

Database Systems

Lab 12

Shazain

Lab 12A

8)

```
mongosh mongodb+srv://<cc> x + v
C:\Users\Shazain>mongosh "mongodb+srv://mango.rhrqa.mongodb.net/" --apiVersion 1 --username yaszq
Enter password: ****
Current Mongosh Log ID: 673ec42944df1060ac0d818f
Connecting to:
  mongodb+srv://<credentials>@mango.rhrqa.mongodb.net/?appName=mongosh+2.3.3
Using MongoDB:
  8.0.3 (API Version 1)
Using Mongosh:
  2.3.3

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

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

Atlas atlas-gcwaib-shard-0 [primary] test>
Atlas atlas-gcwaib-shard-0 [primary] test> use im_in
...
Atlas atlas-gcwaib-shard-0 [primary] test>
Atlas atlas-gcwaib-shard-0 [primary] test> show dbs
sample_mflix 115.15 MiB
admin 328.00 KiB
local 33.43 GiB
Atlas atlas-gcwaib-shard-0 [primary] test> use iba_db
switched to db iba_db
Atlas atlas-gcwaib-shard-0 [primary] iba_db> db.createCollection("test")
{ ok: 1 }
Atlas atlas-gcwaib-shard-0 [primary] iba_db> db.test.insertOne({"name": "testname", "marks": 80})
{ acknowledged: true,
  insertedId: ObjectId('673ec4a144df1060ac0d8190') }
Atlas atlas-gcwaib-shard-0 [primary] iba_db> db.test.find()
[
  {
    _id: ObjectId('673ec4a144df1060ac0d8190'),
    name: 'testname',
    marks: 80
  }
]
Atlas atlas-gcwaib-shard-0 [primary] iba_db> db.createCollection("restaurants")
{ ok: 1 }
Atlas atlas-gcwaib-shard-0 [primary] iba_db> show dbs
iba_db 48.00 KiB
sample_mflix 115.15 MiB
admin 328.00 KiB
local 33.43 GiB
Atlas atlas-gcwaib-shard-0 [primary] iba_db>
```

Lab 12B

1) 1

The screenshot shows the MongoDB Compass application interface. The top bar indicates the connection is 'mongo.rhrqa.mongodb.net/Shell'. The left sidebar contains a 'Connections' panel with a search bar and a list of databases: 'admin', 'config', 'iba_db' (expanded to show 'restaurants' and 'test'), 'local', and 'sample_mflix'. The main panel is divided into two tabs: 'restaurants' and 'mongo.rhrqa...'. The 'restaurants' tab is active, showing a list of database sizes: 'sample_mflix' (115.16 MiB), 'admin' (328.00 KiB), and 'local' (33.43 GiB). Below this, a query is executed: '> db.restaurants.find()'. The result is displayed in a JSON format, showing a single document with the following fields: '_id' (ObjectId), 'address' (building: '1007', coord: [-73.856077, 40.848447], street: 'Morris Park Ave', zipcode: '10462'), 'borough' ('Bronx'), 'cuisine' ('Bakery'), and 'grades' (an array of three objects with date, grade, and score).

```
>_MONGOSH
sample_mflix 115.16 MiB
admin        328.00 KiB
local        33.43 GiB
> db.restaurants.find()
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  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
    },
    {
      date: 2011-11-23T00:00:00.000Z,
      grade: 'A',
      score: 9
    }
  ]
}
```

2) 2

The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' panel with a tree view of the database structure. The main panel shows the 'restaurants' collection in the 'mongo' database. A query is executed, and the results are displayed in a JSON format.

