1. Escriu una consulta per mostrar tots els documents en la col·lecció Restaurants

Donde S2\_04\_Querys\_Restaurante es el nombre de la bbdd

S2\_04\_Querys\_Restaurante.restaurants.find().pretty()

O bien

USE S2\_04\_Querys\_Restaurante;

db.getCollection('restaurants').find().pretty()

1. Escriu una consulta per mostrar el restaurant\_id, name, borough i cuisine per tots els documents en la col·lecció Restaurants

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

1. Escriu una consulta per mostrar el restaurant\_id, name, borough i cuisine, però excloure el camp \_id per tots els documents en la col·lecció Restaurants

db.getCollection('restaurants').find({},{restaurant\_id:1, name:1, borough:1, cuisine:1,"\_id":0 }).pretty()

o db.getCollection('restaurants').find({},{name:1,cuisine:1}).pretty().projection({"\_id":0,grades:0,address:0})

1. Escriu una consulta per mostrar restaurant\_id, name, borough i zip code, però excloure el camp \_id per tots els documents en la col·lecció Restaurants

db.getCollection('restaurants').find({},{restaurant\_id:1, name:1, borough:1,"\_id":0, "address.zipcode":1 }).pretty()

1. Escriu una consulta per mostrar tot els restaurants que estan en el Bronx

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

1. Escriu una consulta per mostrar els primers 5 restaurants que estan en el Bronx

db.getCollection('restaurants').find({borough:"Bronx"}).pretty().limit(5)

1. Escriu una consulta per mostrar el pròxim 5 restaurants després de saltar els primers 5 del Bronx

db.getCollection('restaurants').find({borough:"Bronx"}).pretty().limit(5).skip(5)

1. Escriu una consulta per trobar els restaurants que tenen un score de més de 90

db.getCollection('restaurants').find({"grades.score":{$gt:90}}).pretty()

1. Escriu una consulta per trobar els restaurants que tenen un score de més que 80 però menys que 100

db.getCollection('restaurants').find({$and: [{"grades.score":{$gt:80}},{"grades.score":{$lt:100}}]}).pretty()

1. Escriu una consulta per trobar els restaurants quins localitzen en valor de latitud menys que -95.754168

db.getCollection('restaurants').find({"address.coord":{$lt:-95.754168}}).pretty()

1. Escriu una consulta de MongoDB per a trobar els restaurants que no preparen cap cuisine de 'American' i el seu puntaje de qualificació superior a 70 i latitud inferior a -65.754168

Hay un error en los datos, se carga “American “, por lo que he agregado el espacio en la qry.

db.getCollection('restaurants').find({$and:[{"address.coord":{$lt:-65.754168}},{"cuisine":{$ne: "American "}},{"grades.score":{$gt:70}}]}).pretty()

en caso contrario

db.getCollection('restaurants').find({$and:[{"address.coord":{$lt:-65.754168}},{"cuisine":{$ne: "American"}},{"grades.score":{$gt:70}}]}).pretty()

1. Escriu una consulta per trobar els restaurants quins no preparen cap cuisine de 'American' i va aconseguir un marcador més que 70 i localitzat en la longitud menys que -65.754168. Nota : Fes aquesta consulta sense utilitzar $and operador

db.getCollection('restaurants').find({"address.coord":{$lt:-65.754168},"cuisine":{$ne: "American "},"grades.score":{$gt:70}})

1. Escriu una consulta per trobar els restaurants quins no preparen cap cuisine de 'American ' i va aconseguir un punt de grau 'A' no pertany a Brooklyn. S'ha de mostrar el document segons la cuisine en ordre descendent

db.getCollection('restaurants').find({"borough":{$ne:"Brooklyn"},"cuisine":{$ne: "American "},"grades.grade":"A"}).sort({"cuisine":-1})

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants quin contenir 'Wil' com les tres primeres lletres en el seu nom

db.getCollection('restaurants').find({"name": /^Wil/} ,{restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 }).pretty()

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants quin contenir 'ces' com les últim tres lletres en el seu nom

db.getCollection('restaurants').find({"name": /ces$/},{restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 }).pretty()

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants quin contenir 'Reg' com tres lletres en algun lloc en el seu nom

db.getCollection('restaurants').find({"name": /Reg/},{restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 }).pretty()

1. Escriu una consulta per trobar els restaurants quins pertanyen al Bronx i va preparar qualsevol plat American o xinès

Si hago db.getCollection('restaurants').distinct("cuisine") el nombre en ingles.

db.getCollection('restaurants').find({borough:{$eq:"Bronx"},"cuisine":{$in: ["American ", "Chinese", "Xines"]}}, {restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants que pertanyen a Staten Island o Queens o Bronxor Brooklyn

db.getCollection('restaurants').find({borough:{$in:["Bronx","Staten Island","Queens","Brooklyn"]}},{restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants que no pertanyen a Staten Island o Queens o Bronxor Brooklyn

db.getCollection('restaurants').find({borough:{$nin:["Bronx","Staten Island","Queens","Brooklyn"]}},{restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants que aconsegueixin un marcador quin no és més que 10

db.getCollection('restaurants').find({"grades.score": {$not:{$gt: 10}}},{"borough": 1,"cuisine": 1,"name": 1,"restaurant\_id": 1, "\_id":0,"grades.score":1});

