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 ». **use starter**

Créer une collections products. **db.createCollection('products')**

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

**mongoimport -d starter -c products --type csv --file C:\Users\Youcode\Desktop\products.csv --headerline**

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() ; **db.products.find()**

Combien de documents sont inclus dans la collection « products **».db.products.find().count()**

Modifier la taille du batch de MongoDB pour afficher juste 10**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 ? **la première commande ne pas afficher(s) ms la 2eme commande active (S).**

Sélectionner les documents qui ont un SupplierID égale à 19 et UnitPrice moins que 10. **db.products.find({supplierID:{$eq:19},UnitPrice:{}$lt :10}).pretty();**

Dans le shell, créer la variable suivante :

var allProducts=db.products.find();

Exécuter le bout de code suivant dans votre shell

**while(allproducts.hasNext()){ printjson(allproducts.next()); };**

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

Dans le shell Mongo, executer la commande suivante

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

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

**Db.products.creatIndex({productName : ‘text’})**

Ensuite, executer la commande suivante:

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

Afficher les documents de products ayant un « CategoryID » égal à un, par ordre ascendant de ProductName. **db.products.find()({ CategoryID: 1 }).sort({productName :1}).limit(1).pretty()**

Limiter le nombre des documents à afficher à 2 ;**db.products.find()limit(2)**

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

Qu’affiche la commande ci-hauts. **le dernier products**

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 ? **insert le produit dans base de donner starter 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

}

) ;

**La même chose insert le Produite dans la base de donner products**

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

}

]

);

**Insert deux produits avec un seul commande dans base de donner products**

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.

**Ajout un nouveaux fonction "DateFirstShip": new Date()**

Exécuter la commande suivante :

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

**afficher le dernier document avec la date de insertion**

Que renvoi-t-elle? **La date de insertion le document**

Exécuter la commande suivante:

db.shutdownServer();

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

Exécuter quit() ; **quitte de mongo**