Connections:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

```
db.restaurants.find({}, {"restaurant_id": 1, "name": 1, "borough": 1, "cuisine": 1})
```

Results:

```
{
  "_id": ObjectId('673ec62f3a635a3395746f3c'),
  "borough": 'Bronx',
  "cuisine": 'Bakery',
  "name": 'Morris Park Bake Shop',
  "restaurant_id": '30075445'
},
{
  "_id": ObjectId('673ec62f3a635a3395746f3d'),
  "borough": 'Brooklyn',
  "cuisine": 'Hamburgers',
  "name": 'Wendy'S',
  "restaurant_id": '30112340'
},
{
  "_id": ObjectId('673ec62f3a635a3395746f3e'),
  "borough": 'Manhattan',
  "cuisine": 'Irish',
  "name": 'Dj Reynolds Pub And Restaurant',
  "restaurant_id": '30191841'
},
{
  "_id": ObjectId('673ec62f3a635a3395746f3f'),
  "borough": 'Brooklyn',
  "cuisine": 'American ',
  "name": 'Riviera Caterer',
  "restaurant_id": '40356018'
},
{
  "_id": ObjectId('673ec62f3a635a3395746f40'),
  "borough": 'Queens',
  "cuisine": 'Jewish/Kosher',
  "name": 'Tov Kosher Kitchen',
  "restaurant_id": '40356068'
},
{
  "_id": ObjectId('673ec62f3a635a3395746f41'),
  "borough": 'Queens',
  "cuisine": 'Jewish/Kosher',
  "name": 'Tov Kosher Kitchen',
  "restaurant_id": '40356068'
}
```

3) 3

The screenshot shows the MongoDB Compass application interface. The top bar indicates the connection is 'mongo.rhrqa.mongodb.net/Shell'. The left sidebar contains a 'Connections' panel with a search bar and a tree view of the database structure. The main panel is divided into two tabs: 'restaurants' and 'mongosh: mongo.rhrqa...'. The 'mongosh' tab is active, displaying a MongoDB shell prompt and a query. The query is `db.restaurants.find({}, {"restaurant_id": 1, "name": 1, "borough": 1, "cuisine": 1, "_id": 0})`. The results are displayed as a JSON array of six restaurant documents.

Connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query Results

```
> db.restaurants.find({}, {"restaurant_id": 1, "name": 1, "borough": 1, "cuisine": 1, "_id": 0})
< [
  {
    borough: 'Bronx',
    cuisine: 'Bakery',
    name: 'Morris Park Bake Shop',
    restaurant_id: '30075445'
  },
  {
    borough: 'Brooklyn',
    cuisine: 'Hamburgers',
    name: 'Wendy'S',
    restaurant_id: '30112340'
  },
  {
    borough: 'Manhattan',
    cuisine: 'Irish',
    name: 'Dj Reynolds Pub And Restaurant',
    restaurant_id: '30191841'
  },
  {
    borough: 'Brooklyn',
    cuisine: 'American ',
    name: 'Riviera Caterer',
    restaurant_id: '40356018'
  },
  {
    borough: 'Queens',
    cuisine: 'Jewish/Kosher',
    name: 'Tov Kosher Kitchen',
    restaurant_id: '40356068'
  },
  {
    borough: 'Queens',
    cuisine: 'American ',
    name: 'Brunos On The Boulevard',
    restaurant_id: '40356151'
  }
]
```

4) 4

The screenshot shows the MongoDB Compass interface. On the left, the 'Connections' panel lists the 'mango.rhrqa.mongodb.net' connection, with the 'restaurants' collection selected under the 'iba_db' database. The main panel displays the MongoDB Shell with the following command and result:

