


SeSAC 4기, 

웹 풀스택 과정 Docker 수업

WITH 팀 리처드

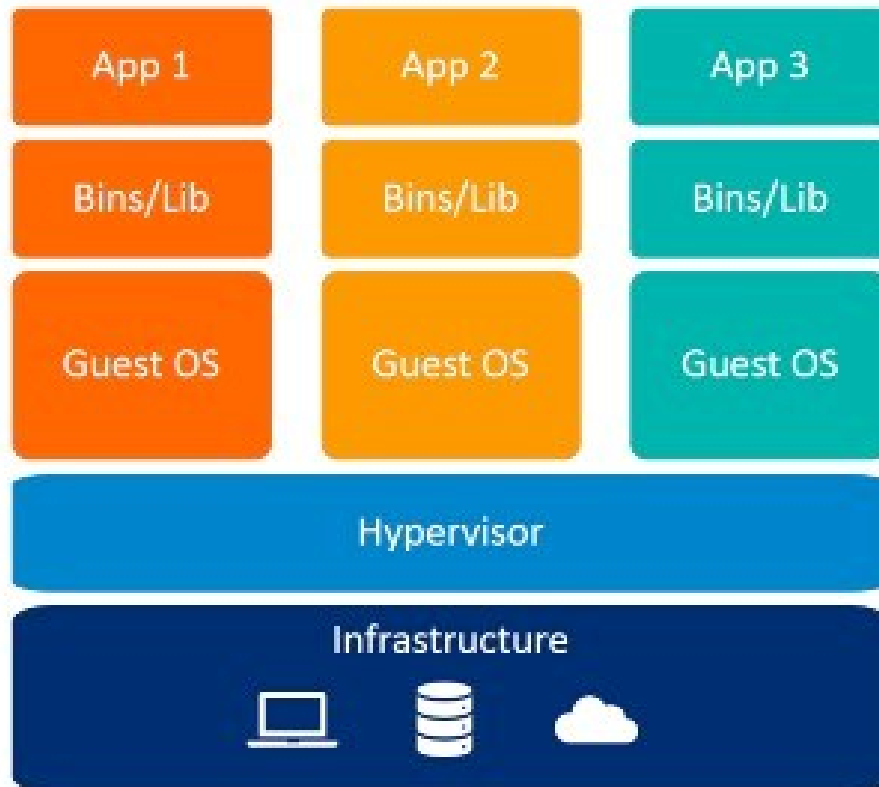


Container

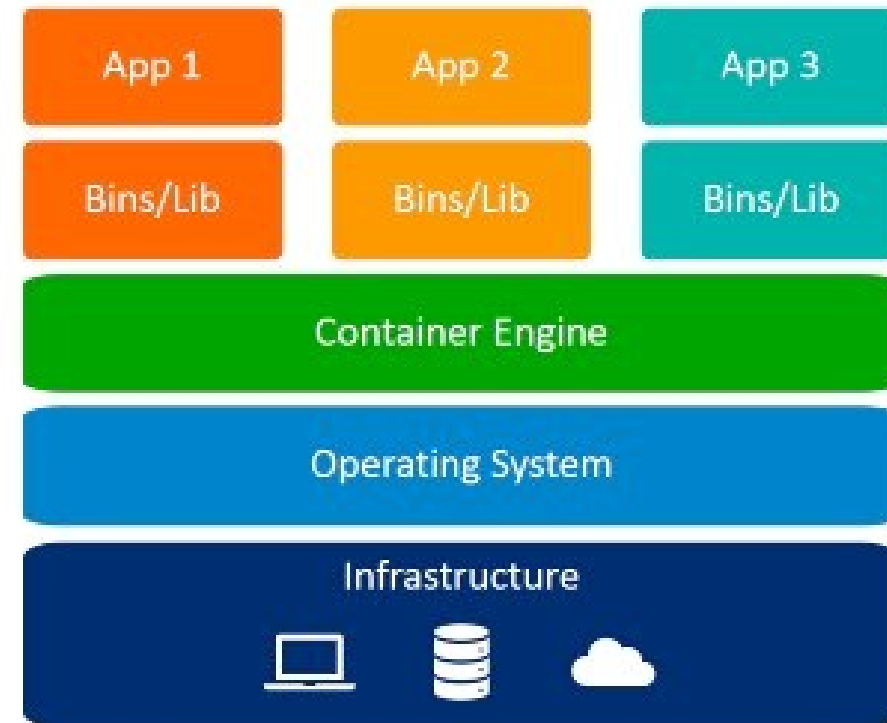
Container란?

- Linux 기반의 기술 중 하나로 필요한 라이브러리와 애플리케이션을 모아 별도의 서버처럼 구성한 것
- 네트워크 설정, 환경 변수 등의 시스템 자원을 독립적으로 소유하고 있다.



