TP2 – NoSQL Cassandra

Restaurant Inspections

Manon GARDIN

Matias OTTENSEN

Alexandre GARNIER

Tiphaine KACHKACHI

# Chapter 1 – Create the database

## Files transfer

We drag and drop the restaurants.json into the files of our Cassandra container.

Une image contenant texte, logiciel, Icône d’ordinateur, nombre

Description générée automatiquement

## Create the keyspace

In the CLI, use the command :

CREATE KEYSPACE IF NOT EXISTS RESTO\_INSPEC

WITH REPLICATION =

{ 'class': 'SimpleStrategy', 'replication\_factor': 3 };

And then,

USE RESTO\_INSPEC;

Une image contenant texte, Police, capture d’écran, algèbre

Description générée automatiquement

## Create Tables

Let’s understand the JSON structure of our dataset with the structure of one insert :

[

{

"address":

{

"building": "1007",

"coord": {

"type": "Point",

"coordinates" : [-73.856077, 40.848447]

},

"street": "Morris Park Ave",

"zipcode": "10462"

},

"borough": "Bronx",

"cuisine": "Bakery",

"grades": [

{

"date": {

"$date": 1393804800000

},

"grade": "A",

"score": 2

},

{

"date": {

"$date": 1378857600000

},

"grade": "A",

"score": 6

},

{

"date": {

"$date": 1358985600000

},

"grade": "A",

"score": 10

},

{

"date": {

"$date": 1322006400000

},

"grade": "A",

"score": 9

},

{

"date": {

"$date": 1299715200000

},

"grade": "B",

"score": 14

}

],

"name": "Morris Park Bake Shop",

"restaurant\_id": "30075445"

},

INSERT2…

]

Visually, we can see that the first property address will need a full table containing building, coord, street and zipcode. In each coord, we have a type of coordinate and the X and Y coordinates. In order to not create another table, we can add them as different attributes.

Also the property grade is an array with different elements. We can conclude that we will need a grades table with the three attributes grade, date and score. One restaurant can have several grades, so we add the restaurant\_id attributes as the equivalent of a foreign key.

And finally the table restaurants that will have all the other attributes.

Create the tables in file CreaTable :

CREATE TABLE restaurants (

restaurant\_id text PRIMARY KEY,

name text,

borough text,

cuisine text

);

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

CREATE TABLE addresses (

address\_id text PRIMARY KEY,

building text,

street text,

zipcode text,

coord\_type text,

coord\_X float,

coord\_Y float

);

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

CREATE TABLE grades (

restaurant\_id text,

date timestamp,

grade text,

score int,

PRIMARY KEY (restaurant\_id, date)

);

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

Une image contenant texte, capture d’écran, Police, algèbre

Description générée automatiquementNow, we open TablePlus, and select the database we created.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

## Fixing Json file

We found out that the format of the Json is not correct, so we needed to do a script to correct the file. What was wrong in that file ?

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

The file lack the first and last '['.

We did the fixing\_json.py :

Une image contenant texte, capture d’écran, logiciel, affichage

Description générée automatiquement

In which we created a whole new file “restaurants\_fixed.json”. We added the '[' character and we join all the lines of the original files separated by a comma.

## Import the data

Then, we execute the data\_importation.py file in the Cassandra container.

In this code below, we setup the connection to the database :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

And then, we add the data with a query, for each table. (See in the data\_importation.py file)

Here is the command :

docker exec -it Cassandra python3 data\_importation.py

# Chapter 2 – Querying Cassandra