

Git 学习笔记

一、Git 概念

Git 是目前最先进的开源的分布式版本控制系统。

二、Git 与 Github

Git 是用于 Linux 内核开发的版本控制工具，采用了分布式版本库的方式，不必利用服务器端软件支持，可在本地建立仓库，存储代码的各个版本。

Github 是面向开源及私有软件项目的托管平台，只支持 git 作为惟一的版本库格式进行托管，是远程代码仓库、

三、Git 组成

三个区域：

1. 工作区：克隆项目到本地后，项目所在的文件夹目录；
2. 暂存区：用于存储工作区中添加的变更（新增、修改、删除）的地方；
3. 仓库：又可分为本地仓库与远程仓库，本地仓库用于存储本地工作区和暂存区提交上来的变更（新增、修改、删除），远程仓库用于在远端服务器上存储项目。

四、创建版本库

1. 设置用户名与邮箱

```
$ git config --global user.name "Name"
```

```
$ git config --global user.email "email"
```

2. 初始化

```
$ git init
```

3. 添加文件到版本库

```
$ git add 文件名
```

```
$ git commit -m "提交说明"
```

五、版本管理

1. 查看当前状态

```
$ git status
```

2. 查看修改内容

```
$ git diff 文件名
```

3. 查看提交历史

```
$ git log
```

4. 版本回退

```
$ git reset --hard HEAD^
```

5. 撤销修改

```
$ git checkout --文件名
```

6. 删除文件

```
$ git rm 文件名
```

六、分支管理

1. 查看分支

```
$ git branch
```

2. 创建分支

```
$ git branch <新建分支名>
```

3. 切换分支

```
$ git switch <分支名>
```

4. 删除分支

```
$ git branch -d<分支名>
```

5. 合并分支到当前分支

```
$ git merge <待合并分支名>
```

七、远程仓库

1. 查看远程仓库

```
$ git remote -v
```

2. 新建远程仓库

```
$ git remote add origin <url>
```

3. 拉取远程分支

```
$ git fetch origin master
```

4. 拉取远程分支并合并

```
$ git pull origin master
```

5. 推送到远程

```
$ git push origin dev
```

6. 删除远程仓库

```
$ git remote remove origin
```