

EMPLOYEE KEYSpace

```
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ java -version
openjdk version "11.0.21" 2023-10-17
OpenJDK Runtime Environment (build 11.0.21+9-post-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.21+9-post-Ubuntu-0ubuntu122.04, mixed mode, sharing)
bmscecse@bmscecse-HP-Elite-Tower-600-G9-Desktop-PC:~$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.0.0 | Cassandra 4.0.5 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh> CREATE KEYSPACE Employee WITH replication =
... { 'class' : 'SimpleStrategy',
... 'replication_factor':1};
AlreadyExists: Keyspace 'employee' already exists
cqlsh> DESCRIBE KEYSPACES;

bemployee  cycling  my_keyspace  system_auth      system_views
bstudents   employee  students    system_distributed  system_virtual_schema
btech       han       students1   system_schema      vehicle
company     library   system      system_traces

cqlsh> CREATE KEYSPACE P_Employee WITH replication = { 'class' : 'SimpleStrategy', 'replication_factor':1};
cqlsh> DESCRIBE KEYSPACES;

bemployee  cycling  my_keyspace  system      system_traces
bstudents   employee  p_employee  system_auth  system_views
btech       han       students    system_distributed  system_virtual_schema
company     library   students1   system_schema  vehicle

cqlsh> USE P_Employee;

cqlsh:p_employee> CREATE TABLE Employee_Info (
... Emp_Id INT PRIMARY KEY,
... Emp_Name TEXT,
... Designation TEXT,
... Date_of_Joining DATE,
... Salary FLOAT,
... Dept_Name TEXT
... );
cqlsh:p_employee> DESCRIBE TABLE Employee_Info;

CREATE TABLE p_employee.employee_info (
  emp_id int PRIMARY KEY,
  date_of_joining date,
  dept_name text,
  designation text,
  emp_name text,
  salary float
) 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 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:p_employee> BEGIN BATCH
... INSERT INTO Employee_Info (Emp_Id, Emp_Name, Designation, Date_of_Joining, Salary,
... Dept_Name)
... VALUES (120, 'Priyanka GH', 'Developer', '2024-05-06', 100000, 'Engineering');
... INSERT INTO Employee_Info (Emp_Id, Emp_Name, Designation, Date_of_Joining, Salary,
... Dept_Name)
... VALUES (121, 'Shreya', 'Developer', '2024-05-06', 90000, 'Management');
... INSERT INTO Employee_Info (Emp_Id, Emp_Name, Designation, Date_of_Joining, Salary,
... Dept_Name)
... VALUES (122, 'Sadhana', 'Engineer', '2024-05-07', 120000, 'Engineering');
... APPLY BATCH;
```

```

cqlsh:p_employee> SELECT * FROM Employee_Info;
+-----+-----+-----+-----+-----+-----+
| emp_id | date_of_joining | dept_name | designation | emp_name | salary |
+-----+-----+-----+-----+-----+-----+
| 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | 1e+05 |
| 122 | 2024-05-07 | Engineering | Engineer | Sadhana | 1.2e+05 |
| 121 | 2024-05-06 | Management | Developer | Shreya | 90000 |
+-----+-----+-----+-----+-----+-----+
(3 rows)

cqlsh:p_employee> UPDATE Employee_Info
... SET Emp_Name = 'Rachana', Dept_Name = 'HR'
... WHERE Emp_Id = 121;
cqlsh:p_employee> SELECT * FROM Employee_Info;
+-----+-----+-----+-----+-----+-----+
| emp_id | date_of_joining | dept_name | designation | emp_name | salary |
+-----+-----+-----+-----+-----+-----+
| 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | 1e+05 |
| 122 | 2024-05-07 | Engineering | Engineer | Sadhana | 1.2e+05 |
| 121 | 2024-05-06 | HR | Developer | Rachana | 90000 |
+-----+-----+-----+-----+-----+-----+
(3 rows)

cqlsh:p_employee> SELECT * FROM Employee_Info
... ALLOW FILTERING;
+-----+-----+-----+-----+-----+-----+
| emp_id | date_of_joining | dept_name | designation | emp_name | salary |
+-----+-----+-----+-----+-----+-----+
| 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | 1e+05 |
| 122 | 2024-05-07 | Engineering | Engineer | Sadhana | 1.2e+05 |
| 121 | 2024-05-06 | HR | Developer | Rachana | 90000 |
