First download the MongoDB database.

then download the Mongo Database Tools,& the mongo compass.

we use the mongo Shell like this:

**• Create database's name : « starter ».**

**• createCollections products.**

* **Importe the products file csv in ur data base,**
* **Add the installation path (link) in the Path variable.**
* **then i ur console, run the mongoimport request to import your csv file into the “starter” database and into the “products” collection:**

C:\Program Files\MongoDB\Server\4.4\bin>mongoimport -d starter -c products --type csv --file C:\Users\Youcode\Desktop\products.csv --headerline

* **Make the following requests:**

**• Display the data of the collection using db.products.find ();**

db.products.find() ;

**• How many documents are included in the "products" collection.**

db.products.find().count()

77

**• Change MongoDB batch size to display just 10 items,(\*the batchSize is not working so i change it to limit\*)**

db.products.find().limit(10).pretty()

**• Execute the following two commands one after the other**

db.products.find({supplierID:{$eq:19}}).pretty();

db.products.find({SupplierID:{$eq:19}}).pretty();

**• WHAT DO U THINK ?**

* The first command does not work because supplierID in 'S' in lowercase on the other hand the second command which contains SupplierID in 'S' in uppercase
* The second command searches for the documents that we request it ID = 19

**• Select the documents that have a SupplierID equal to 19 and UnitPrice less than 10.**

""db.products.find({SupplierID:{$eq:19},UnitPrice:{$lt:10}}).pretty()""

db.products.find({SupplierID:19,UnitPrice:{$lt:10}}).pretty()

{

"\_id" : ObjectId("6005a91920fb9f3712740512"),

"ProductID" : 41,

"ProductName" : "Jack's New England Clam Chowder",

"SupplierID" : 19,

"CategoryID" : 8,

"QuantityPerUnit" : "12 - 12 oz cans",

"UnitPrice" : 9.65,

"UnitsInStock" : 85,

"UnitsOnOrder" : 0,

"ReorderLevel" : 10,

"Discontinued" : 0

}

**•In the shell, create the following variable:**

var allProducts = db.products.find ();

**•Run the following snippet of code in your shell**

while (allProducts.hasNext ()) {printjson (allProducts.next ()); };

**•Display all cursor items: db.products.find () using forEach function.**

db.products.find (). forEach (printjson);

**•In the Mongo shell, run the following command**

db.products.find ({CategoryID: 4}, {ProductName: 1}). pretty ();

db.products.find ({CategoryID: 4}, {ProductName: 1}). pretty ();

**• Using the shell, create an index of type "text" for "ProductName":**

db.products.createIndex ({ProductName: "text"});

db.products.find ({CategoryID: 4}, {ProductName: 1}). pretty ();

**• Then run the following command:**

db.products.find ({CategoryID: 4}, {ProductName: 1}). sort ({ProductName: -1}). pretty ();

**• Display product documents with a "CategoryID" equal to one, in ascending order of ProductName.**

db.products.find (). sort ({CategoryID: 1}). pretty ();

**• Limit the number of documents to be displayed to 2;**

db.products.find (). limit (2) .pretty ();

db.products.find (). sort ({ProductID: -1}). limit (1) .pretty ();

**• What does the above command display.**

db.products.find (). sort ({ProductID: -1}). limit (1) .pretty ();

{

"\_id": ObjectId ("6005a91920fb9f3712740536"),

"ProductID": 77,

"ProductName": "Original Frankfurter grÃ¼ne SoÃŸe",

"SupplierID": 12,

"CategoryID": 2,

"QuantityPerUnit": "12 boxes",

"Unit price": 13,

"Units in stock": 32,

"UnitsOnOrder": 0,

"ReorderLevel": 15,

"Discontinued": 0

}

**• Run the following command:**

db.products.insert (

{

"\_id": ObjectId ("60054a6cd020a8ea8dfd8064"),

"ProductID": 78,

"ProductName": "Moroccan Raclette",

"SupplierID": 28,

"CategoryID": 4,

"QuantityPerUnit": "5 kg pqt.",

"Unit price": 65,

"Units in Stock": 179,

"UnitsOnOrder": 10,

"ReorderLevel": 0,

"Discontinued": 0

}

);

**• What do you notice?**

inserted

**• Now do the following command:**

db.products.insert (

{

"ProductID": 78,

"ProductName": "Moroccan Raclette",

"SupplierID": 28,

"CategoryID": 4,

"QuantityPerUnit": "5 kg pqt.",

"Unit price": 65,

"Units in Stock": 179,

"UnitsOnOrder": 10,

"ReorderLevel": 0,

"Discontinued": 0

}

);

**• Now do the following command:**

db.products.insertMany (

[{

"ProductID": 79,

"ProductName": "Moroccan maple syrup",

"SupplierID": 29,

"CategoryID": 2,

"QuantityPerUnit": "24 bottles of 500 ml",

"Unit price": 28.5,

"Units in stock": 113,

"UnitsOnOrder": 0,

"ReorderLevel": 25,

"Discontinued": 0

}

,

{

"ProductID": 80,

"ProductName": "Moroccan Steeleye Stout",

"SupplierID": 16,

"CategoryID": 1,

"QuantityPerUnit": "24 to 12 oz bottles",

"Unit price": 18,

"Units in stock": 20,

"UnitsOnOrder": 0,

"ReorderLevel": 15,

"Discontinued": 0

}

]

);

**• Run the following command:**

db.products.insert (

{

"ProductID": 81,

"ProductName": "Special Moroccan Raclette",

"SupplierID": 28,

"CategoryID": 4,

"QuantityPerUnit": "5 kg pqt.",

"Unit price": 65,

"Units in Stock": 179,

"UnitsOnOrder": 10,

"ReorderLevel": 0,

"Discontinued": 0,

"DateFirstShip": new date ()

}

);

**• What is the difference between this document and the documents already inserted.**

**• Run the following command:**

db.products.find ({DateFirstShip: {$ exists: true}});

**• What does she refer?**

db.products.find ({DateFirstShip: {$ exists: true}}). pretty ();

{

"\_id": ObjectId ("6007409a7c76594081756939"),

"ProductID": 81,

"ProductName": "Special Moroccan Raclette",

"SupplierID": 28,

"CategoryID": 4,

"QuantityPerUnit": "5 kg pqt.",

"Unit price": 65,

"Units in Stock": 179,

"UnitsOnOrder": 10,

"ReorderLevel": 0,

"Discontinued": 0,

"DateFirstShip": ISODate ("2021-01-19T20: 27: 06.550Z")

}

**• Run the following command:**

db.shutdownServer ();

**• Make the change, so that the previous method works fine.**

db.shutdownServer ();

**• The shutdown command only works with the admin database; try 'use admin'**

> use admin

switched to db admin

> db.shutdownServer ();

the server should be down ...

**• Execute quit (); to leave();**

> quit();

C: \ Program Files \ MongoDB \ Server \ 4.4 \ bin>

AND THAT’S IT TEACHER Brief 1 is Done

**Thank u ☺**