

Projet DB_165



Rapport de Projet

Zeqiri Amir – CID2A
ETML, Vennes – A01
32 Périodes
Maître : M. Mveng

Table des matières

| | | |
|----------|-----------------------------------------------------------------------------|----------|
| 1 | DESCRIPTION DU PROJET DANS SON ENSEMBLE | 3 |
| 1.1 | SUJET | 3 |
| 1.2 | SPÉCIFICITÉS | 3 |
| 1.2.1 | <i>Chapitre A – Importer les données et le schéma de la base de données</i> | 3 |
| 1.2.2 | <i>Chapitre F – Création des index.....</i> | 3 |
| 1.2.3 | <i>Chapitre G – Backup/Restore</i> | 3 |
| 2 | PRATIQUE | 4 |
| 2.1 | A – IMPORTER LES DONNÉES ET LE SCHÉMA DE LA BASE DE DONNÉES | 4 |
| 2.2 | F – CRÉATION DES INDEX | 4 |
| 2.3 | G – BACKUP/RESTORE | 5 |

1 DESCRIPTION DU PROJET DANS SON ENSEMBLE

Sujet

Implémenter et utiliser une base de données MongoDB.

Spécificités

1.1.1 Chapitre A – Importer les données et le schéma de la base de données

- Restaurer les collections dans une base de données nommé « db_mflix » dans une instance de MongoDB.
- Expliquer en détail la commande utilisée.

1.1.2 Chapitre F – Création des index

- Dans votre base de données « db_mflix », identifiez les champs qui bénéficieraient le plus de la création d'index.

1.1.3 Chapitre G – Backup/Restore

- Expliquez chaque étape de ces processus et les raisons pour lesquelles ces étapes sont importantes.

2 PRATIQUE

A – Importer les données et le schéma de la base de données

```
DOCKER EXEC -i MONGO MONGORESTORE --URI=MONGODB://ROOT:ADMIN@LOCALHOST:27017 --GZIP --  
ARCHIVE=BACKUPDB/DB_MFLIX.GZ
```

Explications de la commande utilisée :

docker exec: Cela signifie que nous utilisons Docker pour exécuter une commande à l'intérieur d'un conteneur Docker.

-i: Cela dit à Docker d'exécuter la commande de manière interactive, en prenant l'entrée standard de l'utilisateur pour cette commande.

mongo: C'est le nom du conteneur Docker où se trouve MongoDB, le système de gestion de base de données.

mongorestore: C'est la commande spécifique à MongoDB que nous utilisons pour restaurer une base de données à partir d'une sauvegarde.

--uri=mongodb://root:admin@localhost:27017: C'est l'adresse pour se connecter à MongoDB. "root" est le nom d'utilisateur et "admin" est le mot de passe pour se connecter à MongoDB. "localhost:27017" spécifie l'emplacement et le port de MongoDB.

--gzip: Cela signifie que l'archive de sauvegarde est compressée avec Gzip. Cette option indique à la commande qu'elle doit décompresser l'archive avant de restaurer la base de données.

--archive=backupdb/db_mflix.gz: C'est l'emplacement de l'archive de sauvegarde que nous voulons restaurer. Elle est située dans le répertoire backupdb à l'intérieur du conteneur Docker.

F – Création des index

Les réponses se trouvent dans le fichier playground :

"X-165-P-DB-NoSQL-ZeqiriAmir-playground-F-indexes.mongodb.js"

G – Backup/Restore

2.1.1 Backup (sauvegarde)

Pour sauvegarder la base de données, j'ai utilisé la commande `mongodump`. Voici la commande :

```
docker exec -i mongo mongodump --uri=mongodb://root:admin@localhost:27017 --
authenticationDatabase=admin --db="db_mflix" --gzip --archive=./backupdb/mongodump-
db_mflix.gz
```

Explications de la commande utilisée :

`docker exec`: Cette commande est utilisée pour exécuter une commande à l'intérieur d'un conteneur Docker.

`-i`: Cette option signifie que nous voulons exécuter la commande de manière interactive, en utilisant l'entrée standard.

`mongo`: C'est le nom du conteneur Docker dans lequel nous voulons exécuter la commande. Ce conteneur doit héberger une instance de MongoDB.

`mongodump`: C'est la commande spécifique à MongoDB pour sauvegarder une base de données.

`--uri=mongodb://root:admin@localhost:27017`: C'est l'adresse pour se connecter à MongoDB. "root" est le nom d'utilisateur et "admin" est le mot de passe pour se connecter à MongoDB. "localhost:27017" spécifie l'emplacement et le port de MongoDB.


`--authenticationDatabase=admin`: Cette option spécifie la base de données utilisée pour l'authentification, qui est dans ce cas admin.

`--db="db_mflix"`: Cela spécifie la base de données à sauvegarder, qui est db_mflix.

`--gzip`: Cette option indique que nous voulons compresser la sauvegarde.

`--archive=./backupdb/mongodump-db_mflix.gz`: C'est l'emplacement et le nom du fichier dans lequel nous voulons stocker la sauvegarde. L'option `--archive` spécifie le chemin de l'archive de sauvegarde, ici `./backupdb/mongodump-db_mflix.gz`

Voici le fichier « gzipé » qui prend peu de place :

| | | | |
|------------------------------------------------------------------------------------------------------------------------|------------------|------------|-----------|
|  <code>mongodump-db_mflix.gz</code> | 15.03.2024 15:25 | Fichier GZ | 13 980 Ko |
|------------------------------------------------------------------------------------------------------------------------|------------------|------------|-----------|

Maintenant, voici sans le `--gzip` :

| | | | |
|------------------------------------------------------------------------------------------------------------------------|------------------|------------|-----------|
|  <code>mongodump-db_mflix.gz</code> | 15.03.2024 15:26 | Fichier GZ | 45 106 Ko |
|------------------------------------------------------------------------------------------------------------------------|------------------|------------|-----------|

L'option `--gzip` permet de compresser le fichier et donc de prendre moins de place.

2.1.2 Restore (restauration)

Pour restaurer la base de données, j'ai utilisé la commande mongorestore. Voici la commande :

```
docker exec -i mongo mongorestore --uri=mongodb://root:admin@localhost:27017 --gzip --authenticationDatabase=admin --drop --archive=./backupdb/mongodump-db_mflix.gz
```

docker exec: Cette commande est utilisée pour exécuter une commande à l'intérieur d'un conteneur Docker.

-i: Cette option indique que la commande sera exécutée de manière interactive, utilisant l'entrée standard.

mongo: C'est le nom du conteneur Docker dans lequel nous voulons exécuter la commande. Ce conteneur doit héberger une instance de MongoDB.

mongorestore: C'est la commande spécifique à MongoDB pour restaurer une base de données à partir d'une sauvegarde.

--uri=mongodb://root:admin@localhost:27017: C'est l'URI de connexion à MongoDB. Il spécifie l'hôte (localhost) et le port (27017) ainsi que le nom d'utilisateur (root) et le mot de passe (admin) pour se connecter à la base de données.

--authenticationDatabase=admin: Cette option spécifie la base de données utilisée pour l'authentification.

--drop: Cette option indique à MongoDB de supprimer toutes les collections existantes dans la base de données cible avant de restaurer à partir de la sauvegarde.

--gzip: Cette option indique que l'archive de sauvegarde est compressée avec Gzip.

--archive=./backupdb/mongodump-db_mflix.gz: C'est l'emplacement et le nom du fichier dans lequel nous voulons stocker la sauvegarde. L'option **--archive** spécifie le chemin de l'archive de sauvegarde, qui est **./backupdb/mongodump-db_mflix.gz**.