**Document Insert, Update, Delete**

**Cassandra Query Language**

**Author: Vũ Quang Sơn**

**Date created: 02-08-2016**

1. **Tool using for this document:**
2. NoSQL Manager for Cassandra:
   1. URL: <http://www.mongodbmanager.com/download>
   2. Some notes:
      1. This tool is commercial, you need to purchase 75$/year license
      2. Free trial 30 days
      3. There is problem with export csv or excel, the data is not correct, using command lines instead
      4. Some functions sometimes do not work as expected, using command lines instead
3. Alternative tool:
   1. DBeaver (<http://dbeaver.jkiss.org/>)
   2. Datastax DevCenter (<http://www.planetcassandra.org/cassandra/>)
4. Using commands line
   1. Using command line ***cqlsh*** on the server
   2. For import and export schema or data from cql using following commands:
      1. For exporting data to csv file:

***COPY keyspace.table1(column1, column2) TO 'temp.csv';***

* + 1. For import data from csv file:

***COPY keyspace.table2(column1, column2) FROM 'temp.csv';***

* + 1. To export keyspace schema

***cqlsh -e "DESC KEYSPACE user" > user\_schema.cql***

* + 1. To export entire datebase schema

***cqlsh -e "DESC SCHEMA" > db\_schema.cql***

* + 1. To import schema

***source 'user\_schema.cql'***

***source 'db\_schema.cql'***

1. **INSERT query in Cassandra:**
2. Syntax:

**INSERT** **INTO** [ *keyspace\_name*. ] *table\_name* ( *column\_name* [ , *column\_name* ] [ . . . ] )

**VALUES** ( *column\_value* [ , *column\_value* [ . . . ] )

[ **IF** **NOT** **EXISTS** ]

[ **USING** { **TTL** *time\_value* | **TIMESTAMP** *timestamp\_value* } ]

1. Contraint:
   1. The command must specify the row by primary key. If table already contains a row with the same primary key, the command writes (or overwrites) its values to the existing row. (Exception: if the command includes IF NOT EXISTS, and the row already exists, the command fails.)
   2. INSERT does not return any result unless it includes IF NOT EXISTS
   3. INSERT and UPDATE preform the same operation. A key difference: UPDATE supports counters, INSERT does not.
2. Parameters:
   1. *keyspace\_name*: name of keyspace – hocvalam
   2. *table\_name*: name of table – post
   3. *column\_name*: list of one or more columns - post\_id, created\_date
   4. *column\_value*:
      1. A Literal
         1. *blob*:

hexadecimal 0[xX](hex)+

* + - 1. *boolean*:

true or false, case-insensitive, without quotation

* + - 1. *numeric constant*:

format '-'?[0-9]+('.'[0-9]\*)?([eE][=-]?[0-9+])?

* + - 1. *identifier:*

Names of tables, columns, and other objects. If an identifier is not a keyspace or table name, any characters are allowed if enclosed in double quotation marks

* + - 1. *integer*
      2. *string literal:*

enclose in single quotation use '' to make dog plural: dog''s

* + - 1. *uuid:*

32 hex digits, 0-9 or a-f, separated by dashes, -, after the 8th, 12th, 16th, and 20th digits. For example: 01234567-0123-0123-0123-0123456789ab

* + - 1. *timeuuid:*

For example: d2177dd0-eaa2-11de-a572-001b779c76e3

* + - 1. *whitespace*
    1. A Set:

{ *literal* [, ... ] }

* + 1. A List:

[ *literal* [, ... ] ]

* + 1. A map collection:

{ *literal* : *literal* [, ... ] }

* 1. *time\_value:* The value for TTL is a number of seconds
  2. *timestamp\_value:* If TIMESTAMP parameter is used, the inserted column is marked with its value – a timestamp in microseconds
  3. *note:* 
     1. IF NOT EXISTS and USING TIMESTAMP cannot be used in the same INSERT
     2. Cannot insert multiple VALUES ( ), ( ), ( ) at a time => have to write multiple insert query

1. Practical
   1. Insert into hocvalam.post

INSERT INTO hocvalam.post (

author\_id,

date\_created,

post\_id,

content,

date\_updated,

images,

location,

publicity,

videos

)

