Islamic University of Technology (IUT)

Report on Lab 9

Submitted By

Shanta Maria (ID:200042172)

CSE 4410 Database Management Systems II Lab

Submitted To

Dr. Abu Raihan Mostofa Kamal

Professor, Department of CSE

Zannatun Naim Srsity

Lecturer, Department of CSE

Md. Rafid Haque

Lecturer, Department of CSE

April 11, 2023

Question:

- Let's assume you want to build a recommendation engine for your online bookshop. Your online bookshop sells a variety of books and you want to improve the customer's buying experience by recommending books that they are likely to purchase by analyzing shared personal details and monitoring which books go together in one's purchase list.

  This type of scenario generally includes four types of nodes-

  - Customer: Contains information about customers such as customer ID, name, phone_no, and demographic information like age, gender, country etc.
  - Genre: It helps to filter different books according to the genre.
  - Author: Contains information about authors such as name, country, date_of_birth etc.
  - Book: Contains information like title, published_year, language, page_count, price etc.

  And the relations are the following-

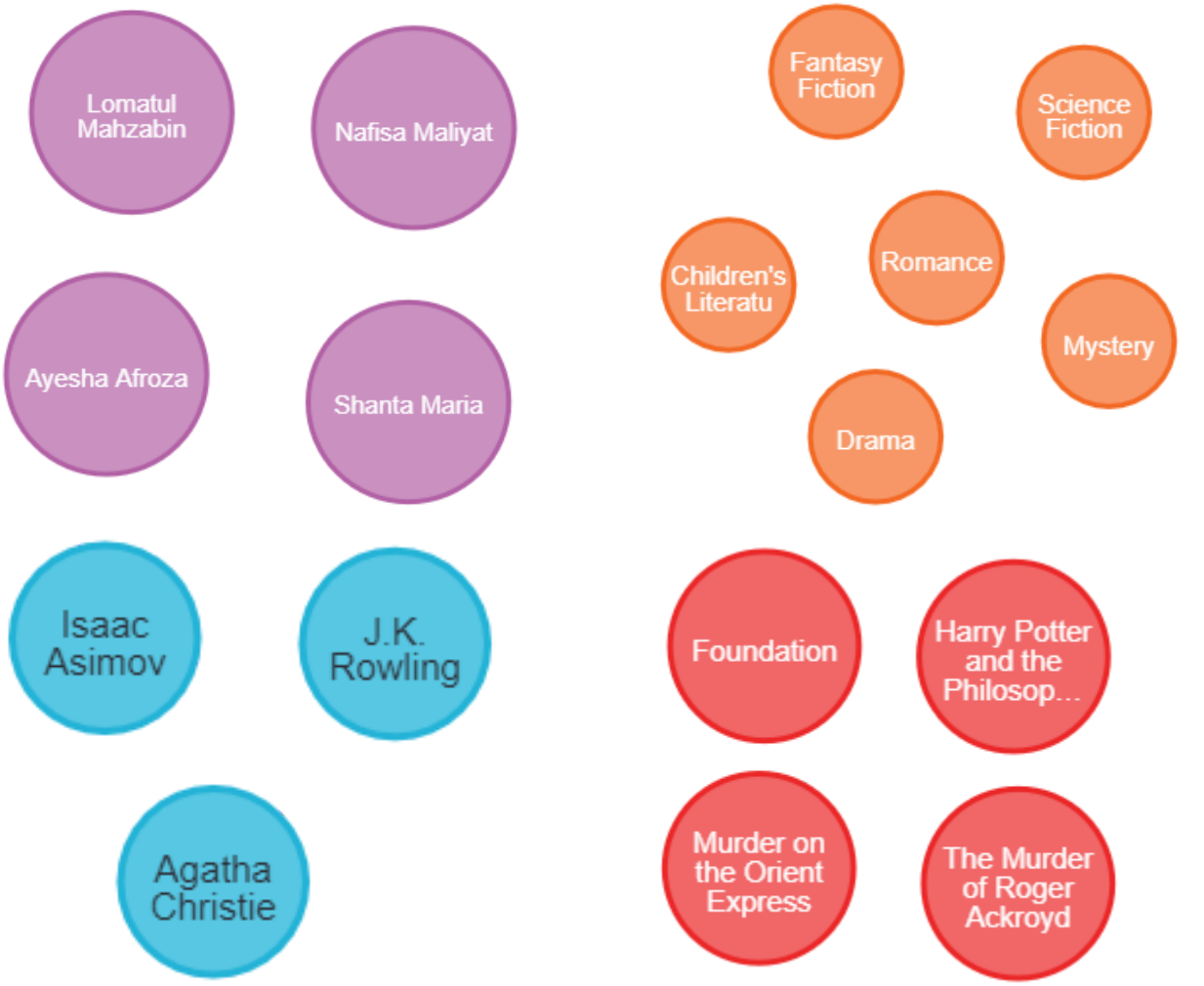
  - Customers purchase or rate books. The purchase information also includes purchasing_date and amount.
  - Customer can also rate authors(this is different from rating a book).
  - Books can be of different genres.
  - Books can have multiple volumes.
  - Authors write books. And this includes the writing_year.

