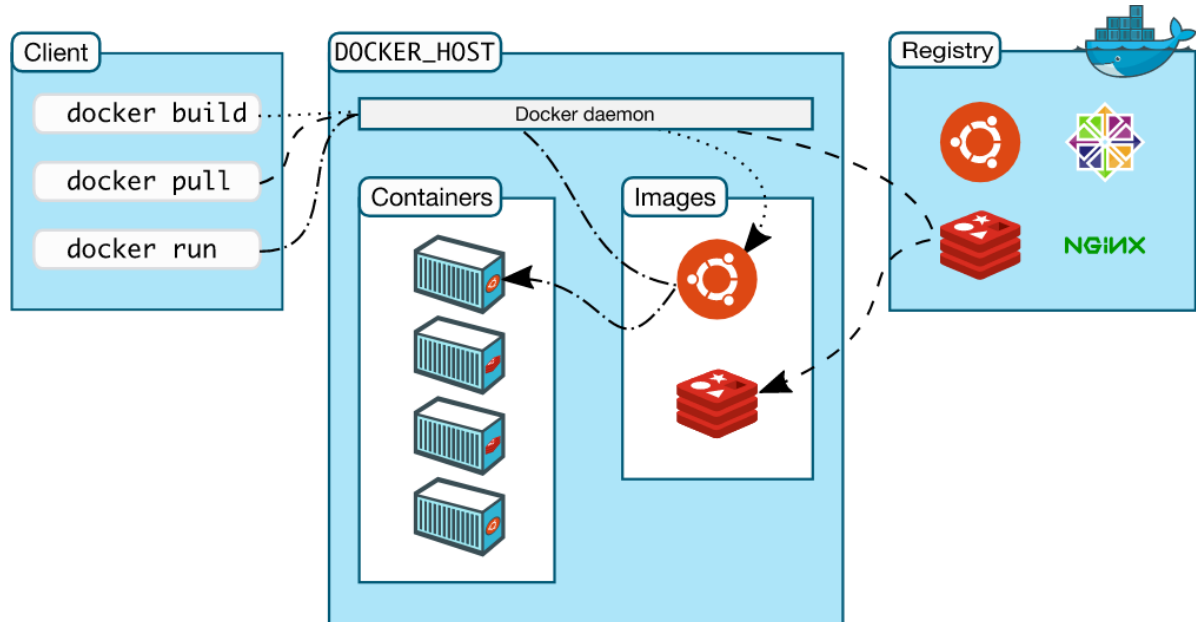


## Docker란?

도커는 컨테이너 기반의 오픈소스 가상화 플랫폼 중 하나이다. Container와 VM의 차이는 [다음](#) 링크를 참조.

아키텍처는 아래와 같다.



Docker는 Client-Server 아키텍처이다. Docker Client와 Docker Demon이 RespAPI를 사용하여 통신한다.

### Docker 데몬

Docker Api 요청수신, 이미지, 컨테이너, 네트워크와 같은 도커 객체 및 도커 서비스 관리

### Docker 클라이언트

Docker 사용자가 Docker와 상호작용하기 위한 방법. 기본적인 도커명령어를 통해서 Docker 데몬과 통신

### Docker 레지스트리

Docker 이미지를 저장, Docker hub이라는 공용 레지스트리와 개인private한 레지스트리가 있다. 일반적으로 공용 레지스트리에서 실행

