

常用命令

命令	含义
mkdir learn git	创建learn git文件夹
cd learn git	进入learn git文件夹
pwd	查看路径
ls	查看当前目录内的文件名
git add <file>	添加多个文件
git add .	把当前仓库中所有最新修改的文件都提交到暂存区
git add -u	包含修改和删除的，但是不包含新增的
git add -A	同时具备.和-u 的特征
git status	查看工作区状态
git commit -m'备注'	提交到历史区（-m'备注'）
git log	查看历史（按小写q退出）
git log -3 --oneline	查看最近的3条历史
git log --oneline	查看历史（精简版）
git log --pretty=oneline	查看提交历史（全版-漂亮的一条线）
git reflog --pretty=oneline	查看命令历史
git diff	查看修改的内容（工作区，暂存区看不到）
git diff HEAD --<file>	查看工作区和版本库里面最新版本的差别（工作区和暂存区都能看到）
git reset --hard head^	回退上一个版本
git reset --hard head^^	回退上两个版本

命令	含义
git reset --hard head~n	回退上n个版本
git reset --hard 1094a	回退到某个版本id (几位数都可以, 不要太少, 避免相同版本)
git checkout -- <file>	工作区的修改全部撤销(暂存区有东西则回到添加暂存区后的状态, 没有则回到版本库一样的状态)
git reset HEAD <file>	撤销暂存区的修改, 重新放回工作区:
git push -u origin master	第一次推送master分支的所有内容 (加上-u可以把本地master和远程master关联)
git remote add origin 地址	关联远程库origin
git push origin master	推送最新修改
git branch	查看当前分支
git branch dev	创建分支dev
git branch -d dev	删除分支dev
git checkout dev	切换已有分支dev
git checkout -b dev	创建并且切换到新的分支 'dev'
git switch dev	切换已有分支dev
git switch -c dev	创建并且切换到新的分支 'dev'(新版本、更科学)
git merge dev	合并分支dev

更新git笔记

```
commit 86944e6834645470190a73fe2b16f1c
Author: dayu <540552972@qq.com>
Date: Thu Jan 9 20:46:02 2020 +0800
```

注意

遇到这种情况按 小q

```
$ git merge dev
Updating d46f35e..b17d20e
Fast-forward //快进模式, 直接改变指针
```

```
readme.txt | 1 +  
1 file changed, 1 insertion(+)
```

场景1：当你改乱了工作区某个文件的内容，想直接丢弃工作区的修改时，用命令`git checkout --file`。

场景2：当你不但改乱了工作区某个文件的内容，还添加到了暂存区时，想丢弃修改，分两步，第一步用命令`git reset HEAD <file>`，就回到了场景1，第二步按场景1操作。

场景3：已经提交了不合适的修改到版本库时，想要撤销本次提交，参考版本回退一节，不过前提是没有推送到远程库。