BIG DATA LAB

Name: Sarthak Kesarwani
USN: 1NT19IS141

Date:31.05.2022

1.Problem Statement & Dataset

Create a collection named "Employee" under the "EmployeeDB" database with each document in the format shown below Table

>use Sarthak_EmployeeDB
>db.createCollection(“Employee”)
>show collections

```
> use Sarthak_EmployeeDB
switched to db Sarthak_EmployeeDB
> db.createCollection("Employee")
{ "ok" : 1 }
```

Inserting the values:

>db. Employee InsertMany ([(Firstname: "John", Middlename: "C" Lastname: "Cena", Age:29, Salary: 70000, Designation: "Vice President", Rol "Middlename:"S", Lastname: "Singh", Age:24, Salary: 65000, Designation: "Associate", Role: "Team Lead"), (Firstname: "Juan", Middlename: 46000, Designation: "President", Role: "Software Developer"), (Firstname: "Mohan", Middlename: "D", Lastname: "Rai", Age: 30, Salary: 40000, are Designer")]);

```
> db.Employee.insertMany([{Firstname:"John",Middlename:"C",Lastname:"Cena",Age:29,Salary:70000,Designation:"Vice President",Rol
",Middlename:"S",Lastname:"Singh",Age:24,Salary:65000,Designation:"Associate",Role:"Team Lead"}, {Firstname:"Juan",Middlename:"
46000,Designation:"President",Role:"Software Developer"}, {Firstname:"Mohan",Middlename:"D",Lastname:"Rai",Age:30,Salary:40000,
are Designer"}]);
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("6295992b3105a79a7b9eaa65"),
                ObjectId("6295992b3105a79a7b9eaa66"),
                ObjectId("6295992b3105a79a7b9eaa67"),
                ObjectId("6295992b3105a79a7b9eaa68")
        ]
}
```

2. Queries

1. Populate the database with at least 15 documents

>db.Employee.find.pretty()

```
> db.Employee.find().pretty()
{
        "_id" : ObjectId("629594050b0b41a60c06cc85"),
        "Firstname" : "Ryan",
        "Middlename" : "Lee",
        "Lastname" : "Orton",
        "Age" : 25,
        "Salary" : 40000,
        "Designation" : "Manager",
        "Role" : "Software Developer"
}
{
        "_id" : ObjectId("6295947a0b0b41a60c06cc86"),
        "Firstname" : "Johnyy",
        "Middlename" : "Del",
        "Lastname" : "Depp",
        "Age" : 30,
        "Salary" : 50000,
        "Designation" : "Vice President",
        "Role" : "TeamLead"
}
```

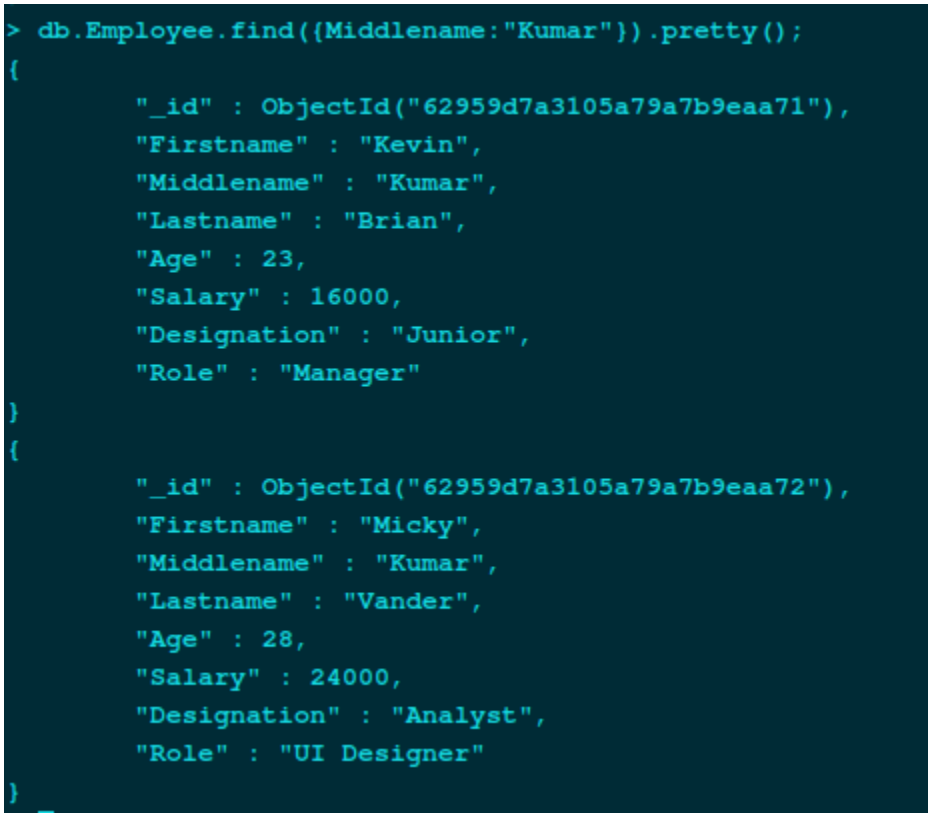
2. List all the records having salary in the range of 20000 - 35000(Exclusive)

