Name: Chuqi Wang

Student ID: 79167724

Q1:

A:

i:

```
LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Users.csv' AS row

MERGE (u:User {user_id: row.user_id})

ON CREATE SET

   u.email = row.email,

   u.joined_date = date(row.joined_date),

   u.nickname = row.nickname,

   u.street = row.street,

   u.city = row.city,

   u.state = row.state,

   u.zip = toInteger(row.zip),

   u.genres = split(row.genres, ",");


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Listeners.csv' AS row

MERGE (u:User {user_id: row.user_id})

SET u:Listener,

   u.subscription = row.subscription,

   u.first_name = row.first_name,

   u.last_name = row.last_name;
```

```
LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Artists.csv' AS row

MERGE (u:User {user_id: row.user_id})

SET u:Artist,

   u.bio = row.bio,

   u.stagename = row.stagename;


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Records.csv' AS row

MERGE (r:Record {record_id: row.record_id})

ON CREATE SET

   r.title = row.title,

   r.genre = row.genre;


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Albums.csv' AS row

MERGE (r:Record {record_id: row.record_id})

SET r:Album,

   r.description = row.description;


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Singles.csv' AS row

MATCH (r:Record {record_id: row.record_id})

SET r:Single,

   r.video_url = row.video_url;
```

```
LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Reviews.csv' AS row

MERGE (r:Review {review_id: row.review_id})

ON CREATE SET

   r.rating = toInteger(row.rating),

   r.body = row.body;
```

ii:

```
CREATE INDEX user_id_index FOR (u:User) ON (u.user_id);

CREATE INDEX record_id_index FOR (r:Record) ON (r.record_id);
```

iii:

```
LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Artists_Releases_Record.csv' AS row

MATCH (a:Artist {user_id: row.artist_user_id})

MATCH (r:Record {record_id: row.record_id})

MERGE (a)-[:RELEASE {release_date: date(row.release_date)}]->(r);


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Users_Post_Reviews.csv' AS row

MATCH (u:User {user_id: row.user_id})

MATCH (r:Review {review_id: row.review_id})

MERGE (u)-[:POSTED_AT {posted_at: datetime(replace(row.posted_at, " ", "T"))}]->(r);


LOAD CSV WITH HEADERS FROM
'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Reviews_About_Record.csv' AS row

MATCH (r:Review {review_id: row.review_id})

MATCH (rec:Record {record_id: row.record_id})
```

MERGE (r)-[:ABOUT]->(rec);

LOAD CSV WITH HEADERS FROM 'file:///Users/chuqiwang/Desktop/UCI/CS224P/assignments/HW4/zot-music-dataset-assignment4/Upvotes.csv' AS row

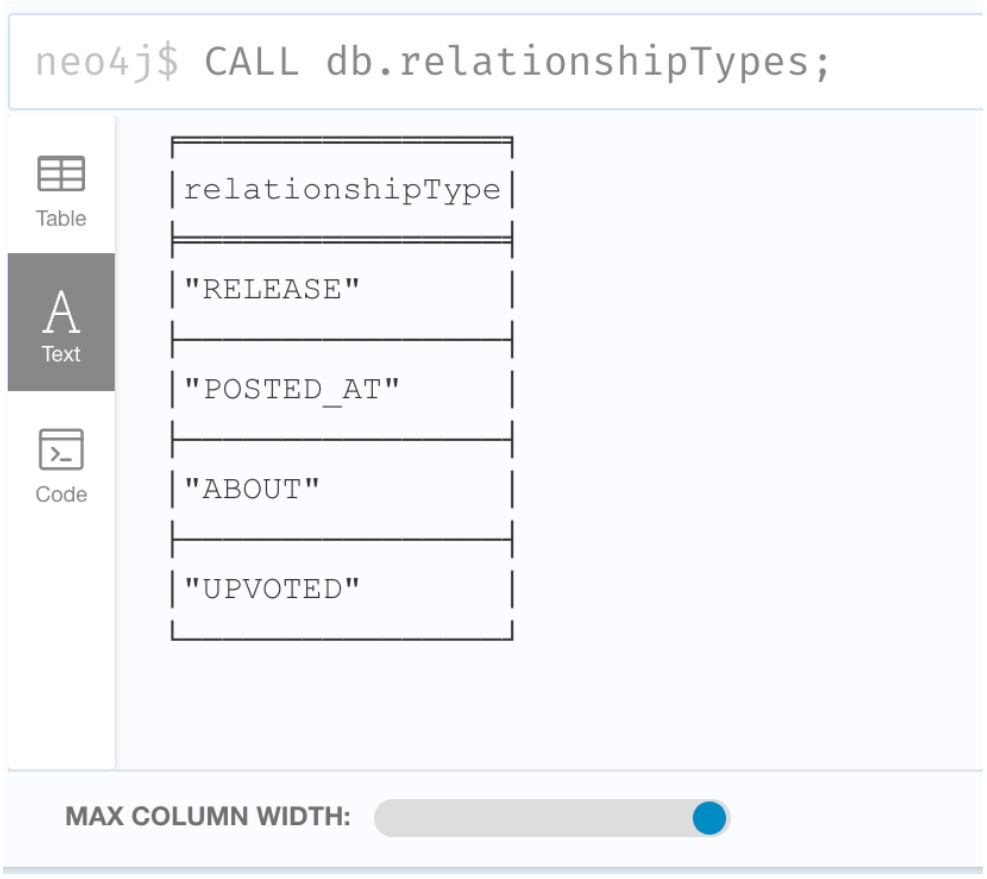
MATCH (u:User {user_id: row.user_id})

