



git

everything-is-local

Author: Shine

Version: 9.0.2

- 一、引言
- 二、介绍
- 三、Git安装
 - 3.1 下载Git
 - 3.2 安装
 - 3.3 基本配置
 - 3.4 测试
- 四、架构
- 五、仓库
 - 5.1 新建仓库
 - 5.2 工作区
 - 5.3 暂存区
 - 5.4 分支
- 六、基本操作
 - 6.1 查看仓库状态
 - 6.2 暂存文件
 - 6.3 提交文件
- 七、远程仓库
 - 7.1 远程仓库工作模式
 - 7.2 远程仓库选型
 - 7.3 基本操作
 - 7.3.1 注册git服务器账号
 - 7.3.2 新建远程仓库
 - 7.3.3 本地关联远程仓库
 - 7.3.4 推送文件到远程仓库
 - 7.3.5 克隆远程仓库
 - 7.3.6 代码共享
 - 7.3.7 命令汇总
- 八、分支

8.1 分支简介	
8.2 分支操作	
8.2.1 查看分支	
8.2.2 创建分支	
8.2.3 切换分支	
8.3 新建分支细节	
8.3.1 新分支初始内容	
8.3.2 多分支走向	
8.3.3 分支提交日志	
8.4 分支合并	
8.4.1 快速合并	
8.4.2 三方合并	
8.4.3 合并冲突	
8.4.3.1 冲突演示	
8.4.3.2 冲突解决	
九、Idea关联Git	
9.1 关联Git	
9.2 创建仓库	
9.3 提交-commit	
9.4 创建分支	
9.5 上传到远程仓库(push)	
9.6 复制到本地仓库(clone)	
9.7 更新本地项目	
9.8 冲突解决	
十、多人协同开发	
10.1 项目管理员(项目经理)	
10.2 开发人员	
十一、经典问题	

一、引言

- 在单人开发过程中，需要进行版本管理，以利于开发进度的控制。
- 在多人开发过程中，不仅需要**版本管理**，还需要进行**多人协同控制**。

二、介绍

- Git是一个**开源的分布式版本控制系统**，用于敏捷高效地处理任何或小或大的项目。
- Git 是 Linus Torvalds 为了帮助管理 Linux 内核开发而开发的一个开放源码的版本控制软件。
- 官网: <https://git-scm.com/>

三、Git安装

3.1 下载Git

下载Git <https://git-scm.com/downloads>

下载git



3.2 安装

安装，除了安装位置外，其他一直下一步即可

3.3 基本配置

- 安装后，打开git的cmd窗口，自报家门
- 如下信息会在提交代码时要使用，记录在你的每次提交中。以后才知道哪次提交是谁做的。

```
git config --global user.name "Your Name" #用户名
git config --global user.email "email@example.com" #邮箱
# 查看信息
git config -l
```