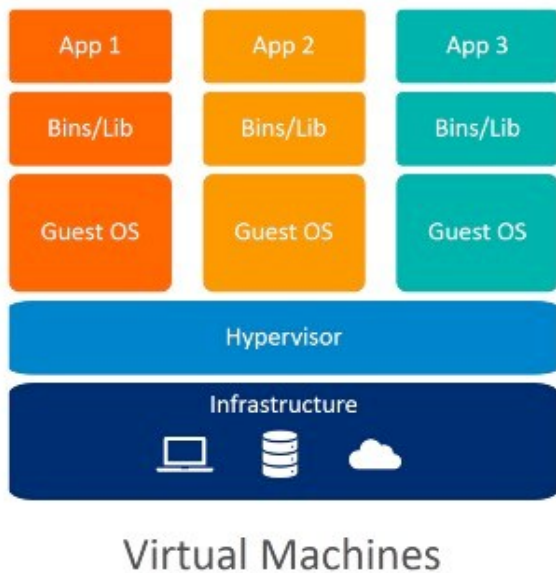


Virtual Machines



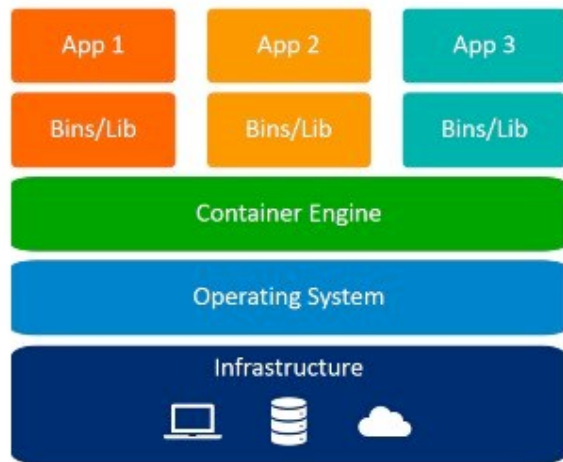
Containers

VM 가상화 플랫폼



- Hypervisor 라는 것을 통해 컴퓨터가 가지고 있는 리소스들에 대해 VM 별로 배분하고, 각 VM이 독립적인 os를 갖고 동작한다.
- 확장성 ↓
 - 독립적인 플랫폼을 하나 증가시킬 때마다 OS를 계속 만들어야 한다.
- 메모리나 자원이 유동적으로 관리되는 것이 아닌 처음부터 정해놓고 실행하기 때문에 비효율적이다.

Container 플랫폼



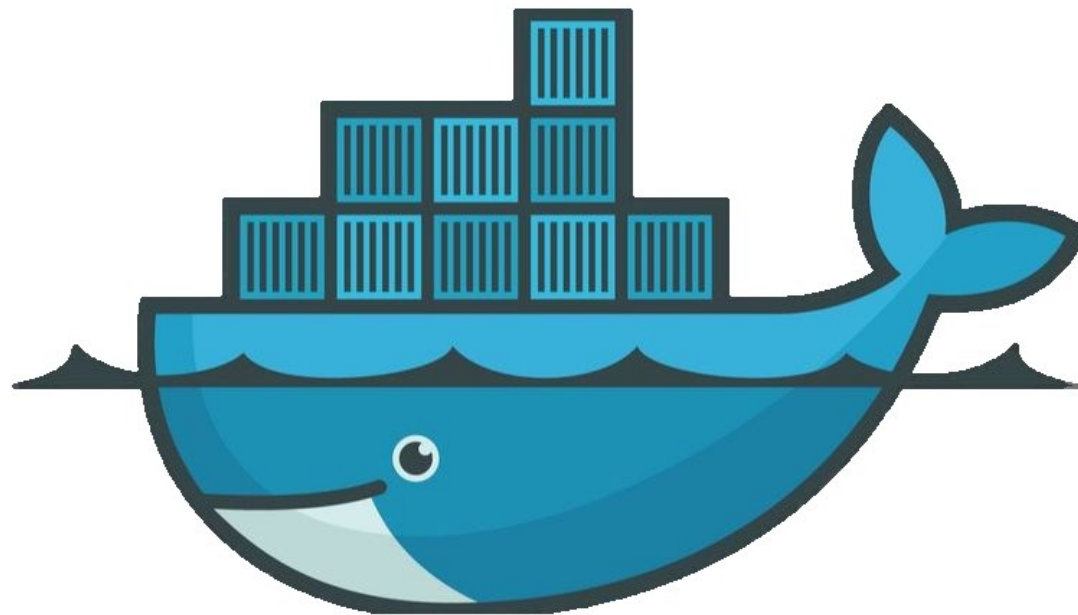
Containers

- 하나의 Host OS 위에서 각각의 독립적인 프로그램처럼 관리되고 실행된다.
- 확장성 ↑ 속도 ↑
 - 불필요한 os 만드는 작업이나 인프라를 독립적으로 나눌 필요 없이 실행할 수 있다.
- VM 보다 비교적 빠르게 세팅할 수 있고 VM과 비교해 크기가 작기 때문에 메모리를 덜 소모한다.

