

I. Installation with Docker

1. Create a docker-compose.yml

The file sequentially lists the network gitea, the gitea service gitea-server, and the database service gitea-mysql-server.

```
PS D:\apps_data\docker\git-lib> docker-compose up -d
time="2025-12-07T21:51:39+08:00" level=warning msg="D:\apps_data\docker\git-lib\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 18/18
  ✓ server Pulled
    ✓ f84edd1c8d8b Pull complete 16.2s
    ✓ edff09febfa9 Pull complete 9.0s
    ✓ 2d35ebdb57d9 Pull complete 8.9s
    ✓ 510f786cf94b Pull complete 5.0s
    ✓ ceee20f5a0c8 Pull complete 4.1s
    ✓ 712adf69e56e Pull complete 3.7s
  ✓ db Pulled
    ✓ 7a5e1e917526 Pull complete 28.4s
    ✓ 57bd38a2d740 Pull complete 10.7s
    ✓ 782c57ddb1a7 Pull complete 3.1s
    ✓ f13e37513b38 Pull complete 10.9s
    ✓ 9e2f31baa56a Pull complete 21.6s
    ✓ 17ff14fd324e Pull complete 21.6s
    ✓ e84ca20366cc Pull complete 2.7s
    ✓ fea5342c4477 Pull complete 2.7s
    ✓ 54c71e5d2da5 Pull complete 11.7s
    ✓ 24b65e61bf70 Pull complete 2.7s
[+] Running 3/3
  ✓ Network git-lib_gitea Created 0.0s
  ✓ Container gitea-mysql-server Started 1.2s
  ✓ Container gitea-server Started 1.0s
```

2. Start up

Open PowerShell and enter the directory that docker-compose.yml belongs to. Enter the following command: **docker-compose up -d**.

```
PS D:\apps_data\docker\git-lib> docker-compose ps
time="2025-12-07T21:53:55+08:00" level=warning msg="D:\apps_data\docker\git-lib\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME                IMAGE                COMMAND                SERVICE    CREATED        STATUS        P
gitea-mysql-server  mysql:8             "docker-entrypoint.s..." db          About a minute ago Up About a minute 0.
0.0.0:3307->3306/tcp, [::]:3307->3306/tcp, 0.0.0.0:33070->33060/tcp, [::]:33070->33060/tcp
gitea-server        gitea/gitea:latest  "/usr/bin/entrypoint..." server      About a minute ago Up About a minute 0.
0.0.0:3000->3000/tcp, [::]:3000->3000/tcp, 0.0.0.0:2222->22/tcp, [::]:2222->22/tcp
```

Note: the gitea and mysql:8 mirrors are pulled to your local machine in advance. If they are not pulled in advance, the information you see will show the additional pulling process.

3. Access

Once the above service starts successfully, you can access Gitea. By default, you can access it using localhost:3000 or your local IP address:3000.

- Database setting: Gitea supports various database types including MySQL, PostgreSQL, MSSQL, and SQLite3. This practice uses MySQL, so MySQL is selected. Other configuration values should remain consistent with the database configuration in 'docker-compose.yml'.

Note: The database host configuration 'db:3306' means that when gitea-server accesses data, it will directly access the data via network port 3306, without needing to access it through port 3307 exposed by the database service.

安装页面 - Gitea: Git with a cup of tea

localhost:3000

初始配置

如果您正在使用 Docker 容器运行 Gitea，请务必先仔细阅读 [官方文档](#) 后再对本页面进行填写。

数据库设置

Gitea 需要使用 MySQL、PostgreSQL、MSSQL、SQLite3 或 TiDB (MySQL协议) 等数据库

数据库类型 * MySQL

数据库主机 * db:3306

用户名 * gitea

数据库用户密码 *

数据库名称 * gitea

- General Settings

General settings are used to configure information related to the Gitea server container.
The configurable items are shown below.

一般设置

站点名称 * Gitea: Git with a cup of tea
您可以在这里输入您公司的名称。

仓库根目录 * /data/git/repositories
所有远程 Git 仓库将保存到此目录。

LFS根目录 /data/git/lfs
存储为Git LFS的文件将被存储在此目录。留空禁用LFS

以用户名运行 * git
输入 Gitea 运行的操作系统用户名。请注意，
此用户必须具有对仓库根路径的访问权限。

4. Does Gitea support LFS? Prove it.

Version: 1.25.2

Git LFS setup

To use Gitea's built-in LFS support, you must update the `app.ini` file:

```
[server]
; Enables git-Lfs support. true or false, default is false.
LFS_START_SERVER = true

[lfs]
; Where your Lfs files reside, default is data/Lfs.
PATH = /home/gitea/data/lfs
```

NOTE

LFS server support needs at least Git v2.1.2 installed on the server

Git LFS Pure SSH protocol

The LFS Pure SSH protocol supports making LFS connections purely over SSH (without having to expose an HTTP endpoint for the Gitea server). Support for it can be enabled with the config option

`server.LFS_ALLOW_PURE_SSH`:

```
[server]
LFS_ALLOW_PURE_SSH = true
```

LFS根目录

存储为Git LFS的文件将被存储在此目录。留空禁用LFS

5. Create a repo with a large file (1G+)

Here Git is needed. I installed it according to https://blog.csdn.net/weixin_73435127/article/details/155746513.

