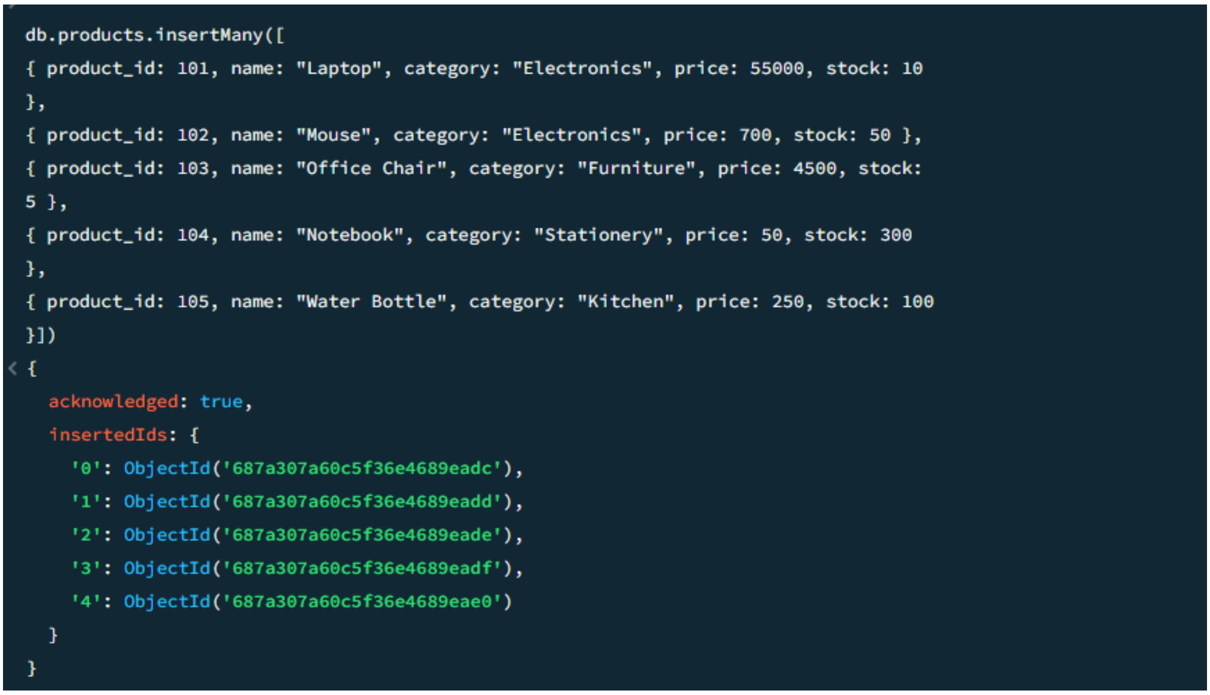
Exercise: products Collection

Step 1: Create and Insert Data into product

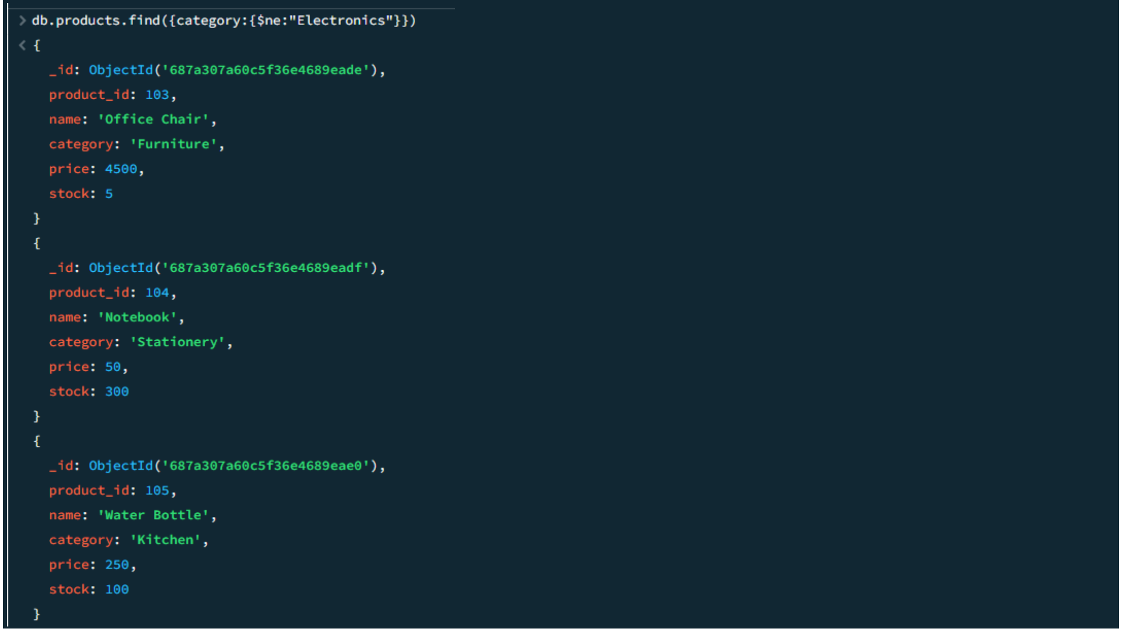


```
db.products.insertMany([
{ product_id: 101, name: "Laptop", category: "Electronics", price: 55000, stock: 10
},
{ product_id: 102, name: "Mouse", category: "Electronics", price: 700, stock: 50 },
{ product_id: 103, name: "Office Chair", category: "Furniture", price: 4500, stock:
5 },
{ product_id: 104, name: "Notebook", category: "Stationery", price: 50, stock: 300
