

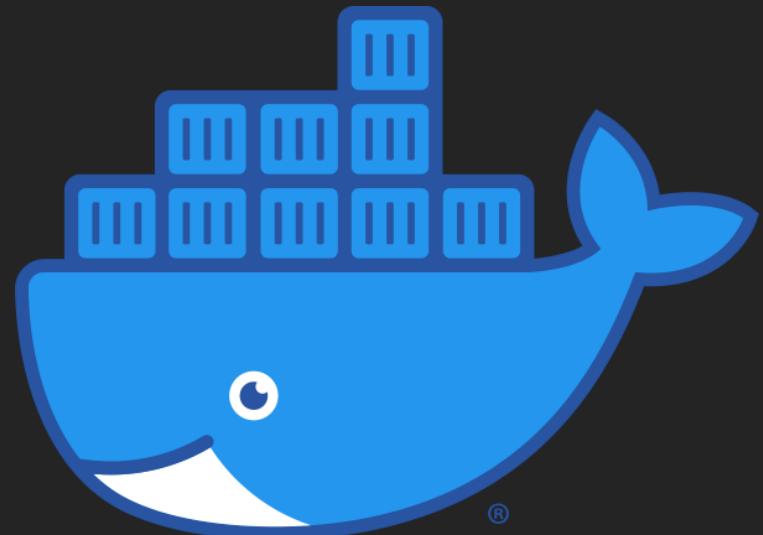
<Infrastructure Engineering>

Bootcamp

Chapter 1

แนะนำบทเรียน

KBTG
KAMPUS
ClassNest



Course Overview

Lecture 1 แนะนำบทเรียน

Lecture 2 ประวัติ Java และเทคโนโลยีเว็บ

Lecture 3 รู้จักกับ Container Technology

Lecture 4 ทำความเข้าใจกับ Docker

Lecture 5 รู้จัก และ ใช้งาน Hub

Lecture 6 ความหมาย และ การใช้งาน Docker Image

Lecture 7 การใช้ DockerFile

Lecture 8 การใช้ Instruction Command

Lecture 9 การสร้าง Docker File

Course Overview

- Lecture 10 การทำงานร่วมกับ Container
- Lecture 11 การใช้งาน Docker พื้นฐาน Container และ Host
- Lecture 12 การใช้งาน Shell ร่วมกับ Docker Container
- Lecture 13 การจัดการ Port ใน Docker
- Lecture 14 การเชื่อมโยงระหว่าง Container ใน Docker
- Lecture 15 Networking ใน Docker
- Lecture 16 เรื่องราวในการจัดการ Storage space
- Lecture 17 Concept และ การใช้งาน Docker Compose
- Lecture 18 การใช้งานกับ Public Repository
- Lecture 19 สรุปบทเรียนสำหรับพื้นฐาน Docker

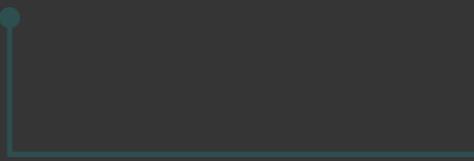
<Infrastructure Engineering>

Bootcamp

Chapter 2

ประวัติ Java และเทคโนโลยีเว็บ







Desktop GUI

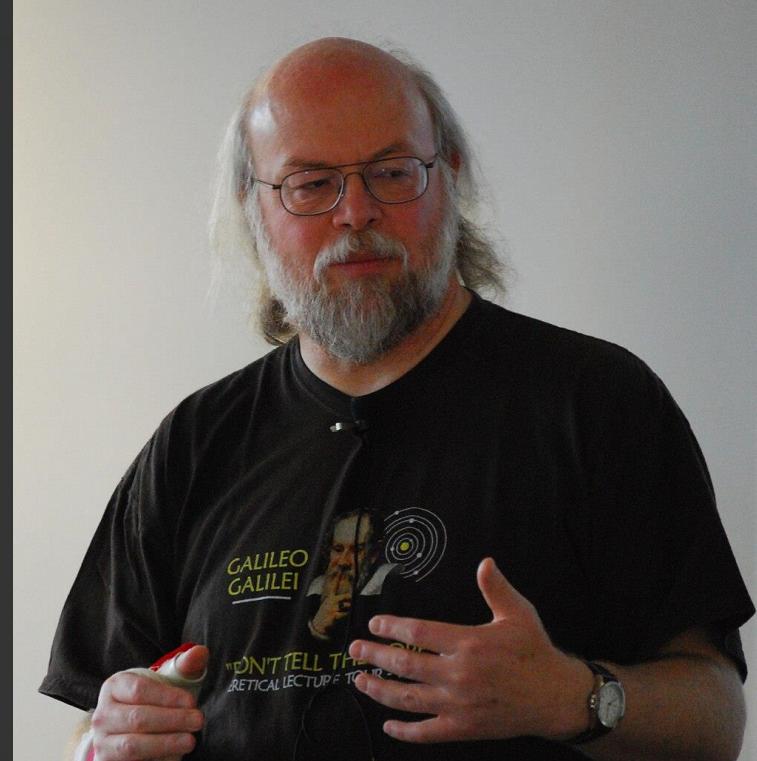
Embedded System

Web Server

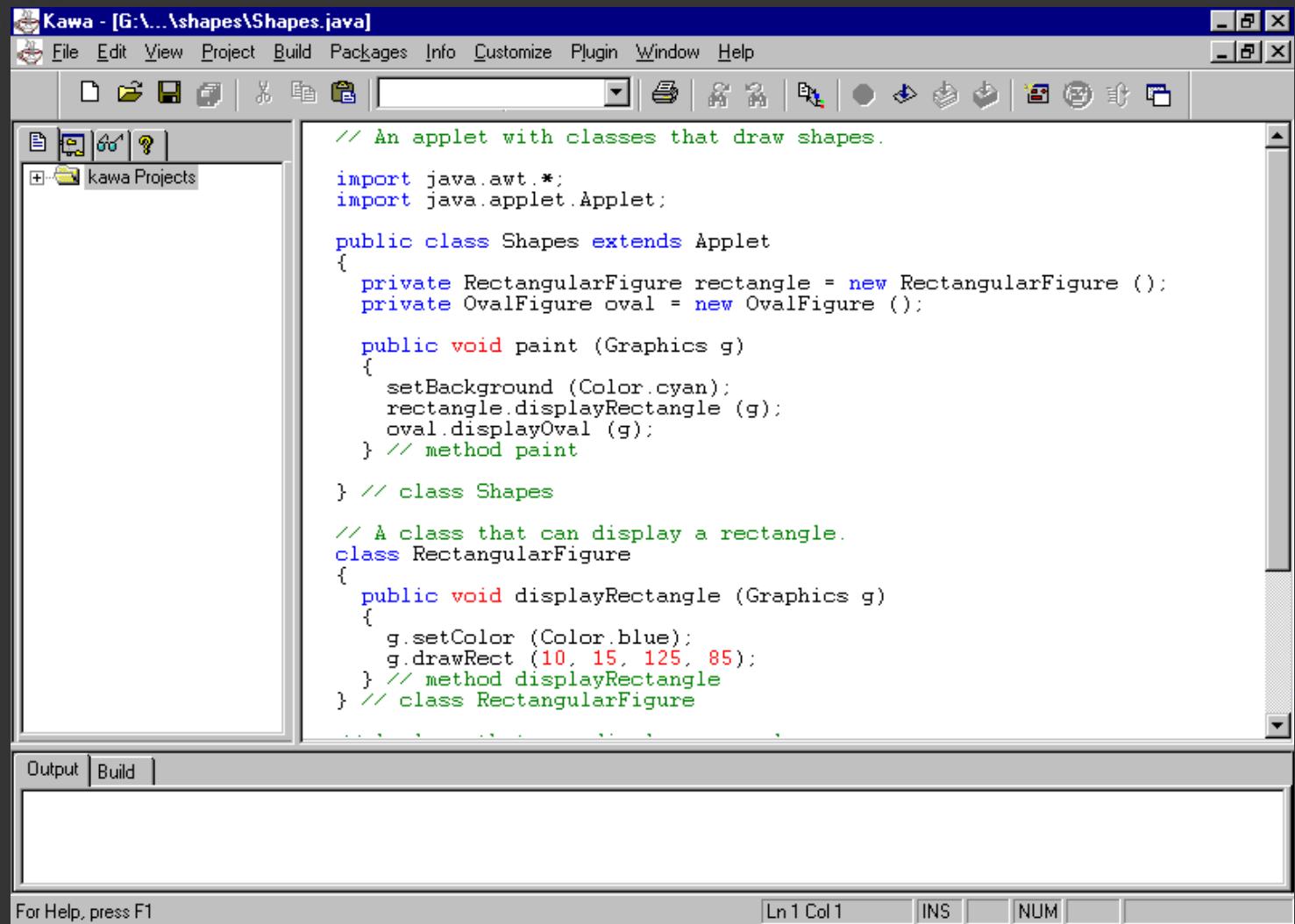
Mobile Application

Web Application

James Gosling

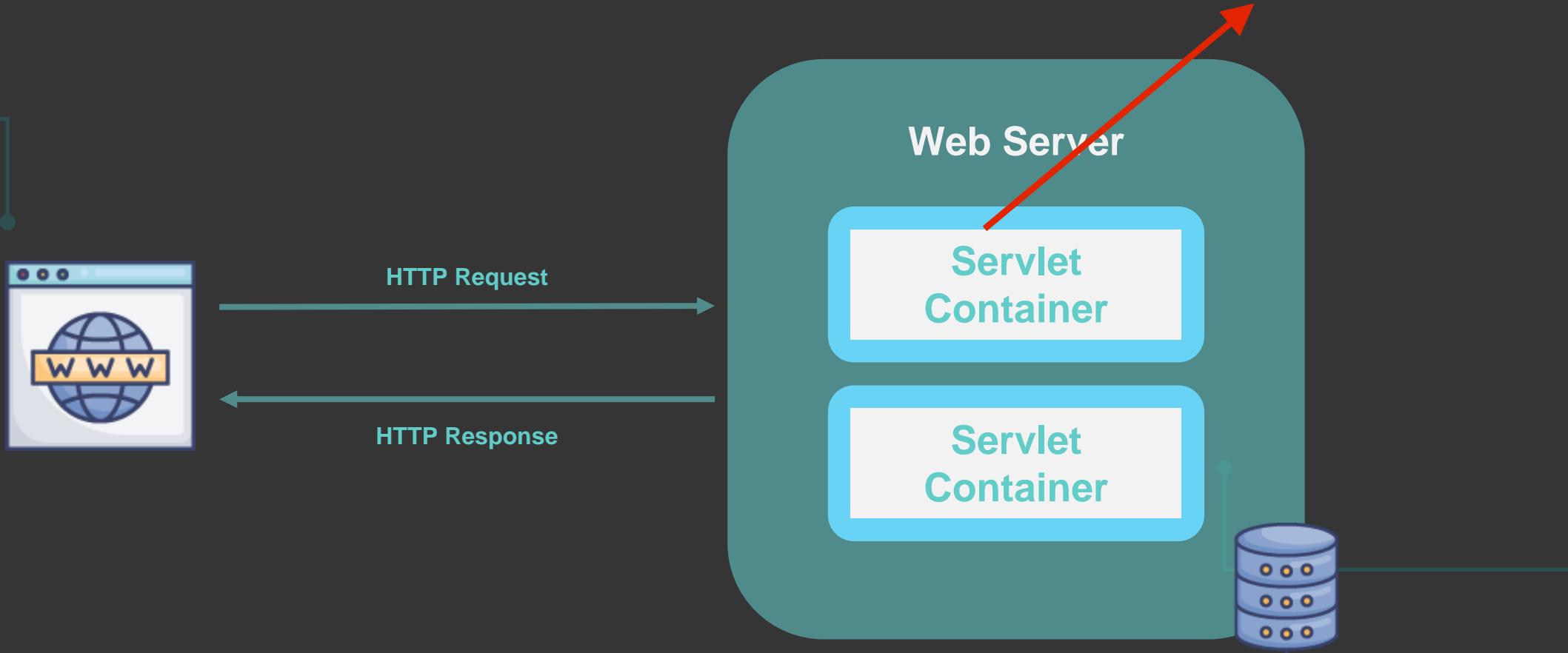


Java Applets



Java Servlets

ข้างในมี code อญ্ত





IBM WebSphere

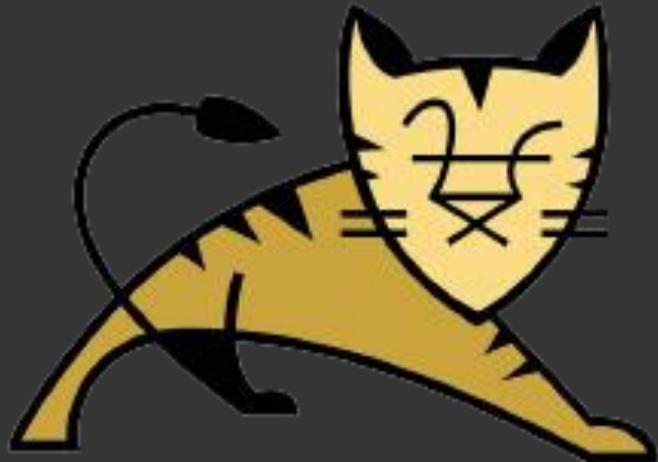
Text

Java Code

IBM WebSphere Container

Java Virtual Machine (JVM)

Apache Tomcat



Lightweight

Open-source web server

Servlet container

Client



Database

```
● ● ●  ~%2 ./mvnw spring-boot:run --quiet
+ demo ./mvnw spring-boot:run --quiet

:: Spring Boot :: (v2.2.4.RELEASE)

2020-02-14 16:16:47.746  INFO 4838 --- [           main] com.example.demo.DemoApplication      : Starting DemoApplication
on Brians-MacBook-Pro.local with PID 4838 (/Users/bclozel/workspace/tmp/demo/target/classes started by bclozel in /Users/bclozel/workspace/tmp/demo)
2020-02-14 16:16:47.748  INFO 4838 --- [           main] com.example.demo.DemoApplication      : No active profile set, falling back to default profiles: default
2020-02-14 16:16:48.272  INFO 4838 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2020-02-14 16:16:48.279  INFO 4838 --- [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2020-02-14 16:16:48.279  INFO 4838 --- [           main] o.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.30]
2020-02-14 16:16:48.323  INFO 4838 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2020-02-14 16:16:48.324  INFO 4838 --- [           main] o.s.web.context.ContextLoader          : Root WebApplicationContext
t: initialization completed in 532 ms
2020-02-14 16:16:48.438  INFO 4838 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2020-02-14 16:16:48.533  INFO 4838 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s)
: 8080 (http) with context path ''
2020-02-14 16:16:48.535  INFO 4838 --- [           main] com.example.demo.DemoApplication      : Started DemoApplication in 1.006 seconds (JVM running for 1.248)
```

Demo

<Infrastructure Engineering>

Bootcamp

Chapter 3

รู้จักกับ Container Technology



What is container technology?





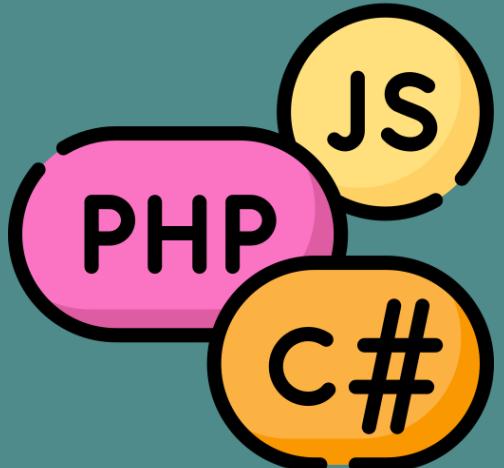


Method to package an application so it can be run, with its dependencies, isolated from other processes





