

git tag

1. Git 标签

- **Git 标签 (Tag)** 是版本控制系统中的特殊标识，用于标记特定的提交。它们通常用于发布生产版本，以便用户可以轻松找到代码的特定版本。
- 标签是永久性的，一旦创建，就不会被删除。它们与提交紧密关联，可以追溯到代码库中的具体更改。
- 标签可以用于创建分支，以便在开发新功能时保持生产版本的稳定性。例如，你可以基于一个生产版本创建一个新的分支，而不会影响生产版本本身。

2. 创建和删除标签

命令	描述	示例
git tag	创建新的标签	git tag (版本号) 提交哈希
git tag -d	删除指定的标签	git tag -d (版本号)
git tag -l	列出所有标签	git tag -l (显示所有标签)
git tag -f	强制覆盖现有标签	git tag -f (版本号) 提交哈希

3. 使用标签

3.1 Lightweight Tag (轻量级标签)

- 轻量级标签只是指向特定提交的指针，没有附加任何元数据。
- 创建轻量级标签的命令是 `git tag`，后面跟版本号、提交哈希或分支名。例如：

```
git tag v1.0.0
```

3.2 Annotated Tag (带注释的标签)

- 创建新分支并切换到该分支
- 提交更改，并推送到远程仓库。GPG 签名 提交

```
git tag -a v1.0.0 -m "Release version 1.0.0"
```

4. ?? ??? ??

4.1 ?? ?? ??

```
git tag
```

4.2 ?? ??? ?? ??

```
git tag v1 <tag message>
```

4.3 ?? ????

```
git checkout v1
```

- 分支 : 分支 **HEAD(detached HEAD)** 指向 提交
- → 分支 指向 提交，分支 指向 提交 提交 提交

```
git checkout -b new-branch-from-v1
```

4.4 ?? ??

```
git tag -d v1
```

4.5 ?? ??? ?? ??

```
git push origin v1
```

- 分支 分支 分支

```
git push origin --tags
```

4.6 ????? ?? ??

```
git push origin --delete tag v1
```

5. git describe: ?? ??? ?? ??

5.1 ???

```
git describe
```

- HEAD 指向 2 个 commit 的 父 commit

```
v1.0.0-2-g9f23a1c
```

- v1.0.0 → 1 个 commit 的 父 commit
- -2 → 2 个 commit 的 父 commit
- g9f23a1c → 2 个 commit 的 父 commit

6. ?????

- 创建分支 fix-hotbug，并切换到该分支
- 修复 bug 并提交到该分支

```
git checkout -b fix-hotbug v1.0.0
```

7. ?? ?? ??

```
# 创建分支 v2.0.0  
git tag v2.0.0
```

```
# 创建分支 v2.0.0
```

```
git tag -a v1.0.0 <tag> -m "Initial release"
```

```
# tag
git tag
```

```
# push
git push origin --tags
```

? ??? ??

操作	コマンド
タグを作成	git tag v1.0.0
タグを作成 (メッセージ付き)	git tag -a v1.0.0 -m "tag message"
タグを列出す	git tag
タグをプッシュ	git push origin v1.0.0
タグとコードをプッシュ	git push origin --tags
タグを削除	git tag -d v1.0.0
タグとコードを削除してプッシュ	git push origin --delete tag v1.0.0
タグとコミット履歴をプッシュ	git describe