Market Basket Analysis Tutorial using NEO4j

//Create the Customers

CREATE (c1: Customer {Name: "Orren"})

CREATE (c21: Customer {Name: "Peter"})

CREATE (c31: Customer {Name: "Preeti"})

CREATE (c41: Customer {Name: "John"})

CREATE (c51: Customer {Name: "Angus"})

CREATE (c61: Customer {Name: "Pamela"})

CREATE (c71: Customer {Name: "Jaipaul"})

CREATE (c81: Customer {Name: "Jane"})

CREATE (c91: Customer {Name: "Sunil"})

// Create the groups

CREATE (g1: Group {Name: "Breakfast"})

CREATE (g21: Group {Name: "Meat"})

CREATE (g31: Group {Name: "Vegetable"})

CREATE (g41: Group {Name: "Fruit"})

CREATE (g51: Group {Name: "Pastry"})

CREATE (g61: Group {Name: "Other"})

//create the items and relationships to groups

CREATE (i1: Item {Name: "Bread"})

with i1

MATCH (g1:Group{Name:"Pastry"})

CREATE (i1)-[a:IN\_GROUP]->(g1);

CREATE (i2: Item {Name: "Milk"})

with i2

MATCH (g2:Group{Name:"Beverage"})

CREATE (i2)-[a2:IN\_GROUP]->(g2);

CREATE (i3: Item {Name: "Cheese"})

with i3

MATCH (g3:Group{Name:"Other"})

CREATE (i3)-[a3:IN\_GROUP]->(g3);

CREATE (i4: Item {Name: "Pasta"})

with i4

MATCH (g4:Group{Name:"Other"})

CREATE (i4)-[a4:IN\_GROUP]->(g4);

CREATE (i5: Item {Name: "Tomato"})

with i5

MATCH (g5:Group{Name:"Vegetable"})

CREATE (i5)-[a5:IN\_GROUP]->(g5);

CREATE (i6: Item {Name: "Beef"})

with i6

MATCH (g6:Group{Name:"Meat"})

CREATE (i6)-[a6:IN\_GROUP]->(g6);

CREATE (i7: Item {Name: "Fish"})

with i7

MATCH (g7:Group{Name:"Meat"})

CREATE (i7)-[a7:IN\_GROUP]->(g7);

CREATE (i8: Item {Name: "Onions"})

with i8

MATCH (g8:Group{Name:"Vegetable"})

CREATE (i8)-[a8:IN\_GROUP]->(g8);

CREATE (i9: Item {Name: "Cake"})

with i9

MATCH (g9:Group{Name:"Pastry"})

CREATE (i9)-[a9:IN\_GROUP]->(g9);

CREATE (i10: Item {Name: "Muffin"})

with i10

MATCH (g10:Group{Name:"Pastry"})

CREATE (i10)-[a10:IN\_GROUP]->(g10);

CREATE (i11: Item {Name: "Burgers"})

with i11

MATCH (g11:Group{Name:"Meat"})

CREATE (i11)-[a11:IN\_GROUP]->(g11);

CREATE (i12: Item {Name: "Coconut"})

with i12

MATCH (g12:Group{Name:"Beverage"})

CREATE (i12)-[a12:IN\_GROUP]->(g12);

CREATE (i13: Item {Name: "Carrots"})

with i13

MATCH (g13:Group{Name:"Vegetable"})

CREATE (i13)-[a13:IN\_GROUP]->(g13);

CREATE (i14: Item {Name: "Lettuce"})

with i14

MATCH (g14:Group{Name:"Vegetable"})

CREATE (i14)-[a14:IN\_GROUP]->(g14);

CREATE (i15: Item {Name: "Lamb"})

with i15

MATCH (g15:Group{Name:"Meat"})

CREATE (i15)-[a15:IN\_GROUP]->(g15);

CREATE (i16: Item {Name: "Chicken"})

with i16

MATCH (g16:Group{Name:"Meat"})

CREATE (i16)-[a16:IN\_GROUP]->(g16);

