

Сидоренко Катерина, ФБ-11мп

## Task 2 - Робота з базовими функціями документо-орієнтованої БД на прикладі MongoDB

1. Створіть декілька товарів з різним набором властивостей Phone/TV/Smart

```
mybd> db.mydb.insert({"category": "Smartwatch", "model": "MI band 3", "producer":  
"XiaoMi", "price": 20})  
{  
  acknowledged: true,  
  insertedIds: { '0': ObjectId("61b7805195bfe796959f3d3a") }  
}  
mybd> db.mydb.insert({"category": "Smartweight", "model": "Body Scale 2", "produc  
er": "XiaoMi", "price": 100})  
{  
  acknowledged: true,  
  insertedIds: { '0': ObjectId("61b7806495bfe796959f3d3b") }  
}  
mybd> db.mydb.insert({"category": "Laptop", "model": "Uehk737 Light", "producer":  
"Samsung", "price": 1000})  
{  
  acknowledged: true,  
  insertedIds: { '0': ObjectId("61b7807b95bfe796959f3d3c") }  
}  
mybd> db.mydb.insert({"category": "Phone", "model": "Galaxy A52", "producer": "Sa  
msung", "price": 600})  
{  
  acknowledged: true,  
  insertedIds: { '0': ObjectId("61b7809295bfe796959f3d3d") }  
}
```

2. Напишіть запит, який виводить усі товари (відображення у JSON)

```
mybd> db.mydb.find()  
[  
  {  
    _id: ObjectId("61b77ee295bfe796959f3d32"),  
    category: 'Phone',  
    model: 'iPhone 6',  
    producer: 'Apple',  
    price: 600  
  },  
  {  
    _id: ObjectId("61b7803395bfe796959f3d39"),  
    category: 'TV',  
    model: 'LED 3D 300',  
    producer: 'Samsung',  
    price: 2000  
  },  
  {  
    _id: ObjectId("61b7805195bfe796959f3d3a"),  
    category: 'Smartwatch',  
    model: 'MI band 3',  
    producer: 'XiaoMi',  
    price: 20  
  },  
]
```

```

    {
      _id: ObjectId("61b7806495bfe796959f3d3b"),
      category: 'Smartweight',
      model: 'Body Scale 2',
      producer: 'XiaoMi',
      price: 100
    },
    {
      _id: ObjectId("61b7807b95bfe796959f3d3c"),
      category: 'Laptop',
      model: 'Uehk737 Light',
      producer: 'Samsung',
      price: 1000
    },
    {
      _id: ObjectId("61b7809295bfe796959f3d3d"),
      category: 'Phone',
      model: 'Galaxy A52',
      producer: 'Samsung',
      price: 600
    }
  ]
}

```

`pretty()` в нашому випадку працює аналогічно

```

mybd> db.mydb.find().pretty()
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
  {
    _id: ObjectId("61b7803395bfe796959f3d39"),
    category: 'TV',
    model: 'LED 3D 300',
    producer: 'Samsung',
    price: 2000
  },
]

```

3. Підрахуйте скільки товарів у певній категорії

```

mybd> db.mydb.countDocuments({category: "Phone"})
2
mybd> db.mydb.countDocuments({category: "Laptop"})
1

```

4. Підрахуйте скільки є різних категорій товарів

```

mybd> db.mydb.distinct('category')
[ 'Laptop', 'Phone', 'Smartwatch', 'Smartweight', 'TV' ]
mybd> db.mydb.distinct('category').length
5

```

5. Виведіть список всіх виробників товарів без повторів

```

mybd> db.mydb.distinct('producer')
[ 'Apple', 'Samsung', 'XiaoMi' ]