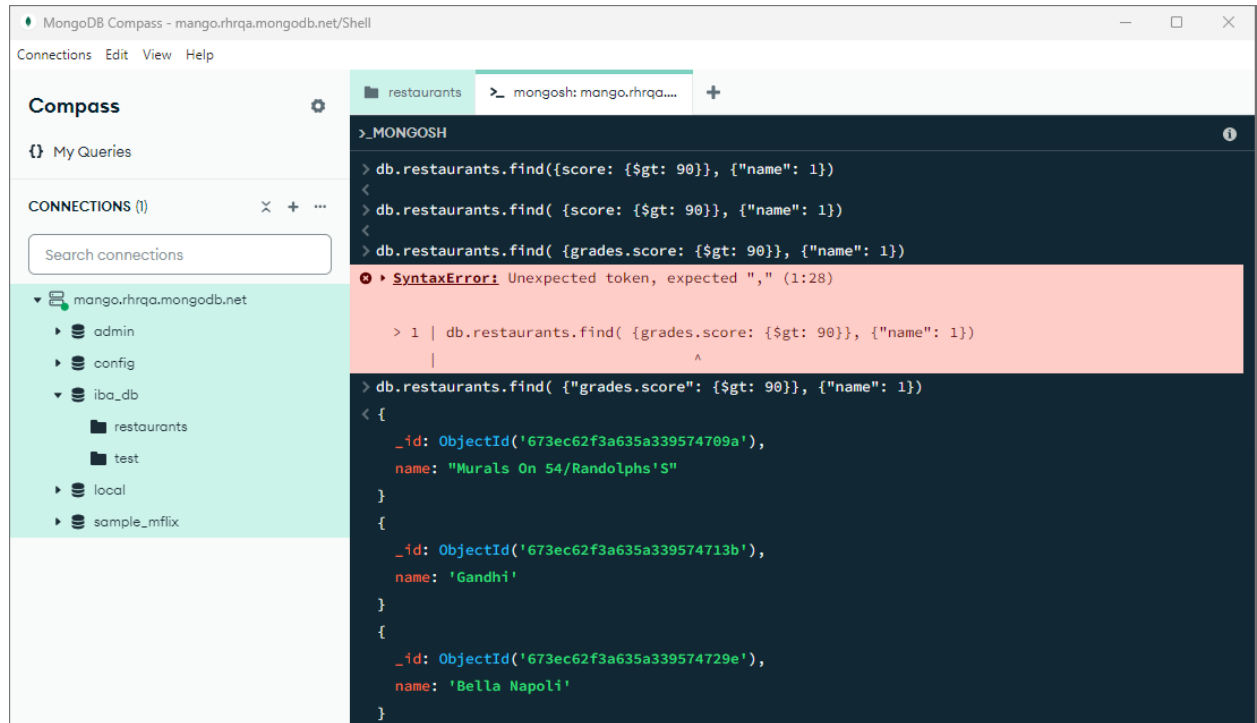
```
> db.restaurants.find({"borough": "Bronx"})
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  address: {
    building: '1007',
    coord: [
      -73.856077,
      40.848447
    ],
    street: 'Morris Park Ave',
    zipcode: '10462'
  },
  borough: 'Bronx',
  cuisine: 'Bakery',
  grades: [
    {
      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
    },
    {
      date: 2011-11-23T00:00:00.000Z,
      grade: 'A',
      score: 9
    },
    {
      date: 2011-03-10T00:00:00.000Z,
```

5) 5

The screenshot shows the MongoDB Compass interface. On the left, the 'Connections' panel lists the 'mango.rhrqa.mongodb.net' connection, with the 'restaurants' collection selected under the 'iba_db' database. The main panel displays the MongoDB Shell with the following command and result:

```
> db.restaurants.countDocuments({"borough": "Bronx"})
< 309
Atlas atlas-gcwa10-shard-0 [primary] iba_db>
```

6) 6



7) 7

MongoDB Compass - mango.rhrqa.mongodb.net/Shell

Connections Edit View Help

Compass

{ } My Queries

CONNECTIONS (1)

Search connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

restaurants mongosh: mango.rhrqa...

>_MONGOSH

```
> db.restaurants.find(
  { "grades": { $elemMatch: { score: { $gt: 80, $lt: 100 } } } },
  { "name" : 1, "grades": 1 }
)
< [
  {
    _id: ObjectId('673ec62f3a635a339574713b'),
    grades: [
      {
        date: 2014-09-15T00:00:00.000Z,
        grade: 'A',
        score: 5
      },
      {
        date: 2014-01-14T00:00:00.000Z,
        grade: 'A',
        score: 8
      },
      {
        date: 2013-05-30T00:00:00.000Z,
        grade: 'A',
        score: 12
      },
      {
        date: 2013-04-24T00:00:00.000Z,
        grade: 'P',
        score: 2
      },
      {
        date: 2012-10-01T00:00:00.000Z,
        grade: 'A',
        score: 9
      },
      {
        date: 2012-04-06T00:00:00.000Z,
        grade: 'C',
        score: 92
      }
    ]
  }
]
```

8) 8

The screenshot shows the MongoDB Compass application interface. The left sidebar contains a 'Connections' panel with a search bar and a list of connections. The 'restaurants' collection is selected in the top navigation bar. The main panel displays a MongoDB query and its results.

