

转载

吃小孩的果冻

于 2021-05-29 09:20:20 发布 93 收藏 1

版权

分类专栏:

学习笔记

文章标签:

git



学习笔记 专栏收录该内容

0 订阅

9 篇文章

订阅专栏

## 常用git命令

### 前言:

原链接: [生哥哥牛逼\(破音\)](#)

不争气的生哥哥链接404了, 害得我要重新整理

配置用户名和邮箱:

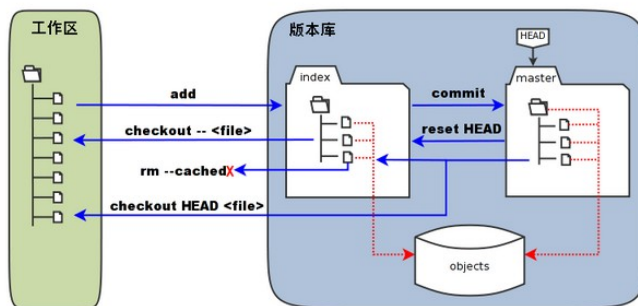
```
git config [--global] user.name "Coding-Ever"
```

```
git config [--global] user.email "qizhiyin3510@163.com"
```

### 一、概念:

#### 1.基本概念

- 工作区: 就是你在电脑里能看到的目录。
- 暂存区: 英文叫 stage 或 index。一般存放在 .git 目录下的 index 文件 (.git/index) 中, 所以我们把暂存区有时也叫作索引 (index)。
- 版本库: 工作区有一个隐藏目录 .git, 这个不算工作区, 而是 Git 的版本库。下面这个图展示了工作区、版本库中的暂存区和版本库之间的关系:



<https://blog.csdn.net/yibushao>

图中左侧为工作区, 右侧为版本库。在版本库中标记为“index”的区域是暂存区 (stage/index), 标记为“master”的是 master 分支所代表的目录树。

- 图中我们可以看出此时“HEAD”实际是指向 master 分支的一个“游标”。所以图示的命令中出现 HEAD 的地方可以用 master 来替换。
- 图中的 objects 标识的区域为 Git 的对象库, 实际位于“.git/objects”目录下, 里面包含了创建的各种对象及内容。
- 当对工作区修改 (或新增) 的文件执行 git add 命令时, 暂存区的目录树被更新, 同时工作区修改 (或新增) 的文件内容被写入到对象库中的一个新的对象中, 而该对象的ID被记录在暂存区的文件索引中。
- 当执行提交操作 (git commit) 时, 暂存区的目录树写到版本库 (对象库) 中, master 分支会做相应的更新。即 master 指向的目录树就是提交时暂存区的目录树。

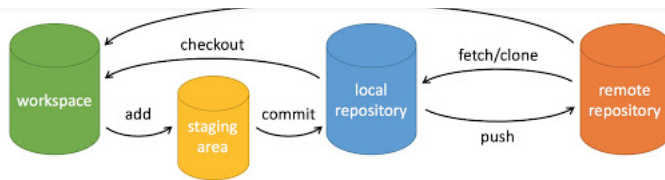
### 2.git操作示意图



吃小孩的果冻

关注





说明:

- workspace: 工作区
- staging area: 暂存区/缓存区
- local repository: 版本库或本地仓库
- remote repository: 远程仓库

## 二、命令:

### 1.基本操作:

#### 1. 状态查看

`git status` 查看工作区、暂存区状态

```
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   aa.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   a.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    bb.txt
CSDN @吃小孩的果冻
```

- 红色文件代表新建或者修改过的文件，仅存在于工作区，未被提交到版本库
- 绿色文件代表已被版本库管理，即存在于工作区和版本库中，未提交到本地仓库

#### 2. 添加

- `git add [file-name1] [file-name2] ...` 将工作区的“新建/修改”添加到暂存区
- `git add .` 将工作区的被修改的文件和新增的文件提交到暂存区，不包括被删除的文件
- `git add -A` A指all，将工作区被修改、被删除、新增的文件都提交到暂存区

#### 3. 提交

- `git commit -m [message]` 将暂存区所有文件添加到本地仓库
- `git commit [file-name-1] [file-name-2] -m [message]` 将暂存区指定文件添加到本地仓库
- `git commit --amend` 用于将本次提交和上次提交合并为一次提交，执行命令之后会进入vim编辑模式，用于编辑合并提交的message

#### 4. 推送

- `git push` 将文件添加到远程仓库
- `git push -f` 强制推送，当我们本地reset到旧的版本时，然后普通push会被拦截，因为回退之后本地HEAD指向比远程库还要旧，需要用强制推送才可以推送到远程仓库
- `git push origin [branch-name]` 推送当前本地分支到指定远程分支

#### 5. 查看历史记录

- `git log` 查看提交历史记录，显示如下图，会展示提交的详细历史记录，包括每次提交的全commit id，分:



```
Author: Jelly Bean <1439720833@qq.com>
Date: Sun Jan 2 21:43:00 2022 +0800
```

第一次提交的修改

`git log --pretty=oneline` , `git log --oneline` 这两个命令都是展示简略的提交历史记录, 这两个的区别在于前者是显示全commit id, 后者显示局部commit id

`git reflog` 查看操作历史记录, 显示如下图, 其中会包括 HEAD@{移动到当前版本需要多少步}, 此命令可以查看回退所丢失提交版本的commit id, 用于回退的找回

```
$ git reflog
ca7c1ee (HEAD -> master, origin/master, origin/HEAD) HEAD@{0}: commit (amend):
第一次提交的修改
6f0861f HEAD@{1}: commit: 第一次提交
352528c HEAD@{2}: clone: from https://gitee.com/yibushao/hello-git.git
```

