

GitHub Gist



guweigang / git_tutorial

Last active 21 hours ago

git命令大全

git_tutorial

```
1  git init                                # 初始化本地git仓库（创建新仓库）
2  git config --global user.name "xxx"    # 配置用户名
3  git config --global user.email "xxx@xxx.com" # 配置邮件
4  git config --global color.ui true      # git status等命令自动着色
5  git config --global color.status auto
6  git config --global color.diff auto
7  git config --global color.branch auto
8  git config --global color.interactive auto
9  git config --global --unset http.proxy # remove proxy configuration on git
10 git clone git+ssh://git@192.168.53.168/VT.git # clone远程仓库
11 git status                             # 查看当前版本状态（是否修改）
12 git add xyz                           # 添加xyz文件至index
13 git add .                             # 增加当前目录下所有更改过的文件至index
14 git commit -m 'xxx'                   # 提交
15 git commit --amend -m 'xxx'           # 合并上一次提交（用于反复修改）
16 git commit -am 'xxx'                  # 将add和commit合为一步
17 git rm xxx                           # 删除index中的文件
18 git rm -r *                           # 递归删除
19 git log                               # 显示提交日志
20 git log -1                             # 显示1行日志 -n为n行
21 git log -5
22 git log --stat                         # 显示提交日志及相关变动文件
23 git log -p -m
24 git show dfb02e6e4f2f7b573337763e5c0013802e392818 # 显示某个提交的详细内容
25 git show dfb02                         # 可只用commitid的前几位
26 git show HEAD                         # 显示HEAD提交日志
27 git show HEAD^                         # 显示HEAD的父（上一个版本）的提交日志 ^^为上两个版本 ^5为上5个版本
28 git tag                               # 显示已存在的tag
29 git tag -a v2.0 -m 'xxx'              # 增加v2.0的tag
30 git show v2.0                         # 显示v2.0的日志及详细内容
31 git log v2.0                           # 显示v2.0的日志
32 git diff                               # 显示所有未添加至index的变更
33 git diff --cached                      # 显示所有已添加index但还未commit的变更
34 git diff HEAD^                         # 比较与上一个版本的差异
35 git diff HEAD -- ./lib                # 比较与HEAD版本lib目录的差异
36 git diff origin/master..master        # 比较远程分支master上有本地分支master上没有的
37 git diff origin/master..master --stat # 只显示差异的文件，不显示具体内容
38 git remote add origin git+ssh://git@192.168.53.168/VT.git # 增加远程定义（用于push/pull/fetch）
39 git branch                             # 显示本地分支
40 git branch --contains 50089            # 显示包含提交50089的分支
41 git branch -a                          # 显示所有分支
42 git branch -r                          # 显示所有原创分支
43 git branch --merged                    # 显示所有已合并到当前分支的分支
44 git branch --no-merged                 # 显示所有未合并到当前分支的分支
45 git branch -m master master_copy      # 本地分支改名
46 git checkout -b master_copy            # 从当前分支创建新分支master_copy并检出
47 git checkout -b master master_copy    # 上面的完整版
48 git checkout features/performance     # 检出已存在的features/performance分支
49 git checkout --track hotfixes/BJVEP933 # 检出远程分支hotfixes/BJVEP933并创建本地跟踪分支
50 git checkout v2.0                      # 检出版本v2.0
51 git checkout -b devel origin/develop  # 从远程分支develop创建新本地分支devel并检出
52 git checkout -- README                 # 检出head版本的README文件（可用于修改错误回退）
53 git merge origin/master                # 合并远程master分支至当前分支
54 git cherry-pick ff44785404a8e         # 合并提交ff44785404a8e的修改
55 git push origin master                 # 将当前分支push到远程master分支
56 git push origin :hotfixes/BJVEP933    # 删除远程仓库的hotfixes/BJVEP933分支
57 git push --tags                       # 把所有tag推送到远程仓库
58 git fetch                             # 获取所有远程分支（不更新本地分支，另需merge）
59 git fetch --prune                      # 获取所有原创分支并清除服务器上已删掉的分支
60 git pull origin master                 # 获取远程分支master并merge到当前分支
61 git mv README README2                  # 重命名文件README为README2
62 git reset --hard HEAD                  # 将当前版本重置为HEAD（通常用于merge失败回退）
63 git rebase
64 git branch -d hotfixes/BJVEP933        # 删除分支hotfixes/BJVEP933（本分支修改已合并到其他分支）
65 git branch -D hotfixes/BJVEP933        # 强制删除分支hotfixes/BJVEP933
66 git ls-files                           # 列出git index包含的文件
67 git show-branch                       # 图示当前分支历史
68 git show-branch --all                  # 图示所有分支历史
69 git whatchanged                        # 显示提交历史对应的文件修改
```

70	git revert dfb02e6e4f2f7b573337763e5c0013802e392818	# 撤销提交dfb02e6e4f2f7b573337763e5c0013802e392818
71	git ls-tree HEAD	# 内部命令：显示某个git对象
72	git rev-parse v2.0	# 内部命令：显示某个ref对于的SHA1 HASH
73	git reflog	# 显示所有提交，包括孤立节点
74	git show HEAD@{5}	
75	git show master@{yesterday}	# 显示master分支昨天的状态
76	git log --pretty=format:'%h %s' --graph	# 图示提交日志
77	git show HEAD~3	
78	git show -s --pretty=raw 2be7fcb476	
79	git stash	# 暂存当前修改，将所有至为HEAD状态
80	git stash list	# 查看所有暂存
81	git stash show -p stash@{0}	# 参考第一次暂存
82	git stash apply stash@{0}	# 应用第一次暂存
83	git grep "delete from"	# 文件中搜索文本“delete from”
84	git grep -e '#define' --and -e SORT_DIRENT	
85	git gc	
86	git fsck	



xiaochunping commented on 13 Oct 2014

I just need, thank you



john-yuan commented on 20 Jan 2015

nice! 收藏了，谢谢分享！



08hjz commented on 16 Mar 2015

多谢



omeweb commented on 14 Apr 2015

good



zhkuo24 commented on 23 Apr 2015

nice



gamelife1314 commented on 21 May 2015

非常感谢



courri commented on 2 Jun 2015

thx!



chenliang613 commented on 31 Jul 2015

thanks for good sharing.



fruitsnoodle commented on 21 Aug 2015

謝謝！！



luinnx commented on 13 Oct 2015

well done

cracker8090 commented on 29 Oct 2015



thanks a lot!



beanlee commented on 6 Nov 2015

Thanks a lot ! forked :)



wujunze commented on 17 Nov 2015

很好



JZzhao commented on 18 Dec 2015

great, thks, dud



Wanrem commented on 28 Dec 2015

非常感谢!!!



TimeMosaic commented on 11 Jan

厉害了



akreturn commented on 21 Jan

very nice.ths



244200898 commented on 3 Feb

thank you



qiangzai commented 25 days ago

thank you very much!



imsandy commented 22 days ago

thanks



cousin1994 commented 18 days ago

thanks



songjie commented 13 days ago

猴塞雷



shijiaoliang commented 10 days ago

thks