# Using Docker compose

In this lesson we will learn about Docker Compose and how it can help us simplify the way we run our infrastructure services.

## Do I need yet another Docker tool?

We already started using Docker to run our MongoDB database, and we are doing this via a simple docker run command. However, starting with the next module we will need to start bringing in many other infrastructure services to support our microservices. For instance, in the next module we will use RabbitMQ as our message broker, and in future modules we will use Seq, Prometheus and Grafana to enable a series of observability related components.

So, if we keep just using docker run to execute our infrastructure services we will have to perform too many steps to setup them and we will need to remember too many arguments for each of them, like the right environment variables, ports and volumes. Also, some of our containers might need to talk to each other, like Grafana and Prometeus, and some of them might not be able to work at all without another container be up and running first, which again is the case of Grafana, which won't be able to do much without Prometeus being available first.

## What is Docker Compose

That's where Docker Compose come into play. Compose is a tool for defining and running multi-container Docker applications.

We will now be able to define all of our infrastructure services in a single file called the docker-compose yaml file. This will include the definition of the container to use in each case, environment variables, ports, volumes and even dependencies between them.

Then, we will execute a single docker-compose up command and all the containers will start in the right order and with the correct configurations. Also, compose provides a default network that all containers are joined to, in case they need to talk to each other.

So with docker compose we are able to document the way to configure all of our infrastructure services in a single file and we get to start all of them with just one line as opposed to multiple commands. Also, they all join a default network.

In the next lesson we will create our docker compose file and move our current MongoDB configuration there.