Create a repo named as large-file-test

 工单 合并请求 里程碑 探索   + 

创建仓库

代码仓库包含了所有的项目文件，包括版本历史记录。已经在其他地方托管了？[迁移仓库](#)。

所有者 *

 gitea

由于最大仓库数量限制，一些组织可能不会显示在下拉列表中。

仓库名称 *

large-file-test

理想的仓库名称应由简短、有意义和独特的关键词组成。[.profile] 和 [.profile-private] 可用于为用户/组织添加 README.md。

可见性

☒ 将仓库设为私有

只有组织所有人或拥有权利的组织成员才能看到。

描述

输入简要描述 (可选)

In a local Gitea environment, a 1GB file was successfully created, committed, and pushed using Git LFS. Git only manages pointers, while LFS is responsible for storing and transferring the actual large file.


Command:

```
git clone http://localhost:3000/gitea/large-file-test.git
cd large-file-test
git lfs track "*.dat"
git add .gitattributes
fsutil file createnew bigfile.dat 1073741824
git add bigfile.dat
git commit -m "Add 1GB large test file"
git push
```

```

PS D:\apps-data\docker\git-lib> git clone http://localhost:3000/gitea/large-file-test.git
Cloning into 'large-file-test'...
warning: use of unencrypted HTTP remote URLs is not recommended; see https://aka.ms/gcm/unsafere
otes for more information.
warning: You appear to have cloned an empty repository.
PS D:\apps-data\docker\git-lib> cd large-file-test
PS D:\apps-data\docker\git-lib\large-file-test> git lfs track "*.dat"
Tracking "*.dat"
PS D:\apps-data\docker\git-lib\large-file-test> git add .gitattributes
PS D:\apps-data\docker\git-lib\large-file-test> fsutil file createnew bigfile.dat 1073741824
已创建文件 D:\apps-data\docker\git-lib\large-file-test\bigfile.dat
PS D:\apps-data\docker\git-lib\large-file-test> git add bigfile.dat
PS D:\apps-data\docker\git-lib\large-file-test> git commit -m "Add 1GB large test file"
[main (root-commit) c505342] Add 1GB large test file
 2 files changed, 4 insertions(+)
 create mode 100644 .gitattributes
 create mode 100644 bigfile.dat
PS D:\apps-data\docker\git-lib\large-file-test> git push
Locking support detected on remote "origin". Consider enabling it with:
  $ git config lfs.http://localhost:3000/gitea/large-file-test.git/info/lfs.locksverify true
Uploading LFS objects: 100% (1/1), 1.1 GB | 78 MB/s, done.
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 418 bytes | 418.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote: . Processing 1 references
remote: Processed 1 references in total
To http://localhost:3000/gitea/large-file-test.git
 * [new branch]      main -> main




```


gitea / large-file-test
私有库
📡
👁️ 取

<> 代码
🕒 工单
🔗 合并请求
🔄 工作流
📦 软件包
📁 项目

🕒 1 提交
🔗 1 分支
🏷️ 0 Git 标签

🔗 main ▾
🔗
转到文件
添加文件 ▾
<> 代码 ▾

 Agoni-Y	c505342cdd Add 1GB large test file	2分钟前
 .gitattributes	Add 1GB large test file	2分钟前
 bigfile.dat	Add 1GB large test file	2分钟前

6. Backup all data to an external harddrive/USB stick and restore it from another computer.

Copy D:\apps_data\docker\git-lib\ to a USB drive, then paste it to another computer (using the same directory in this case) and save it as D:\apps_data\docker\git-lib\. Open the directory in cmd and execute D:\apps_data\docker\git-lib\. Then gitea will restore the user, restore all repos, restore LFS files, etc.

Install Gitea without Docker

First, download gitea-1.23.6-windows-4.0-amd64.exe from <https://dl.gitea.com/gitea/1.23.6/> and save it to D:\gitea. Then, open CMD in that directory and type the web command ***gitea-1.23.6-windows-4.0-amd64.exe*** to launch Gitea. The following setting is similar with the above one. However, the database is chosen as SQLite3 because MySQL is very complicated.

```
D:\gitea>gitea-1.23.6-windows-4.0-amd64.exe web
2025/12/11 14:33:58 cmd/web.go:253:runWeb() [I] Starting Gitea on PID: 17084
2025/12/11 14:33:58 cmd/web.go:112:showWebStartupMessage() [I] Gitea version: 1.23.6 built with GNU Make 4.3, go1.23.7 :
bindata, sqlite, sqlite_unlock_notify
2025/12/11 14:33:58 cmd/web.go:113:showWebStartupMessage() [I] * RunMode: prod
2025/12/11 14:33:58 cmd/web.go:114:showWebStartupMessage() [I] * AppPath: D:/gitea/gitea-1.23.6-windows-4.0-amd64.exe
2025/12/11 14:33:58 cmd/web.go:115:showWebStartupMessage() [I] * WorkPath: D:\gitea
2025/12/11 14:33:58 cmd/web.go:116:showWebStartupMessage() [I] * CustomPath: D:\gitea\custom
2025/12/11 14:33:58 cmd/web.go:117:showWebStartupMessage() [I] * ConfigFile: D:\gitea\custom\conf\app.ini
2025/12/11 14:33:58 cmd/web.go:118:showWebStartupMessage() [I] Prepare to run install page
2025/12/11 14:33:58 cmd/web.go:315:listen() [I] Listen: http://0.0.0.0:3000
2025/12/11 14:33:58 cmd/web.go:319:listen() [I] AppURL(ROOT_URL): http://localhost:3000/
2025/12/11 14:33:58 ...s/graceful/server.go:50:NewServer() [I] Starting new Web server: tcp:0.0.0.0:3000 on PID: 17084
```



