

???? ???? ?

- `git commit -m "fix bug"` & `git push`
- #1. `git checkout -b feature` `git pull` `git merge` `git push`
- #2. `www` `503` `git`
- #3. `DB` & `git` `git`
- #4. `git` `git` `git` `git` `git` `git`
- #5. `Git` `README.md` `git` `git` `git` `git` `push` `git` `git`
- #6. The current branch master has no upstream branch. `git`

? ???? ?? & ?? ?? ???

?? ???? ??

- Docker 環境 BookStack + MariaDB

```
./DainWiki/
├── bookstack/          # BookStack のソースコード
│   └── ...              # 詳細は README を参照
├── bookstack_db/       # MariaDB のデータベース
│   └── ...              # (.frm, .ibd の DB ファイル)
├── certs/              # SSL 証明書 (必要に応じて)
├── html/               # nginx の静的コンテンツ
├── vhost.d/            # 仮想ホスト設定 (必要に応じて)
└── docker-compose.yml  # Docker コンテナ設定
```

bookstack container

```
/app/www/
├── .env                # Laravel の環境変数
├── artisan             # Laravel CLI の実行ファイル
├── config/             # 各種設定ファイル
│   ├── app.php         # アプリケーション設定
│   └── database.php     # データベース接続設定
├── routes/             # ルーティング設定
│   └── web.php          # Web アプリケーションのルーティング
├── resources/          # リソースファイル
│   ├── views/          # Blade テンプレート
│   └── lang/            # 言語ファイル
├── storage/            # 各種ストレージ
│   ├── logs/           # ログファイル
│   ├── app/            # アプリケーションのストレージ
│   └── framework/      # フレームワークのストレージ
└── public/             # Web サイトの公開ディレクトリ
    └── uploads/        # アップロードされたファイル
```

bookstack_db container

```
./
├── var/
│   └── lib/
│       ├── mysql/                # ← DB: MySQL DB
│       │   ├── bookstack/        # BookStack DB
│       │   │   ├── users.ibd
│       │   │   ├── pages.ibd
│       │   │   └── ...
│       │   ├── mysql/            # MySQL DB (root, etc)
│       │   ├── performance_schema/
│       │   └── ibdata1, ib_logfile0
├── etc/
│   └── my.cnf                    # MariaDB config
└── Linux ...
```

1. SSH? ??? ?? ??

Linux, macOS, WSL, `ssh` , VSCode `ssh`

```
ssh -i ~/.ssh.pem ubuntu@your-lightsail-ip
```

`ssh` (Windows PowerShell / CMD)

```
ssh -i "C:\Users\aweso\pem\ssh.pem" ubuntu@your-lightsail-ip
```

2. home ??

- `.yml`` `ssh` `ssh`
- `docker ps`, `docker-compose up -d`, `docker-compose down`)

```
cd DainWiki
```

3. ?? ????? ??? ?????

```
docker exec -it bookstack bash
```

4. Laravel ? ??? ??

- ``.env`` 文件 文件

```
docker exec -it bookstack bash
cd /app/www
ls -al .env
```

5. Laravel ?? ?? ?? ? ?????? ?? ??

- DB 文件 文件
- DB 文件 (文件) 文件

```
php artisan config:clear
php artisan cache:clear
php artisan view:clear
php artisan migrate --force
```

6. maria db ??? ?? ?

```
docker exec -it bookstack_db bash 文件 文件
mysql -u root -p
文件 rootpassword










文件文件
Welcome to the MariaDB monitor...
MariaDB [(none)]>
```

7. ?? ??? ?? ??

```
df -h
```


#1. ?? ???? ?? ??????? ?? SSL
?? ?? ?

?? ??

- (Docker  AWS Lightsail  )
- HTTPS   'img alt="HTTPS logo" data-bbox="315 855 345 875"/>  'img alt="HTTPS logo" data-bbox="455 855 485 875"/>  

? ?? ??

- SSL

--	--

--	--

? ? ? ? ?

