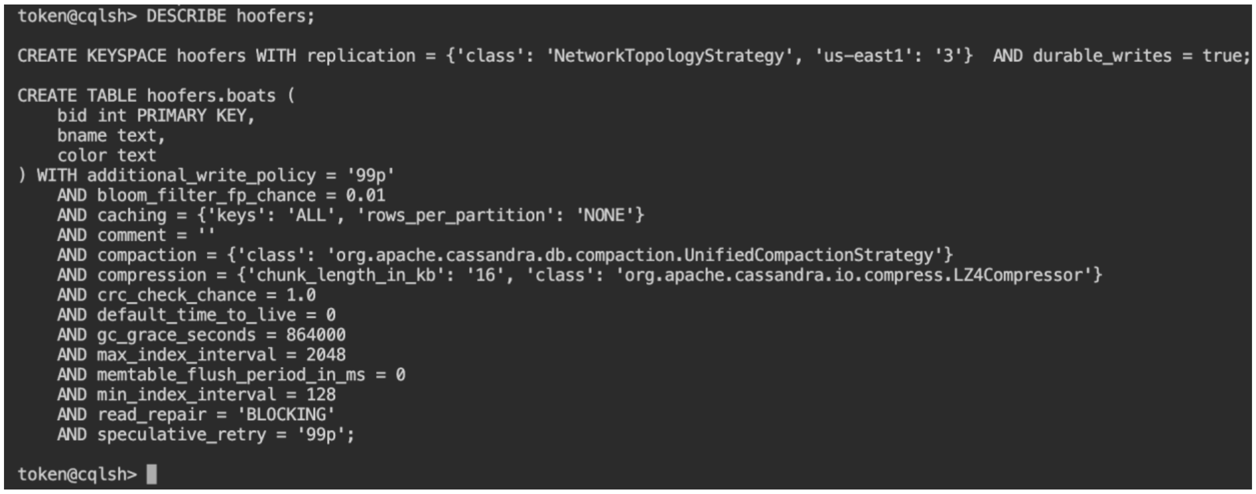
Student Name: Ameya Gaitonde UCInetID: ameyag

1.

a) CQL Query:

DESCRIBE hoofers;

b) Result:



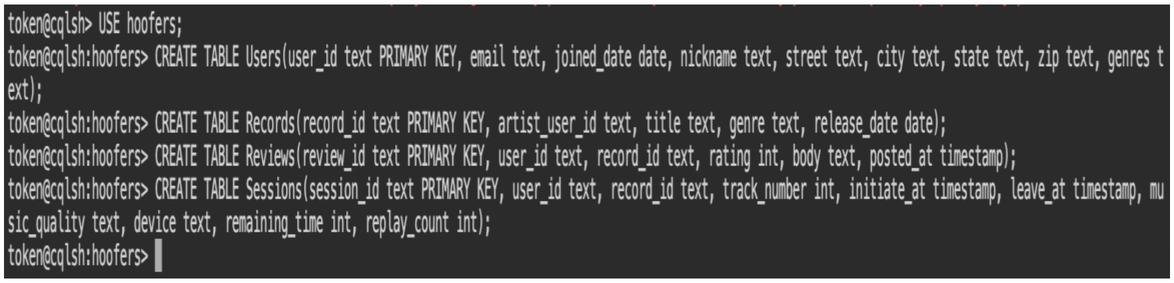
```
token@cqlsh> DESCRIBE hoofers;

CREATE KEYSPACE hoofers WITH replication = {'class': 'NetworkTopologyStrategy', 'us-east1': '3'}  AND durable_writes = true;

CREATE TABLE hoofers.boats (
    bid int PRIMARY KEY,
    bname text,
    color text
) WITH additional_write_policy = '99p'
    AND bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.UnifiedCompactionStrategy'}
    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 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';

token@cqlsh> 
```

c) Answers:

The Hoofers keyspace maintains 3 copies of the data. It resides in the 'us-east1' cloud region. The Read and Write quorum sizes should each be 2 in this case. This means that at least 2 copies (a majority of 3) must accept the read or write operation to successfully occur.

2.

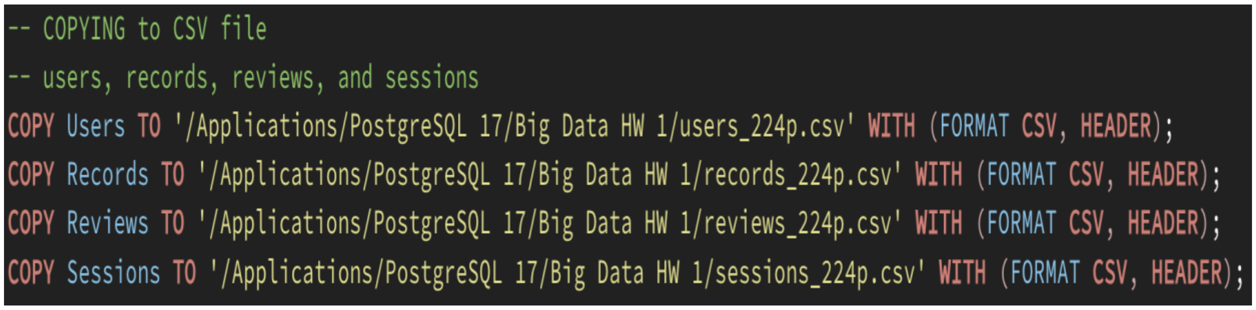
a) CQL CREATE Statements:



```
token@cqlsh> USE hoofers;
token@cqlsh:hoofers> CREATE TABLE Users(user_id text PRIMARY KEY, email text, joined_date date, nickname text, street text, city text, state text, zip text, genres t
ext);
token@cqlsh:hoofers> CREATE TABLE Records(record_id text PRIMARY KEY, artist_user_id text, title text, genre text, release_date date);
token@cqlsh:hoofers> CREATE TABLE Reviews(review_id text PRIMARY KEY, user_id text, record_id text, rating int, body text, posted_at timestamp);
token@cqlsh:hoofers> CREATE TABLE Sessions(session_id text PRIMARY KEY, user_id text, record_id text, track_number int, initiate_at timestamp, leave_at timestamp, mu
sic_quality text, device text, remaining_time int, replay_count int);
token@cqlsh:hoofers>
```

