

Docker Command

Docker Hub 로그인&로그아웃

#docker login [옵션] [서버명]

옵션	설명
-u	사용자명
-p	패스워드
-e	이메일 주소

#docker logout [서버명]

#docker login

username :

*서버명을 입력하지 않으면 Docker Hub에 액세스

password :

#docker logout

```
root@ubuntu:/# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't
te one.
Username: soraland
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store
Login Succeeded
root@ubuntu:/# docker logout
Removing login credentials for https://index.docker.io/v1/
root@ubuntu:/#
```

#docker search centos

* Docker Hub에 공개된 'centos' 관련 docker 이미지 목록 검색

```
root@ubuntu:/# docker search centos
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
centos	The official build of CentOS.	6465	[OK]	
ansible/centos7-ansible	Ansible on Centos7	133		[OK]
consol/centos-xfce-vnc	Centos container with "headless" VNC session...	127		[OK]
jdeathe/centos-ssh	OpenSSH / Supervisor / EPEL/IUS/SCL Repos - ...	117		[OK]
centos/systemd	systemd enabled base container.	97		[OK]
centos/mysql-57-centos7	MySQL 5.7 SQL database server	87		
imagine10255/centos6-lnmp-php56	centos6-lnmp-php56	58		[OK]
tutum/centos	Simple CentOS docker image with SSH access	46		
kinogmt/centos-ssh	CentOS with SSH	29		[OK]
pivotaldata/centos-gpdb-dev	CentOS image for GPDB development. Tag names...	13		
guyton/centos6	From official centos6 container with full up...	10		[OK]
centos/tools	Docker image that has systems administration...	7		[OK]
drecom/centos-ruby	centos ruby	6		[OK]
pivotaldata/centos	Base centos, freshened up a little with a Do...	5		
pivotaldata/centos-gcc-toolchain	CentOS with a toolchain, but unaffiliated wi...	3		
pivotaldata/centos-mingw	Using the mingw toolchain to cross-compile t...	3		
darksheer/centos	Base Centos Image -- Updated hourly	3		[OK]
mamohr/centos-java	Oracle Java 8 Docker image based on Centos 7	3		[OK]
indigo/centos-maven	Vanilla CentOS 7 with Oracle Java Developmen...	2		[OK]
amd64/centos	The official build of CentOS.	2		
mcnaughton/centos-base	centos base image	1		[OK]
blacklabelops/centos	CentOS Base Image! Built and Updates Daily!	1		[OK]
pivotaldata/centos6.8-dev	CentosOS 6.8 image for GPDB development	0		
pivotaldata/centos7-dev	CentosOS 7 image for GPDB development	0		
smartentry/centos	centos with smartentry	0		[OK]

```
root@ubuntu:/#
```

#docker search webserver

* Docker Hub에 공개된 'webserver' 관련 docker 이미지 목록 검색

```
root@ubuntu:/# docker search webserver
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
nazarpc/webserver	WebServer (MariaDB, PHP-FPM, Nginx) composed...	31		[OK]
yejune/webserver	php nginx phalcon webserver	3		[OK]
bitbull/webserver	Webserver image with optimized Apache webser...	2		
campanda/webserver-mock	Customizable HTTP/HTTPS webserver to be used...	2		[OK]
suchitagarwal/webserver	Webserver base image	1		
klambt/webserver	Apache 2.4.25+, mod_brotli, Google PageSpeed...	1		[OK]
healthplatforms/webserver	Webservers Apache 2.4 + php 5.6	1		
itisfoundation/webserver		1		
pcrespov/webserver		0		
versantus/webserver-drupal	drupal webserver	0		[OK]
anderegg/webserver		0		
eguilhon/webserver-hub		0		
gettestedcovid19/webserver	https://get-tested-covid19.org webserver	0		
nazarpc/webserver-apps	Carefully crafted images that are intended t...	0		[OK]
ak413/webserver		0		
opsway/webserver	Webserver docker images	0		[OK]
jack1007533560/webserver		0		
physk/webserver		0		
odeimaiz/webserver		0		
webserverhu/spamassassin	Mail filter to identify spam using a wide ra...	0		[OK]
octarinesec/webserver_for_tests		0		
brightideainc/webserver-newrelic	Webserver with New Relic with PHP 7.3 with C...	0		
stefanscherrer/webserver-windows	A minimal webserver written in Go to run in ...	0		
hubhk/webserver		0		
suhuruli/webserver		0		

```
root@ubuntu:/#
```

```

root@ubuntu:/# docker search webserver
NAME                                DESCRIPTION                                STARS    OFFICIAL    AUTOMATED
nazarpc/webserver                  WebServer (MariaDB, PHP-FPM, Nginx) compos...  31       [OK]
yejune/webserver                   php nginx phalcon webserver                 3        [OK]