- `docker-compose.yml` `2` Docker `SSL` `Nginx reverse proxy (jwilder/nginx-proxy),` `Let's Encrypt companion (JrCs/letsencrypt-nginx-proxy-companion)`

--	--	--	--

```

└─ DainWiki/
  └─ bookstack/
    └─ bookstack_db/
      └─ docker-compose.yml

```

yml ☐ ☐ ☐

```

DainWiki/
├── bookstack/           # BookStack のインストール
├── bookstack_db/        # MariaDB のインストール
├── certs/               # SSL 証明書 (openssl)
├── html/                # nginx の静的コンテンツ
├── vhost.d/            # nginx の仮想ホスト設定

```

└─ docker-compose.yml # 00 00 00

```
ubuntu@ip-172-26-5-34:~/DainWiki$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  NAMES              CREATED        STATUS
d25363af4d9b   lscr.io/linuxserver/bookstack:late "/init"                  bookstack          6 minutes ago  Up 6 minutes
e3662714bf27   jrcs/letsencrypt-nginx-proxy-companion "/bin/bash /app/entr... letsencrypt         6 minutes ago  Up 6 minutes
fa3cf91211e6   jwilder/nginx-proxy                "/app/docker-entrypo... nginx-proxy         6 minutes ago  Up 6 minutes
:80->80/tcp, :::80->80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp
613018760a99   mariadb:10.4                      "docker-entrypoint.s... bookstack_db        6 minutes ago  Up 6 minutes
p
```

01 ① Let's Encrypt 000 000 00

```
service:
- VIRTUAL_HOST=dainwiki.com
- LETSENCRYPT_HOST=dainwiki.com
- LETSENCRYPT_EMAIL=your@email.com
```

01 ② APP_URL 0 http → https 000

```
- APP_URL=http://dainwiki.com
+ APP_URL=https://dainwiki.com
```

→ 0000 0000 https 000 000 000 .

01 ③ nginx-proxy 0 letsencrypt 000 00

```
nginx-proxy:
  image: jwilder/nginx-proxy
  container_name: nginx-proxy
  restart: always
  ports:
    - "80:80"
    - "443:443"
  volumes:
    - /var/run/docker.sock:/tmp/docker.sock:ro
    - ./certs:/etc/nginx/certs
    - ./vhost.d:/etc/nginx/vhost.d
    - ./html:/usr/share/nginx/html
  networks:
    - bookstack_net
```

```
letsencrypt:
  image: jrcs/letsencrypt-nginx-proxy-companion
  container_name: letsencrypt
  restart: always
  environment:
    - NGINX_PROXY_CONTAINER=nginx-proxy
    - DEFAULT_EMAIL=your@email.com
  volumes:
    - /var/run/docker.sock:/var/run/docker.sock:ro
    - ./certs:/etc/nginx/certs
    - ./vhost.d:/etc/nginx/vhost.d
    - ./html:/usr/share/nginx/html
  depends_on:
    - nginx-proxy
  networks:
    - bookstack_net
```

→ ☐ ☐ + SSL ☐ ☐

?? ??

- ## 503 Service Temporarily Unavailable

nginx

? ? ? ?

- ? ? ? ?

- Docker [] nginx-proxy + letsencrypt-nginx-proxy-companion [] []
- .env [] docker-compose.yml [] [] [] [] []

```
VIRTUAL_HOST=dainwiki.com,www.dainwiki.com
LETSENCRYPT_HOST=dainwiki.com,www.dainwiki.com
LETSENCRYPT_EMAIL=you@example.com
```

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

❏ ❸❸❸ ❸❸

- www.dainwiki.com❏ ❸❸ SSL ❸❸❸❸ ❸❸ ❸❸❸
- [www ❸❸ ❸❸](http://www.❸❸.❸❸) [https://❸❸ ❸❸](https://❸❸.❸❸) ❸❸❸❸❸

#3. DB ??? & ??? ?? ?????

