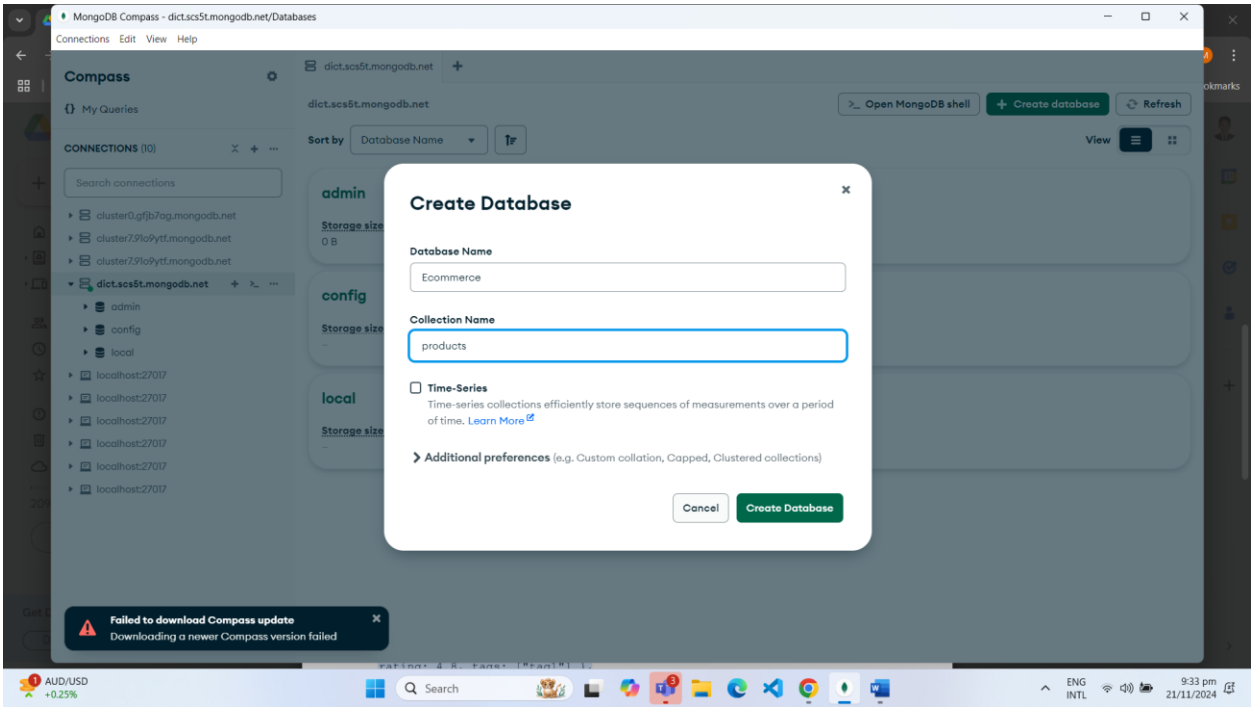
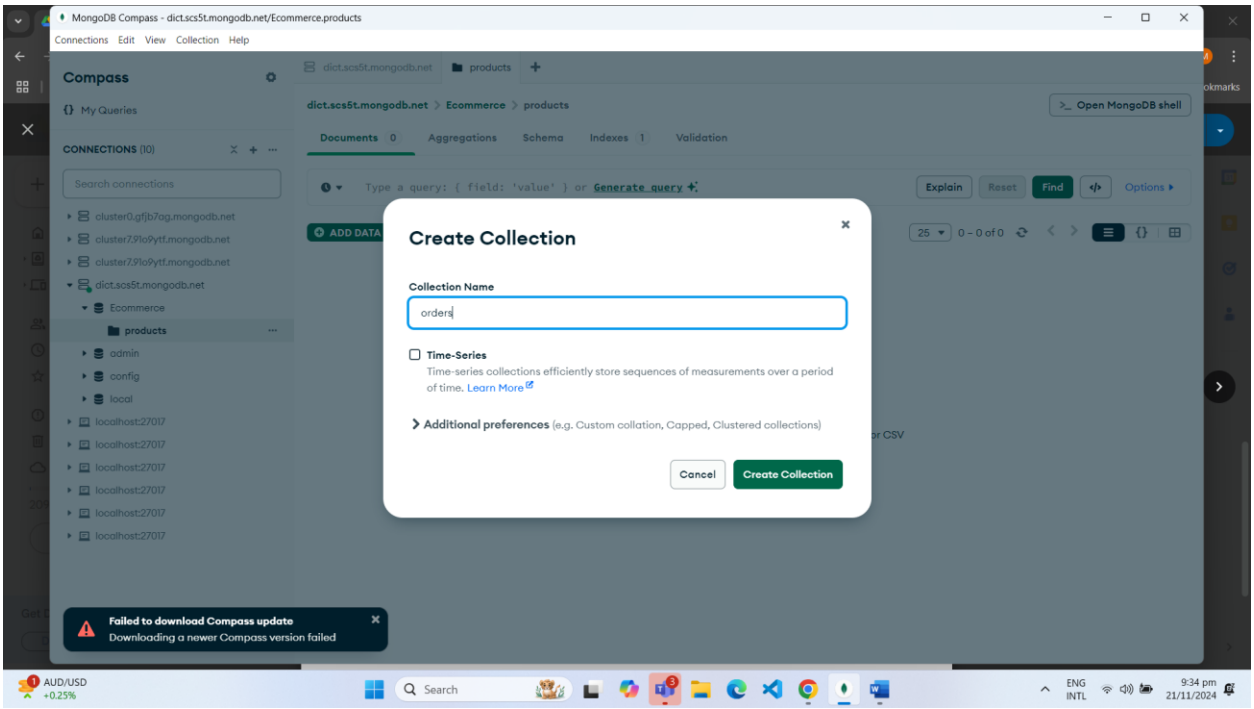
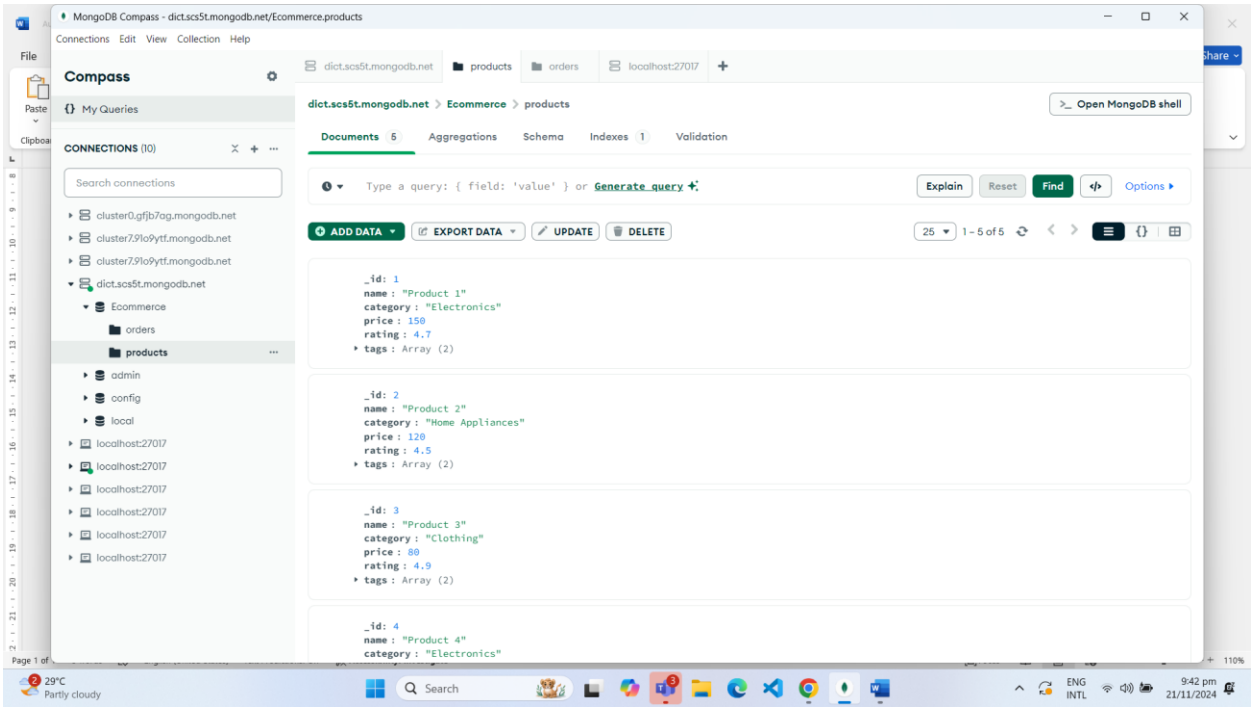
Activity 1: Query Optimization Challenge

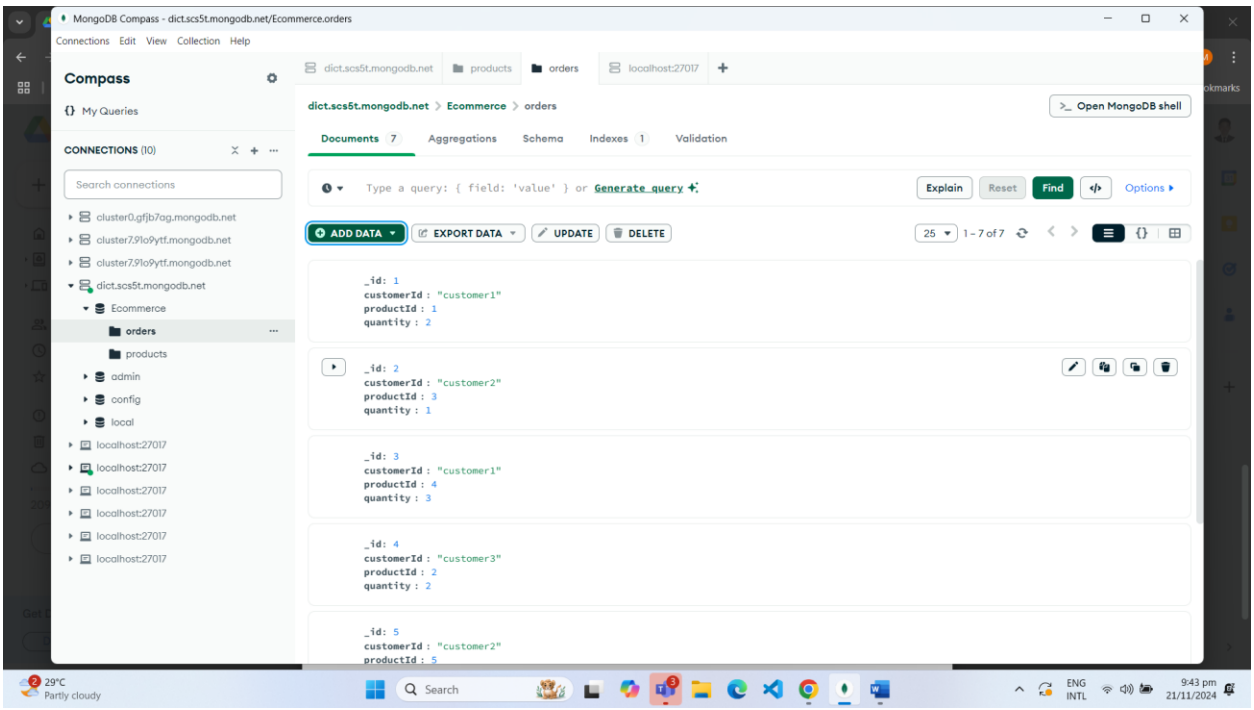




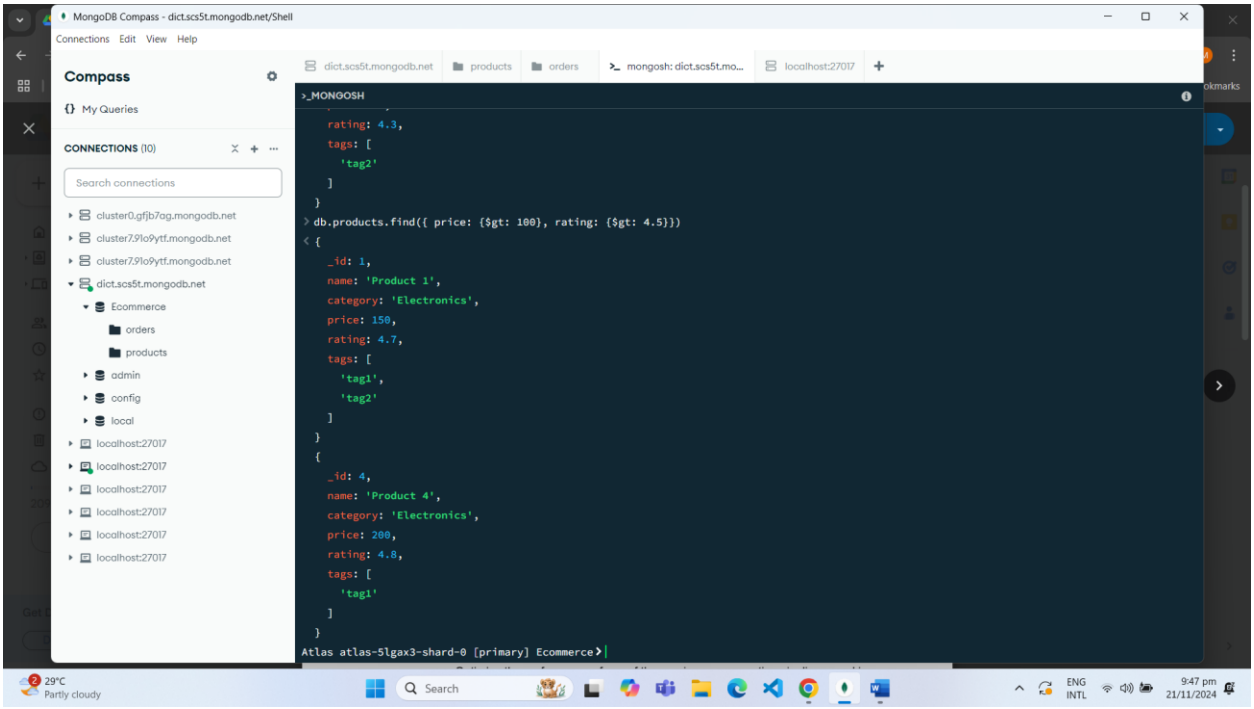
Inserted Products