Now, your task is to:

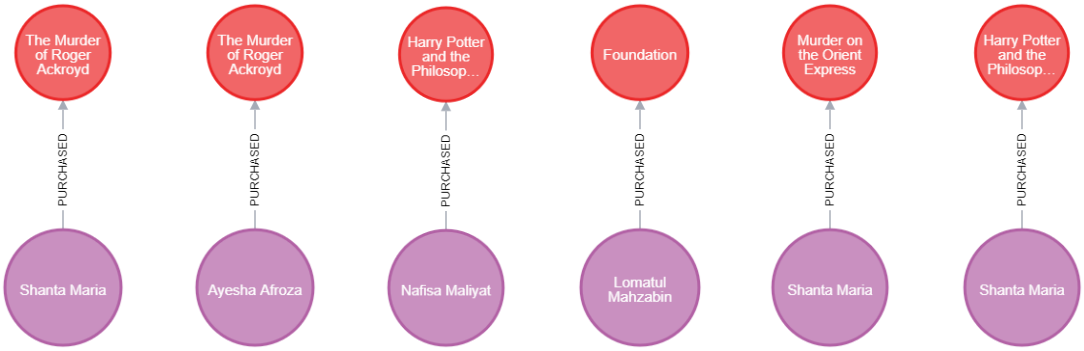
1. Create necessary nodes and relations with properties.
2. Cypher Query
   (a) Find the total revenue generated by each book.
   (b) Find the average rating for each genre.
   (c) Find books purchased by a customer 'N' within a specific time range.
   (d) Find the customer who buys the maximum number of books.
   (e) Find the best-seller books by the number of purchases.
   (f) Find the customer who bought or rated a certain book. for example 'A'
   (g) Find the customer who bought the books of a certain author. for example 'X'
   (h) Find books frequently purchased together.

Task 01:

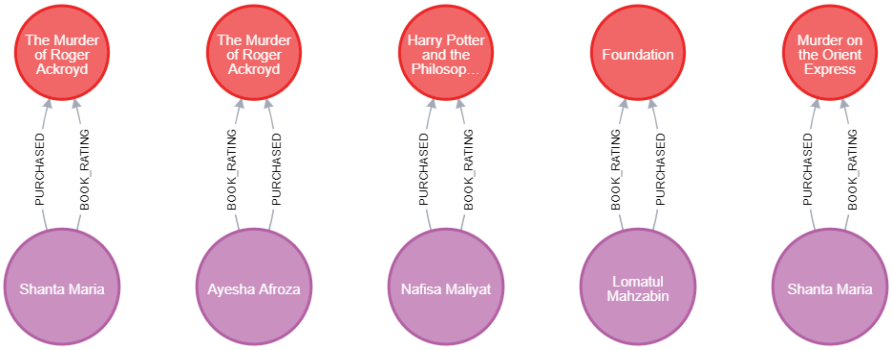
```
//All Nodes

--------------------------Customer nodes--------------------------
CREATE (a:Customer {customer_id: '172', name: 'Shanta Maria', phone_no: '1234567890', age: 24, gender: 'Female', country: 'UK'}),
       (b:Customer {customer_id: '133', name: 'Nafisa Maliyat', phone_no: '0987654321', age: 28, gender: 'Female', country: 'Canada'}),
       (c:Customer {customer_id: '113', name: 'Lomatul Mahzabin', phone_no: '5551234567', age: 35, gender: 'Female', country: 'Australia'}),
       (d:Customer {customer_id: '106', name: 'Ayesha Afroza', phone_no: '274017292370', age: 30, gender: 'Male', country: 'USA'})
RETURN a, b, c, d

--------------------------Genre nodes--------------------------
CREATE (a:Genre {name: 'Mystery'}),
       (b:Genre {name: 'Romance'}),
       (c:Genre {name: 'Science Fiction'}),
       (d: Genre {name: 'Fantasy Fiction'}),
       (e: Genre {name: 'Children\'s Literature'}),
       (f: Genre {name: 'Drama'})
RETURN a, b, c, d, e, f

--------------------------Author nodes--------------------------
CREATE (a:Author {name: 'Agatha Christie', country: 'UK', date_of_birth: date('1890-09-15')}),
       (b:Author {name: 'J.K. Rowling', country: 'UK', date_of_birth: date('1965-07-31')}),
       (c:Author {name: 'Isaac Asimov', country: 'Russia', date_of_birth: date('1920-01-02')})
RETURN a, b, c

--------------------------Book nodes--------------------------
CREATE (a:Book {title: 'The Murder of Roger Ackroyd', published_year: 1926, language: 'English', page_count: 296, price: 10.99}),
       (b:Book {title: 'Harry Potter and the Philosopher\'s Stone', published_year: 1997, language: 'English', page_count: 223, price: 12.99}),
       (c:Book {title: 'Foundation', published_year: 1951, language: 'English', page_count: 255, price: 9.99}),
       (d:Book {title: 'Murder on the Orient Express', published_year: 1934, language: 'English', page_count: 347, price: 11.99})
RETURN a, b, c, d
```



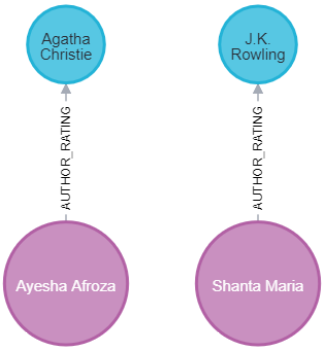
```
//All Relationships

--------------Relationship between customer and book nodes (purchasing)--------------
MATCH (a:Customer {customer_id: '172'}), (b:Book {title: 'The Murder of Roger Ackroyd'})
CREATE (a)-[r:PURCHASED {purchasing_date: date('2022-10-15'), amount: 12.15}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '106'}), (b:Book {title: 'The Murder of Roger Ackroyd'})
CREATE (a)-[r:PURCHASED {purchasing_date: date('2022-08-21'), amount: 12.15}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '133'}), (b:Book {title: 'Harry Potter and the Philosopher\'s Stone'})
CREATE (a)-[r:PURCHASED {purchasing_date: date('2023-01-01'), amount: 14.00}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '113'}), (b:Book {title: 'Foundation'})
CREATE (a)-[:PURCHASED {purchasing_date: date('2022-09-30'), amount: 10.23}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '172'}), (b:Book {title: 'Murder on the Orient Express'})
CREATE (a)-[:PURCHASED {purchasing_date: date('2022-11-25'), amount: 12.55}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '172'}), (b:Book {title: 'Harry Potter and the Philosopher\'s Stone'})
CREATE (a)-[:PURCHASED {purchasing_date: date('2023-03-24'), amount: 14.00}]->(b)
RETURN a, b
```




```
--------------Relationship between customer and book nodes (rating)--------------
MATCH (a:Customer {customer_id: '172'}), (b:Book {title: 'The Murder of Roger Ackroyd'})
CREATE (a)-[r:BOOK_RATING {book_rating: 7.3}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '106'}), (b:Book {title: 'The Murder of Roger Ackroyd'})
CREATE (a)-[r:BOOK_RATING {book_rating: 8.0}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '133'}), (b:Book {title: 'Harry Potter and the Philosopher\'s Stone'})
CREATE (a)-[r:BOOK_RATING {book_rating: 9.4}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '113'}), (b:Book {title: 'Foundation'})
CREATE (a)-[:BOOK_RATING {book_rating: 7.4}]->(b)
RETURN a, b

MATCH (a:Customer {customer_id: '172'}), (b:Book {title: 'Murder on the Orient Express'})
CREATE (a)-[:BOOK_RATING {book_rating: 9.2}]->(b)
RETURN a, b
```