CREATE (i17: Item {Name: "Oats"})

with i17

MATCH (g17:Group{Name:"Breakfast"})

CREATE (i17)-[a17:IN\_GROUP]->(g17);

CREATE (i18: Item {Name: "Cornflakes"})

with i18

MATCH (g18:Group{Name:"Breakfast"})

CREATE (i18)-[a18:IN\_GROUP]->(g18);

CREATE (i19: Item {Name: "Eggs"})

with i19

MATCH (g19:Group{Name:"Breakfast"})

CREATE (i19)-[a19:IN\_GROUP]->(g19);

CREATE (i20: Item {Name: "Pancakes"})

with i20

MATCH (g20:Group{Name:"Breakfast"})

CREATE (i20)-[a20:IN\_GROUP]->(g20);

CREATE (i21: Item {Name: "Bacon"})

with i21

MATCH (g21:Group{Name:"Breakfast"})

CREATE (i21)-[a21:IN\_GROUP]->(g21);

CREATE (i22: Item {Name: "Biscuit"})

with i22

MATCH (g22:Group{Name:"Pastry"})

CREATE (i22)-[a22:IN\_GROUP]->(g22);

CREATE (i23: Item {Name: "Coffee"})

with i23

MATCH (g23:Group{Name:"Beverage"})

CREATE (i23)-[a23:IN\_GROUP]->(g23);

CREATE (i24: Item {Name: "Tea"})

with i24

MATCH (g24:Group{Name:"Beverage"})

CREATE (i24)-[a24:IN\_GROUP]->(g24);

CREATE (i25: Item {Name: "Oranges"})

with i25

MATCH (g25:Group{Name:"Fruit"})

CREATE (i25)-[a25:IN\_GROUP]->(g25);

CREATE (i26: Item {Name: "Apples"})

with i26

MATCH (g26:Group{Name:"Fruit"})

CREATE (i26)-[a26:IN\_GROUP]->(g26);

//create the customer transactions one at a time

MATCH (c1: Customer {Name: "Orren"}) , (i1: Item {Name: "Milk"})  , (i2: Item {Name: "Tomato"}) , (i3: Item {Name: "Burgers"})  , (i4: Item {Name: "Coffee"})

CREATE (c1)-[r:BUYS]->(i1)

CREATE (c1)-[r1:BUYS]->(i2)

CREATE (c1)-[r2:BUYS]->(i3)

CREATE (c1)-[r3:BUYS]->(i4);

Match (c1: Customer {Name: "Peter"}), (i1: Item {Name: "Onions"}) ,

 (i2: Item {Name: "Carrots"}) , (i3: Item {Name: "Eggs"}) ,(i4: Item {Name: "Milk"}) ,  (i5: Item {Name: "Lettuce"}), (i6: Item {Name: "Bacon"})

CREATE (c1)-[r:BUYS]->(i1)

CREATE (c1)-[r1:BUYS]->(i2)

CREATE (c1)-[r2:BUYS]->(i3)

CREATE (c1)-[r3:BUYS]->(i4)

CREATE (c1)-[r4:BUYS]->(i5)

CREATE (c1)-[r5:BUYS]->(i6);

Match (c1: Customer {Name: "Preeti"}), (i1: Item {Name: "Bread"}) , (i2: Item {Name: "Pancakes"}) ,(i3: Item {Name: "Bacon"})  , (i4: Item {Name: "Beef"})  , (i5: Item {Name: "Lamb"})  , (i6: Item {Name: "Pasta"})  , (i7: Item {Name: "Tomato"}) , (i8: Item {Name: "Onions"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6)

CREATE (c1)-[r7:BUYS]->(i7)

CREATE (c1)-[r8:BUYS]->(i8);

MATCH (c1: Customer {Name: "John"}), (i1: Item {Name: "Chicken"})

, (i2: Item {Name: "Beef"}),  (i3: Item {Name: "Lettuce"})  , (i4: Item {Name: "Tea"}) ,

(i5: Item {Name: "Oranges"}) , (i6: Item {Name: "Biscuit"})  ,(i7: Item {Name: "Fish"}) ,

