

Abstract geometric lines in the top left corner.

Introduction to docker

Docker

■ 컨테이너

- » 호스트 OS상에 논리적인 구획(컨테이너)를 만들고 애플리케이션 작동에 필요한 라이브러리나 애플리케이션 등을 하나로 모아 별도의 서버인 것처럼 사용할 수 있는 도구
- » 호스트 OS의 리소스를 논리적으로 분리시키고 여러 컨테이너가 공유해서 사용
- » OS, IP 주소 등과 같은 시스템 자원을 마치 각 애플리케이션이 점유하는 것처럼 운영
- » 마이크로 서비스형 애플리케이션과 친화성 높음

■ 도커

- » 컨테이너 관리 서비스
- » 컨테이너 기술을 사용하여 애플리케이션의 실행 환경을 구축 및 운용하기 위한 플랫폼
- » 애플리케이션 실행에 필요한 환경을 하나의 이미지로 모아두고 그 이미지를 사용하여 다양한 환경에서 애플리케이션 실행 환경을 구축 및 운용할 수 있는 오픈소스 플랫폼

Docker 설치

- download url → <https://www.docker.com/get-started/>

Get Started with Docker

Build applications faster and more securely with Docker for developers

Learn how to install Docker

Download for Windows



Download for Mac - Intel Chip



Download for Mac - Apple Chip



Download for Linux



An experience you'll love

Customize your development experience with tools that enhance your tech stack and optimize your development process.

Docker Hub

- GitHub, Bitbucket과 같은 소스코드 관리 도구와 연계하여 코드를 빌드하는 기능이나 실행 가능한 애플리케이션의 이미지를 관리하는 기능을 갖춘 Docker의 공식 리포지토리 서비스

The screenshot shows the Docker Hub interface with a search bar containing 'ubuntu linux'. The results are sorted by 'Best Match'. The first result is 'ubuntu', which is a Docker Official Image. It has over 1 billion downloads, 10K+ stars, and 26,561,427 pulls in the last week. The second result is 'oraclelinux', an Official Docker build of Oracle Linux, with over 10 million downloads, 1.0K stars, and 43,074 pulls in the last week. The third result is 'alpine', also with over 1 billion downloads and 10K+ stars, and 10,379,631 pulls in the last week. The left sidebar shows filters for Products (Images, Extensions, Plugins), Trusted Content (Docker Official Image, Verified Publisher, Sponsored OSS), and Operating Systems (Linux, Windows).

Filters

Products

- ☐ Images
- ☐ Extensions
- ☐ Plugins



Trusted Content



- ☐ Docker Official Image ⓘ
- ☐ Verified Publisher ⓘ
- ☐ Sponsored OSS ⓘ



Operating Systems

- ☐ Linux
- ☐ Windows

1 - 25 of 10,000 results for **ubuntu linux**. Best Match

**ubuntu** 
Updated 14 days ago
Ubuntu is a Debian-based Linux operating system based on free software.
Linux ARM 64 PowerPC 64 LE IBM Z 386 riscv64 x86-64 ARM
Pulls: 26,561,427
Last week
[Learn more](#)

**oraclelinux** 
Updated 16 days ago
Official Docker builds of Oracle Linux.
Linux x86-64 ARM 64
Pulls: 43,074
Last week
[Learn more](#)

**alpine** 
Pulls: 10,379,631
Last week

도커 명령

■ 이미지 관리 명령

명령 내용 / 명령 형식
■ 이미지 다운로드
■ <code>docker image pull [option] image-name[:tag-name]</code>
■ 이미지 목록 표시
■ <code>docker image ls [option] [repository-name]</code>
■ 이미지 상세 정보 표시
■ <code>docker image inspect [option] image-name[:tag-name]</code>
■ 이미지 태그 이름 지정
■ <code>docker image tag image-name new-image-name</code>
■ 이미지 검색
■ <code>docker search [option] search-keyword</code>
■ 이미지 제거
■ <code>docker image rm [option] image-name [image-name ...]</code>

도커 명령

■ 컨테이너 관리 명령

명령 내용 / 명령 형식
<ul style="list-style-type: none">지정된 이미지를 기반으로 컨테이너 생성 및 실행 (이미지가 없으면 <code>docker image pull</code> 실행)<code>docker container run [option] image-name[:tag-name] [parameters]</code>
<ul style="list-style-type: none">(가동) 컨테이너 목록 표시<code>docker container ls [option]</code>
<ul style="list-style-type: none">컨테이너 가동 확인<code>docker container stats [container-identifier]</code>
<ul style="list-style-type: none">컨테이너 시작<code>docker container start [option] container-identifier</code>
<ul style="list-style-type: none">컨테이너 정지<code>docker container stop [option] container-identifier</code>
<ul style="list-style-type: none">컨테이너 재시작<code>docker container restart [option] container-identifier</code>

