

Git是目前世界上最先进的分布式版本控制系统（没有之一）。

- 集中式版本控制系统最大的毛病就是必须联网才能工作
- 分布式版本控制系统根本没有“中央服务器”，每个人的电脑上都是一个完整的版本库

所有的版本控制系统，其实只能跟踪文本文件的改动。

图片、视频这些二进制文件，没法跟踪文件的变化，s只知道图片从100KB改成了120KB，但到底改了啥，版本控制系统不知道，也没法知道。

## 安装git

```
sudo apt-get install git
```

安装完成后，配置用户名和邮箱：

```
$ git config --global user.name "Your Name"
$ git config --global user.email "email@example.com"
```

注意：git config 命令的 --global 参数，用了这个参数，表示你这台机器上所有的Git仓库都会使用这个配置。

## 创建版本库

repository，可以简单理解成一个目录，这个目录里面的所有文件都可以被Git管理起来，每个文件的修改、删除，Git都能跟踪，以便任何时刻都可以追踪历史，或者在将来某个时刻可以“还原”。

1. 选择一个合适的地方，创建一个空目录：

```
$ mkdir learngit
$ cd learngit
$ pwd
/Users/michael/learngit
```

2. git init 命令把这个目录变成Git可以管理的仓库：

```
$ git init
Initialized empty Git repository in /Users/michael/learngit/.git/
```

当前目录下多了一个.git的目录，这个目录是Git来跟踪管理版本库的，不要手动修改这个目录。

## 把文件添加进版本库

编写一个文本文件 例如 readme.md，放入 repository 目录下面。

1. 用命令 git add 告诉Git，把文件添加到仓库：

```
$ git add readme.txt
```

2. 用命令 git commit 告诉Git，把文件提交到仓库：

```
$ git commit -m "wrote a readme file"
```

`-m` 后面输入的是本次提交的说明，可以输入任意内容，

为什么Git添加文件需要`add`，`commit`一共两步呢？因为`commit`可以一次提交很多文件，所以你可以多次`add`不同的文件，比如：

```
$ git add file1.txt  
$ git add file2.txt file3.txt  
$ git commit -m "add 3 files."
```