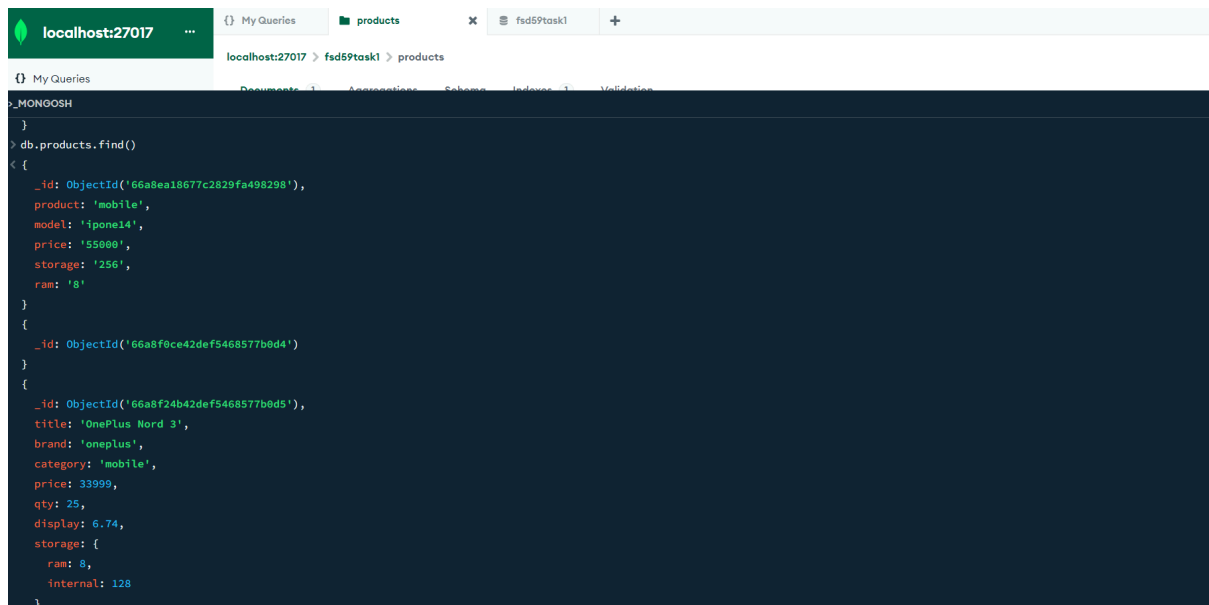


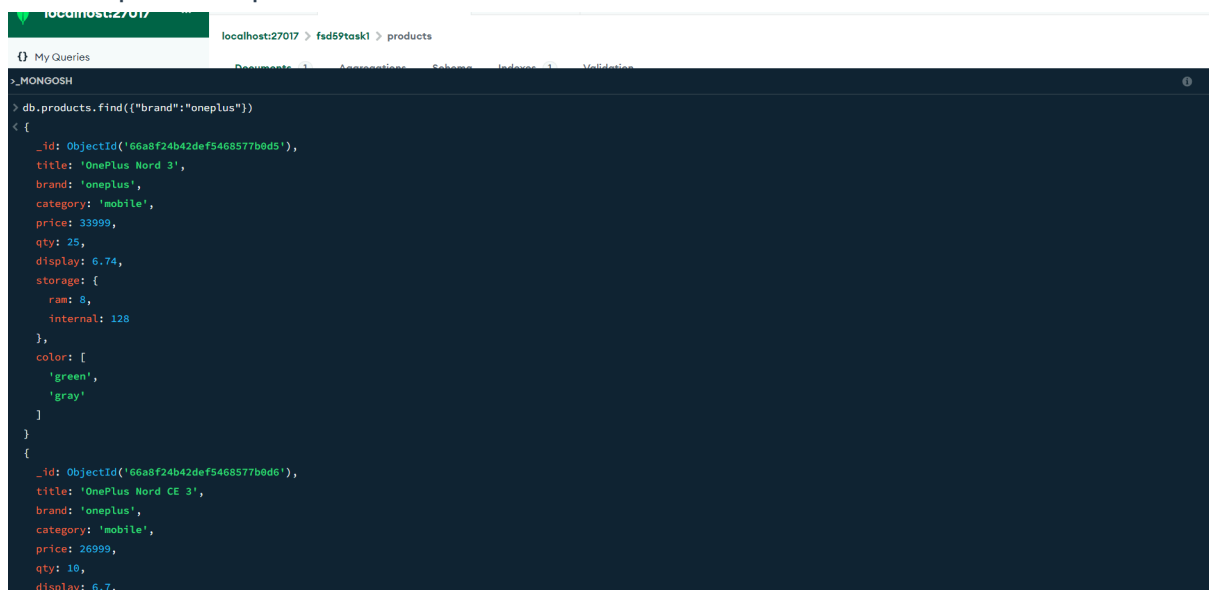
1. Find all the information about each products.