+-----+-----+-----+-----+-----+-----+
(3 rows)

cqlsh:p_employee> ALTER TABLE Employee_Info
... ADD Projects set<text>;
cqlsh:p_employee> DESCRIBE TABLE Employee_Info;

CREATE TABLE p_employee.employee_info (
    emp_id int PRIMARY KEY,
    date_of_joining date,
    dept_name text,
    designation text,
    emp_name text,
    salary float,
    projects set<text>
) 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 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:p_employee> UPDATE Employee_Info
... SET Projects = ['Project A', 'Project B']
... WHERE Emp_Id = 120;
cqlsh:p_employee> UPDATE Employee_Info
... SET Projects = ['Project C']
... WHERE Emp_Id = 121;
cqlsh:p_employee> UPDATE Employee_Info
... SET Projects = ['Project A', 'Project D']
... WHERE Emp_Id = 122;
cqlsh:p_employee> SELECT Emp_Id, Emp_Name, Projects FROM Employee_Info;
+-----+-----+-----+
| emp_id | emp_name | projects |
+-----+-----+-----+
| 120 | Priyanka GH | {'Project A', 'Project B'} |
| 122 | Sadhana | {'Project A', 'Project D'} |
| 121 | Rachana | {'Project C'} |
+-----+-----+-----+
(3 rows)
cqlsh:p_employee>

```

```
cqlsh:p_employee> UPDATE Employee_Info
... USING TTL 15
... SET Salary = 0
... WHERE Emp_Id = 121;
cqlsh:p_employee> SELECT * FROM Employee_Info;

emp_id | date_of_joining | dept_name | designation | emp_name | projects | salary
-----+-----------------+-----------+-------------+-----+-----+-----+-----+
 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | {'Project A', 'Project B'} | 1e+05
 122 | 2024-05-07 | Engineering | Engineer | Sadhana | {'Project A', 'Project D'} | 1.2e+05
 121 | 2024-05-06 | HR | Developer | Rachana | {'Project C'} | 0
(3 rows)
cqlsh:p_employee>
```

```
cqlsh:p_employee> UPDATE Employee_Info
... USING TTL 15
... SET Salary = 0
... WHERE Emp_Id = 121;
cqlsh:p_employee> SELECT * FROM Employee_Info;

emp_id | date_of_joining | dept_name | designation | emp_name | projects | salary
-----+-----------------+-----------+-------------+-----+-----+-----+-----+
 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | {'Project A', 'Project B'} | 1e+05
 122 | 2024-05-07 | Engineering | Engineer | Sadhana | {'Project A', 'Project D'} | 1.2e+05
 121 | 2024-05-06 | HR | Developer | Rachana | {'Project C'} | 0
(3 rows)
cqlsh:p_employee> SELECT * FROM Employee_Info;

emp_id | date_of_joining | dept_name | designation | emp_name | projects | salary
-----+-----------------+-----------+-------------+-----+-----+-----+-----+
 120 | 2024-05-06 | Engineering | Developer | Priyanka GH | {'Project A', 'Project B'} | 1e+05
 122 | 2024-05-07 | Engineering | Engineer | Sadhana | {'Project A', 'Project D'} | 1.2e+05
 121 | 2024-05-06 | HR | Developer | Rachana | {'Project C'} | null
(3 rows)
cqlsh:p_employee>
```

STUDENTS KEYSpace

```

cqlsh> CREATE KEYSPACE Students WITH REPLICATION =
... . {'class':'SimpleStrategy','replication_factor':1};
AlreadyExists: Keyspace 'students' already exists
cqlsh> CREATE KEYSPACE P_Students WITH REPLICATION = {'class':'SimpleStrategy','replication_factor':1};
cqlsh> DESCRIBE KEYSPACES;
bemployee employee p_students system_distributed vehicle
bstudents han students system_schema
btech library students1 system_traces
company my_keyspace system system_views
cycling p_employee system_auth system_virtual_schema

cqlsh> SELECT * FROM system.schema_keyspaces;
InvalidRequest: Error from server: code=2200 [Invalid query] message="table schema_keyspaces does not exist"
cqlsh> USE Students;
cqlsh: Students> EXIT
bnseccegbnsecce-HP-Elite-Tower-600-G9-Desktop-PC:-$ CQLSH
CQLSH: command not found
bnseccegbnsecce-HP-Elite-Tower-600-G9-Desktop-PC:-$ cqlsh
Connected to Test Cluster at 127.0.0.1:9642
[cqlsh 6.0.0 | Cassandra 4.0.5 | CQL spec 3.4.5 | Native protocol v5]
Use HELP for help.
