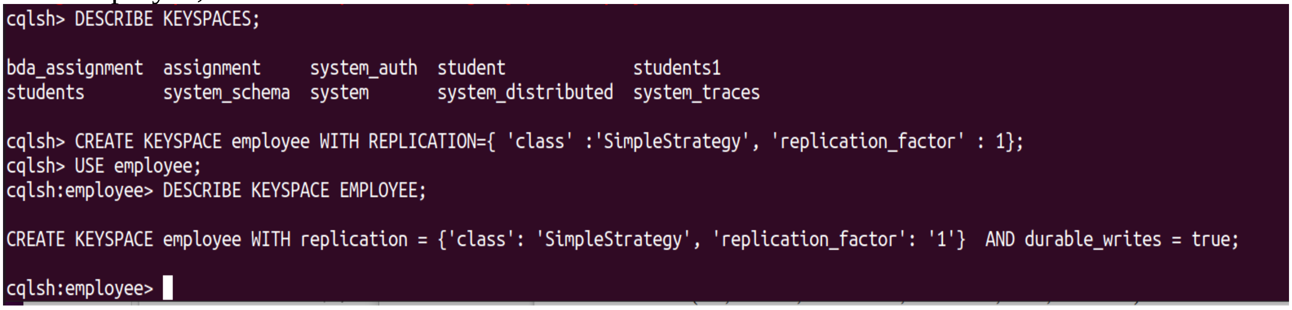
1BM18CS089
SAHANA L

1. Create a keyspace by name Employee

CREATE KEYSPACE employee WITH REPLICATION={ 'class' : 'SimpleStrategy', 'replication_factor' : 1};
USE employee;



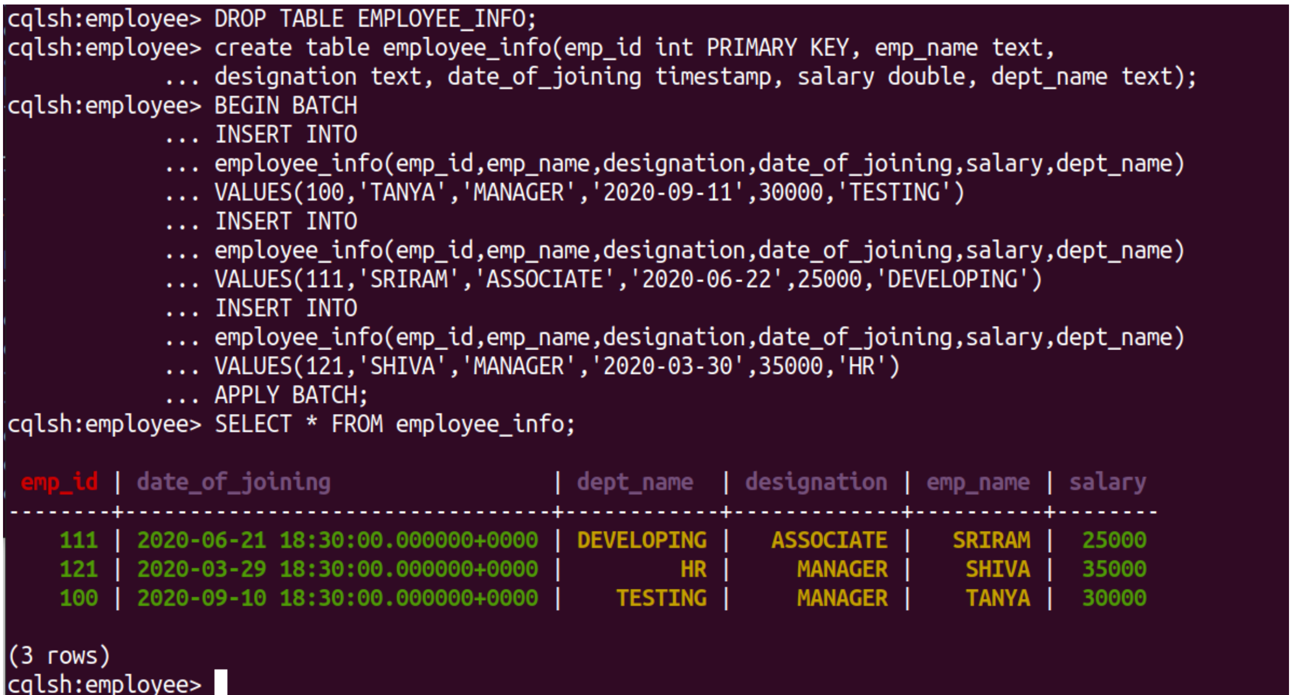
```
cqlsh> DESCRIBE KEYSPACES;

bda_assignment  assignment     system_auth  student             students1
students        system_schema  system       system_distributed  system_traces

cqlsh> CREATE KEYSPACE employee WITH REPLICATION={ 'class' :'SimpleStrategy', 'replication_factor' : 1};
cqlsh> USE employee;
cqlsh:employee> DESCRIBE KEYSPACE EMPLOYEE;

CREATE KEYSPACE employee WITH replication = {'class': 'SimpleStrategy', 'replication_factor': '1'}  AND durable_writes = true;

cqlsh:employee> 
```

