

ICP-7

Name : Anusha Jasti

Class ID : 10

Cassandra

```
Administrator: Command Prompt
cassandra.db.marshall.TimeUDTType),partitionColumns=[]) [coordinator exception_message exception_stacktrace finished_at parent_id range_begin range_end started_at status participant
ts]],partitionKeyColumns=[keyspace_name, columnfamily_name],clusteringColumns=[id],keyValidator=org.apache.cassandra.db.marshall.CompositeType(org.apache.cassandra.db.marshall.UTF8Type
org.apache.cassandra.db.marshall.UTF8Type),columnMetadata=[status, id, coordinator, finished_at, participants, exception_stacktrace, parent_id, range_end, range_begin, exception_m
essage, keyspace_name, started_at, columnfamily_name],droppedColumns={},triggers=[],indexes=[], org.apache.cassandra.config.CFMetaData@265a75a[cfId=deabd734-b9d0-3b9c-92e5-fd92eb5ab
f14,ksName=system_distributed,cfName=parent_repair_history,flags=[COMPOUND],params=TableParams{comment=Repair history, read_repair_chance=0.0, dclocal_read_repair_chance=0.0, bloom
filter_fp_chance=0.01, crc_check_chance=1.0, gc_grace_seconds=864000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_interval=128, max_index_interval=2048, s
peculative_retry=99PERCENTILE, caching={keys: 'ALL', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db.compaction.SizeTieredCompactionStra
tegy, options={min_threshold=4, max_threshold=32}}, compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(),partitionCo
lums=[]) [exception_message exception_stacktrace finished_at keyspace_name started_at columnfamily_name options requested_ranges successful_ranges]],partitionKeyColumns=[parent_
id],clusteringColumns=[],keyValidator=org.apache.cassandra.db.marshall.TimeUDTType,columnMetadata=[requested_ranges, exception_message, keyspace_name, successful_ranges, started_at,
finished_at, options, exception_stacktrace, parent_id, columnfamily_name],droppedColumns={},triggers=[],indexes=[], org.apache.cassandra.config.CFMetaData@54a8efb9[cfId=5582eb59f-
8e4e-35e1-b913-3acada51eb04,ksName=system_distributed,cfName=view_build_status,flags=[COMPOUND],params=TableParams{comment=Materialized View build status, read_repair_chance=0.0, dc
local_read_repair_chance=0.0, bloom_filter_fp_chance=0.01, crc_check_chance=1.0, gc_grace_seconds=864000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_inte
rval=128, max_index_interval=2048, speculative_retry=99PERCENTILE, caching={keys: 'ALL', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db
.compaction.SizeTieredCompactionStrategy, options={min_threshold=4, max_threshold=32}}, compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(org.apache.cassandra.db.marshall.UUIDType),partitionColumns=[]) [status]],partitionKeyColumns=[keyspace_name, view_name],clusteringColumns=[host_id],keyValid
ator=org.apache.cassandra.db.marshall.CompositeType(org.apache.cassandra.db.marshall.UTF8Type,org.apache.cassandra.db.marshall.UTF8Type),columnMetadata=[view_name, status, keyspace_nam
e, host_id],droppedColumns={},triggers=[],indexes=[], views=[], functions=[], types=[]]
INFO [MigrationStage:1] 2019-06-30 17:09:32,236 ViewManager.java:137 - Not submitting build tasks for views in keyspace system_distributed as storage service is not initialized
INFO [MigrationStage:1] 2019-06-30 17:09:32,241 ColumnFamilyStore.java:430 - Initializing system_distributed.parent_repair_history
INFO [MigrationStage:1] 2019-06-30 17:09:32,248 ColumnFamilyStore.java:430 - Initializing system_distributed.repair_history
INFO [MigrationStage:1] 2019-06-30 17:09:32,256 ColumnFamilyStore.java:430 - Initializing system_distributed.view_build_status
INFO [main] 2019-06-30 17:09:32,269 StorageService.java:1483 - JOINING: Finish joining ring
INFO [main] 2019-06-30 17:09:32,417 MigrationManager.java:337 - Create new keyspace: KeyspaceMetadata{name=system_auth, params=KeyspaceParams{durable_writes=true, replication=Repli
cationParams{class=org.apache.cassandra.locator.SimpleStrategy, replication_factor=1}}, tables=[org.apache.cassandra.config.CFMetaData@43c434e6[cfId=5bc52802-de25-35ed-aeb-188eeceb
b090,ksName=system_auth,cfName=roles,flags=[COMPOUND],params=TableParams{comment=role definitions, read_repair_chance=0.0, dclocal_read_repair_chance=0.0, bloom_filter_fp_chance=0.0
