MangoDB Class

#MangoDB Notes

MongoDb Aggregate Functions

Distinct() Method

- **Distinct() method will return distinct (Unique) values from the collections.**

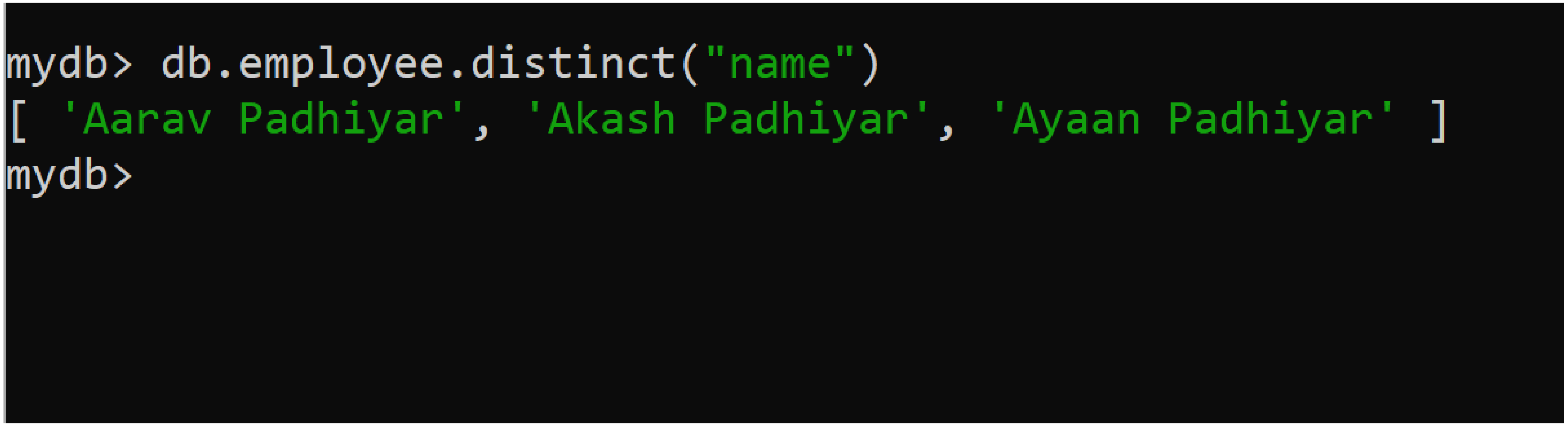
- Syntax :-

```
db.COLLECTION_NAME.distinct(“KeyName”)
```

- Example :-

```
db.employee.distinct(“name”)
```

Distinct() Method

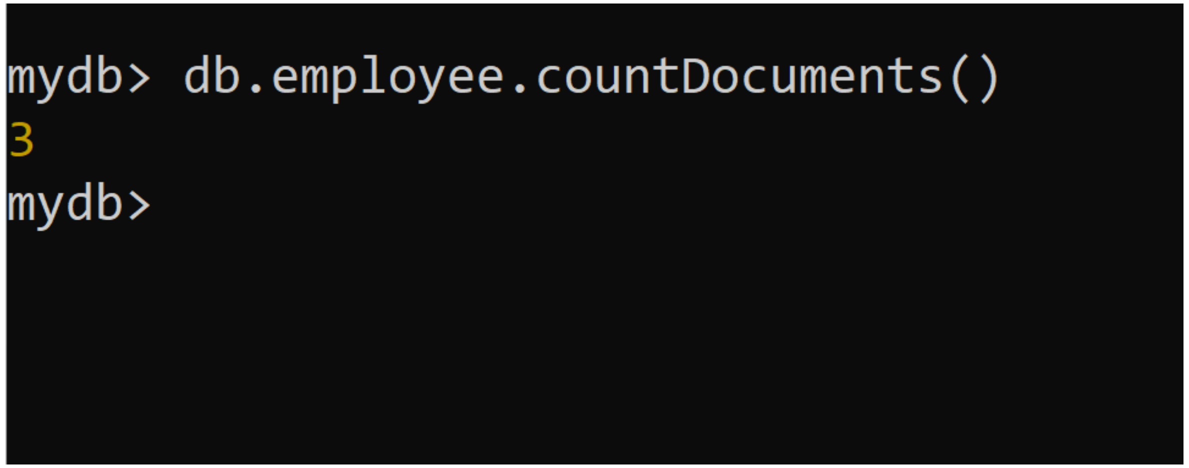


```
mydb> db.employee.distinct("name")
[ 'Aarav Padhiyar', 'Akash Padhiyar', 'Ayaan Padhiyar' ]
mydb>
```

Aggregate Functions

- **countDocuments() method counts the number of documents that matches to the selection criteria**
- Syntax :-

db.COLLECTION_NAME.countDocuments()



```
mydb> db.employee.countDocuments()
3
mydb>
```

