

git reset 命令

适用场景：使用 Git 的 `commit` 命令提交代码后，发现本次的提交有错误的内容。此时有 2 种解决的办法：

- 1、修改错误的内容，重新提交一次。
- 2、使用 `git reset` 命令撤销此次的错误提交。

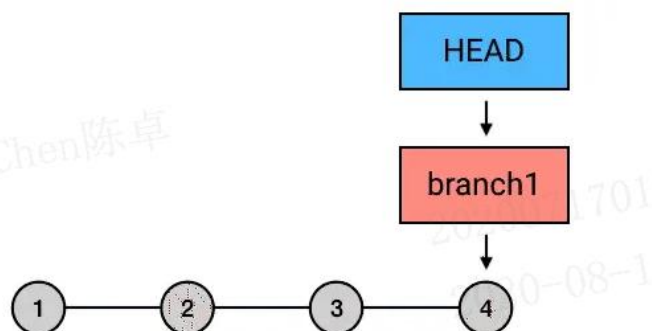
但第一种方法，会有一个明显的弊端，那就是会在记录中留下一次错误的版本。

`git reset` 的三种模式分别为：

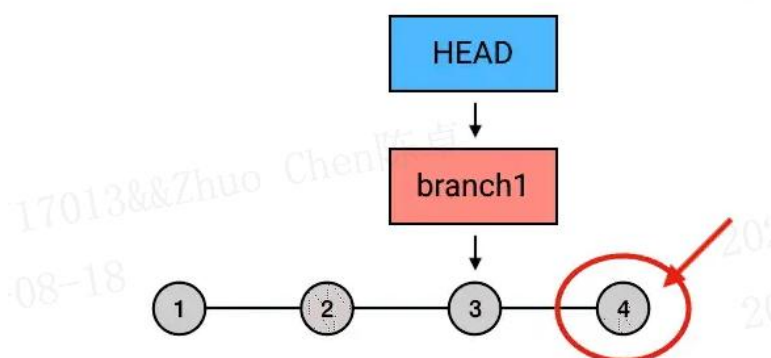
```
git reset --hard
git reset --soft
git reset --mixed (默认)
```

1、`reset --hard`:重置暂存区和工作目录中的内容

回滚前：



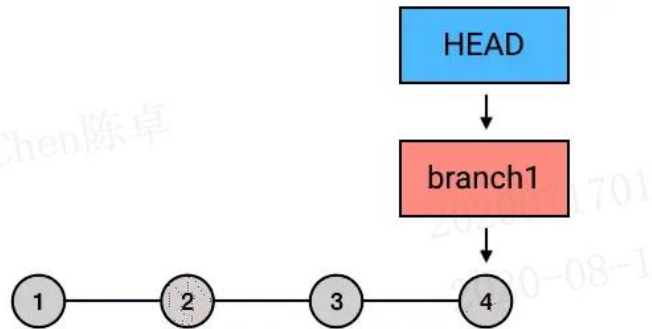
回滚后：



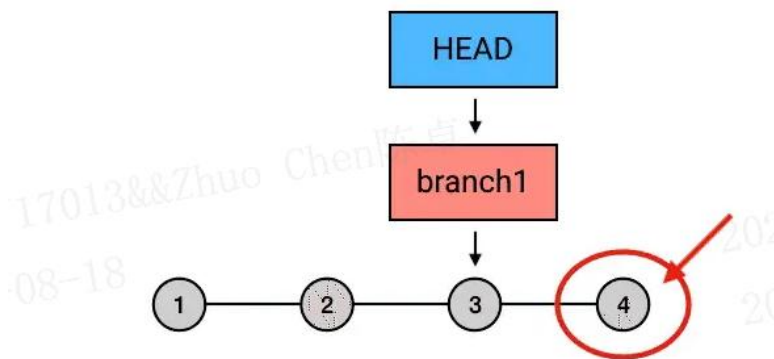
`git reset --hard` 会在重置 `HEAD` 和 `branch` 的同时，重置暂存区和工作目录当中的内容。假设你在镜像 4 的位置使用了 `git reset --hard` 命令回滚到了镜像 3，那么你的暂存区和工作目录中的内容就会完全重置为和 `HEAD` 所指向新位置相同的内容。即：没有 `commit` 的修改都会被擦除。

2、reset --soft:保留工作目录和暂存区中的内容，并把重置 HEAD 所带来的差异放入暂存区

重置前：



重置后：



什么是“重置 HEAD 所带来的差异”？

当从镜像 4 回滚到镜像 3 时，原节点镜像 4 相对于 reset 目标节点镜像 3 的变化 $\Delta_{4,3}$ 就是重置 HEAD 所带来的差异。

3、reset –mixed（不加参数默认为 mixed）:保留工作目录，清空暂存区，同时将暂存区中的内容以及重置 HEAD 所带来的差异放入工作目录中。

hard、soft 和 mixed 三个参数之间的比较

（假设由 a_n 节点回滚到了 a_1 节点）

方式	工作目录 (w)	暂存区 (s)	说明
hard	$W=0$	$S=0$	没有 commit 的内容被完全擦除
soft	W 不变	$S=S+\Delta_{n1}$	
mixed	$W=W+S+\Delta_{n1}$	$S=0$	