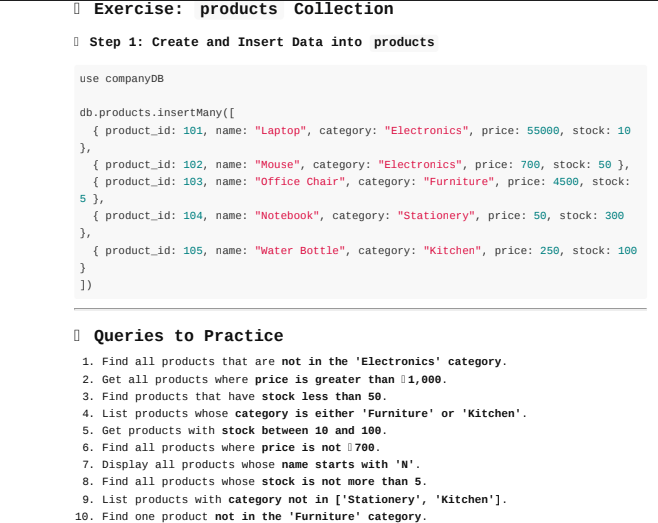
**Assignment:**



**Step 1: Create and Insert Data into products**

db.products.insertMany([

{ product\_id: 101, name: "Laptop", category: "Electronics", price: 55000, stock: 10 },

{ product\_id: 102, name: "Mouse", category: "Electronics", price: 700, stock: 50 },

{ product\_id: 103, name: "Office Chair", category: "Furniture", price: 4500, stock: 5 },

{ product\_id: 104, name: "Notebook", category: "Stationery", price: 50, stock: 300 },

{ product\_id: 105, name: "Water Bottle", category: "Kitchen", price: 250, stock: 100 }

])

**Task 2: Queries to Practice**

**1.Find all products that are not in the Electronics category**

db.products.find({category:{$ne:"Electronics"}})

output:



**2. Get all products where price is greater than 1000**

db.products.find({price:{$gt:1000}})

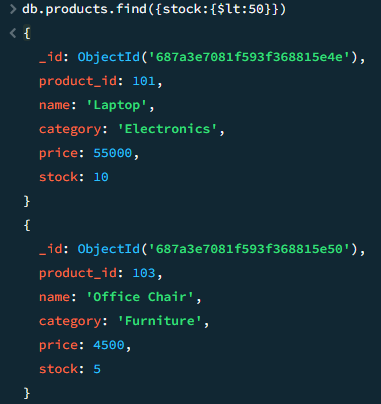
output:



**3. Find products that have stock less than 50.**

db.products.find({stock:{$lt:50}})

output:



**4.List products whose category is either 'Furniture' or 'Kitchen'.**

db.products.find({category:{$in:[“Furniture”,”Kitchen”]}})

Output:



**5. Get products with stock between 10 and 100.**

db.products.find({stock:{$gte:10,$lte:100}})

Output:



**6. Find all products where price is not 700.**

db.products.find({price:{$ne:700}})

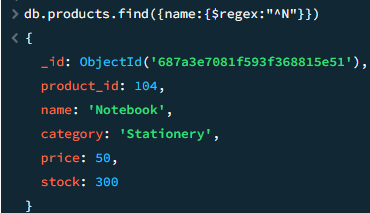
Output:



**7. Display all products whose name starts with 'N'.**

db.products.find({name:{$regex:"^N"}})

Output:



**8. Find all products whose stock is not more than 5.**

db.products.find({stock:{$lte:5}})

Output:



**9. List products with category not in ['Stationery', 'Kitchen'].**

db.products.find({category:{$nin:["Stationery","Kitchen"]}})

Output:



**10. Find one product not in the 'Furniture' category.**

db.products.findOne({category:{$ne:"Furniture"}})

Output:

