

P/

# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB

---

## **P/3**

QU'EST CE QUE MONGODB ?

## **P/6**

MISE EN PLACE D'UNE BASE DE  
DONNÉES MONGODB

## **P/11**

INTERROGATIONS D'UNE BASE  
MONGODB AVEC COMPASS

## **P/19**

UN EXEMPLE DE CODE  
UTILISANT MONGODB

## **P/20 CRÉDITS**

---



### Au préalable

Pour acquérir cette compétence, vous avez besoin de connaître les systèmes de gestion de bases de données relationnels (pour bien comprendre cette approche qui est différente) et le **JSON** (Javascript Object Notation)



### De l'aide

La documentation est essentielle pour bien comprendre le fonctionnement et le développement. Comme tout bon développeur, ayez le réflexe d'avoir la documentation de MongoDB ou encore le site des communautés à disposition

[Site officiel](#)

[Documentation sur les bases de données NoSQL](#)

## QU'EST CE QUE MONGODB ?



### Définition

MongoDB est un Système de Gestion de Base de Données orienté **documents**. Il fait partie de la famille des systèmes **NoSQL**. MongoDB permet de manipuler des objets structurés au format **BSON** (**JSON** binaire), sans schéma prédéterminé. En d'autres termes, des clés peuvent être ajoutées à tout moment « à la volée », sans reconfiguration de la base.

Les données prennent la forme de **documents** enregistrés eux-mêmes dans des **collections**, une collection contenant un nombre quelconque de documents. Les collections sont comparables aux tables, et les documents aux enregistrements des bases de données relationnelles. Contrairement aux bases de données relationnelles, les champs d'un enregistrement sont libres et peuvent être différents d'un enregistrement à un autre au sein d'une même collection. Le seul champ commun et obligatoire est le champ de clé principale ("id"). Par ailleurs, MongoDB ne permet ni les requêtes très complexes standardisées, ni les JOIN, mais permet de programmer des requêtes spécifiques en JavaScript.

Ce système de gestion de base de données est utilisé notamment par eBay, PagesJaunes, SourceForge.

Source : Wikipedia



### Un exemple de base de données MongoDB

Nous disposons de cette structure de base données.

ID	Nom	Prénom	Âge
6	DUMOND	Jean	43
7	PELLERIN	Franck	NULL
8	MARTIN	Emile	62
9	KING	NULL	51

Dans notre base de données MongoDB, elle sera écrite de cette manière dans une collection "personnes".

```
{
  "_id": ObjectId("4efa8d2b7d284dad101e4bc7"),
  "Nom": "DUMOND",
  "Prénom": "Jean",
  "Âge": 43
},

{
  "_id": ObjectId("4efa8d2b7d284dad101e4bc8"),
  "Nom": "PELLERIN",
  "Prénom": "Franck",
  "Adresse": "1 chemin des Loges",
  "Ville": "VERSAILLES"
},

{
  "_id": ObjectId("4efa8d2b7d284dad101e4bc9"),
  "Nom": "KING",
  "Âge": "51",
  "Adresse": "36 quai des Orfèvres",
  "Ville": "PARIS"
}
```

Les documents d'une collection MongoDB peuvent comporter des champs différents (note : le champ "\_id" est un champ obligatoire, généré et ajouté par MongoDB, c'est un index unique qui permet d'identifier le document).

Dans un document, des champs peuvent être ajoutés, supprimés, modifiés et renommés à tout moment. Contrairement aux bases de données relationnelles, il n'y a pas de schéma prédéfini. La structure d'un document est très simple et se compose de paires clef/valeur à la manière des tableaux associatifs, la clef est le nom du champ, la valeur son contenu (voir le format JSON). Les deux sont séparés d'un signe deux-points ":" comme le montre l'exemple ci-dessus. Une "valeur" peut être un nombre ou du texte, mais aussi une donnée binaire (comme une image) ou une collection d'autres paires clefs/valeurs comme le montre l'exemple ci-dessous :

```
{
  "_id": ObjectId("4efa8d2b7d284dad101e4bc7"),
  "Nom": "PELLERIN",
  "Prénom": "Franck",
  "Âge": 29,
  "Adresse":
    {
      "Rue" : "1 chemin des Loges",
      "Ville": "VERSAILLES"
    }
}
```

Ici sont imbriqués des documents : le champ "Adresse" contient un document de deux champs : "Rue" et "Ville".

Certaines opérations sur les champs telles que l'incrémentations peuvent être effectuées de façon atomique, c'est-à-dire sans déplacement ni copie du document.

### MongoDB et les langages de programmation

Mongo DB fournit des pilotes pour la plupart des langages de programmation.

[Les pilotes MongoDB par langage de programmation](#)

# MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB

## Installation de MongoDB Server Community Edition

MongoDB (Server) est à télécharger et à installer : <https://www.mongodb.com/download-center/community>

Les données sont stockés dans de le répertoire "data" du dossier d'installation de MongoDB.  
Ce chemin peut être modifié en éditant le fichier mongod.cfg dans le répertoire d'installation ou par une ligne de commande :

**mongod --dbpath=*nouveau chemin***

(ceci prendra effet après redémarrage du service MongoDB)

## Installation de MongoDB Compass Community

MongoDB Compass est un outil graphique permettant de faire des opérations sur les bases de données MongoDB (créer des bases, des collections, des documents,...)

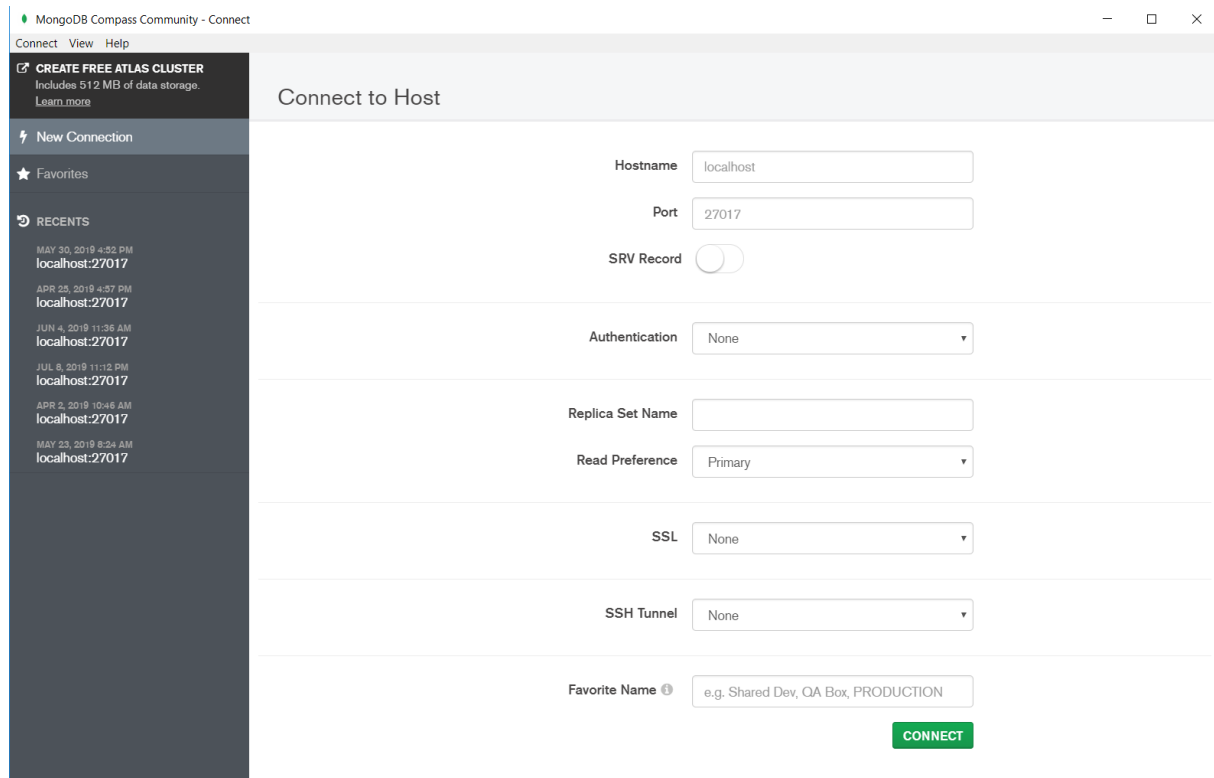
MongoDB Compass est à télécharger et installer : <https://www.mongodb.com/download-center/compass>

Toutes ces opérations sont possibles aussi en lignes de commande

## Visualisation des données avec Compass

Avec compass, on va se connecter à notre serveur MongoDB (serveur : localhost, port MongoDB par défaut : 27017).

