

git 笔记

1. 工作区、缓存区和仓库区

- 工作区：当前编辑的文件所在区域
- 缓存区：存储准备提交到仓库区的文件列表, 记录文件的改动信息
- 仓库区：存储已经修改过的文件

2. 基本操作

- 新建代码仓库

```
git init [dir-name]

git init
```

- 配置

显示当前配置：

```
git config --list
```

编辑当前配置：

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

- 工作区和暂存区

添加文件到暂存区：

```
git add [file] [file2]
```

添加目录到暂存区：

```
git add [dir-name]
```

删除工作区文件：

```
git rm [file]
```

停止追踪但不删除文件：

```
git rm --cached [file]
```

撤回暂存区的文件

```
git reset HEAD file-name
```

- 暂存区和仓库区

暂存区提交到仓库区：

```
git commit [file1] [file2] -m "message"

-a: 提交工作区的所有改动直接到仓库区

-v 提交时显示所有的改动信息
```

- 分支

新建分支：

```
git branch [branch-name]
```

切换分支

```
git checkout branch-name
```

删除分支

```
git branch -d branch-name
```

合并到当前分支

```
git merge branch-name
```

- 查看信息

显示有变更的文件：

```
git status
```

显示commit的历史：

```
git log

--stat //显示每次commit发生变更的文件

-S key 根据关键字查找提交历史

--follow filename 显示文件的版本历史，包括文件的改名

-p file 显示文件的每一改动
```

显示改动文件的人和时间信息：

```
git blame file-name
```

显示暂存区和工作区的差异：

```
git diff

-cached file-name 显示暂存区和上一次commit的差异
```

显示工作区和当前分支最新commit之间的差异：

```
git diff HEAD
```

- 远程仓库的操作

显示远程仓库：

```
git remote -v
```

下载远程仓库的变动：

```
git fetch [remote]
```

拉取远程仓库的变化并与本地分支合并：

```
git pull origin branch-name

等于git fetch + git merge
```

上传本地分支到远程仓库

```
git push origin branch-name
```

上传所有分支：

```
git push --all
```