

1 Lab14. docker-compose commands

1. docker-compose pull

-pull 뒤에 이미지를 적지 않으면 모든 이미지를 다운로드한다.

1)\$ cat docker-compose.yml

```
version: '3'
```

```
services:
```

```
  nginx:
```

```
    image: nginx:latest
```

```
    ports:
```

```
      - 8000:80
```

2)\$ docker-compose pull

2. docker-compose build

-build 뒤에 이미지를 적지 않으면 필요한 이미지를 모두 빌드한다.

1)index.html

```
<font size='7' color='blue'>Hello, Docker Compose</font>
```

2)Dockerfile

```
FROM      nginx:latest
```

```
RUN       ["chmod", "755", "/usr/share/nginx/html"]
```

```
RUN       ["chmod", "644", "/usr/share/nginx/html/index.html"]
```

```
ADD       ./index.html /usr/share/nginx/html/
```

3)\$ cat docker-compose.yml

```
version: '3'
```

```
services:
```

```
  nginx:
```

```
    build:
```

```
      context: . <---context가 현재 위치를 의미하고 있음. 같은 위치에 있는 Dockerfile을 사용할 것임.
```

```
      dockerfile: ./dockerfile
```

```
    ports:
```

```
      - 8000:80
```

4)\$ docker-compose build

3. docker-compose up

-pull과 build를 실행했기 때문에 서비스를 구동한다.

1)\$ docker-compose up

-브라우저에서 확인

--Ctrl + C로 중지

2)\$ docker-compose up --build

-필요한 이미지를 강제로 빌드

3)\$ docker-compose up --force-recreate

-이미 실행중인 컨테이너에 뭔가 문제가 생겼거나 docker-compose file을 수정했다면 컨테이너를 새로 생성하면 된다.

4)\$ docker-compose up -d

-서비스를 background로 실행, 실행 후 다시 터미널로 돌아옴.

4. docker-compose ps

1)서비스가 잘 실행되고 있는지 확인

```
61 2)Name의 앞부분은 현재 실행되고 있는 디렉토리
62 $ docker-compose ps
63
64
65 5. docker-compose logs
66 1)컨테이너별 로그를 보고 싶을 때
67 -$ docker-compose logs
68
69 -$ docker-compose logs -f
70 --브라우저에서 Refresh로 log가 계속 update되는 것을 확인
71
72
73 6. docker-compose top
74 $ docker-compose top
75 -지금 실행하고 있는 nginx의 Process를 확인
76
77
78 7. docker-compose stop
79 $ docker-compose stop
80 -stop뒤에 서비스 이름을 넣지 않으면 모든 서비스 중단
81
82 $ docker-compose ps
83 -중단된 서비스 확인
84 -State가 Exit됨을 확인
85
86
87 8. docker-compose start
88 -멈춘 서비스 다시 실행
89 $ docker-compose start
90
91
92 9. docker-compose exec
93 1)서비스 안에서 어떤 명령을 실행
94 -$ docker-compose exec nginx bash
95 /# touch TEST <---run과의 명령어 차이를 구별하기 위해 빈 파일 하나 생성
96 /# ls <---TEST 파일 확인
97 /# exit
98
99
100 10. docker-compose run
101 1)해당 서비스에 컨테이너 하나 더 실행할 때
102 $ docker-compose run -d nginx
103 $ docker-compose ps <---하나 더 컨테이너 있는 것을 확인
104 $ docker-compose run nginx bash
105 /# ls <--- TEST 파일 없음을 확인
106
107
108 11. docker-compose down
109 1)중지하고 삭제
110 $ docker-compose down
111
112 2)Error 발생시
113 $ docker compose down
114 [+] Running 1/0
115 X Network ubuntu_default Error 0.0s
116 failed to remove network ubuntu_default: Error response from daemon: error while removing network:
network ubuntu_default id a17bac259bb0bd387f51f3dea20382bdcf25ae086a1e366bc88af2bbebb0a5e6
has active endpoints
117
118 -다음의 명령을 사용
119 $ docker-compose down --remove-orphans
```

120	[+] Running 3/2		
121	✓ Container ubuntu-nginx-run-ad778dfe067a	Removed	0.2s
122	✓ Container ubuntu-nginx-run-95e8d6846106	Removed	0.0s
123	✓ Network ubuntu_default	Removed	0.1s