MATCH (r:Review {review_id: row.review_id})

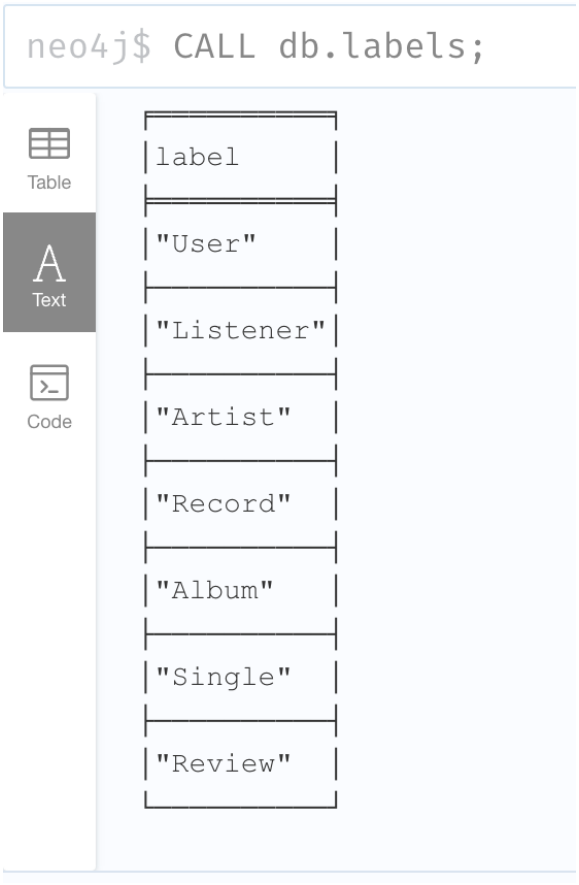
MERGE (u)-[:UPVOTED]->(r);

B:

i:



ii:



Q2:

Query:

MATCH (r:Record)

WITH r LIMIT 1000

UNWIND keys(r) AS recordKeys

RETURN DISTINCT recordKeys;

Results (screenshot below):

```
1 MATCH (r:Record)
2 WITH r LIMIT 1000
3 UNWIND keys(r) AS recordKeys
4 RETURN DISTINCT recordKeys;
```

Table

Text

Code

```
╒═══════════════╕
│recordKeys     │
╞═══════════════╡
│"record_id"    │
├───────────────┤
│"title"        │
├───────────────┤
│"genre"        │
├───────────────┤
│"video_url"    │
├───────────────┤
│"description"  │
└───────────────┘
```

Q3 A:

Query:

MATCH (u:User)

RETURN u

ORDER BY u.joined_date DESC

LIMIT 5;

Results (screenshot below):

Q3 B:

Query:

MATCH (u:User {user_id: "user_WKWGxlZa"})-[rel:RELEASE]->(r:Record)

RETURN r, rel.release_date

ORDER BY r.record_id ASC;

Results (screenshot below):

```
MATCH (u:User {user_id: "user_WKWGxlZa"})-[rel:RELEASE]→(r:Record)
RETURN r, rel.release_date
ORDER BY r.record_id ASC;
```