```

항목	설명
NAME	Docker Image명 (예) <u>nazarpc/webserver</u> 저장소명(사용자명) / 이미지명
DESCRIPTION	Docker Image 설명
STARS	해당 이미지가 받은 별(star) 수 별 수가 많을 수록 인기 있는 docker 이미지임을 알 수 있음
OFFICIAL	공식 이미지 여부
AUTOMATED	Dockerfile을 기반으로 자동 생성된 이미지 여부

#docker search ubuntu

#docker search centos

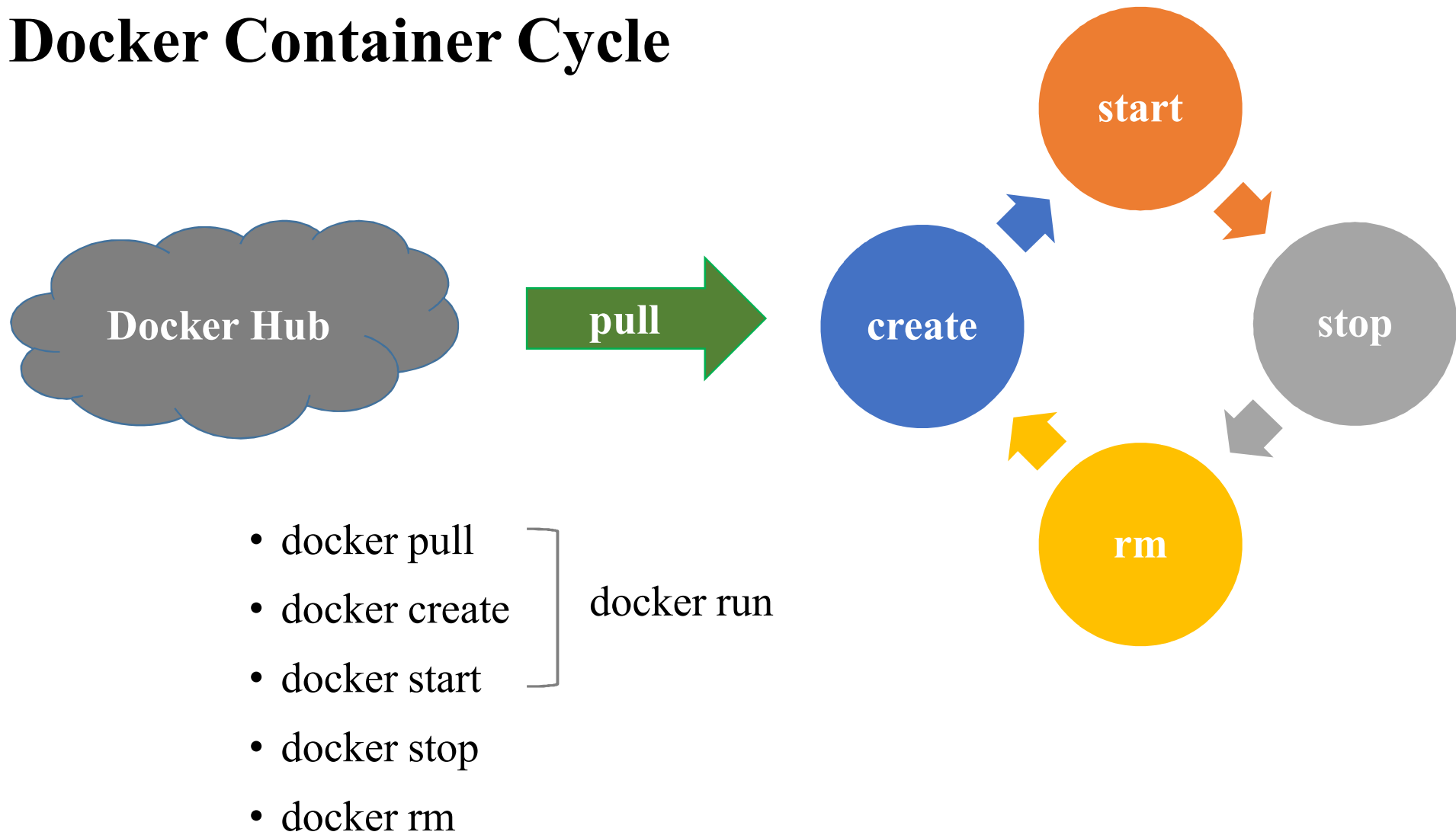
#docker search fedora

#docker search mysql

#docker search wordpress

#docker search java

Docker Container Cycle



Docker 기본 명령어

- Container 생성
- Container 구동
- Container 접속
- Container 중지
- Container 삭제
- Container 재시작
- Container 일시정지 및 재시작

1) Container 생성과 실행

- Docker image pull, docker container create, docker container start 기능을 하나로 합친 것
- 컨테이너 생성과 실행 명령어 : **docker run**

#docker run [옵션] <이미지명>[:태그명][값]

옵션 형식	설명
--name 컨테이너명	컨테이너이름 지정
-p 호스트포트번호:컨테이너포트번호	포트번호 지정
-v 호스트디스크:컨테이너디렉터리	볼륨을 마운트함
--net=네트워크이름	컨테이너를 네트워크에 연결
-e 환경변수명=값	환경변수를 설정
-d	백그라운드로 실행
-i	컨테이너에 터미널(키보드)를 연결
-t	특수 키를 사용 가능하도록 함

2) Container 목록 확인

- 컨테이너 목록 확인 명령어 : `docker ps`

#docker ps [옵션]

옵션	설명
-a	구동, 중지 상태의 모든 컨테이너 표시
-l	마지막에 구동된 컨테이너 표시
-q	컨테이너 ID만 표시
-f	목록에 표시할 컨테이너를 필터링
-s	파일 사이즈 표시

#docker ps

#docker ps -a

```
root@docker:/# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