Connections:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

```
> db.restaurants.find(
  {"grades.score": {$lt: 5}},
  {"name": 1, "grades.score": 1}
)
```

Results:

```
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  grades: [
    { score: 2 },
    { score: 6 },
    { score: 10 },
    { score: 9 },
    { score: 14 }
  ],
  name: 'Morris Park Bake Shop'
}
{
  _id: ObjectId('673ec62f3a635a3395746f3e'),
  grades: [
    { score: 2 },
    { score: 11 },
    { score: 12 }
  ],
  name: 'Morris Park Bake Shop'
}
```


9) 9

The screenshot shows the MongoDB Compass web interface. The left sidebar contains a 'Connections' panel with a search bar and a tree view of databases and collections. The 'restaurants' collection is selected. The main panel shows a MongoDB query in the 'MONGOSH' tab:

```
>_MONGOSH
>
> db.restaurants.find( {
  "grades": { $all: [
    {$elemMatch: {score: 10}},
    {$elemMatch: {grade: "A"}}
  ]
}
})
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  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
    },
  ],
}
```

10) 10

The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' panel with a tree view of the database structure. The 'restaurants' collection is selected. The main panel shows a query in the 'MongoShell' tab. The query is a find operation on the 'restaurants' collection, filtering for documents where the 'cuisine' is 'American', the 'grades' score is greater than or equal to 70, and the 'address.coord.0' is less than -65.75. The result is a single document with the following fields: '_id', 'address' (building, coord), 'borough', 'cuisine', 'grades' (array of objects with date, grade, score), 'street', and 'zipcode'.

```
> db.restaurants.find({
  "cuisine": {$ne: "American "},
  "grades": {$elemMatch: {score: {$gt: 70}}},
  "address.coord.0": {$lt: -65.75}}
)
< {
  _id: ObjectId('673ec62f3a635a339574713b'),
  address: {
    building: '345',
    coord: [
      -73.9864626,
      40.7266739
    ],
    street: 'East 6 Street',
    zipcode: '10003'
  },
  borough: 'Manhattan',
  cuisine: 'Indian',
  grades: [
    {
      date: 2014-09-15T00:00:00.000Z,
      grade: 'A',
      score: 5
    },
    {
      date: 2014-01-14T00:00:00.000Z,
      grade: 'A',
      score: 8
    },
    {
      date: 2013-05-30T00:00:00.000Z,
      grade: 'A',
      score: 12
    },
    {
      date: 2013-04-24T00:00:00.000Z,
      grade: 'P',
      score: 2
    }
  ]
}
```

11) 11

The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' panel with a tree view of the database structure. The 'restaurants' collection is selected under the 'iba_db' database. The main panel shows the MongoDB Shell with a query that filters restaurants by cuisine, grade, and borough, and sorts them by cuisine. The query results are displayed in a table format.

Connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

```
> db.restaurants.find({
  "cuisine": {$ne: "American "},
  "grades": {$elemMatch: {grade: "A"}},
  "borough": {$ne: "Brooklyn"}},
  {"name": 1, "borough": 1,
  "grades": 1, "cuisine": 1}
).sort({"cuisine": -1});
```

Results:

_id	borough	cuisine	grades	date	grade	score
ObjectId('673ec6303a635a3395747701')	Queens	Vietnamese/Cambodian/Malaysia	[{"date": 2014-06-12T00:00:00.000Z, "grade": 'B', "score": 21}]	2014-06-12T00:00:00.000Z	B	21
			[{"date": 2013-05-20T00:00:00.000Z, "grade": 'A', "score": 13}]	2013-05-20T00:00:00.000Z	A	13
			[{"date": 2012-12-26T00:00:00.000Z, "grade": 'A', "score": 10}]	2012-12-26T00:00:00.000Z	A	10
			[{"date": 2012-12-03T00:00:00.000Z, "grade": 'P', "score": 5}]	2012-12-03T00:00:00.000Z	P	5
			[{"date": 2012-05-04T00:00:00.000Z, "grade": 'B', "score": 27}]	2012-05-04T00:00:00.000Z	B	27

12) 12

The screenshot shows the MongoDB Compass web interface. The left sidebar contains a 'Connections' panel with a search bar and a list of connections. The 'mongo.rhrqa.mongodb.net' connection is expanded, showing databases 'admin', 'config', 'iba_db', and 'test'. The 'restaurants' collection is selected under the 'iba_db' database. The main panel displays a MongoDB query and its results.