},
{ product_id: 105, name: "Water Bottle", category: "Kitchen", price: 250, stock: 100
}])
< {
    acknowledged: true,
    insertedIds: {
      '0': ObjectId('687a307a60c5f36e4689eadc'),
      '1': ObjectId('687a307a60c5f36e4689eadd'),
      '2': ObjectId('687a307a60c5f36e4689eade'),
      '3': ObjectId('687a307a60c5f36e4689eadf'),
      '4': ObjectId('687a307a60c5f36e4689eae0')
    }
  }
```

Queries to Practice

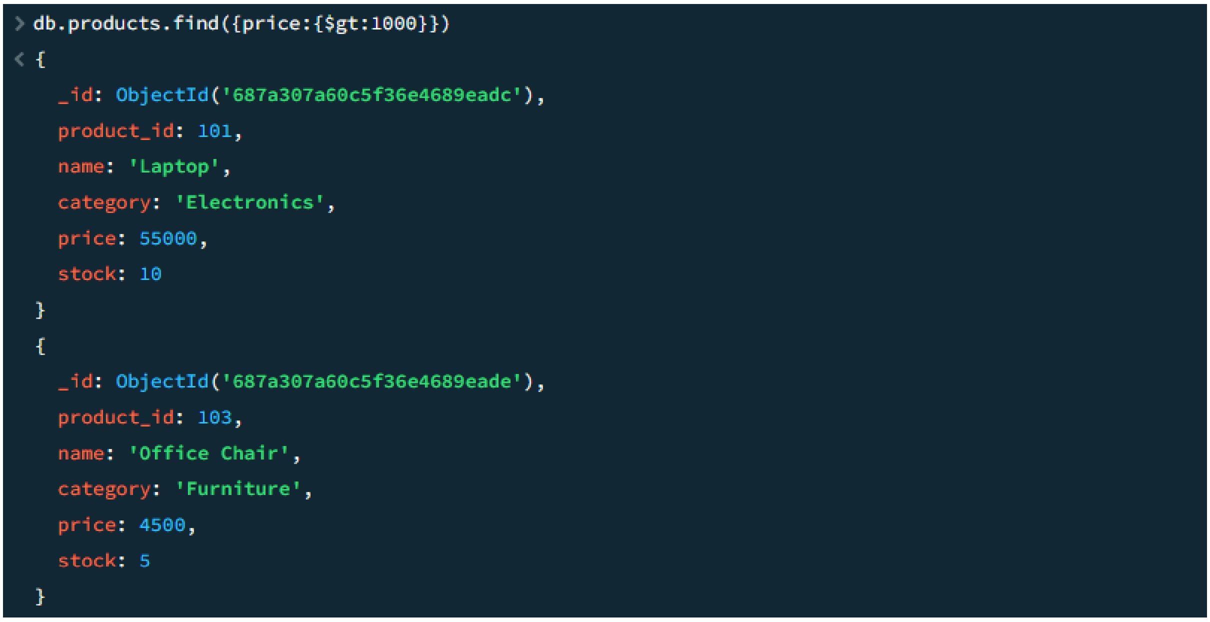
1. Find all products that are not in the 'Electronics' category.

