MongoDB queries (AmmuPA)

1. Retrieve all products in the "Electronics" category

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

```
db.products.find({ category: "Electronics" })
([{ _id:
ObjectId("123456789012345678901234"){product_name:"Smartp
hone",
brand:"BrandX",
price: 599,
stock_quantity: 100,
category: "Electronics",
attributes: {colour: "Black", storage"128GB",RAM:"8GB"},
Reviews:
[{user_id: ObjectId("987654321098765432198765"),
rating: 4,comment: "Great phone!"},
{user_id: ObjectId("567890123456789012345678"),
rating: 5,comment: "Best phone I've ever had!"}]},
{product_name: "Laptop",
brand: "Xiaomi",
price: 230.99,stock_quantity: 50,
category: ["Electronics", "Laptop"],
attributes: {colour: "White",storage: "256GB",
RAM: "16GB"},
reviews: [{user_id: ObjectId("651654321098765432100531"),
rating: 6,comment: "Amazing...Really Amazing!"},
{user_id: ObjectId("921790123456789012345312"),
rating: 3,comment: "Not Bad!"}]}]);
```

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

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

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

{ _id: ObjectId("987654321012345678901234"),
product_name: "Laptop",
 brand: "Xiaomi",
price: 230.99,
 stock_quantity: 50,
category: ["Electronics", "Laptop"],
 attributes: { colour: "White", storage: "256GB", RAM:
"16GB" },
 reviews: [ { user_id:
ObjectId("651654321098765432100531"),
 rating: 6,
comment: "Amazing...Really Amazing!" },
{ user_id: ObjectId("921790123456789012345312"),
rating: 3,
 comment: "Not Bad!" } ] }
```

3. Retrieve all products with a rating greater than 4

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

```
db.products.find({ "reviews.rating": { $gt: 4 } })
{  product_name: "Laptop",
   brand: "Xiaomi",
   price: 230.99,
   stock_quantity: 50,
   category: ["Electronics", "Laptop"],
   attributes: { colour: "White", storage: "256GB",
RAM: "16GB" },
   reviews: [
     { user_id:
ObjectId("65165432109876543210531"), rating: 6,
comment: "Amazing...Really Amazing!" } ]}
{product_name: "Camera",
   brand: "Canon",
   price: 899.99,
   stock_quantity: 20,
   category: ["Electronics", "Photography"],
   attributes: { colour: "Black", resolution:
"24MP" },
   reviews: [ {
user_id:ObjectId("8765432109876543210987 65"),
rating: 5, comment: "Excellent camera!" }]}
```

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

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

```
db.products.find({ "attributes.color": "Black" })
{product_name: "Smartphone",
  brand: "BrandX",
  price: 599.99,
  stock_quantity: 100,
  category: ["Electronics", "Mobile"],
  attributes: { colour: "Black", storage: "128GB",
RAM: "8GB" },
  reviews: [ {
user_id:ObjectId("9876543210987654321098 76"),
rating: 4, comment: "Great phone!" },{ user_id:
ObjectId("5678901234567890 12345678"), rating: 5,
comment: "Best phone I've ever had!" }]}
{
 product_name: "Smartwatch",
  brand: "BrandZ",
  price: 199.99,
  stock_quantity: 80,
  category: ["Electronics", "Wearable"],
  attributes: { colour: "Black", features: ["Heart
Rate", "GPS"] },
  reviews: [{ user_id:
ObjectId("3456789012 34567890123456"), rating: 4,
comment: "Very useful!" }]}
```

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

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

```
db.products.find().sort({ price: -1 })
{
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599.99,
  stock_quantity: 100,
  category: ["Electronics", "Mobile"],
  attributes: {colour: "Black",storage:
"128GB",RAM: "8GB" },
  reviews: [
  {user_id: ObjectId("98765432109876543210 9876"),
   rating: 4,comment: "Great phone!"},
 {user_id: ObjectId("567890123456789012345678"),
rating: 5,comment: "Best phone I've ever had!"} ]}
{
  product_name: "Laptop",
  brand: "Xiaomi",
  price: 230.99,
  stock_quantity: 50,
  category: ["Electronics", "Laptop"],
  attributes: {colour: "White", storage:
"256GB",RAM:"16GB"}, reviews: [
  {user_id: ObjectId("65165432109876543210 0531"),
 rating: 6,comment: "Amazing...Really Amazing!" },
 { user_id: ObjectId("921790123456789012345312"),
      rating: 3,comment: "Not Bad!" } ]}
```

6. Calculate the average rating of all products

```
db.products.aggregate([
  { $unwind: "$reviews" },
  { $group: { _id: null, averageRating: { $avg:
"$reviews.rating" } } }
])
```

```
{
  id: null,
  averageRating: 4.5
}
```

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

db.products.find({ stock_quantity: { $lt: 10 } }).pretty()

```
db.products.find({ stock_quantity: { $lt: 10 } })
{
  product_name: "Tablet",
  brand: "BrandY",
  price: 299.99,
  stock_quantity: 5,
  category: ["Electronics", "Mobile"],
  attributes: {
    colour: "Silver",
    storage: "64GB",
    RAM: "4GB"},
  reviews: [
    {
      user_id: ObjectId("5678901234567890123456789"),
      rating: 4,
      comment: "Good tablet!"} ]}
```

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

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