3.

a) PostgreSQL COPY commands:



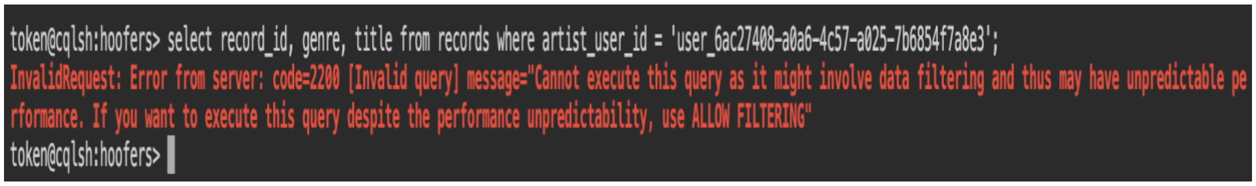
```
-- COPYING to CSV file
-- users, records, reviews, and sessions
COPY Users TO '/Applications/PostgreSQL 17/Big Data HW 1/users_224p.csv' WITH (FORMAT CSV, HEADER);
COPY Records TO '/Applications/PostgreSQL 17/Big Data HW 1/records_224p.csv' WITH (FORMAT CSV, HEADER);
COPY Reviews TO '/Applications/PostgreSQL 17/Big Data HW 1/reviews_224p.csv' WITH (FORMAT CSV, HEADER);
COPY Sessions TO '/Applications/PostgreSQL 17/Big Data HW 1/sessions_224p.csv' WITH (FORMAT CSV, HEADER);
```

4.

a) First CQL Query:

```
select record_id, genre, title
from records
where artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3';
```

b) Result:

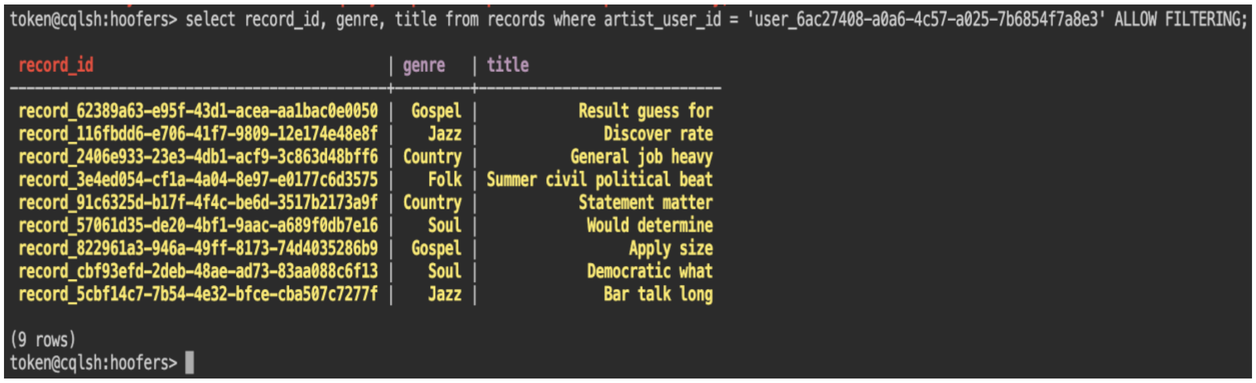


```
token@cqlsh:hoofers> select record_id, genre, title from records where artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3';
InvalidRequest: Error from server: code=2200 [Invalid query] message="Cannot execute this query as it might involve data filtering and thus may have unpredictable pe
rformance. If you want to execute this query despite the performance unpredictability, use ALLOW FILTERING"
token@cqlsh:hoofers>
```

c) Modified CQL Query:

```
SELECT record_id, genre, title
FROM records
WHERE artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3'
ALLOW FILTERING;
```

b) Result:



```
token@cqlsh:hoofers> select record_id, genre, title from records where artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3' ALLOW FILTERING;

 record_id                                      | genre   | title
------------------------------------------------+---------+----------------------------
 record_62389a63-e95f-43d1-acea-aa1bac0e0050 |  Gospel |           Result guess for
 record_116fbdd6-e706-41f7-9809-12e174e48e8f |    Jazz |              Discover rate
 record_2406e933-23e3-4db1-acf9-3c863d48bff6 | Country |          General job heavy
 record_3e4ed054-cf1a-4a04-8e97-e0177c6d3575 |    Folk | Summer civil political beat
 record_91c6325d-b17f-4f4c-be6d-3517b2173a9f | Country |           Statement matter
 record_57061d35-de20-4bf1-9aac-a689f0db7e16 |    Soul |            Would determine
 record_822961a3-946a-49ff-8173-74d4035286b9 |  Gospel |                 Apply size
 record_cbf93efd-2deb-48ae-ad73-83aa088c6f13 |    Soul |             Democratic what
 record_5cbf14c7-7b54-4e32-bfce-cba507c7277f |    Jazz |              Bar talk long

(9 rows)
token@cqlsh:hoofers>
```

5.

a) CQL Create Statement:

CREATE TABLE records_q5(artist_user_id text, record_id text, genre text, release_date date, title text,
PRIMARY KEY (artist_user_id, record_id));

