클라우드컴퓨팅 실습 3

Container 및 Docker 기초

조교 김민창

naxcco@gmail.com

공지사항

이용 가이드라인

- 매일 저녁 8:30 ~ 9시 사이에 **모든 인스턴스 삭제** **→** custom image으로 저장 하시길
  - **9시 이후에** 다시 작업하시기 바랍니다
- Instance 불필요한 자원/configuration 설정시 삭제 예정
- **과제 마감 되면 모든 CVM, custom image 삭제 예정**
- 사용하지 않을때 CVM instance을 shutdown

추가 사항

- 과제는 **미리 수행해 주시기 바랍니다!**
  - Cloud 플랫폼에 예기치 못한 일시적 장애가 발생할 수 있으므로,
    제출 마감일 이전에 충분한 시간을 확보하여 작업해 주시기 바랍니다

Section 1:
Containerization Explained

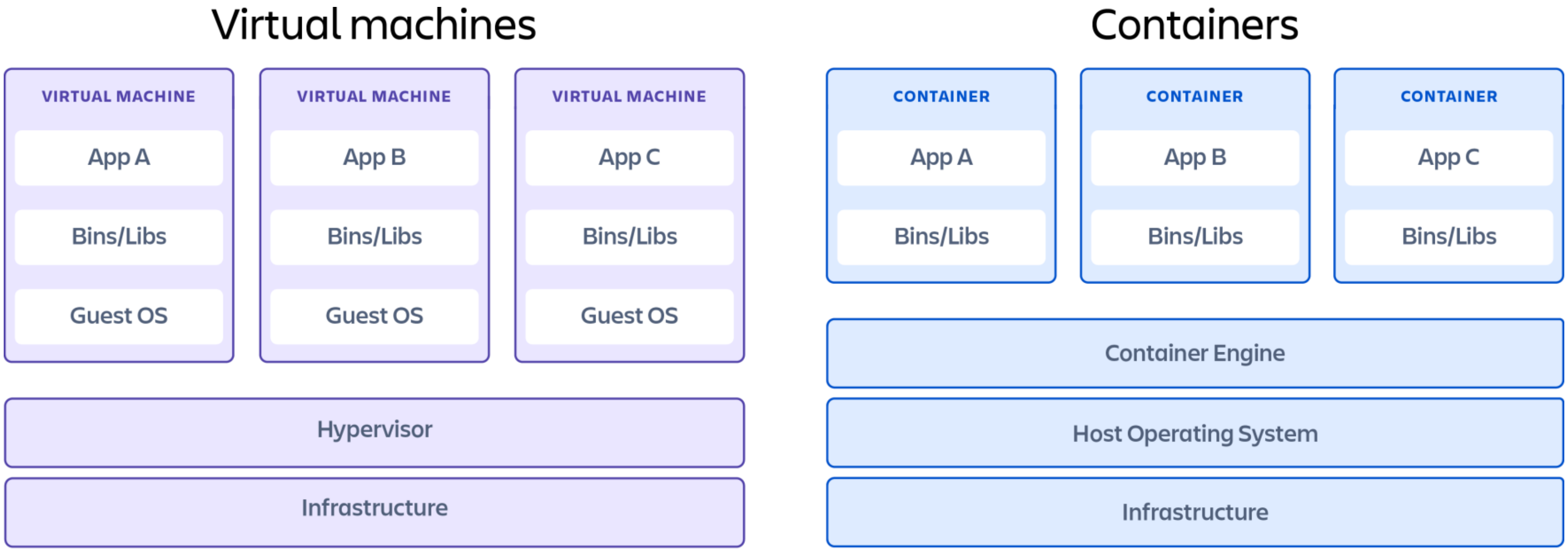
Containerization Explained

What is a container?

- A lightweight virtualized environment at the process level
  - Uses Linux kernel features: chroot, namespace, cgroup
  - Provides isolated execution environments while sharing the host OS kernel

- **Key characteristics:**
  - Minimal performance overhead
  - Only contains essential libraries and binaries
  - Fast startup and low memory usage

Containerization Explained



Virtual machines

VIRTUAL MACHINE

App A

Bins/Libs

Guest OS

VIRTUAL MACHINE

App B

Bins/Libs

Guest OS

VIRTUAL MACHINE

App C

Bins/Libs

Guest OS

Hypervisor

Infrastructure

Containers

CONTAINER

App A

Bins/Libs

CONTAINER

App B

Bins/Libs

CONTAINER

App C

Bins/Libs

Container Engine

Host Operating System

Infrastructure

Containerization Explained

Containerization Benefits

- (1) Simplified Development and Deployment
  - Containers run in isolated environments on the host OS
  - You can install software or modify configs **without affecting the host**
  - Once ready, package your container as an **image** and deploy it directly — no need to reinstall dependencies
  - Avoids “it works on my machine” issues

Containerization Explained

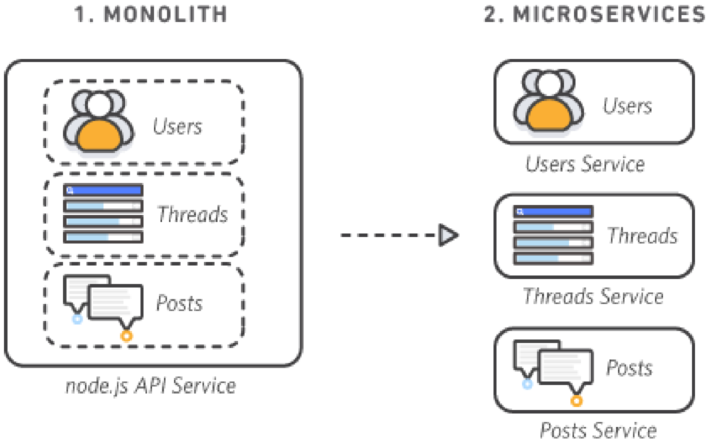
Containerization Benefits

- (2) Independence and Scalability
  - Containers start in seconds
  - Deployment to various nodes at the same time
  - Each component can run independently → perfect for **Microservice Architecture (MSA)**

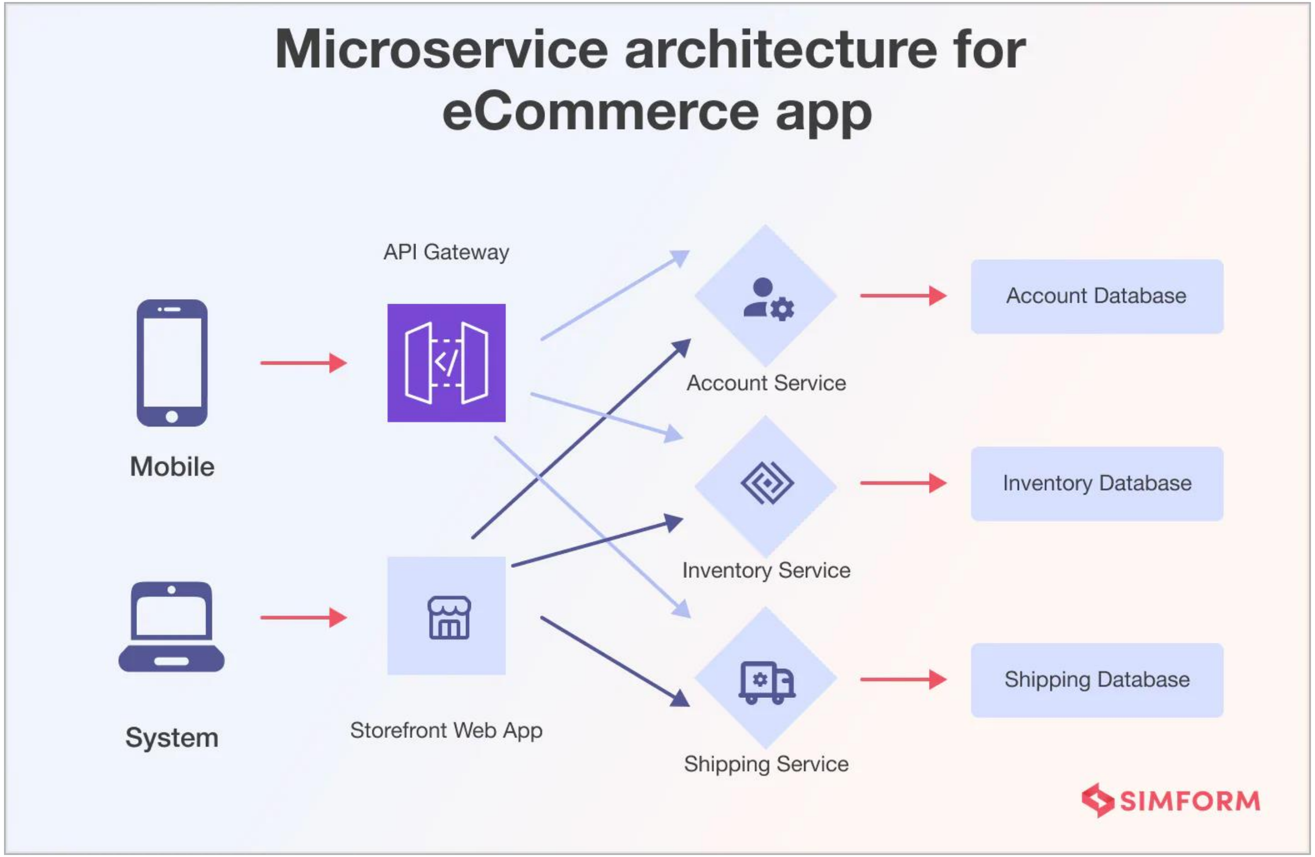
Containerization Explained

What is the Microservice Architecture (MSA)?

- Software is divided into **independent modules (services)** that interact through APIs
- Contrast with **Monolithic** architecture, where all logic runs in one process
- Advantages:
  - Easier maintenance and updates
  - Independent scaling of components
  - Language-agnostic flexibility



Containerization Explained



Section 2:
Introducing Docker

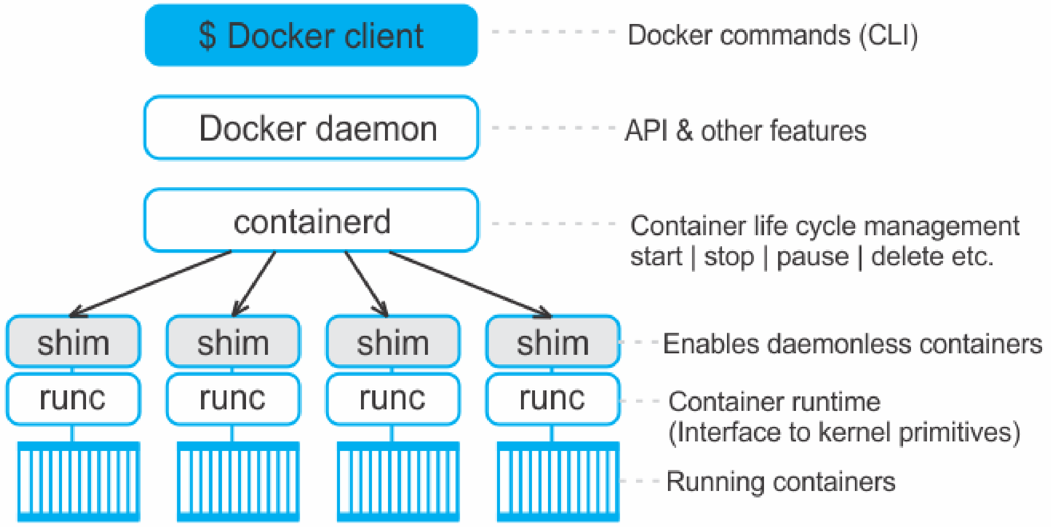
Introducing Docker

What Is Docker?