```

6. Напишіть запити, які вибирають товари за різними критеріям і їх сукупності

а) категорія та ціна (в проміжку) - конструкція *\$and*

```

mybd> db.mydb.find({$and:[{'category':'Phone'}, {'price':{$gt:30, $lt:1000}}]})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
  {
    _id: ObjectId("61b7809295bfe796959f3d3d"),
    category: 'Phone',
    model: 'Galaxy A52',
    producer: 'Samsung',
    price: 600
  }
]
mybd> db.mydb.find({$and:[{'category':'Phone'}, {'price':{$gt:30, $lt:100}}]})
mybd> db.mydb.find({$and:[{'category':'Phone'}, {'price':{$gt:700, $lt:800}}]})

```

Якщо проміжок мається на увазі [gt,lt], тобто, включно, тоді:

```

mybd> db.mydb.find({$and:[{'category':'Phone'}, {'price':{$gte:600, $lte:600}}]})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
  {
    _id: ObjectId("61b7809295bfe796959f3d3d"),
    category: 'Phone',
    model: 'Galaxy A52',
    producer: 'Samsung',
    price: 600
  }
]

```

b) модель чи одна чи інша - конструкція \$or,

```

mybd> db.mydb.find({$or:[{'model':'Galaxy A52'}, {'model':'iPhone 6'}]})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
  {
    _id: ObjectId("61b7809295bfe796959f3d3d"),
    category: 'Phone',
    model: 'Galaxy A52',
    producer: 'Samsung',
    price: 600
  }
]

```

с) виробники з переліку - конструкція

\$in

```

mybd> db.mydb.find({'producer':{'$in': ['Apple', 'XiaoMi']}})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
  {
    _id: ObjectId("61b7805195bfe796959f3d3a"),
    category: 'Smartwatch',
    model: 'MI band 3',
    producer: 'XiaoMi',
    price: 20
  },
  {
    _id: ObjectId("61b7806495bfe796959f3d3b"),
    category: 'Smartweight',
    model: 'Body Scale 2',
    producer: 'XiaoMi',
    price: 100
  }
]
mybd> db.mydb.find({'producer':{'$in': db.mydb.distinct('producer')}})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 600
  },
]

```

виводить всі товари, оскільки у find усі виробники

7. Оновіть певні товари, змінивши існуючі значення і додайте нові властивості (характеристики) усім товарам за певним критерієм



```

mybd> db.mydb.updateOne({'model':'iPhone 6'}, {$set : {'price':650}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
mybd> db.mydb.find({'model':'iPhone 6'})
[
  {
    _id: ObjectId("61b77ee295bfe796959f3d32"),
    category: 'Phone',
    model: 'iPhone 6',
    producer: 'Apple',
    price: 650
  }
]

```

Оновили ціну iPhone

```

mybd> db.mydb.updateMany({'producer':'Samsung'}, { $set: { 'price': 300}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
mybd> db.mydb.find({'producer':'Samsung'})
[
  {
    _id: ObjectId("61b7803395bfe796959f3d39"),
    category: 'TV',
    model: 'LED 3D 300',
    producer: 'Samsung',
    price: 300
  },
  {
    _id: ObjectId("61b7807b95bfe796959f3d3c"),
    category: 'Laptop',
    model: 'Uehk737 Light',
    producer: 'Samsung',
    price: 300
  },
]

```

Оновили ціну всім товарам марки Самсунг

//Через некорректну роботу ВМ довелося перезаписати ДБ (\_id будуть іншу, початкова ціна збереглася)

```

mybd> db.mydb.updateMany({'price':{'$lte:1000}}, { $set: {'description':'SUPERSALE'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}

```

Критерій - ціна менше 1000, додали нове поле(характеристику/властивість)

8. Знайдіть товари у яких є (присутнє поле) певні властивості

```
mydb> db.mydb.find({'description': {'$exists: true}})
[
  {
    _id: ObjectId("61b7aafd233a929659219c0b"),
    category: 'Smartwatch',
    model: 'MI band 3',
    producer: 'XiaoMi',
    price: 20,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ab34233a929659219c0e"),
    category: 'Phone',
    model: 'Galaxy A52',
    producer: 'Samsung',
    price: 300,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ac30233a929659219c0f"),
    category: 'Smartweight',
    model: 'Body Scale 2',
    producer: 'XiaoMi',
    price: 100,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ac91233a929659219c11"),
    category: 'Laptop',
```