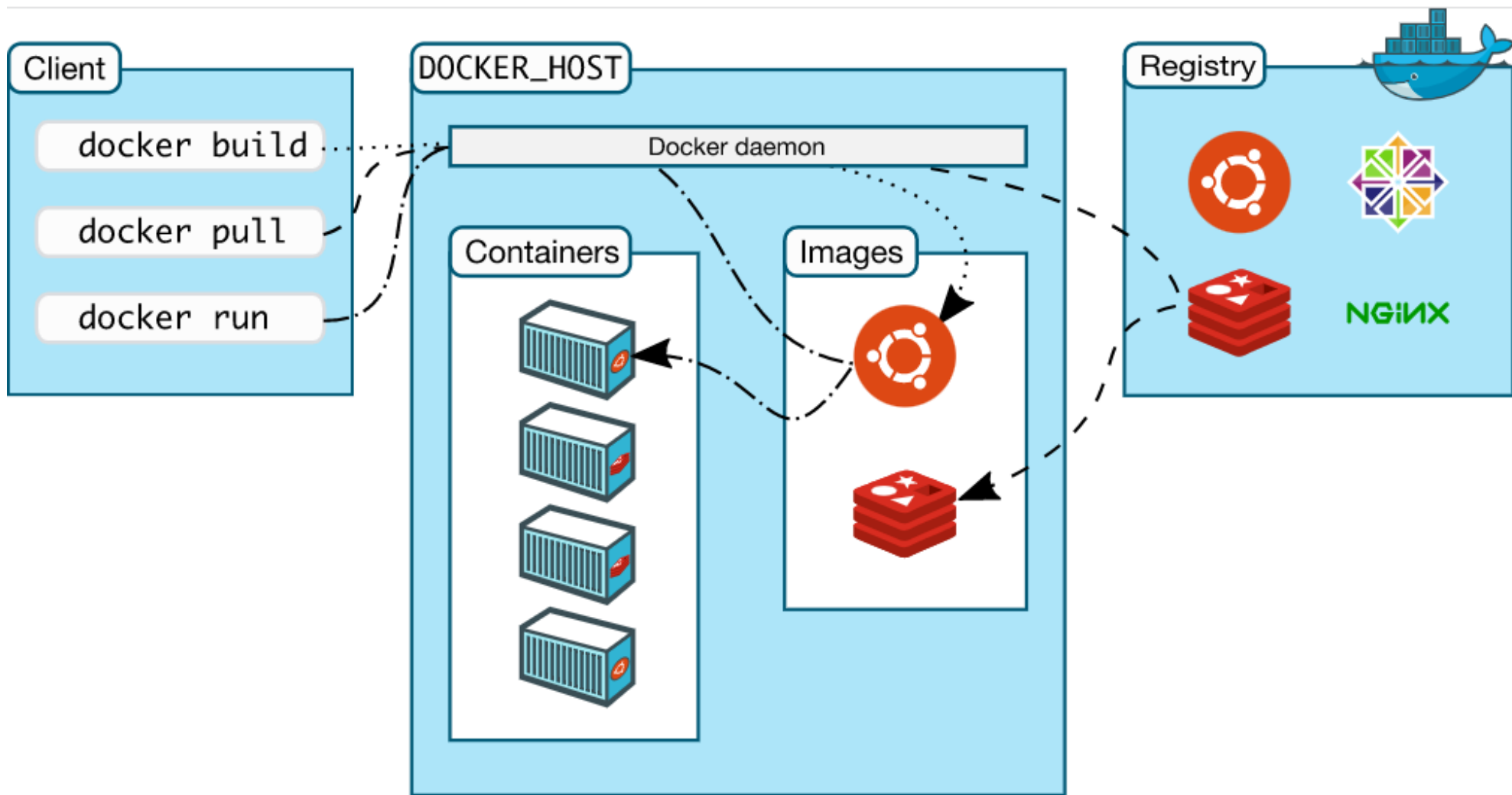
Docker

Docker란?

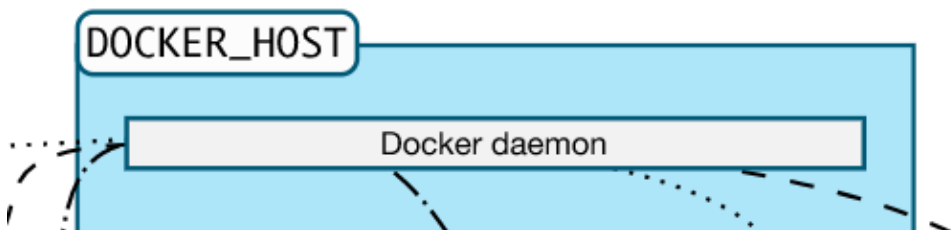
- Go 언어로 작성된 **Linux 컨테이너 기반**으로 하는 오픈소스 가상화 플랫폼
- 애플리케이션 실행 환경을 코드로 작성할 수 있고, 하나의 Host OS를 공유해 사용할 수 있다.



Docker 아키텍처

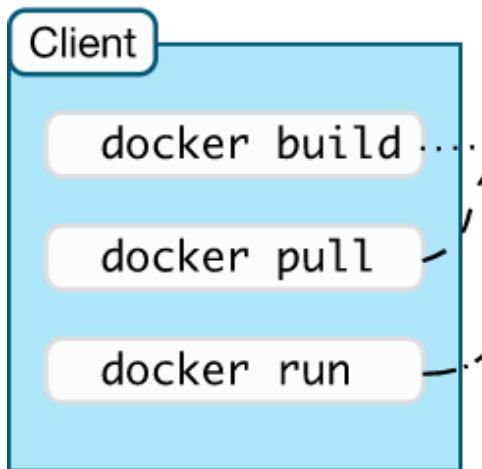


Docker Daemon



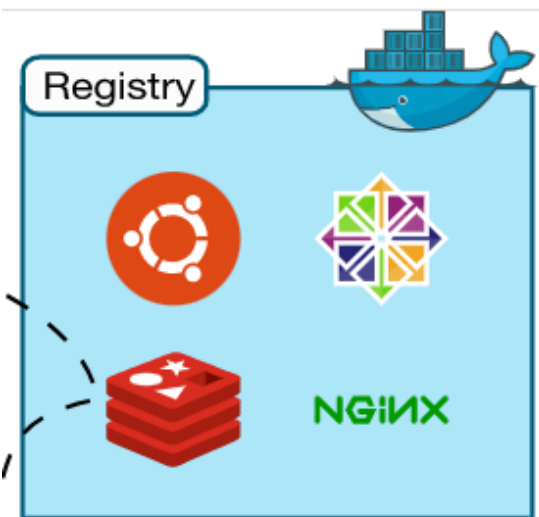
- Docker API 요청 수신, Image, Container, 네트워크와 같은 Docker 객체 및 Docker 서비스를 관리한다.

Docker Client



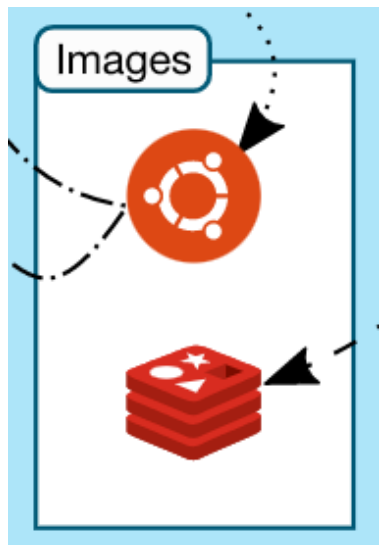
- Docker 사용자가 Docker와 상호작용하기 위한 방법으로 기본적인 Docker 명령어를 통해 Docker Daemon과 통신한다.
- **docker run** 과 같은 명령어를 이용해 Docker을 실행할 수 있다.

Docker Registry



- Docker 이미지 저장한다.
- Docker Hub이라는 공용 Registry와 개인 private한 Registry에서 Docker 이미지를 관리한다.
- 일반적으로는 Docker Hub이라는 공용 Registry에서 실행한다.

