**Lab 08**

**Name : Shani Patel**

**Seneca ID :152243192**

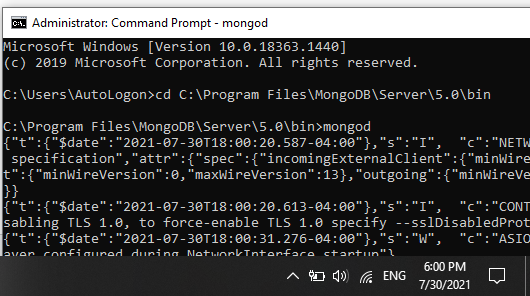
**Submission data : 30/07/2021**

Open your Windows command prompt and go the following directory where MongoDB is installed:

* cd C:\Program Files\MongoDB\Server\4.2\**bin**

To run MongoDB, execute ***mongod***

* mongod

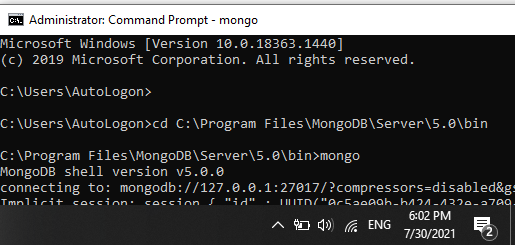


When MongoDB starts successfully, open another Windows command prompt and go the same *bin* directory:

* cd C:\Program Files\MongoDB\Server\4.2\**bin**

and execute ***mongo***

* mongo



Or you execute a batch file to start up MongoDB.

You will import products.json to the *inventory* database. To import data, go to the *bin* directory:

* cd C:\Program Files\MongoDB\Server\4.2\**bin**

Execute the following command:

* mongoimport --db inventory --collection products --file ..\dataset\products.json

To import the *json* file, provide the full path to the products.json. After executing the command, the data is imported to the *inventory* database. To make sure data is imported successfully, go to the MongoDB shell and execute the following command to see the imported documents:

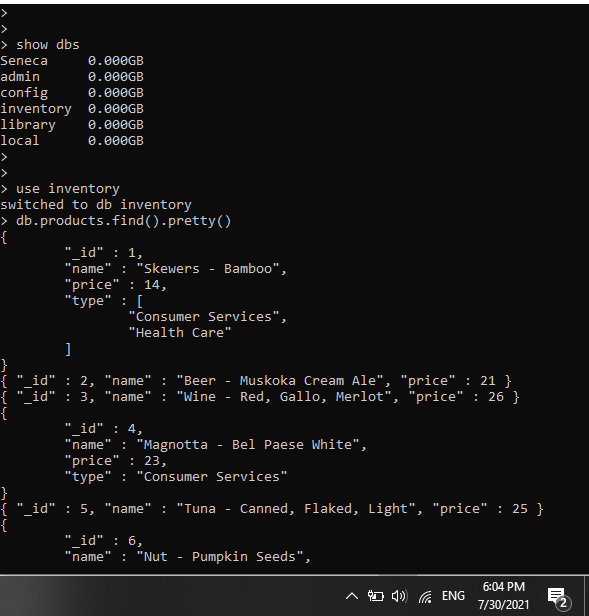
* show dbs

You should see the database inventory added to the list of your databases. To see the documents inside the database:

* use inventory
* db.products.find().forEach(printjson)

or

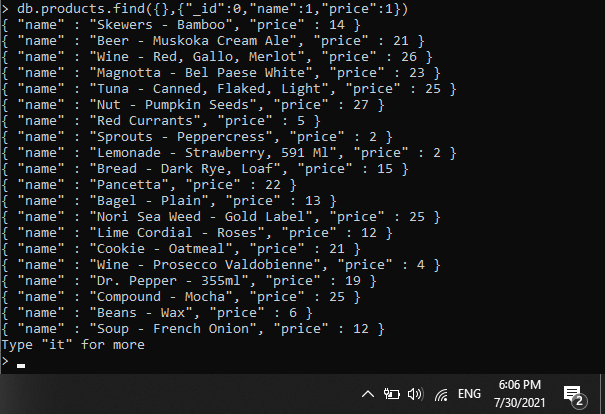
* db.products.find().pretty()



Extra questions

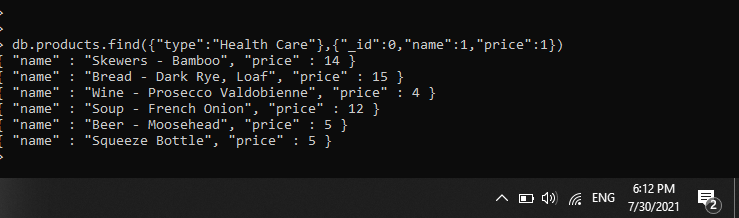
1. Write a query to return *name* and *price* of each product in the *inventory* database.

|  |
| --- |
| db.products.find({},{"\_id":0,"name":1,"price":1}) |



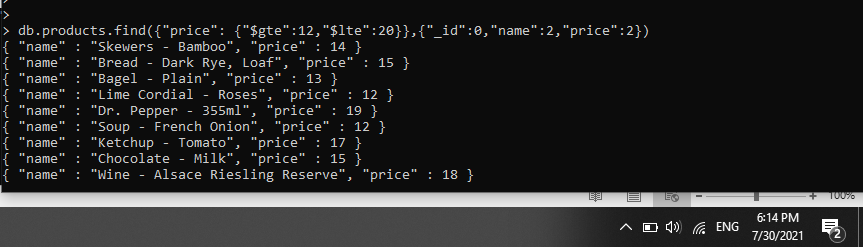
1. Write a query to return *name* and *price* for products of type “Health Care” in the *inventory* database.

|  |
| --- |
| db.products.find({"type":"Health Care"},{"\_id":0,"name":1,"price":1}) |



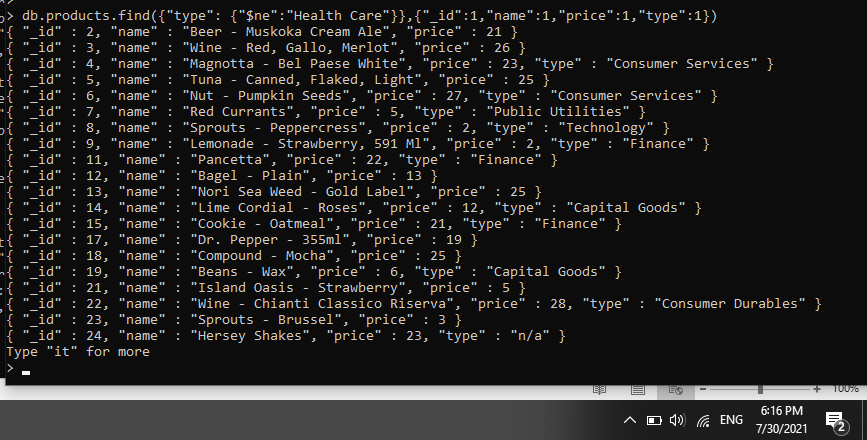
1. Write a query to return *name* and *price* for products with price between $12 and $20 (Values *12* and *20* are included).

|  |
| --- |
| db.products.find({"price": {"$gte":12,"$lte":20}},{"\_id":0,"name":2,"price":2}) |



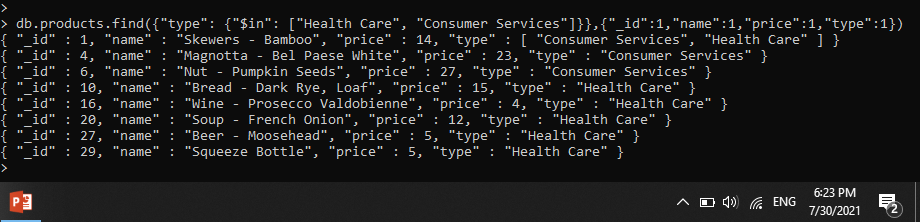
1. Write a query to return *id*, *name*, *price*, and *type* for products that are not of type “Health Care”.

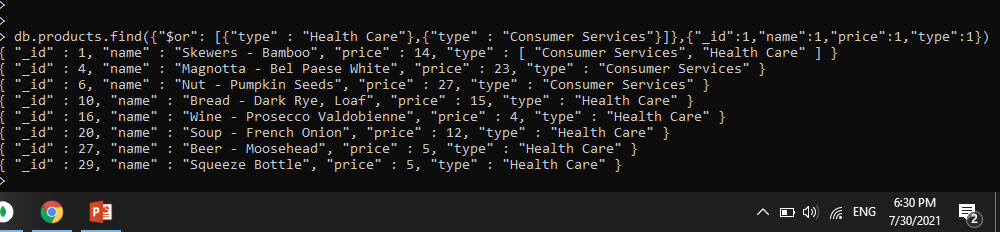
|  |
| --- |
| db.products.find({"type": {"$ne":"Health Care"}},{"\_id":1,"name":1,"price":1,"type":1}) |



1. Write a query to return *id*, *name*, *price*, and type for products with type “Health Care” or “Consumer Services.”

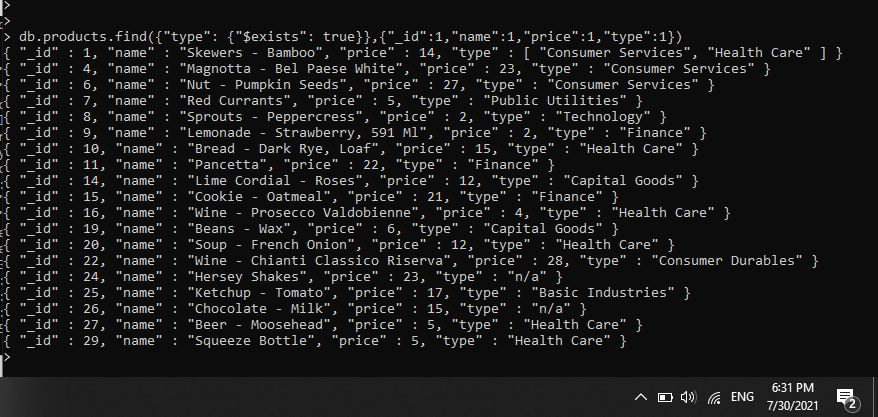
|  |
| --- |
| db.products.find({"type": {"$in": ["Health Care", "Consumer Services"]}},{"\_id":1,"name":1,"price":1,"type":1})  db.products.find({"$or": [{"type" : "Health Care"},{"type" : "Consumer Services"}]},{"\_id":1,"name":1,"price":1,"type":1}) |





1. Write a query to return *id*, *name*, *price*, and *type* for products that do have the *type* key.

|  |
| --- |
| db.products.find({"type": {"$exists": true}},{"\_id":1,"name":1,"price":1,"type":1}) |



1. Write a query to return *id*, *name*, *price*, and *type* for products that their type is both Health Care and Consumer Services.

|  |
| --- |
| db.products.find({"type": {"$all": ["Health Care", "Consumer Services"]}},{"\_id":1,"name":1,"price":1,"type":1}) |

