

Result of lab 7

The image shows two screenshots of the MongoDB Compass interface. The top screenshot displays the 'students' collection in the 'mydatabase' database on 'localhost:27017'. It shows two documents with fields like '_id', 'name', 'age', and 'major'. The bottom screenshot shows the 'Restaurant' collection, displaying documents with fields like '_id', 'address', 'borough', 'cuisine', 'grades', 'name', and 'restaurant_id'. A notification at the bottom left of the second screenshot states 'Import completed. 3773 documents imported.'

Top Screenshot: students collection

Connections: localhost:27017

Documents: 2 | Aggregations | Schema | Indexes: 1 | Validation

Type a query: { field: 'value' } or [Generate query](#)

Buttons: ADD DATA, EXPORT DATA, UPDATE, DELETE

Display: 25 | 1-2 of 2

Document 1:

```
{
  "_id": ObjectId('6874747ca6b8873672b7a67f'),
  "name": "Hà",
  "age": 12,
  "major": "IT"
}
```

Document 2:

```
{
  "_id": ObjectId('68747489a6b8873672b7a680'),
  "name": "Quang",
  "age": 13,
  "major": "IT"
}
```

Bottom Screenshot: Restaurant collection

Connections: localhost:27017

Documents: 3.8K | Aggregations | Schema | Indexes: 1 | Validation

Type a query: { field: 'value' } or [Generate query](#)

Buttons: ADD DATA, EXPORT DATA, UPDATE, DELETE

Display: 25 | 1-25 of 3773

Document 1:

```
{
  "_id": ObjectId('6782122afa4945d74d0aaaf'),
  "address": {
    "borough": "Bronx",
    "cuisine": "Bakery"
  },
  "grades": [5],
  "name": "Morris Park Bake Shop",
  "restaurant_id": "30075445"
}
```

Document 2:

```
{
  "_id": ObjectId('6782122afa4945d74d0aaaf0'),
  "address": {
    "borough": "Brooklyn",
    "cuisine": "Hamburgers"
  },
  "grades": [4],
  "name": "Wendy's",
  "restaurant_id": "30112340"
}
```

Document 3:

```
{
  "_id": ObjectId('6782122afa4945d74d0aaaf1'),
  "address": {
    "borough": "Manhattan",
    "cuisine": "Irish"
  },
  "grades": [4],
  "name": "Dj Reynolds Pub And Restaurant",
  "restaurant_id": "30191841"
}
```

Document 4:

```
{
  "_id": ObjectId('6782122afa4945d74d0aaaf2'),
  "address": {
    "borough": "Manhattan",
    "cuisine": "Irish"
  },
  "grades": [4],
  "name": "Dj Reynolds Pub And Restaurant",
  "restaurant_id": "30191841"
}
```

Notification: Import completed. 3773 documents imported.

>_MONGOSH

test> db["Stock"].find()