усі товари, у яких існує поле description

9. Для знайдених товарів збільшіть їх вартість на певну суму

```
mydb> db.mydb.updateMany({'description': {'$exists: true}}, {'$inc': {'price': 50}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}
```

```
mydb> db.mydb.find({'description': {$exists: true}})
[
  {
    _id: ObjectId("61b7aafd233a929659219c0b"),
    category: 'Smartwatch',
    model: 'MI band 3',
    producer: 'XiaoMi',
    price: 70,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ab34233a929659219c0e"),
    category: 'Phone',
    model: 'Galaxy A52',
    producer: 'Samsung',
    price: 350,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ac30233a929659219c0f"),
    category: 'Smartweight',
    model: 'Body Scale 2',
    producer: 'XiaoMi',
    price: 150,
    description: 'SUPERSALE'
  },
  {
    _id: ObjectId("61b7ac91233a929659219c11"),
```

2.1 Створіть кілька замовлень з різними наборами товарів, але так щоб один з товарів був у декількох замовленнях

```
mydb> db.mydb.insert({ "order_number" : 3000, "date" : ISODate("2021-04-11"), "total_sum" : 700.7, "customer" : { "name" : "Andrii", "surname" : "Rodinov", "phones" : [ 9876543, 1234567 ], "address" : "PTI, Peremohy 37, Kyiv, UA" }, "payment" : { "card_owner" : "Andrii Rodionov", "cardId" : 12345678 }, "order_items_id" : [ { "$ref" : "mydb", "$id": ObjectId("61b7acac233a929659219c13") }, { "$ref" : "mydb", "$id": ObjectId("61b7ac91233a929659219c11") } ] })
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7b827233a929659219c14") }
}

mydb> db.mydb.insert({ "order_number" : 3001, "date" : ISODate("2021-05-11"), "total_sum" : 1750.7, "customer" : { "name" : "John", "surname" : "Doe", "phones" : [ 01522218934 ], "address" : "Somestreet, 13, Kyiv, UA" }, "payment" : { "card_owner" : "John Doe", "cardId" : 123839847 }, "order_items_id" : [ { "$ref" : "mydb", "$id" : ObjectId("61b7ac91233a929659219c11") }, { "$ref" : "mydb", "$id" : ObjectId("61b7ac30233a929659219c0f") }, { "$ref" : "mydb", "$id" : ObjectId("61b7ac5f233a929659219c10") } ] })
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7b9bd233a929659219c16") }
}
```



```
mydb> db.mydb.insert({ "order_number" : 3002,"date" : ISODate("2021-05-15"),"total_sum" : 7365,"customer" : { "name" : "Erika", "surname" : "Burg", "phones" : [3015222134], "address" : "Strasse, 38a, Kyiv, UA" }, "payment" : { "card_owner" : "Peter Burg", "cardId" : 2846847 }, "order_items_id" : [ { "$ref" : "mydb", "$id" : ObjectId("61b7ac91233a929659219c11") }, { "$ref" : "mydb", "$id" : ObjectId("61b7ac30233a929659219c12") }, { "$ref" : "mydb", "$id" : ObjectId("61b7ac5f233a929659219c13") } ] })
```