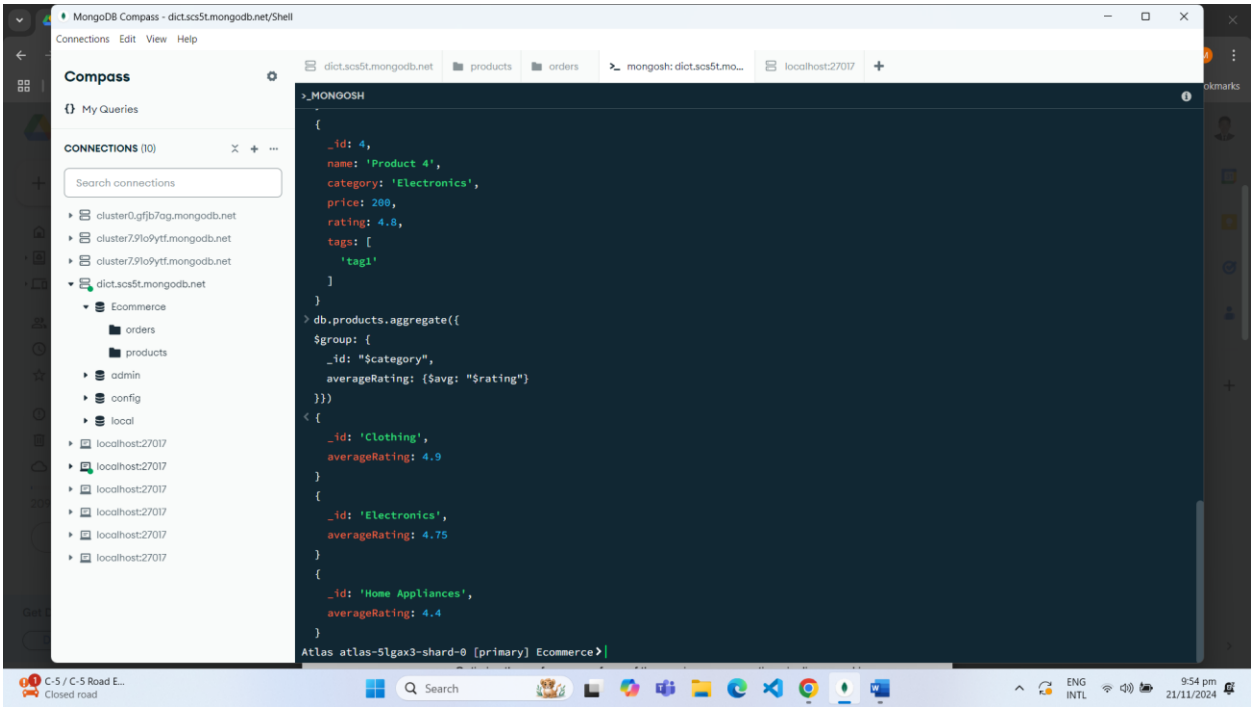
Inserted orders



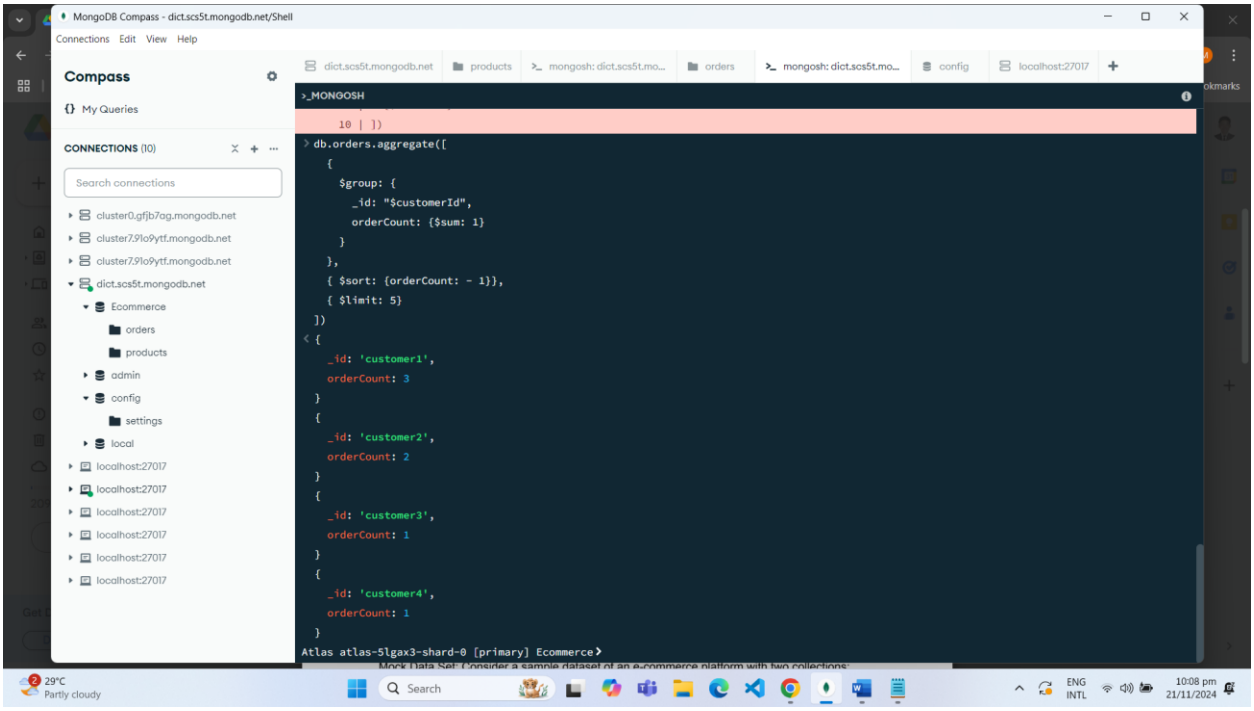
Step 1: Find all products with a price greater than $100 and a rating above 4.5:



Step 2: Find the average rating for products in each category:

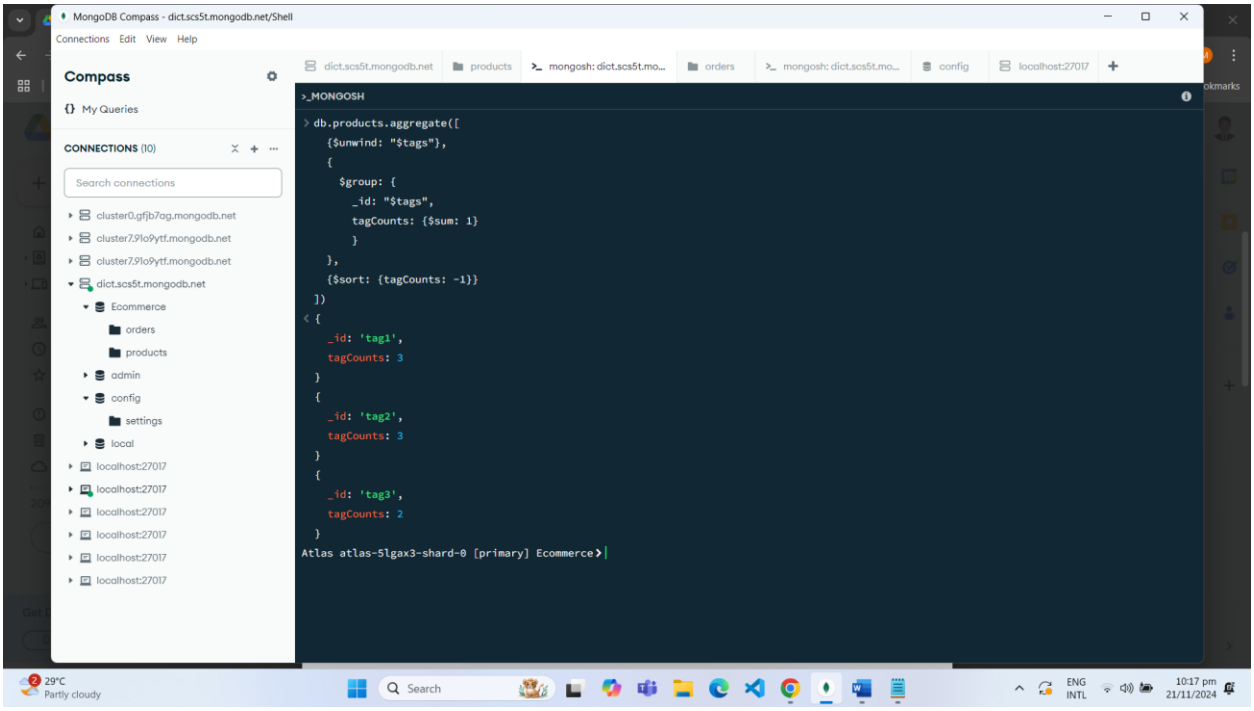


