**Big Data Analytics**

**Experiment No. 02**

**Title :** To install and configure Cassandra to execute NoSQL commands.

**Theory :**

Cassandra is highly scalable; it allows to add more hardware to accommodate more customers and more data as per requirements. It accommodates all possible data formats including: structured, semi-structured and unstructured data. It can dynamically accommodate changes to our data structured according to need. It provides the flexibility to distribute data where you need by replicating data across multiple datacenters. Cassandra supports properties like Atomicity, Consistency, Isolation, Durability (ACID ).

Components of Cassandra -

Node : It is the place where data is stored.

Data center : It is a collection of related nodes.

Cluster : A cluster is a component that contains one or more data centers.

Commit log : The commit log is a crash- recovery mechanism in Cassandra. Every

write operation is written to the commit log.

Mem-table : A mem-table is a memory-resident data structure. After commit log,

the data will be written to the mem-table. Sometimes, for a single-column

family, there will be multiple mem-tables.

SSTable : It is a disk file to which the data is flushed from the mem-table when its

contents reach a threshold value.

Bloom filter : These are nothing but quick, nondeterministic, algorithms for

testing whether an element is a member of a set. It is a special kind of

cache. Bloom filters are accessed after every query.

Steps to Install :

Prerequisites -

* The latest version of Java 8, either the Oracle Java Standard Edition 8 or OpenJDK 8. To verify that you have the correct version of java installed, type java -version.
* For using cqlsh, the latest version of Python 2.7. To verify that you have the correct version of Python installed, type python --version.
* Create JAVA\_HOME environment variable

JAVA\_HOME =C:\Program Files\Java\jdk1.8.0\_221

* Also add following path to path variable
* C:\Program Files\Java\jdk1.8.0\_221\bin
* C:\Python27

Cassandra Installation -

* Download the latest stable release from the Apache Cassandra downloads websites
* [**http://cassandra.apache.org/**](http://cassandra.apache.org/)

[**https://dlcdn.apache.org/cassandra/3.11.13/apache-cassandra-3.11.13-bin.tar.gz**](https://dlcdn.apache.org/cassandra/3.11.13/apache-cassandra-3.11.13-bin.tar.gz)

* Extract apache-cassandra - 3.11.13-bin and copy apache - cassandra - 3.11.1 folder into the C:\ drive
* Open edit the system environment variables then Environment Variables in system variables add new path as C:\apache-cassandra-3.11.13\bin
* Open cmd and start cassandra database by

cassandra –f

* Open Windows PowerShell and Run as administrator and run

Set-ExecutionPolicy Unrestricted

A

* Again Open cmd and start cassandra database by

cassandra –f

r

* open new cmd and run

Cqlsh

Cqlsh> Type your command here

For example

CREATE KEYSPACE GIT WITH replication =

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

Commands :

**1. Creating a Keyspace**

A keyspace in Cassandra is a namespace that defines data replication on nodes. A cluster contains one keyspace per node.

Syntax: CREATE KEYSPACE <identifier> WITH <properties>

Example:

cqlsh> CREATE KEYSPACE Admin

WITH replication = {'class': ‘SimpleStrategy’, 'replication\_factor' : 3};

**2. Creating a Table**

You can create a table using the command CREATE TABLE. Given below is the syntax for creating a table.

Syntax:

CREATE (TABLE | COLUMNFAMILY) <tablename>

('<column-definition>' , '<column-definition>')

(WITH <option> AND <option>)

Example:

cqlsh> USE Admin;

cqlsh:Admin> CREATE TABLE emp(

emp\_id int PRIMARY KEY,

emp\_name text,

emp\_city text,

emp\_sal varint,

emp\_phone varint

);

**3. Creating Data in a Table**

You can insert data into the columns of a row in a table using the command INSERT.

Syntax:

INSERT INTO <tablename>

(<column1 name>, <column2 name>....)

VALUES (<value1>, <value2>....)

USING <option>

Example:

cqlsh:Admin> INSERT INTO emp (emp\_id, emp\_name, emp\_city,

emp\_phone, emp\_sal) VALUES(1,'ram', 'Hyderabad', 9848022338, 50000);

cqlsh:Admin> INSERT INTO emp (emp\_id, emp\_name, emp\_city,

emp\_phone, emp\_sal) VALUES(2,'robin', 'Hyderabad', 9848022339, 40000);

cqlsh:Admin> INSERT INTO emp (emp\_id, emp\_name, emp\_city,

emp\_phone, emp\_sal) VALUES(3,'rahman', 'Chennai', 9848022330, 45000);

**4. Updating Data in a Table**

UPDATE is the command used to update data in a table. The following keywords are used while updating data in a table

• Where − This clause is used to select the row to be updated.

• Set − Set the value using this keyword.

• Must − Includes all the columns composing the primary key.

While updating rows, if a given row is unavailable, then UPDATE creates a fresh row.

Syntax:

UPDATE <tablename>

SET <column name> = <new value>

<column name> = <value>....

WHERE <condition>

Example: cqlsh:Admin> UPDATE emp SET emp\_city='Delhi',emp\_sal=50000 WHERE emp\_id=2;

**5. Reading Data from Table**

SELECT clause is used to read data from a table in Cassandra. Using this clause, you can read a whole table, a single column, or a particular cell.

Syntax: SELECT FROM <tablename>

Example: cqlsh:Admin> select \* from emp;

emp\_id | emp\_city | emp\_name | emp\_phone | emp\_sal

--------+-----------+----------+------------+---------

1 | Hyderabad | ram | 9848022338 | 50000

2 | null | robin | 9848022339 | 50000

3 | Chennai | rahman | 9848022330 | 50000

4 | Pune | rajeev | 9848022331 | 30000

**6. Deleting Datafrom a Table**

You can delete data from a table using the command DELETE.

Syntax: DELETE FROM <identifier> WHERE <condition>;

Example: cqlsh:Admin> DELETE emp\_sal FROM emp WHERE emp\_id=3;

**Conclusion :**

We installed, configured and executed NoSQL commands in Cassandra.