root@docker:/# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED          STATUS          PORTS   NAMES
26ec0da67218   centos    "/bin/bash"   About a minute ago   Exited (0) 24 seconds ago   TEST01
```

항목	설명
Container ID	컨테이너 ID
Image	컨테이너 기반이 된 IMAGE
COMMAND	컨테이너에서 실행 중인 명령어
CREATED	컨테이너 생성 후 경과 시간
STATUS	컨테이너 상태 (restarting running paused exited)
PORTS	할당된 포트
NAMES	컨테이너 명

#docker ps

#docker ps -a

#docker start 662c

#docker ps

#docker attach 662c

- 컨테이너 이름 대신 ID 사용 가능
- ID가 너무 길때 앞의 2~3자만 입력해도 됨

```
root@ubuntu:/# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
root@ubuntu:/# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
662c059f9350        centos              "/bin/bash"        6 minutes ago       Exited (0) 6 minutes ago              TEST01
1a92731e3701        ubuntu:14.04        "/bin/bash"        18 minutes ago      Exited (127) 8 minutes ago           vigorous_vt
vesvaraya
root@ubuntu:/# docker start 662c
662c
root@ubuntu:/# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
662c059f9350        centos              "/bin/bash"        11 minutes ago      Up 4 seconds                TEST01
root@ubuntu:/# docker attach 662c
[root@662c059f9350 /]#
```

