

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



**Báo Cáo Giữa Kỳ**

**Môn học:** Phân tích dữ liệu lớn

**Đề tài**: Apache Hive và Apache Cassandra

GVHD: Th.s Lê Thị Minh Châu

Sinh viên thực hiện:

Trần Công Tuấn Mạnh 19133035

Phan Hoàng Việt 17133072

Lê Ngọc Đoan 17133014

Tp. HCM, ngày 09 tháng 06 năm 2022

MỤC LỤC

[**I - Giới thiệu Cassandra**](#_sl4pbmtliaf2) **2**

[Cassandra là gì ?](#_307vdxucsvql) 2

[Đặc điểm về Cassandra](#_i061ufdwejue) 2

[2.1. Một số tính năng của Cassandra](#_ar9bqlvjffib) 2

[2.2. Một số điểm đáng chú ý của Cassandra](#_1ddta7u32wq4) 2

[Kiến trúc của Cassandra](#_gudqkf7luhnn) 3

[**II - Cài đặt Cassandra**](#_t4in9hr1l7d5) **5**

[**III - Cú pháp Cassandra**](#_so305v6yhfqk) **8**

[3.1 Cassandra Keyspace Operations](#_r4708hedq1l9) 8

[3.1.1. Tạo Keyspace:](#_45m0ldczuhqk) 8

[3.1.2. Alter Keyspace Cassandra:](#_dcoih7cddt1x) 9

[3.1.3. Drop Keyspace:](#_duzopgjn3bwp) 9

[3. 2. Cassandra Table](#_74mdp6ssxyil) 10

[3.2.1 Create Table](#_ugxv8pgwtf6s) 10

[3.2.2 Alter Table](#_fh05ud1aovor) 11

[3.2.3 Drop Table](#_wedkc329bfa9) 11

[3.2.4 Truncate Table](#_ehwc94d7d595) 11

[3.2.5 Create Index](#_yv351jhwepa7) 11

[3.2.6 Drop Index](#_oaxsv8lioq7e) 12

[3.2.7 Batch](#_5bwp64ny0lga) 12

[3.3 Cassandra CURD](#_pgck1q5omfag) 12

[3.3.1 Create Data](#_z4ahyktb8jjh) 12

[3.3.2 Update Data](#_e0vb09vjt0ee) 13

[3.3.3 Read Data](#_irohdd130qgb) 13

[3.3.4 Delete Data](#_uyaec17gdo5n) 14

[**IV - Demo**](#_bio3ba8ud7bo) **14**

[5.1 Truy vấn Cassandra và vẽ Dashboard](#_md9vaqs96iks) 14

[5.1.1 Kết nối Cassandra bằng Jupyter notebook](#_c1neoe5qjayg) 14

[5.1.2 Gộp dataset](#_ulwy6vgm542g) 15

[5.1.3 Kết nối Cassandra Server và tạo keyspaces](#_xahj45n1x416) 16

[5.1.4 Tạo bảng, load dữ liệu vào bảng và truy vấn](#_gh7we6wbml7p) 17

[5.2 Kết nối Cassandra và PowerBI](#_2cpkxbye0ja) 22

[5.3 Độ che phủ đất](#_ji4nu56fgxko) 27

**Phân Công Công Việc**

—--------------------------------------------------------------------------------------------------

Cassandra:

Trần Công Tuấn Mạnh - Cài đặt Cassandra, Tạo bảng, Tạo Dashboard, Lý thuyết Cassandra

—--------------------------------------------------------------------------------------------------

Cassandra:

Phan Hoàng Việt - Load data, Truy vấn, Lý thuyết Cassandra

—--------------------------------------------------------------------------------------------------

Apache Hive:

Lê Ngọc Đoan - Demo Apache Hive, Lý thuyết Cassandra

—--------------------------------------------------------------------------------------------------

# I - Giới thiệu Cassandra

## Cassandra là gì ?

Apache Cassandra là một hệ thống lưu trữ (cơ sở dữ liệu) mã nguồn mở, phân tán và phi tập trung / phân tán, để quản lý một lượng rất lớn dữ liệu có cấu trúc trải rộng trên toàn thế giới, cung cấp dịch vụ có tính khả dụng cao mà không có điểm lỗi nào. Nó là một loại cơ sở dữ liệu NoSQL.

## Đặc điểm về Cassandra

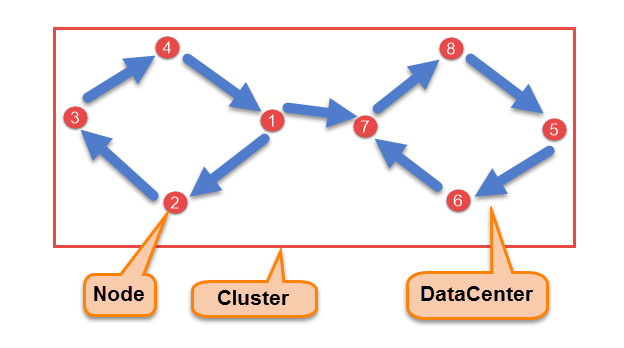
### 2.1. Một số tính năng của Cassandra

* Elastic scalability: có khả năng mở rộng cao, nó cho phép thêm nhiều phần cứng hơn để đáp ứng nhiều khách hàng hơn và nhiều dữ liệu hơn theo yêu cầu.
* Always on architecture: không có điểm nào bị lỗi và nó liên tục có sẵn cho các ứng dụng quan trọng của doanh nghiệp không thể xảy ra lỗi.
* Fast linear-scale performance: Khi nhiều nút được thêm vào, hiệu suất của Cassandra sẽ tăng lên.
* Flexible data storage: đáp ứng tất cả các định dạng dữ liệu có thể có bao gồm: có cấu trúc, bán cấu trúc và không có cấu trúc. Nó có thể tự động điều chỉnh các thay đổi đối với cấu trúc dữ liệu của bạn theo nhu cầu của bạn.
* Easy data distribution: cung cấp tính linh hoạt để phân phối dữ liệu ở những nơi bạn cần bằng cách sao chép dữ liệu trên nhiều trung tâm dữ liệu.
* Data Compression: Cassandra có thể nén tới 80% dữ liệu mà không cần bất kỳ chi phí nào.
* Transaction support: hỗ trợ các thuộc tính như Atomicity, Consistency, Isolation, and Durability (ACID)
* Fast writes: Thực hiện ghi cực nhanh và có thể lưu trữ hàng trăm terabyte dữ liệu mà không làm giảm hiệu quả đọc.

### 2.2. Một số điểm đáng chú ý của Cassandra

* Nó là một cơ sở dữ liệu hướng cột.
* Thiết kế phân phối của nó dựa trên Dynamo của Amazon và mô hình dữ liệu của nó trên Bigtable của Google.
* Được tạo ra tại Facebook, nó khác hẳn với các hệ thống quản lý cơ sở dữ liệu quan hệ.
* Cassandra triển khai mô hình sao chép kiểu Dynamo mà không có điểm lỗi nào, nhưng bổ sung thêm một mô hình dữ liệu “họ cột” mạnh mẽ hơn.
* Cassandra đang được sử dụng bởi một số công ty lớn nhất như Facebook, Twitter, Cisco, Rackspace, ebay, Twitter, Netflix, v.v.

## Kiến trúc của Cassandra



* Node: Là nơi lưu trữ dữ liệu.
* Data center: Nó là một tập hợp các nút liên quan.
* Cluster (Cụm): Một cụm là một thành phần có chứa một hoặc nhiều trung tâm dữ liệu.
* Commit log (Nhật ký cam kết): Mọi thao tác ghi đều được ghi vào Nhật ký cam kết. Nhật ký cam kết được sử dụng để khôi phục sự cố.
* Mem-table (Bảng ghi nhớ): Bảng ghi nhớ là một cấu trúc dữ liệu thường trú trong bộ nhớ. Sau khi đăng nhập cam kết, dữ liệu sẽ được ghi vào bảng ghi nhớ. Đôi khi, đối với một họ cột đơn, sẽ có nhiều bảng ghi nhớ.
* SSTable : Nó là một tệp đĩa mà dữ liệu được xóa khỏi bảng ghi nhớ khi nội dung của nó đạt đến một giá trị ngưỡng.
* Bloom filter: Đây là những thuật toán nhanh chóng, không xác định, để kiểm tra xem một phần tử có phải là thành viên của một tập hợp hay không. Nó là một loại bộ nhớ cache đặc biệt. Bộ lọc Bloom được truy cập sau mỗi truy vấn.

1. Mô hình dữ liệu của Cassandra

4.1. Keyspace:

là vùng chứa ngoài cùng cho dữ liệu trong Cassandra. Các thuộc tính cơ bản của Keyspace trong Cassandra là:  
 - Replication factor: Là số lượng máy trong cụm sẽ nhận được các bản sao của cùng một dữ liệu.  
 - Replica placement strategy: Có các strategy như strategy đơn giản (rack-aware strategy), strategy cấu trúc liên kết mạng (datacenter-shared strategy)  
 - Column families: Keyspace là vùng chứa danh sách một hoặc nhiều họ cột.

4.2. Column Family: là một vùng chứa cho một tập hợp các hàng có thứ tự. Bảng sau liệt kê các điểm phân biệt họ cột với bảng cơ sở dữ liệu quan hệ.

|  |  |
| --- | --- |
| Relational Table | Cassandra column Family |
| Một lược đồ trong mô hình quan hệ là cố định. Khi chúng ta xác định các cột nhất định cho bảng, trong khi chèn dữ liệu, trong mỗi hàng, tất cả các cột phải được điền ít nhất bằng giá trị null. | Trong Cassandra, mặc dù column family được xác định, nhưng các cột thì không. Bạn có thể tự do thêm bất kỳ cột nào vào bất kỳ column family bất kỳ lúc nào. |
| Bảng quan hệ chỉ xác định các cột và người dùng điền vào bảng các giá trị. | Trong Cassandra, một bảng chứa các cột, hoặc có thể được định nghĩa là một super column family. |

Column Family Cassandra có các thuộc tính sau:  
 - Key\_cached: Nó đại diện cho số lượng vị trí cần lưu trong bộ nhớ cache trên mỗi SSTable.  
 - Row\_cached: Nó đại diện cho số hàng có toàn bộ nội dung sẽ được lưu trong bộ nhớ.  
 - Preload\_row\_cache: Nó chỉ định xem bạn có muốn điền trước vào bộ đệm hàng hay không.  
4.3. Column: Một cột là cấu trúc dữ liệu cơ bản của Cassandra với ba giá trị, đó là tên khóa hoặc cột, giá trị và dấu thời gian. Dưới đây là cấu trúc của một cột.  
   
4.4. SuperColumn: là một cột đặc biệt, do đó, nó cũng là một key-value.

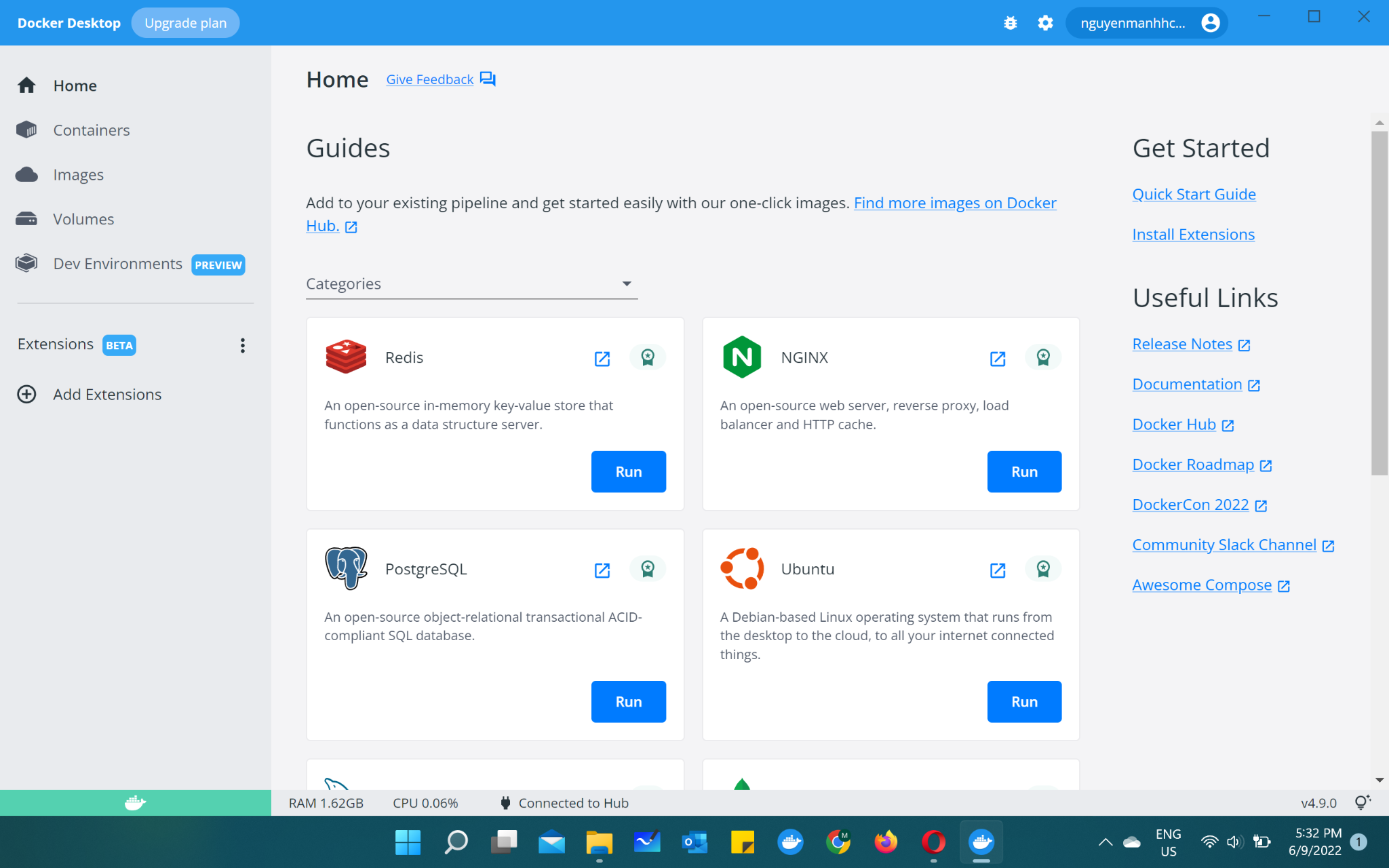
# II - Cài đặt Cassandra

Cài đặt Cassandra trên môi trường docker Windows.

