

명령어

FROM

- 빌드 프로세스를 시작하는 기본 이미지를 정의
- FROM 명령어는 도커 런타임에 사용할 도커 이미지를 지정

LABEL

- 도커 이미지에 메타데이터를 추가
- 메타데이터는 키-값 쌍으로 되어 있음
- 이미지 작성 목적으로 버전, 타이틀, 설명, 라이선스 정보 등을 작성
- 1개 이상 작성 가능

ARG

- 사용자가 docker build 명령을 사용하여 빌더에 전달할 수 있는 변수를 정의
- docker bulid 시점에서 변수값을 전달하기 위해 -- build-arg=인자를 정의하여 사용

```
ARG db_name
```

```
docker build --build-arg db_name=mysql .
```

COPY

- 원본의 새 파일, 디렉터리 또는 리모트 파일 URL을 복사하고 지정된 대상 경로에 생성 중인 이미지의 파일 시스템에 추가

ADD

- 호스트에서 파일을 복사하고 컨테이너 대상 위치에 추가

VOLUME

- 컨테이너의 마운트 지점을 만듦
- 동일한 이미지를 사용해서 새 컨테이너를 만들 때 이전 볼륨과 격리되는 새 볼륨을 생성

RUN

- 명령과 해당 매개변수를 받아 이미지에서 컨테이너를 실행
- 보통 소프트웨어 패키지를 설치하는데 이 명령어를 사용

CMD

- ENTRYPOINT에 매개변수를 제공
- 이 명령어는 docker run과 유사하지만 컨테이너가 인스턴스화 된 후에만 실행될 수 있음
- 여러 개의 CMD를 작성해도 마지막 하나만 처리됨

ENTRYPOINT

- 실행 파일로 실행할 컨테이너를 구성
- CMD와 마찬가지로 생성된 이미지가 컨테이너로 실행될 때 사용하지만, 컨테이너가 실행될 때 명령어 및 인자 값을 전달하여 실행한다는 점

WORKDIR

- 컨테이너상에서 작업할 경로(디렉터리) 전환을 위해 작성
- WORKDIR을 설정하면 RUN, CMD, ENTRYPOINT, COPY, ADD 명령문은 해당 디렉터리를 기준으로 실행

ENV

- 환경 변수를 설정
- 이미지 안에 각종 환경 변수를 지정하는 경우 작성
- 애플리케이션 사용을 쉽게 하려면 사전에 구성되어야 하는 환경 변수들이 있음
 - 자바 홈 디렉터리

- 특정 실행 파일의 경로를 보장하기 위해 절대 경로 지정을 위한 PATH 설정
- 프로그램 버전 등을 사전에 설정
- 또한, 반복된 표현이 사용되는 경우에도 환경 변수 설정을 권장

```
ENV JAVA_HOME /usr/lib/jvm/java-8-oracle
```

EXPOSE

- 컨테이너가 호스트 네트워크를 통해 들어오는 트래픽을 리스닝(listening)하는 포트와 프로토콜을 지정하기 위해 작성
- docker run 사용시 -p 옵션을 통해 사용
- 컨테이너 생성 시에는 반드시 -p 옵션을 사용해야 실제 포트가 바인딩 되기 때문에 EXPOSE는 다른 사용자에게 해당 dockerfile에서 포트를 사용한다고 알려주는 관점이 강함

ENTRYPOINT와 CMD의 차이점

- 커맨드와 동일하게 컨테이너가 시작할 때 수행할 명령을 지정한다는 점은 동일
- entrypoint는 커맨드를 인자로 사용할 수 있는 스크립트의 역할을 할 수 있음

```
(base) gen2@:~$ docker run -it --name no_entrypoint
ubuntu:14.04 /bin/bash
root@be7ce6c45fb5:/#

-----

(base) gen2@:~$ docker run -it --entrypoint="echo"
--name yes_entrypoint ubuntu:14.04 /bin/bash
/bin/bash
(base) gen2@DESKTOP-V59NI0S:~$
```

- entrypoint로 전달한 값이 터미널에 출력
- 컨테이너에 entrypoint가 설정되면 run 명령어의 맨 마지막에 입력된 cmd를 인자로 삼아 명령어를 출력
- 위의 예제에서는 echo /bin/bash가 컨테이너에서 실행되어 /bin/bash가 출력

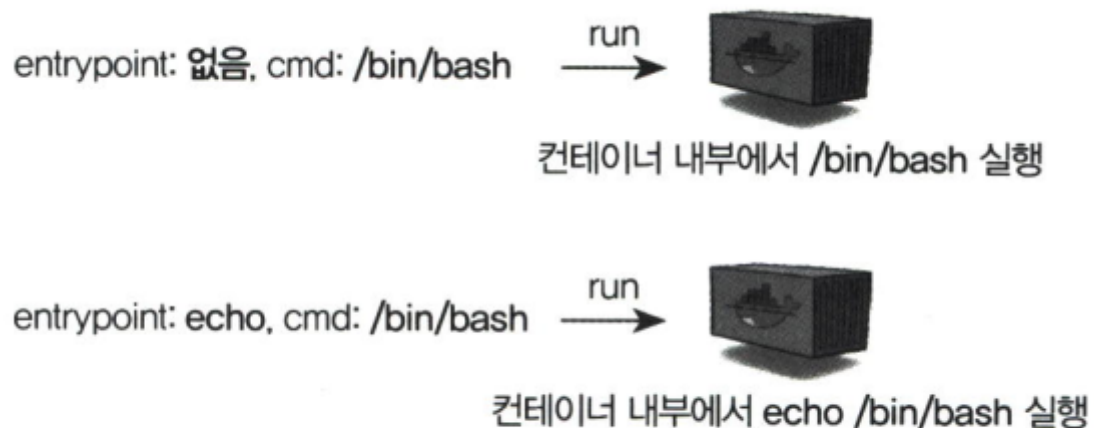


그림 2.60 entrypoint와 cmd의 조합

- CMD와 entrypoint가 둘 다 설정되지 않으면 컨테이너는 생성되지 않고 에러를 출력하므로 반드시 둘 중 하나

나는 설정해야 함

- ENTRYPOINT는 도커 컨테이너 실행 시 항상 수행해야 하는 명령어를 지정(웹 서버나 데이터베이스등의 데몬 실행)
- CMD는 도커 컨테이너 실행 시 다양한 명령어를 지정하는 경유 유용