Application Code



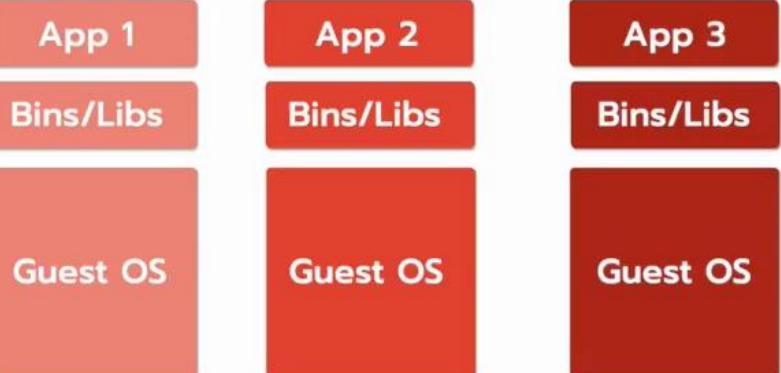
OS



Lib / Bin



Virtual Machines



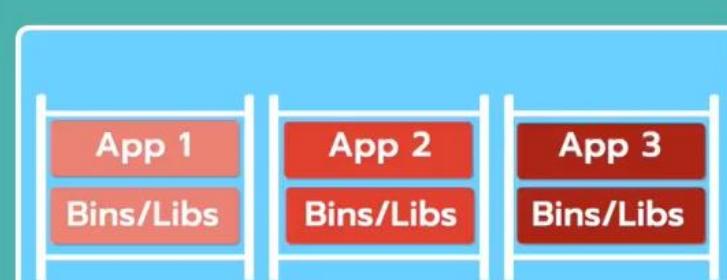
Hypervisor

Host Operating System

Infrastructure



Containers



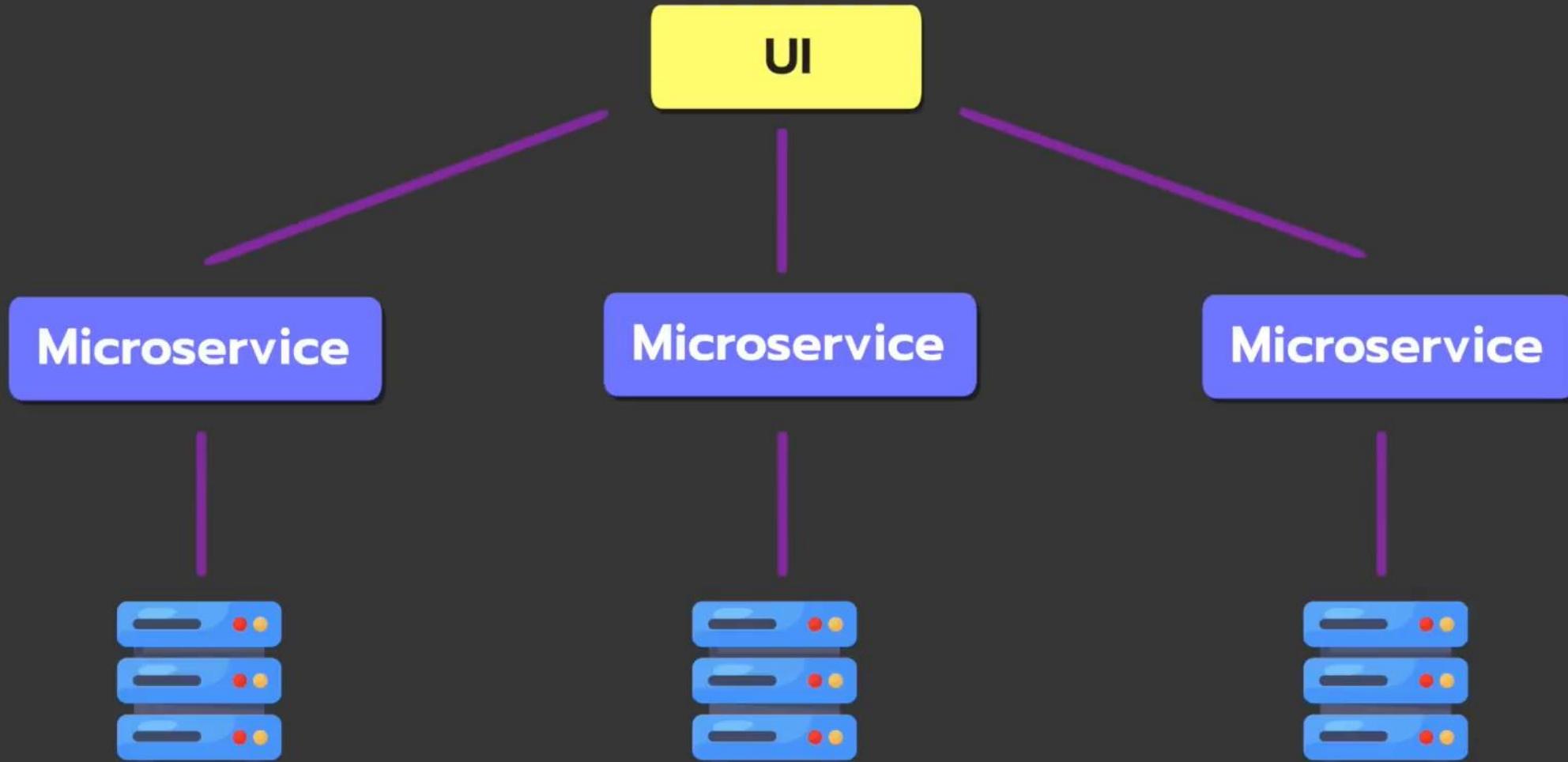
Container Engine

Operating System

Infrastructure



Microservices







Containerization of Front-end



Database Containerization



Containerization of Back-end Services



Scalability and Load Balancing



Containerization of Front-end



**Containerization of
Back-end Services**



**Containerization of
Back-end Services**

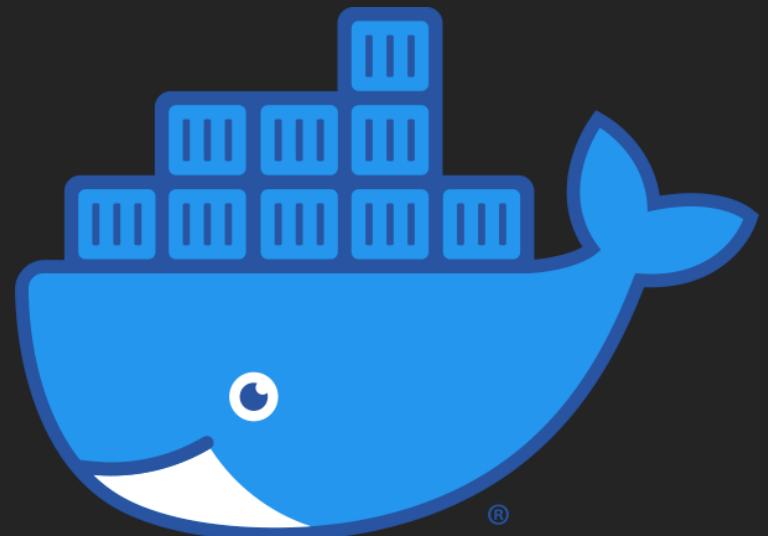
<Infrastructure Engineering>

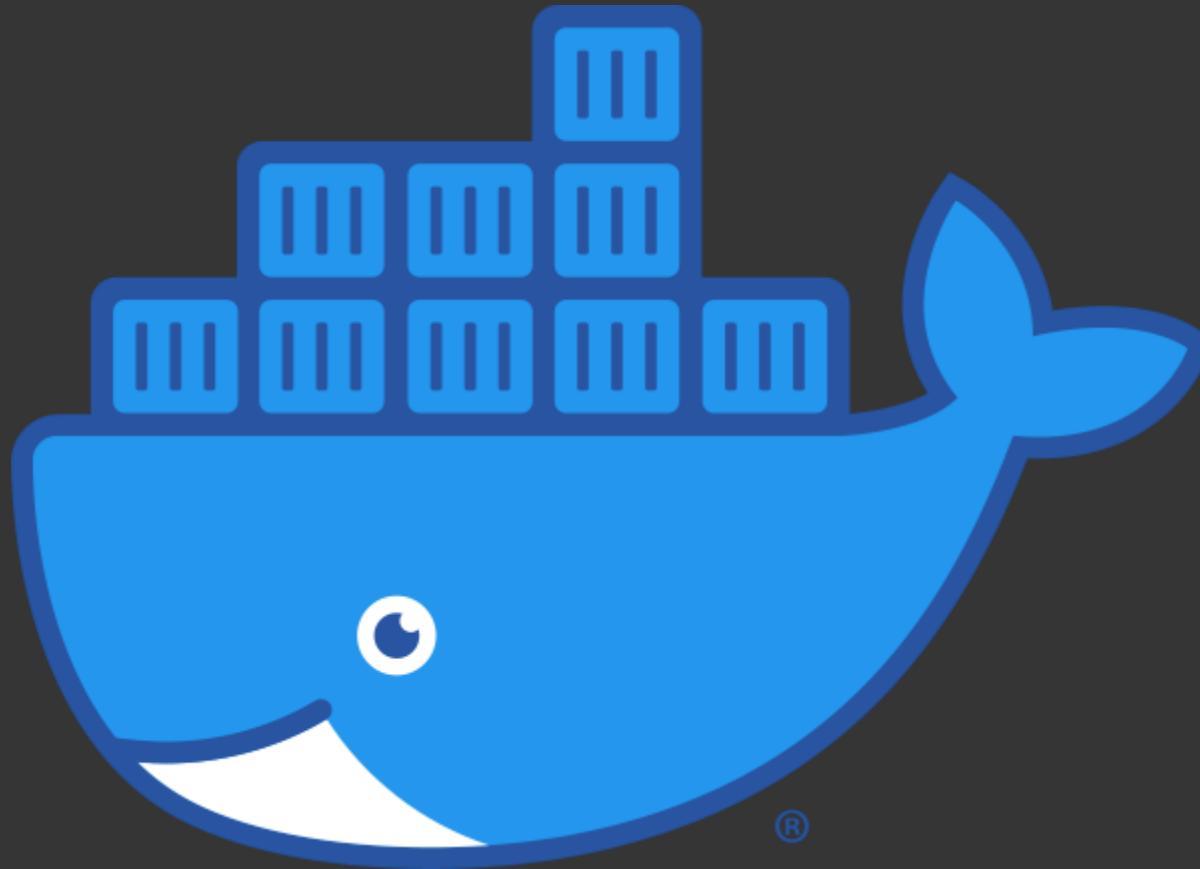
Bootcamp

Chapter 4

ทำความเข้าใจกับ Docker

KBTG
KAMPUS
ClassNest





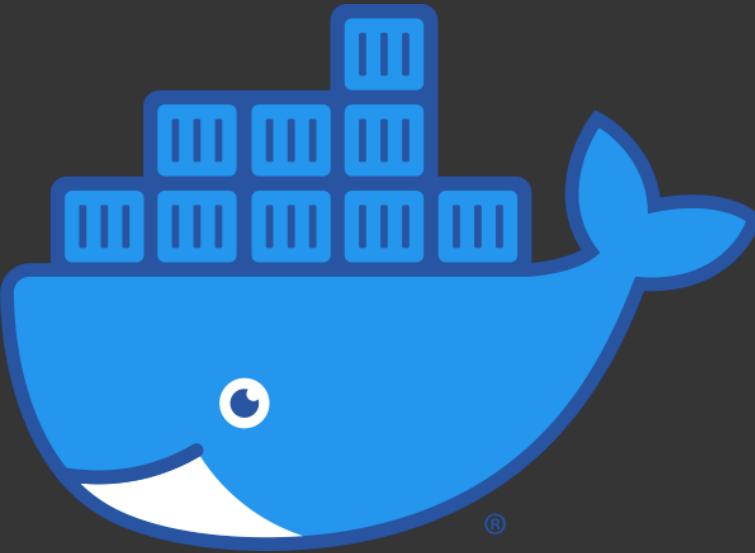
Containers



Container Engine

Operating System

Infrastructure



Docker is an open platform
for developing, shipping,
and running applications.

Docker Client



```
Command Prompt      + | - X
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

C:\Users\siras>docker

Usage: docker [OPTIONS] COMMAND
      A self-sufficient runtime for containers

Common Commands:
  run      Create and run a new container from an image
  exec     Execute a command in a running container
  ps       List containers
  build    Build an image from a Dockerfile
  pull     Download an image from a registry
  push     Upload an image to a registry
  images   List images
  login    Log in to a registry
  logout   Log out from a registry
  search   Search Docker Hub for images
  version  Show the Docker version information
  info     Display system-wide information

Management Commands:
  builder   Manage builds
  buildx*   Docker Buildx (Docker Inc., v0.11.0)
  compose*  Docker Compose (Docker Inc., v2.19.1)
  container Manage containers
  context   Manage contexts
  dev*     Docker Dev Environments (Docker Inc., v0.1.0)
```

Docker Desktop

Docker Desktop Upgrade plan

Q Search for images, containers, volumes, extensions and more... Ctrl+K

Containers Images Volumes Dev Environments BETA Docker Scout EARLY ACCESS Learning Center

Extensions Add Extensions

Container CPU usage 26.66% / 1000% (10 cores allocated)

Container memory usage 723.4MB / 11.1GB

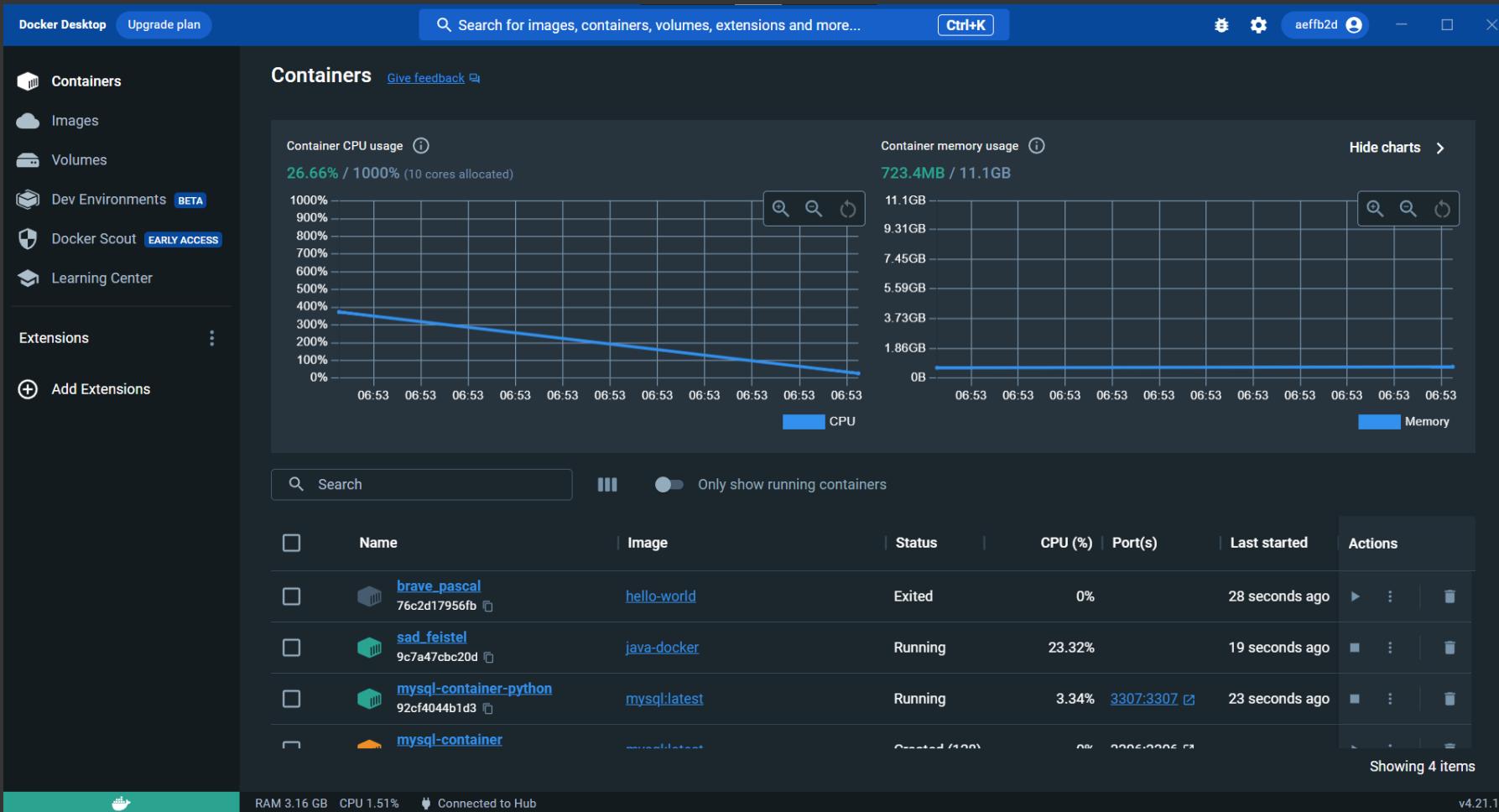
Hide charts

Search Only show running containers

Name	Image	Status	CPU (%)	Port(s)	Last started	Actions		
brave_pascal 76c2d17956fb	hello-world	Exited	0%	28 seconds ago				
sad_feistel 9c7a47cbc20d	java-docker	Running	23.32%	19 seconds ago				
mysql-container-python 92cf4044b1d3	mysql:latest	Running	3.34%	3307:3307	23 seconds ago			
mysql-container	mysql:latest	Created 1 day ago	0%	3306:3306	2 days ago			

Showing 4 items

RAM 3.16 GB CPU 1.51% Connected to Hub v4.21.1



Docker Registry

 docker hub Explore Pricing Sign In Register

Filters

Products

- Images
- Extensions
- Plugins

Trusted Content

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

Operating Systems

- Linux
- Windows

Architectures

- ARM
- ARM 64
- IBM POWER
- IBM Z

1 - 25 of 10,000 available results. Suggested

 alpine DOCKER OFFICIAL IMAGE • 1B+ • 10K+
Updated 2 months ago

A minimal Docker image based on Alpine Linux with a complete package index and only 5 ...

Linux IBM Z riscv64 x86-64 ARM ARM 64 386 PowerPC 64 LE

Pulls: 9,412,366 Last week 
[Learn more](#)

 nginx DOCKER OFFICIAL IMAGE • 1B+ • 10K+
Updated 6 days ago

Official build of Nginx.

Linux ARM 64 386 mips64le PowerPC 64 LE IBM Z x86-64 ARM

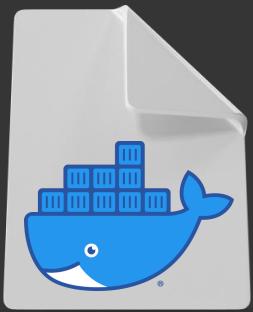
Pulls: 20,000,887 Last week 
[Learn more](#)

 busybox DOCKER OFFICIAL IMAGE • 1B+ • 3.1K
Updated 16 days ago

Busybox base image.

Linux mips64le PowerPC 64 LE riscv64 IBM Z x86-64 ARM ARM 64 386

Docker Objects



**images, containers, networks,
volumes, plugins, and other objects**

Get Docker

Get Docker

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

You can download and install Docker on multiple platforms. Refer to the following section and choose the best installation path for you.



Docker Desktop for Mac

A native application using the macOS sandbox security model which delivers all Docker tools to your Mac.



Docker Desktop for Windows

A native Windows application which delivers all Docker tools to your Windows computer.



