello git v3" >> test.txt  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Changes not staged for commit:  (use "git add <file>..." to update what will be committed)  (use "git restore <file>..." to discard changes in working directory)  modified: test.txt  no changes added to commit (use "git add" and/or "git commit -a")  # 使用git checkout命令撤销工作区中的操作 # 撤销操作固定写法“git checkout -- 文件” L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git checkout -- test.txt  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master nothing to commit, working tree clean |

|  |
| --- |
| 说明：git checkout -- file命令中的--很重要，没有--，就变成了“切换到另一个分支”的命令，我们在后面的分支管理中会再次遇到git checkout命令。 |

**4）总结**

1. git restore <file>...和git checkout -- <file>...命令原理：
2. 这里有两种情况：
3. 一种是test.txt自修改后还没有被放到暂存区，现在的撤销修改就回到和版本库一模一样的状态；
4. 一种是test.txt已经添加到暂存区后，又作了修改，现在的撤销修改就回到添加到暂存区后的状态。
5. 撤销命令是一个危险的命令，这个很重要，你对该文件做的任何修改都会消失。原理是你拷贝了该文件在暂存区或者本地版本库中的副本，来覆盖工作区的该文件。
6. 即：工作区的文件变化一旦被撤销，就无法找回了，除非你确实不想要这个文件了。

|  |
| --- |
| 所以我的理解：撤销命令实际的操作是重置（原文件覆盖），而实际的效果上是撤回。 |

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/da3542af4dfa 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。