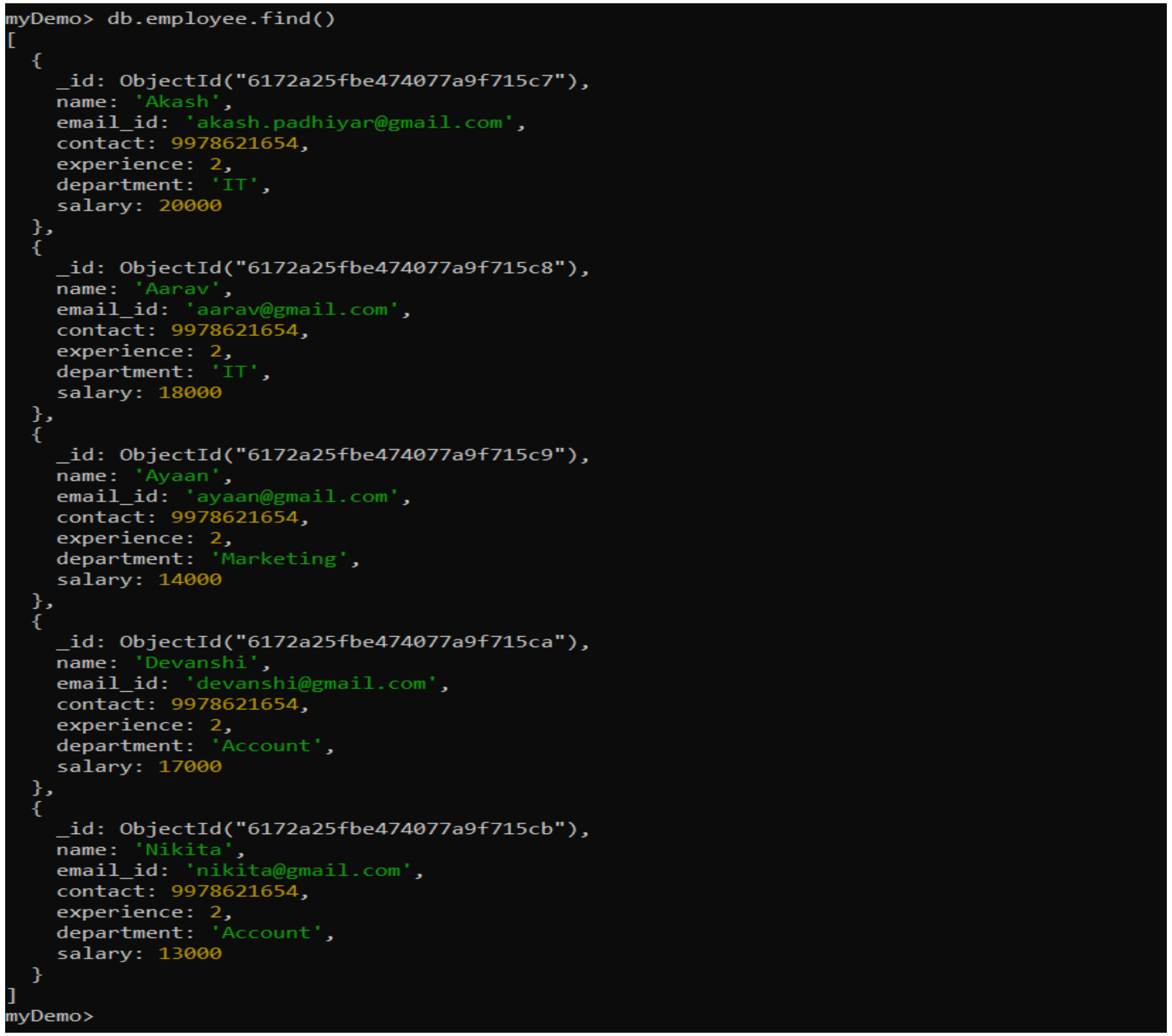
Aggregate Functions With Group By Clause

Different expressions used by Aggregate function

Expression	Description
$sum	Sums up the defined value from all documents in the collection.
$avg	Calculates the average values from all the documents in a collection
$min	Return the minimum of all values of documents in a collection
$max	Return the maximum of all values of documents in a collection

$addToSet	Inserts values to an array but no duplicates in the resulting document
$push	Inserts values to an array in the resulting document
$first	Returns the first document from the source document
$last	Returns the last document from the source document

Employee Collection’s Output



```
myDemo> db.employee.find()
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    name: 'Akash',
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 20000
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c8"),
    name: 'Aarav',
    email_id: 'aarav@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 18000
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c9"),
    name: 'Ayaan',
    email_id: 'ayaan@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'Marketing',
    salary: 14000
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715ca"),
    name: 'Devanshi',
    email_id: 'devanshi@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'Account',
    salary: 17000
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715cb"),
    name: 'Nikita',
    email_id: 'nikita@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'Account',
    salary: 13000
  }
]
myDemo>
```

Aggregate Function – SUM

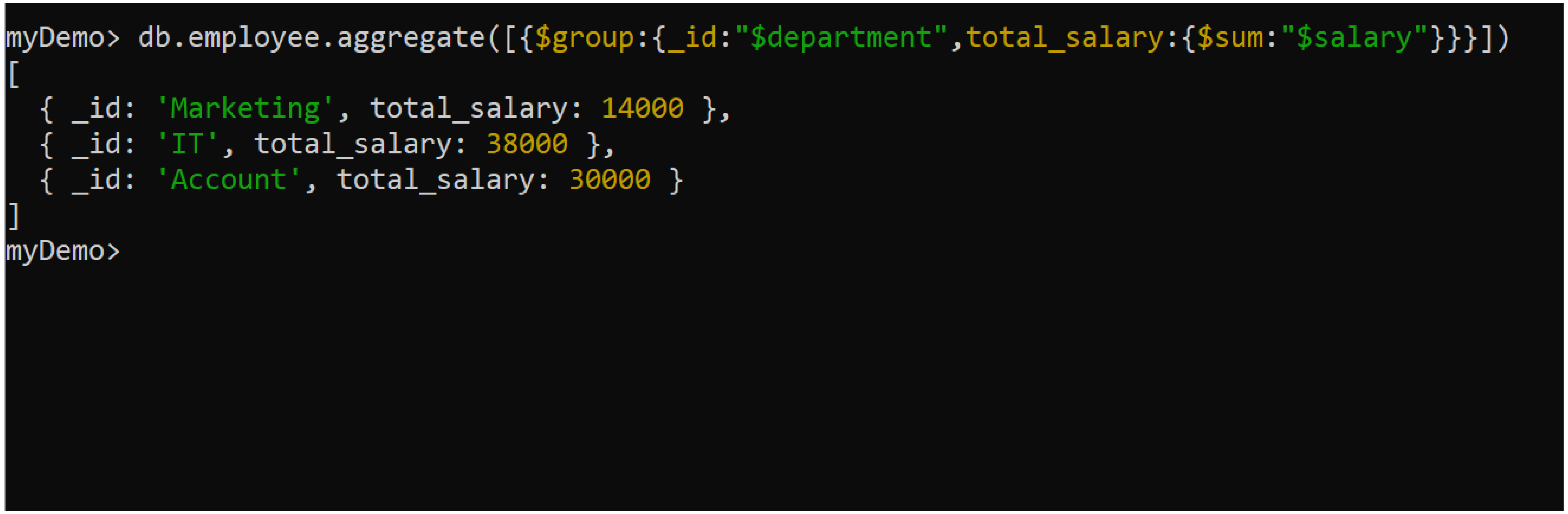
- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
          $group:
          {_id:"$department",
          total_salary:{$sum:"$salary"}
           }
}])
```

Sum



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",total_salary:{$sum:"$salary"}}}])
[
  { _id: 'Marketing', total_salary: 14000 },
  { _id: 'IT', total_salary: 38000 },
  { _id: 'Account', total_salary: 30000 }
]
myDemo>
```

