

Dockek 03

[도커 이미지 생성해보기]

따라하며 배우는 도커

John Ahn

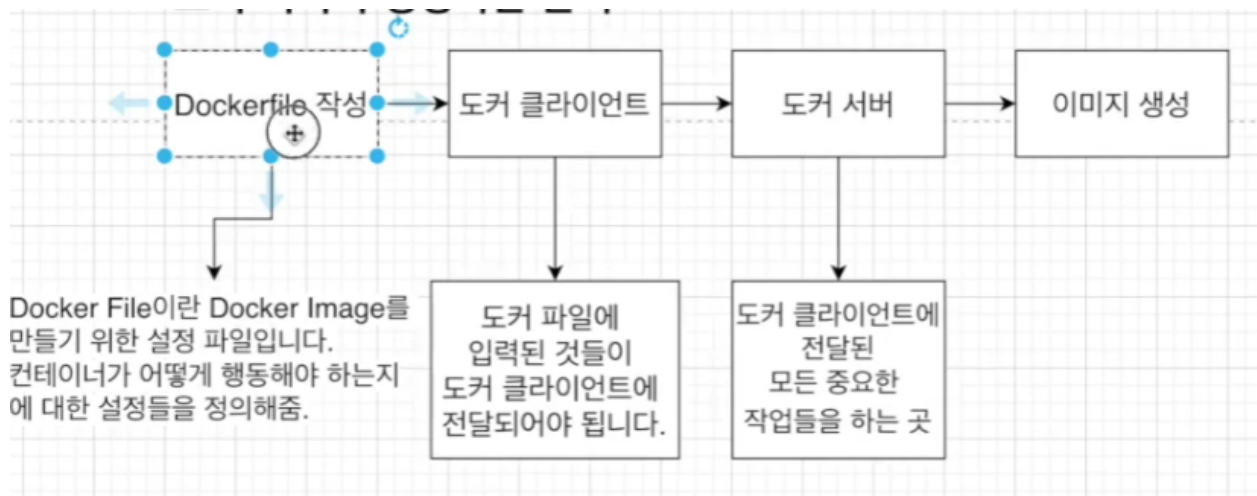
2023.01.15

도커 이미지 만들기

도커 이미지 : 컨테이너를 만들기 위해 필요한 설정이나 종속성들을 갖고있는 소프트웨어 패키지

도커 이미지를 이용해서 도커 컨테이너 생성

`docker create <이미지 이름>`



순서대로 만들어본다.

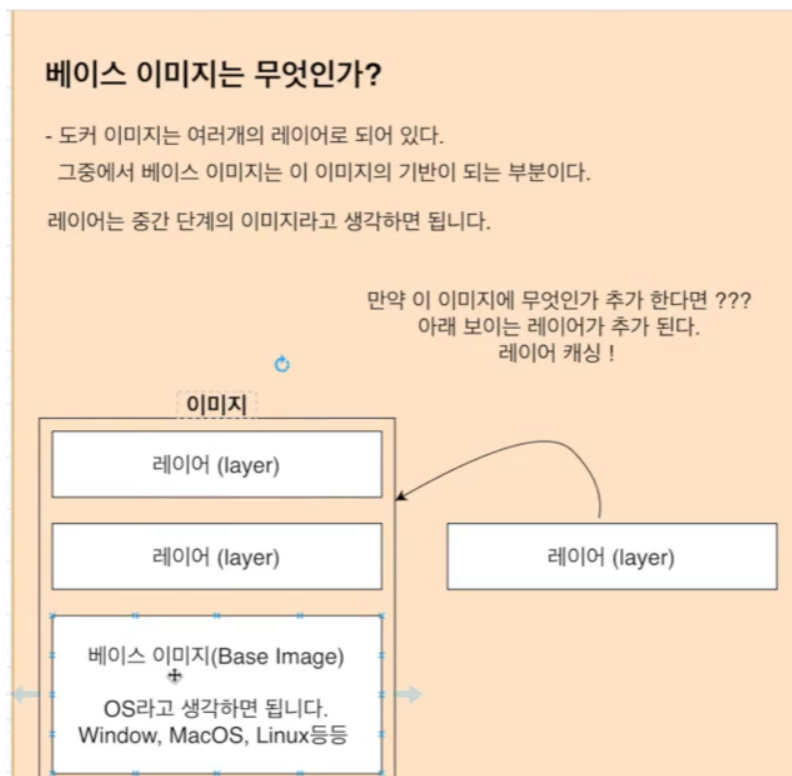
도커 파일 만들기

도커 파일이란?

도커 이미지를 만들기 위한 설정 파일 , 컨테이너가 어떻게 행동해야 하는지에 대한 설정들을 정의해 준다.

도커 파일 만드는 순서

1. 베이스 이미지를 명시해 준다.
2. 추가적으로 필요한 파일을 다운 받기위한 몇가지 명령어를 명시해 준다.
3. 컨테이너 시작시 실행 될 명령어를 명시해준다.



