



链滴

git 【基本操作】

作者: [haxLook](#)

原文链接: <https://ld246.com/article/1617285578488>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Git 常用命令

git branch 查看本地所有分支

git status 查看当前状态

git commit 提交

git branch -a 查看所有的分支

git branch -r 查看本地所有分支

git commit -am "init" 提交并且加注释

git remote add origin git@192.168.1.119:ndshow

git push origin master 将文件给推到服务器上

git remote show origin 显示远程库origin里的资源

git push origin master:develop

git push origin master:hb-dev 将本地库与服务器上的库进行关联

git checkout --track origin/dev 切换到远程dev分支

git branch -D master develop 删除本地库develop

git checkout -b dev 建立一个新的本地分支dev

git merge origin/dev 将分支dev与当前分支进行合并

git checkout dev 切换到本地dev分支

git remote show 查看远程库

git add .

git rm 文件名(包括路径) 从git中删除指定文件

git clone [git://github.com/schacon/grit.git](https://link.jianshu.com?t=git%3A%2F%2Fgithub.com%2Fschacon%2Fgrit.git) 从服务器上将代码给拉下来

git clone -b develop [git://github.com/schacon/grit.git](https://link.jianshu.com?t=git%3A%2F%2Fgithub.com%2Fschacon%2Fgrit.git)
从服务器上将develop分支的代码给拉下来

git config --list 看所有用户

git ls-files 看已经被提交的

git rm [file name] 删除一个文件

git commit -a 提交当前repos的所有改变

git add [file name] 添加一个文件到git index

git commit -v 当你用 -v 参数的时候可以看commit的差异

git commit -m "This is the message describing the commit" 添加commit信息

git commit -a -a是代表add, 把所有的change加到git index里然后再commit

git commit -a -v 一般提交命令

git log 看你commit的日志

git diff 查看尚未暂存的更新

git rm a.a 移除文件(从暂存区和工作区中删除)

git rm --cached a.a 移除文件(只从暂存区中删除)

git commit -m "remove" 移除文件(从Git中删除)

git rm -f a.a 强行移除修改后文件(从暂存区和工作区中删除)

git diff --cached 或 \$ git diff --staged 查看尚未提交的更新

git stash push 将文件给push到一个临时空间中

git stash pop 将文件从临时空间pop下来

git remote add origin [git@github.com](https://link.jianshu.com?t=mailto%3Agit%40github.com):username/Hello-World.git

git push origin master 将本地项目给提交到服务器中

Git命令参考手册(文本版)

git init # 初始化本地git仓库 (创建新仓库)

git config --global user.name "xxx" # 配置用户名

git config --global [user.email](https://link.jianshu.com?t=http%3A%2F%2Fuser.email) "[xxx@xx.com](https://link.jianshu.com?t=mailto%3Axxx%40xxx.com)" # 配置邮件

git config --global color.ui true # git status等命令自动着色

git config --global color.status auto

git config --global color.diff auto

git config --global color.branch auto

git config --global color.interactive auto

```

git clone [git+ssh://git@192.168.53.168/VT.git](https://link.jianshu.com?t=git%2Bssh%3A%2
%2Fgit%40192.168.53.168%2FVT.git)      # clone远程仓库

git status                                # 查看当前版本状态 (是否修改)

git add xyz                               # 添加xyz文件至index

git add .                                  # 增加当前子目录下所有更改过的文件至index

git commit -m 'xxx'                       # 提交

git commit --amend -m 'xxx'              # 合并上一次提交 (用于反复修改)

git commit -am 'xxx'                     # 将add和commit合为一步

git rm xxx                                # 删除index中的文件

git rm -r *                               # 递归删除

git log                                    # 显示提交日志

git log -1                                # 显示1行日志 -n为n行

git log -5

git log --stat                            # 显示提交日志及相关变动文件

git log -p -m

git show dfb02e6e4f2f7b573337763e5c0013802e392818    # 显示某个提交的详细内容

git show dfb02                            # 可只用commitid的前几位

git show HEAD                             # 显示HEAD提交日志

git show HEAD^                            # 显示HEAD的父 (上一个版本) 的提交日志 ^^为
两个版本 ^5为上5个版本

git tag                                    # 显示已存在的tag

git tag -a v2.0 -m 'xxx'                  # 增加v2.0的tag

git show v2.0                             # 显示v2.0的日志及详细内容

git log v2.0                              # 显示v2.0的日志

git diff                                  # 显示所有未添加至index的变更

git diff --cached                          # 显示所有已添加index但还未commit的变更

git diff HEAD^                            # 比较与上一个版本的差异

git diff HEAD -- ./lib                    # 比较与HEAD版本lib目录的差异

