

Bài 1. Thêm document vào collection

Tạo tập tin xe.json chứa thông tin của xe như hình bên dưới. Thông tin mỗi xe gồm có: mã xe, tên, năm sản

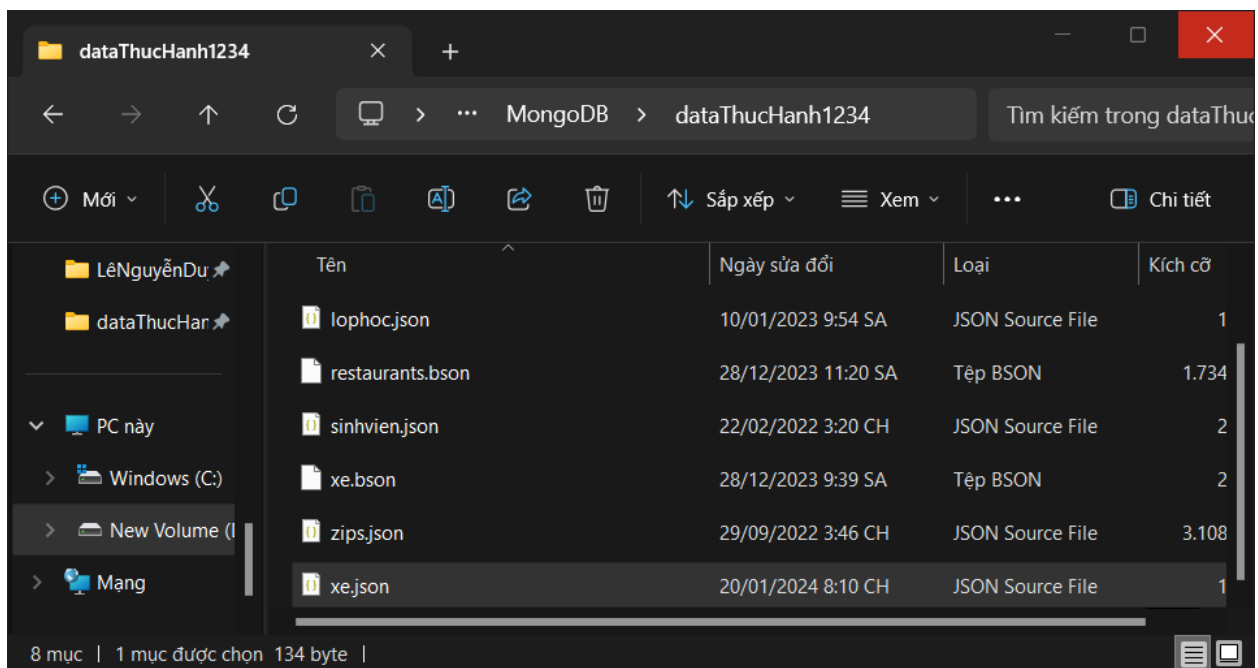
xuất, giá, hình ảnh, loại xe. Trong đó, loại xe chứa thông tin về mã loại và tên loại.

xe.json

```
{ma: 'Ya001',ten: 'Yamaha',namsx: 1992,gia: 12000000,hinhanh: 'h2.jpg',loai:{ maloi: '001Xemay',  
tenloai: 'Xe máy' }}
```

Thực hiện đưa dữ liệu từ tập tin xe.json vào CSDL với tên database là qlXe, tên collection là thôngTinXe.

Dùng lệnh insertOne() để chèn thêm document tùy ý vào collection. Xem kết quả bằng lệnh find().



```
D:\HOC\NAM2\MongoDB\dataThucHanh1234>mongoimport -d qlXe -c thôngTinXe xe.js  
on  
2024-01-20T20:16:34.602+0700    connected to: mongodb://localhost/  
2024-01-20T20:16:34.628+0700    1 document(s) imported successfully. 0 docum  
ent(s) failed to import.
```

```
D:\HOC\NAM2\MongoDB\dataThucHanh1234>|
```

```
qlXe> db.thôngTinXe.insertOne({ma: 'Ya002',ten: 'Yamaha',namsx: 1992,gia: 16  
000000,hinhanh: 'h1.jpg',loai:{ maloi: '001Xemay', tenloai: 'Xe máy' }})  
{  
  acknowledged: true,  
  insertedId: ObjectId('65abc815230eaf029d27172f')  
}  
qlXe> |
```

```
qlXe> db.thongTinXe.find()
[
  {
    _id: ObjectId('65abc7b2a559ecd5a9ee7724'),
    ma: 'Ya001',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 12000000,
    hinhanh: 'h2.jpg',
    loai: { maloai: '001Xemay', tenloai: 'Xe máy' }
  },
  {
    _id: ObjectId('65abc815230eaf029d27172f'),
    ma: 'Ya002',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 16000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '001Xemay', tenloai: 'Xe máy' }
  }
]
qlXe> |
```

Bài 2. Cài đặt ràng buộc dữ liệu (document validate)

1. Cài đặt validate rule cho collection thongTinXe: giá của xe “Honda” từ 15 triệu đến 50 triệu.

a. Thêm document sau vào thongtinXe và cho nhận xét:

```
{ma: 'HD005', ten: 'Honda', namsx: 2015, gia: 12000000, hinhanh: 'h5.jpg', loai:{maloai: '002Xemay',tenloai: 'Xe máy'}}
```

```
{
  $and: [
    {
      ten: {
        $eq: 'Honda'
      }
    },
    {
      gia: {
        $gte: 15000000,
        $lte: 50000000
      }
    }
  ]
}
```

```

qlXe> db.thongTinXe.insertOne({ma: 'HD005', ten: 'Honda', namsx: 2015, gia:
12000000, hinhanh: 'h5.jpg', loai:{maloai: '002Xemay',tenloai: 'Xe máy'}})
Uncaught:
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('65abc957230eaf029d271730'),
  details: {
    operatorName: '$and',
    clausesNotSatisfied: [
      {
        index: 1,
        details: {
          operatorName: '$eq',
          specifiedAs: { gia: { gt: 15000000, lt: 50000 } },
          reason: 'comparison failed',
          consideredValue: 12000000
        }
      }
    ]
  }
}
qlXe> |

```

Lỗi vì không thoả ràng buộc về giá

b. Thêm document sau vào thongtinXe và cho nhận xét: {ma: 'HD005', ten: 'Honda', namsx: 2015, gia: 18000000, hinhanh: 'h5.jpg', loai:{maloai: '002Xemay',tenloai: 'Xe máy'}}

```

qlXe> db.thongTinXe.insertOne({ma: 'HD005', ten: 'Honda', namsx: 2015, gia:
18000000, hinhanh: 'h5.jpg', loai:{maloai: '002Xemay',tenloai: 'Xe máy'}})
{
  acknowledged: true,
  insertedId: ObjectId('65abcbb9230eaf029d271734')
}
qlXe> |

```

Thoả ràng buộc nên chèn được

2. Cài đặt validate rule cho collection thongTinXe: năm sản xuất của xe là sau năm 2000.

a. Thêm document sau vào thongtinXe và cho nhận xét:

```

{"ma":"Fd006","ten":"Ford","namsx":1999,"gia":555000000,"hinhanh":"h3.jpg","loai":{"maloai":"001Ot
o","tenloai":"Ô tô"}}

