

Name: Cem Emir Senyurt

Student ID: 45064076

Date: 05/28/2023

Q1:

A:

i.

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_users.csv' AS u
MERGE (user:User {user_id: u.user_id})
ON CREATE SET user += u;
```

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_sellers.csv' AS s
MERGE (seller:User {user_id: s.user_id})
ON MATCH SET seller += s
SET seller:Seller;
```

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_buyers.csv' AS b
MERGE (buyer:User {user_id: b.user_id})
ON MATCH SET buyer += b
SET buyer:Buyer;
```

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_items.csv' AS i
MERGE (item:Item {item_id: i.item_id})
ON CREATE SET item += i, item.price = toFloat(i.price);
```

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_goods.csv' AS g

MERGE (good:Item {item_id: g.item_id})

ON MATCH SET good += g

SET good:Good;
```

```
LOAD CSV WITH HEADERS
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_services.csv' AS serv

MERGE (service:Item {item_id: serv.item_id})

ON MATCH SET service += serv

SET service:Service;
```

ii.

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

CREATE INDEX item_id FOR (i:Item) ON (i.item_id);

CALL db.awaitIndexes();
```

iii.

```
LOAD CSV WITH HEADERS FROM
'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_sells.csv' AS s

MATCH (seller:Seller {user_id: s.seller_id})

MATCH (item:Item {item_id: s.item_id})

MERGE (seller)-[sell:Sells]->(item)

ON CREATE SET sell.list_date = s.list_date;
```

```
LOAD CSV WITH HEADERS FROM
'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_buys.csv' AS b

MATCH (buyer:Buyer {user_id: b.buyer_id})

MATCH (item:Item {item_id: b.item_id})

MERGE (buyer)-[buy:Buys]->(item)
```

```
ON CREATE SET buyer.purchase_date = b.purchase_date;
```

LOAD CSV WITH HEADERS

```
FROM 'file:///Users/cememirsenyurt/Desktop/Assignment5/data/Interchange_data_ratings.csv' AS r
```

```
MATCH (buyer:Buyer {user_id: r.buyer_id})
```

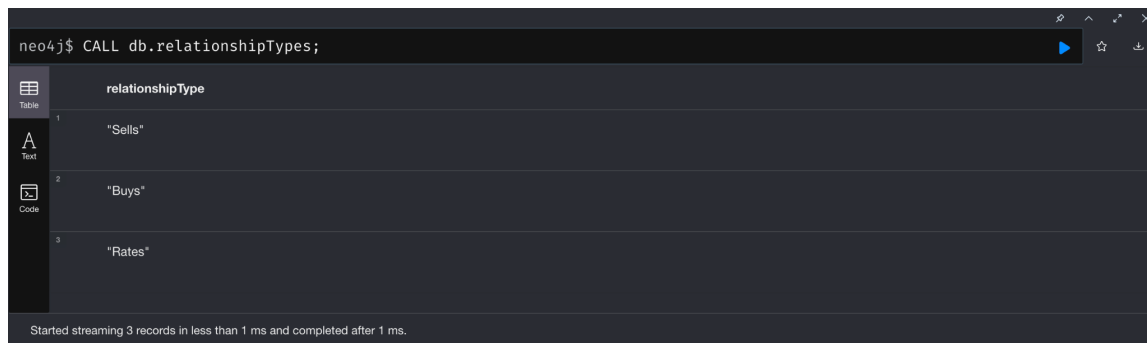
```
MATCH (seller:Seller {user_id: r.seller_id})
```

```
MERGE (buyer)-[rates:Rates]->(seller)
```

```
ON CREATE SET rates.delivery = toInteger(r.delivery), rates.pricing = toInteger(r.pricing), rates.quality = toInteger(r.quality), rates.rating_date = r.rating_date;
```

B:

I

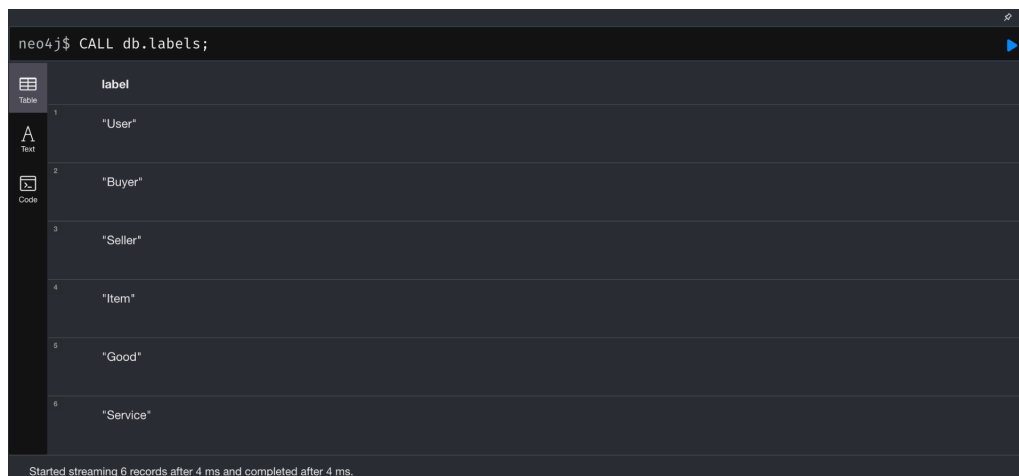


neo4j\$ CALL db.relationshipTypes;

	relationshipType
1	"Sells"
2	"Buys"
3	"Rates"

Started streaming 3 records in less than 1 ms and completed after 1 ms.

ii



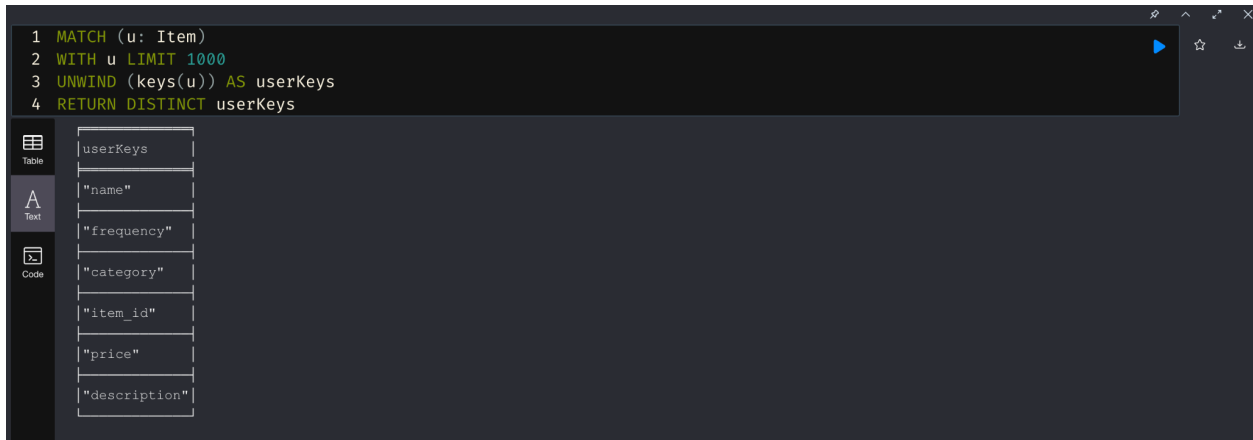
neo4j\$ CALL db.labels;

	label
1	"User"
2	"Buyer"
3	"Seller"
4	"Item"
5	"Good"
6	"Service"

Started streaming 6 records after 4 ms and completed after 4 ms.

Q2:

- i) Users have a Buys, Sells, and Rates relationship.
- ii) The labels of the other nodes that these nodes are connected to are: Seller, Good, Item, and Service.
- iii) The directions of each relationship are from the User to the other node.