3) Container 실행 중지

- 중지되어 있는 컨테이너 구동 명령어 : `docker stop`

`#docker stop <컨테이너명 또는 ID>`

4) Container 삭제

- 구동중인 컨테이너에 접속 명령어 : `docker rm`

`#docker rm [옵션] <컨테이너명 또는 ID>`

항목	설명
-f	구동중인 컨테이너를 강제 삭제

- 생성된 컨테이너가 많은 경우 한번에 생성되어 있는 모든 컨테이너 삭제

`docker container prune`

5) 이미지 삭제

- 이미지 삭제 명령어

#docker rmi [옵션] <이미지명>

옵션	설명
-f	이미지 강제 삭제

실습 1. Apache container 생성

- ❶ `docker run --name webservice -d httpd`
- ❷ `docker ps`
- ❸ `docker stop webservice`
- ❹ `docker ps`
- ❺ `docker ps -a`
- ❻ `docker rm webservice`



실습 2. 외부에서 Apache container에 접속

❶ `docker run --name apache -d -p 8080:80 httpd`

❷ `docker ps`

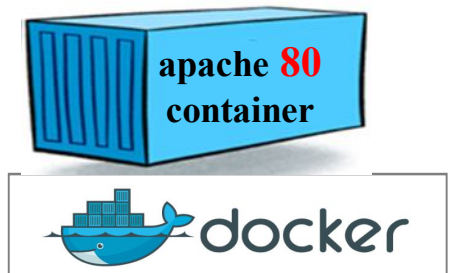
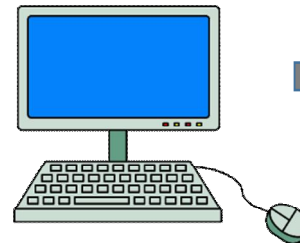
❸ `http://192.168.10.10:8080`

❹ `docker stop apache`

❺ `docker ps -a`

❻ `docker rm apache`

`http://192.168.10.10:8080`



6) Container 접속 명령어

- docker exec 또는 docker attach
 - 컨테이너가 실행되고 있을 때 사용
 - docker ps 를 입력했을 때 실행 중인 컨테이너가 없다면 실행할 수 없음

<code>docker exec <CONTAINER ID> <OPTIONS></code>	실행 중인 컨테이너에 명령어를 전달 (외부 -> 내부)
<code>docker exec -it <CONTAINER ID> /bin/bash</code>	실행 중인 컨테이너에 직접 들어가 명령어를 실행 (내부 접근)
<code>docker run -it --name <CONTAINER NAME> /bin/bash</code> <code>docker attach <CONTAINER ID></code>	실행 중인 컨테이너에 직접 들어가 명령어를 실행 (내부 접근) */bin/bash로 컨테이너를 생성해야 attach로 접속 가능