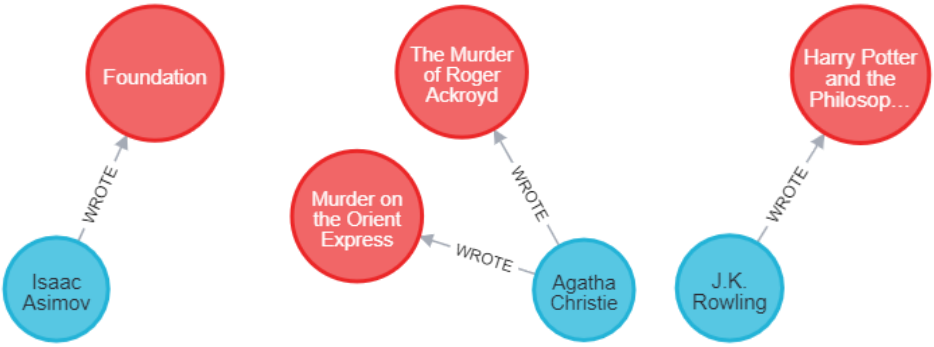



```
--------------Relationship between customer and author nodes--------------
MATCH (c:Customer {customer_id: '106'}), (a:Author {name: 'Agatha Christie'})
CREATE (c)-[:AUTHOR_RATING {author_rating: 4}]->(a)
RETURN c, a

MATCH (c:Customer {customer_id: '172'}), (a:Author {name: 'J.K. Rowling'})
CREATE (c)-[:AUTHOR_RATING {author_rating: 7.5}]->(a)
RETURN c, a
```

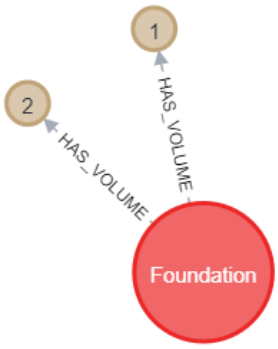



```
--------------Relationship between book and genre nodes--------------
MATCH (b:Book {title: 'Harry Potter and the Philosopher\'s Stone'})
MATCH (g1:Genre {name: 'Fantasy Fiction'}), (g2:Genre {name: 'Children\'s Literature'})
CREATE (b)-[:IN_GENRE]->(g1), (b)-[:IN_GENRE]->(g2)
RETURN b, g1, g2

MATCH (b:Book {title: 'Murder on the Orient Express'})
MATCH (g1:Genre {name: 'Mystery'})
CREATE (b)-[:IN_GENRE]->(g1)
RETURN b, g1

MATCH (b:Book {title: 'The Murder of Roger Ackroyd'})
MATCH (g1:Genre {name: 'Mystery'})
CREATE (b)-[:IN_GENRE]->(g1)
RETURN b, g1

MATCH (b:Book {title: 'Foundation'})
MATCH (g1:Genre {name: 'Science Fiction'}), (g2:Genre {name: 'Drama'})
CREATE (b)-[:IN_GENRE]->(g1), (b)-[:IN_GENRE]->(g2)
RETURN b, g1, g2
```


```
112  --------------Relationship between book and author nodes--------------
113  MATCH (a:Author {name: 'Agatha Christie'})
114  MATCH (b1:Book {title: 'The Murder of Roger Ackroyd'}), (b2:Book {title: 'Murder on the Orient Express'})
115  CREATE (a)-[:WROTE {writing_year: 1926}]->(b1), (a)-[:WROTE {writing_year: 1934}]->(b2)
116  RETURN a, b1, b2
117
118  MATCH (a:Author {name: 'J.K. Rowling'})
119  MATCH (b1:Book {title: 'Harry Potter and the Philosopher\'s Stone'})
120  CREATE (a)-[:WROTE {writing_year: 1997}]->(b1)
121  RETURN a, b1
122
123  MATCH (a:Author {name: 'Isaac Asimov'})
124  MATCH (b1:Book {title: 'Foundation'})
125  CREATE (a)-[:WROTE {writing_year: 1951}]->(b1)
126  RETURN a, b1
```




```
128  --------------Relationship between book and author nodes--------------
129  MATCH (b:Book {title: 'Foundation'})
130  CREATE (v1:Volume {volume_number: 1})
131  CREATE (v2:Volume {volume_number: 2})
132  CREATE (b)-[:HAS_VOLUME]->(v1)
133  CREATE (b)-[:HAS_VOLUME]->(v2)
134  RETURN b, v1, v2
```



