

05506007 Operating Systems (1/2568)

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

รหัสนักศึกษา.....ชื่อ-สกุล.....

ปฏิบัติการ ครั้งที่ 3 – Containerization Tool

วัตถุประสงค์ของการปฏิบัติการ

- สามารถใช้งาน Docker Container และสร้าง Docker Images ได้

1. Docker Images และ Docker Container

เปิดการใช้งาน Docker Desktop WSL 2 backend on Windows

1.1. Download และติดตั้ง Docker Desktop for Windows ([Download Here](#))

- ตรวจสอบ WSL mode พิมพ์คำสั่งที่ Power Shell “wsl -l -v”
- กำหนด WSL ให้เป็น Version 2 พิมพ์คำสั่งที่ Power Shell “wsl --set-default-version 2”
- ติดตั้ง Distro (VM) บน WSL พิมพ์คำสั่งที่ Power Shell “wsl --install <distro name>”
- กำหนด Default Distro พิมพ์คำสั่งที่ Power Shell “wsl --set-default <distro name>”

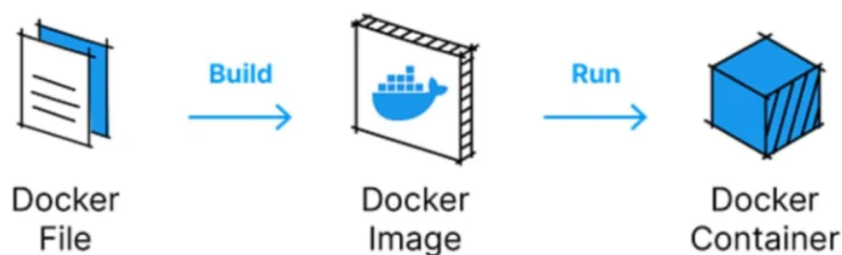
1.2. Login เข้าสู่ WSL Distro (VM) และติดตั้ง Docker ใน Distro พิมพ์คำสั่งที่ Power Shell เพื่อ Login เข้าสู่ WSL Distro “wsl ~ -u root -d <distro name>”

- เมื่อเข้าสู่ Distro (VM) ให้ทำการติดตั้ง Docker ด้วยลำดับคำสั่งต่อไปนี้
“sudo apt-get update; sudo apt-get upgrade; sudo apt-get install docker”
- หลังจากติดตั้ง Docker ใน Linux OS ให้รันคำสั่ง “docker ps -a” เพื่อตรวจสอบการติดตั้งและการทำงานของ Docker Desktop WSL 2 backend on Windows
- ทดสอบรันคำสั่ง “docker info” อธิบายผลลัพธ์
- ทดสอบรันคำสั่ง “docker --help” อธิบายผลลัพธ์

1.3.สมัคร Docker Hub Account เพื่อใช้งาน Docker Hub (<https://hub.docker.com/>) จากนั้นทดสอบ

Login สู่ Docker Hub จาก WSL Distro (VM) ด้วยคำสั่ง “docker login -u”

- ทดสอบรันคำสั่ง “docker images” บน WSL Distro (VM) พร้อมอธิบายผล
- ทำการ Download docker image (hello-world image) จาก Docker Hub ด้วยการรันคำสั่ง “docker pull hello-world” และถ้าต้องการรัน hello-world image ให้เป็น container จะต้องรันอย่างไร
- ทดสอบรันคำสั่ง “docker run -it ubuntu bash” บน WSL Distro (VM) พร้อมอธิบายผล
- หากต้องการตรวจสอบรายการ Docker Images ที่เก็บอยู่บนเครื่องจะต้องใช้คำสั่งใด
- หากต้องการตรวจสอบรายการ Docker Container ที่รันอยู่บนเครื่องจะต้องใช้คำสั่งใด
- ทดสอบรันคำสั่ง “docker container stats” บน WSL Distro (VM) พร้อมอธิบายผล
- หากต้องการรัน Nginx Container เวอร์ชัน 1.27.0 แบบ Background Mode พร้อม Publish Port (Nginx run with port 80) จะต้องใช้คำสั่งใด
- หากต้องการหยุดการทำงานของ Nginx Container จะต้องรันคำสั่งใด
- หากต้องการลบ Nginx Container ออกเพื่อเรียกคืนทรัพยากร จะต้องรันคำสั่งใด



รูปที่ 1 แสดงกระบวนการรัน Docker Container

1.4. สร้าง Docker Image (ทดสอบโดยการสร้าง Docker Image ของ nodejs)

ทดสอบการรัน Nodejs Application บน WSL Distro (VM)

