Sreeti Ravi

Assignment 2

**Part 1**

Q1: db . test . find ( {age : {$lt : 24 , $gt : 17 }} );

This query finds documents in the collection test where age is less than 24 and greater than 17.

SQL: SELECT \* FROM test WHERE age < 24 AND age > 17

Q2: db.restaurants.find(

                           {

                             "cuisine" : {$ne : "American "},

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

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

                            }

                     );

This query finds documents in the collection restaurants that aren’t American cuisine with a grade score above 70 and coordinates less than -65.754168.

SQL: SELECT \* FROM restaurants WHERE cuisine = ‘American’ AND grades.score > 70 AND address.coord < -65.754168

Q3: db.restaurants.find(

                     {

                     "borough": "Bronx" ,

                       $or : [

                            { "cuisine" : "American " },

                           { "cuisine" : "Chinese" }

                       ]

                   }

               );

This query finds documents in the collection restaurants that are in the Bronx and either American or Chinese cuisine.

SQL: SELECT \* FROM restaurants WHERE borough = “Bronx” AND (cuisine = “American” OR cuisine = “Chinese”)

Q4:  db.movies.find(

           {

             "genre": {$ne: ""}

              },

             {\_id:0,title:1, genre:1}

            ).limit(4);

This query finds 4 documents in the collection movies where genre is not null and returns the fields title and genre and excludes the \_id field.

SQL: SELECT title, genre FROM movies LIMIT 4

Q5: db.restaurants.find(

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

          {

             "restaurant\_id" : 1,

             "name":1,"borough":1,

              "cuisine" :1

           }

         );

This query finds restaurants in the boroughs of Staten Island, Queens, Bronx and Brooklyn and prints only the restaurant ID, name, borough and cuisine.

SQL: SELECT restaurant\_id, name, borough, cuisine FROM restaurants WHERE borough IN (“Staten Island”, “Queens”, “Bronx”, “Brooklyn”)

Q6: db.restaurants.find(

                     {"address.street" :

                         { $exists : true }

                     }

                   );

This query finds restaurants where the street address is null.

SQL: SELECT \* FROM restaurants WHERE address.street IS NULL

Part 2

Q7: SELECT customer\_id , status    FROM customers;

db.customers.find (

{“customer\_id”: 1, “status”:1}

)

Q8: SELECT \*  FROM customers    WHERE age > 25;

db.customers.find (

{“age”: {$gt: 25}}

)

Q9: SELECT \*  FROM customers  WHERE status = "A" OR rating = 6    LIMIT 5;

db.customers.find (

{

$or: [{“status”: “A”}, {“rating”: 6}]

}

).limit(5);

Q10:

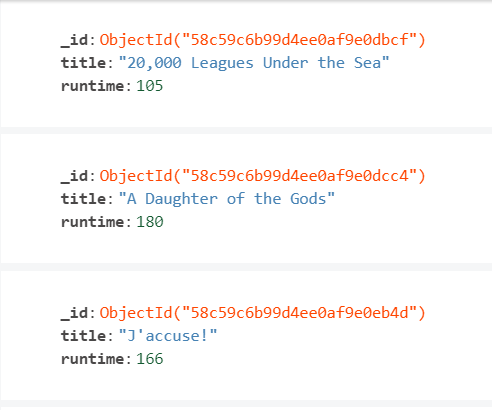
\_id: 0



\_id: 1



No \_id:



\_id: 0 means that the result will not include an ID field. \_id:1 will generate an ObjectId that acts as the primary key and unique identifier. Not including \_id at all will automatically provide the same results as \_id:1. \_id:0 can be used to exclude certain columns that we don’t want.