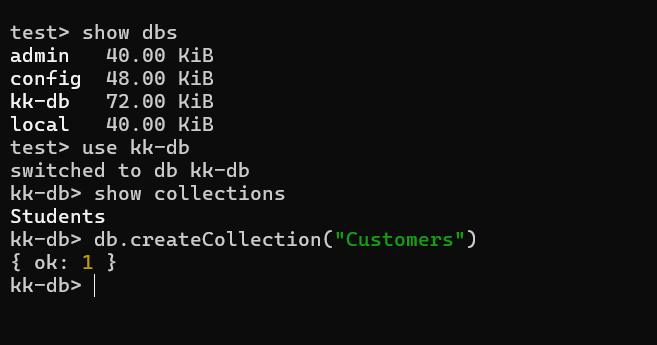
**AGGREGATE PIPELINE IN MONGODB**

Create Database **“kk-db”** and collection **“Customers”**



Insert the documents



**Aggregation Pipeline**

structure

*db.collection.aggregate([*

*{stage1},*

*{stage2},*

*{stage3},*

*])*

**$match** – Filters documents to pass only those that match the given condition(s).

**$group** – Groups documents by some field(s) and performs aggregation operations like sum, avg, min, max, etc.

**$sort** - Sorts the documents by specified fields

**$project** – Reshapes each document by adding or removing fields

**$limit** – Limits the number of documents passed to the next stage.

**$skip** – Skips the specified number of documents

**$unwind** – Deconstructs an array field from the input documents to output a document for each element of the array

**$lookup** – Joins documents from another collection (like SQL JOIN).

**Calculate the total revenue generated from all completed orders**

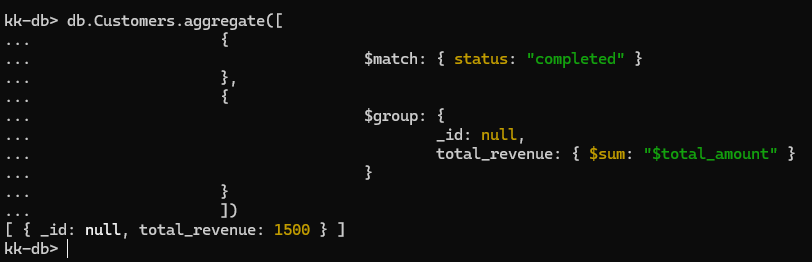
[Hint: Find completed orders and sum the amounts to get total revenue]

*db.Customers.aggregate([*

*{ $match: { status: "completed" } },*

*{ $group: { \_id: null, total\_revenue: { $sum: "$total\_amount" }}}*

*])*



**Find the average order value for all completed orders.**

[Hint: Find completed orders and average the amount]

*db.Customers.aggregate([*

*{ $match: { status: "completed" }},*

*{ $group: { \_id: null, avg\_order: { $avg: "$total\_amount" } }}*

*])*



**Identify the customer who has spent the most across all their orders.**

[Hint: Group Customers by their spending, sort in descending order and get top record]

*db.Customers.aggregate([*

*{$group: {*

*\_id: "$customer\_name",*

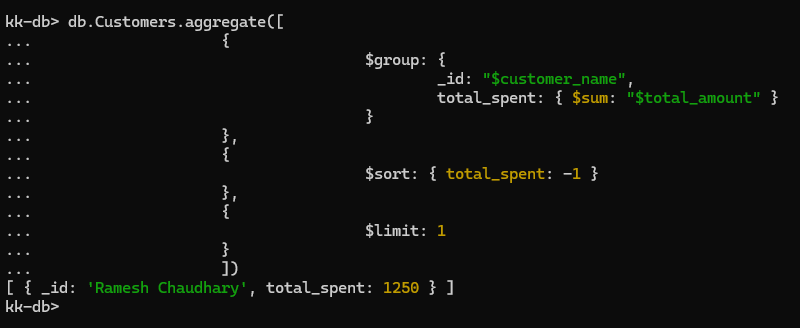
*total\_spent: { $sum: "$total\_amount" } }*

