1. Use the provided Cypher script to create the graph database

- ♣ You could use any names for your project and the graph database
- ♣ Copy the ENTIRE Cypher code in the script and paste it in ne4j$ prompt

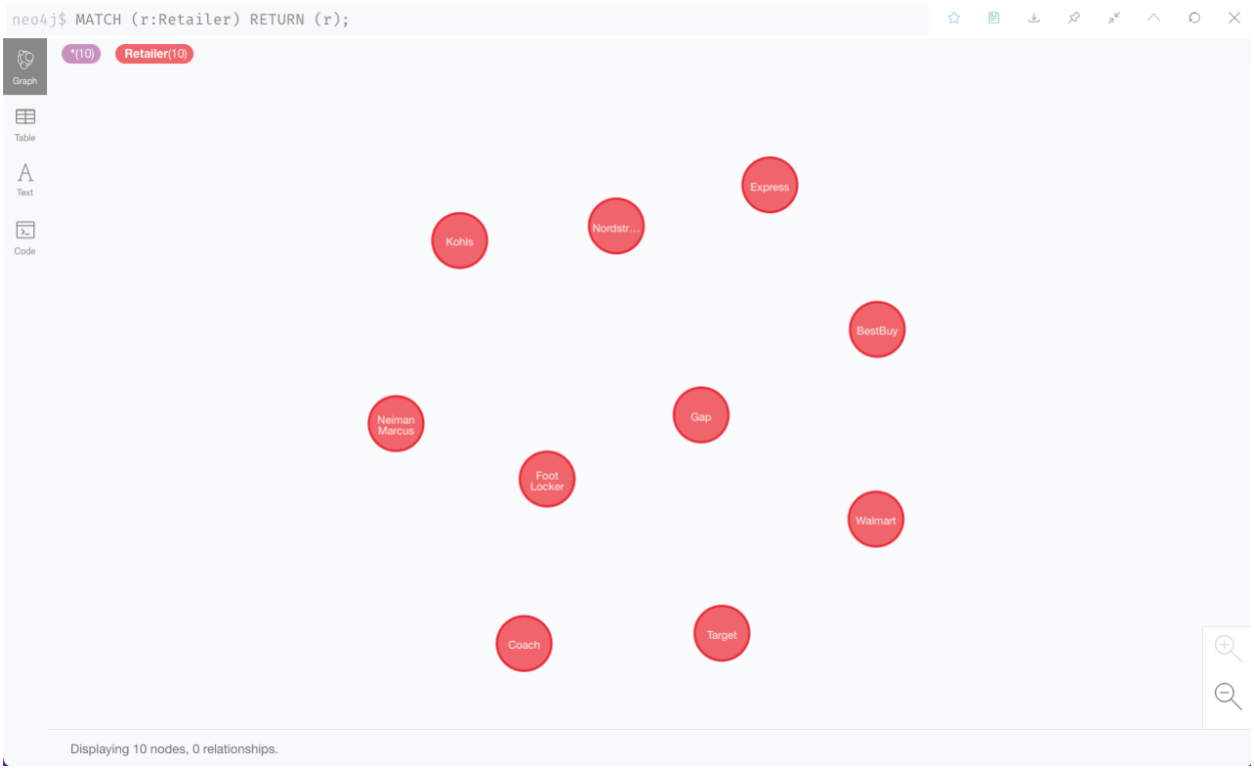
  and then click the play button on the right.

  DONE

2. Execute the following Cypher code to get the list of retailers:

   MATCH (r:Retailer)