The screenshot shows the MongoDB Shell interface. The top bar indicates the connection is to localhost:27017. The current database is 'products'. The command prompt shows the command `db.products.find()` has been executed. The output displays three documents from the 'products' collection. The first document is a mobile phone with a price of 55000. The second document is a OnePlus Nord 3 with a price of 33999. The third document is a OnePlus Nord CE 3 with a price of 26999.

```
> db.products.find()
{
  "_id": ObjectId("66a8ea18677c2829fa498298"),
  "product": "mobile",
  "model": "ipone14",
  "price": "55000",
  "storage": "256",
  "ram": "8"
}
{
  "_id": ObjectId("66a8f0ce42def5468577b0d4")
}
{
  "_id": ObjectId("66a8f24b42def5468577b0d5"),
  "title": "OnePlus Nord 3",
  "brand": "oneplus",
  "category": "mobile",
  "price": 33999,
  "qty": 25,
  "display": 6.74,
  "storage": {
    "ram": 8,
    "internal": 128
  },
  "color": [
    "green",
    "gray"
  ]
}
{
  "_id": ObjectId("66a8f24b42def5468577b0d6"),
  "title": "OnePlus Nord CE 3",
  "brand": "oneplus",
  "category": "mobile",
  "price": 26999,
  "qty": 10,
  "display": 6.7,
```

2. The following example selects from the products collection all documents where the brand equals "oneplus".



The screenshot shows the MongoDB Shell interface. The command prompt shows the command `db.products.find({"brand": "oneplus"})` has been executed. The output displays two documents from the 'products' collection, both of which are OnePlus phones. The first document is a OnePlus Nord 3 with a price of 33999. The second document is a OnePlus Nord CE 3 with a price of 26999.

```
> db.products.find({"brand": "oneplus"})
{
  "_id": ObjectId("66a8f24b42def5468577b0d5"),
  "title": "OnePlus Nord 3",
  "brand": "oneplus",
  "category": "mobile",
  "price": 33999,
  "qty": 25,
  "display": 6.74,
  "storage": {
    "ram": 8,
    "internal": 128
  },
  "color": [
    "green",
    "gray"
  ]
}
{
  "_id": ObjectId("66a8f24b42def5468577b0d6"),
  "title": "OnePlus Nord CE 3",
  "brand": "oneplus",
  "category": "mobile",
  "price": 26999,
  "qty": 10,
  "display": 6.7,
```

3. The following example retrieves all documents in the products collection where the category equals "mobile" and qty is less than (\$lt) 20.

```

      storage: {
        ram: 12,
        internal: 256
      },
      color: [
        'aqua'
      ]
    }
  }
}
db.products.find(
  {
    "category": "mobile",
    "qty": { $lt: 20 }
  }
)
{
  _id: ObjectId('66a8f24b42def5468577b0d6'),
  title: 'OnePlus Nord CE 3',
  brand: 'oneplus',
  category: 'mobile',
  price: 26990,
  qty: 10,
  display: 6.7,
  storage: {
    ram: 12,
    internal: 256
  },
  color: [
    'aqua'
  ]
}

```

4. The following example retrieves all documents in the collection where the category equals "mobile" **or** qty is less than (\$lt) 20.

```

> _MONGOSH
> db.products.find(
  {
    $or: [
      { "category": "mobile" },
      { "qty": { $lt: 20 } }
    ]
  }
)
< {
  _id: ObjectId('66a8f24b42def5468577b0d5'),
  title: 'OnePlus Nord 3',
  brand: 'oneplus',
  category: 'mobile',
  price: 33990,
  qty: 25,
  display: 6.74,
  storage: {
    ram: 8,
    internal: 128
  },
  color: [
    'green',
    'gray'
  ]
}
{
  _id: ObjectId('66a8f24b42def5468577b0d6'),
  title: 'OnePlus Nord CE 3',