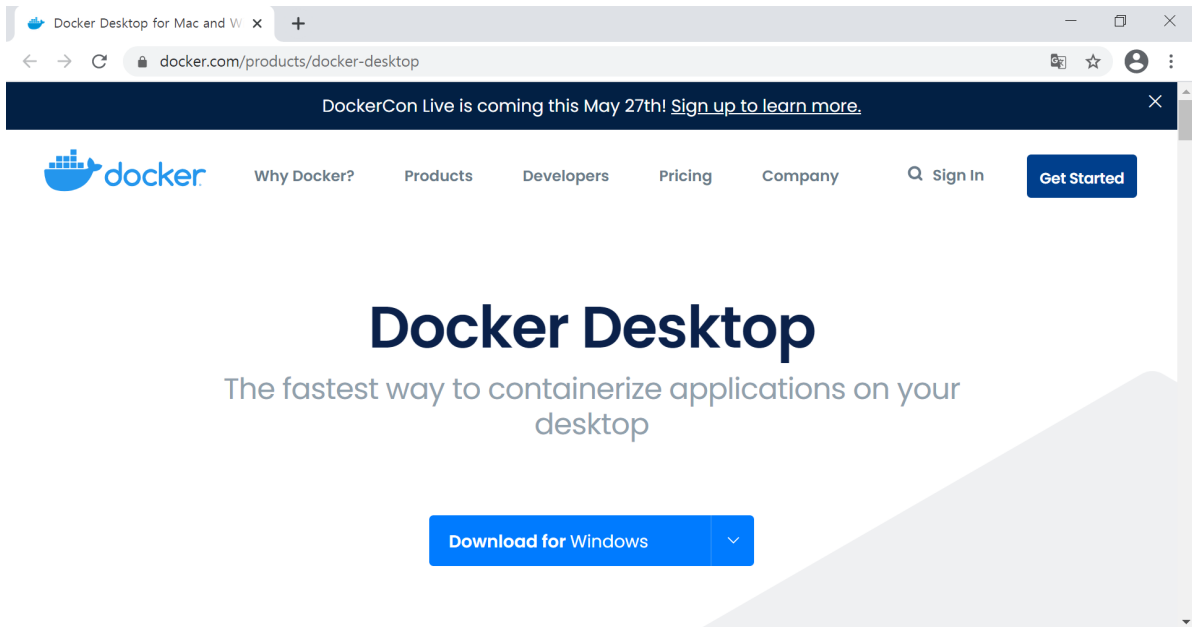
### Docker 객체

도커 이미지: 도커 이미지는 컨테이너 실행에 필요한 파일과 설정값등을 포함하고 있다.