1.4.1 ทำการติดตั้ง Nodejs และ npm package ดังนี้

- ติดตั้ง Nodejs บน WSL Distro (VM) ด้วยการรันคำสั่ง
 - “sudo apt-get update; sudo apt-get install nodejs”
- ติดตั้ง npm package บน WSL Distro (VM) ด้วยการรันคำสั่ง “sudo apt-get install npm”
- ทดสอบรันคำสั่ง “node -v” และ “npm -v” บน WSL Distro (VM) พร้อมอธิบายผลลัพธ์

1.4 สร้าง Web Application ด้วย Express Framework บน Nodejs ดังนี้

- ให้นักศึกษาสร้าง Directory ชื่อ “nodejs_<รหัสนักศึกษา>” เช่น “nodejs_1234” จากนั้น เข้าไปยัง Directory ดังกล่าว
- รันคำสั่ง “npm init” เพื่อสร้าง package.json
- รันคำสั่ง “npm install express” เพื่อติดตั้ง Express Framework ใน Directory
- สร้าง Web Application ด้วย javascript โดยตั้งชื่อไฟล์ว่า “app<รหัสนักศึกษา>.js” เช่น “app1234.js” และพิมพ์ Code ที่ 1 ลงในไฟล์

Code ที่ 1

| No. | File: app<รหัสนักศึกษา>.js |
|-----|---|
| 1 | const express = require('express') |
| 2 | const app = express() |
| 3 | const port = 3000 |
| 4 | |
| 5 | app.get('/', (req, res) => { |
| 6 | res.send('Hello CS KMITL !!!') |
| 7 | }) |
| 8 | |
| 9 | app.listen(port, () => { |
| 10 | console.log(`Listening at http://localhost:\${port}`) |
| 11 | }) |

- รันคำสั่ง “node app<รหัสนักศึกษา>.js” เพื่อรัน Web Application และทดสอบโดยการพิมพ์ `http://localhost:3000` ที่ Web browser พร้อมแสดงผลลัพธ์

1.5.เขียน Docker File สำหรับสร้าง Docker Image ของ Nodejs Application

- สร้าง Docker File ใน Directory “nodejs_<รหัสนักศึกษา>” ตั้งชื่อว่า “Dockerfile” โดยใส่คำสั่งของ Docker File ดัง Code ที่ 2

Code ที่ 2

| No. | File: Dockerfile |
|-----|--------------------------------------|
| 1 | FROM node:18-alpine |
| 2 | WORKDIR /nodejs_<รหัสนักศึกษา> |
| 3 | COPY package.json . |
| 4 | RUN npm install |
| 5 | COPY . . |
| 6 | EXPOSE 3000 |
| 7 | CMD [“node”, “app<รหัสนักศึกษา>.js”] |

- สร้างไฟล์ “.dockerignore” เพื่อกำหนดไม่ให้ Copy ไฟล์หรือ Directory ที่ไม่ต้องการ ไปยัง Docker Image โดยใส่ข้อมูล ดัง Code ที่ 3

Code ที่ 3

| No. | File: .dockerignore |
|-----|---------------------|
| 1 | node_modules |
| 2 | npm-debug.log |

- ทดสอบรันคำสั่ง “docker build -t nodejs-app-<รหัสนักศึกษา> .” บน WSL Distro (VM) เพื่อสร้าง Docker Image ของ Nodejs Application
 - คำถาม Option “-t” ทำหน้าที่อะไร
- ตรวจสอบ Docker Images ที่สร้างด้วยคำสั่งใด
- รัน Docker Container จาก Docker Image ที่สร้าง ด้วยคำสั่ง
“docker run -d -p 8080:3000 nodejs-app-<รหัสนักศึกษา>” สังเกตผลลัพธ์จากการรัน

- คำถาม option “-d” ทำหน้าที่อะไร
- คำถาม option “-p” ทำหน้าที่อะไร
- ทดสอบเปิด Web Application ที่เกิดจากการรัน Docker Container ด้วย Web Browser ที่ URL-> http://localhost:3000 และ http://localhost:8080 สังเกตผลลัพธ์พร้อมอธิบายผล
.....
.....
.....

1.6.สำรวจ Docker Image ไปเก็บบน Docker Hub (https://hub.docker.com/)

- เข้าสู่ Docker Hub (https://hub.docker.com/) ผ่าน Web Browser
 - จากหน้าแรกของ Docker Hub ให้กดที่ปุ่ม “Create repository” เพื่อสร้างที่เก็บ Docker Image จากนั้นกำหนด Repository Name เป็น “nodejs-app-<รหัสนักศึกษา>” กด Create ดังรูปที่ 2

The screenshot shows the Docker Hub 'Create repository' page. The top navigation bar includes 'dockerhub', 'Explore', 'Repositories', and 'Organizations'. Below the navigation bar, there's a search bar and a button to '+K'. The main content area is titled 'Create repository' and includes a 'Namespace' dropdown set to 'abanjong', a 'Repository Name' input field containing 'nodejs-demo-1234', and a 'Short description' text area. To the right, there's a 'Pushing images' section with CLI commands: 'docker tag local-image:tagname new-repo:tagname' and 'docker push new-repo:tagname'. Below that, a note says 'Make sure to replace tagname with your desired image repository tag.' At the bottom, there's a 'Visibility' section with two radio buttons: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Appears in Docker Hub search results', and the 'Private' option is 'Only visible to you'. At the very bottom, there are 'Cancel' and 'Create' buttons.