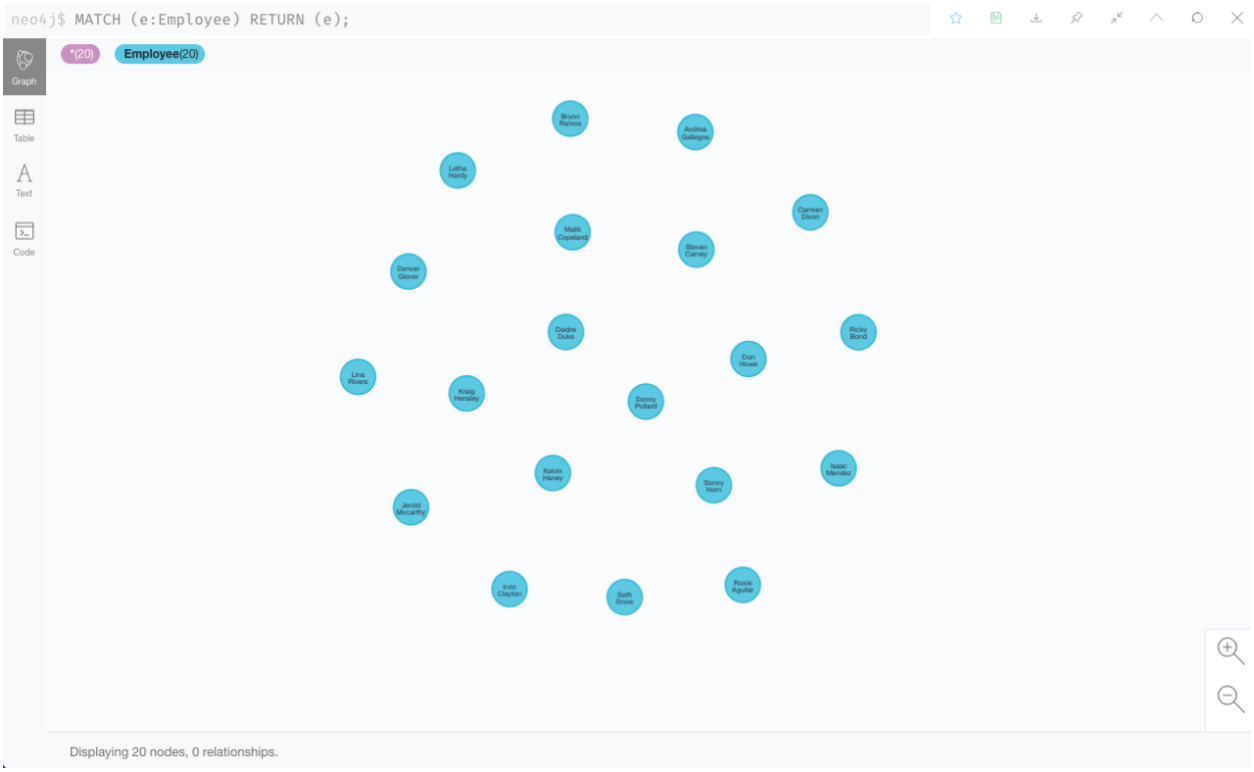
   RETURN (r);



