

MongoDB

ONLINE SERVER

In this option, we run a MongoDB engine on Atlas, then we have a couple of options to connect to it. To begin with, you can watch the instructions in [this video](#) until 5:51. Up to that point, you have created an account on the official website, and set a cluster to run your database. Now, as Fig. 1 suggests, click on connect to see those options.

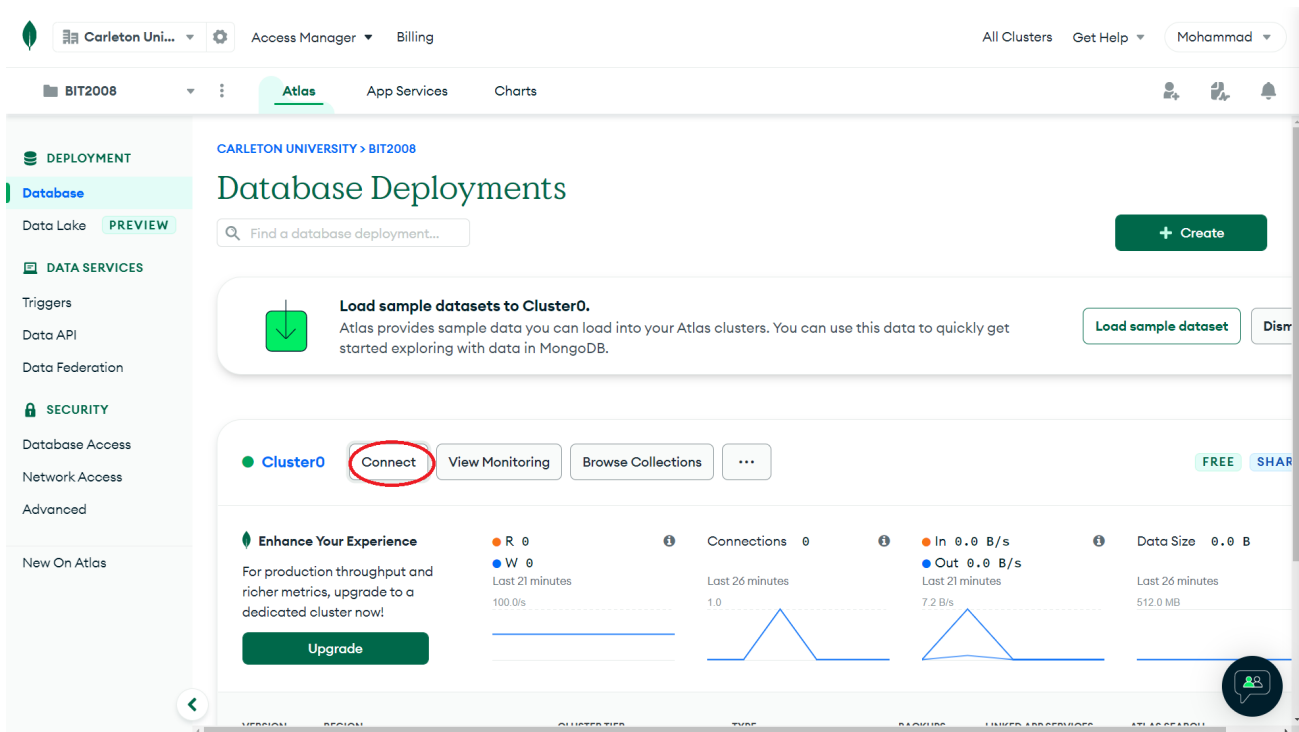


FIGURE 1. Trying to connect to online MongoDB engine.

Afterward, choose the first option as Fig. 2 demonstrates. Then, assuming that you do not have a MongoDB shell installed, you should click to download the version that is shown in the prompt (item 1). You do not have to go through the second item which is adding the file to your path. Instead, extract the file that you just downloaded into a folder of your choice, then open PowerShell to that folder, change your directory to find **mongosh** in *bin* folder, and run the third item with the replacements that are needed for the command to fit your scenario. Basically, the name of the database, and the username may vary from case to case. Also, the username and password that you need to use here, are the same as what you created under *Database Access*. Finally, you can type in your MongoDB commands inside this shell and it runs on your online cluster of yours.

Please feel free to follow the rest of the video as far as you wish to find a more convenient way.

