Lesson 3 : Containerization

The goal of the TME is to create docker config to launch an app in containers.

You need to install docker.

(And optionnally nodejs and postgreSQL to test the code outside docker)

Exercise 1 Containerize server

Try to launch the server : https://github.com/arthurescriou/crud

It is a basic crud server written in JS. There are instructions to run the code in the readme.

It needs a postgreSQL Database to work. (cf next exercise)

To configure the server all you have to do is to change values in the .env file.

Then write a Dockerfile to put this server in a docker container.

Then use docker build to build you image and run it with docker run.

Exercise 2 Containerize database

The server need a postgreSQL connection to work. We will put a postgreSQL database in a docker instance as well.

I built an example of what your postgreSQL Dockerfile looks like here https://github.com/arthurescriou/crud-database

Follow the readme to build it.

Exercise 3 Connect containers together

Now that you can run both server and database, you can run the whole application in docker instance.

Just make sure that the environment variables are correct.

Exercise 4 Use docker-compose

Our application uses several docker instances and we have to launch it with several commands.

Now we want to be able to launch it in one time.

For that we will use docker-compose.

To run a docker-compose up command we need to write a docker-compose.yml file.

Note

Some commands that could help you :

build an image (You have to run it in folder with a Dockerfile)

```
docker build -t imageName .
```

run an image

```
docker run imageName -d
```

stop docker instance

```
docker stop instanceID
```

get log of an instance

```
docker exec -it instanceID sh
```

list of all docker images on your device

```
docker images
```

list of all instances running on your device

```
docker ps
```

build and run a docker-compose stack (You have to run it in folder with a docker-compose.yml)

```
docker-compose up
```