3. Execute the following Cypher code to get the list of employees:

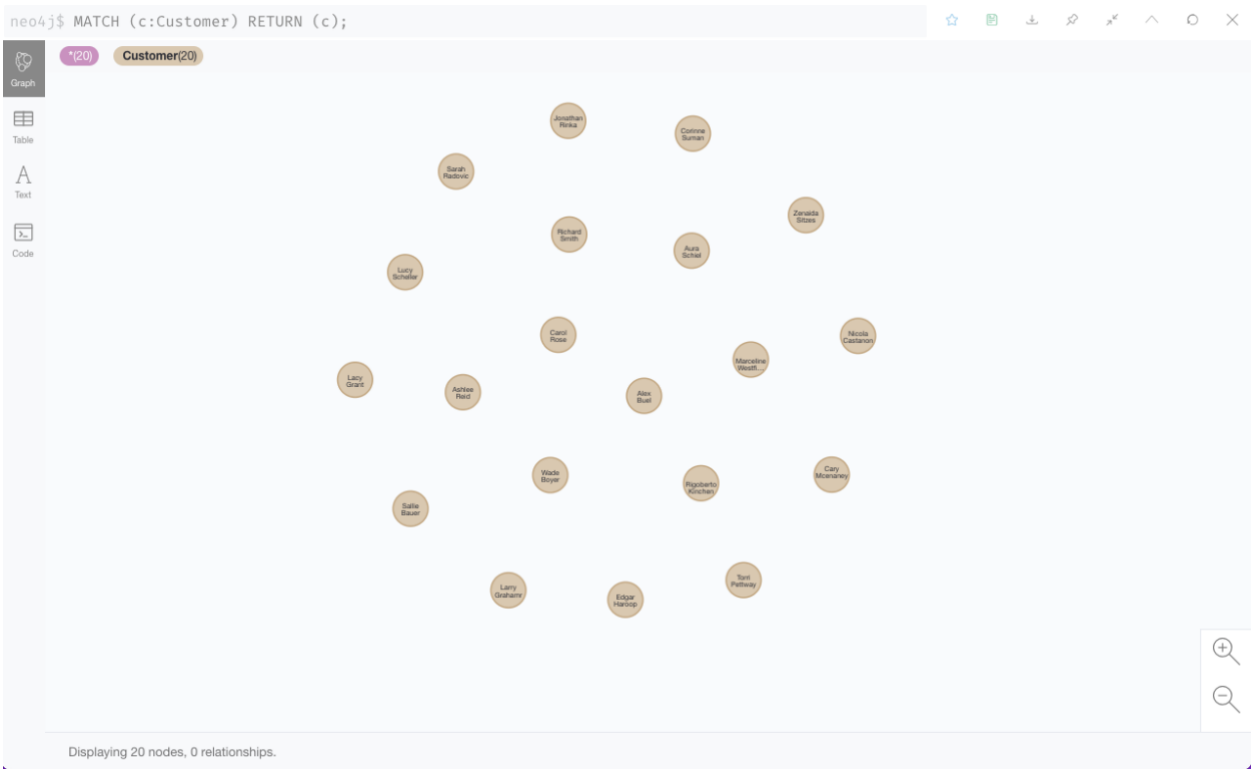
   MATCH (e:Employee)

   RETURN (e);



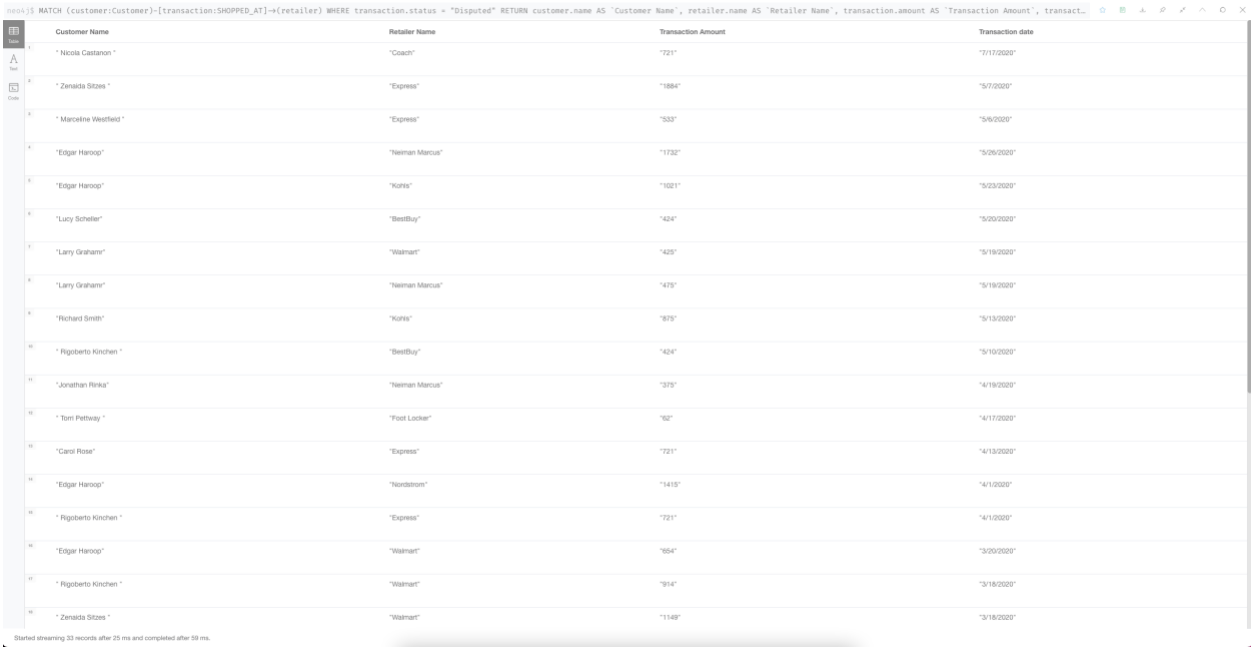
4. Execute the following Cypher code to get the list of customers:

```
MATCH (c:Customer)

RETURN (c);
```