Aggregate Function – AVERAGE

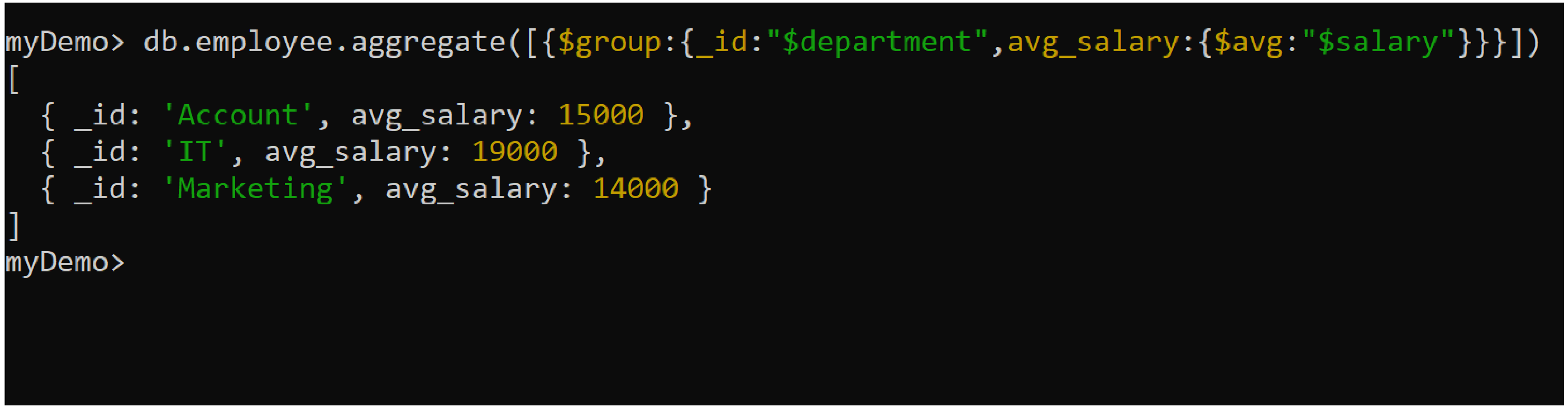
- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
          $group:
          {_id:"$department",
          avg_salary:{$avg:"$salary"}
           }
}])
```

AVERAGE



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",avg_salary:{$avg:"$salary"}}}])
[
  { _id: 'Account', avg_salary: 15000 },
  { _id: 'IT', avg_salary: 19000 },
  { _id: 'Marketing', avg_salary: 14000 }
]
myDemo>
```