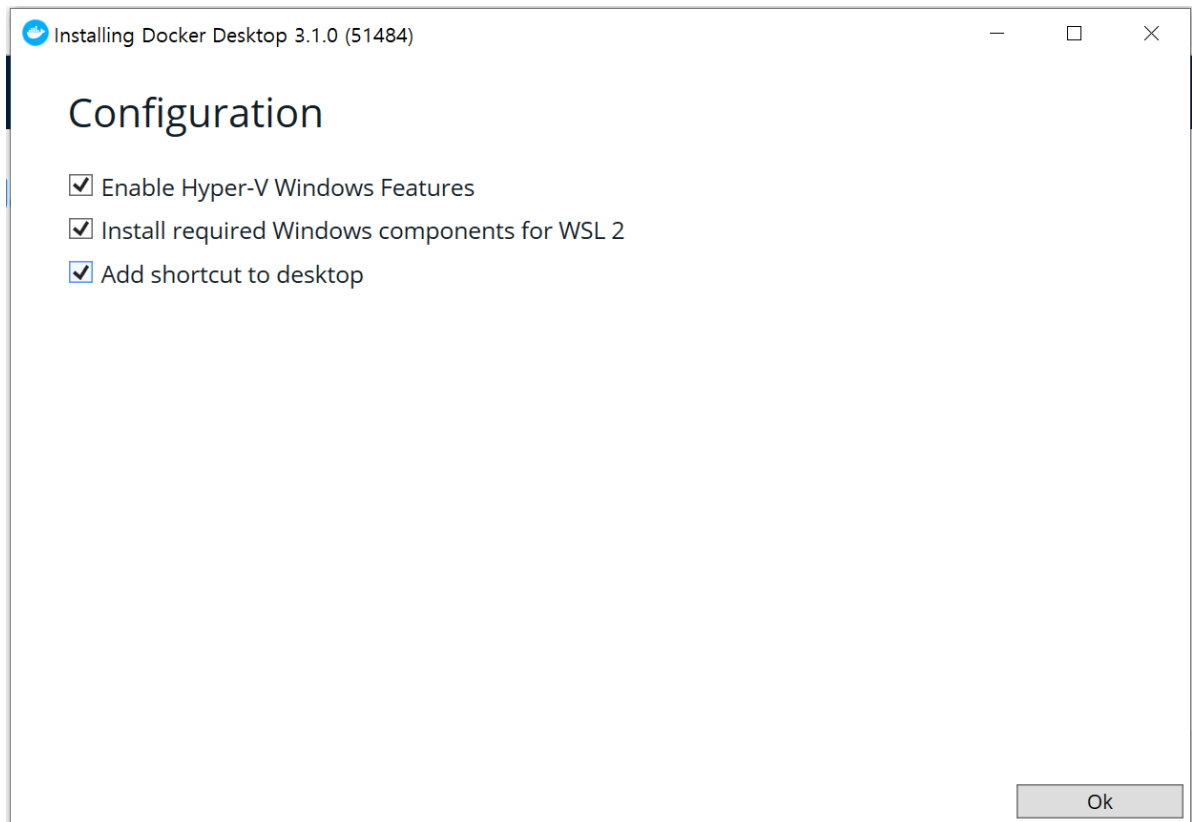
컨테이너 : 컨테이너는 도커 이미지의 실행 가능한 인스턴스

## Docker 설치

<https://www.docker.com/products/docker-desktop>에서 자신에게 맞는 OS의 Docker를 다운로드 받는다.

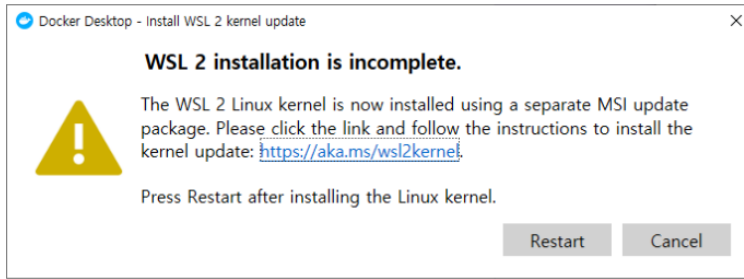


다운로드 받은 설치파일을 실행시키고 OK를 누르면 설치가 시작된다.



## Docker 실행 시 WSL installation 오류

Docker 실행시 다음과 같은 오류 메시지가 나오면 해당 링크로 가서 관련 파일을 설치해 준 후 재기동 하면 된다.



## 4단계 - Linux 커널 업데이트 패키지 다운로드

1. 최신 패키지를 다운로드합니다.

- x64 머신용 최신 WSL2 Linux 커널 업데이트 패키지

### ① 참고

ARM64 머신을 사용하는 경우 ARM64 패키지를 대신 다운로드하세요. 사용하고 있는 머신의 종류를 잘 모르는 경우 명령 프롬프트 또는 PowerShell을 열고 `systeminfo | find "System Type"`을 입력합니다.

## Docker Image 다운로드

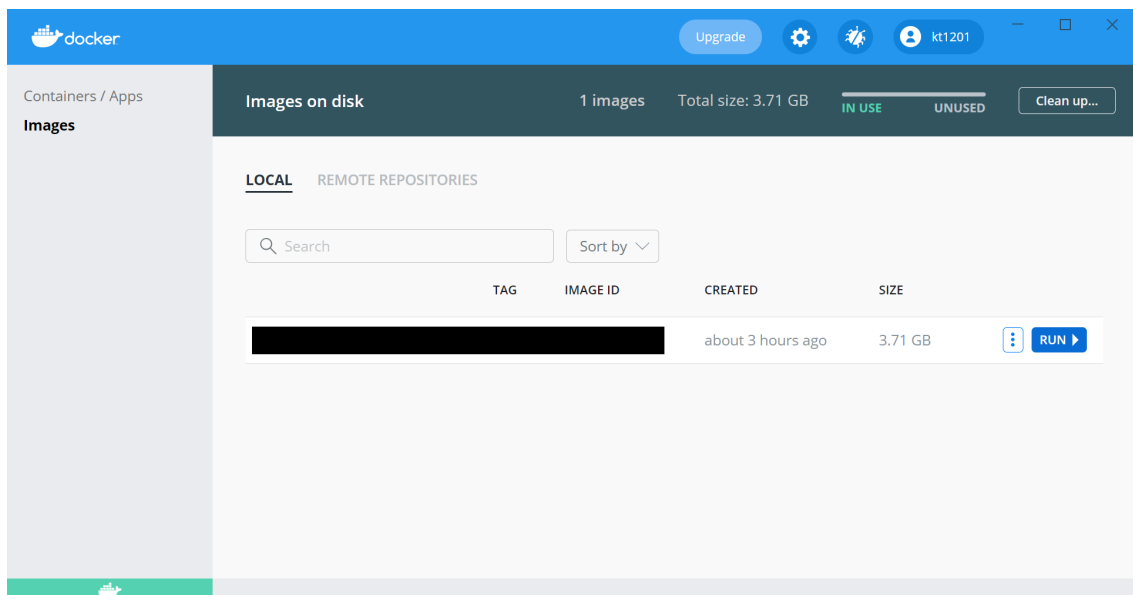
Docker 설치가 완료되면 cmd나 terminal을 열고 아래 명령어로 Docker Image를 다운로드 받는다.

```
$ docker pull 이미지명
```