5. Execute the following Cypher code to get the list of all Disputed transactions

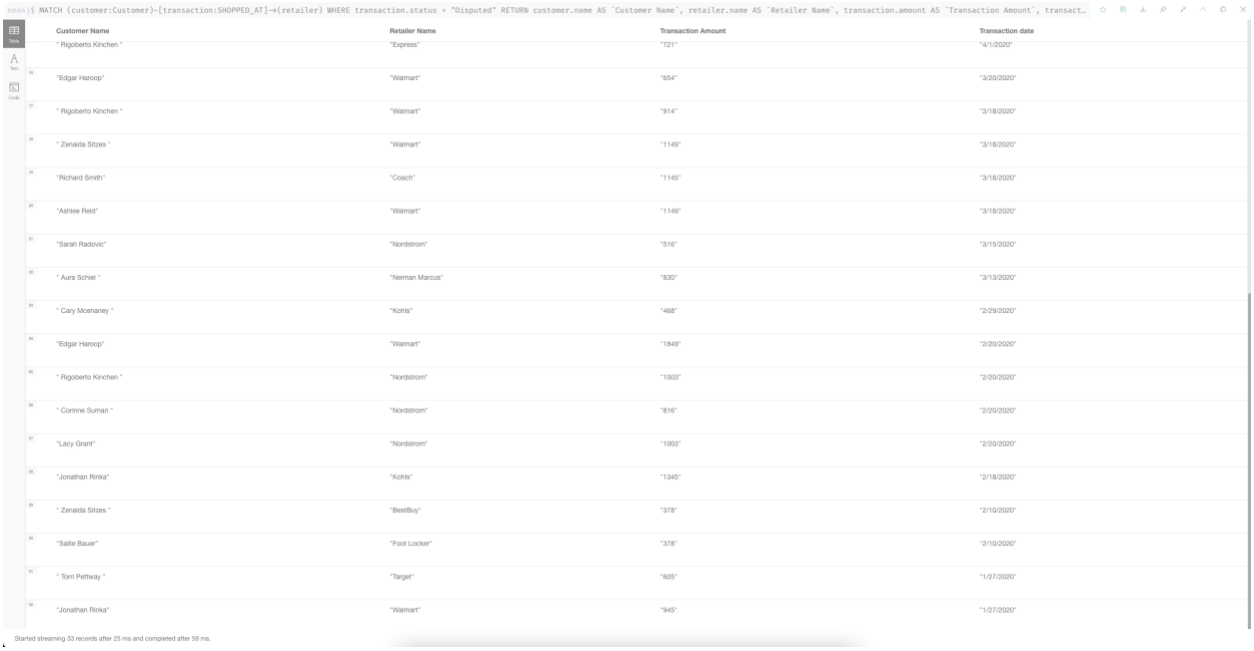
```
MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE transaction.status = "Disputed"

RETURN customer.name AS `Customer Name`, retailer.name AS `Retailer Name`, transaction.amount AS `Transaction Amount`, transaction.date AS `Transaction date`

ORDER BY `Transaction date` DESC
```



neo4j$ MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE transaction.status = "Disputed" RETURN customer.name AS `Customer Name`, retailer.name AS `Retailer Name`, transaction.amount AS `Transaction Amount`, transact…