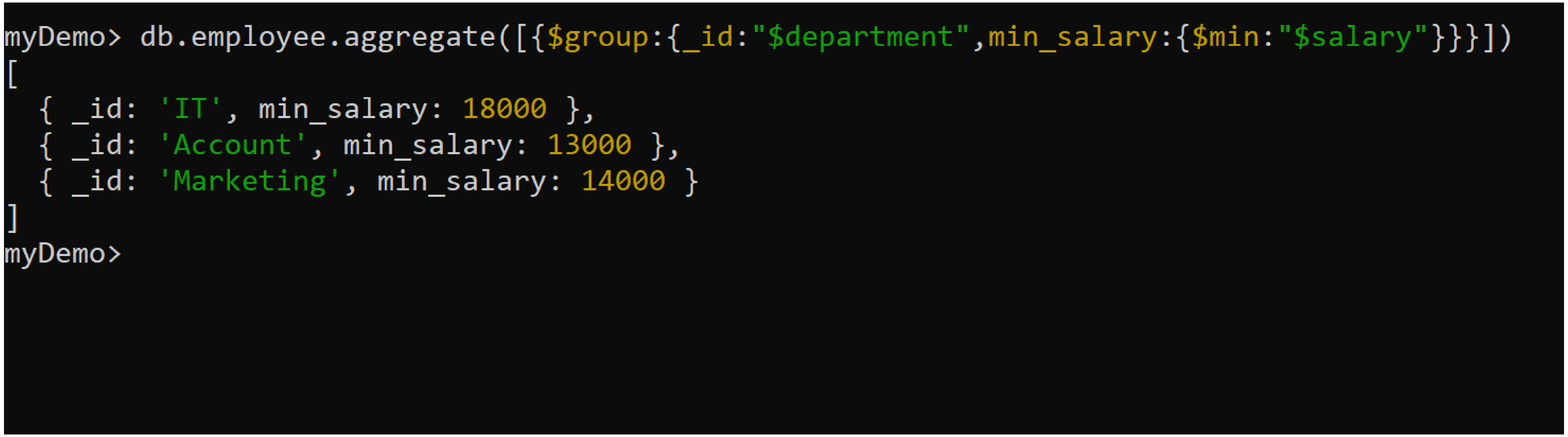
Aggregate Function – MIN

- Syntax :-

```
db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)
```

- Example :-

```
db.employee.aggregate([ {
          $group:
          {_id:"$department",
          min_salary:{$min:"$salary"}
           }
}])
```


```
myDemo> db.employee.aggregate([{$group:{_id:"$department",min_salary:{$min:"$salary"}}}])
[
  { _id: 'IT', min_salary: 18000 },
  { _id: 'Account', min_salary: 13000 },
  { _id: 'Marketing', min_salary: 14000 }
]
myDemo>
```

Aggregate Function – MAX

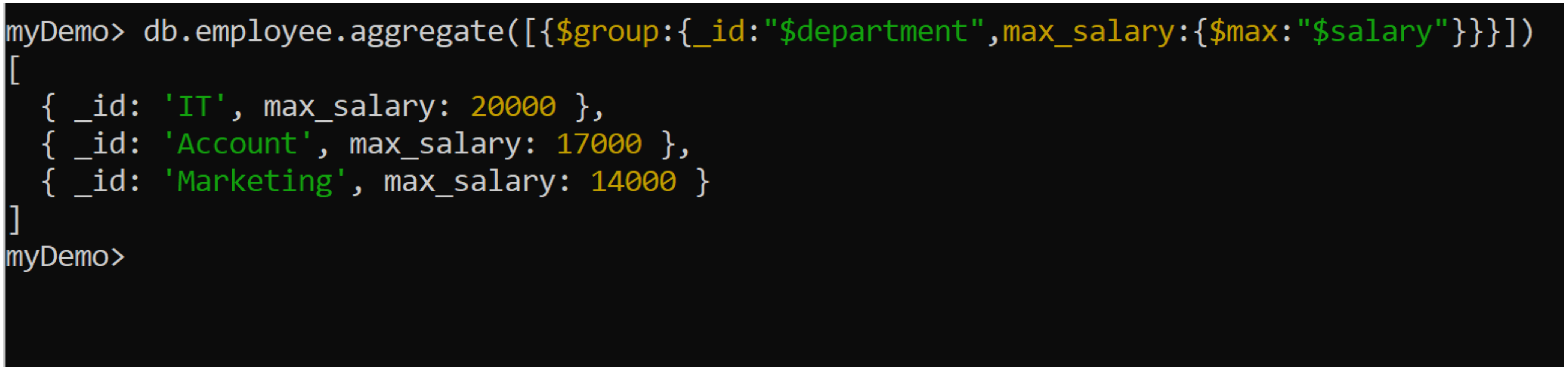
- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
        $group:
        {_id:"$department",
        max_salary:{$max:"$salary"}
         }
}])
```

Max



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",max_salary:{$max:"$salary"}}}])
[
  { _id: 'IT', max_salary: 20000 },
  { _id: 'Account', max_salary: 17000 },
  { _id: 'Marketing', max_salary: 14000 }
]
myDemo>
```

Aggregate Function – PUSH

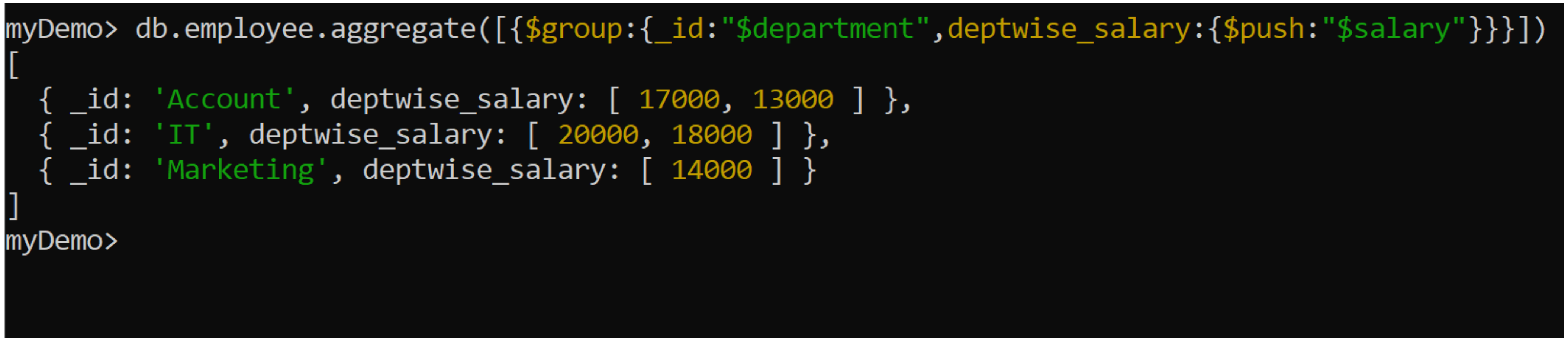
- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
        $group:
        {_id:"$department",
        deptwise_salary:{$push:"$salary"}
         }
}])
```

Push



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",deptwise_salary:{$push:"$salary"}}}])
[
  { _id: 'Account', deptwise_salary: [ 17000, 13000 ] },
  { _id: 'IT', deptwise_salary: [ 20000, 18000 ] },
  { _id: 'Marketing', deptwise_salary: [ 14000 ] }
]
myDemo>
```