Bước 1: Tải docker desktop cho windows, tham khảo link dưới:

<https://docs.docker.com/desktop/windows/install/>

* Nhấn double click vào file vừa tải về. Sau đó cài đặt.
* Sau khi cài xong docker sẽ yêu cầu tải thêm WSL, có thể tải theo link dưới: <https://docs.microsoft.com/en-us/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package>
* Sau khi cài đặt hoàn tất hai phần trên, tìm docker desktop trong máy và mở lên.
* Giao diện sau khi cài:

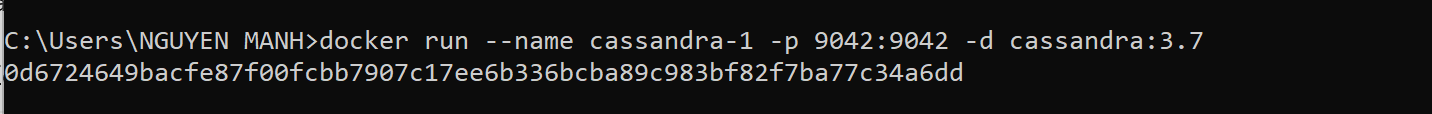


Bước 2: Cài Cassandra

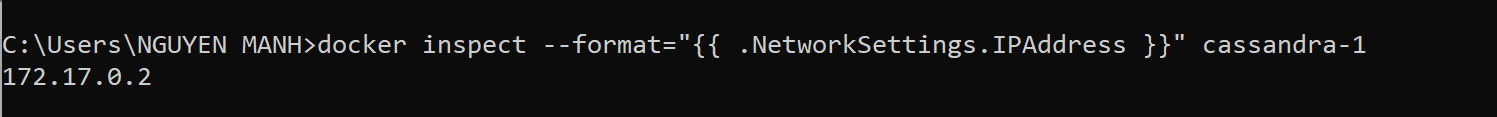
* Mở Commandline
* Để cài Cassandra trên docker, ta dùng lệnh *docker pull cassandra*



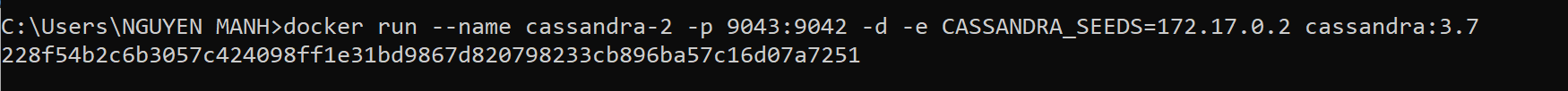
* Sau khi lệnh chạy xong, ta vào docker desktop
* Tạo node cassandra 1:



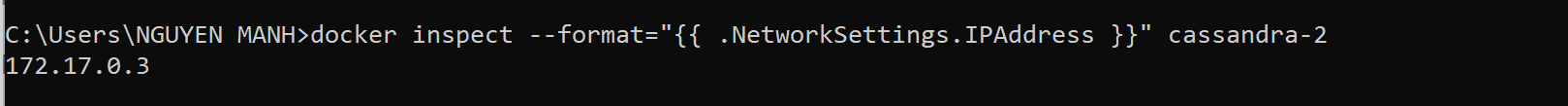
* Kiểm tra IP node 1:



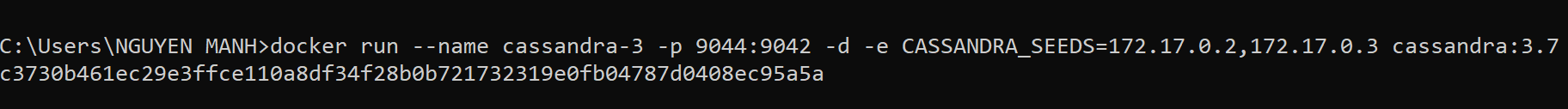
* Tạo node 2 với IP node 1:



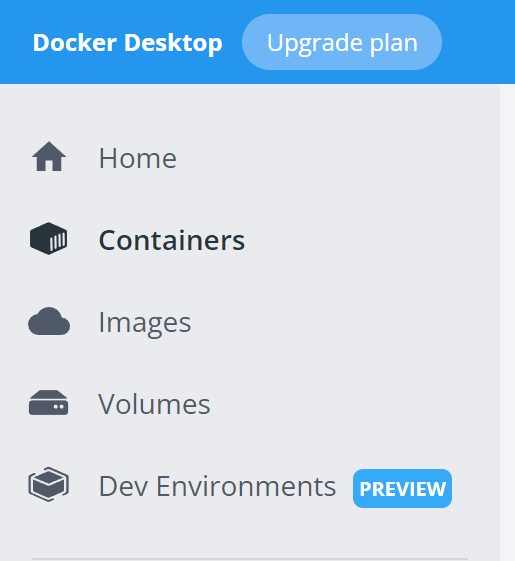
* Kiểm tra IP node 2:



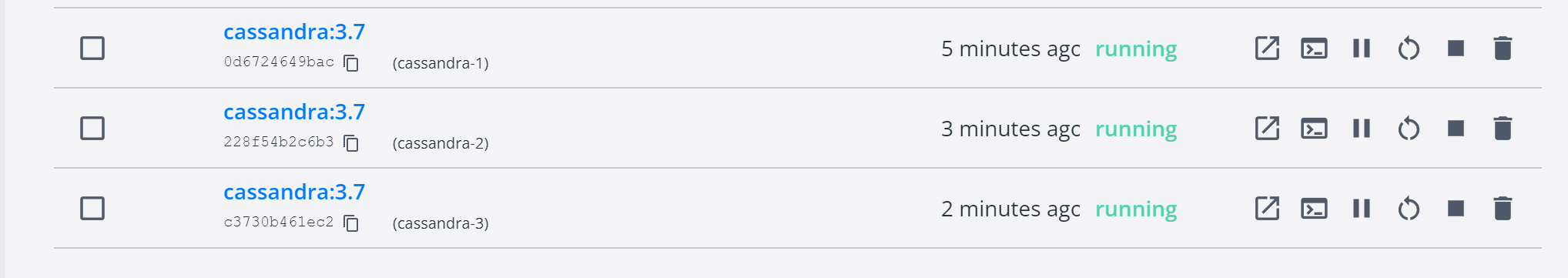
* Tạo node 3 với IP của node 1, 2:



Vào docker desktop kiểm tra container:

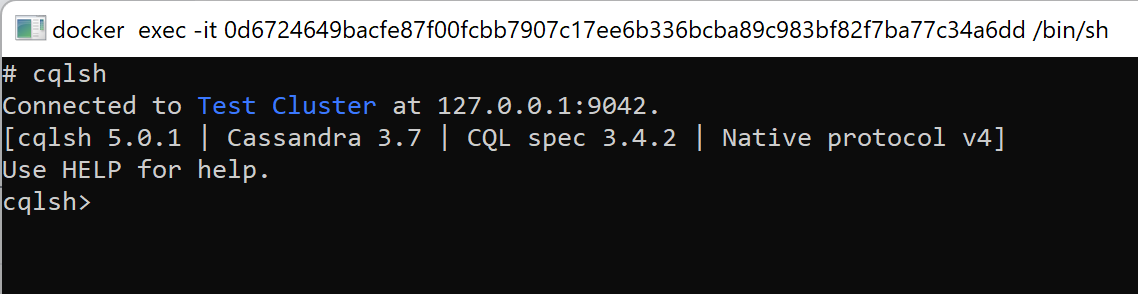


Nhấn vào Container:



Ta thấy có 3 container node đang run. Để vào giao diện sql shell, nhấn vào biểu tượng terminal bất kì trên 3 node như bên dưới:





Gõ lệnh cqlsh để vào CQL shell như trên.

Bước cài đặt đến đây là hoàn tất.

# III - Cú pháp Cassandra

## 3.1 Cassandra Keyspace Operations

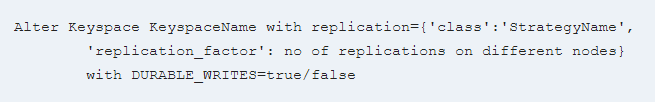
### 3.1.1. Tạo Keyspace:

Keyspace là một đối tượng chứa các họ cột, kiểu do người dùng xác định. Lệnh “ Create Keyspace ” được sử dụng để tạo keyspace trong Cassandra.

# 

Các thành phần khác nhau của Cassandra Keyspace  
 Strategy: Trong khi khai báo tên strategy trong Cassandra. Có hai loại strategy được khai báo trong Cú pháp Cassandra. Simple Strategy: Chỉ định một yếu tố sao chép đơn giản cho cụm.  
 Network Topology Strategy: cấu trúc liên kết mạng được sử dụng khi bạn có nhiều trung tâm dữ liệu. Trong chiến lược này, bạn phải cung cấp hệ số nhân rộng cho từng trung tâm dữ liệu riêng biệt.

### 3.1.2. Alter Keyspace Cassandra:

Lệnh “Alter Keyspace” thay đổi hệ số replication, tên strategy và các thuộc tính durable writes trong keyspace đã tạo trong Cassandra.  
 Syntax:  
  
 Các thuộc tính của ALTER KEYSPACE giống như CREATE KEYSPACE. Nó có hai thuộc tính : replication và durable\_writes.  
 Replication: Tùy chọn sao chép chỉ định strategy đặt bản sao và số lượng bản sao mong muốn.  
 Durable\_writes: Sử dụng tùy chọn này, bạn có thể hướng dẫn Cassandra có sử dụng commitlog cho các bản cập nhật trên KeySpace hiện tại hay không. Tùy chọn này không bắt buộc và theo mặc định, nó được đặt thành true.

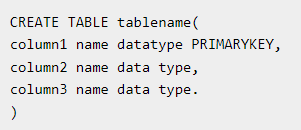
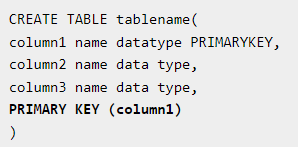
### 3.1.3. Drop Keyspace:

Lệnh Drop Keyspace bao gồm tất cả dữ liệu, họ cột, loại và index do người dùng xác định từ Cassandra.  
 Syntax:  
 

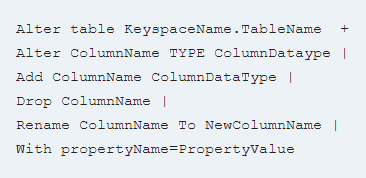
## 3. 2. Cassandra Table

### 3.2.1 Create Table

Bạn có thể tạo bảng bằng lệnh CREATE TABLE . Dưới đây là cú pháp để tạo một bảng.  
 

Primary key: Khóa chính là một cột được sử dụng để xác định duy nhất một hàng. Do đó, việc xác định khóa chính là bắt buộc trong khi tạo bảng.   
Syntax:  
   
hoặc:  
 

### 3.2.2 Alter Table

Lệnh 'Alter Table' được sử dụng để bỏ cột, thêm cột mới, thay đổi tên cột, thay đổi kiểu cột và thay đổi thuộc tính của bảng.  
 Syntax:  
 

### 3.2.3 Drop Table

Bạn có thể drop một bảng bằng lệnh ‘Drop Table’ chỉ định bao gồm tất cả dữ liệu từ keyspace. Cú pháp của nó như sau:  
 

### 3.2.4 Truncate Table

Bạn có thể cắt bớt một bảng bằng lệnh TRUNCATE. Khi bạn cắt bớt một bảng, tất cả các hàng của bảng sẽ bị xóa vĩnh viễn. Dưới đây là cú pháp của lệnh này.  
 Syntax:  
 

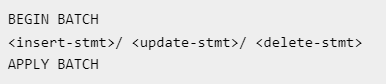
### 3.2.5 Create Index

Lệnh 'create index' trên cột do người dùng chỉ định. Nếu dữ liệu đã tồn tại cho cột bạn muốn lập index, Cassandra sẽ tạo index trên dữ liệu trong quá trình thực thi câu lệnh 'create index'.  
Syntax:  
 

### 3.2.6 Drop Index

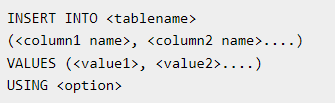
Bạn có thể giảm một index bằng lệnh DROP INDEX . Syntax:  
   
 Lưu ý:  
 - Nếu index không tồn tại, nó sẽ trả về lỗi trừ khi sử dụng IF EXISTS sẽ trả về no-op.