?? ??

- 1000 BookStack API 10 10 10 1000 ((Apache + PHP + MySQL) 10 10 DB 1000 10)
- 503 10 1000 1000 1000 10 1000 1000 10 10 1000 10 1000

? ?? ??

- MySQL 1000 10 1000 htdocs 1000 10 1000 1000
- 10 1000 10 10 10 1000 1000 10
- 10 1000 10 (volume, data 1000) 10

? ?? ??

- 10 : AWS Lightsail 10 10 , Docker + Volume 1000 1000 10 10
- 10 : docker-compose up -d 10 10 10 , 1000 1000 volume: 10 1000
- 10 10 :
10 10 60GB 1000 10 1000 10000 10 1000 10 10
10 mysqldump 10 10 10 1000 10 10
- 1000 : AWS Lightsail \$10 10 (2GB RAM, vCPU 1, SSD 60GB)

10 10 10

```
1000DainWiki/
├─ 1000bookstack/      # BookStack 10 10, 1000
├─ 1000bookstack_db/   # MariaDB 1000 (10 1000, 10 10)
├─ 1000docker-compose.yml
└─ 1000.env
```

10 yml. 10 1000 10 10

```
bookstack:
  volumes:
    - ./bookstack_data:/config

bookstack_db:
  volumes:
    - ./db_data:/var/lib/mysql
```

容器 名称 (bind mount)容器 名称 .

- 容器 名称 (./bookstack_data)容器 名称 (/config)容器 名称
- 容器 名称 (=Lightsail 容器 名称)容器 名称 容器 名称 .
- 容器 名称 (/)容器 名称 ~/DainWiki/bookstack_data, db_data 容器 名称 (容器 名称 : /home/ubuntu/DainWiki/bookstack_data)

容器 名称 容器 名称 容器 名称

```
docker inspect bookstack
```

- 容器 名称 Mounts 容器 名称 容器 名称

容器 名称 容器 名称

```
df -h
```

```
ubuntu@ip-172-26-5-34:~/DainWiki$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        58G   3.9G   54G   7% /
tmpfs            956M    0   956M   0% /dev/shm
tmpfs            383M   1.3M   381M   1% /run
tmpfs            5.0M    0    5.0M   0% /run/lock
efivarfs         128K   3.6K   120K   3% /sys/firmware/efi/efivars
/dev/nvme0n1p16  881M   136M   684M  17% /boot
/dev/nvme0n1p15  105M    6.1M    99M   6% /boot/efi
tmpfs            192M    12K   192M   1% /run/user/1000
```

?? ??

- ? ?? ??

- ? ?? ??

SSL ☐ ☐ ☐ ☐

```
docker logs letsencrypt
```

```
Reloading nginx proxy (nginx-proxy)...
2025/05/18 04:30:43 Generated '/etc/nginx/conf.d/default.conf' from 4 containers
2025/05/18 04:30:43 [warn] 63#63: "ssl_stapling" ignored, no OCSP responder URL in the certificate "/etc/nginx/certs/www.dainwiki.com.crt"
2025/05/18 04:30:43 [notice] 63#63: signal process started
Reloading nginx proxy (nginx-proxy)...
2025/05/18 04:30:43 Contents of /etc/nginx/conf.d/default.conf did not change. Skipping notification ''
2025/05/18 04:30:43 [warn] 77#77: "ssl_stapling" ignored, no OCSP responder URL in the certificate "/etc/nginx/certs/www.dainwiki.com.crt"
2025/05/18 04:30:43 [notice] 77#77: signal process started
```

```
docker exec -it nginx-proxy sh
```

```
docker exec -it nginx-proxy bash
```

ls /etc/nginx/certs/

- daidwiki.com/, www.daidwiki.com/ fullchain.pem, cert.pem, key.pem
- dhparam.pem

```
ubuntu@ip-172-26-5-34:~/DaidWiki$ docker exec -it nginx-proxy bash
root@c44cb4cfd4ae:/app# ls /etc/nginx/certs/
daidwiki.com          daidwiki.com.dhparam.pem  www.daidwiki.com.chain.pem  www.daidwiki.com.key
daidwiki.com.chain.pem  daidwiki.com.key          www.daidwiki.com.crt
daidwiki.com.crt       dhparam.pem               www.daidwiki.com.dhparam.pem
```

#5. Git? README.md ??? ??? ??? ?? push ?? ??

