

오픈소스기초설계(가) 4 팀

20192910 양재훈, 20180368 박세진
20180392 이창한

Docker 실습 내용 정리

파이썬 웹 프레임워크 Flask 라이브러리를 사용하여 간단하게 웹서버를 실행 시켜볼 것이다.

Dockerfile 작성

```
FROM ubuntu:20.04

ENV TZ=Asia/Seoul
RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezone

RUN sed -ie 's/archive.ubuntu.com/mirror.kakao.com/g' /etc/apt/sources.list

RUN apt-get update && apt-get -y upgrade && apt-get -y autoremove

RUN apt-get install -y python3 \
    python3-pip \
    vim

COPY ./ /chang-web
WORKDIR /chang-web

RUN pip3 install flask

EXPOSE 3000
CMD python3 runserver.py
```

Line by Line 설명

1. Ubuntu 20.04 버전을 베이스 이미지 지정

```
FROM ubuntu:20.04
```

2. Time zone 을 서울로 수정

```
ENV TZ=Asia/Seoul
```

```
RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezone
```

3. apt 패키지 다운로드 서버를 카카오로 바꾸기

```
RUN sed -ie 's/archive.ubuntu.com/mirror.kakao.com/g' /etc/apt/sources.list
```

4. apt 업데이트 및 업그레이드

```
RUN apt-get update && apt-get -y upgrade && apt-get -y autoremove
```

5. 필요한 프로그램 설치

```
RUN apt-get install -y python3 python3-pip vim
```

6. 현재 디렉토리를 도커 이미지 환경에 복사

```
COPY ./ /chang-web
```

7. 현재 작업 디렉토리를 /chang-web 으로 변경

```
WORKDIR /chang-web
```

8. flask 파이썬 라이브러리 설치

```
RUN pip3 install flask
```

9. 3000 포트를 개방하겠다 명시

```
EXPOSE 3000
```

10. runserver.py 을 실행 -> 웹서버 실행

```
CMD python3 runserver.py
```

Docker Image 빌드

```
$ docker build -t chang-web .
```

- -t 는 tag 을 지정하는 옵션이다.
- . 을 하게 되면 현재 디렉토리에 있는 Dockerfile 으로 빌드하게 된다.

```
$ docker images
```

- chang-web 이미지 확인 가능

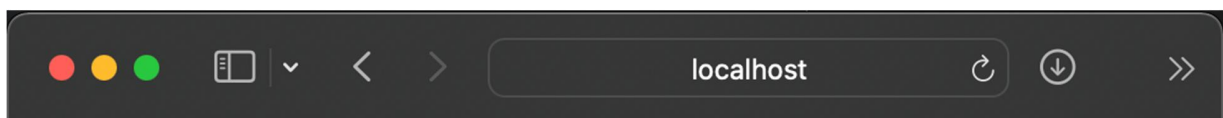
Docker Container 실행

```
$ docker run -it -d -p 3000:3000 chang-web
```

- -d 옵션 daemon 으로 실행
- -it 옵션 상호작용이 되도록
- -p 옵션 포트 지정

```
$ docker ps -a
```

- Status 을 통해 실행 중인지 종료된 것인지 알 수 있다.



Example open source

- 웹 브라우저에서 확인

Docker Container 종료

```
$ docker stop {container_id}
```

Docker Image 삭제

```
$ docker rmi {image_id}
```

역할 분담

- 양재훈: 팀장, 도커 서버 설계
- 박세진 : 코드 작성
- 이창한 : Docker 연동