

Gitee 入门指南

Gitee 是开源中国社区 2013 年推出的基于 Git 的代码托管服务，目前已经成为国内知名的代码托管平台，致力于为国内开发者提供优质稳定的托管服务。Gitee 除了提供最基础的 Git 代码托管之外，还提供代码在线查看、历史版本查看、Fork、Pull Request、打包下载任意版本、Issue、Wiki、保护分支、代码质量检测、PaaS 项目演示等方便管理、开发、协作、共享的功能。

一、注册 gitee 账号

个人开发者可以通过访问 <https://gitee.com/>，从首页右上角点击「注册」或点击「加入 Gitee」即可注册个人账号。



在 Gitee 的注册界面依次填入各项，需注意的是：邮箱建议填写国内邮箱如 163 邮箱/QQ 邮箱，以免因为众所周知的原因无法接收激活邮件。当然，也可以通过微信等第三方登录平台授权登录，授权登录后按照引导提示填写相关信息即可完成注册。

二、创建仓库

在注册完成并成功登录 Gitee 账号后，可以开始创建自己的第一个仓库。

1、通过网站右上角的「+」号，选择「新建仓库」，进入新建仓库页面。



2、在新建仓库页面填写仓库信息。

新建仓库

在其他网站已经有仓库了吗? [点击导入](#)

仓库名称 * ✓

repo

归属

路径 * ✓

Z Zachary

/

repo

仓库地址: <https://gitee.com/Sunzhkt/repo>

仓库介绍 21/200

This is a repository.

☐ 开源 (所有人可见) ?

☒ 私有 (仅仓库成员可见)

☐ 企业内部开源 (仅企业成员可见) ?

☐ 初始化仓库 (设置语言、.gitignore、开源许可证)

☐ 设置模板 (添加 README、Issue、Pull Request 模板文件)

☐ 选择分支模型 (仓库创建后将根据所选模型创建分支)

创建

仓库相关概念说明如下：

- 仓库名称：仓库的名称，用于仓库命名。
- 归属：仓库归属账户，可以是个人账号/组织/企业中的一种，创建成功后该账户默认为仓库的拥有者（管理员）。
- 路径：仓库的 git 访问路径，由个人空间地址+仓库路径名称组成。创建仓库后用户将通过该路径访问仓库。
- 仓库介绍：仓库的简单介绍。
- 是否开源：设置仓库是否为公开仓库，公开仓库对所有人可见，私有仓库仅限仓库成员可见（**注意：由于空仓库不允许设置为公开，需向仓库提交代码后通过仓库「管理」修改为公开**）。

3、填写完仓库的相关信息后点击「创建」，即可在 Gitee 上创建你的第一个仓库。

三、提交代码

在创建完仓库之后，可以通过如下方式，可以向仓库提交第一行代码。

此处我们以名为 repo 的仓库为例。对应的仓库地址为：<https://gitee.com/用>

户个性地址/repo.git，在实际实践中，请将用户个性地址替换为自己的地址。

方法 1、从本地上传文件

1、点击「初始化 readme 文件」。



2、点击「文件」选择「上传文件」。



3、拖拽文件或点击上传文件。

The screenshot shows the Gitee web interface for a repository named 'Zachary / repo'. The interface is in Chinese. At the top, there are navigation links for '代码' (Code), 'Issues', 'Pull Requests', 'Wiki', '统计' (Statistics), '流水线' (Pipeline), '服务' (Services), and '管理' (Management). The main section is titled '上传文件' (Upload File). It shows a file named 'test.cpp' (0.2 KB) being uploaded. Below the file upload area, there is a '提交信息' (Commit Message) section with the text 'test commit'. There is also an '扩展信息' (Extended Information) section for additional commit details. At the bottom, there is a '目标分支' (Target Branch) section set to 'master' and a '提交' (Commit) button.

4、填写提交信息后点击「提交」。

方法 2、先将仓库 clone 到本地，修改后再 push 到 Gitee 的仓库

```
$ git clone https://gitee.com/用户个性地址/repo.git #将远程仓库克隆到本地
```

在克隆过程中，如果仓库是一个私有仓库，将会要求用户输入 Gitee 的账号和密码。按照提示输入即可。

当然，用户也可以通过配置本地的 git 配置信息，执行 `git config` 命令预先配置好相关的用户信息，配置执行如下：

```
$ git config --global user.name "你的名字或昵称"
$ git config --global user.email "你的邮箱"
```

在 Gitee 平台中，强烈建议用户在【设置-多邮箱管理】中的“提交邮箱”与上面配置信息中的邮箱地址保持一致，这样平台就能及时地统计用户在平台中提交代码的贡献了。

修改代码后，在仓库目录下执行下面命令

```
$ git add . #将当前目录所有文件添加到 git 暂存区
$ git commit -m "my first commit" #提交并备注提交信息
$ git push origin master #将本地提交推送到远程仓库
```

方法3、本地初始化一个仓库，设置远程仓库地址后再做 push

和方法2的差别，在于先创建仓库。

```
$ git init
$ git remote add origin https://gitee.com/用户个性地址/repo.git
```

这样就完成了版本的一次初始化。

接下来，进入用户已经初始化好的或者克隆仓库的目录，然后执行：

```
$ git pull origin master
```

修改/添加文件，否则与原文件相比就没有变动。

```
$ git add .
$ git commit -m "第一次提交"
$ git push origin master
```

然后如果需要账号密码的话就输入账号密码，这样就完成了一次提交。此时，用户可以在个人面板、仓库主页查看到你的提交记录。

在新建仓库时，如果在 Gitee 平台仓库上已经存在 readme 或其他文件，在提交时可能会存在冲突，这时你需要选择的是保留线上的文件或者舍弃线上的文件，如果需要舍弃线上的文件，则在推送时选择强制推送，强制推送需要执行下面的命令(默认不推荐该行为)：

```
$ git push origin master -f
```

如果选择保留线上的 readme 文件,则需要先执行：

```
$ git pull origin master
```

四、将仓库设置为公开

在仓库「管理」的「是否开源」选项中，选择「开源」，并点击「保存」。

Zachary / repo

Watch

代码

Issues 0

Pull Requests 0

Wiki

统计

流水线

服务

管理

仓库设置

仓库成员管理

部署公钥管理

仓库挂件

WebHooks

只读文件管理

保护分支设置

推送规则设置

仓库设置

基本信息

功能设置

仓库状态

转移仓库

清空仓库

存储库 GC

删除仓库

代码审查设置

仓库镜像管理 限时开放

基本信息

仓库名称 *

路径 *

介绍

主页

语言

默认分支

是否开源

repo

https://gitee.com/Sunzhkt/

repo

This is a repository.

输入一个有效的 http 链接

请选择语言

master

开源 (所有人可见)

私有 (仅仓库成员可见)

为保障你的合法权益，请点此 选择 合适的开源许可证

保存