

1 Lab8. Docker Container Storage

1. MySQL DB Data 영구 보존하기
2. Web Data를 ReadOnly 서비스로 지원하기
3. Container 사이의 데이터 공유하기

1. MySQL DB Data 영구 보존하기

1){Host path}:{Container Mount Path}

```
$ mkdir demo
```

```
$ cd demo
```

```
$ ls <-- dbdata folder 없음 확인.
```

```
$ sudo docker run -d --name db ₩
```

```
-v ./dbdata:/var/lib/mysql ₩
```

```
-p 3306:3306 ₩
```

```
-e MYSQL_ROOT_PASSWORD=password mysql:5.7
```

```
$ sudo docker ps -a
```

```
$ sudo docker exec -it db /bin/bash
```

```
/# mysql -h localhost -u root -p
```

```
password:
```

```
mysql> show databases;
```

```
mysql> create database sample;
```

```
mysql> show databases;
```

```
mysql> exit
```

```
/#exit
```

```
$ cd dbdata
```

```
$ ls <---sample 폴더 확인
```

```
$ sudo docker rm -f db
```

```
$ sudo docker rmi -f mysql:5.7
```

```
$ ls <--- 여전히 sample 폴더 확인, 영구적 데이터 보존
```

```
$ sudo docker run -d --name db ₩
```

```
-v /dbdata:/var/lib/mysql ₩
```

```
-p 3306:3306 ₩
```

```
-e MYSQL_ROOT_PASSWORD=password mysql:5.7
```

```
$ sudo docker ps -a
```

```
$ sudo docker exec -it db /bin/bash
```

```
/# mysql -h localhost -u root -p
```

```
password:
```

```
mysql> show databases; <--- sample database 여전히 존재함.
```

2){Container Mount Path}

```
$ sudo docker run -d --name db ₩
```

```
-v /var/lib/mysql ₩
```

```
-p 3306:3306 ₩
```

```
-e MYSQL_ROOT_PASSWORD=password mysql:5.7
```

```
$ sudo docker inspect db
```

```
- "Mounts"> "Source" --> "/var/lib/docker/volumes/kddfk.../_data <--- 확인할 것  
, "Destination" 확인할 것
```

```
-Source Directory를 기술하지 않았을 경우
```

```
/var/lib/docker/volumes/UUID/_data 에 생김
```

```
62
63 $ su -
64 # cd /var/lib/docker/volumes/kddfk.../_data
65
66 -Container를 삭제해도 데이터 폴더는 남아있음.
67
68 # docker volume ls
69
70 -Volume 삭제시 한개씩 삭제할 경우
71   # docker volume rm UUID
72
73 -Volume 삭제시 한꺼번에 삭제할 경우
74   # docker volume prune
75   Are you sure you want to continue? [y/N] y
76
77 -현재 실행중인 Docker Container의 Volume은 남아있음.
```

```
78
79
80
81 2. Web Data를 ReadOnly 서비스로 지원하기
82 $ mkdir demo
83 $ cd demo
84 $ mkdir /webdata
85 $ cd /webdata
86 $ echo "<h1>Hello, World</h1>" > index.html
87 $ cat index.html
88
89 $ docker run -d --name webserver -p 80:80 \
90   -v /webdata:/usr/share/nginx/html:ro nginx:1.24
91
92 $ docker ps -a
93
94 -Web Browser로 확인
95
96 -index.html 수정하기
97   $ sudo nano /webdata/index.html
98   <h1>Hello, Docker</h1>
99
100 -수정한 내용을 다시 Web Browser로 확인할 것
```

```
101
102
103 3. Container 사이의 데이터 공유하기
104 $ df -h <---Disk 사용량 모니터링
105
106 $ mkdir demo
107 $ cd demo
108 $ cat df.sh
109   #!/bin/bash
110   mkdir ./webdata
111   while true
112   do
113       df -h / > ./webdata/index.html
114       sleep 10
115   done
116
117 $ chmod +x ./df.sh
118
119 $ cat Dockerfile
120   FROM          ubuntu:22.04
121   ADD            df.sh /bin/df.sh
122   RUN            chmod +x /bin/df.sh
```

```
123     ENTRYPOINT  ["/bin/df.sh"]
124
125 $ docker build -t df:latest .
126
127 $ docker images
128
129 $ docker run -v ./webdata:/webdata -d --name df df:latest
130 $ docker run -d --name webserver -p 80:80 -v ./webdata:/usr/share/nginx/html:ro nginx:1.24
```