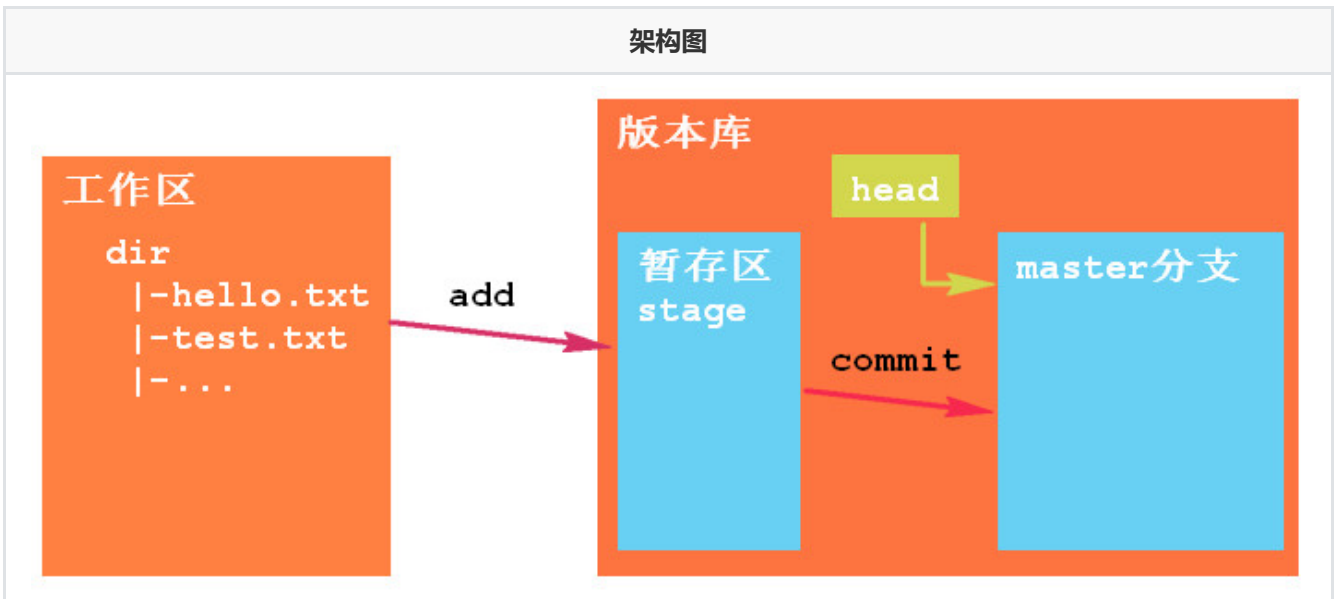
3.4 测试

测试：cmd中执行,查看git版本

```
git version
```

四、架构

- 版本库：工作区中有一个隐藏目录 `.git`，这个目录不属于工作区，而是git的 版本库，是git管理的所有内容
- 暂存区：版本库中包含一个临时区域，保存下一步要提交的文件。
- 分支：版本库中包含若干分支，提交的文件存储在分支中



五、仓库

对应的就是一个[目录](#)，这个目录中的所有文件被git管理起来。

以后会将一个[项目的根目录](#)，作为仓库。

仓库中的每个文件的改动 都由git跟踪。

5.1 新建仓库

选择一个目录，执行指令：[git init](#)

新建仓库	仓库目录
<pre>C:\Windows\System32\cmd.exe D:\repo1>git init Initialized empty Git repository in D:/repo1/.git/ D:\repo1></pre> <p>D:\repo1目录中，会新建一个.git目录 .git目录即仓库对应目录</p>	<p>电脑 > D盘 (D:) > repo1</p> <p>名称</p> <p>.git ← 仓库</p>