- **Open-source platform** to build, ship, and run applications in containers
- Simplifies container creation and management
- Provides:
  - **Docker Engine (dockerd)** – main service controlling containers
  - **Docker CLI** – user interface for commands
  - **Docker Hub** – registry for sharing images

Introducing Docker

Docker Engine Architecture



Docker commands (CLI)

API & other features

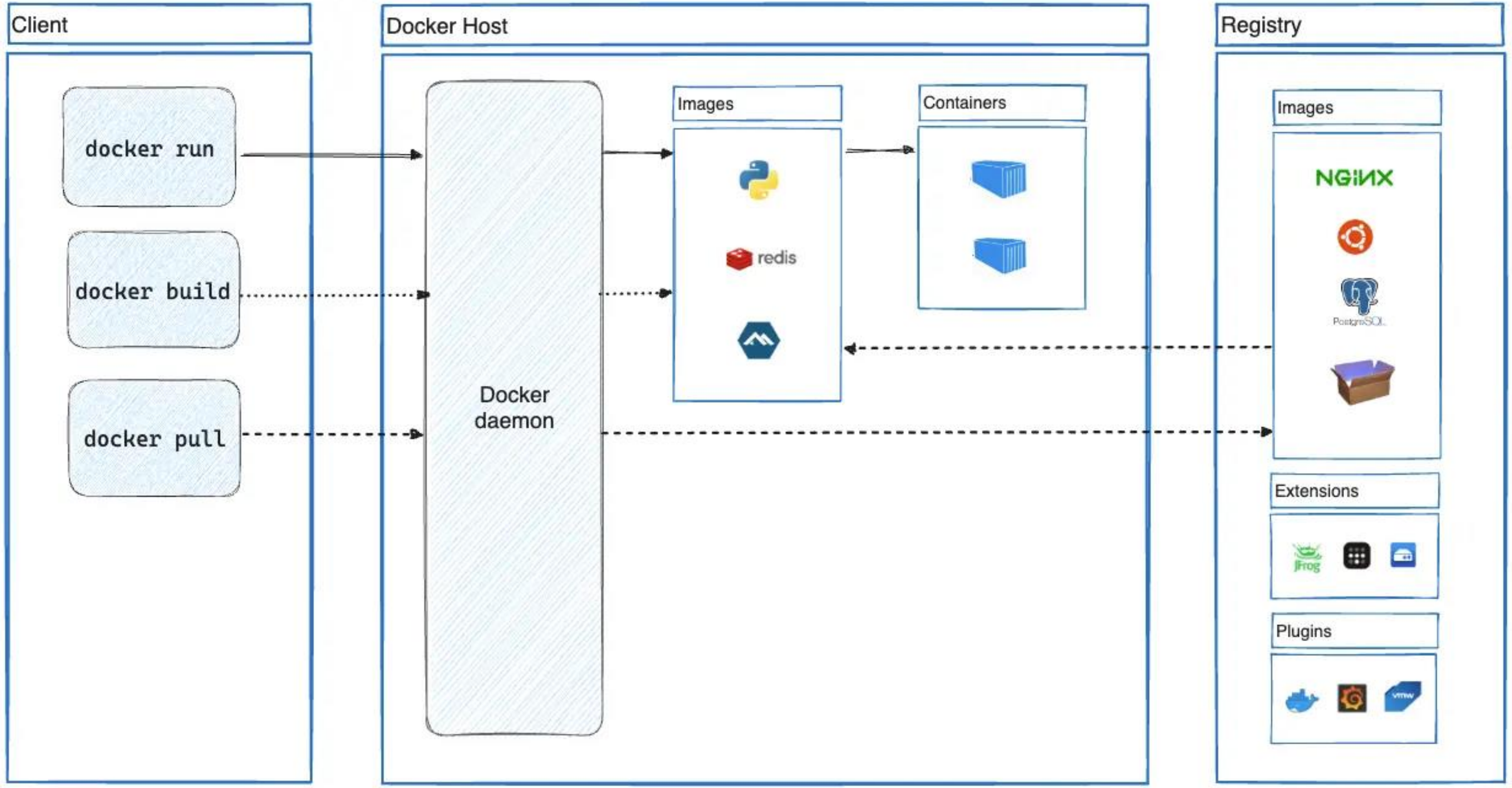
Container life cycle management
start | stop | pause | delete etc.

Enables daemonless containers

Container runtime
(Interface to kernel primitives)

Running containers

Introducing Docker

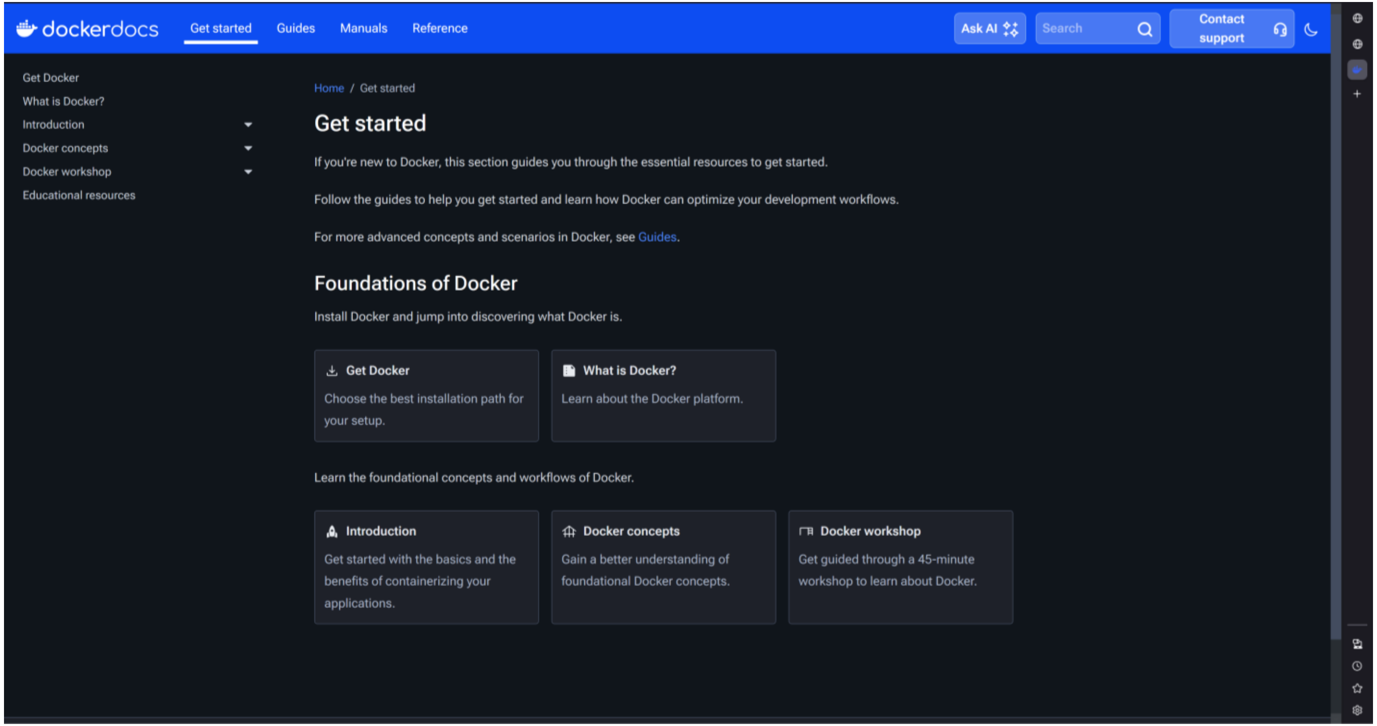


Section 3:
Practice (실습)

실습

Docker official documentation

- https://docs.docker.com/get-started/



Get started

If you're new to Docker, this section guides you through the essential resources to get started.

Follow the guides to help you get started and learn how Docker can optimize your development workflows.

For more advanced concepts and scenarios in Docker, see Guides.

Foundations of Docker

Install Docker and jump into discovering what Docker is.

Get Docker
Choose the best installation path for your setup.

What is Docker?
Learn about the Docker platform.

Learn the foundational concepts and workflows of Docker.

Introduction
Get started with the basics and the benefits of containerizing your applications.

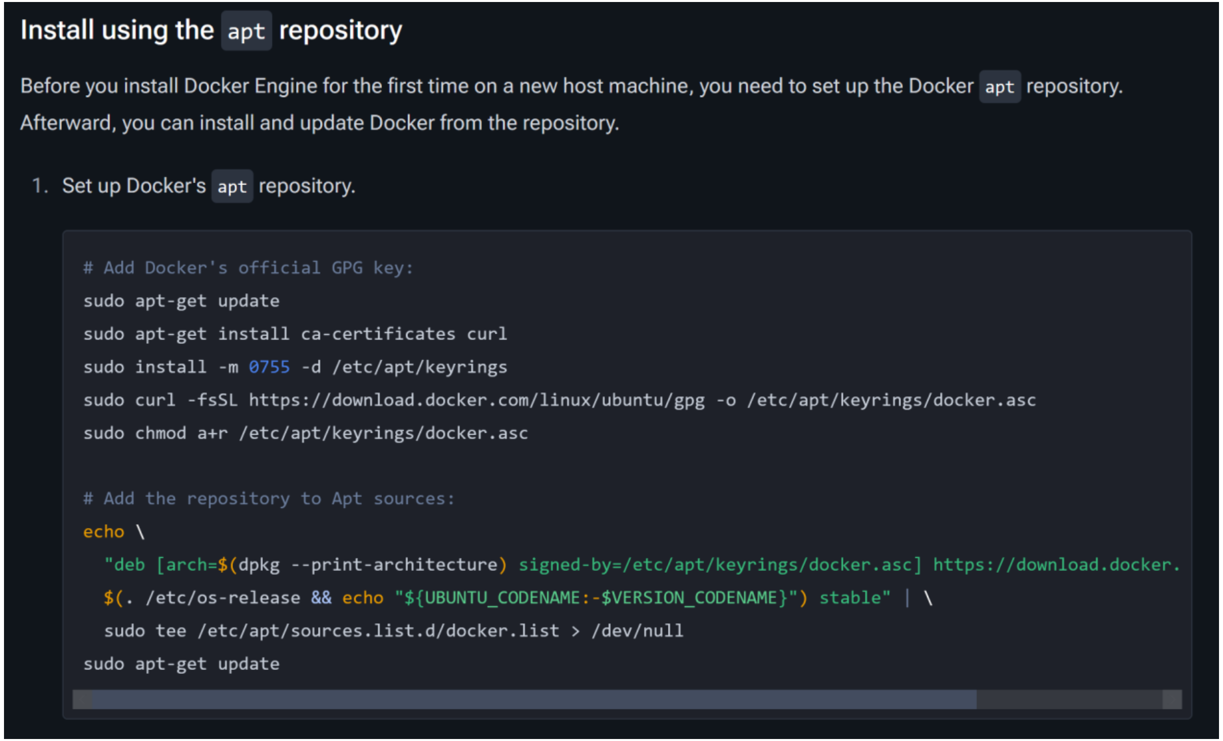
Docker concepts
Gain a better understanding of foundational Docker concepts.

Docker workshop
Get guided through a 45-minute workshop to learn about Docker.

실습

Docker Engine installation (docs link)

- Add Docker package to `apt` registry and sign with GPG key



Install using the `apt` repository

Before you install Docker Engine for the first time on a new host machine, you need to set up the Docker `apt` repository. Afterward, you can install and update Docker from the repository.

1. Set up Docker's `apt` repository.

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

실습

Docker Engine installation (docs link)

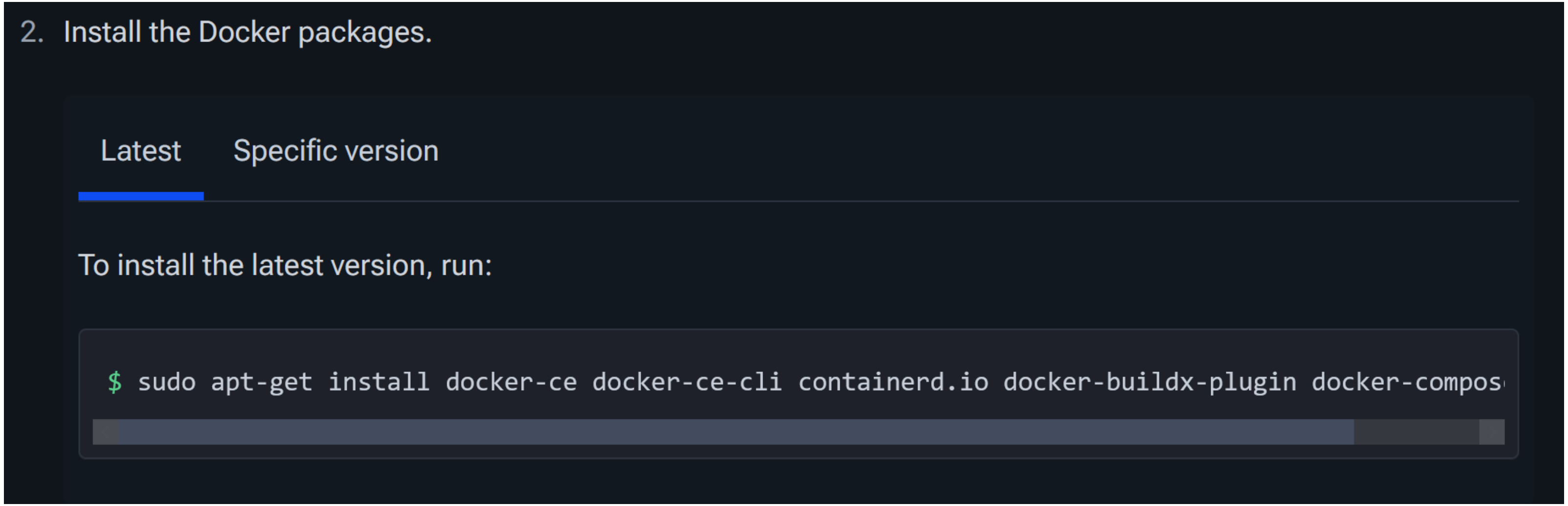
- Add Docker package to `apt` registry and sign with GPG key

```
# Add Docker's official GPG key
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources (for installing later on)
echo ₩
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu ₩
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | ₩
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

실습

Docker Engine installation (docs link)

- Install packages using the `apt-get` command



2. Install the Docker packages.

To install the latest version, run:

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compos
```

실습

Docker Engine installation (docs link)

