E-commerce Website Database Management

Group 10
Anshita Aishwarya
Vignesh Selvanayagam

Implementation of Ecommerce Relational Model via NoSQL:

Creation of tables:

1. Customer:

```
LOAD CSV WITH HEADERS FROM 'file:///Customer.csv' AS row WITH row WHERE row.CustomerID IS NOT NULL
MERGE (c:Customer {CustomerID:row.CustomerID})
ON CREATE SET c.FirstName = row.FirstName, c.LastName = row.LastName, c.DOB = row.DOB, c.Gender = row.Gender;
```

2. Category:

```
LOAD CSV WITH HEADERS FROM 'file:///Category.csv' AS row WITH row WHERE row.CategoryID IS NOT NULL
MERGE (ct:Category {CategoryID:row.CategoryID})
ON CREATE SET ct.CategoryName = row.CategoryName, ct.Description= row.Description;
```

3. Product:

```
LOAD CSV WITH HEADERS FROM 'file:///Product.csv' AS row WITH row WHERE row.ProductID IS NOT NULL
MERGE (p:Product {ProductID: row.ProductID})
ON CREATE SET p.ProductName = row.ProductName, p.Description = row.Description, p.UnitPrice = toFloat(row.UnitPrice), p.UnitWeight = toFloat(row.UnitWeight), p.Manufacturer = row.Manufacturer, p.UnitsinStock = row.UnitsinStock, p.Rating = toFloat(row.Rating), p.CategoryID = row.CategoryID;
```

4. Orders:

```
LOAD CSV WITH HEADERS FROM 'file:///Orders.csv' AS row WITH row WHERE row.OrderID IS NOT NULL
MERGE (o:Order {OrderID: row.OrderID})
ON CREATE SET o.OrderDate = row.OrderDate, o.CustomerID = row.CustomerID, o.TotalAmount = toFloat(row.TotalAmount), o.Discount = toFloat(row.Discount), o.ShippedDate = row.ShippedDate, o.ShippedAddress = row.ShippedAddress, o.ShippedCity = row.ShippedCity, o.ShippedState = row.ShippedState, o.Pincode = row.Pincode, o.ContactNumber = row.ContactNumber;
```

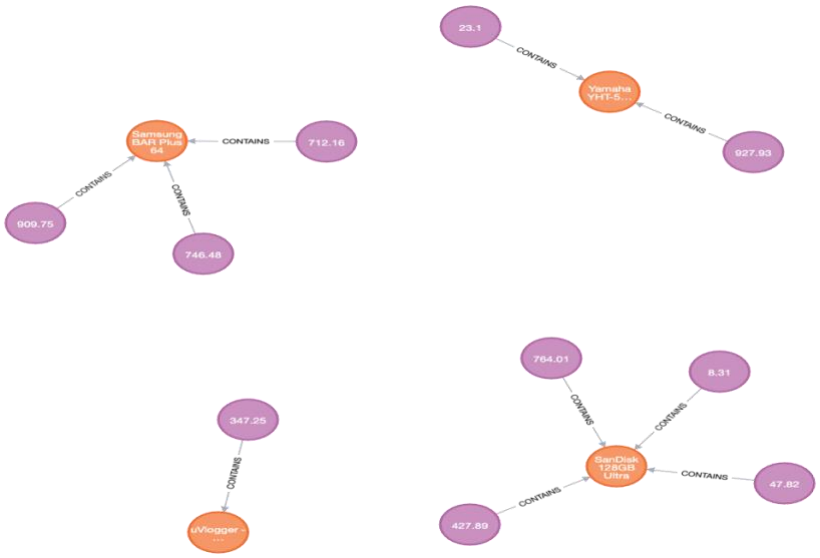
5. Supplier:

```
LOAD CSV WITH HEADERS FROM 'file:///Supplier.csv' AS row WITH row WHERE row.SupplierID
IS NOT NULL
MERGE (s:Supplier {SupplierID: row.SupplierID})
ON CREATE SET s.CompanyName = row.CompanyName, s.Phone = row.Phone;
```

Creation of relationships:

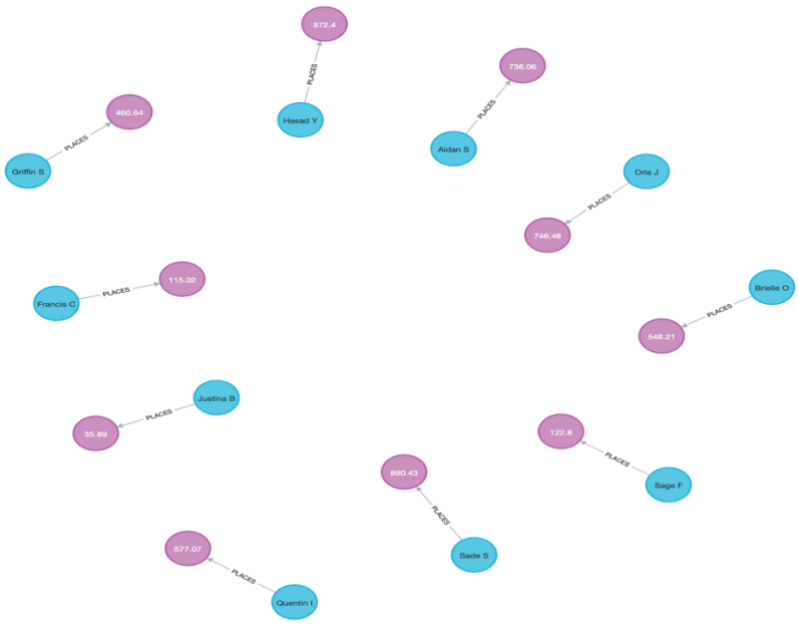
1. Contains (relationship between orders and products):

```
LOAD CSV WITH HEADERS FROM 'file:///Orders.csv' AS row
MATCH (o:Order {OrderID: row.OrderID})
MATCH (p:Product {ProductID: row.ProductID})
MERGE (o)-[op:CONTAINS]->(p)
```



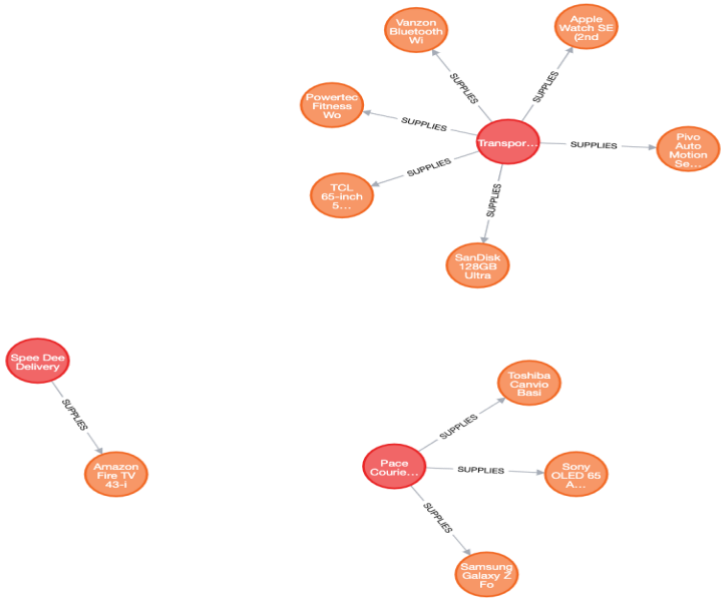
2. Places (relationship between customers and orders)

```
LOAD CSV WITH HEADERS FROM 'file:///Orders.csv' AS row
MATCH (o:Order {OrderID: row.OrderID})
MATCH (c:Customer {CustomerID: row.CustomerID})
MERGE (c)-[:PLACES]->(o);
```



3. Supplies (relationship between suppliers and products):

```
LOAD CSV WITH HEADERS FROM 'file:///Product.csv' AS row
MATCH (p:Product {ProductID: row.ProductID})
MATCH (s:Supplier {SupplierID: row.SupplierID})
MERGE (s)-[:SUPPLIES]->(p);
```



4. Belongs to (relationship between products and categories):

```
LOAD CSV WITH HEADERS FROM 'file:///Product.csv' AS row
MATCH (p:Product {ProductID: row.ProductID})
MATCH (ct:Category {CategoryID: row.CategoryID})
MERGE (p)-[:BELONGS_TO]->(ct);
```