5.2 工作区

执行[git init](#)的目录即为工作区，如上例，[D:\repo1](#)目录即为工作区【不包含[.git](#)目录】

所有文件，都首先在工作区新建，然后可以存入仓库(版本库)，进行版本控制。

5.3 暂存区

暂存区也在 [.git](#) 目录内，工作区的文件进入仓库时，要先进入暂存区。

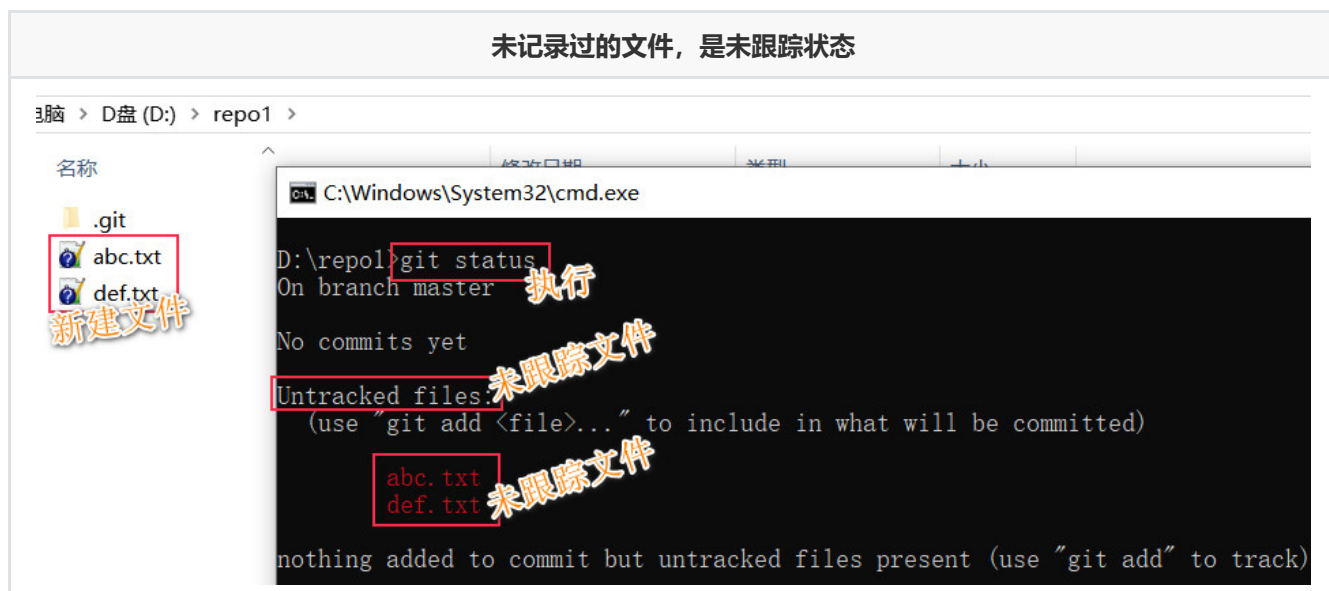
5.4 分支

版本控制，简单说，就是记录文件的诸多版本，分支就是这些版本的最终记录位置。

六、基本操作

6.1 查看仓库状态

执行 `git status` 可以看到工作区中文件的状态



6.2 暂存文件

执行 `git add`，将工作区中的文件全部存入暂存区



6.3 提交文件

执行 `git commit -m "这里写提交的描述信息"` 作用是将暂存区的文件存入分支，形成一个版本

提交文件，形成一个版本

窗 > D盘 (D:) > repo1

名称

修改日期

类型

大小

.git

abc.txt

def.txt

C:\Windows\System32\cmd.exe

提交文件

```
D:\repo1>git commit -m "first commit"
[master (root-commit) 762f568] first commit
2 files changed, 0 insertions(+), 0 deletion
create mode 100644 abc.txt
create mode 100644 def.txt
```