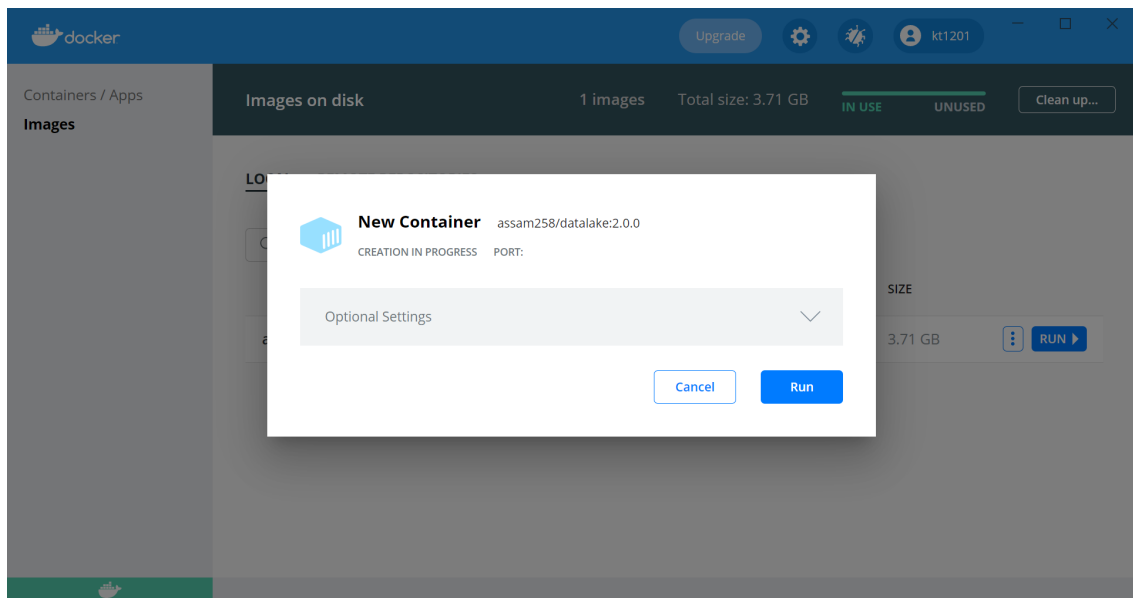
## Docker Container 생성 및 구동

GUI로 구성된 Docker Desktop에서 Container를 생성하고 구동하는 방법은 아래와 같다.