실습 3. docker attach로 Container 내부에 접근하여 명령어 실행

❶ docker run -it --name apache -d httpd /bin/bash

❷ docker attach apache

❸ ls

❹ pwd

❺ exit

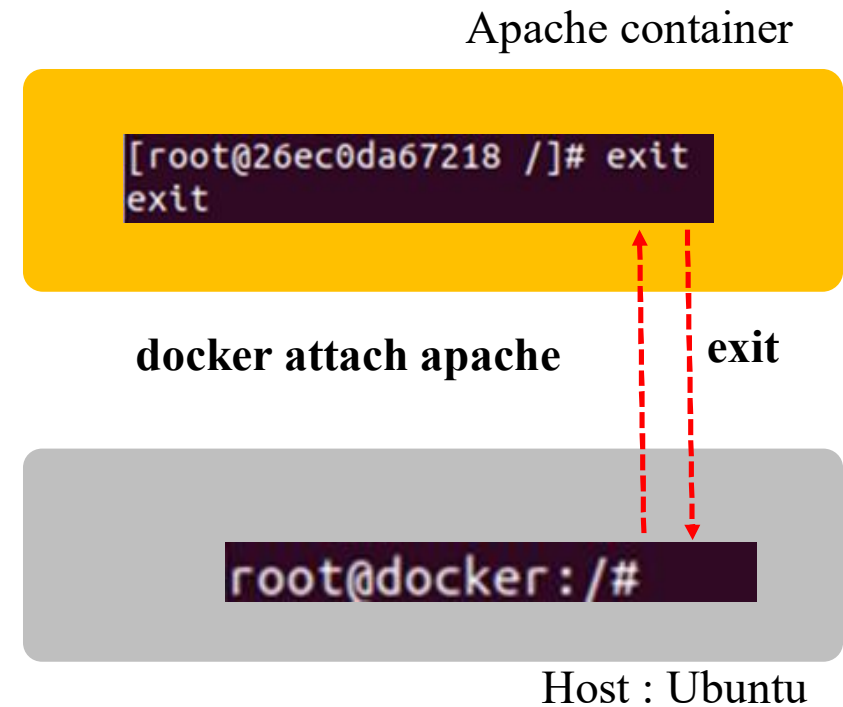
❻ docker start apache

❼ docker stop apache

❽ docker rm apache

❾ docker rmi apahce

***/bin/bash로 컨테이너를 생성해야 attach로 접속 가능**



실습 4. docker exec를 이용한 Container에 명령어 전달

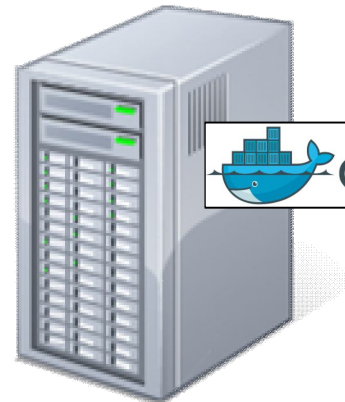
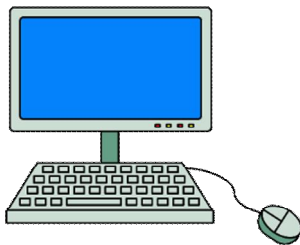
- ❶ `docker run --name apache -d httpd`
- ❷ `docker exec apache ls`
- ❸ `docker exec apache pwd`
- ❹ `docker exec apache cat /usr/local/apache2/htdocs/index.html`

실습 5. docker exec로 Container 내부에 접근하여 명령어 실행

- ❶ `docker exec -it apache /bin/bash`
- ❷ `ls`
- ❸ `pwd`
- ❹ `cat /usr/local/apache2/htdocs/index.html`
- ❺ `exit`