```

git diff origin/master..master 的 # 比较远程分支master上有本地分支master上没的

git diff origin/master..master --stat # 只显示差异的文件，不显示具体内容

git remote add origin [git+ssh://git@192.168.53.168/VT.git](https://link.jianshu.com?t=git%2ssh%3A%2F%2Fgit%40192.168.53.168%2FVT.git) # 增加远程定义（用于push/pull/fetch）

git branch # 显示本地分支

git branch --contains 50089 # 显示包含提交50089的分支

git branch -a # 显示所有分支

git branch -r # 显示所有原创分支

git branch --merged # 显示所有已合并到当前分支的分支

git branch --no-merged # 显示所有未合并到当前分支的分支

git branch -m master master_copy # 本地分支改名

git checkout -b master_copy # 从当前分支创建新分支master_copy并检出

git checkout -b master master_copy # 上面的完整版

git checkout features/performance # 检出已存在的features/performance分支

git checkout --track hotfixes/BJVEP933 # 检出远程分支hotfixes/BJVEP933并创建本跟踪分支

git checkout v2.0 # 检出版本v2.0

git checkout -b devel origin/develop # 从远程分支develop创建新本地分支devel并出

git checkout -- README # 检出head版本的README文件（可用于修改错回退)

git merge origin/master # 合并远程master分支至当前分支

git cherry-pick ff44785404a8e # 合并提交ff44785404a8e的修改

git push origin master # 将当前分支push到远程master分支

git push origin :hotfixes/BJVEP933 # 删除远程仓库的hotfixes/BJVEP933分支

git push --tags # 把所有tag推送到远程仓库

git fetch # 获取所有远程分支（不更新本地分支，另需merge)

git fetch --prune # 获取所有原创分支并清除服务器上已删掉的分支

git pull origin master # 获取远程分支master并merge到当前分支

git mv README README2 # 重命名文件README为README2

git reset --hard HEAD # 将当前版本重置为HEAD (通常用于merge失败回)

git rebase

git branch -d hotfixes/BJVEP933 # 删除分支hotfixes/BJVEP933 (本分支修改已并到其他分支)

git branch -D hotfixes/BJVEP933 # 强制删除分支hotfixes/BJVEP933

git ls-files # 列出git index包含的文件

git show-branch # 图示当前分支历史

git show-branch --all # 图示所有分支历史

git whatchanged # 显示提交历史对应的文件修改

git revert dfb02e6e4f2f7b573337763e5c0013802e392818 # 撤销提交dfb02e6e4f2f7b57337763e5c0013802e392818

git ls-tree HEAD # 内部命令: 显示某个git对象

git rev-parse v2.0 # 内部命令: 显示某个ref对于的SHA1 HASH

git reflog # 显示所有提交, 包括孤立节点

git show HEAD@{5}

git show master@{yesterday} # 显示master分支昨天的状态

git log --pretty=format:'%h %s' --graph # 图示提交日志

git show HEAD~3

git show -s --pretty=raw 2be7fcb476

git stash # 暂存当前修改, 将所有至为HEAD状态

git stash list # 查看所有暂存

git stash show -p stash@{0} # 参考第一次暂存

git stash apply stash@{0} # 应用第一次暂存

git grep "delete from" # 文件中搜索文本 "delete from"

git grep -e '#define' --and -e SORT_DIRENT

git gc

git fsck