# BIG DATA LAB PROGRAM – I

**Name** :Chandu.M
**USN** : 1NT19IS044
**Sec** : A

**Employee Data Base**

```
> use EmployeeDB
switched to db EmployeeDB
> db.createCollection("Employee")
{ ok: 1 }
> show dbs
EmployeeDB  8.19 kB
admin        41 kB
config     36.9 kB
local      73.7 kB
```

## Queries

### 1. Populate the database with atleast 15 documents

```
> db.Employee.insert({"Name":
{"Firstname":"chandu","Middlename":"M","LastName":"manju"},"Age":25,"Salary"
:50000,"Designation":"Employee","Role":["Manager", "Team Lead","Software
Developer"]})

>
db.Employee.insert({"Name":{"Firstname":"Rajesh","Middlename":"Kumar","LastN
ame":"p"},"Age":27,"Salary":50000,"Designation":"Scientist","Role":["Supervisor",
"Project Lead"]})

>
db.Employee.insert({"Name":{"Firstname":"punith","Middlename":"Kumar","LastN
ame":"R"},"Age":29,"Salary":75000,"Designation":"Employee","Role":["Manager",
"Team Lead","Data Analyst"]})

>
db.Employee.insert({"Name":{"Firstname":"Koush","Middlename":"varma","LastNa
me":"C"},"Age":27,"Salary":60000,"Designation":"Employee","Role":["Manager","
Data Scientist"]})
```

>
db.Employee.insert({"Name":{"Firstname":"Amir","Middlename":"Kumar","LastName":"M"},"Age":29,"Salary":65000,"Designation":"Employee","Role":["Team Lead","Software Developer"]})

>
db.Employee.insert({"Name":{"Firstname":"Krishna","Middlename":"Sai","LastName":"G"},"Age":26,"Salary":55000,"Designation":"Employee","Role":["Manager","UI/UX Designer"]})

>
db.Employee.insert({"Name":{"Firstname":"Vishnu","Middlename":"Tej","Lastname":"T"},"Age":35,"Salary":75000,"Designation":"Employee","Role":["UI Designer","Software Developer","Team Lead"]});

>
db.Employee.insert({"Name":{"Firstname":"Amay","Middlename":"Nayak","Lastname":"R"},"Age":25,"Salary":35000,"Designation":"Scientist","Role":["Manager","Software Developer","ML Scientist"]});
{
>
db.Employee.insert({"Name":{"Firstname":"Harish","Middlename":"Kumar","Lastname":"J"},"Age":27,"Salary":30000,"Designation":"Employee","Role":["Manager","Software Developer"]});

>
db.Employee.insert({"Name":{"Firstname":"Ajay","Middlename":"Raj","Lastname":"N"},"Age":28,"Salary":35000,"Designation":"Employee","Role":["Team Lead","Manager","Software Developer"]});

>
db.Employee.insert({"Name":{"Firstname":"Ram","Middlename":"Charan","Lastname":"R"},"Age":26,"Salary":39000,"Designation":"Scientist","Role":["Manager","Software Developer","ML Scientist"]});

>
db.Employee.insert({"Name":{"Firstname":"Vijay","Middlename":"Kumar","Lastname":"K"},"Age":28,"Salary":33000,"Designation":"Scientist","Role":["Team Lead","ML Scientist"]});

>
db.Employee.insert({"Name":{"Firstname":"Arjun","Middlename":"Kumar","Lastname":"C"},"Age":30,"Salary":30000,"Designation":"Employee","Role":["Team Lead","Tester"]});

```
>
db.Employee.insert({"Name":{"Firstname":"Charan","Middlename":"Sai","Lastname
":"B"},"Age":26,"Salary":23000,"Designation":"Employee","Role":["Team
Lead","ML Scientist"]});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("628a1bf8149e859a20395f82") }
}
>
db.Employee.insert({"Name":{"Firstname":"Yashwin","Middlename":"Raj","Lastna
me":"D"},"Age":28,"Salary":26000,"Designation":"Employee","Role":["Manager","
ML Scientist"]});
```

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

```
> db.Employee.find({"Salary":{$gte:20000,$lt:35000}})
[
  {
    _id: ObjectId("628a18e7149e859a20395f7d"),
    Name: { Firstname: 'Harish', Middlename: 'Kumar', Lastname: 'y' },
    Age: 27,
    Salary: 30000,
    Designation: 'Employee',
    Role: [ 'Manager', 'Software Developer' ]
  },
  {
    _id: ObjectId("628a1a5d149e859a20395f80"),
    Name: { Firstname: 'Vijay', Middlename: 'Kumar', Lastname: 'K' },
    Age: 28,
    Salary: 33000,
    Designation: 'Scientist',
    Role: [ 'Team Lead', 'ML Scientist' ]
  },
  {
    _id: ObjectId("628a1b4b149e859a20395f81"),
    Name: { Firstname: 'Arjun', Middlename: 'Kumar', Lastname: 'C' },
    Age: 30,
    Salary: 30000,
    Designation: 'Employee',
    Role: [ 'Team Lead', 'Tester' ]
  },
```

```
  {
    _id: ObjectId("628a1bf8149e859a20395f82"),
    Name: { Firstname: 'Charan', Middlename: 'Sai', Lastname: 'B' },
    Age: 26,
    Salary: 23000,
    Designation: 'Employee',
    Role: [ 'Team Lead', 'ML Scientist' ]
  },
  {
    _id: ObjectId("628a1c42149e859a20395f83"),
    Name: { Firstname: 'Yashwin', Middlename: 'Raj', Lastname: 'D' },
    Age: 28,
    Salary: 26000,
    Designation: 'Employee',
    Role: [ 'Manager', 'ML Scientist' ]
  }
]
```