1. Escriu una consulta per trobar el restaurant\_id, name, borough i cuisine per a aquells restaurants que preparen peix excepte 'American' i 'Chinees' o el name del restaurant comença amb lletres 'Wil'

No retorna nada, pero abajo pruebo con Caribean y si lo hace, lo que quiere decir que no hay de Fish

db.getCollection('restaurants').find({$or: [{name: /^Wil/} ,{$and:[{cuisine:{$nin: ["American ", "Chinese"]}},{cuisine: /Fish/}]}]}, {restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 })

db.getCollection('restaurants').find({$or: [{name: /^Wil/} ,{$and:[{cuisine:{$nin: ["American ", "Chinese"]}},{cuisine: /Caribbean/}]}]}, {restaurant\_id:1,name:1,"borough":1,cuisine:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name, i grades per a aquells restaurants que aconsegueixin un grau "A" i un score 11 en dades d'estudi ISODate "2014-08-11T00:00:00Z"

db.getCollection('restaurants').find({grades:{ $elemMatch:{grade:"A",score:{$eq:11},date:{$eq: new ISODate ("2014-08-11T00:00:00Z")}}}}, {restaurant\_id:1,name:1,grades:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name i grades per a aquells restaurants on el 2n element de varietat de graus conté un grau de "A" i marcador 9 sobre un ISODate "2014-08-11T00:00:00Z"

db.getCollection('restaurants').find({$and:[{"grades.1.grade":{$eq:"A"}},{"grades.1.score":{$eq:9}},{"grades.1.date":{$eq: new ISODate ("2014-08-11T00:00:00Z")}}]},{restaurant\_id:1,name:1,grades:1,"\_id":0 })

1. Escriu una consulta per trobar el restaurant\_id, name, adreça i ubicació geogràfica per a aquells restaurants on el segon element del array coord conté un valor quin és més que 42 i fins a 52

db.getCollection('restaurants').find({$and:[{"address.coord.1":{$gt:42}},{"address.coord.1":{$lt:52}}]},{restaurant\_id:1,name:1,address:1,borough:1,"\_id":0 })

1. Escriu una consulta per organitzar el nom dels restaurants en ordre ascendent juntament amb totes les columnes

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

1. Escriu una consulta per organitzar el nom dels restaurants en descendir juntament amb totes les columnas

db.getCollection('restaurants').find().sort({name: -1, borough:-1, cuisine:-1, restaurant\_id:-1}).pretty()

1. Escriu una consulta a organitzar el nom de la cuisine en ordre ascendent i per el mateix barri de cuisine. Ordre descendint

db.getCollection('restaurants').find({},{cuisine:1, borough:1,"\_id":0}).sort({cuisine:1, borough:-1}).pretty()

1. Escriu una consulta per saber tant si totes les direccions contenen el carrer o no

Todas tienen calle

db.getCollection('restaurants').find({"address.street":{$exists:true}},{name: 1,"address.street":1,"\_id":0}).sort({"address.street":1})

O

db.restaurants.aggregate([{$match:{"address.street":{$exists:true}}},{$count: "HAS\_STREET"}])

Ninguna no tiene calle

db.getCollection('restaurants').find({"address.street":{$exists:false}},{name: 1,"address.street":1,"\_id":0}).sort({"address.street":1})

o

db.restaurants.aggregate([{$match:{"address.street":{$exists:false}}},{$count: "HAS\_not\_STREET"}])

1. Escriu una consulta quin seleccionarà tots el documents en la col·lecció de restaurants on el valor del camp coord és Double

db.getCollection('restaurants').find({"address.coord":{$type:["double"]}},{name: 1,"address.coord":1,"\_id":0}).sort({"name":1})

o

db.getCollection('restaurants').find({"address.coord":{$type:["double"]}})

1. Escriu una consulta quin seleccionarà el restaurant\_id, name i grade per a aquells restaurants quins retorns 0 com a resta després de dividir el marcador per 7

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

1. Escriu una consulta per trobar el name de restaurant, borough, longitud i altitud i cuisine per a aquells restaurants que contenen 'mon' com tres lletres en algun lloc del seu name

db.getCollection('restaurants').find({ name:/mon/ }, {name:1,"borough": 1,"address.coord":1,cuisine: 1,"\_id":0}).pretty()

o

db.getCollection('restaurants').find({"\_id": {"$in": db.restaurants.distinct("\_id",{ name:/mon/ })}}, {name:1,"borough": 1,cuisine: 1, "address.coord":1,"\_id":0}).pretty()

1. Escriu una consulta per trobar el name de restaurant, borough, longitud i latitud i cuisine per a aquells restaurants que conteinen 'Mad' com primeres tres lletres del seu name

db.getCollection('restaurants').find({ name:/^Mad/ }, {name:1,borough: 1,"address.coord":1,cuisine: 1,"\_id":0}).pretty()