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);

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(100,'TANYA','MANAGER','2020-09-11',30000,'TESTING')
... INSERT INTO
employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
VALUES(111,'SRIRAM','ASSOCIATE','2020-06-22',25000,'DEVELOPING')
... INSERT INTO
employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
VALUES(121,'SHIVA','MANAGER','2020-03-30',35000,'HR')
... APPLY BATCH;
SELECT * FROM employee_info;



```
cqlsh:employee> DROP TABLE EMPLOYEE_INFO;
cqlsh:employee> create table employee_info(emp_id int PRIMARY KEY, emp_name text,
            ... designation text, date_of_joining timestamp, salary double, dept_name text);
cqlsh:employee> BEGIN BATCH
            ... INSERT INTO
            ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
            ... VALUES(100,'TANYA','MANAGER','2020-09-11',30000,'TESTING')
            ... INSERT INTO
            ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
            ... VALUES(111,'SRIRAM','ASSOCIATE','2020-06-22',25000,'DEVELOPING')
            ... INSERT INTO
            ... employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
            ... VALUES(121,'SHIVA','MANAGER','2020-03-30',35000,'HR')
            ... APPLY BATCH;
cqlsh:employee> SELECT * FROM employee_info;

 emp_id | date_of_joining                 | dept_name  | designation | emp_name | salary
--------+---------------------------------+------------+-------------+----------+--------
    111 | 2020-06-21 18:30:00.000000+0000 | DEVELOPING |   ASSOCIATE |   SRIRAM |  25000
    121 | 2020-03-29 18:30:00.000000+0000 |         HR |     MANAGER |    SHIVA |  35000
    100 | 2020-09-10 18:30:00.000000+0000 |    TESTING |     MANAGER |    TANYA |  30000

(3 rows)
cqlsh:employee> 
```

1BM18CS089
SAHANA L

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

UPDATE employee_info SET emp_name = 'SHAAN' WHERE emp_id = 121;
SELECT * FROM employee_info;

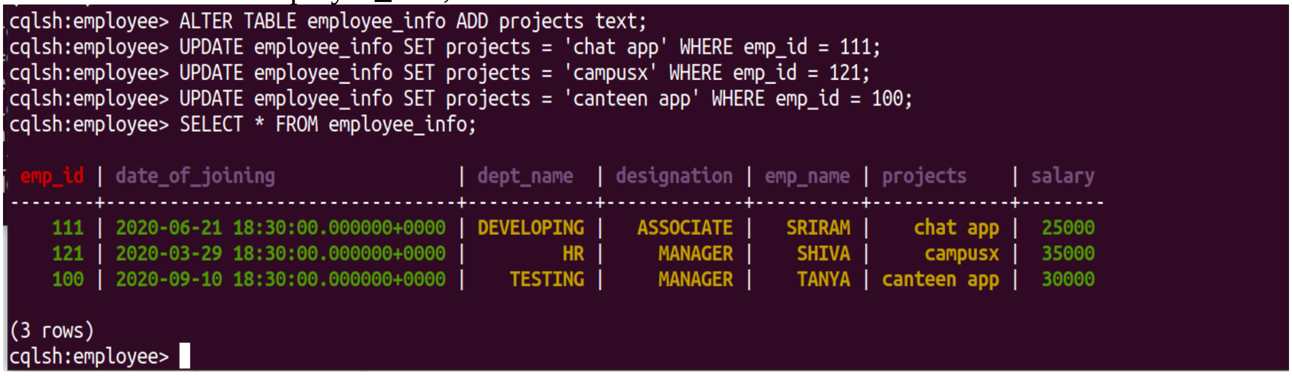
5. Sort the details of Employee records based on salary

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.

