GROUP PROJECT

Team Name: Cluster Crafters (Mobile Phone Supply Chain Management)
PART 5: Implementation of the Given Schema in MongoDB

1. Introduction

In this report, we detail the implementation of a schema in a NoSQL distributed database system, leveraging MongoDB. The implementation focuses on ensuring high availability, scalability, and fault tolerance of the NoSQL system. The CRUD (Create, Read, Update, Delete) operations have been coded, and additional sample queries have been included to showcase the system's capabilities.

2. Implementation of CRUD Operations

The CRUD operations have been implemented using the Python programming language and the PyMongo driver for MongoDB. The operations are as follows:

- Create Operation:
The `crud_create` function inserts a new document into the specified collection. It demonstrates the insertion of a sample document representing a new phone product.

- Read Operation:
The `crud_read` function queries a document based on a specific field (e.g., name) and prints the result. It demonstrates the retrieval of a document representing a new phone.

- Update Operation:
The `crud_update` function updates a document based on a specific field (e.g., name) with new data. It showcases the modification of the price and stock of a phone.

- Delete Operation:
The `crud_delete` function deletes a document based on a specific field (e.g., name). It demonstrates the removal of a document representing a phone.

```
OEC0758:Task5 snagen11$ python3 task5.py
Number of documents in 'supplychain.supplychain_collection': 2440

CREATE (INSERT) OPERATION:
Document successfully inserted with ID: 656546c8fa52126152e675d2

READ (QUERY) OPERATION:
Found document:
{'_id': ObjectId('656546c8fa52126152e675d2'), 'name': 'New Phone', 'brand': 'Brand XYZ', 'price': 599.99, 'stock': 100}

UPDATE OPERATION:
Document successfully updated. Modified 1 field(s).

DELETE OPERATION:
Document successfully deleted. Deleted 1 document(s).
```

3. Connection and Database Information

The MongoDB connection string is provided in the code, connecting to a MongoDB server hosted at MongoDb cloud The database named "supplychain" and the collection named "supplychain_collection" are accessed for the implementation.

The number of documents in the specified collection is counted and printed for reference.

4. Sample Queries and Data Retrieval

After executing the CRUD operations, a set of sample queries is performed to retrieve data from the implemented NoSQL system. The queries include:

- Retrieve Number of Suppliers:
The `retrieve_suppliers` function retrieves all suppliers from the "supplier" collection, counts them, and displays a sample.

- Retrieve MobilePhones Manufactured in a Specific Year:
The `retrieve_manufacturer` function retrieves mobile phones manufactured in the year 2021, counts them, and displays a sample.

- Retrieve Inventory Items in a Specific Warehouse:
The `retrieve_inventory` function retrieves inventory items from a specific warehouse (WarehouseID: 80) and displays them.

- Retrieve OrderDetails for a Specific MobilePhone:
The `retrieve_order` function retrieves order details for a specific mobile phone (MobilePhoneID: 68) and displays them.

