ขั้นตอนการใช้งาน Docker เบื้อต้น

ขั้นตอนที่ 1: สร้างโฟลเดอร์โปรเจกต์

สมมติว่าโฟลเดอร์โปรเจกต์ชื่อ mydocker-shop

โครงสร้างไฟล์จะเป็นแบบนี้:





ขั้นตอนที่ 2: สร้างไฟล์ Dockerfile สำหรับเว็บ PHP

ไฟล์นี้จะสร้างอิมเมจ PHP 8.1 พร้อม Apache และ mysqli extension

ไฟล์ Dockerfile (สร้างในโฟลเดอร์ mydocker-shop)



FROM php:8.1-apache

ติดตั้ง mysqli extension สำหรับเชื่อม MySQL/MariaDB

RUN docker-php-ext-install mysqli

เปิด mod rewrite ของ Apache (ถ้าต้องใช้)

RUN a2enmod rewrite

กำหนดโฟลเดอร์งานภายใน container

WORKDIR /var/www/html

ไฟล์ docker-compose.yml

```
yaml

	☐ Copy

                                                                                          ⁰ Edit
version: '3.8'
services:
  web:
    build:
     context: .
      dockerfile: Dockerfile
    container_name: shop-web
    volumes:
     - ./www:/var/www/html
    ports:
      - "8080:80"
    depends_on:
      - mariadb
  mariadb:
    image: mariadb:10.5
    container_name: shop-db
    environment:
      MARIADB_ROOT_PASSWORD: rootpass
      MARIADB_DATABASE: shop_db
      MARIADB_USER: shopuser
      MARIADB_PASSWORD: shoppass
      - "3306:3306"
    volumes:
      db_data:/var/lib/mysql
  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: shop-pma
    environment:
     PMA_HOST: mariadb
     PMA_PORT: 3306
     PMA_USER: shopuser
                                                \downarrow
      PMA_PASSWORD: shoppass
    ports:
      - "8081:80"
    depends on:
      - mariadb
volumes:
  db_data:
```

```
services:
 web:
  build:
   context: .
   dockerfile: Dockerfile
  container_name: shop-web
  volumes:
   - ./www:/var/www/html
  ports:
   - "8080:80"
  depends_on:
   - mariadb
 mariadb:
  image: mariadb:10.5
  container_name: shop-db
  environment:
   MARIADB ROOT PASSWORD: rootpass
   MARIADB DATABASE: shop db
   MARIADB USER: shopuser
   MARIADB_PASSWORD: shoppass
  ports:
   - "3306:3306"
  volumes:
   - db_data:/var/lib/mysql
 phpmyadmin:
  image: phpmyadmin/phpmyadmin
  container_name: shop-pma
  environment:
   PMA HOST: mariadb
   PMA PORT: 3306
   PMA USER: shopuser
   PMA PASSWORD: shoppass
  ports:
   - "8081:80"
  depends_on:
   - mariadb
volumes:
```

db data:

version: '3.8'

ขั้นตอนที่ 4: สร้างไฟล์เว็บตัวอย่าง

สร้างโฟลเดอร์ชื่อ www แล้วสร้างไฟล์ index.php ข้างใน:

ไฟล์ www/index.php

```
<?php
$mysqli = new mysqli("mariadb", "shopuser", "shoppass", "shop_db");
if ($mysqli->connect_errno) {
   echo "เชื่อมต่อฐานข้อมูลไม่สำเร็จ: " . $mysqli->connect_error;
   exit();
}
echo "เชื่อมต่อฐานข้อมูลสำเร็จ!";
?>
```

ขั้นตอนที่ 5: รันคำสั่งสร้างและเปิด container

เปิด terminal/PowerShell ไปที่โฟลเดอร์ mydocker-shop แล้วรัน:

```
bash

docker-compose up -d --build
```

docker-compose up -d -build

ขั้นตอนที่ 6: ทดสอบเข้าถึงเว็บและ phpMyAdmin

- เข้าเว็บ: http://localhost:8080 ควรเห็นข้อความ "เชื่อมต่อฐานข้อมูลสำเร็จ!"
- เข้า phpMyAdmin: http://localhost:8081
 - Login ด้วย username: shopuser
 - รหัสผ่าน: shoppass
 - Server: mariadb



