

概要

- 表題を利用した環境を作る必要があったので、メモします。
- 合わせて日本語化など

環境

- Docker + Docker-compose が利用できること。
- Python3.6
- Django3.1(最新)

手順

ディレクトリの構成

- 下記構成でディレクトリとファイルを作成します。

```
./WORKDIR
|--- docker-compose.yml      # Docker-compose本体
|---.gitignore               # git対象外用（今回は作るだ、gitを使うと
|---nginx
|   |---nginx.conf           # nginxの設定ファイル
|   |---uwsgi_params          # uwsgi用パラメータ
|---django
|   |---Dockerfile            # Dockerfile
|   |---requirements.txt      # pipでインストールするリスト
|   |---.env                  # django用の環境変数
|---mysql
|   |---.env                  # mysql用の環境変数
|---sql
|   |---init.sql              # DB起動時(初回のみ)に流すスクリプト
```

ファイルの作成と編集

- docker-compose.yml

docker-compose.yml

```
version: '3'

services:
  nginx:
    image: nginx:1.13

    ports:
      - "8000:8000"

    volumes:
      - ./nginx/nginx.conf:/etc/nginx/nginx.conf
      - ./nginx/uwsgi_params:/etc/nginx/uwsgi_params
      - ./static:/static

    depends_on:
      - django

  mysql:
    image: mysql:5.7

    command: mysqld --character-set-server=utf8mb4 --collation-server=utf8mb4_unicode_ci

    ports:
      - "3306:3306"

    env_file:
      - ./mysql/.env

    volumes:
      - ./mysql:/var/lib/mysql
      - ./sql:/docker-entrypoint-initdb.d
```

```
django:

  build: ./django

  command: uwsgi --socket :8001 --module app.wsgi --py-autoreload 1 --logto /tmp/mylog.log

  volumes:

    - ./src:/code

    - ./static:/static

  expose:

    - "8001"

  env_file:

    - ./django/.env

  depends_on:

    - mysql
```

- .gitignore

```
mysql/*
```

- django/Dockerfile

```
FROM python:3.6
ENV PYTHONUNBUFFERED 1
RUN mkdir /code
WORKDIR /code
ADD requirements.txt /code/
RUN pip install -r requirements.txt
ADD . /code/
```

- `django/requirements.txt`

django/requirements.txt

```
Django
uwsgi
PyMySQL
```

- `django/.env`

django/.env

```
MYSQL_ROOT_PASSWORD=root
MYSQL_DATABASE=db
MYSQL_USER=bbt-user
MYSQL_PASSWORD=password
```

- `nginx/nginx.conf`

nginx/nginx.conf

```
user    nginx;

worker_processes  1;

error_log  /var/log/nginx/error.log warn;

pid        /var/run/nginx.pid;


events {
    worker_connections  1024;
}
```

```
http {  
  
    include /etc/nginx/mime.types;  
  
    default_type application/octet-stream;  
  
    log_format main '$remote_addr - $remote_user [$time_local] "$request" '  
                    '$status $body_bytes_sent "$http_referer" '  
                    '$http_user_agent' "$http_x_forwarded_for";  
  
    access_log /var/log/nginx/access.log main;  
  
    sendfile on;  
  
    #tcp_nopush on;  
  
    keepalive_timeout 65;  
  
    #gzip on;  
  
  
    upstream django_8001 {  
  
        ip_hash;  
  
        server django:8001;  
    }  
  
  
    server {  
  
        listen 8000;  
  
        server_name 127.0.0.1;  
  
        charset utf-8;  
  
  
        location /static {  
  
            alias /static;  
        }  
  
  
        location / {  
  
            uwsgi_pass django_8001;  
  
            include /etc/nginx/uwsgi_params;  
        }  
    }  
}
```

```
server_tokens off;

}
```

- nginx/uwsgi_params

```
uwsgi_param QUERY_STRING      $query_string;
uwsgi_param REQUEST_METHOD    $request_method;
uwsgi_param CONTENT_TYPE      $content_type;
uwsgi_param CONTENT_LENGTH     $content_length;

uwsgi_param REQUEST_URI        $request_uri;
uwsgi_param PATH_INFO           $document_uri;
uwsgi_param DOCUMENT_ROOT      $document_root;
uwsgi_param SERVER_PROTOCOL    $server_protocol;
uwsgi_param REQUEST_SCHEME     $scheme;
uwsgi_param HTTPS               $https if_not_empty;

uwsgi_param REMOTE_ADDR        $remote_addr;
uwsgi_param REMOTE_PORT        $remote_port;
uwsgi_param SERVER_PORT        $server_port;
uwsgi_param SERVER_NAME        $server_name;
```

- mysql/.env

mysql/.env

```
MYSQL_ROOT_PASSWORD=root
MYSQL_DATABASE=db
MYSQL_USER=bbt-user
```

```
MYSQL_PASSWORD=password
TZ='Asia/Tokyo'
```

- sql/init.sql

sql/init.sql

```
GRANT ALL PRIVILEGES ON db.* TO 'user'@'%' ;

FLUSH PRIVILEGES;
```

構築

- プロジェクトを作成する。

```
$ docker-compose run django django-admin.py startproject app .
```

- 下記が作成される。

```
./WORKDIR
|--- docker-compose.yml
|---.gitignore
|---nginx
    |---nginx.conf
    |---uwsgi_params
|---django
    |---Dockerfile
    |---requirements.txt
```

```

    |---.env
|---src
    |---app
        |---__init__.py__  # pythonであることの意味（編集しない）
        |---asgi.py        # 利用しない
        |---settings.py    # django本体の設定ファイル
        |---urls.py         # URL処理をする
        |---wsgi.py         # wsgiファイル
        |---manage.py       # 管理用コマンド
|---mysql
    |---.env
    |--- ~~                # 複数作成されるが編集しないため割愛
|---sql
    |---init.sql

```

- src/app/settings.py を編集する

```

#### 追加
import os
import pymysql
STATIC_ROOT = '/static'
pymysql.install_as_MySQLdb()

#### 修正
ALLOWED_HOSTS = ['*']
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': os.environ.get('MYSQL_DATABASE'),
        'USER': os.environ.get('MYSQL_USER'),
        'PASSWORD': os.environ.get('MYSQL_PASSWORD'),
        'HOST': 'mysql',
    }
}

```



```
        'PORT' : '3306',  
    }  
}  
LANGUAGE_CODE = 'ja'  
TIME_ZONE = 'Asia/Tokyo'
```

- DBのマイグレーション

```
$ docker-compose run django ./manage.py makemigrations  
$ docker-compose run django ./manage.py migrate
```

- CSSを反映する

- ```
$ docker-compose run django ./manage.py collectstatic
```

- 起動

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

## 参考

- [Django+Nginx+MySQLの開発環境をDockerで構築する](https://laptrinhx.com/docker-compose-django-mysql-nginx-uwsgi-wo-shitta-huan-jingno-chu-qi-she-ding-850978113/)
  - 自分用に修正しただけでほぼお世話になってます。