Graph | Table | Text | Code

r	rel.release_date
(:Record:Single {record_id: "record_1487gZZ9",video_url: "http://www.williams.net/",genre: "Indie",title: "Civil government find"})	"2020-08-08"
(:Album:Record {record_id: "record_696_41KT",genre: "Techno",description: "Result notice as Congress fact recently safe. Behavior hear what.\nOthers often worker make notice. Security affect picture tell matter tax example teach.",title: "Hope there scientist"})	"2022-12-26"
(:Album:Record {record_id: "record_9PGDade0",genre: "Gospel",title: "Bring official"})	"2020-07-04"
(:Record:Single {record_id: "record_9WSnH7xj",video_url: "https://www.romero.com/",genre: "Indie",title: "Become better"})	"2022-01-07"
(:Record:Single {record_id: "record_9i6Xv84O",video_url: "http://www.wright-campos.com/",genre: "Folk",title: "Impact wind true"})	"2022-04-11"
(:Album:Record {record_id: "record__rjS126-",genre: "Gospel",description: "Mouth thousand vote price employee member. Region just record face floor sport ball. Method scene low building buy we however.",title: "Whom"})	"2023-11-09"
(:Album:Record {record_id: "record_dVCtabgv",genre: "Rock",description: "Forget order hundred light successful. Career participant agree. Light detail one threat clear. Hold action treatment firm difference feeling.\nCertainly major person road along board field you.",title: "Court relate"})	"2023-05-21"
(:Album:Record {record_id: "record_ffeLJn0d",genre: "Techno",description: "Care four use always act. Edge compare occur real. Unit answer take involve white administration item.",title: "Professional than"})	"2021-06-01"
(:Album:Record {record_id: "record_pCcJKFJf",genre: "Blues",description: "Position thus look billion movement discuss. Discuss early time mention feeling better future again.",title: "Edge lay"})	"2023-08-01"

Q3 C:

Query:

MATCH (l:Listener)

WHERE NOT (l)-[:POSTED_AT]->(:Review)

RETURN COUNT(l) AS count_of_listeners_without_reviews;

Results (screenshot below):

```
1  MATCH (l:Listener)
2  WHERE NOT (l)-[:POSTED_AT]→(:Review)
3  RETURN COUNT(l) AS count_of_listeners_without_reviews;
4
```

Table | Text | Code

```
╒═══════════════════════════════════╕
│count_of_listeners_without_reviews│
╞═══════════════════════════════════╡
│10                                 │
└───────────────────────────────────┘
```

Q3 D:

Query:

MATCH (l:Listener)-[:POSTED_AT]->(r:Review)-[:ABOUT]->(rec:Record {title: "Audience star apply"})

WHERE r.rating = 5

RETURN DISTINCT l.first_name, l.last_name

ORDER BY l.first_name ASC;

Results (screenshot below):

```
1  MATCH (l:Listener)-[:POSTED_AT]→(r:Review)-[:ABOUT]→(rec:Record {title: "Audience star apply"})
2  WHERE r.rating = 5
3  RETURN DISTINCT l.first_name, l.last_name
4  ORDER BY l.first_name ASC;
5
```

Table | Text | Code

```
╒═══════════╤══════════╕
│l.first_name│l.last_name│
╞═══════════╪══════════╡
│"Doris"    │"Baker"   │
├───────────┼──────────┤
│"Mark"     │"Allen"   │
└───────────┴──────────┘
```

Q3 E:

Query:

MATCH (rec:Record)<-[:ABOUT]-(r:Review)

WITH rec, COUNT(r) AS review_count

WHERE review_count > 25

RETURN rec.record_id, rec.title, review_count

ORDER BY review_count DESC;

Results (screenshot below):

```
1 MATCH (rec:Record)←[:ABOUT]-(r:Review)
2 WITH rec, COUNT(r) AS review_count
3 WHERE review_count > 25
4 RETURN rec.record_id, rec.title, review_count
5 ORDER BY review_count DESC;
6
```

Table

Text

Code

rec.record_id	rec.title	review_count
"record_51zx5w5v"	"Focus idea defense"	26

Q3 F:

Query:

MATCH (l:Listener)-[:POSTED_AT]->(r:Review)

WITH l, COUNT(r) AS review_count

WHERE review_count > 60

MATCH (l)-[:UPVOTED]->(u:Review)