VALUES (

3ba50468-0359-46c4-96ac-ec8b5c1c1ba0,

1470149492000,

89f777bf-ff44-43ad-b566-121cd2c91fa1,

'Sample Content',

1470149492000,

{2e9be4f6-9614-4dd6-8fb4-c43d01a2f2d1, 9c62f6d8-699a-4f5b-adf7-b92218f6ea4f },

{long:47.387215,lat:-61.97391, text:'6894 Freedom Circle, Santa Clara'},

'1',

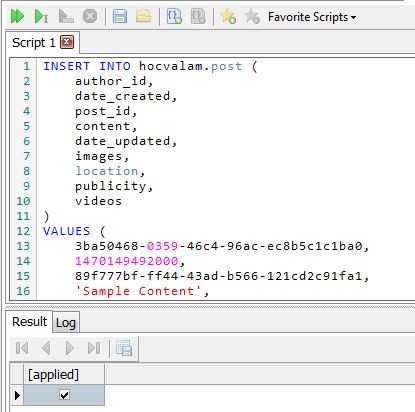
{c3918389-34cf-4685-9f5f-af758ab94980, f042d888-0ad0-4e77-8bbb-44a5d5c72f8c}

)

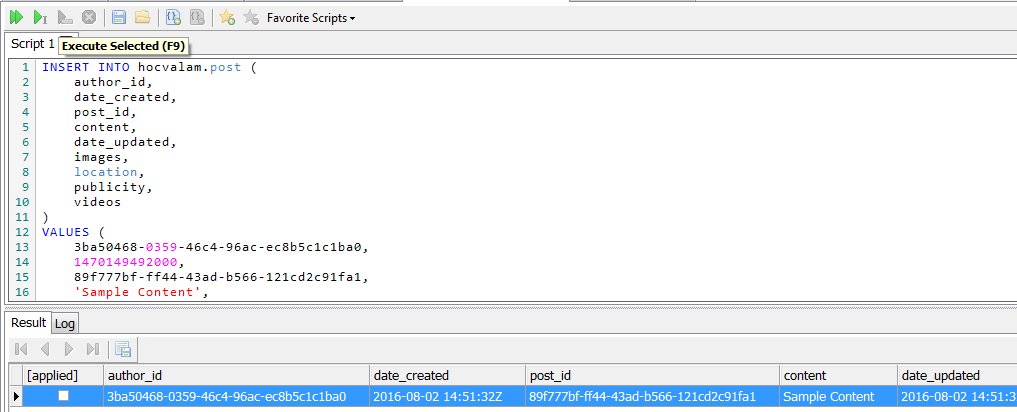
IF NOT EXISTS

USING TTL 86400;

* 1. Result with no existing record: return applied = yes



* 1. Result with existing record: return the existing record and applied = no



1. Some example about datatypes
   1. Query:

CREATE KEYSPACE simplex WITH replication = {

'class': 'SimpleStrategy',

'replication\_factor': 1

};

USE simplex;

CREATE TABLE values (

id int PRIMARY KEY,

bigint\_value bigint,

decimal\_value decimal,

double\_value double,

float\_value float,

int\_value int,

varint\_value varint,

timestamp\_value timestamp,

blob\_value blob,

uuid\_value uuid,

timeuuid\_value timeuuid,

inet\_value inet

);

INSERT INTO values (

id,

bigint\_value,

decimal\_value,

double\_value,

float\_value,

int\_value,

varint\_value,

timestamp\_value,

blob\_value,

uuid\_value,

timeuuid\_value,

inet\_value

)

VALUES (

0,

-765438000,

1313123123.234234234234234234123,

3.141592653589793,

3.14,

4,

67890656781923123918798273492834712837198237,

1425691864001,

varcharAsBlob('0x000000'),

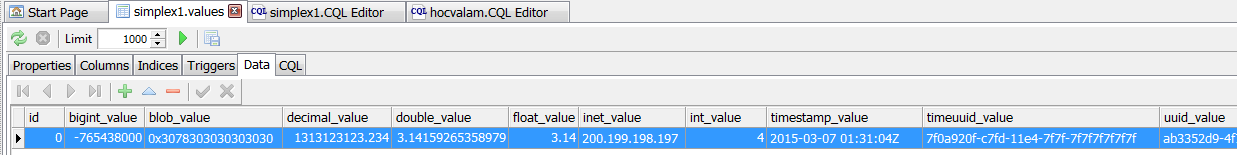
ab3352d9-4f7f-4007-a35a-e62aa7ab0b19,

maxTimeuuid('2015-03-11 14:47:10+0000'),

'200.199.198.197'

)

* 1. Result:



1. **DELETE query in Cassandra:**
   * + 1. Syntax:

**DELETE** [ *column\_name* [ , *column\_name* ] [ . . . ] | *column\_name* [ *term* ] ]

**FROM** [ *keyspace\_name*. ] *table\_name*

[ **USING** **TIMESTAMP** *timestamp\_value* ]

**WHERE** **row\_specification**

[ { **IF** **EXISTS** | **IF** **condition** [ **AND** **condition** ] [ . . . ] } ]