□ เข้าเว็บ: http://localhost:8080 — ควรเห็นข้อความ "เชื่อมต่อฐานข้อมูลสำเร็จ!"
 □ เข้า phpMyAdmin: http://localhost:8081
 Login ด้วย username: shopuser
 รหัสผ่าน: shoppass
 Server: mariadb

ขั้นตอน 7 (ถ้ามีไฟล์ SQL ต้องการนำเข้า)

สมมติไฟล์ SQL ชื่อ shop_db.sql อยู่ในโฟลเดอร์ mydocker-shop รันคำสั่งนี้เพื่อ import:

bash

docker exec -i shop-db mysql -u shopuser -pshoppass shop_db < shop_db.sql

1. ตรวจสอบว่า container รันอยู่ไหม

ให้รับคำสั่ง:

bash

docker ps

ผลลัพธ์ควรจะเห็น container ดังนี้:

CONTAINER NAME	STATUS	PORTS
shop-web	Up	0.0.0.0:8080->80/tcp
shop-pma	Up	0.0.0.0:8081->80/tcp
shop-db	Up	0.0.0.0:3306->3306/tcp

ถ้าไม่เจอ shop-web หรือ shop-pma หรือมีสถานะ Exited, ให้รันคำสั่งนี้เพื่อดู log:

bash

docker-compose logs web
docker-compose logs phpmyadmin

2. ลองหยุดแล้วรันใหม่

bash

docker-compose down
docker-compose up -d --build

รอให้ระบบโหลดขึ้นให้ครบ (ใช้ docker ps ดูสถานะอีกครั้ง)

3. ปิด Firewall ชั่วคราว (Windows)

หากยังไม่ได้ อาจเป็นเพราะ Windows Firewall หรือ antivirus บางตัวบล็อกพอร์ต 8080 หรือ 8081

ให้ลองชั่วคราวโดย:

- เปิด Windows Defender Firewall
- เลือก Turn Windows Defender Firewall off
- ลองเข้า http://localhost:8080 และ http://localhost:8081 อีกครั้ง

(เปิดกลับหลังทดสอบเสร็จนะครับ เพื่อความปลอดภัย)

4. เช็ดว่า Docker ใช้ Linux container อยู่

คลิกขวาที่ Docker ที่ Taskbar แล้วดูว่าเป็น "Switch to Windows containers" หรือ "Switch to Linux containers"

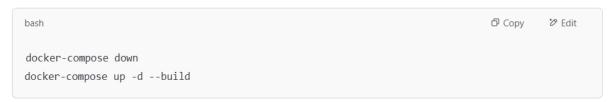
🔽 ต้องเป็น Linux containers

5. ตรวจสอบว่า Docker ใช้พอร์ตซ้ำไหม

อาจมีโปรแกรมอื่นจองพอร์ต 8080 หรือ 8081 ไว้ ลองเปลี่ยนพอร์ตใน docker-compose.yml เป็น:



แล้วรันใหม่:



เข้าเว็บที่ http://localhost:8888

🔽 วิธีแก้ปัญหา: สร้าง environment ใหม่ทั้งหมด โดยไม่กระทบข้อมูลเดิม

ขั้นตอน 1: หยุด container เก่า (mysql:5.7 ที่ใช้ชื่อ shop-db)

bash

docker stop shop-db

docker rename shop-db shop-db-old

🔽 การ rename จะช่วย เก็บ container เดิมไว้ โดยไม่ลบทิ้ง เผื่อข้อมูลสำคัญในฐานข้อมูล