## 6. 切换

`git checkout -- [file-name]`

用暂存区的文件覆盖掉工作区的文件

## 7. 回退

`git reset --参数 [commit id]` 将代码回退到指定版本处, 参数包括soft, mixed (推荐), hard. 若远程仓库中的代码版本比本地回退之后的版本新, 可用 `git push -f` 命令进行强制推送。

**-soft 参数:** 仅仅在本地库移动 HEAD 指针

**-mixed 参数:** 在本地库移动 HEAD 指针, 重置暂存区, 配合命令6可完成保留指定内容的保留和回退

**-hard 参数:** 在本地库移动 HEAD 指针, 重置暂存区, 重置工作区

使用hard参数回退所删除的代码可找回 (注: 删除前, 文件存在时的状态必须提交到了本地库)。具体操作见示例:

假设我现在的版本为1111111, 我进行了代码修改并提交了修改代码后的新版本为2222222, 然后我使用命令 `git reset --hard 1111111` 回退到了上个版本, 我现在突然不想回退了。

当前 `git reflog` 获取的操作记录如下:

1111111 HEAD@{0}: reset: moving to 1111111

2222222 HEAD@{1}: commit: 因回退删除的版本

1111111 HEAD@{2}: commit: 原版本

找回的操作步骤:

1. 首先使用 `git reflog` 命令获取操作记录, 查找到本次回退所误删的commit id为 222222 ,

2. 使用命令 `git reset --hard 222222` 即可找回这次回退而误删的代码

## 8. 比对

`git diff [file-name]` 将工作区中的文件和暂存区进行比较

`git diff [commit-id] [file-name]` 将工作区中的文件和暂存区进行比较

## 2、分支管理

### 1. `git fetch` 更新本地仓库

### 2. `git checkout -b 本地新分支名 origin/远程新分支名` 将远程的新分支拉到本地并创建一个新分支

### 3. 存放

`git stash` 将所有未提交的修改 (工作区和暂存区) 保存至堆栈中, 用于后续恢复当前工作目录

`git stash list` 查看当前stash中的内容

`git stash pop` 将当前stash中的内容弹出, 并应用到当前分支对应的工作目录上

`git stash apply` 将堆栈中的



吃小孩的果冻

关注



命令3一般用于

- 1.当前分支正在开发，需要临时切换到其他分支进行其他操作（比如bug修复）但是又不想commit的情况，可以用git stash将内容保存起来，切到bug修复分支操作后，再次切回开发分支恢复内容即可。
- 2.误在master上进行开发，需要将已开发的内容搬到dev分支上去，可以用git stash将内容保存起来，切到dev分支后，再次恢复内容即可。

### 三、其他学习git比较好的文章地址：

1. 廖雪峰Git教程
2. 菜鸟教程

#### 文章知识点与官方知识档案匹配，可进一步学习相关知识

CS入门技能树 Git入门 Git简介 1757 人正在系统学习中

#### Git 命令详解及常用命令整理

01-10

Git 命令详解及常用命令 Git作为常用的版本控制工具，多了解一些命令，将能省去很多时间，...

#### git常用命令详解

lansetudou的博客 317

本文不是教程大全，也不过多探讨底层原理，重点在于讲解常用命令，及其正确的使用方法。...

#### 参与评论

抢沙发



请发表有价值的评论，博客评论不欢迎灌水，良好的社区氛围需大家一起评论

#### git常用命令-详解版\_小武~~的博客\_git常用命令

3-24

git常用命令-详解版 原博客地址:https://blog.csdn.net/ithomer/article/details/7529022 一、Git ...

#### Git工作常用命令详解(工作当中很实用)\_java小童鞋的博客

4-28

Git工作常用命令详解(工作当中很实用) Git 1、git status详解:(太多了,不逐一演示,复制粘贴到...

#### Git常用命令(总结说明超详细)

念念不忘，必有回响 604

命令 作用 git init 创建本地库 git add filename 将文件添加到暂存区 git add . 添加所有文件到...

#### Git分布式版本控制工具

ayu6\_1的博客 185

Git分布式版本控制工具 1. Git概述 1.1 Git历史 Git 诞生于一个极富纷争大举创新的年代。Linu...

#### git常用命令全解\_程序员xiaoLe的博客

4-8

Git 常用命令速查 git branch 查看本地所有分支 git status 查看当前状态 git commit 提交 git br...

#### 【转】Git 常用命令详解 (二) ----不错

weixin\_33753003的博客 202

原文网址: http://blog.csdn.net/ithomer/article/details/7529022 Git 是一个很强大的分布式版...

#### Git 常用命令详解 (二)

编程开发资料库 3310

Git 是一个很强大的分布式版本管理工具，它不但适用于管理大型开源软件的源代码（如：linu...

#### Git 常用操作命令 (解析 + 案例)

zzvar的博客 1830

Git 操作命令详解 一、Git 概念 Git 是一个开源的分布式版本控制系统，用于敏捷高效地处理...

#### git常用命令 (补充)

86

Git常用操作命令： 1) 远程仓库相关命令 检出仓库：\$ git clone git://github.com/jquery/jquery....

#### git常用命令与详解

qq\_38915078的博客 146

1、git与SVN区别：说一下自己的理解，git是分布式版本管理系统，每个PC都是一个独立且...

#### git 常用操作命令

菜鸟\_zhengke的博客 1327

git clone git仓库地址克隆分支 git pull 更新代码 git add . (提交全部 注意add后面 有个点) git c...

#### Git常用命令 很全很详细

RFANI FAF的博客 37

转载出处: http://www.ruanyifeng.com/blc



吃小孩的果冻

关注