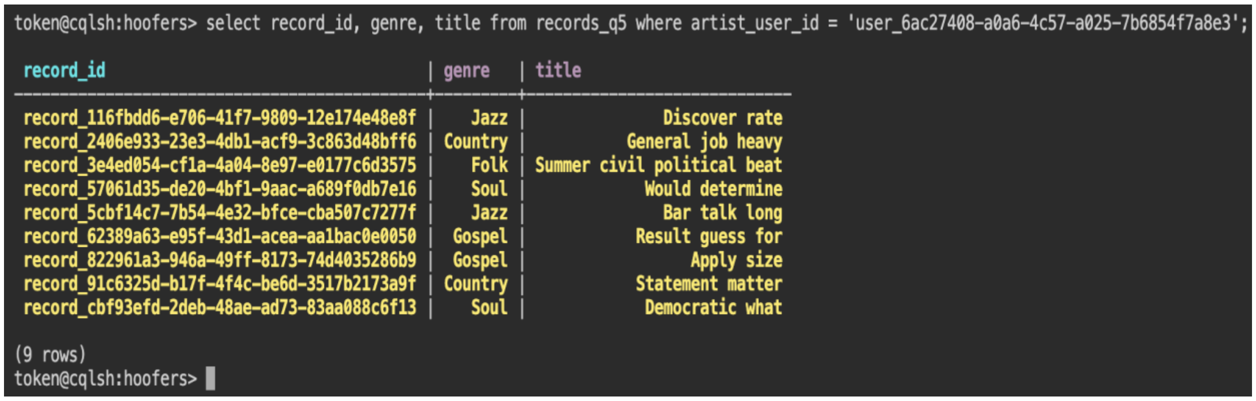
b) CQL Query:

SELECT record_id, genre, title

FROM records_q5

WHERE artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3';

c) Result:



```
token@cqlsh:hoofers> select record_id, genre, title from records_q5 where artist_user_id = 'user_6ac27408-a0a6-4c57-a025-7b6854f7a8e3';

 record_id                                   | genre   | title
---------------------------------------------+---------+-----------------------------
 record_116fbdd6-e706-41f7-9809-12e174e48e8f |    Jazz |               Discover rate
 record_2406e933-23e3-4db1-acf9-3c863d48bff6 | Country |           General job heavy
 record_3e4ed054-cf1a-4a04-8e97-e0177c6d3575 |    Folk | Summer civil political beat
 record_57061d35-de20-4bf1-9aac-a689f0db7e16 |    Soul |             Would determine
 record_5cbf14c7-7b54-4e32-bfce-cba507c7277f |    Jazz |               Bar talk long
 record_62389a63-e95f-43d1-acea-aa1bac0e0050 |  Gospel |           Result guess for
 record_822961a3-946a-49ff-8173-74d4035286b9 |  Gospel |                  Apply size
 record_91c6325d-b17f-4f4c-be6d-3517b2173a9f | Country |            Statement matter
 record_cbf93efd-2deb-48ae-ad73-83aa088c6f13 |    Soul |             Democratic what

(9 rows)
token@cqlsh:hoofers>
```

d) Explanation:

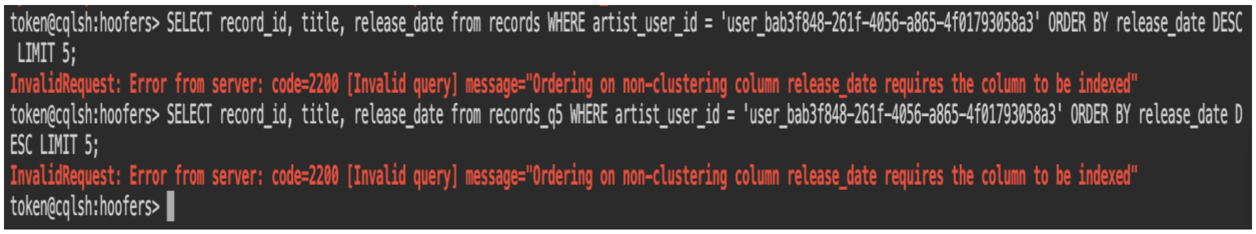
Using the partition key created partitions on the artist_user_id ran without any error from Cassandra. This is because Cassandra no longer has to scan all the partitions to find the relevant data, so we have greater efficiency in retrieving the records. Therefore, there is no error or warning from Cassandra. We include record_id in the primary key so that we can uniquely identify each record given the artist_user_id and record_id.

6.

a) CQL Query:

```
SELECT record_id, title, release_date
FROM records
WHERE artist_user_id = 'user_bab3f848-261f-4056-a865-4f01793058a3'
ORDER BY release_date DESC
LIMIT 5;
```

(I tried this query on both the records and records_q5 tables. Error in both cases).

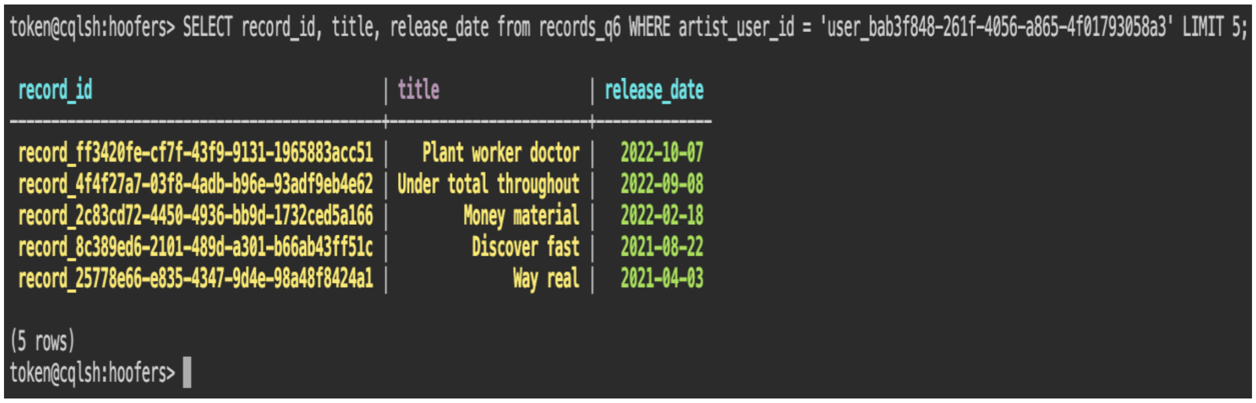


```
token@cqlsh:hoofers> SELECT record_id, title, release_date from records WHERE artist_user_id = 'user_bab3f848-261f-4056-a865-4f01793058a3' ORDER BY release_date DESC
 LIMIT 5;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Ordering on non-clustering column release_date requires the column to be indexed"
token@cqlsh:hoofers> SELECT record_id, title, release_date from records_q5 WHERE artist_user_id = 'user_bab3f848-261f-4056-a865-4f01793058a3' ORDER BY release_date D
ESC LIMIT 5;
InvalidRequest: Error from server: code=2200 [Invalid query] message="Ordering on non-clustering column release_date requires the column to be indexed"
token@cqlsh:hoofers>
```