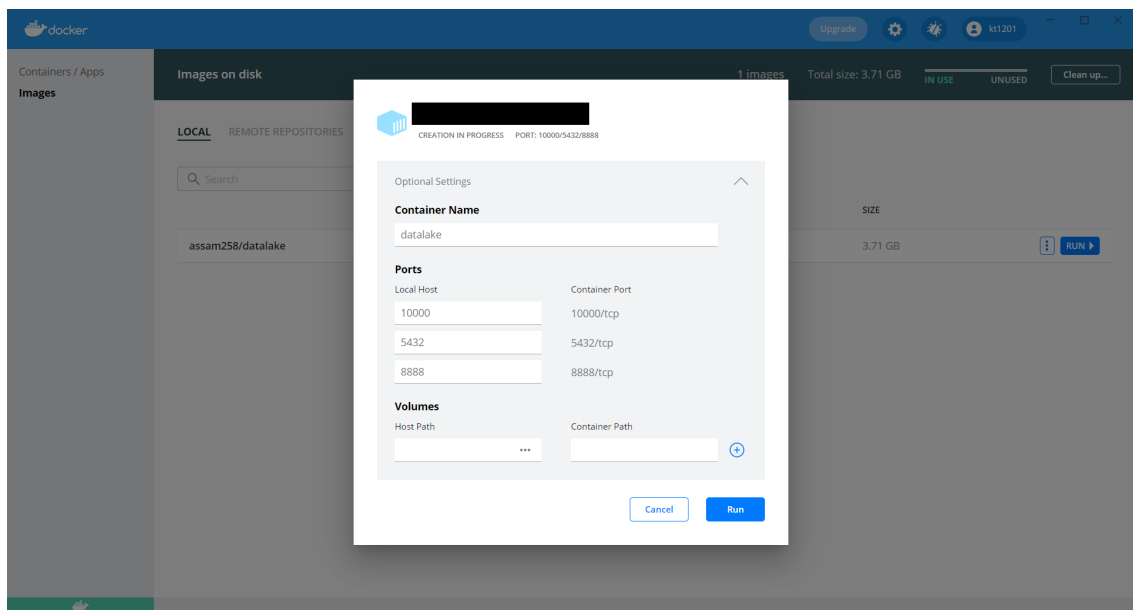
- 설치된 Docker Desktop을 실행시키고 왼쪽메뉴에 Images를 눌러보면 아래와 같이 다운받았던 Docker image가 생성되어 있다.



- 생성된 Docker 이미지의 RUN 버튼을 누르면 아래와 같은 팝업이 뜬다.



- Optional Settings를 눌러준다.



- 위 그림과 같이 연결 포트를 설정해 줄 수 있다. 나의 경우엔 아래와 같이 hive, postgres, tomcat의 포트를 설정해 주었다. 로컬에 쓰지 않는 포트로 설정해 주면 된다.

(예를 들어, 로컬에 이미 기존 postgres가 5432포트를 쓰고있는 경우 5431과 같이 바꿔주면 된다.)

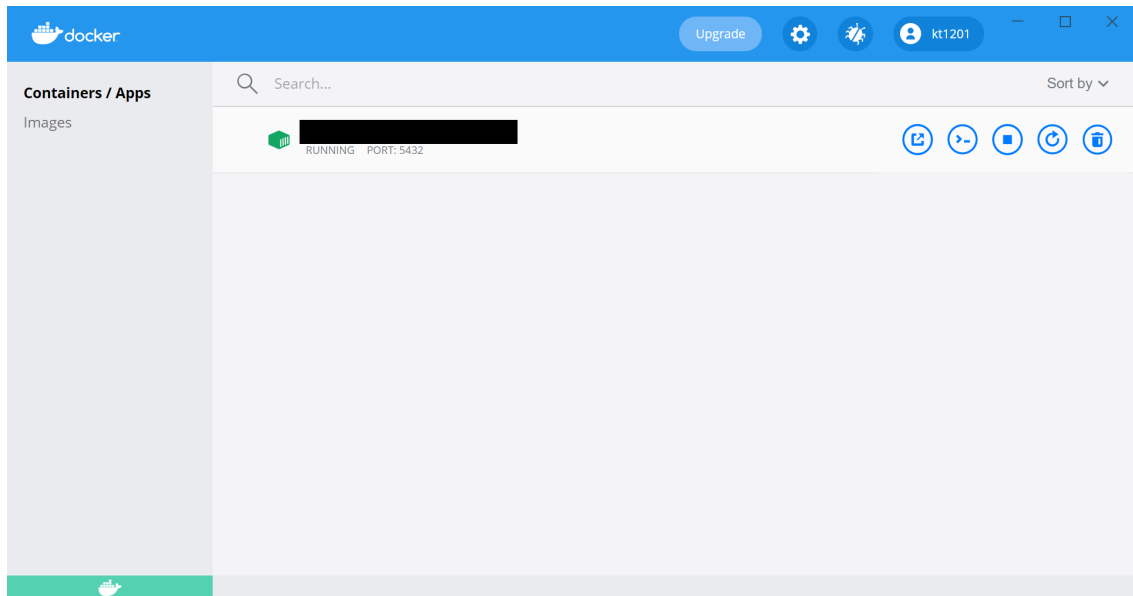
Optional Settings	
Container Name	datalake(자유대로 명명)
Port	10000 10000/tcp (hive) 5432 5432/tcp (postgres) 8888 8888/tcp (tomcat)