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



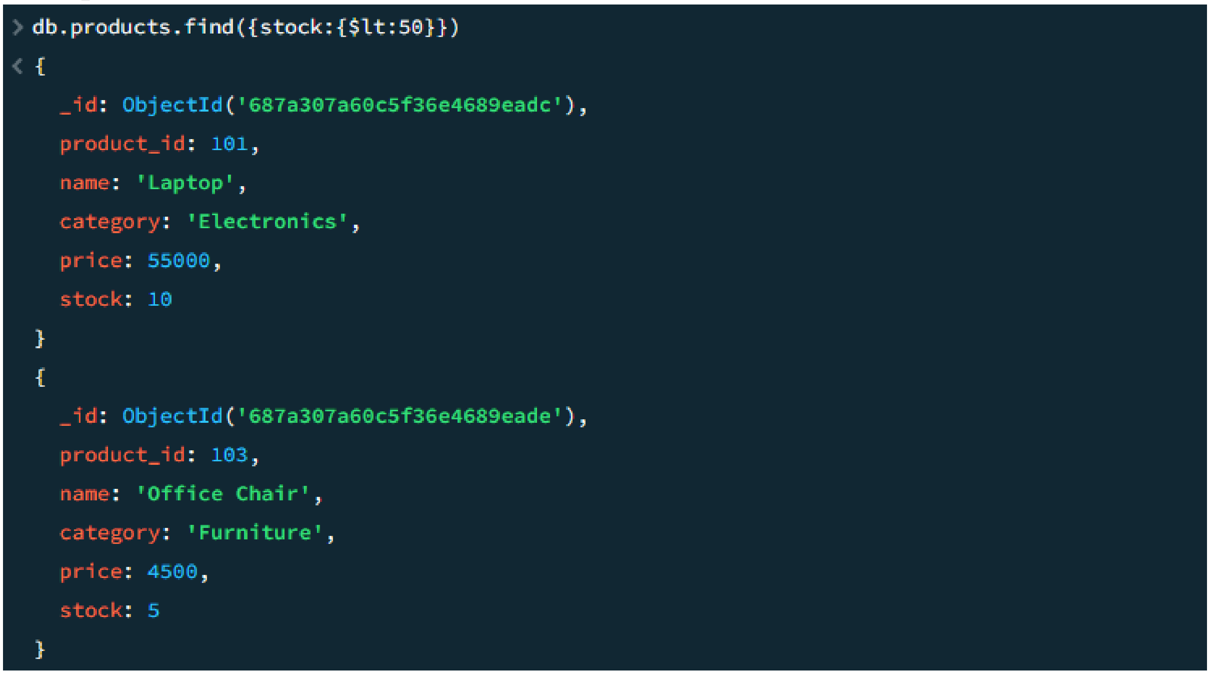
```
> db.products.find({category:{$ne:"Electronics"}})
< {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eadf'),
    product_id: 104,
    name: 'Notebook',
    category: 'Stationery',
    price: 50,
    stock: 300
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eae0'),
    product_id: 105,
    name: 'Water Bottle',
    category: 'Kitchen',
    price: 250,
    stock: 100
  }
```