Aggregate Function – ADDTOSET

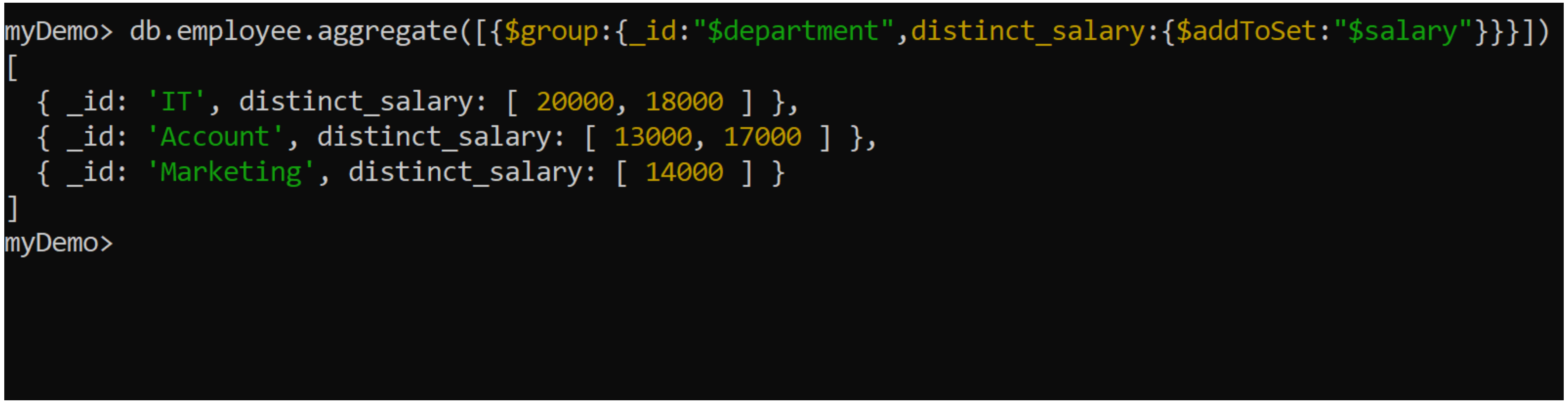
- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
        $group:
        {_id:"$department",
        distinct_salary:{$addToSet:"$salary"}
         }
}])
```

addToSet



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",distinct_salary:{$addToSet:"$salary"}}}])
[
  { _id: 'IT', distinct_salary: [ 20000, 18000 ] },
  { _id: 'Account', distinct_salary: [ 13000, 17000 ] },
  { _id: 'Marketing', distinct_salary: [ 14000 ] }
]
myDemo>
```

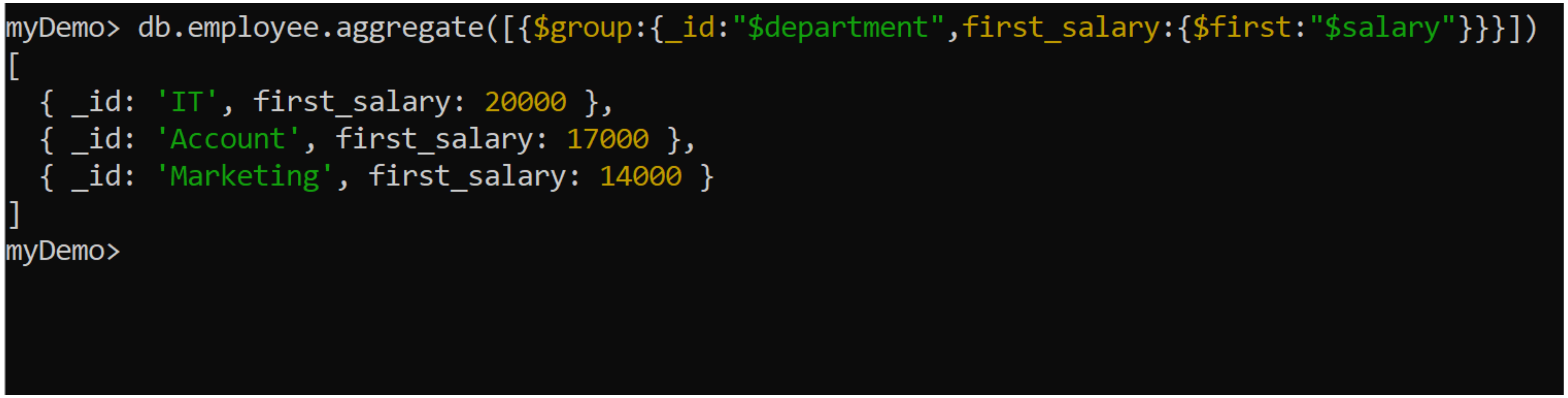
Aggregate Functions – FIRST

- Syntax :-

db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)

- Example :-

```
db.employee.aggregate([ {
        $group:
        {_id:"$department",
        first_salary:{$first:"$salary"}
         }
}])
```