Connections:

- mongo.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

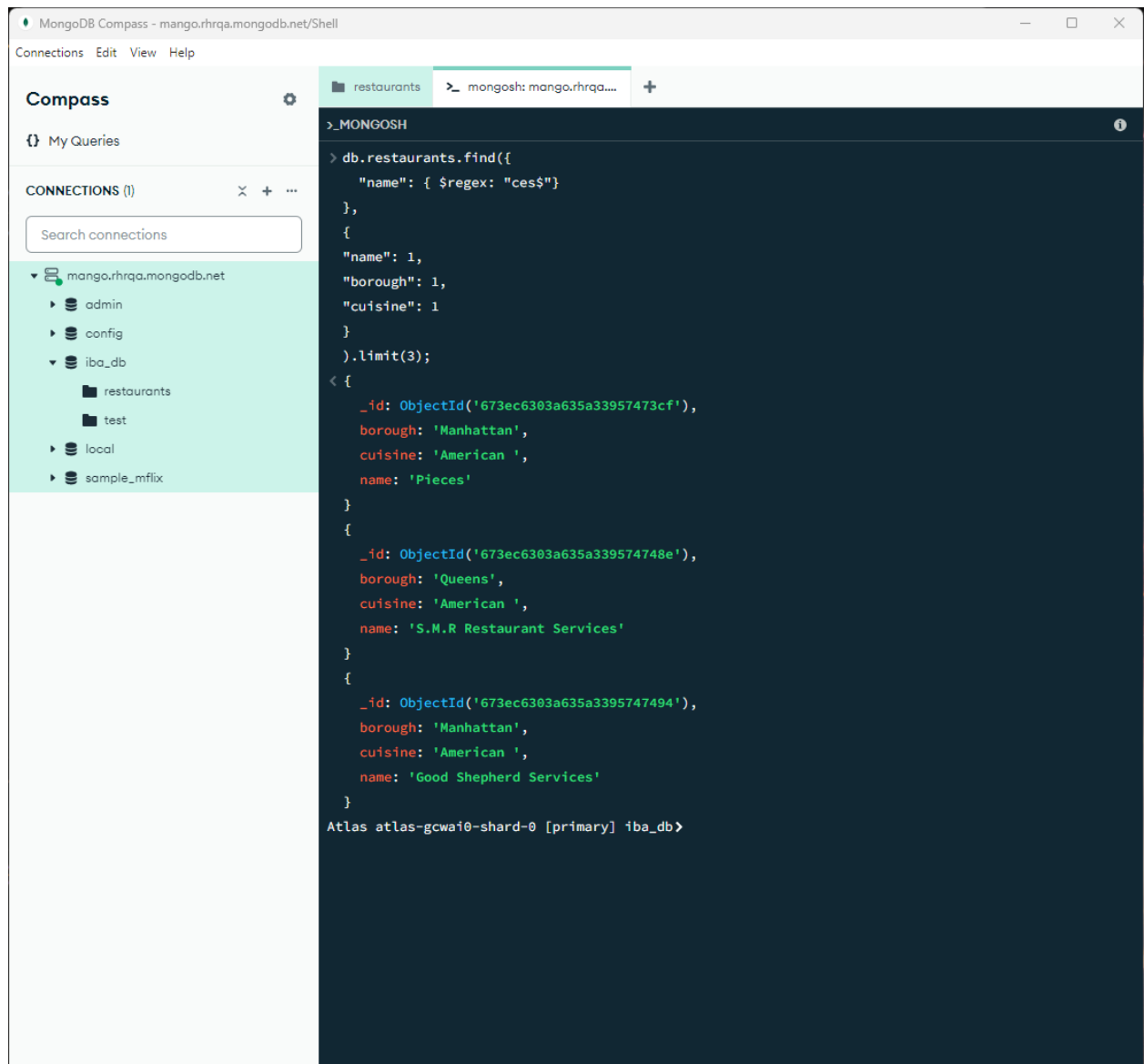
```
> db.restaurants.find({
  "name": { $regex: /^wil/i}
},
{
  "restaurant_id": 1,
  "name": 1,
  "borough": 1,
  "cuisine": 1
}).sort({"cuisine": -1});
```