Docker Desktop for Linux

A native Linux application which delivers all Docker tools to your Linux computer.

docker version

```
Command Prompt      X + ▾  
C:\Users\siras>docker version  
Client:  
  Cloud integration: v1.0.35  
  Version:          24.0.2  
  API version:     1.43  
  Go version:      go1.20.4  
  Git commit:       cb74dfc  
  Built:            Thu May 25 21:53:15 2023  
  OS/Arch:          windows/amd64  
  Context:          default  
  
Server: Docker Desktop 4.21.1 (114176)  
Engine:  
  Version:          24.0.2  
  API version:     1.43 (minimum version 1.12)  
  Go version:      go1.20.4  
  Git commit:       659604f  
  Built:            Thu May 25 21:52:17 2023  
  OS/Arch:          linux/amd64  
  Experimental:    false  
containerd:  
  Version:          1.6.21  
  GitCommit:        3dce8eb055cbb6872793272b4f20ed16117344f8  
runc:
```

docker info

```
PS C:\Users\siras>docker info
Client:
  Version: 24.0.2
  Context: default
  Debug Mode: false
  Plugins:
    buildx: Docker Buildx (Docker Inc.)
      Version: v0.11.0
      Path: C:\Program Files\Docker\cli-plugins\docker-buildx.exe
    compose: Docker Compose (Docker Inc.)
      Version: v2.19.1
      Path: C:\Program Files\Docker\cli-plugins\docker-compose.exe
    dev: Docker Dev Environments (Docker Inc.)
      Version: v0.1.0
      Path: C:\Program Files\Docker\cli-plugins\docker-dev.exe
    extension: Manages Docker extensions (Docker Inc.)
      Version: v0.2.20
      Path: C:\Program Files\Docker\cli-plugins\docker-extension.exe
    init: Creates Docker-related starter files for your project (Docker Inc.)
      Version: v0.1.0-beta.6
      Path: C:\Program Files\Docker\cli-plugins\docker-init.exe
    sbom: View the packaged-based Software Bill Of Materials (SBOM) for an image (Anchore Inc.)
      Version: 0.6.0
      Path: C:\Program Files\Docker\cli-plugins\docker-sbom.exe
    scan: Docker Scan (Docker Inc.)
      Version: v0.26.0
      Path: C:\Program Files\Docker\cli-plugins\docker-scan.exe
    scout: Command line tool for Docker Scout (Docker Inc.)
      Version: 0.16.1
      Path: C:\Program Files\Docker\cli-plugins\docker-scout.exe
```

docker run hello-world

```
Command Prompt x + ▾  
C:\Users\siras>docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
719385e32844: Pull complete  
Digest: sha256:926fac19d22aa2d60f1a276b66a20eb765fbeea2db5dbdaafeb456ad8ce81598  
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/
```

The screenshot shows the Docker Hub interface for the 'hello-world' image. At the top, there's a blue header bar with the Docker Hub logo and a search bar containing 'hello-world'. Below the header, a breadcrumb navigation shows 'Explore > Official Images > hello-world'. The main content area features a large image of a terminal window displaying '>hello world'. To the right of the image, the repository name 'hello-world' is displayed in bold, with a badge indicating it's a 'DOCKER OFFICIAL IMAGE'. It also shows download statistics ('1B+') and a star rating ('2.1K'). A brief description follows: 'Hello World! (an example of minimal Dockerization)'. Below this, there are two tabs: 'Overview' (which is underlined in blue, indicating it's the active view) and 'Tags'. A large section titled 'Quick reference' contains two bullet points: 'Maintained by: the Docker Community' and 'Where to get help: the Docker Community Slack, Server Fault, Unix & Linux, or Stack Overflow'. At the bottom, a callout box highlights 'Supported tags and respective Dockerfile links'.

hello-world

DOCKER OFFICIAL IMAGE • 1B+ • 2.1K

Hello World! (an example of minimal Dockerization)

Overview Tags

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

Docker Desktop Upgrade plan

Search for images, containers, volumes, extensions and more... Ctrl+K

Containers Give feedback

Container CPU usage 4.58% / 1000% (10 cores allocated)

Container memory usage 397MB / 11.1GB

Hide charts

Container CPU usage chart (06-30 to 07-01)

Container memory usage chart (06-30 to 07-01)

Extensions

Add Extensions

Search Only show running containers

Name	Image	Status	CPU (%)	Port(s)	Last started	Actions	
brave_pascal 76c2d17956fb	hello-world	Exited	0%	15 minutes ago			
sad_feistel 9c7a47cbc20d	java-docker	Exited (1)	0%	15 minutes ago			
mysql-container-python 92cf4044b1d3	mysql:latest	Running	4.58% 3307:3307	15 minutes ago			
mysql-container	mysql:latest	Created (1m)	0%	2025-07-01 15:45:56.573			

Showing 4 items

RAM 2.97 GB CPU 0.96% Connected to Hub v4.21.1

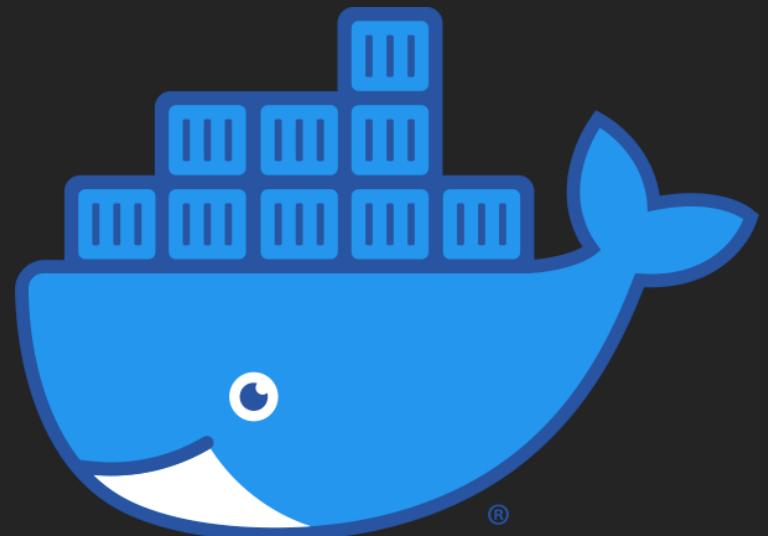
<Infrastructure Engineering>

Bootcamp

Chapter 5

รู้จัก และ ใช้งาน Hub

KBTG
KAMPUS
ClassNest



 docker hub Explore Repositories Organizations Help ▾ Upgrade  aeffb2d ▾

Filters 1 - 25 of 10,000 available results. Suggested

Products

- Images
- Extensions
- Plugins

Trusted Content

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

Operating Systems

- Linux
- Windows

Architectures

- ARM
- ARM 64
- IBM POWER

alpine  DOCKER OFFICIAL IMAGE · 1B+ · 10K+ Updated 2 months ago A minimal Docker image based on Alpine Linux with a complete package index and only 5 ... Pulls: 9,412,366 Last week Learn more ⓘ

nginx  DOCKER OFFICIAL IMAGE · 1B+ · 10K+ Updated 8 days ago Official build of Nginx. Pulls: 20,000,887 Last week Learn more ⓘ

busybox  DOCKER OFFICIAL IMAGE · 1B+ · 3.1K Updated 17 days ago Busybox base image. Pulls: 10,337,819 Last week Learn more ⓘ

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



—

—

Demo

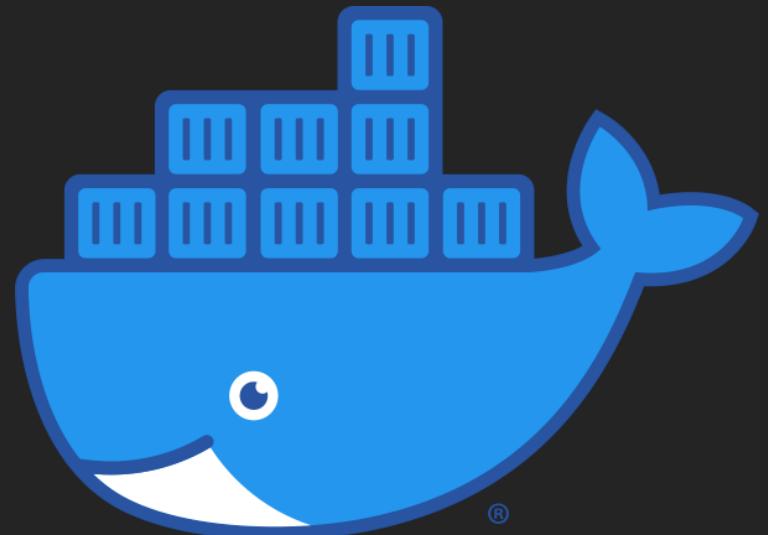
<Infrastructure Engineering>

Bootcamp

Chapter 6

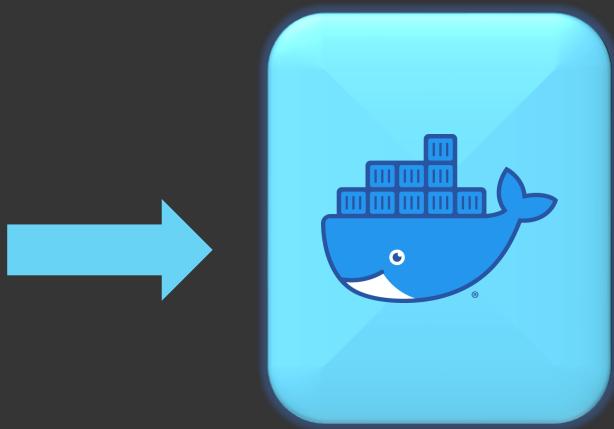
ความหมาย และ การใช้งาน Docker Image

KBTG
KAMPUS
ClassNest





Dockerfile



Docker Image



Docker Container



Docker Image

**Docker image is a file
used to execute code in
a Docker container**



Docker Image



Application code



Libraries



Tools



Dependencies

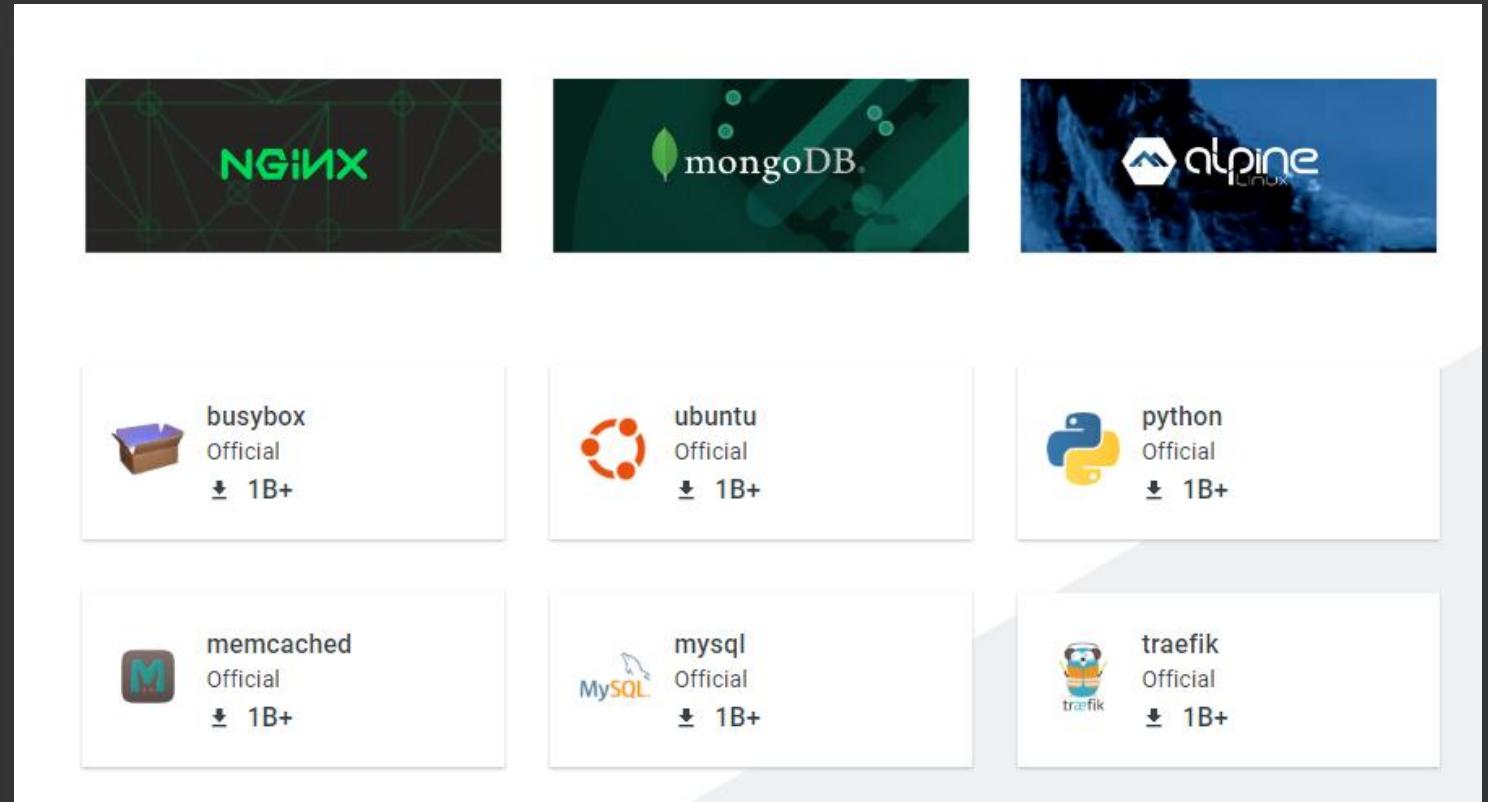


Configuration



Other Files

Docker Official Images



Anatomy of a Docker image



Exposed Ports (e.g., 80)

Entrypoint and CMD (Application)

Metadata (Labels)

Environment Variables and Config

Runtime and Libraries (e.g., Node)

Filesystem Layers (Read-only)

Filesystem Layers (Read-only)

Base Image (e.g., Alpine Linux)

ຕໍ່າວິ່ງການ pull image ມາ

Explore > Official Images > [ubuntu](#)



ubuntu DOCKER OFFICIAL IMAGE • 1B+ • 10K+
Ubuntu is a Debian-based Linux operating system based on free software.

[docker pull ubuntu](#) 📄

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

Quick reference

- Maintained by:
[Canonical](#)
- Where to get help:
[the Docker Community Slack](#), [Server Fault](#), [Unix & Linux](#), or [Stack Overflow](#)

Recent Tags

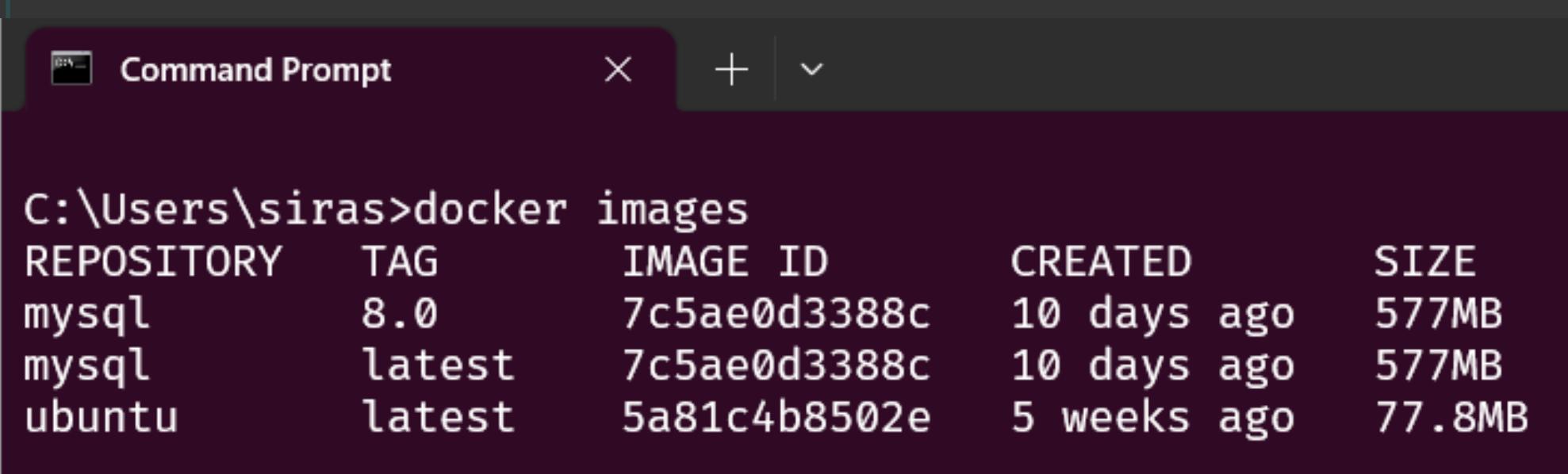
[rolling](#) [mantic-20230801](#) [mantic](#) [lunar-20230731](#)
[lunar](#) [focal-20230801](#) [focal](#) [devel](#) [23.10](#) [23.04](#)

About Official Images

Pulling Docker images

```
Command Prompt × + ▾  
C:\Users\siras>docker pull ubuntu  
Using default tag: latest  
latest: Pulling from library/ubuntu  
3153aa388d02: Already exists  
Digest: sha256:0bc...  
Status: Downloaded newer image for ubuntu:latest  
docker.io/library/ubuntu:latest  
  
What's Next?  
View summary of image vulnerabilities and recommendations → docker scout quickview ubuntu
```

Listing downloaded images



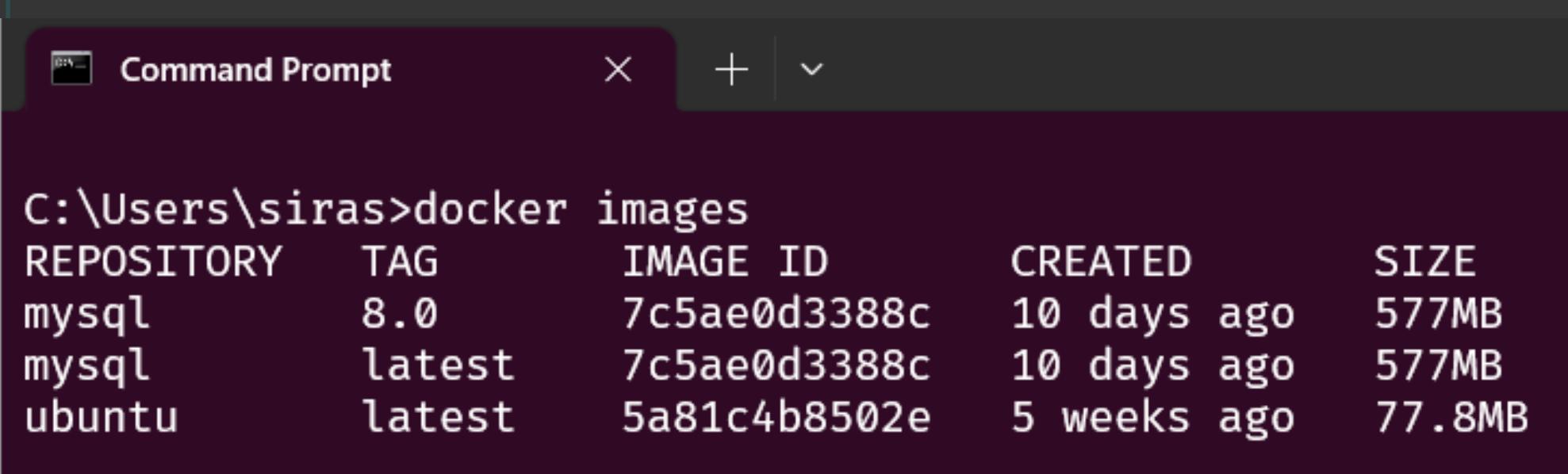
A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the output of the command "docker images". The output lists three Docker images:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mysql	8.0	7c5ae0d3388c	10 days ago	577MB
mysql	latest	7c5ae0d3388c	10 days ago	577MB
ubuntu	latest	5a81c4b8502e	5 weeks ago	77.8MB