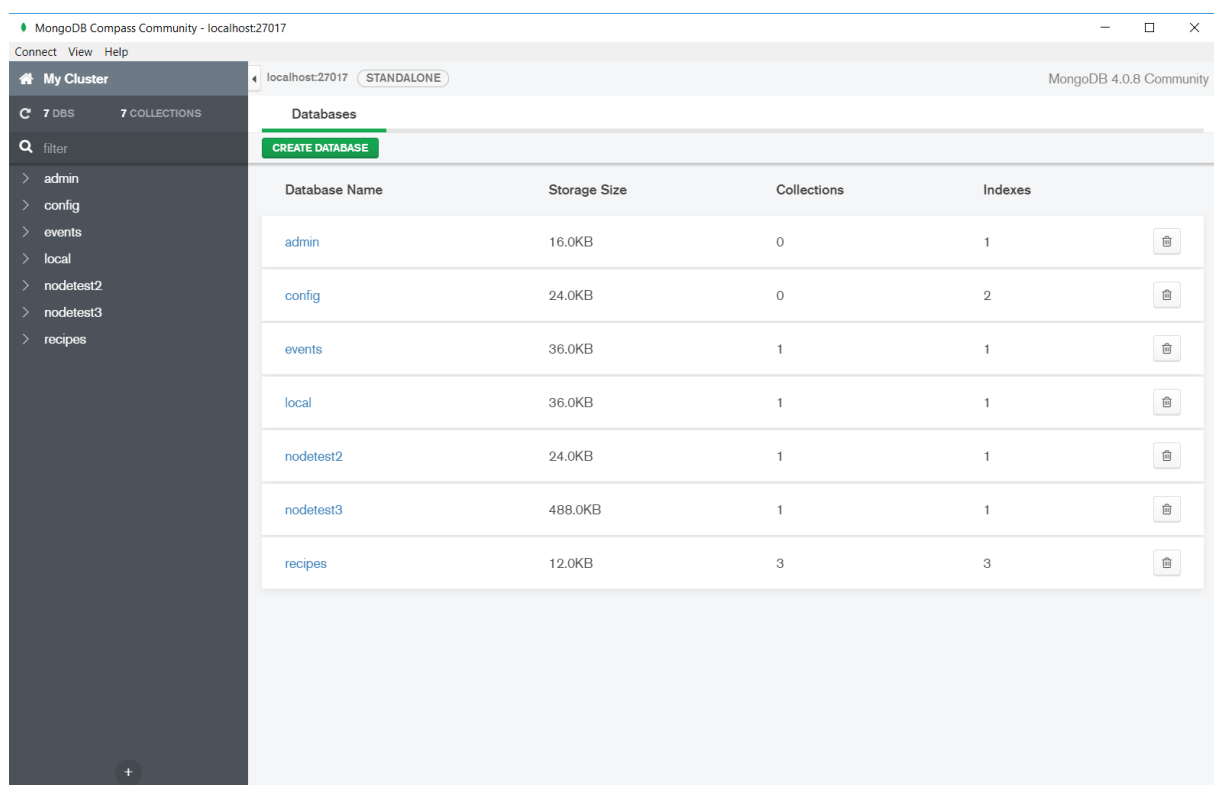
# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB



The screenshot shows the 'Connect to Host' dialog in MongoDB Compass. The left sidebar contains links for 'CREATE FREE ATLAS CLUSTER', 'New Connection', 'Favorites', and 'RECENTS'. The main area has fields for 'Hostname' (localhost), 'Port' (27017), 'SRV Record' (toggle), 'Authentication' (None), 'Replica Set Name', 'Read Preference' (Primary), 'SSL' (None), 'SSH Tunnel' (None), and 'Favorite Name' (e.g. Shared Dev, QA Box, PRODUCTION). A green 'CONNECT' button is at the bottom right.

Field	Value
Hostname	localhost
Port	27017
SRV Record	<input type="checkbox"/>
Authentication	None
Replica Set Name	
Read Preference	Primary
SSL	None
SSH Tunnel	None
Favorite Name	e.g. Shared Dev, QA Box, PRODUCTION

Après la connexion, on peut voir les bases de données qui sont présentes et en créer de nouvelles.

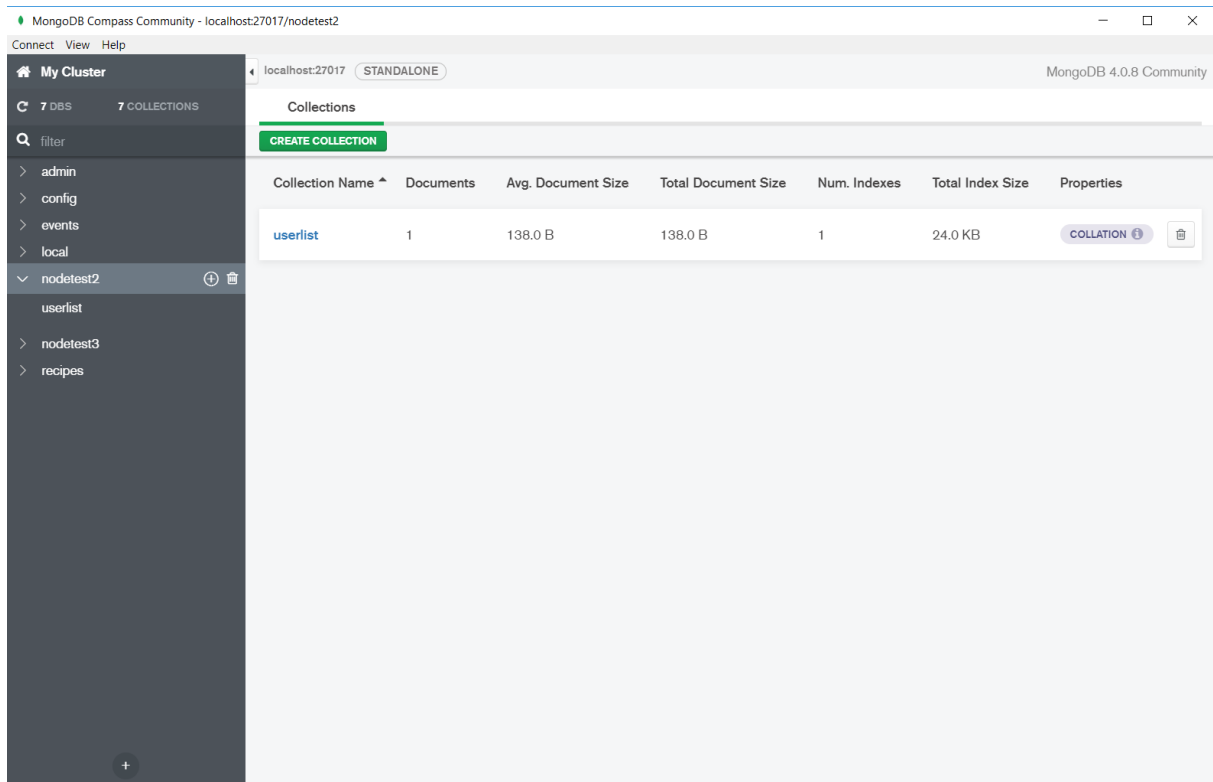


The screenshot shows the 'Databases' view in MongoDB Compass after connecting to localhost:27017. The left sidebar shows a tree view of databases and collections. The main area displays a table of databases with columns: Database Name, Storage Size, Collections, and Indexes. A green 'CREATE DATABASE' button is at the top left of the table.

