

Git 教程

作者：人工智能001贾明达

一、Github和Gitee简介

什么是Github?

Github 是一个面向开源及私有软件项目的托管平台，因为只支持 Git 作为唯一的版本库格式进行托管，故名 Github。

GitHub 于 2008 年 4 月 10 日正式上线，除了 Git 代码仓库托管及基本的 Web 管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。目前，其托管版本数量非常之多，而且其中有很多的开源项目。

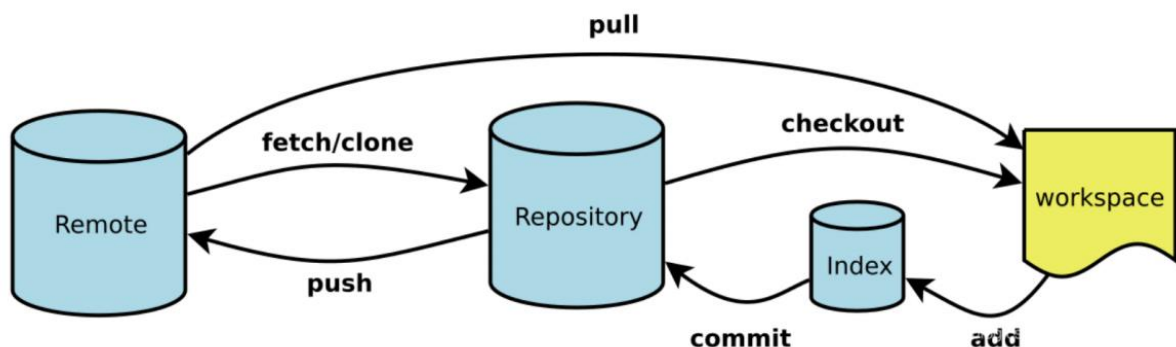


Git是一个 **分布式** 的**版本控制系统**，每一份拷贝出的仓库都可以独立使用，任何两个仓库之间不一致的地方都可以进行合并。

Github可以托管各种Git库，并提供一个Web界面。

Github的优点在于项目分支管理的简易性，贡献代码十分方便，通过站点的 **Fork** 之后进行代码的修改更新，然后再通过 **pull request** 机制向管理员申请合并，即可完成代码的统一。

Github同样是一个资源丰富的开源社区，在Github里有你想了解的几乎任何编程相关的知识，不过需要一丢丢的英语能力。



什么是Gitee?

可以理解为国内版的Github

为什么用Gitee而不是Github?

- 1.github一般都需要挂梯子，也就是科学上网才能快速访问（现在传播梯子的犯法的）。
- 2.gitee毕竟是国内的东西，github是国外的东西，可能说不用不了就会用不了（参考俄罗斯）。
- 3.gitee目前也非常成熟了，而且全面支持中文，入门门槛相当ok

