**Problem Statement 1: MongoDB Aggregation**

You have been given a dataset containing details about different books. Each book has the following fields:

● title: The title of the book

● author:Theauthor of the book

● genre:Thegenreof the book (e.g., Fiction, Non-Fiction, Mystery, Sci-Fi)

● price:Theprice of the book

● published\_date: The date the book was published

The dataset has been stored in a MongoDB collection named books. Task:

Using the MongoDB aggregation framework, perform the following tasks:

1. Find the average price of all books.

2. find the count of books in each genre.

3. Foreachgenre, find the most expensive book.

4. Find the authors who have written more than 3 books.

5. Sort the books by published\_date in descending order, and then by price in ascending order

**1. Find the Average Price of All Books**

**javascript**

**Copy code**

db.books.aggregate([

{

$group: {

\_id: null,

averagePrice: { $avg: "$price" }

}

},

{

$project: {

\_id: 0,

averagePrice: 1

}

}

])

**2. Find the Count of Books in Each Genre**

db.books.aggregate([

{

$group: {

\_id: "$genre",

bookCount: { $sum: 1 }

}

},

{

$project: {

genre: "$\_id",

bookCount: 1,

\_id: 0

}

}

])

**3. For Each Genre, Find the Most Expensive Book**

db.books.aggregate([

{

$sort: { price: -1 }

},

{

$group: {

\_id: "$genre",

mostExpensiveBook: { $first: "$$ROOT" }

}

},

{

$project: {

genre: "$\_id",

title: "$mostExpensiveBook.title",

price: "$mostExpensiveBook.price",

\_id: 0

}

}

])

**4. Find Authors Who Have Written More Than 3 Books**

db.books.aggregate([

{

$group: {

\_id: "$author",

bookCount: { $sum: 1 }

}

},

{

$match: {

bookCount: { $gt: 3 }

}

},

{

$project: {

author: "$\_id",

bookCount: 1,

\_id: 0

}

}

])

**5. Sort Books by published\_date (Descending) and price (Ascending)**

db.books.aggregate([

{

$sort: {

published\_date: -1,

price: 1

}

}

])

**Problem Statement 2: MongoDB Aggregation**

2. You are working with an e-commerce platform that sells various products across different categories. You have access to a MongoDB collection named orders which keeps a record of all the purchases. Each document in this collection has the following fields:

● orderId: Aunique identifier for each order

● customerId: Aunique identifier for each customer

● items:Anarray of objects where each object represents a product purchased and contains:

● productId: The ID of the product

● productName:Nameoftheproduct

● category: The category of the product (e.g., Electronics, Clothing, Groceries)

● price:Theprice of the product

● quantity: The quantity of the product ordered

● orderDate: The date and time when the order was placed

● shippingAddress: The address to which the order will be shipped

● status: The status of the order (e.g., 'Shipped', 'Delivered', 'Pending', 'Cancelled')

**Tasks:**

1. Identify the top 3 most popular product categories based on the number of items sold.

2. Find out the total amount spent by each customer.

3. Identify the top 5 customers who've spent the most on the platform.

4. OrderInsights: For each month:

● Calculate the total revenue

● Identify the most sold product

● Determinetheaverage number of items per order

● Findoutthepercentage of orders that were cancelled

5. Shipping Analysis: Group orders based on the zip code in the shippingAddress (assuming the zip code is the last 5 characters of the address). For each group, calculate the total number of orders and the average order amount.

**1. Identify the Top 3 Most Popular Product Categories**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: "$items.category",

itemsSold: { $sum: "$items.quantity" }

}

},

{

$sort: { itemsSold: -1 }

},

{

$limit: 3

}

])

**2. Find the Total Amount Spent by Each Customer**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: "$customerId",

totalSpent: { $sum: { $multiply: ["$items.price", "$items.quantity"] } }

}

},

{

$project: {

customerId: "$\_id",

totalSpent: 1,

\_id: 0

}

}

])

**3. Identify the Top 5 Customers Who’ve Spent the Most**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: "$customerId",

totalSpent: { $sum: { $multiply: ["$items.price", "$items.quantity"] } }

}

},

{

$sort: { totalSpent: -1 }

},

{

$limit: 5

}

])

**4. Order Insights (Monthly)**

**a. Calculate the Total Revenue for Each Month**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: { $month: "$orderDate" },

totalRevenue: { $sum: { $multiply: ["$items.price", "$items.quantity"] } }

}

}

])

**b. Identify the Most Sold Product Each Month**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: { month: { $month: "$orderDate" }, productName: "$items.productName" },

quantitySold: { $sum: "$items.quantity" }

}

},

{

$sort: { "quantitySold": -1 }

},

{

$group: {

\_id: "$\_id.month",

mostSoldProduct: { $first: "$\_id.productName" },

quantitySold: { $first: "$quantitySold" }

}

}

])

**c. Determine the Average Number of Items per Order Each Month**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: { $month: "$orderDate" },

itemsCount: { $sum: "$items.quantity" },

orderCount: { $sum: 1 }

}

},

{

$project: {

\_id: 0,

month: "$\_id",

avgItemsPerOrder: { $divide: ["$itemsCount", "$orderCount"] }

}

}

])

**d. Find the Percentage of Orders that Were Cancelled Each Month**

db.orders.aggregate([

{

$group: {

\_id: { $month: "$orderDate" },

totalOrders: { $sum: 1 },

cancelledOrders: { $sum: { $cond: [{ $eq: ["$status", "Cancelled"] }, 1, 0] } }

}

},

{

$project: {

month: "$\_id",

cancellationPercentage: { $multiply: [{ $divide: ["$cancelledOrders", "$totalOrders"] }, 100] },

\_id: 0

}

}

])

**5. Shipping Analysis (Group by Zip Code)**

**a. Calculate the Total Number of Orders and Average Order Amount per Zip Code**

db.orders.aggregate([

{

$unwind: "$items"

},

{

$group: {

\_id: { zipCode: { $substr: ["$shippingAddress", -5, 5] } },

totalOrders: { $sum: 1 },

totalAmount: { $sum: { $multiply: ["$items.price", "$items.quantity"] } }

}

},

{

$project: {

zipCode: "$\_id.zipCode",

totalOrders: 1,

averageOrderAmount: { $divide: ["$totalAmount", "$totalOrders"] },

\_id: 0

}

}

])