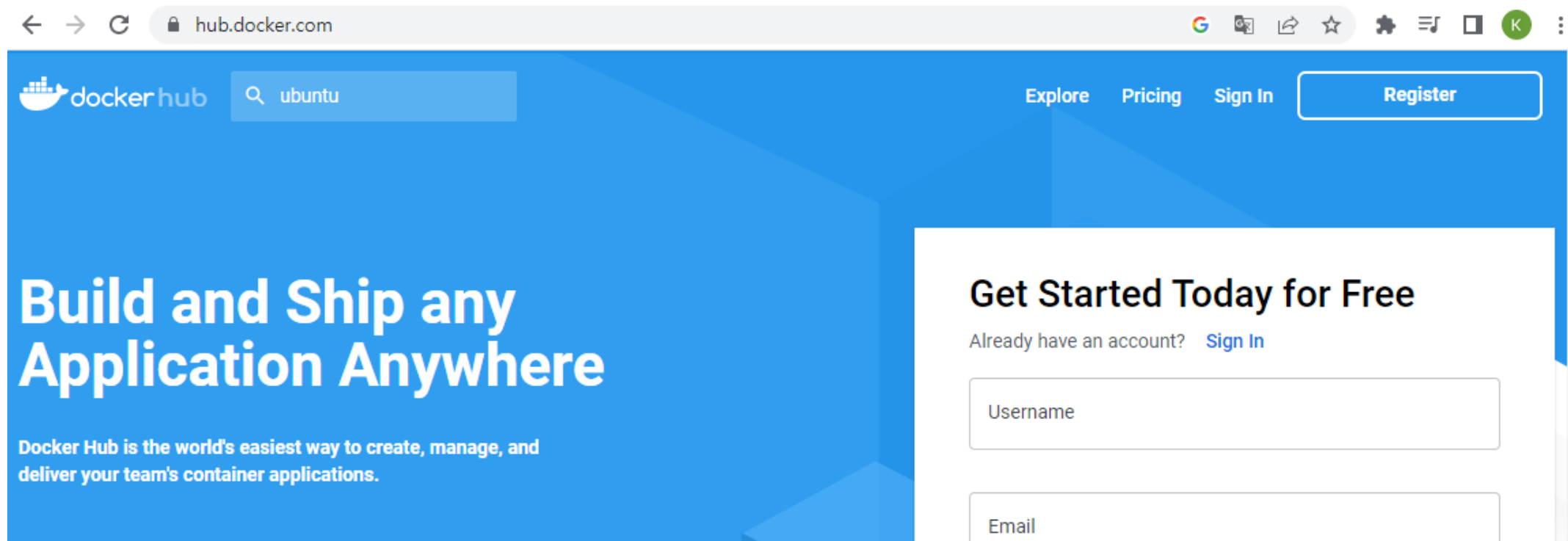
Docker Image



- 이미지 = 실행파일
- 컨테이너 실행에 필요한 파일과 설정값 등을 포함하고 있는 것으로 **Immutable(불변적)** 하다.
- 이미지는 Docker Hub에 저장되고, 그곳에서 다운받아 사용할 수 있다.

Docker Hub

<https://hub.docker.com/>



The screenshot shows the Docker Hub website in a web browser. The address bar displays 'hub.docker.com'. The page has a blue header with the Docker Hub logo, a search bar containing 'ubuntu', and navigation links for 'Explore', 'Pricing', 'Sign In', and a 'Register' button. The main content area features a large blue background with the text 'Build and Ship any Application Anywhere' and a subtext 'Docker Hub is the world's easiest way to create, manage, and deliver your team's container applications.' On the right side, there is a white box titled 'Get Started Today for Free' with a link 'Already have an account? Sign In' and two input fields for 'Username' and 'Email'.

hub.docker.com

dockerhub ubuntu

Explore Pricing Sign In Register

Build and Ship any Application Anywhere

Docker Hub is the world's easiest way to create, manage, and deliver your team's container applications.

Get Started Today for Free

Already have an account? [Sign In](#)