>db.Employee.find({$and:[{Salary:{$gt:20000}},{Salary:{$lt:35000}}]})

```
> db.Employee.find({Salary:{$gt:20000},Salary:{$lt:35000}}).pretty();
{
        "_id" : ObjectId("62959d033105a79a7b9eaa6d"),
        "Firstname" : "Kevin",
        "Middlename" : "O",
        "Lastname" : "Brian",
        "Age" : 23,
        "Salary" : 26000,
        "Designation" : "Vice President",
        "Role" : "Manager"
}
{
        "_id" : ObjectId("62959d033105a79a7b9eaa6e"),
        "Firstname" : "Micky",
        "Middlename" : "F",
        "Lastname" : "Vander",
        "Age" : 28,
        "Salary" : 30000,
        "Designation" : "Analyst",
        "Role" : "UI Designer"
}
```

3. List all the Employees whose Middle name is "Kumar"

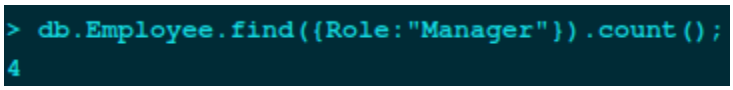
> db.Employee.find({Middlename:"Kumar"}).pretty();



```
> db.Employee.find({Middlename:"Kumar"}).pretty();
{
        "_id" : ObjectId("62959d7a3105a79a7b9eaa71"),
        "Firstname" : "Kevin",
        "Middlename" : "Kumar",
        "Lastname" : "Brian",
        "Age" : 23,
        "Salary" : 16000,
        "Designation" : "Junior",
        "Role" : "Manager"
}
{
        "_id" : ObjectId("62959d7a3105a79a7b9eaa72"),
        "Firstname" : "Micky",
        "Middlename" : "Kumar",
        "Lastname" : "Vander",
        "Age" : 28,
        "Salary" : 24000,
        "Designation" : "Analyst",
        "Role" : "UI Designer"
}
```

4. Count the number of Employees who has the role "Manager" in the Role field

> db.Employee.find({Role:"Manager"}).count();



```
> db.Employee.find({Role:"Manager"}).count();
4
```

5. Find out all the documents who have age < 35 and salary in the range of 30000-50000

>db.Employee.find({$and:[{Salary:{$gt:30000}},{Salary:{$lt:50000}},{Age:{$lt:35}}]}).pretty();

```
> db.Employee.find({$and:[{Age:{$lt:35}},{Salary:{$gte:30000}},{Salary:{$lte:35000}}]}).pretty();
{
        "_id" : ObjectId("6295955a0b0b41a60c06cc89"),
        "Firstname" : "Erell",
        "Middlename" : "H",
        "Lastname" : "Huelevan",
        "Age" : 28,
        "Salary" : 35000,
        "Designation" : "Junior Officer",
        "Role" : "Tester"
}
{
        "_id" : ObjectId("62959d033105a79a7b9eaa6e"),
        "Firstname" : "Micky",
        "Middlename" : "F",
        "Lastname" : "Vander",
        "Age" : 28,
        "Salary" : 30000,
        "Designation" : "Analyst",
        "Role" : "UI Designer"
}
```