```

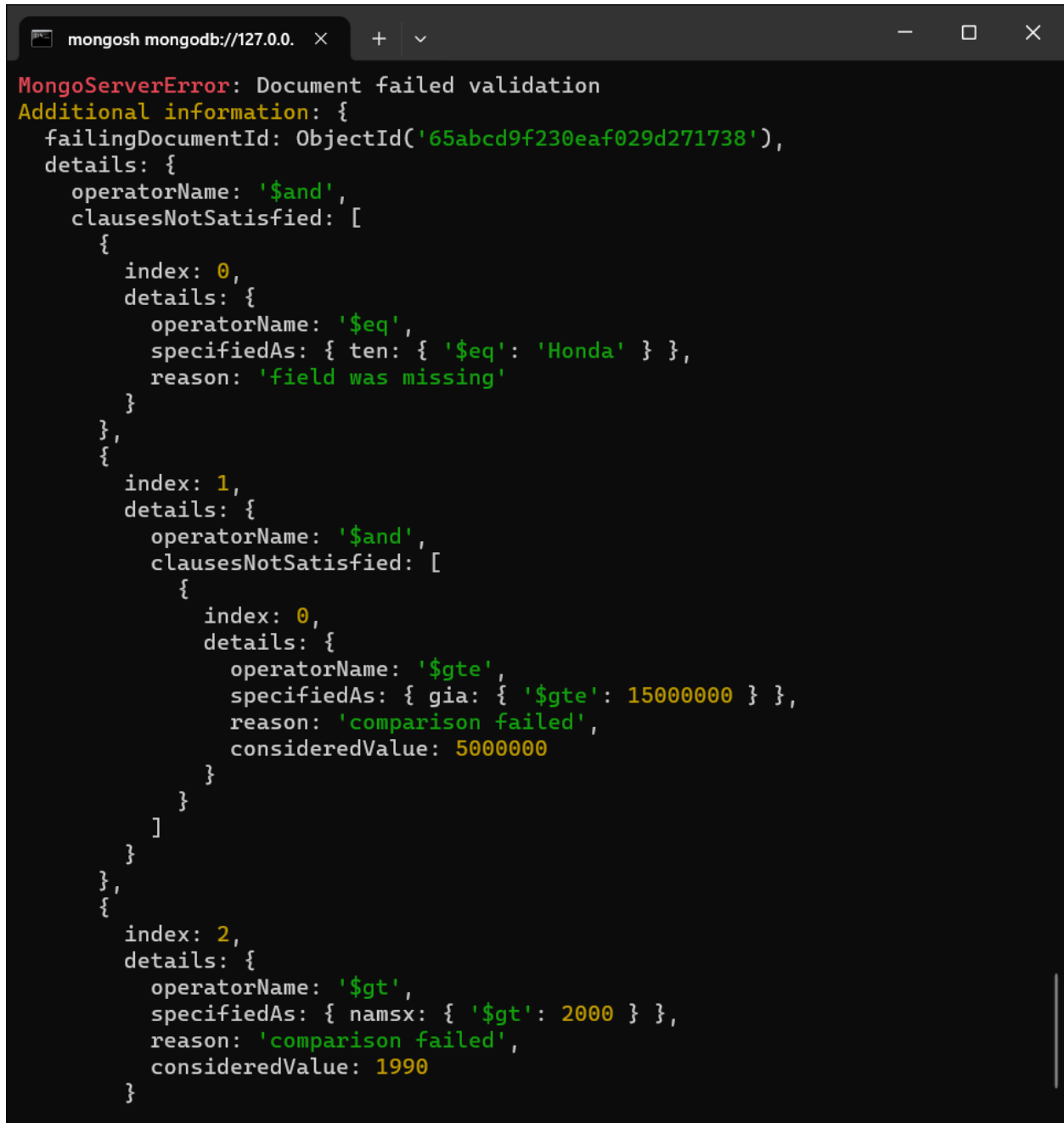
```
1 ▼ {
2 ▼   $and: [
3 ▼     {
4 ▼       ten: {
5         $eq: 'Honda'
6       }
7     },
8 ▼     {
9 ▼       gia: {
10        $gte: 15000000,
11        $lte: 50000000
12      }
13    },
14 ▼    {
15 ▼      namsx: {
16        $gt: 2000
17      }
18    }
19  ]
20 }
```

```
mongosh mongodb://127.0.0.1:27017/
> use test
> db.products.find(
  {
    'details.details.operatorName': '$and',
    'details.details.clausesNotSatisfied': [
      {
        index: 0,
        details: {
          operatorName: '$eq',
          specifiedAs: { ten: { '$eq': 'Honda' } },
          reason: 'comparison failed',
          consideredValue: 'Ford'
        }
      },
      {
        index: 1,
        details: {
          operatorName: '$and',
          clausesNotSatisfied: [
            {
              index: 1,
              details: {
                operatorName: '$lt',
                specifiedAs: { gia: { '$lt': 50000000 } },
                reason: 'comparison failed',
                consideredValue: 555000000
              }
            }
          ]
        }
      }
    ]
  },
  {
    index: 2,
    details: {
      operatorName: '$gt',
      specifiedAs: { namsx: { '$gt': 2000 } },
      reason: 'comparison failed',
      consideredValue: 1999
    }
  }
}
```

3. Bật tắt validate rule:

a. Thêm document sau vào thông tin xe và cho nhận xét:

```
{"ma": "Fd007", "hang": "Ford", "namsx": 1990, "gia": 5000000, "hinhanh": "h7.jpg", "loai": {"maloi": "002Xe may", "tenloai": "Xe gắn máy"}}
```

A screenshot of a MongoDB terminal window. The title bar shows 'mongosh mongodb://127.0.0.1'. The terminal displays a 'MongoServerError: Document failed validation' message. Below this, it shows 'Additional information' with a 'failingDocumentId' and 'details'. The details include three validation rules (index 0, 1, 2) that failed. Index 0 failed because the 'ten' field is missing in a '\$eq' comparison. Index 1 failed because the 'gia' field (5000000) is less than the specified value (15000000) in a '\$gte' comparison. Index 2 failed because the 'namsx' field (1990) is greater than the specified value (2000) in a '\$gt' comparison.

```
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('65abcd9f230eaf029d271738'),
  details: {
    operatorName: '$and',
    clausesNotSatisfied: [
      {
        index: 0,
        details: {
          operatorName: '$eq',
          specifiedAs: { ten: { '$eq': 'Honda' } },
          reason: 'field was missing'
        }
      },
      {
        index: 1,
        details: {
          operatorName: '$and',
          clausesNotSatisfied: [
            {
              index: 0,
              details: {
                operatorName: '$gte',
                specifiedAs: { gia: { '$gte': 15000000 } },
                reason: 'comparison failed',
                consideredValue: 5000000
              }
            }
          ]
        }
      },
      {
        index: 2,
        details: {
          operatorName: '$gt',
          specifiedAs: { namsx: { '$gt': 2000 } },
          reason: 'comparison failed',
          consideredValue: 1990
        }
      }
    ]
  }
}
```

