



# Docker Cheat sheet

## Cài đặt

Cài đặt **Docker Engine** (<https://docs.docker.com/engine/install>) (chỉ dành cho Linux) hoặc **Docker Desktop** (<https://docs.docker.com/desktop>) (Linux, macOS và Windows).

## Lệnh Container

Command	Mô tả
<code>docker run [image]</code>	Tạo và chạy một container mới
<code>docker run -p 8080:80 [image]</code>	Công khai cổng 80 của container ra cổng 8080 của máy chủ
<code>docker run -d [image]</code>	Chạy một container ở chế độ nền
<code>docker run -v [host]:[container] [image]</code>	Gắn một thư mục từ máy chủ vào container
<code>docker ps</code>	Liệt kê các container đang chạy
<code>docker ps --all</code>	Liệt kê tất cả container (đang chạy hoặc đã dừng)
<code>docker logs [container_name]</code>	Lấy nhật ký của một container
<code>docker logs -f [container_name]</code>	Lấy và theo dõi nhật ký của một container
<code>docker stop [container_name]</code>	Dừng một container đang chạy
<code>docker start [container_name]</code>	Khởi động lại một container đã dừng
<code>docker rm [container_name]</code>	Xóa một container

## Thực thi lệnh trong Container

Command	Mô tả
<code>docker exec [container_name] [command]</code>	Thực thi một lệnh trong container đang chạy
<code>docker exec -it [container_name] bash</code>	Mở một shell trong container đang chạy

## Lệnh Image

Command	Mô tả
<code>docker build -t [image] .</code>	Xây dựng một image từ Dockerfile
<code>docker images</code>	Liệt kê tất cả các image
<code>docker rmi [image]</code>	Xóa một image

## Lệnh Registry

Command	Mô tả
---------	-------

<code>docker login</code>	Đăng nhập vào một registry Docker
<code>docker login [server]</code>	Đăng nhập vào một registry Docker cụ thể
<code>docker logout</code>	Đăng xuất khỏi một registry Docker
<code>docker logout [server]</code>	Đăng xuất khỏi một registry Docker cụ thể
<code>docker push [image]</code>	Đẩy một image lên registry
<code>docker pull [image]</code>	Tải một image từ registry
<code>docker search [image]</code>	Tìm kiếm một image trong registry

## Lệnh Hệ thống

Command	Mô tả
<code>docker system df</code>	Hiển thị mức sử dụng đĩa của Docker
<code>docker system prune</code>	Xóa dữ liệu không sử dụng
<code>docker system prune -a</code>	Xóa toàn bộ dữ liệu không sử dụng

## Lệnh Docker Compose

Command	Mô tả
<code>docker compose up</code>	Tạo và khởi động các container
<code>docker compose up -d</code>	Tạo và khởi động các container ở chế độ nền
<code>docker compose up --build</code>	Xây dựng image và khởi động các container
<code>docker compose stop</code>	Dừng các container
<code>docker compose down</code>	Dừng và xóa các container
<code>docker compose ps</code>	Liệt kê các container
<code>docker compose logs</code>	Xem nhật ký cho tất cả các dịch vụ
<code>docker compose logs [service]</code>	Xem nhật ký cho một dịch vụ cụ thể
<code>docker compose logs -f</code>	Theo dõi nhật ký cho tất cả các dịch vụ
<code>docker compose pull</code>	Tải image của các dịch vụ
<code>docker compose build</code>	Xây dựng image của các dịch vụ
<code>docker compose build --pull</code>	Xây dựng image của các dịch vụ với việc tải image cơ sở mới nhất

## Hướng dẫn Dockerfile

Instruction	Mô tả
<code>FROM [image]</code>	Đặt image cơ sở
<code>FROM [image] AS [name]</code>	Đặt image cơ sở và đặt tên cho giai đoạn xây dựng
<code>RUN [command]</code>	Thực thi một lệnh trong quá trình xây dựng
<code>RUN ["exec", "param1", "param2"]</code>	Thực thi một lệnh trong quá trình xây dựng
<code>CMD ["exec", "param1", "param2"]</code>	Thực thi một lệnh khi container khởi động

ENTRYPOINT ["exec", "param1"]	Cấu hình container để chạy như một tệp thực thi
ENV [key]=[value]	Đặt một biến môi trường
EXPOSE [port]	Công khai một cổng
COPY [src] [dest]	Sao chép tệp từ nguồn đến đích
COPY --from=[name] [src] [dest]	Sao chép tệp từ một giai đoạn xây dựng đến đích
WORKDIR [path]	Đặt thư mục làm việc
VOLUME [path]	Tạo một điểm gắn kết
USER [user]	Đặt người dùng
ARG [name]	Định nghĩa một đối số xây dựng
ARG [name]=[default]	Định nghĩa một đối số xây dựng với giá trị mặc định
LABEL [key]=[value]	Đặt một nhãn siêu dữ liệu
HEALTHCHECK [command]	Đặt một lệnh kiểm tra sức khỏe

Để xem tài liệu tham khảo đầy đủ về Dockerfile, truy cập: <https://docs.docker.com/engine/reference/builder/>.

## Khóa Docker Compose

Key	Mô tả
name	Đặt tên cho dự án
services	Danh sách các dịch vụ được định nghĩa trong tệp
services.[name].image	Đặt image để sử dụng hoặc xây dựng
services.[name].build	Bối cảnh và tùy chọn xây dựng
services.[name].build.context	Bối cảnh xây dựng (mặc định là thư mục hiện tại)
services.[name].build.dockerfile	Dockerfile để sử dụng (mặc định là Dockerfile)
services.[name].build.target	Giai đoạn xây dựng để sử dụng
services.[name].build.args	Các đối số xây dựng
services.[name].command	Ghi đè lệnh mặc định cho container
services.[name].entrypoint	Ghi đè điểm vào mặc định cho container
services.[name].volumes	Gắn các volume vào container
services.[name].ports	Công khai các cổng của container ra máy chủ
services.[name].environment	Đặt các biến môi trường trong container
services.[name].restart	Chính sách khởi động lại (no/always/on-failure/unless-stopped)
services.[name].scale	Đặt số lượng container để chạy
services.[name].networks	Danh sách các mạng để kết nối container
services.[name].depends_on	Danh sách các dịch vụ cần khởi động trước dịch vụ này
services.[name].labels	Đặt các nhãn siêu dữ liệu cho container

<code>networks</code>	Danh sách các mạng được định nghĩa trong tệp
<code>networks.[name].driver</code>	Đặt trình điều khiển mạng
<code>networks.[name].external</code>	Không tạo mạng, sử dụng mạng hiện có
<code>volumes</code>	Danh sách các volume được định nghĩa trong tệp
<code>volumes.[name].name</code>	Đặt tên cho volume
<code>volumes.[name].driver</code>	Đặt trình điều khiển volume
<code>configs</code>	Danh sách các cấu hình được định nghĩa trong tệp
<code>secrets</code>	Danh sách các bí mật được định nghĩa trong tệp

## Ví dụ compose.yaml

Ghi chú: Tệp này trước đây được gọi là `docker-compose.yaml`, nhưng hiện tại `compose.yaml` được ưu tiên hơn.

```
services:
  service1:
    image: [image]
    build:
      context: .
    volumes:
      - ./code:ro
    ports:
      - "8000:80"
    environment:
      KEY: value
```

Xem trực tiếp tại [devops.vn](#)