cqlsh> USE p_Students;
cqlsh:p_Students> CREATE TABLE Students_Info (Roll_No int PRIMARY KEY, StudName text,
... DateOfJoining timestamp, last_exam_Percent double);
cqlsh:p_Students> DESCRIBE TABLES;
students_info

cqlsh:p_Students> DESCRIBE TABLE students_info;

CREATE TABLE p_Students.students_info (
    roll_no int PRIMARY KEY,
    dateofjoining timestamp,
    last_exam_percent double,
    studname text
) 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 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';
    AND write_repair_chance = 0.01

cqlsh:p_Students> BEGIN BATCH
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Asha','2012-03-12',79.9)
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Krian','2012-03-12',89.9)
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Tarun','2012-03-12',78.9)
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Samrth','2012-03-12',90.9)
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Smitha','2012-03-12',67.9)
... INSERT INTO Students_Info(Roll_No, StudName, DateOfJoining, last_exam_Percent)
... VALUES (1,'Rohan','2012-03-12',56.9)
... APPLY BATCH;
cqlsh:p_Students> SELECT * FROM Students_Info;

roll_no | dateofjoining | last_exam_percent | studname
-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 |      90.9 | Tarun

(1 rows)
cqlsh:p_Students> SELECT * FROM Students_Info WHERE Roll_No IN (1,2,3);

roll_no | dateofjoining | last_exam_percent | studname
-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 |      90.9 | Tarun

(1 rows)
cqlsh:p_Students> select * from students_info where Studname= 'Asha';
Invalid syntax at char 45
  select * from students_info where Studname= 'Asha';

cqlsh:p_Students> select * from students_info where Studname= 'Asha';
^
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot execute this query as it might involve data filtering and thus may have u
query despite the performance unpredictability, use ALLOW FILTERING"
cqlsh:p_Students> CREATE INDEX ON Students_Info ( StudName );
cqlsh:p_Students> select * from students_info where Studname= 'Asha';

roll_no | dateofjoining | last_exam_percent | studname
-----+-----+-----+-----+
  1 |          null |          null |      Tarun

(0 rows)
cqlsh:p_Students> select Roll_No, StudName from students_info LIMIT 2;

roll_no | studname
-----+-----+
  1 |      Tarun

(1 rows)

```

```

cqlsh:p_students> UPDATE students_info SET StudName='David Sheen' WHERE Roll_No=2;
cqlsh:p_students> UPDATE students_info SET roll_no=6 WHERE roll_no=3;
InvalidRequest: Error from server: code=200 [Invalid query] message="PRIMARY KEY part roll_no found in SET part"
cqlsh:p_students> select * from students_info;

roll_no | dateofjoining | last_exam_percent | studname
-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 |      90.9 | Tarun
  2 |           null |           null | David Sheen

(2 rows)
cqlsh:p_students> DELETE LastExamPercent FROM students_info WHERE Roll_No=2;
InvalidRequest: Error from server: code=200 [Invalid query] message="Undefined column name lastexampercent in table p_students.students_info"
cqlsh:p_students> DELETE Last_Exam_Percent FROM students_info WHERE Roll_No=2;
cqlsh:p_students> DELETE FROM student_info WHERE Roll_No=2;
InvalidRequest: Error from server: code=200 [Invalid query] message="table student_info does not exist"
cqlsh:p_students> select * from students_info;

roll_no | dateofjoining | last_exam_percent | studname