*},*

*{ $sort: { total\_spent: -1 } },*

*{ $limit: 1 }*

*])*



**Determine the most popular product based on the total quantity sold.**

[Hint: Separate with items, group with total quantities, sort in descending order and select top record]

*db.Customers.aggregate([*

*{ $unwind: "$items" },*

*{$group: {*

*\_id: "$items.product",*

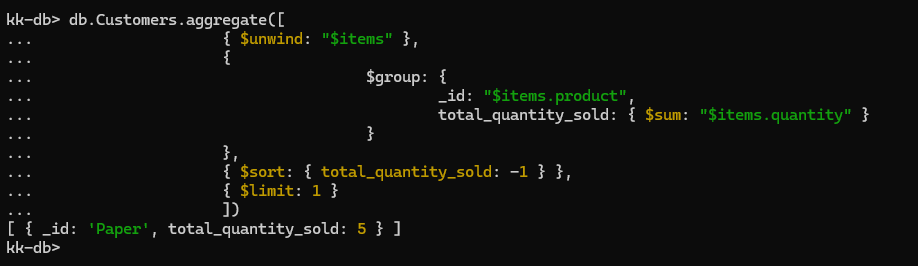
*total\_quantity\_sold: { $sum: "$items.quantity" }}*

*},*

*{ $sort: { total\_quantity\_sold: -1 } },*

*{ $limit: 1 }*

*])*



**Count the number of orders in each status.**

[Hint: Group by customer status and count]

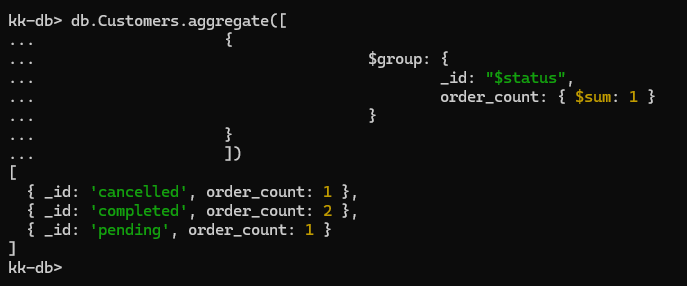
*db.Customers.aggregate([*

*{$group: {*

*\_id: "$status",*

*order\_count: { $sum: 1 } }}*

*])*



**Calculate the total revenue generated by each customer.**

[Hint: Group customer by their total revenue]

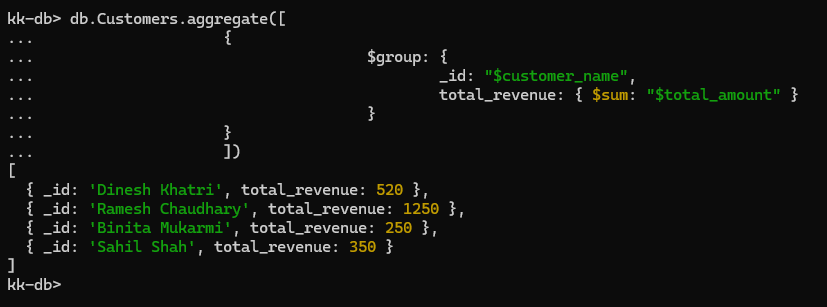
*db.Customers.aggregate([*

*{$group: {*

*\_id: "$customer\_name",*

*total\_revenue: { $sum: "$total\_amount" } }}*

*])*



**Find all orders placed between October 1,2023 and October 1,2023**

[Hint: Filter order with dates ]

*db.Customers.aggregate([*

*{$match: {*

*order\_date: {*

*$gte: ISODate("2023-10-01T00:00:00Z"),*

*$lte: ISODate("2023-10-03T23:59:59Z")*

*}}}*

*])*



**Calculate the total quantity sold for each product**

[Hint: separates with item and group item with total quantity sold]

*db.Customers.aggregate([*

*{$unwind: "$items" },*

*{*

*$group: {*

*\_id: "$items.product",*

*total\_quantity\_sold: { $sum: "$items.quantity" } }*

*},*

*{$sort: { total\_quantity\_sold: -1 }}*

*])*