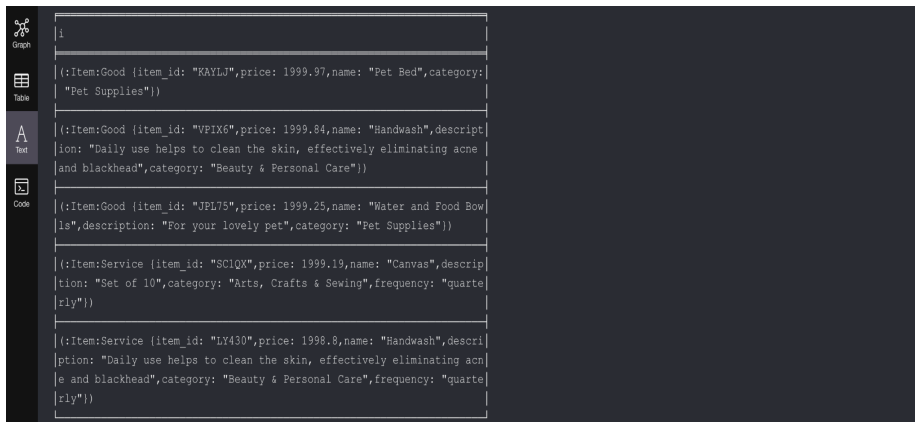


Q3 A:

Query:



Results (screenshot below):



Q3 B:

Query:

```

MATCH (u:User)-[b:Buys]->(item:Item)

WHERE u.user_id = 'JNP1L'

RETURN item, b.purchase_date ORDER BY item.item_id ASC;

```

Results (screenshot below):

item	b.purchase_date
(:Item:Service {item_id: "3C5S5",price: 514.47,name: "Barbie",description: "Unbox the package to find a soft, plush, sparkly doll!",category: "Toys & Games",frequency: "once"})	"2022-10-28"
(:Item:Service {item_id: "H78YV",price: 1610.32,name: "Chair",description: "for everyday use",category: "Others",frequency: "monthly"})	"2022-10-29"
(:Item:Good {item_id: "ICCZ6",price: 856.59,name: "Notebook",description: "for everyday use",category: "Others"})	"2022-03-06"
(:Item:Service {item_id: "N1OKM",price: 412.25,name: "Bath Soap",category: "Beauty & Personal Care",frequency: "weekly"})	"2022-05-24"

Q3 C:

Query:

```

MATCH (u:User)

WHERE NOT (u) - [:Buys] ->(:Item) AND NOT (u) - [:Sells] ->(:Item)

RETURN COUNT(u) as num_of_users;

```

Results (screenshot below):

Table	num_of_users
Text	798

Q3 D:

Query:

```

MATCH(s:Seller)-[:Sells]->(i:Item)<-[:Buys]-(b:Buyer)

WITH s, COUNT(i) as item_count

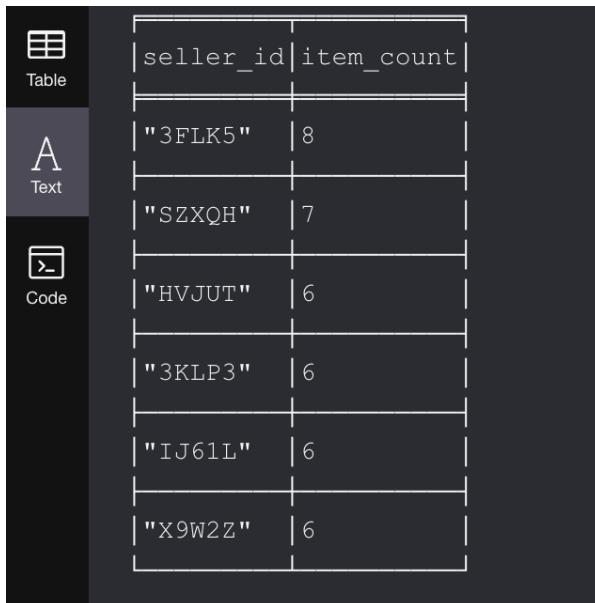
WHERE item_count > 5

RETURN s.user_id as seller_id, item_count

ORDER BY item_count DESC;

```

Results (screenshot below):



seller_id	item_count
"3FLK5"	8
"SZXQH"	7
"HVJUT"	6
"3KLP3"	6
"IJ61L"	6
"X9W2Z"	6

Q3 E:

Query:

```

MATCH (b:Buyer)-[r:Rates{quality : 5}]->(u:Seller)-[s:Sells]->(i:Item {category: "Electronics"})

RETURN DISTINCT u.first_name, u.last_name

ORDER BY u.first_name, u.last_name

LIMIT 5;

```

Results (screenshot below):

Table	u.first_name	u.last_name
Text	"Adrian"	"Blackwell"
Code	"Adrian"	"Frey"
	"Amanda"	"Bentley"
	"Amanda"	"Sanchez"
	"Amber"	"Ward"