Queries:

```
------------(a)------------
MATCH (c:Customer)-[p:PURCHASED]->(b:Book)
RETURN b.title AS Book, SUM(p.amount) AS Revenue
ORDER BY Revenue DESC
```

Book	Revenue
"Harry Potter and the Philosopher's Stone"	28.0
"The Murder of Roger Ackroyd"	24.3
"Murder on the Orient Express"	12.55
"Foundation"	10.23

```
------------(b)------------
//WITH allows us to pass results from one query to the next

MATCH (g:Genre)<-[:IN_GENRE]-(b:Book)<-[p:BOOK_RATING]-(c:Customer)
WITH g, AVG(p.book_rating) AS avg_rating
RETURN g.name AS genre, avg_rating
```

genre	avg_rating
"Fantasy Fiction"	9.4
"Children's Literature"	9.4
"Mystery"	8.166666666666666
"Science Fiction"	7.4
"Drama"	7.4

```
------------(c)------------
MATCH (c:Customer {name: 'Shanta Maria'})-[r:PURCHASED]->(b:Book)
WHERE r.purchasing_date >= date('2023-01-01') AND r.purchasing_date <= date('2023-12-31')
RETURN c.name AS Customer, b.title AS Book
```

Customer	Book
"Shanta Maria"	"Harry Potter and the Philosopher's Stone"

```
------------(d)------------
MATCH (c:Customer)-[p:PURCHASED]->(b:Book)
WITH c, COUNT(p) AS Purchases
RETURN c.name AS Customer, Purchases
ORDER BY Purchases DESC LIMIT 1
```

Customer	Purchases
"Shanta Maria"	3

```
------------(e)------------
MATCH (c:Customer)-[p:PURCHASED]->(b:Book)
RETURN b.title AS Book, COUNT(p) AS Purchases
ORDER BY Purchases DESC LIMIT 1
```

Book	Purchases
"The Murder of Roger Ackroyd"	2

```
------------(f)------------
//Here frequency eliminates duplicates since a single customer could have rated and purchased the same book

MATCH (c:Customer)-[:PURCHASED|:BOOK_RATING]->(b:Book {title: 'The Murder of Roger Ackroyd'})
RETURN c.name AS Customer, b.title AS Book, COUNT(*) AS Frequency
ORDER BY Frequency DESC
```

Customer	Book	Frequency
"Ayesha Afroza"	"The Murder of Roger Ackroyd"	2
"Shanta Maria"	"The Murder of Roger Ackroyd"	2

```
------------(g)------------
MATCH (a:Author {name: 'Agatha Christie'})-[:WROTE]->(b:Book)<-[:PURCHASED]-(c:Customer)
RETURN c.name AS Customer, COUNT(b) AS Num_books_bought
ORDER BY Num_books_bought DESC
LIMIT 1
```

Customer	Num_books_bought
"Shanta Maria"	2

```
------------(h)------------
MATCH (b1:Book)<-[p1:PURCHASED]-(c:Customer)-[p2:PURCHASED]->(b2:Book)
WHERE b1 <> b2
RETURN b1.title, b2.title, COUNT(DISTINCT c) AS frequency
ORDER BY frequency DESC


//Matches all pairs of Book nodes that are purchased by the same Customer node using the PURCHASED relationship
//and then counts the number of distinct Customer nodes that have purchased each pair of books
//exclude pairs of books where both nodes represent the same book using the <> operator
//sort the results in descending order based on the frequency of purchase.
```

b1.title	b2.title	frequency
"Harry Potter and the Philosopher's Stone"	"The Murder of Roger Ackroyd"	1
"Murder on the Orient Express"	"The Murder of Roger Ackroyd"	1
"Harry Potter and the Philosopher's Stone"	"Murder on the Orient Express"	1
"The Murder of Roger Ackroyd"	"Murder on the Orient Express"	1
"Murder on the Orient Express"	"Harry Potter and the Philosopher's Stone"	1
"The Murder of Roger Ackroyd"	"Harry Potter and the Philosopher's Stone"	1