그 아래 Volumes을 통하여 local과 container간에 이어지는 공유폴더를 생성할 수 있다.

- Host Path에 local 경로인 C:\test를 적고, Container Path에 /home/hadoop/test를 적고, local의 test폴더에 파일을 올려놓으면 Container안의 test 폴더에서도 볼 수 있다.

설정을 한 후 Run을 누르면 Docker Container가 생성 및 구동된다.

왼쪽 메뉴의 Containers/Apps을 누르면 아래 그림과 같이 Container가 생성되어 있는 것을 볼 수 있다.



명령어로 Container를 생성하고 구동하는 방법은 아래와 같다.

- `docker create`으로 Container를 생성하고, `docker start`로 구동한다. 명령어는 아래와 같다.

```
$ docker create --name datalake -p 10000:10000 -p 5432:5432 -p 8888:8888 이미지명
$ docker start datalake
```

- 위 두 개의 명령어를 `docker run`으로 한번에 생성, 구동시킬 수도 있다.

```
$ docker run --name datalake -p 10000:10000 -p 5432:5432 -p 8888:8888 이미지명
```

## Docker Terminal 실행

GUI로 구성된 Docker Desktop에서 Terminal을 실행하는 방법은 아래와 같다.

- RUNNING 상태의 Container의 CLI를 Click

명령어로 터미널을 실행하는 방법은 아래와 같다.

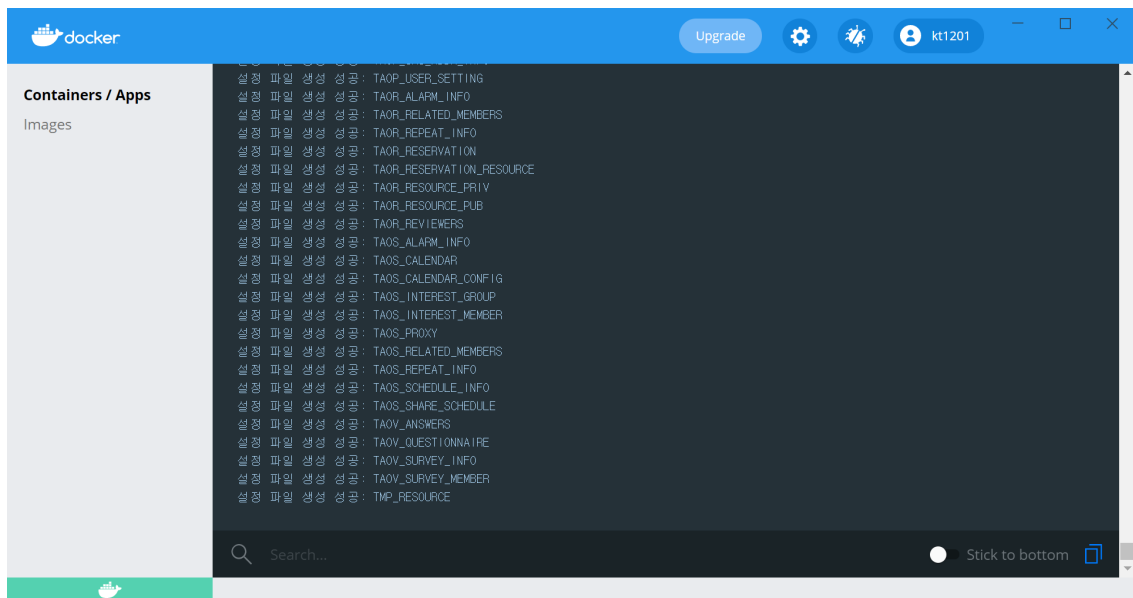
```
$ docker exec -it datalake bash
```

## Docker Log

GUI로 구성된 Docker Desktop에서 Log를 보는 방법은 아래와 같다.