ALTER TABLE employee_info ADD projects text;

7. Update the altered table to add project names.

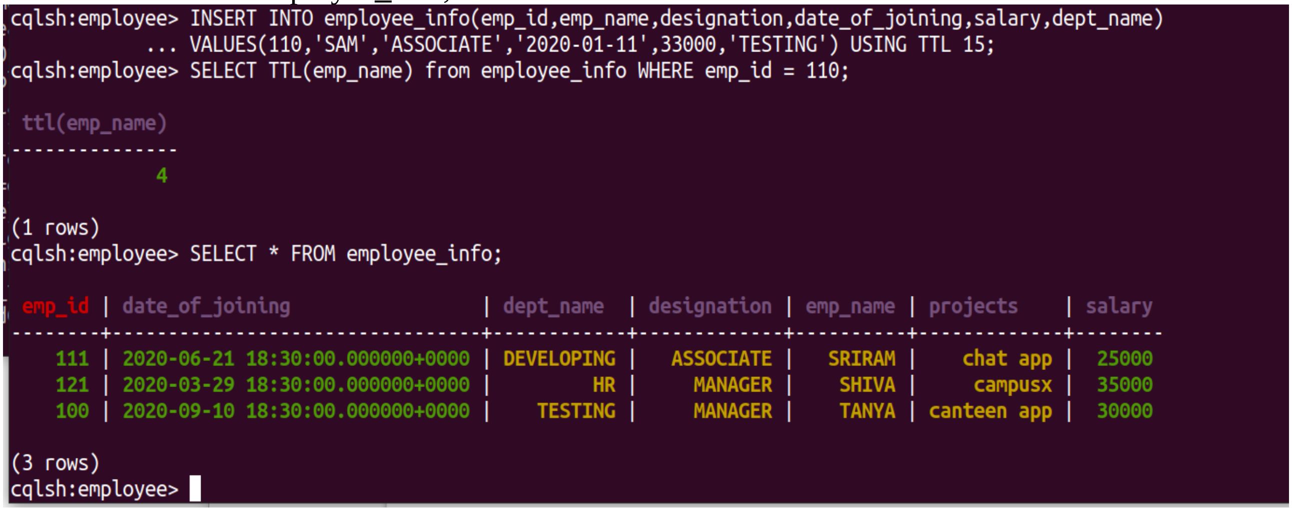
UPDATE employee_info SET projects = 'chat app' WHERE emp_id = 111;
UPDATE employee_info SET projects = 'campusx' WHERE emp_id = 121;
UPDATE employee_info SET projects = 'canteen app' WHERE emp_id = 100;
SELECT * FROM employee_info;



```
cqlsh:employee> ALTER TABLE employee_info ADD projects text;
cqlsh:employee> UPDATE employee_info SET projects = 'chat app' WHERE emp_id = 111;
cqlsh:employee> UPDATE employee_info SET projects = 'campusx' WHERE emp_id = 121;
cqlsh:employee> UPDATE employee_info SET projects = 'canteen app' WHERE emp_id = 100;
cqlsh:employee> SELECT * FROM employee_info;

 emp_id | date_of_joining                 | dept_name  | designation | emp_name | projects    | salary
--------+---------------------------------+------------+-------------+----------+-------------+--------
    111 | 2020-06-21 18:30:00.000000+0000 | DEVELOPING |   ASSOCIATE |   SRIRAM |    chat app |  25000
    121 | 2020-03-29 18:30:00.000000+0000 |         HR |     MANAGER |    SHIVA |     campusx |  35000
    100 | 2020-09-10 18:30:00.000000+0000 |    TESTING |     MANAGER |    TANYA | canteen app |  30000

(3 rows)
cqlsh:employee>
```

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

INSERT INTO
employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
VALUES(110,'SAM','ASSOCIATE','2020-01-11',33000,'TESTING') USING TTL 15;
SELECT TTL(emp_name) from employee_info WHERE emp_id = 110;
SELECT * FROM employee_info;



```
cqlsh:employee> INSERT INTO employee_info(emp_id,emp_name,designation,date_of_joining,salary,dept_name)
            ... VALUES(110,'SAM','ASSOCIATE','2020-01-11',33000,'TESTING') USING TTL 15;
cqlsh:employee> SELECT TTL(emp_name) from employee_info WHERE emp_id = 110;

 ttl(emp_name)
---------------
             4

(1 rows)
cqlsh:employee> SELECT * FROM employee_info;

 emp_id | date_of_joining                 | dept_name  | designation | emp_name | projects    | salary
--------+---------------------------------+------------+-------------+----------+-------------+--------
    111 | 2020-06-21 18:30:00.000000+0000 | DEVELOPING |   ASSOCIATE |   SRIRAM |    chat app |  25000
    121 | 2020-03-29 18:30:00.000000+0000 |         HR |     MANAGER |    SHIVA |     campusx |  35000
    100 | 2020-09-10 18:30:00.000000+0000 |    TESTING |     MANAGER |    TANYA | canteen app |  30000

(3 rows)
cqlsh:employee>
```