**MONGODB QUERIES**

**1.Retrieve all products in the "Electronics" category.**

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

****

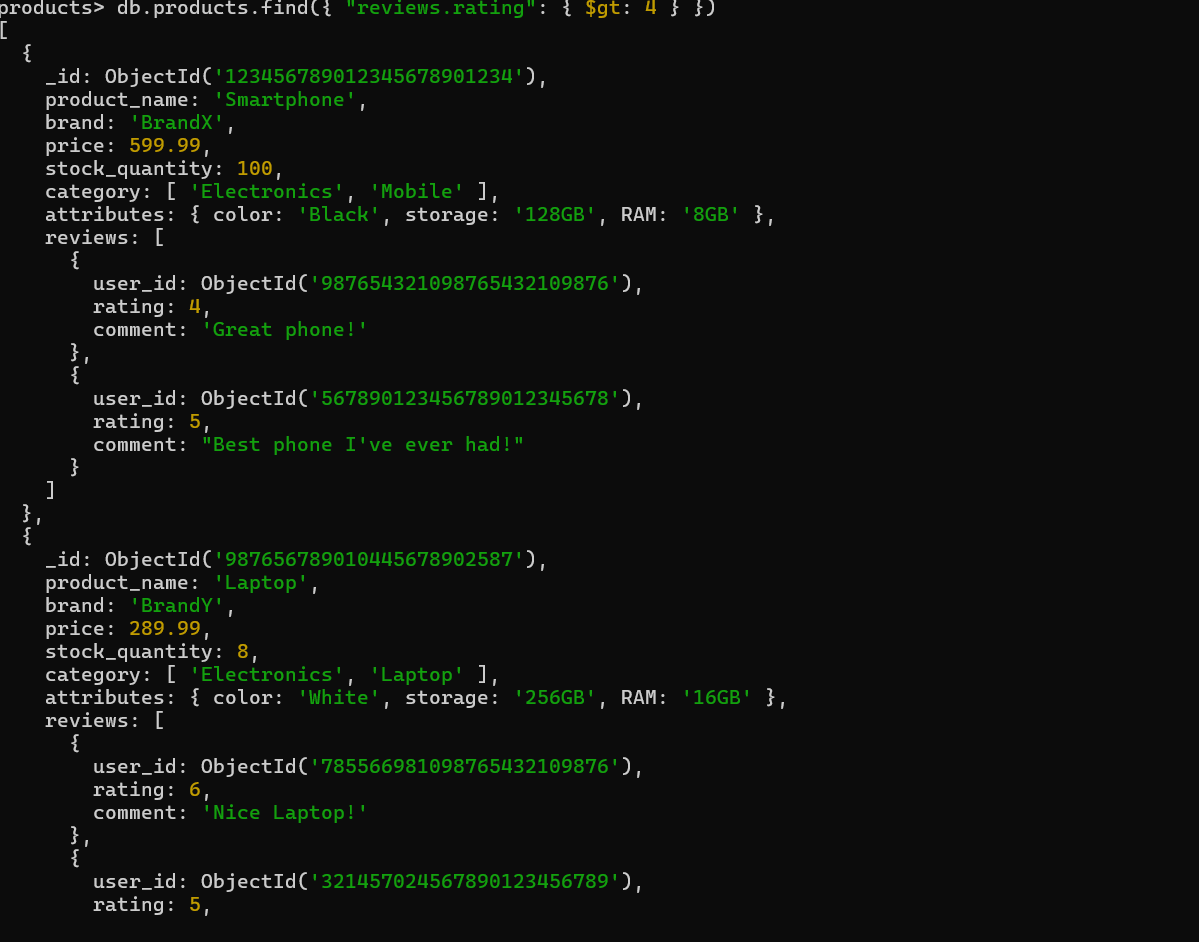
**2.Find all products with a price less than $500.**