七、远程仓库

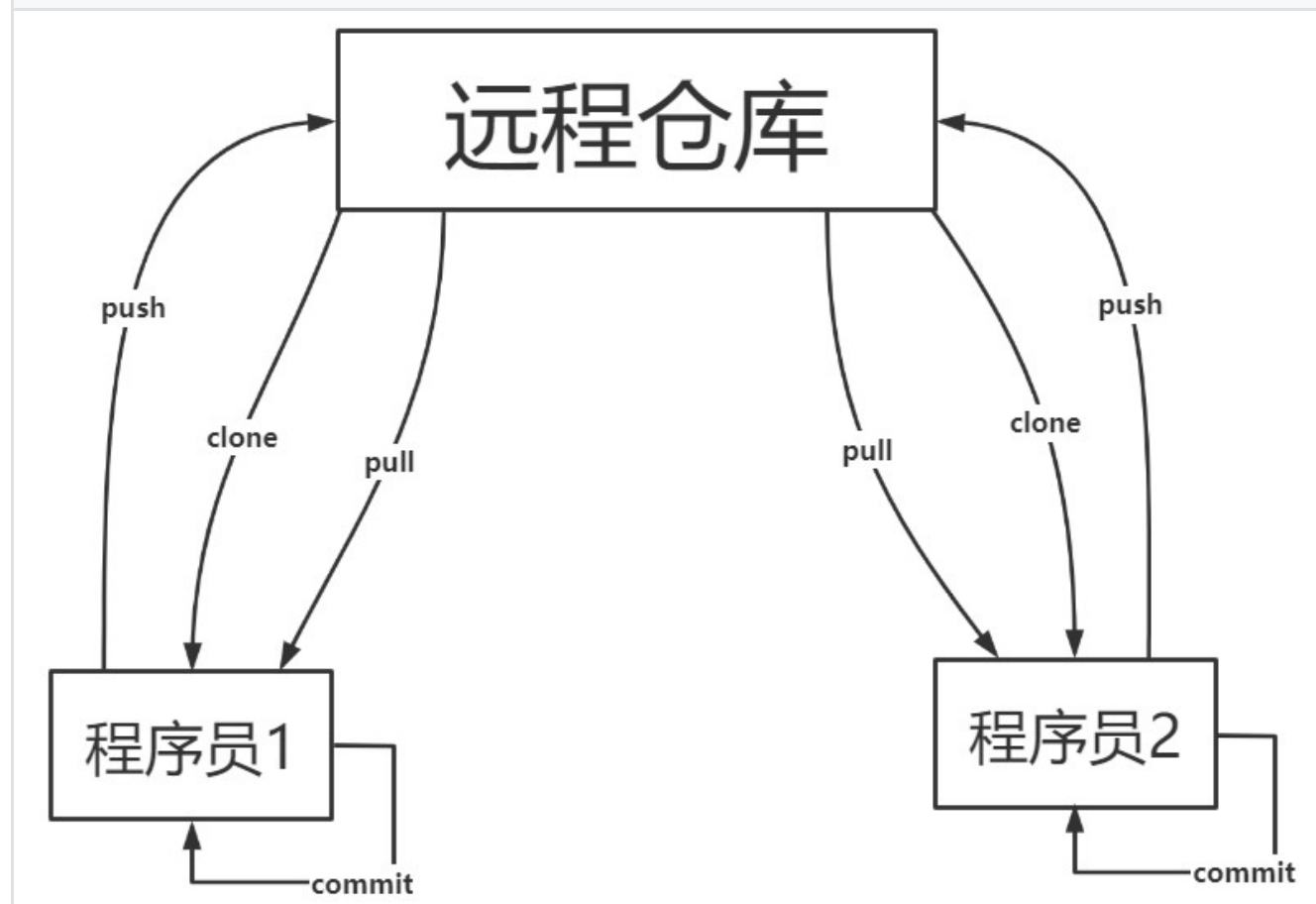
第5章中的仓库，其实是本地仓库。

当多人协同开发时，每人都在自己的本地仓库维护版本。

但很重要的一点是，多人之间需要共享代码、合并代码，此时就需要一个[远程仓库](#)。

7.1 远程仓库工作模式

远程仓库工作模式



7.2 远程仓库选型

- 有很多远程仓库可以选择，比如 github(<https://github.com/>),码云(<https://gitee.com/>);
此两种可以注册自己测试使用，但如果是商业项目，需要更多支持需要付费。
- 公司内部也可以有自己构建的远程仓库(http://qianfeng.qfjava.cn:8087/users/sign_in)。

7.3 基本操作

每个开发人员，在面对远程仓库时，会面临的一些基本操作。

7.3.1 注册git服务器账号

在 [码云](#) 注册账号，并登录。

进入公司后，很可能会使用公司自己搭建的git服务器，则账号向领导索要即可

点击注册，完成注册过程



开源软件

企业版

特惠

高校版

博客

搜索项目

登录

注册

登录后显示主页



开源软件

企业版

特惠

高校版

博客

我的码云



shine

用户名

个人主页



动态

7.3.2 新建远程仓库

创建远程私有仓库-01



gitee

开源软件

企业版

特惠

还没有组织，[立即创建](#)

仓库

公开的

私有的

私有，不是所有人可见

添加仓库





开源软件

企业版

特惠

高校版

博客

我的码云

搜索项目

新建仓库

仓库名称

仓库名称

git_shine

归属

路径

s shine

/ git_shine

仓库地址: https://gitee.com/shine10/git_shine

仓库介绍 非必填

用简短的语言来描述一下吧

是否开源

☒ 私有☐ 公开

私有仓库的非仓库成员无法访问该仓库的代码和其他任何形式的资源

私有仓库最多支持 5 人协作 (如拥有多个私有仓库, 所有协作人数总计不得超过 5 人)

企业仓库, 更适合使用码云企业版, [了解更多 >>](#)

选择语言

添加 .gitignore

请选择语言

请选择 .gitignore 模板

☐ 使用Readme文件初始化这个仓库☐ 使用Issue模板文件初始化这个仓库 ⓘ☐ 使用Pull Request模板文件初始化这个仓库 ⓘ

选择分支模型 (仓库初始化后将根据所选分支模型创建分支)

单分支模型 (只创建 master 分支)

[导入已有仓库](#)