2.2 Виведіть всі замовлення

```
mydb> db.mydb.find({'order_number': {$exists: true}})
[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    date: ISODate("2021-04-11T00:00:00.000Z"),
    total_sum: 700.7,
    customer: {
      name: 'Andrii',
      surname: 'Rodionov',
      phones: [ 9876543, 1234567 ],
      address: 'PTI, Peremohy 37, Kyiv, UA'
    },
    payment: { card_owner: 'Andrii Rodionov', cardId: 12345678 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7acac233a929659219c13")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11"))
    ]
  },
  {
    _id: ObjectId("61b7b9bd233a929659219c16"),
    _id: ObjectId("61b7b9bd233a929659219c16"),
    order_number: 3001,
    date: ISODate("2021-05-11T00:00:00.000Z"),
    total_sum: 1750.7,
    customer: {
      name: 'John',
      surname: 'Doe',
      phones: [ 1522218934 ],
      address: 'Somestreet, 13, Kyiv, UA'
    },
    payment: { card_owner: 'John Doe', cardId: 123839847 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac30233a929659219c0f")),
      DBRef("mydb", ObjectId("61b7ac5f233a929659219c10"))
    ]
  }
]
```



```

},
{
  _id: ObjectId("61b7bb49233a929659219c17"),
  order_number: 3002,
  date: ISODate("2021-05-15T00:00:00.000Z"),
  total_sum: 7365,
  customer: {
    name: 'Erika',
    surname: 'Burg',
    phones: [ 3015222134 ],
    address: 'Strasse, 38a, Kyiv, UA'
  },
  payment: { card_owner: 'Peter Burg', cardId: 2846847 },
  order_items_id: [
    DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
    DBRef("mydb", ObjectId("61b7ac30233a929659219c12")),
    DBRef("mydb", ObjectId("61b7ac5f233a929659219c13"))
  ]
}
]

```

2.3 Виведіть замовлення з вартістю більше певного значення

```

mydb> db.mydb.find({'total_sum': {$gt: 7300}})
[
  {
    _id: ObjectId("61b7bb49233a929659219c17"),
    order_number: 3002,
    date: ISODate("2021-05-15T00:00:00.000Z"),
    total_sum: 7365,
    customer: {
      name: 'Erika',
      surname: 'Burg',
      phones: [ 3015222134 ],
      address: 'Strasse, 38a, Kyiv, UA'
    },
    payment: { card_owner: 'Peter Burg', cardId: 2846847 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac30233a929659219c12")),
      DBRef("mydb", ObjectId("61b7ac5f233a929659219c13"))
    ]
  }
]

```

2.4 Знайдіть замовлення зроблені одним замовником

//додали ще замовлення для Erika, щоб перевірити, що виводить всі

```
mydb> db.mydb.find({'$and': [{'customer.name': 'Erika'}, {'customer.surname': 'Burg'}]})
[
  {
    _id: ObjectId("61b7bb49233a929659219c17"),
    order_number: 3002,
    date: ISODate("2021-05-15T00:00:00.000Z"),
    total_sum: 7365,
    customer: {
      name: 'Erika',
      surname: 'Burg',
      phones: [ 3015222134 ],
      address: 'Strasse, 38a, Kyiv, UA'
    },
    payment: { card_owner: 'Peter Burg', cardId: 2846847 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac30233a929659219c12")),
      DBRef("mydb", ObjectId("61b7ac5f233a929659219c13"))
    ]
  },
  {
    _id: ObjectId("61b7bfff233a929659219c18"),
    order_number: 3002,
    date: ISODate("2021-05-15T00:00:00.000Z"),
    total_sum: 7365,
    customer: {
      name: 'Erika',
```

2.5 Знайдіть всі замовлення з певним товаром (товарами) (шукати можна по ObjectId)

```
mydb> db.mydb.find({"order_items_id.$id": ObjectId("61b7ac91233a929659219c0e")})
[
  {
    _id: ObjectId("61b7bfff233a929659219c18"),
    order_number: 3002,
    date: ISODate("2021-05-15T00:00:00.000Z"),
    total_sum: 7365,
    customer: {
      name: 'Erika',
      surname: 'Burg',
      phones: [ 3015222134 ],
      address: 'Strasse, 38a, Kyiv, UA'
    },
    payment: { card_owner: 'Peter Burg', cardId: 2846847 },
    order_items_id: [ DBRef("mydb", ObjectId("61b7ac91233a929659219c0e")) ]
  }
]
```