List all containers (including stopped ones)

```
Command Prompt      X + ▾
C:\Users\siras>docker ps -a
CONTAINER ID        IMAGE          COMMAND       CREATED        STATUS         PORTS          NAMES
C:\Users\siras>docker container ls -a
CONTAINER ID        IMAGE          COMMAND       CREATED        STATUS         PORTS          NAMES
C:\Users\siras>|
```

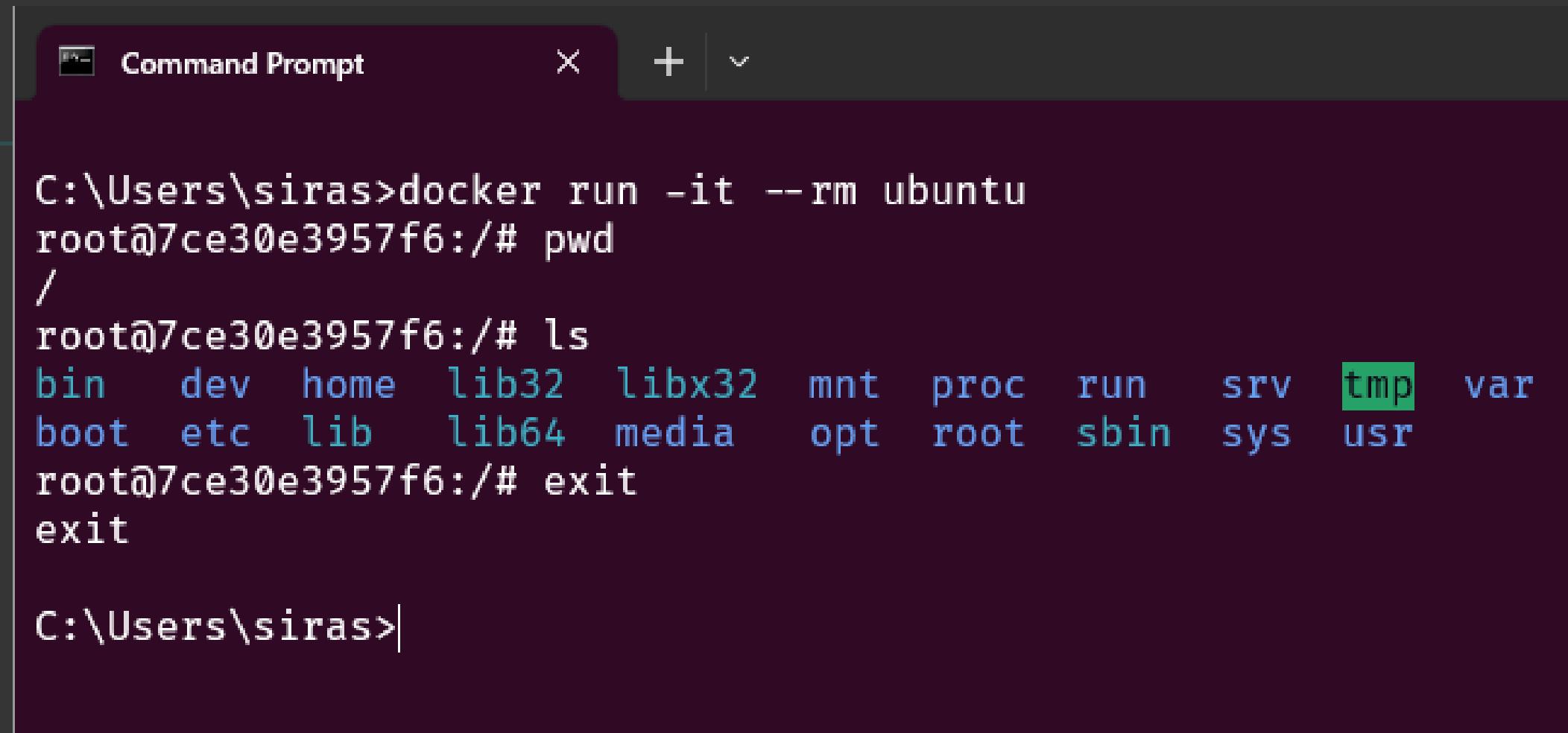
Listing downloaded images



A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the output of the command "docker images". The output lists three Docker images:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mysql	8.0	7c5ae0d3388c	10 days ago	577MB
mysql	latest	7c5ae0d3388c	10 days ago	577MB
ubuntu	latest	5a81c4b8502e	5 weeks ago	77.8MB

How to run image



A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the following terminal session:

```
C:\Users\siras>docker run -it --rm ubuntu
root@7ce30e3957f6:/# pwd
/
root@7ce30e3957f6:/# ls
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr
root@7ce30e3957f6:/# exit
exit

C:\Users\siras>
```

The "tmp" directory is highlighted in green.

How to run image

The screenshot shows the Docker Desktop application interface. The left sidebar includes links for Containers, Images, Volumes, Dev Environments (BETA), Docker Scout (EARLY ACCESS), and Learning Center. Below these are sections for Extensions and a button to Add Extensions. The main area is titled 'Containers' and displays four charts: 'Container CPU usage' (No containers are running), 'Container memory usage' (No containers are running), 'Hide charts', and two other charts labeled 'No containers are running'. At the bottom, there is a search bar, a filter for 'Only show running containers' (which is off), and a table of containers. The table has columns for Actions, Last started, Port(s), CPU (%), Status, Image, and Name. One container is listed: 'boring_hodgki' (ubuntu) with ID '5b44d65eb36f', status 'Exited', and last started 7 minutes ago.

Actions	Last started	Port(s)	CPU (%)	Status	Image	Name	Actions
	7 minutes ago	N/A		Exited	ubuntu	boring_hodgki	

RAM 2.76 GB Connected to Hub v4.21.1

Remove image

```
Command Prompt × + ▾  
C:\Users\siras>docker images  
REPOSITORY TAG IMAGE ID CREATED SIZE  
mysql 8.0 7c5ae0d3388c 10 days ago 577MB  
mysql latest 7c5ae0d3388c 10 days ago 577MB  
ubuntu latest 5a81c4b8502e 5 weeks ago 77.8MB  
  
C:\Users\siras>docker rmi -f 5a81c4b8502e  
Untagged: ubuntu:latest  
Untagged: ubuntu@sha256:0bc...  
Deleted: sha256:5a81c4b8502e4979e75bd8f91343b95b0d695ab67f241dbed0d1530a35bde1eb
```

Demo

<Infrastructure Engineering>

Bootcamp

Chapter 7

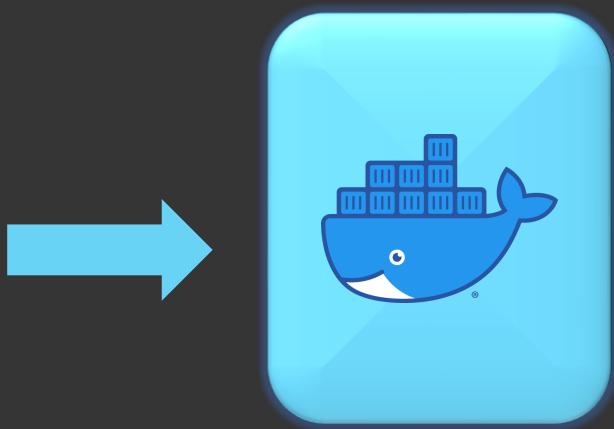
การใช้ DockerFile

KBTG
KAMPUS
ClassNest





Dockerfile

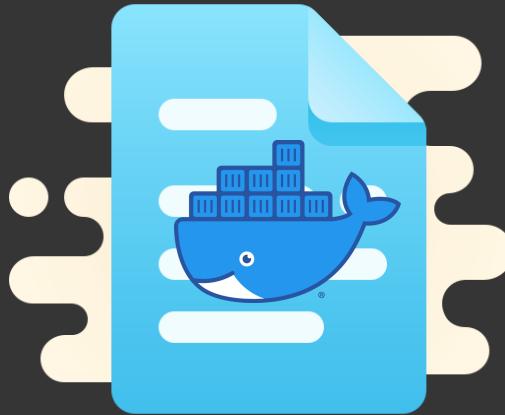


Docker Image



Docker Container

What is a Dockerfile?



Dockerfile is a simple text file that consists of instructions to build Docker images.



Dockerfile



```
FROM eclipse-temurin:17-jdk-jammy

WORKDIR /app

COPY .mvn/ .mvn
COPY mvnw pom.xml .
RUN ./mvnw dependency:resolve

COPY src ./src

CMD ["./mvnw", "spring-boot:run"]
```



Dockerfile

```
FROM eclipse-temurin:17-jdk-jammy

WORKDIR /app

COPY .mvn/ .mvn
COPY mvnw pom.xml ./
RUN ./mvnw dependency:resolve

COPY src ./src

CMD ["./mvnw", "spring-boot:run"]
```



eclipse-temurin

DOCKER OFFICIAL IMAGE

50M+

389

Official Images for OpenJDK binaries built by Eclipse Temurin.

Overview

Tags

Quick reference

- Maintained by:
[Adoptium](#)
- Where to get help:
[Adoptium Slack](#); [Adoptium Support](#)

Supported tags and respective Dockerfile links

Note: the description for this image is longer than the Hub length limit of 25000, so the "Supported tags" list has been trimmed to compensate. See also [docker/hub-feedback#238](#) and [docker/roadmap#475](#).

- See "Supported tags and respective Dockerfile links" at <https://github.com/docker-library/docs/tree/master/eclipse-temurin/README.md>

Quick reference (cont.)

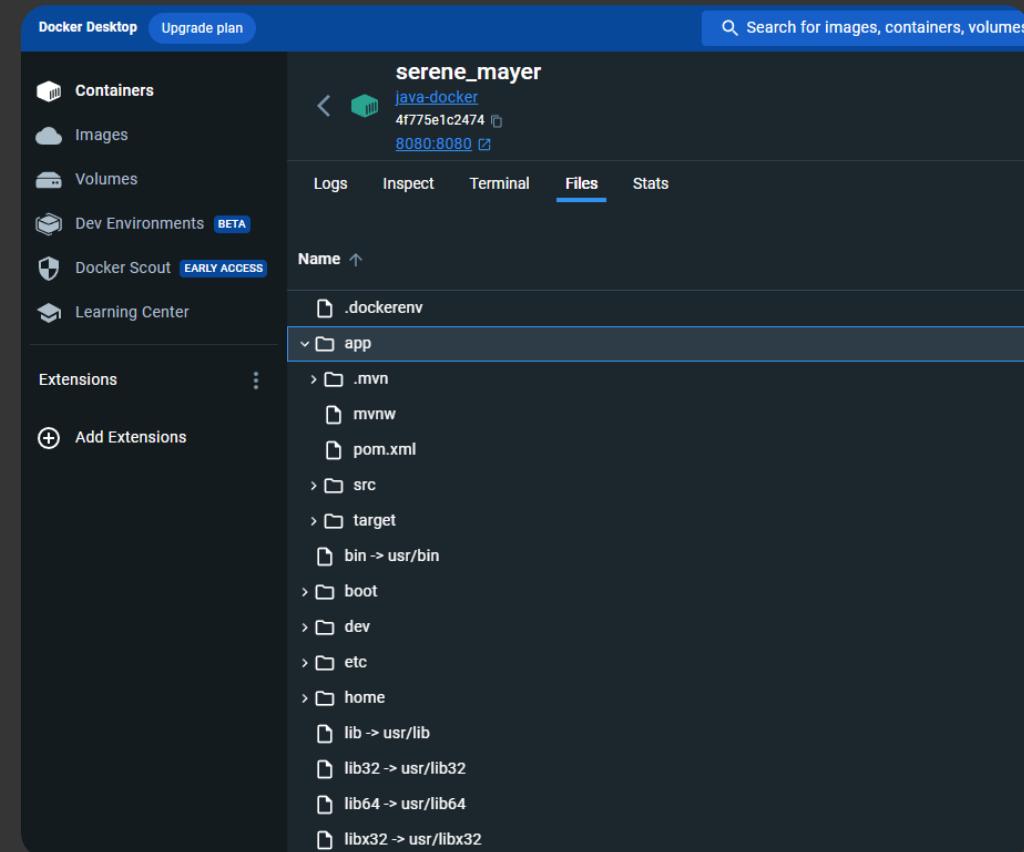
```
FROM eclipse-temurin:17-jdk-jammy

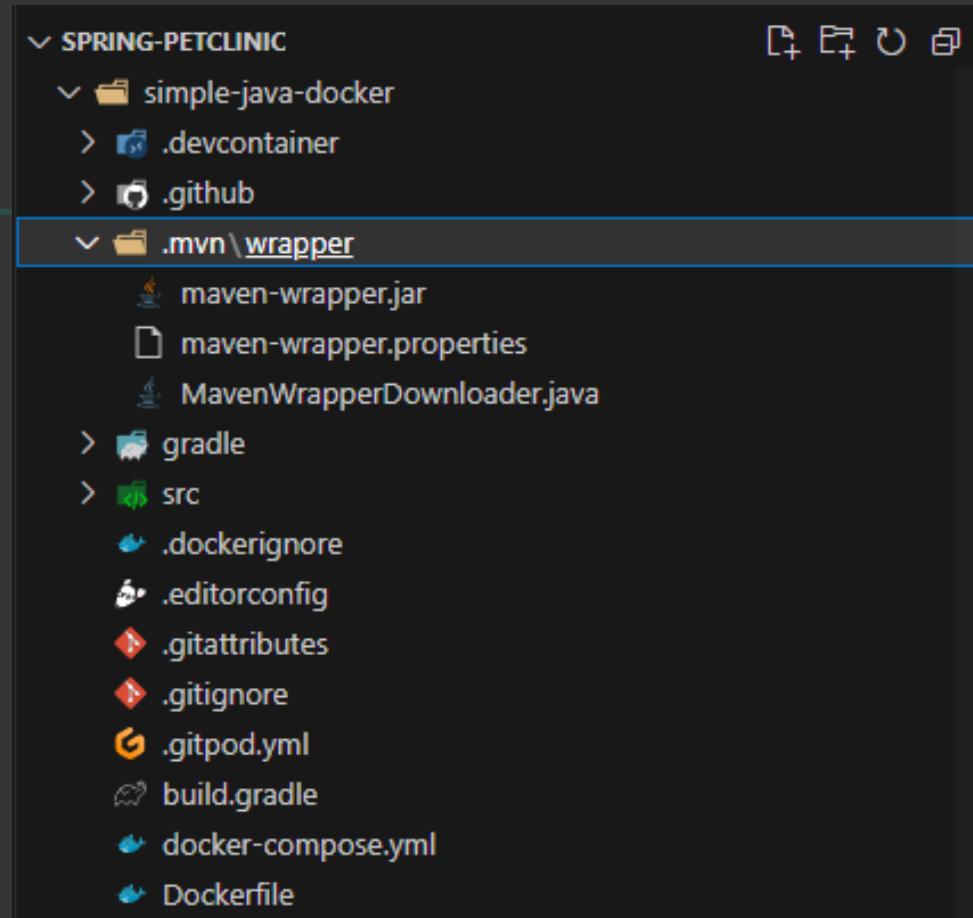
WORKDIR /app

COPY .mvn/.mvn
COPY mvnw pom.xml ./
RUN ./mvnw dependency:resolve

COPY src ./src

CMD ["./mvnw", "spring-boot:run"]
```





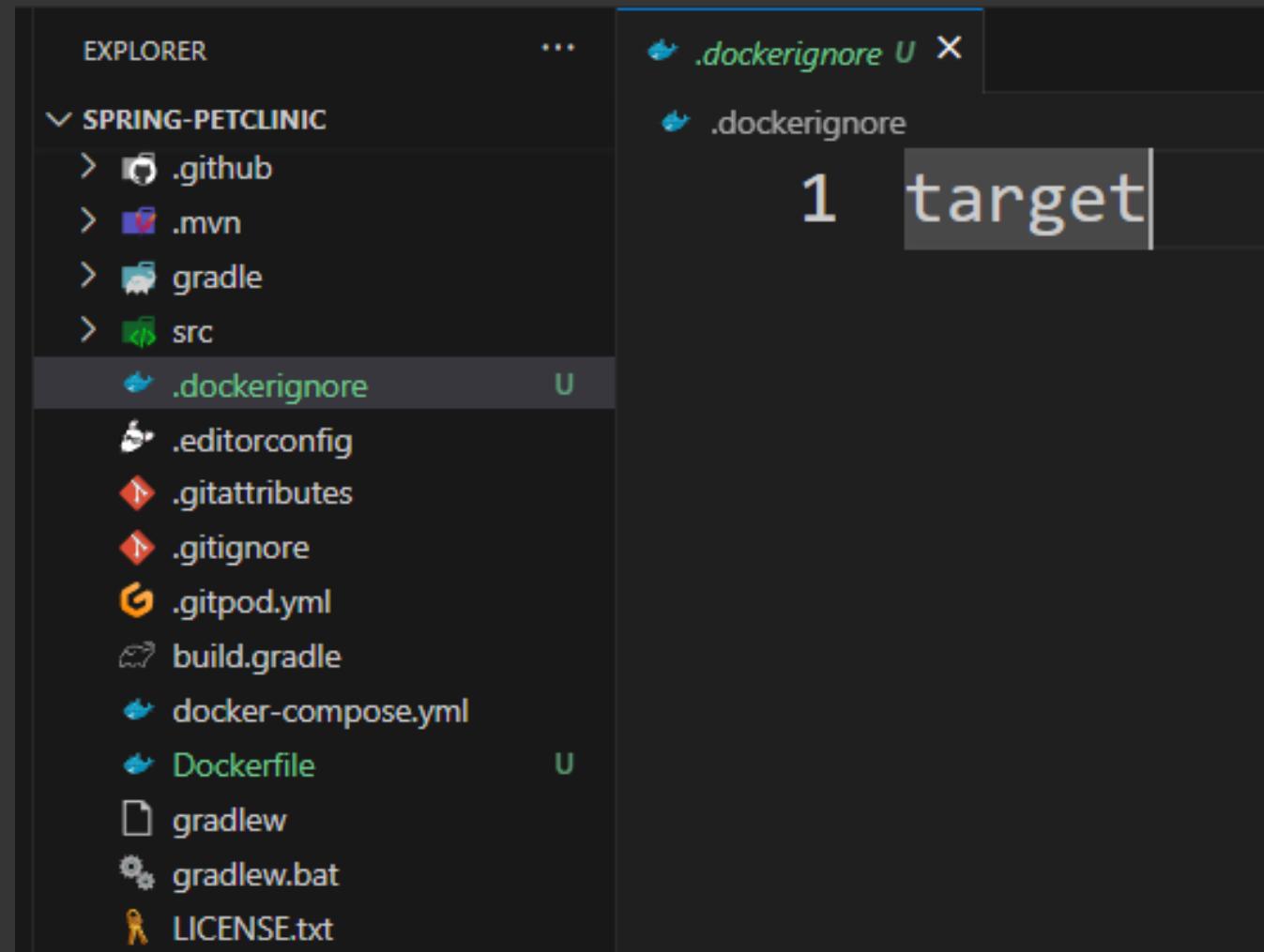
The screenshot shows the Docker Container Explorer with the following details:

Container: serene_mayer
Image: java-docker
ID: 4f775e1c2474
Port: 8080:8080

The **Files** tab is selected, showing the contents of the container's file system:

Name	Note
.dockerenv	
app	MODIFIED
.mvn	
wrapper	
maven-wrapper.jar	
maven-wrapper.properties	
MavenWrapperDownloader.java	

.dockerignore



<Infrastructure Engineering>

Bootcamp

Chapter 8

การใช้ Instruction Command



A screenshot of a Mac OS X terminal window titled "Dockerfile". The window has the standard red, yellow, and green close buttons. The content of the terminal is a Dockerfile:

```
FROM eclipse-temurin:17-jdk-jammy

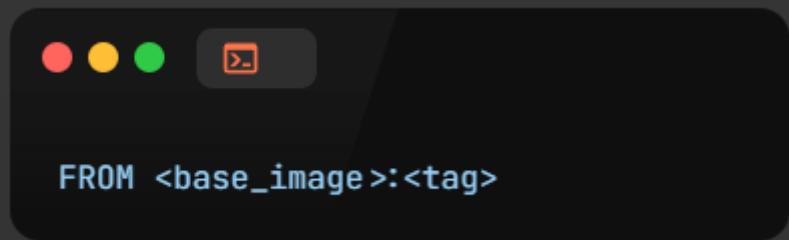
WORKDIR /app

COPY .mvn/ .mvn
COPY mvnw pom.xml ./
RUN ./mvnw dependency:resolve

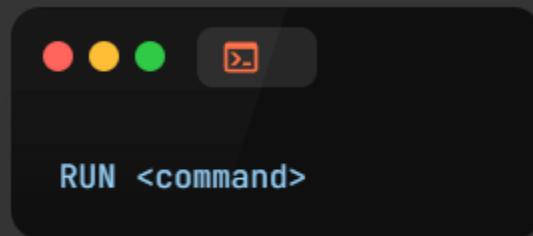
COPY src ./src

CMD ["./mvnw", "spring-boot:run"]
```

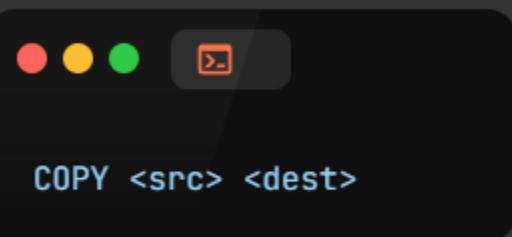
FROM



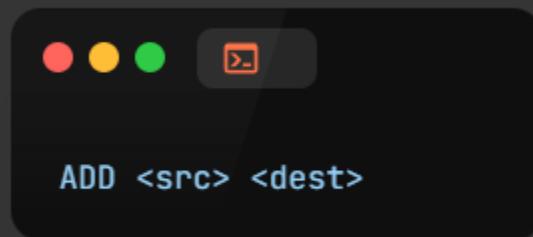
RUN



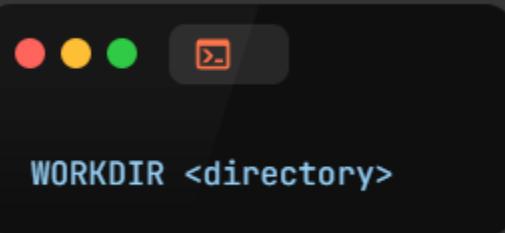
COPY



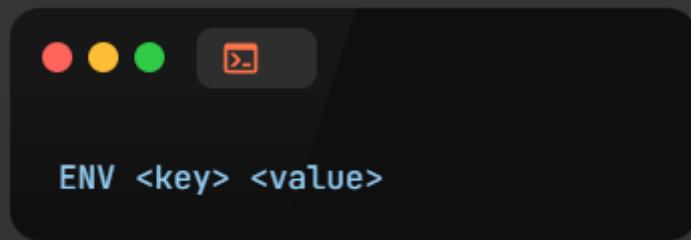
ADD



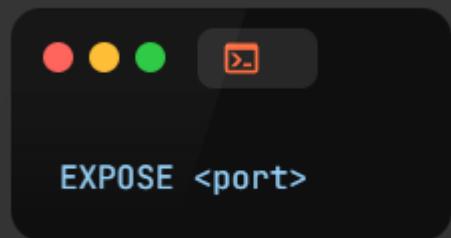
WORKDIR



ENV



EXPOSE



CMD



ENTRYPOINT



```
ENTRYPOINT ["executable","param1","param2"]
```

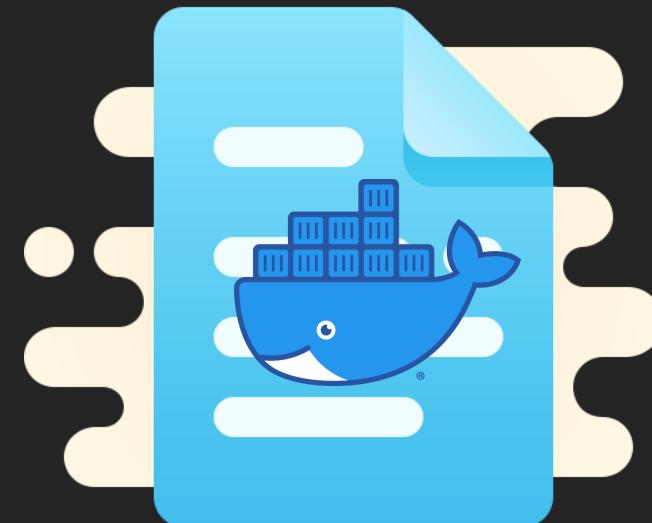
<Infrastructure Engineering>

Bootcamp

Chapter 9

การสร้าง Docker File

KBTG
KAMPUS
ClassNest



Example Spring Boot Project



<https://github.com/spring-projects/spring-petclinic>

readme.md

Spring PetClinic Sample Application

Java CI with Maven passing

[Open in Gitpod](#) [Open in GitHub Codespaces](#)

Understanding the Spring Petclinic application with a few diagrams

[See the presentation here](#)

Running petclinic locally

Petclinic is a Spring Boot application built using [Maven](#) or [Gradle](#). You can build a jar file and run it from the command line (it should work just as well with Java 17 or newer):

```
git clone https://github.com/spring-projects/spring-petclinic.git
cd spring-petclinic
./mvnw package
java -jar target/*.jar
```

You can then access petclinic at <http://localhost:8080/>

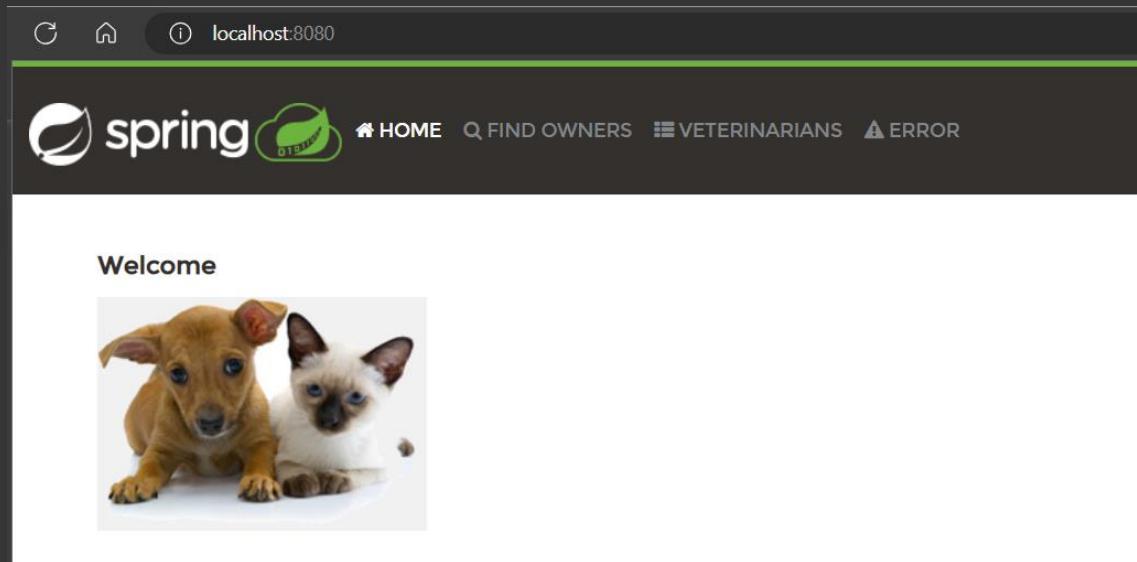
The screenshot shows the Spring PetClinic application running locally. The top navigation bar includes links for HOME, FIND OWNERS (which is highlighted in green), VETERINARIANS, and ERROR. The main content area is titled "Owners" and displays a table with the following data:

Name	Address	City	Telephone	Pets
Jeff Black	1450 Oak Blvd.	Monona	6085555387	Lucky
Jean Coleman	105 N. Lake St.	Monona	6085552654	Max Samantha
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085551749	Basil
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Maria Escobito	345 Maple St.	Madison	6085557683	Mulligan
Carlos Estaban	2335 Independence La.	Waunakee	6085555487	Lucky Sly
George Franklin	110 W. Liberty St.	Madison	6085551023	Leo
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	Jewel Rosy
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy

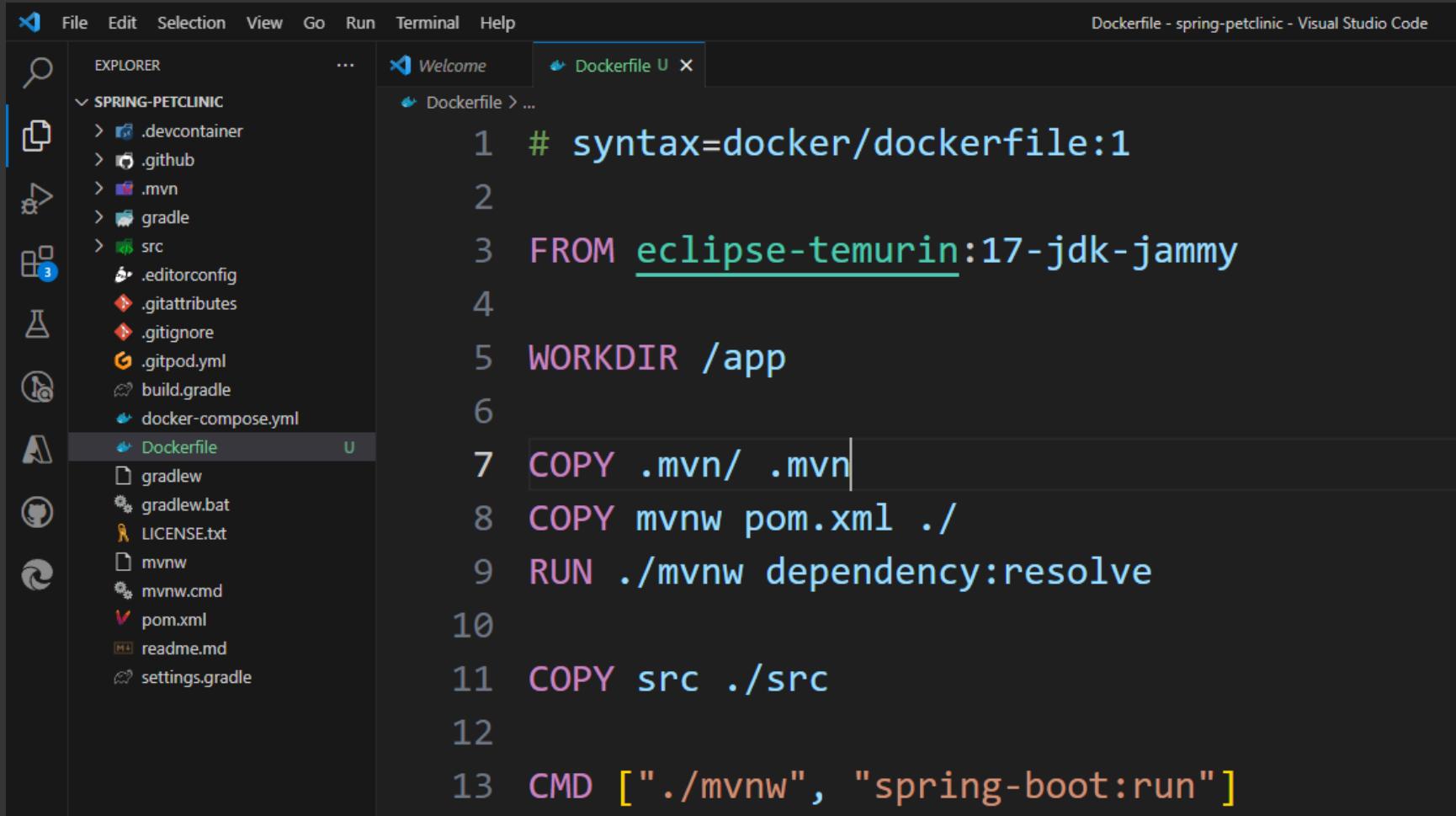
[spring](#) by Pivotal

Clone simple application

```
git clone https://github.com/spring-projects/spring-petclinic.git  
cd spring-petclinic
```



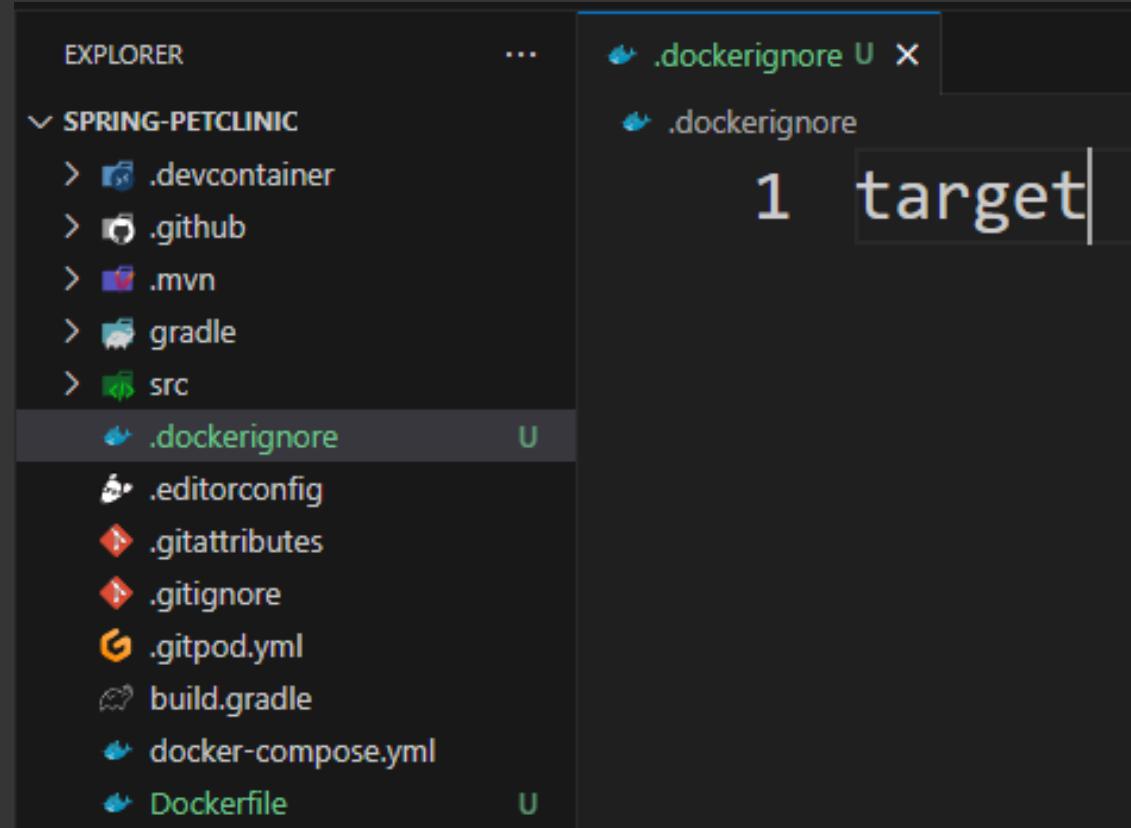
Write a Dockerfile



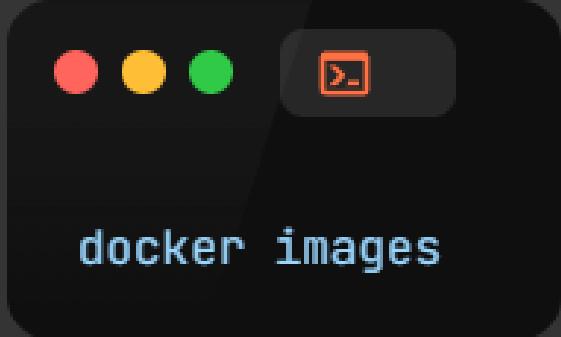
The screenshot shows a Visual Studio Code interface with a dark theme. The left sidebar has icons for Explorer, Search, Open, Find, Git, Projects, Task, Terminal, and Help. The Explorer view shows a project named "SPRING-PETCLINIC" containing files like ".devcontainer", ".github", ".mvn", "gradle", "src", ".editorconfig", ".gitattributes", ".gitignore", ".gitpod.yml", "build.gradle", "docker-compose.yml", and "Dockerfile". The "Dockerfile" tab is selected. The main editor area displays a Dockerfile:

```
1 # syntax=docker/dockerfile:1
2
3 FROM eclipse-temurin:17-jdk-jammy
4
5 WORKDIR /app
6
7 COPY .mvn/.mvn|
8 COPY mvnw pom.xml .
9 RUN ./mvnw dependency:resolve
10
11 COPY src ./src
12
13 CMD ["./mvnw", "spring-boot:run"]
```

.dockerignore file



View local images



```
Command Prompt
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

C:\Users\siras>docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
java-docker     latest       ec04d0dbec6a   3 days ago   584MB

C:\Users\siras>
```

A screenshot of a Windows Command Prompt window. The title bar says 'Command Prompt'. The window shows the output of the 'docker images' command. It lists one image: 'java-docker' with the 'latest' tag, having an image ID of 'ec04d0dbec6a', created 3 days ago, and a size of 584MB. The prompt 'C:\Users\siras>' is at the bottom.