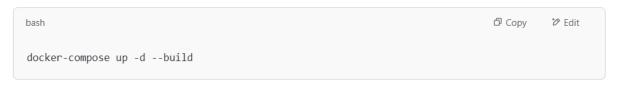
ขั้นตอน 2: สร้างระบบใหม่ของคุณ (mydocker-shop-new)

สมมุติว่าในโฟลเดอร์ mydocker-shop-new ของคุณมี:

- Dockerfile
- docker-compose.yml
- โฟลเดอร์ www (ที่มีไฟล์เว็บไซต์ของคุณ)

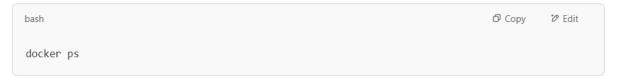


จากในโฟลเดอร์นั้น ให้รันคำสั่งนี้:



ขั้นตอน 3: ตรวจสอบว่า container ใหม่ขึ้นครบไหม

รันคำสั่ง:



ควรเห็น:

- shop-web (PORTS: 0.0.0.0:8080->80/tcp)
- shop-pma (PORTS: 0.0.0.0:8081->80/tcp)
- shop-db (ตัวใหม่จาก mariadb:10.5)



ขั้นตอน 4: ทดสอบเข้าใช้งาน

- เว็บไซต์หลัก: http://localhost:8080
- phpMyAdmin: http://localhost:8081
 វិឃ័ user: shopuser / pass: shoppass

ในไฟล์ docker-compose ที่คุณสร้างขึ้น บริการ MariaDB ถูกตั้งชื่อว่า mariadb (และ PMA_HOST ก็ใช้ "mariadb") ดัง นั้นในไฟล์ db_connect.php คุณควรเปลี่ยนค่า host จาก "db" เป็น "mariadb"

ตัวอย่างการแก้ไขใน db_connect.php:

```
php

$host = "mariadb"; // เปลี่ยนจาก "db" เป็น "mariadb"

$user = "shopuser";

$pass = "shoppass";

$db = "shop_db";

$conn = new mysqli($host, $user, $pass, $db);

if ($conn->connect_error) {
    die("เชื่อมต่อฐานข้อมูลไม่สำเร็จ: " . $conn->connect_error);
}

}
```

หลังจากแก้ไขแล้ว ให้ rebuild container และทดสอบใหม่อีกครั้ง

```
โดยสั่ง:
```

```
bash

docker-compose up -d --build

Docker-compose up -d --build
```

ต่อไปคุณสามารถทำอะไรได้บ้าง?

🔽 1. พัฒนาเว็บไซต์

- เพิ่มไฟล์ .php , .html , CSS, JS ลงในโฟลเดอร์ www/
- ปรับปรุงระบบ login/register, ระบบตะกร้าสินค้า, ระบบแอดมิน ฯลฯ

🔽 2. จัดการฐานข้อมูล

- เข้าผ่าน phpMyAdmin
- นำเข้าหรือสำรองข้อมูล .sql

🔽 3. สำรอง/นำเข้าฐานข้อมูล

- สำรองฐานข้อมูล: ผ่าน phpMyAdmin หรือ docker exec ก็ได้
- นำเข้า: ใช้ phpMyAdmin หรือคำสั่ง:

```
bash

docker exec -i shop-db mysql -ushopuser -pshoppass shop_db < /path/to/your.sql
```

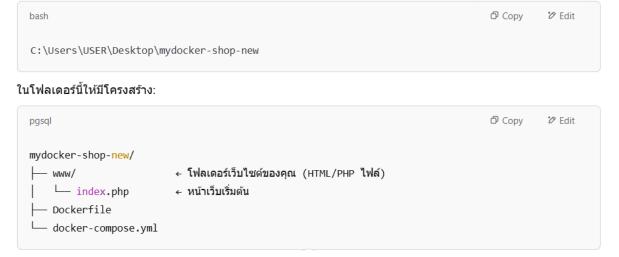
(ต้อง copy ไฟล์ .sql เข้า container ก่อน หรือ mount volume ให้ถูก)