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",first_salary:{$first:"$salary"}}}])
[
  { _id: 'IT', first_salary: 20000 },
  { _id: 'Account', first_salary: 17000 },
  { _id: 'Marketing', first_salary: 14000 }
]
myDemo>
```

Aggregate Functions – LAST

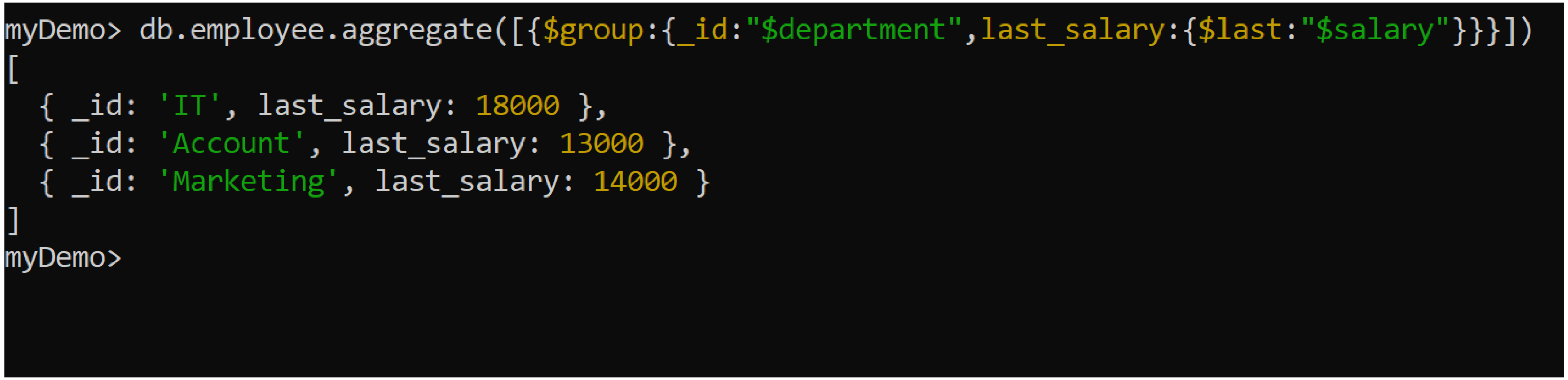
- Syntax :-

```
db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)
```

- Example :-

```
db.employee.aggregate([ {
          $group:
          {_id:"$department",
          last_salary:{$last :"$salary"}
           }
}])
```

Last



```
myDemo> db.employee.aggregate([{$group:{_id:"$department",last_salary:{$last:"$salary"}}}])
[
  { _id: 'IT', last_salary: 18000 },
  { _id: 'Account', last_salary: 13000 },
  { _id: 'Marketing', last_salary: 14000 }
]
myDemo>
```

Incrementing value of documents in collection

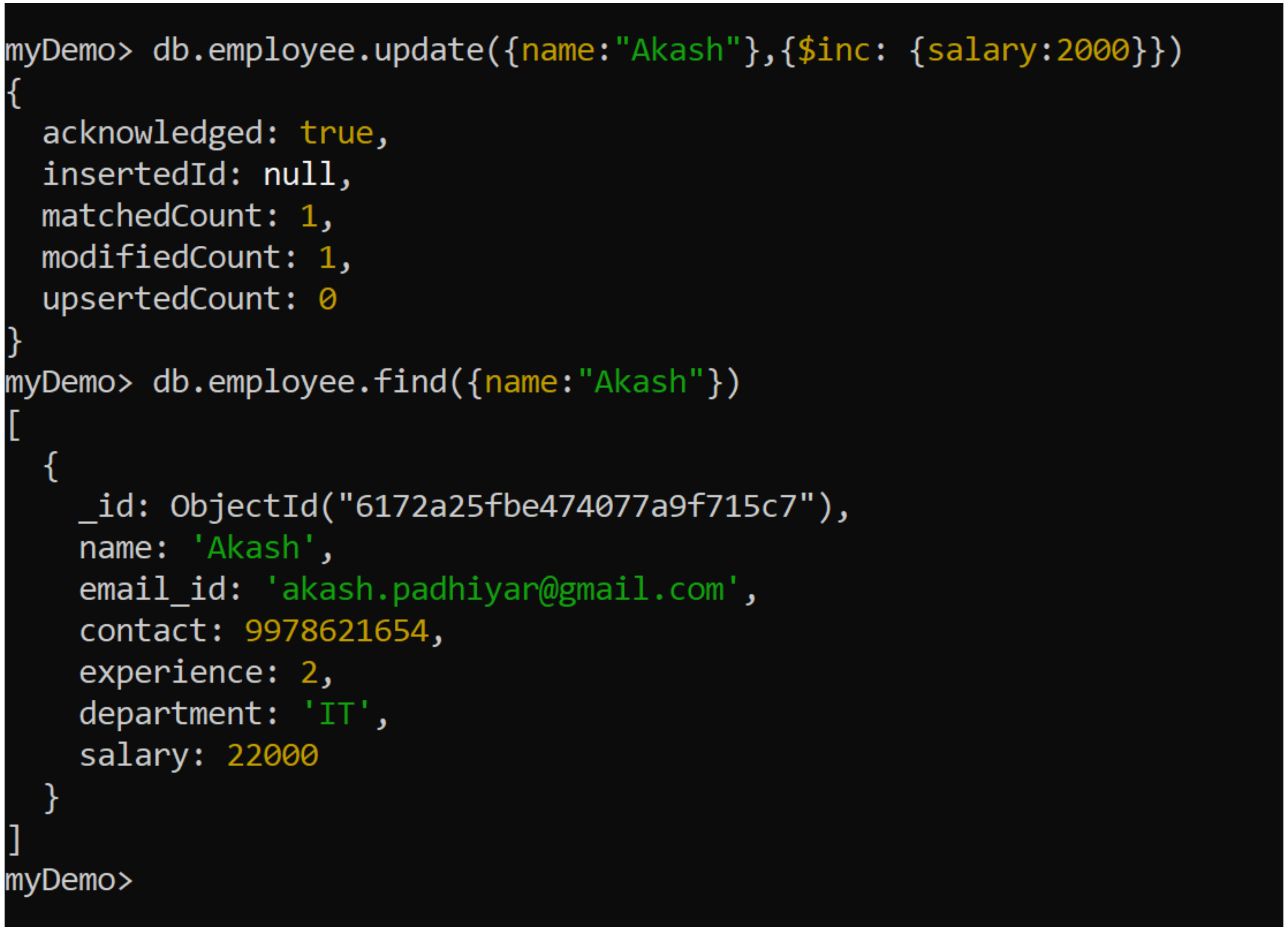
- **$inc is used to increment and decrement scalar values of documents in collection.**

- Syntax :-

  db.COLLECTION_NAME.update({},{$inc:{“key” : “value”}}, {multi:true})

- Example :-

  db.employee.update({name: "Akash Padhiyar"},{$inc: {"salary":2000}})