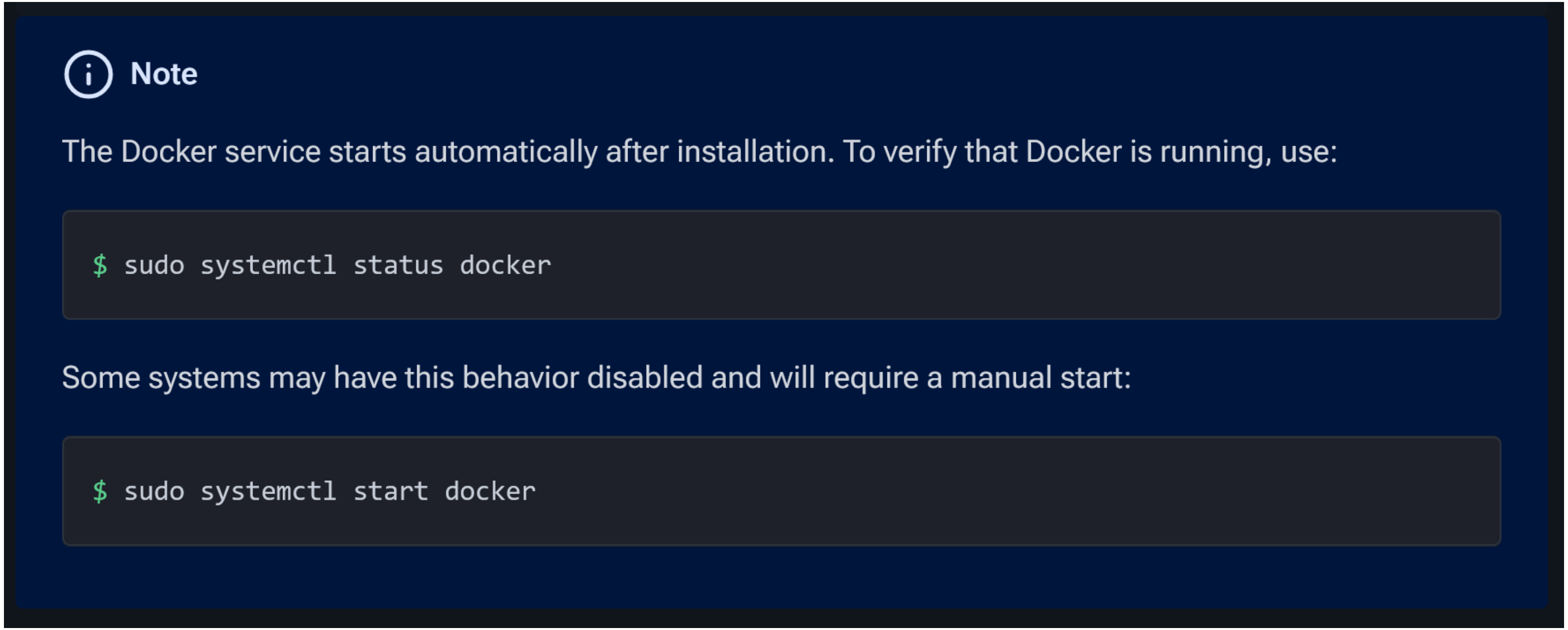
- Install packages using the `apt-get` command

```
# Install Docker Engine packages (Container Engine (CE), CLI, containerd runtime, buildx plugin)
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
```

실습

Docker Engine installation (docs link)

- Install packages using the `apt-get` command – verify installation



Note

The Docker service starts automatically after installation. To verify that Docker is running, use:

```
$ sudo systemctl status docker
```

Some systems may have this behavior disabled and will require a manual start:

```
$ sudo systemctl start docker
```

실습

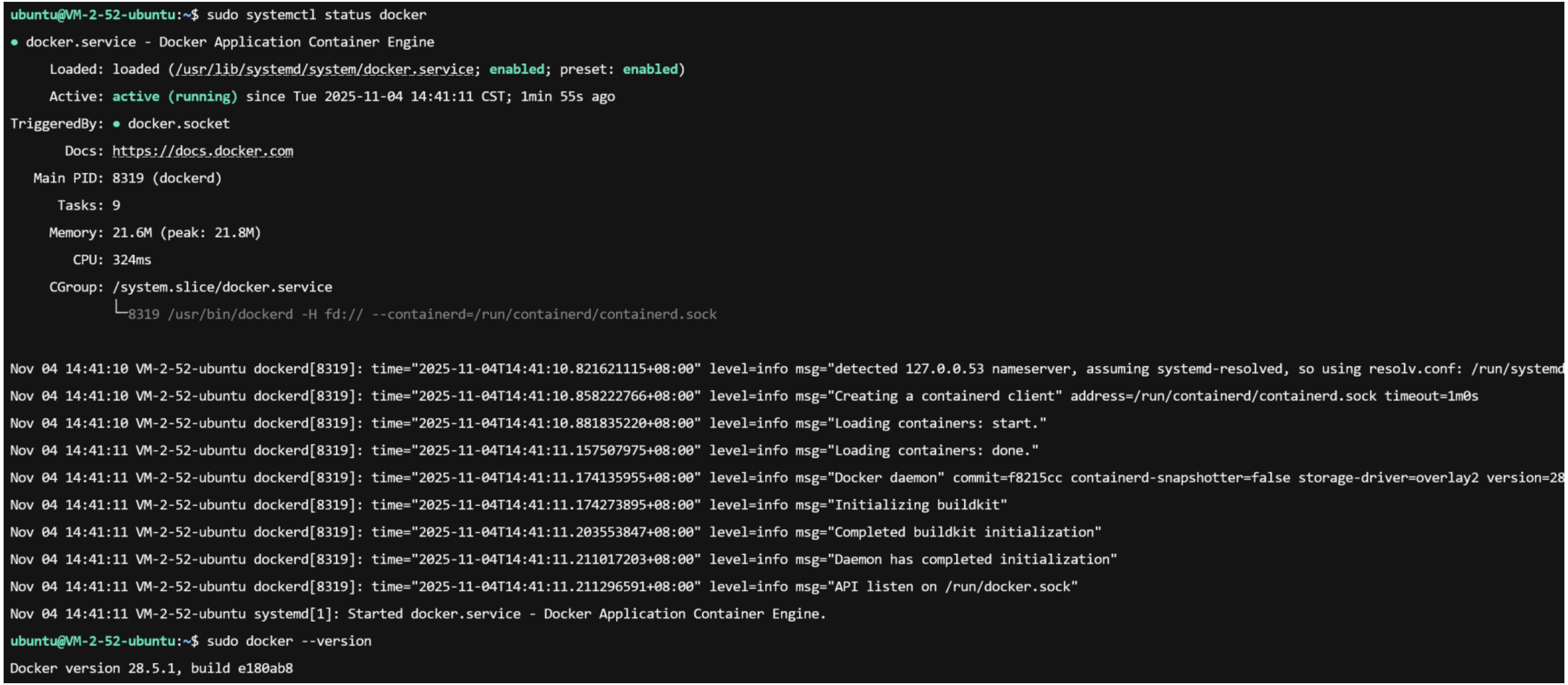
Docker Engine installation (docs link)

- Install packages using the `apt-get` command – verify installation

```
# Make sure to check Docker is installed properly and running
sudo systemctl status docker
sudo docker --version

# If daemon doesn’t run automatically, try starting it manually
sudo systemctl start docker
```

실습

Docker Engine installation (docs link)

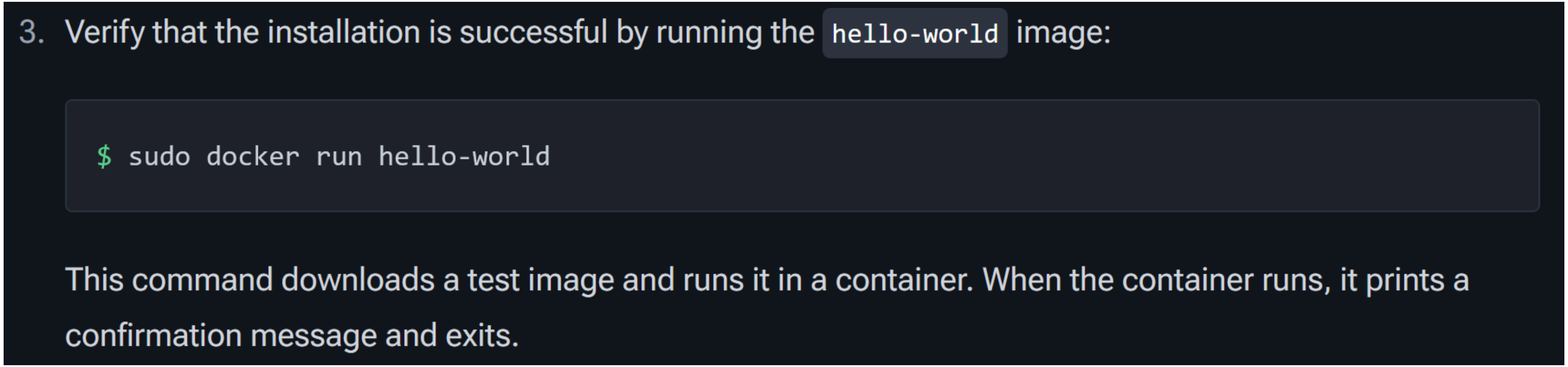


```
ubuntu@VM-2-52-ubuntu:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-11-04 14:41:11 CST; 1min 55s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 8319 (dockerd)
      Tasks: 9
     Memory: 21.6M (peak: 21.8M)
        CPU: 324ms
     CGroup: /system.slice/docker.service
             └─8319 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Nov 04 14:41:10 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:10.821621115+08:00" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, so using resolv.conf: /run/systemd
Nov 04 14:41:10 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:10.858222766+08:00" level=info msg="Creating a containerd client" address=/run/containerd/containerd.sock timeout=1m0s
Nov 04 14:41:10 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:10.881835220+08:00" level=info msg="Loading containers: start."
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.157507975+08:00" level=info msg="Loading containers: done."
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.174135955+08:00" level=info msg="Docker daemon" commit=f8215cc containerd-snapshotter=false storage-driver=overlay2 version=28
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.174273895+08:00" level=info msg="Initializing buildkit"
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.203553847+08:00" level=info msg="Completed buildkit initialization"
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.211017203+08:00" level=info msg="Daemon has completed initialization"
Nov 04 14:41:11 VM-2-52-ubuntu dockerd[8319]: time="2025-11-04T14:41:11.211296591+08:00" level=info msg="API listen on /run/docker.sock"
Nov 04 14:41:11 VM-2-52-ubuntu systemd[1]: Started docker.service - Docker Application Container Engine.
ubuntu@VM-2-52-ubuntu:~$ sudo docker --version
Docker version 28.5.1, build e180ab8
```

실습

Docker Engine installation (docs link)

- Test run Docker using sample image



3. Verify that the installation is successful by running the `hello-world` image:

```
$ sudo docker run hello-world
```

This command downloads a test image and runs it in a container. When the container runs, it prints a confirmation message and exits.

실습

Docker Engine installation (docs link)

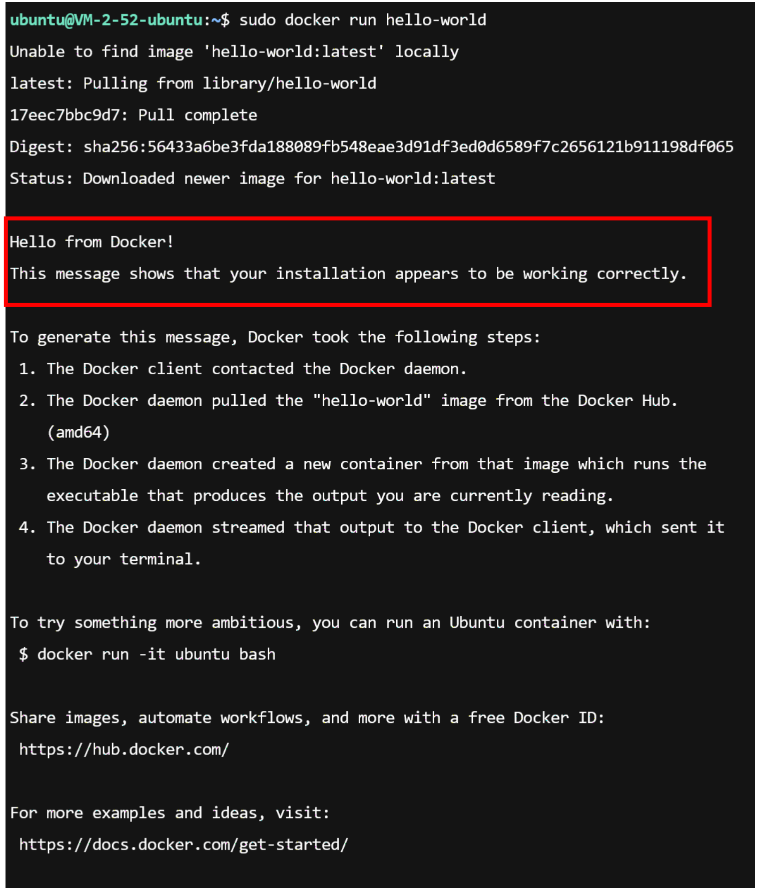
- Test run Docker using sample image

```
# `docker run` checks for the image `hello-world`
# Since we don’t have it locally, it pulls it from the remote repository (DockerHub)
# A confirmation message saying “Hello from Docker!” should be printed
sudo docker run hello-world
```

실습

Docker Engine installation (docs link)

