Installer the MongoDB database.

Installer l’outil [Mongo Database Tool](https://docs.mongodb.com/database-tools/installation/installation-windows/), en plus du mongo compass.

En utilisant mongo Shell :

* Créer une base de données nommé : « starter ».
* Créer une collections products.

Importer le fichier ci-joint dans votre base de données mongo, le fichier csv ci-joint. Pour ce faire :

Ajouter son chemin d’installation dans la variable Path.

Depuis votre console, exécuter la requête mongoimport pour importer votre fichier csv dans la base de données « starter » et dans la collection « products » :

Faire les requêtes suivantes :

Afficher les données de la collection en utilisant db.products.find() ;

Combien de documents sont inclus dans la collection « products ».

77 documents

Modifier la taille du batch de MongoDB pour afficher juste 10 éléments.

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

Exécuter les deux commandes suivantes l’une après l’autre

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

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

Que remarquez-vous ?

Il affiche juste les SupplierID eguale 19 avec S majuscule

Sélectionner les documents qui ont un SupplierID égale à 19 et UnitPrice moins que 10.

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

Dans le shell, créer la variable suivante :

var allProducts=db.products.find();

on a déclarer le variable

Exécuter le bout de code suivant dans votre shell

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

ok : on a executer le code.

Afficher tous les éléments du curseur : db.products.find() en utilisant la fonction forEach.

db.products.find().forEach((doc)=>printjson(doc))

Dans le shell Mongo, executer la commande suivante

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

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

{

"\_id" : ObjectId("6005b050c46e512e14fd11c9"),

"ProductName" : "Queso Cabrales"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ca"),

"ProductName" : "Queso Manchego La Pastora"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e4"),

"ProductName" : "Gorgonzola Telino"

}

{ "\_id" : ObjectId("6005b050c46e512e14fd11e5"), "ProductName" : "Geitost" }

{

"\_id" : ObjectId("6005b050c46e512e14fd11e6"),

"ProductName" : "Mascarpone Fabioli"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ff"),

"ProductName" : "Camembert Pierrot"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1203"),

"ProductName" : "Raclette Courdavault"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd120f"),

"ProductName" : "Mozzarella di Giovanni"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1211"),

"ProductName" : "Flotemysost"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1212"),

"ProductName" : "Gudbrandsdalsost"

}

En utilisant le shell, créer un index de type « text » pour « ProductName » :

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

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 1,

"numIndexesAfter" : 2,

"ok" : 1

}

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

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

{

"\_id" : ObjectId("6005b050c46e512e14fd11c9"),

"ProductName" : "Queso Cabrales"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ca"),

"ProductName" : "Queso Manchego La Pastora"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e4"),

"ProductName" : "Gorgonzola Telino"

}

{ "\_id" : ObjectId("6005b050c46e512e14fd11e5"), "ProductName" : "Geitost" }

{

"\_id" : ObjectId("6005b050c46e512e14fd11e6"),

"ProductName" : "Mascarpone Fabioli"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ff"),

"ProductName" : "Camembert Pierrot"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1203"),

"ProductName" : "Raclette Courdavault"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd120f"),

"ProductName" : "Mozzarella di Giovanni"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1211"),

"ProductName" : "Flotemysost"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1212"),

"ProductName" : "Gudbrandsdalsost"

}

Ensuite, executer la commande suivante:

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

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

{

"\_id" : ObjectId("6005b050c46e512e14fd1203"),

"ProductName" : "Raclette Courdavault"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ca"),

"ProductName" : "Queso Manchego La Pastora"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11c9"),

"ProductName" : "Queso Cabrales"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd120f"),

"ProductName" : "Mozzarella di Giovanni"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e6"),

"ProductName" : "Mascarpone Fabioli"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1212"),

"ProductName" : "Gudbrandsdalsost"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e4"),

"ProductName" : "Gorgonzola Telino"

}

{ "\_id" : ObjectId("6005b050c46e512e14fd11e5"), "ProductName" : "Geitost" }

{

"\_id" : ObjectId("6005b050c46e512e14fd1211"),

"ProductName" : "Flotemysost"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ff"),

"ProductName" : "Camembert Pierrot"

}

Afficher les documents de products ayant un « CategoryID » égal à un, par ordre ascendant de ProductName.

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

Limiter le nombre des documents à afficher à 2 ;

db.products.find().sort({ProductID:-1}).limit(2).pretty();

{

"\_id" : ObjectId("6005b050c46e512e14fd120d"),

"ProductID" : 77,

"ProductName" : "Original Frankfurter grÃ¼ne SoÃŸe",

"SupplierID" : 12,

"CategoryID" : 2,

"QuantityPerUnit" : "12 boxes",

"UnitPrice" : 13,

"UnitsInStock" : 32,

"UnitsOnOrder" : 0,

"ReorderLevel" : 15,

"Discontinued" : 0

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1210"),

"ProductID" : 76,

"ProductName" : "LakkalikÃ¶Ã¶ri",

"SupplierID" : 23,

"CategoryID" : 1,

"QuantityPerUnit" : "500 ml",

"UnitPrice" : 18,

"UnitsInStock" : 57,

"UnitsOnOrder" : 0,

"ReorderLevel" : 20,

"Discontinued" : 0

}

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

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

{

"\_id" : ObjectId("6005b050c46e512e14fd120d"),

"ProductID" : 77,

"ProductName" : "Original Frankfurter grÃ¼ne SoÃŸe",

"SupplierID" : 12,

"CategoryID" : 2,

"QuantityPerUnit" : "12 boxes",

"UnitPrice" : 13,

"UnitsInStock" : 32,

"UnitsOnOrder" : 0,

"ReorderLevel" : 15,

"Discontinued" : 0

}

Qu’affiche la commande ci-hauts.

