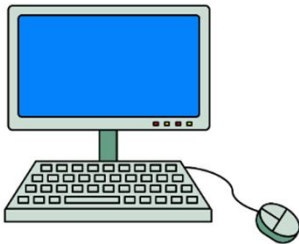


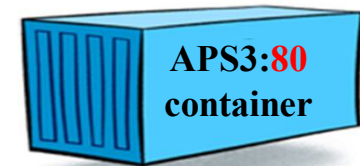
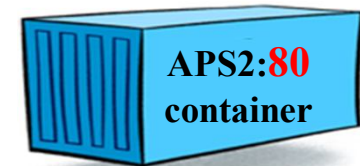
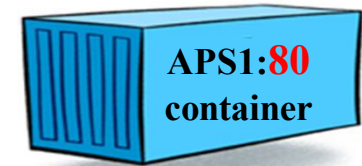
Container 생성/접근/삭제

실습 1. Apache Web Server Container 생성과 실행

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



192.168.10.10/24



Container name	APS1	APS2	APS3
Image name	httpd	httpd	httpd
Port No.	8081:80	8082:80	8083:80

❶ `docker run --name APS1 -d -p 8081:80 httpd`

❷ `docker images`

`docker ps`

❸ `docker run --name APS2 -d -p 8082:80 httpd`

`docker run --name APS3 -d -p 8083:80 httpd`

❹ `docker images`

`docker ps`

`docker ps -a`

⑤ `docker exec -it APSX/bin/bash`

`cd /usr/local/apache2/htdocs`

`cat index.html`

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

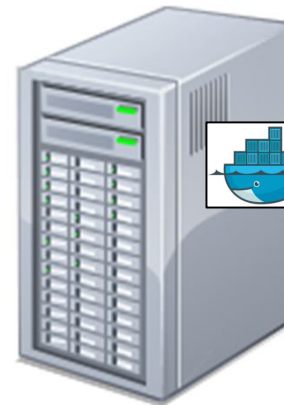
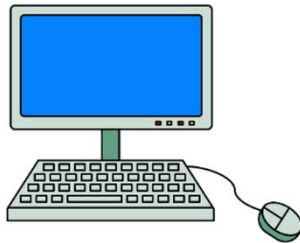
⑥ `http://192.168.10.10:8081`

`http://192.168.10.10:8082`

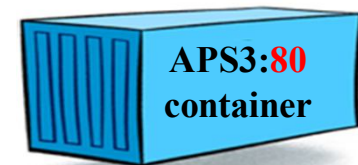
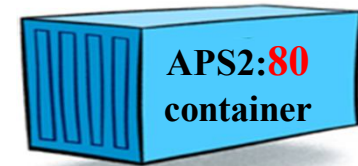
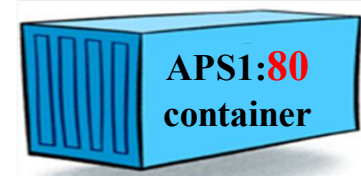
`http://192.168.10.10:8083`

