



Universidade Federal de Itajubá

Atividade 9

COM231 – Banco de Dados II

Fábio Piovani Viviani

2017006774

Ygor Salles Aniceto Carvalho

2017014382

Professora: Vanessa Cristina Oliveira de Souza

Novembro

2020

Questão 1: Crie um novo banco de dados no servidor MongoDB.

use restaurants

Questão 2: Crie uma nova coleção no banco de dados criado na questão 1.

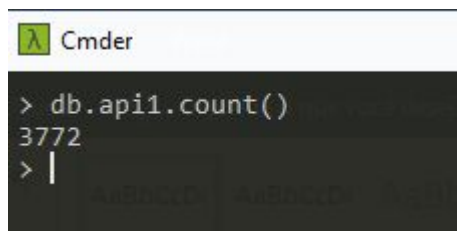
db.createCollection("api1")

Questão 3: Importe os documentos arquivo restaurants.json para a coleção criada na questão 2. Use o mongoimport

mongoimport --db restaurants --collection api1 --file restaurants.json

Questão 4: Dê um count na coleção depois de executar a questão 3.

db.api1.count()



```
> db.api1.count()
3772
> |
```

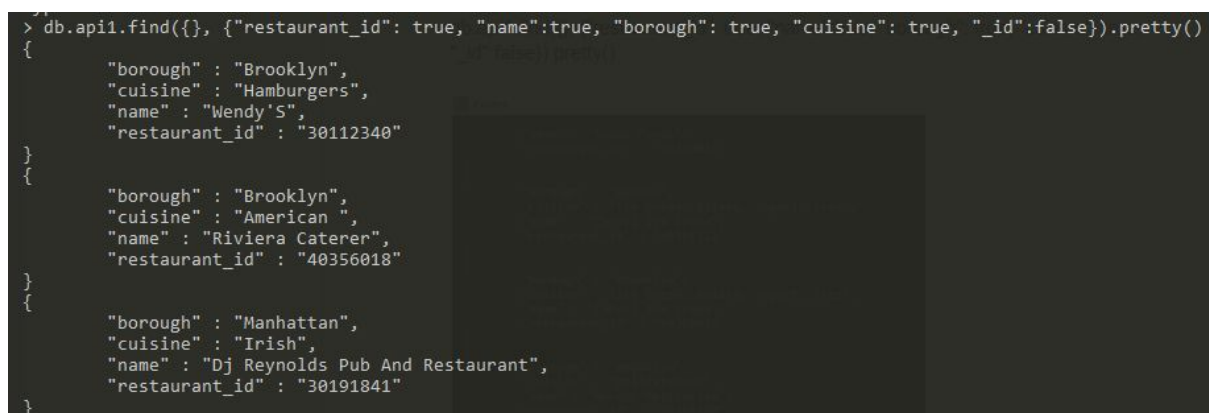
Questão 5: Com base na coleção criada nas questões anteriores responda as questões:

a) Listar todos os documentos da coleção.

db.api1.find().pretty()

b) Listar o id, nome, bairro (borough) e tipo de cozinha de todos os documentos da coleção.

db.api1.find({}, {"restaurant_id": true, "name": true, "borough": true, "cuisine": true, "_id": false}).pretty()



```
> db.api1.find({}, {"restaurant_id": true, "name": true, "borough": true, "cuisine": true, "_id": false}).pretty()
{
  "borough" : "Brooklyn",
  "cuisine" : "Hamburgers",
  "name" : "Wendy'S",
  "restaurant_id" : "30112340"
}
{
  "borough" : "Brooklyn",
  "cuisine" : "American ",
  "name" : "Riviera Caterer",
  "restaurant_id" : "40356018"
}
{
  "borough" : "Manhattan",
  "cuisine" : "Irish",
  "name" : "Dj Reynolds Pub And Restaurant",
  "restaurant_id" : "30191841"
}
```

c) Listar o nome, bairro (borough) e tipo de cozinha de todos os documentos da coleção, excluindo-se o id.

```
db.api1.find({}, {"name":true, "borough": true, "cuisine": true, "_id":false}).pretty()
```

```
> db.api1.find({}, {"name":true, "borough": true, "cuisine": true, "_id":false}).pretty()
{ "borough" : "Brooklyn", "cuisine" : "Hamburgers", "name" : "Wendy'S" }
{
  "borough" : "Brooklyn",
  "cuisine" : "American ",
  "name" : "Riviera Caterer"
}
{
  "borough" : "Manhattan",
  "cuisine" : "Irish",
  "name" : "Dj Reynolds Pub And Restaurant"
}
{
  "borough" : "Bronx",
  "cuisine" : "Bakery",
  "name" : "Morris Park Bake Shop"
}
```

d) Listar todos os restaurantes do bairro Bronx.

```
db.api1.find({"borough": "Bronx"}).pretty()
```

e) Listar os primeiros 5 restaurantes do bairro Bronx.

```
db.api1.find({"borough": "Bronx"}).limit(5).pretty()
```

f) Listar os restaurantes que tenham alguma nota maior que 90.

```
db.api1.find({grades : { $elemMatch:{ "score":{$gt : 90}}}}).pretty()
```

g) Listar os restaurantes que tenham alguma nota maior que 80 e menor que 100.

```
db.api1.find({grades : { $elemMatch:{ "score":{$gt : 80 , $lt :100}}}}).pretty()
```

h) Listar os restaurantes que estejam localizados numa latitude menor que -95.754168.

```
db.api1.find({"address.coord" : {$lt : -95.754168}}).pretty()
```

i) Listar os restaurantes que não preparem cozinha 'American', tenham nota menor que 70 e latitude menor que -65.754168. Os documentos devem ser mostrados de acordo com o tipo de cozinha, em ordem descendente.

```
db.api1.find({$and: [{"cuisine" : { $ne: "American"}}, {"grades.score" : {$lt : 70}}, {"address.coord" : {$lt : -65.754168}}]}.sort({"cuisine": -1}).pretty()
```

j) Listar o id, nome, bairro e cozinha dos restaurantes que não estejam nos bairros 'Staten Island', 'Queens' ou 'Bronxor Brooklyn'

```
db.api1.find({"borough" : { $nin: ["Staten Island", "Queens", "Bronxor", "Brooklyn"] }, {"_id" : true, "name" : true, "borough" : true, "cuisine" : true}).pretty()
```

k) Listar a quantidade de restaurantes por tipo de cozinha;

```
db.api1.aggregate([ {$group : { _id : "$cuisine", count: {$sum : 1}} }]).pretty()
```

l) Listar a quantidade de restaurantes de comida “Mediterranean” por bairro;

```
db.api1.aggregate([ {$match : {"cuisine" : "Mediterranean" }} , {$group : { _id : "$borough", count: {$sum : 1}} }]).pretty()
```

m) Alterar o nome do atributo ‘borough’ para ‘bairro’ em todos os documentos

```
db.api1.update({}, {$rename : {borough : "bairro"}}, {multi : true})
```

n) Alterar o zipcode dos restaurantes cujo valor é “11215” para “112215”

```
db.api1.update({"address.zipcode" : "11215"}, {$set : {"address.zipcode" : "112215"}}, {multi : true})
```

o) Apagar os documentos referentes a cozinha ‘Barbecue’

```
db.api1.remove({"cuisine" : "Barbecue"}, {multi : true})
```