	Customer Name	Retailer Name	Transaction Amount	Transaction date
1	" Nicola Castanon "	"Coach"	"721"	"7/17/2020"
2	" Zenaida Sitzes "	"Express"	"1884"	"5/7/2020"
3	" Marceline Westfield "	"Express"	"533"	"5/6/2020"
4	"Edgar Haroop"	"Neiman Marcus"	"1732"	"5/26/2020"
5	"Edgar Haroop"	"Kohls"	"1021"	"5/23/2020"
6	"Lucy Scheller"	"BestBuy"	"424"	"5/20/2020"
7	"Larry Grahamr"	"Walmart"	"425"	"5/19/2020"
8	"Larry Grahamr"	"Neiman Marcus"	"475"	"5/19/2020"
9	"Richard Smith"	"Kohls"	"875"	"5/13/2020"
10	" Rigoberto Kinchen "	"BestBuy"	"424"	"5/10/2020"
11	"Jonathan Rinka"	"Neiman Marcus"	"375"	"4/19/2020"
12	" Toni Pettway "	"Foot Locker"	"62"	"4/17/2020"
13	"Carol Rose"	"Express"	"721"	"4/13/2020"
14	"Edgar Haroop"	"Nordstrom"	"1415"	"4/1/2020"
15	" Rigoberto Kinchen "	"Express"	"721"	"4/1/2020"
16	"Edgar Haroop"	"Walmart"	"654"	"3/20/2020"
17	" Rigoberto Kinchen "	"Walmart"	"914"	"3/18/2020"
18	" Zenaida Sitzes "	"Walmart"	"1149"	"3/18/2020"

Started streaming 33 records after 25 ms and completed after 59 ms.



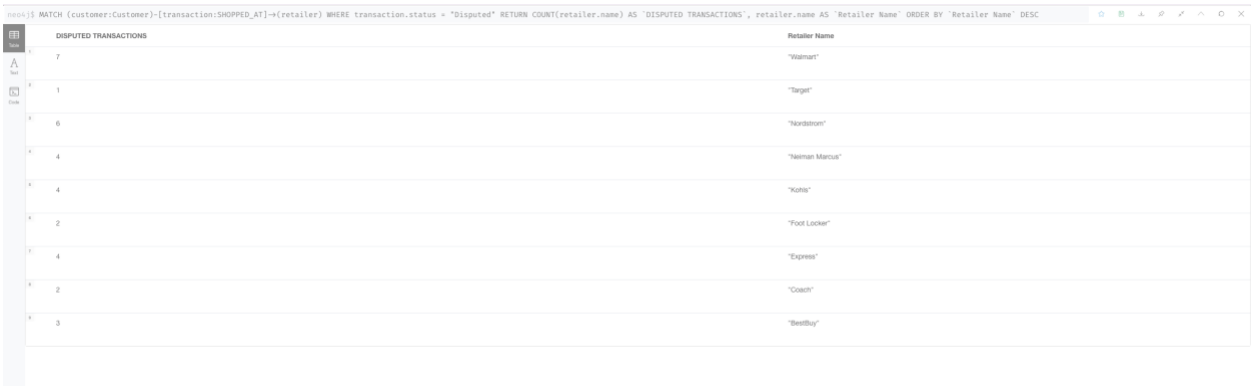
neo4j$ MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE transaction.status = "Disputed" RETURN customer.name AS `Customer Name`, retailer.name AS `Retailer Name`, transaction.amount AS `Transaction Amount`, transact…