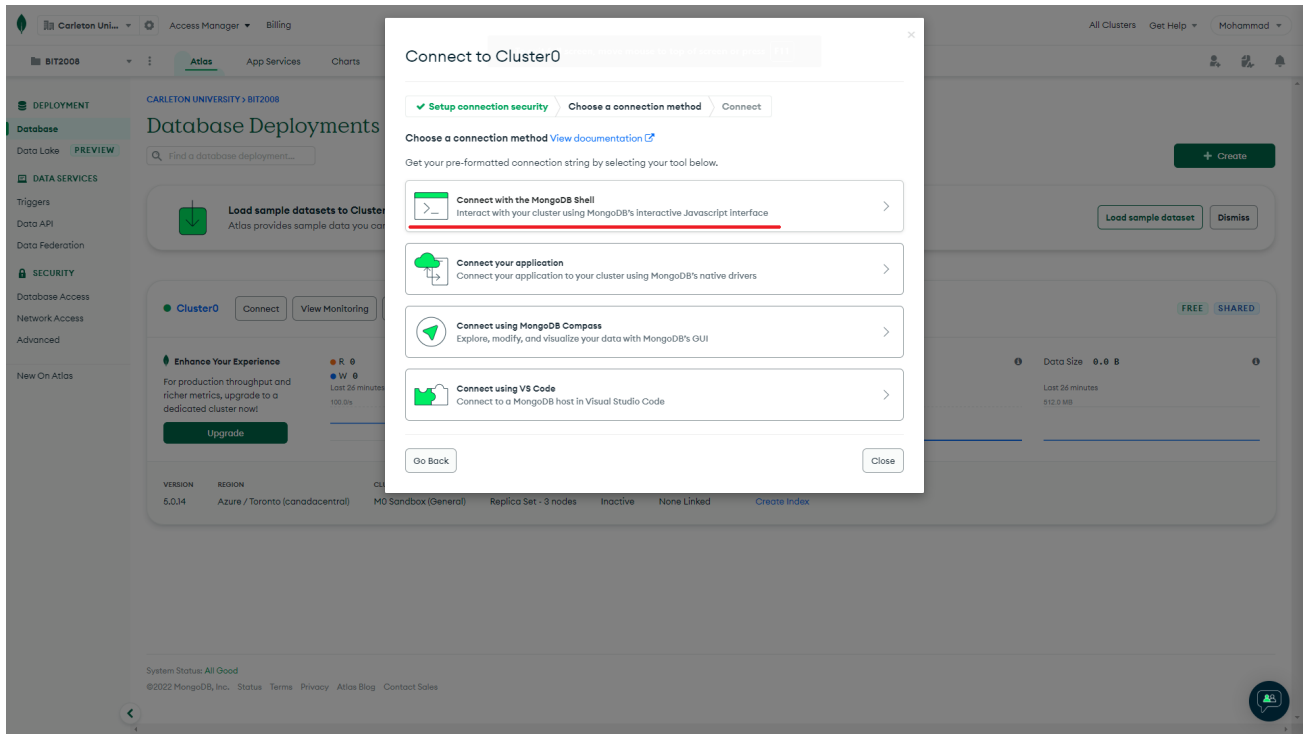


FIGURE 2. Choosing MongoDB shell option.

DOCKER

You can create your own Docker container to run MongoDB. First, you need to create a *docker-compose.yml* file in a new folder. As you know, this file contains the configurations of the containers. It should look like the code sheet below. The file is also shared with you on BrightSpace.

```

1 version: '3.8'
2
3 services:
4
5   mongo:
6     image: mongo
7     restart: unless-stopped
8     environment:
9       MONGO_INITDB_ROOT_USERNAME: root
10      MONGO_INITDB_ROOT_PASSWORD: example
11     volumes:
12       - ./data/db:/data/db
13       - ./:/code/
14     ports:
15       - 27017:27017
16
17   mongo-express:
18     image: mongo-express
19     restart: unless-stopped
20     ports:
21       - 8081:8081
22     environment:
23       ME_CONFIG_MONGODB_ADMINUSERNAME: root
24       ME_CONFIG_MONGODB_ADMINPASSWORD: example
25       ME_CONFIG_MONGODB_URL: mongodb://root:example@mongo:27017/

```

Then, you should open a Powershell with the same directory as the `docker-compose.yml` file, leave the docker desktop application open, and type

```
1 docker compose up
```

Then you have two options. First, there is *mongo-express* container that you can click on and choose **open with browser** or, like previous containers you can open a terminal to **mongo** container as Fig. 3 shows. For our lectures, we use the terminal mode and all that you need to do is execute the command below:

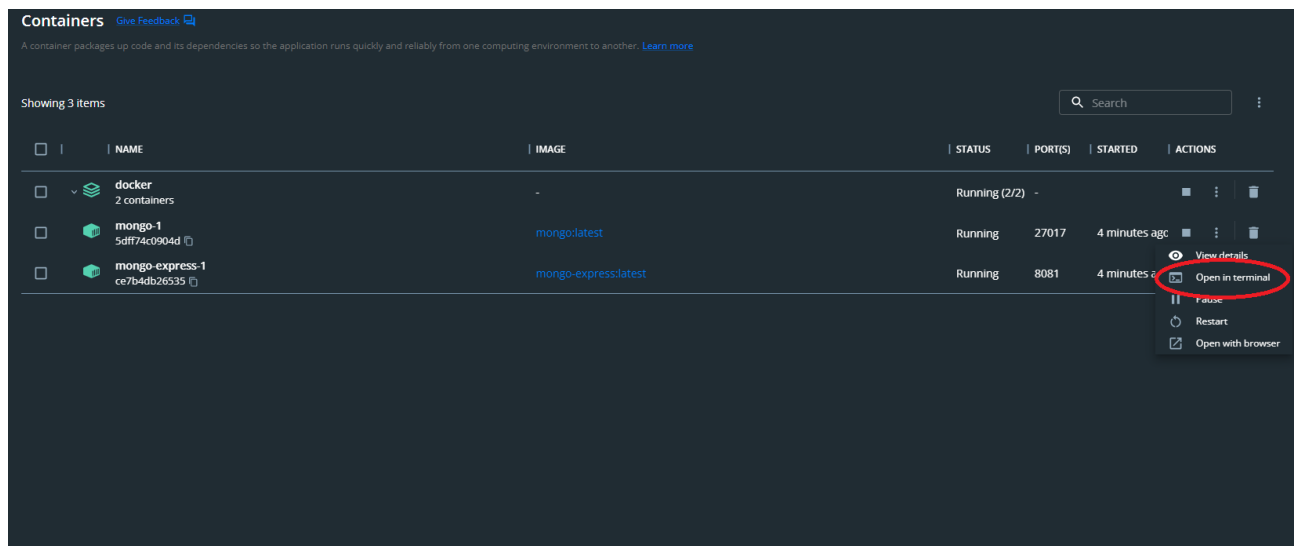


FIGURE 3. Opening a terminal to MongoDB container.

```
1 mongosh -u root -p example
```

The username and password can be modified via the *docker-compose.yml* file.

****** Please note that these instructions or any other instructions that are meant to set up our programming environment are **not** a part of course evaluation.