Không chèn được vì không thỏa điều kiện về tên, giá và năm sản xuất

b. Tắt validate rule trên trường năm sản xuất, thực hiện thêm lại document trên vào và cho nhận xét.

```
▼ {
▼   $and: [
▼     {
▼       ten: {
▼         $eq: 'Honda'
▼       }
▼     },
▼     {
▼       gia: {
▼         $gte: 15000000,
▼         $lte: 50000000
▼       }
▼     }
▼   ]
▼ }
```

```
mongosh mongodb://127.0.0.1:27017
qlXe> db.thongTinXe.insertOne({"ma":"Fd007","hang":"Ford","namsx":1990,"gia":5000000,"hinhanh":"h7.jpg","loai":{"maloi":"002Xemay","tenloai":"Xe gắn máy"}})
Uncaught:
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('65abce2d230eaf029d271739'),
  details: {
    operatorName: '$and',
    clausesNotSatisfied: [
      {
        index: 0,
        details: {
          operatorName: '$eq',
          specifiedAs: { ten: { '$eq': 'Honda' } },
          reason: 'field was missing'
        }
      },
      {
        index: 1,
        details: {
          operatorName: '$and',
          clausesNotSatisfied: [
            {
              index: 0,
              details: {
                operatorName: '$gte',
                specifiedAs: { gia: { '$gte': 15000000 } },
                reason: 'comparison failed',
                consideredValue: 5000000
              }
            }
          ]
        }
      }
    ]
  }
}
qlXe>
```

Vẫn không chèn được vì không thỏa ràng buộc về tên và giá

c. Thêm document sau vào thongTinXe và cho nhận xét:

```
{"ma":"Hd001","hang":"Honda","namsx":1990,"gia":5000000,"hinhanh":"h1.jpg","loai":{"maloi":"002Xemay","tenloai":"Xe gắn máy"},cuahang:["Cong Hoa", "An Thanh", "Phat Tien"]}
```



```
mongosh mongodb://127.0.0.1:27020/qlXe> db.thongTinXe.insertOne({"ma":"Hd001","hang":"Honda","namSX":1990,"gia":5000000,"hinhAnh":"h1.jpg","loai":{"maLoai":"002XeMay","tenLoai":"Xe gán m áy"},"cuahang":["Cong Hoa","An Thanh","Phat Tien"]})
Uncaught:
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('65abce8d230eaf029d27173a'),
  details: {
    operatorName: '$and',
    clausesNotSatisfied: [
      {
        index: 0,
        details: {
          operatorName: '$eq',
          specifiedAs: { ten: { '$eq': 'Honda' } },
          reason: 'field was missing'
        }
      },
      {
        index: 1,
        details: {
          operatorName: '$and',
          clausesNotSatisfied: [
            {
              index: 0,
              details: {
                operatorName: '$gte',
                specifiedAs: { gia: { '$gte': 15000000 } },
                reason: 'comparison failed',
                consideredValue: 5000000
              }
            }
          ]
        }
      }
    ]
  }
}
```

Lỗi mất field name nên không thể kiểm tra ràng buộc về tên. Ràng buộc về giá cũng không thoả
d. Tắt hết tất cả các validate rule có trong thongTinXe.

```
1 {}
```

e. Chèn lại document ở trên vào thôngTinXe và cho nhận xét.

```
qlXe> db.thongTinXe.insertOne({"ma":"Hd001","hang":"Honda","namSX":1990,"gia":5000000,"hinhanh":"h1.jpg","loai":{"maLoai":"002Xemay","tenLoai":"Xe gắn m
áy"}, "cuahang":["Cong Hoa", "An Thanh", "Phat Tien"]})
{
  acknowledged: true,
  insertedId: ObjectId('65abcf28230eaf029d27173b')
}
qlXe> |
```

Chèn được vì không còn ràng buộc nữa

Bài 3. Truy vấn collection xe

Thực hiện đưa dữ liệu từ file xe.bson (cho sẵn) vào collection xe, viết các truy vấn sau:

```
D:\HOC\NAM2\MongoDB\dataThucHanh1234>mongorestore -d qlXe -c xe xe.bson
2024-01-20T20:49:29.195+0700 checking for collection data in xe.bson
2024-01-20T20:49:29.221+0700 restoring qlXe.xe from xe.bson
2024-01-20T20:49:29.283+0700 finished restoring qlXe.xe (12 documents, 0
failures)
2024-01-20T20:49:29.283+0700 12 document(s) restored successfully. 0 docu
ment(s) failed to restore.

D:\HOC\NAM2\MongoDB\dataThucHanh1234>|
```

1. Xuất toàn bộ danh sách xe máy.

```
mongosh mongodb://127.0.0.1:27021/qlXe
switched to db qlXe
qlXe> db.xe.find()
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc4'),
    ma: 'Hd002',
    ten: 'Honda',
    namsx: 1990,
    gia: 10000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc5'),
    ma: 'Fd006',
    ten: 'Ford',
    namsx: 1999,
    gia: 555000000,
    hinhanh: 'h3.jpg',
    loai: { maloai: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc6'),
    ma: 'Ya002',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 14000000,
    hinhanh: 'h2.jpg',
    loai: { maloai: '001Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc7'),
    ma: 'Ya001',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 12000000,
    hinhanh: 'h2.jpg',
    loai: { maloai: '001Xemay', tenloai: 'Xe máy' }
  },
]
```

```
{
  _id: ObjectId('658cdf4fe428794a57451bc8'),
  ma: 'Hd001',
  ten: 'Honda',
  namsx: 1990,
  gia: 5000000,
  hinhanh: 'h1.jpg',
  loai: { ma_loai: '002Xemay', ten_loai: 'Xe gắn máy' }
},
{
  _id: ObjectId('658cdf4fe428794a57451bc9'),
  ma: 'Ya003',
  ten: 'Yamaha',
  namsx: 1992,
  gia: 13000000,
  hinhanh: 'h2.jpg',
  loai: { ma_loai: '001Xemay', ten_loai: 'Xe gắn máy' }
},
{
  _id: ObjectId('658cdf4fe428794a57451bca'),
  ma: 'Hd003',
  ten: 'Honda',
  namsx: 1990,
  gia: 25000000,
  hinhanh: 'h1.jpg',
  loai: { ma_loai: '002Xemay', ten_loai: 'Xe gắn máy' }
},
{
  _id: ObjectId('658cdf4fe428794a57451bcb'),
  ma: 'Hd004',
  ten: 'Honda',
  namsx: 1990,
  gia: 15000000,
  hinhanh: 'h1.jpg',
  loai: { ma_loai: '002Xemay', ten_loai: 'Xe gắn máy' }
},
{
  _id: ObjectId('658cdf4fe428794a57451bcc'),
  ma: 'Ya004',
```

```
mongosh mongodb://127.0.0.1:27020/qlXe> use qlXe
qlXe> find('xe')
{
  "_id": ObjectId("658cdf4fe428794a57451bcc"),
  "ma": "Ya004",
  "ten": "Yamaha",
  "namsx": 1992,
  "gia": 80000000,
  "hinhanh": "h2.jpg",
  "loai": { "ma_loai": "001Xemay", "ten_loai": "Xe gắn máy" }
},
{
  "_id": ObjectId("658cdf4fe428794a57451bcd"),
  "ma": "Fd010",
  "ten": "Ford",
  "namsx": 2001,
  "gia": 655000000,
  "hinhanh": "h3.jpg",
  "loai": { "ma_loai": "0010to", "ten_loai": "Ô tô" }
},
{
  "_id": ObjectId("658cdf4fe428794a57451bce"),
  "ma": "Fd110",
  "ten": "Ford",
  "namsx": 2002,
  "gia": 755000000,
  "hinhanh": "h3.jpg",
  "loai": { "ma_loai": "0010to", "ten_loai": "Ô tô" }
},
{
  "_id": ObjectId("658cdf4fe428794a57451bcf"),
  "ma": "Fd005",
  "ten": "Ford",
  "namsx": 2009,
  "gia": 655000000,
  "hinhanh": "h3.jpg",
  "loai": { "ma_loai": "0010to", "ten_loai": "Ô tô" }
}
]
```