- Test run Docker using sample image

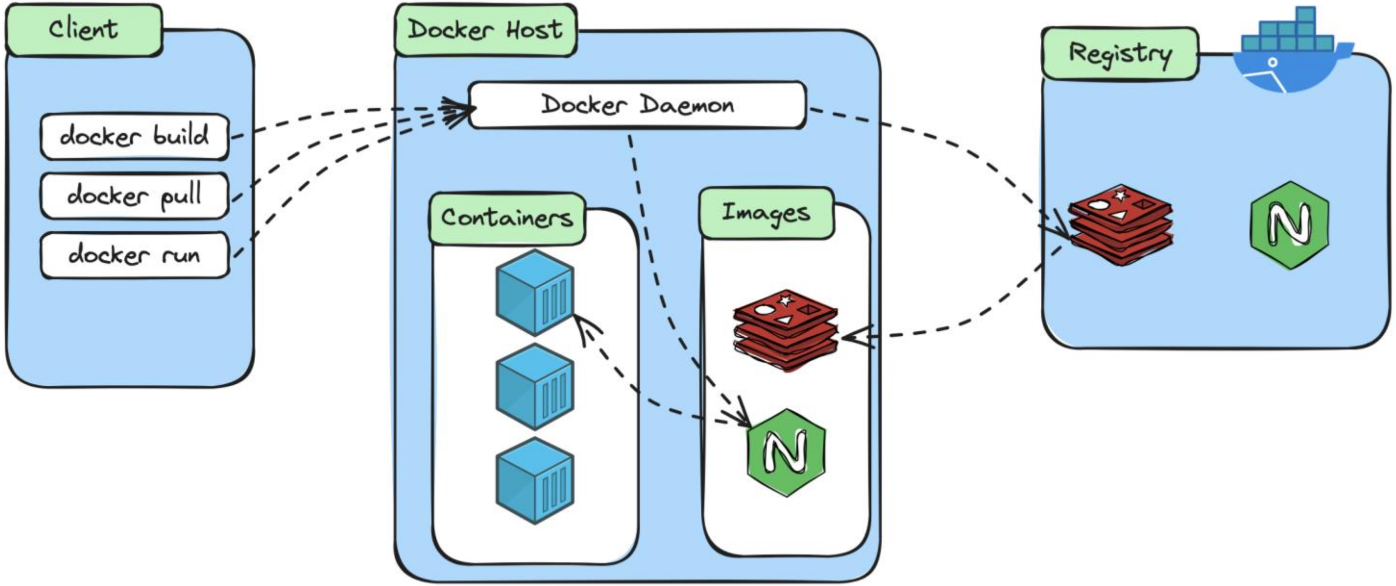


```
ubuntu@VM-2-52-ubuntu:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:56433a6be3fda188089fb548eae3d91df3ed0d6589f7c2656121b911198df065
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

실습

Docker command usage

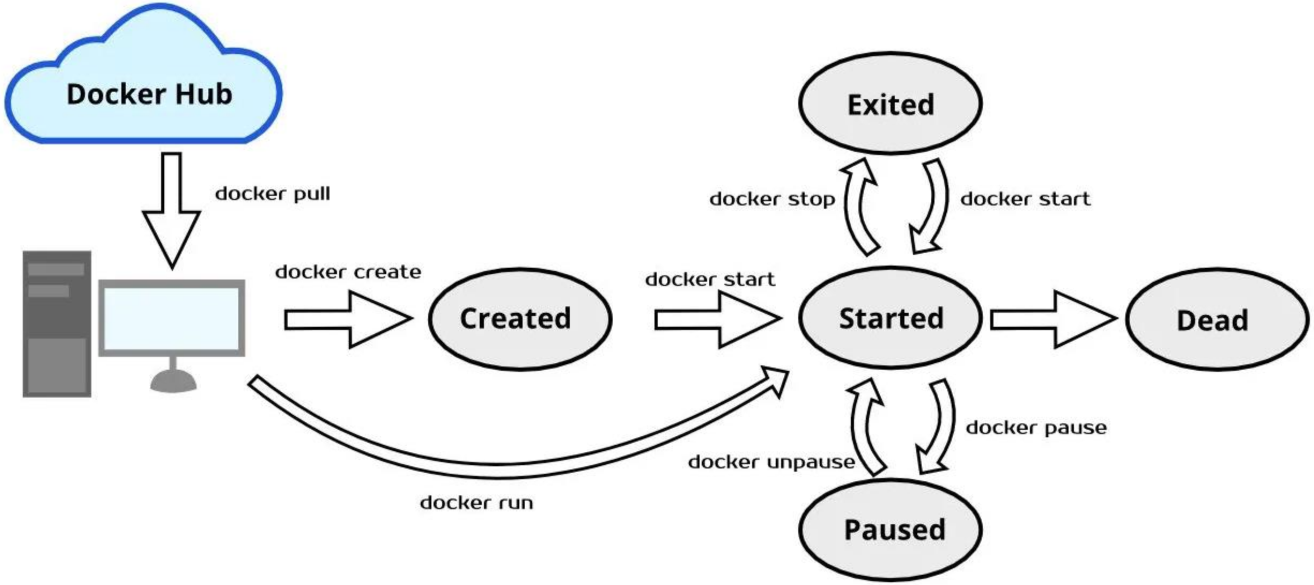
- `docker run` - create and **run a new container** from an image
- `docker pull` - bring image from a remote to local registry (**Dockerhub**)



실습

Docker command usage – docker run

- \`docker run\` = **docker pull** + docker create + docker start



실습

Docker command usage – docker run

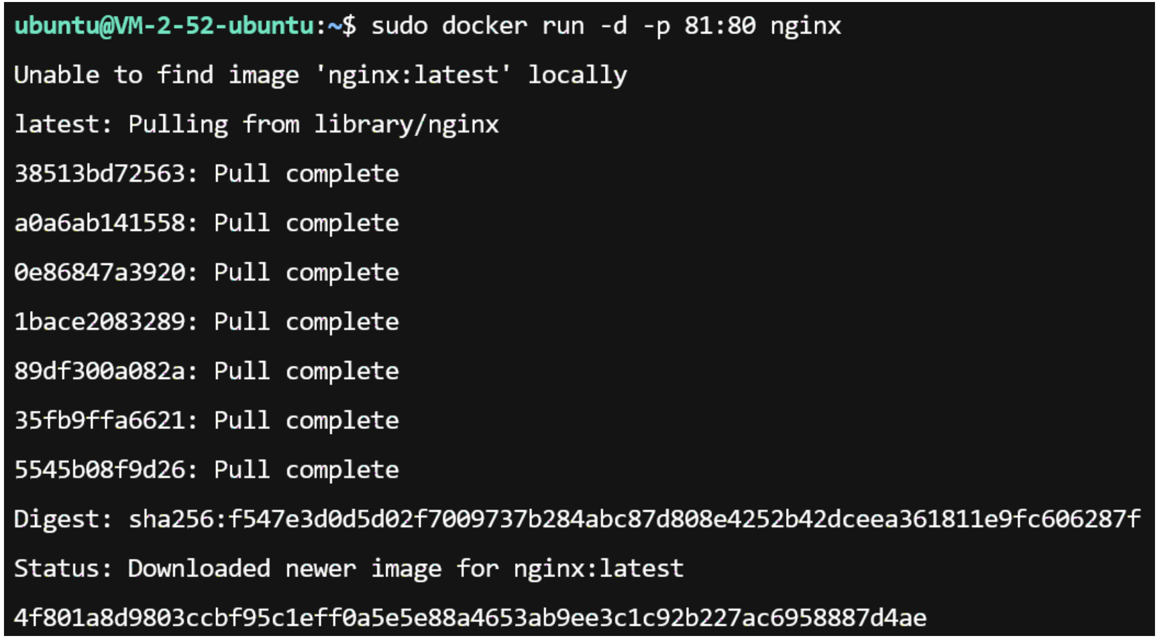
- (ex) deploying a Nginx server as a container

```
# Run the official Nginx image
# `-d` detached (run in background); `-p` publish/expose (bind host port to container port)
sudo docker run -p 81:80 -d nginx

# Displays currently running containers (stopped containers will not be listed!)
sudo docker ps

# `-a` option displays all created containers (both stopped and running)
sudo docker ps -a
```

실습

Docker command usage – docker run

- (ex) deploying a Nginx server as a container

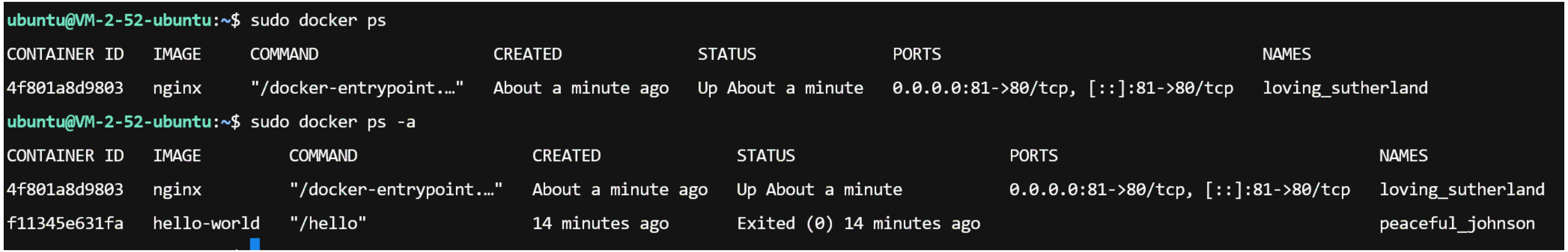


```
ubuntu@VM-2-52-ubuntu:~$ sudo docker run -d -p 81:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
38513bd72563: Pull complete
a0a6ab141558: Pull complete
0e86847a3920: Pull complete
1bace2083289: Pull complete
89df300a082a: Pull complete
35fb9ffa6621: Pull complete
5545b08f9d26: Pull complete
Digest: sha256:f547e3d0d5d02f7009737b284abc87d808e4252b42dceea361811e9fc606287f
Status: Downloaded newer image for nginx:latest
4f801a8d9803ccbf95c1eff0a5e5e88a4653ab9ee3c1c92b227ac6958887d4ae
```

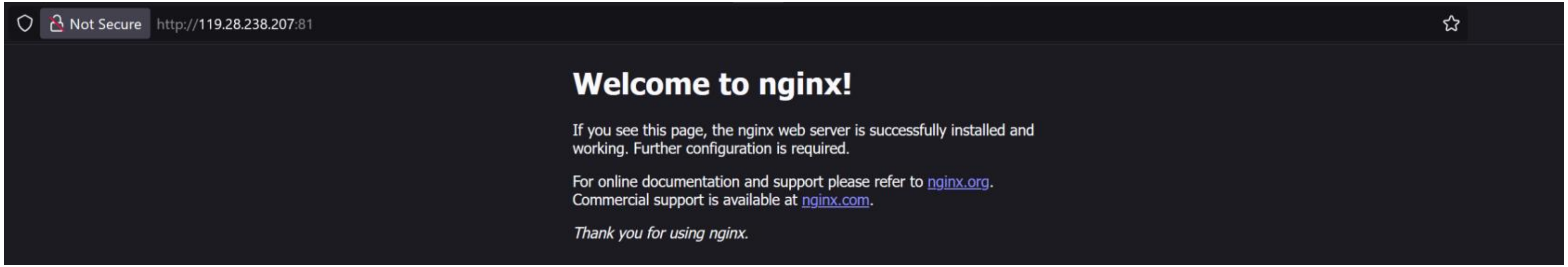
실습

Docker command usage – docker run

- (ex) deploying a Nginx server as a container