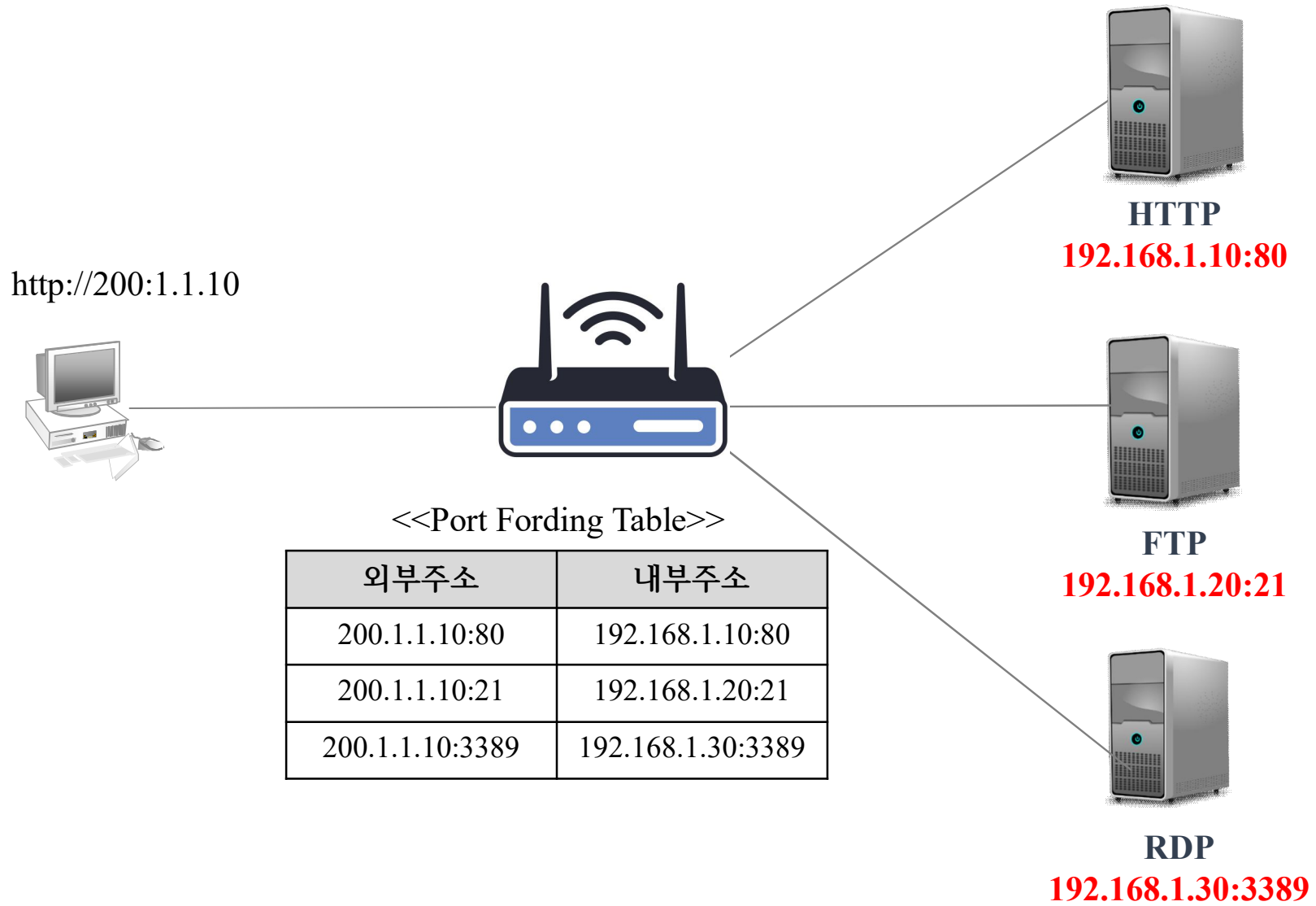
실습 6. Port Forwarding을 이용하여 외부에서 Apache 접속

