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

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

Table des matières

Importation du ficher json dans le container Cassandra 2

Création des tables 3

Importation des données du fichier json 5

Simple Queries 1

Complex Queries 1

Hard Queries 1

|  |
| --- |
| Importation du json dans Cassandra |

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 :
  + Docker cp "C:\Users\emine\Documents\Ecole\A4\S8\Advanced\_topics\_in\_NoSql\_databases\TP\_Cassandra\companies2.json" Cassandra:/
* On entre dans le terminal Cassandra :
  + cqlsh (dans le terminal Docker)
* On lance la création d’un KeySpace :
  + CREATE KEYSPACE IF NOT EXISTS Companies2\_Cassandra WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication\_factor': 3 };
* On se met dans ce KeySpace :
  + USE Companies2\_Cassandra;

|  |
| --- |
| Création des tables |

Script de création des Types que l’on va utiliser :

* + relationshipsType :
    - CREATE TYPE IF NOT EXISTS relationshipsType (is\_past BOOLEAN,title TEXT,person  map<TEXT,TEXT>);
  + investmentsType :
    - CREATE TYPE IF NOT EXISTS investmentsType(company map<TEXT,TEXT>,financial\_org  map<TEXT,TEXT>, person map<TEXT,TEXT);
  + roundType :
    - CREATE TYPE IF NOT EXISTS roundType(id  INT , round\_code TEXT, source\_url TEXT, source\_description TEXT, raised\_amount INT, raised\_currency\_code TEXT, funder\_year INT, funder\_month INT, funder\_day INT, investments list frozen(investmentsType));

Script de création de la table Company :

CREATE TABLE  Company(\_id map<TEXT,TEXT>, name TEXT, permalink TEXT, crunchbase\_url TEXT, homepage\_url TEXT, blog\_url TEXT, blog\_feed\_url TEXT , twitter\_username TEXT, category\_code TEXT, number\_of\_employees INT, founded\_year INT, founded\_month INT, founded\_day INT, deadpooled\_year INT, deadpooled\_month INT, deadpooled\_day INT, deadpooled\_url TEXT, tag\_list TEXT, alias\_list TEXT, email\_address TEXT, phone\_number TEXT, description TEXT, created\_at TEXT, updated\_at TEXT, overview TEXT, total\_money\_raised TEXT

PRIMARY KEY (\_id, permalink)

);

ALTER TABLE Company WITH GC\_GRACE\_SECONDS = 0;

CREATE INDEX IF NOT EXISTS  company\_id ON Company(\_id) ;

CREATE INDEX IF NOT EXISTS  permalink\_id ON Company(permalink);

Script de création de la table Product :

CREATE TABLE Product (\_id map<TEXT,TEXT>,products list<map<TEXT,TEXT>>

PRIMARY KEY (\_id,permalink)

);

ALTER TABLE Product WITH GC\_GRACE\_SECONDS = 0;

CREATE INDEX IF NOT EXISTS  permalink\_id ON product(permalink) ;

|  |
| --- |
| Création des tables |

Script de création de la table Office\_Company :

CREATE TABLE Office\_Company(\_id map<TEXT,TEXT>,offices list<frozen<map<TEXT,TEXT>>>,

coord frozen<map<TEXT,frozen<map<TEXT,list<double>>>>>

PRIMARY KEY (\_id,zip\_code)

);

ALTER TABLE Office\_Company WITH GC\_GRACE\_SECONDS = 0;

Script de création de la table Relationships :

CREATE TABLE Relationships(\_id map<TEXT,TEXT>,relationships list frozen<relationshipsType>

PRIMARY KEY (\_id,) //faut ajouter une autre clé primaire

);

ALTER TABLE Relationships WITH GC\_GRACE\_SECONDS = 0;

Script de création de la table RoundTable :

CREATE TABLE  RoundTable(id map<TEXT,TEXT>,list frozen< roundType>);

ALTER TABLE RoundTable WITH GC\_GRACE\_SECONDS = 0;

|  |
| --- |
| Importation des données du fichier json |

|  |
| --- |
| Simple Queries |

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

* select \* from Companies where category\_code=”nanotech”;

1. Get all the Companies that founded in 2008 :

* select \* from Companies where founded\_year =2008 ALLOW FILTERING;

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

* select count(\*) from Companies where number\_of\_employees >100 ALLOW FILTERING;

Ou

* select count(\*) from Companies where token(\_id)>100;

1. Get all companies founded in May 2019 :

* select \* from Companies where founded\_year =2019 and founded\_month=5 ALLOW FILTERING;

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

* SELECT \* FROM companies WHERE name LIKE 'F%' ALLOW FILTERING;

1. Get all comapnies that raised more than 10 million dollars :

* SELECT \* FROM companies WHERE total\_money\_raised > 10000000 ALLOW FILTERING;

|  |
| --- |
| Complex Queries |

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

* select \_id,number\_of\_employees from Companies where category\_code =”software” group by\_id;

1. Get the product names that have a permalink =”ripcode-v4” :

* CREATE OR REPLACE FUNCTION project(key text, tab map) RETURNS NULL ON NULL INPUT RETURNS INT LANGUAGE Java AS 'return tab.get(key);';
* select project(‘name', products ) fromProduct WHERE(select project(‘permalink ', products ) =”ripcode-v4”) ALLOW FILTERING;

1. Give the product names thab belong to the campanie “5a5c533c942d09e481c15829” :

* select project(‘name', products ) fromProduct WHERE(select project(‘$oid’, \_id ) =”5a5c533c942d09e481c15829”) ALLOW FILTERING;

|  |
| --- |
| Hard Queries |