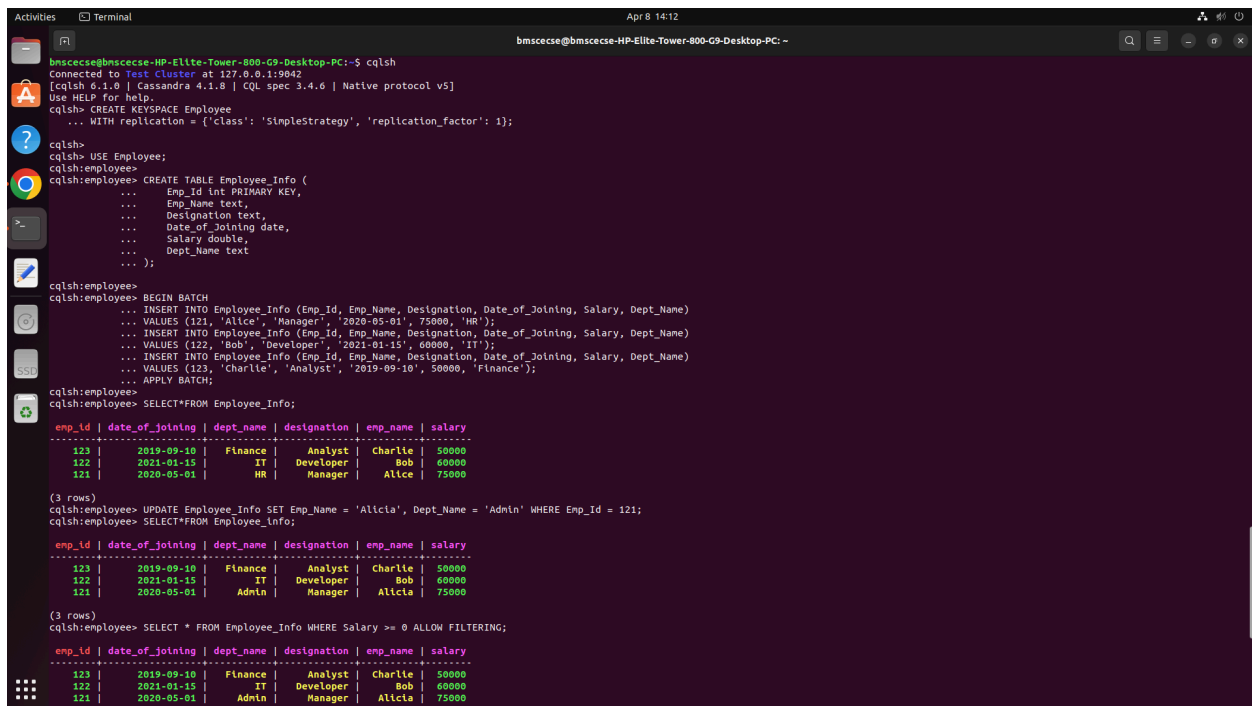# EXPERIMENT-3

Perform the following DB operations using Cassandra.
a) Create a keyspace by name Employee
b) Create a column family by name
Employee-Info with attributes
Emp_Id Primary Key, Emp_Name,
Designation, Date_of_Joining,
Salary, Dept_Name
c) Insert the values into the table in batch
d) Update Employee name and Department of Emp-Id
121
e) Sort the details of Employee records based on salary
f) Alter the schema of the table Employee_Info to add a
column Projects which stores a set of Projects done by
the corresponding Employee.
g) Update the altered table to add project names.
h) Create a TTL of 15 seconds to display the values of
Employees.
CODE WITH SCREENSHOTS:

bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC: ~

```
        121 |      2020-05-01 |        HR |   Manager |  Alice |  75000

(3 rows)
cqlsh:employee> UPDATE Employee_Info SET Emp_Name = 'Alicia', Dept_Name = 'Admin' WHERE Emp_Id = 121;
cqlsh:employee> SELECT*FROM Employee_info;

 emp_id | date_of_joining | dept_name | designation | emp_name | salary
--------+-----------------+-----------+-------------+----------+--------
    123 |      2019-09-10 |   Finance |     Analyst |  Charlie |  50000
    122 |      2021-01-15 |        IT |   Developer |      Bob |  60000
    121 |      2020-05-01 |     Admin |     Manager |   Alicia |  75000

(3 rows)
cqlsh:employee> SELECT * FROM Employee_Info WHERE Salary >= 0 ALLOW FILTERING;

 emp_id | date_of_joining | dept_name | designation | emp_name | salary
--------+-----------------+-----------+-------------+----------+--------
    123 |      2019-09-10 |   Finance |     Analyst |  Charlie |  50000
    122 |      2021-01-15 |        IT |   Developer |      Bob |  60000
    121 |      2020-05-01 |     Admin |     Manager |   Alicia |  75000

(3 rows)
cqlsh:employee> ALTER TABLE Employee_Info ADD Projects set<text>;
cqlsh:employee>
cqlsh:employee> SELECT*FROM Employee_Info;

 emp_id | date_of_joining | dept_name | designation | emp_name | projects | salary
--------+-----------------+-----------+-------------+----------+----------+--------
    123 |      2019-09-10 |   Finance |     Analyst |  Charlie |     null |  50000
    122 |      2021-01-15 |        IT |   Developer |      Bob |     null |  60000
    121 |      2020-05-01 |     Admin |     Manager |   Alicia |     null |  75000

(3 rows)
cqlsh:employee> UPDATE Employee_Info SET Projects = {'Payroll System', 'Attendance Tracker'} WHERE Emp_Id = 121;
cqlsh:employee> SELECT*FROM Employee_Info;

 emp_id | date_of_joining | dept_name | designation | emp_name | projects                                     | salary
--------+-----------------+-----------+-------------+----------+----------------------------------------------+--------
    123 |      2019-09-10 |   Finance |     Analyst |  Charlie |                                         null |  50000
    122 |      2021-01-15 |        IT |   Developer |      Bob |                                         null |  60000
    121 |      2020-05-01 |     Admin |     Manager |   Alicia | {'Attendance Tracker', 'Payroll System'}     |  75000

(3 rows)
cqlsh:employee> INSERT INTO Employee_Info (Emp_Id, Emp_Name, Designation, Date_Of_Joining, Salary, Dept_Name)
            ... VALUES (124, 'David', 'Tester', '2022-03-05', 45000, 'QA') USING TTL 15;
cqlsh:employee> select*from employee_info;

 emp_id | date_of_joining | dept_name | designation | emp_name | projects                                     | salary
--------+-----------------+-----------+-------------+----------+----------------------------------------------+--------
    123 |      2019-09-10 |   Finance |     Analyst |  Charlie |                                         null |  50000
    122 |      2021-01-15 |        IT |   Developer |      Bob |                                         null |  60000
    121 |      2020-05-01 |     Admin |     Manager |   Alicia | {'Attendance Tracker', 'Payroll System'}     |  75000

(3 rows)
cqlsh:employee> ▮
```

bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC: ~

```
bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.8 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> CREATE KEYSPACE Library
   ... WITH replication = {'class': 'SimpleStrategy', 'replication_factor': 1};
cqlsh> USE Library;
cqlsh:library> CREATE TABLE Library_Info (
           ...      Stud_Id int PRIMARY KEY,
           ...      Counter_value counter,
           ...      Stud_Name text,
           ...      Book_Name text,
           ...      Book_Id int,
           ...      Date_of_issue date
           ... );
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot mix counter and non counter columns in the same table"
cqlsh:library> CREATE TABLE Library_Info (
           ...      Stud_Id int PRIMARY KEY,
           ...      Stud_Name text,
           ...      Book_Name text,
           ...      Book_Id int,
           ...      Date_of_issue date
           ... );
cqlsh:library>
cqlsh:library> CREATE TABLE Library_Counter (
           ...      Stud_Id int PRIMARY KEY,
           ...      Counter_value counter
           ... );
cqlsh:library>
cqlsh:library> SELECT*FROM Library_Info;

 stud_id | book_id | book_name | date_of_issue | stud_name
---------+---------+-----------+---------------+-----------

(0 rows)
cqlsh:library> CREATE TABLE Library_Info1(
           ...      Stud_Id int,
           ...      Book_Name text,
           ...      Counter_Value counter,
           ...      PRIMARY KEY (Stud_Id, Book_Name)
           ... );
cqlsh:library> DROP TABLE LIBRARY_INFO;


cqlsh:library>
cqlsh:library>
cqlsh:library> DROP TABLE LIBRARY_INFO1;
cqlsh:library> CREATE TABLE Library_Info (
           ...      Stud_Id int,
           ...      Book_Name text,
           ...      Counter_Value counter,
           ...      PRIMARY KEY (Stud_Id, Book_Name)
           ... );
cqlsh:library>
```

```
...          Counter_Value counter,
...          PRIMARY KEY (Stud_Id, Book_Name)
... );

cqlsh:library>
cqlsh:library> BEGIN BATCH
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 112 AND Book_Name = 'BDA';
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 113 AND Book_Name = 'DSA';
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 112 AND Book_Name = 'BDA';
           ... APPLY BATCH;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot include a counter statement in a logged batch"
cqlsh:library> BEGIN UNLOGGED BATCH
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 112 AND Book_Name = 'BDA';
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 113 AND Book_Name = 'DSA';
           ... UPDATE Library_Info SET Counter_Value = Counter_Value + 1 WHERE Stud_Id = 112 AND Book_Name = 'BDA';
           ... APPLY BATCH;
cqlsh:library> SELECT*FROM Library_info;

 stud_id | book_name | counter_value
---------+-----------+---------------
     113 |       DSA |             1
     112 |       BDA |             2

(2 rows)
cqlsh:library> SELECT * FROM Library_Info WHERE Stud_Id = 112 AND Book_Name = 'BDA';

 stud_id | book_name | counter_value
---------+-----------+---------------
     112 |       BDA |             2

(1 rows)
cqlsh:library> COPY Library_Info TO '/home/your_username/library_info.csv' WITH HEADER = TRUE;
Using 16 child processes
Can't open '/home/your_username/library_info.csv' for writing: [Errno 2] No such file or directory: '/home/your_username/library_info.csv'
cqlsh:library> COPY library_info (stud_id, book_name, counter_value) TO '/home/bmscecse/library_info.csv';
Using 16 child processes

Starting copy of library.library_info with columns [stud_id, book_name, counter_value].
Processed: 2 rows; Rate:      22 rows/s; Avg. rate:      22 rows/s
2 rows exported to 1 files in 0.105 seconds.
cqlsh:library> COPY library_info (stud_id, book_name, counter_value) FROM '/home/bmscecse/library_info.csv' WITH HEADER = true;
Using 16 child processes

Starting copy of library.library_info with columns [stud_id, book_name, counter_value].
Processed: 1 rows; Rate:       2 rows/s; Avg. rate:       3 rows/s
1 rows imported from 1 files in 0.358 seconds (0 skipped).
cqlsh:library> select*from library.library_info;

 stud_id | book_name | counter_value
---------+-----------+---------------
     113 |       DSA |             1
     112 |       BDA |             4

(2 rows)
cqlsh:library>
```