(i8: Item {Name: "Onions"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6)

CREATE (c1)-[r7:BUYS]->(i7)

CREATE (c1)-[r8:BUYS]->(i8);

MATCH (c1: Customer {Name: "Angus"}) , (i1: Item {Name: "Apples"})  , (i2: Item {Name: "Burgers"})  , (i3: Item {Name: "Cornflakes"})  ,

(i4: Item {Name: "Eggs"})  , (i5: Item {Name: "Cake"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5);

MATCH (c1: Customer {Name: "Pamela"}) , (i1: Item {Name: "Muffin"})  ,

(i2: Item {Name: "Burgers"}) , (i3: Item {Name: "Coconut"}) ,

(i4: Item {Name: "Carrots"})  ,(i5: Item {Name: "Lettuce"})  ,

(i6: Item {Name: "Lamb"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6);

MATCH (c1: Customer {Name: "Jaipaul"}) , (i1: Item {Name: "Muffin"})  ,

(i2: Item {Name: "Burgers"}) , (i3: Item {Name: "Coconut"}) ,

(i4: Item {Name: "Carrots"})  ,(i5: Item {Name: "Lettuce"})  ,

(i6: Item {Name: "Lamb"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6);

Match(c1: Customer {Name: "Jane"}) , (i1: Item {Name: "Tea"}) , (i2: Item {Name: "Bread"})  ,

(i3: Item {Name: "Milk"})  ,(i4: Item {Name: "Cheese"})  ,(i5: Item {Name: "Pasta"})  ,

(i6: Item {Name: "Tomato"})  ,(i7: Item {Name: "Beef"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6)

CREATE (c1)-[r7:BUYS]->(i7);

Match(c1: Customer {Name: "Sunil"}) , (i1: Item {Name: "Tea"}) , (i2: Item {Name: "Bread"})  ,

(i3: Item {Name: "Milk"})  ,(i4: Item {Name: "Cheese"})  ,(i5: Item {Name: "Pasta"})  ,

(i6: Item {Name: "Tomato"})  ,(i7: Item {Name: "Beef"})

CREATE (c1)-[r1:BUYS]->(i1)

CREATE (c1)-[r2:BUYS]->(i2)

CREATE (c1)-[r3:BUYS]->(i3)

CREATE (c1)-[r4:BUYS]->(i4)

CREATE (c1)-[r5:BUYS]->(i5)

CREATE (c1)-[r6:BUYS]->(i6)

CREATE (c1)-[r7:BUYS]->(i7);

Market Basket scenarios

### Recommend products to a customer

MATCH (c:Customer{Name:"Orren"})-[:BUYS]->(i:Item)<-[:BUYS]-(oc:Customer)-[:BUYS]->(oi:Item)

RETURN oi.Name as `Recommended Item`,count(oc) as `Score`

ORDER BY count(oc) DESC

### What's getting ordered

MATCH (p:Item)<-[r:BUYS]-()

RETURN p.Name, count(r) AS timesOrdered

ORDER BY timesOrdered DESC

LIMIT 20

### What's purchased with Lamb?

MATCH (p:Item {Name: "Lamb"})<-[:BUYS]-()-[:BUYS]->(otherProduct:Item)

RETURN otherProduct.Name, count(\*) as occurrences

ORDER BY occurrences DESC

### Buying Beef and Lamb? What else should you buy?

MATCH (o:Customer)-[:BUYS]-(p:Item),

 (o)-[:BUYS]->(p2:Item {Name: "Pasta"}),

 (o)-[:BUYS]-(otherp:Item)

 WHERE otherp <> p AND otherp <> p2 AND p.Name CONTAINS "Bread"

 RETURN otherp.Name, count(\*) AS occurrences

 ORDER BY occurrences DESC

 LIMIT 10

### What's getting ordered by group – could be used for isle settings

MATCH (customer)-[r1:BUYS]->(p2:Item)-[g:IN\_GROUP]-(group)

RETURN group.Name, count(g) AS timesOrdered

ORDER BY timesOrdered DESC

LIMIT 20