**Phần 18 Docker Compose**

Docker Compose sẽ thực thi ứng dụng đa container dựa trên YAML. Tệp YAML bao gồm tất cả các cấu hình cần thiết để triển khai container Docker Compose, được tích hợp với Docker Swarm và cung cấp hướng dẫn để xây dựng và triển khai container. Với Docker Compose, mỗi container được xây dựng để chạy trên một máy chủ duy nhất.

**Cài đặt Docker Compose**

Chúng ta có thể chạy Docker Compose trên hệ điều hành MacOs, Widows và Linux 64-bit.

* Đối với bất kỳ hoạt động quan trọng nào, Docker Compose đều phụ thuộc vào Docker Engine. Tùy thuộc vào sự sắp xếp của bạn, chúng tôi phải đảm bảo Docker Engine được cài đặt cục bộ hoặc từ xa.
* Hệ thống máy tính để bàn như Docker dành cho Mac và Windows được cài đặt sẵn Docker Compose.
* Trước tiên, hãy cài đặt Docker theo hướng dẫn trong phần Cài đặt Docker trên hệ thống Linux trước khi bắt đầu cài đặt Docker Compose.

**Tại sao nên dùng Docker Compose?**

Như đã thảo luận trước đó, một ứng dụng thực tế có một container riêng cho mỗi dịch vụ của nó. Và chúng ta biết rằng mỗi container cần có một Dockerfile. Điều đó có nghĩa là chúng ta sẽ phải viết có thể hàng trăm tệp docker và sau đó quản lý mọi thứ về các container riêng lẻ,

Do đó, chúng tôi sử dụng docker-compose, một công cụ hỗ trợ định nghĩa và chạy các ứng dụng đa container. Với sự trợ giúp của Docker Compose, bạn có thể bắt đầu và dừng các dịch vụ bằng cách sử dụng tệp YAML của nó. Docker-compose cho phép chúng tôi bắt đầu và dừng tất cả các dịch vụ chỉ bằng một vài lệnh đơn giản và một tệp YAML duy nhất cho mỗi cấu hình.

Ngược lại với việc sử dụng hình ảnh dựng sẵn từ Docker Hub, mà bạn có thể cấu hình bằng tệp docker-compose.yaml, nếu bạn đang sử dụng hình ảnh tùy chỉnh, bạn sẽ cần khai báo cấu hình của nó trong một Dockerfile riêng. Đây là các tính năng mà docker-compose hỗ trợ:

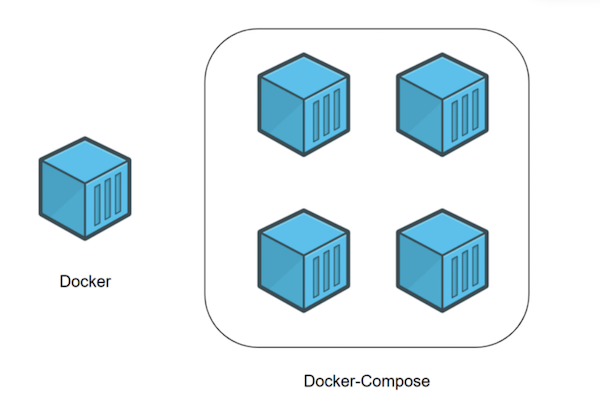
* Tất cả các dịch vụ đều được chạy riêng biệt trên một máy chủ duy nhất.
* Các container chỉ được tạo lại khi có sự thay đổi.
* Dữ liệu ổ đĩa không bị thiết lập lại khi tạo vùng chứa mới, ổ đĩa vẫn được bảo toàn.
* Sự di chuyển của các biến số và thành phần trong môi trường.
* Nó tạo ra một mạng ảo để dễ dàng tương tác trong các môi trường.