실습 2. Nginx Web Server Container 생성과 실행

http://192.168.10.10:8084



Container name	NGX
Image name	nginx
Port No.	8084:80



❶ `docker run --name NGX -d -p 8084:80 nginx`

❷ `docker images`

`docker ps`

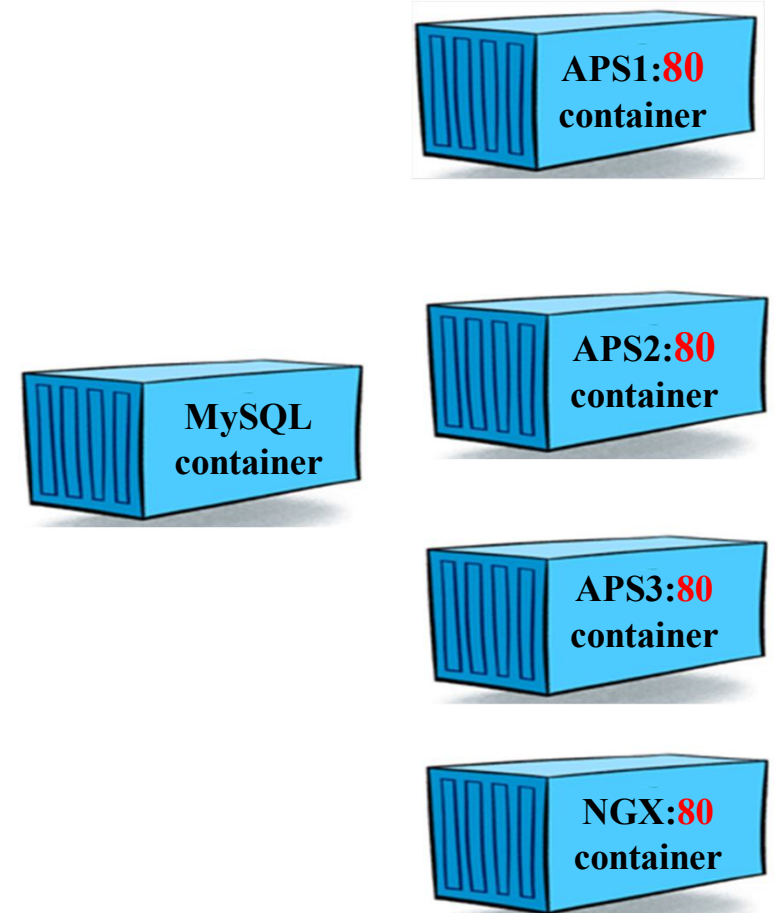
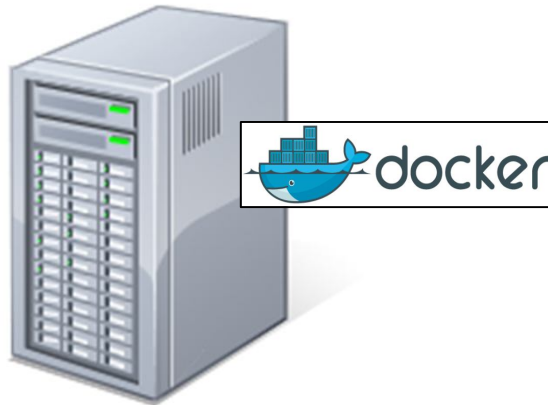
❸ `docker exec -it NXS /bin/bash`

`echo "<h1>Nginx Web Server </h1>" > /usr/share/nginx/html/index.html`

❹ `http://192.168.10.10:8084`

실습 3. MySQL Container 생성과 실행

Container name	MySQL
Image name	mysql
MySQL root Password	1234567



❶ `docker run --name MySQL --dit -e MYSQL_ROOT_PASSWORD=1234567 mysql`

❷ `docker images`

`docker ps`

❸ `docker exec -it MySQL /bin/bash`

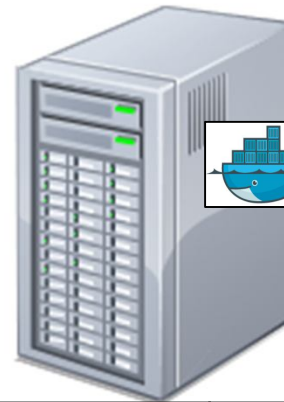
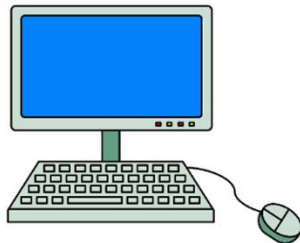
`mysql -u root -p`