1, crc_check_chance=1.0, gc_grace_seconds=7776000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_interval=128, max_index_interval=2048, speculative_retry=99
PERCENTILE, caching={keys: 'ALL', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy, options={min_
threshold=4, max_threshold=32}}, compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(),partitionColumns=[]) [can_lo
gin is_superuser, salted_hash, member_of_role, can_login, is_superuser],droppedColumns={},triggers=[],indexes=[], org.apache.cassandra.config.CFMetaData@42b9c296[cfId=0ecd8a87-f8fb-3e08-88d1-74fb36f5c0d,ksName=system_a
uth,cfName=role_members,flags=[COMPOUND],params=TableParams{comment=role memberships lookup table, read_repair_chance=0.0, dclocal_read_repair_chance=0.0, bloom_filter_fp_chance=0.0
1, crc_check_chance=1.0, gc_grace_seconds=7776000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_interval=128, max_index_interval=2048, speculative_retry=99
PERCENTILE, caching={keys: 'ALL', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy, options={min_
threshold=4, max_threshold=32}}, compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(org.apache.cassandra.db.marshall.
UTF8Type),partitionColumns=[]) [role]],partitionKeyColumns=[role],clusteringColumns=[member],keyValidator=org.apache.cassandra.db.marshall.UTF8Type,columnMetadata=[role, member],droppe
dColumns={},triggers=[],indexes=[]], org.apache.cassandra.config.CFMetaData@312390b0[cfId=3afbe9f-2194-31a7-ddd7-f5ab9d0b0e9c,ksName=system_auth,cfName=role_permissions,flags=[COMP
OUND],params=TableParams{comment=permissions granted to db roles, read_repair_chance=0.0, dclocal_read_repair_chance=0.0, bloom_filter_fp_chance=0.01, crc_check_chance=1.0, gc_gra
ce_seconds=7776000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_interval=128, max_index_interval=2048, speculative_retry=99PERCENTILE, caching={keys: 'AL
L', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy, options={min_threshold=4, max_threshold=32}},
compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(org.apache.cassandra.db.marshall.UTF8Type),partitionColumns=[]) [
[permissions]],partitionKeyColumns=[role],clusteringColumns=[resource],keyValidator=org.apache.cassandra.db.marshall.UTF8Type,columnMetadata=[role, resource, permissions],droppedColu
ms={},triggers=[],indexes=[]], org.apache.cassandra.config.CFMetaData@c8f8a2d6[cfId=5f2fbdad-91f1-3e46-bd25-d5da3a5c35ec,ksName=system_auth,cfName=resource_role_permissions_index,fl
ags=[COMPOUND],params=TableParams{comment=index of db roles with permissions granted on a resource, read_repair_chance=0.0, dclocal_read_repair_chance=0.0, bloom_filter_fp_chance=0
.01, crc_check_chance=1.0, gc_grace_seconds=7776000, default_time_to_live=0, memtable_flush_period_in_ms=3600000, min_index_interval=128, max_index_interval=2048, speculative_retry=9
9PERCENTILE, caching={keys: 'ALL', 'rows_per_partition': 'NONE'}}, compaction=CompactionParams{class=org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy, options={min_
threshold=4, max_threshold=32}}, compression=org.apache.cassandra.schema.CompressionParams@179f9be9, extensions={}, cdc=false},comparator=comparator(org.apache.cassandra.db.marshall
.UTF8Type),partitionColumns=[]) [role]],partitionKeyColumns=[resource],clusteringColumns=[role],keyValidator=org.apache.cassandra.db.marshall.UTF8Type,columnMetadata=[resource, role],d
roppedColumns={},triggers=[],indexes=[]], views=[], functions=[], types=[]]
INFO [MigrationStage:1] 2019-06-30 17:09:32,645 ViewManager.java:137 - Not submitting build tasks for views in keyspace system_auth as storage service is not initialized
INFO [MigrationStage:1] 2019-06-30 17:09:32,648 ColumnFamilyStore.java:430 - Initializing system_auth.resource_role_permissions_index
INFO [MigrationStage:1] 2019-06-30 17:09:32,656 ColumnFamilyStore.java:430 - Initializing system_auth.role_members
INFO [MigrationStage:1] 2019-06-30 17:09:32,662 ColumnFamilyStore.java:430 - Initializing system_auth.role_permissions
INFO [MigrationStage:1] 2019-06-30 17:09:32,668 ColumnFamilyStore.java:430 - Initializing system_auth.roles
```

