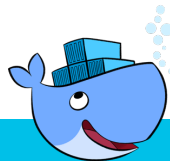


Tự động hóa triển khai Container 🤖



Docker Compose

- Vì sao cần tự động hóa?
 - Môi trường có nhiều container
 - Chuẩn hóa môi trường
 - Mô tả lại toàn hệ thống trong 1 file
- Sử dụng:
 - File mô tả bằng YAML
 - CLI `docker-compose`

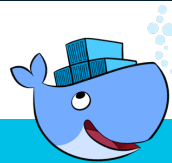


docker-compose.yml

- Có nhiều version 1, 2, 2.1, 3, 3.1... Khuyến nghị sử dụng 3.x trở lên
- Sử dụng cùng CLI `docker-compose`
- Sử dụng trong môi trường cluster (`docker swarm`)
- Đặt tên `docker-compose.yml`. Muốn sử dụng tên khác thì cần dùng option

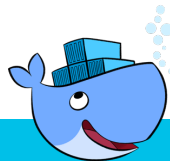
`docker-compose -f filename`

Compose file format	Docker Engine release
Compose specification	19.03.0+
3.8	19.03.0+
3.7	18.06.0+
3.6	18.02.0+
3.5	17.12.0+
3.4	17.09.0+
3.3	17.06.0+
3.2	17.04.0+
3.1	1.13.1+
3.0	1.13.0+
2.4	17.12.0+
2.3	17.06.0+
2.2	1.13.0+
2.1	1.12.0+
2.0	1.10.0+



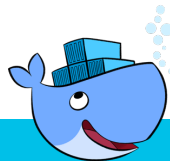
docker-compose CLI

- Chủ yếu dùng cho môi trường Test/Dev. Môi trường prod thường dùng stack trong docker swarm.
- 2 câu lệnh thường dùng:
 - `docker-compose up` → set up network, volume và bật containers
 - `docker-compose down` → tắt hết container và xóa network, volume



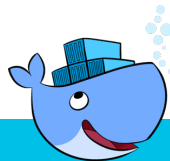
Bài tập 7 – Soạn một docker-compose file

- Xây dựng một compose file để chạy ứng dụng Drupal
- Sử dụng image `drupal` cho app và `postgres` cho database
- Cấu hình để truy cập được drupal qua `http://localhost:8080`
- Lưu ý dùng biến môi trường `POSTGRES_PASSWORD`
- Dùng volume để lưu trữ dữ liệu của Drupal
- Cài đặt nốt Drupal trên giao diện web. Lưu ý mặc định Drupal sẽ kết nối đến database tên `localhost`, cần sửa thành tên của db service



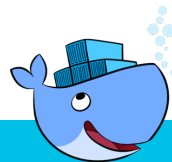
Dùng docker compose để build image

- `docker-compose up` **sẽ build** nếu không tìm thấy image trong cache
- Có thể kết hợp `docker-compose up --build`



Bài tập 8 – Build và Run Compose

- Build một custom image của Drupal
- Dùng compose file từ Bài tập 7
- Tạo `Dockerfile` và `docker-compose.yml` trong cùng thư mục



Bài tập 8 – Build và Run Compose

● Dockerfile

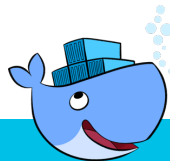
- Build Dockerfile từ base `drupal:8.8.2`
- Sau đó RUN apt package manager command để cài đặt git: `apt-get update && apt-get install -y git`
- Clean up apt install với lệnh `rm -rf /var/lib/apt/lists/*`, dùng `\` và `&&` để kết nối nhiều lệnh sheell.
- Chuyển thư mục làm việc về `/var/www/html/themes`
- Sử dụng git clone để copy code về thư mục theme: `RUN git clone --branch 8.x-3.x --single-branch --depth 1 https://git.drupalcode.org/project/bootstrap.git`
- Change owner về đúng người dùng cần thiết để chạy ứng dụng: `chown -R www-data:www-data bootstrap.`
- Chuyển thư mục làm việc về thư mục `/var/www/html`

● Compose File

- Đặt tên image tag là `custom-drupal`
- Sử dụng Dockerfile trong cùng thư mục làm việc hiện tại build: `.` cho drupal.
- Sử dụng `postgres:12.1` cho postgres service, lưu ý phần environment và volume. Volume cần thiết cho DB để bảo toàn dữ liệu khi compose down.

● Compose Down

- Dùng `docker-compose down -v` và `--rm` để xóa hết image và volume



Thank you