	Customer Name	Retailer Name	Transaction Amount	Transaction date
1	" Rigoberto Kinchen "	"Express"	"721"	"4/1/2020"
2	"Edgar Haroop"	"Walmart"	"654"	"3/20/2020"
3	" Rigoberto Kinchen "	"Walmart"	"914"	"3/18/2020"
4	" Zenaida Sitzes "	"Walmart"	"1149"	"3/18/2020"
5	"Richard Smith"	"Coach"	"1145"	"3/18/2020"
6	"Ashlee Reid"	"Walmart"	"1149"	"3/18/2020"
7	"Sarah Radovic"	"Nordstrom"	"516"	"3/15/2020"
8	" Aura Schiel "	"Neiman Marcus"	"830"	"3/13/2020"
9	" Cary Mcenaney "	"Kohls"	"468"	"2/29/2020"
10	"Edgar Haroop"	"Walmart"	"1849"	"2/20/2020"
11	" Rigoberto Kinchen "	"Nordstrom"	"1003"	"2/20/2020"
12	" Corinne Suman "	"Nordstrom"	"816"	"2/20/2020"
13	"Lacy Grant"	"Nordstrom"	"1003"	"2/20/2020"
14	"Jonathan Rinka"	"Kohls"	"1345"	"2/18/2020"
15	" Zenaida Sitzes "	"BestBuy"	"378"	"2/10/2020"
16	"Sallie Bauer"	"Foot Locker"	"378"	"2/10/2020"
17	" Toni Pettway "	"Target"	"605"	"1/27/2020"
18	"Jonathan Rinka"	"Walmart"	"945"	"1/27/2020"

Started streaming 33 records after 25 ms and completed after 59 ms.

6. Write the Cypher code to get the number of disputed transactions for every retailer

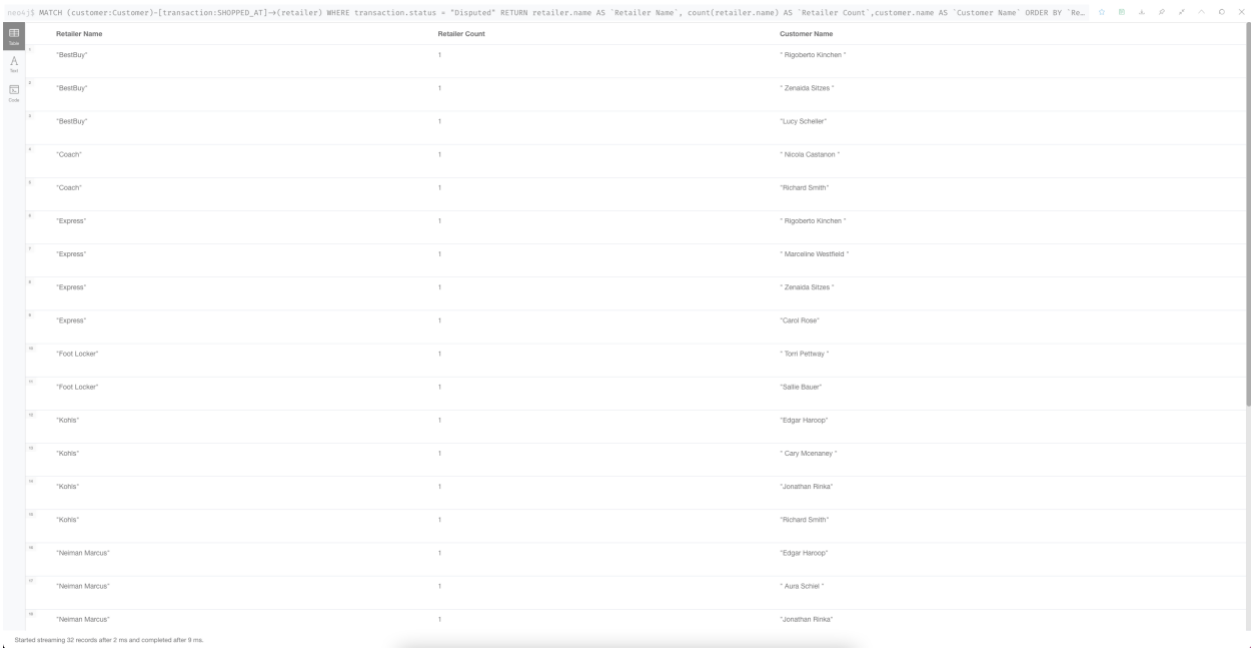
```
MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE
transaction.status = "Disputed"

RETURN COUNT(retailer.name) AS `DISPUTED TRANSACTIONS`, retailer.name AS `Retailer
Name`

ORDER BY `Retailer Name` DESC
```