### 3.2.7 Batch

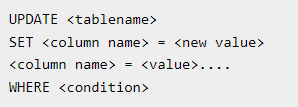
Sử dụng BATCH, bạn có thể thực hiện đồng thời nhiều câu lệnh sửa đổi (chèn, cập nhật, xóa).   
Syntax:  
 

## 3.3 Cassandra CURD

### 3.3.1 Create Data

Bạn có thể chèn dữ liệu vào các cột của một hàng trong bảng bằng lệnh INSERT.  
Syntax:  
 

### 3.3.2 Update Data

UPDATE là lệnh dùng để cập nhật dữ liệu trong bảng. Các từ khóa sau được sử dụng khi cập nhật dữ liệu trong bảng:  
 - Where: Mệnh đề này được sử dụng để chọn hàng sẽ được cập nhật.   
 - Set: Đặt giá trị bằng từ khóa này.   
 - Must: Bao gồm tất cả các cột tạo khóa chính.  
Syntax:  
 

### 3.3.3 Read Data

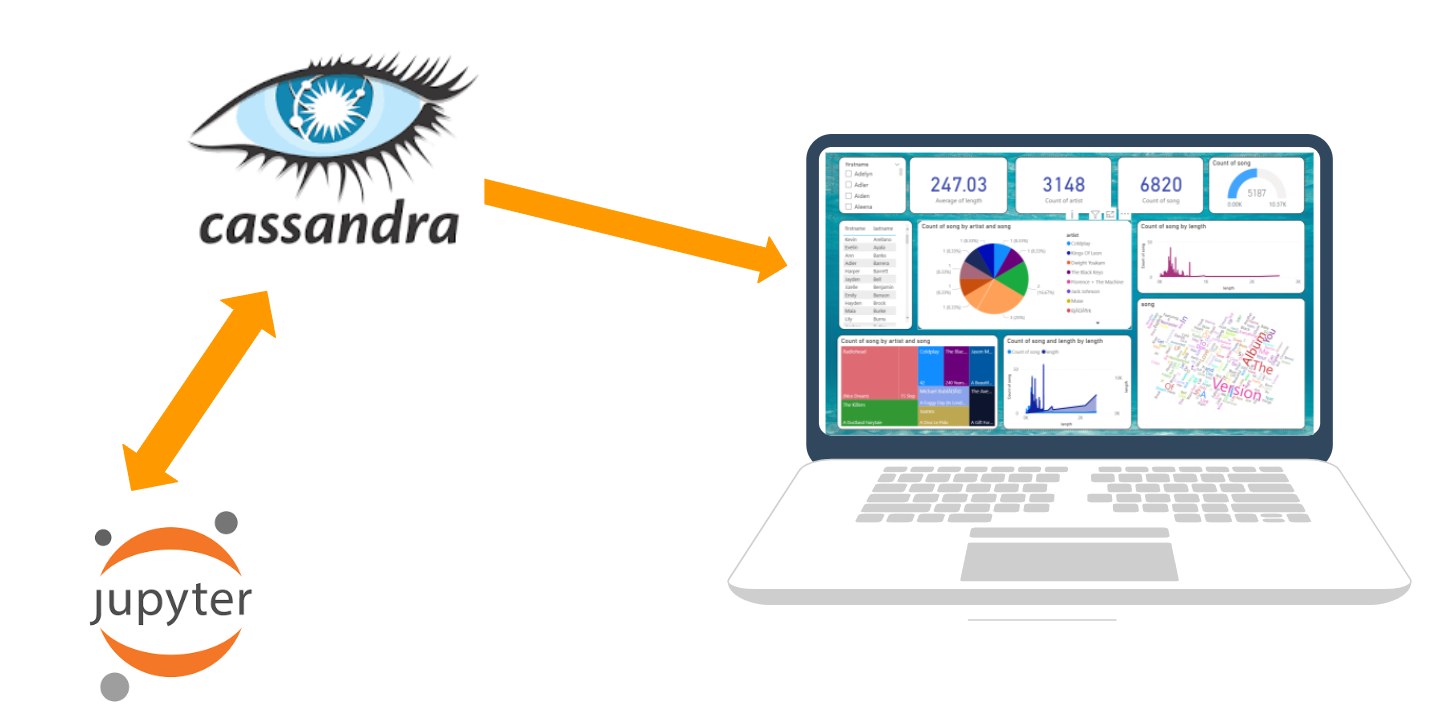
Mệnh đề SELECT được sử dụng để đọc dữ liệu từ một bảng trong Cassandra. Sử dụng mệnh đề này, bạn có thể đọc toàn bộ bảng, một cột hoặc một ô cụ thể.  
Syntax:  
   
 Sử dụng mệnh đề WHERE, bạn có thể đặt một ràng buộc đối với các cột bắt buộc.  
Syntax:  
 

### 3.3.4 Delete Data

Bạn có thể xóa dữ liệu khỏi bảng bằng lệnh DELETE .  
Syntax:  
 

# IV - Demo

## 5.1 Truy vấn Cassandra và vẽ Dashboard



Ý tưởng demo: Sử dụng jupyter notebook kết nối với Cassandra để thực hiện các thao tác tạo bảng, truy vấn. Sau đó kết nối Cassandra với PowerBI để vẽ Dashboard phục vụ cho việc phân tích dữ liệu.

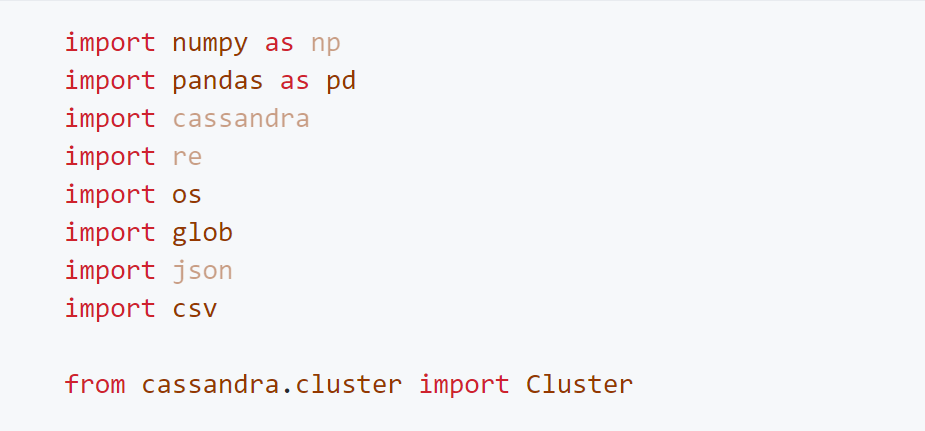
### 5.1.1 Kết nối Cassandra bằng Jupyter notebook

* Để kết nối Jupyter notebook với Cassandra, ta cần dừng một thư viện python là cassandra-driver.

Cài thư viện bằng lệnh sau:

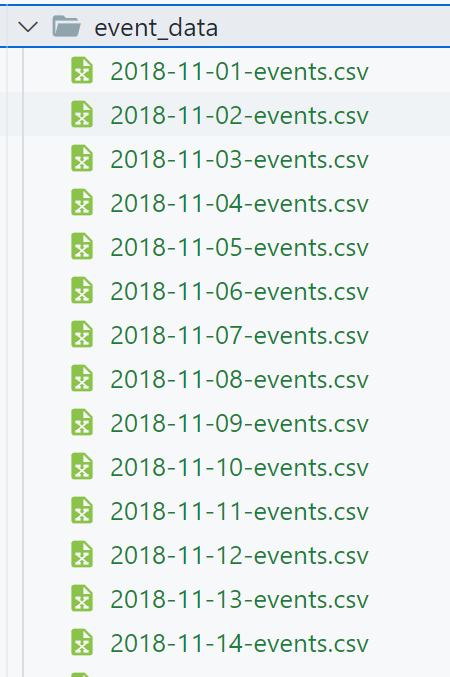
*pip install cassandra-driver*

Sau khi cài thư viện, cần import nó và một số thư viện khác cần thiết vào code python.



### 5.1.2 Gộp dataset

Bởi vì data của demo này gồm nhiều file nhỏ, chúng cần kết nối chúng lại với nhau thành một file hoàn chỉnh.



Ta dùng code python để gộp chúng lại với nhau:



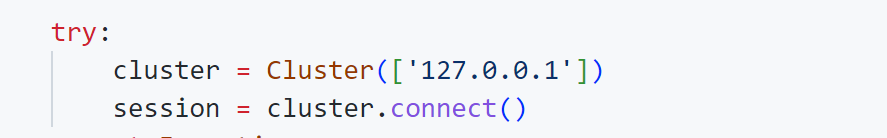
Code trên đây là để đọc các file trong thư mục chưa nó.



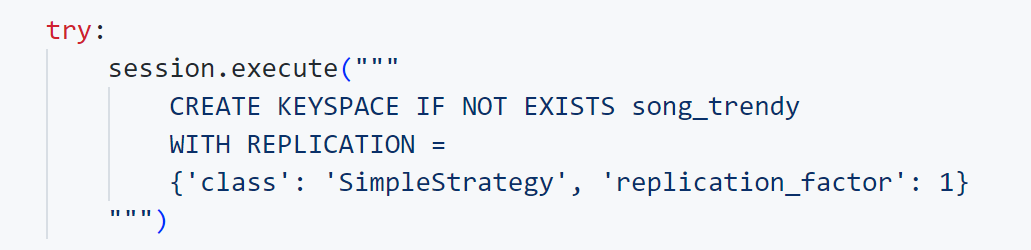
Kết nối chúng với nhau thông qua bước ghi dữ liệu vào một file mới.

### 5.1.3 Kết nối Cassandra Server và tạo keyspaces

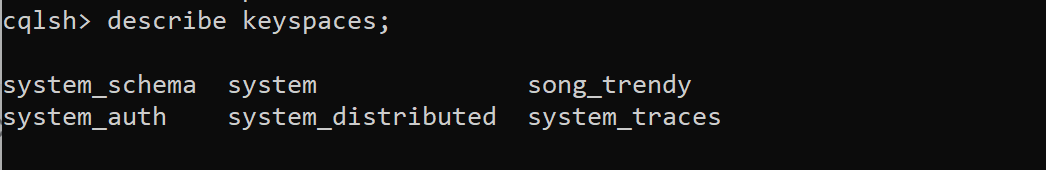
* Để kết nối với 1 cụm Cassandra, ta dùng lệnh bên dưới:



* Nó sẽ tự động kết nối tới địa chỉ IP mặc định của Cassandra Server( 127.0.0.1:9042).
* Tạo bảng bằng lệnh bên dưới, trong câu lệnh session.execute của python, chưa lệnh tạo bảng của Cassandra. Thực thi và tạo bảng *song\_trendy* như bên dưới.



Nếu muốn kiểm tra xem keyspace đã được tạo trong Cassandra thay chưa. Ta truy cập trực tiếp trên CQL shell.

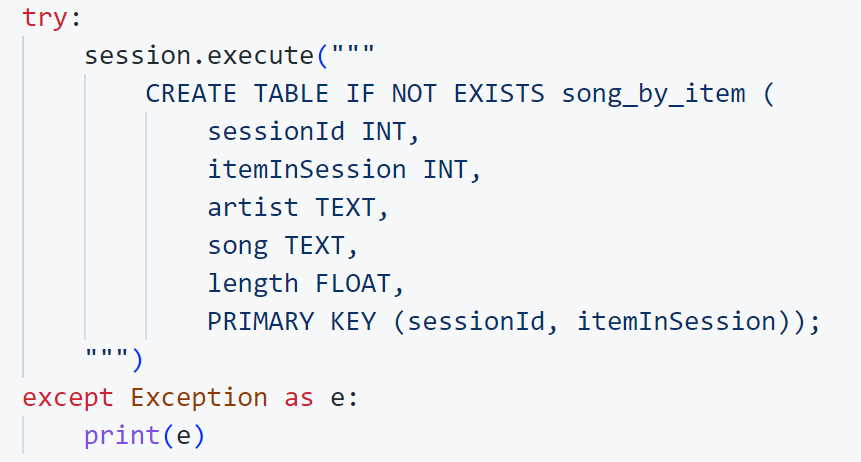


* Như vậy là bước tạo keyspace đã hoàn tất.

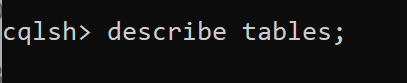
### 5.1.4 Tạo bảng, load dữ liệu vào bảng và truy vấn

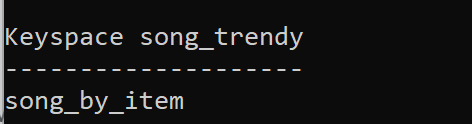
Query 1: Cung cấp tên nghệ sĩ, tên bài hát và thời lượng bài hát trong lịch sử âm nhạc đã được nghe trong sessionId = 388 và itemInSession = 4

* Tạo bảng *song\_by\_item* chứa các cột như session, item, artist, song, length.