创建仓库

创建

创建远程私有仓库-03

shine / git_shine Unwatch 1 Star 0

用户名/仓库名

代码 Issues Pull Requests 附件 Wiki DevOps 服务 管理

快速设置— 如果你知道该怎么操作, 直接使用下面的地址

HTTPS SSH `https://gitee.com/shine10/git_shine.git`

仓库地址

我们强烈建议所有的git仓库都有 README, LICENSE, .gitignore 文件

Git入门? 查看 帮助, Visual Studio / TortoiseGit / Eclipse / Xcode 下如何连接本站, 如何导入仓库

简易的命令行入门教程:

Git 全局设置:

```
git config --global user.name "shine"
git config --global user.email "shine_0214@163.com"
```

创建 git 仓库:

```
mkdir git_shine
cd git_shine
git init
touch README.md
git add README.md
git commit -m "first commit"
git remote add origin https://gitee.com/shine10/git_shine.git
git push -u origin master
```

已有仓库?

```
cd existing_git_repo
git remote add origin https://gitee.com/shine10/git_shine.git
git push -u origin master
```

有此提示, 说明是裸库

7.3.3 本地关联远程仓库

本文档中选用仓库的 https协议的地址, 将此地址关联到本地git中

如此后, 本地即可用“origin”代指远程仓库

C:\WINDOWS\system32\cmd.exe

```
nothing to commit, working tree clean
D:\repol>git remote add origin https://gitee.com/shine10/git_shine.git
D:\repol>git remote -v
origin https://gitee.com/shine10/git_shine.git (fetch)
origin https://gitee.com/shine10/git_shine.git (push)
```

关联远程仓库地址

查看远程仓库地址

7.3.4 推送文件到远程仓库

将本地仓库中已经commit的内容push到远程仓库, 以共享自己的代码。

push

C:\WINDOWS\system32\cmd.exe

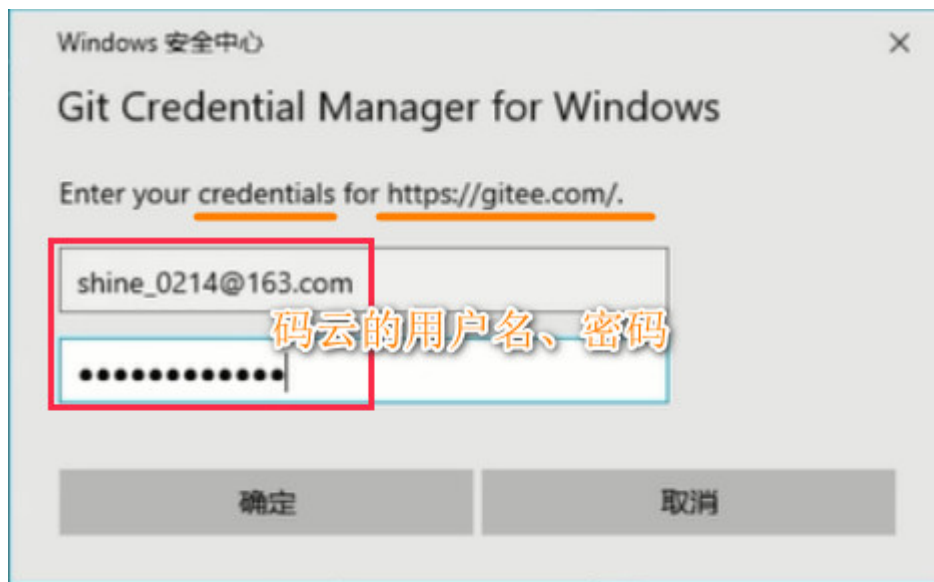
```
D:\repol>git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta 1, 381 bytes | 381.00 KiB/s, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 381 bytes | 381.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Powered by GITEE.COM [GNK-5.0]
To https://gitee.com/shine10/git_shine.git
 * [new branch]      master -> master
D:\repol>
```

将本地master分支，上传到远程master分支

上传时会有弹窗，需要输入码云的账号和密码

本地master > 远程master

输入凭证，如此码云才能接受此次上传的内容



7.3.5 克隆远程仓库

如果仓库已经由别人创建完毕，我们需要其中的内容，则可以通过 [git clone](#) 将其复制到本地。