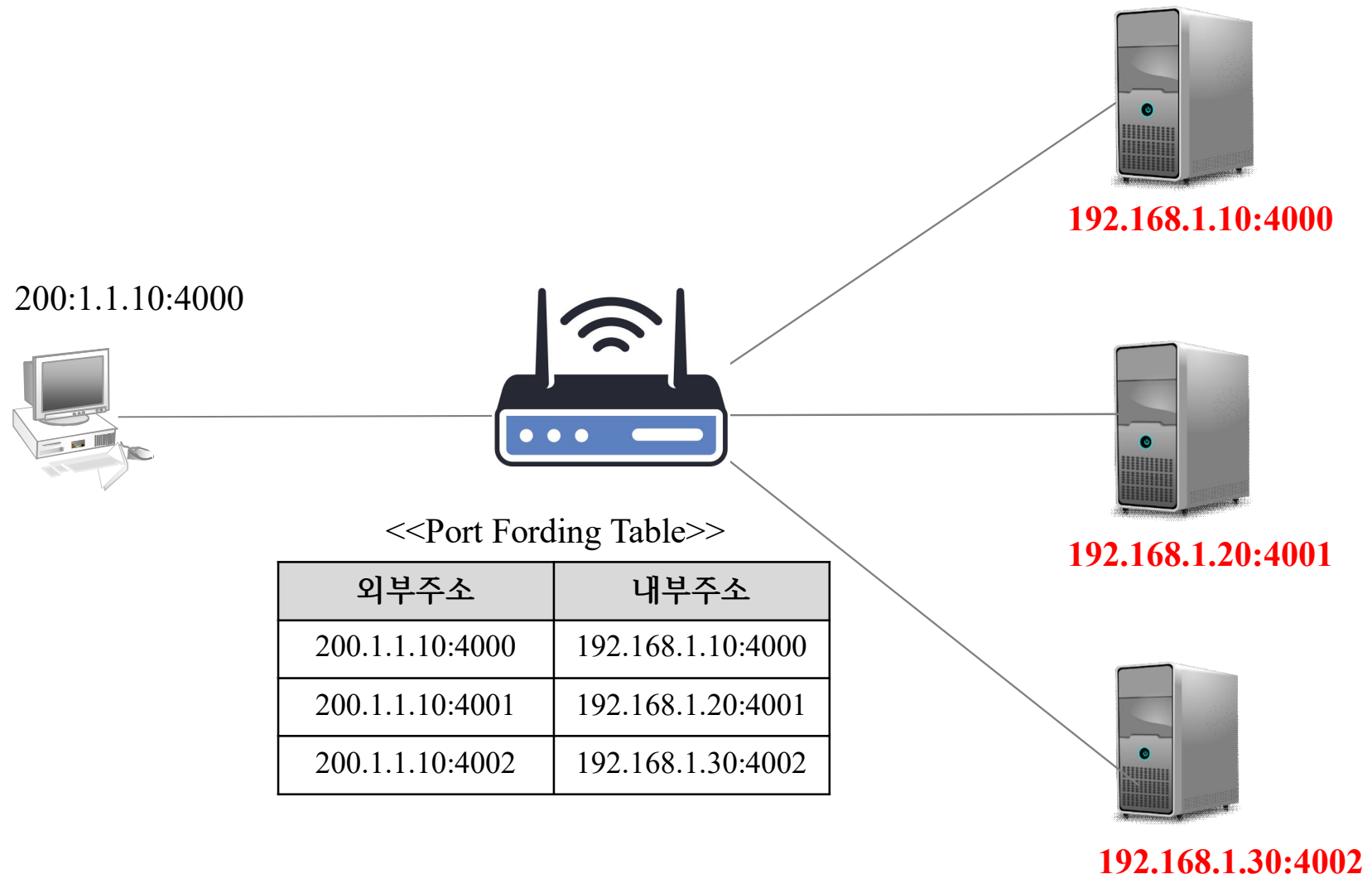
http://192.168.10.10:8080
http://192.168.10.10:8081
http://192.168.10.10:8082



192.168.10.10:8080	WS01:80
192.168.10.10:8081	WS02:80
192.168.10.10:8082	WS03:80







❶ `docker run --name WS01 -d -p 8081:80 httpd`

`docker run --name WS02 -d -p 8082:80 httpd`

`docker run --name WS03 -d -p 8083:80 httpd`

❷ `docker ps`

❸ `docker exec -it WS01 /bin/bash`

`cd /usr/local/apache2/htdocs`

`cat index.html`

`echo "<h1>Apache Server NO.1</h1>" > /usr/local/apache2/htdocs/index.html`

❹ `http://192.168.10.10:8080`

⑤ `docker stop WS01`

`docker stop WS02`

`docker stop WS03`

⑥ `docker ps`

⑦ `docker ps -a`

⑧ `docker rm WS01`

`docker rm WS02`

`docker rm WS03`

7) Container 일시 정지와 재시작

- 구동중인 컨테이너에서 실행중인 프로세스를 모두 일시 정지 명령어 : `docker pause`

`#docker pause <컨테이너명 또는 ID>`

- 일시 정지된 프로세스 재시작 명령어 : `docker unpause`

`#docker unpause <컨테이너명 또는 ID>`

8) Container 정지와 재시작

- 컨테이너 정지 명령어 : `docker stop`

`#docker stop <컨테이너명 또는 ID>`

- 컨테이너 재시작 명령어 : `docker restart`

`#docker restart [옵션] <컨테이너명 또는 ID>`

항목	설명
-t	컨테이너 재시작 시간을 지정(default는 10초)

