

DBMS Lab File

(2024)

|  |  |
| --- | --- |
| Made by – | |
| Name  Course  Sap Id | Satvik Raj  B. tech CSE  500119624 |

# EXPERIMENT-20: To understand the concepts of NoSQL Database

1. Write a MongoDB query to arrange the name of the cuisine in ascending order and for that same cuisine Borough should be in descending order.

db.hotels.find({}, {cuisine: 1, borough: 1})

.sort({cuisine: 1, borough: -1});

2. Write a MongoDB query to know whether all the addresses contain the street or not.

db.hotels.find({"address.street": {$exists: true}});

3. Write a MongoDB query which will select all documents in the hotels collection where the coord field value is Double.

db.hotels.find({"address.coord": {$type: "double"}});

4. Write a MongoDB query which will select the hotel Id, name, and grades for those hotels which return 0 as a remainder after dividing the score by 7.

db.hotels.find({"grades.score": {$mod: [7, 0]}}, {hotel\_id: 1, name: 1, grades: 1});

5. Write a MongoDB query to find the hotel name, Borough, longitude, latitude, and cuisine for those hotels which contain 'mon' as three letters somewhere in their name.

db.hotels.find(

{name: {$regex: "mon", $options: "i"}},

{name: 1, borough: 1, "address.coord": 1, cuisine: 1}

);

6. Write a MongoDB query to find the hotel name, Borough, longitude, latitude, and cuisine for those hotels which contain 'Mad' as the first three letters of their name.

db.hotels.find(

{name: {$regex: "^Mad", $options: "i"}},

{name: 1, borough: 1, "address.coord": 1, cuisine: 1}

);

# EXPERIMENT-21: To understand the concepts of NoSQL Database

1. Write a MongoDB query to find the hotels which do not prepare any cuisine of 'American', achieved a score more than 70, and are located in the longitude less than -65.754168.

db.hotels.find({

cuisine: {$ne: "American"},

"grades.score": {$gt: 70},

"address.coord.0": {$lt: -65.754168}

});

2. Write a MongoDB query to find the hotels which do not prepare any cuisine of 'American', achieved a grade point 'A', and do not belong to the Borough Brooklyn. Display the document according to cuisine in descending order.

db.hotels.find({

cuisine: {$ne: "American"},

"grades.grade": "A",

borough: {$ne: "Brooklyn"}

}).sort({cuisine: -1});

3. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which contain 'ces' as the last three letters of their name.

db.hotels.find(

{name: {$regex: "ces$", $options: "i"}},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

4. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which contain 'Reg' as three letters somewhere in their name.

db.hotels.find(

{name: {$regex: "Reg", $options: "i"}},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

5. Write a MongoDB query to find the hotels which belong to the Borough Bronx and prepare either American or Chinese cuisine.

db.hotels.find({

borough: "Bronx",

cuisine: {$in: ["American", "Chinese"]}

});

6. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which belong to the Borough Staten Island, Queens, or Hyatt.

db.hotels.find(

{borough: {$in: ["Staten Island", "Queens", "Hyatt"]}},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

7. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which do not belong to the Borough New Delhi, Queens, or Hyatt.

db.hotels.find(

{borough: {$nin: ["New Delhi", "Queens", "Hyatt"]}},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

8. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which achieved a score not more than 10.

db.hotels.find(

{"grades.score": {$lte: 10}},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

9. Write a MongoDB query to find the hotel Id, name, Borough, and cuisine for those hotels which prepare dishes other than 'American' and 'Chinese' or whose name begins with the letter 'Wil'.

db.hotels.find(

{

$or: [

{cuisine: {$nin: ["American", "Chinese"]}},

{name: {$regex: "^Wil", $options: "i"}}

]

},

{hotel\_id: 1, name: 1, borough: 1, cuisine: 1}

);

10. Write a MongoDB query to find the hotel Id, name, and grades for those hotels which achieved a grade of "A" and scored 11.

db.hotels.find(

{"grades.grade": "A", "grades.score": 11},

{hotel\_id: 1, name: 1, grades: 1}

);