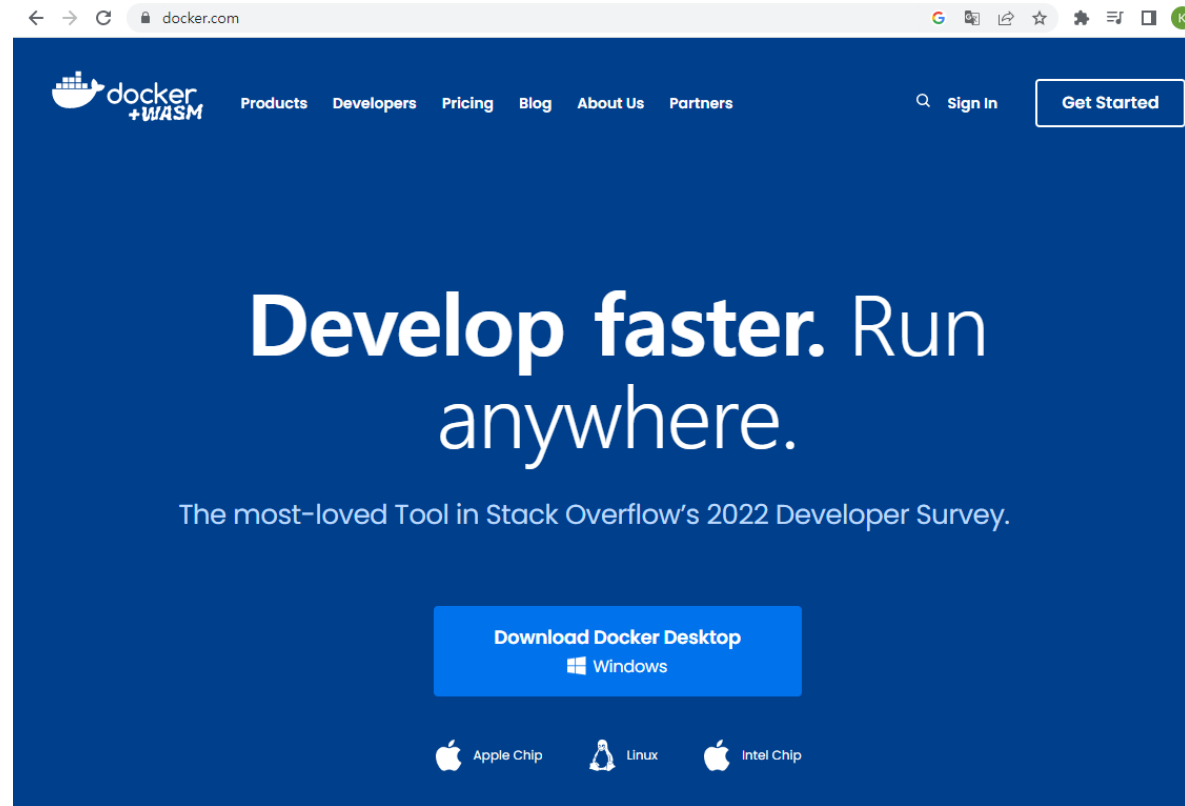
Username

Email

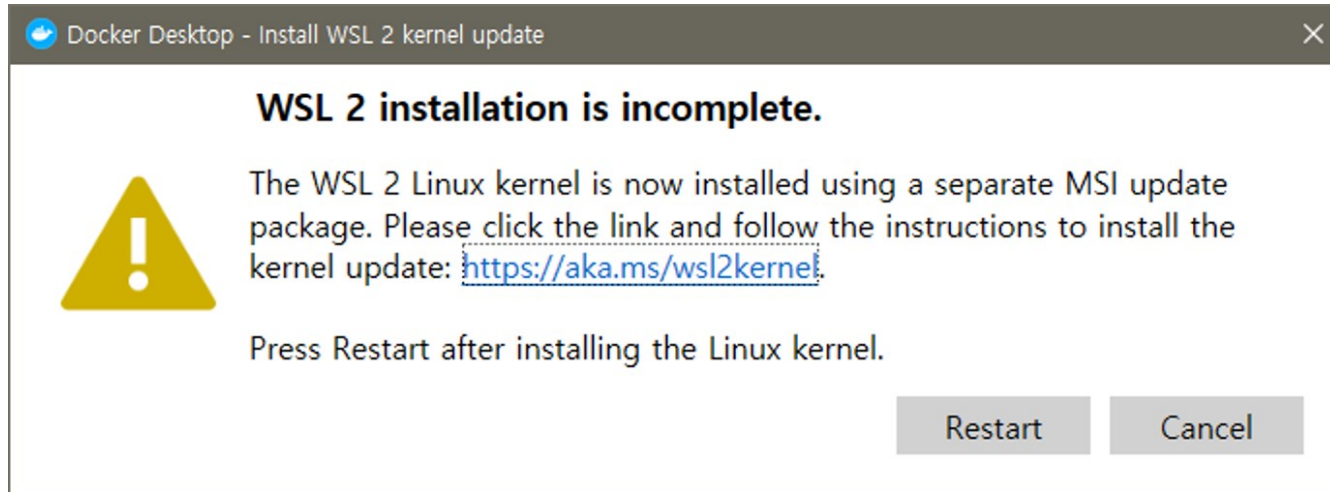
Docker 설치

Docker 설치하기

- <https://www.docker.com/>



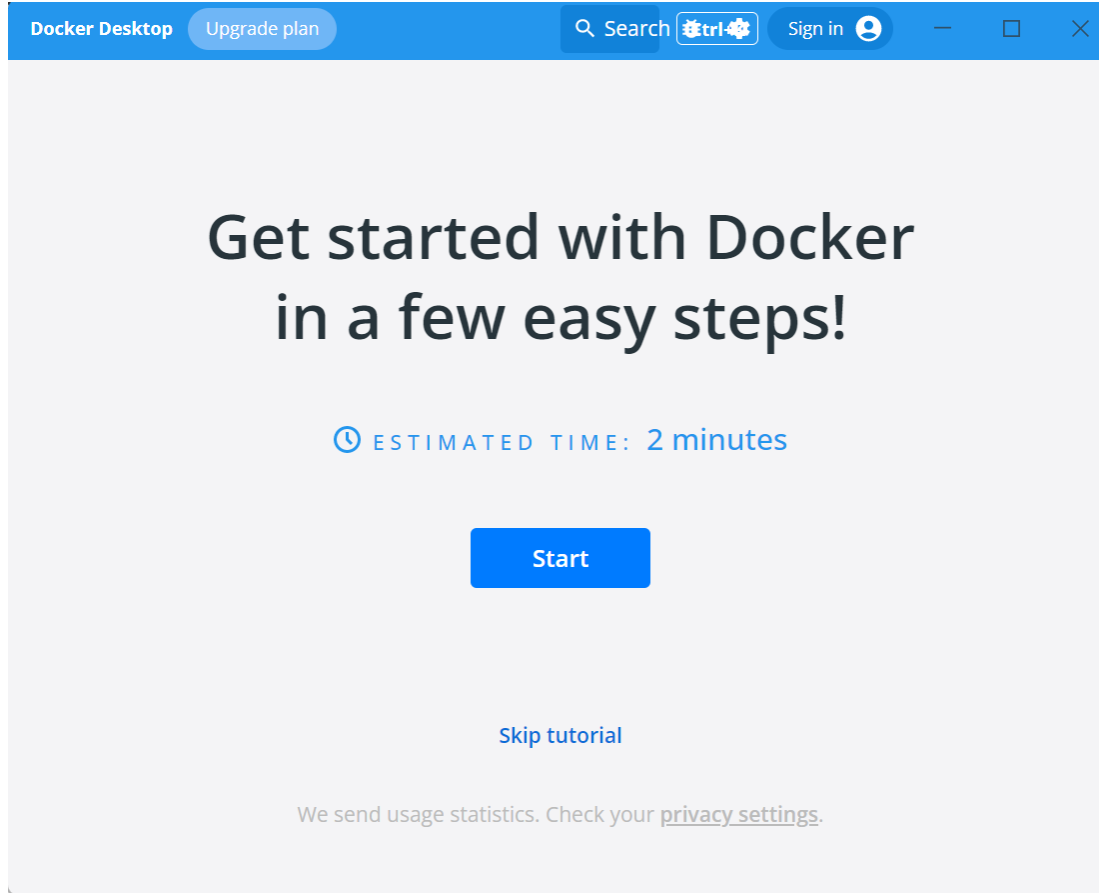
Window 오류 해결



- 위의 오류가 발생한다면? 아래 링크로 참고해 해결하기

[\[Docker\] 설치 \(notion.site\)](#)