2. Xuất danh sách xe ô tô.

```
mongosh mongodb://127.0.0.1:27020/xe
qlXe> db.xe.find({"loai.tenloai" : "Ô tô"})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc5'),
    ma: 'Fd006',
    ten: 'Ford',
    namsx: 1999,
    gia: 555000000,
    hinhanh: 'h3.jpg',
    loai: { maloai: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcd'),
    ma: 'Fd010',
    ten: 'Ford',
    namsx: 2001,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloai: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bce'),
    ma: 'Fd110',
    ten: 'Ford',
    namsx: 2002,
    gia: 755000000,
    hinhanh: 'h3.jpg',
    loai: { maloai: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcf'),
    ma: 'Fd005',
    ten: 'Ford',
    namsx: 2009,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloai: '0010to', tenloai: 'Ô tô' }
  }
]
```

3. Xuất danh sách xe máy Yamaha.

```
mongosh mongodb://127.0.0.1:27020/xe
> use xe
switched to db xe
> show collections
collection
> use collection
switched to db collection
> find({ten:"Yamaha"})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc6'),
    ma: 'Ya002',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 14000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc7'),
    ma: 'Ya001',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 12000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc9'),
    ma: 'Ya003',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 13000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcc'),
    ma: 'Ya004',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 8000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  }
]
```

4. Xuất danh sách xe máy Honda.

```
mongosh mongodb://127.0.0.1
qlXe> db.xe.find({ten:"Honda"})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc4'),
    ma: 'Hd002',
    ten: 'Honda',
    namsx: 1990,
    gia: 10000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc8'),
    ma: 'Hd001',
    ten: 'Honda',
    namsx: 1990,
    gia: 5000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bca'),
    ma: 'Hd003',
    ten: 'Honda',
    namsx: 1990,
    gia: 25000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcb'),
    ma: 'Hd004',
    ten: 'Honda',
    namsx: 1990,
    gia: 15000000,
    hinhanh: 'h1.jpg',
    loai: { maloai: '002Xemay', tenloai: 'Xe gắn máy' }
  }
]
```

5. Xuất danh sách xe máy Yamaha sản xuất năm 1992.


```
mongosh mongodb://127.0.0.1:27020/
> use xe
xe> db.xe.find({$and:[{ten:"Yamaha"},{namsx:1992}]})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc6'),
    ma: 'Ya002',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 14000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc7'),
    ma: 'Ya001',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 12000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bc9'),
    ma: 'Ya003',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 13000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcc'),
    ma: 'Ya004',
    ten: 'Yamaha',
    namsx: 1992,
    gia: 8000000,
    hinhanh: 'h2.jpg',
    loai: { maloi: '001Xemay', tenloai: 'Xe gắn máy' }
  }
]
```

6. Xuất danh sách xe máy Honda có giá từ 15.000.000 trở lên.

```
qlXe> db.xe.find({$and:[{ten:"Honda"},{gia:{$gte:15000000}}]})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bca'),
    ma: 'Hd003',
    ten: 'Honda',
    namsx: 1990,
    gia: 25000000,
    hinhanh: 'h1.jpg',
    loai: { maloi: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcb'),
    ma: 'Hd004',
    ten: 'Honda',
    namsx: 1990,
    gia: 15000000,
    hinhanh: 'h1.jpg',
    loai: { maloi: '002Xemay', tenloai: 'Xe gắn máy' }
  }
]
qlXe> |
```

7. Xuất ra các danh sách các xe có năm sản xuất là 1999.

```
qlXe> db.xe.find({namsx:1999})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc5'),
    ma: 'Fd006',
    ten: 'Ford',
    namsx: 1999,
    gia: 555000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  }
]
qlXe> |
```

8. Xuất ra danh sách xe ô tô sản xuất từ năm 2000.

```
qlXe> db.xe.find({$and:[{"loai.tenloai":"Ô tô"},{namsx:{$gte:2000}}]})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bcd'),
    ma: 'Fd010',
    ten: 'Ford',
    namsx: 2001,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bce'),
    ma: 'Fd110',
    ten: 'Ford',
    namsx: 2002,
    gia: 755000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcf'),
    ma: 'Fd005',
    ten: 'Ford',
    namsx: 2009,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  }
]
qlXe> |
```

9. Đếm có bao nhiêu xe có năm sản xuất 1992.

```
qlXe> db.xe.countDocuments({namsx:1992})
4
qlXe> |
```

10. Xuất ra các xe có giá trên 25.000.000 dưới 655.000.000.

```
mongosh mongodb://127.0.0.1:27027
qlXe> db.xe.find({gia:{$gte:25000000,$lte:655000000}})
[
  {
    _id: ObjectId('658cdf4fe428794a57451bc5'),
    ma: 'Fd006',
    ten: 'Ford',
    namsx: 1999,
    gia: 555000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bca'),
    ma: 'Hd003',
    ten: 'Honda',
    namsx: 1990,
    gia: 25000000,
    hinhanh: 'h1.jpg',
    loai: { maloi: '002Xemay', tenloai: 'Xe gắn máy' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcd'),
    ma: 'Fd010',
    ten: 'Ford',
    namsx: 2001,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  },
  {
    _id: ObjectId('658cdf4fe428794a57451bcf'),
    ma: 'Fd005',
    ten: 'Ford',
    namsx: 2009,
    gia: 655000000,
    hinhanh: 'h3.jpg',
    loai: { maloi: '0010to', tenloai: 'Ô tô' }
  }
]
```

11. Xuất ra tên và giá của tất cả xe 'Yamaha'

```
qlXe> db.xe.find({ten:"Yamaha"},{ten:1,gia:1,_id:0})
[
  { ten: 'Yamaha', gia: 14000000 },
  { ten: 'Yamaha', gia: 12000000 },
  { ten: 'Yamaha', gia: 13000000 },
  { ten: 'Yamaha', gia: 8000000 }
]
qlXe> |
```

Bài 4. Truy vấn collection restaurants

Sử dụng file restaurants.bson đưa dữ liệu vào MongoDB, thực hiện các truy vấn theo yêu cầu.
restaurants

```
{address: {building: '7715', coord: [ -73.9973325, 40.61174889999999 ], street: '18  
Avenue', zipcode: '11214'}, borough: 'Brooklyn', cuisine: 'American ', grades: [{date:  
ISODate("2014-04-16T00:00:00.000Z"), grade: 'A', score: 5},{date: ISODate("2013-04-  
23T00:00:00.000Z"), grade: 'A', score: 2},{date: ISODate("2012-04-24T00:00:00.000Z"), grade:  
'A', score: 5},{date: ISODate("2011-12-16T00:00:00.000Z"), grade: 'A', score: 2}], name: 'C & C  
Catering Service',  
restaurant_id: '40357437'}
```