c. Step 3: Find the top 5 customers with the highest number of orders:



Step 4: Find the most common tags across all products:

Write an aggregation pipeline that groups products by tags and counts their occurrences. Sort the results in descending order based on the tag count.


MongoDB Compass - dict.scs5t.mongodb.net/Shell
Connections Edit View Help
Compass
My Queries
CONNECTIONS (10)
Search connections
cluster0.gfjb7ag.mongodb.net
cluster7.9lo9ytf.mongodb.net
cluster7.9lo9ytf.mongodb.net
dict.scs5t.mongodb.net
Ecommerce
orders
products
admin
config
settings
local
localhost:27017
localhost:27017
localhost:27017
localhost:27017
localhost:27017
localhost:27017
dict.scs5t.mongodb.net
products
mongosh: dict.scs5t.mo...
orders
mongosh: dict.scs5t.mo...
config
localhost:27017
>_MONGOSH
> db.products.aggregate([
{$unwind: "$tags"},
{
$group: {
_id: "$tags",
tagCounts: {$sum: 1}
}
},
{$sort: {tagCounts: -1}}
])
< {
_id: 'tag1',
tagCounts: 3
}
{
_id: 'tag2',
tagCounts: 3
}
{
_id: 'tag3',
tagCounts: 2
}
Atlas atlas-5lgax3-shard-0 [primary] Ecommerce>
29°C
Partly cloudy
Search
ENG
INTL
10:17 pm
21/11/2024