Perform the following DB operations using Cassandra.

1.Create a keyspace by name Employee

```
create keyspace Employee with replication = {
   ... 'class':'SimpleStrategy',
   ... 'replication_factor':1
   ... };
```

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

```
create table employee_info ( Employee_id int PRIMARY KEY, Emp_name text, Designation text,
Dateofjoining timestamp, Salary double, Dept_name text);
```

3. Insert the values into the table in batch

```
BEGIN BATCH insert into employee_info (Employee_id, Emp_name, Designation,
Dateofjoining, Salary, Dept_name) values (1, 'Michael Storm', 'HR', '2018-09-22', 200000, 'A')
insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary,
Dept_name) values (2, 'Stephen Fox', 'Manager', '2019-05-02', 150000, 'B') insert into
employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name)
values (3, 'David', 'AVP', '2019-03-01', 500000, 'C') insert into employee_info (Employee_id,
Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (121, 'Ian', 'AVP',
'2018-03-10', 250000, 'D') APPLY BATCH;
```

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

```
UPDATE employee_info SET Emp_name = 'Charlie', Dept_name = 'E' WHERE
Employee_id=121;
```

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 4.1.4 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> create keyspace Employee with replication = {
   ... 'class':'SimpleStratergy',
   ... 'replication_factor':1
   ... };
ConfigurationException: Unable to find replication strategy class 'org.apache.cassandra.locator.SimpleStratergy'
cqlsh> create keyspace Employee with replication = { 'class':'SimpleStrategy', 'replication_factor':1 };
cqlsh> USE Employee;
cqlsh:employee> Create table employee_info(
   ... Employee_id PRIMARY KEY,
   ...
cqlsh:employee> create table employee_info ( Employee_id int PRIMARY KEY, Emp_name text, Designation text, Dateofjoining timestamp, Salary double, Dept_name text);
cqlsh:employee> describe Employee_info

CREATE TABLE employee.employee_info (
    employee_id int PRIMARY KEY,
    dateofjoining timestamp,
    dept_name text,
    designation text,
    emp_name text,
    salary double
) WITH additional_write_policy = '99p'
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND cdc = false
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32', 'min_threshold': '4'}
    AND compression = {'chunk_length_in_kb': '16', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
    AND memtable = 'default'
    AND crc_check_chance = 1.0
    AND default_time_to_live = 0
    AND extensions = {}
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair = 'BLOCKING'
    AND speculative_retry = '99p';
cqlsh:employee> describe table employee_info

CREATE TABLE employee.employee_info (
    employee_id int PRIMARY KEY,
    dateofjoining timestamp,
    dept_name text,
    designation text,
    emp_name text,
    salary double
) WITH additional_write_policy = '99p'
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND cdc = false
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max_threshold': '32', 'min_threshold': '4'}
```

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

```
    AND extensions = {}
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair = 'BLOCKING'
    AND speculative_retry = '99p';
cqlsh:employee> insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (1, 'Michael Storm', 'HR', '2018-09-22', 200000, 'A')
   ... insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (2, 'Stephen Fox', 'Manager', '2019-05-02', 150000, 'B')
   ... insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (3, 'David', 'AVP', '2019-03-01', 500000, 'C')
   ... insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (121, 'Ian', 'AVP', '2018-03-10', 250000, 'D')
   ... APPLY BATCH;
SyntaxException: line 2:0 mismatched input 'insert' expecting EOF (... '2018-09-22', 200000, 'A')[insert]...)
cqlsh:employee> BEGIN BATCH insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (1, 'Michael Storm', 'HR', '2018-09-22', 200000, 'A') insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (2, 'Stephen Fox', 'Manager', '2019-05-02', 150000, 'B') insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (3, 'David', 'AVP', '2019-03-01', 500000, 'C') insert into employee_info (Employee_id, Emp_name, Designation, Dateofjoining, Salary, Dept_name) values (121, 'Ian', 'AVP', '2018-03-10', 250000, 'D') APPLY BATCH;
cqlsh:employee> select * from employee_info
   ... select * from employee_info
cqlsh:employee> select * from employee_info;

 employee_id | dateofjoining                   | dept_name | designation | emp_name      | salary
-------------+---------------------------------+-----------+-------------+---------------+--------
           1 | 2018-09-21 18:30:00.000000+0000 |         A |          HR | Michael Storm |  2e+05
           2 | 2019-05-01 18:30:00.000000+0000 |         B |     Manager |   Stephen Fox | 1.5e+05
         121 | 2018-03-09 18:30:00.000000+0000 |         D |         AVP |           Ian | 2.5e+05
           3 | 2019-02-28 18:30:00.000000+0000 |         C |         AVP |         David |  5e+05

(4 rows)
cqlsh:employee> update employee_info SET Emp_name = 'Charlie', Dept = 'E' WHERE Employee_id=121;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Undefined column name dept in table employee.employee_info"
cqlsh:employee> update employee_info SET Emp_name = 'Charlie' Dept = 'E' WHERE Employee_id=121;
SyntaxException: line 1:46 no viable alternative at input 'Dept' (...employee_info SET Emp_name = 'Charlie' [Dept]...)
cqlsh:employee> update employee_info SET Emp_name = 'Charlie' SET  Dept = 'E' WHERE Employee_id=121;
SyntaxException: line 1:46 no viable alternative at input 'SET' (...employee_info SET Emp_name = 'Charlie' [SET]...)
cqlsh:employee> UPDATE employee_info SET Emp_name = 'Charlie' SET  Dept = 'E' WHERE Employee_id=121;
SyntaxException: line 1:46 no viable alternative at input 'SET' (...employee_info SET Emp_name = 'Charlie' [SET]...)
cqlsh:employee> UPDATE employee_info SET Emp_name = 'Charlie'  Dept = 'E' WHERE Employee_id=121;
SyntaxException: line 1:47 no viable alternative at input 'Dept' (...employee_info SET Emp_name = 'Charlie'  [Dept]...)
cqlsh:employee> UPDATE employee_info SET Emp_name = 'Charlie',  Dept_name = 'E' WHERE Employee_id=121;
SyntaxException: line 1:47 no viable alternative at input 'Dept_name' (...employee_info SET Emp_name = 'Charlie'  [Dept_name]...)
cqlsh:employee> UPDATE employee_info SET Emp_name = 'Charlie', Dept_name = 'E' WHERE Employee_id=121;
cqlsh:employee> UPDATE employee_info SET Emp_name = 'Charlie', Dept_name = 'E' WHERE Employee_id=121;
cqlsh:employee> select * from employee_info;

 employee_id | dateofjoining                   | dept_name | designation | emp_name      | salary
-------------+---------------------------------+-----------+-------------+---------------+--------
           1 | 2018-09-21 18:30:00.000000+0000 |         A |          HR | Michael Storm |  2e+05
           2 | 2019-05-01 18:30:00.000000+0000 |         B |     Manager |   Stephen Fox | 1.5e+05
         121 | 2018-03-09 18:30:00.000000+0000 |         E |         AVP |       Charlie | 2.5e+05
           3 | 2019-02-28 18:30:00.000000+0000 |         C |         AVP |         David |  5e+05

(4 rows)
cqlsh:employee> select * from employee_info order by salary;
InvalidRequest: Error from server: code=2200 [Invalid query] message="ORDER BY is only supported when the partition key is restricted by an EQ or an IN."
cqlsh:employee> select * from employee_info order by salary ASC;
InvalidRequest: Error from server: code=2200 [Invalid query] message="ORDER BY is only supported when the partition key is restricted by an EQ or an IN."
```