Docker 설치 확인

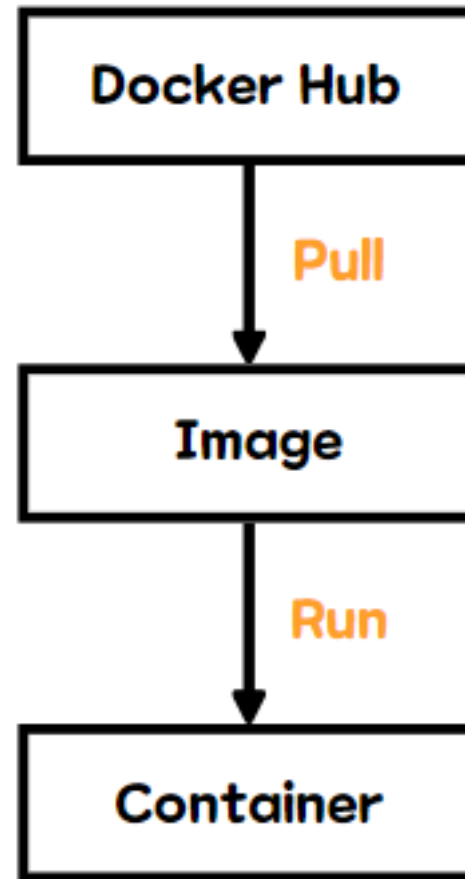


docker images
명령어로 확인하기


```
C:\Users\W...>docker images
REPOSITORY    TAG       IMAGE ID   CREATED   SIZE
```

Docker 실행

Docker 실행 순서




Docker Image Pull



Explore Pricing Sign In [Register](#)

Explore > Official Images > node




node

DOCKER OFFICIAL IMAGE • 1B+ • 10K+

Node.js is a JavaScript-based platform for server-side and networking applications.

[Overview](#) [Tags](#)

```
docker pull node
```



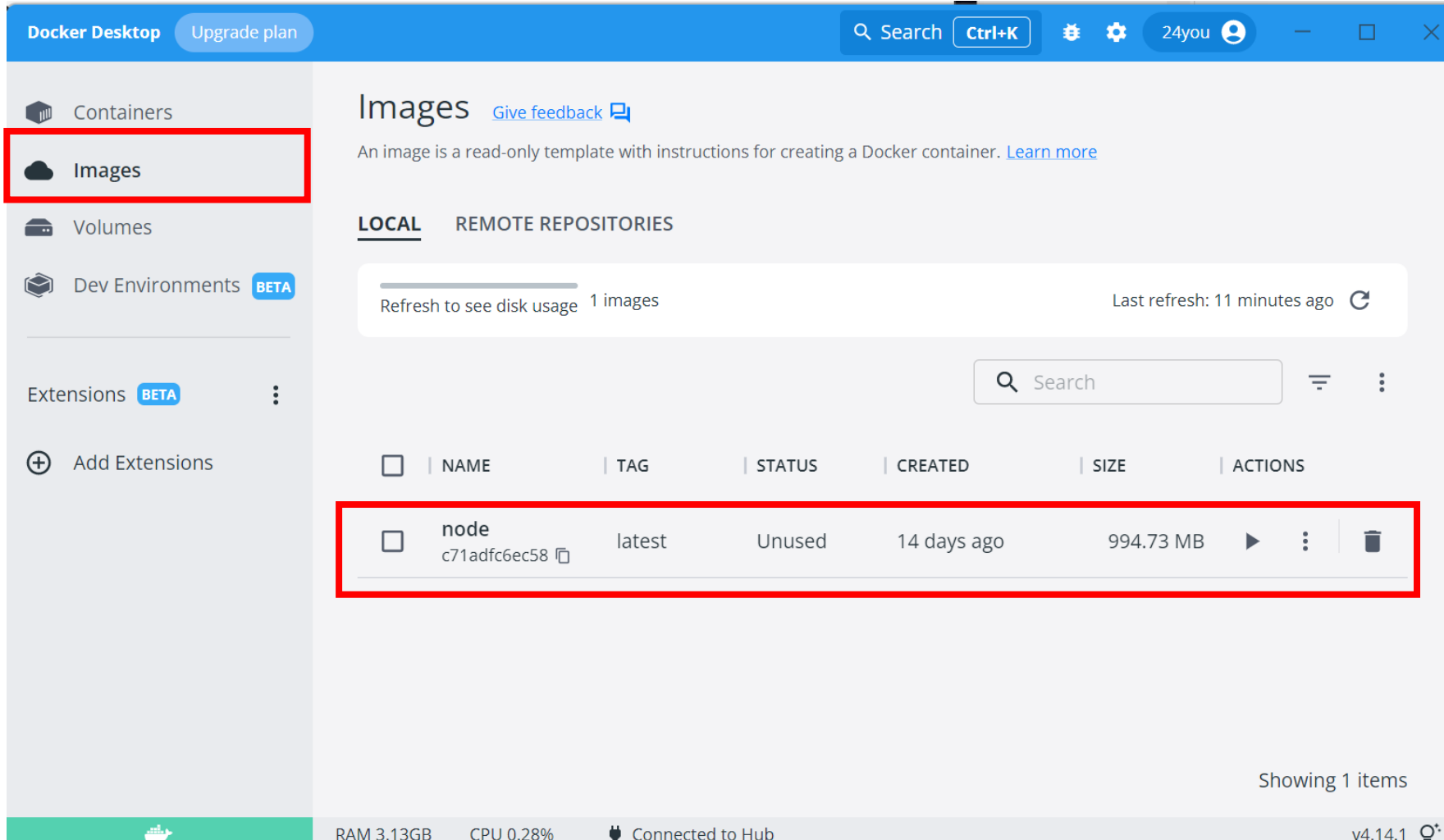
Docker Image Pull

```
docker pull node  
dockr images
```

```
C:\Users\SBA5>docker pull node  
Using default tag: latest  
latest: Pulling from library/node  
a8ca11554fce: Pull complete  
e4e46864aba2: Pull complete  
c85a0be79bfb: Pull complete  
195ea6a58ca8: Pull complete  
157f16ed0a0c: Pull complete  
3745f00fc1cc: Pull complete  
5bbc49fd3a32: Pull complete  
bf8c7ae7c46e: Pull complete  
dc0e02f1aec6: Pull complete  
Digest: sha256:bff0e689cb433913ab411af7a58253d54c7fd8c3134ffeb25287cdf24d9a5972  
Status: Downloaded newer image for node:latest  
docker.io/library/node:latest
```

```
C:\Users\SBA5>docker images  
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE  
node          latest    c71adfc6ec58   13 days ago   995MB
```