Create a Keyspace test

```
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.
WARNING: pyreadline dependency missing. Install to enable tab completion.
cqlsh> create keyspace test with replication={'class': 'SimpleStrategy', 'replication_factor': 1};
SyntaxError: line 1:7: no viable alternative at input 'keyspae' (([create] keyspae...))
cqlsh> create keyspace test with replication={'class': 'SimpleStrategy', 'replication_factor': 1};
cqlsh>
```

Use the keyspace test to perform queries

Creating table emp

```
cqlsh:test> create table emp(employee_id int PRIMARY KEY,e_name text,last_name text,e_no int,department text,dept_no int,job_title text,salary int,annual_salary int,h
ire_date date,years_with_company int);
cqlsh:test> select * from emp;

 employee_id | annual_salary | department | dept_no | e_name | e_no | hire_date | job_title | last_name | salary | years_with_company
-----
(0 rows)
cqlsh:test>
```

Inserting values into the table emp

```

cqlsh:test> INSERT INTO emp(employee_id,e_name,last_name,e_no,department,dept_no,job_title,salary,annual_salary,hire_date,years_with_company) VALUES (1,'Anusha','Jasti',303,'Engineering',30,'clerk',10000,120000,'1999-01-31',20);
cqlsh:test> select * from emp;

employee_id | annual_salary | department | dept_no | e_name | e_no | hire_date | job_title | last_name | salary | years_with_company
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
1 | 120000 | Engineering | 30 | Anusha | 303 | 1999-01-31 | clerk | Jasti | 10000 | 20
(1 rows)
cqlsh:test> INSERT INTO emp(employee_id,e_name,last_name,e_no,department,dept_no,job_title,salary,annual_salary,hire_date,years_with_company) VALUES (2,'Apoorva','Geeta',506,'Engineering',28,'clerk',1000,12000,'2018-12-17',1);
cqlsh:test> INSERT INTO emp(employee_id,e_name,last_name,e_no,department,dept_no,job_title,salary,annual_salary,hire_date,years_with_company) VALUES (3,'Bindu','Hima',405,'Marketing',38,'assistant',500,60000,'2018-12-16',1);
cqlsh:test> INSERT INTO emp(employee_id,e_name,last_name,e_no,department,dept_no,job_title,salary,annual_salary,hire_date,years_with_company) VALUES (4,'James','Howard',987,'Sales',34,'clerk',200,2400,'2017-06-24',2);
cqlsh:test> INSERT INTO emp(employee_id,e_name,last_name,e_no,department,dept_no,job_title,salary,annual_salary,hire_date,years_with_company) VALUES (5,'Jacob','Gonzalez',256,'Engineering',23,'clerk',300,3600,'2015-11-23',4);

```

The output of the table

```

cqlsh:test> select * from emp;

employee_id | annual_salary | department | dept_no | e_name | e_no | hire_date | job_title | last_name | salary | years_with_company
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
5 | 3600 | Engineering | 23 | Jacob | 256 | 2015-11-23 | clerk | Gonzalez | 300 | 4
1 | 120000 | Engineering | 30 | Anusha | 303 | 1999-01-31 | clerk | Jasti | 10000 | 20
2 | 12000 | Engineering | 28 | Apoorva | 506 | 2018-12-17 | clerk | Geeta | 1000 | 1
4 | 2400 | Sales | 34 | James | 987 | 2017-06-24 | clerk | Howard | 200 | 2
3 | 60000 | Marketing | 38 | Bindu | 405 | 2018-12-16 | assistant | Hima | 500 | 1
(5 rows)
cqlsh:test>

```

The following are the queries performed on the table emp.

