## Setup

Pull **mongo:latest** and **mongo-express:latest** images from Docker Hub:

Text

Description automatically generated

## Network

Simple diagram of what the network will look like:Graphical user interface, application

Description automatically generated with medium confidence

Check the networks that currently exists:

Text

Description automatically generated

Create a new network called **mongo-network** for MongoDB and Mongo Express containers to run in:



Confirm creation of the new network:

Text

Description automatically generated

## Creating Containers

Run the **mongo:latest** image in a container with the following options:

-d >> detached mode

-p >> map host port to container port

-e >> environment variables to create a new user (<https://hub.docker.com/_/mongo>) Graphical user interface, text, application

Description automatically generated

--name >> name of the container

--network >> specify which Docker network to run the container in

Text

Description automatically generated

Check logs to ensure creation was successful:

 \_\_



Run the **mongo-express:latest** image in a container with the following options:

-d >> detached mode

-p >> map host port to container port

-e >> environment variables to identify mongodb container as server and authenticate using credentials used for the database (<https://hub.docker.com/_/mongo-express>)

Table

Description automatically generated

--network >> specify which Docker network to run the container in

--name >> name of the container

Text

Description automatically generated

Check logs to ensure creation was successful:

Text

Description automatically generated

Head to <http://localhost:8081> to view the Mongo Express GUI connected to MongoDB:

Graphical user interface, text, application, email

Description automatically generated

Make changes to the default user profile:

A picture containing text, screenshot, mammal, dog

Description automatically generated

See the change reflected in the database:

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

## Volumes - Data Persistence

Data volume is used to persist data so information can be saved and retrieved even after a container is restarted:

Graphical user interface

Description automatically generated with low confidence

Graphical user interface

Description automatically generated

The best way to create volume is using named volumes:

Creating a variable to reference the folder that's automatically generated by Docker on the host machine for each container (/var/lib/docker/volumes) and mapping it to a location on the container

Graphical user interface, application, Word

Description automatically generated