-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 |      90.9 | Tarun
  2 |           null |           null | David Sheen

(2 rows)
cqlsh:p_students> DELETE FROM students_info WHERE Roll_No=2;
cqlsh:p_students> ALTER TABLE students_info ADD hobbies set<text>;
cqlsh:p_students> select * from students_info;

roll_no | dateofjoining | hobbies | last_exam_percent | studname
-----+-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 |    null |      90.9 | Tarun

(1 rows)
cqlsh:p_students> ALTER TABLE students_info ADD language list<text>;
cqlsh:p_students> UPDATE students_info
... SET hobbies=hobbies+{'Chess,Table Tennis'}
... WHERE RollNo=1;
InvalidRequest: Error from server: code=200 [Invalid query] message="Undefined column name rollno in table p_students.students_info"
cqlsh:p_students> UPDATE students_info
... SET hobbies=hobbies+{'Chess,Table Tennis'}
... WHERE Roll_No=1;
cqlsh:p_students> select * from students_info;

roll_no | dateofjoining | hobbies | language | last_exam_percent | studname
-----+-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 | {'Chess,Table Tennis'} |    null |      90.9 | Tarun

(1 rows)

cqlsh:p_students> SELECT * from students_info WHERE Roll_No=1;

roll_no | dateofjoining | hobbies | language | last_exam_percent | studname
-----+-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 | {'Chess,Table Tennis'} |    null |      90.9 | Tarun

(1 rows)
cqlsh:p_students> UPDATE students_info
... SET language=language+['Hindi,English']
... WHERE Roll_No=1;
cqlsh:p_students> select * from students_info;

roll_no | dateofjoining | hobbies | language | last_exam_percent | studname
-----+-----+-----+-----+-----+
  1 | 2012-03-11 18:30:00.000000+0000 | {'Chess,Table Tennis'} | ['Hindi,English'] |      90.9 | Tarun

(1 rows)
cqlsh:p_students> CREATE TABLE library_book(counter_value counter, book_name varchar, stud_name varchar,
... PRIMARY KEY(book_name,stud_name));
cqlsh:p_students> UPDATE library_book SET countrr value=countrr value+1 WHERE book_name='Big data Analytics'
... AND studname='jeet';
SyntaxException: line 1:32 no viable alternative at input 'value' (UPDATE library_book SET countrr [value]...)
cqlsh:p_students> UPDATE library_book SET counter value=counter_value+1 WHERE book_name='Big data Analytics' AND studname='jeet';
SyntaxException: line 1:32 no viable alternative at input 'value' (UPDATE library_book SET counter [value]...)
cqlsh:p_students> select * from library_book;

book_name | stud_name | counter_value
-----+-----+-----+
(0 rows)
cqlsh:p_students> UPDATE library_book SET counter value=counter_value+1 WHERE book_name='Big data Analytics' AND stud_name='jeet';
SyntaxException: line 1:32 no viable alternative at input 'value' (UPDATE library_book SET counter [value]...)
cqlsh:p_students> UPDATE library_book SET counter_value=counter_value+1 WHERE book_name='Big data Analytics' AND stud_name='jeet';
cqlsh:p_students> CREATE TABLE userlogin(userid int PRIMARY KEY, password text);
cqlsh:p_students> INSERT INTO userlogin(userid, password) VALUES (1,'infy') USING TTL 30;
Invalid syntax at char 51
  INSERT INTO userlogin(userid, password) VALUES (1,'infy') USING TTL 30;
^
cqlsh:p_students> INSERT INTO userlogin(userid, password) VALUES (1,'infy') USING TTL 30;
cqlsh:p_students> SELECT TTL(password) FROM userlogin WHERE userid=1;

ttl(password)
-----
  19

(1 rows)
cqlsh:p_students>

```