6. Delete an Employee whose "Firstname" is "Rajesh" and having the designation as "Scientist"

>db.Employee.remove({$and:[{Firstname:"Rajesh"},{Designation:"Scientist"}]})

```
> db.Employee.remove({$and:[{FirstName:"Rajesh"},{Designation:"Scientist"}]});
WriteResult({ "nRemoved" : 0 })
```

7. Update all the Employees whose role is "Team Lead" with a salary of 55650 INR

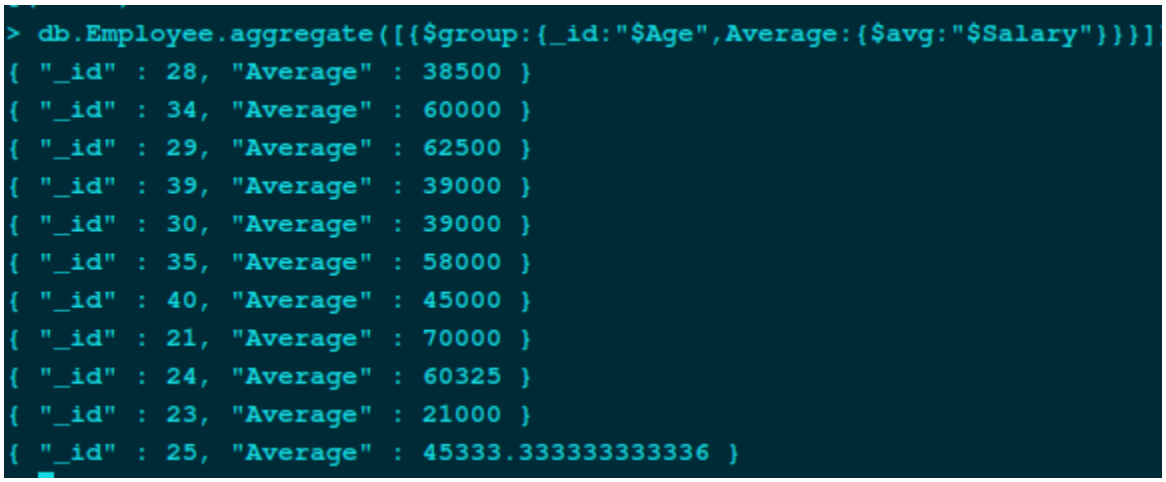
> db.Employee.updateMany({Role:"Team Lead"},{$set:{Salary:55650}})

```
> db.Employee.updateMany({Role:"Team Lead"},{$set:{Salary:55650}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

```
{
        "_id" : ObjectId("6295992b3105a79a7b9eaa66"),
        "Firstname" : "Priya",
        "Middlename" : "S",
        "Lastname" : "Singh",
        "Age" : 24,
        "Salary" : 55650,
        "Designation" : "Associate",
        "Role" : "Team Lead"
}
```

8. Group all the Employees by their age(common age should be there) and calculate the average salary obtained in each group

>db.Employee.aggregate([{$group:{_id:"$Age",Average:{$avg:"$Salary"}}}])



9. Apply the map-reduce to perform the above operation and obtain the results

> var mapfunction=function(){emit(this.Age,this.Salary)}
> var reducefunction=function(key,values){return Array.avg(values)}
>db.Employee.mapReduce(mapfunction,reducefunction,('out':'result'})
>db.Employee.mapReduce(mapfunction,reducefunction,{'out':'result'})
{ "result" : "result", "ok" : 1 }

> db.result.find()

```
> var mapfunction = function(){emit(this.Age,this.Salary)}
> var reducefunction = function(key,values){return Array.avg(values)}
> db.Employee.mapReduce(mapfunction,reducefunction,{'out':'result'});
{ "result" : "result", "ok" : 1 }
> db.result.find();
{ "_id" : 28, "value" : 38500 }
{ "_id" : 34, "value" : 60000 }
{ "_id" : 29, "value" : 62500 }
{ "_id" : 39, "value" : 39000 }
{ "_id" : 30, "value" : 39000 }
{ "_id" : 35, "value" : 58000 }
{ "_id" : 40, "value" : 45000 }
{ "_id" : 21, "value" : 70000 }
{ "_id" : 24, "value" : 60325 }
{ "_id" : 23, "value" : 21000 }
{ "_id" : 25, "value" : 45333.333333333336 }
```