WITH l, review_count, COUNT(u) AS upvote_count

WHERE upvote_count > 500

RETURN l.user_id

ORDER BY l.user_id ASC

LIMIT 5;

Results (screenshot below):

```
1  MATCH (l:Listener)-[:POSTED_AT]→(r:Review)
2  WITH l, COUNT(r) AS review_count
3  WHERE review_count > 60
4  MATCH (l)-[:UPVOTED]→(u:Review)
5  WITH l, review_count, COUNT(u) AS upvote_count
6  WHERE upvote_count > 500
7  RETURN l.user_id
8  ORDER BY l.user_id ASC
9  LIMIT 5;
10
```

Table

Text

Code

l.user_id
"user_RFwrYnro"
"user_V0VNPkG9"
"user_kH57AloB"
"user_s20mtRZS"
"user_vJ-Cm6Mk"

Q3 G:

Query:

MATCH (l1:Listener)-[:POSTED_AT]->(r1:Review)-[:ABOUT]->(rec:Record)<-[:ABOUT]-(r2:Review)<-[:POSTED_AT]-(l2:Listener)

WHERE l1 <> l2 AND r1.rating = r2.rating AND id(l1) < id(l2)

RETURN DISTINCT l1.last_name AS listener1_last_name, l2.last_name AS listener2_last_name, rec.record_id

ORDER BY rec.record_id ASC

LIMIT 10;

Results (screenshot below):

```
1 MATCH (l1:Listener)-[:POSTED_AT]→(r1:Review)-[:ABOUT]→(rec:Record)←[:ABOUT]-(r2:Review)←
  [:POSTED_AT]-(l2:Listener)
2 WHERE l1 <> l2 AND r1.rating = r2.rating AND id(l1) < id(l2)
3 RETURN DISTINCT l1.last_name AS listener1_last_name, l2.last_name AS listener2_last_name,
  rec.record_id
4 ORDER BY rec.record_id ASC
5 LIMIT 10;
6
```

Table | Text | Warn | Code

listener1_last_name	listener2_last_name	rec.record_id
"Wyatt"	"Young"	"record_--4_Ht0N"
"Myers"	"Sanders"	"record_--4_Ht0N"
"Sanders"	"Gallegos"	"record_--4_Ht0N"
"Myers"	"Gallegos"	"record_--4_Ht0N"
"Mullins"	"Sherman"	"record_--4_Ht0N"
"Copeland"	"Sherman"	"record_--4_Ht0N"
"Mullins"	"Copeland"	"record_--4_Ht0N"
"Brown"	"Wright"	"record_-2WQnPOg"
"Copeland"	"Wright"	"record_-2WQnPOg"
"Allen"	"Taylor"	"record_-2WQnPOg"

Q3 H:

Query:

MATCH (a:User)-[:POSTED_AT]->(r1:Review)-[:ABOUT]->(rec1:Record)<-[:RELEASE]-(b:User),

(b)-[:POSTED_AT]->(r2:Review)-[:ABOUT]->(rec2:Record)<-[:RELEASE]-(a)

WHERE id(a) < id(b)

WITH DISTINCT a.nickname AS user_a_nickname, b.nickname AS user_b_nickname

RETURN user_a_nickname, user_b_nickname

ORDER BY user_a_nickname ASC

LIMIT 10;

Results (screenshot below):

```
MATCH (a:User)-[:POSTED_AT]→(r1:Review)-[:ABOUT]→(rec1:Record)←[:RELEASE]-(b:User),
      (b)-[:POSTED_AT]→(r2:Review)-[:ABOUT]→(rec2:Record)←[:RELEASE]-(a)
WHERE id(a) < id(b)
WITH DISTINCT a.nickname AS user_a_nickname, b.nickname AS user_b_nickname
RETURN user_a_nickname, user_b_nickname
ORDER BY user_a_nickname ASC
LIMIT 10;
```

Table | Text | Warn | Code

	user_a_nickname	user_b_nickname
1	"amandawatkins"	"velasquezmargaret"
2	"amandawatkins"	"nataliecoffey"
3	"amandawatkins"	"smithhelen"
4	"anthony59"	"chelsealawson"
5	"anthony59"	"smithhelen"
6	"anthony59"	"zjohnson"
7	"anthony59"	"courtney36"
8	"anthony59"	"candace45"
9	"bettyhahn"	"freemanlaurie"
10	"bettyhahn"	"sandersjanet"