>_MONGOSH

> use Lab6

< switched to db Lab6

> db["Restaurant"].find()

```
< {
  _id: ObjectId('6782122afa4945d74d0aaaef'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      date: 2014-03-03T00:00:00.000Z,
      grade: 'A',
      score: 2
    },
    {
      date: 2013-09-11T00:00:00.000Z,
      grade: 'A',
      score: 6
    },
    {
      date: 2013-01-24T00:00:00.000Z,
```

```
> db["Restaurant"].find({}, {restaurant_id: 1, name: 1, borough: 1, cuisine: 1})
```

```
< {
  _id: ObjectId('6782122afa4945d74d0aaaf1'),
  borough: 'Bronx',
  cuisine: 'Bakery',
  name: 'Morris Park Bake Shop',
  restaurant_id: '30075445'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf0'),
  borough: 'Brooklyn',
  cuisine: 'Hamburgers',
  name: 'Wendy'S',
  restaurant_id: '30112340'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf1'),
  borough: 'Manhattan',
  cuisine: 'Irish',
  name: 'Dj Reynolds Pub And Restaurant',
  restaurant_id: '30191841'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf2'),
  borough: 'Brooklyn',
  cuisine: 'American ',
  name: 'Riviera Caterer',
  restaurant_id: '40356018'
}
```

```
> db["Restaurant"].find({}, {restaurant_id: 1, name: 1, borough: 1, cuisine: 1})
```

```
< {
  _id: ObjectId('6782122afa4945d74d0aaaf1'),
  borough: 'Bronx',
  cuisine: 'Bakery',
  name: 'Morris Park Bake Shop',
  restaurant_id: '30075445'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf0'),
  borough: 'Brooklyn',
  cuisine: 'Hamburgers',
  name: 'Wendy'S',
  restaurant_id: '30112340'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf1'),
  borough: 'Manhattan',
  cuisine: 'Irish',
  name: 'Dj Reynolds Pub And Restaurant',
  restaurant_id: '30191841'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaf2'),
  borough: 'Brooklyn',
  cuisine: 'American ',
  name: 'Riviera Caterer',
  restaurant_id: '40356018'
}
```

```

> db["Restaurant"].find({}, { name: 1, address: 1 }).pretty()
< {
  _id: ObjectId('6782122afa4945d74d0aaaef'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  name: 'Morris Park Bake Shop'
}
{
  _id: ObjectId('6782122afa4945d74d0aaaef0'),
  address: {
    building: '469',
    coord: [
      -73.961704,
      40.662942
    ],
    street: 'Flatbush Avenue',
    zipcode: '11225'
  },
}

```

```

> db["Restaurant"].find({borough: "Bronx"})
< {
  _id: ObjectId('6782122afa4945d74d0aaaef'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      date: 2014-03-03T00:00:00.000Z,
      grade: 'A',
      score: 2
    },
    {
      date: 2013-09-11T00:00:00.000Z,
      grade: 'A',
      score: 6
    },
  ],
}

```

```

> db["Restaurant"].find({cuisine: "Bakery"})
< {
  _id: ObjectId('6782122afa4945d74d0aaaf'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      date: 2014-03-03T00:00:00.000Z,
      grade: 'A',
      score: 2
    },
    {
      date: 2013-09-11T00:00:00.000Z,
      grade: 'A',
      score: 6
    },
    {
      date: 2013-01-24T00:00:00.000Z,
      grade: 'A',
      score: 10
    }
  ]
}

```

```

> db["Restaurant"].find({borough: "Bronx"}).limit(5)
< {
  _id: ObjectId('6782122afa4945d74d0aaaf'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      date: 2014-03-03T00:00:00.000Z,
      grade: 'A',
      score: 2
    },
    {
      date: 2013-09-11T00:00:00.000Z,
      grade: 'A',
      score: 6
    },
    {

```

```
> db["Restaurant"].find({borough: "Bronx"}).skip(5)
< {
  _id: ObjectId('6782122afa4945d74d0aab2c'),
  address: {
    building: '658',
    coord: [
      -73.81363999999999,
      40.82941100000001
    ],
    street: 'Clarence Ave',
    zipcode: '10465'
  },
  borough: 'Bronx',
  cuisine: 'American ',
  grades: [
    {
      date: 2014-06-21T00:00:00.000Z,
      grade: 'A',
      score: 5
    },
    {
      date: 2012-07-11T00:00:00.000Z,
      grade: 'A',
      score: 10
    }
  ],
  name: 'Manhem Club',
  restaurant_id: '40364363'
```

```
> db["Restaurant"].find({borough:"Manhattan", "grades.grade":"A"}).limit(10).pretty()
< {
  _id: ObjectId('6782122afa4945d74d0aaaf1'),
  address: {
    building: '351',
    coord: [
      -73.98513559999999,
      40.7676919
    ],
    street: 'West 57 Street',
    zipcode: '10019'
  },
  borough: 'Manhattan',
  cuisine: 'Irish',
  grades: [
    {
      date: 2014-09-06T00:00:00.000Z,
      grade: 'A',
      score: 2
    },
    {
      date: 2013-07-22T00:00:00.000Z,
      grade: 'A',
      score: 11
    },
    {
      date: 2012-07-31T00:00:00.000Z,
      grade: 'A',