b) CQL CREATE Statement:

```
CREATE TABLE records_q6 (
        artist_user_id text,
        record_id text,
        release_date date,
        genre text,
        title text,
        PRIMARY KEY (artist_user_id, release_date, record_id)
) WITH CLUSTERING ORDER BY (release_date DESC);
```

c) Results:



```
token@cqlsh:hoofers> SELECT record_id, title, release_date from records_q6 WHERE artist_user_id = 'user_bab3f848-261f-4056-a865-4f01793058a3' LIMIT 5;

 record_id                                    | title                | release_date
----------------------------------------------+----------------------+--------------
 record_ff3420fe-cf7f-43f9-9131-1965883acc51 |   Plant worker doctor |   2022-10-07
 record_4f4f27a7-03f8-4adb-b96e-93adf9eb4e62 | Under total throughout |   2022-09-08
 record_2c83cd72-4450-4936-bb9d-1732ced5a166 |        Money material |   2022-02-18
 record_8c389ed6-2101-489d-a301-b66ab43ff51c |         Discover fast |   2021-08-22
 record_25778e66-e835-4347-9d4e-98a48f8424a1 |              Way real |   2021-04-03

(5 rows)
token@cqlsh:hoofers>
```

d) Explanation:

The new clustering key of release_date allows Cassandra to order on that column without the column being indexed. This is why the error from before is no longer raised after we include release_date as the clustering key within our CREATE TABLE statement.

7.

a) CQL Create Statement:

```
CREATE TABLE table_7a (
        user_id text,
        review_id text,
        record_id text,
        rating int,
        PRIMARY KEY (user_id, rating, review_id) )
        WITH CLUSTERING ORDER BY (rating DESC);
```

b) CQL Create Statement:

```
CREATE TABLE table_7b (
        genre text,
        record_id text,
        PRIMARY KEY (genre, record_id));
```

c) CQL Create Statement:

```
CREATE TABLE table_7c (
        artist_user_id text,
         posted_at timestamp,
        review_id text,
        record_id text,
        title text,
        rating int,
        PRIMARY KEY (artist_user_id, posted_at, review_id) )
        WITH CLUSTERING ORDER BY (posted_at DESC);
```

d) CQL Create Statement:

```
CREATE TABLE table_7d (
        user_id text,
        initiate_at timestamp,
        session_id text,
        replay_count int,
        PRIMARY KEY (user_id, initiate_at, session_id)
);
```

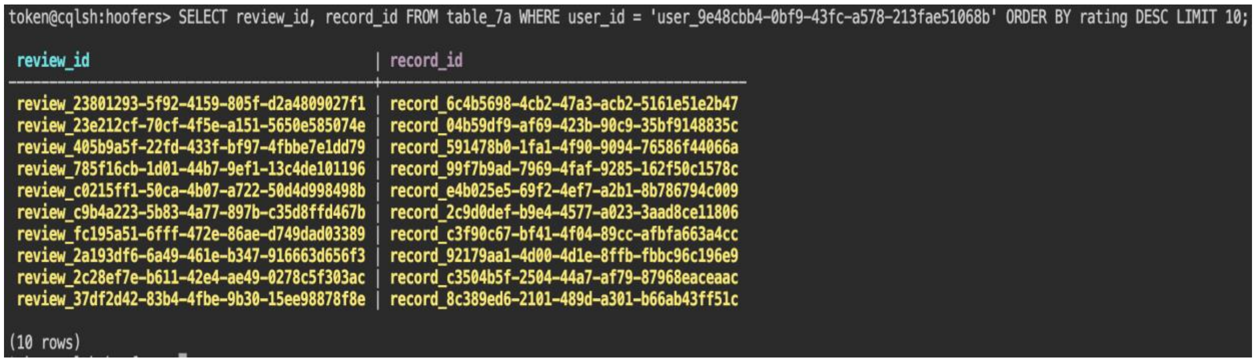
8.

a)

- CQL Query:

```
SELECT review_id, record_id
FROM table_7a WHERE user_id = ' user_9e48cbb4-0bf9-43fc-a578-213fae51068b'
ORDER BY rating DESC LIMIT 10;
```

- Result:



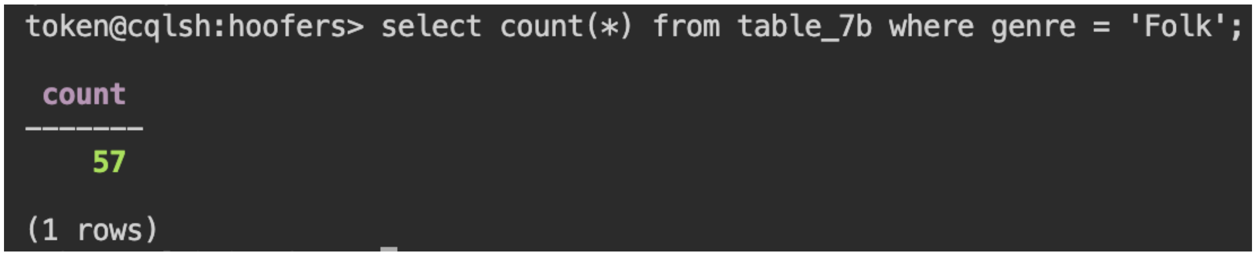
```
token@cqlsh:hoofers> SELECT review_id, record_id FROM table_7a WHERE user_id = 'user_9e48cbb4-0bf9-43fc-a578-213fae51068b' ORDER BY rating DESC LIMIT 10;

 review_id                                     | record_id
-----------------------------------------------+-----------------------------------------------
 review_23801293-5f92-4159-805f-d2a4809027f1 | record_6c4b5698-4cb2-47a3-acb2-5161e51e2b47
 review_23e212cf-70cf-4f5e-a151-5650e585074e | record_04b59df9-af69-423b-90c9-35bf9148835c
 review_405b9a5f-22fd-433f-bf97-4fbbe7e1dd79 | record_591478b0-1fa1-4f90-9094-76586f44066a
 review_785f16cb-1d01-44b7-9ef1-13c4de101196 | record_99f7b9ad-7969-4faf-9285-162f50c1578c
 review_c0215ff1-50ca-4b07-a722-50d4d998498b | record_e4b025e5-69f2-4ef7-a2b1-8b786794c009
 review_c9b4a223-5b83-4a77-897b-c35d8ffd467b | record_2c9d0def-b9e4-4577-a023-3aad8ce11806
 review_fc195a51-6fff-472e-86ae-d749dad03389 | record_c3f90c67-bf41-4f04-89cc-afbfa663a4cc
 review_2a193df6-6a49-461e-b347-916663d656f3 | record_92179aa1-4d00-4d1e-8ffb-fbbc96c196e9
 review_2c28ef7e-b611-42e4-ae49-0278c5f303ac | record_c3504b5f-2504-44a7-af79-87968eaceaac
 review_37df2d42-83b4-4fbe-9b30-15ee98878f8e | record_8c389ed6-2101-489d-a301-b66ab43ff51c

(10 rows)
```

b)

- CQL Query:

SELECT COUNT(*) FROM table_7b WHERE genre = 'Folk';

- Result:



```
token@cqlsh:hoofers> select count(*) from table_7b where genre = 'Folk';

 count
-------
    57

(1 rows)
```

c)

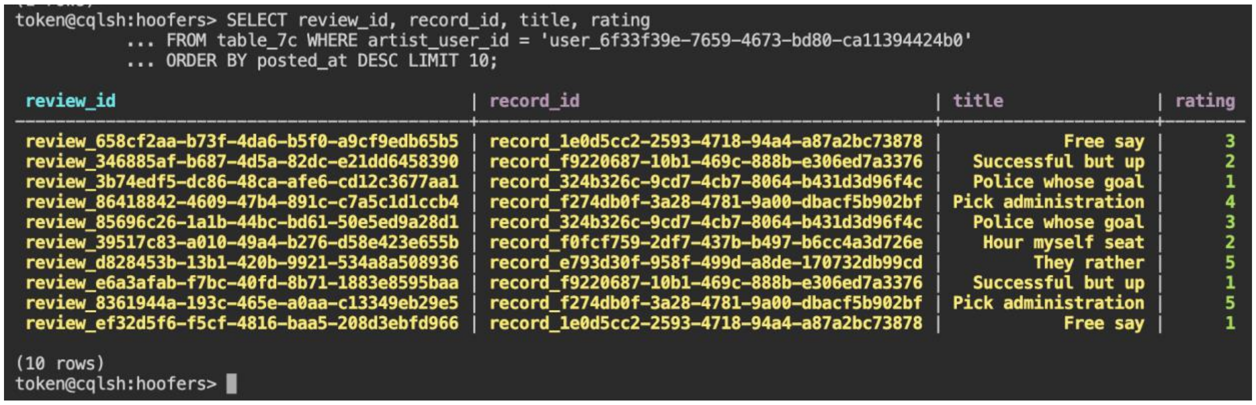
- CQL Query:

SELECT review_id, record_id, title, rating

FROM table_7c WHERE artist_user_id = 'user_6f33f39e-7659-4673-bd80-ca11394424b0'

ORDER BY posted_at DESC LIMIT 10;

- Result:



```
token@cqlsh:hoofers> SELECT review_id, record_id, title, rating
           ... FROM table_7c WHERE artist_user_id = 'user_6f33f39e-7659-4673-bd80-ca11394424b0'
           ... ORDER BY posted_at DESC LIMIT 10;

 review_id                                   | record_id                                   | title                  | rating
---------------------------------------------+---------------------------------------------+------------------------+--------
 review_658cf2aa-b73f-4da6-b5f0-a9cf9edb65b5 | record_1e0d5cc2-2593-4718-94a4-a87a2bc73878 |               Free say |      3
 review_346885af-b687-4d5a-82dc-e21dd6458390 | record_f9220687-10b1-469c-888b-e306ed7a3376 |    Successful but up |      2
 review_3b74edf5-dc86-48ca-afe6-cd12c3677aa1 | record_324b326c-9cd7-4cb7-8064-b431d3d96f4c |      Police whose goal |      1
 review_86418842-4609-47b4-891c-c7a5c1d1ccb4 | record_f274db0f-3a28-4781-9a00-dbacf5b902bf | Pick administration |      4
 review_85696c26-1a1b-44bc-bd61-50e5ed9a28d1 | record_324b326c-9cd7-4cb7-8064-b431d3d96f4c |      Police whose goal |      3
 review_39517c83-a010-49a4-b276-d58e423e655b | record_f0fcf759-2df7-437b-b497-b6cc4a3d726e |      Hour myself seat |      2
 review_d828453b-13b1-420b-9921-534a8a508936 | record_e793d30f-958f-499d-a8de-170732db99cd |           They rather |      5
 review_e6a3afab-f7bc-40fd-8b71-1883e8595baa | record_f9220687-10b1-469c-888b-e306ed7a3376 |    Successful but up |      1
 review_8361944a-193c-465e-a0aa-c13349eb29e5 | record_f274db0f-3a28-4781-9a00-dbacf5b902bf | Pick administration |      5
 review_ef32d5f6-f5cf-4816-baa5-208d3ebfd966 | record_1e0d5cc2-2593-4718-94a4-a87a2bc73878 |               Free say |      1

(10 rows)
token@cqlsh:hoofers> 
```

d)