Q3 F:

Query:

```
MATCH (u:Buyer)-[b:Buys]->(i:Item), (s:Seller)-[r:Rates]-(u:Buyer)
WITH u.user_id as user_id, COUNT(i) as item_count, COUNT(r) as rate_count
WHERE item_count > 2 AND rate_count > 2
RETURN user_id
ORDER BY user_id
LIMIT 5;
```

Results (screenshot below):

user_id
"010JB"
"04AN7"
"05K3U"
"08NIU"
"0HQSK"

Q3 G:

Query:

```
MATCH (u1: User)-[:Rates]->(s: Seller)<-[:Rates]-(u2: User)
WHERE elementId(u1) < elementId(u2)
RETURN u1.last_name as buyer1_last_name, u2.last_name as buyer2_last_name, s.first_name as
seller_first_name,s.last_name as seller_last_name, s.user_id as seller_user_id
ORDER BY s.user_id
LIMIT 10;
```

Results (screenshot below):

buyer1_last_name	buyer2_last_name	seller_first_name	seller_last_name	seller_user_id
"Wood"	"Chavez"	"Teresa"	"Kelly"	"05M7F"
"Allison"	"Chavez"	"Teresa"	"Kelly"	"05M7F"
"Allison"	"Wood"	"Teresa"	"Kelly"	"05M7F"
"Chavez"	"Bowers"	"Teresa"	"Kelly"	"05M7F"
"Wood"	"Bowers"	"Teresa"	"Kelly"	"05M7F"
"Allison"	"Bowers"	"Teresa"	"Kelly"	"05M7F"
"Barrett"	"Moss"	"Juan"	"Clayton"	"07800"
"Proctor"	"Snyder"	"Tracy"	"Zamora"	"07RB8"
"Holmes"	"Snyder"	"Tracy"	"Zamora"	"07RB8"
"Cannon"	"Snyder"	"Tracy"	"Zamora"	"07RB8"

Q3 H:



Query:

```
MATCH (u1: Buyer)-[:Rates]->(s: Seller)-[:Sells]->(i: Item)

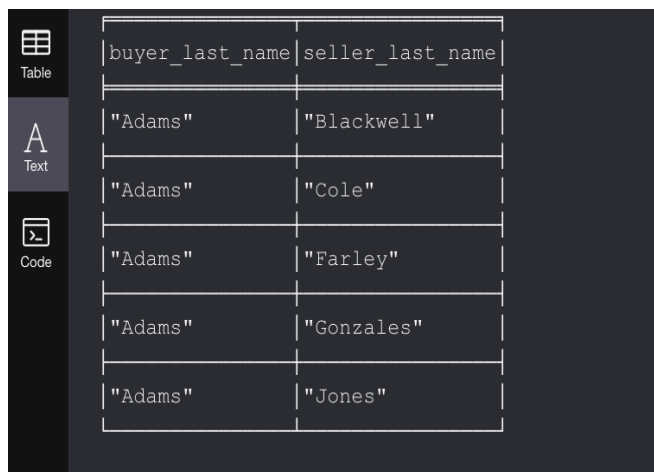
WHERE i:Good OR i:Service

RETURN DISTINCT u1.last_name as buyer_last_name, s.last_name as seller_last_name

ORDER BY buyer_last_name, seller_last_name ASC

LIMIT 5;
```

Results (screenshot below):



buyer_last_name	seller_last_name
"Adams"	"Blackwell"
"Adams"	"Cole"
"Adams"	"Farley"
"Adams"	"Gonzales"
"Adams"	"Jones"

Q3 I:

i

Query:

```
MATCH (start:Item {item_id: "P8WKJ"})

MATCH (end:Item)

WHERE start <> end

RETURN MIN(length(shortestPath((start)-[*]-(end)))) AS shortest_path_length;
```

Results (screenshot below):

Table	shortest_path_length
Text	2

ii

Query:

```
MATCH (start:Item {item_id: "P8WKJ"})-[*3]-(end:Item)
WHERE start <> end
RETURN DISTINCT end.item_id AS item_id
ORDER BY item_id ASC
LIMIT 5;
```

Results (screenshot below):

Table	item_id
Text	"0QHP8"
Text	"0VJPX"
Code	"1PZ5H"
Code	"2OSVX"
Code	"3UISG"