二、Gitee的简单实例

git安装与卸载

```
apt-get install git
apt-get remove git
```

git配置

配置用户名

```
git config --global user.name "your name"
```

配置邮箱

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

查看配置信息

```
git config --global --list
```

生成公钥

```
ssh-keygen -t rsa -C "your email"
```

生成公钥后在ssh目录下，就是 `~/.ssh/id_rsa.pub` 中的内容拷贝至码云

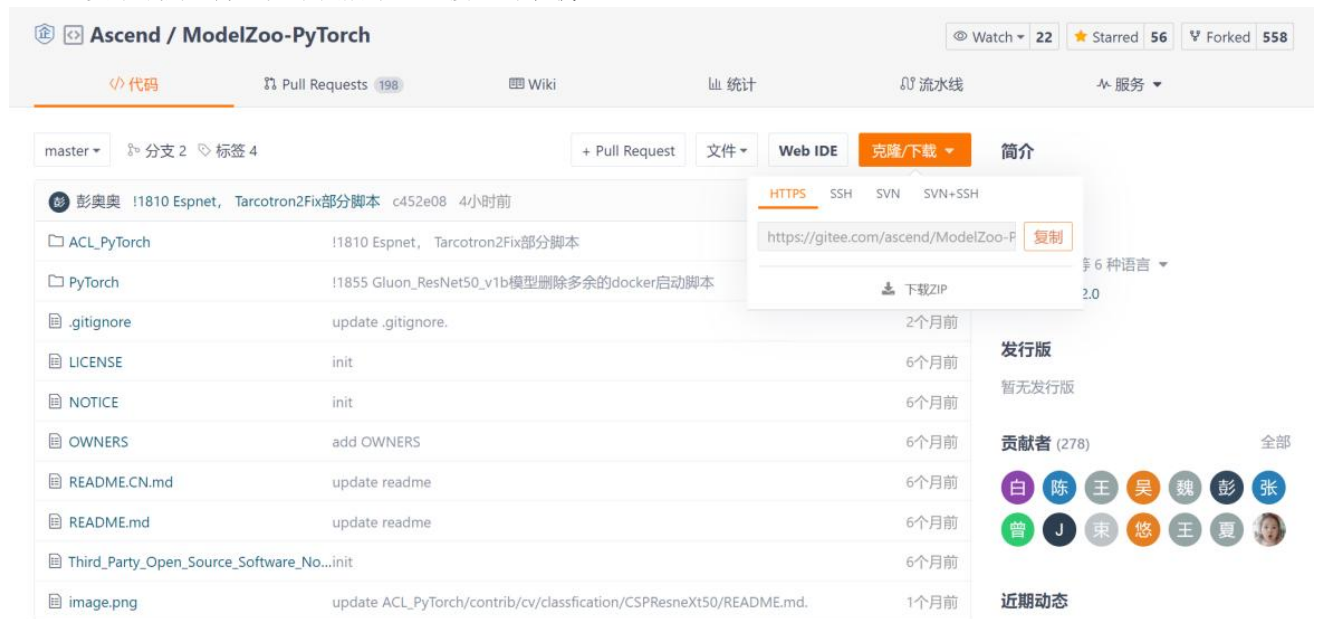


测试配置是否成功

```
ssh -T git@gitee.com
```

git下载代码

进入想要下载的仓库，如下图所示，直接点击**下载ZIP**



或者用命令行操作（随便举个栗子）

```
wget https://gitee.com/undark/vision-code-reading.git
```

之后就可以使用仓库了

git创建仓库

创建一个新仓库，全部步骤放一起

```
# 创建文件夹
mkdir my_git
cd my_git

# 仓库初始化
git init

# 与远程仓库创建链接，需要查看自己仓库的地址
git remote add origin xxx.git

# 创建并写入文件
touch test.py
vim test.py

# 将该目录下所有文件添加到 暂存区
git add .

# 将更改内容和日志消息一起存储在新的提交中
git commit -m "first commit"

# 将代码推至远程仓库
git push -u origin master
```

三、git分支管理

分支的作用

简单来说，就是进行仓库的版本管理和版本记录

master

从项目创建之初，有且唯一的分支就是主分支。如果之后再创建分支，就是一个一个的从分支，主分支被叫做 **master**

Git 的 **master** 分支并不是一个特殊分支。它就跟其它分支完全没有区别。之所以几乎每一个仓库都有 **master** 分支，是因为 `git init` 命令默认创建它，并且大多数人都懒得去改动它。

HEAD

对一个项目提交了很多个分支，如果有两个分支指向相同提交历史，Git 又是怎么知道当前在哪个分支上呢？

很简单，它有一个名为 **HEAD** 的特殊指针，指向当前所在的本地分支（译注：将 **HEAD** 想象为当前分支的别名）。

切换分支的本质就是移动 **HEAD** 指针



基本操作

1.查看分支

```
git branch -v
```

2.创建分支

```
git branch 分支名
```

3.切换分支

```
git checkout 分支名
```

4.合并分支 （在某一支支下，对另外的分支做合并）

```
git merge 分支名
```