간단히 생각하면 OS 라고 생각하면 된다.

dockerfile을 생성

```
Dockerfile X
Dockerfile
1  #베이스 이미지를 명시해 준다.
2  FROM baseImage
3
4  #추가적으로 필요한 파일들을 다운로드 받는다.
5  RUN command
6
7  # 컨테이너 시작시 실행 될 명령어를 입력해준다.
8  CMD ["executable"]
```

```
Dockerfile X
Dockerfile
1  #베이스 이미지를 명시해 준다.
2  FROM alpine
3
4  #추가적으로 필요한 파일들을 다운로드 받는다.
5  # RUN command
6
7  # 컨테이너 시작시 실행 될 명령어를 입력해준다.
8  CMD ["echo", "hello"]
```

간단한 hello 출력하는 도커 파일

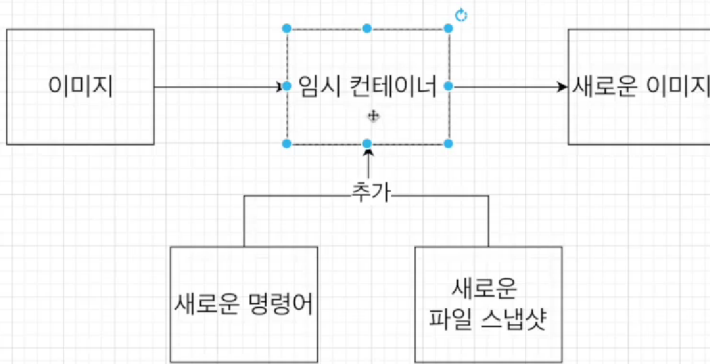
docker build ./ 또는 docker build .

Build 명령어는

- 해당 디렉토리 내에서 dockerfile이라는 파일을 찾아서 도커 클라이언트에 전달시켜준다.
- docker build 뒤에 ./ 와 . 는 둘다 현재 디렉토리를 가르킨다.

결론

- 베이스 이미지에서 다른 종속성이나 새로운 커맨드를 추가 할때는
임시 컨테이너를 만든 후 그 컨테이너를 토대로 새로운 이미지를 만든다 !!!
그리고 그 임시 컨테이너는 지워준다.



베이스 이미지에 추가로 명령어나 파일 스냅샷을 추가해서 새로운 이미지를 만들어 내는것이다.

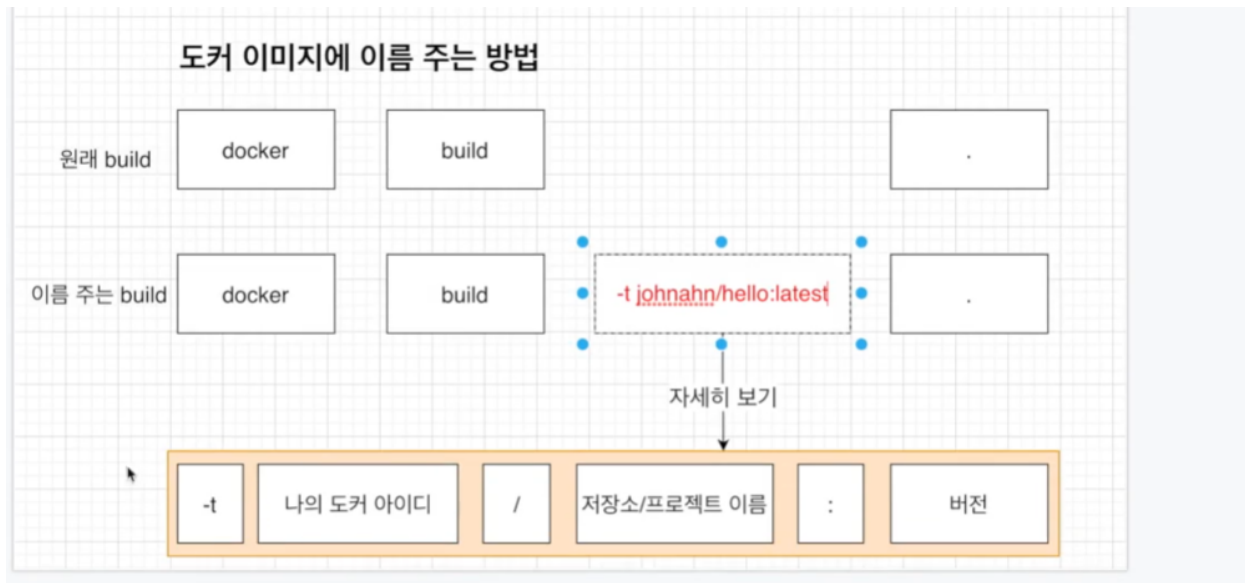
```
Sending build context to Docker daemon 2.048kB
Step 1/2 : FROM alpine
---> 49176f190c7e
Step 2/2 : CMD ["echo", "hello"]
---> Running in 800693864ea2
Removing intermediate container 800693864ea2
---> e4bfd829115d
Successfully built e4bfd829115d
```

베이스 alpine (800-) 에서 e4bf- 컨테이너가 생성된것을 볼 수 있다.

도커 이미지 기억하기 쉬운 이름 주기

```
C:\Users\Lece6>docker run -it e4bfd829115d
hello
```

너무 길고 불편하다.



생성 할 때 이름을 줄 수 있다.

```
D:\dockerResource\docker_example\dockerfile-folder>docker build -t lece619/hello:latest .
Sending build context to Docker daemon 2.048kB
Step 1/2 : FROM alpine
--> 49176f190c7e
Step 2/2 : CMD ["echo", "hello"]
--> Using cache
--> e4bfd829115d
Successfully built e4bfd829115d
Successfully tagged lece619/hello:latest
```

생성완료 실행해보기

```
C:\Users\Lece6>docker run -it lece619/hello:latest
hello
```

문제없이 완료 된다.