?? ??

- GitHub `README.md` , `Git` `git push` .
- GitHub `pull` `push` .
- `git pull --rebase` .

? ?? ??

명령어	효과	비고
<code>git pull origin main --allow-unrelated-histories</code>	원본 브랜치에서 변경사항을 가져옴	Git 2.14.0 이상 버전에서 사용 가능
<code>git pull origin main --rebase</code>	원본 브랜치에서 변경사항을 가져옴 (리베이스)	rebase 옵션을 사용하면 변경사항을 리베이스

```
1 pull --allow-unrelated-histories
```

```
[ ] [ ][ ][ ][ ] '[ ][ ] [ ] '[ ][ ] [ ][ ] Git[ ][ ][ ][ ] [ ][ ][ ][ ] --rebase[ ][ ][ ] .
```

```
--allow-unrelated-histories  1 1 11 1 1 1 111 11 1 111 11 1 1
1 1 ,
```


 $X \rightarrow A \rightarrow B \rightarrow C$

? ?? ??

```
# 1. 仓库初始化
git remote add origin https://github.com/xxxxx/xxxxx.git
```

```
# 2. 分支合并到主分支并允许无关历史
git pull origin main --allow-unrelated-histories
```

```
# 3. 分支合并到主分支
```

```
# 4. 推送代码到远程仓库
```

```
git push origin main
```

或者使用 `git pull` 命令来拉取远程仓库的代码并合并到本地分支。

- `git pull --rebase origin main` → 将本地分支与远程分支合并，并基于远程分支的最新状态进行合并。
- `git pull origin main` → 将本地分支与远程分支合并，并基于本地分支的最新状态进行合并。

#6. The current branch master has no upstream branch. ??

?? ??

- Git Bash 打开 终端 窗口 输入 命令

The current branch master has no upstream branch.

\$ git push fatal: The current branch master has no upstream branch. To push the current branch and set the remote as upstream, use git push --set-upstream origin master To have this happen automatically for branches without a tracking upstream, see 'push.autoSetupRemote' in 'git help config'.

? ?? ??

- 在 本地 创建 (master) 分支 并 切换 到 该 分支
- master 分支 设置 上游 分支 --set-upstream 指定 上游 分支
- main 分支 设置 上游 分支 并 推送 到 远程 仓库

? ?? ??

- master 分支 设置 上游 分支 并 推送 到 远程 仓库
- 在 本地 创建 分支 + GitHub(远程 仓库 origin) 分支 设置 上游 分支 并 推送 到 远程 仓库

```
git branch -m master main
git push -u origin main
# (GitHub 默认 分支 是 main)
git push origin --delete master # (删除)
```

1. 在 本地 创建 分支 并 切换 到 该 分支

```
git branch -m master main
```

- -m = move (rename)

2. `origin` 分支 重命名

```
git push -u origin main
```

- -u `main` 分支 与 `upstream` 分支 关联

3. GitHub 仓库 默认分支 从 `master` 改为 `main`

1. GitHub 仓库 设置 → 分支 设置
2. Settings → Branches 分支 设置
3. Default branch `master` → `main` 分支 设置

4. 删除 `master` 分支

```
git push origin --delete master
```

本地分支

本地 `master` 分支 与 `origin/master` 分支 关联

```
git push --set-upstream origin master
```

本地 `origin/master` 分支 与 `master` 分支 (origin) 关联。

本地 `master` 分支 与 `origin/master` 分支 关联，

本地 `origin/master` 分支 与 `master` 分支 关联。