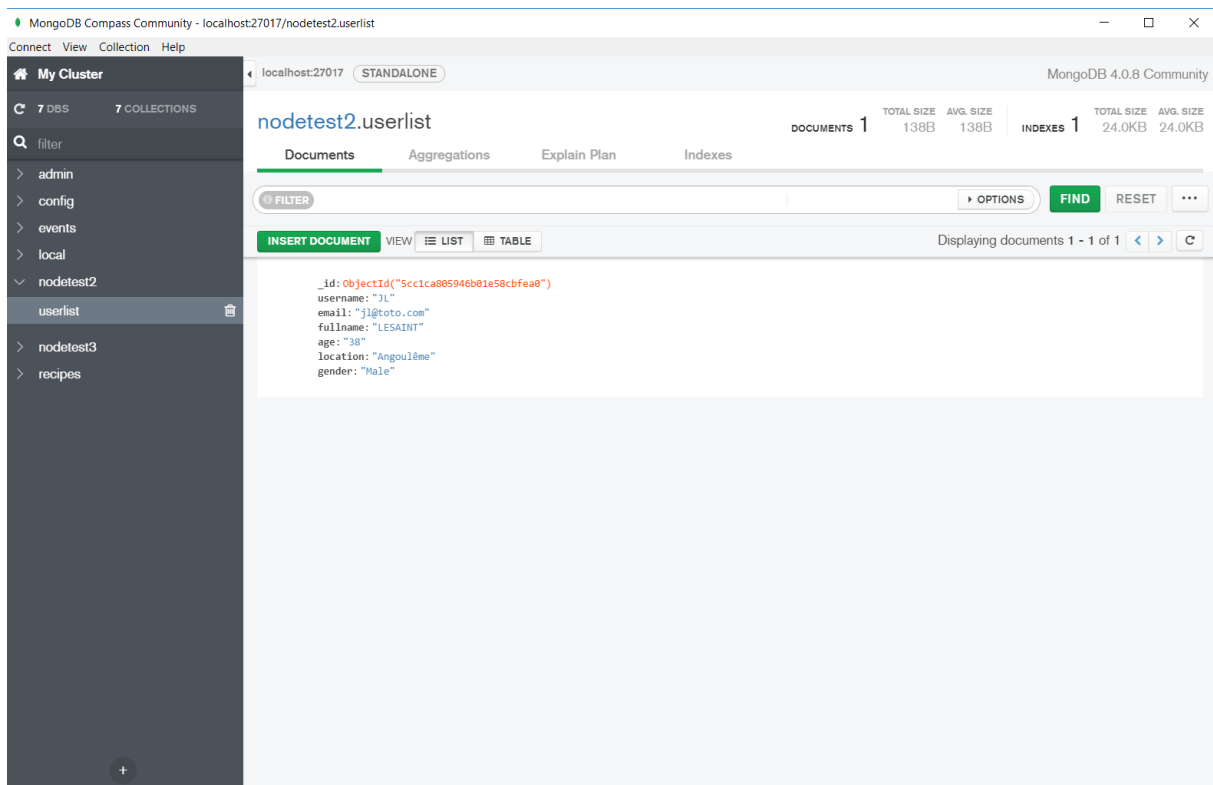
Database Name	Storage Size	Collections	Indexes
admin	16.0KB	0	1
config	24.0KB	0	2
events	36.0KB	1	1
local	36.0KB	1	1
nodetest2	24.0KB	1	1
nodetest3	488.0KB	1	1
recipes	12.0KB	3	3

On va à l'intérieur d'une créée, on peut visualiser les collections et en créer de nouvelles.

# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > **MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB**



On va à l'intérieur d'une collection, on peut visualiser les documents et en créer de nouveaux.



À nous de jouer, on va intégrer une base

On a récupéré une base de données de films que l'on va intégrer à notre serveur.

Ce document zip (cf. *movies.zip*) contient le fichier movies.json.



Avec Compass, on crée une nouvelle base de données "movies" avec une collection "movies"

Create Database

Database Name

movies

Collection Name

movies

☐ Capped Collection ⓘ

☐ Use Custom Collation ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL CREATE DATABASE

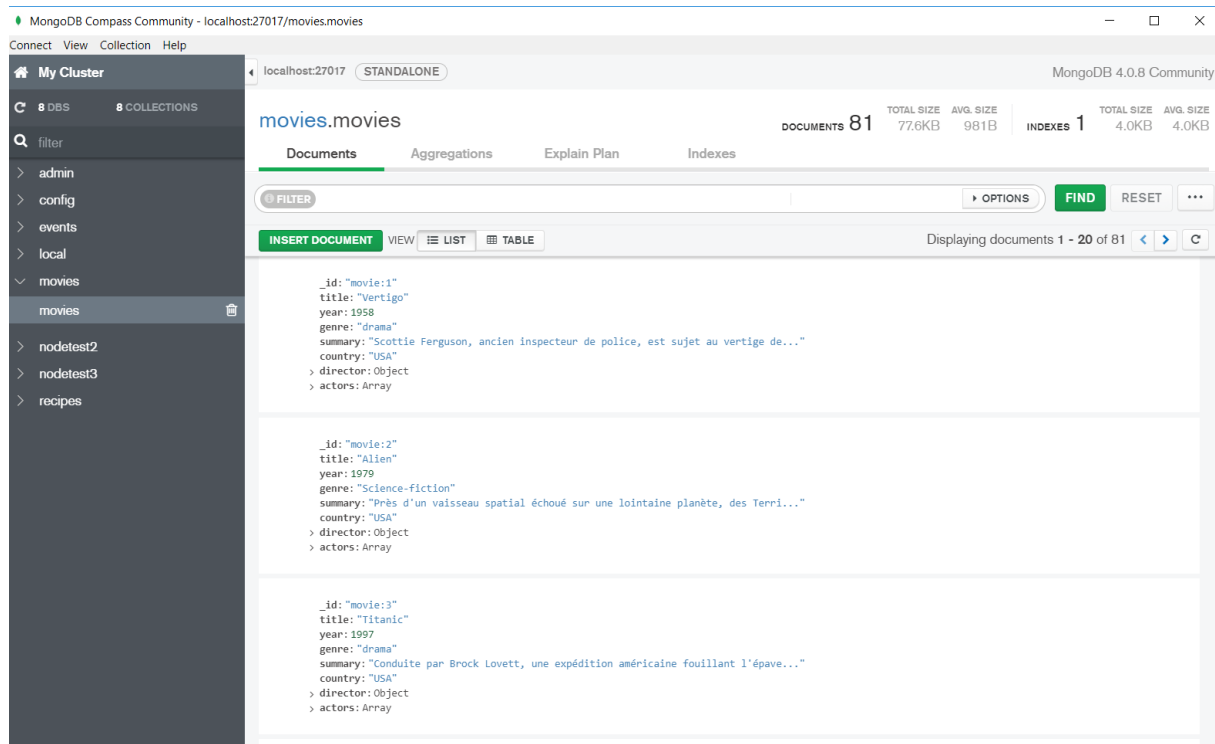
Après avoir créé cette base, on pointe dessus et, dans le menu "Collections", on clique sur "Import Data"

On va chercher notre fichier JSON que l'on a enregistré au préalable pour l'importer. Si on passe par un validator JSON, le contenu JSON n'est pas valide.