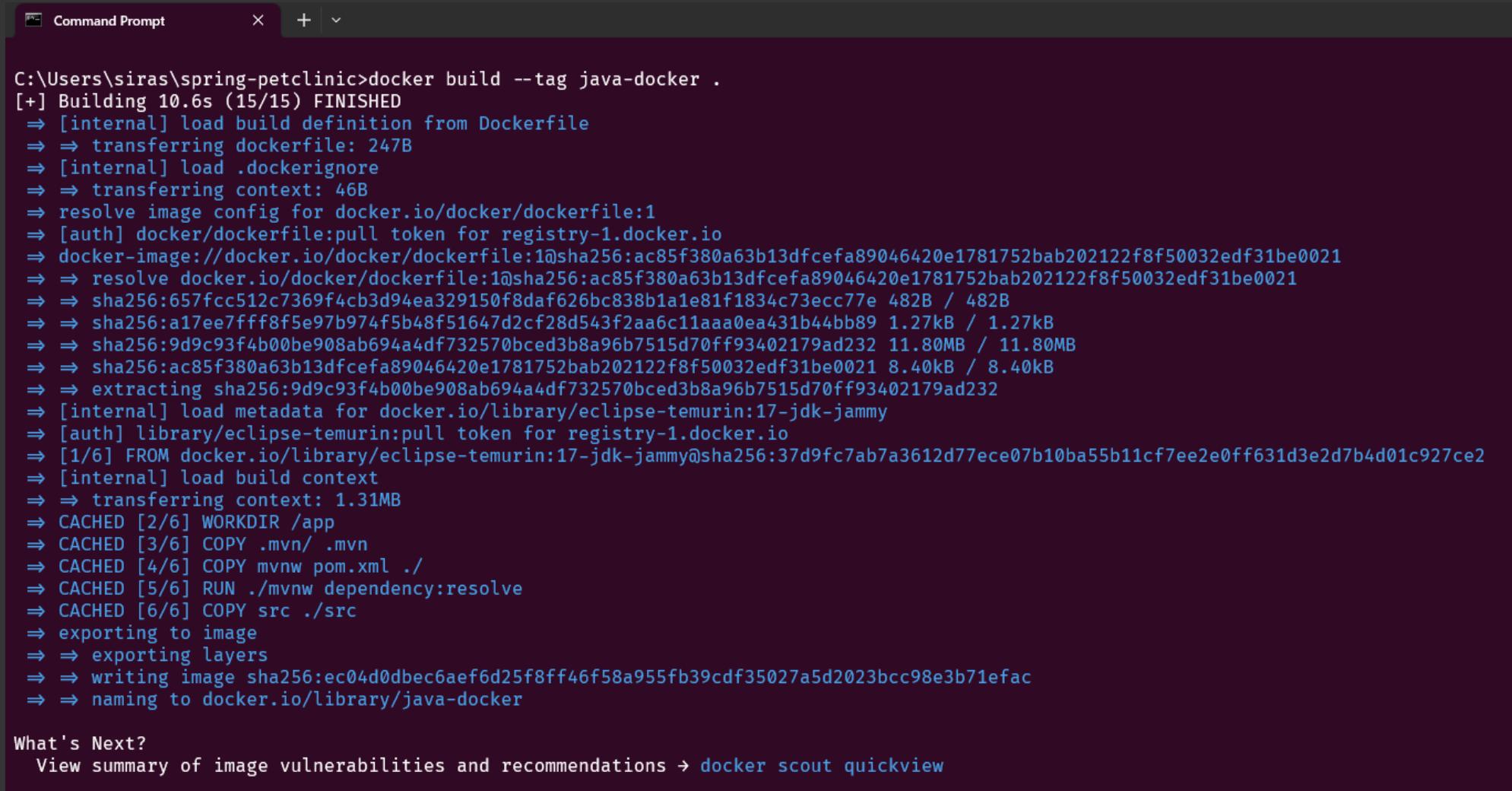
Build an image from a Dockerfile

```
docker build [OPTIONS] PATH | URL | -
```

Example

```
docker build --tag java-docker .
```

Build an image



```
C:\Users\siras\spring-petclinic>docker build --tag java-docker .
[+] Building 10.6s (15/15) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 247B
=> [internal] load .dockerignore
=> => transferring context: 46B
=> resolve image config for docker.io/docker/dockerfile:1
=> [auth] docker/dockerfile:pull token for registry-1.docker.io
=> docker-image://docker.io/docker/dockerfile:1@sha256:ac85f380a63b13dfcefa89046420e1781752bab202122f8f50032edf31be0021
=> => resolve docker.io/docker/dockerfile:1@sha256:ac85f380a63b13dfcefa89046420e1781752bab202122f8f50032edf31be0021
=> => sha256:657fcc512c7369f4cb3d94ea329150f8daf626bc838b1a1e81f1834c73ecc77e 482B / 482B
=> => sha256:a17ee7fff8f5e97b974f5b48f51647d2cf28d543f2aa6c11aaa0ea431b44bb89 1.27kB / 1.27kB
=> => sha256:9d9c93f4b00be908ab694a4df732570bcd3b8a96b7515d70ff93402179ad232 11.80MB / 11.80MB
=> => sha256:ac85f380a63b13dfcefa89046420e1781752bab202122f8f50032edf31be0021 8.40kB / 8.40kB
=> => extracting sha256:9d9c93f4b00be908ab694a4df732570bcd3b8a96b7515d70ff93402179ad232
=> [internal] load metadata for docker.io/library/eclipse-temurin:17-jdk-jammy
=> [auth] library/eclipse-temurin:pull token for registry-1.docker.io
=> [1/6] FROM docker.io/library/eclipse-temurin:17-jdk-jammy@sha256:37d9fc7ab7a3612d77ece07b10ba55b11cf7ee2e0ff631d3e2d7b4d01c927ce2
=> [internal] load build context
=> => transferring context: 1.31MB
=> CACHED [2/6] WORKDIR /app
=> CACHED [3/6] COPY .mvn/ .mvn
=> CACHED [4/6] COPY mvnw pom.xml ./
=> CACHED [5/6] RUN ./mvnw dependency:resolve
=> CACHED [6/6] COPY src ./src
=> exporting to image
=> => exporting layers
=> => writing image sha256:ec04d0dbec6aef6d25f8ff46f58a955fb39cdf35027a5d2023bcc98e3b71efac
=> => naming to docker.io/library/java-docker
```

What's Next?

View summary of image vulnerabilities and recommendations → [docker scout quickview](#)

Run Container

```
Command Prompt - docker n + × - □ ×

:: Built with Spring Boot :: 3.1.1

2023-08-04T06:54:35.191Z INFO 238 — [           main] o.s.s.petclinic.PetClinicApplication : Starting PetClinicApplication using Java 17.0.8 with PID 238 (/app/target/classes starte
d by root in /app)
2023-08-04T06:54:35.196Z INFO 238 — [           main] o.s.s.petclinic.PetClinicApplication : No active profile set, falling back to 1 default profile: "default"
2023-08-04T06:54:36.272Z INFO 238 — [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2023-08-04T06:54:36.338Z INFO 238 — [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 54 ms. Found 2 JPA repository interfaces.
2023-08-04T06:54:36.971Z INFO 238 — [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2023-08-04T06:54:36.981Z INFO 238 — [           main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2023-08-04T06:54:36.982Z INFO 238 — [           main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.10]
2023-08-04T06:54:37.090Z INFO 238 — [           main] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2023-08-04T06:54:37.093Z INFO 238 — [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1803 ms
2023-08-04T06:54:37.272Z INFO 238 — [           main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting ...
2023-08-04T06:54:37.522Z INFO 238 — [           main] com.zaxxer.hikari.pool.HikariPool : HikariPool-1 - Added connection conn0: url=jdbc:h2:mem:189e86fd-73ca-40d1-b26f-2aad59d5b
3fc user=SA
2023-08-04T06:54:37.523Z INFO 238 — [           main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2023-08-04T06:54:37.725Z INFO 238 — [           main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2023-08-04T06:54:37.848Z INFO 238 — [           main] org.hibernate.Version : HHH000412: Hibernate ORM core version 6.2.5.Final
2023-08-04T06:54:37.852Z INFO 238 — [           main] org.hibernate.cfg.Environment : HHH000406: Using bytecode reflection optimizer
2023-08-04T06:54:37.989Z INFO 238 — [           main] o.h.b.i.BytecodeProviderInitiator : HHH000021: Bytecode provider name : bytebuddy
2023-08-04T06:54:38.178Z INFO 238 — [           main] o.s.o.j.p.SpringPersistenceUnitInfo : No LoadTimeWeaver setup: ignoring JPA class transformer
2023-08-04T06:54:38.519Z INFO 238 — [           main] o.h.b.i.BytecodeProviderInitiator : HHH000021: Bytecode provider name : bytebuddy
2023-08-04T06:54:39.397Z INFO 238 — [           main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2023-08-04T06:54:39.401Z INFO 238 — [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2023-08-04T06:54:39.666Z INFO 238 — [           main] o.s.d.j.r.query.QueryEnhancerFactory : Hibernate is in classpath; If applicable, HQL parser will be used.
2023-08-04T06:54:40.753Z INFO 238 — [           main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 13 endpoint(s) beneath base path '/actuator'
2023-08-04T06:54:40.844Z INFO 238 — [           main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2023-08-04T06:54:40.859Z INFO 238 — [           main] o.s.s.petclinic.PetClinicApplication : Started PetClinicApplication in 6.103 seconds (process running for 6.447)
```



localhost:8080

[HOME](#)[FIND OWNERS](#)[VETERINARIANS](#)[ERROR](#)

Welcome



<Infrastructure Engineering>

Bootcamp

Chapter 10

การทำงานร่วมกับ Container

KBTG
KAMPUS
ClassNest



Docker Hub

Screenshot of the Docker Hub user interface showing a repository page for user 'aefffb2d'.

The top navigation bar includes:

- Docker Hub logo
- Search bar: Search Docker Hub
- Navigation links: Explore, Repositories, Organizations, Help
- User profile: Upgrade, Fingerprint icon, Username 'aefffb2d'

The main search bar shows the user's name 'aefffb2d'. Below it, there are filters: 'Search by repository name' with a magnifying glass icon, 'All Content' with a dropdown arrow, and a 'Create repository' button.

A repository card for 'aefffb2d / nginx' is displayed, containing the following information:

- Repository name: aefffb2d / nginx
- Description: Contains: Image | Last pushed: 11 days ago
- Status: Inactive
- Star count: 0
- Download count: 0
- Visibility: Public

To the right, there is a call-to-action box for organizations:

- Icon: Three interconnected shapes (red circle, blue hexagon, green triangle) with a lock icon.
- Text: Create an Organization
Manage Docker Hub repositories
with your team

Docker Hub

```
Command Prompt      X + ▾

C:\Users\siras>docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
java-docker     latest    1ebe35c5afc0  5 minutes ago  646MB

C:\Users\siras>docker tag java-docker java-docker:1.0

C:\Users\siras>docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
java-docker     1.0      1ebe35c5afc0  5 minutes ago  646MB
java-docker     latest    1ebe35c5afc0  5 minutes ago  646MB

C:\Users\siras>docker login
Authenticating with existing credentials ...
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/

C:\Users\siras>docker tag java-docker aeffb2d/java-docker:1.0

C:\Users\siras>docker push aeffb2d/java-docker:1.0
The push refers to repository [docker.io/aeffb2d/java-docker]
c635cbc177da: Pushed
6abc22df5f0c: Pushed
678cf466f7d7: Pushed
0a4365111efd: Pushed
05fd14ba219a: Pushed
406e3a01a3ed: Pushed
3bccb20dc942: Mounted from library/eclipse-temurin
ed32c2a47a0a: Mounted from library/eclipse-temurin
535126407d49: Mounted from library/eclipse-temurin
59c56aee1fb4: Mounted from library/ubuntu
1.0: digest: sha256:ef13bab055ee76531a1c89b10a92f2024e7c8feccc1d2e1aaafe5501f046b133 size: 2419

C:\Users\siras>
```

Docker Hub

Screenshot of the Docker Hub user interface showing the profile of user 'aeffb2d'.

The top navigation bar includes:

- Docker Hub logo
- Search bar: Search Docker Hub
- Navigation links: Explore, Repositories, Organizations, Help
- User account information: Upgrade, Fingerprint icon, Username 'aeffb2d', and a dropdown menu

Profile settings and repository search filters are displayed:

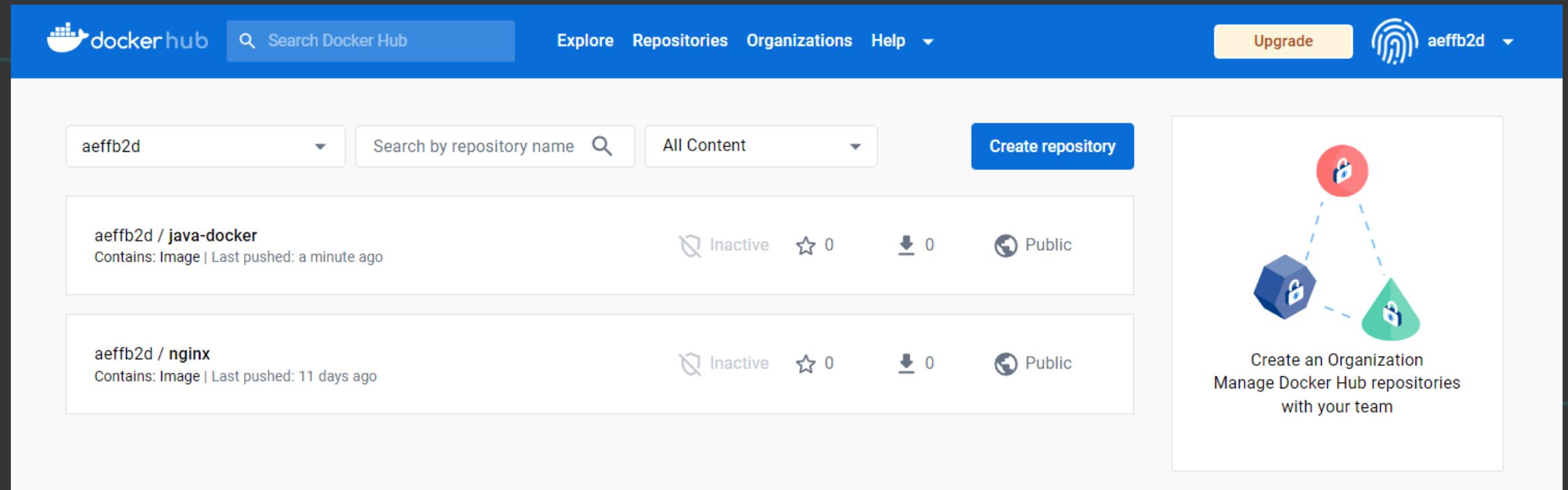
- Selected user: aeffb2d
- Search by repository name: Search by repository name
- Content type: All Content
- Create repository button

Two repositories are listed:

- aeffb2d / java-docker**
Contains: Image | Last pushed: a minute ago
Status: Inactive | Stars: 0 | Downloads: 0 | Public
- aeffb2d / nginx**
Contains: Image | Last pushed: 11 days ago
Status: Inactive | Stars: 0 | Downloads: 0 | Public

A sidebar on the right side of the profile page provides options for organization management:

- Create an Organization
- Manage Docker Hub repositories with your team



aeffb2d

Repositories

java-docker

General

Using 0 of 1 private repositories. [Get more](#)

General

Tags

Builds

Collaborators

Webhooks

Settings

 Add a short description for this repository

Update

The short description is used to index your content on Docker Hub and in search engines. It's visible to users in search results.

 aeffb2d / java-docker

Description

This repository does not have a description

 Last pushed: 18 minutes ago

Docker commands

Public View

To push a new tag to this repository,

```
docker push aeffb2d/java-docker:tagname
```

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
1.0		Image	---	18 minutes ago

[See all](#)[Go to Advanced Image Management](#)

Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#)[Upgrade](#)Repository overview An overview describes what your repository is and how to run it. It displays in [the public view of your repository](#).[Add overview](#)

Demo

<Infrastructure Engineering>

Bootcamp

Chapter 11

การใช้งาน Docker พื้นฐาน Container และ Host



Run a Container



```
docker run [options] <image_name:tag>
```

List running containers



```
docker ps
```

Stop a running container



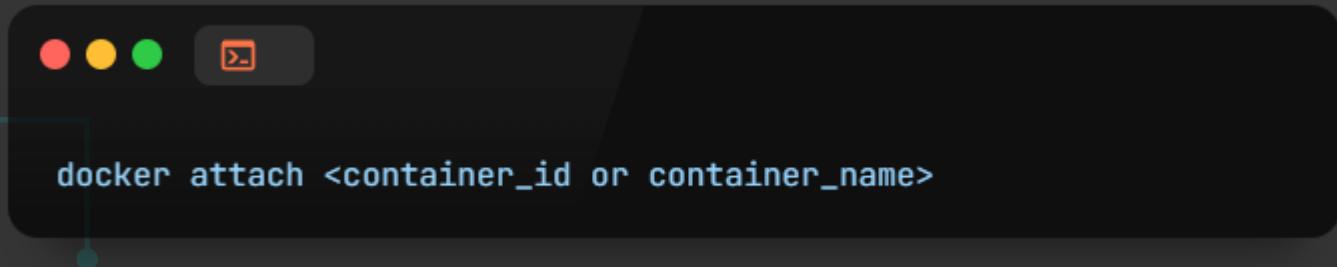
```
docker stop <container_id or container_name>
```