初始配置

如果您正在使用 Docker 容器运行 Gitea，请务必先仔细阅读 [官方文档](#) 后再对本页面进行填写。

数据库设置

Gitea 需要使用 MySQL、PostgreSQL、MSSQL、SQLite3 或 TiDB (MySQL协议) 等数据库

数据库类型 *

SQLite3

数据库文件路径 *

D:\gitea\data\gitea.db

SQLite3 数据库的文件路径。
如果以服务的方式运行 Gitea，请输入绝对路径。

一般设置

站点名称 *

Gitea: Git with a cup of tea

您可以在这里输入您公司的名称。

仓库根目录 *

D:\gitea\data\gitea-repositories

所有远程 Git 仓库将保存到此目录。

LFS根目录

D:\gitea\data\lfs

存储为Git LFS的文件将被存储在此目录。留空禁用LFS

以用户名运行 *

陈芷薇

输入 Gitea 运行的操作系统用户名。请注意，此用户必须具有对仓库根路径的访问权限。

服务器域名 *

localhost

服务器的域名或主机地址。

SSH 服务端点

22



Summary

Port conflicts are likely to occur when installing without Docker.

```
D:\gitea>gitea-1.23.6-windows-4.0-amd64.exe web
2025/12/11 14:04:38 cmd/web.go:253:runWeb() [I] Starting Gitea on PID: 26116
2025/12/11 14:04:38 cmd/web.go:112:showWebStartupMessage() [I] Gitea version: 1.23.6 built with GNU Make 4.3, go1.23.7 :
bindata, sqlite, sqlite_unlock_notify
2025/12/11 14:04:38 cmd/web.go:113:showWebStartupMessage() [I] * RunMode: prod
2025/12/11 14:04:38 cmd/web.go:114:showWebStartupMessage() [I] * AppPath: D:/gitea/gitea-1.23.6-windows-4.0-amd64.exe
2025/12/11 14:04:38 cmd/web.go:115:showWebStartupMessage() [I] * WorkPath: D:/gitea
2025/12/11 14:04:38 cmd/web.go:116:showWebStartupMessage() [I] * CustomPath: D:/gitea/custom
2025/12/11 14:04:38 cmd/web.go:117:showWebStartupMessage() [I] * ConfigFile: D:/gitea/custom/conf/app.ini
2025/12/11 14:04:38 cmd/web.go:118:showWebStartupMessage() [I] Prepare to run install page
2025/12/11 14:04:38 cmd/web.go:315:listen() [I] Listen: http://0.0.0.0:3000
2025/12/11 14:04:38 cmd/web.go:319:listen() [I] AppURL(ROOT_URL): http://localhost:3000/
2025/12/11 14:04:38 ...s/graceful/server.go:50:NewServer() [I] Starting new Web server: tcp:0.0.0.0:3000 on PID: 26116
2025/12/11 14:04:38 ...s/graceful/server.go:76:ListenAndServe() [E] Unable to Getlistener: listen tcp 0.0.0.0:3000: bind
: Only one usage of each socket address (protocol/network address/port) is normally permitted.
2025/12/11 14:04:38 cmd/web.go:364:listen() [E] Failed to start server: listen tcp 0.0.0.0:3000: bind: Only one usage of
each socket address (protocol/network address/port) is normally permitted.
2025/12/11 14:04:38 cmd/web.go:366:listen() [I] HTTP Listener: 0.0.0.0:3000 Closed
2025/12/11 14:04:38 cmd/web.go:150:serveInstall() [E] Unable to open listener for installer. Is Gitea already running?
2025/12/11 14:04:38 .../graceful/manager.go:168:doHammerTime() [W] Setting Hammer condition
2025/12/11 14:04:39 .../graceful/manager.go:184:doTerminate() [W] Terminating
2025/12/11 14:04:39 cmd/web.go:156:serveInstall() [I] PID: 26116 Gitea Web Finished
Command error: listen tcp 0.0.0.0:3000: bind: Only one usage of each socket address (protocol/network address/port) is n
ormally permitted.
```