2. Get all products where price is greater than 1,000.



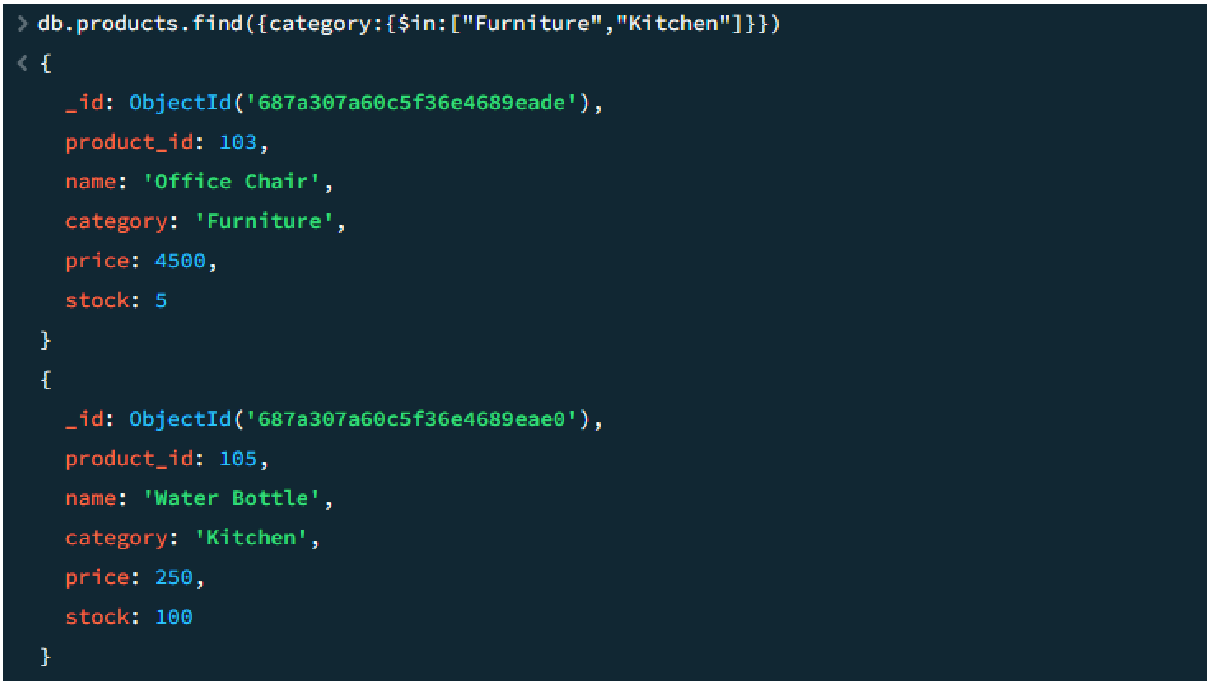
```
> db.products.find({price:{$gt:1000}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
```

3. Find products that have stock less than 50.



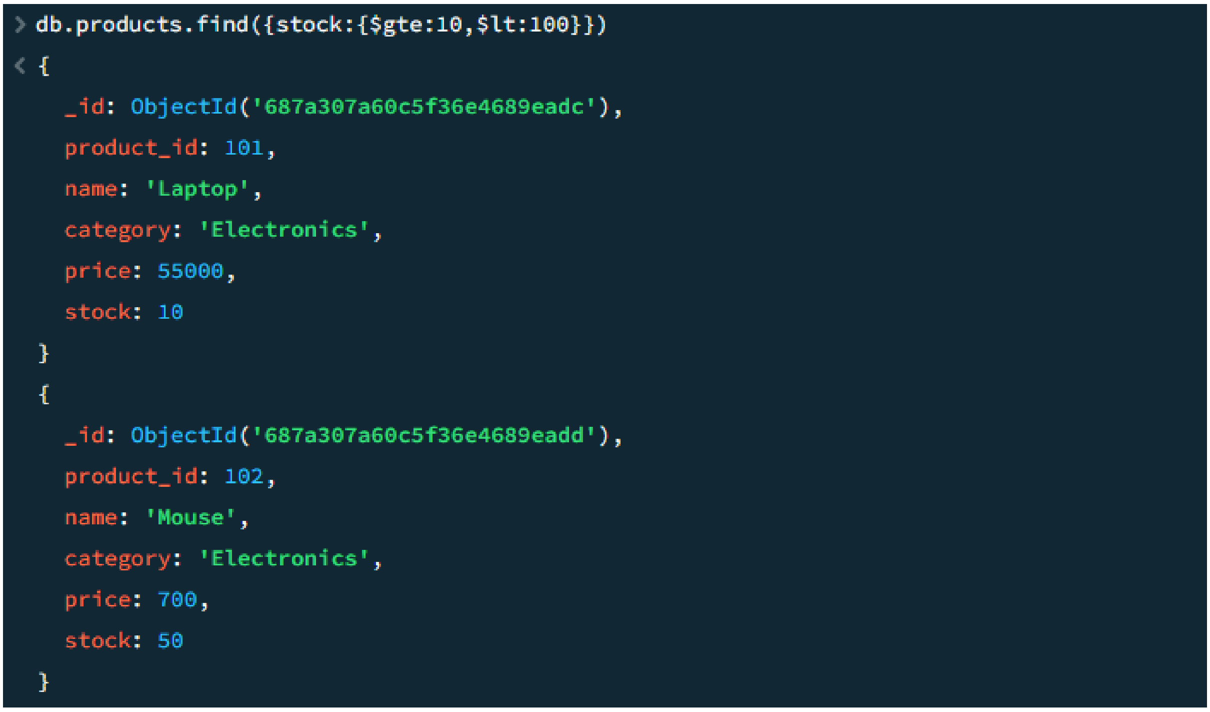
```
> db.products.find({stock:{$lt:50}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
```