1. Query to display the e_no,e_name,job_title,hire_date.

```

cqlsh:test> select e_no,e_name,job_title,hire_date from emp;

e_no | e_name | job_title | hire_date
-----+-----+-----+-----
256 | Jacob | clerk | 2015-11-23
303 | Anusha | clerk | 1999-01-31
506 | Apoorva | clerk | 2018-12-17
987 | James | clerk | 2017-06-24
405 | Bindu | assistant | 2018-12-16
(5 rows)
cqlsh:test>

```

2. Query to display name and salary of employee who are clerks.

```

cqlsh:test> select e_name,salary from emp WHERE job_title='clerk' ALLOW FILTERING;

e_name | salary
-----+-----
Jacob | 300
Anusha | 10000
Apoorva | 1000
James | 200
(4 rows)
cqlsh:test>

```

3. Query to display name,job,salary of employees who joined on 'Jan 31 1999'.

```
cqlsh:test> select e_name,job_title,salary FROM emp WHERE hiredate='1999-01-31' ALLOW FILTERING;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Undefined column name hiredate"
cqlsh:test> select e_name,job_title,salary FROM emp WHERE hire_date='1999-01-31' ALLOW FILTERING;

e_name | job_title | salary
-----+-----+-----
Anusha | clerk    | 10000

(1 rows)
cqlsh:test>
```

4. Query to display name and annual salary of all employees.

```
(1 rows)
cqlsh:test> select e_name,annual_salary FROM emp;

e_name | annual_salary
-----+-----
Jacob  | 3600
Anusha | 120000
Apoorva | 12000
James  | 2400
Bindu  | 60000

(5 rows)
cqlsh:test>
```

5. Query to display name,dept no for the departments having dept no greater than 20.

```
(1 rows)
cqlsh:test> select department,dept_no FROM emp WHERE dept_no>=20 ALLOW FILTERING;

department | dept_no
-----+-----
Engineering | 23
Engineering | 30
Engineering | 28
Sales       | 34
Marketing   | 38

(5 rows)
cqlsh:test>
```

Bonus :

I have chosen the dataset books for the bonus part.

- Creating the table books and importing data from csv file.

```
cqlsh:test> CREATE TABLE books (book_id int,category text,title text,number_pages int,publisher text,pub_date timestamp,author text,PRIMARY KEY(publisher, pub_date,book_id));
cqlsh:test> COPY books (book_id, category, title, author, publisher, pub_date, number_pages) FROM 'C:\Users\kite\Downloads\Data\books\book_entries.csv' WITH HEADER=True AND DELIMITER='|';
Using 7 child processes

Starting copy of test.books with columns [book_id, category, title, author, publisher, pub_date, number_pages].
PPProcess ImportProcess-44: 5 rows/s; Avg. rate: 5 rows/s
PTProcess ImportProcess-41:
Process ImportProcess-46:
Process ImportProcess-43:
Process ImportProcess-47:
Process ImportProcess-45:
```

```
cqlsh:test> select * from books;
```

| publisher | pub_date | book_id | author | category | number_pages | title |
|-----------|---------------------------------|---------|----------------|-----------------|--------------|----------------------|
| TOR | 1985-02-14 18:00:00.000000+0000 | 3 | Isaac Asimov | science fiction | 550 | Foundation |
| TOR | 1999-04-21 18:00:00.000000+0000 | 4 | David Weaver | fantasy | 600 | On Your Honor |
| Penguin | 2013-08-11 18:00:00.000000+0000 | 1 | Dorothy Sawyer | mystery | 500 | Murder By Alibi |
| Penguin | 2014-09-29 18:00:00.000000+0000 | 2 | Tom Jones | mystery | 500 | Murder at the School |

```
(4 rows)
```

- Performing few queries on the table books.

- Query to display author,book id whose category is fantasy.

```
cqlsh:test> select author,book_id FROM books WHERE category='fantasy' ALLOW FILTERING;
```

| author | book_id |
|--------------|---------|
| David Weaver | 4 |

```
(1 rows)
cqlsh:test>
```

- Query to display the maximum number of pages in all the books.

```
cqlsh:test> select max(number_pages) FROM books;
```

| system.max(number_pages) |
|--------------------------|
| 600 |

```
(1 rows)

Warnings :
Aggregation query used without partition key
```

- Query to display author,book id,publisher whose number of pages=500.

```
cqlsh:test> select author,book_id,publisher FROM books WHERE number_pages = 500 ALLOW FILTERING;
```

| author | book_id | publisher |
|----------------|---------|-----------|
| Dorothy Sawyer | 1 | Penguin |
| Tom Jones | 2 | Penguin |

```
(2 rows)
cqlsh:test>
```

- Query to display min number of pages in all the books.


```
cqlsh:test> select min(number_pages) FROM books;

system.min(number_pages)
-----
500

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:test>
```

5. Query to display author,category whose publisher is Penguin.

```
cqlsh:test> select author,category FROM books WHERE publisher='Penguin' ALLOW FILTERING;

author          | category
-----+-----
Dorothy Sawyer | mystery
Tom Jones      | mystery

(2 rows)

cqlsh:test>
```