

mongo

Voici le développement détaillé de chaque point des modules de la formation sur MongoDB pour débutants :

Module 1 : Introduction à MongoDB

Qu'est-ce que MongoDB ?

- ****Définition et avantages par rapport aux bases de données relationnelles**** : MongoDB est une base de données NoSQL orientée documents qui stocke les données sous forme de documents JSON. Contrairement aux bases de données relationnelles (SQL), MongoDB ne nécessite pas de schéma rigide, ce qui permet une flexibilité accrue et une adaptation facile aux modifications de données.
- ****Concepts de NoSQL et pourquoi choisir MongoDB**** : Les bases de données NoSQL sont conçues pour gérer des quantités massives de données non structurées ou semi-structurées. MongoDB est particulièrement adapté aux applications nécessitant un développement agile, des modifications fréquentes de schéma et une scalabilité horizontale.

Installation de MongoDB

- ****Guide étape par étape pour l'installation sur Windows, macOS, et Linux**** :
 - ***Windows*** : Télécharger l'installateur depuis le site officiel de MongoDB, suivre l'assistant d'installation et configurer le chemin d'accès (`PATH`) pour `mongod` et `mongo`.
 - ***macOS*** : Utiliser Homebrew avec la commande `brew install mongodb-community@5.0`.
 - ***Linux*** : Utiliser les paquets `.deb` ou `.rpm` en ajoutant le dépôt officiel de MongoDB.
- ****Lancement du service MongoDB**** : Comment démarrer le serveur MongoDB (`mongod`) et se connecter au shell MongoDB (`mongo`).

Premier contact avec MongoDB

- ****Introduction à MongoDB Compass et MongoDB Shell**** :
 - ***MongoDB Compass*** : Interface graphique permettant de visualiser et manipuler les données sans lignes de commande.
 - ***MongoDB Shell*** : Environnement en ligne de commande pour exécuter des requêtes et gérer la base de données.

Module 2 : Les bases de MongoDB

Concepts fondamentaux

- ****Différence entre bases de données, collections, et documents**** :
 - ***Base de données*** : Regroupe plusieurs collections.
 - ***Collection*** : Ensemble de documents similaires, équivalent aux tables en SQL.
 - ***Document*** : Unité de base contenant des paires clé-valeur, similaire à un enregistrement en SQL.
- ****Structure des documents en BSON**** : BSON est une représentation binaire de JSON utilisée par MongoDB pour des performances accrues et une compatibilité avec différents types de données.

Premières commandes

- ****Création d'une base de données et insertion de documents**** :
 - ``use nomDeLaBase`` : Pour créer ou se déplacer dans une base de données.
 - ``db.nomDeLaCollection.insertOne()`` et ``insertMany()`` : Pour insérer des documents.
- ****Lecture des données**** :
 - ``db.nomDeLaCollection.find()`` : Retourne tous les documents.
 - ``findOne()`` : Retourne le premier document correspondant aux critères.

Mise à jour et suppression

- ****Utilisation des commandes `update`, `updateMany`, et `delete`**** :
 - ``updateOne()`` et ``updateMany()`` : Pour modifier un ou plusieurs documents avec des opérateurs tels que `\$set`

et ``$unset``.

- ``deleteOne()`` et ``deleteMany()`` : Pour supprimer des documents spécifiques.

Module 3 : Manipulation des données

Filtrage et requêtes avancées

- ****Utilisation de filtres, projections et tri**** :
 - Les filtres permettent de cibler des documents selon des critères spécifiques (``{champ: valeur}``).
 - Les projections (``find({}, {champ: 1})``) limitent les champs retournés.
 - Utiliser ``sort()`` pour trier les résultats.

Opérations de mise à jour complexes

- ****Les opérateurs de mise à jour tels que ``$set``, ``$unset``, et ``$inc``**** :
 - ``$set`` : Modifie ou ajoute un champ.
 - ``$unset`` : Supprime un champ.
 - ``$inc`` : Incrémente un champ numérique.

Agrégation des données

- ****Introduction au pipeline d'agrégation et exemples de cas pratiques**** :
 - Utilisation de ``aggregate()`` avec des étapes comme ``$match``, ``$group``, ``$project``, etc.
 - Exemple : calculer le total des ventes par produit.

Module 4 : Indexation et optimisation

Création et gestion des index

- ****Importance des index pour les performances**** :
 - Les index réduisent le temps de recherche et améliorent la rapidité des requêtes.
- ****Création d'index simples et composés**** :
 - ``createIndex({champ: 1})`` : Pour indexer un champ.
 - Index composés pour optimiser les requêtes sur plusieurs champs.

Analyse des performances

- ****Utilisation des outils de suivi des performances et optimisation des requêtes**** :
 - Analyse avec ``explain()`` pour vérifier l'efficacité des requêtes.
 - Optimisation en utilisant des index et en améliorant les requêtes.

Module 5 : Sécurité et sauvegarde

Gestion des utilisateurs et des permissions

- ****Création d'utilisateurs et configuration des rôles**** :
 - ``db.createUser()`` pour ajouter des utilisateurs.
 - Configuration des rôles tels que ``readWrite``, ``dbAdmin``.

Sauvegarde et restauration

- ****Utilisation de ``mongodump`` et ``mongorestore``**** :
 - ``mongodump`` pour sauvegarder la base de données.
 - ``mongorestore`` pour restaurer les données à partir de la sauvegarde.

Module 6 : Travaux pratiques et projet final

Exercices pratiques

- ****Création et gestion d'une base de données de gestion des contacts**** :
 - Implémenter des commandes de création, lecture, mise à jour et suppression sur une base de données fictive.

Projet final

- ****Développement d'une mini-application CRUD avec MongoDB comme backend**** :
- Conception d'une application web basique connectée à MongoDB.
- Réalisation des opérations de création, lecture, mise à jour, et suppression de données.

Module 7 : Introduction à l'intégration avec des applications

Connexion avec des langages de programmation

- ****Utilisation de MongoDB avec Node.js, Python, ou Java**** :
- Installation des bibliothèques comme `mongodb` pour Node.js, `pymongo` pour Python.
- Exemple de connexion et exécution de requêtes simples.

Requêtes et gestion des données à partir de l'application

- ****Implémentation de requêtes CRUD via le code**** :
- Scripts d'insertion, lecture et manipulation des données dans une application.

Cette structure est conçue pour offrir un apprentissage complet et progressif de MongoDB, de la base à l'utilisation avancée et intégration dans des applications réelles.