

```

1 Lab. Docker Container Storage
2 1. MySQL DB Data 영구 보존하기
3 2. Web Data를 ReadOnly 서비스로 지원하기
4 3. Container 사이의 데이터 공유하기
5
6 1. MySQL DB Data 영구 보존하기
7 1){Host path}:{Container Mount Path}
8     $ mkdir demo
9     $ cd demo
10    $ ls <-- dbdata folder 없음 확인.
11
12    $ sudo docker run -d --name db \
13        -v /dbdata:/var/lib/mysql \
14        -p 3306:3306 \
15        -e MYSQL_ROOT_PASSWORD=password mysql:5.7
16
17    $ sudo docker ps -a
18
19    $ sudo docker exec -it db /bin/bash
20    /# mysql -h localhost -u root -p
21    password:
22    mysql> show databases;
23
24    mysql> create database sample;
25    mysql> show databases;
26    mysql> exit
27    /#exit
28
29    $ cd /dbdata
30    $ ls <---sample 폴더 확인
31
32    $ sudo docker rm -f db
33    $ sudo docker rmi -f mysql:5.7
34    $ ls <--- 여전히 sample 폴더 확인, 영구적 데이터 보존
35
36    $ sudo docker run -d --name db \
37        -v /dbdata:/var/lib/mysql \
38        -p 3306:3306 \
39        -e MYSQL_ROOT_PASSWORD=password mysql:5.7
40
41    $ sudo docker ps -a
42
43    $ sudo docker exec -it db /bin/bash
44    /# mysql -h localhost -u root -p
45    password:
46    mysql> show databases; <--- sample database 여전히 존재함.
47
48
49 2){Container Mount Path}
50    $ sudo docker run -d --name db \
51        -v /var/lib/mysql \
52        -p 3306:3306 \
53        -e MYSQL_ROOT_PASSWORD=password mysql:5.7
54
55    $ sudo docker inspect db
56        -"Mounts"> "Source" --> "/var/lib/docker/volumes/kddfk.../_data <--- 확인할 것
57        , "Destination" 확인할 것
58
59    -Source Directory를 기술하지 않았을 경우
60        /var/lib/docker/volumes/UUID/_data 에 생김
61
62    $ su -
63    # cd /var/lib/docker/volumes/kddfk.../_data
64
65    -Container를 삭제해도 데이터 폴더는 남아있음.
66
67    # docker volume ls

```

```
68 -Volume 삭제시 한개씩 삭제할 경우
69 # docker volume rm UUID
70
71 -Volume 삭제시 한꺼번에 삭제할 경우
72 # docker volume prune
73 Are you sure you want to continue? [y/N] y
74
75 -현재 실행중인 Docker Container의 Volume은 남아있음.
```

## 2. Web Data를 ReadOnly 서비스로 지원하기

```
80 $ mkdir demo
81 $ cd demo
82 $ mkdir /webdata
83 $ cd /webdata
84 $ echo "<h1>Hello, World</h1>" > index.html
85 $ cat index.html
86
87 $ docker run -d --name webserver -p 80:80 \
88   -v /webdata:/usr/share/nginx/html:ro nginx:1.14
89
90 $ docker ps -a
91
92 -Web Browser로 확인
93
94 -index.html 수정하기
95 $ sudo nano /webdata/index.html
96 <h1>Hello, Docker</h1>
97
98 -수정한 내용을 다시 Web Browser로 확인할 것
```

## 3. Container 사이의 데이터 공유하기

```
102 $ df -h <---Disk 사용량 모니터링
103
104 $ mkdir demo
105 $ cd demo
106 $ cat df.sh
107 #!/bin/bash
108 mkdir /webdata
109 while true
110 do
111     df -h / > /webdata/index.html
112     sleep 10
113 done
114
115 $ chmod +x ./df.sh
116
117 $ cat Dockerfile
118 FROM ubuntu:18.04
119 ADD df.sh /bin/df.sh
120 RUN chmod +x /bin/df.sh
121 ENTRYPOINT ["/bin/df.sh"]
122
123 $ docker build -t df:latest .
124
125 $ docker images
126
127 $ docker run -v /webdata:/webdata -d --name df df:latest
128 $ docker run -d --name webserver -p 80:80 -v /webdata:/usr/share/nginx/html:ro nginx:1.14
```