**db.products.find({ "price": { $lt: 500 } })**

****

**3.Retrieve all products with a rating greater than 4.**

**db.products.find({ "reviews.rating": { $gt: 4 } })**

****

**4.Find all products with a specific color (e.g., Black).**

**mongodb**

**db.products.find({ "attributes.color": "Black" })**

**A computer screen shot of a black screen

Description automatically generated**

**5.Retrieve all products sorted by price in descending order.**

**mongodb**

**db.products.find().sort({ "price": -1 })**

****

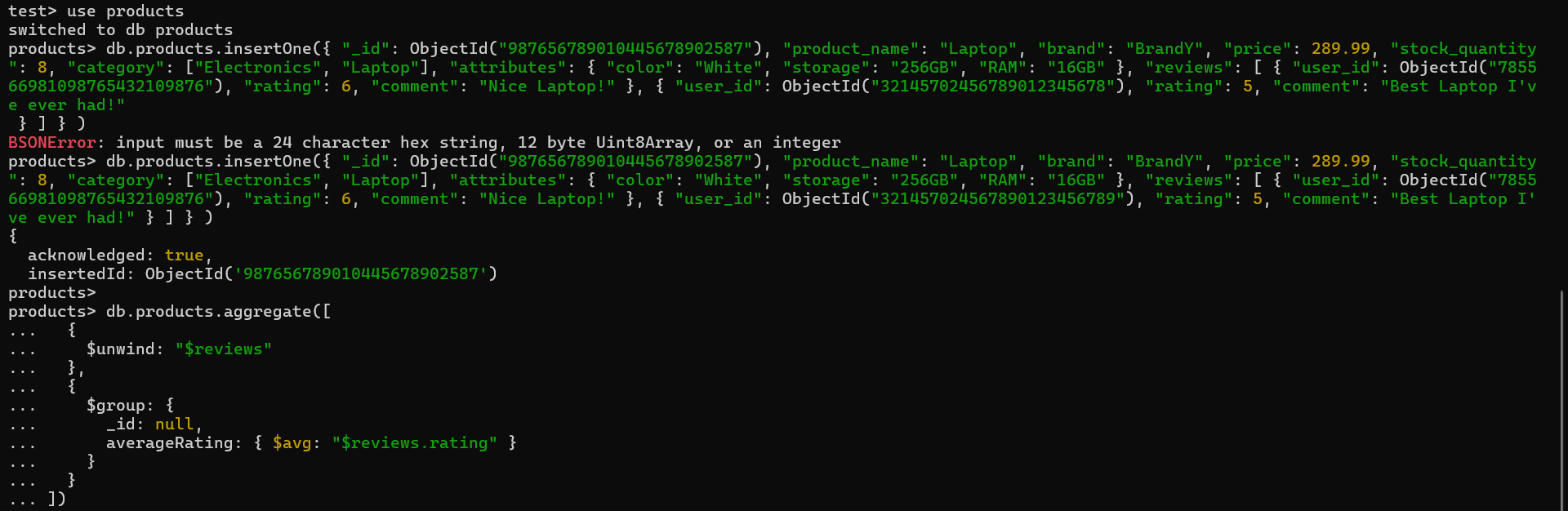
**6.Calculate the average rating of all products.**

**db.products.aggregate([ { $unwind: "$reviews" },**

**{**

**$group: {\_id: null,**

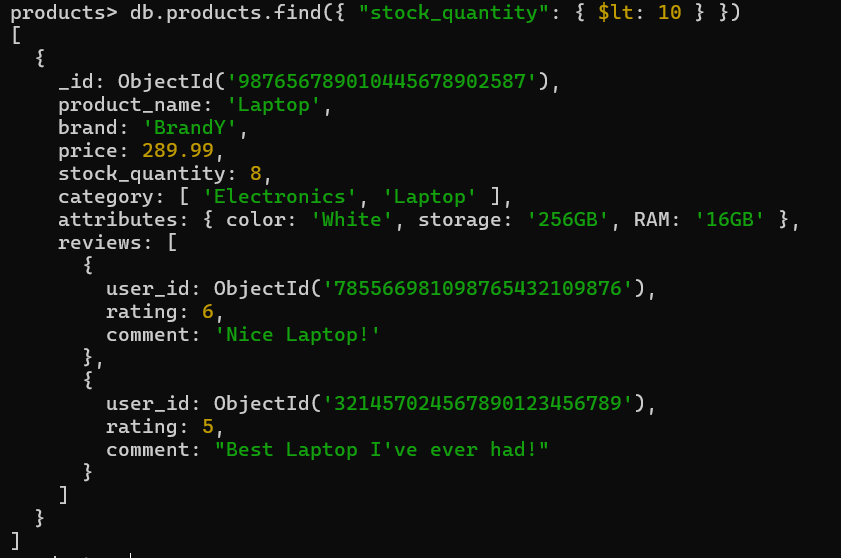
**averageRating: { $avg: "$reviews.rating" } } }])**

