

Lecture 6: MongoDB Manipulating Data (Intermediate Query)

Big Data Systems Design

Part 4

PRACTICE

Import dataset

❖ Restaurant dataset

```
{
  "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"
}
```

Intermediate Queries

❖ Query embedded document

- To specify a query condition on fields, use **dot notation**
 - (*"field.nestedField"*)

❖ Query 1

- Find the restaurant Id, name, borough and cuisine for those restaurants which achieved grade A
 - **db.restaurants.find({"grades.grade": "A"},
{_id:0, restaurant_id:1, name:1, borough:1, cuisine:1})**

See the result

❖ Task 1

- Find the restaurants that do not prepare any cuisine of 'American' and their grade score more than 70

Intermediate Queries

❖ Query an array

- To specify equality condition on an array, use the query document
 - `{ <field>: <value> }`
 - <value> is the EXACT array to match, including the order of the elements

❖ Query 2

- All restaurants that are located in latitude -73.856077 and longitude 40.848447
 - **`db.restaurants.find({"address.coord": [-73.856077, 40.848447]})`**

See the result

❖ Task 2

- Find the restaurants which prepare American or Chinese dish and achieved a score more than 60 and located in the latitude less than -74.

Intermediate Queries

❖ Query an array of documents

- Using dot notation, you can specify query conditions for field in a document at a particular index or position of the array

❖ Query 3

- Find the restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00.000+00:00"
 - **`db.restaurants.find({$and: [`**
 `{"grades.1.grade":"A"},`
 `{"grades.1.score": 9},`
 `{"grades.1.date": ISODate("2014-08-11T00:00:00.000+00:00")}])`

See the result

❖ Task 3

- Find the restaurant Id, name, and scores for those restaurants where the eighth element of grades contains a score greater than 30

Intermediate Queries

❖ Regular expression

- Evaluation operators: \$regex
 - To use \$regex, use one of the following syntaxes
 - { <field>: { \$regex: 'pattern', \$options: '<options>' } }

❖ Query 4

- Find the restaurants that have Pizza on its name
 - **db.restaurants.find({name : {\$regex : "Pizza"}})**

See the result

❖ Task 4

- Find the restaurants that are located either in Staten Island or Queens and contain 'Wen' as first three letters for its name

Intermediate Queries

❖ The query matches documents that do not contain the item field

- *$\{ item : \{ \$exists: false \} \}$*

- Task 5

- Find whether all the addresses contains the street or not

❖ \$type

- *$\{ field: \{ \$type: <BSON type> \} \}$*

- Task 6

- Check if all address coordinate values are in an array format

Final Task

- ❖ Download one of the datasets provided by MongoDB
 - <https://github.com/neelabalan/mongodb-sample-dataset>
 - <https://github.com/ozlerhakan/mongodb-json-files>
 - Optional: You can also use your own dataset
- ❖ Create three queries studied during the Lecture 6
 - Use dot notation
 - Use \$regex operator
 - Use \$exist operator
- ❖ Demo
 - Your dataset, your queries and their description, and result screen

Questions?

SEE YOU NEXT TIME!