**MSSV: 19127333**

**HỌ VÀ TÊN: Nguyễn Trung Anh**

BTTH2: MONGODB

Bài1:

Viết script json như sau:

{

    "address":{

        "building":"1007",

        "coord":[-73.856077,40.848447],

        "street":"Morris Park Ave",

        "zipcode":"10462"

    },

    "borough":"Bronx",

    "cuisine":"Bakery",

    "grades":[

        {"date":{"$date":1393804800000},"grade":"A","score":2},

        {"date":{"$date":1378857600000},"grade":"A","score":6},

        {"date":{"$date":1358985600000},"grade":"A","score":10},

        {"date":{"$date":1322006400000},"grade":"A","score":9},

        {"date":{"$date":1299715200000},"grade":"B","score":14}

    ],

    "name":"Morris Park Bake Shop",

    "restaurant\_id":"30075445"

}

Sau đó lưu lại với tên restaurants.json

(Lưu ý đã cài mongoSH,Mongo Commandline tool)

Mở mongo CML TOOL, import file restaurants.json.

Text

Description automatically generated

Mongo compass đã có collection restaurant, sau đây sẽ là những câu truy vấn được thực hiện trên mongosh.

Graphical user interface, text, application, email

Description automatically generated

-Cho biết tất cả documents trong collection restaurants

db.restaurants.find({})

Text

Description automatically generated

2. Cho biết restaurant\_id, name, borough và cuisine của tất cả document trong restaurants

db.restaurants.find({}, {restaurant\_id: 1, name: 1, borough: 1, cuisine: 1})

Graphical user interface, text

Description automatically generated

1 thể hiện giá trị sẽ trả về, kết quả sẽ in màu xanh

3. Cho biết restaurant\_id, name, borough và zip code của tất cả document trong restaurants.

db.restaurants.find({}, {restaurant\_id: 1, name: 1, borough: 1, "address.zipcode": 1})

Text

Description automatically generated

4. Cho biết restaurant có borough = Bronx

db.restaurants.find({borough: "Bronx"})

Text

Description automatically generated

5. Cho biết restaurants có score trong khoản 80 đến 100

db.restaurants.find({ "grades.score": { $gte: 80, $lte: 100 } })

6. Cho biết restaurant có vĩ độ nhỏ hơn -60.754168 và có score lớn hơn 70

db.restaurants.find({ "address.coord": { $lt: [-60.754168] }, "grades.score": { $gt: 70 } })

7. Cho biết restaurants có điểm A.

db.restaurants.find({ "grades.grade": "A" })

Text

Description automatically generated

Bài 2:

// Tạo collection movies

db.createCollection("movies", {

validator: {

$jsonSchema: {

bsonType: "object",

required: ["title", "director\_id", "release\_year", "imdb", "actors"],

properties: {

\_id: {

bsonType: "objectId"

},

title: {

bsonType: "string"

},

director\_id: {

bsonType: "objectId"

},

release\_year: {

bsonType: "int"

},

imdb: {

bsonType: "object",

required: ["rating", "votes"],

properties: {

rating: {

bsonType: "double"

},

votes: {

bsonType: "int"

}

}

},

actors: {

bsonType: "array",

items: {

bsonType: "object",

required: ["person\_id", "as"],

properties: {

person\_id: {

bsonType: "objectId"

},

as: {

bsonType: "string"

}

}

}

}

}

}

}

});

// Tạo collection people

db.createCollection("people", {

validator: {

$jsonSchema: {

bsonType: "object",

required: ["first\_name", "last\_name", "birth\_date"],

properties: {

\_id: {

bsonType: "objectId"

},

first\_name: {

bsonType: "string"

},

last\_name: {

bsonType: "string"

},

birth\_date: {

bsonType: "date"

}

}

}

}

});

//Nhập dữ liệu clt Movies

db.movies.insertMany([

{

title: "Tom and Jerry",

director\_id: ObjectId("61f53c39e3d3b80e942b51c7"),

release\_year: 2021,

imdb: {

rating: 4.5,

votes: 100

},

actors: [

{

person\_id: ObjectId("61f53c39e3d3b80e942b51c8"),

as: "Tom"

