**Git后悔药 - reset版本回退说明**

**1、什么版本回退**

版本回退也可以叫**回滚**。

若修改过的文件，不仅添加到了暂存区，还提交到了本地版本库，还能撤销吗？ 已经无法撤销修改了，但是可以回退到修改前的版本。

版本回退也是撤销操作的一种，但我更愿意和前两种撤销操作分开理解。

|  |
| --- |
| 引用廖雪峰比喻：  你不断对文件进行修改，然后不断提交修改到版本库里，就好比玩RPG游戏时，每通过一关就会自动把游戏状态存盘，如果某一关没过去，你还可以选择读取前一关的状态。有些时候，在打Boss之前，你会手动存盘，以便万一打Boss失败了，可以从最近的地方重新开始。  Git也是一样，每当你觉得文件修改到一定程度的时候，就可以“保存一个快照”，这个快照在Git中被称为commit。一旦你把文件改乱了，或者误删了文件，还可以从最近的一个commit恢复，然后继续工作，而不是把几个月的工作成果全部丢失。 |

**2、需要了解两个知识点**

能够了解是什么即可。

**（1）HEAD是什么**

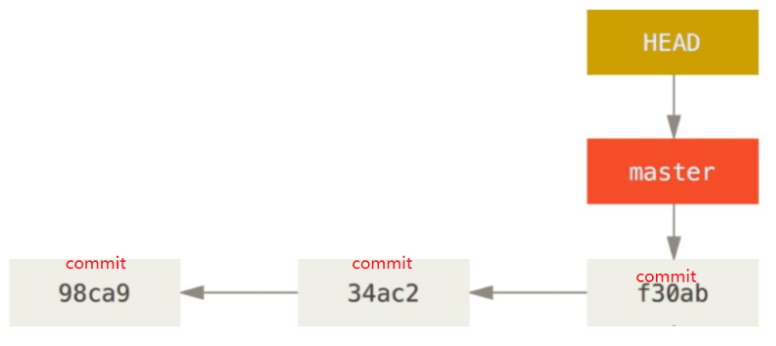
HEAD也可以称为HEAD指针。

Git仓库初始化之后，会默认创建一个master分支，即主分支。这是Git对版本进行管理的时间线，即Git的每次提交，都会自动把它们串成一条时间线，这条时间线就是一个分支。

如果没有新建分支，那么就只有一条时间线，即只有一个分支，master分支（主分支）。在这条master时间线上有很多版本的时间节点（提交commit），而HEAD指针则指向的是当前分支上，刚刚提交的版本时间节点。

如果上面看不太懂的话，就记住：HEAD指针指向分支，分支指向具体commit。所以HEAD指针代表的就是当前工作分支上最新一次的提交。

（具体原理以后会总结，这里不详细展开）



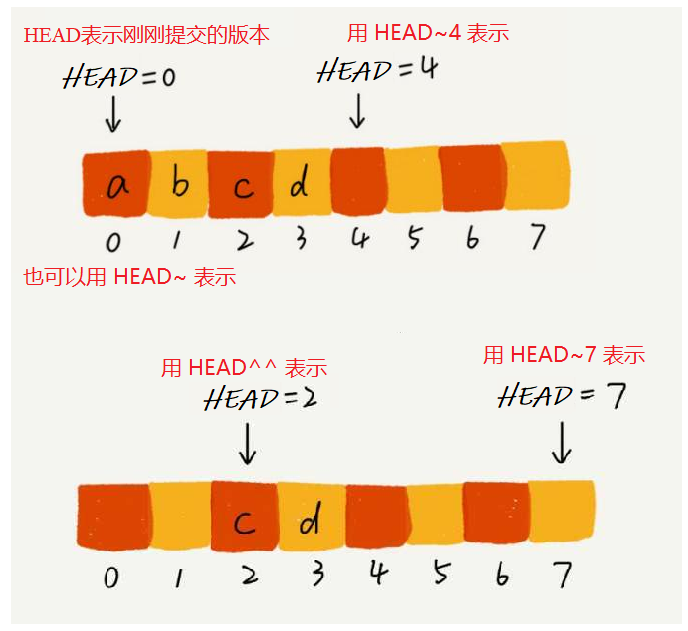
**（2）HEAD指针用法**

上面说了，HEAD指针代表的就是当前工作分支上最新一次的提交。换句话说HEAD这个词就等同于当前工作分支上最新一次提交的别名。

HEAD指针可以表示相对位置，有几种表示方法：

* 单写HEAD，表示当前工作分支的最新提交版本。
* 可以通过^表示前一个版本，例如HEAD^，如要表示前两个版本，则需要加两个^，示例：HEAD^^，以此类推。
* 但是要选择之前版本跨度较多的时候，^的数量就不是很好数了（书写可读性差），此时可以使用~加数字表示当前版本之前的第几个版本。例如：HEAD~100表示为最新版本的往前第100个版本。
* 当时我们也可以使用HEAD@{0}的方式进行表示，HEAD@{0}等于HEAD。HEAD@{1}等于HEAD^，也就是前一个版本。值的数字越小，表示版本越新，数字越大表示版本越旧。

HEAD指针的作用示意图：



**3、git reflog命令介绍**

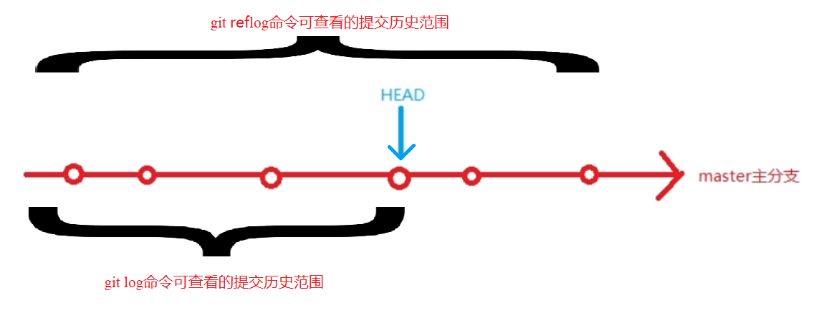
git reflog命令是用来恢复本地错误操作很重要的一个命令，他和git log命令一样都可以查看本地版本库的历史提交信息。

但是不同的是，git reflog命令能够查看可引用历史提交版本。

可引用历史提交版本，什么意思？

* 使用git log命令只可以查看到HEAD指针及其之前的版本信息，如果版本发生过回退操作，则可能会出现，HEAD指针之后仍存在历史提交版本的情况，而这些提交版本信息通过git log命令是看不到的。
* 我们可以通过使用git reflog命令，就可查看到所有历史版本信息。由于查看所有历史版本信息的目的，大多是为了进行版本回退或恢复操作所使用，从中找到所需的commit索引，所以该命令被命名为reflog，即：引用日志。

git log命令与git reflog命令作用范围示意图：



作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/859161daf265 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。