2.6 Додайте в усі замовлення з певним товаром ще один товар і збільште існуючу вартість замовлення на деяке значення X

```

mydb> db.mydb.find({"order_items_id.$id":ObjectId("61b7acac233a929659219c13")})

[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    date: ISODate("2021-04-11T00:00:00.000Z"),
    total_sum: 700.7,
    customer: {
      name: 'Andrii',
      surname: 'Rodionov',
      phones: [ 9876543, 1234567 ],
      address: 'PTI, Peremohy 37, Kyiv, UA'
    },
    payment: { card_owner: 'Andrii Rodionov', cardId: 12345678 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7acac233a929659219c13")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11"))
    ]
  }
]

mydb> db.mydb.updateMany({"order_items_id.$id":ObjectId("61b7acac233a929659219c13")},{$push: {"order_items_id" : {"$ref" : "mydb", "$id":ObjectId("61b7ac91233a929659219c0b")}}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

mydb> db.mydb.updateMany({"order_items_id.$id":ObjectId("61b7acac233a929659219c13")}, {$inc: {'total_sum':70}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

mydb> db.mydb.find({"order_items_id.$id":ObjectId("61b7acac233a929659219c13")})

[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    date: ISODate("2021-04-11T00:00:00.000Z"),
    total_sum: 770.7,
    customer: {
      name: 'Andrii',
      surname: 'Rodionov',
      phones: [ 9876543, 1234567 ],
      address: 'PTI, Peremohy 37, Kyiv, UA'
    },
    payment: { card_owner: 'Andrii Rodionov', cardId: 12345678 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7acac233a929659219c13")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c0b"))
    ]
  }
]

```

## 2.7 Виведіть кількість товарів в певному замовленні



```

mydb> db.mydb.find({'order_number': 3000})
[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    date: ISODate("2021-04-11T00:00:00.000Z"),
    total_sum: 770.7,
    customer: {
      name: 'Andrii',
      surname: 'Rodionov',
      phones: [ 9876543, 1234567 ],
      address: 'PTI, Peremohy 37, Kyiv, UA'
    },
    payment: { card_owner: 'Andrii Rodionov', cardId: 12345678 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7acac233a929659219c13")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c0b"))
    ]
  }
]
mydb> db.mydb.aggregate([ { $match: {'order_number': 3000} }, { $project: {'order_number': 3000, items_count: { $size: "$order_items_id" }}} ] )
[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    items_count: 3
  }
]

```

2.8 Виведіть тільки інформацію про кастомера і номери кредитної карт, для замовлень вартість яких перевищує певну суму

```

mydb> db.mydb.mapReduce(function(){return emit(this.customer.name, this.payment.cardId);}, function(key,values) {return Array.sum(values);}, {query: { $and : [ {'total_sum' : { $gte :1000}} ] }, out: 'map_red'});
{ result: 'map_red', ok: 1 }

mydb> db.map_red.find()
[ { _id: 'Erika', value: 2846847 }, { _id: 'John', value: 123839847 } ]
mydb>

```

2.9 Видаліть товар з замовлень, зроблених за певний період дат

```

mydb> db.mydb.find({"date":{"$gte:ISODate("2021-05-16"),$lte:ISODate("2021-05-28")}})
[
  {
    _id: ObjectId("61b7bb49233a929659219c17"),
    order_number: 3002,
    date: ISODate("2021-05-16T00:00:00.000Z"),
    total_sum: 7365,
    customer: {
      name: 'Erika',
      surname: 'Burg',
      phones: [ 3015222134 ],
      address: 'Strasse, 38a, Kyiv, UA'
    },
    payment: { card_owner: 'Peter Burg', cardId: 2846847 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac30233a929659219c12")),
      DBRef("mydb", ObjectId("61b7ac5f233a929659219c13"))
    ]
  }
]
mydb> db.mydb.remove({"date":{"$gte:ISODate("2021-05-16"),$lte:ISODate("2021-05-28")}})
DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.
{ acknowledged: true, deletedCount: 1 }
mydb> db.mydb.find({"date":{"$gte:ISODate("2021-05-16"),$lte:ISODate("2021-05-28")}})
mydb>

```

2.10 Перейменуйте у всіх замовленнях прізвище замовника

//на момент виконання завдання існувало лише 3 замовлення (Еріка, Андрій та Джон 3000-3002)