รูปที่ 2 กำหนด Repository บน Docker Hub

- รันคำสั่ง
“docker tag nodejs-demo-<รหัสนักศึกษา> <docker hub’s username>/nodejs-demo-<รหัสนักศึกษา>:latest” บน WSL Distro (VM) เพื่อระบุ Tag และกำหนด Repository ให้กับ Docker Image ที่ต้องการเก็บบน Docker Hub

- รันคำสั่ง
“docker push <docker hub’s username>/nodejs-app-<รหัสนักศึกษา>:latest” บน WSL Distro (VM) เพื่อ Upload Docker Image ไปเก็บที่ Docker Hub Repository ที่กำหนด
- ต้องการลบ Docker Container ชื่อ “nodejs-app-<รหัสนักศึกษา>” จะต้องรันคำสั่งใด
.....
- ต้องการลบ Docker Image ชื่อ “nodejs-demo-<รหัสนักศึกษา>” จะต้องรันคำสั่งใด
.....
- ต้องการ Download Docker Image ชื่อ “nodejs-demo-<รหัสนักศึกษา>” จาก Docker Hub จะต้องรันคำสั่งใด
.....

1.7. ทดสอบสร้าง Docker Image สำหรับ mysql

- สร้าง Directory “mysql_<รหัสนักศึกษา>” เช่น “mysql_1234” ที่ Home Directory ของ WSL Distro (VM)
- เข้าไปที่ Directory “mysql_<รหัสนักศึกษา>” แล้วสร้างไฟล์ชื่อ “Dockerfile” ใส่คำสั่งของ Docker File ดัง Code ที่ 4

Code ที่ 4

| No. | File: Dockerfile |
|-----|---|
| 1 | FROM mysql:8.0.32 |
| 2 | WORKDIR /app |
| 3 | MAINTAINER <email นักศึกษา> |
| 4 | ENV MYSQL_ROOT_PASSWORD <กำหนด root password> |
| 5 | ENV MYSQL_USER <ชื่อนักศึกษา> |
| 6 | ENV MYSQL_PASSWORD <กำหนด user password> |
| 7 | ADD database.sql /docker-entrypoint-initdb.d |
| 8 | EXPOSE 3306 |

- จากนั้นสร้างไฟล์ชื่อ “database.sql” ใน Directory “mysql_<รหัสนักศึกษา>” เพื่อกำหนดคำสั่ง SQL ให้สร้างฐานข้อมูลเริ่มต้น โดยใส่คำสั่ง SQL ดัง Code ที่ 5

Code ที่ 5

| No. | File: database.sql |
|-----|--|
| 1 | RANT ALL PRIVILEGES ON *.* TO '<ชื่อนักศึกษา>@'%'; |
| 2 | CREATE DATABASE test; |
| 3 | USE test; |
| 4 | CREATE TABLE user (|
| 5 | id INTEGER NOT NULL AUTO_INCREMENT, |
| 6 | email VARCHAR(255) NOT NULL, |
| 7 | fullname VARCHAR(255) NOT NULL, |
| 8 | PRIMARY KEY (id) |
| 9 |); |
| 10 | INSERT INTO user (email, fullname) |
| 11 | VALUES ('test@email.com', 'CS KMITL'); |

○ หากต้องการสร้าง Docker Image ของ mysql จะต้องรันคำสั่งใด

.....

○ หลังจากสร้าง Docker Image ของ mysql ทำการรัน Docker Container ของ mysql ด้วย

คำสั่ง “docker run --name mysql-server --restart unless-stopped -d -p 3306:3306

<mysql docker image> --default-authentication-plugin=mysql_native_password”

- คำถาม Option “--restart unless-stopped” ทำหน้าที่อะไร

.....