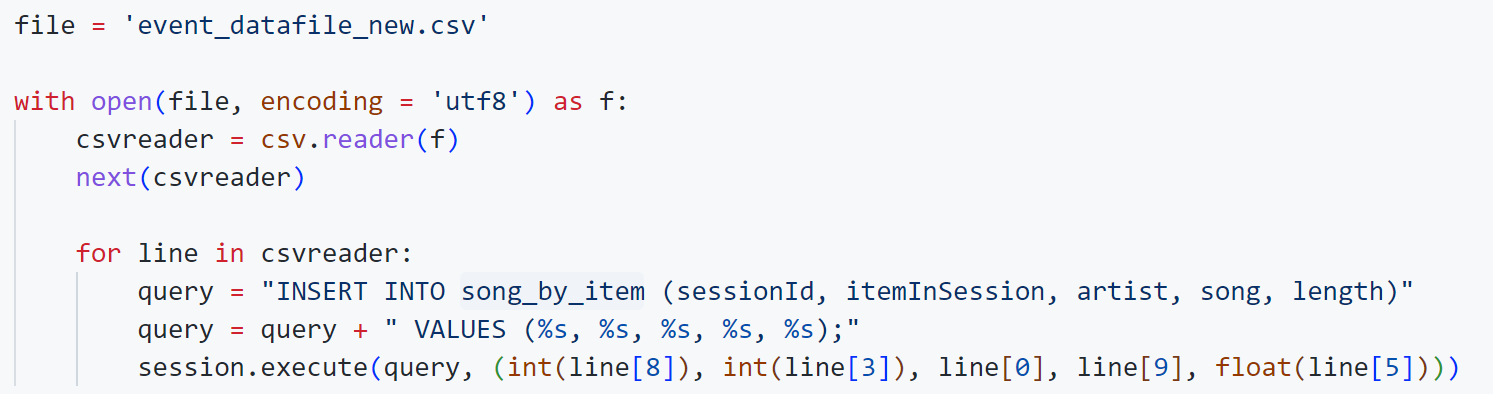
* Tương tự như trên, để xem bảng đã có trong Cassandra hay chưa, ta dùng CQL shell để xem:



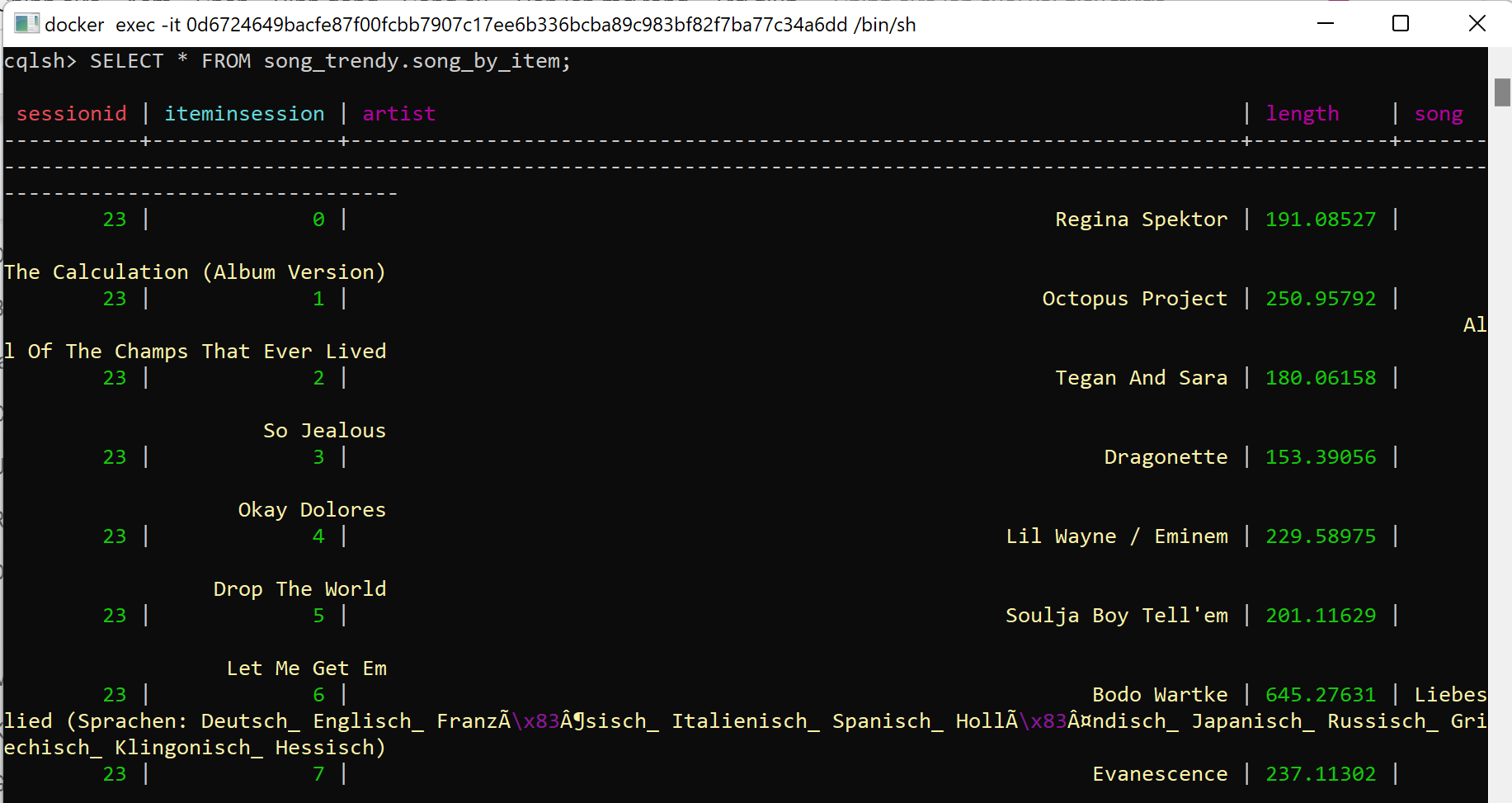


* Như vậy lad đã có bảng *song\_by\_item*.

Bước tiếp theo là load dữ liệu vào bảng, ta sử code như bên dưới;

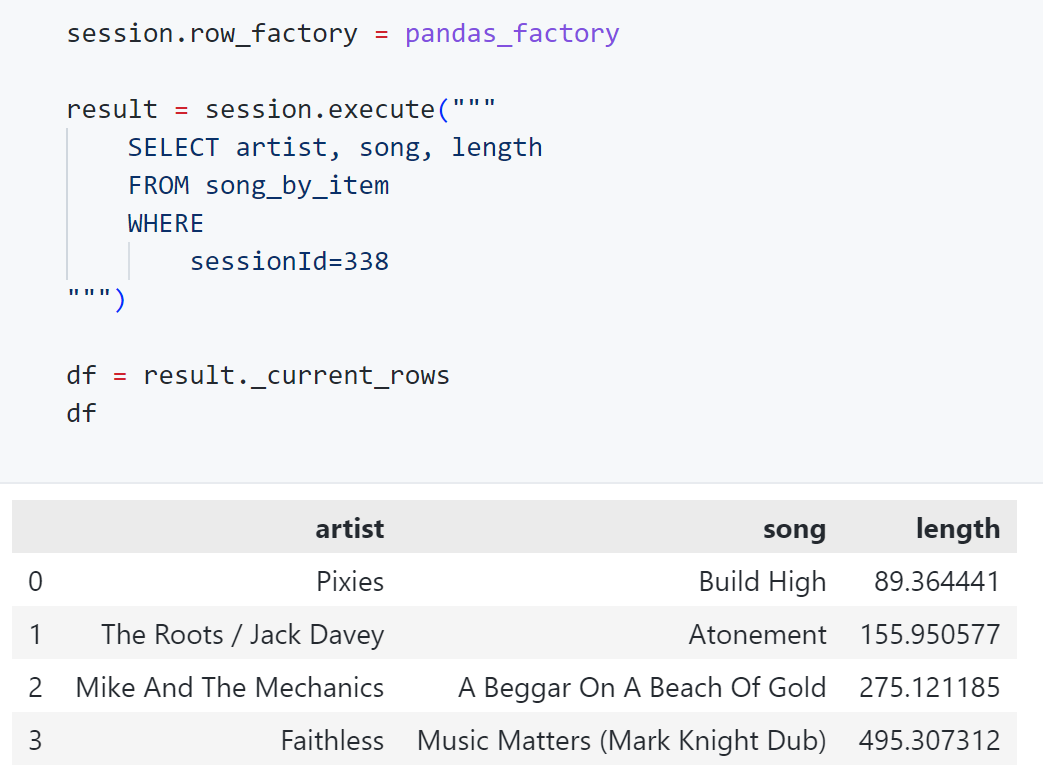


Sau khi load dữ liệu vào bảng xong, ta kiểm tra dữ liệu đã tồn tại trong bảng chưa bằng hai cách, dùng code python hoặc CQL shell:

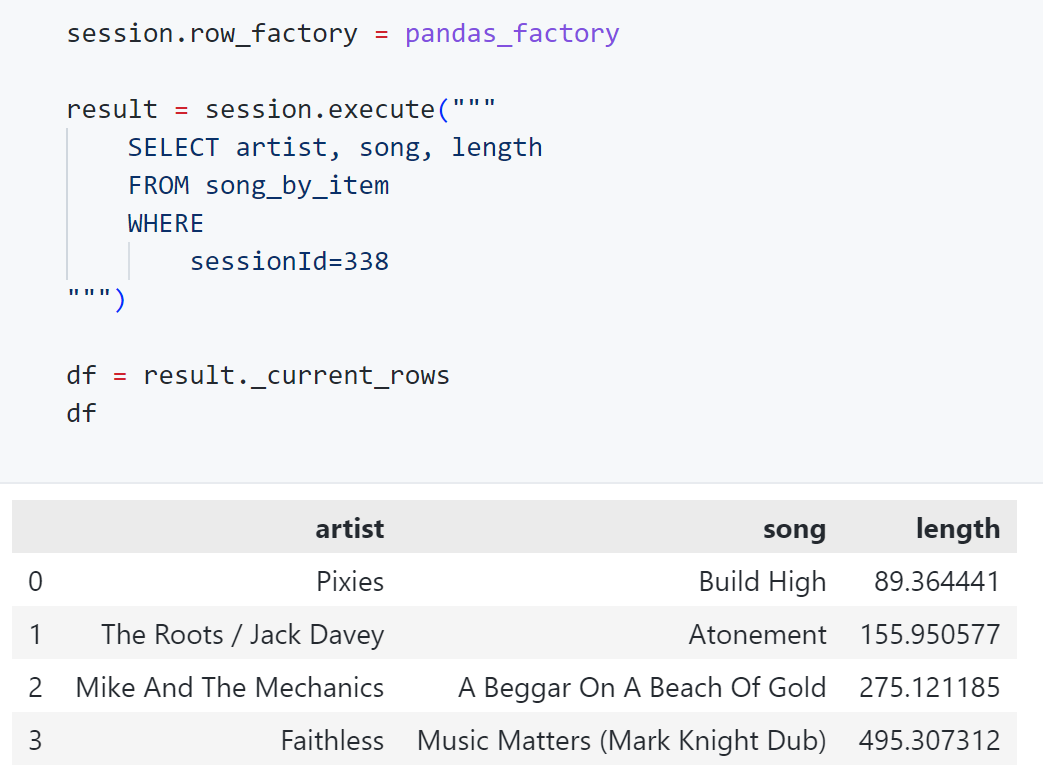


Như vậy là dữ liệu đã vào bảng hoàn tất.

