

# Git--cmd

---

## 常见命令：

## 合集：

```
#常见操作
#创建版本库(在本地文件夹创建版本库)
git init
#工作区的内容放入版本库的暂存区
git add readme.txt
#暂存区的内容提交到当前分支：
git commit -m "说明文字"
#拉取分支（远程拉取分支）
git pull origin master From https://github.com/ElvesYuki/hello-world
#推送分支
git push origin master
#推送其他分支
git push origin dev
```

## 分支操作

```
#创建分支dev,并切换到分支
git checkedout -b dev
#查看当前分支
git branch
#切换本地分支
git checkout <name>
#删除本地分支dev
git branch -d dev
#合并dev分支到当前分支(master)
git merge dev
```

## 文件操作

```
#git忽略某个指定的文件(不从版本库中删除)
git update-index --assume-unchanged config.conf
#git取消忽略某个指定的文件
git update-index --no-assume-unchanged config.conf
```

## 版本操作

```
#回退到上一个版本
git reset --hard HEAD^
#回退到上上一个版本
git reset --hard HEAD^^
#显示日志
git log
#回退到指定版本
git reset --hard commit_id(此处填的是id)
```

## 问题:

### 1、fatal: refusing to merge unrelated histories

(拒绝合并不相关的历史)

解决: 出现这个问题的最主要原因还是在于本地仓库和远程仓库实际上是独立的两个仓库。

将本地仓库与远程仓库 强制连接 使用命令:

```
git pull origin master --allow-unrelated-histories
```

### 2、.gitignore规则不生效

.gitignore只能忽略那些原来没有被track的文件, 如果某些文件已经被纳入了版本管理中, 则修改.gitignore是无效的。

解决方法就是先把本地缓存删除(改变成未track状态), 然后再提交:

```
git rm -r --cached .
git add .
git commit -m "update .gitignore"
```