- 생성된 Container를 Click 한다.

아래와 같이 Log를 볼 수 있다.



명령어로 Log를 보는 방법은 아래와 같다.

```
$ docker logs -f datalake
```

## Docker Container 종료

GUI로 구성된 Docker Desktop에서 Container를 종료하는 방법은 아래와 같다.

- RUNNING상태의 Container의 Stop을 Click

명령어로 Container를 종료하는 방법은 아래와 같다.

```
$ docker exec -it datalake shutdown.sh
```

## Docker Container 구동

GUI로 구성된 Docker Desktop에서 Container를 구동하는 방법은 아래와 같다.

- 기존에 생성되어 있던 Container의 Start를 Click

명령어로 Container를 구동하는 방법은 아래와 같다.

```
$ docker start datalake
```

## Docker Container 제거

GUI로 구성된 Docker Desktop에서 Container를 제거하는 방법은 아래와 같다.

- 기존에 생성되어 있던 Container의 Delete를 Click

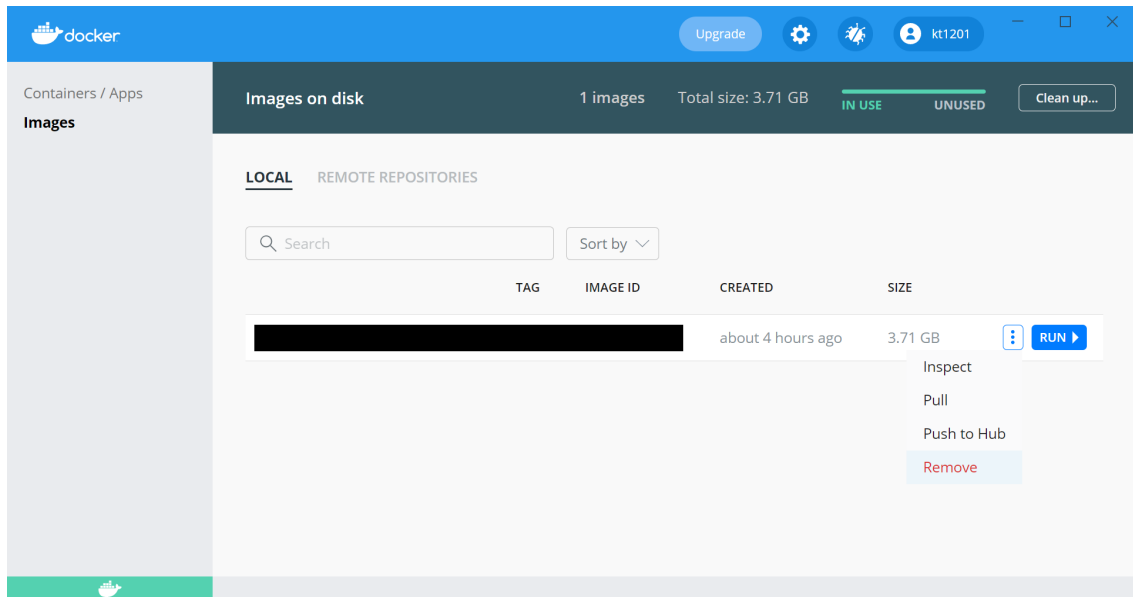
명령어로 Container를 제거하는 방법은 아래와 같다.

```
$ docker rm datalake
```

# Docker Image 제거

GUI로 구성된 Docker Desktop에서 Image를 제거하는 방법은 아래와 같다.

- 기존에 있던 Image의 Remove를 Click



명령어로 Container를 제거하는 방법은 아래와 같다.

```
$ docker rmi 이미지명
```