

# MongoDB

Building DB without constrain without architecture. You built it fast

Document: json object (key: value)

Collection: correspond to a table in relational DB

## Installation - Docker

1) from the website or 2) Docker (the MongoDB instance running on Docker)

Mongo Express: GUI interface to interact with the DB

So in this course we choose to use the docker option so let's install Docker

I) install Docker

II) in the terminal check for its version `docker --version`

Docker has a `.yaml`  $\Rightarrow$  `docker-compose.yaml` configuration file for all its

dependencies, like `maven pom.xml`. Under services I have the dependencies.

In this case Mongo and Mongo Express

#) navigate to the folder where the `yaml` file is in

`trg` `cat docker-compose` it should <sup>show</sup> you the content of the file (It on mac)

-) `docker compose -f docker-compose.yaml up`

if I named the file `docker-compose`, I can do this  $\rightarrow$  `docker compose up` } is 2 files  
Docker  
mean part

this create ~~2~~<sup>x</sup> containers according to what you defined in the setting file

`up`: build the containers. If I want to remove them

`ctrl+c`: to stop the process/containers. `ctrl+z`: to have the line of command

$\rightarrow$  `docker compose down`: remove the containers

● `docker compose up -d`: detached mode

to stop the containers: `docker compose stop`

to start them, after stopping: `docker compose start`

`docker ps`: show you the containers

## Connect with the DB

1) with GUI client (Mongo Express, DataGrip, MongoDB Compass)

2) Mongo Shell (cmd)

A) find out the id of the container (`docker ps`)

b) `docker exec -it id bash`  $\Rightarrow$  now I am inside the DB but not really inside.

I need to enter the user, password etc.

● `mongo mongodb://localhost:27017 -u rootuser -p rootpassword`  

$\downarrow$   
port  
27017

$\downarrow$   
username  
rootuser

$\downarrow$   
password  
rootpassword

Deleting an item from an array of values

\$pull

letters: [x, y, z, a] I want to delete y  
key value

update ( { \_id: ... }, { \$pull: { letters: 'y' } } )

Adding an item to an array

\$push

the same but instead of pull I push

Deleting Docs

deleteOne()

every id is one document

• deleteOne ( { \_id: ... } )

• deleteOne() will delete a document without me having any control of which doc

deleteMany ( { sex: 'm' } ) delete all the documents who has sex is set to 'm'

deleteMany() : will delete all the docs in this collection