PRADYOTH SINGENAHALLI PRABHU
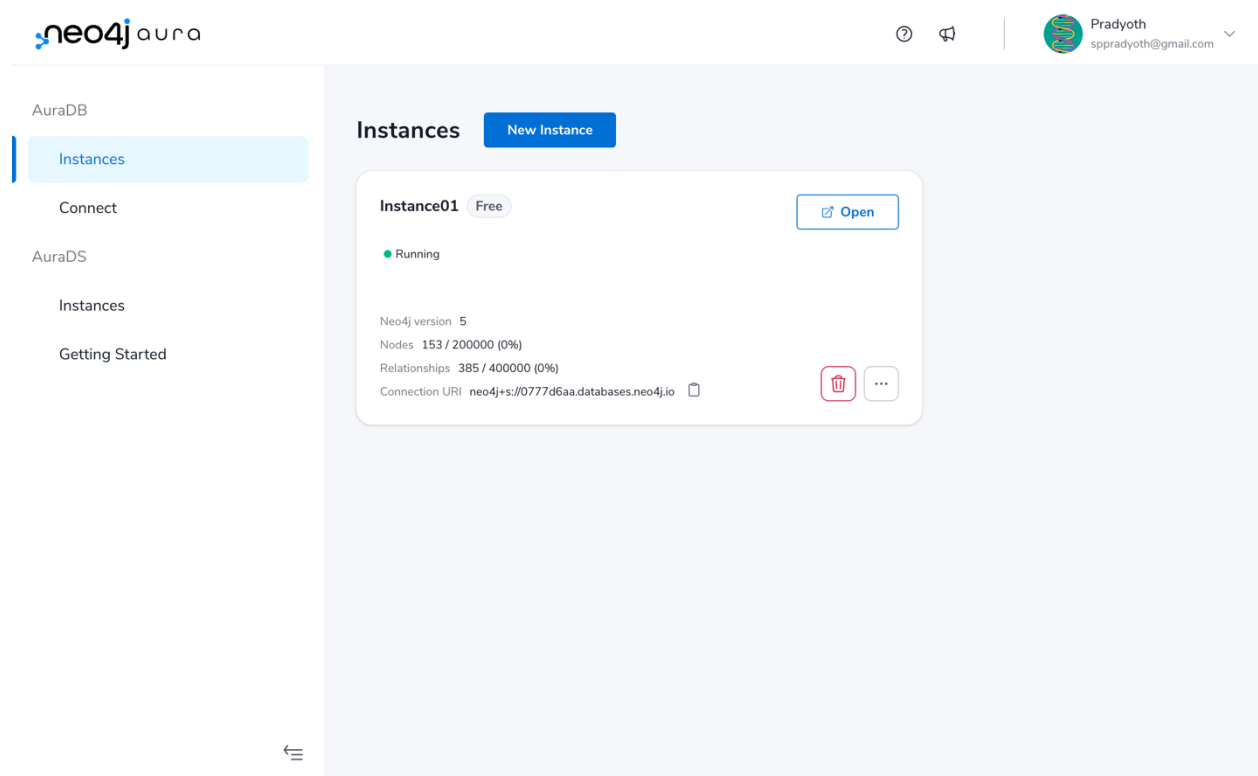02071847
psingenahalliprabhu@umassd.edu

CIS 552: Database Design – Homework 7

Introduction:

Neo4j is a highly scalable, native graph database that stores and manages large, interconnected datasets. It uses a property graph model to organize data into nodes and relationships for efficient querying. Neo4j is well-suited for social networking, recommendation engines, fraud detection, and identity and access management use cases. It offers a flexible data model, powerful query language, and indexing capabilities.

Creating a Database:

The above image shows the step to create a database.

After creating the database, I need to connect to the database via the neo4j website with the user and Password provided.

Inserting Nodes and Edges:

After inserting the Nodes and Edges via the cypher code in the Ch14_FCC.txt. This is the result we get as shown below.

Query Database using Cypher Query Language:

1. Retrieve Restaurants and Cuisines Owned by Member Cecil Esparza:

```
1  MATCH (m:Member {fname: 'Cecil', lname: 'Esparza'})-[:OWNS]→(r:Restaurant)
2  OPTIONAL MATCH (r)-[:TYPE_OF]→(c:Cuisine)
3  RETURN m.fname AS memberFirstName, m.lname AS memberLastName, r.name AS restaurantName, c.name AS
   cuisineName
```

memberFirstName	memberLastName	restaurantName	cuisineName
"Cecil"	"Esparza"	"Over the Counter Foods"	"Sandwich and Salad"

Showing **1** of **1** results

Show   50 ⌄

2. Retrieve Restaurant, Owner, and Cuisine Information for Restaurant with ID 4625:

```
1  MATCH (r:Restaurant {rid: 4625})
2  OPTIONAL MATCH (r)←[:OWNS]-(m:Member)
3  OPTIONAL MATCH (r)-[:TYPE_OF]→(c:Cuisine)
4  RETURN r.name AS restaurantName, r.street AS restaurantStreet, r.city AS restaurantCity, r.state AS
   restaurantState, r.zip AS restaurantZip,
5      m.fname AS ownerFirstName, m.lname AS ownerLastName, m.username AS ownerUsername, m.email AS
   ownerEmail,
6      c.name AS cuisineName
```

Table   RAW

restaurantName	restaurantStreet	restaurantCity	restaurantState	restaurantZip	ownerFirstName
"Beijing Table"	"1980 Woodstock Circle"	"South Egremont"	"TN"	"1252"	"Rocco"

Showing **1** of **1** results   Show 50 ⌄

3. Retrieve Restaurant Review and Member Information for Member Cecil Esparza and Restaurant with ID 4657:

```
1  MATCH (m:Member {fname: 'Cecil', lname: 'Esparza'})-[:REVIEWED]→(r:Restaurant {rid: 4657})
2  RETURN m.fname AS memberFirstName, m.lname AS memberLastName, r.name AS restaurantName,
3      r.phone AS restaurantPhone, r.price AS restaurantPrice,
4      r.service AS reviewService, r.taste AS reviewTaste, r.atmosphere AS reviewAtmosphere, r.value AS
   reviewValue
```
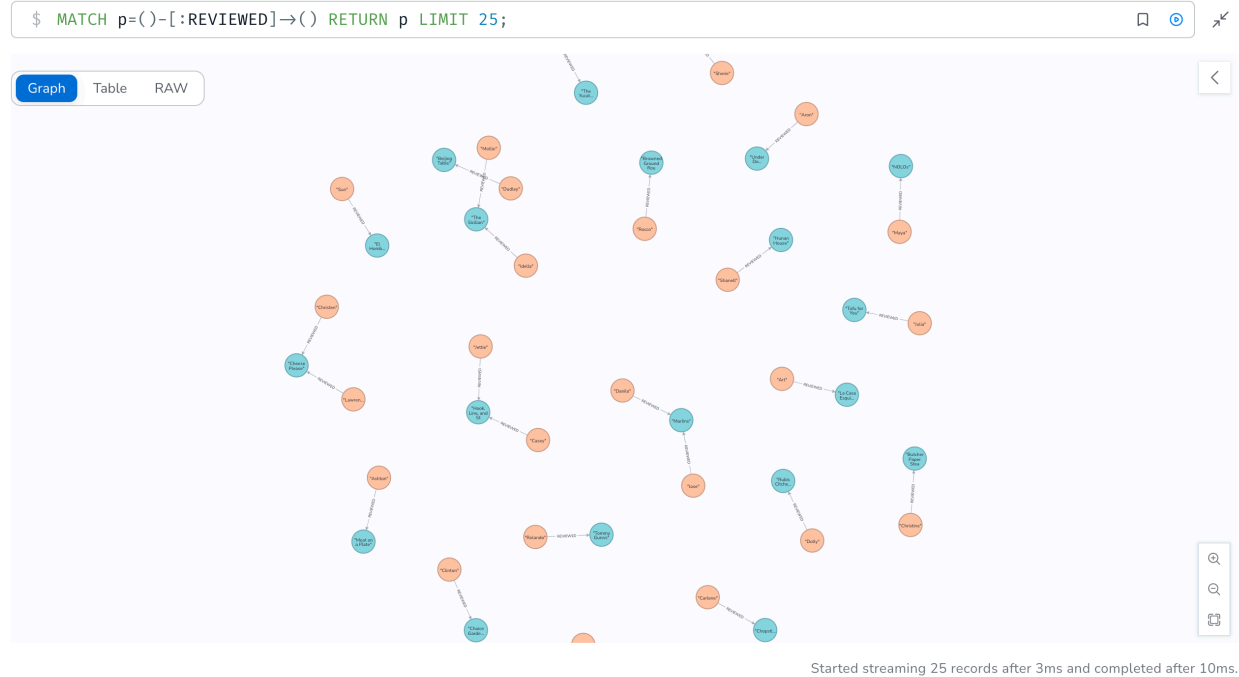
Table   RAW

memberFirstName	memberLastName	restaurantName	restaurantPhone	restaurantPrice	reviewService
"Cecil"	"Esparza"	"Over the Counter Foods"	"(615) 555-7641"	1	*null*

Showing **1** of **1** results   Show 50 ⌄

4. Retrieve 25 Reviews with Their Associated Nodes:

```
$ MATCH p=()-[:REVIEWED]→() RETURN p LIMIT 25;
```


Started streaming 25 records after 3ms and completed after 10ms.

Conclusion:

In conclusion, Neo4j is a widely recognized and efficient graph database management solution that enables data storage and retrieval in a graph structure. It is particularly useful for complex use cases, such as social networks and fraud detection, due to its ability to represent and query intricate relationships. The Cypher query language used by Neo4j provides a flexible and expressive syntax for querying graph data, allowing developers to create and query complex relationships with ease. Neo4j's graph model and indexing capabilities make it a powerful tool for data analysis and insights. Overall, Neo4j is a valuable resource for developers, data scientists, and business analysts seeking to leverage graph database technology for their projects.