* Tiếp theo là truy vấn trong bảng này.
* Để truy vấn và hiển thị dữ liệu dễ dàng và đẹp mắt, ta dùng code python để truy vấn như bên dưới:

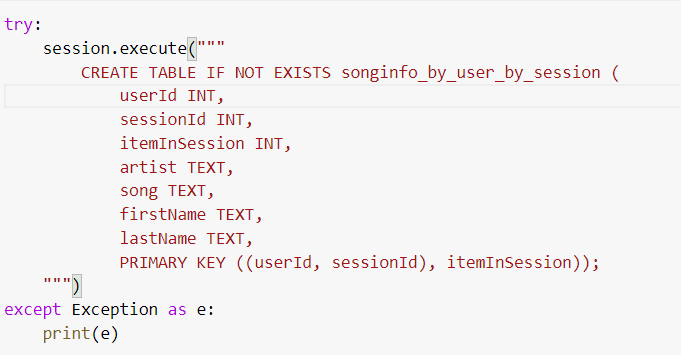


* Ta sẽ tìm kiếm nghệ sĩ, bài hát và độ dài của lượt nghe đó với sessionID bằng 338.
* Kết quả:

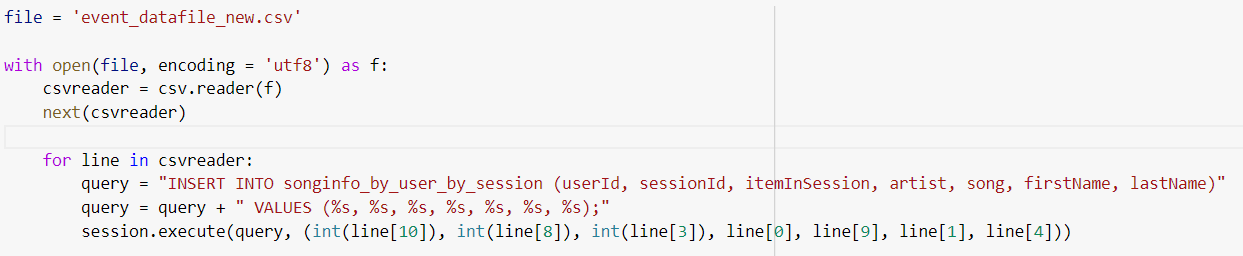


Query 2: Cung cấp cho tôi những thông tin sau: tên nghệ sĩ, bài hát (được sắp xếp theo itemInSession) và người dùng (họ và tên) cho userId = 10, sessionId = 182

* Tạo bảng *songinfo\_by\_user\_by\_session* chứa userID , sessionId, iteamInSession, artist ,song, fistName, lastName

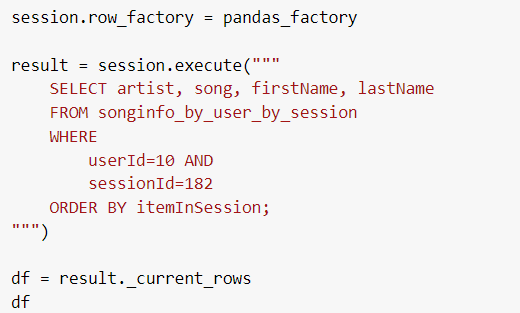


Bước tiếp theo là load dữ liệu vào bảng



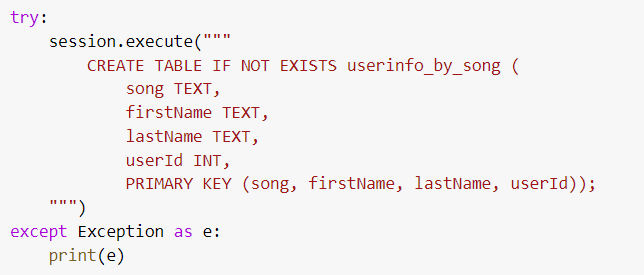
Như vậy là dữ liệu đã vào bảng hoàn tất.

Tiếp theo là truy vấn:

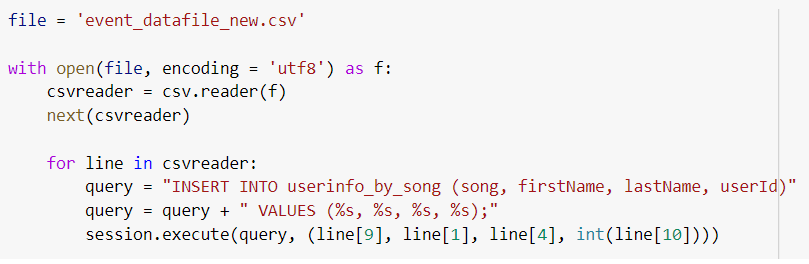


Query 3: Cung cấp cho tôi mọi tên người dùng (họ và tên) những người đã nghe bài hát 'All Hands Against His Own'

* Tạo bảng *userinfo\_\_by\_song* chứa song, firstName, lastName, userId.

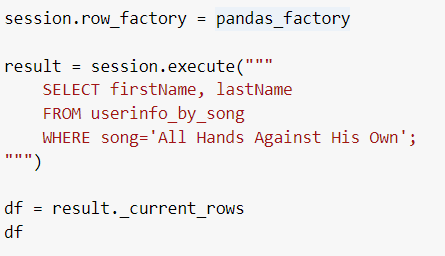


Bước tiếp theo là load dữ liệu vào bảng



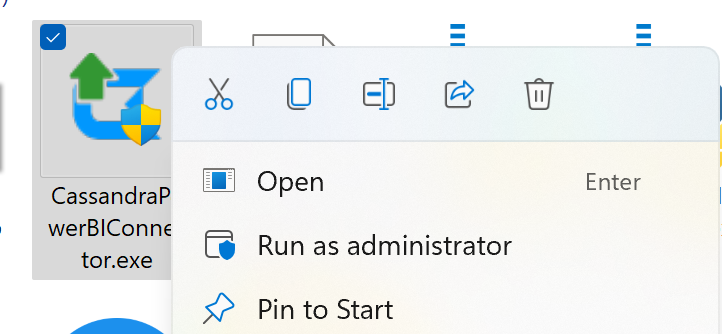
Như vậy là dữ liệu đã vào bảng hoàn tất.

Tiếp theo là truy vấn:

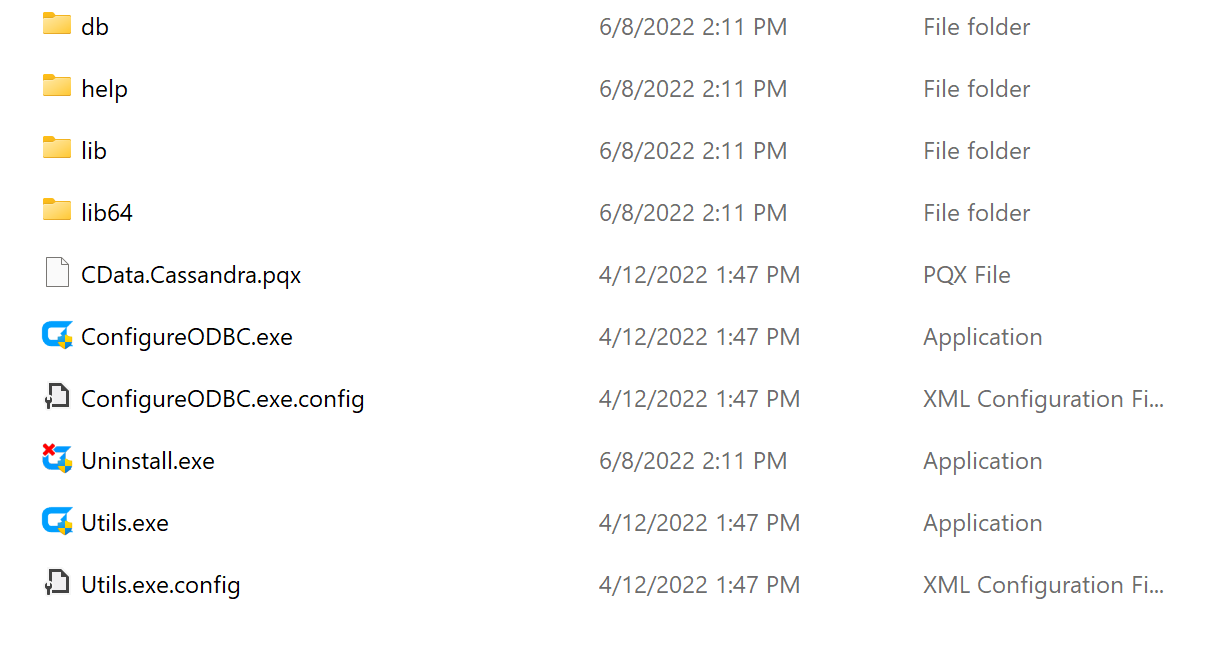


### 5.1.5 Kết nối Cassandra và PowerBI

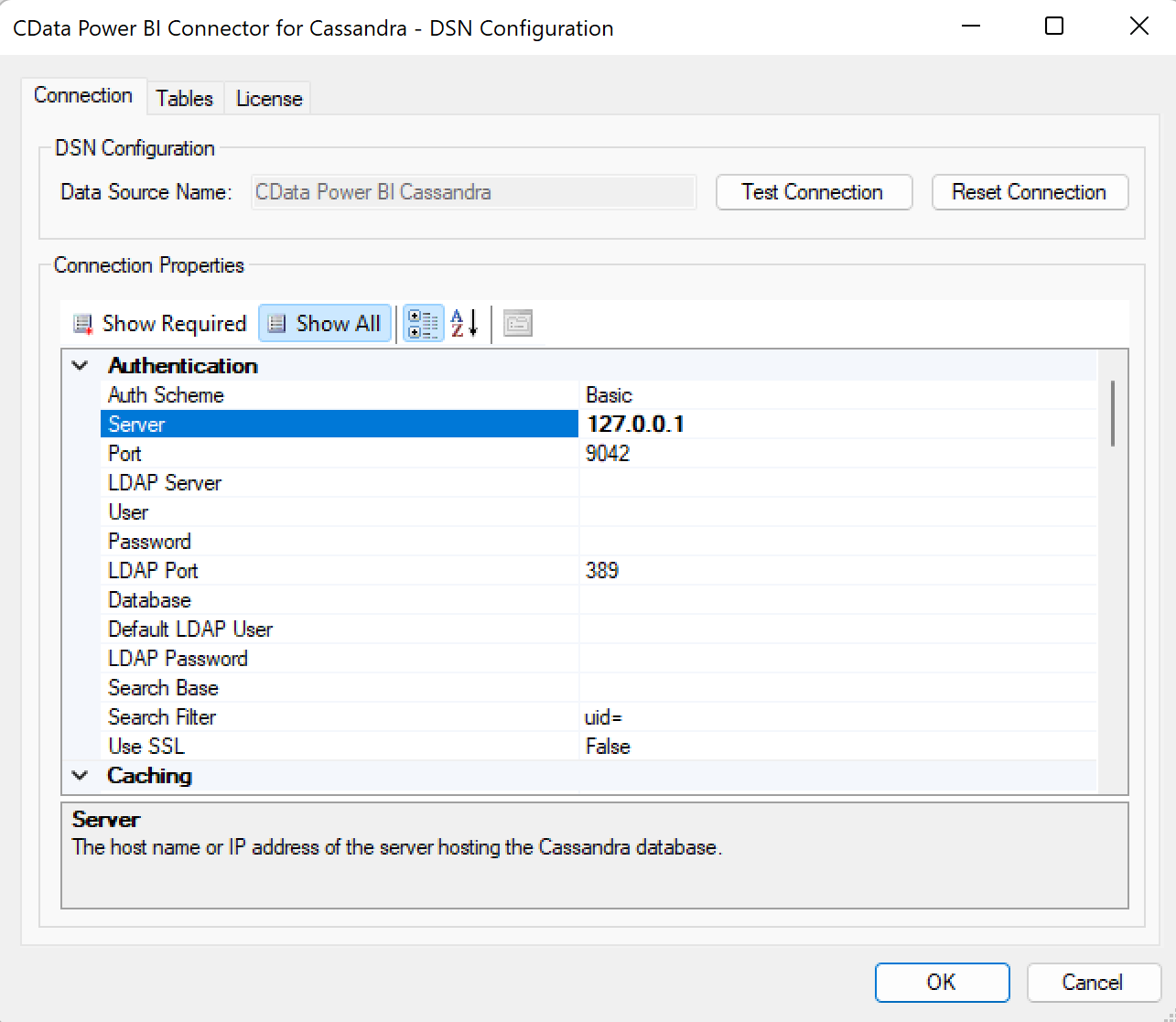
* Hiện tại PowerBI không hỗ trợ việc kết nối cở sở dữ liệu Cassandra, nhưng ta có thể sử dụng ODBC( Open Database Connectivity) để kết nối chúng lại với nhau.
* Trong bản demo này, ta sẽ sử dụng ODBC Cassandra từ nhà cung cấp **cdata.**
* Ta có thể tải ODBC theo link này: <https://www.cdata.com/drivers/cassandra/odbc/download/>
* Sau khi tải xong, cần run file .exe để build app.



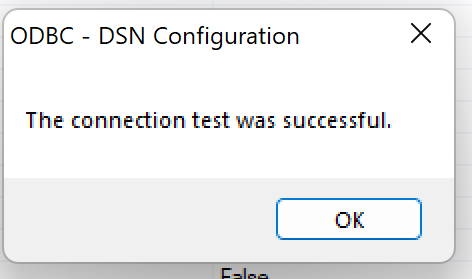
* Nhấp vào Open để run.
* Sau khi cài xong, mở thư mục đã configure lên;



* Chạy file ConfigureODBC.exe.
* Giao diện hiện lên, ta nhập vào mục Server giá trị “127.0.0.1” để kết nối.



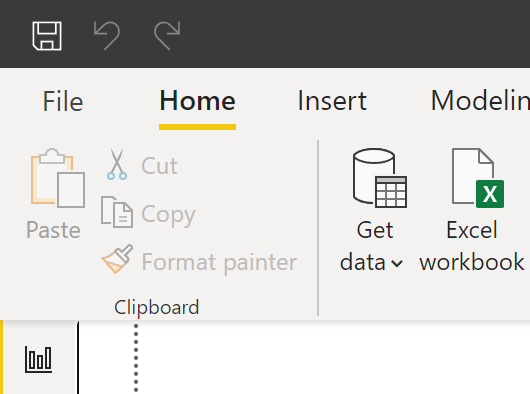
* Nhấp vào *Test Connection* ở phía trên để kiểm tra kết nối.