```
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599.99,
  stock_quantity: 100,
  category: ["Electronics", "Mobile"],
  attributes: {
    colour: "Black",
    storage: "128GB",
    RAM: "8GB"
  },
  reviews: [
    {
      user_id: ObjectId("987654321098765432109876"),
      rating: 4,
      comment: "Great phone!"
    },
    {
      user_id: ObjectId("567890123456789012345678"),
      rating: 5,
      comment: "Best phone I've ever had!"
    }
  ]
}
```

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

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

```
db.products.find({ "attributes.storage": "128GB",
"attributes.RAM": "8GB" })
{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599.99,
  stock_quantity: 100,
  category: ["Electronics", "Mobile"],
  attributes: {
    colour: "Black",
    storage: "128GB",
    RAM: "8GB"
  },
  reviews: [
    {
      user_id: ObjectId("987654321098765432109876"),
      rating: 4,
      comment: "Great phone!"
    },
    {
      user_id: ObjectId("567890123456789012345678"),
      rating: 5,
      comment: "Best phone I've ever had!"
    }
  ]
}
```

10. Retrieve all products with a specific user's review (e.g., user_id: ObjectId("98765432109876543210**9**876"))

db.products.find({ "reviews.user_id": ObjectId("98765432109876543210**9**876") }).pretty()

```
{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599
```

11.Calculate the total number of reviews for each product
db.products.find({ "reviews.comment": "Great phone!"
}).pretty()

db.products.aggregate([
 {$project: {
   product_name: 1,
   total_reviews: { $size: "$reviews" } }}])

```
db.products.aggregate([
  {
    $project: {
      product_name: 1,
      total_reviews: { $size: "$reviews" }
    }
  }
])

{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  total_reviews: 2
}
{
  _id: ObjectId("987654321012345678901234"),
  product_name: "Laptop",
  total_reviews: 2
}
```

12. Find all products with a price between $400 and $600
db.products.find({ price: { $gte: 400, $lte: 600 } }).pretty()

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

{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599,
  stock_quantity: 100,
  category: "Electronics",
  attributes: { colour: "Black", storage: "128GB",
RAM: "8GB" },
  reviews: [
    { user_id: ObjectId("987654321098765432198765"),
rating: 4, comment: "Great phone!" },
    { user_id: ObjectId("567890123456789012345678"),
rating: 5, comment: "Best phone I've ever had!" }
  ]
}
```

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 } }).pretty()

```
db.products.find({ category: "Mobile", stock_quantity: {
$gt: 50 } })

{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599,
  stock_quantity: 100,
  category: "Electronics",
  attributes: { colour: "Black", storage: "128GB", RAM:
"8GB" },
  reviews: [
    { user_id: ObjectId("987654321098765432198765"),
rating: 4, comment: "Great phone!" },
    { user_id: ObjectId("567890123456789012345678"),
rating: 5, comment: "Best phone I've ever had!" }
  ]
}
```

14. Find the highest rated product

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

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

{
  _id: ObjectId("9876543210123456789011234"),
  product_name: "Laptop",
  reviews: { rating: 6 }
}
```

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 }).pretty()

```
db.products.find().sort({ brand: 1, price: -1 })
{
  _id: ObjectId("12345678901234567890123 4"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599,
  stock_quantity: 100,
  category: "Electronics",
  attributes: { color: "Black", storage: "128GB", RAM: "8GB" },
  reviews: [
    { user_id: ObjectId("98765432109876543219876 5"), rating: 4, comment:
"Great phone!" },
    { user_id: ObjectId("56789012345678901234567 8"), rating: 5, comment:
"Best phone I've ever had!" }
  ]}
{
  _id: ObjectId("34567890123456789012345 7"),
  product_name: "Smartwatch",
  brand: "BrandZ",
  price: 199.99,
  stock_quantity: 80,
  category: ["Electronics", "Wearable"],
  attributes: { colour: "Black", features: ["Heart Rate", "GPS"] },
  reviews: [
    { user_id: ObjectId("34567890123456789012345 6"), rating: 4, comment:
"Very useful!" }
  ]
}
{
  _id: ObjectId("98765432101234567890123 4"),
  product_name: "Laptop",
  brand: "Xiaomi",
  price: 230.99,
  stock_quantity: 50,
  category: ["Electronics", "Laptop"],
  attributes: { colour: "White", storage: "256GB", RAM: "16GB" },
  reviews: [
    { user_id: ObjectId("65165432109876543210053 1"), rating: 6, comment:
"Amazing...Really Amazing!" },
    { user_id: ObjectId("92179012345678901234531 2"), rating: 3, comment:
"Not Bad!" }]}
```

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

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

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

{
  _id: ObjectId("123456789012345678901234"),
  product_name: "Smartphone",
  brand: "BrandX",
  price: 599,
  stock_quantity: 100,
  category: "Electronics",
  attributes: { color: "Black", storage: "128GB", RAM: "8GB"
},
  reviews: [
    { user_id: ObjectId("987654321098765432198765"), rating:
4, comment: "Great phone!" },
    { user_id: ObjectId("567890123456789012345678"), rating:
5, comment: "Best phone I've ever had!" }
  ]
}
```