

Initiation au NoSQL

Exercices 05 – MongoDB appliquée

MongoDB en application

Installations préalables

Installation de Robot 3T

Url de téléchargement: <https://robomongo.org/download>

Installer le soft (vous n'êtes pas obligé de laisser votre mail ni votre nom cliquez sur finish).

Fonctionnement de MongoDB

- Lancez votre docker-compose
- Depuis la fenêtre d'accueil de Robo 3T
 - Cliquez sur Create (en haut à gauche)
 - Laissez Direct Connexion
 - Choisissez un nom
 - Remplacez localhost par 192.168.99.100 (sauf linuxiens)
 - Sélectionnez la connexion
 - Cliquez sur select
 - Si la connexion se fait c'est que MongoDB est fonctionnelle

Exercices

Prise en main

Sélection d'une BDD

Sélectionnez la bdd 'mabdd'

Rafraîchissez la liste au niveau de New Connection. Que constatez-vous ?

Insertion simple

Créez un item de collection 'macollection' avec les entrées x=1 et y=2

Rafraîchissez la liste au niveau de New Connection. Que constatez-vous ?

Insertion multiple

Créez-en 1 requêtes plusieurs items dans 'macollection' avec les propriétés suivantes:

x	y
2	2
3	3
4	4

Lecture

Affichez l'ensemble de la collection (!!!sans double clic sur l'élément!!!)

Insertion avec _id

Insérer un item avec les entrée x=200, y=50 et l'_id:1

Insérer un deuxième item avec les entrée x=200, y=600 et l'_id:1

Que constatez-vous sur la deuxième insertion ? Expliquez

Modification

La bonne combinaison de valeur pour l'_id:1 est x=200 et y=600 (y à modifier), trouvez un moyen de modifier les valeurs.

Avez-vous réussi du premier coup ?

Suppression

Supprimer la collection

On passe la seconde

Q1

Insérez votre prénom et votre téléphone ainsi que celui de votre voisin dans 2 items de collection contact.

Q2

Insérez une nouvelle ligne sans name mais avec le phone = '0606060606'

Faites une requête pour n'afficher que les noms dans toute la collection.

Que constatez-vous sur le résultat ?

Q3

Saisissez le cmd suivante :

```
db.createCollection( "contacts1",
{
  validator: { $and:
    [
      { phone: { $type: "string" } },
      { status: { $in: [ "Unknown", "Incomplete" ] } }
    ]
  }
}
)
```

Saisissez ensuite les cmd 1 par une :

```
db.contacts1.insert({_id:1,phone:'505050505',status:'Unknow'})
db.contacts1.insert({_id:1,phone:505050505,status:'Unknown'})
db.contacts1.insert({_id:1,phone:'505050505',status:'Unknown'})
```

Que constatez-vous pour chaque ligne ? Expliquez

Q4

Saisissez la cmd suivante :

```
db.createCollection( "contacts2",
{
  validator: { $or:
    [
```

```
{ phone: { $type: "string" } },  
  { status: { $in: [ "Unknown", "Incomplete" ] } }  
]  
}  
}  
)
```

Saisissez ensuite les cmd 1 par une:

```
db.contacts2.insert({_id:1,phone:'505050505',status:'Unknow'})  
db.contacts2.insert({phone:505050505,status:'Unknown'})  
db.contacts2.insert({phone:'505050505',status:'Unknown'})
```

Que constatez-vous pour chaque ligne? Expliquez

Q5

Supprimez l'éléments de _id=1 de la collection contact2

Q6

Supprimez les 3 collections créées

Les choses sérieuses

Insertion des données

Créer une job Talend insérant les préférences dans une collection preferences de la DB (1000 itérations)

Q7

Retournez une liste unique des noms des utilisateurs

Q8

Comptez le nombre d'itération de chaque nom et prenez la plus grande itération pour la suite du travail

Q9

Comptez la somme de la limite du nom le plus utilisé

Q10

Idem pour la moyenne des valeurs de limit

Q11

Idem pour les valeurs min et max

Que constatez-vous pour max ?