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

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

{ "\_id" : ObjectId("6005b050c46e512e14fd11da"), "ProductName" : "Chai" }

{ "\_id" : ObjectId("6005b050c46e512e14fd11d9"), "ProductName" : "Chang" }

{

"\_id" : ObjectId("6005b050c46e512e14fd1200"),

"ProductName" : "Chartreuse verte"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11eb"),

"ProductName" : "CÃ´te de Blaye"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11d7"),

"ProductName" : "GuaranÃ¡ FantÃ¡stica"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11ef"),

"ProductName" : "Ipoh Coffee"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1210"),

"ProductName" : "LakkalikÃ¶Ã¶ri"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd1207"),

"ProductName" : "Laughing Lumberjack Lager"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd120a"),

"ProductName" : "Outback Lager"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd120e"),

"ProductName" : "RhÃ¶nbrÃ¤u Klosterbier"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e7"),

"ProductName" : "Sasquatch Ale"

}

{

"\_id" : ObjectId("6005b050c46e512e14fd11e8"),

"ProductName" : "Steeleye Stout"

}

Exécuter la commande suivante :

db.products.insert(

{

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

"ProductID" : 78,

"ProductName" : "Raclette Marocaine",

"SupplierID" : 28,

"CategoryID" : 4,

"QuantityPerUnit" : "5 kg pkg.",

"UnitPrice" : 65,

"UnitsInStock" : 179,

"UnitsOnOrder" : 10,

"ReorderLevel" : 0,

"Discontinued" : 0

}

) ;

Que remarquez-vous ?

WriteResult({ "nInserted" : 1 }) , j’ai remarquer que on a ajouter un ID = 78

Maintenant, exécuter la commande suivante :

db.products.insert(

{

"ProductID" : 78,

"ProductName" : "Raclette Marocaine",

"SupplierID" : 28,

"CategoryID" : 4,

"QuantityPerUnit" : "5 kg pkg.",

"UnitPrice" : 65,

"UnitsInStock" : 179,

"UnitsOnOrder" : 10,

"ReorderLevel" : 0,

"Discontinued" : 0

}

) ;

Maintenant, exécuter la commande suivante :

db.products.insertMany(

[{

"ProductID" : 79,

"ProductName" : "Sirop d'érable Marocain",

"SupplierID" : 29,

"CategoryID" : 2,

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

"UnitPrice" : 28.5,

"UnitsInStock" : 113,

"UnitsOnOrder" : 0,

"ReorderLevel" : 25,

"Discontinued" : 0

}

,

{

"ProductID" : 80,

"ProductName" : "Moroccan Steeleye Stout",

"SupplierID" : 16,

"CategoryID" : 1,

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

"UnitPrice" : 18,

"UnitsInStock" : 20,

"UnitsOnOrder" : 0,

"ReorderLevel" : 15,

"Discontinued" : 0

}

]

);

Exécuter la commande suivante :

db.products.insert(

{

"ProductID" : 81,

"ProductName" : "Raclette Marocaine Spéciale",

"SupplierID" : 28,

"CategoryID" : 4,

"QuantityPerUnit" : "5 kg pkg.",

"UnitPrice" : 65,

"UnitsInStock" : 179,

"UnitsOnOrder" : 10,

"ReorderLevel" : 0,

"Discontinued" : 0,

"DateFirstShip": new Date()

}

) ;

Quel est la différence entre ce document et les documents déjà inséré.

Je constate que insert et inserer un objet son id et insertMany il a donnée des id

Exécuter la commande suivante :

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

Que renvoi-t-elle?

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

{ "\_id" : ObjectId("600800dd847b9793cbb239f8"), "ProductID" : 81, "ProductName" : "Raclette Marocaine Spéciale", "SupplierID" : 28, "CategoryID" : 4, "QuantityPerUnit" : "5 kg pkg.", "UnitPrice" : 65, "UnitsInStock" : 179, "UnitsOnOrder" : 10, "ReorderLevel" : 0, "Discontinued" : 0, "DateFirstShip" : ISODate("2021-01-20T10:07:25.562Z") }

Exécuter la commande suivante:

db.shutdownServer();

db.getSiblingDB("admin").shutdownServer()

server should be down...

Faites la modification, pour que la méthode précédente s’exécute bien.

Exécuter quit() ;

> quit()

C:\Users\Youcode>