

3주차

Prerequisite

1. github repo clone

```
https://github.com/BOAZ-bigdata/19Engineering_BASE.git
```

```
https://github.com/BOAZ-bigdata/19Engineering_BASE/tree/week_3
```

2. docker installed: `docker -v`
3. python3 installed: `python3 --version`

Dockerfile

```
# base image

# Use an official Python runtime as a parent image
FROM python:3.8.6

# Set the working directory to /app
WORKDIR /app

# update fundamentals
RUN apt-get update && apt-get install -y

# Copy the current directory contents into the container at /app
COPY . app/

# Install any needed packages specified in requirements.txt
RUN pip3 install -r app/requirements.txt

# Make port 9080 available to the world outside this container
EXPOSE 9080

# Define environment variable
# ENV NAME World
```

```
RUN cd app/

# Run app.py when the container launches
CMD [ "python3", "app/server.py"]
```

FROM 베이스 이미지

주로 **ubuntu**와 같은 OS를 지정
ubuntu:16.04 처럼 **:version** 을 붙여 버전을 명시해줄 수 있고,
따로 버전을 명시해주지 않으면 **latest** 버전으로 이미지를 가져온다.

WORKDIR 작업 디렉토리 지정

Linux의 **cd** 와 같은 개념
directory가 없으면 새로 생성한다.

COPY 호스트의 파일 또는 디렉토리를 이미지로 copy

ADD **COPY**와 비슷지만 **URL**을 지정할 수 있고, 압축을 자동으로 풀어준다.
일반 파일에 대해서는 주로 **COPY** 를 사용한다.

RUN 셸 커맨드를 도커 이미지에서 실행

필요한 패키지 install 등등

EXPOSE 컨테이너를 외부에 열어줄 포트 지정

EXPOSE 8080 이라고 작성했다면,
해당 이미지로 생성된 컨테이너는 외부로 8080번 포트를 개방하겠다는 뜻

EXPOSE 에 포트 번호를 작성했다고 해서 사용자가 해당 포트로 컨테이너에 접근 가능한 것은 아님.

도커파일에서 **EXPOSE**를 작성해 외부로 개방하려는 포트 번호를 지정하고,

호스트 OS에서 해당 컨테이너를 실행시킬 때 위 포트로 매핑시키는 것까지 해야 컨테이너 접근 가능



호스트 OS로 들어오는 포트와 개방된 컨테이너 포트를 매핑해야 컨테이너 접근 가능

`docker build run` 의 `-p` 옵션으로 매핑
`-p 8080:2222`

콜론기준 **왼쪽**은 호스트 OS로 들어오는(사용자가 접근하는)포트,
오른쪽은 매핑시킬 컨테이너의 포트

ex)

```
# 3개의 포트 개방
EXPOSE 2222
EXPOSE 3333
EXPOSE 4444
```

`docker build run -p 8080:2222`

사용자가 8080포트에 접근했을때 컨테이너의 2222포트로 매핑

CMD 도커 컨테이너가 시작할 때 실행할 커맨드 지정

ENTRYPOINT CMD와 비슷하지만 **ENTRYPOINT**는 **override** 불가능

ENV 환경변수 설정

참고 <https://wooono.tistory.com/123>

참고 https://rampart81.github.io/post/dockerfile_instructions/

참고 <https://imkh.dev/docker-expose-ports/>

실습

Work1:



socket-server를 이미지로 빌드하고 컨테이너로 실행시키기

힌트) 리드미 참고

Work2:



socket-client의 client.py 실행시키기

힌트) 리드미 참고

Work3:



포트포워딩 해서 포트 변경해보기

Work4:

multi-client로 변경해보기

Hint) multi-client.py, multi-server.py 참고!
