

Mongo bd task

db setup

```
C:\Users\Alphans>mongosh
Current Mongosh Log ID: 65d88a7152cb5fb28e5c732c
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.1.5
Using MongoDB:      7.0.5
Using Mongosh:       2.1.5

For mongosh info see: https://docs.mongodb.com/mongosh-shell/

-----
  The server generated these startup warnings when booting
  2024-02-22T22:39:23.039+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----

test> db.createCollection("product")
{ ok: 1 }
test> use product
switched to db product
product>
```

1. Find all the information about each products

db.product.find()

```
test> db.product.find()
[
  {
    _id: ObjectId('65d893b4d81668e39b665e04'),
    id: '1',
    product_name: 'Intelligent Fresh Chips',
    product_price: 655,
    product_material: 'Concrete',
    product_color: 'mint green'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e05'),
    id: '2',
    product_name: 'Practical Fresh Sausages',
    product_price: 911,
    product_material: 'Cotton',
    product_color: 'indigo'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e06'),
    id: '3',
    product_name: 'Refined Steel Car',
    product_price: 690,
    product_material: 'Rubber',
    product_color: 'gold'
  },
]
```

2. Find the product price which are between 400 to 800

db.product.find({product_price:{ \$gt:400,\$lt:800}})

```

test> db.product.find({product_price:{ $gt:400,$lt:800}})
[
  {
    _id: ObjectId('65d893b4d81668e39b665e04'),
    id: '1',
    product_name: 'Intelligent Fresh Chips',
    product_price: 655,
    product_material: 'Concrete',
    product_color: 'mint green'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e06'),
    id: '3',
    product_name: 'Refined Steel Car',
    product_price: 690,
    product_material: 'Rubber',
    product_color: 'gold'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e07'),
    id: '4',
    product_name: 'Gorgeous Plastic Pants',
    product_price: 492,
    product_material: 'Soft',
    product_color: 'plum'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e09'),
    id: '6',
    product_name: 'Awesome Wooden Towels',
    product_price: 474,
    product_material: 'Plastic',
    product_color: 'orange'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e0a'),
    id: '7',
    product_name: 'Practical Soft Shoes',
    product_price: 500,
    product_material: 'Rubber',
    product_color: 'pink'
  }
]
test> 

```

3. Find the product price which are not between 400 to 600

```
db.product.find({$nor : [{product_price: {$gt : 400,$lt : 600}}]});
```

```
test> db.product.find({$nor : [{product_price: {$gt : 400,$lt : 600}}]});
[
  {
    _id: ObjectId('65d893b4d81668e39b665e04'),
    id: '1',
    product_name: 'Intelligent Fresh Chips',
    product_price: 655,
    product_material: 'Concrete',
    product_color: 'mint green'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e05'),
    id: '2',
    product_name: 'Practical Fresh Sausages',
    product_price: 911,
    product_material: 'Cotton',
    product_color: 'indigo'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e06'),
    id: '3',
    product_name: 'Refined Steel Car',
    product_price: 690,
    product_material: 'Rubber',
    product_color: 'gold'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e08'),
    id: '5',
    product_name: 'Sleek Cotton Chair',
    product_price: 33,
    product_material: 'Fresh',
    product_color: 'black'
  },
  {

```

4. List the four product which are greater than 500 in price

```
db.product.find({product_price : {$gt:500}}).limit(4)
```


db.product.find({id:"10"})

```
test> db.product.find({id:"10"})
[
  {
    _id: ObjectId('65d893b4d81668e39b665e0d'),
    id: '10',
    product_name: 'Generic Wooden Pizza',
    product_price: 84,
    product_material: 'Frozen',
    product_color: 'indigo'
  }
]
```

7. Find only the product name and product material

db.product.find({}, {product_name:true, product_material:true})

```
test> db.product.find({}, {product_name:true, product_material:true})
[
  {
    _id: ObjectId('65d893b4d81668e39b665e04'),
    product_name: 'Intelligent Fresh Chips',
    product_material: 'Concrete'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e05'),
    product_name: 'Practical Fresh Sausages',
    product_material: 'Cotton'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e06'),
    product_name: 'Refined Steel Car',
    product_material: 'Rubber'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e07'),
    product_name: 'Gorgeous Plastic Pants',
    product_material: 'Soft'
  },
  {

```

8. Find all products which contain the value of soft in product material

db.product.find({product_material:"Soft"})

```

test> db.product.find({product_material:"Soft"})
[
  {
    _id: ObjectId('65d893b4d81668e39b665e07'),
    id: '4',
    product_name: 'Gorgeous Plastic Pants',
    product_price: 492,
    product_material: 'Soft',
    product_color: 'plum'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e0c'),
    id: '9',
    product_name: 'Awesome Wooden Ball',
    product_price: 28,
    product_material: 'Soft',
    product_color: 'azure'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e0e'),
    id: '11',
    product_name: 'Unbranded Wooden Cheese',
    product_price: 26,
    product_material: 'Soft',
    product_color: 'black'
  },
  {
    _id: ObjectId('65d893b4d81668e39b665e16'),
    id: '19',
    product_name: 'Intelligent Cotton Chips',
    product_price: 46,
    product_material: 'Soft',
    product_color: 'azure'
  }
]
test>

```

9. Find products which contain product color indigo and product price 492.00

```
db.product.find({$and : [{product_color: "indigo", product_price : 492.00}]});
```

Note = product not available

```

test> db.product.find({$and : [{product_color: "indigo", product_price : 492.00}]});
test>

```

10. Delete the products which product price value are same

```

db.product.aggregate(
...   [
...     {
...       $group: {_id: "$product_price",

```

```
...         count: { $sum: 1 },
...         docs: { $push: "$_id" }}
...     },
...     {$match: {count: { $gt: 1 }}}
... ].forEach(function(doc)
... {
...     doc.docs.shift();
...     db.product.deleteMany({ _id: { $in: doc.docs } });
... }
... );
```