```
ubuntu@VM-2-52-ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS                                 NAMES
4f801a8d9803   nginx     "/docker-entrypoint.…"   About a minute ago   Up About a minute   0.0.0.0:81->80/tcp, [::]:81->80/tcp   loving_sutherland
ubuntu@VM-2-52-ubuntu:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED              STATUS                      PORTS                                 NAMES
4f801a8d9803   nginx         "/docker-entrypoint.…"   About a minute ago   Up About a minute           0.0.0.0:81->80/tcp, [::]:81->80/tcp   loving_sutherland
f11345e631fa   hello-world   "/hello"                 14 minutes ago       Exited (0) 14 minutes ago                                         peaceful_johnson
```



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

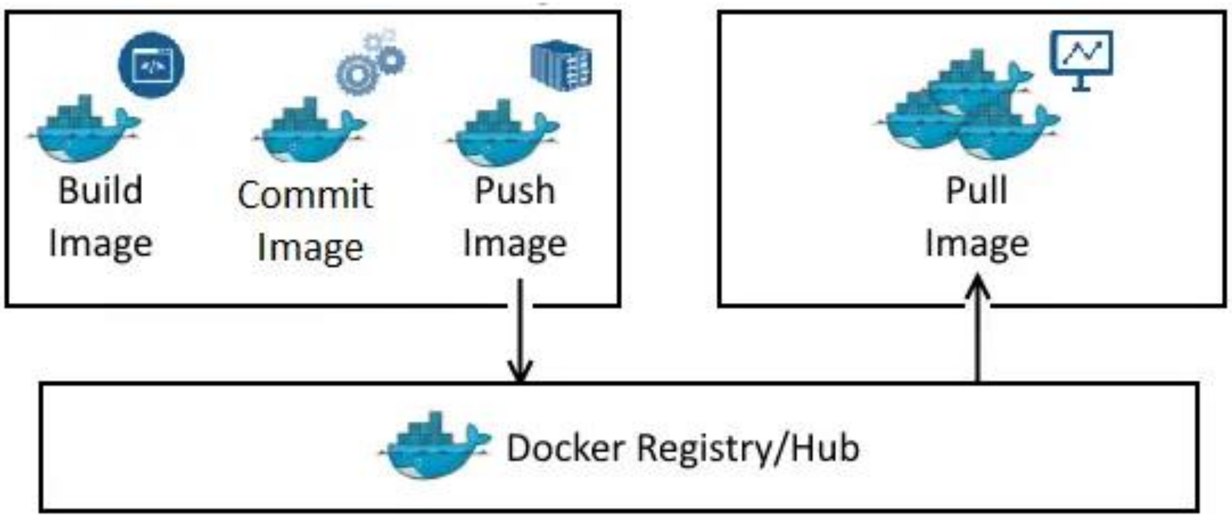
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

실습

Docker command usage – docker pull

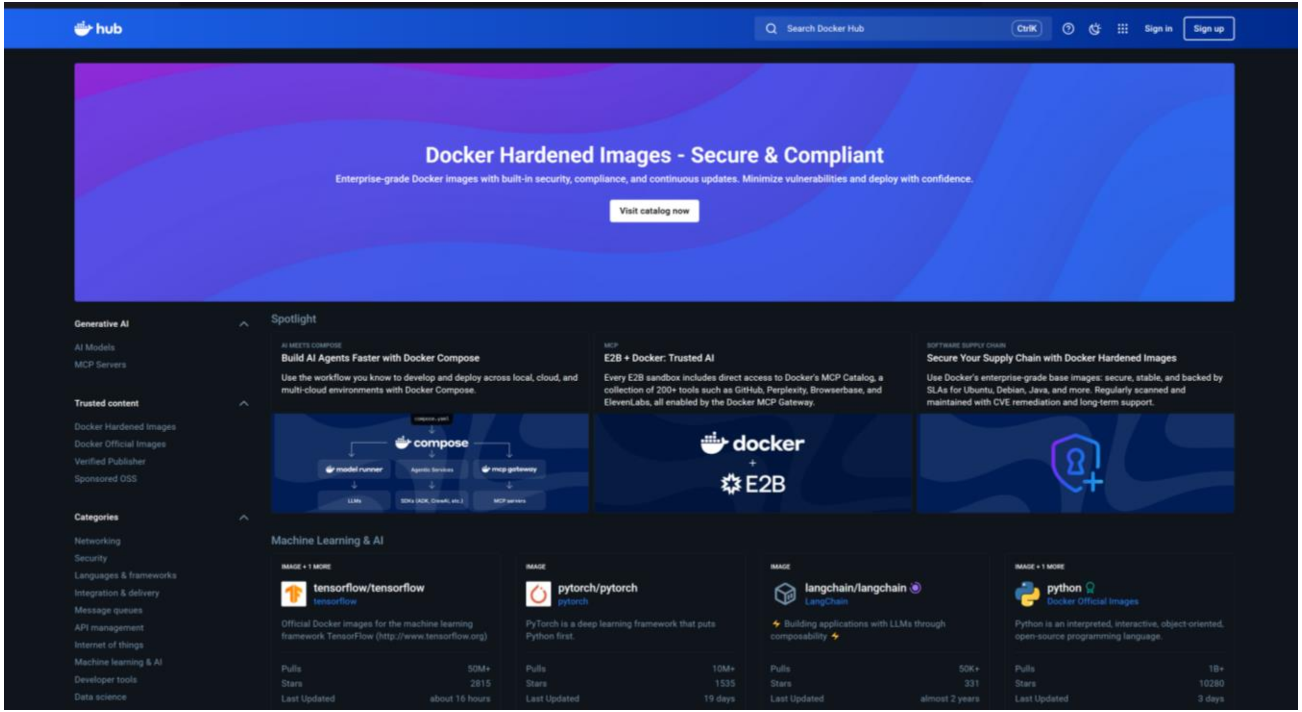
- `docker pull` - bring image from a remote to local registry (**Dockerhub**)



실습

Docker command usage – docker pull

- Docker Hub – registry for saved images (like **Github** for images)



Docker Hardened Images - Secure & Compliant

Enterprise-grade Docker images with built-in security, compliance, and continuous updates. Minimize vulnerabilities and deploy with confidence.

Spotlight

AI MEETS COMPOSE
Build AI Agents Faster with Docker Compose
Use the workflow you know to develop and deploy across local, cloud, and multi-cloud environments with Docker Compose.

MCP
E2B + Docker: Trusted AI
Every E2B sandbox includes direct access to Docker's MCP Catalog, a collection of 200+ tools such as GitHub, Perplexity, Browserbase, and ElevenLabs, all enabled by the Docker MCP Gateway.

SOFTWARE SUPPLY CHAIN
Secure Your Supply Chain with Docker Hardened Images
Use Docker's enterprise-grade base images: secure, stable, and backed by SLAs for Ubuntu, Debian, Java, and more. Regularly scanned and maintained with CVE remediation and long-term support.

Machine Learning & AI

tensorflow/tensorflow
tensorflow
Official Docker images for the machine learning framework TensorFlow (http://www.tensorflow.org)
Pulls 50M+
Stars 2815
Last Updated about 16 hours

pytorch/pytorch
pytorch
PyTorch is a deep learning framework that puts Python first.
Pulls 10M+
Stars 1535
Last Updated 19 days

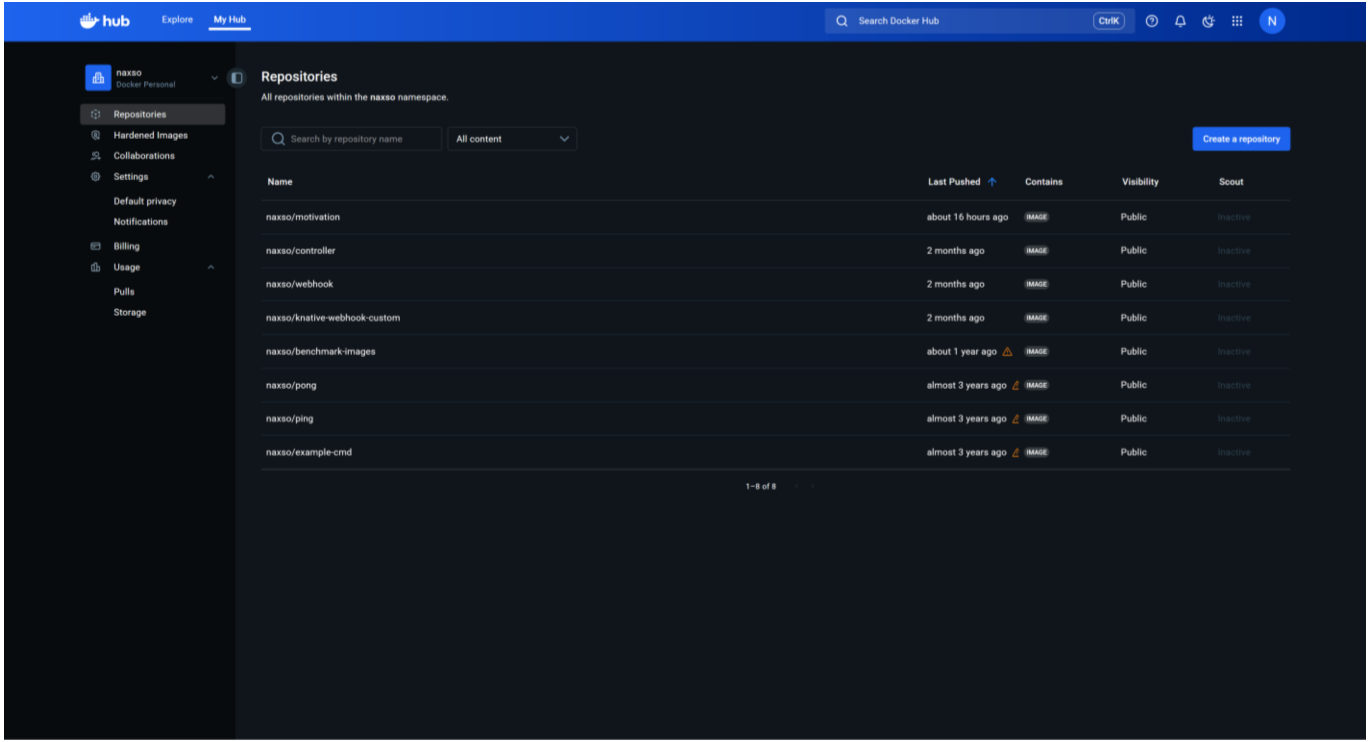
langchain/langchain
LangChain
⚡ Building applications with LLMs through composability ⚡
Pulls 50K+
Stars 331
Last Updated almost 2 years

python
Docker Official Images
Python is an interpreted, interactive, object-oriented, open-source programming language.
Pulls 1B+
Stars 10280
Last Updated 3 days

실습

Docker command usage – docker pull

- Docker Hub – registry for saved images



Repositories

All repositories within the **naxso** namespace.

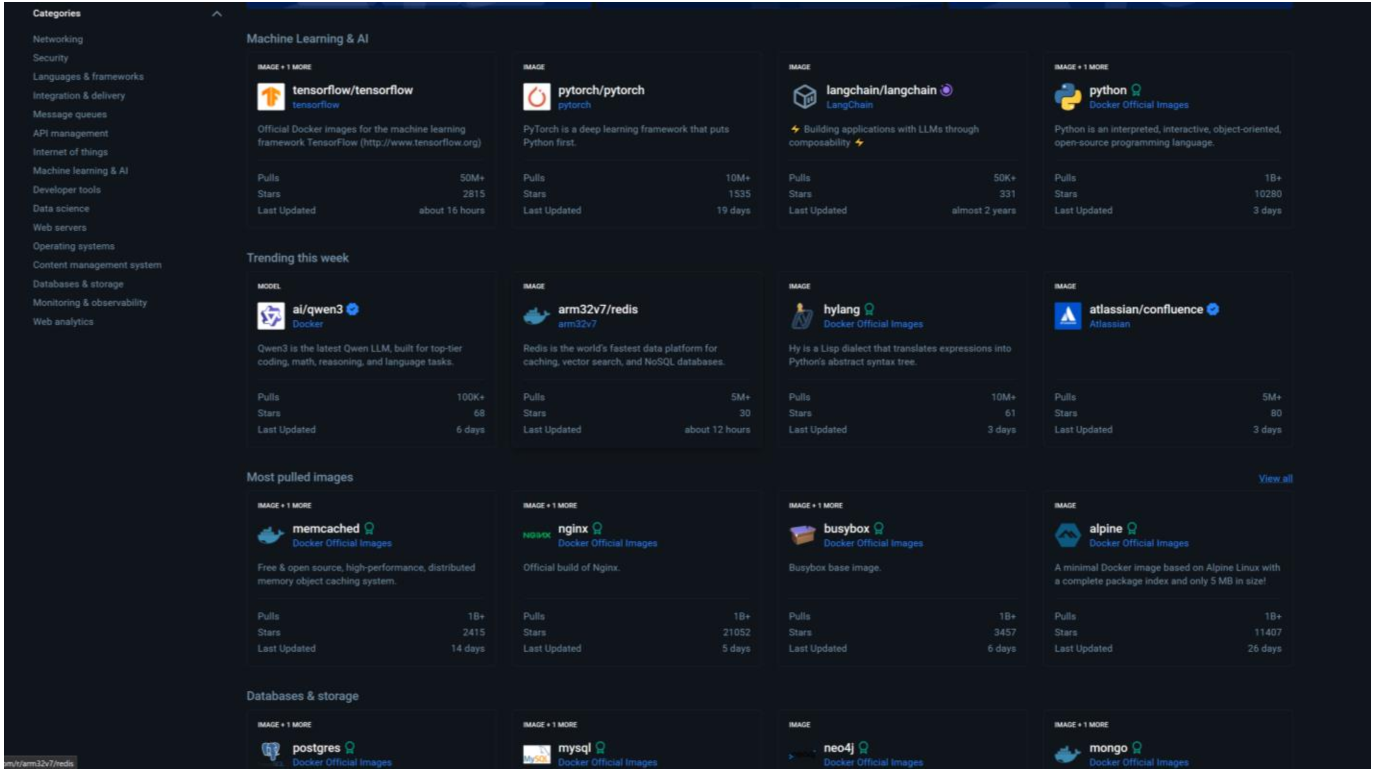
Name	Last Pushed	Contains	Visibility	Scout
naxso/motivation	about 16 hours ago	IMAGE	Public	Inactive
naxso/controller	2 months ago	IMAGE	Public	Inactive
naxso/webhook	2 months ago	IMAGE	Public	Inactive
naxso/knative-webhook-custom	2 months ago	IMAGE	Public	Inactive
naxso/benchmark-images	about 1 year ago ⚠	IMAGE	Public	Inactive
naxso/pong	almost 3 years ago ⚠	IMAGE	Public	Inactive
naxso/ping	almost 3 years ago ⚠	IMAGE	Public	Inactive
naxso/example-cmd	almost 3 years ago ⚠	IMAGE	Public	Inactive

1–8 of 8

실습

Docker command usage – docker pull

- Docker Hub – registry for saved images



Categories

- Networking
- Security
- Languages & frameworks
- Integration & delivery
- Message queues
- API management
- Internet of things
- Machine learning & AI
- Developer tools
- Data science
- Web servers
- Operating systems
- Content management system
- Databases & storage
- Monitoring & observability
- Web analytics

Machine Learning & AI

IMAGE + 1 MORE
tensorflow/tensorflow
tensorflow
Official Docker images for the machine learning framework TensorFlow (http://www.tensorflow.org)
Pulls 50M+
Stars 2815
Last Updated about 16 hours

IMAGE
pytorch/pytorch
pytorch
PyTorch is a deep learning framework that puts Python first.
Pulls 10M+
Stars 1535
Last Updated 19 days

IMAGE
langchain/langchain
LangChain
⚡ Building applications with LLMs through composability ⚡
Pulls 50K+
Stars 331
Last Updated almost 2 years

IMAGE + 1 MORE
python
Docker Official Images
Python is an interpreted, interactive, object-oriented, open-source programming language.
Pulls 1B+
Stars 10280
Last Updated 3 days

Trending this week

MODEL
ai/qwen3
Docker
Qwen3 is the latest Qwen LLM, built for top-tier coding, math, reasoning, and language tasks.
Pulls 100K+
Stars 68
Last Updated 6 days

IMAGE
arm32v7/redis
arm32v7
Redis is the world's fastest data platform for caching, vector search, and NoSQL databases.
Pulls 5M+
Stars 30
Last Updated about 12 hours

IMAGE
hylang
Docker Official Images
Hy is a Lisp dialect that translates expressions into Python's abstract syntax tree.
Pulls 10M+
Stars 61
Last Updated 3 days

IMAGE
atlassian/confluence
Atlassian
Pulls 5M+
Stars 80
Last Updated 3 days

Most pulled images

View all

IMAGE + 1 MORE
memcached
Docker Official Images
Free & open source, high-performance, distributed memory object caching system.
Pulls 1B+
Stars 2415
Last Updated 14 days

IMAGE + 1 MORE
nginx
Docker Official Images
Official build of Nginx.
Pulls 1B+
Stars 21052
Last Updated 5 days

IMAGE + 1 MORE
busybox
Docker Official Images
Busybox base image.
Pulls 1B+
Stars 3457
Last Updated 6 days

IMAGE
alpine
Docker Official Images
A minimal Docker image based on Alpine Linux with a complete package index and only 5 MB in size!
Pulls 1B+
Stars 11407
Last Updated 26 days

Databases & storage

IMAGE + 1 MORE
postgres
Docker Official Images

IMAGE + 1 MORE
mysql
Docker Official Images

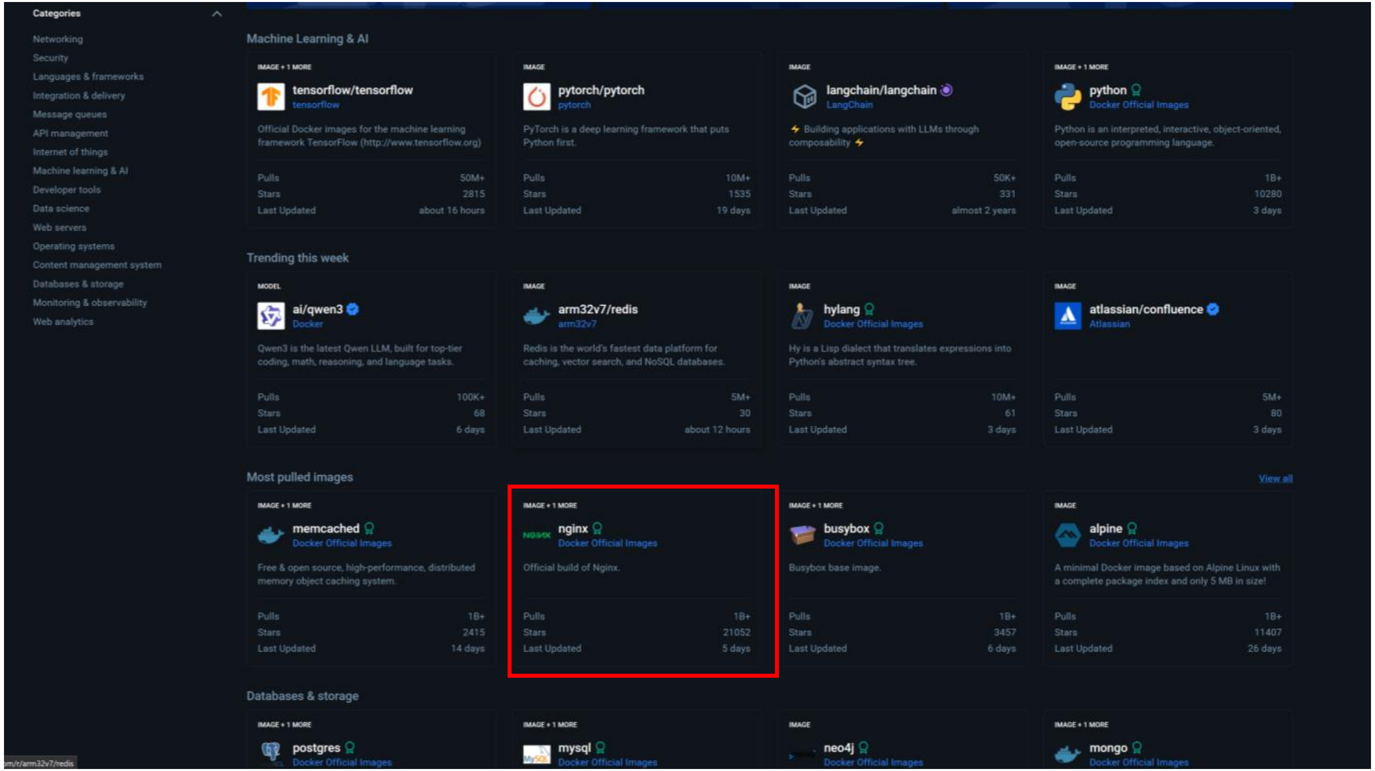
IMAGE
neo4j
Docker Official Images

IMAGE + 1 MORE
mongo
Docker Official Images

실습

Docker command usage – docker pull

- Docker Hub – registry for saved images



Machine Learning & AI

IMAGE + 1 MORE
tensorflow/tensorflow
tensorflow
Official Docker images for the machine learning framework TensorFlow (http://www.tensorflow.org)
Pulls 50M+
Stars 2815
Last Updated about 16 hours

IMAGE
pytorch/pytorch
pytorch
PyTorch is a deep learning framework that puts Python first.
Pulls 10M+
Stars 1535
Last Updated 19 days

IMAGE
langchain/langchain
LangChain
⚡ Building applications with LLMs through composability ⚡
Pulls 50K+
Stars 331
Last Updated almost 2 years

IMAGE + 1 MORE
python
Docker Official Images
Python is an interpreted, interactive, object-oriented, open-source programming language.
Pulls 1B+
Stars 10280
Last Updated 3 days

Trending this week

MODEL
ai/qwen3
Docker
Qwen3 is the latest Qwen LLM, built for top-tier coding, math, reasoning, and language tasks.
Pulls 100K+
Stars 68
Last Updated 6 days

IMAGE
arm32v7/redis
arm32v7
Redis is the world's fastest data platform for caching, vector search, and NoSQL databases.
Pulls 5M+
Stars 30
Last Updated about 12 hours

IMAGE
hylang
Docker Official Images
Hy is a Lisp dialect that translates expressions into Python's abstract syntax tree.
Pulls 10M+
Stars 61
Last Updated 3 days

IMAGE
atlassian/confluence
Atlassian
Pulls 5M+
Stars 80
Last Updated 3 days

Most pulled images
View all

IMAGE + 1 MORE
memcached
Docker Official Images
Free & open source, high-performance, distributed memory object caching system.
Pulls 1B+
Stars 2415
Last Updated 14 days

IMAGE + 1 MORE
nginx
Docker Official Images
Official build of Nginx.
Pulls 1B+
Stars 21052
Last Updated 5 days

IMAGE + 1 MORE
busybox
Docker Official Images
Busybox base image.
Pulls 1B+
Stars 3457
Last Updated 6 days

IMAGE
alpine
Docker Official Images
A minimal Docker image based on Alpine Linux with a complete package index and only 5 MB in size!
Pulls 1B+
Stars 11407
Last Updated 26 days

Databases & storage

IMAGE + 1 MORE
postgres
Docker Official Images

IMAGE + 1 MORE
mysql
Docker Official Images

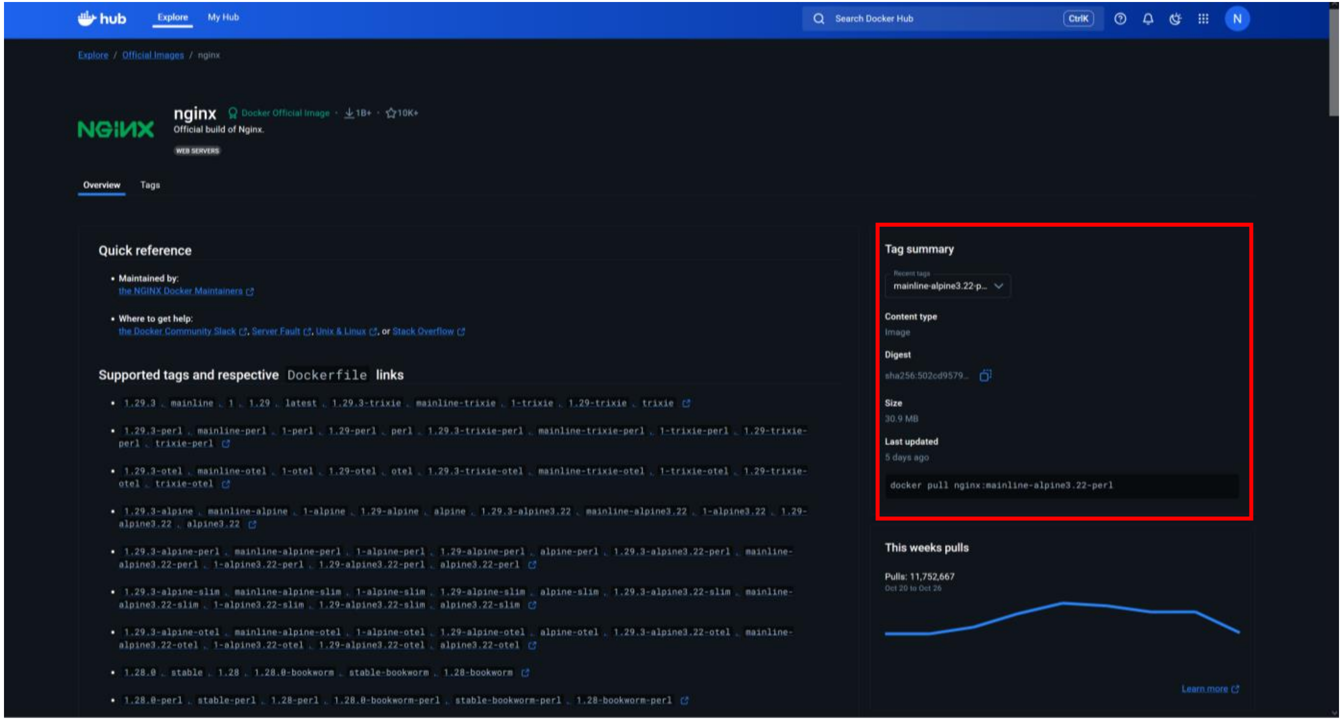
IMAGE
neo4j
Docker Official Images

IMAGE + 1 MORE
mongo
Docker Official Images

실습

Docker command usage – docker pull

- Docker Hub – https://hub.docker.com/_/nginx



nginx Docker Official Image · 1B+ · 10K+

Official build of Nginx.

WEB SERVERS

Overview Tags

Quick reference

- **Maintained by:**
  the NGINX Docker Maintainers
- **Where to get help:**
  the Docker Community Slack, Server Fault, Unix & Linux, or Stack Overflow

Supported tags and respective `Dockerfile` links

- `1.29.3`, `mainline`, `1`, `1.29`, `latest`, `1.29.3-trixie`, `mainline-trixie`, `1-trixie`, `1.29-trixie`, `trixie`
- `1.29.3-perl`, `mainline-perl`, `1-perl`, `1.29-perl`, `perl`, `1.29.3-trixie-perl`, `mainline-trixie-perl`, `1-trixie-perl`, `1.29-trixie-perl`, `trixie-perl`
- `1.29.3-otel`, `mainline-otel`, `1-otel`, `1.29-otel`, `otel`, `1.29.3-trixie-otel`, `mainline-trixie-otel`, `1-trixie-otel`, `1.29-trixie-otel`, `trixie-otel`
- `1.29.3-alpine`, `mainline-alpine`, `1-alpine`, `1.29-alpine`, `alpine`, `1.29.3-alpine3.22`, `mainline-alpine3.22`, `1-alpine3.22`, `1.29-alpine3.22`, `alpine3.22`
- `1.29.3-alpine-perl`, `mainline-alpine-perl`, `1-alpine-perl`, `1.29-alpine-perl`, `alpine-perl`, `1.29.3-alpine3.22-perl`, `mainline-alpine3.22-perl`, `1-alpine3.22-perl`, `1.29-alpine3.22-perl`, `alpine3.22-perl`
- `1.29.3-alpine-slim`, `mainline-alpine-slim`, `1-alpine-slim`, `1.29-alpine-slim`, `alpine-slim`, `1.29.3-alpine3.22-slim`, `mainline-alpine3.22-slim`, `1-alpine3.22-slim`, `1.29-alpine3.22-slim`, `alpine3.22-slim`
- `1.29.3-alpine-otel`, `mainline-alpine-otel`, `1-alpine-otel`, `1.29-alpine-otel`, `alpine-otel`, `1.29.3-alpine3.22-otel`, `mainline-alpine3.22-otel`, `1-alpine3.22-otel`, `1.29-alpine3.22-otel`, `alpine3.22-otel`
- `1.28.0`, `stable`, `1.28`, `1.28.0-bookworm`, `stable-bookworm`, `1.28-bookworm`
- `1.28.0-perl`, `stable-perl`, `1.28-perl`, `1.28.0-bookworm-perl`, `stable-bookworm-perl`, `1.28-bookworm-perl`

Tag summary

Recent tags: mainline-alpine3.22-p...

Content type
Image

Digest
sha256:502cd9579...

Size
30.9 MB

Last updated
5 days ago