Docker Image Pull



Docker Desktop Upgrade plan

Search **Ctrl+K** 24you

Containers

Images

Volumes

Dev Environments **BETA**

Extensions **BETA**

Add Extensions

Images

[Give feedback](#)

An image is a read-only template with instructions for creating a Docker container. [Learn more](#)

LOCAL REMOTE REPOSITORIES

Refresh to see disk usage 1 images Last refresh: 11 minutes ago

Search

	NAME	TAG	STATUS	CREATED	SIZE	ACTIONS
<input type="checkbox"/>	node c71adfc6ec58	latest	Unused	14 days ago	994.73 MB	

Showing 1 items

RAM 3.13GB CPU 0.28% Connected to Hub v4.14.1

Docker Run(1)

Images [Give feedback](#)





An image is a read-only template with instructions for creating a Docker container. [Learn more](#)

LOCAL REMOTE REPOSITORIES


Refresh to see disk usage 1 images


Last refresh: 19 minutes ago 

 Search

<input type="checkbox"/>	NAME	TAG	STATUS	CREATED	SIZE	ACTIONS
<input type="checkbox"/>	node c71adfc6ec58 	latest	Unused	14 days ago	994.73 MB	  

Docker Run(1)

 **Run a new container**
node:latest

Optional settings 

A random name is generated if you do not provide one.

Ports
No ports exposed in this image

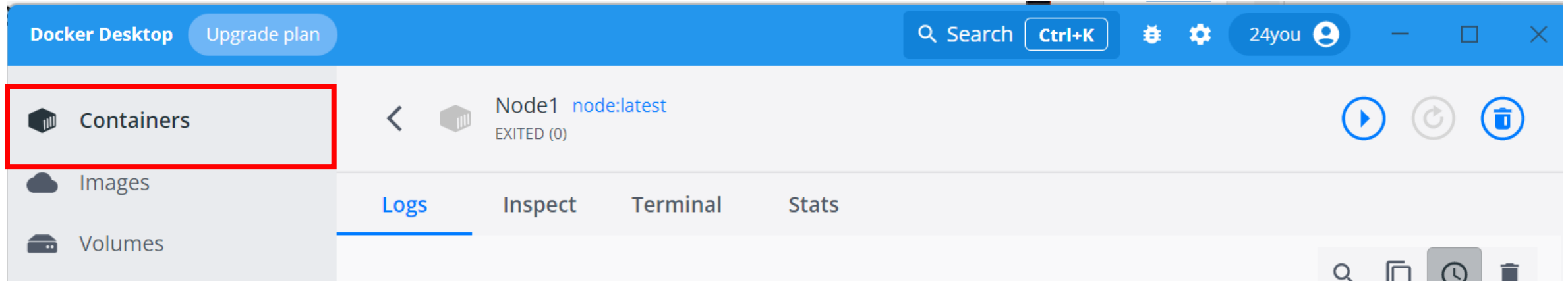
Volumes

...

+

Environment variables

Docker Run(1)



Docker Run(2)

```
docker run 이미지이름  
docker run --name 컨테이너이름 이미지 이름  
docker rmi 이미지이름
```

```
docker stop 이름 (컨테이너 ID)
```

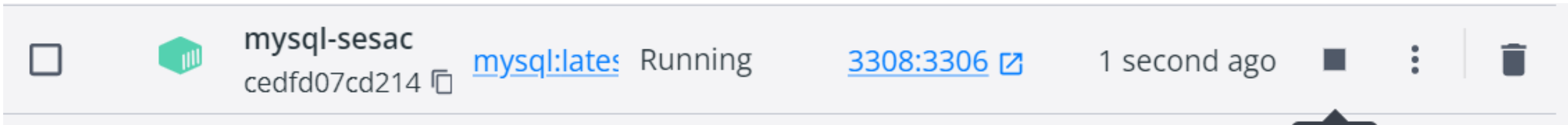
```
docker start 이름 (컨테이너 ID)
```

```
docker rm 이름 (컨테이너 ID)
```

```
docker ps -a
```

실습. Docker 실습

- Mysql 이미지를 받아 mysql-sesac 이름의 컨테이너로 실행해보기



힌트! Mysql은 실행 시 환경변수 설정이 필수!

Run 실행 시 옵션에 추가하기

`-d -p 3308:3306 -e MYSQL_ROOT_PASSWORD=1234`

A terminal window showing the command to run a MySQL container. The command is: `-d -p 3308:3306 -e MYSQL_ROOT_PASSWORD=1234 mysql`. The first part of the command is partially obscured by a blue rectangular box.

Docker 실행

Dockerfile 이란?