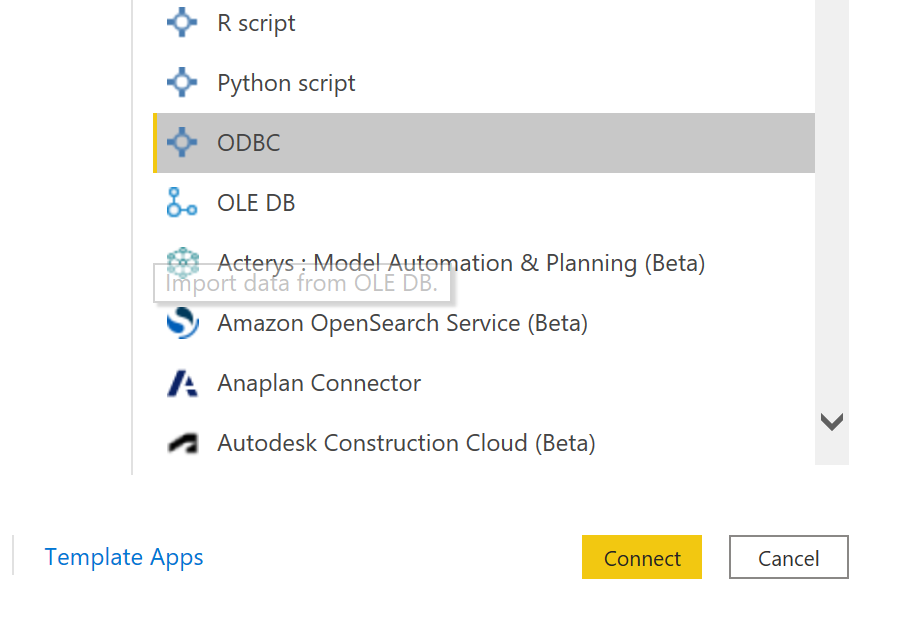
* Như vậy là đã kết nối thành công.

Tiếp theo cần mở PowerBI để kết nối với Cassandra.

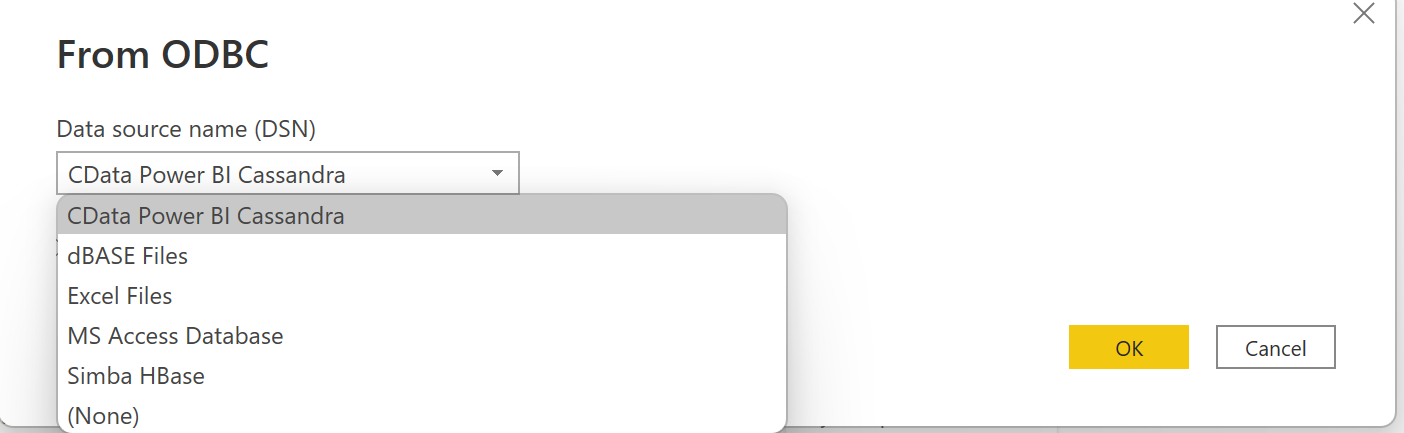
* Nhấp vào Get data.



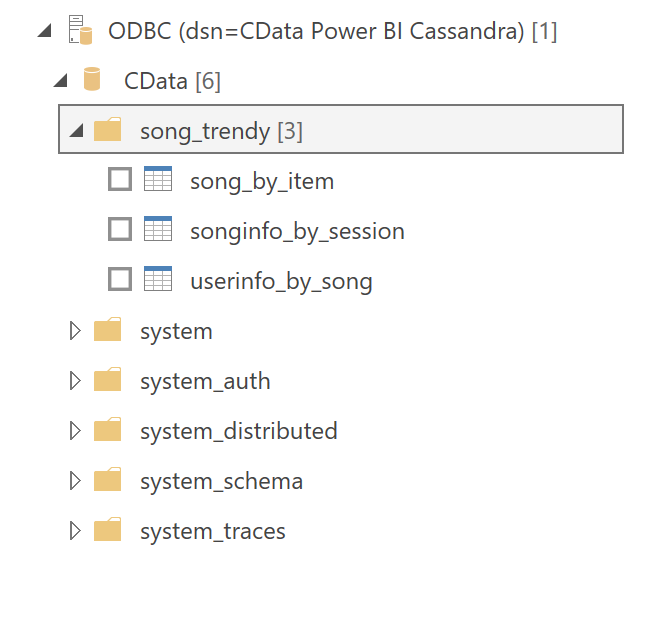
* Chọn ODBC và nhấn Connect.



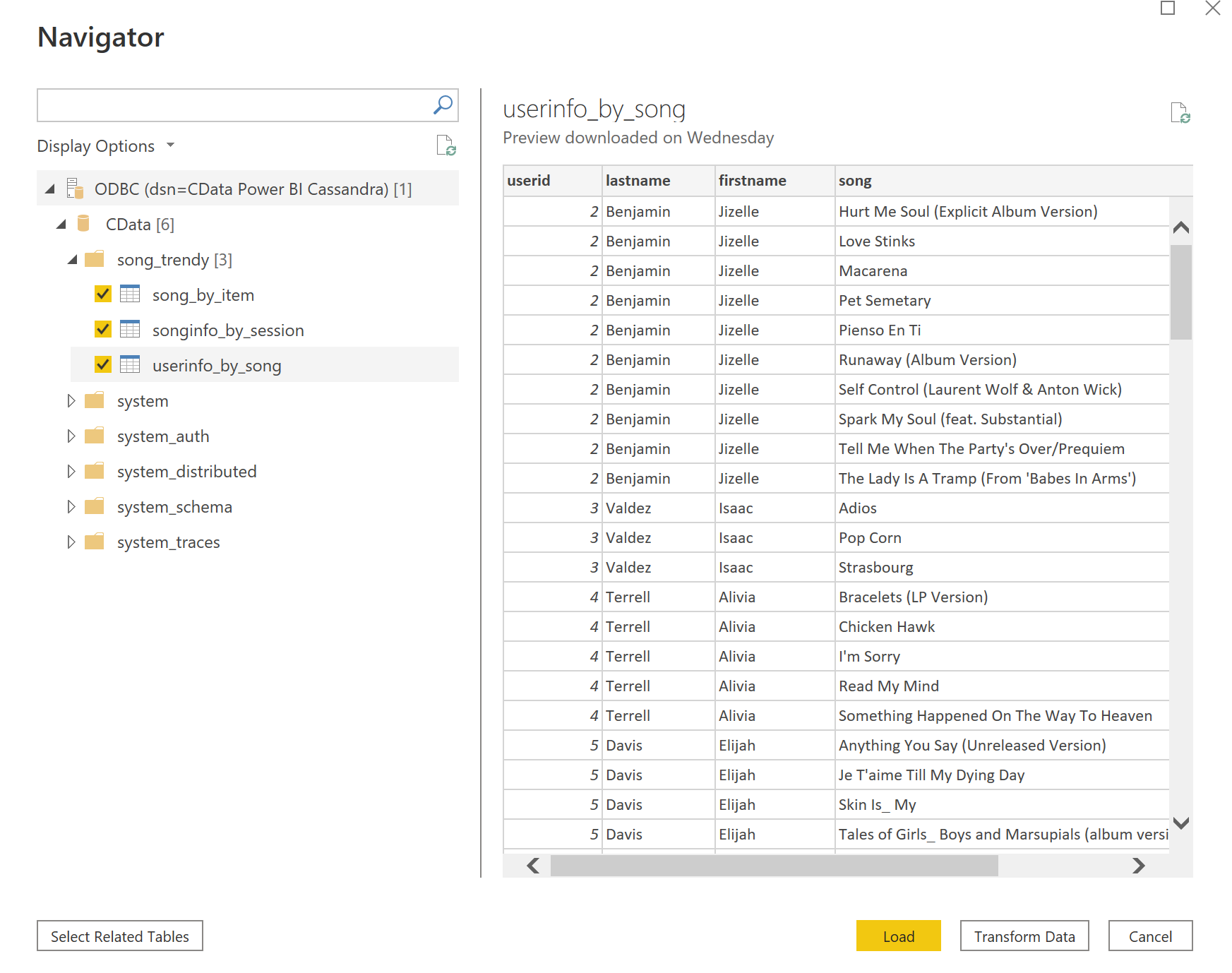
* Chọn CData Power BI Cassandra và nhấn OK.



* Sau khi kết nối thành công, bảng dữ liệu sẽ được hiện ra như dưới:

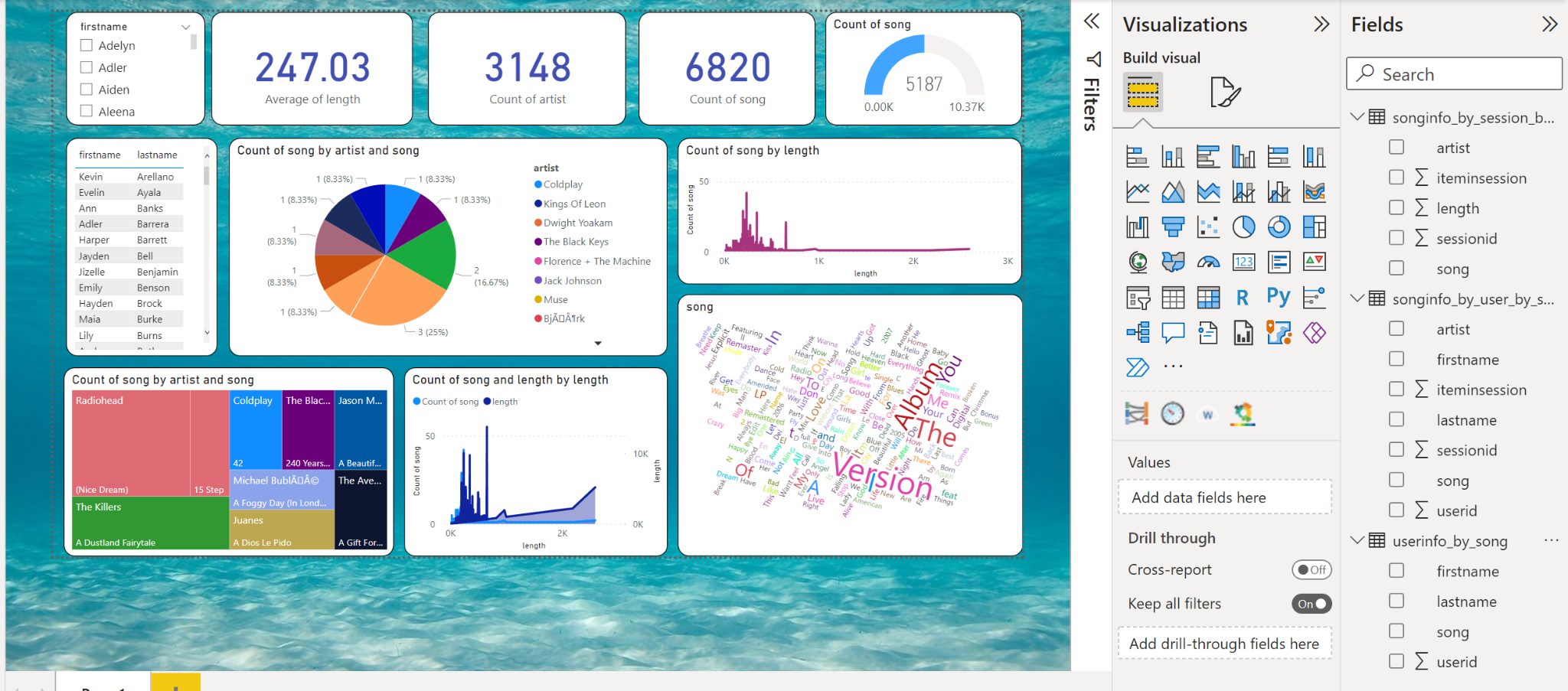


* Tích chọn hết từng bảng và load:



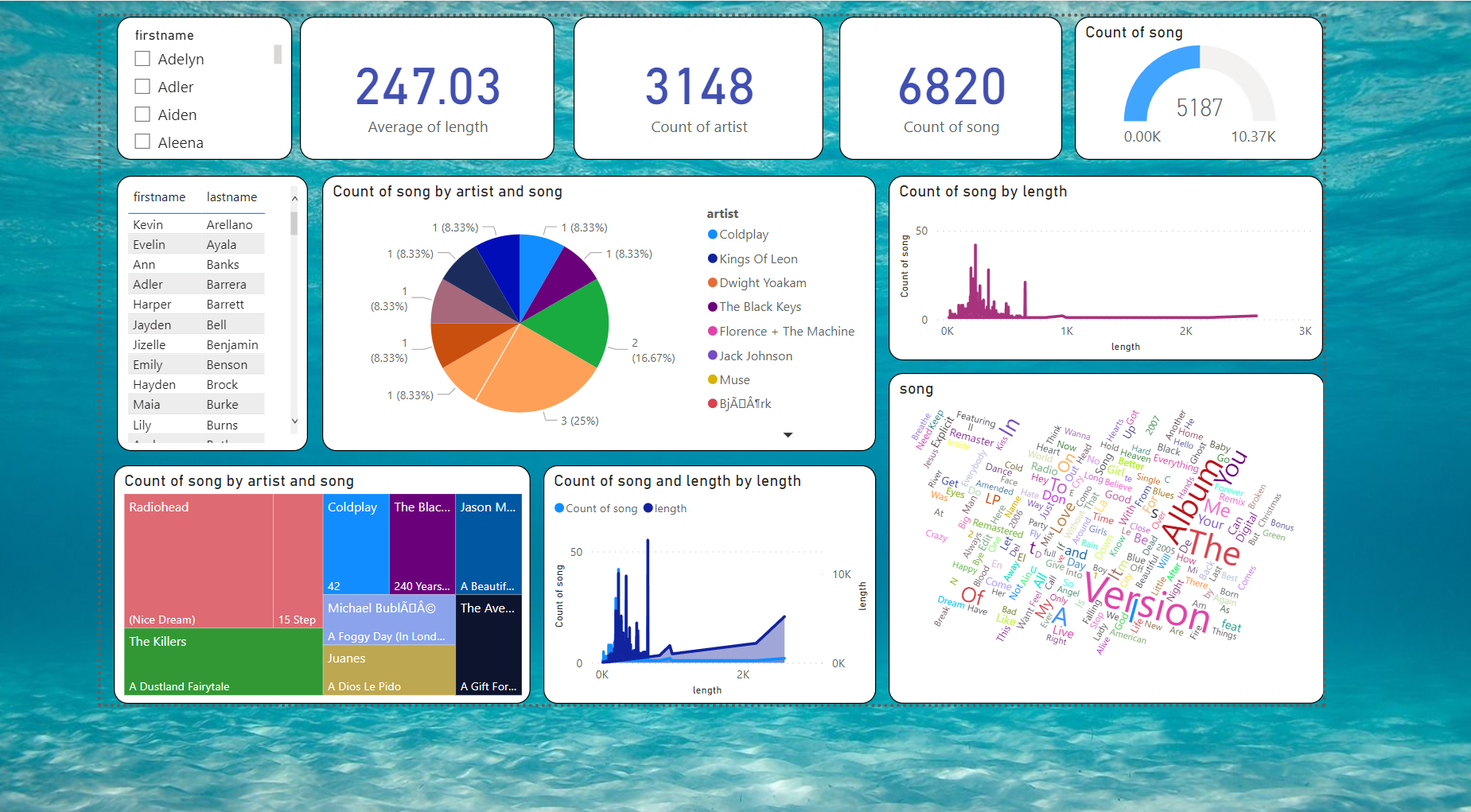
* Dữ liệu đã được load vào PowerBI, tiếp theo ta sẽ tiến hành vẽ Dashborad.

### 5.1.6 Vẽ Dashboard



* Cột data ở bên phải, ta có thể chọn vào áp dụng cho từng biểu đồ.

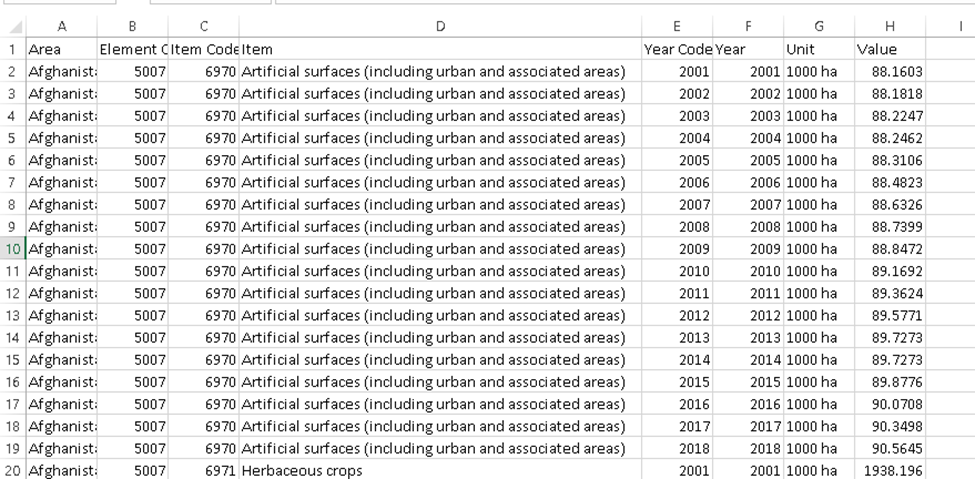
Dashboard hoàn chỉnh:



## 5.3. Độ che phủ đất

- Sử dụng dữ liệu Độ che phủ đất của toàn cầu được lấy từ trang web:<http://www.fao.org/faostat/en/#data/GT>

- Download về file csv và bỏ bớt những cột thật sự không cần thiết:



- Tập dữ liệu bao gồm 34720 hàng và 8 thuộc tính:

o Area: Tên nước

o Element Code: Mã nguồn dữ liệu được thu thập

o Item Code : Mã loại khu vực che phủ

o Item : Tên loại khu vực che phủ

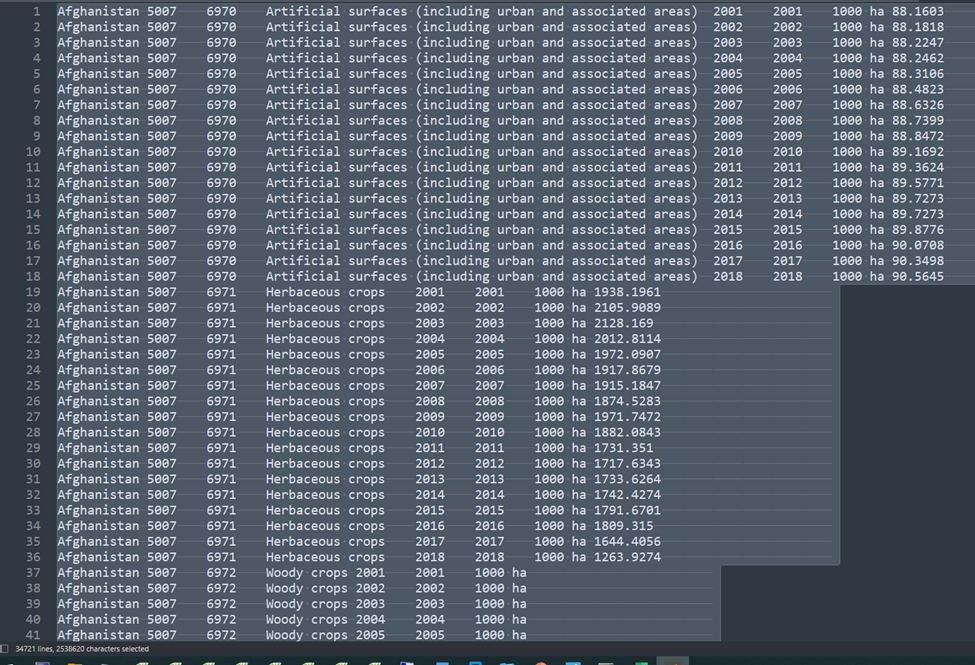
o Year Code : Mã năm

o Year : Năm

o Unit : Đơn vị (được tính là 1000 ha)

o Value : Diện tích

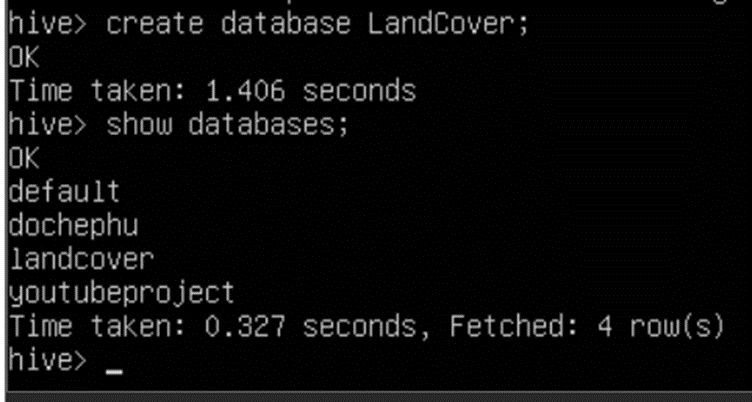
- Chuyển dữ liệu từ file .csv sang file .txt :



Tính tổng diện tích các loại khu vực trong năm 2018 toàn cầu

Tạo cơ sở dữ liệu:

# create database LandCover;



Tạo bảng:

CREATE TABLE dochephu (

Country string,

ElementCode int,

ItemCode int,

Item string,

YearCode int,

Year int,

Unit string,

Value double

)

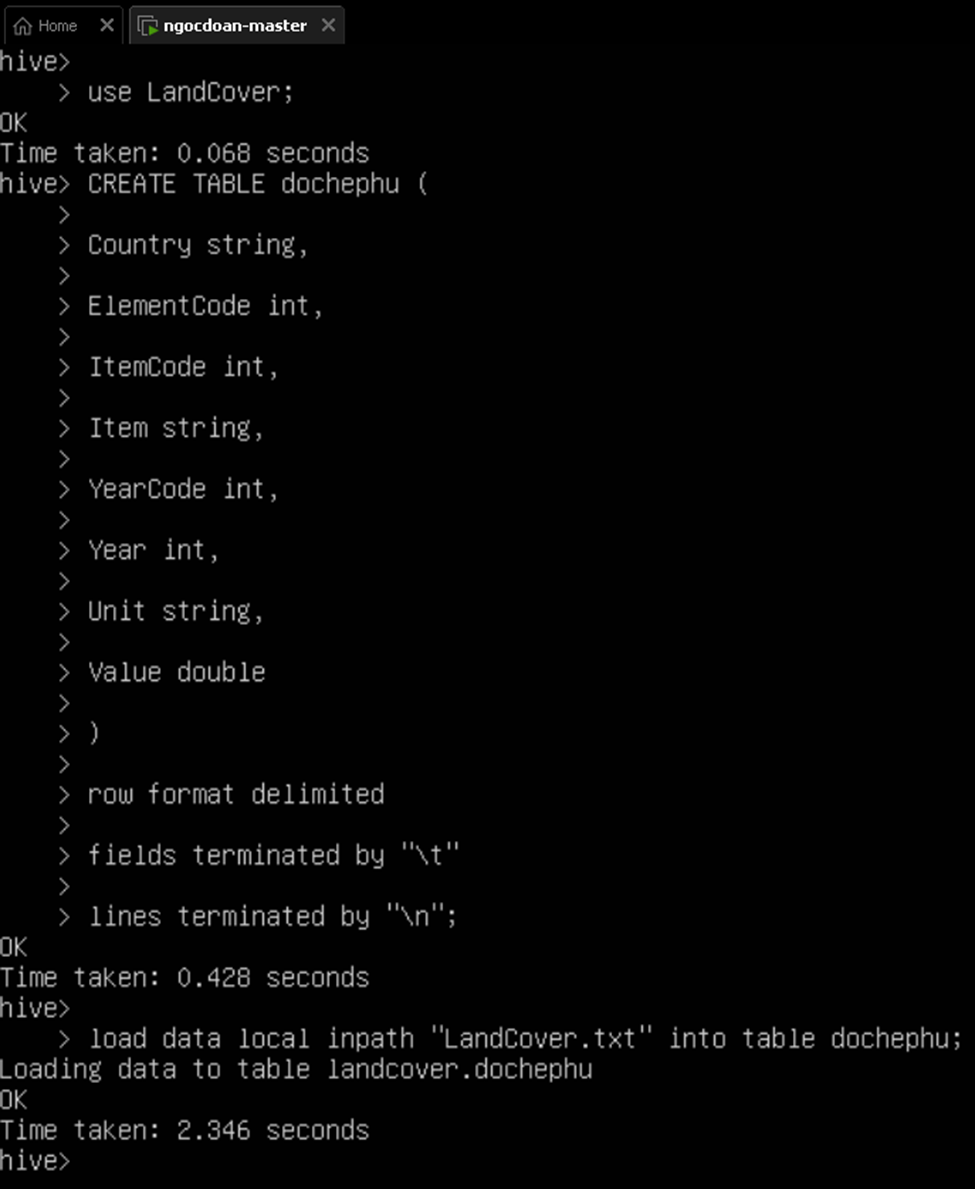
row format delimited

fields terminated by "\t"

lines terminated by "\n";

Tải dữ liệu vào bảng:

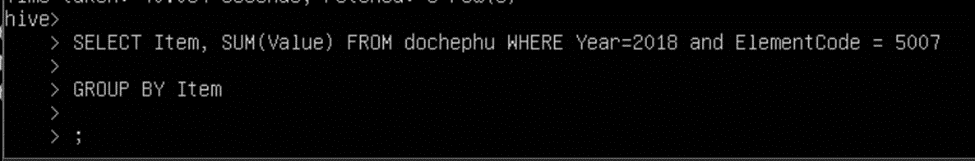
load data local inpath "LandCover.txt" into table dochephu;