### 3. List all the Employee whose Middle name is "Kumar"

```
> db.Employee.find({"Name.Middlename":"Kumar"})
[
  {
    _id: ObjectId("628a0d03149e859a20395f76"),
    Name: { Firstname: 'Rajesh', Middlename: 'Kumar', LastName: 'p' },
    Age: 27,
    Salary: 50000,
    Designation: 'Scientist',
    Role: [ 'Supervisor', 'Project Lead' ]
  },
  {
    _id: ObjectId("628a0db9149e859a20395f77"),
    Name: { Firstname: 'punith', Middlename: 'Kumar', LastName: 'R' },
    Age: 29,
    Salary: 75000,
    Designation: 'Employee',
    Role: [ 'Manager', 'Team Lead', 'Data Analyst' ]
  },
  {
    _id: ObjectId("628a0e7f149e859a20395f79"),
    Name: { Firstname: 'Amir', Middlename: 'Kumar', LastName: 'M' },
    Age: 29,
    Salary: 65000,
    Designation: 'Employee',
    Role: [ 'Team Lead', 'Software Developer' ]
```

```
  },
  {
    _id: ObjectId("628a18e7149e859a20395f7d"),
    Name: { Firstname: 'Harish', Middlename: 'Kumar', Lastname: 'J' },
    Age: 27,
    Salary: 30000,
    Designation: 'Employee',
    Role: [ 'Manager', 'Software Developer' ]
  },
  {
    _id: ObjectId("628a1a5d149e859a20395f80"),
    Name: { Firstname: 'Vijay', Middlename: 'Kumar', Lastname: 'K' },
    Age: 28,
    Salary: 33000,
    Designation: 'Scientist',
    Role: [ 'Team Lead', 'ML Scientist' ]
  },
  {
    _id: ObjectId("628a1b4b149e859a20395f81"),
    Name: { Firstname: 'Arjun', Middlename: 'Kumar', Lastname: 'C' },
    Age: 30,
    Salary: 30000,
    Designation: 'Employee',
    Role: [ 'Team Lead', 'Tester' ]
  }
]
```

### 4. Count the number of Employees who has a role "Manager" in the Role eld

```
> db.Employee.find({"Role":{$in:["Manager"]}}).count();
9
```

### 5. Find out all the documents who have age < 35 and salary in the range of 30000-35000

```
> db.Employee.find({$and:[{"Age":{$lt:35}},{"Salary":{$gte:30000,$lt:35000}}]})
[
  {
    _id: ObjectId("628a18e7149e859a20395f7d"),
xx    Name: { Firstname: 'Harish', Middlename: 'Kumar', Lastname: 'J' },
    Age: 27,
    Salary: 30000,
    Designation: 'Employee',
```

```
    Role: [ 'Manager', 'Software Developer' ]
  },
  {
    _id: ObjectId("628a1a5d149e859a20395f80"),
    Name: { Firstname: 'Vijay', Middlename: 'Kumar', Lastname: 'K' },
    Age: 28,
    Salary: 33000,
    Designation: 'Scientist',
    Role: [ 'Team Lead', 'ML Scientist' ]
  },
  {
    _id: ObjectId("628a1b4b149e859a20395f81"),
    Name: { Firstname: 'Arjun', Middlename: 'Kumar', Lastname: 'C' },
    Age: 30,
    Salary: 30000,
    Designation: 'Employee',
    Role: [ 'Team Lead', 'Tester' ]
  }
]
```

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

```
>
db.Employee.remove({$and:[{"Name.Firstname":"Rajesh"},{"Designation":"Scientis
t"}]})

{ acknowledged: true, deletedCount: 1 }
```

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

```
> db.Employee.updateMany({"Role":{$in:["Team
Lead"]}},{$set:{"Salary":55650}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 8,
  modifiedCount: 8,
  upsertedCount: 0
}
```

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

```
> db.Employee.aggregate([{$group : {_id: "$Age", avg_salary : {$avg : "$Salary"}}}])
[
  { _id: 30, avg_salary: 55650 },
  { _id: 28, avg_salary: 45766.666666666664 },
  { _id: 26, avg_salary: 49883.333333333336 },
  { _id: 29, avg_salary: 55650 },
  { _id: 35, avg_salary: 55650 },
  { _id: 27, avg_salary: 45000 },
  { _id: 25, avg_salary: 45325 }
]
```

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

```
> var mac=function(){emit(this.Age,this.Salary)}
[Function: mac]
> var rec=function(key,avg_salary){return Array.avg(avg_salary)}
[Function: rec]
> db.Employee.mapReduce(mac,rec,{out:"res"});
{ result: 'res', ok: 1 }
> db.res.find().pretty()
[
  { _id: 27, value: 45000 },
  { _id: 25, value: 45325 },
  { _id: 29, value: 55650 },
  { _id: 35, value: 55650 },
  { _id: 26, value: 49883.333333333336 },
  { _id: 28, value: 45766.666666666664 },
  { _id: 30, value: 55650 }
]
```