Il répond aux caractéristiques d'importation avec Compass : <https://docs.mongodb.com/compass/master/import-export/>

Nous avons importé nos movies :

# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB



Nous allons faire maintenant des interrogations à notre base de données

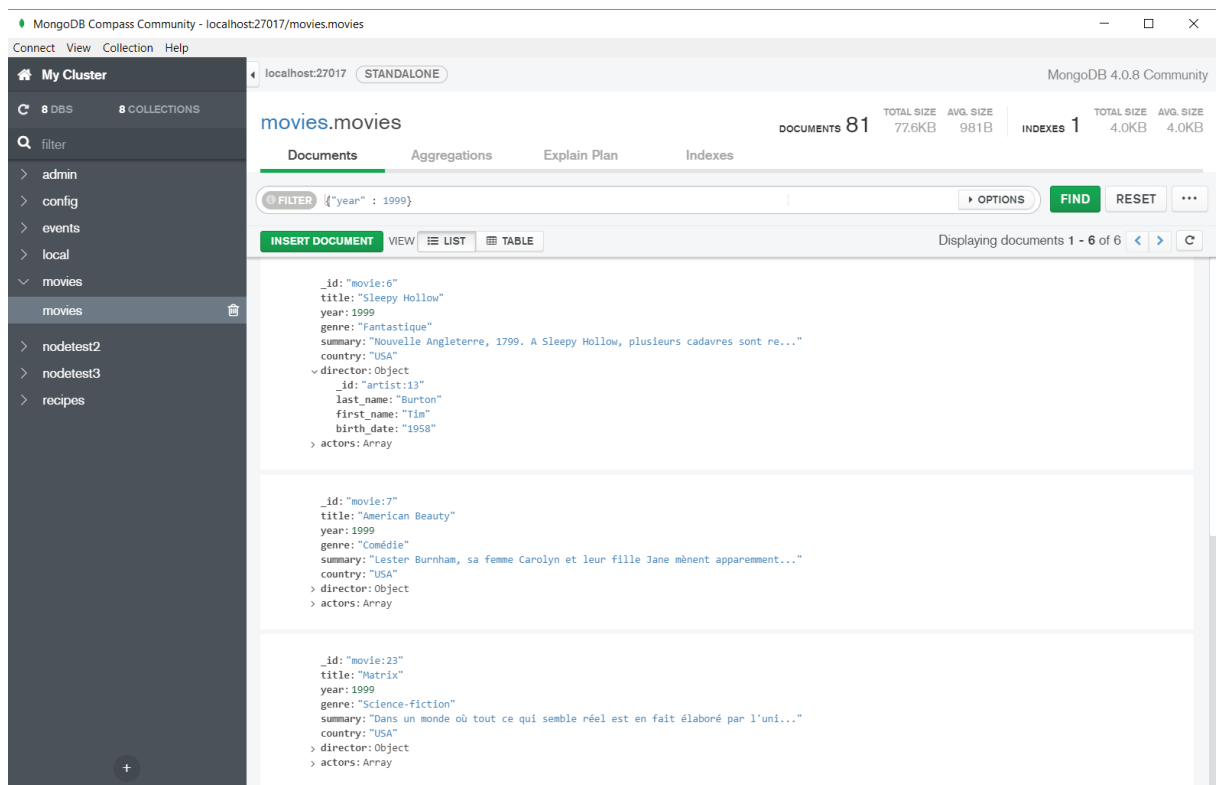
# INTERROGATIONS D'UNE BASE MONGODB AVEC COMPASS

La base de données utilisée "movies"

Dans le dernier chapitre nous avons intégré la base de données "movies". C'est celle-ci que nous allons utiliser.

## Les films de l'année 1999

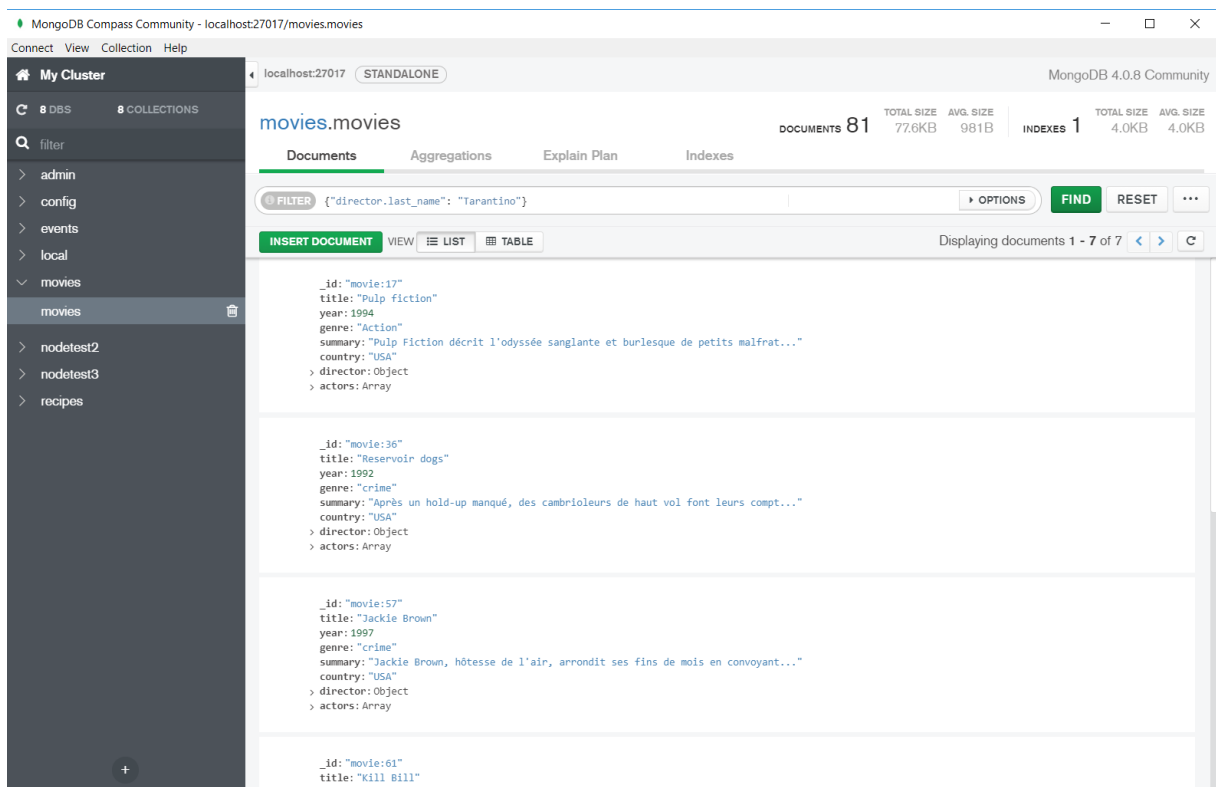
On va mettre dans "filter" le critère de sélection `{ year : 1999 }` et on fait **FIND**