```
myDemo> db.employee.update({name:"Akash"},{$inc: {salary:2000}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
myDemo> db.employee.find({name:"Akash"})
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    name: 'Akash',
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 22000
  }
]
myDemo>
```

Decrementing value of documents in collection

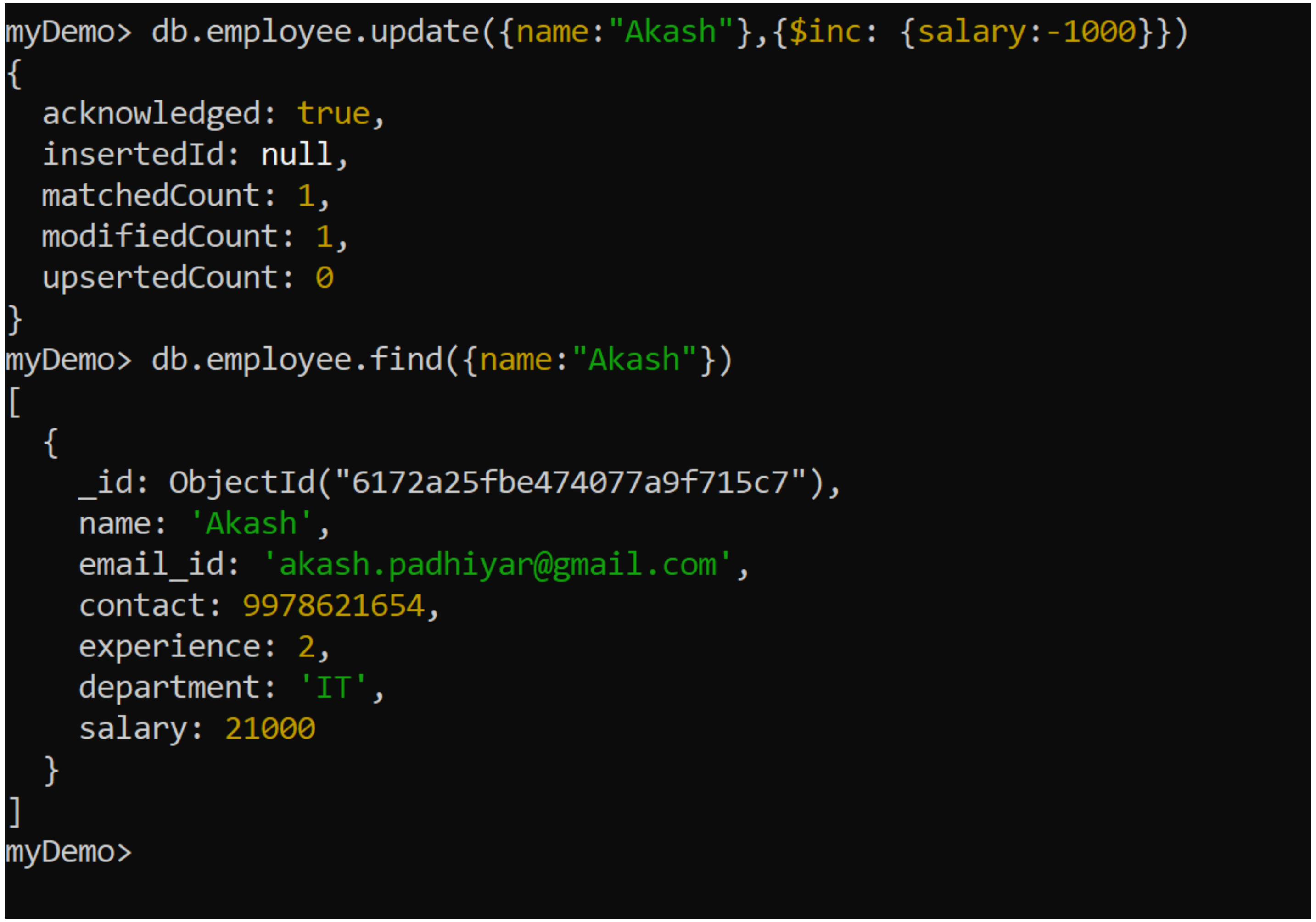
- **$inc is used to increment and decrement scalar values of documents in collection.**

- Syntax :-

```
db.COLLECTION_NAME.update({},{$inc:{“key” : “value”}}, {multi:true})
```

- Example :-