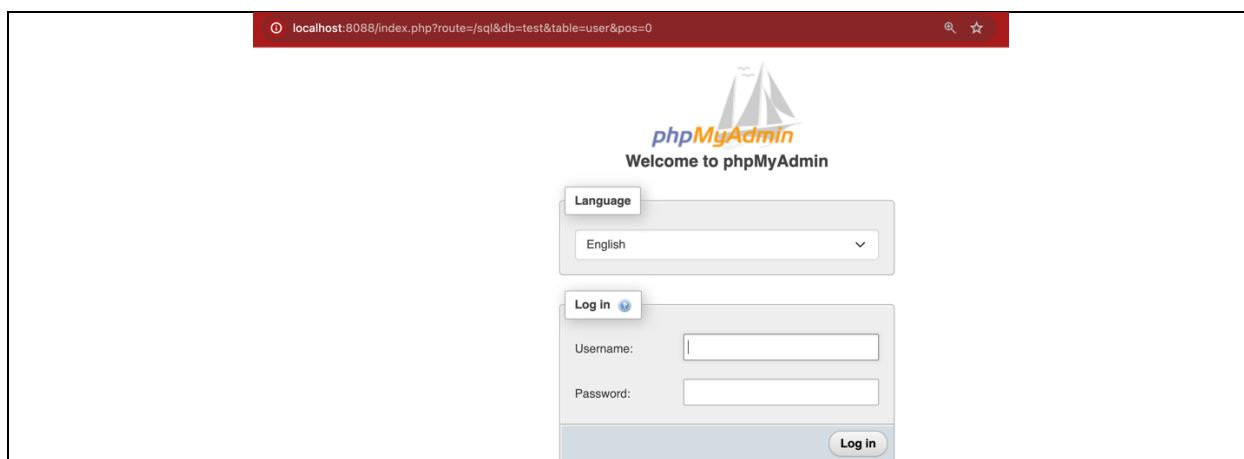
1.8.ทดสอบรัน phpmyadmin docker container และเชื่อมต่อกับ mysql docker container โดยการรัน

คำสั่ง “docker run --name phpmyadmin -d --restart unless-stopped -p 8088:80 --link mysql-server:db phpmyadmin”

- คำถาม Option “--link mysql-server:db” ทำหน้าที่อะไร

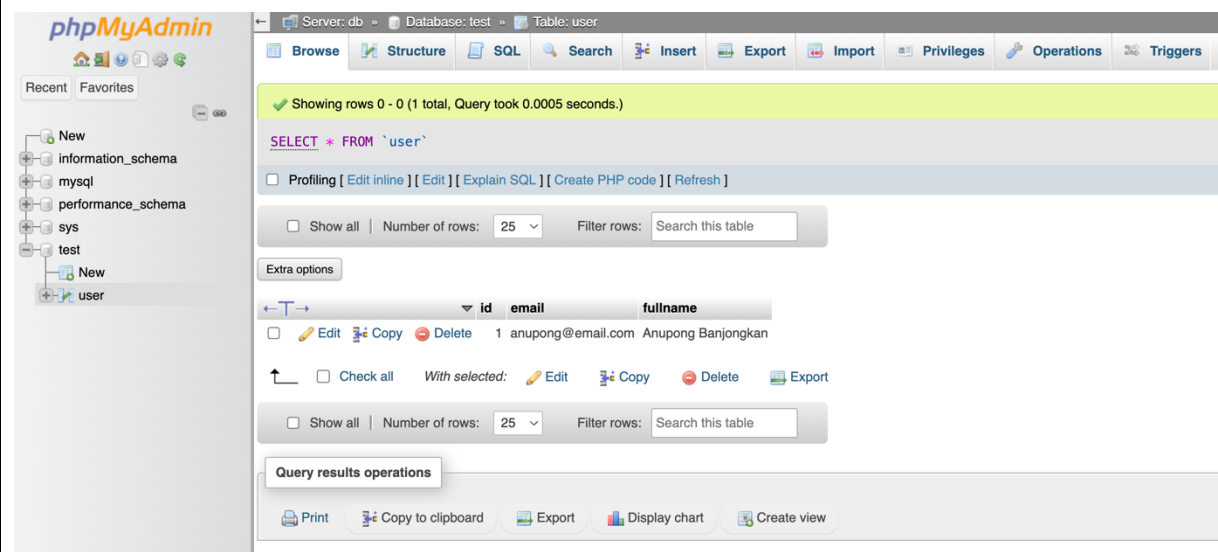
.....

1.9.ทดสอบเข้าใช้งาน phpMyAdmin ผ่าน Web Browser ด้วย URL -> “http://localhost:8088” จะแสดงหน้า Login ของ phpMyAdmin ดังรูปที่ 3



รูปที่ 3 แสดงหน้า Login ของ phpMyAdmin ที่รันผ่าน Docker Container

- 1.10. เมื่อ Login เข้า phpMyAdmin ด้วย username และ password ที่กำหนดในไฟล์ “Dockerfile” ของ mysql docker image ก็จะสามารถเห็น database และข้อมูลที่ผ่านการรันคำสั่ง SQL ที่กำหนดในไฟล์ “database.sql” ดังรูปที่ 4



รูปที่ 4 แสดงฐานข้อมูลและข้อมูลใน mysql docker container