도커 명령

■ 컨테이너 관리 명령

명령 내용 / 명령 형식
■ 컨테이너 중단/재개
■ <code>docker container pause/unpause container-identifier</code>
■ 가동 중인 컨테이너에 연결
■ <code>docker container attach container-identifier</code>
■ 가동 중인 컨테이너에서 새 프로세스 실행
■ <code>docker container exec [option] container-identifier command [parameters]</code>
■ 가동 중인 컨테이너의 프로세스 확인
■ <code>docker container top container-identifier</code>
■ 가동 중인 컨테이너의 포트 확인
■ <code>docker container port container-identifier</code>
■ 컨테이너 이름 변경
■ <code>docker container rename old-name new-name</code>

도커 명령

- 컨테이너 관리 명령

명령 내용 / 명령 형식

- | |
|--|
| <ul style="list-style-type: none">■ 컨테이너와 호스트간 파일 복사■ <code>docker container cp container-identifier:file-path host-file-path</code>■ <code>docker container cp host-file-path container-identifier:file-path</code> |
|--|

도커 명령

■ 이미지 생성 명령

명령 내용 / 명령 형식
<ul style="list-style-type: none">컨테이너로부터 이미지 생성
<ul style="list-style-type: none"><code>docker container commit [option] container-identifier [image-name[:tag-name]]</code>
<ul style="list-style-type: none">컨테이너를 tar 파일로 출력
<ul style="list-style-type: none"><code>docker container export container-identifier > file-name.tar</code>
<ul style="list-style-type: none">tar 파일로부터 이미지 작성 (export 명령으로 만들어진 파일에 적용)
<ul style="list-style-type: none"><code>docker image import file-or-url - image-name[:tag-name]</code>
<ul style="list-style-type: none">이미지 저장
<ul style="list-style-type: none"><code>docker image save [option] file-name image-name</code>
<ul style="list-style-type: none">이미지 읽기 (save 명령으로 만들어진 파일에 적용)
<ul style="list-style-type: none"><code>docker image load [option]</code>
<ul style="list-style-type: none">사용하지 않는 이미지 일괄 삭제
<ul style="list-style-type: none"><code>docker image prune</code>

- 컨테이너로부터 이미지 생성

- `docker container commit [option] container-identifier [image-name[:tag-name]]`

- 컨테이너를 tar 파일로 출력

- `docker container export container-identifier > file-name.tar`

- tar 파일로부터 이미지 작성 (export 명령으로 만들어진 파일에 적용)

- `docker image import file-or-url - image-name[:tag-name]`

- 이미지 저장

- `docker image save [option] file-name image-name`

- 이미지 읽기 (save 명령으로 만들어진 파일에 적용)

- `docker image load [option]`

- 사용하지 않는 이미지 일괄 삭제

- `docker image prune`

Dockerfile 기반 구성 관리

- Dockerfile
 - » Docker 상에서 작동시킬 컨테이너의 구성 정보를 기술하기 위한 파일
 - » `docker build` 명령은 Dockerfile에 기술된 구성 정보를 바탕으로 Docker 이미지 작성
- Dockerfile 작성 기초
 - » text file
 - » 기본 파일 이름은 Dockerfile (다른 파일 이름을 사용할 수 있으나 빌드할 때 파일 이름을 명시적으로 지정해야 함)
 - » 기본 형식
 - › 명령 인수
 - › 명령은 대소문자를 구분하지 않지만 일반적으로 대문자로 작성
 - › 주석은 #으로 표시
 - › 베이스 이미지 설정은 필수 (`FROM` 명령)

Dockerfile 기반 구성 관리

- 명령 종류

명령	설명
FROM	베이스 이미지 지정
RUN	명령 실행
CMD	컨테이너 실행 명령
LABEL	라벨 설정
EXPOSE	포트 개방
ENV	환경변수
ADD	파일/디렉터리 추가
COPY	파일 복사
ENTRYPOINT	컨테이너 실행 명령

명령	설명
VOLUME	볼륨 마운트
USER	사용자 지정
WORKDIR	작업 디렉터리
ARG	Dockerfile 안의 변수
ONBUILD	빌드 완료 후 실행되는 명령
STOPSIGNAL	시스템 콜 시그널 설정
HEALTHCHECK	컨테이너의 헬스 체크
SHELL	기본 셸 설정