LAB 6 QUESTION 1:

1. Create a keyspace by name Employee

```
cqlsh> CREATE KEYSPACE "employee" WITH replication={'class':'SimpleStrategy', 'replication_fact
or':1};
cqlsh> DESCRIBE keyspaces;

"Student"       system_auth  student             employee
system_schema   system       system_distributed  system_traces
```

2. Create a column family by name Employee-Info with attributes Emp_Id Primary Key, Emp_Name, Designation, Date_of_Joining, Salary, Dept_Name

```
cqlsh> USE employee;
cqlsh:employee> CREATE COLUMNFAMILY employeeInfo(
            ... employeeID INT PRIMARY KEY,
            ... name TEXT,
            ... designation TEXT,
            ... dateofJoining DATE,
            ... salary FLOAT,
            ... departmentName TEXT
            ... );
```

3. Insert the values into the table in batch

```
cqlsh:employee> BEGIN BATCH
            ... INSERT INTO employeeInfo(employeeID,name,designation,dateofJoining,salary,depar
tmentName)VALUES(121,'Divyanshu','SDE 1','2020-3-21',20000,'Software');
            ... INSERT INTO employeeInfo(employeeID,name,designation,dateofJoining,salary,depar
tmentName)VALUES(122,'Ajith','Power Engineer','2020-6-23',20000,'Electric');
            ... INSERT INTO employeeInfo(employeeID,name,designation,dateofJoining,salary,depar
tmentName)VALUES(123,'Asish','Network Engineer','2020-7-12',40000,'Hardware');
            ... INSERT INTO employeeInfo(employeeID,name,designation,dateofJoining,salary,depar
tmentName)VALUES(124,'Alok','Analyst','2020-9-28',50000,'Security');
            ... APPLY BATCH;
cqlsh:employee> SELECT * FROM employeeInfo;

 employeeid | dateofjoining | departmentname | designation      | name      | salary
------------+---------------+----------------+------------------+-----------+--------
        123 |    2020-07-12 |       Hardware | Network Engineer |     Asish |  40000
        122 |    2020-06-23 |       Electric |   Power Engineer |     Ajith |  20000
        121 |    2020-03-21 |       Software |            SDE 1 | Divyanshu |  20000
        124 |    2020-09-28 |       Security |          Analyst |      Alok |  50000

(4 rows)
```

4. Update Employee name and Department of Emp-Id 121

```
cqlsh:employee> UPDATE employeeInfo SET name='John', department='Law' WHERE employeeID=121;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Undefined column name dep
artment"
cqlsh:employee> UPDATE employeeInfo SET name='John', departmentname='Law' WHERE employeeID=121;
cqlsh:employee> SELECT * FROM employeeInfo;

 employeeid | dateofjoining | departmentname | designation      | name  | salary
------------+---------------+----------------+------------------+-------+--------
        123 |    2020-07-12 |       Hardware | Network Engineer | Asish |  40000
        122 |    2020-06-23 |       Electric |   Power Engineer | Ajith |  20000
        121 |    2020-03-21 |            Law |            SDE 1 |  John |  20000
        124 |    2020-09-28 |       Security |          Analyst |  Alok |  50000

(4 rows)
```

5. Alter the schema of the table Employee_Info to add a column Projects which stores a set of Projects done by the corresponding Employee.
6. Update the altered table to add project names.

```
cqlsh:employee> ALTER TABLE employeeInfo
           ... ADD projects set<text>;
cqlsh:employee> UPDATE employeeInfo
           ... SET projects={'Information Security','Networks'}
           ... WHERE employeeID=121;
cqlsh:employee> SELECT * FROM employeeInfo;

 employeeid | dateofjoining | departmentname | designation      | name  | projects
            | salary
------------+---------------+----------------+------------------+-------+--------------------
------------+--------
        123 |    2020-07-12 |       Hardware | Network Engineer | Asish |
          null |  40000
        122 |    2020-06-23 |       Electric |   Power Engineer | Ajith |
          null |  20000
        121 |    2020-03-21 |            Law |            SDE 1 |  John | {'Information Securit
y', 'Networks'} |  20000
        124 |    2020-09-28 |       Security |          Analyst |  Alok |
          null |  50000

(4 rows)
```

7. Create a TTL of 15 seconds to display the values of Employees.

```
cqlsh:employee> INSERT INTO employeeInfo (employeeId,name,designation,dateofJoining,salary,depa
rtmentname)VALUES(126,'Jane','Web Developer','2020-2-13',30000,'IT')USING TTL 15;
cqlsh:employee> SELECT * FROM employeeInfo;

 employeeid | dateofjoining | departmentname | designation      | name  | projects
            | salary
------------+---------------+----------------+------------------+-------+--------------------
------------+--------
        123 |    2020-07-12 |       Hardware | Network Engineer | Asish |
          null |  40000
        122 |    2020-06-23 |       Electric |   Power Engineer | Ajith |
          null |  20000
        121 |    2020-03-21 |            Law |            SDE 1 |  John | {'Information Securit
y', 'Networks'} |  20000
        126 |    2020-02-13 |             IT |    Web Developer |  Jane |
          null |  30000
        124 |    2020-09-28 |       Security |          Analyst |  Alok |
          null |  50000

(5 rows)
cqlsh:employee>
```