


GIT 中如何打标签 (git tag)

 工具篇 专栏收录该内容

0 订阅 8 篇文章

订阅专栏

- 前言
- 0x1 GIT 标签常用指令
- 0x2 轻量标签
- 0x3 附注标签
- 0x4 远程仓库中的标签

前言

在我们开发的过程中，可能经过多次 `commit` 提交才会确定一个版本，那么除了用 `commit comments` 来标识一次提交记录，还有什么更好的办法呢？GIT 为我们提供了标签这个功能，使用标签可以对某一次提交记录做上一个标记，就像我们看书的时候，看到很有意思想要做个记号，可以用小的标签纸贴在那一页的边上，这让我们以后可以更方便的索引。

0x1 GIT 标签常用指令

GIT 中的标签分为两种，一种是轻量标签（lightweight tag），一种是附注标签（annotated tag）。以下是一些常用的与标签相关的命令：

- `git tag <lightweight_name>`：为当前分支所在的提交记录打上轻量标签
- `git tag -a <anotated_name> -m <tag_message>`：为当前分支所在的提交记录打上附注标签。
- `git tag`：列出所有的标签名。
- `git tag -d <tag_name>`：删除某个标签，本质上就是移除 `.git/refs/tags/` 中对应的文件。
- `git show <tag_name>`：显示标签对应提交记录的具体信息。
- `git push <remote> <tag_name>`：推送某个标签到远程仓库。
- `git push <remote> --tags`：推送所有标签到远程仓库。
- `git push <remote> --delete <tag_name>`：删除远程仓库中的某个标签。

[注]：标签与分支不相关。

0x2 轻量标签

轻量标签的创建十分简单，使用命令 `git tag <lightweight_name>` 则会为当前所在的提交记录打上一个轻量标签。那么这个命令会发生什么呢？其实很简单，在使用这个命令后，会在 `.git/refs/tags/` 目录下生成一个以 `<lightweight_name>` 为名字的文件，文件中存放了标签所对应提交记录的 `commit` 对象的 SHA-1 值。如下图所示：

```
nino@ubuntu:~/Workspace/GitLearn$ git log
commit f3e7a0723af205dae99c22866fd2752abf654c8b (HEAD -> master)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 21:14:07 2021 -0700

    third commit

commit e98cc5d318b3ff994771b9b8af12a3d236c550a4
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 20:19:14 2021 -0700

    second commit

commit e4fc5e2eec8a0daf29b55e6260ef35103f86cf63
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 18:25:09 2021 -0700

    first commit
nino@ubuntu:~/Workspace/GitLearn$ git tag v1.0.0.lightweight
nino@ubuntu:~/Workspace/GitLearn$ cat .git/refs/tags/v1.0.0.lightweight
f3e7a0723af205dae99c22866fd2752abf654c8b
```

假设我想为上述三个提交记录中的第一次提交打上版本 0 的标签，也是很容易做到的，同时使用 `git log` 命令时也会显示出标签的名称：

```
nino@ubuntu:~/Workspace/GitLearn$ git tag v0.0.0.lightweight e4fc5e
nino@ubuntu:~/Workspace/GitLearn$ git log
commit f3e7a0723af205dae99c22866fd2752abf654c8b (HEAD -> master, tag: v1.0.0.lightweight)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 21:14:07 2021 -0700

    third commit

commit e98cc5d318b3ff994771b9b8af12a3d236c550a4
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 20:19:14 2021 -0700

    second commit

commit e4fc5e2eec8a0daf29b55e6260ef35103f86cf63 (tag: v0.0.0.lightweight)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 18:25:09 2021 -0700

    first commit
```

0x3 附注标签

创建附注标签也非常简单，只要加上 `-a` 选项即可。同时还可以通过 `-m` 选项指定一条存储在标签中的信息，如果没有指定则 GIT 会打开编辑器要求你输入。假如我们为当前的提交记录打上一个附注标签，可以查看标签如下：

```
nino@ubuntu:~/Workspace/GitLearn$ git tag -a v1.0.0.anotated -m "This our first version"
nino@ubuntu:~/Workspace/GitLearn$ git log
commit f3e7a0723af205dae99c22866fd2752abf654c8b (HEAD -> master, tag: v1.0.0.lightweight, tag: v1.0.0.anotated)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 21:14:07 2021 -0700

    third commit

commit e98cc5d318b3ff994771b9b8af12a3d236c550a4
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 20:19:14 2021 -0700

    second commit

commit e4fc5e2eec8a0daf29b55e6260ef35103f86cf63 (tag: v0.0.0.lightweight)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 18:25:09 2021 -0700

    first commit
nino@ubuntu:~/Workspace/GitLearn$ git show v1.0.0.anotated
tag v1.0.0.anotated
Tagger: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Thu Oct 14 18:57:13 2021 -0700

This our first version

commit f3e7a0723af205dae99c22866fd2752abf654c8b (HEAD -> master, tag: v1.0.0.lightweight, tag: v1.0.0.anotated)
Author: user_sunwenchuan <user_swczzzycs@outlook.com>
Date:   Wed Oct 13 21:14:07 2021 -0700

    third commit

diff --git a/bye b/bye
index 05222e7..cbda105 100644
--- a/bye
+++ b/bye
@@ -1,2 +1,3 @@
 Bye GIT
+modify file bye
+modify file bye again
```

附注标签同样也在 `.git/refs/tags/` 目录下生成对应的文件，但是这个文件中不再存储该标签对应的 `commit` 对象的 SHA-1 值，而是存放了一个称为 `tag` 对象的 SHA-1 值，这个 `tag` 对象中就存储了与这个附注标签相关的信息（包括打标签者的名字、电子邮件地址、日期时间、一个标签信息）和这个标签所对应的 `commit` 对象的 SHA-1 值。我们可以查看如下：

```
nino@ubuntu:~/Workspace/GitLearn$ cat .git/refs/tags/v1.0.0.anotated
b73469b2abe3fd57ef25879f857e1a07d29faa75
nino@ubuntu:~/Workspace/GitLearn$ git cat-file -t b73469
tag
nino@ubuntu:~/Workspace/GitLearn$ git cat-file -p b73469
object f3e7a0723af205dae99c22866fd2752abf654c8b
type commit
tag v1.0.0.anotated
tagger user_sunwenchuan <user_swczzzycs@outlook.com> 1634263033 -0700

This our first version
```

0x4 远程仓库中的标签

我们想想标签文件所在的文件夹和其中保存的信息，是不是觉得其和分支很像？这使得他也有一个特点，就是在我们使用 `git push` 命令推送某个分支的时候，标签并不会被推送到远程仓库，所以我们必须显式的推送标签到远程仓库。

使用下面两种命令则可以推送标签到远程仓库：

- `git push <remote> <tag_name>`：推送某个标签到远程仓库。
- `git push <remote> --tags`：推送所有标签到远程仓库。

同样，在删除本地的标签后，要想删除远程仓库的标签，也必须使用的显式的命令：

- `git push <remote> --delete <tag_name>`：删除远程仓库中的某个标签。
- `git push <remote> :refs/tags/<tagname>`：删除远程仓库某个标签的等价方式，相当于将冒号前面的空值推送到远程标签名，从而高效地删除它。