* + - 1. Parameters:
         1. *column\_name:* The name of the column / list columns to be deleted
         2. *term:*

If column\_name refers to a collection (a list or map), the parameter in parentheses indicates the term in the collection to be deleted. For each collection type, use term to specify:

list: index of the element in the list (the first item in the list has index 0, etc.)

map: the key of the element to be deleted

* + - * 1. *keyspace\_name:* name of keyspace – hocvalam
        2. *table\_name:* name of table – post
        3. *timestamp\_value:* If a TIMESTAMP is specified, the command only deletes elements older than the timestamp\_value (which match the WHERE and optional IF conditions)
        4. *row\_specification:*

The WHERE clause must identify the row or rows to be deleted by primary key

To specify one row, use primary\_key\_name = primary\_key\_value. If the primary key is a combination of elements, follow this with AND primary\_key\_name = primary\_key\_value

To specify more than one row, use primary\_key\_name IN ( primary\_key\_value, primary\_key\_value … )

Note: To delete a static column, only the partition key must be specified

* + - * 1. *IF EXISTS / IF condition:*

IF EXISTS to make the DELETE fail if there are no rows that match the WHERE conditions

Use IF to specify one or more conditions that must test true for the values in the specified row or rows

* + - * 1. *IF NOT EXISTScondition:* Use IF NOT EXISTS to make the DELETE fail if there are rows that match the WHERE conditions
      1. Practical:
         1. Delete column

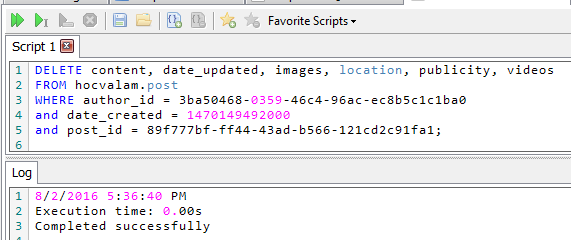
DELETE content, date\_updated, images, location, publicity, videos

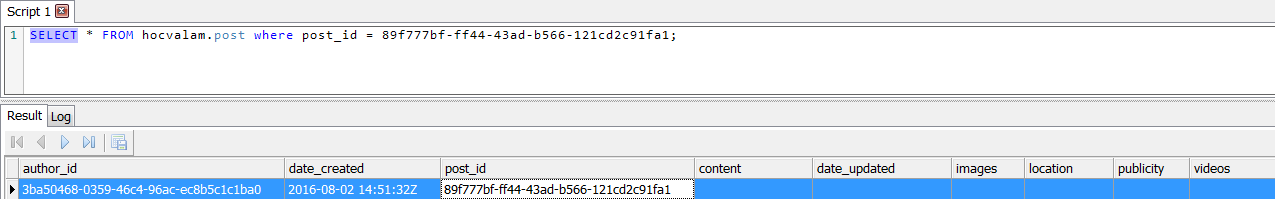
FROM hocvalam.post

WHERE author\_id = 3ba50468-0359-46c4-96ac-ec8b5c1c1ba0

and date\_created = 1470149492000

and post\_id = 89f777bf-ff44-43ad-b566-121cd2c91fa1;





* + - * 1. Delete entire row

DELETE FROM hocvalam.post

WHERE author\_id = 3ba50468-0359-46c4-96ac-ec8b5c1c1ba0 and date\_created = 1470149492000 and post\_id = 89f777bf-ff44-43ad-b566-121cd2c91fa1;

* + - * 1. Delete entire row with condition

DELETE FROM hocvalam.post WHERE author\_id = 3ba50468-0359-46c4-96ac-ec8b5c1c1ba0 and date\_created = 1470149492000 and post\_id = 89f777bf-ff44-43ad-b566-121cd2c91fa1

IF EXISTS;

1. **UPDATE query in Cassandra:**
2. Syntax:

**UPDATE** [ *keyspace\_name*. ] *table\_name*

[ **USING** { **TTL** *time\_value* | **TIMESTAMP** *timestamp\_value* } ]

**SET** **assignment** [, **assignment** ] [ . . . ]

**WHERE** **row\_specification**

[ { **IF** **EXISTS** | **condition** [ **AND** **condition** ] [ . . . ] } ]

1. Description
   1. UPDATE writes one or more column values to a row.
   2. If the specified row does not exist, the command creates it.
   3. UPDATE cannot update the values of a row's primary key fields.
   4. To update a counter column value in a counter table, specify the increment or decrement to the counter column.
   5. Note: Unlike the INSERT command, the UPDATE command supports counters. Otherwise, the UPDATE and INSERT operations are identical internally.