Самостоятельная работа 3

Создаём контейнер из официального образа с помощью команды docker run -d -p 127.0.0.1:4321:4321 --name mongo-exp-project mongo
Заходим в mongo и создаём базу данных и коллекцию с которой будем работать:

```
root@579905-nm0zg0v0y:/# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                 NAMES
f0d15757371c   mongo     "docker-entrypoint.s…"   24 minutes ago   Up 24 minutes   127.0.0.1:4321->4321/tcp, 27017/tcp   mongo-exp-project
root@579905-nm0zg0v0y:/# docker exec -it f0d15757371c mongo
MongoDB shell version v4.4.6
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("57c30d0f-ef55-4a71-88d3-3c3acca2b3e2") }
MongoDB server version: 4.4.6
---
The server generated these startup warnings when booting:
        2021-07-06T09:42:17.358+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/c
ore/prodnotes-filesystem
        2021-07-06T09:42:17.955+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
---
---
        Enable MongoDB's free cloud-based monitoring service, which will then receive and display
        metrics about your deployment (disk utilization, CPU, operation statistics, etc).

        The monitoring data will be available on a MongoDB website with a unique URL accessible to you
        and anyone you share the URL with. MongoDB may use this information to make product
        improvements and to suggest MongoDB products and deployment options to you.

        To enable free monitoring, run the following command: db.enableFreeMonitoring()
        To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
> show databases;
admin   0.000GB
config  0.000GB
local   0.000GB
> use newDatabases;
switched to db newDatabases
> show databases;
admin   0.000GB
config  0.000GB
local   0.000GB
> show collections;
> db.createCollection("users");
{ "ok" : 1 }
> db.users.insert({"name": "Admin", "age": 21});
WriteResult({ "nInserted" : 1 })
>
```

Узнаём с помощью команды ip-адрес для подключения к контейнеру: docker container inspect f0d15757371c


```
"Networks": {
    "bridge": {
        "IPAMConfig": null,
        "Links": null,
        "Aliases": null,
        "NetworkID": "46ee90a85e013ceea2690dea3d9a40bac2309f4d54b3b553a507ebf961ee2f58",
        "EndpointID": "2ce272430ea59d90d5ac0515e8d32d7b7838118edd23e23ee6d34b04d4540e6e",
        "Gateway": "172.17.0.1",
        "IPAddress": "172.17.0.2",
        "IPPrefixLen": 16,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "MacAddress": "02:42:ac:11:00:02",
        "DriverOpts": null
    }
}
```


После чего выходим из контейнера и создаём node.js проект с реализацией CRUD:

```
const mongoose = require("mongoose");
const Schema = mongoose.Schema;;
const url = "mongodb://172.17.0.2/4321";

const userSchema = new Schema({
        name: String,
        age: Number
});

mongoose.connect(url, { useUnifiedTopology: true, useNewUrlParser: true });

const User = mongoose.model("User", userSchema);

/* Вставка */
const user = new User({
        name: "newUser",
        age: 0
});

user.save(function (err, comment) {
        if (err) console.log(err);
        else console.log('Вставлен объект', user);
});


/* Вывод всех объектов */
User.find({}, (err, docs) => {
        if (err) console.log(err);
        else console.log(docs);
});


/* Удаление объекта */
User.deleteOne({name: "root"}, (err, result) => {
        if (err) console.log(err);
        else console.log("Объект удалён");
});


/* Обновление объекта */
User.updateOne({name: "Admin"}, {age: 22}, (err, result) => {
        if (err) console.log(err);
        else console.log("Объект обновлён");
});
```

Результаты:

```
root@579905-nm0zg0v0y:~/nodeProject# node .
[ { _id: 60e448f4ebb4cd0f9d723473, name: 'newUser', age: 0, __v: 0 } ]
Объект обновлён
Вставлен объект { _id: 60e4490cf454e10fadc72873, name: 'newUser', age: 0, __v: 0 }
Объект удалён
```