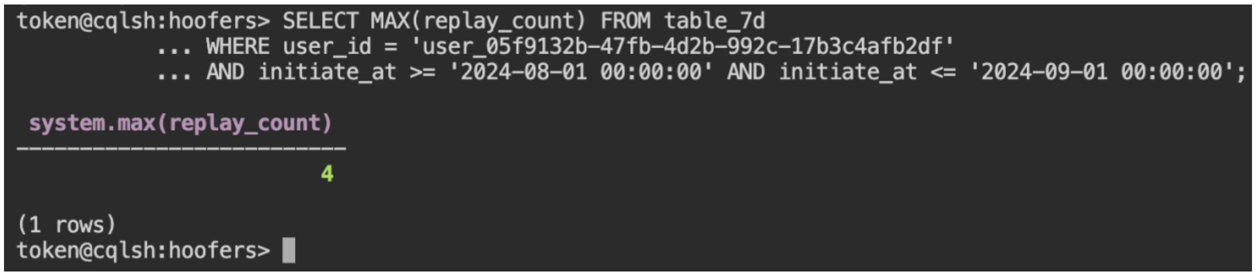
- CQL Query:

SELECT MAX(replay_count) FROM table_7d

WHERE user_id = 'user_05f9132b-47fb-4d2b-992c-17b3c4afb2df'

AND initiate_at >= '2024-08-01 00:00:00' AND initiate_at <= '2024-09-01 00:00:00';

- Result:



```
token@cqlsh:hoofers> SELECT MAX(replay_count) FROM table_7d
           ... WHERE user_id = 'user_05f9132b-47fb-4d2b-992c-17b3c4afb2df'
           ... AND initiate_at >= '2024-08-01 00:00:00' AND initiate_at <= '2024-09-01 00:00:00';

 system.max(replay_count)
--------------------------
                        4

(1 rows)
token@cqlsh:hoofers> 
```

9.

- CQL INSERT statements:

```
INSERT INTO records (record_id, artist_user_id, title, genre, release_date)
VALUES ('record_d2f498f8-d7ff-4f1c-a967-7090417751f5', 'user_38eaa9f8-e8fc-4ce4-a8ae-
ffb882c1786c', 'Blue by You', 'Rock', '2024-10-07');
```

```
INSERT INTO users (user_id, city, email, genres, joined_date, nickname, state, street, zip) VALUES
('user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c', NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL);
```

```
INSERT INTO reviews (review_id, body, posted_at, rating, record_id, user_id) VALUES ('review_new',
NULL, NULL, NULL, 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5', 'user_38eaa9f8-e8fc-4ce4-a8ae-
ffb882c1786c');
```

```
INSERT INTO sessions (session_id, device, initiate_at, leave_at, music_quality, record_id, remaining_time,
replay_count, track_number, user_id) VALUES ('session_new', NULL, NULL, NULL, NULL,
'record_d2f498f8-d7ff-4f1c-a967-7090417751f5', NULL, NULL, NULL, 'user_38eaa9f8-e8fc-4ce4-a8ae-
ffb882c1786c');
```

```
INSERT INTO table_7a (user_id, rating, review_id, record_id) VALUES ('user_38eaa9f8-e8fc-4ce4-a8ae-
ffb882c1786c', 5, 'review_new', 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5');
```

```
INSERT INTO table_7b (genre, record_id) VALUES ('Rock', 'record_d2f498f8-d7ff-4f1c-a967-
7090417751f5');
```

```
INSERT INTO table_7c (artist_user_id, posted_at, review_id, rating, record_id, title) VALUES
('user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c', '2024-10-08', 'review_new', 5, 'record_d2f498f8-d7ff-
4f1c-a967-7090417751f5', 'Blue By You');
```

```
INSERT INTO table_7d (user_id, initiate_at, session_id, replay_count) VALUES ('user_38eaa9f8-e8fc-4ce4-
a8ae-ffb882c1786c', '2024-10-08T11:00:00Z', 'session_new', 1);
```

- Verification queries:

SELECT * FROM records WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5';

SELECT * FROM users WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c';

SELECT * FROM reviews WHERE review_id = 'review_new';

SELECT * FROM sessions WHERE session_id = 'session_new';

SELECT * FROM table_7a WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c';