якби замовлення були в окремій колекції, замість {'order\_number':{'\$exists: true}}, було б {}

```

mydb> db.mydb.updateMany({'order_number':{'$exists: true}}, {$set: {'Customer.surname': 'New_surname'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}

```

```

mydb> db.mydb.find({'Customer.surname':{$exists: true}})
[
  {
    _id: ObjectId("61b7b827233a929659219c14"),
    order_number: 3000,
    date: ISODate("2021-04-11T00:00:00.000Z"),
    total_sum: 770.7,
    payment: { card_owner: 'Andrii Rodionov', cardId: 12345678 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7acac233a929659219c13")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac91233a929659219c0b"))
    ],
    Customer: {
      name: 'Andrii',
      surname: 'New_surname',
      phones: [ 9876543, 1234567 ],
      address: 'PTI, Peremohy 37, Kyiv, UA'
    }
  },
  {
    _id: ObjectId("61b7b9bd233a929659219c16"),
    order_number: 3001,
    date: ISODate("2021-05-11T00:00:00.000Z"),
    total_sum: 1750.7,
    payment: { card_owner: 'John Doe', cardId: 123839847 },
    order_items_id: [
      DBRef("mydb", ObjectId("61b7ac91233a929659219c11")),
      DBRef("mydb", ObjectId("61b7ac30233a929659219c0f")),
      DBRef("mydb", ObjectId("61b7ac5f233a929659219c10"))
    ],
    Customer: {
      name: 'John',
      surname: 'New_surname',
      phones: [ 1522218934 ],
      address: 'Somestreet, 13, Kyiv, UA'
    }
  },
  {
    _id: ObjectId("61b7bfff233a929659219c18"),
    order_number: 3002,
    date: ISODate("2021-05-15T00:00:00.000Z"),
    total_sum: 7365,
    payment: { card_owner: 'Peter Burg', cardId: 2846847 },
    order_items_id: [ DBRef("mydb", ObjectId("61b7ac91233a929659219c0e")) ],
    Customer: {
      name: 'Erika',
      surname: 'New_surname',
      phones: [ 3015222134 ],
      address: 'Strasse, 38a, Kyiv, UA'
    }
  }
]

```

2.11 Знайдіть замовлення зроблені одним замовником, і виведіть тільки інформацію про кастомера та товари у замовленні підставивши замість



ObjectId("\*\*\*") назви товарів та їх вартість (аналог join-а між таблицями *orders* та *items*).

```
mydb> db.customer.aggregate([ {$match: {'customer.name':'John'}}, {$lookup:
{ from: "item", localField: "order_items_id.$id", foreignField: "_id", as: "
oritem" }}, {$project: {customer: 1, _id:0, 'oritem.model':1, 'oritem.price':
1}}])
[
  {
    customer: {
      name: 'John',
      surname: 'Doe',
      phones: [ 1522218934 ],
      address: 'Somestreet, 13, Kyiv, UA'
    },
    oritem: [
      { model: 'MI band 3', price: 20 },
      { model: 'Body Scale 2', price: 100 }
    ]
  }
]
```