## Les films de Tarantino

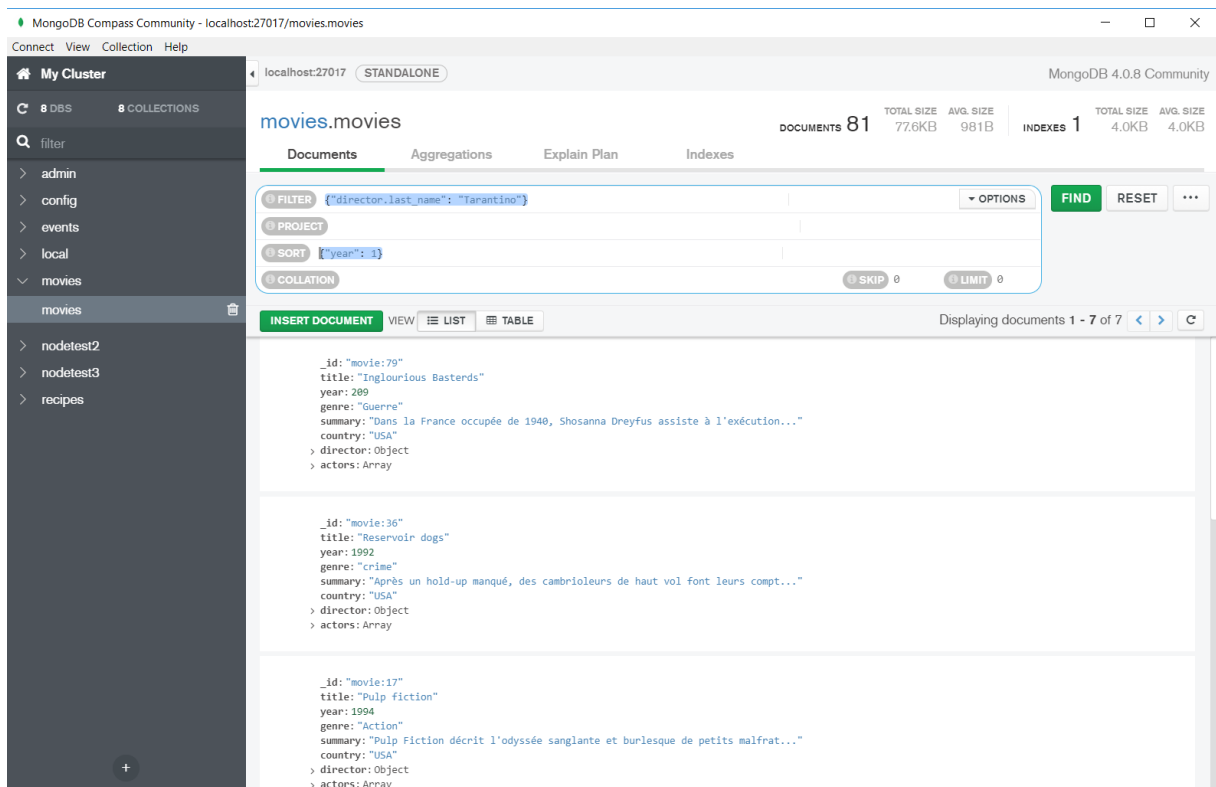
On va mettre dans "filter" le critère de sélection `{ "director.last_name": "Tarantino" }` et on fait **FIND**

# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > INTERROGATIONS D'UNE BASE MONGODB AVEC COMPASS



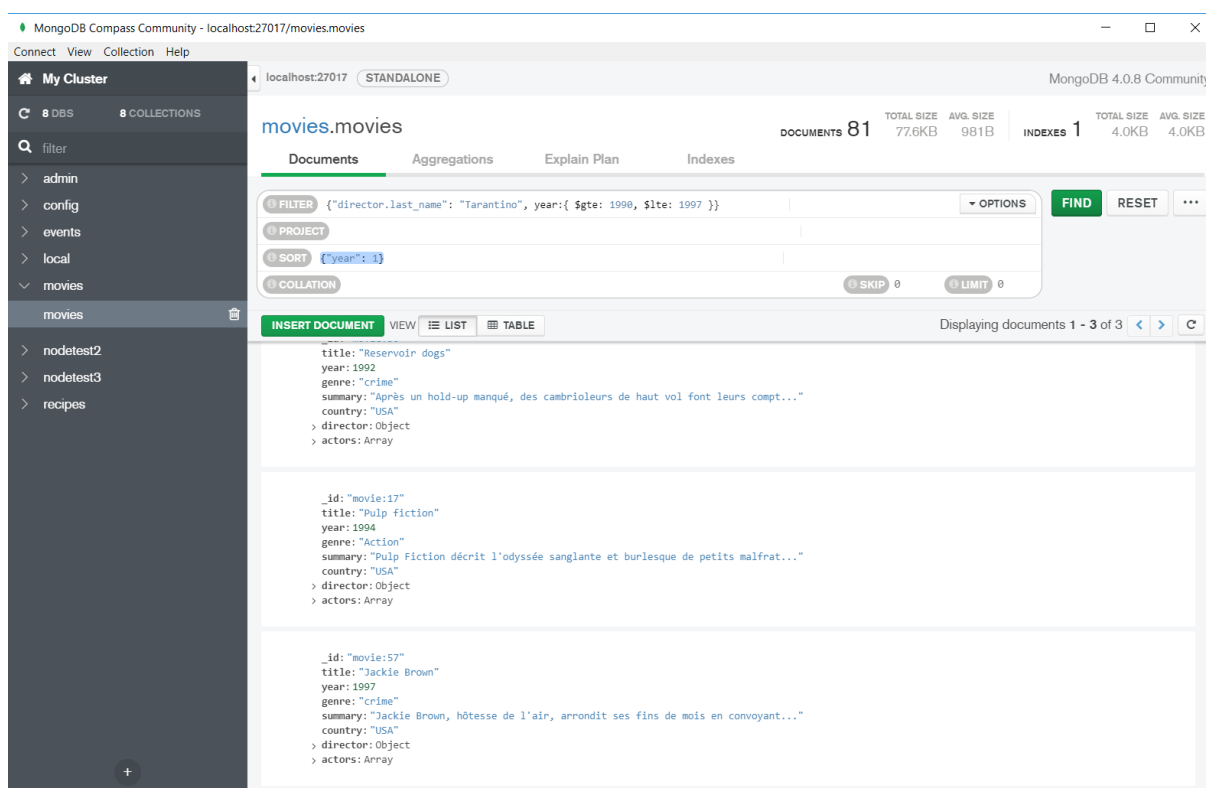
## Les films de Tarantino triés par année

On va mettre dans "filter" le critère de sélection `{"director.last_name": "Tarantino"}` et dans "sort" (options) `{"year": 1}` pour avoir un tri ascendant. Il suffira de mettre "-1" pour obtenir le tri descendant.



Les films de Tarantino entre l'année 1990 (incluse) et l'année 2000 (incluse) triés par année

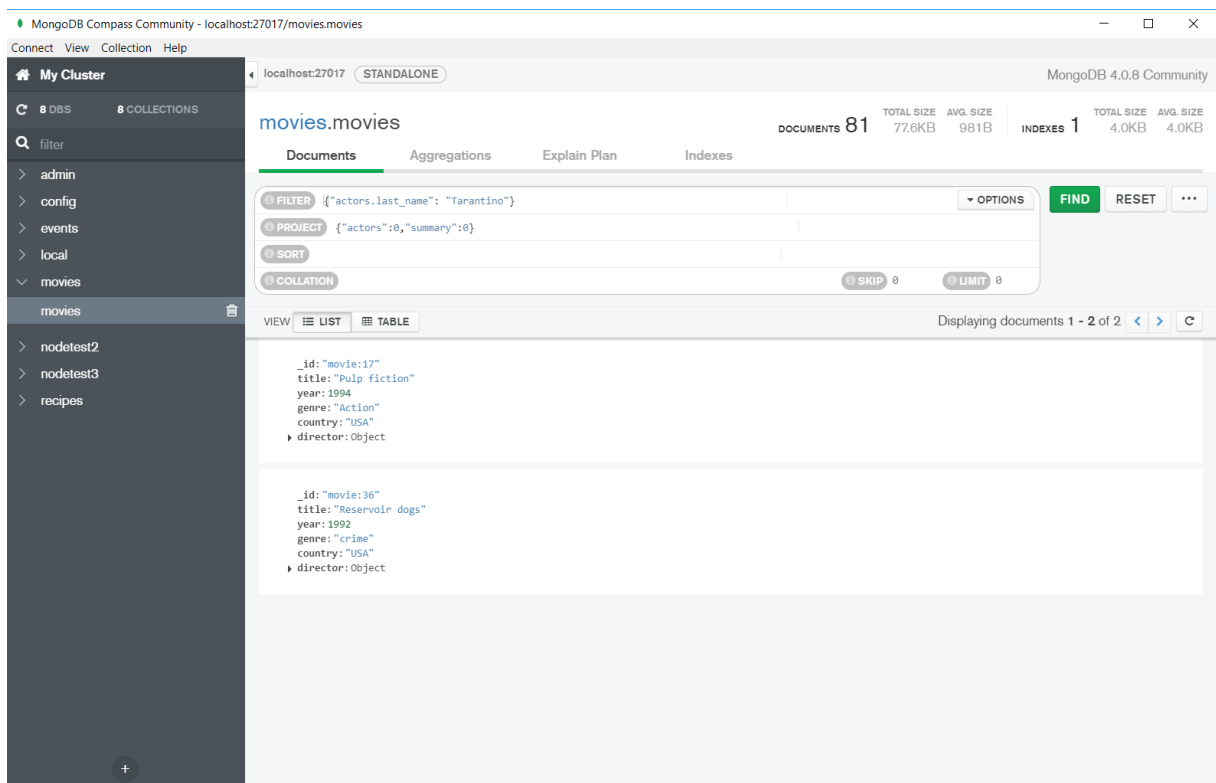
On va mettre dans "filter" le critère de sélection `{"director.last_name": "Tarantino", year: { $gte: 1990, $lte: 2000 }}` et dans "sort" (options) `{"year": 1}`



