Implementation in NOSQL (MongoDB)

1)Find All Customers

- This query retrieves all documents from the customer collection:

db.customer.find({});

```
{
  _id: ObjectId('674bac5f7375693cca052147'),
  c_id: 1,
  name: 'Newton Harnor',
  'phone ': '253-419-1771',
  address: 'Tacoma',
  email: 'newton.harnor@gmail.com'
}
{
  _id: ObjectId('674bac5f7375693cca052148'),
  c_id: 2,
  name: 'Tomaso Voisey',
  'phone ': '330-079-3654',
  address: 'Akron',
  email: 'tomaso.voisey@yahoo.co.uk'
}
{
  _id: ObjectId('674bac5f7375693cca052149'),
  c_id: 3,
  name: 'Margery Glover',
  'phone ': '785-292-5555',
  address: 'Topeka',
  email: 'margery.glover@hotmail.com'
}
```

2) Find All Products Purchased in a Specific Order

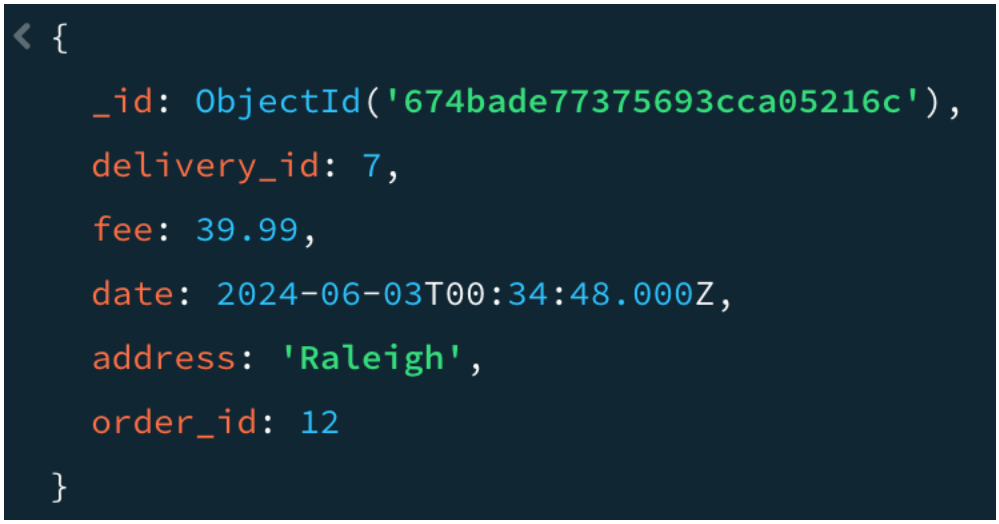
- This query is used to fetch the product details for a specific order (e.g., order_id: 12):

db.sales_order_detail.find({ order_id: 12 });

3) Find Deliveries in a Specific Date Range

- This query is used to find deliveries within a specific date range (e.g., between 2024-06-01 and 2024-06-30):

```
db.Delivery.find({
  date: {
    $gte: ISODate("2024-06-01T00:00:00.000Z"), // Greater than or equal to June 1, 2024
    $lte: ISODate("2024-06-30T23:59:59.999Z")  // Less than or equal to June 30, 2024
  }
});
```



4) Find the Total Payments Received by the Customers

- This query groups payment records by customer_id and calculates the total amount paid.

```
db.payments.aggregate([
  {
    $group: {
      _id: "$customer_id", // Group by customer_id
      totalPayments: { $sum: "$amount" } // Sum up the 'amount' field
    }
  },
  {
    $sort: { totalPayments: -1 } // Sort by total payments in descending order
  }
]);
```

5) Find Customer Details Along with Their Orders

- This query uses $lookup to join the customer collection with the SalesOrder collection to fetch customer details and their associated orders:

```
db.customer.aggregate([
  {
    $lookup: {
      from: "SalesOrder", // Join with the SalesOrder collection
      localField: "customer_id", // Field in the 'customer' collection
      foreignField: "customer_id", // Matching field in the 'SalesOrder' collection
      as: "orders" // Name for the joined data
    }
  },
  {
    $project: {
      customer_id: 1,
      name: 1,
      email: 1,
      orders: 1 // Include only relevant fields
    }
  }
]);
```

```
{
  _id: ObjectId('674bac5f7375693cca052147'),
  name: 'Newton Harnor',
  email: 'newton.harnor@gmail.com',
  orders: [
    {
      _id: ObjectId('674bae7d7375693cca0522a4'),
      order_id: 1,
      c_id: 7,
      payment_id: 7,
      order_date: '2/29/2024',
      order_type: 'RETAIL'
    },
    {
      _id: ObjectId('674bae7d7375693cca0522a5'),
      order_id: 2,
      c_id: 18,
      payment_id: 18,
      order_date: '5/1/2024',
      order_type: 'WHOLESALE',
      discount: 29
    },
```