**Assignment 3**

**Graph Databases**

***Sanjhana Rangaraj (sr55737)***

**User Nodes:**

1. User ID: 1, Name: Isaac, Last Name: Newton, Age 20

2. User ID: 2, Name: Richard, Last Name: Feynman, Age 30

3. User ID: 3, Name: Chandrasekhara, Last Name: Raman, Age 40

4. User ID: 4, Name: Srinivasa, Last Name: Ramanujam, Age 25

5. User ID: 5, Name: Galileo, Last Name: Galilei, Age 78

6. User ID: 6, Name: Marie, Last Name: Curie, Age 35

7. User ID: 7, Name: Albert, Last Name: Einstein, Age 100

8. User ID: 8, Name: Oviyasri, Last Name: Aryabhata, Age 50

9. User ID: 9, Name: Rabindranath, Last Name: Tagore, Age 90

10. User ID: 10, Name: Paul, Last Name: Erdos, Age 75

**Group Nodes:**

11. Group ID: 101, Name: Isotope Enthusiasts

12. Group ID: 102, Name: Food Lovers

13. Group ID: 103, Name: Tech Innovators

**Post Nodes:**

14. Post ID: 201, Title: "Exploring Radium"

15. Post ID: 202, Title: "Best Pizza Places"

16. Post ID: 203, Title: "AI Breakthroughs"

**Comment Nodes:**

17. Comment ID: 301, Text: "Awesome photos!"

18. Comment ID: 302, Text: "I want to visit the moon."

19. Comment ID: 303, Text: "That pizza looks delicious!"

**Relationships:**

- Isaac created Post 201

- Richard created Post 202

- Chandrasekhara created Post 203

- Isaac commented on Post 202

- Richard commented on Post 201

- Chandrasekhara commented on Post 201

- Srinivasa liked Post 201

- Srinivasa liked Comment 302

- Galileo liked Comment 301

- Galileo liked Comment 303

- Marie created Group 101

- Albert created Group 102

- Oviyasri created Group 103

- Isaac is a member of Group 101

- Richard is a member of Group 102

- Chandrasekhara is a member of Group 103

- Paul is a member of Group 101

- Rabindranath is a member of Group 102

**Questions:**

1. **Find all posts created by Isaac and list the users who liked those posts.**

**Solution:**

**Creating Nodes:**

// Create User Nodes

CREATE (u1:User {id: 1, name: 'Isaac', last\_name: 'Newton', age: 20})

CREATE (u2:User {id: 2, name: 'Richard', last\_name: 'Feynman', age: 30})

CREATE (u3:User {id: 3, name: 'Chandrasekhara', last\_name: 'Raman', age: 40})

CREATE (u4:User {id: 4, name: 'Srinivasa', last\_name: 'Ramanujam', age: 25})

CREATE (u5:User {id: 5, name: 'Galileo', last\_name: 'Galilei', age: 78})

CREATE (u6:User {id: 6, name: 'Marie', last\_name: 'Curie', age: 35})

CREATE (u7:User {id: 7, name: 'Albert', last\_name: 'Einstein', age: 100})

CREATE (u8:User {id: 8, name: 'Oviyasri', last\_name: 'Aryabhata', age: 50})

CREATE (u9:User {id: 9, name: 'Rabindranath', last\_name: 'Tagore', age: 90})

CREATE (u10:User {id: 10, name: 'Paul', last\_name: 'Erdos', age: 75})

// Create Group Nodes

CREATE (g11:Group {id: 101, name: 'Isotope Enthusiasts'})

CREATE (g12:Group {id: 102, name: 'Food Lovers'})

CREATE (g13:Group {id: 103, name: 'Tech Innovators'})

// Create Post Nodes

CREATE (p14:Post {id: 201, title: 'Exploring Radium'})

CREATE (p15:Post {id: 202, title: 'Best Pizza Places'})

CREATE (p16:Post {id: 203, title: 'AI Breakthroughs'})

// Create Comment Nodes

CREATE (c17:Comment {id: 301, text: 'Awesome photos!'})

CREATE (c18:Comment {id: 302, text: 'I want to visit the moon.'})

CREATE (c19:Comment {id: 303, text: 'That pizza looks delicious!'})

// Create Relationships

CREATE (u1)-[:CREATED]->(p14)

CREATE (u2)-[:CREATED]->(p15)

CREATE (u3)-[:CREATED]->(p16)

CREATE (u1)-[:COMMENTED]->(p15)

CREATE (u2)-[:COMMENTED]->(p14)

CREATE (u3)-[:COMMENTED]->(p14)

CREATE (u4)-[:LIKED]->(p14)

CREATE (u4)-[:LIKED]->(c18)

CREATE (u5)-[:LIKED]->(c17)

CREATE (u5)-[:LIKED]->(c19)

CREATE (u6)-[:CREATED]->(g11)

CREATE (u7)-[:CREATED]->(g12)

CREATE (u8)-[:CREATED]->(g13)

CREATE (u1)-[:MEMBER\_OF]->(g11)