- 필요한 최소한의 패키지를 설치하고 동작하기 위한 자신만의 설정을 담고 있는 파일
- 즉, 기본 이미지가 아닌 기본 이미지를 바탕으로 설정이 담긴 이미지를 만들기 위해 사용하는 파일

Dockerfile 만들기

```
FROM mysql
```

```
ENV MYSQL_ROOT_PASSWORD=password
```

```
EXPOSE 3309
```

Dockerfile로 이미지 만들기

```
docker build -t 이름 도커파일경로
```

```
$ docker build -t mysql_self .
```

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mysql_self	latest	5425ac8d4b0a	12 days ago	538MB

Dockerfile 로 코드 실행하기

```
// index.js
const express = require('express');
const app = express();

app.get('/', (req, res) => {
  res.send(req.query.name);
});

app.listen(8000, ()=>{
  console.log("Server is open, port : 8000");
});
```

```
// Dockerfile
# 부모 이미지 지정
FROM node:16
# 작업 디렉토리 생성
WORKDIR /app
# 소스 추가
COPY . /app
# 패키지 설치
RUN npm install
# 포트 매핑
EXPOSE 8000
# 실행 명령
CMD [ "node", "index.js" ]
```

Dockerfile 로 코드 실행하기

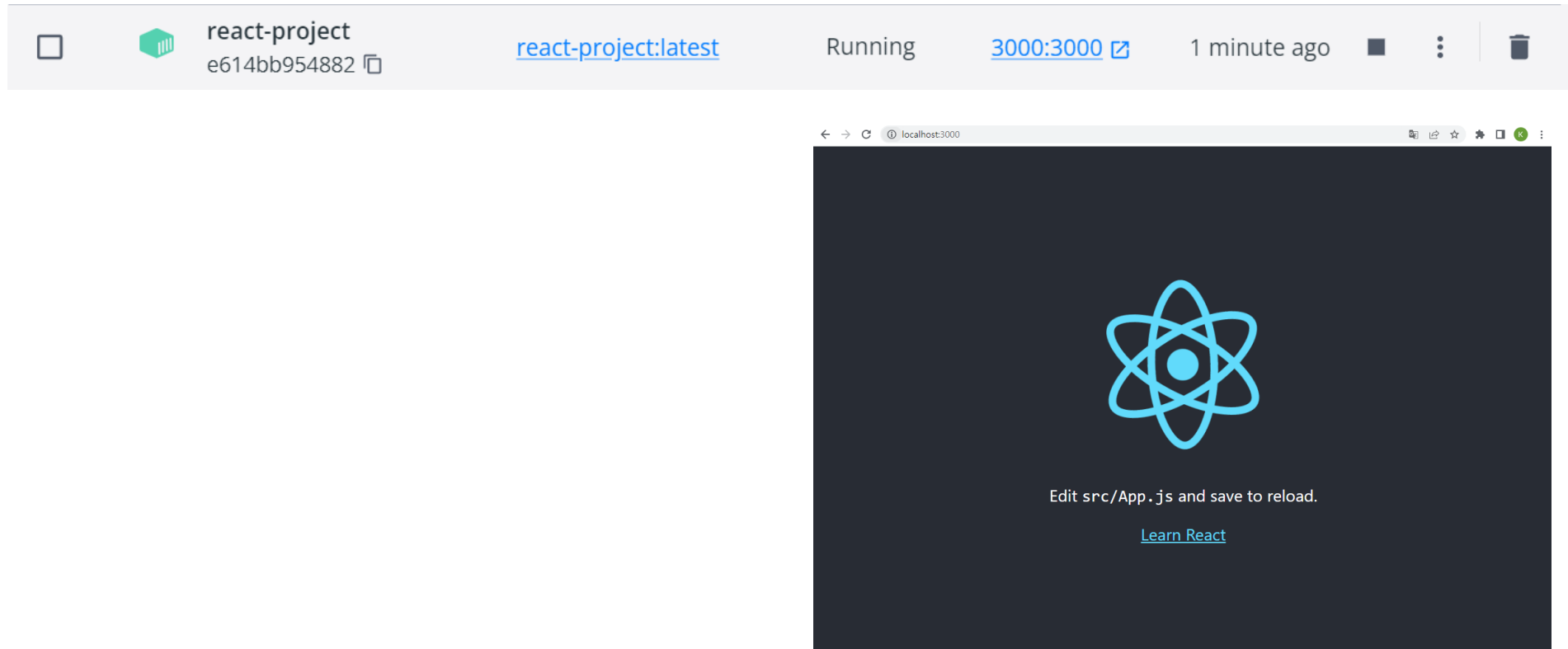
```
Simple Dockerfile (master)  
$ ls -a  
./ ../ Dockerfile index.js node_modules/ package-lock.json package.json
```

```
$ docker build -t node-project .
```

```
$ docker run --name node-project -p 8080:8000 node-project
```

실습. Dockerfile 실습

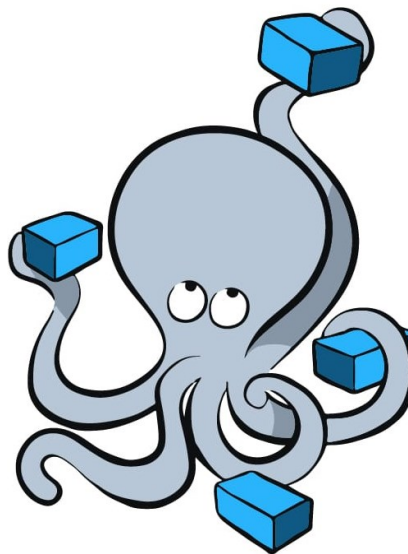
- Dockerfile을 만들고 React를 Docker로 돌리기



Docker Compose

Docker Compose란?

- 여러 개의 컨테이너를 하나로 묶어주는 역할을 하는 툴
- 여러 개의 컨테이너로부터 이루어진 서비스를 구축, 실행하는 순서를 자동으로 하여 관리를 가능하게 해준다.



docker
Compose