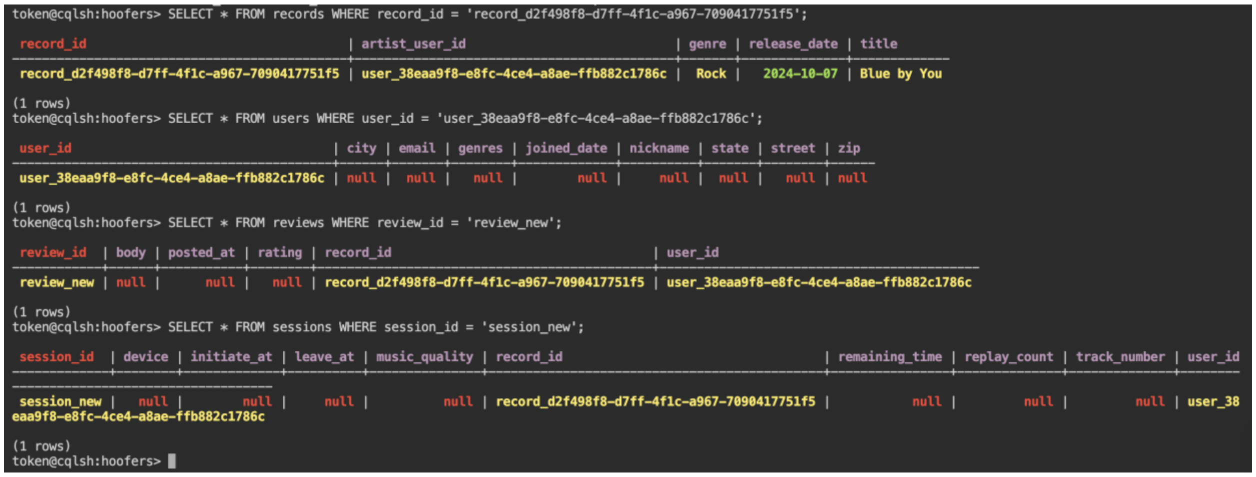
SELECT * FROM table_7b WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5';

SELECT * FROM table_7c WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5';

SELECT * FROM table_7d WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c';

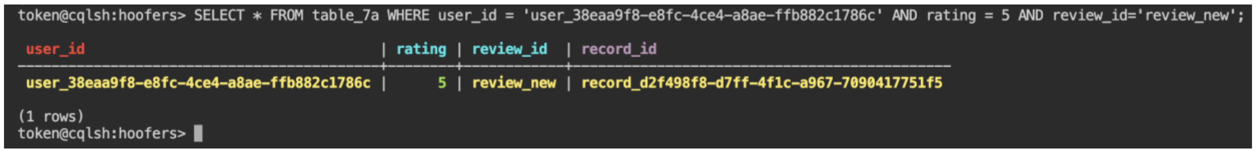
- Result:

First 4 queries above.



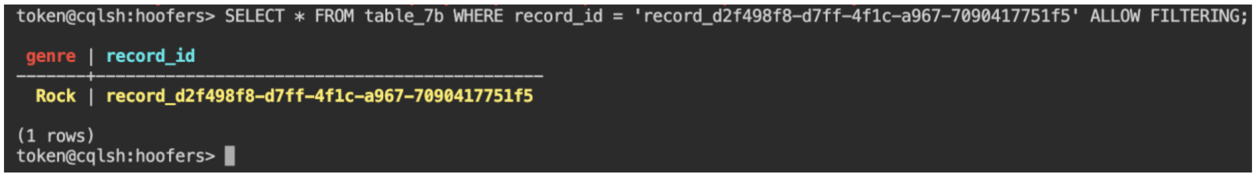
```
token@cqlsh:hoofers> SELECT * FROM records WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5';

 record_id                                   | artist_user_id                            | genre | release_date | title
---------------------------------------------+-------------------------------------------+-------+--------------+-------------
 record_d2f498f8-d7ff-4f1c-a967-7090417751f5 | user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c |  Rock |   2024-10-07 | Blue by You

(1 rows)
token@cqlsh:hoofers> SELECT * FROM users WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c';

 user_id                                   | city | email | genres | joined_date | nickname | state | street | zip
-------------------------------------------+------+-------+--------+-------------+----------+-------+--------+------
 user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c | null |  null |   null |        null |     null |  null |   null | null

(1 rows)
token@cqlsh:hoofers> SELECT * FROM reviews WHERE review_id = 'review_new';

 review_id  | body | posted_at | rating | record_id                                   | user_id
------------+------+-----------+--------+---------------------------------------------+-------------------------------------------
 review_new | null |      null |   null | record_d2f498f8-d7ff-4f1c-a967-7090417751f5 | user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c

(1 rows)
token@cqlsh:hoofers> SELECT * FROM sessions WHERE session_id = 'session_new';

 session_id  | device | initiate_at | leave_at | music_quality | record_id                                   | remaining_time | replay_count | track_number | user_id
-------------+--------+-------------+----------+---------------+---------------------------------------------+----------------+--------------+--------------+---------
-------------------------------------
 session_new |   null |        null |     null |          null | record_d2f498f8-d7ff-4f1c-a967-7090417751f5 |           null |         null |         null | user_38
eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c

(1 rows)
token@cqlsh:hoofers> 
```

Verifying table_7a



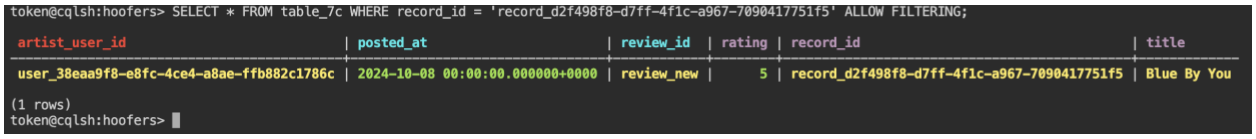
```
token@cqlsh:hoofers> SELECT * FROM table_7a WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c' AND rating = 5 AND review_id='review_new';

 user_id                                   | rating | review_id  | record_id
-------------------------------------------+--------+------------+---------------------------------------------
 user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c |      5 | review_new | record_d2f498f8-d7ff-4f1c-a967-7090417751f5

(1 rows)
token@cqlsh:hoofers> 
```