Les films où Tarantino a été acteur en n'affichant pas les informations des acteurs et de la description sur les films

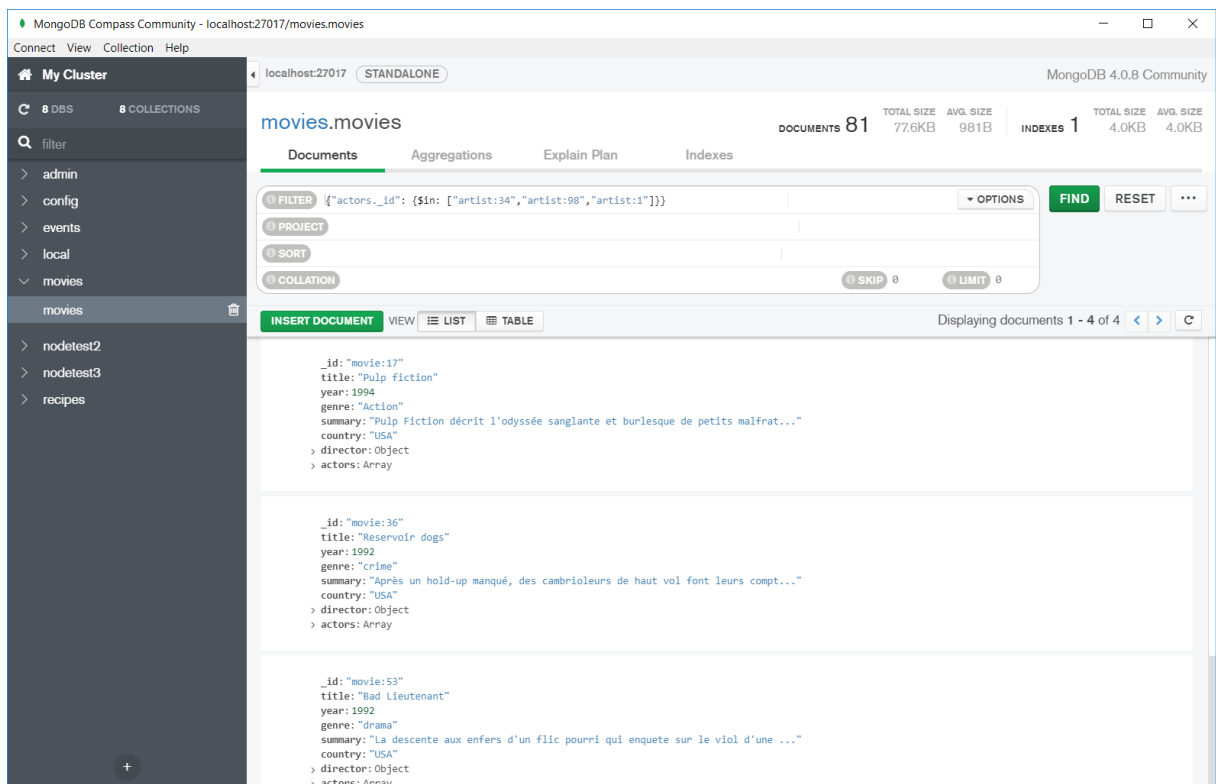
On va mettre dans "filter" le critère de sélection `{"actors.last_name": "Tarantino"}` et dans "project" (options) `{"actors":0,"summary":0}` (0 : ne pas afficher)

# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > INTERROGATIONS D'UNE BASE MONGODB AVEC COMPASS



Les films dans lesquels joue au moins un des artistes dans une liste (on suppose que l'on connaît l'identifiant).

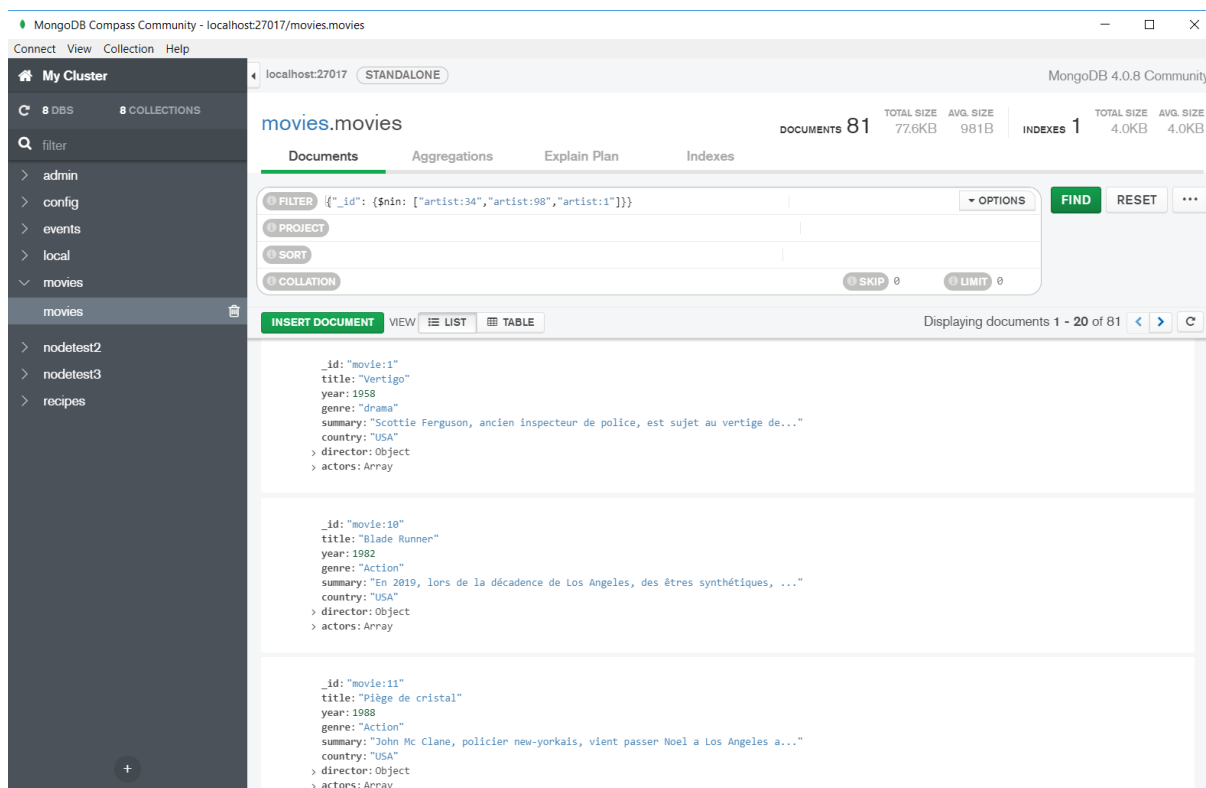
On va mettre dans "filter" le critère de sélection `{ "actors._id": { $in: ["artist:34", "artist:98", "artist:1"] } }`





Les films dans lesquels ne joue pas au moins un des artistes dans une liste (on suppose que l'on connaît l'identifiant).