9) Container 이름 변경

- 컨테이너 이름 변경 명령어 : `docker rename`

#`docker rename` 이전컨테이너명 변경컨테이너명

```
root@ubuntu:/# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
e05c598255d6       300e315adb2f      "/bin/bash"        4 minutes ago      Up 31 seconds        
compassionate_albattani

root@ubuntu:/# docker rename compassionate_albattani TEST02
root@ubuntu:/# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
e05c598255d6       300e315adb2f      "/bin/bash"        4 minutes ago      Up 56 seconds        
TEST02
root@ubuntu:/#
```

#docker images

#docker ps -a

#docker rmi centos

#docker images

```
root@ubuntu:/# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
centos               latest             300e315adb2f       3 months ago       209MB
ubuntu               14.04             df043b4f0cf1       6 months ago       197MB
root@ubuntu:/# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
1a92731e3701       ubuntu:14.04       "/bin/bash"        2 hours ago        Exited (127) 2 hours ago
root@ubuntu:/# docker rmi centos
Untagged: centos:latest
Untagged: centos@sha256:5528e8b1b1719d34604c87e11dcd1c0a20bedf46e83b5632cdeac91b8c04efc1
Deleted: sha256:300e315adb2f96afe5f0b2780b87f28ae95231fe3bdd1e16b9ba606307728f55
Deleted: sha256:2653d992f4ef2bfd27f94db643815aa567240c37732cae1405ad1c1309ee9859
root@ubuntu:/# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu               14.04             df043b4f0cf1       6 months ago       197MB
root@ubuntu:/#
```

10) Docker 버전확인

- Docker 버전
- Go 언어버전
 - * 도커는 Go 언어^{Go}로 개발
- OS 확인
- 서버와 클라이언트정보확인

#docker version

```
root@ubuntu:/# docker version
Client:
 Version:           19.03.8
 API version:       1.40
 Go version:        go1.13.8
 Git commit:        afacb8b7f0
 Built:             Wed Mar 11 23:42:35 2020
 OS/Arch:           linux/amd64
 Experimental:      false

Server:
 Engine:
  Version:          19.03.8
  API version:      1.40 (minimum version 1.12)
  Go version:       go1.13.8
  Git commit:       afacb8b7f0
  Built:            Wed Mar 11 22:48:33 2020
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.3.3-0ubuntu2
  GitCommit:
 runc:
  Version:          spec: 1.0.1-dev
  GitCommit:
 docker-init:
  Version:          0.18.0
  GitCommit:
```

11) Docker 실행환경 확인

#docker info

```
root@ubuntu:/# docker info
Client:
 Debug Mode: false

Server:
 Containers: 1
  Running: 1
  Paused: 0
  Stopped: 0
 Images: 3
 Server Version: 19.03.8
 Storage Driver: overlay2
  Backing Filesystem: <unknown>
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-
 Swarm: inactive
 Runtimes: runc
 Default Runtime: runc
 Init Binary: docker-init
 containerd version:
 runc version:
```

컨테이너 수

```
init version:
Security Options:
 apparmor
 seccomp
  Profile: default
 Kernel Version: 5.4.0-26-generic
 Operating System: Ubuntu 20.04 LTS
 OType: linux
 Architecture: x86_64
 CPUs: 1
 Total Memory: 1.914GiB
 Name: ubuntu
 ID: B15Q:T5JV:CNV5:JKCE:3LM3:2T7B:4R6V:
 Docker Root Dir: /var/lib/docker
 Debug Mode: false
 Registry: https://index.docker.io/v1/
 Labels:
 Experimental: false
 Insecure Registries:
  127.0.0.0/8
 Live Restore Enabled: false

WARNING: No swap limit support
```

Docker호스트명

Docker저장소명

12) 컨테이너 프로세스 확인

- 구동중인 컨테이너에서 실행 중인 프로세스 확인 확인 명령어 :

#docker top [옵션]

```
root@ubuntu:/# docker top e05c598255d6
UID          PID          PPID         C            STIME        TTY          TIME
```