```
docker pull nginx:mainline-alpine3.22-perl
```

This weeks pulls

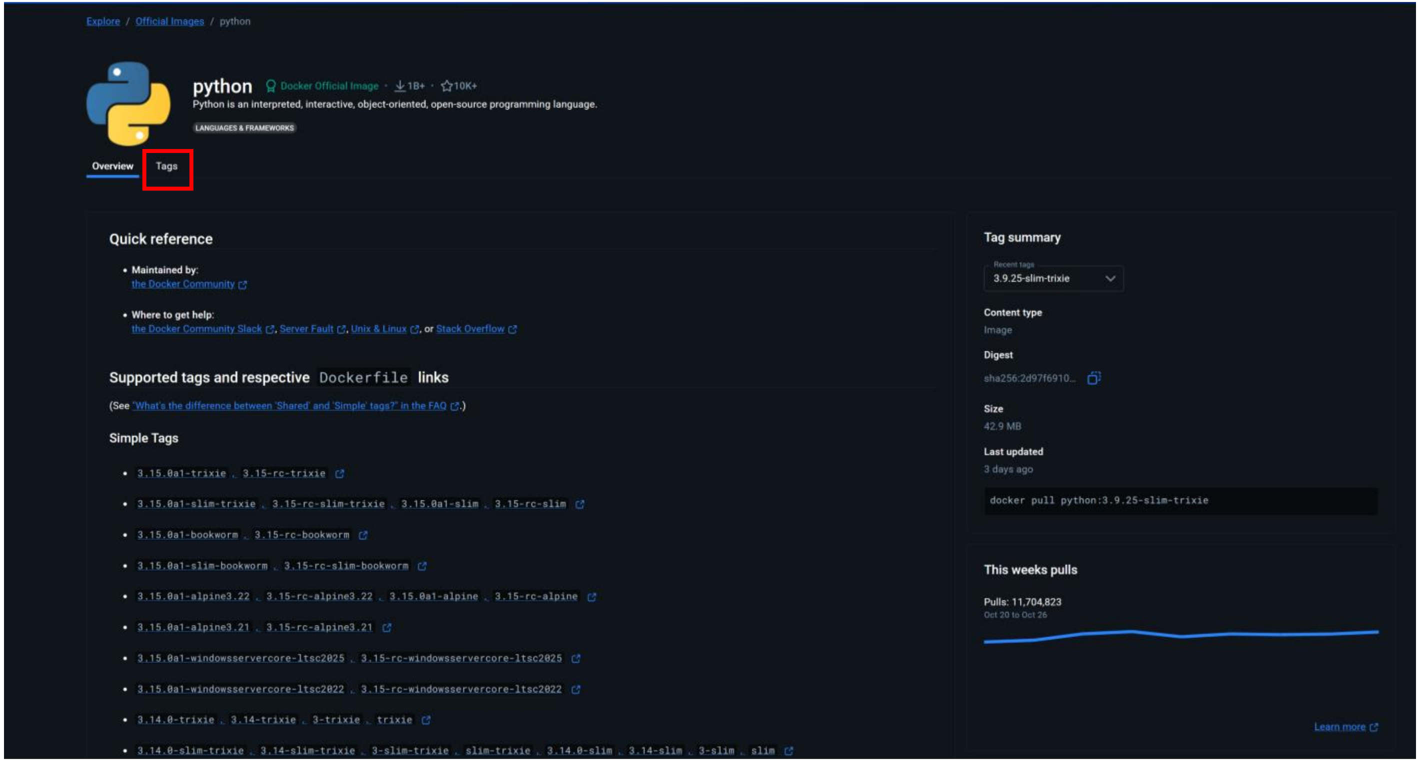
Pulls: 11,752,667
Oct 20 to Oct 26

Learn more

실습

Docker command usage – docker pull

- Docker Hub – https://hub.docker.com/_/python



Explore / Official Images / python

python Docker Official Image · 1B+ · 10K+

Python is an interpreted, interactive, object-oriented, open-source programming language.

Quick reference

- Maintained by:
  the Docker Community
- Where to get help:
  the Docker Community Slack, Server Fault, Unix & Linux, or Stack Overflow

Supported tags and respective `Dockerfile` links

(See "What's the difference between 'Shared' and 'Simple' tags?" in the FAQ.)

Simple Tags

- `3.15.0a1-trixie`, `3.15-rc-trixie`
- `3.15.0a1-slim-trixie`, `3.15-rc-slim-trixie`, `3.15.0a1-slim`, `3.15-rc-slim`
- `3.15.0a1-bookworm`, `3.15-rc-bookworm`
- `3.15.0a1-slim-bookworm`, `3.15-rc-slim-bookworm`
- `3.15.0a1-alpine3.22`, `3.15-rc-alpine3.22`, `3.15.0a1-alpine`, `3.15-rc-alpine`
- `3.15.0a1-alpine3.21`, `3.15-rc-alpine3.21`
- `3.15.0a1-windowsservercore-ltsc2025`, `3.15-rc-windowsservercore-ltsc2025`
- `3.15.0a1-windowsservercore-ltsc2022`, `3.15-rc-windowsservercore-ltsc2022`
- `3.14.0-trixie`, `3.14-trixie`, `3-trixie`, `trixie`
- `3.14.0-slim-trixie`, `3.14-slim-trixie`, `3-slim-trixie`, `slim-trixie`, `3.14.0-slim`, `3.14-slim`, `3-slim`, `slim`

Tag summary

Recent tags: 3.9.25-slim-trixie

Content type: Image

Digest: sha256:2d97f6910...

Size: 42.9 MB

Last updated: 3 days ago