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



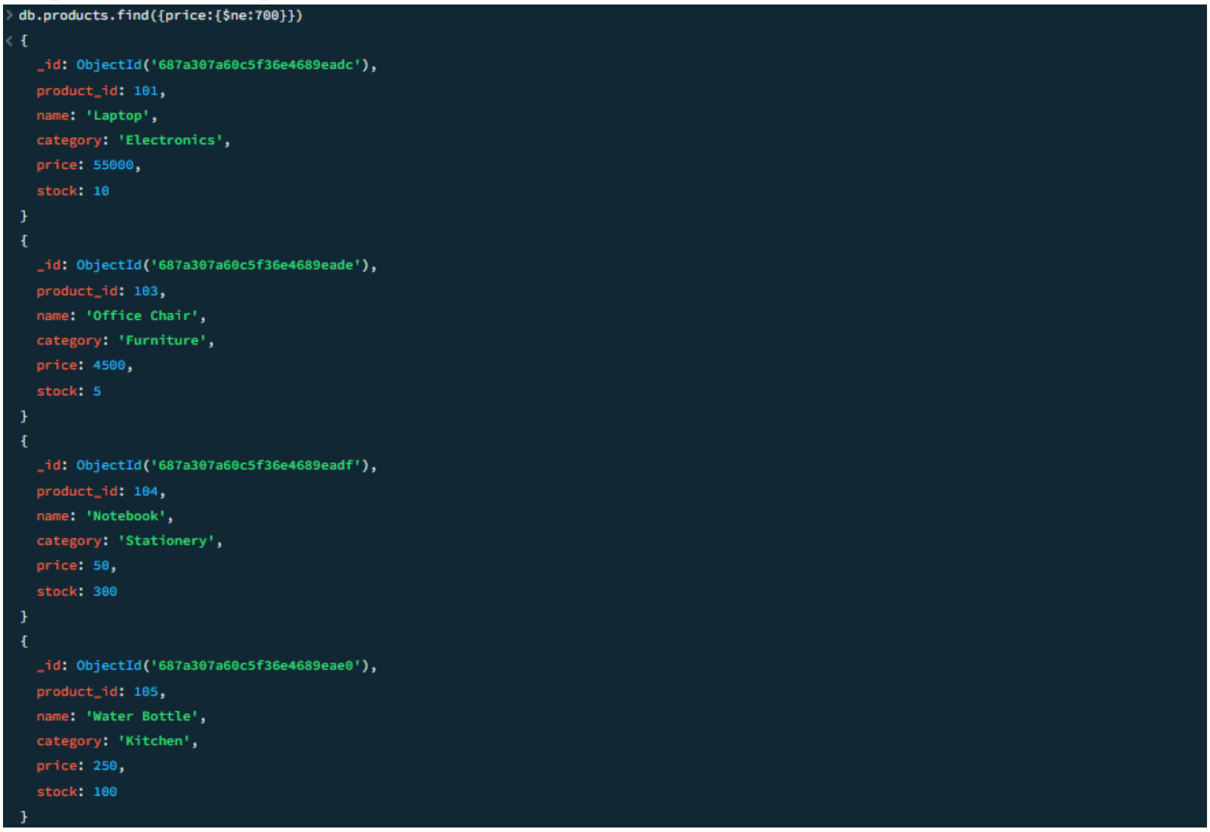
```
> db.products.find({category:{$in:["Furniture","Kitchen"]}})
< {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eae0'),
    product_id: 105,
    name: 'Water Bottle',
    category: 'Kitchen',
    price: 250,
    stock: 100
  }
```