- Retrieve Warehouses in a Specific Location:
The `retrieve_warehouses` function retrieves warehouses in a specific location (Sweden) and displays them.

```
SAMPLE QUERIES and DATA RETRIEVAL

 Retrieving Number of suppliers :  20
 Sample retrieval data from supplier: {'_id': ObjectId('6563b33805ce91e111017957'), 'SupplierID': 20, 'SupplierName': 'Alcatel', 'ContactInfo': '363-
529-7108', 'Address': '438 Buell Way'}


 Number of mobilephones manufactured in 2021 :  4
Sample retrieval data from manufacturer : {'_id': ObjectId('6563b32405ce91e11101713d'), 'MobilePhoneID': 49, 'ManufactureID': 5, 'ModelName': 'W', 'P
rice': 2687.23, 'ReleaseYear': 2021}
```

```
Inventory items present in warehouse 80
{'_id': ObjectId('6563b31505ce91e111017093'), 'InventoryID': 1, 'MobilePhoneID': 42, 'WarehouseID': 80, 'Quantity': 14788, 'PurchaseDate': '4/22/20'}
{'_id': ObjectId('6563b31505ce91e11101709d'), 'InventoryID': 11, 'MobilePhoneID': 24, 'WarehouseID': 80, 'Quantity': 13987, 'PurchaseDate': '1/2/20'}
```

```
Retrieve OrderDetails for a Specific MobilePhone 68
{'_id': ObjectId('6563b33105ce91e11101755b'), 'OrderDetailID': 1, 'OrderID': 277, 'MobilePhoneID': 68, 'Quantity': 1}
{'_id': ObjectId('6563b33105ce91e111017598'), 'OrderDetailID': 62, 'OrderID': 346, 'MobilePhoneID': 68, 'Quantity': 3}
{'_id': ObjectId('6563b33105ce91e1110175bb'), 'OrderDetailID': 97, 'OrderID': 982, 'MobilePhoneID': 68, 'Quantity': 7}
{'_id': ObjectId('6563b33105ce91e1110175c6'), 'OrderDetailID': 108, 'OrderID': 228, 'MobilePhoneID': 68, 'Quantity': 3}
{'_id': ObjectId('6563b33105ce91e111017617'), 'OrderDetailID': 189, 'OrderID': 721, 'MobilePhoneID': 68, 'Quantity': 5}
{'_id': ObjectId('6563b33105ce91e11101763a'), 'OrderDetailID': 224, 'OrderID': 856, 'MobilePhoneID': 68, 'Quantity': 8}
{'_id': ObjectId('6563b33105ce91e11101769c'), 'OrderDetailID': 322, 'OrderID': 798, 'MobilePhoneID': 68, 'Quantity': 10}
{'_id': ObjectId('6563b33105ce91e1110176c9'), 'OrderDetailID': 367, 'OrderID': 149, 'MobilePhoneID': 68, 'Quantity': 8}
{'_id': ObjectId('6563b33105ce91e11101772b'), 'OrderDetailID': 465, 'OrderID': 453, 'MobilePhoneID': 68, 'Quantity': 1}
{'_id': ObjectId('6563b33105ce91e11101773d'), 'OrderDetailID': 483, 'OrderID': 774, 'MobilePhoneID': 68, 'Quantity': 10}
{'_id': ObjectId('6563b33105ce91e111017762'), 'OrderDetailID': 520, 'OrderID': 655, 'MobilePhoneID': 68, 'Quantity': 3}
{'_id': ObjectId('6563b33105ce91e111017793'), 'OrderDetailID': 569, 'OrderID': 367, 'MobilePhoneID': 68, 'Quantity': 5}
{'_id': ObjectId('6563b33105ce91e1110177cf'), 'OrderDetailID': 629, 'OrderID': 914, 'MobilePhoneID': 68, 'Quantity': 4}
{'_id': ObjectId('6563b33105ce91e1110178c4'), 'OrderDetailID': 874, 'OrderID': 102, 'MobilePhoneID': 68, 'Quantity': 4}
```

```
Retrieve Warehouses in a Specific Location – Sweden
{'_id': ObjectId('6563b33f05ce91e111017983'), 'WarehouseID': 43, 'Location': 'Sweden', 'Capacity': 27083}
{'_id': ObjectId('6563b33f05ce91e111017984'), 'WarehouseID': 44, 'Location': 'Sweden', 'Capacity': 395131}
{'_id': ObjectId('6563b33f05ce91e1110179c6'), 'WarehouseID': 110, 'Location': 'Sweden', 'Capacity': 153384}
{'_id': ObjectId('6563b33f05ce91e1110179de'), 'WarehouseID': 134, 'Location': 'Sweden', 'Capacity': 632256}
{'_id': ObjectId('6563b33f05ce91e1110179f1'), 'WarehouseID': 153, 'Location': 'Sweden', 'Capacity': 827942}
OEC0758:Task5 snagen11$
```

Conclusion

This report outlines the successful implementation of a schema in a NoSQL distributed database system using MongoDB. The CRUD operations and sample queries demonstrate the functionality, high availability, scalability, and fault tolerance of the NoSQL system in a real-world supply chain scenario.