****

**7.Find all products where the stock quantity is less than 10.**

**mongodb**

**db.products.find({ "stock\_quantity": { $lt: 10 } })**

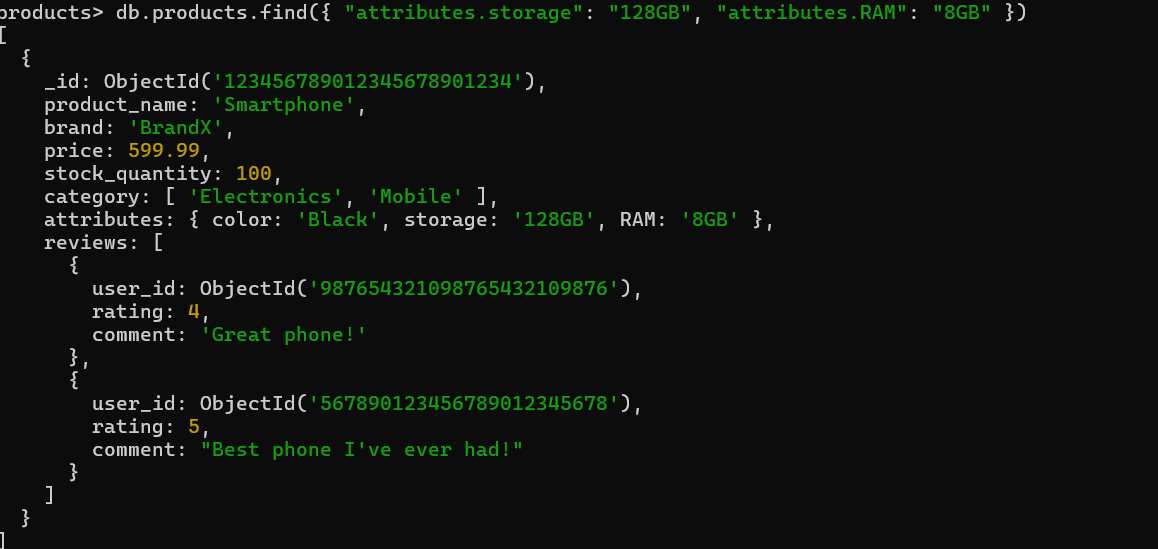
****

**8.Retrieve all products with a specific brand (e.g., BrandX).**

**db.products.find({ "brand": "BrandX" })**

****

**9.Find all products where the storage is 128GB and RAM is 8GB.**

**db.products.find({ "attributes.storage": "128GB", "attributes.RAM": "8GB" })**

**10.Retrieve all products with a specific user's review (e.g., user\_id: ObjectId("987654321098765432109876")).**

**db.products.find({ "reviews.user\_id": ObjectId("987654321098765432109876") })**

**A computer screen with green text

Description automatically generated**

**11.Calculate the total number of reviews for each product.**

**db.products.aggregate([ {**

**$project: {**

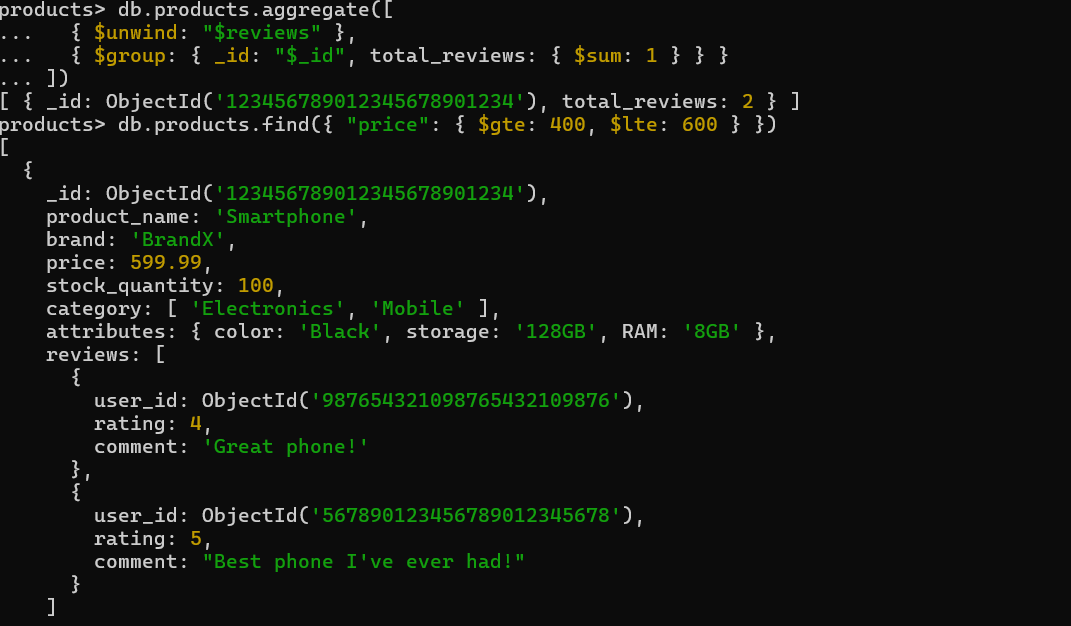