CREATE (u2)-[:MEMBER\_OF]->(g12)

CREATE (u3)-[:MEMBER\_OF]->(g13)

CREATE (u10)-[:MEMBER\_OF]->(g11)

CREATE (u9)-[:MEMBER\_OF]->(g12)

A diagram of a network

Description automatically generated

**Finding all posts created by Isaac and the list of people who liked it:**

**Query:**

MATCH (user:User {name: 'Isaac'})-[:CREATED]->(post:Post)

MATCH (post)<-[:LIKED]-(liker:User)

RETURN post, COLLECT(liker) AS liked\_by\_users

**Result:**

|  |  |
| --- | --- |
| **post** | **liked\_by\_users** |
| **(:Post {id: 201,title: "Exploring Radium"})** | [(:User {name: "Srinivasa",last\_name: "Ramanujam",id: 4,age: 25})] |

A close-up of a circle

Description automatically generated

1. **Write a query to identify posts that received more than two comments and display the number of comments for each of these posts.**

**Solution:**

**Query:**

MATCH (post:Post)

OPTIONAL MATCH (post)<-[:COMMENTED]-(comment:Comment)

WITH post, COUNT(comment) AS commentCount

WHERE commentCount > 2

RETURN post, commentCount

**Result:**

A white background with black and white clouds

Description automatically generated with medium confidence

1. **Calculate the user engagement score for each user, where the score is the sum of the likes received on their posts and comments. List users ranked by their engagement score.**

**Query:**

MATCH (user:User)

OPTIONAL MATCH (user)-[:CREATED]->(post:Post)

OPTIONAL MATCH (post)<-[:LIKED]-(liker:User)

OPTIONAL MATCH (post)<-[:COMMENTED]-(comment:Comment)

WITH user, COLLECT(DISTINCT post) AS posts, COALESCE(SUM(DISTINCT CASE WHEN liker IS NOT NULL THEN 1 ELSE 0 END), 0) AS totalLikes, COALESCE(SUM(DISTINCT CASE WHEN comment IS NOT NULL THEN 1 ELSE 0 END), 0) AS totalComments

WITH user, totalLikes + totalComments AS engagementScore

RETURN user, engagementScore

ORDER BY engagementScore DESC

**Result:**

|  |  |
| --- | --- |
| **user** | **engagementScore** |
| **(:User {name: "Isaac",last\_name: "Newton",id: 1,age: 20})** | 1 |
| **(:User {name: "Richard",last\_name: "Feynman",id: 2,age: 30})** | 0 |
| **(:User {name: "Chandrasekhara",last\_name: "Raman",id: 3,age: 40})** | 0 |
| **(:User {name: "Srinivasa",last\_name: "Ramanujam",id: 4,age: 25})** | 0 |
| **(:User {name: "Galileo",last\_name: "Galilei",id: 5,age: 78})** | 0 |
| **(:User {name: "Marie",last\_name: "Curie",id: 6,age: 35})** | 0 |
| **(:User {name: "Albert",last\_name: "Einstein",id: 7,age: 100})** | 0 |
| **(:User {name: "Oviyasri",last\_name: "Aryabhata",id: 8,age: 50})** | 0 |
| **(:User {name: "Rabindranath",last\_name: "Tagore",id: 9,age: 90})** | 0 |
| **(:User {name: "Paul",last\_name: "Erdos",id: 10,age: 75})** | 0 |

1. **Calculate the average number of comments on posts created by each user. List users with their average comment counts.**

**Query:**

MATCH (user:User)-[:CREATED]->(p:Post)<-[:COMMENTED]-(commenter:User)

WITH user, COUNT(DISTINCT p) AS total\_posts, COUNT(commenter) AS total\_comments, COUNT(commenter)/COUNT(DISTINCT p) AS avg\_comments

RETURN user, total\_posts, total\_comments, avg\_comments

ORDER BY total\_comments DESC

**Result:**

|  |  |  |  |
| --- | --- | --- | --- |
| **user** | **total\_posts** | **total\_comments** | **avg\_comments** |
| **(:User {name: "Isaac",last\_name: "Newton",id: 1,age: 20})** | 1 | 2 | 2 |
| **(:User {name: "Richard",last\_name: "Feynman",id: 2,age: 30})** | 1 | 1 | 1 |

1. **Propose a real-world use case where this social network graph database could be applied effectively, considering user engagement, content creation, and interactions. Describe the scenario and how the database would be used.**

A graph database would ingest an influencer's Instagram data to model followers, posts, and their interactions. By analyzing these nodes and relationships, the database can calculate engagement scores, gain audience insights, recommend content ideas, identify collaborations, and optimize posting strategy and growth. Specifically, the influencer can use the graph analytics to segment their audience, determine top-performing content, target engagement campaigns, monitor metrics, and foster their community. Adherence to data privacy and Instagram's terms would be critical. In summary, the connectivity and relationships modeled in the graph database would empower the influencer to boost engagement, refine content creation, maximize reach, and build a loyal following through data-driven decisions and community building.