Restart a container

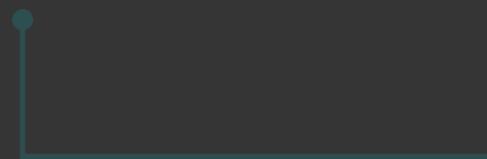


```
docker restart <container_id or container_name>
```

Attach to a running container



```
docker attach <container_id or container_name>
```



Demo

<Infrastructure Engineering>

Bootcamp

Chapter 12

การใช้งาน Shell ร่วมกับ Docker Container





```
docker run -it <image_name:tag> /bin/bash
```

Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

```
C:\Users\siras>docker run -it ubuntu:latest /bin/bash  
root@17495f89b65d:/# |
```

```
C:\Users\siras>docker run -it ubuntu:latest /bin/bash  
root@17495f89b65d:/# ls  
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var  
root@17495f89b65d:/# |
```



```
docker run -it <image_name:tag> /bin/bash
```

```
C:\Users\siras>docker run -itd ubuntu:latest /bin/bash  
728137dc1ca57ca8adcf6124f05a8ee5e6cf2bff0010207c75985b5cbbb8faf  
C:\Users\siras>
```



```
docker exec -it <container_id or container_name> /bin/bash
```

```
C:\Users\siras>docker exec -it competent_jepsen /bin/bash
root@728137dc1ca5:/# |
```

```
C:\Users\siras>docker exec -it java-pet /bin/bash
root@74f6a535bb05:/app# ls
mvnw pom.xml src target
root@74f6a535bb05:/app# |
```



```
docker exec -it -u <username> <container_id or container_name> /bin/bash
```

Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

```
C:\Users\siras>docker exec -it -u my_user java-pet /bin/bash  
unable to find user my_user: no matching entries in passwd file
```

```
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.
```

```
C:\Users\siras>docker exec -it -u my_user java-pet /bin/bash
unable to find user my_user: no matching entries in passwd file
```

```
C:\Users\siras>
C:\Users\siras>docker exec -it java-pet cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin

C:\Users\siras>docker exec -it -u root java-pet /bin/bash
root@74f6a535bb05:/app# |
```

<Basic Docker Commands>

```
C:\Users\siras>docker exec java-pet ls -l
total 36
-rwxr-xr-x 1 root root 11290 Aug  4 06:11 mvnw
-rwxr-xr-x 1 root root 15802 Aug  4 06:11 pom.xml
drwxr-xr-x 5 root root  4096 Aug  4 06:11 src
drwxr-xr-x 7 root root  4096 Aug  6 00:42 target
```



<Basic Docker Commands>

```
C:\Users\siras>docker run --rm ubuntu:latest ls -l
total 48
lrwxrwxrwx  1 root root    7 Jun 24 02:02 bin → usr/bin
drwxr-xr-x  2 root root 4096 Apr 18 2022 boot
drwxr-xr-x  5 root root   340 Aug  6 00:57 dev
drwxr-xr-x  1 root root 4096 Aug  6 00:57 etc
drwxr-xr-x  2 root root 4096 Apr 18 2022 home
lrwxrwxrwx  1 root root    7 Jun 24 02:02 lib → usr/lib
lrwxrwxrwx  1 root root    9 Jun 24 02:02 lib32 → usr/lib32
lrwxrwxrwx  1 root root    9 Jun 24 02:02 lib64 → usr/lib64
lrwxrwxrwx  1 root root   10 Jun 24 02:02 libx32 → usr/libx32
drwxr-xr-x  2 root root 4096 Jun 24 02:02 media
drwxr-xr-x  2 root root 4096 Jun 24 02:02 mnt
drwxr-xr-x  2 root root 4096 Jun 24 02:02 opt
dr-xr-xr-x 320 root root    0 Aug  6 00:57 proc
drwx----- 2 root root 4096 Jun 24 02:06 root
drwxr-xr-x  5 root root 4096 Jun 24 02:06 run
lrwxrwxrwx  1 root root    8 Jun 24 02:02 sbin → usr/sbin
drwxr-xr-x  2 root root 4096 Jun 24 02:02 srv
dr-xr-xr-x 11 root root    0 Aug  6 00:57 sys
drwxrwxrwt  2 root root 4096 Jun 24 02:06 tmp
drwxr-xr-x 14 root root 4096 Jun 24 02:02 usr
drwxr-xr-x 11 root root 4096 Jun 24 02:06 var
```

Demo

<Infrastructure Engineering>

Bootcamp

Chapter 13

การจัดการ Port ใน Docker

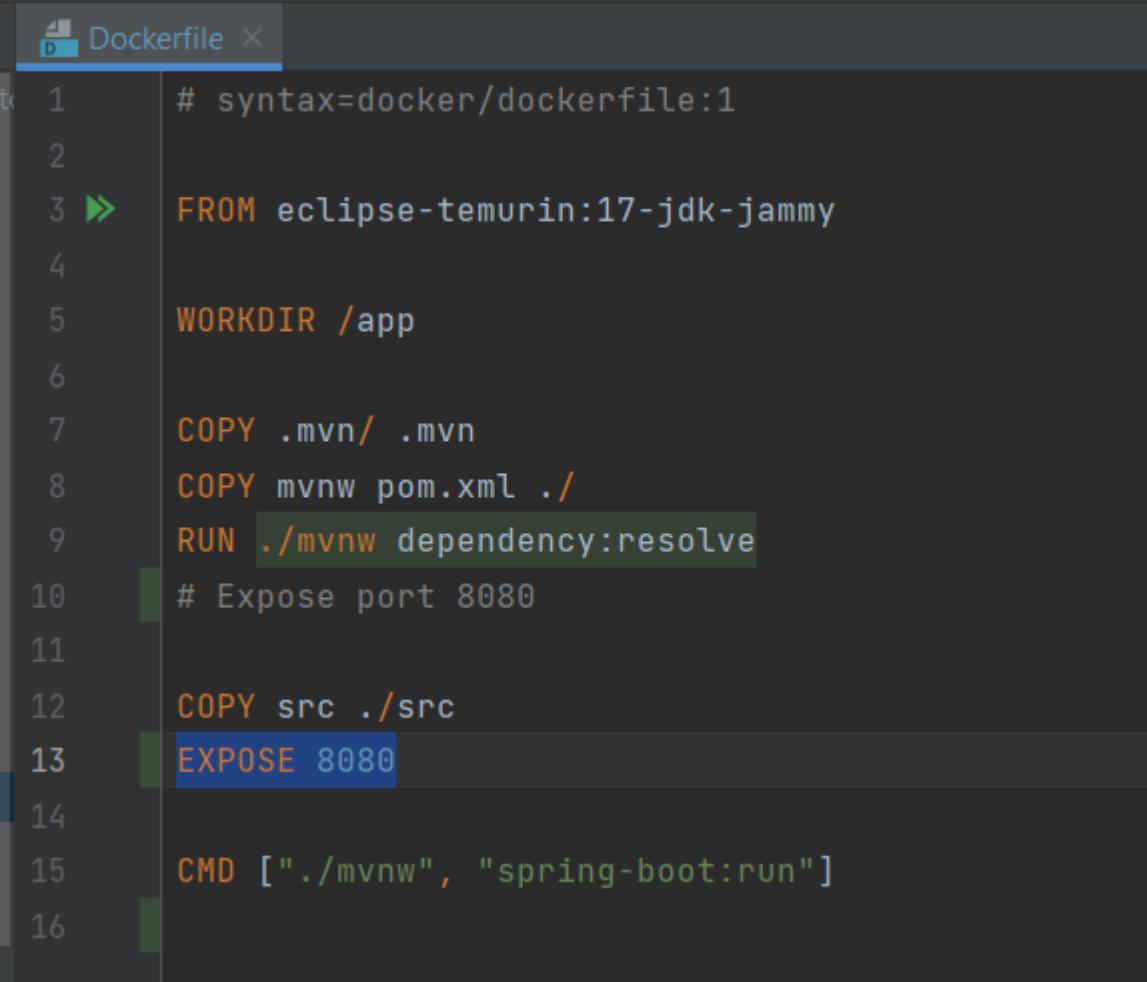


+-----+ <td> Main Web <td> API </td><td>+-----+</td></td>	Main Web <td> API </td> <td>+-----+</td>	API	+-----+
Application :8080 <td> Service :9090 </td> <td></td> <td></td>	Service :9090		
(Frontend) <td> (RESTful APIs) </td> <td></td> <td></td>	(RESTful APIs)		
+-----+	+-----+		

Check Port

```
C:\Users\siras>docker port java-pet2  
8080/tcp → 0.0.0.0:8081
```

Docker file

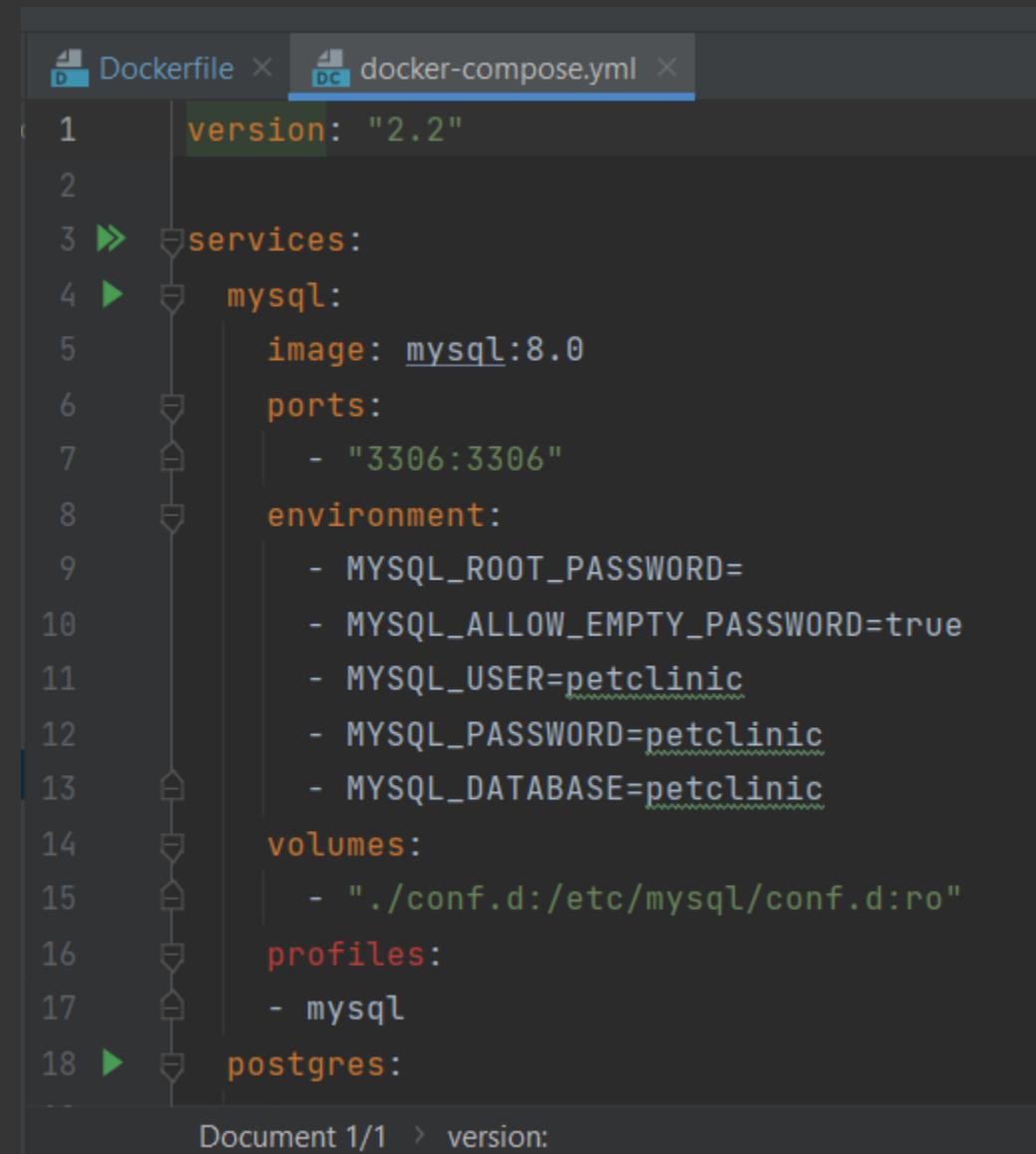


A screenshot of a code editor window titled "Dockerfile". The editor shows a Dockerfile with the following content:

```
1 # syntax=docker/dockerfile:1
2
3 ► FROM eclipse-temurin:17-jdk-jammy
4
5 WORKDIR /app
6
7 COPY .mvn/.mvn
8 COPY mvnw pom.xml ./
9 RUN ./mvnw dependency:resolve
10 # Expose port 8080
11
12 COPY src ./src
13 EXPOSE 8080
14
15 CMD ["./mvnw", "spring-boot:run"]
16
```

The code editor has syntax highlighting and line numbers. A green highlight covers the "RUN" command, and a blue highlight covers the "EXPOSE" command.

Docker compose



```
version: "2.2"

services:
  mysql:
    image: mysql:8.0
    ports:
      - "3306:3306"
    environment:
      - MYSQL_ROOT_PASSWORD=
      - MYSQL_ALLOW_EMPTY_PASSWORD=true
      - MYSQL_USER=petclinic
      - MYSQL_PASSWORD=petclinic
      - MYSQL_DATABASE=petclinic
    volumes:
      - "./conf.d:/etc/mysql/conf.d:ro"
    profiles:
      - mysql
  postgres:
```

Document 1/1 > version:

Demo

<Infrastructure Engineering>

Bootcamp

Chapter 14

เรื่องราวนในการจัดการ Storage space



Demo

<Infrastructure Engineering>

Bootcamp

Chapter 15

Networking ၏ Docker

KBTG
KAMPUS
ClassNest



Docker Network

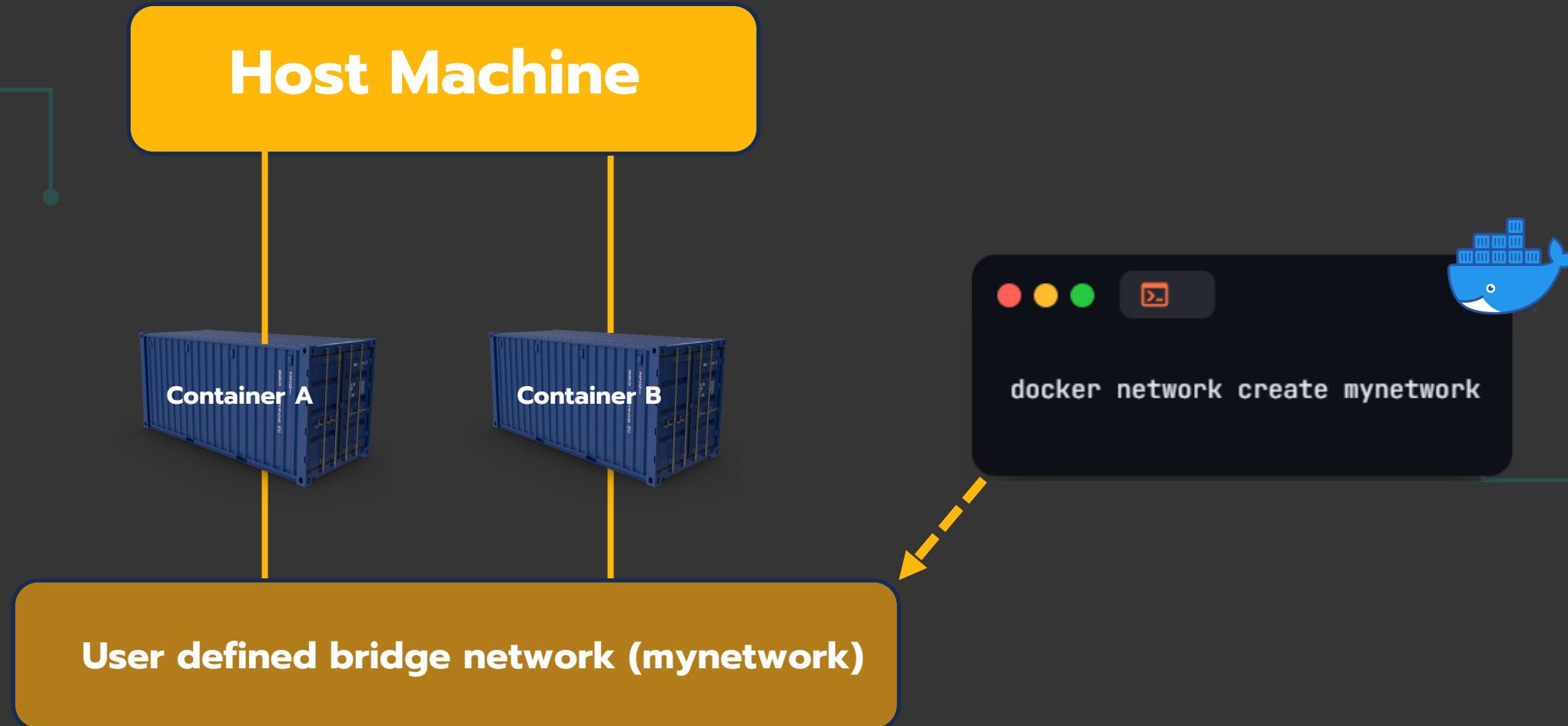
- **Default network scope**
- **User defined bridge network scope**