```
docker pull python:3.9.25-slim-trixie
```

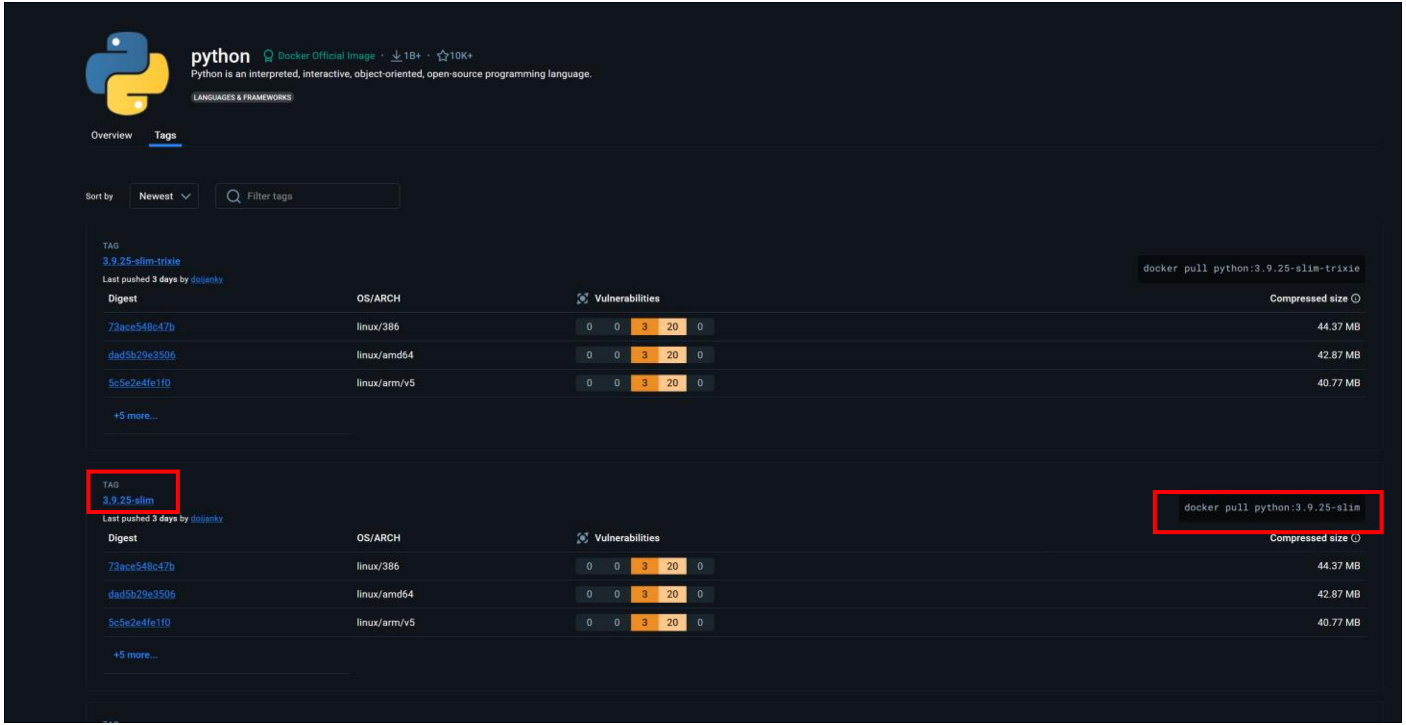
This weeks pulls

Pulls: 11,704,823
Oct 20 to Oct 26

실습

Docker command usage – docker pull

- Docker Hub – https://hub.docker.com/_/python



python Docker Official Image · 1B+ · 10K+

Python is an interpreted, interactive, object-oriented, open-source programming language.

TAG
3.9.25-slim-trixie
Last pushed 3 days by doijanky

```
docker pull python:3.9.25-slim-trixie
```

Digest	OS/ARCH	Vulnerabilities					Compressed size
73ace548c47b	linux/386	0	0	3	20	0	44.37 MB
dad5b29e3506	linux/amd64	0	0	3	20	0	42.87 MB
5c5e2e4fe1f0	linux/arm/v5	0	0	3	20	0	40.77 MB

+5 more...

TAG
3.9.25-slim
Last pushed 3 days by doijanky

```
docker pull python:3.9.25-slim
```

Digest	OS/ARCH	Vulnerabilities					Compressed size
73ace548c47b	linux/386	0	0	3	20	0	44.37 MB
dad5b29e3506	linux/amd64	0	0	3	20	0	42.87 MB
5c5e2e4fe1f0	linux/arm/v5	0	0	3	20	0	40.77 MB

+5 more...

실습

Docker command usage – docker pull

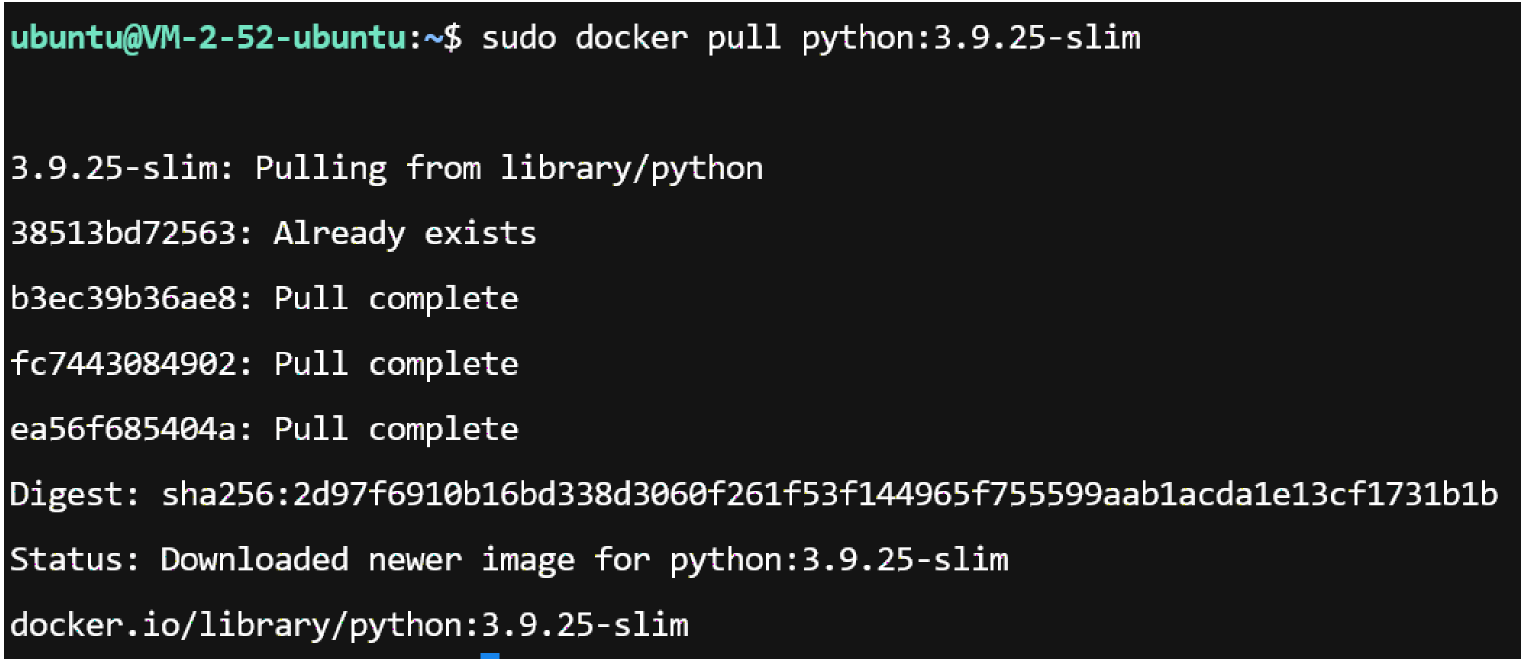
- (Practice) Pull the official Python image and run an interactive container