1. Hiển thị tất cả các documents.

```
mongosh mongodb://127.0.0.1:27020/restaurant
switched to db restaurants
restaurants> db.restaurants.find()
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce58'),
    address: {
      building: '7114',
      coord: [ -73.9068506, 40.6199034 ],
      street: 'Avenue U',
      zipcode: '11234'
    },
    borough: 'Brooklyn',
    cuisine: 'Delicatessen',
    grades: [
      {
        date: ISODate('2014-05-29T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2014-01-14T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2013-08-03T00:00:00.000Z'),
        grade: 'A',
        score: 8
      },
      {
        date: ISODate('2012-07-18T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2012-03-09T00:00:00.000Z'),
        grade: 'A',
        score: 13
      }
    ]
  }
]
```

```
    },
    {
      date: ISODate('2012-03-09T00:00:00.000Z'),
      grade: 'A',
      score: 13
    },
    {
      date: ISODate('2011-10-14T00:00:00.000Z'),
      grade: 'A',
      score: 9
    }
  ],
  name: "Wilken'S Fine Food",
  restaurant_id: '40356483'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce59'),
  address: {
    building: '2206',
    coord: [ -74.1377286, 40.6119572 ],
    street: 'Victory Boulevard',
    zipcode: '10314'
  },
  borough: 'Staten Island',
  cuisine: 'Jewish/Kosher',
  grades: [
    {
      date: ISODate('2014-10-06T00:00:00.000Z'),
      grade: 'A',
      score: 9
    },
    {
      date: ISODate('2014-05-20T00:00:00.000Z'),
      grade: 'A',
      score: 12
    },
    {
      date: ISODate('2013-04-04T00:00:00.000Z'),
      grade: 'A',
```

```
mongosh mongodb://127.0.0.1:27020/restaurant
> use restaurant
switched to db restaurant
> find()
{
  _id: ObjectId('658cf72d52088adb4ccbce5a'),
  address: {
    building: '2780',
    coord: [ -73.98241999999999, 40.579505 ],
    street: 'Stillwell Avenue',
    zipcode: '11224'
  },
  borough: 'Brooklyn',
  cuisine: 'American',
  grades: [
    {
      date: ISODate('2014-06-10T00:00:00.000Z'),
      grade: 'A',
      score: 5
    },
    {
      date: ISODate('2013-06-05T00:00:00.000Z'),
      grade: 'A',
      score: 7
    },
    {
      date: ISODate('2012-04-13T00:00:00.000Z'),
      grade: 'A',
      score: 12
    },
    {
      date: ISODate('2011-10-12T00:00:00.000Z'),
      grade: 'A',
      score: 9
    }
  ],
  name: 'Kosher Island',
  restaurant_id: '40356442'
}
```

...


```
mongosh mongodb://127.0.0.1:27020/restaurant
> use restaurant
switched to db restaurant
> find()
{
  _id: ObjectId('658cf72d52088adb4ccbce6b'),
  address: {
    building: '97-22',
    coord: [ -73.8601152, 40.7311739 ],
    street: '63 Road',
    zipcode: '11374'
  },
  borough: 'Queens',
  cuisine: 'Jewish/Kosher',
  grades: [
    {
      date: ISODate('2014-11-24T00:00:00.000Z'),
      grade: 'Z',
      score: 20
    },
    {
      date: ISODate('2013-01-17T00:00:00.000Z'),
      grade: 'A',
      score: 13
    },
    {
      date: ISODate('2012-08-02T00:00:00.000Z'),
      grade: 'A',
      score: 13
    },
    {
      date: ISODate('2011-12-15T00:00:00.000Z'),
      grade: 'B',
      score: 25
    }
  ],
  name: 'Tov Kosher Kitchen',
  restaurant_id: '40356068'
}
]
Type "it" for more
restaurants>
```

2. Hiển thị tất cả các document, nhưng chỉ xuất các field `_id`, `restaurant_id`, `name`, `borough` và `cuisine`.

```
mongosh mongodb://127.0.0.1:27027/restaurant
restaurants> db.restaurants.find({}, {_id:1,restaurant_id:1,name:1,borough:1,cuisine:1})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce58'),
    borough: 'Brooklyn',
    cuisine: 'Delicatessen',
    name: 'Wilken'S Fine Food',
    restaurant_id: '40356483'
  },
  {
    _id: ObjectId('658cf72d52088adb4ccbce59'),
    borough: 'Staten Island',
    cuisine: 'Jewish/Kosher',
    name: 'Kosher Island',
    restaurant_id: '40356442'
  },
  {
    _id: ObjectId('658cf72d52088adb4ccbce5a'),
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Riviera Caterer',
    restaurant_id: '40356018'
  },
  {
    _id: ObjectId('658cf72d52088adb4ccbce5b'),
    borough: 'Brooklyn',
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'Taste The Tropics Ice Cream',
    restaurant_id: '40356731'
  },
  {
    _id: ObjectId('658cf72d52088adb4ccbce5c'),
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Regina Caterers',
    restaurant_id: '40356649'
  },
]
```

```
mongosh mongodb://127.0.0.1:27020/restaurant
> use restaurant
switched to db restaurant
> find()
{
  _id: ObjectId('658cf72d52088adb4ccbce5d'),
  borough: 'Bronx',
  cuisine: 'American ',
  name: 'Wild Asia',
  restaurant_id: '40357217'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce5e'),
  borough: 'Brooklyn',
  cuisine: 'Chinese',
  name: 'May May Kitchen',
  restaurant_id: '40358429'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce5f'),
  borough: 'Brooklyn',
  cuisine: 'American ',
  name: 'C & C Catering Service',
  restaurant_id: '40357437'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce60'),
  borough: 'Brooklyn',
  cuisine: 'Jewish/Kosher',
  name: 'Seuda Foods',
  restaurant_id: '40360045'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce61'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: '1 East 66Th Street Kitchen',
  restaurant_id: '40359480'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce62'),
  borough: 'Queens',
  cuisine: 'American ',
  name: 'The Greenhouse',
  restaurant_id: '40359481'
}
```

```
mongosh mongodb://127.0.0.1:27020/restaurant?authSource=admin
```

```
{
  "_id": "658cf72d52088adb4ccbce62",
  "borough": "Queens",
  "cuisine": "Ice Cream, Gelato, Yogurt, Ices",
  "name": "Carvel Ice Cream",
  "restaurant_id": "40361322"
},
{
  "_id": "658cf72d52088adb4ccbce63",
  "borough": "Brooklyn",
  "cuisine": "Ice Cream, Gelato, Yogurt, Ices",
  "name": "Carvel Ice Cream",
  "restaurant_id": "40360076"
},
{
  "_id": "658cf72d52088adb4ccbce64",
  "borough": "Brooklyn",
  "cuisine": "Hamburgers",
  "name": "Wendy'S",
  "restaurant_id": "30112340"
},
{
  "_id": "658cf72d52088adb4ccbce65",
  "borough": "Brooklyn",
  "cuisine": "American",
  "name": "The Movable Feast",
  "restaurant_id": "40361606"
},
{
  "_id": "658cf72d52088adb4ccbce66",
  "borough": "Manhattan",
  "cuisine": "Irish",
  "name": "Dj Reynolds Pub And Restaurant",
  "restaurant_id": "30191841"
},
{
  "_id": "658cf72d52088adb4ccbce67",
  "borough": "Queens",
  "cuisine": "Delicatessen",
  "name": "Sal'S Deli",
  "restaurant_id": "40361322"
}
```

```
mongosh mongodb://127.0.0.1:27020/restaurant
> use restaurant
switched to db restaurant
> find()
{
  _id: ObjectId('658cf72d52088adb4ccbce67'),
  borough: 'Queens',
  cuisine: 'Delicatessen',
  name: "Sal'S Deli",
  restaurant_id: '40361618'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce68'),
  borough: 'Manhattan',
  cuisine: 'Delicatessen',
  name: "Bully'S Deli",
  restaurant_id: '40361708'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce69'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: 'Glorious Food',
  restaurant_id: '40361521'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce6a'),
  borough: 'Queens',
  cuisine: 'Delicatessen',
  name: "Steve Chu'S Deli & Grocery",
  restaurant_id: '40361998'
},
{
  _id: ObjectId('658cf72d52088adb4ccbce6b'),
  borough: 'Queens',
  cuisine: 'Jewish/Kosher',
  name: 'Tov Kosher Kitchen',
  restaurant_id: '40356068'
}
]
Type "it" for more
restaurants>
```

