

```

1 1. hub.docker.com에 Container Upload & Download
2 1)Docker Hub에서 Image 검색하기
3   - 'httpd' 검색
4   $ su -
5   # docker search httpd
6   # docker pull httpd:latest
7   # docker images
8
9   # docker login
10  Username :
11  Password :
12  Login Succeeded
13
14 2)Docker Hub Login 상태 확인
15  -Syntax
16   $ docker info | grep Username
17  -만일 로그인되어 있지 않다면 아무것도 출력되지 않는다.
18
19   $ docker info | grep Username
20   Username: devtrainer
21
22 3)Docker Public Registry에 Push 하기
23   # docker tag httpd:latest {{dockerhub's 계정}}/httpd:latest
24   # docker images
25   # docker push {{dockerhub's 계정}}/httpd:latest
26
27  -Docker Hub에서 확인할 것
28
29
30 2. Private Registry 운영하기
31 1)registry Container
32
33   # docker ps -a <-- 동작중인 Container 없음을 확인
34
35   # docker run -dp 5000:5000 --restart always --name registry registry:2
36
37   # hostname
38   docker-ubuntu.example.com <---VirtualBox 에서 실행시
39   ip-10-0-10.23 <---AWS EC2에서 실행 시(Sample)
40
41 2)이름 3가지
42  -ubuntu:22.04 <---docker.com이 직접 관리하는 Official Image
43  -계정/ubuntu:22.04 <---일반 계정이 관리하는 Image
44  -localhost:5000/ubuntu:22.04
45
46
47 3)이름 확인하기
48   # docker images
49   # docker images httpd:latest
50   # docker tag httpd:latest localhost:5000/httpd:latest
51
52   # docker images --> 아래와 같이 3개의 Image ID가 같음.
53   $ docker images
54   REPOSITORY          TAG          IMAGE ID      CREATED      SIZE
55   devtrainer/httpd    latest      4b7fc736cb48  12 days ago  145MB
56   httpd               latest      4b7fc736cb48  12 days ago  145MB
57   localhost:5000/httpd latest      4b7fc736cb48  12 days ago  145MB
58   registry            2          8db46f9d7550  3 weeks ago  24.2MB
59
60   # docker push localhost:5000/httpd:latest
61
62
63 4)Private Registry 위치 확인하기
64   # docker exec -it registry sh
65   /# ls
66   /# cd /var/lib/registry/docker/registry/v2/repositories
67   /# ls <--- httpd 확인
68
69   # docker images localhost:5000/httpd

```