On va mettre dans "filter" le critère de sélection `{"_id": {"$nin": ["artist:34","artist:98","artist:1"]}}`



Les films qui n'ont pas description

On va mettre dans "filter" le critère de sélection `{"summary": {"$exists": false}}`

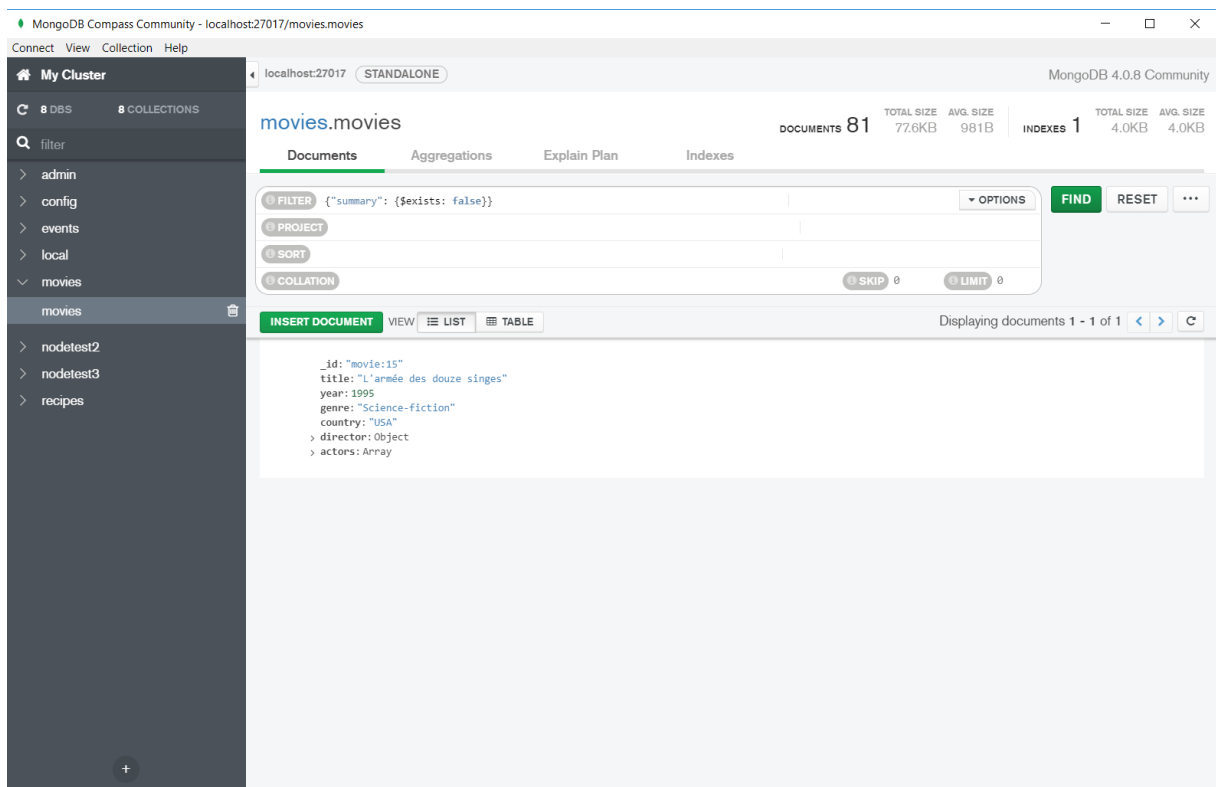
On remarque qu'aucun film dans notre base de films n'a pas de description.

Nous allons modifier le document avec l' "\_id" : "movie :15", "L'armée des douze singes" puisque celui-ci a une description dont la valeur est nulle.

Il suffit de supprimer la clé "summary" pour que celle-ci soit enlevé du document.

Nous relançons notre sélection.

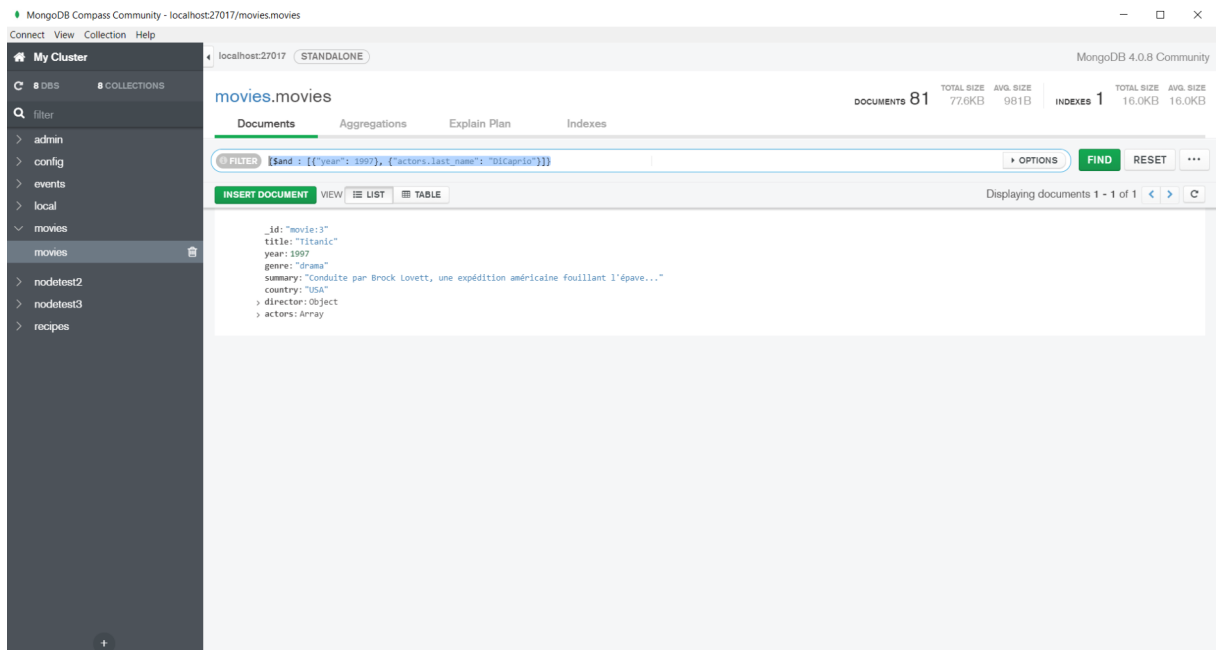
# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > INTERROGATIONS D'UNE BASE MONGODB AVEC COMPASS



## Les films tournés avec Leonardo DiCaprio en 1997

On va mettre dans "filter" le critère de sélection `{ $and : [ {"year": 1997}, {"actors.last_name": "DiCaprio"} ] }`

Nous utilisons ici un "et" pour faire du multi-critères

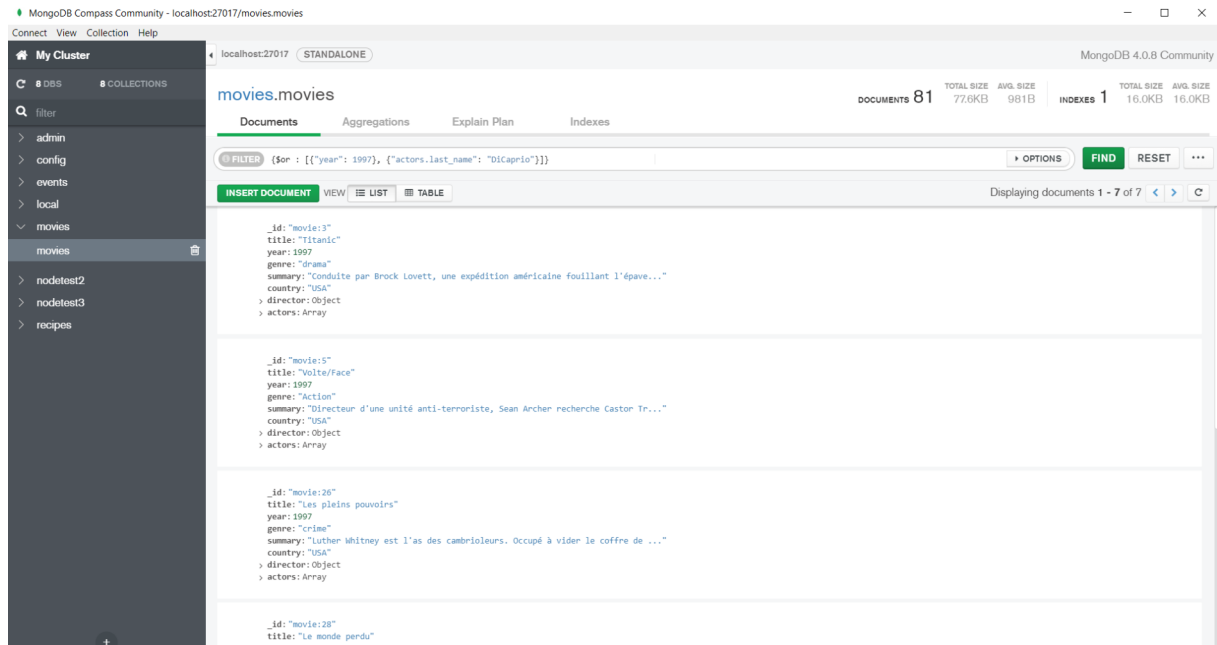




## Les films parus en 1997 ou avec Leonardo DiCaprio

On va mettre dans "filter" le critère de sélection `{ $or : [ {"year": 1997}, {"actors.last_name": "DiCaprio"} ] }`

Nous utilisons ici un "ou" pour faire du multi-critères



## Le nombre de films de "Tarantino"

On va utiliser les fonctions d'aggrégation **\$match** et **\$count** pour pouvoir se faire.

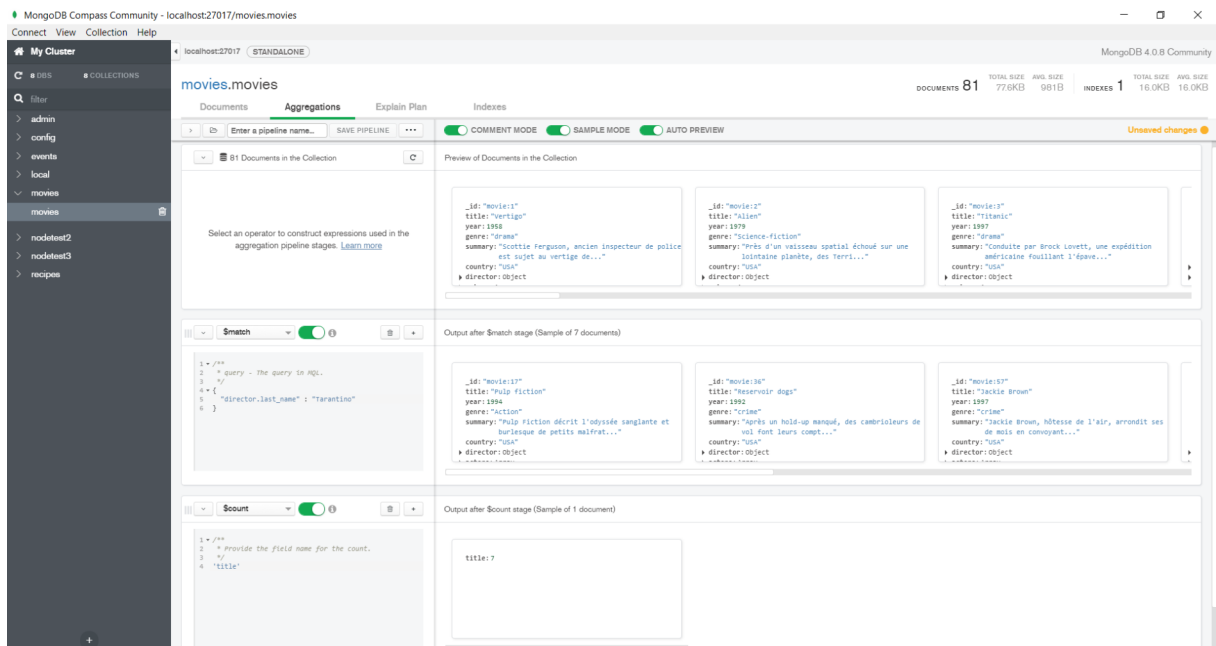
**\$match**

```
{  
  "director.last_name" : "Tarantino"  
}
```

**\$count**

'title'

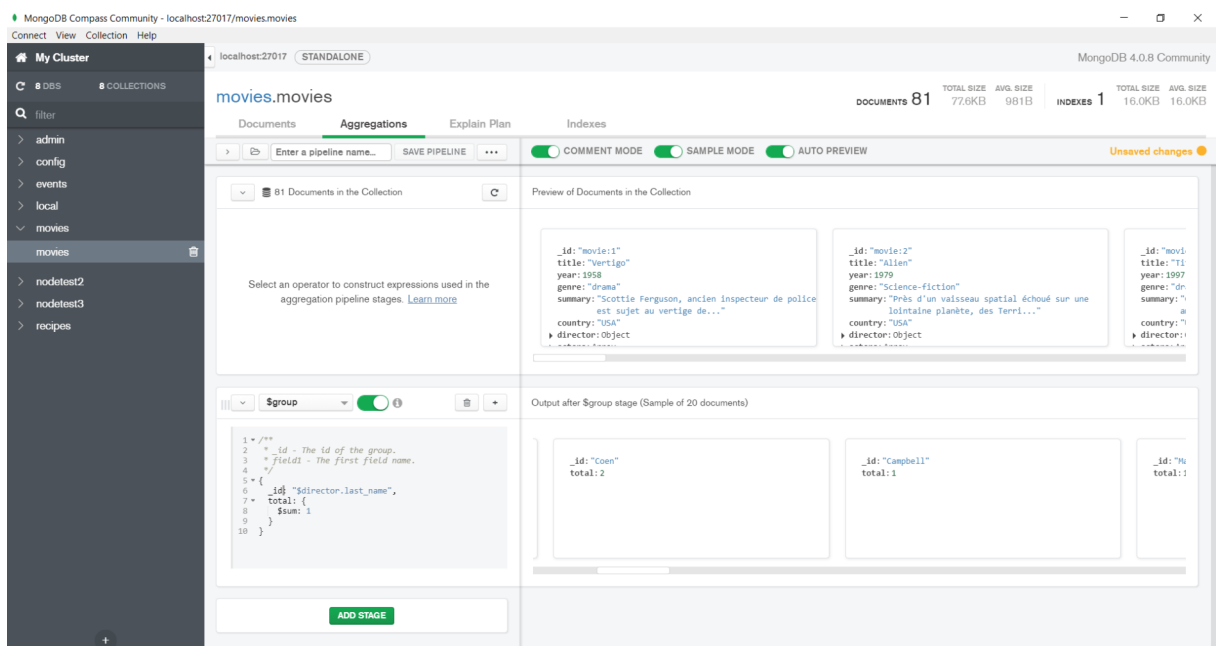
# PRINCIPE GÉNÉRAL ET MISE EN PLACE D'UNE BASE DE DONNÉES MONGODB > INTERROGATIONS D'UNE BASE MONGODB AVEC COMPASS



## Le nombre de films par directeur

On va utiliser la fonction d'agrégation **\$group** pour pouvoir se faire.

```
{
  _id: "$director.last_name",
  total: {
    $sum: 1
  }
}
```



## UN EXEMPLE DE CODE UTILISANT MONGODB

Le code présenté a été écrit en javascript. Il est lancé sur un Node JS avec les bibliothèques appropriées pour interroger du MongoDB (MongoDB et Monk)

*// Database*

```
var mongo = require('mongodb');  
var monk = require('monk');  
var db = monk('localhost:27017/movies');  
var collection = db.get('movies');  
collection.find({}, {}, function(e, docs){  
  console.log(docs);  
});
```

# CRÉDITS

**OEUVRE COLLECTIVE DE L'AFPA**  
Sous le pilotage de la Direction de l'ingénierie

© **AFPA**

**Reproduction interdite**

*Article L 122-4 du code de la propriété intellectuelle.*

« Toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction par un art ou un procédé quelconques. »