Universidade do Minho
Escola de Engenharia

Bases de Dados NoSQL

Mestrado em Bioinformática

Resolução da Ficha de Trabalho 3

Docente: António Abelha / Cristiana Neto

Aluno

Duarte Alves Velho A96815

2023/2024

➔ **FE03 – “**Introdução ao MongoDB”

Ex 1- Costumers

1- show dbs

2- use costumers

3- db

4- db.createCollection("customers")

5- show collections

6- db.customers.insertOne({first_name: "John", last_name: "Doe", age: 30})

7- db.customers.insertMany([{first_name: "Steven", last_name:"Williams", gender: "male"},{first_name: "Mary", last_name: "Troy", age: 19}])

8- db.customers.insertOne({first_name: "Ric", last_name: "Foe", address: {street: "4 main st", city: "Boston"}})

9- db.customers.insertOne({first_name: "Ana", last_name: "Durant", degree:["phD","Msc"], address: {street: "4 Square Garden", city: "New York"}, age: 32})

10- db.customers.insertOne({first_name: "Natalia", last_name: "Will", age: 44, gender: "female"})

11- db.customers.find()

12- db.customers.updateOne({first_name: 'Ric',last_name: 'Foe'}, {$set:{ age: 45}})

13- db.customers.find({last_name:'Will'})

14- db.customers.updateOne({first_name: 'Seteven',last_name: 'Williams'}, {$set:{ age: 35}})

15- db.customers.updateOne({first_name: 'Ana',age:{$gt: 30}}, {$inc:{ age: 10}})

16- db.customers.updateOne({first_name: 'Ric',last_name: 'Foe'}, {$unset:{ age: ''}})

17- db.customers.updateOne({first_name: 'Jimmy'}, {$set:{first_name: "Jimmy", last_name: "Connors", age: 25, gender:" male"}},{upsert: true})

18- db.customers.find({age: {$gte:25}})

19- db.customers.find({gender: 'male'})

20- db.customers.deleteOne({first_name:'Mary'})

21- db.customers.find({$or: [{first_name:'Ana'},{first_name:'Ric'}]})

Ex 2- Restaurants

```
1- db.restaurants.find()

2- db.restaurants.find({},{restaurant_id:1, name:1, borough:1, cuisine:1})

3- db.restaurants.find({},{_id:0,restaurant_id:1, name:1, borough:1, cuisine:1})

4- db.restaurants.find({},{_id:0,restaurant_id:1, name:1, borough:1,'adress.zipcode':1})

5- db.restaurants.find({borough:'Bronx'})

6- db.restaurants.find({borough:'Bronx'}). limit(5)

7- db.restaurants.find({borough:'Bronx'}).skip(5). limit(5)

8- db.restaurants.find({'grades.score':{$gt:90}})

9- db.restaurants.find({'grades.score':{$gt:80, $lt:100}})

10- db.restaurants.find({'address.coord.coordinates.0':{$lt:-95.754168}})

11- db.restaurants.find({ $and: [ { cuisine: { $ne: "American" } }, { 'grades.score': { $gt: 70 } }, {
'address.coord.coordinates.0': { $lt: -65.754168 }}]} )

12- db.restaurants.find({ 'cuisine': { $ne: "American" }, 'grades.score': { $gt: 70 },
'address.coord.coordinates.0': { $lt: -65.754168 } })

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

14- db.restaurants.find({ 'borough': "Bronx", 'cuisine': { $in: ["American", "Chinese"] } })

15- db.restaurants.find({'address.street': {$exists: true}})

16- db.restaurants.find().sort({'cuisine': 1, 'borough': -1})

17- db.restaurants.find({ 'address.coord.coordinates.1': { $gt: 42, $lte: 52 } }, { 'restaurant_id': 1,
'name': 1, 'address': 1, 'coord': 1 })

18- db.restaurants.find({ 'grades.score': { $not: { $gt: 10 } }
},{restaurante_id:1,name:1,borough:1,cuisine:1, _id:0})

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

20- db.restaurants.updateOne( { 'restaurant_id': "30191841" }, { $set: { 'name':
"NewName Restaurant" } } )
```