**product\_name: 1,**

**totalReviews: { $size: "$reviews" } } }**

****

**]) 12.Find all products with a price between $400 and $600.**

**db.products.find({ "price": { $gte: 400, $lte: 600 } })**

****

**13.Find all products with a specific category (e.g., "Mobile") and a stock quantity greater than 50.**

**db.products.find({ "category": "Mobile", "stock\_quantity": { $gt: 50 } })**

**A screenshot of a computer

Description automatically generated**

**14.Find the highest rated product.**

**db.products.aggregate([{ $unwind: "$reviews" }, { $sort: { "reviews.rating": -1 } }, { $limit: 1 }])**

**A black screen with green and white text

Description automatically generated**

**15.Retrieve all products sorted by brand in ascending order and then by price in descending order.**

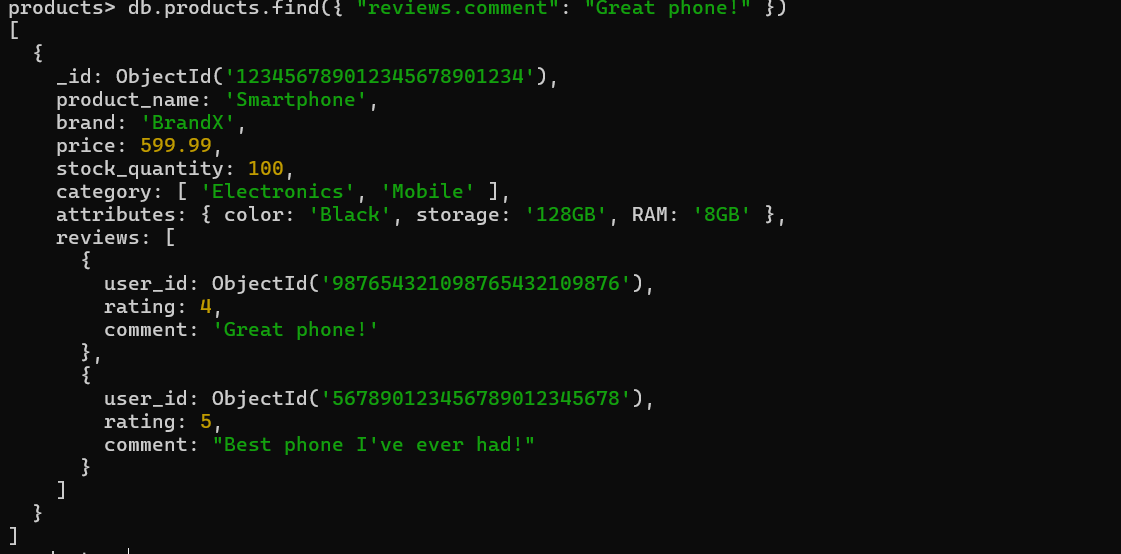
**db.products.find().sort({ "brand": 1, "price": -1 })**

**A screenshot of a computer program

Description automatically generated**

**16.Find all products with a specific comment in their reviews (e.g., "Great phone!").**

**db.products.find({ "reviews.comment": "Great phone!" })**

****