Verifying table_7b



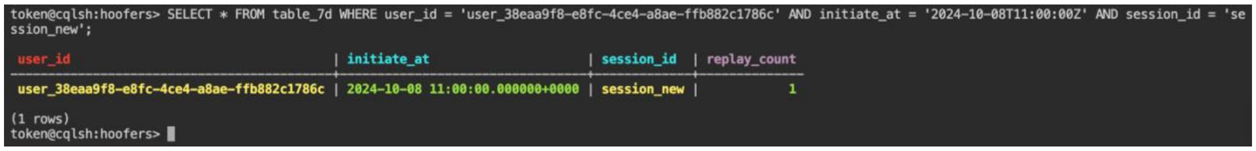
```
token@cqlsh:hoofers> SELECT * FROM table_7b WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5' ALLOW FILTERING;

 genre | record_id
-------+---------------------------------------------
  Rock | record_d2f498f8-d7ff-4f1c-a967-7090417751f5

(1 rows)
token@cqlsh:hoofers> 
```

Verifying table_7c



```
token@cqlsh:hoofers> SELECT * FROM table_7c WHERE record_id = 'record_d2f498f8-d7ff-4f1c-a967-7090417751f5' ALLOW FILTERING;

 artist_user_id                            | posted_at                       | review_id  | rating | record_id                                   | title
-------------------------------------------+---------------------------------+------------+--------+---------------------------------------------+-------------
 user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c | 2024-10-08 00:00:00.000000+0000 | review_new |      5 | record_d2f498f8-d7ff-4f1c-a967-7090417751f5 | Blue By You

(1 rows)
token@cqlsh:hoofers> 
```

Verifying table_7d



```
token@cqlsh:hoofers> SELECT * FROM table_7d WHERE user_id = 'user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c' AND initiate_at = '2024-10-08T11:00:00Z' AND session_id = 'se
ssion_new';

 user_id                                   | initiate_at                     | session_id  | replay_count
-------------------------------------------+---------------------------------+-------------+--------------
 user_38eaa9f8-e8fc-4ce4-a8ae-ffb882c1786c | 2024-10-08 11:00:00.000000+0000 | session_new |            1

(1 rows)
token@cqlsh:hoofers> 
```

10.

Python script:

from cassandra.cluster import Cluster

from cassandra.query import SimpleStatement

from cassandra.auth import PlainTextAuthProvider

from datetime import datetime

# Connect to your Cassandra cluster

# Path to your secure connect bundle

secure_connect_bundle = '/Users/ameya/Desktop/DataLoading/secure-connect-cs224p-fall.zip'

# Astra DB credentials

# For client_id and client_secret, I have not filled them out here, but in my code I would fill out the
# necessary values from the downloaded JSON token file.

client_id = 'client_id'

client_secret = 'client_secret'

# Set up authentication and connection

auth_provider = PlainTextAuthProvider(client_id,client_secret)

cluster = Cluster(cloud={'secure_connect_bundle': secure_connect_bundle}, auth_provider=auth_provider)

session = cluster.connect('hoofers')

```
# This function replicates the query from question 9 to add the record to all the tables.

# It takes as input all the possible parameters that may be of interest, but we might only use a few of them per table

# The ‘%s’ serves as a place holder for the relevant variables for each table.

def addRecord(session, record_id, artist_user_id, title, genre, release_date, user_id, review_id, rating, session_id):

        # Insert into records table

        session.execute( SimpleStatement( "INSERT INTO records (record_id, artist_user_id, title, genre, release_date) VALUES (%s, %s, %s, %s, %s)", consistency_level=ConsistencyLevel.ONE ), (record_id, artist_user_id, title, genre, release_date) )


        # Insert into users table

        session.execute( SimpleStatement( "INSERT INTO users (user_id, city, email, genres, joined_date, nickname, state, street, zip) VALUES (%s, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)", consistency_level=ConsistencyLevel.ONE ), (user_id,) )


        # Insert into reviews table

        session.execute( SimpleStatement( "INSERT INTO reviews (review_id, body, posted_at, rating, record_id, user_id) VALUES (%s, NULL, NULL, %s, %s, %s)", consistency_level=ConsistencyLevel.ONE ), (review_id, rating, record_id, user_id) )


        # Insert into sessions table

        session.execute( SimpleStatement( "INSERT INTO sessions (session_id, device, initiate_at, leave_at, music_quality, record_id, remaining_time, replay_count, track_number, user_id) VALUES (%s, NULL, NULL, NULL, NULL, %s, NULL, NULL, NULL, %s)", consistency_level=ConsistencyLevel.ONE ), (session_id, record_id, user_id) )


        # Insert into table_7a

        session.execute( SimpleStatement( "INSERT INTO table_7a (user_id, rating, review_id, record_id) VALUES (%s, %s, %s, %s)", consistency_level=ConsistencyLevel.ONE ), (user_id, rating, review_id, record_id) )


        # Insert into table_7b

        session.execute( SimpleStatement( "INSERT INTO table_7b (genre, record_id) VALUES
(%s, %s)", consistency_level=ConsistencyLevel.ONE ), (genre, record_id) )
        # Insert into table_7c
        session.execute( SimpleStatement( "INSERT INTO table_7c (artist_user_id, posted_at, review_id, rating, record_id, title) VALUES (%s, %s, %s, %s, %s, %s)", consistency_level=ConsistencyLevel.ONE ), (artist_user_id, datetime.now().isoformat(), review_id, rating, record_id, title) )

        # Insert into table_7d
        session.execute( SimpleStatement( "INSERT INTO table_7d (user_id, initiate_at, session_id, replay_count) VALUES (%s, %s, %s, %s)", consistency_level=ConsistencyLevel.ONE ), (user_id, datetime.now().isoformat(), session_id, 1) )



# Query to call the function and add the given record to the database, connected to ‘hoofers’ #keyspace.
session = connect_to_cassandra()
add_record( session,
‘record_632fe768-eecb-4596-9780-cc21734feec5’,
 ‘user_b91cf915-487b-42fc-b6b8-6c17935bb755’,
 ‘One Sour Day’,
 'R&B',
 '2024-10-07',
‘user_b91cf915-487b-42fc-b6b8-6c17935bb755’,
 'review_new',
 5,
 'session_new' )

session.shutdown()
```