Results:

```
< {
  _id: ObjectId('673ec6313a635a3395747d4b'),
  borough: 'Bronx',
  cuisine: 'Pizza',
  name: 'Wilbel Pizza',
  restaurant_id: '40871979'
}
{
  _id: ObjectId('673ec62f3a635a3395746f43'),
  borough: 'Brooklyn',
  cuisine: 'Delicatessen',
  name: 'Wilken'S Fine Food',
  restaurant_id: '40356483'
}
{
  _id: ObjectId('673ec62f3a635a3395746f46'),
  borough: 'Bronx',
  cuisine: 'American ',
  name: 'Wild Asia',
  restaurant_id: '40357217'
}
```

Atlas atlas-gcwai0-shard-0 [primary] iba_db>

13) 13



**** PLEASE IGNORE THE .limit AT
THE END AFTER THIS QUERY****

14) 14

The screenshot shows the MongoDB Compass interface. On the left, the 'Connections' panel lists several connections, with 'mango.rhrqa.mongodb.net' selected. Under this connection, the 'restaurants' database is expanded, showing a 'restaurants' collection. The main panel on the right displays a MongoDB query in the 'MONGOSH' shell:

```
> db.restaurants.find(
  {
    "name": {$regex: /reg/i}
  },
  {
    "restaurant_id": 1,
    "name": 1,
    "borough": 1,
    "cuisine": 1
  }
)
< {
  _id: ObjectId('673ec62f3a635a3395746f44'),
  borough: 'Brooklyn',
  cuisine: 'American ',
  name: 'Regina Caterers',
  restaurant_id: '40356649'
}
{
  _id: ObjectId('673ec62f3a635a3395747041'),
  borough: 'Manhattan',
  cuisine: 'Café/Coffee/Tea',
  name: 'Caffe Reggio',
  restaurant_id: '40369418'
}
{
  _id: ObjectId('673ec62f3a635a3395747150'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: 'Regency Hotel',
  restaurant_id: '40382679'
}
{
  _id: ObjectId('673ec6303a635a339574746d'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: 'Regency Whist Club',
  restaurant_id: '40400000'
```

15) 15

The screenshot shows the MongoDB Compass web interface. The left sidebar contains a 'Connections' panel with a search bar and a tree view of the database structure. The main area is split into two panes: a 'Compass' pane on the left and a 'mongo' shell pane on the right. The 'mongo' shell pane shows a query being executed in the 'restaurants' collection of the 'iba_db' database. The query filters for restaurants in the 'Bronx' borough with either 'American' or 'Chinese' cuisine, sorted by 'restaurant_id' in descending order, and limited to 3 results. The output shows three documents with their respective IDs, boroughs, cuisines, and names.

Connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Compass

My Queries

mongo

```
> db.restaurants.find({
  $or: [
    {"cuisine": "American "},
    {"cuisine": "Chinese"}
  ],
  "borough": "Bronx"
},
{
  "restaurant_id": 1,
  "name": 1,
  "borough": 1,
  "cuisine": 1
}).limit(3);
< {
  _id: ObjectId('673ec62f3a635a3395746f46'),
  borough: 'Bronx',
  cuisine: 'American ',
  name: 'Wild Asia',
  restaurant_id: '40357217'
}
{
  _id: ObjectId('673ec62f3a635a3395746f5f'),
  borough: 'Bronx',
  cuisine: 'Chinese',
  name: 'Happy Garden',
  restaurant_id: '40363289'
}
{
  _id: ObjectId('673ec62f3a635a3395746f71'),
  borough: 'Bronx',
  cuisine: 'Chinese',
  name: 'Happy Garden',
  restaurant_id: '40364296'
}
Atlas atlas-gcwa10-shard-0 [primary] iba_db>
```

16) 16

The screenshot shows the MongoDB Compass application interface. The top bar indicates the connection is 'mongo.rhrqa.mongodb.net/Shell'. The left sidebar contains a 'Connections' panel with a search bar and a tree view of the database structure. The tree shows a database named 'iba_db' with two collections: 'restaurants' and 'test'. The main panel on the right is the 'mongoosh' shell, where a query is being executed. The query filters for restaurants in 'Staten Island', 'Queens', 'Bronx', or 'Brooklyn', with a limit of 3 results. The output shows three documents, each with an ObjectId, borough, cuisine, and name.

Connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

```
> db.restaurants.find({
  "borough": {
    $in: ["Staten Island", "Queens", "Bronx", "Brooklyn"]
  },
  {
    "name": 1,
    "borough": 1,
    "cuisine": 1
  }
}).limit(3);
```

Result:

```
< [ {
  _id: ObjectId('673ec62f3a635a3395746f3e'),
  borough: 'Manhattan',
  cuisine: 'Irish',
  name: 'Dj Reynolds Pub And Restaurant'
}, {
  _id: ObjectId('673ec62f3a635a3395746f49'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: '1 East 66Th Street Kitchen'
}, {
  _id: ObjectId('673ec62f3a635a3395746f4e'),
  borough: 'Manhattan',
  cuisine: 'American ',
  name: 'Glorious Food'
} ]
```

Atlas atlas-gcwa10-shard-0 [primary] iba_db>

17) 17

The screenshot shows the MongoDB Compass application interface. The left sidebar contains the 'Compass' section with 'My Queries' and 'CONNECTIONS (1)'.

The 'CONNECTIONS (1)' section shows a search bar and a list of connections:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

The main panel displays the MongoDB shell with the following query and results:

```
> db.restaurants.find({
  $or: [
    {"cuisine": {
      $nin: ["American ", "Chinese"]}},
    {"name": {$regex: "^Wil"}}
  ]},
  {
    "name": 1,
    "borough": 1,
    "address.building": 1
  }
).limit(3);
```

The results are displayed as an array of three objects:

```
< [
  {
    _id: ObjectId('673ec62f3a635a3395746f3c'),
    address: {
      building: '1007'
    },
    borough: 'Bronx',
    name: 'Morris Park Bake Shop'
  },
  {
    _id: ObjectId('673ec62f3a635a3395746f3d'),
    address: {
      building: '469'
    },
    borough: 'Brooklyn',
    name: 'Wendy'S'
  },
  {
    _id: ObjectId('673ec62f3a635a3395746f3e'),
    address: {
      building: '351'
    },
    borough: 'Manhattan',
    name: 'Dj Reynolds Pub And Restaurant'
  }
]
```

At the bottom of the shell, the text 'Atlas atlas-gcwai@shard=0 [primary] iba_db' is visible.

18) 18

The screenshot shows the MongoDB Compass application interface. The top bar indicates the connection to 'mongo.rhrqa.mongodb.net/Shell'. The left sidebar contains a 'Connections' panel with a search bar and a tree view of the database structure. The main area is split into two panes: 'My Queries' and 'MONGOSH'. The 'MONGOSH' pane displays a JavaScript query that finds documents in the 'restaurants' collection based on specific criteria.

Connections Panel:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

MONGOSH Query:

```
> db.restaurants.find(
  {
    "grades": {
      $elemMatch: {
        grade: "A",
        score: 11,
        date: new Date("2014-08-11")
      }
    },
    {
      "name": 1,
      "grades": 1
    }
  }
)
< {
  _id: ObjectId('673ec62f3a635a3395747095'),
  grades: [
    {
      date: 2014-08-11T00:00:00.000Z,
      grade: 'A',
      score: 11
    },
    {
      date: 2013-12-10T00:00:00.000Z,
      grade: 'A',
      score: 9
    },
    {
      date: 2013-06-10T00:00:00.000Z,
      grade: 'A',
      score: 12
    },
    {
      date: 2012-06-08T00:00:00.000Z,
      grade: 'A',
      score: 13
    }
  ]
}
```

19) 19

The screenshot shows the MongoDB Compass application interface. The top bar indicates the connection is 'mongo.rhrqa.mongodb.net/Shell'. The left sidebar contains a 'Connections' panel with a search bar and a tree view of databases and collections. The 'iba_db' database is expanded, showing the 'restaurants' collection. The main panel displays the 'mongosh' shell with a query that filters restaurants by coordinates and lists their details.

