

# 毕小烦的学习笔记

每天进步一点点 请关注毕小烦的微博 [weibo.com/wirelessqa](http://weibo.com/wirelessqa)

目录视图

摘要视图

RSS 订阅

我的作品

  
Markdown  
Typora  
GitBook  
互联网写作工具  
**一本通**  
毕小烦 著

  
Android Studio  
实用指南  
毕小烦 著

个人资料

  
毕小烦



访问： 2290115次

积分： 16461

等级： **BLOG** 7

排名： 第613名

原创： 188篇 转载： 25篇

译文： 4篇 评论： 864条

联系方式

微博: [wirelessqa](#)

扣扣: 782670627

微信公众号: [wirelessqa](#)



文章搜索

赠书 | 异步2周年,技术图书免费送 每周荐书: 渗透测试、K8s、架构 (评论送书) 项目管理+代码托管+云原生 开发手册 流畅

## Git入门指南十:应用标签

2014-02-28 19:15

4659人阅读

评论

分类：

玩转 Git (13) ▼

版权声明：本文为博主原创文章，未经博主允许不得转载。

目录(?)

[+]

## 十四. 应用标签

Git tag参数：

用法: git tag [-a|-s|-u <key-id>] [-f] [-m <说明>|-F <文件>] <标签名> [<头>]

或: git tag -d <标签名>...

或: git tag -l [-n[<num>]] [--contains <提交>] [--points-at <对象>]

[<模式>...]

或: git tag -v <标签名>...

-l, --list 列出tag名称

-n[<n>] 每个 tag 信息打印 <n> 行

-d, --delete 删除 tags

-v, --verify 验证 tags

Tag 创建选项

-a, --annotate 注解 tag, 需要一个说明

-m, --message <说明> tag 说明

-F, --file <文件> 从文件中读取提交说明

-s, --sign 注解并 GPG 签名的 tag

--cleanup <模式> 设置如何删除提交说明里的空格和#注释

-u, --local-user <key id>

使用另外的私钥签名 tag

-f, --force 如果存在, 替换现有的 tag

--column[=<风格>] 以列的方式显示 tag

Tag 列表选项

--contains <提交> 只打印包含提交的tags

--points-at <对象> 只打印tags对象

### 14.1 查看标签

Git tag -l <name>, <name>支持通配符

# 查看某个tag

bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark\$ git tag -l v2.2

v2.2

# 列出所有的tag名称

bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark\$ git tag -l

v2.2

# 查看某个tag的详情(有问题)

bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark\$ git tag -v v2.2

object 2c2e8b14d37d26c85fd07a5d21620f51eaad01fe

type commit

tag v2.2

tagger xiaopeng.bxp <xiaopeng.bxp@\*\*\*\*.com> 1378099160 +0800

Release version v2.2

### 14.2. 新建标签

git tag -a -m 如果是需要给某个历史提交节点打标签, 则加上对应hashcode即可

博客专栏



玩转Ubuntu  
文章：11篇  
阅读：58572



每天一个Linux命令  
文章：23篇  
阅读：177957

文章分类

- 玩转 Android开发 (26)
- 玩转 Android自动化 (17)
- 杂的文 (4)
- 玩转 Java (10)
- 玩转 AndroidStudio (31)
- 玩转 Android测试 (9)
- 玩转 mac OS (17)
- 玩转 Eclipse (6)
- 玩转 Git (14)
- 玩转 Maven (3)
- 玩转 工具 (6)
- 玩转 JavaScript (0)
- 玩转 Python (14)
- 玩转 Jenkins (5)
- 玩转 iOS 测试 (1)
- 玩转 Linux (11)
- 每天一个Linux命令 (26)
- 玩转 Ubuntu (13)
- 玩转 SQLite (5)
- 玩转 Docker (0)
- Markdown 实用指南 (5)

文章存档

- 2017年07月 (1)
- 2017年05月 (1)
- 2017年04月 (3)
- 2016年12月 (3)
- 2016年11月 (2)

展开

阅读排行

- Android Studio入门指南 (180438)
- Git入门指南十一:Git brar (120398)
- 【Git配置技巧】01. 配置 (92071)
- 【Android数据传递】Inte (70564)
- Git入门指南八:Git撤消操 (56661)
- 使用python编写appium月 (52673)
- python时间处理之datetin (48669)
- Android视频录制命令scn (46142)
- Android截图命令screenc (45621)
- 【Android测试工具】02. (41408)

评论排行

- Android Studio入门指南 (570)
- Android截图命令screenc (18)

```
#1. 新建一个tag
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git tag -a v2.2 -m "Release version v2.2"
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git tag -l
v2.2
#2. 把新建的分支push到远程
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git push origin v2.2
Counting objects: 1, done.
Writing objects: 100% (1/1), 169 bytes | 0 bytes/s, done.
Total 1 (delta 0), reused 0 (delta 0)
remote: To git@mirror.gitlab.****.com:/home/git/repositories/sospartan/spark.git
remote: * [new tag] v2.2 -> v2.2
To git@gitlab.****:sospartan/spark.git
* [new tag] v2.2 -> v2.2
# 扩展阅读
$ git push origin -tags # 将本地所有标签一次性提交到git服务器
```

14.3. 删除标签

git tag -d<name>

```
# 1. 删除本地tag v2.2
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git tag -d v2.2
已删除 tag 'v2.2' (曾为 78fa3cc)
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git tag -l
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$

# 2. 删除远程tag
bixiaopeng@bixiaopeng-To-be-filled-by-O-E-M:~/workspace2/spark$ git push origin :v2.2
remote: To git@mirror.gitlab.****.com:/home/git/repositories/sospartan/spark.git
remote: - [deleted] v2.2
To git@gitlab.****c.com:sospartan/spark.git
- [deleted] v2.2
```

值得注意的是，如果tag名字和分支名字一样的时候，push需要指定refs的详细路径，因为分支和tag在git内部是这样表示的（只是引用），tag只存储所指向的那一次提交：

refs/tags/{tagnane} refs/heads/branches/{branchnane}

所以要这样push相同名字的tag和分支

```
$: git push origin refs/tags/product
$: git push origin refs/heads/branches/product
```

14.4. 提交标签

git push origin <分支名><tag name>

14.5. 恢复代码到某个标签点

git show<tagname> 由此得到对应的提交哈希值，然后git reset 回退代码

微信公众帐号: wirelessqa



关于作者：

作者: 毕小朋 | 老毕 邮箱: wirelessqa.me@gmail.com