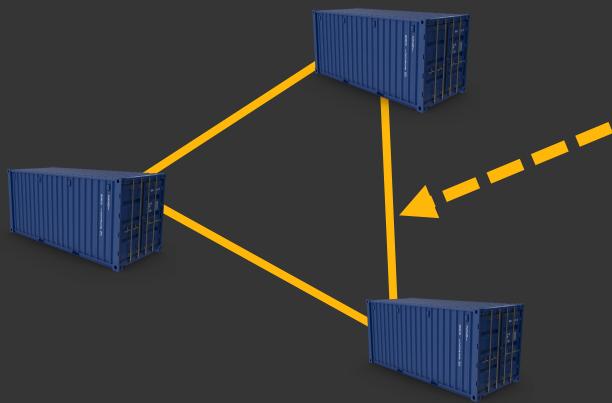
Docker Network

Default Bridge Network (bridge)

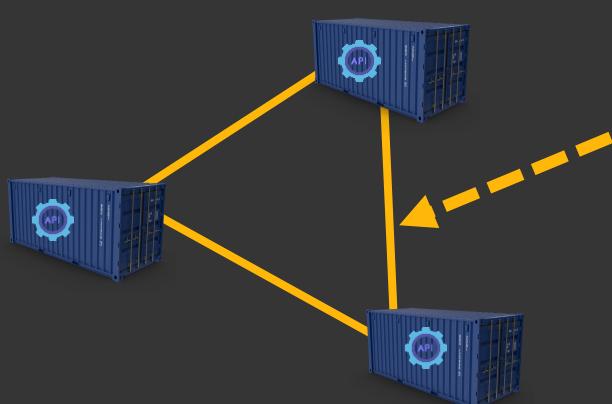


Docker Network



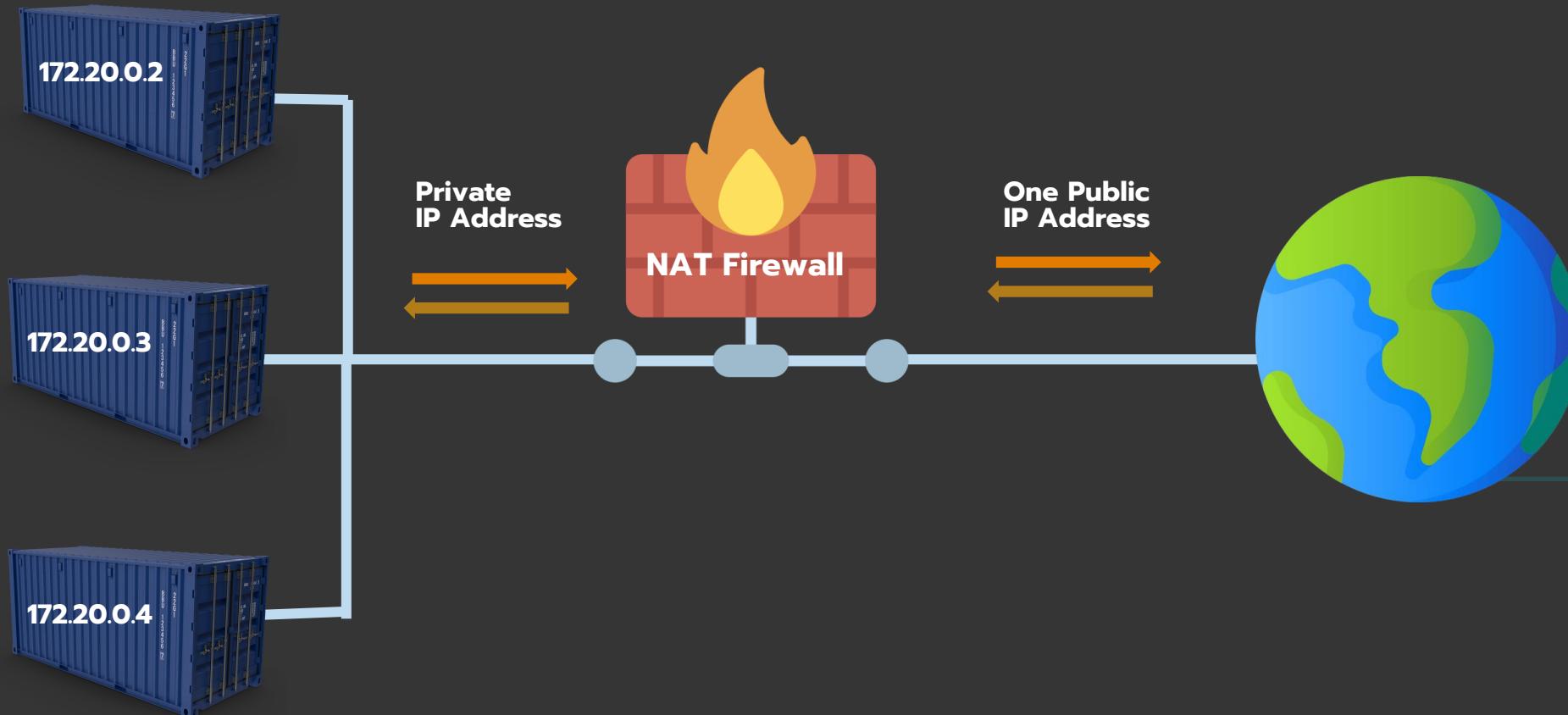


Bridge Network "web-app"



Bridge Network "web-api"

NAT Firewall



<Infrastructure Engineering>

Bootcamp

Chapter 16

การเชื่อมโยงระหว่าง Container ใน Docker



Demo

<Infrastructure Engineering>

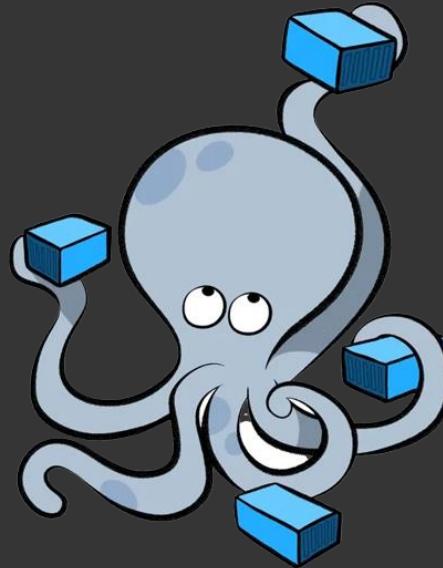
Bootcamp

Chapter 17

Concept และ การใช้งาน Docker Compose

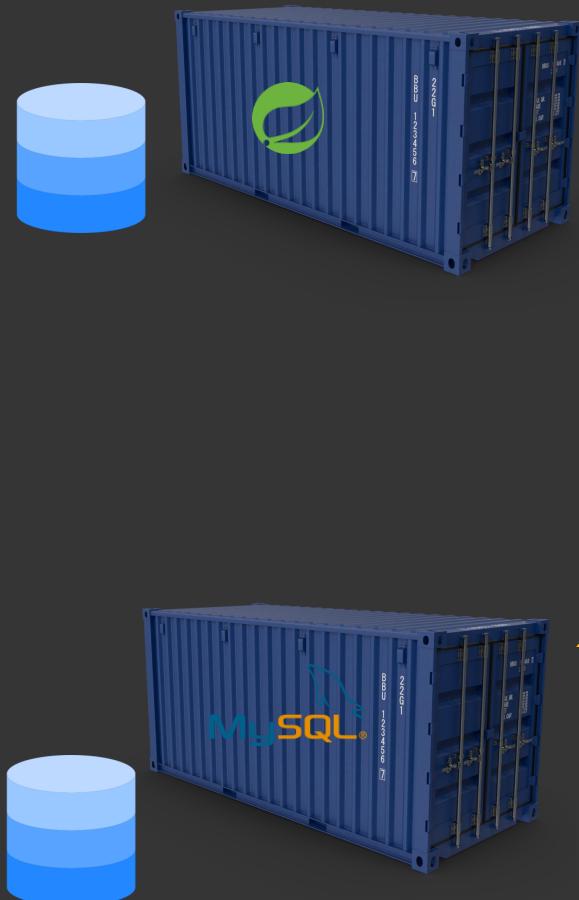


Docker Compose



Docker Compose is a tool that helps you define and share multi-container applications.

Docker Compose file



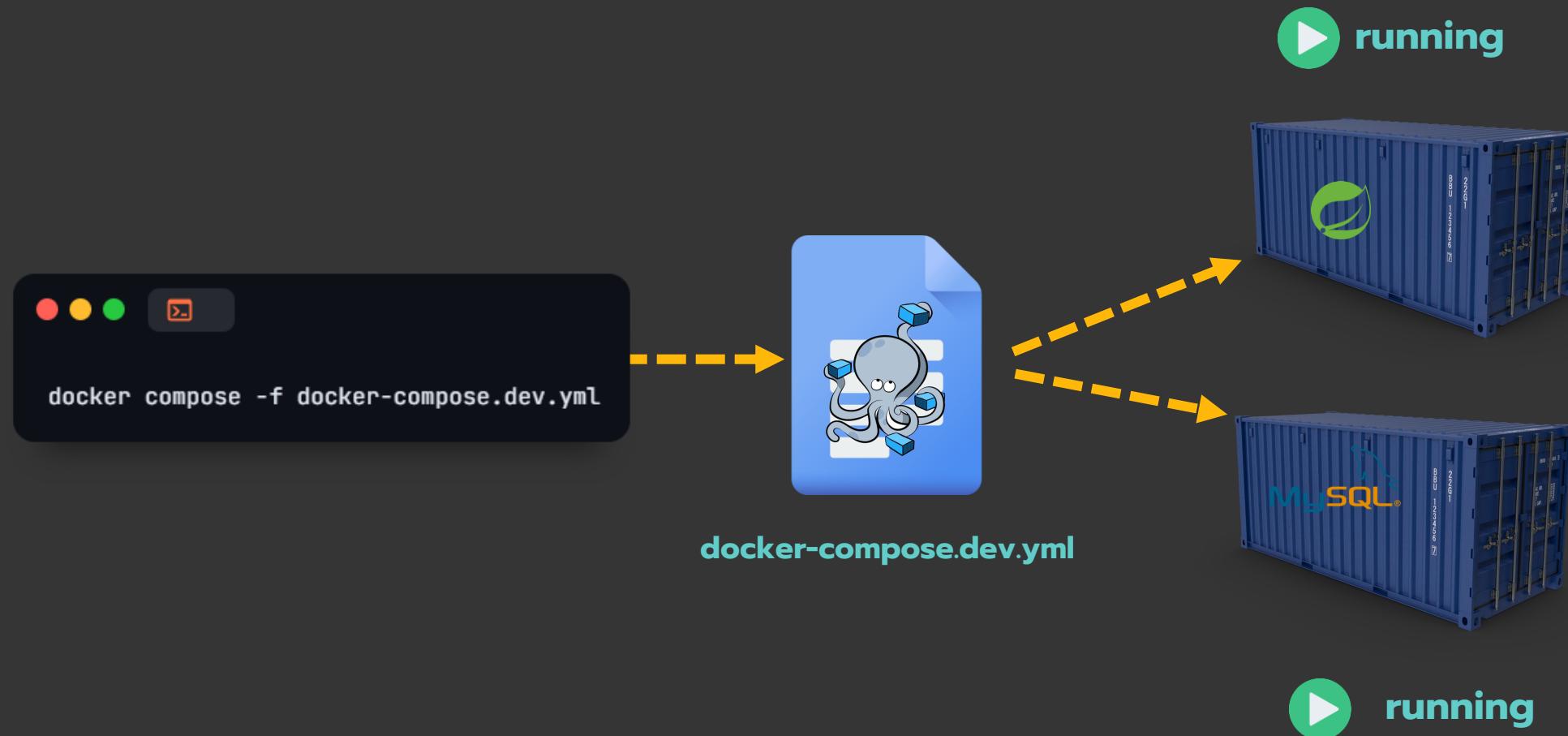
A screenshot of a Mac OS X-style window titled "docker-compose.yml". The window contains the following YAML configuration:

```
version: '3.8'
services:
  petclinic:
    build:
      context: .
      target: development
    ports:
      - "8000:8000"
      - "8080:8080"
    environment:
      - SERVER_PORT=8080
      - MYSQL_URL=jdbc:mysql://mysqlserver/petclinic
    volumes:
      - ./:/app
    depends_on:
      - mysqlserver

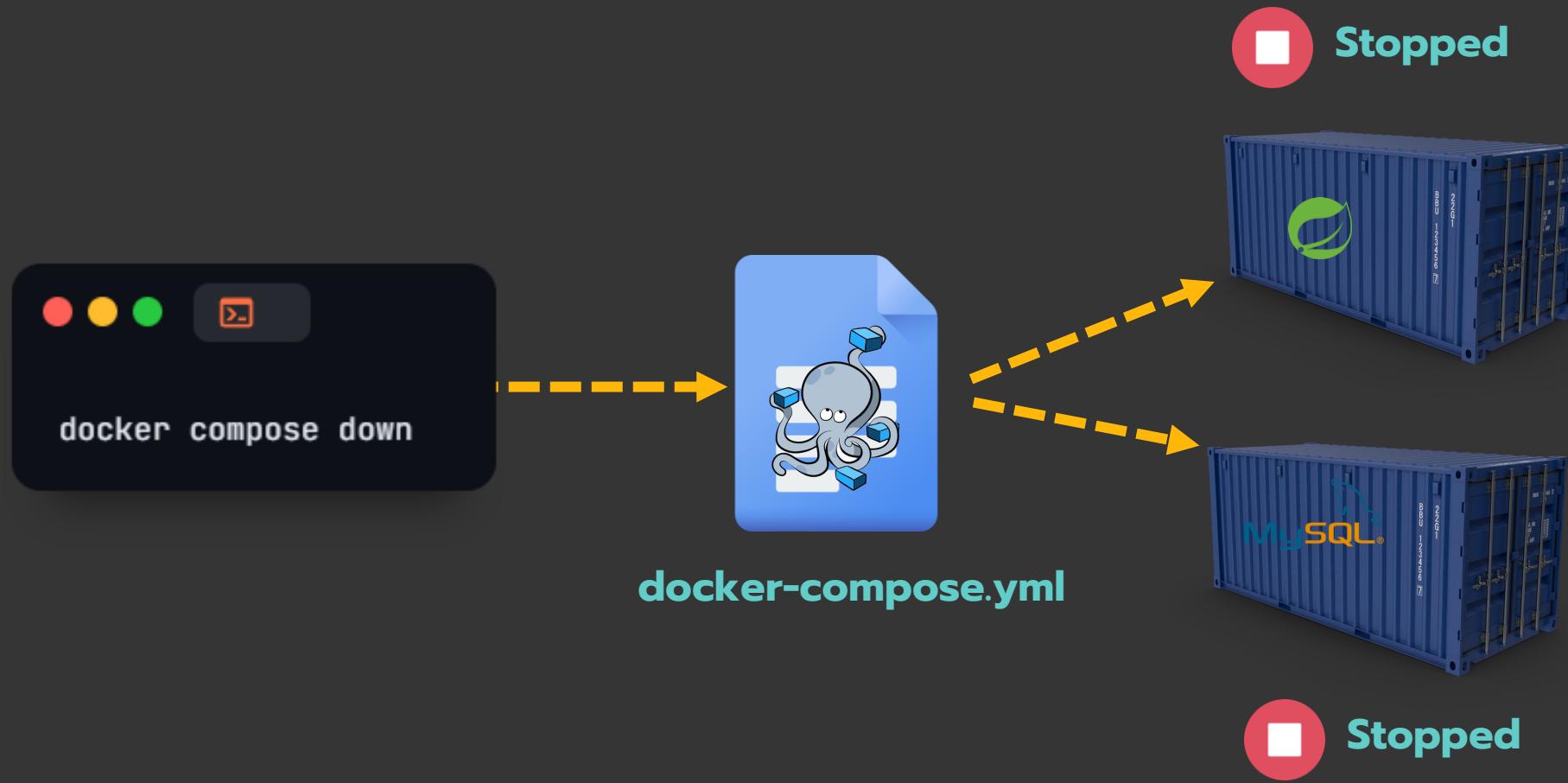
mysqlserver:
  image: mysql:8.0
  ports:
    - "3306:3306"
  environment:
    - MYSQL_ROOT_PASSWORD=
    - MYSQL_ALLOW_EMPTY_PASSWORD=true
    - MYSQL_USER=petclinic
    - MYSQL_PASSWORD=petclinic
    - MYSQL_DATABASE=petclinic
  volumes:
    - mysql_data:/var/lib/mysql
    - mysql_config:/etc/mysql/conf.d
volumes:
  mysql_data:
  mysql_config:
```

The window also features a small blue octopus icon with a wrench and a gear, positioned in the top right corner.

docker compose -f <compose_file.yml>



docker compose down



Demo

<Infrastructure Engineering>

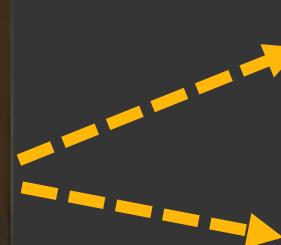
Bootcamp

Chapter 18

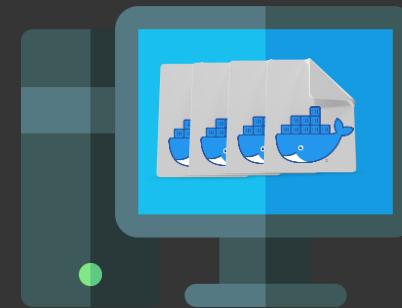
การใช้งานกับ Public Repository



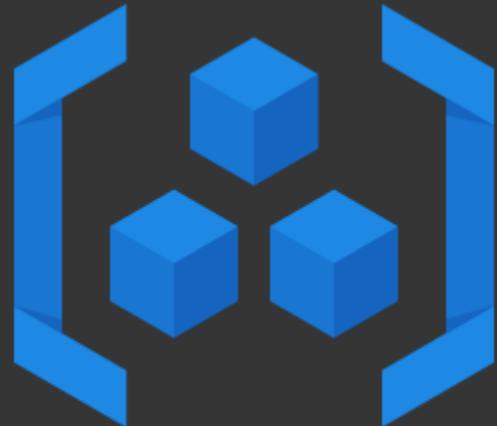
Container Registry



Local
Public



Container Registry



Demo

<Infrastructure Engineering>

Bootcamp

Chapter 19

สรุปบทเรียนสำหรับพื้นฐาน Docker

KBTG
KAMPUS
ClassNest