🔽 4. เชื่อมต่อระบบกับเว็บไซต์

• ให้ไฟล์ PHP เชื่อมต่อกับฐานข้อมูลด้วย host mariadb และ user/pass ที่ตั้งไว้

🔽 ขั้นตอนที่ 1: โครงสร้างโฟลเดอร์

สร้างโฟลเดอร์โปรเจกต์:



🔽 ขั้นตอนที่ 2: ไฟล์ index.php (ในโฟลเดอร์ www)

php php Copy bedit cho "<h1>ยินดีต้อนรับสู่ mydocker-shop-new</h1>"; }>

☑ ขั้นตอนที่ 3: สร้าง Dockerfile

Dockerfile

Dockerfile

FROM php:8.1-apache

RUN docker-php-ext-install mysqli

RUN a2enmod rewrite

☑ ขั้นตอนที่ 4: เขียน docker-compose.yml

docker-compose.yml

```
	☐ Copy

                                                                                            ℃ Edit
yaml
services:
  web:
    build:
      context: .
      dockerfile: Dockerfile
    container_name: myshop-web
    volumes:
      - ./www:/var/www/html
    ports:
      - "8080:80"
    depends_on:
      - mariadb
  mariadb:
    image: mariadb:10.5
                                                \downarrow
    container_name: myshop-db
  mariadb:
    image: mariadb:10.5
    container name: myshop-db
    environment:
     MARIADB_ROOT_PASSWORD: rootpass
     MARIADB_DATABASE: shop_db
     MARIADB_USER: shopuser
     MARIADB_PASSWORD: shoppass
    ports:
      - "3306:3306"
    volumes:
      - db_data:/var/lib/mysql
  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: myshop-pma
    environment:
      PMA_HOST: mariadb
     PMA_PORT: 3306
      PMA_USER: shopuser
     PMA_PASSWORD: shoppass
    ports:
      - "8081:80"
    depends_on:
      - mariadb
volumes:
  db_data:
```

🔽 ขั้นตอนที่ 5: สั่งรันโปรเจกต์

เปิด Terminal ในโฟลเดอร์ mydocker-shop-new แล้วรัน:

bash

docker-compose up -d --build

docker-compose up -d --build

🔽 ขั้นตอนที่ 6: เข้าดูผลลัพธ์

- เข้าเว็บ: http://localhost:8080
- เข้า phpMyAdmin: http://localhost:8081

Username: shopuser
Password: shoppass

หากมีไฟล์ .sql สำหรับฐานข้อมูลเดิม สามารถนำเข้าได้ผ่าน phpMyAdmin หรือใช้คำสั่ง docker exec เพื่อเข้า MariaDB โดยตรงก็ได้ครับ

🔽 รายละเอียดของแต่ละ service

1. web

- ใช้ php:8.1-apache (จาก Dockerfile)
- เปิดพอร์ต 8080 → ใช้งานเว็บไซต์ผ่าน http://localhost:8080
- เชื่อมกับฐานข้อมูล mariadb

mariadb

- ใช้ mariadb:10.5
- สร้างฐานข้อมูลชื่อ shop_db
- ชื่อผู้ใช้: shopuser , รหัสผ่าน: shoppass
- พอร์ต 3306 ใช้งานภายในระบบ

3. phpmyadmin

- ใช้ phpmyadmin/phpmyadmin
- เปิดใช้งานบนพอร์ต 8081 → เข้าจาก http://localhost:8081
- เชื่อมกับ mariadb ด้วย user shopuser

🔽 การใช้งาน

สิ่งที่คุณต้องการ	เข้าใช้งานที่ไหน	ð
เว็บไซด์	http://localhost:8080	
phpMyAdmin	http://localhost:8081	
เข้าฐานข้อมูล	user: shopuser / pass: shoppass	

🔽 คำสั่งรัน

ให้เปิดเทอร์มินัลที่โฟลเดอร์ mydocker-shop-new แล้วรัน:

