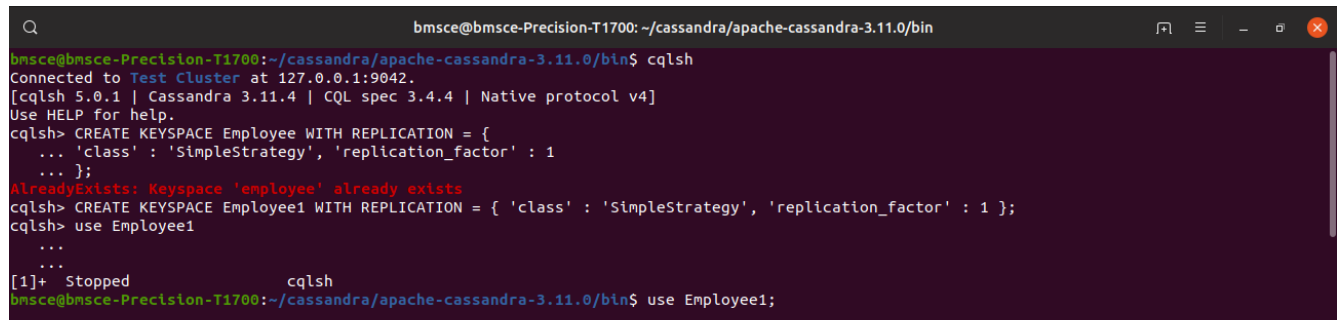


Program 1. Perform the following DB operations using Cassandra.

1. Create a key space by name Employee

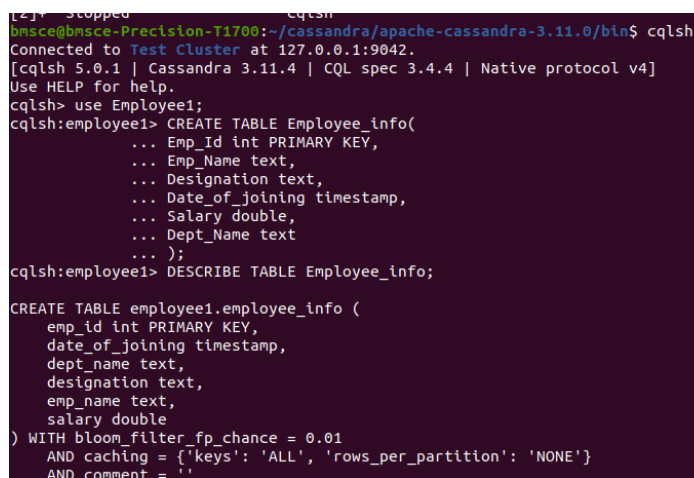
```
CREATE KEYSPACE Employee WITH REPLICATION = {  
  ... 'class' : 'SimpleStrategy', 'replication_factor' : 1  
  ... };
```



```
bmsce@bmsce-Precision-T1700: ~/cassandra/apache-cassandra-3.11.0/bin$ cqlsh  
Connected to Test Cluster at 127.0.0.1:9042.  
[cqlsh 5.0.1 | Cassandra 3.11.4 | CQL spec 3.4.4 | Native protocol v4]  
Use HELP for help.  
cqlsh> CREATE KEYSPACE Employee WITH REPLICATION = {  
  ... 'class' : 'SimpleStrategy', 'replication_factor' : 1  
  ... };  
AlreadyExists: Keyspace 'employee' already exists  
cqlsh> CREATE KEYSPACE Employee1 WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 1 };  
cqlsh> use Employee1  
...  
[1]+  Stopped                  cqlsh  
bmsce@bmsce-Precision-T1700: ~/cassandra/apache-cassandra-3.11.0/bin$ use Employee1;
```

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

```
CREATE TABLE Employee_info(  
  ... Emp_Id int PRIMARY KEY,  
  ... Emp_Name text,  
  ... Designation text,  
  ... Date_of_joining timestamp,  
  ... Salary double,  
  ... Dept_Name text  
  ... );
```



```
[2]+  Stopped                  cqlsh  
bmsce@bmsce-Precision-T1700: ~/cassandra/apache-cassandra-3.11.0/bin$ cqlsh  
Connected to Test Cluster at 127.0.0.1:9042.  
[cqlsh 5.0.1 | Cassandra 3.11.4 | CQL spec 3.4.4 | Native protocol v4]  
Use HELP for help.  
cqlsh> use Employee1;  
cqlsh:employee1> CREATE TABLE Employee_info(  
  ... Emp_Id int PRIMARY KEY,  
  ... Emp_Name text,  
  ... Designation text,  
  ... Date_of_joining timestamp,  
  ... Salary double,  
  ... Dept_Name text  
  ... );  
cqlsh:employee1> DESCRIBE TABLE Employee_info;  
  
CREATE TABLE employee1.employee_info (  
  emp_id int PRIMARY KEY,  
  date_of_joining timestamp,  
  dept_name text,  
  designation text,  
  emp_name text,  
  salary double  
) WITH bloom_filter_fp_chance = 0.01  
   AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}  
   AND comment = ''
```

3. Insert the values into the table in batch

```
BEGIN BATCH
... INSERT INTO
Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name)
... VALUES(1,'Prema','Manager','2023-08-28','50000','Executive')
... INSERT INTO
Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name)
... VALUES(121,'Pooja','Product-Manager','2023-08-28','60000','Product-Management')
... INSERT INTO
Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name)
... VALUES(118,'Prathiksha','Design-Lead','2023-09-28','70000','Design')
... INSERT INTO
Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name)
... VALUES(119,'Prathibha','CTO','2024-09-28','70000','Development')
... INSERT INTO
Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name)
... VALUES(116,'Prateek.G','Sales-Manager','2022-09-28','80000','Sales')
... APPLY BATCH;
```

```
cqlsh:employee> BEGIN BATCH INSERT INTO Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name) VALUES(1,'Prema','Manager','2023-08-28','50000','Executive') INSERT INTO Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name) VALUES(121,'Pooja','Product-Manager','2023-08-28','60000','Product-Management') INSERT INTO Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name) VALUES(118,'Prathiksha','Design-Lead','2023-09-28','70000','Design') INSERT INTO Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name) VALUES(119,'Prathibha','CTO','2024-09-28','70000','Development') INSERT INTO Employee_info(Emp_Id,Emp_Name,Designation,Date_of_joining,Salary,Dept_Name) VALUES(116,'Prateek.G','Sales-Manager','2022-09-28','80000','Sales') APPLY BATCH;
cqlsh:employee> SELECT * FROM Employee_info;
```

emp_id	date_of_joining	dept_name	designation	emp_name	salary
118	2023-09-28	Design	Design-Lead	Prathiksha	70000
1	2023-08-28	Executive	Manager	Prema	50000
116	2022-09-28	Sales	Sales-Manager	Prateek.G	80000
121	2023-08-28	Product-Management	Product-Manager	Pooja	60000
119	2024-09-28	Development	CTO	Prathibha	70000

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

```
UPDATE Employee_info SET Emp_Name = 'Chinmayee',Dept_Name = 'Sales'
WHERE Emp_Id = 121;
```

```
cqlsh:employee> UPDATE Employee_info SET Emp_Name = 'Chinmayee',Dept_Name = 'Sales' WHERE Emp_Id = 121;
cqlsh:employee> SELECT * FROM Employee_info;
```

emp_id	date_of_joining	dept_name	designation	emp_name	salary
118	2023-09-28	Design	Design-Lead	Prathiksha	70000
1	2023-08-28	Executive	Manager	Prema	50000
116	2022-09-28	Sales	Sales-Manager	Prateek.G	80000
121	2023-08-28	Sales	Product-Manager	Chinmayee	60000
119	2024-09-28	Development	CTO	Prathibha	70000

(5 rows)
cqlsh:employee> █

5. Sort the details of Employee records based on salary

6. Alter the schema of the table Employee_Info to add a column Projects which

stores a set of Projects done by the corresponding Employee.

7. Update the altered table to add project names.

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