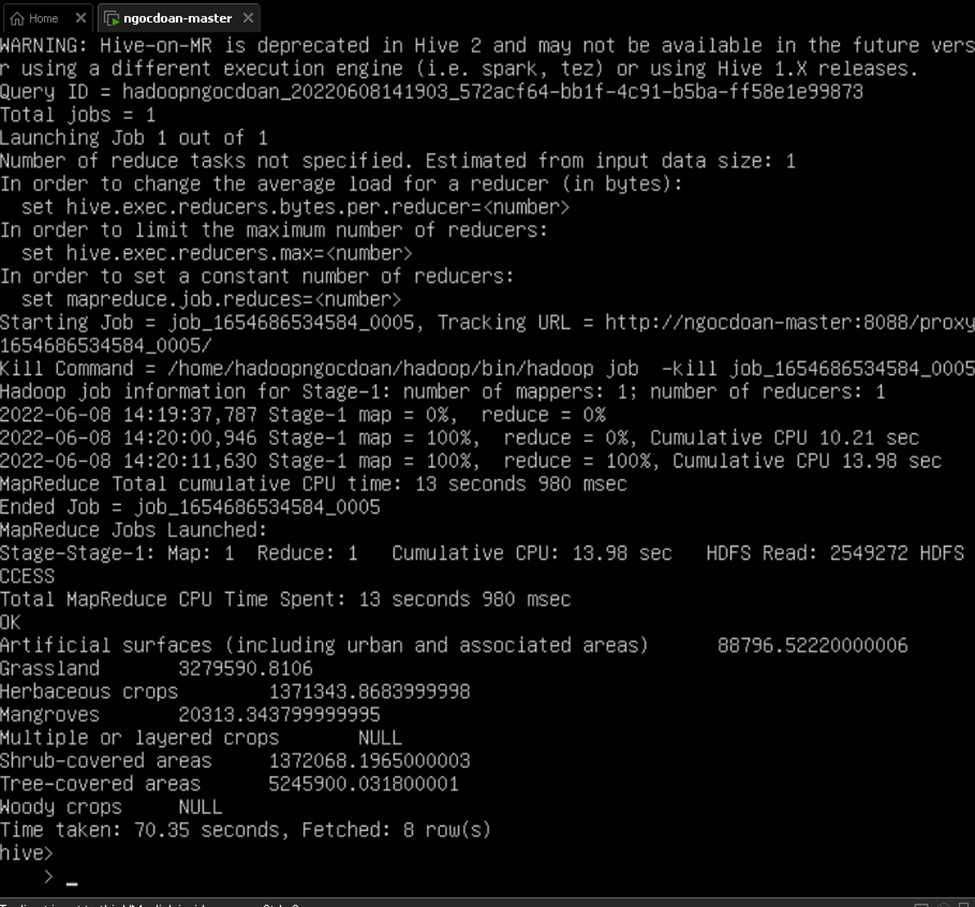
Tính tổng diện tích che phủ của loại khu vực trong năm 2018 và có mã nguồn thu thập dữ liệu là 5007

SELECT Item, SUM(Value) FROM dochephu WHERE Year=2018 and ElementCode = 5007

GROUP BY Item;



Kết quả đạt được:



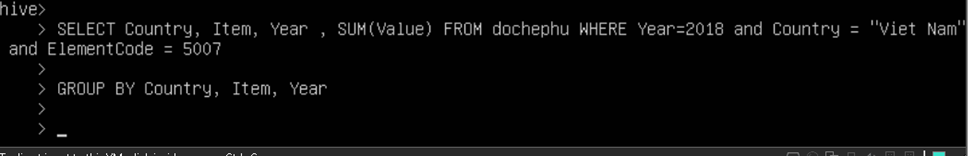
Kết quả gồm cột tên loại khu vực, diện tích của khu vực đó: Hiển thị 8 loại khu vực, trong đó 2 loại khu vực Woody crops và Multiple or layered crops do không có dữ liệu được thu thập. Qua đó ta thấy được diện tích khu vực nào đang chiếm nhiều nhất và ít nhất.

Thống kê diện tích các loại khu vực che phủ của Việt Nam năm 2018

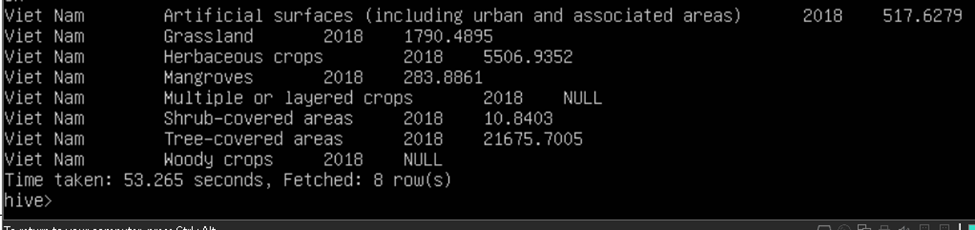
Tính diện tích bao phủ của khu vực che phủ nước Việt Nam năm 2018 có mã nguồn thu thập dữ liệu là 5007 của từng loại khu vực

SELECT Country, Item, Year , SUM(Value) FROM dochephu WHERE Year=2018 and Country = “Viet Nam” and ElementCode = 5007

GROUP BY Country, Item, Year;



Kết quả ra được:



Bao gồm: Tên nước Việt Nam, tên các loại khu vực, năm 2018, diện tích bao phủ của khu vực che phủ

Ta thấy diện tích cây che phủ là nhiều nhất, và cây bụi là ít nhất

Thống kê về diện tích khu vực Mangroves(rừng ngập mặn) của Việt Nam từ năm 2000 tới 2018

Hive là một công cụ tốt để thực hiện các truy vấn trên các bộ dữ liệu lớn - đặc biệt là các bộ dữ liệu yêu cầu quét toàn bộ bảng. Nhưng khá thường xuyên, có những trường hợp người dùng cần lọc dữ liệu trên các giá trị cột cụ thể. Và đó là nơi phân vùng phát huy tác dụng. Một phân vùng không có gì ngoài một thư mục chứa khối dữ liệu. Khi chúng tôi phân vùng, chúng tôi tạo một phân vùng cho mỗi giá trị duy nhất của cột.

# CREATE TABLE IF NOT EXISTS Country\_part(Item string,Year int, Value double, ItemCode int, ElementCode int) PARTITIONED BY(Country string);

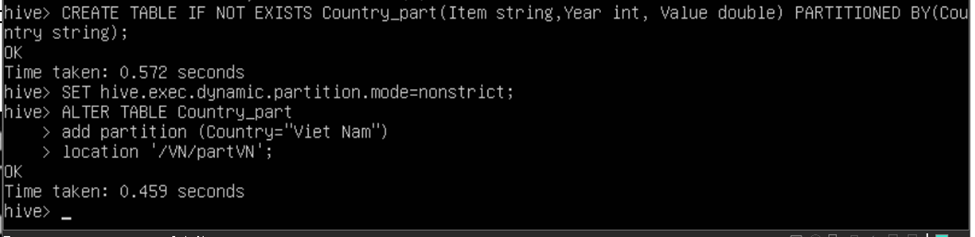
# SET hive.exec.dynamic.partition.mode=nonstrict;

Nếu hive.exec.dynamic.partition.modeđược đặt thành strict, thì bạn cần thực hiện ít nhất một phân vùng tĩnh. Trong chế độ không nghiêm ngặt, tất cả các phân vùng được phép động.

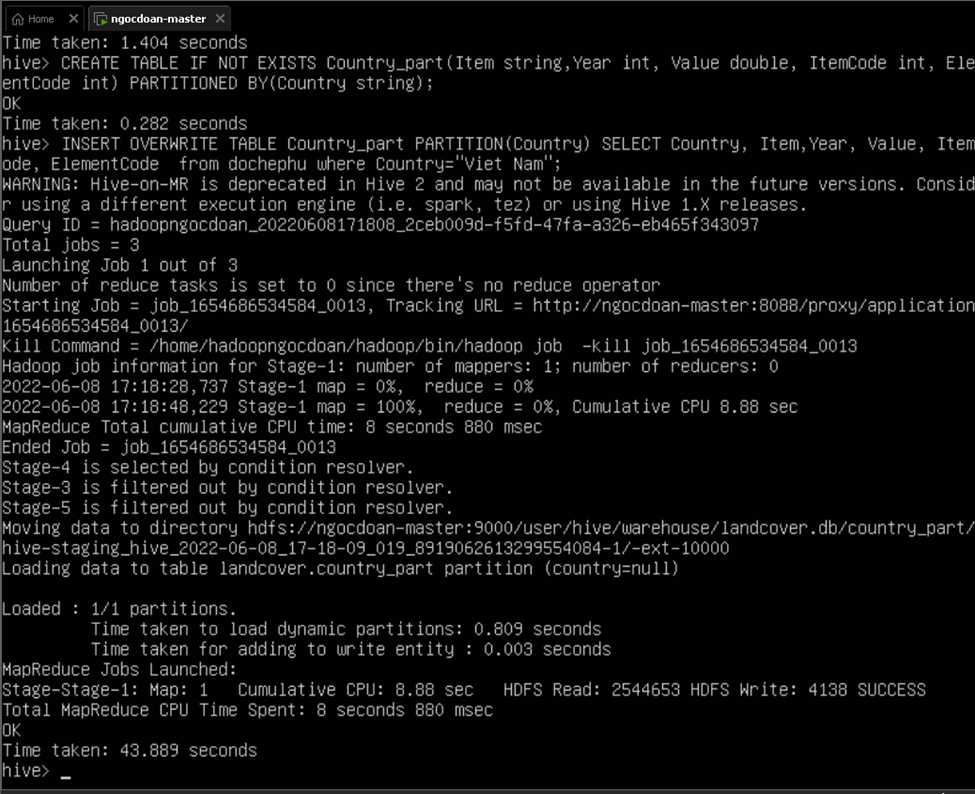
# ALTER TABLE Country\_part

# ADD PARTITION (Country=”Viet Nam”)

# location '/VN/partVN';



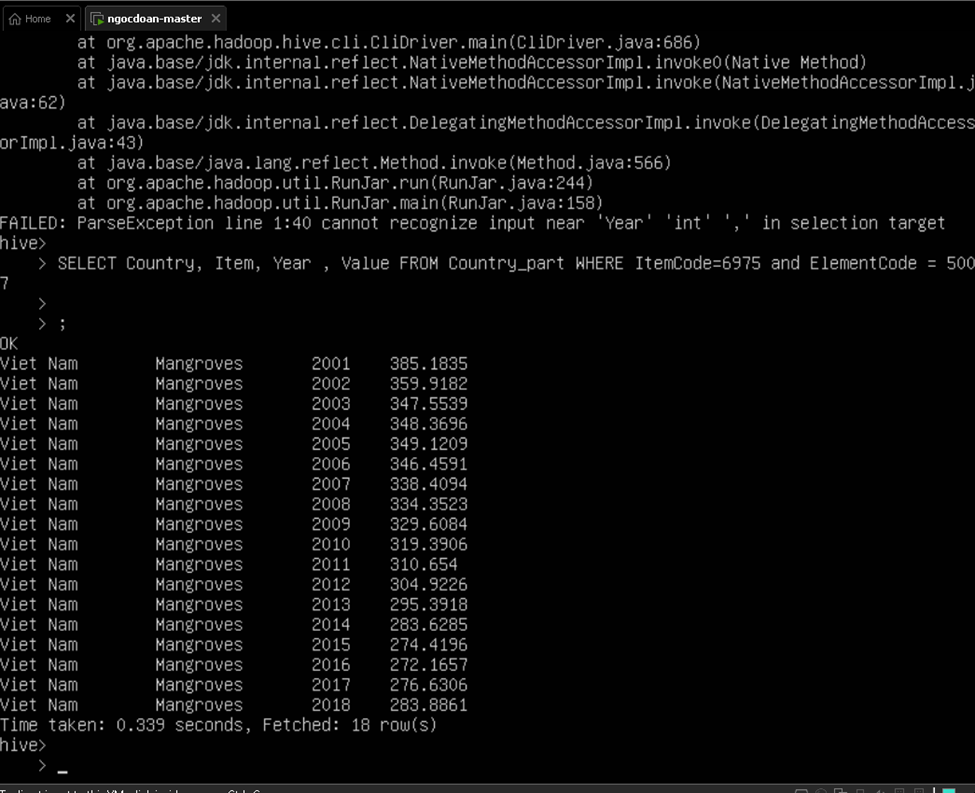
INSERT OVERWRITE TABLE Country\_part PARTITION(Country) SELECT Item,Year, Value, ItemCode, ElementCode, Country from dochephu where Country=”Viet Nam”;



SELECT Country, Item, Year , Value FROM Country\_part WHERE ItemCode=6975 and ElementCode = 5007

;

Kết quả hiển thị:



Qua kết quả thống kê trên ta thấy có sự sụt giảm về diện tích rừng ngập mặn từ năm 2006 – 2016. Sau đó đã có sự tăng trưởng nhẹ trở lại. Cho thấy được những cuộc vận động mở rộng diện tích rừng ngập mặn những năm gần đây của các cơ quan chức năng có hiệu quả.

Tham khảo

<https://www.tutorialspoint.com/cassandra/index.htm>

<https://github.com/kenhanscombe/project-cassandra>

<https://www.cloudduggu.com/cassandra/>

https://www.tutorialspoint.com/hive/index.htm