`show databases;`

`exit`

실습 4. Container & Image 삭제하기

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 stop *containername*

❷ docker ps

docker ps -a

❸ docker rm *containername*

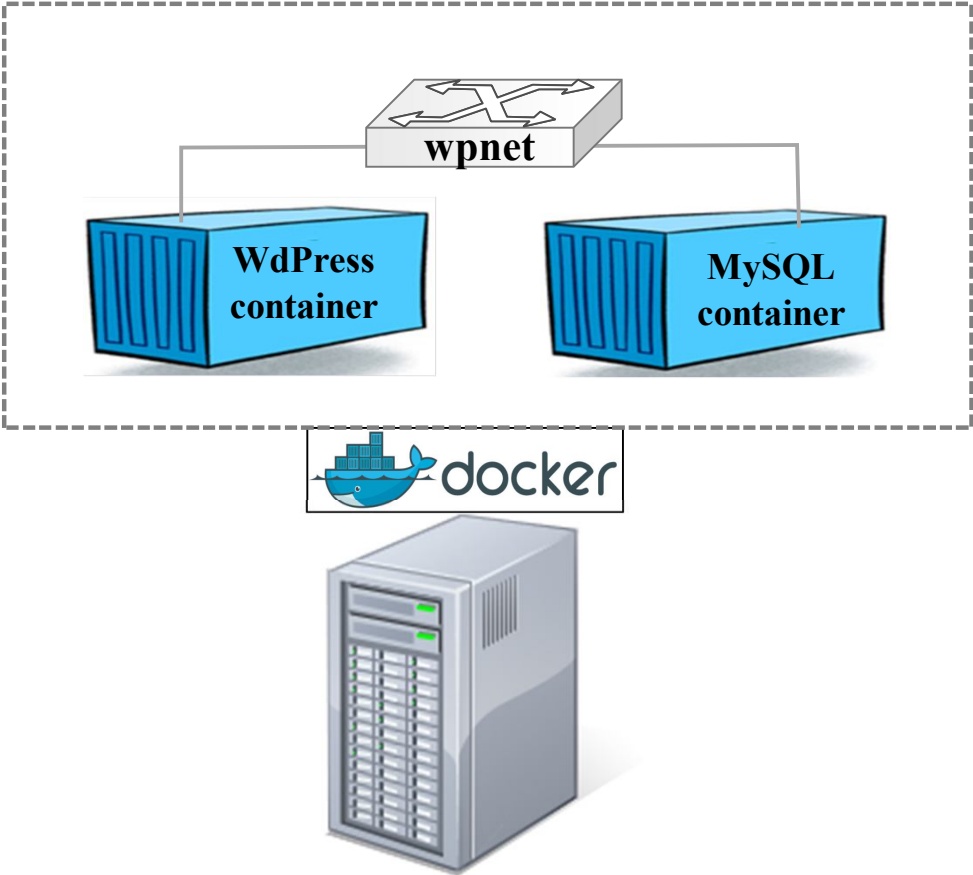
❹ docker images

docker rmi *imagename*

WordPress Site 구축

<< Wordpress를 사용한 블로그 서버 구축 >>

Container name	wdserver	mysqlserver
Image name	wordpress	mysql
Network name	wpnet	



❶ Network 생성

```
docker network create wpnet
```

❷ MySQL Container 생성

```
docker run --name mysqlserver -dit --net=wpnet  
-e MYSQL_ROOT_PASSWORD=1234567  
-e MYSQL_DATABASE=wordpressdb  
-e MYSQL_USER=test  
-e MYSQL_PASSWORD=1234 mysql  
--character-set-server=utf8mb4  
--collation-server=utf8mb4_unicode_ci  
--default-authentication-plugin=mysql_native_password
```

③ WordPress Container 생성

```
docker run --name wdserver --dit --net=wpnet -p 8085:80
```

```
    -e WORDPRESS_DB_HOST=mysqlserver
```

```
    -e WORDPRESS_DB_NAME=wordpressdb
```

```
    -e WORDPRESS_DB_USER=test
```

```
    -e WORDPRESS_DB_PASSWORD=1234 wordpress
```

④ docker ps

⑤ http://192.168.10.10:8085



Basa Jawa
ქართული
Taqbaylit
Қазақ тілі
ភាសាខ្មែរ
ಕನ್ನಡ
한국어
کوردی
Кыргызча
ພາສາລາວ
Lietuvių kalba
Latviešu valoda
Македонски јазик
മലയാളം
Монгол
मराठी
Bahasa Melayu
සමාන
Norsk bokmål
नेपाली
Nederlands



계속

환영합니다

유명한 워드프레스 5분 설치 절차에 오신 것을 환영합니다! 아래의 정보를 입력하기만 하면 세계에서 가장 확장성 있고 강력한 개인 발행 플랫폼을 사용할 수 있습니다.

정보가 필요합니다

다음 정보를 제공해주세요. 걱정하지 마세요. 이 설정을 나중에 언제든지 바꿀 수 있습니다.

사이트 제목

Docker Blog

사용자명

Gildong

사용자명은 알파벳, 숫자, 스페이스, 밑줄, 하이픈, 마침표, @ 기호만 가능합니다.

비밀번호

bSJ09DxTcyTdQmIj!t

숨기기

강함

중요: 로그인할 비밀번호가 필요할 것입니다. 안전한 위치에서 저장해주세요.

이메일 주소

계속하기 전에 이메일 주소를 다시 확인하세요.

검색 엔진 가시성

☐ 검색 엔진이 이 사이트를 검색하는 것을 차단

이 요청이 받아들여지는 것은 전적으로 검색 엔진에 좌우됩니다.

워드프레스 설치



성공!

워드프레스가 설치되었습니다. 감사합니다. 즐거운 시간을!

사용자명 Gildong

비밀번호 *선택한 비밀번호*

[로그인](#)