**MongoDB : Principes de base**

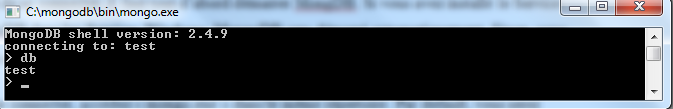
1. **Se connecter à la base de données**

Avant de se connecter, il faut tout d’abord démarrer MongDB. Si vous avez installé le Service MongoDB dans les Services Windows, MongoDB sera démarré automatiquement. Sinon, vous avez à accéder le repertoire qui contient MongoDB et accédez « mongod.exe ».

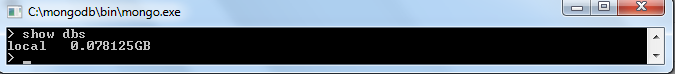
Pour se connecter, accédez « mongo.exe » dans le même répertoire. Par default, vous serez connectés à la base de données « test ».

1. **Sélectionner et ajouter une base de données**

Pour savoir à quelle base de données vous êtes connectés dans un moment donné, il suffit d’exécuter la commande « db ».

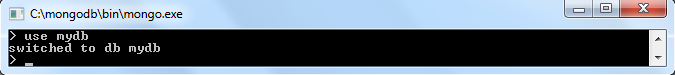


Pour afficher la liste des bases de données, exécutez la commande « show dbs ».



La console affiche « local » puisque nous sommes connectés à la base de données « test » qui est par default la base de données locale.

Pour sélectionner une base de données existante ou ajouter une nouvelle base de données, exécuter la commande « use » suivie du nom de la base de données. Si la base de données existe déjà, elle sera sélectionnée, sinon, une nouvelle base de données sera générée et sélectionnée. Maintenant, on va ajouter une nouvelle base de données « mydb » en exécutant la commande « use mydb ».



Si maintenant on exécute la commande « db », la base de données « mydb » ne sera pas affichée parmi les bases de données existantes. MongoDB ne va pas créer la base de données jusqu’à ce qu’on ajoute des données à elle.

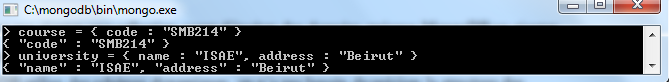
1. **Créer une Collection et ajouter des Documents**

Maintenant on va ajouter des documents dans une collection qu’on va l’appeler « testData » pour faire partir de la base de données « mydb » qu’on a ajouté avant.

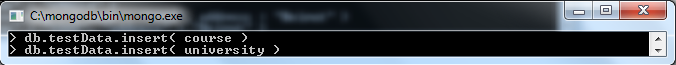
On ne va pas ajouter une collection avant d’insérer des données, puisque MongoDB va ajouter implicitement la collection quand on va l’utiliser pour la première fois. De même, puisque MongoDB utilise des schémas dynamiques, on n’a pas besoin de préciser la structure des documents qu’on va ajouter.

Pour commencer, exécutez la commande « db » pour s’assurer que vous êtes connectés à « mydb ». Sinon, exécutez la commande « use mydb ».

Maintenant en utilisant des opérations Java Script, on va ajouter des documents. En exécutant l’opération « course = { code : "SMB214" } », on va créer un document qui s’appelle « course » avec une clé « code » qui a une valeur « SMB214 ». Ensuite, on ajoute un document « university » ayant une clé « name » et sa valeur « ISAE » et une autre clé « address » et sa valeur « Beirut » en executant l’opération « university = { name : "ISAE", address : "Beirut" } ».

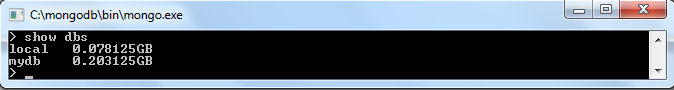


Maintenant on va insérer les documents « course » et « university » dans une collection qu’on va appeler « testData » en exécutant les opérations suivantes : « db.testData.insert( course ) » et «  db.testData.insert( university ) ».

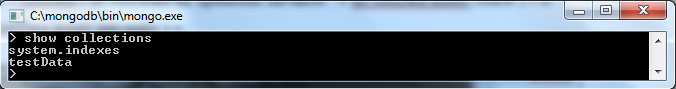


En ajoutant le premier document, MongoDB va ajouter la base de données « mydb » et la collection « testData ».

Pour s’assurer que la base de données « mydb » a été créée on exécute la commande   
« show dbs ».

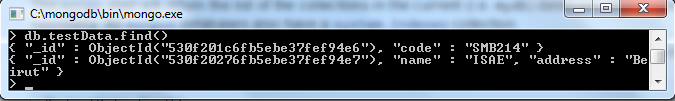


Pour s’assurer que la collection « testData » a été créée on exécute la commande   
« show collections ».



En exécutant cette commande, MongoDB va afficher seulement les collections qui appartiennent à la base de données sélectionnée « mydb ». Ainsi, en plus de la collection « testData » qu’on a ajouté, MongoDB ajoute une collection pour gérer les indexes.

Pour s’assurer que les deux documents qu’on a ajouté existent dans la collection « testData » on exécute une requête sur la collection en utilisant la méthode « find() ». La requête est la suivante : « db.testData.find() ».



MongoDB affiche le contenu des deux documents. Si vous remarquez, chaque document a un « ObjectId » avec une valeur unique. MongoDB a donc créé implicitement les attributs « ObjectId » avant d’insérer les documents dans la base de données.