Q3 I:

i

Query:

MATCH (start:Artist {user_id: "user_0ZIeALBX"}), (target:Artist)

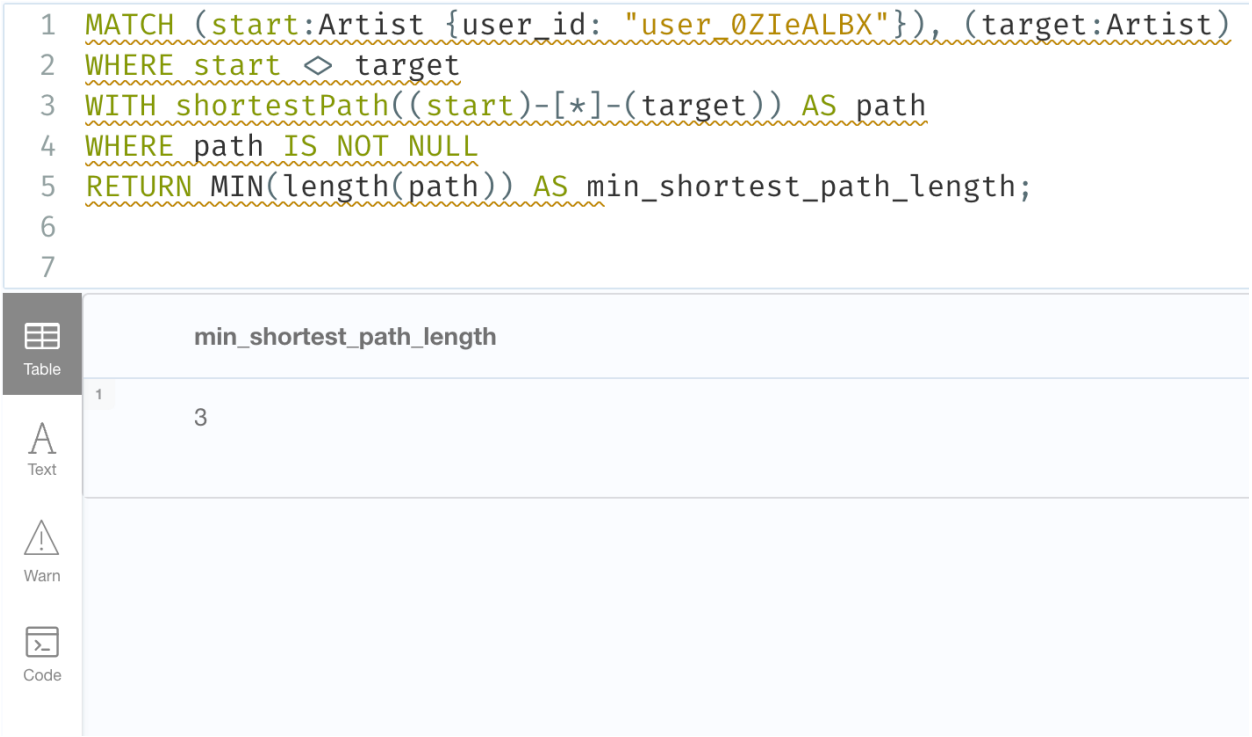
WHERE start <> target

WITH shortestPath((start)-[*]-(target)) AS path

WHERE path IS NOT NULL

RETURN MIN(length(path)) AS min_shortest_path_length;

Results (screenshot below):



ii

Query:

MATCH (start:Artist {user_id: "user_0ZIeALBX"}), (target:Artist)

WHERE start <> target

WITH shortestPath((start)-[*]-(target)) AS path, target

WHERE length(path) = 6

RETURN target.user_id AS artist_user_id

ORDER BY artist_user_id ASC

LIMIT 5;

Results (screenshot below):

```
MATCH (start:Artist {user_id: "user_0ZIeALBX"}), (target:Artist)
WHERE start <> target
WITH shortestPath((start)-[*]-(target)) AS path, target
WHERE length(path) = 6
RETURN target.user_id AS artist_user_id
ORDER BY artist_user_id ASC
LIMIT 5;
```

Table | Text | Warn | Code

	artist_user_id
1	"user_-2Ijew-e"
2	"user_-EzOnF9o"
3	"user_AQrv2BCA"
4	"user_ByXkfiU9"
5	"user_D5t1hYVg"