3. Hiển thị tất cả các document, nhưng chỉ xuất các field restaurant_id, name, borough và zipcode.

4. Hiển thị 5 document đầu tiên với field borough có giá trị là 'Bronx'.

```
mongosh mongodb://127.0.0.1
restaurants> db.restaurants.find({borough:"Bronx"}).limit(5)
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce5d'),
    address: {
      building: '2300',
      coord: [ -73.8786113, 40.8502883 ],
      street: 'Southern Boulevard',
      zipcode: '10460'
    },
    borough: 'Bronx',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-05-28T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2013-06-19T00:00:00.000Z'),
        grade: 'A',
        score: 4
      },
      {
        date: ISODate('2012-06-15T00:00:00.000Z'),
        grade: 'A',
        score: 3
      }
    ],
    name: 'Wild Asia',
    restaurant_id: '40357217'
  },
  {
    _id: ObjectId('658cf72d52088adb4ccbce6d'),
    address: {
      building: '1007',
      coord: [ -73.856077, 40.848447 ],
      street: 'Morris Park Ave',
      zipcode: '10462'
    },
    borough: 'Bronx',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-05-28T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2013-06-19T00:00:00.000Z'),
        grade: 'A',
        score: 4
      },
      {
        date: ISODate('2012-06-15T00:00:00.000Z'),
        grade: 'A',
        score: 3
      }
    ],
    name: 'Wild Asia',
    restaurant_id: '40357217'
  }
]
```

5. Hiển thị 5 document tiếp theo sau khi bỏ qua 5 document đầu tiên với field borough có giá trị là 'Bronx'.

```
mongosh mongodb://127.0.0.1:27027/restaurant
> use restaurant
restaurant> db.restaurants.find({borough:"Bronx"}).limit(5).skip(5)
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce96'),
    address: {
      building: '277',
      coord: [ -73.8941893, 40.8634684 ],
      street: 'East Kingsbridge Road',
      zipcode: '10458'
    },
    borough: 'Bronx',
    cuisine: 'Chinese',
    grades: [
      {
        date: ISODate('2014-03-03T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2013-09-26T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2013-03-19T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2012-08-29T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2011-08-17T00:00:00.000Z'),
        grade: 'A',
        score: 13
      }
    ]
  }
],
```

6. Hiển thị các document có giá trị score trong field grades > 90.


```
restaurants> db.restaurants.find({"grades.score":{"$gt:90}})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbcfb5'),
    address: {
      building: '65',
      coord: [ -73.9782725, 40.7624022 ],
      street: 'West 54 Street',
      zipcode: '10019'
    },
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-08-22T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2014-03-28T00:00:00.000Z'),
        grade: 'C',
        score: 131
      },
      {
        date: ISODate('2013-09-25T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2013-04-08T00:00:00.000Z'),
        grade: 'B',
        score: 25
      },
      {
        date: ISODate('2012-10-15T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {

```

```
mongosh mongodb://127.0.0.1:27027/restaurant
restaurants> db.restaurants.find({"grades.score":{$gt:80,$lt:100}})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbcfb5'),
    address: {
      building: '65',
      coord: [ -73.9782725, 40.7624022 ],
      street: 'West 54 Street',
      zipcode: '10019'
    },
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-08-22T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2014-03-28T00:00:00.000Z'),
        grade: 'C',
        score: 131
      },
      {
        date: ISODate('2013-09-25T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2013-04-08T00:00:00.000Z'),
        grade: 'B',
        score: 25
      },
      {
        date: ISODate('2012-10-15T00:00:00.000Z'),
        grade: 'A',
        score: 11
      }
    ]
  }
]
```

8. Hiển thị các document có vĩ độ (trong coord) < -95.754168.

```
mongosh mongodb://127.0.0.1:27027/restaurant
> use restaurant
> db.restaurants.find({"address.coord.0":{"$lt":-95.754168}})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbd4a2'),
    address: {
      building: '3707',
      coord: [ -101.8945214, 33.5197474 ],
      street: '82 Street',
      zipcode: '11372'
    },
    borough: 'Queens',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-06-04T00:00:00.000Z'),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate('2013-11-07T00:00:00.000Z'),
        grade: 'B',
        score: 19
      },
      {
        date: ISODate('2013-05-17T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2012-08-29T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2012-04-03T00:00:00.000Z'),
        grade: 'A',
        score: 12
      }
    ]
  }
]
```

9. Hiển thị các document thoả các điều kiện field cuisine không được là 'American', score của field grade > 70 và vĩ độ < -65.754168.
11. Hiển thị các document thoả các điều kiện field cuisine không được là 'American', score của field grade > 70 và kinh độ < -65.754168.

```
mongosh mongodb://127.0.0.1:27027/restaurant
2 |
restaurants> db.restaurants.find({$and:[{cuisine:{$ne:"American"}},{ "grades.
score":{$gt:70}},{ "address.coord.0":{$lt:-65.754168}}]})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbcfb5'),
    address: {
      building: '65',
      coord: [ -73.9782725, 40.7624022 ],
      street: 'West 54 Street',
      zipcode: '10019'
    },
    borough: 'Manhattan',
    cuisine: 'American',
    grades: [
      {
        date: ISODate('2014-08-22T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2014-03-28T00:00:00.000Z'),
        grade: 'C',
        score: 131
      },
      {
        date: ISODate('2013-09-25T00:00:00.000Z'),
        grade: 'A',
        score: 11
      },
      {
        date: ISODate('2013-04-08T00:00:00.000Z'),
        grade: 'B',
        score: 25
      },
      {
        date: ISODate('2012-10-15T00:00:00.000Z'),
```

12. Hiển thị các document thoả các điều kiện field cuisine không được là 'American', giá trị grade của field grades là 'A', và field borough không được là 'Brooklyn'. Sau đó sắp xếp các document theo thứ tự tăng dần của field cuisine.

```

restaurants> db.restaurants.find({$and:[{cuisine:{$ne:"American"}},{ "grades.
grade": 'A'},{borough:{$ne:"Brooklyn"}}]}).sort({cuisine:1})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbd543'),
    address: {
      building: '1345',
      coord: [ -73.959249, 40.768076 ],
      street: '2 Avenue',
      zipcode: '10021'
    },
    borough: 'Manhattan',
    cuisine: 'Afghan',
    grades: [
      {
        date: ISODate('2014-10-07T00:00:00.000Z'),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate('2013-10-23T00:00:00.000Z'),
        grade: 'A',
        score: 8
      },
      {
        date: ISODate('2012-10-26T00:00:00.000Z'),
        grade: 'A',
        score: 13
      },
      {
        date: ISODate('2012-04-26T00:00:00.000Z'),
        grade: 'A',
        score: 7
      },
      {
        date: ISODate('2012-01-12T00:00:00.000Z'),
        grade: 'P',