},

{

person\_id: ObjectId("61f53c39e3d3b80e942b51c9"),

as: "Jerry"

}

]

},

{

title: "The Lion King",

director\_id: ObjectId("61f53c39e3d3b80e942b51ca"),

release\_year: 2019,

imdb: {

rating: 7.1,

votes: 500

},

actors: [

{

person\_id: ObjectId("61f53c39e3d3b80e942b51cb"),

as: "Simba"

},

{

person\_id: ObjectId("61f53c39e3d3b80e942b51cc"),

as: "Mufasa"

}

]

}

]);

//Nhập dữ liệu collection people

db.people.insertMany([

{

first\_name: "John",

last\_name: "Doe",

birth\_date: ISODate("1990-01-01T00:00:00Z")

},

{

first\_name: "Jane",

last\_name: "Doe",

birth\_date: ISODate("2000-02-01T00:00:00Z")

},

{

first\_name: "Bob",

last\_name: "Ngô",

birth\_date: ISODate("1980-03-01T00:00:00Z")

}

]);

//tạo và nhập dl clt studios

db.createCollection("studios")

db.studios.insertOne({

\_id: ObjectId("613fd3cb10779c6e3892c3c2"),

name: "Warner Bros",

year\_founded: 1923,

movies: [

ObjectId("613fd5d110779c6e3892c3c5"),

ObjectId("613fd5d110779c6e3892c3c6"),

ObjectId("613fd5d110779c6e3892c3c7")

],

headquarters: {

address: "4000 Warner Blvd",

city: "Burbank",

state: "California",

country: "USA"

}

})

db.studios.insertOne({

\_id: ObjectId("613fd3cb10779c6e3892c3c3"),

name: "Universal Studios",

year\_founded: 1912,

movies: [

ObjectId("613fd5d110779c6e3892c3c5"),

ObjectId("613fd5d110779c6e3892c3c8")

],

headquarters: {

address: "100 Universal City Plaza",

city: "Universal City",

state: "California",

country: "USA"

}

})

db.studios.insertOne({

\_id: ObjectId("613fd3cb10779c6e3892c3c4"),

name: "Walt Disney Pictures",

year\_founded: 1923,

movies: [

ObjectId("613fd5d110779c6e3892c3c9"),

ObjectId("613fd5d110779c6e3892c3ca")

],

headquarters: {

address: "500 S Buena Vista St",

city: "Burbank",

state: "California",

country: "USA"

}

})

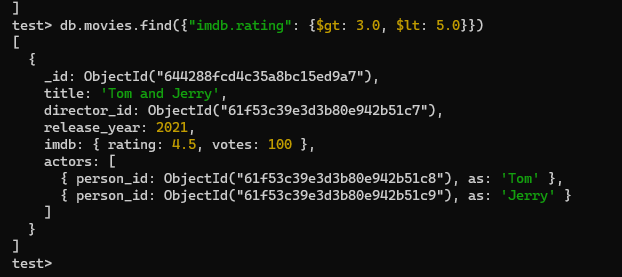
Bắt đầu thực hiện truy vấn theo yêu cầu

1/ 1. Cho biết các movies có title là “Tom and Jerry”

db.movies.find({title: "Tom and Jerry"})

2. Cho biết các movies có rating lớn hơn 3.0 và nhỏ hơn 5.0

db.movies.find({"imdb.rating": {$gt: 3.0, $lt: 5.0}})



3. Cho biết các movies có diễn viên họ “Ngô”

db.movies.find({ actors: { $elemMatch: { last\_name: "Ngô" } } })

4. Cho biết các movies được sản xuất ở “Mỹ” và có hơn 10 người vote

db.movies.find({

"imdb.votes": { $gt: 10 },

"studios": {

$elemMatch: {

"headquarters.country": "usa"

}

}

})

5. Cho biết các movies có diễn viên sinh năm 2000 hoặc có năm phát hành trước 2020

db.movies.find({

$or: [

{ "actors.birth\_date": { $gte: ISODate("2000-01-01T00:00:00Z"), $lt: ISODate("2001-01-01T00:00:00Z") } },

{ release\_year: { $lt: 2020 } }

]

})

Các file json đi kèm được trong thư mục bài nộp