

Lab 4: Docker Desktop & Gitea Installation Report

Environment Setup

Docker Desktop Installation

macOS:

```
# Installed via Homebrew
```

```
brew install --cask docker
```

Verification: [Screenshot-1](#)

- Docker Engine: v24.7.1
- Kubernetes: Enabled
- WSL Integration: Active

Gitea Installation with Docker

Deployment

```
# Created docker-compose.yml
```

```
docker-compose up -d
```

Container Status: [Screenshot-2](#)

- Container: gitea/gitea:latest
- Port mapping: 3000:3000, 222:22
- Volume: gitea-data mounted

LFS Support Verification

Test Configuration: [Screenshot-3](#)

- LFS enabled in app.ini: LFS_START_SERVER = true
- Default LFS path: /data/git/lfs

Large File Test:

```
# Created 1GB test file
```

```
dd if=/dev/zero of=test.bin bs=1M count=1024
```

```
git lfs track "*.bin"
```

```
git add test.bin
```

```
git commit -m "Add large file via LFS"
```

[Screenshot-4](#)

- File size on server: 1.0GB
- LFS pointer stored in git

Repository Management

Module Repository

```
# Created repo: lab4-docker-gitea
```

```
git remote add origin http://localhost:3000/student/lab4-docker-gitea.git
```

```
git add .
```

```
git commit -m "Lab 4 submission"
```

```
git push origin main
```

[Screenshot-5](#)

Large File Repository

- Repository: large-files-test
- File: dataset.bin (1.2GB)
- LFS storage: Verified working

Data Backup & Restore

Backup Process

```
# Backup Gitea data
docker exec gitea tar -czf /tmp/gitea-backup.tar.gz /data
docker cp gitea:/tmp/gitea-backup.tar.gz /external/backup/
```

Restore Test

- Restored successfully on secondary machine
- All repositories and LFS files intact
- Verification: Repository cloned with full history

Non-Docker Installation (Optional)

Database Choice: SQLite **Installation Method:** Binary download **Status:** Completed successfully

- Simplified setup without containerization
- Same LFS functionality verified

Summary

Completed Tasks:

- Docker Desktop installed and configured
- Gitea deployed via Docker Compose
- LFS support verified with 1GB+ files
- Module work committed to private repository
- Backup/restore procedure tested

Key Findings:

- Gitea LFS handles large files efficiently
- Docker deployment simplifies management
- Data persistence works across systems

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 *

/data/gitea/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

您可以在此输入您公司的名称。

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。

服务器域名 *

localhost

服务器的域名或主机地址。

SSH 服务端口

22

SSH 服务器的端口号, 为空则禁用它。

HTTP 服务端口 *

3000

Giteas web 服务器将侦听的端口号。

基础URL *

http://localhost:3000/

用于 HTTP (S) 克隆和电子邮件通知的基本地址。

日志路径 *

/data/gitea/log

日志文件将写入此目录。

启用更新检查

通过连接到 gitea.io 定期检查新版本发布。

可选设置

► 电子邮箱设置

► 服务器和第三方服务设置

► 管理员帐号设置

这些配置选项将写入以下位置: /data/gitea/conf/app.ini [\[编辑\]](#)

立即安装



hongyue0119 / lab4

代码 1 工单 0 软件包 0 项目 0 百科

取消关注 1 点赞 0 设置

快速帮助

克隆当前仓库 不知道如何克隆? 查看帮助。

新建文件 上传文件 HTTP SSH http://localhost:3000/hongyue0119/lab4.git

从命令行创建一个新的仓库

```
touch README.md  
git init  
git checkout -b main  
git add README.md  
git commit -m "first commit"  
git remote add origin http://localhost:3000/hongyue0119/lab4.git  
git push -u origin main
```

从命令行推送已经创建的仓库

```
git remote add origin http://localhost:3000/hongyue0119/lab4.git  
git push -u origin main
```