

# Docker Hands-on Guide

## やること

- Windows WSL(Ubuntu)+Docker / Mac+DockerUbuntu+Docker  
(DockerDesktopを使用する場合は、Dockerコンテナを起動する必要はありません  
※業務使用は条件次第で有償)
- Gitリポジトリのクローン
- Dockerfileを作成
- Dockerコンテナの各種操作

## Gitリポジトリのクローン

自身のPC内で任意の場所にリポジトリを作成してください。

```
$ git clone https://github.com/Daisuke-Ito-fwd/docker_sample.git
```

```
// WSLにGitがない場合
```

```
$ sudo apt update
```

```
$ sudo apt install git
```

```
$ git config user.name "Your Name"
```

```
$ git config user.email "sample@example.com"
```

## ディレクトリ構成

ここから先は WSL(Ubuntu)での操作となります。

(Macの場合はUbuntuコンテナ内)

```
Windows
└─ 3001ポート – WSL Ubuntu
    └─ docker_sample/
        ├── app/ # Flaskアプリケーション
        ├── sh/ # シェルスクリプト
        ├── forMac/ # Mac用のDockerfile
        ├── Dockerfile.dev # 開発環境用
        ├── Dockerfile.prd # 本番環境用
        ├── nginx.conf # Nginx設定ファイル
        ├── ReadMe.md
        └── requirements.txt # Python依存関係
```

```
Mac
└─ 3001ポート - Ubuntu
    └─ docker_sample/
        ├── app/
        ├── sh/
        ├── forMac/
        ├── Dockerfile.dev
        ├── Dockerfile.prd
        ├── nginx.conf
        ├── ReadMe.md
        └── requirements.txt
```

## Dockerのインストール (便利シェル)

### 1. スクリプトをダウンロードして実行

```
$ pwd  
/mnt/docker_sample  
$ cd sh  
$ curl -fsSL https://get.docker.com -o get-docker.sh  
$ sudo sh get-docker.sh
```

### 2. Dockerデーモンをバックグラウンドで起動

```
$ dockerd &
```

# Dockerイメージをビルドする

## 1. ディレクトリ移動

```
$ cd ..  
$ pwd  
/mnt/docker_sample
```

## 2. イメージをビルド Dockerfile.devを使用します

```
$ docker build -t docker_dev -f Dockerfile.dev .
```

## 3. 接続エラー時の対処

```
$ cat /etc/resolv.conf  
$ echo "nameserver 8.8.8.8" > /etc/resolv.conf
```

## コンテナの作成 & 起動 (開発環境)

### 1. コンテナを作成・起動

```
$ pwd  
/mnt/docker_sample  
$ docker run -dit --name docker_dev -p 3001:3001 -v src:/mnt/src docker_dev
```

### 2. ブラウザでアクセス

- <http://localhost:3001>



# コンテナ操作

## 0. コンテナの状態を確認・起動

```
$ docker ps -a  
$ docker start docker_dev
```

## 1. コンテナにアクセス

```
$ docker exec -it docker_dev /bin/bash
```

## 2. コンテナを停止

```
$ docker stop docker_sample
```

### 3. コンテナを削除

```
$ docker rm docker_sample
```

### 4. イメージを削除

```
$ docker rmi <image_id>
```

# Dockerビルド(本番向け) Flask + Gunicorn + Nginx

## 1. Dockerfileをビルド

```
$ docker build -t docker_prod -f Dockerfile.prod .
```

## 2. コンテナを作成・起動

```
$ docker run -dit --name docker_prod -p 3001:3001 -v src:/mnt/src docker_prod
```

## 3. ブラウザでアクセス

- <http://localhost:3001>

## 4. コンテナにアクセス

```
$ docker exec -it docker_prod /bin/bash
```

## Docker関連 VSCodeの便利拡張機能

- Docker
- Remote - WSL
- Remote - Containers
- Remote - SSH

## その他 VSCodeの便利拡張機能

- Git Graph
- GitHub Copilot
- GitHub Pull Requests and Issues
- Marp for VS Code