```

12. Hiển thị các document với các field restaurant_id, name, borough, cuisine với name có chứa 3 ký tự bắt đầu là 'Wil'.

```
restaurants> db.restaurants.find({name:{$regex: '^Wil'}},{_id:0,restaurant_id:1,name:1,borough:1,cuisine:1})
[
  {
    borough: 'Brooklyn',
    cuisine: 'Delicatessen',
    name: 'Wilken'S Fine Food',
    restaurant_id: '40356483'
  },
  {
    borough: 'Bronx',
    cuisine: 'American ',
    name: 'Wild Asia',
    restaurant_id: '40357217'
  },
  {
    borough: 'Bronx',
    cuisine: 'Pizza',
    name: 'Wilbel Pizza',
    restaurant_id: '40871979'
  }
]
restaurants> |
```

13. Hiển thị các document với các field restaurant_id, name, borough, cuisine với name có chứa 3 ký tự cuối cùng là 'ces'.

```
restaurants> db.restaurants.find({name:{$regex: 'ces$'}},{_id:0,restaurant_id:1,name:1,borough:1,cuisine:1})
[
  { borough: 'Manhattan', cuisine: 'American ', name: 'Pieces' },
  {
    borough: 'Queens',
    cuisine: 'American ',
    name: 'S.M.R Restaurant Services'
  },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Good Shepherd Services'
  },
  {
    borough: 'Queens',
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'The Ice Box-Ralph'S Famous Italian Ices'
  },
  { borough: 'Manhattan', cuisine: 'American ', name: 'Re: Sources' },
  { borough: 'Brooklyn', cuisine: 'Jewish/Kosher', name: 'Alices' }
]
restaurants> |
```

15. Hiển thị các document với các field restaurant_id, name, borough, cuisine với name có chứa 3 ký tự 'Reg'.

```

restaurants> db.restaurants.find({name:{$regex:'Reg'}},{_id:0,restaurant_idr
restaurant_id:1,name:1,borough:1,cuisine:1})
[
  {
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Regina Caterers'
  },
  {
    borough: 'Manhattan',
    cuisine: 'Café/Coffee/Tea',
    name: 'Caffe Reggio'
  },
  { borough: 'Manhattan', cuisine: 'American ', name: 'Regency Hotel' },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Regency Whist Club'
  },
  { borough: 'Queens', cuisine: 'American ', name: 'Rego Park Cafe' },
  { borough: 'Queens', cuisine: 'Pizza', name: 'Regina Pizza' },
  {
    borough: 'Manhattan',
    cuisine: 'American ',
    name: 'Regal Entertainment Group'
  }
]
restaurants> |

```

16. Hiển thị các document với borough có giá trị là 'Bronx' và cuisine có giá trị là 'American' hoặc 'Chinese'.

```

restaurants> db.restaurants.find({$and:[{borough:"Bronx"},{cuisine:{$in:["Am
erican","Chinese"]}}]})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce7a'),
    address: {
      building: '1236',
      coord: [ -73.8893654, 40.81376179999999 ],
      street: '238 Spofford Ave',
      zipcode: '10474'
    },
    borough: 'Bronx',
    cuisine: 'Chinese',
    grades: [
      {
        date: ISODate('2013-12-30T00:00:00.000Z'),
        grade: 'A',
        score: 8
      },
      {
        date: ISODate('2013-01-08T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2012-06-12T00:00:00.000Z'),
        grade: 'B',
        score: 15
      }
    ],
    name: 'Happy Garden',
    restaurant_id: '40363289'
  },
  {

```

17. Hiển thị các document với điều kiện field borough có giá trị 'Staten Island', 'Queens' hoặc 'Bronx' hoặc 'Brooklyn', và chỉ xuất ra các field restaurant_id, name, borough, cuisine.

18. Hiển thị các document với field borough không được là các giá trị 'Staten Island', 'Queens' hoặc 'Brooklyn', và chỉ xuất các field restaurant_id, name, borough, cuisine.

```
restaurants> db.restaurants.find({borough:{$in:["Staten Island","Queens","Brooklyn"]}}, {_id:0, restaurant:1, name:1, borough:1, cuisine:1})
[
  {
    borough: 'Staten Island',
    cuisine: 'Jewish/Kosher',
    name: 'Kosher Island'
  },
  {
    borough: 'Queens',
    cuisine: 'Ice Cream, Gelato, Yogurt, Ices',
    name: 'Carvel Ice Cream'
  },
  { borough: 'Queens', cuisine: 'Delicatessen', name: "Sal'S Deli" },
  {
    borough: 'Queens',
    cuisine: 'Delicatessen',
    name: "Steve Chu'S Deli & Grocery"
  },
  {
    borough: 'Queens',
    cuisine: 'Jewish/Kosher',
    name: 'Tov Kosher Kitchen'
  },
  {
    borough: 'Queens',
    cuisine: 'American ',
    name: 'Brunos On The Boulevard'
  },
  { borough: 'Queens', cuisine: 'Chinese', name: 'Ho Mei Restaurant' },
  {
    borough: 'Staten Island',
```

Brooklyn', và chỉ xuất các field restaurant_id, name, borough, cuisine.

19. Hiển thị các document với giá trị score của field grades không lớn hơn 10, chỉ xuất các field restaurant_id, name, borough, cuisine.


```

restaurants> db.restaurants.find({$or:[{cuisine:{$nin:["American","Chinese"]}
}}, {name:{$regex: '^Wil'}}]})
[
  {
    _id: ObjectId('658cf72d52088adb4ccbce58'),
    address: {
      building: '7114',
      coord: [ -73.9068506, 40.6199034 ],
      street: 'Avenue U',
      zipcode: '11234'
    },
    borough: 'Brooklyn',
    cuisine: 'Delicatessen',
    grades: [
      {
        date: ISODate('2014-05-29T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2014-01-14T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2013-08-03T00:00:00.000Z'),
        grade: 'A',
        score: 8
      },
      {
        date: ISODate('2012-07-18T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2012-03-09T00:00:00.000Z'),

```

21. Hiển thị các document thoả các điều kiện field grade = 'A', score = 11, date= "2014-08-11T00:00:00Z"

```

restaurants> db.restaurants.find({$and:[{"grades.grade":"A"}, {"grades.score":11}, {"grades.date":new Date("2014-08-11T00:00:00Z")}]}])
[
  {
    _id: ObjectId('658cf72d52088adb4ccbc7'),
    address: {
      building: '358',
      coord: [ -73.963506, 40.758273 ],
      street: 'East 57 Street',
      zipcode: '10022'
    },
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2014-08-11T00:00:00.000Z'),
        grade: 'A',
        score: 13
      },
      {
        date: ISODate('2013-07-22T00:00:00.000Z'),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate('2013-03-14T00:00:00.000Z'),
        grade: 'A',
        score: 12
      },
      {
        date: ISODate('2012-07-02T00:00:00.000Z'),
        grade: 'A',
        score: 11
      }
    ]
  }
]