DISPUTED TRANSACTIONS	Retailer Name
7	"Walmart"
1	"Target"
6	"Nordstrom"
4	"Neiman Marcus"
4	"Kohls"
2	"Foot Locker"
4	"Express"
2	"Coach"
3	"BestBuy"

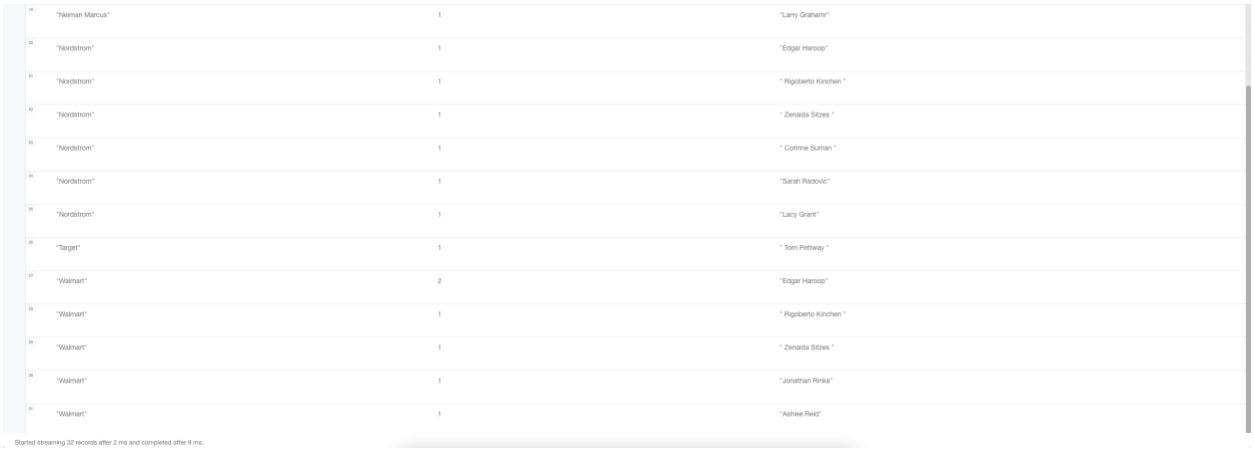
7. Write the Cypher code to get the number of disputed transactions and the list of customer names for these disputed transactions for every retailer

```
MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE
transaction.status = "Disputed"

RETURN  retailer.name AS `Retailer Name`, count(retailer.name) AS `Retailer
Count`,customer.name AS `Customer Name`

ORDER BY `Retailer Name`
```