**Các lệnh cơ bản trong Docker Compose**

* **docker-compose up**: Khởi động các container
* **docker-compose down**: Dừng và xóa các container
* **docker-compose ps**: Hiển thị trạng thái của các container
* **docker-compose build**: Tạo image từ Dockerfile trong mỗi dịch vụ
* **docker-compose restart**: Khởi động lại các container
* **docker-compose stop**: Dừng các container
* **docker-compose rm**: Xóa các container không sử dụng
* **docker-compose logs**: Hiển thị các logs của các container
* **docker-compose config**: Hiển thị các cấu hình của Docker Compose
* **docker-compose exec**: Thực thi một lệnh trên một container
* **docker-compose port**: Hiển thị các port của các container
* **docker-compose top**: Hiển thị các process đang chạy trong các container

Lưu ý: Các lệnh trên phải được thực hiện trong thư mục chứa file docker-compose.yml.

**Cài đặt Docker Compose**

Cài đặt Docker Compose

Theo từng hệ điều hành sẽ có bước cài đặt Docker Compose khác nhau. Bạn nên tải phiên bản mới nhất của Docker Compose từ trang web chính thức của [Docker Compose](https://docs.docker.com/compose/install/?ref=200lab.io).

**Cài đặt Docker Compose trên macOS**

Để sử dụng Docker Compose trên macOS, bạn chỉ cần cài đặt Docker Desktop cho Mac và không cần cài đặt riêng Docker Compose.

**Cài đặt Docker Compose trên Windows**

Để sử dụng Docker Compose trên Windows, bạn chỉ cần cài đặt Docker Desktop cho Windows và không cần cài đặt riêng Docker Compose.

**Cài đặt Docker Compose trên Linux**

* Hãy cấp quyền thực thi cho tệp tin Docker Compose binary bằng lệnh sau:

BASH

copy

sudo chmod +x /path/to/docker-compose

Trong đó, "/path/to/docker-compose" là đường dẫn tới tệp tin Docker Compose binary.

* Di chuyển tệp tin Docker Compose binary vào một trong các đường dẫn $PATH trên hệ thống của bạn để có thể truy cập nó từ bất kỳ thư mục nào.

Ví dụ, để di chuyển tệp tin Docker Compose binary vào thư mục /usr/local/bin trên Linux, bạn có thể sử dụng lệnh sau:

BASH

copy

sudo mv /path/to/docker-compose /usr/local/bin/docker-compose

* Kiểm tra xem Docker Compose đã được cài đặt thành công bằng cách chạy lệnh sau:

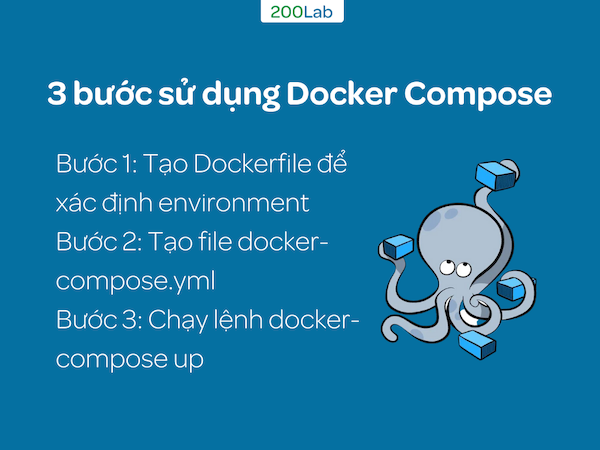
CSS

copy

docker-compose --version

Nếu thành công, lệnh này sẽ hiển thị phiên bản của Docker Compose.

**Sử dụng Docker Compose**

Sử dụng Docker Compose

Để sử dụng Docker Compose, bạn cần thực hiện các bước sau:

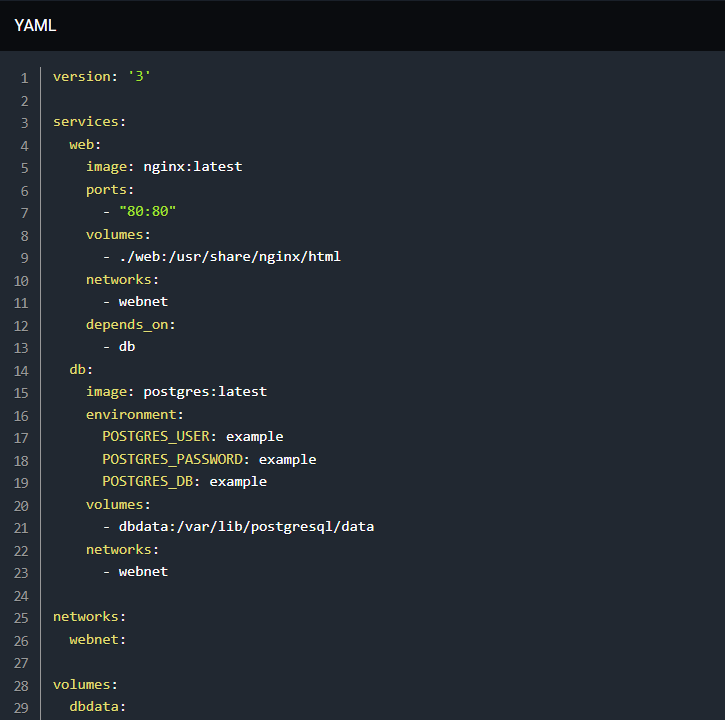
**Bước 1: Cài đặt Docker Compose**

Bạn cần cài đặt Docker Compose trên máy tính của mình. Bạn có thể tải từ trang chủ của Docker hoặc có thể cài đặt thông qua các gói phần mềm của hệ điều hành.

Tiếp đến, bạn sẽ tạo Dockerfile để xác định environment. Mỗi dịch vụ sẽ chạy trên một container riêng và sử dụng image tương ứng.

**Bước 2: Tạo file docker-compose.yml**

Bạn cần tạo một file docker-compose.yml để định nghĩa các container và cấu hình của chúng.



Ý nghĩa của các giá trị trong file docker-compose:

* **version**: phiên bản của Docker Compose file. Ở đây, chúng ta sử dụng phiên bản 3.
* **services**: là khu vực khai báo các services cần thiết cho ứng dụng.
* **web**: dịch vụ web, sử dụng image nginx, chia sẻ volume và network với dịch vụ db. Cổng 80 được định nghĩa để web có thể truy cập được từ bên ngoài.
* **db**: dịch vụ db, sử dụng image postgresql, chia sẻ volume và network với dịch vụ web. Các biến môi trường cài đặt cho dịch vụ postgresql được định nghĩa ở phần environment.
* **networks**: danh sách các networks được sử dụng cho container.
* **webnet**: mạng webnet để chia sẻ giữa dịch vụ web và db.
* **volumes**: là option nên config, volumes cho phép mount data từ container ra máy local. Khi config option này thì mỗi lần stop container data của container đó sẽ không bị mất đi.
* **dbdata**: là container chứa thông tin về database.

Trong ví dụ này, Docker Compose sẽ khởi tạo 2 container, một container sử dụng image nginx và một container sử dụng image postgresql. Container sử dụng image nginx sẽ được kết nối với container sử dụng image postgresql thông qua mạng webnet. Sau đó, chúng sẽ chia sẻ volume dbdata để lưu trữ dữ liệu của postgresql.

**Bước 3: Chạy lệnh docker-compose up**

Sử dụng lệnh **docker-compose up**để khởi động các container được định nghĩa trong file docker-compose.yml. Nếu các image không được tải xuống trước đó, Docker Compose sẽ tự động tải chúng xuống và khởi động các container.

**Quản lý các container**

Bạn có thể quản lý các container bằng các lệnh Docker Compose như docker compose stop để dừng các container, docker compose start khởi động sau khi dừng các container, docker compose restart để khởi động lại các container đã dừng, docker compose -f docker.yaml down để xóa các container đã dừng, docker-compose ps để hiển thị trạng thái của các container, ...

**Tùy chỉnh cấu hình**

Nếu bạn muốn thay đổi cấu hình của các container, bạn chỉ cần chỉnh sửa **file docker-compose.yml**và chạy lại lệnh docker-compose up.

Lưu ý: các lệnh Docker Compose phải được thực hiện trong thư mục chứa file docker-compose.yml.