5. Get products with stock between 10 and 100.



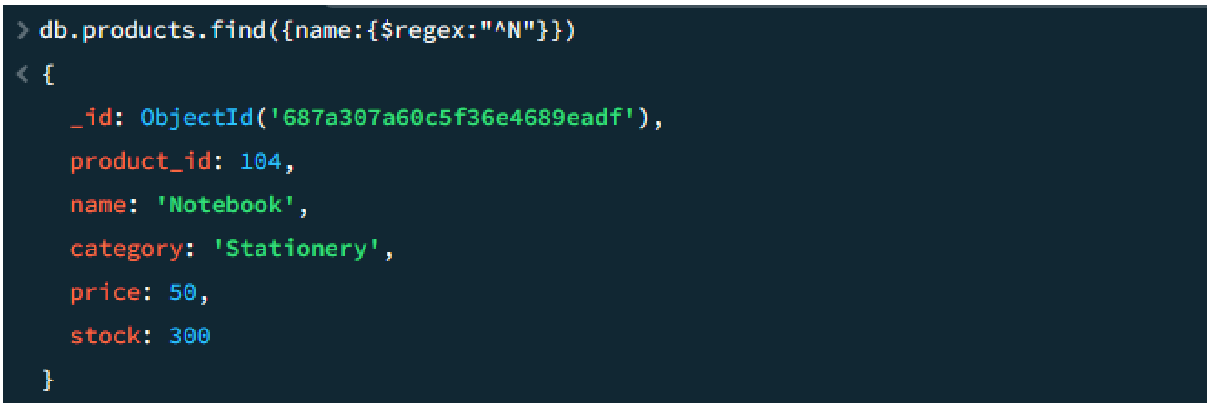
```
> db.products.find({stock:{$gte:10,$lt:100}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eadd'),
    product_id: 102,
    name: 'Mouse',
    category: 'Electronics',
    price: 700,
    stock: 50
  }
```

6. Find all products where price is not 700.
   db.products.find(price:{$ne:700}})



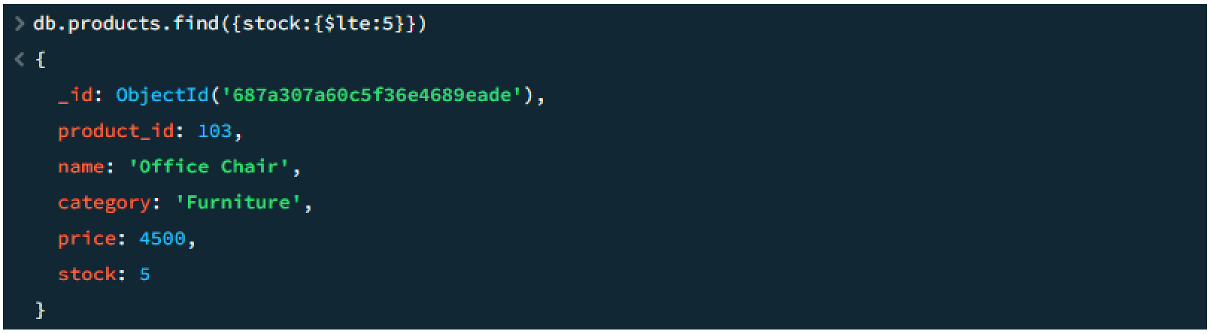
```
> db.products.find({price:{$ne:700}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eadf'),
    product_id: 104,
    name: 'Notebook',
    category: 'Stationery',
    price: 50,
    stock: 300
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eae0'),
    product_id: 105,
    name: 'Water Bottle',
    category: 'Kitchen',
    price: 250,
    stock: 100
  }
```

