

# Docker Compose

Docker Compose is an **additional tool**, offered by the Docker ecosystem, which helps with **orchestration / management** of multiple Containers. It can also be used for single Containers to simplify building and launching.

## Why?

Consider this example:

```
docker network create shop

docker build -t shop-node .

docker run -v logs:/app/logs --network shop --name shope-web shop-node

docker build -t shop-database

docker run -v data:/data/db --network shop --name shop-db shop-database
```

This is a very simple (made-up) example - yet you got **quite a lot of commands** to execute and memorize to bring up all Containers required by this application.

And you have to run (most of) these commands whenever you change something in your code or you need to bring up your Containers again for some other reason.

With Docker Compose, this gets much easier.

You can put your Container configuration into a `docker-compose.yaml` file and then use just one command to bring up the entire environment: `docker-compose up`.

## Docker Compose Files

A `docker-compose.yaml` file looks like this:

```
version: "3.8" # version of the Docker Compose spec which is being used

services: # "Services" are in the end the Containers that your app needs
  web:
    build: # Define the path to your Dockerfile for the image of this container
      context: .
      dockerfile: Dockerfile-web
    volumes: # Define any required volumes / bind mounts
      - logs:/app/logs
  db:
```

```
build:
  context: ./db
  dockerfile: Dockerfile-web
volumes:
  - data:/data/db
```

You can conveniently edit this file at any time and you just have a short, simple command which you can use to bring up your Containers:

```
docker-compose up
```

You can find the full (possibly intimidating - you'll only need a small set of the available options though) **list of configurations** here: <https://docs.docker.com/compose/compose-file/>

**Important to keep in mind:** When using Docker Compose, you **automatically get a Network for all your Containers** - so you don't need to add your own Network unless you need multiple Networks!

## Docker Compose Key Commands

There are two key commands:

- `docker-compose up`: **Start all containers** / services mentioned in the Docker Compose file
  - `-d`: Start in **detached mode**
  - `--build`: Force Docker Compose to re-evaluate / rebuild all images (otherwise, it only does that if an image is missing)
- `docker-compose down`: **Stop and remove** all containers / services
  - `-v`: **Remove all Volumes** used for the Containers - otherwise they stay around, even if the Containers are removed

Of course, there are **more commands**. You'll see more commands in other course sections (e.g. the "**Utility Containers**" and "**Laravel Demo**" sections) but you can of course also already dive into the official command reference: <https://docs.docker.com/compose/reference/>

`docker compose build`: Only builds the image and does not start the container afterwards.

`docker compose up <service_1> <service_2> ...`: Bring up the certain services.

`docker compose run <option> <service> <command>`: Run a one-time command against a service in the docker compose file.

`docker exec <option> <container> <command>`: Execute a command in a running container (the container should be started first). Note that a docker compose file isn't required here as this is a docker command and not a docker compose command.