```

22. Hiển thị các document có phần tử thứ 2 của mảng grades có grade là "A", score là 9 và date là "2014-08-11T00:00:00Z"

```

restaurants> db.restaurants.find({$and:[{"grades.1.grade":"A"}, {"grades.1.score":9}, {"grades.1.date":new Date("2014-08-11T00:00:00Z")}]}))
[
  {
    _id: ObjectId('658cf72d52088adb4ccbd483'),
    address: {
      building: '26',
      coord: [ -73.96936149999999, 40.7658603 ],
      street: 'East 63 Street',
      zipcode: '10065'
    },
    borough: 'Manhattan',
    cuisine: 'American ',
    grades: [
      {
        date: ISODate('2015-01-12T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2014-08-11T00:00:00.000Z'),
        grade: 'A',
        score: 9
      },
      {
        date: ISODate('2014-01-14T00:00:00.000Z'),
        grade: 'A',
        score: 13
      },
      {
        date: ISODate('2013-02-07T00:00:00.000Z'),
        grade: 'A',
        score: 10
      },
      {
        date: ISODate('2012-04-30T00:00:00.000Z'),
        grade: 'A',

```

Bài 5. Truy vấn collection sinhvien, lop

Cho thông tin sinhvien và lop như sau:

sinhvien (file dsSinhvien.bson)

```

{
  _id: ObjectId("6131a2a278b3264add832588"),
  diemTB: 9.5,
  dsDienthoai: [ '0947536844' ],
  email: 'Chuot@gmail.com',
  gioitinh: 'Nam',
  ho: 'Nguyễn Việt',
  malop: 'DHKTPM15B',
  mssv: '19296011',
  ngaysinh: ISODate("2001-09-15T00:00:00.000Z"),

```

```
ten: 'Anh'
}
lop (file dsLophoc.bson)
{
  _id: ObjectId("6131e5e97cf137f22a235cd0"),
  macn: 'KHMT',
  mslop: 'DHKHMT15A',
  sisoDukien: 80,
  tenlop: 'Đại học Khoa học Máy tính 15A'
}
```

Thực hiện các câu truy vấn sau:

1. Đếm số sinh viên có điểm trung bình từ 5 trở lên.

```
qlsv> db.sinhvien.countDocuments({diemTB:{$gte:5}})
220
qlsv> |
```

2. Đếm số sinh viên có điểm trung bình từ 3.0 đến nhỏ hơn 6.5.

```
qlsv> db.sinhvien.countDocuments({diemTB:{$gte:3,$lt:6.5}})
102
qlsv> |
```

3. Liệt kê danh sách sinh viên không có số điện thoại hoặc email.

```
qlsv> db.sinhvien.find({$or:[{dsDienthoai:{$exists:false}},{email:{$exists:false}}]})
qlsv> |
```

4. Liệt kê danh sách sinh viên có từ 2 số điện thoại trở lên.

```

qlsv> db.sinhvien.find({"dsDienthoai.1":{"$exists:true"}})
[
  {
    _id: ObjectId('6131a2a278b3264add83259a'),
    diemTB: 6,
    dsDienthoai: [ '0978289481', '0819490559' ],
    email: 'HaoNguyen.241@gmail.com',
    gioitinh: 'Nam',
    ho: 'Nguyễn Trọng',
    malop: 'DHKTPM13A',
    mssv: '17595581',
    ngaysinh: ISODate('1999-01-26T00:00:00.000Z'),
    ten: 'Hào'
  },
  {
    _id: ObjectId('6131a2a278b3264add83259d'),
    diemTB: 4,
    dsDienthoai: [ '0366497800', '0387773717' ],
    email: 'tino@gmail.com',
    gioitinh: 'Nam',
    ho: 'Nguyễn Chí',
    malop: 'DHKTPM15B',
    mssv: '19608231',
    ngaysinh: ISODate('2001-08-30T00:00:00.000Z'),
    ten: 'Hiếu'
  },
  {
    _id: ObjectId('6131a2a278b3264add8325a4'),
    diemTB: 10,
    dsDienthoai: [ '0973704864', '0393575926' ],
    email: 'hoangk@gmail.com',
    gioitinh: 'Nam',
    ho: 'Nguyễn Văn',
    malop: 'DHKTPM15B',
    mssv: '19363741',
    ngaysinh: ISODate('2001-11-21T00:00:00.000Z'),
    ten: 'Hoàng'
  }
]

```

5. Liệt kê danh sách sinh viên có lót chữ “Văn”, không phân biệt chữ hoa chữ thường.

```

qlsv> db.sinhvien.find({ho:{$regex:'Văn', $options:'i'}})
[
  {
    _id: ObjectId('6131a2a278b3264add83258d'),
    diemTB: 6.5,
    dsDienthoai: [ '0967127210' ],
    email: 'chinhhoang@gmail.com',
    gioitinh: 'Nam',
    ho: 'Hoàng Văn',
    malop: 'DHKTPM15B',
    mssv: '19131531',
    ngaysinh: ISODate('2001-12-22T00:00:00.000Z'),
    ten: 'Chinh'
  },
  {
    _id: ObjectId('6131a2a278b3264add83258e'),
    diemTB: 6,
    dsDienthoai: [ '0329807464' ],
    email: 'fantasiafer@gmail.com',
    gioitinh: 'Nam',
    ho: 'Phạm Văn',
    malop: 'DHKTPM15A',
    mssv: '19433931',
    ngaysinh: ISODate('2001-02-03T00:00:00.000Z'),
    ten: 'Chinh'
  },
  {
    _id: ObjectId('6131a2a278b3264add832595'),
    diemTB: 8.25,
    dsDienthoai: [ '0965494140' ],
    email: 'datngo2032@gmail.com',
    gioitinh: 'Nam',
    ho: 'Ngô Văn',
    malop: 'DHKTPM15B',
    mssv: '19636091',
    ngaysinh: ISODate('2001-04-18T00:00:00.000Z'),

```

6. Liệt kê danh sách sinh viên có họ tên chứa chữ “Minh”, không phân biệt chữ hoa chữ thường.


```
q\sv> db.sinhvien.find({$or:[{ho:{$regex:'Minh',$options:'i'}},{ten:{$regex:'Minh',$options:'i'}}]})
[
  {
    _id: ObjectId('6131a2a278b3264add83258c'),
    diemTB: 5.25,
    dsDienthoai: [ '0795815322' ],
    email: 'minhchaunguyen@gmail.com',
    gioitinh: 'Nữ',
    ho: 'Nguyễn Thị Minh',
    malop: 'DHKTPM15A',
    mssv: '19202821',
    ngaysinh: ISODate('2001-01-24T00:00:00.000Z'),
    ten: 'Châu'
  },
  {
    _id: ObjectId('6131a2a278b3264add83259c'),
    diemTB: 4.75,
    dsDienthoai: [ '0364973888' ],
    email: 'leminhhie7@gmail.com',
    gioitinh: 'Nam',
    ho: 'Lê Minh',
    malop: 'DHKTPM15B',
    mssv: '19729901',
    ngaysinh: ISODate('2001-08-15T00:00:00.000Z'),
    ten: 'Hiếu'
  },
  {
    _id: ObjectId('6131a2a278b3264add8325a9'),
    diemTB: 7.75,
    dsDienthoai: [ '0342409371' ],
    email: 'ndmhuy.c3tqcap@gmail.com',
    gioitinh: 'Nam',
    ho: 'Nguyễn Dương Minh',
    malop: 'DHKTPM15A',
    mssv: '19933421',
  }
]
```