```

```

>
// Lấy danh sách các borough duy nhất
const boroughs = db.Restaurant.distinct("borough")

// Với mỗi borough, đếm số document và in ra
boroughs.forEach(b => {
  const cnt = db.Restaurant.countDocuments({ borough: b })
  print(b + ": " + cnt)
})
< Bronx: 310
< Brooklyn: 684
< Manhattan: 1883
< Queens: 738
< Staten Island: 158
Lab6> |

```

```

db["Restaurant"].find({borough:"Queens"}).count()
738

```

```

> db["Restaurant"].find({"grades.score": {$gte: 90}}).count()
< 3

```

```

> db["Restaurant"].find({"grades.score": {$gte: 80,$lt: 100}}).count()
< 5

```

```

> db["Restaurant"].find({ "address.coord.1": { $lt: -95.754168 } }).count()
< 0

```

```

> db["Restaurant"].find({"grades.grade" : {$lt:10}}).count()
< 0

```

```

> db.Restaurant.find({
  borough: "Queens",
  cuisine: { $ne: "American" }
}).count()
< 738
Lab6> |

```

```
> db.Restaurant.find(
  {
    name: /^M/,           // name bắt đầu bằng "M"
    "grades.score": { $gt: 100 } // ít nhất một score > 100
  },
  {
    restaurant_id: 1,
    name:          1,
    borough:       1,
    cuisine:       1,
    _id:           0      // ẩn trường _id nếu không cần
  }
).pretty()
< {
  borough: 'Manhattan',
  cuisine: 'American ',
  name:    "Murals On 54/Randolphs'S",
  restaurant_id: '40372466'
}
```


Student List

[Add New Student](#) [Show Excellent Students](#) [Show Top Student](#)

<input type="text" value="Enter exact name"/>	<input type="button" value="Search by Name"/>
<input type="text" value="Enter name for fuzzy search"/>	<input type="button" value="Fuzzy Search"/>
<input type="button" value="All Majors"/> <input type="button" value="Filter by Major"/>	
Min Age: <input type="text"/>	Max Age: <input type="text"/> <input type="button" value="Filter by Age"/>
<input type="button" value="All Genders"/> <input type="button" value="Filter by Gender"/>	

Students per Major

Thêm Sinh Viên Mới

Họ Tên:	<input type="text" value="PHAM MANH HA"/>
Tuổi:	<input type="text" value="12"/>
Giới Tính:	<input type="text" value="male"/>
Chuyên Ngành:	<input type="text" value="IT"/>
Điểm Toán:	<input type="text" value="6"/>
Điểm Văn:	<input type="text" value="6"/>
Điểm Anh:	<input type="text" value="6"/>
<input type="button" value="Thêm Sinh Viên"/>	

Student List

[Add New Student](#) [Show Excellent Students](#) [Show Top Student](#)

Enter exact name	Search by Name
Enter name for fuzzy search	Fuzzy Search
All Majors ▾	Filter by Major
Min Age: <input type="text"/>	Max Age: <input type="text"/>
All Genders ▾	Filter by Gender

- PHAM MANH HA - Age: 12 - Gender: mail - Major: IT - Scores: Math: 6, Literature: 6, English: 6 - GPA: 6.00 - Rank: Average [Edit](#) [Delete](#)
- PHAM MANH HA - Age: 12 - Gender: male - Major: IT - Scores: Math: 6, Literature: 6, English: 6 - GPA: 6.00 - Rank: Average [Edit](#) [Delete](#)

Students per Major

- IT: 2

Compass

My Queries

CONNECTIONS (1)

Search connections

localhost:27017

Lab6

Restaurant

admin

config

local

startup_log

mydatabase

students

students_db

Chapter6_Students

Chapter6_Students

localhost:27017 > students_db > Chapter6_Students

Documents 1 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

ADD DATA

EXPORT DATA

UPDATE

DELETE

25 1-1 of 1

```
_id: ObjectId('6875265175ca2f2acc72e85d')
name: "PHAM MANH HA"
age: 12
gender: "mail"
major: "IT"
math: 6
literature: 6
english: 6
```