Fundamentos de MongoDB

Practica 3 Aggregation Framework y Map Reduce

# Objetivo

* Aprender a utilizar los pipelines y operadores de Aggregation framework y el método MapReduce

# Procedimiento

1. Inicia el servidor mongoDB mediante el demonio mongod.exe y conéctate al servidor usando mongo CLI mongo.exe
2. Crea la base de datos devmongodb si es que no existe.
3. Crea la colección people en devmongodb mediante el siguiente comando:

db.people.insertMany([

{name:"Mary",gender:"female",size:1.72,weight:54,phone:"+51 2345679", age:25 ,email:"mary.smith@gmail.com",company:"AWS",isActive:true,address:[{primary:"100 Boulevard Miami",secondary:"303 St. Geneva Rome"}]},

{name:"Charles",gender:"male",size:1.86,weight:90,phone:"+86 7345674", age:35 ,email:"charles.slate@yahoo.com",company:"Redhat",isActive:true},

{name:"Danny",gender:"male",size:1.91,weight:102,phone:"+1 8445663", age:25 ,email:"danny.lasalle@growing.com",company:"AWS",isActive:false,address:[{primary:"102 Bronco Texas",secondary:"404 Borbon Street Lousiana"}]},

{name:"Richard",gender:"male",size:1.82,weight:83,phone:"+86 2545671", age:35 ,email:"richard.jhonson@gmail.com",company:"Open cloud",isActive:true},

{name:"Yenny",gender:"female",size:1.75,weight:56,phone:"+51 2345459", age:29 ,email:"yenny.sullivan@gmail.com",company:"AWS",isActive:false,address:[{primary:"505 Renfer Madrid",secondary:"345 Republica Barcelona"}]},

{name:"Rob",gender:"male",size:1.79,weight:85,phone:"+51 7145679", age:35 ,email:"rob.sax@mshaw.com",company:"Microsoft Inc",isActive:false},

{name:"Brain",gender:"male",size:1.90,weight:92,phone:"+1 8947679", age:45 ,email:"brain.dawner@yahoo.com",company:"AWS",isActive:true} ,

{name:"Jane",gender:"male",size:1.56,weight:55,phone:"+1 8345663", age:25 ,email:"jane.gross@growing.com",company:"MongoDB Inc",isActive:true}

]);

1. Ejecuta los siguientes comandos e interpreta la salida a partir de los resultados

db.people.aggregate({

$project: {

isActive: 1,

company: 1,

name: 1,

age: 1,

addedAge: {

$add: [

"$age",

10

]

},

upperName: {

$toUpper: "$name"

}

}

});

db.people.aggregate({

$project: {

isActive: 1,

name: 1,

company: 1,

age: 1,

mainAddress: "$address.primary"

}

});

1. Ejecuta los siguientes comandos e interpreta la salida a partir de los resultados

db.people.aggregate({

$match: {

isActive: true

}

},

{

$project: {

isActive: 1,

name: 1,

mainAddress: "$address.primary"

}

});

1. Modifica el comando del punto 5 para que muestre solo las personas activas que trabajan en la compañía “AWS”
2. Ejecuta los siguientes comandos e interpreta la salida a partir de los resultados

db.people.aggregate({

$group: {

\_id: {

gender: "$gender"

},

averageAge: {

$avg: "$age"

},

count: {

$sum: 1

}

}

});

db.people.aggregate({

$match: {

isActive: true

}

},

{

$group: {

\_id: {

gender: "$gender",

age: "$age"

},

count: {

$sum: 1

}

}

},

{

$project: {

\_id: 0,

type: "$\_id",

total: "$count"

}

});

db.people.aggregate(

{

$group:

{\_id:"$gender",

totalAge:{$sum:"$age"},

totalPeople:{$sum:1}

}

});

1. Ejecuta los siguientes comandos e interpreta la salida a partir de los resultados

db.people.aggregate({

$limit: 3

});

db.people.aggregate({

$skip: 2

});

db.people.aggregate({

$sort: {

age: -1

}

},

{

$skip: 2

},

{

$limit: 1

})

1. Modifica el comando del punto 8, para que muestre la segunda persona con menos edad de la colección people
2. Ejecuta los siguientes comandos e interpreta la salida a partir de los resultados

db.people.aggregate({

$match: {

name: "Yenny"

}

},

{

$project: {

name: 1,

address: 1

}

});

db.people.aggregate({

$match: {

name: "Yenny"

}

},

{

$project: {

name: 1,

address: 1

}

},

{

$unwind: "$address"

});

¿Cuál es la diferencia entre estos dos comandos?

1. Usando las siguientes funciones y MapReduce calcula el número de personas por género y la suma de las edades.

var map = function () {

emit (this.gender, { age: this.age, count: 1 });

}

var reduce = function(keys,values)

{

var reduced = {

totalPeople:0,

totalAge:0

}

for (var i=0; i < values.length;i++)

{

reduced.totalPeople+=1;

reduced.totalAge+=values[i].age;

}

return reduced;

}

db.people.mapReduce(map,reduce,{out:"MapReduceResult"});

db.MapReduceResult.find().pretty();

Importante:

* Al finalizar la practica enviarla al email del instructor: [carlos.carreno.pe@gmail.com](mailto:carlos.carreno.pe@gmail.com)
* Asunto: MongoDB – Practica 3 - <Nombres y Apellidos>
* No olvidar adjuntar la solución de la practica en formato Word.