Connections Panel:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Mongosh Shell:

```
> db.restaurants.find(
  {
    "address.coord.1": {$gt: 42, $lt: 52}
  },
  {
    "name": 1,
    "address": 1
  }
)
< {
  _id: ObjectId('673ec62f3a635a33957471de'),
  address: {
    building: '47',
    coord: [
      -78.877224,
      42.89546199999999
    ],
    street: 'Broadway @ Trinity Pl',
    zipcode: '10006'
  },
  name: "T.G.I. Friday'S"
}
{
  _id: ObjectId('673ec62f3a635a339574720a'),
  address: {
    building: '1',
    coord: [
      -0.7119979,
      51.6514664
    ],
    street: 'Pennplaza E, Penn Sta',
    zipcode: '10001'
  },
  name: 'T.G.I. Fridays'
}
{
  _id: ObjectId('673ec6303a635a3395747463'),
```

20) 20

The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' panel with a tree view of the database structure. The 'restaurants' collection is selected. The main panel shows the MongoDB Shell with a query that filters documents by the 'name' field and sorts them by 'name'.

Connections:

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query:

```
> db.restaurants.find({}, {"name": 1}).sort({"name": 1})
```

Results:

```
< {
  _id: ObjectId('673ec6313a635a3395747bcc'),
  name: '(Lewis Drug Store) Locanda Vini E Olii'
}
{
  _id: ObjectId('673ec62f3a635a3395746f49'),
  name: '1 East 66Th Street Kitchen'
}
{
  _id: ObjectId('673ec6303a635a339574775b'),
  name: '101 Deli'
}
{
  _id: ObjectId('673ec6303a635a339574764f'),
  name: '101 Restaurant And Bar'
}
{
  _id: ObjectId('673ec6303a635a339574741c'),
  name: '1020 Bar'
}
{
  _id: ObjectId('673ec6303a635a3395747856'),
  name: '104-01 Foster Avenue Coffee Shop(Ups)'
}
{
  _id: ObjectId('673ec6303a635a3395747902'),
  name: '10Th Avenue Pizza & Cafe'
}
{
  _id: ObjectId('673ec6313a635a3395747da4'),
  name: '111 Restaurant'
}
{
  _id: ObjectId('673ec6303a635a3395747974'),
  name: '15 East Restaurant'
}
```

21)21

The screenshot shows the MongoDB Compass web interface. The left sidebar contains a 'Connections' panel with a search bar and a list of connections. The 'mongo.rhrqa.mongodb.net' connection is expanded, showing databases 'admin', 'config', 'iba_db', 'local', and 'sample_mflix'. The 'iba_db' database is selected, and the 'restaurants' collection is visible. The main panel shows a query in the 'restaurants' collection. The query is a find operation with a filter on 'grades.score' and a projection on 'name'. The results are displayed as a list of documents.

Connections

- mango.rhrqa.mongodb.net
 - admin
 - config
 - iba_db
 - restaurants
 - test
 - local
 - sample_mflix

Query

```
> db.restaurants.find(
  {
    "grades.score": {
      $all:
        [2, 6]
    }
  },
  {
    "name": 1
  }
)
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  name: 'Morris Park Bake Shop'
}
{
  _id: ObjectId('673ec62f3a635a3395746f7c'),
  name: 'Polish National Home'
}
{
  _id: ObjectId('673ec62f3a635a3395746f92'),
  name: 'White Horse Tavern'
}
{
  _id: ObjectId('673ec62f3a635a3395746f9e'),
  name: 'Keats Restaurant'
}
{
  _id: ObjectId('673ec62f3a635a3395746fe6'),
  name: 'Little Brown Jug'
}
{
  _id: ObjectId('673ec62f3a635a339574700f'),
  name: 'Fiorellos'
}
{
  _id: ObjectId('673ec62f3a635a33957470a9'),
```

22) 22