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



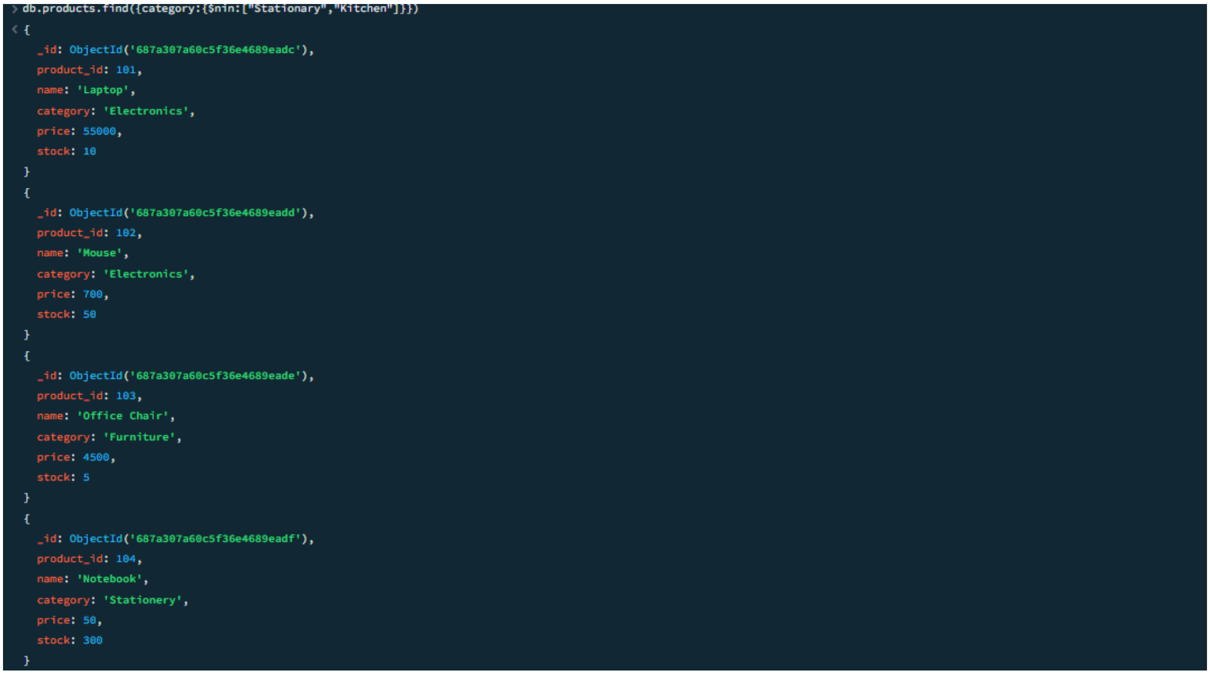
```
> db.products.find({name:{$regex:"^N"}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadf'),
    product_id: 104,
    name: 'Notebook',
    category: 'Stationery',
    price: 50,
    stock: 300
  }
```

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



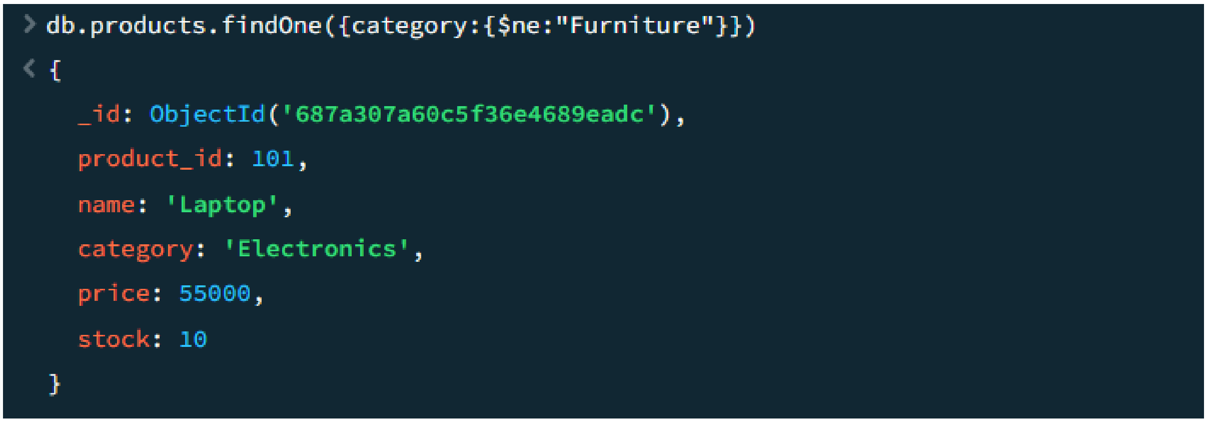
```
> db.products.find({stock:{$lte:5}})
< {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
```

9. List products with category not in ['Stationery', 'Kitchen'].
   db.products.find({category:{$ini:[“Stationary”, ”Kitchen”]}})



```
> db.products.find({category:{$nin:["Stationary","Kitchen"]}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eadd'),
    product_id: 102,
    name: 'Mouse',
    category: 'Electronics',
    price: 700,
    stock: 50
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eade'),
    product_id: 103,
    name: 'Office Chair',
    category: 'Furniture',
    price: 4500,
    stock: 5
  }
  {
    _id: ObjectId('687a307a60c5f36e4689eadf'),
    product_id: 104,
    name: 'Notebook',
    category: 'Stationery',
    price: 50,
    stock: 300
  }
```

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



```
> db.products.findOne({category:{$ne:"Furniture"}})
< {
    _id: ObjectId('687a307a60c5f36e4689eadc'),
    product_id: 101,
    name: 'Laptop',
    category: 'Electronics',
    price: 55000,
    stock: 10
  }
```