|  |
| --- |
| rectangle transparent colorérectangle coloréUne image contenant texte  Description générée automatiquementrectangle transparent coloré  RAPPORT  TP MongoDB |

|  |
| --- |
| NoSQL - DIA2  MANNAI Hasna ()  CHENIK Yassine ()  BOUCHIBA Emine () |

Table des matières

Importation du ficher json dans le container Mongo 2

Simple Queries 3

Complex Queries 9

Hard Queries 11

|  |
| --- |
| Importation du json dans MongoDB |

Dézipper le fichier JSON :

Utilisation de WinRar pour dézipper le fichier et le mettre dans notre répertoire de travail.

Lancement de Docker :

Une fois Docker lancé et le container Cassandra lancé :

* On lance le terminal Windows dans lequel on tape la ligne de commande suivante afin d’importer le fichier JSON dans Docker :
  + Docker cp "C:\Users\emine\Documents\Ecole\A4\S8\Advanced\_topics\_in\_NoSql\_databases\TP\_MongoDB\companies2.json" MongoDB:/
* On importe le fichier json dans le container MongoDB :
  + mongoimport --db DB\_Companies --collection TD --file companies2.json
  + *Ici, on utilise "--file" et non "--jsonArray" parce que le ficher JSON companies2.json ne comporte pas une liste dans laquelle est contenu nos objets JSON comme dans celui du TD. Ici, les objets sont inscrits seuls sans être dans une liste.*
* On entre dans le terminal Cassandra :
  + mongosh (dans le terminal Docker)
* On se met dans la database :
  + USE DB\_Companies;

|  |
| --- |
| Simple Queries |

1. Get all the companies that have a category\_code=”nanotech” :

* db.TD.find({"category\_code":"nanotech"},{"category\_code":1,"name":1})
* *Ici, on veut afficher les entreprises qui sont dans le domaine de la nanotechnologie en affichant le "name", le "category\_code" ainsi que l’id.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Simple Queries |

1. Get all the Companies that founded in 2008 :

* db.TD.find({"founded\_year":2008},{"founded\_year":1,"founded\_month":1,"founded\_day":1,"name":1})
* *Ici, on veut afficher les entreprises qui ont été fondés en 2008. en affichant le "name", le "founded\_year", le "founded\_month", le "founded\_day" ainsi que l’id.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Simple Queries |

1. Get the number of Companies whith more than 100 employees :

* db.TD.find({"number\_of\_employees":{"$gte":100}},{"name":1,"number\_of\_employees":1})
* *Ici, on veut afficher les entreprises qui ont plus de 100 employés en affichant le "name", le "number\_of\_employees" ainsi que l’id.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Simple Queries |

1. Get all companies founded in May 2019 :

* db.TD.aggregate([{"$match":{"category\_code":"security"}},{"$unwind":"$funding\_rounds"},{"$group":{"\_id":"$name","total\_raised\_amount":{"$sum":"$funding\_rounds.raised\_amount"}}},{"$sort":{"total\_raised\_amount":-1}}])
* *Ici nous cherchons à afficher le montant total amassé par les entreprises qui travaillent dans le domaine de la sécurité.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Simple Queries |

1. Get total money raised for each domain :

* db.TD.aggregate([{"$match":{"category\_code":{"$ne":null}}},{"$unwind":"$funding\_rounds"},{"$group":{"\_id":{"category\_code":"$category\_code","name":"$name"},"total\_raised\_amount":{"$sum":"$funding\_rounds.raised\_amount"}}},{"$group":{"\_id":"$\_id.category\_code","total\_raised\_amount":{"$sum":{"$convert":{"input":"$total\_raised\_amount","to":"double"}}}}},{"$sort":{"total\_raised\_amount":-1}}])
* *Ici, nous cherchons à afficher le montant total amassé par les entreprises du même domaine.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Simple Queries |

1. Get all companies that starts with an “F” :

* db.TD.find({"name":{"$regex":"^F"}},{"name":1})
* *Ici, on cherche à afficher toutes les entreprises qui commencent par la lettre F. Pour ce faire, on utilise l’option $regex qui nous permet de reconnaître un motif dans une chaîne de caractères et on lui donne comme argument ‘^F’. De et on affiche les noms des entreprises correspondantes.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Complex Queries |

1. Get the number of employees of each Company that have as category\_code =”software” :

* db.TD.aggregate([{"$match":{"category\_code":"software"}},{"$group":{"\_id":"$name","category\_code":{"$first":"$category\_code"},"total\_employees":{"$sum":"$number\_of\_employees"}}},{"$sort":{"total\_employees":-1}}])
* *Ici, on cherche à avoir le nombre d'employés travaillant au sein d’entreprises dans le domaine du Software et on affiche le nom de l’entreprise, sa category\_code et le nombre d’employés.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Complex Queries |

1. Get the number of products grouped by companies name :

* db.TD.aggregate([{"$group":{"\_id":"$name","total\_products":{"$sum":{"$size":"$products"}}}},{"$sort":{"total\_products":-1}}])
* *Ici, on veut afficher le nombre de produits provenant de chacune des entreprises.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*

|  |
| --- |
| Hard Queries |

1. Get the total amount of money raised by companies founded in each year, sorted in descending order of the total amount raised :

* db.TD.aggregate([{"$match":{"founded\_year":{"$ne":null}}},{"$unwind":"$funding\_rounds"},{"$group":{"\_id":{"year":"$founded\_year","name":"$name"},"total\_raised\_amount":{"$sum":"$funding\_rounds.raised\_amount"}}},{"$group":{"\_id":"$\_id.year","total\_raised\_amount":{"$sum":{"$toDouble":"$total\_raised\_amount"}}}},{"$sort":{"total\_raised\_amount":-1}}])
* *Ici, on cherche à obtenir le montant total d'argent collecté par les entreprises fondées chaque année, trié par ordre décroissant du montant total collecté.*
* *Output :*

*Une image contenant texte

Description générée automatiquement*