```

5. the following query selects all documents where the field storage equals the document { "ram": 8, "internal": 64 }.

```

}
db.products.find({
  "storage":{"ram": 8, "internal": 64 }
})
{
  _id: ObjectId('66a8f24b42def5468577b0dd'),
  title: 'Apple iPad Air',
  brand: 'apple',
  category: 'tab',
  price: 68400,
  qty: 2,
  display: 10.9,
  storage: {
    ram: 8,
    internal: 64
  },
  color: [
    'grey',
    'pink'
  ]
}

```

6.The following example selects all documents where the field ram nested in the storage field equals "8".

```

db.products.find({
  "storage.ram":8
})
{
  _id: ObjectId('66a8f24b42def5468577b0d5'),
  title: 'OnePlus Nord 3',
  brand: 'oneplus',
  category: 'mobile',
  price: 33999,
  qty: 25,
  display: 6.74,
  storage: {
    ram: 8,
    internal: 128
  },
  color: [
    'green',
    'gray'
  ]
}
{
  _id: ObjectId('66a8f24b42def5468577b0dc'),
  title: 'Xiaomi Pad 6',
  brand: 'xiaomi',
  category: 'tab',
  price: 33999,
  qty: 25,
  display: 11.0,
  storage: {
    ram: 8,
    internal: 128
  },
  color: [
    'green',
    'gray'
  ]
}

```

7.The following query uses the less than operator (**\$lt**) on the field **internal** embedded in the **storage** field.

```

    'grey',
    'pink'
  ]
}
> db.products.find({
  "storage.internal": { $lt: 80 }
})
< {
  _id: ObjectId('66a8f24b42def5468577b0dd'),
  title: 'Apple iPad Air',
  brand: 'apple',
  category: 'tab',
  price: 68400,
  qty: 2,
  display: 10.9,
  storage: {
    ram: 8,
    internal: 64
  },
  color: [
    'grey',
    'pink'
  ]
}
fsd59task1>

```

8.The following example queries for all documents where the field tags value is an array with exactly two elements, "silver" and "black", in the specified order.

```

> db.products.find({
  "color": ['silver','black']
});
< {
  _id: ObjectId('66a8f24b42def5468577b0d8'),
  title: 'Samsung Galaxy M14',
  brand: 'samsung',
  category: 'mobile',
  price: 14990,
  qty: 8,
  display: 6.58,
  storage: {
    ram: 4,
    internal: 128
  },
  color: [
    'silver',
    'black'
  ]
}
{
  _id: ObjectId('66a8f24b42def5468577b0da'),
  title: 'Vivo T2x',
  brand: 'vivo',
  category: 'mobile',
  price: 13999,

```

9.The following example queries for all documents where color is an array that contains the string "grey" as one of its elements.

```

}
> db.products.find({
  "color": "grey"
});
{
  _id: ObjectId('66a8f24b42def5468577b0dc'),
  title: 'Xiaomi Pad 6',
  brand: 'xiaomi',
  category: 'tab',
  price: 28999,
  qty: 45,
  display: 11,
  storage: {
    ram: 8,
    internal: 256
  },
  color: [
    'grey',
    'black',
    'blue'
  ]
}
{
  _id: ObjectId('66a8f24b42def5468577b0dd'),
  title: 'Apple iPad Air',
  brand: 'apple',
  category: 'tablet'
}

```

10.The following selects documents where the array color has 3 elements.

```

> db.products.find({
  "color": { $size: 3 }
});
{
  _id: ObjectId('66a8f24b42def5468577b0d7'),
  title: 'Samsung Galaxy M34',
  brand: 'samsung',
  category: 'mobile',
  price: 18999,
  qty: 72,
  display: 6.4,
  storage: {
    ram: 6,
    internal: 128
  },
  color: [
    'black',
    'dark blue',
    'blue'
  ]
}
{
  _id: ObjectId('66a8f24b42def5468577b0dc'),
  title: 'Xiaomi Pad 6',
  brand: 'xiaomi',
  category: 'tab',

```