Створіть [Capped collection](#) яка б містила 5 останніх відгуків на наш інтернет-магазин. Структуру запису визначіть самостійно.

```
mydb> db.createCollection("review", {capped: true, size:371400, max:5})
{ ok: 1 }
mydb> db.review.insert({user:'admin', text:'Soorgngirjgiriughiwuhgiuhgi', date:
ISODate("2021-06-14")})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMan
y, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e0f17bb534f3034e5178") }
}
mydb> db.review.find()
[
  {
    _id: ObjectId("61b7e0f17bb534f3034e5178"),
    user: 'admin',
    text: 'Soorgngirjgiriughiwuhgiuhgi',
    date: ISODate("2021-06-14T00:00:00.000Z")
  }
]
mydb> db.review.insert({user:'allo4ka', text:'Vse good, horosho', date:ISODate(
"2021-02-11")})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e1247bb534f3034e5179") }
}
mydb> db.review.insert({user:'petrovich', text:'Telepon ne ponravilsya', date:I
SODate("2021-12-11")})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e1427bb534f3034e517a") }
}
mydb> db.review.insert({user:'cat2dog', text:'Malo tovarov', date:ISODate("2021
-03-17")})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e1607bb534f3034e517b") }
}
```

```

mydb> db.review.insert({user:'revizor', text:' 1 iz 10, ne rokomendyu', date:ISODate("2021-08-23")})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e1927bb534f3034e517c") }
}

mydb> db.review.find()
[
  {
    _id: ObjectId("61b7e0f17bb534f3034e5178"),
    user: 'admin',
    text: 'Soorgngirjgiriughiwuhgiuhgi',
    date: ISODate("2021-06-14T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1247bb534f3034e5179"),
    user: 'allo4ka',
    text: 'Vse good, horosho',
    date: ISODate("2021-02-11T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1427bb534f3034e517a"),
    user: 'petrovich',
    text: 'Telepon ne ponravilsya',
    date: ISODate("2021-12-11T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1607bb534f3034e517b"),
    user: 'cat2dog',
    text: 'Malo tovarov',
    date: ISODate("2021-03-17T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1927bb534f3034e517c"),
    user: 'revizor',

```

Перевірте що при досягненні обмеження старі відгуки будуть затиратись

```

mydb> db.review.insert({user:'revizor2', text:' 10 iz 10, rokomendyu', date:ISODate("2021-08-08")})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b7e1af7bb534f3034e517d") }
}

mydb> db.review.find()
[
  {
    _id: ObjectId("61b7e1247bb534f3034e5179"),
    user: 'allo4ka',
    text: 'Vse good, horosho',
    date: ISODate("2021-02-11T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1427bb534f3034e517a"),
    user: 'petrovich',
    text: 'Telepon ne ponravilsya',
    date: ISODate("2021-12-11T00:00:00.000Z")
  },
  {
    _id: ObjectId("61b7e1607bb534f3034e517b"),
    user: 'cat2dog',
    text: 'Malo tovarov',
    date: ISODate("2021-03-17T00:00:00.000Z")
  },

```

```

{
  _id: ObjectId("61b7e1927bb534f3034e517c"),
  user: 'revizor',
  text: ' 1 iz 10, ne rokomendyu',
  date: ISODate("2021-08-23T00:00:00.000Z")
},
{
  _id: ObjectId("61b7e1af7bb534f3034e517d"),
  user: 'revizor2',
  text: ' 10 iz 10, rokomendyu',
  date: ISODate("2021-08-08T00:00:00.000Z")
}
ydb> db.review.insert({user:'revizor3', text:'norm', date:ISODate("2021-08-08"
)})
acknowledged: true,
insertedIds: { '0': ObjectId("61b7e2a77bb534f3034e517e") }
ydb> db.review.find()
{
  _id: ObjectId("61b7e1427bb534f3034e517a"),
  user: 'petrovich',
  text: 'Telepon ne ponravilsya',
  date: ISODate("2021-03-17T00:00:00.000Z")
},
{
  _id: ObjectId("61b7e1607bb534f3034e517b"),
  user: 'cat2dog',
  text: 'Malo tovarov',
  date: ISODate("2021-03-17T00:00:00.000Z")
},
{
  _id: ObjectId("61b7e1927bb534f3034e517c"),
  user: 'revizor',
  text: ' 1 iz 10, ne rokomendyu',
  date: ISODate("2021-08-23T00:00:00.000Z")
},
{
  _id: ObjectId("61b7e1af7bb534f3034e517d"),
  user: 'revizor2',
  text: ' 10 iz 10, rokomendyu',
  date: ISODate("2021-08-08T00:00:00.000Z")
},
{
  _id: ObjectId("61b7e2a77bb534f3034e517e"),
  user: 'revizor3',
  text: 'norm',
  date: ISODate("2021-08-08T00:00:00.000Z")
}

```