```
# Pull the official image for Python v3.9.25 (uses the ‘slim’ distro as a base OS)
sudo docker pull python:3.9.25-slim

# Check which images have been pulled into the local registry
sudo docker images

# Create and run a container based on the Python image
# `-it` interactive (binds terminal to container runtime)
sudo docker run -it python:3.9.25-slim
```

실습

Docker command usage – docker pull

- (Practice) Pull the official Python image and run an interactive container

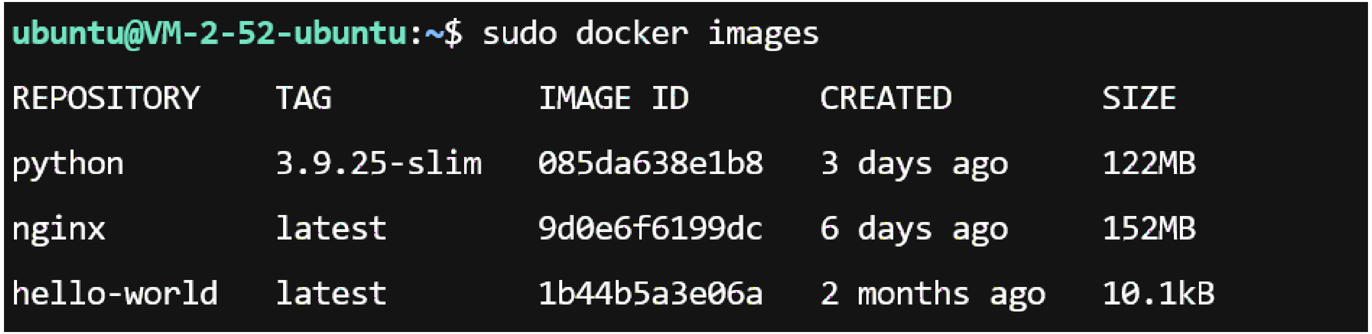


```
ubuntu@VM-2-52-ubuntu:~$ sudo docker pull python:3.9.25-slim


3.9.25-slim: Pulling from library/python
38513bd72563: Already exists
b3ec39b36ae8: Pull complete
fc7443084902: Pull complete
ea56f685404a: Pull complete
Digest: sha256:2d97f6910b16bd338d3060f261f53f144965f755599aab1acda1e13cf1731b1b
Status: Downloaded newer image for python:3.9.25-slim
docker.io/library/python:3.9.25-slim
```

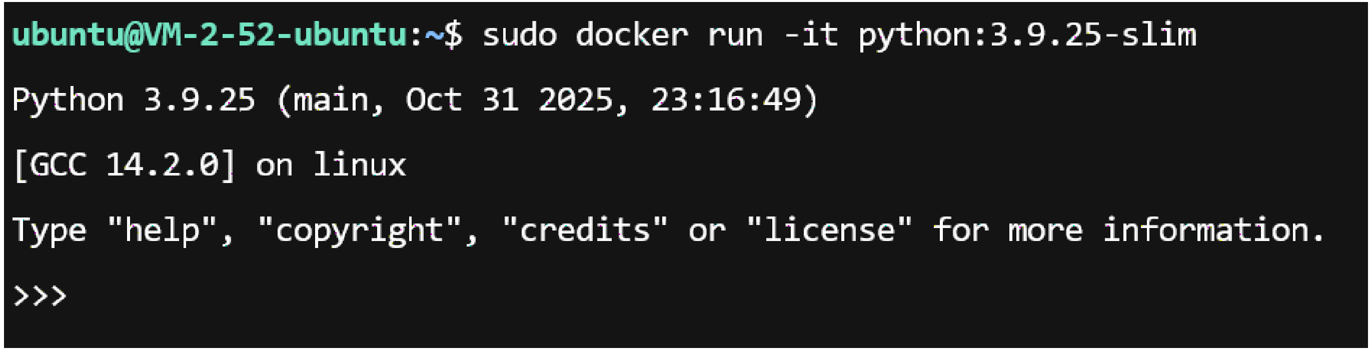
실습

Docker command usage – docker pull

- (Practice) Pull the official Python image and run an interactive container



```
ubuntu@VM-2-52-ubuntu:~$ sudo docker images
REPOSITORY    TAG           IMAGE ID       CREATED         SIZE
python        3.9.25-slim   085da638e1b8   3 days ago      122MB
nginx         latest        9d0e6f6199dc   6 days ago      152MB
hello-world   latest        1b44b5a3e06a   2 months ago    10.1kB
```

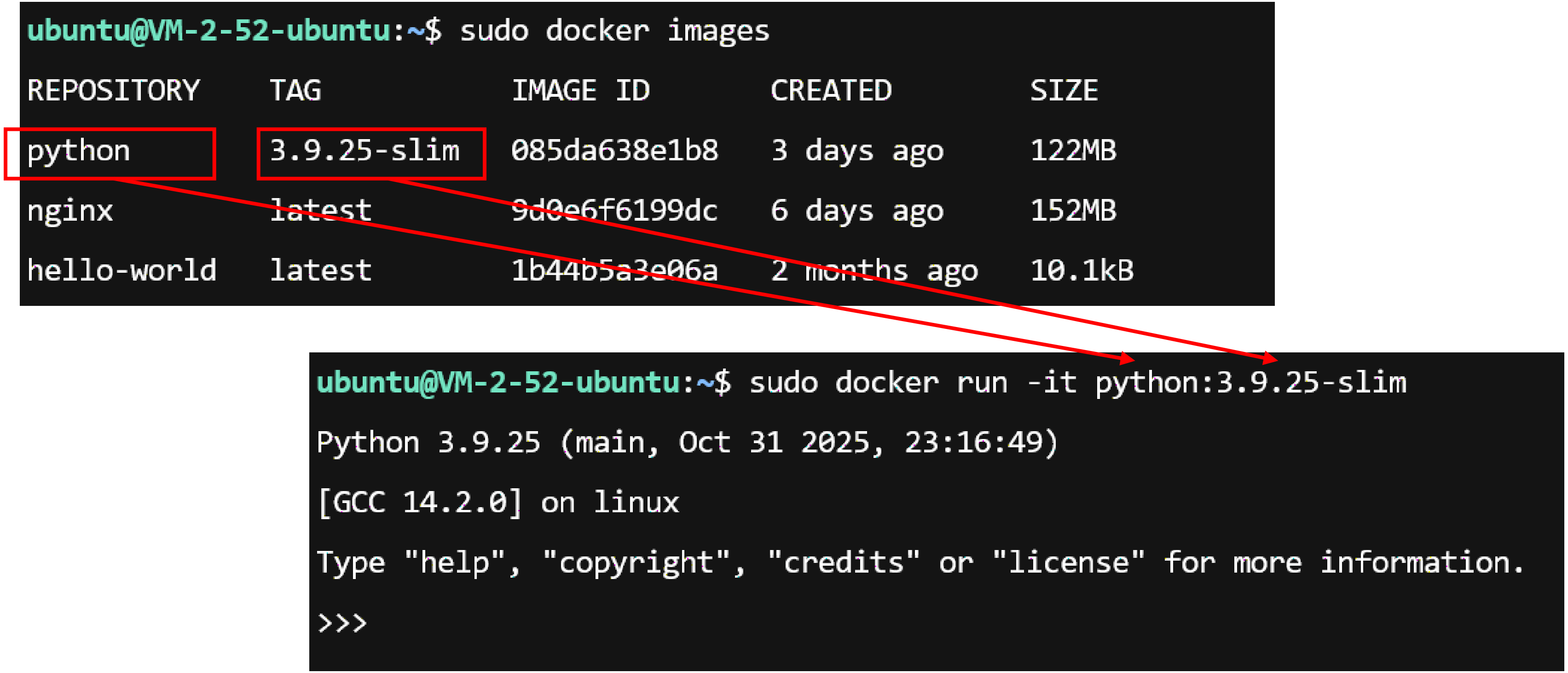


```
ubuntu@VM-2-52-ubuntu:~$ sudo docker run -it python:3.9.25-slim
Python 3.9.25 (main, Oct 31 2025, 23:16:49)
[GCC 14.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

실습

Docker command usage – docker pull

- (Practice) Pull the official Python image and run an interactive container



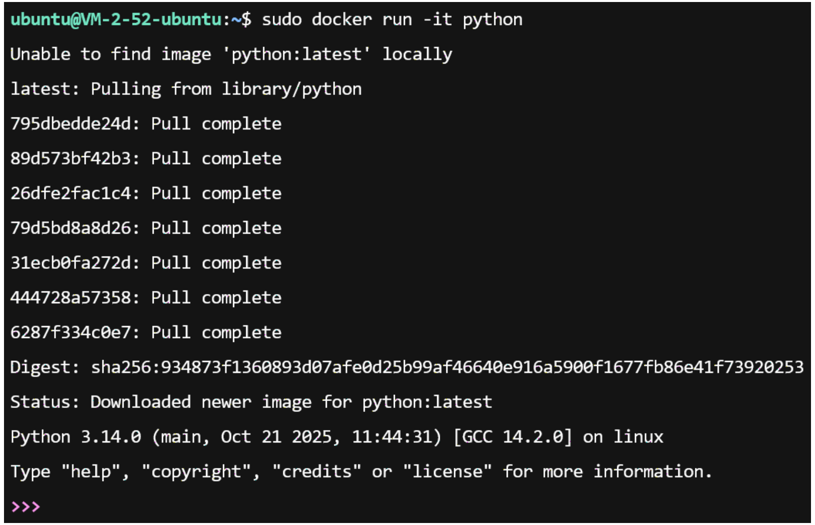
```
ubuntu@VM-2-52-ubuntu:~$ sudo docker images
REPOSITORY    TAG           IMAGE ID       CREATED        SIZE
python        3.9.25-slim   085da638e1b8   3 days ago     122MB
nginx         latest        9d0e6f6199dc   6 days ago     152MB
hello-world   latest        1b44b5a3e06a   2 months ago   10.1kB
```

```
ubuntu@VM-2-52-ubuntu:~$ sudo docker run -it python:3.9.25-slim
Python 3.9.25 (main, Oct 31 2025, 23:16:49)
[GCC 14.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

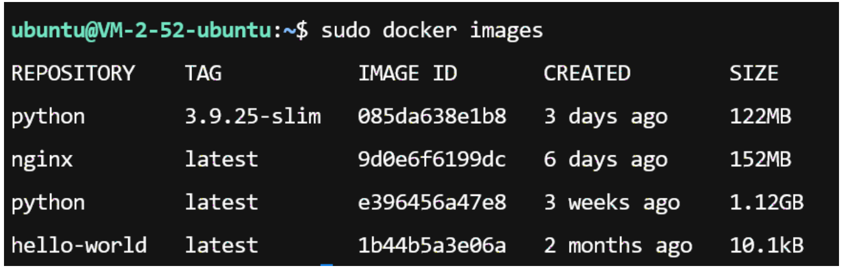
실습

Docker command usage – docker pull

- (Practice) Pull the official Python image and run an interactive container
    - **Make sure to write the exact image name!**



```
ubuntu@VM-2-52-ubuntu:~$ sudo docker run -it python
Unable to find image 'python:latest' locally
latest: Pulling from library/python
795dbedde24d: Pull complete
89d573bf42b3: Pull complete
26dfe2fac1c4: Pull complete
79d5bd8a8d26: Pull complete
31ecb0fa272d: Pull complete
444728a57358: Pull complete
6287f334c0e7: Pull complete
Digest: sha256:934873f1360893d07afe0d25b99af46640e916a5900f1677fb86e41f73920253
Status: Downloaded newer image for python:latest
Python 3.14.0 (main, Oct 21 2025, 11:44:31) [GCC 14.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```



```
ubuntu@VM-2-52-ubuntu:~$ sudo docker images
REPOSITORY    TAG           IMAGE ID       CREATED        SIZE
python        3.9.25-slim   085da638e1b8   3 days ago     122MB
nginx         latest        9d0e6f6199dc   6 days ago     152MB
python        latest        e396456a47e8   3 weeks ago    1.12GB
hello-world   latest        1b44b5a3e06a   2 months ago   10.1kB
```

실습

Conclusion

- Learned about containerization
- Installed Docker engine
- Utilized the basic Docker commands (`docker run` and `docker pull`)

Next time

- We will be building our own custom image (`docker build` and Dockerfile)
- Attaching volumes for data storage
- Understand and utilize Docker networking capabilities

Q&A

Index - Installation

https://docs.docker.com/engine/install/ubuntu/#install-using-the-repository

```
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

echo ₩
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu ₩
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | ₩
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
sudo systemctl status docker
```

Index – docker run

https://docs.docker.com/reference/cli/docker/container/run/

https://docs.docker.com/reference/cli/docker/container/ls/

```
sudo docker run hello-world
```

```
sudo docker run -d -p 81:80 nginx
```

```
sudo docker ps
sudo docker ps -a
```

Index – docker pull

https://docs.docker.com/reference/cli/docker/image/pull/

https://docs.docker.com/reference/cli/docker/image/ls/

```
sudo docker pull python:3.9.25-slim
```

```
sudo docker images
```

```
sudo docker run -it python:3.9.25-slim
```