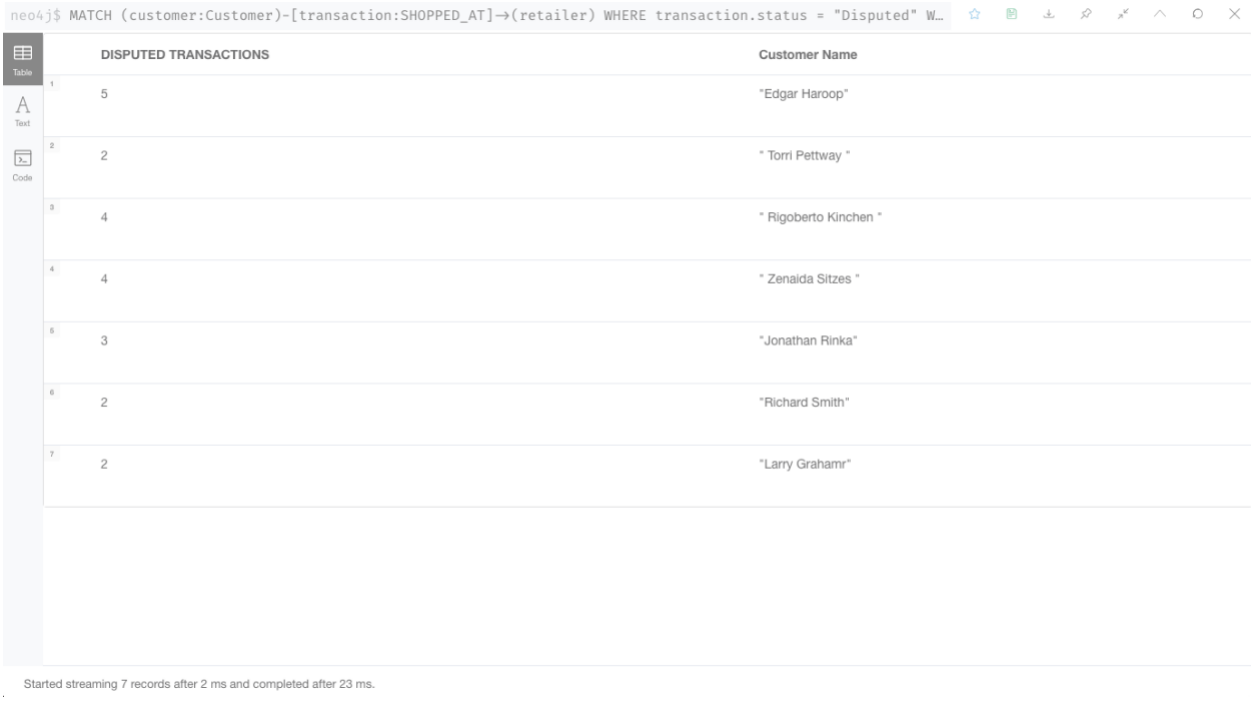
Retailer Name	Retailer Count	Customer Name
"BestBuy"	1	" Rigoberto Kinchen "
"BestBuy"	1	" Zenaida Sitzes "
"BestBuy"	1	"Lucy Scheller"
"Coach"	1	" Nicola Castanon "
"Coach"	1	"Richard Smith"
"Express"	1	" Rigoberto Kinchen "
"Express"	1	" Marceline Westfield "
"Express"	1	" Zenaida Sitzes "
"Express"	1	"Carol Rose"
"Foot Locker"	1	" Toni Pettway "
"Foot Locker"	1	"Sallie Bauer"
"Kohls"	1	"Edgar Haroop"
"Kohls"	1	" Cary Mcenaney "
"Kohls"	1	"Jonathan Rinka"
"Kohls"	1	"Richard Smith"
"Neiman Marcus"	1	"Edgar Haroop"
"Neiman Marcus"	1	" Aura Schiel "
"Neiman Marcus"	1	"Jonathan Rinka"

Started streaming 32 records after 2 ms and completed after 9 ms.



8. Write the Cypher code to get the number of disputed transactions for every customer that has more than one disputed transaction

```
MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer) WHERE transaction.status = "Disputed"

WITH count(customer.name) as `DISPUTED TRANSACTIONS`, customer.name as `Customer Name`

WHERE `DISPUTED TRANSACTIONS`>1

RETURN `DISPUTED TRANSACTIONS`, `Customer Name`
```



9. Write the Cypher code to get the list of stores on LaSalle street that have disputed transactions and the number of disputed transactions for every store; the store list must be sorted by store name in ascending order.

```
MATCH (customer:Customer)-[transaction:SHOPPED_AT]->(retailer)

WHERE transaction.status = "Disputed" AND (retailer.street  =~ '(?i).*LaSalle.*')

RETURN COUNT(retailer.name) AS `DISPUTED TRANSACTIONS`,retailer.name AS `Retailer Name`

ORDER BY `Retailer Name`
```

10. Write the Cypher code to get the list of Employees who work in at least 2 stores where disputed transactions reported in these retailers.

```
MATCH (employee:Employee)-[works:WORKS_AT]->(retailer) WHERE retailer.name<>"Gap"

WITH count(works) as `Number of Stores Worked At`, employee.name as `Employee Name`

WHERE `Number of Stores Worked At`>=2

RETURN `Employee Name`
```