```
db.employee.update({name: "Akash Padhiyar"},{$inc: {"salary": -1000}})
```



```
myDemo> db.employee.update({name:"Akash"},{$inc: {salary:-1000}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
myDemo> db.employee.find({name:"Akash"})
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    name: 'Akash',
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 21000
  }
]
myDemo>
```

Multiply value of documents in collection

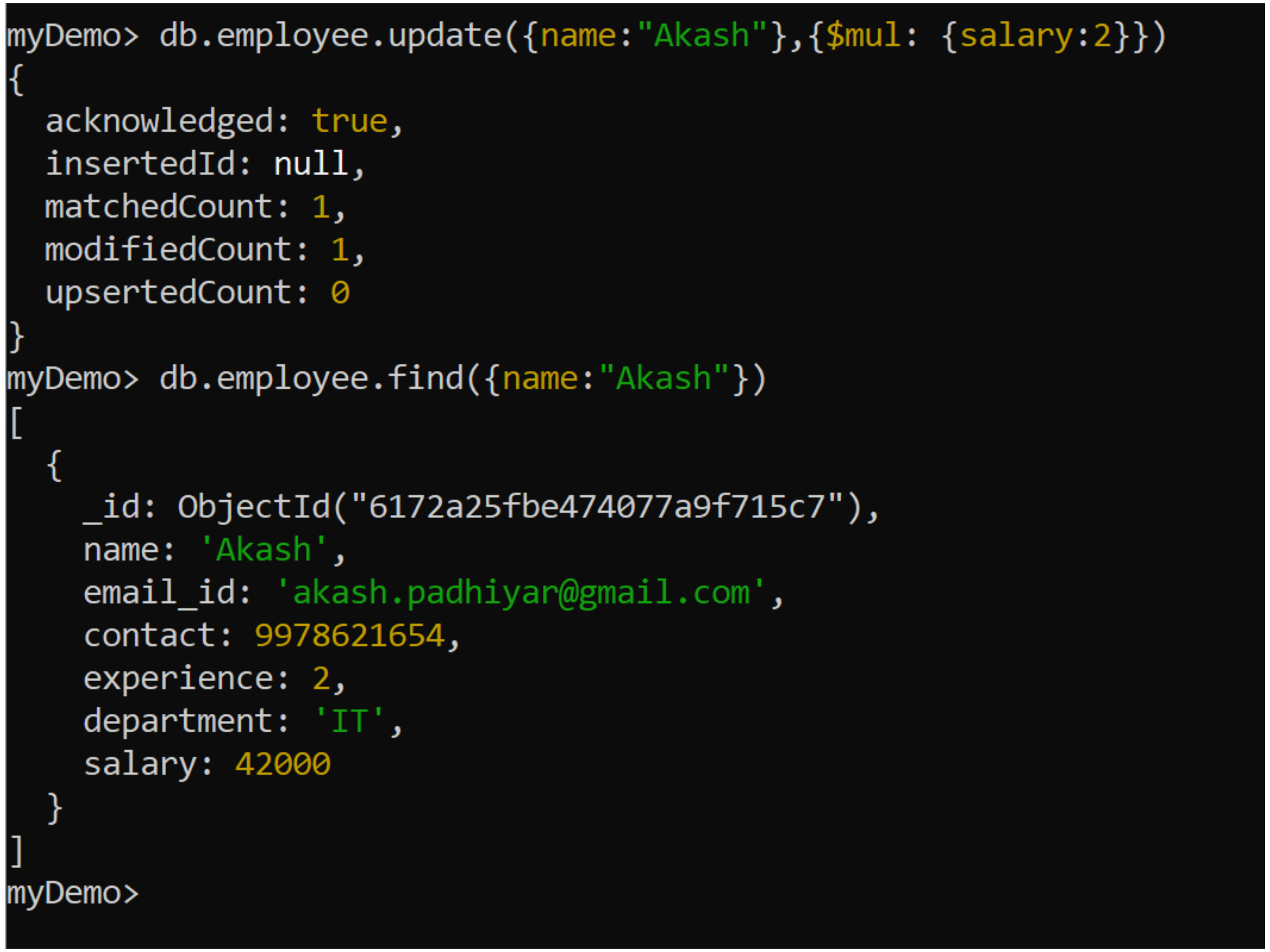
- **$mul is used to multiply the field by the given value. It allows both positive and negative values.**

- Syntax :-

  db.COLLECTION_NAME.update({},{$mul:{“key”: value}}, {multi:true})

- Example :-

  db.employee.update({name: "Akash Padhiyar"},{$mul: {"salary": 2}})



```
myDemo> db.employee.update({name:"Akash"},{$mul: {salary:2}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
myDemo> db.employee.find({name:"Akash"})
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    name: 'Akash',
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 42000
  }
]
myDemo>
```

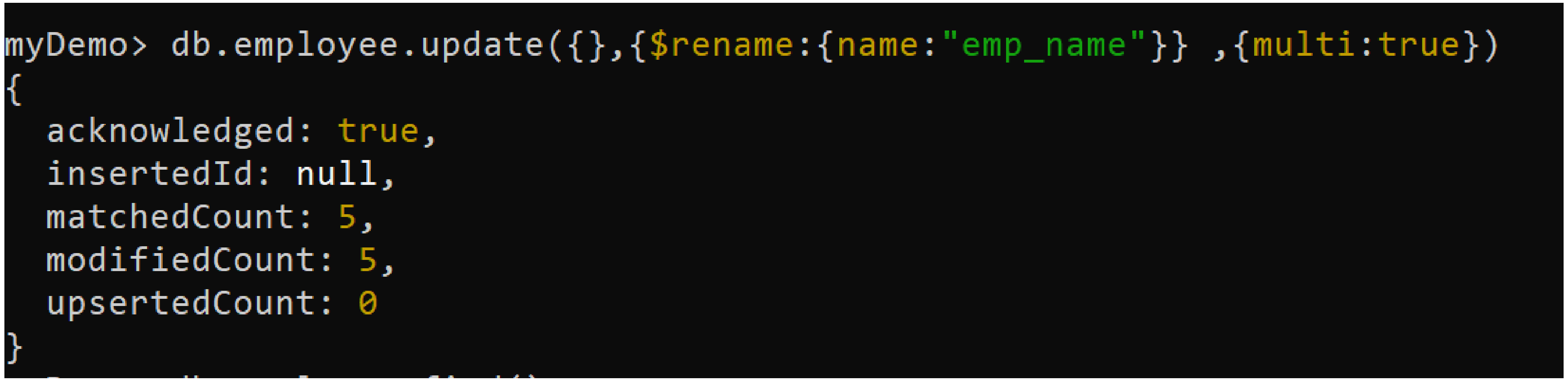
Renaming the name of the field of document in collection

- Syntax :-

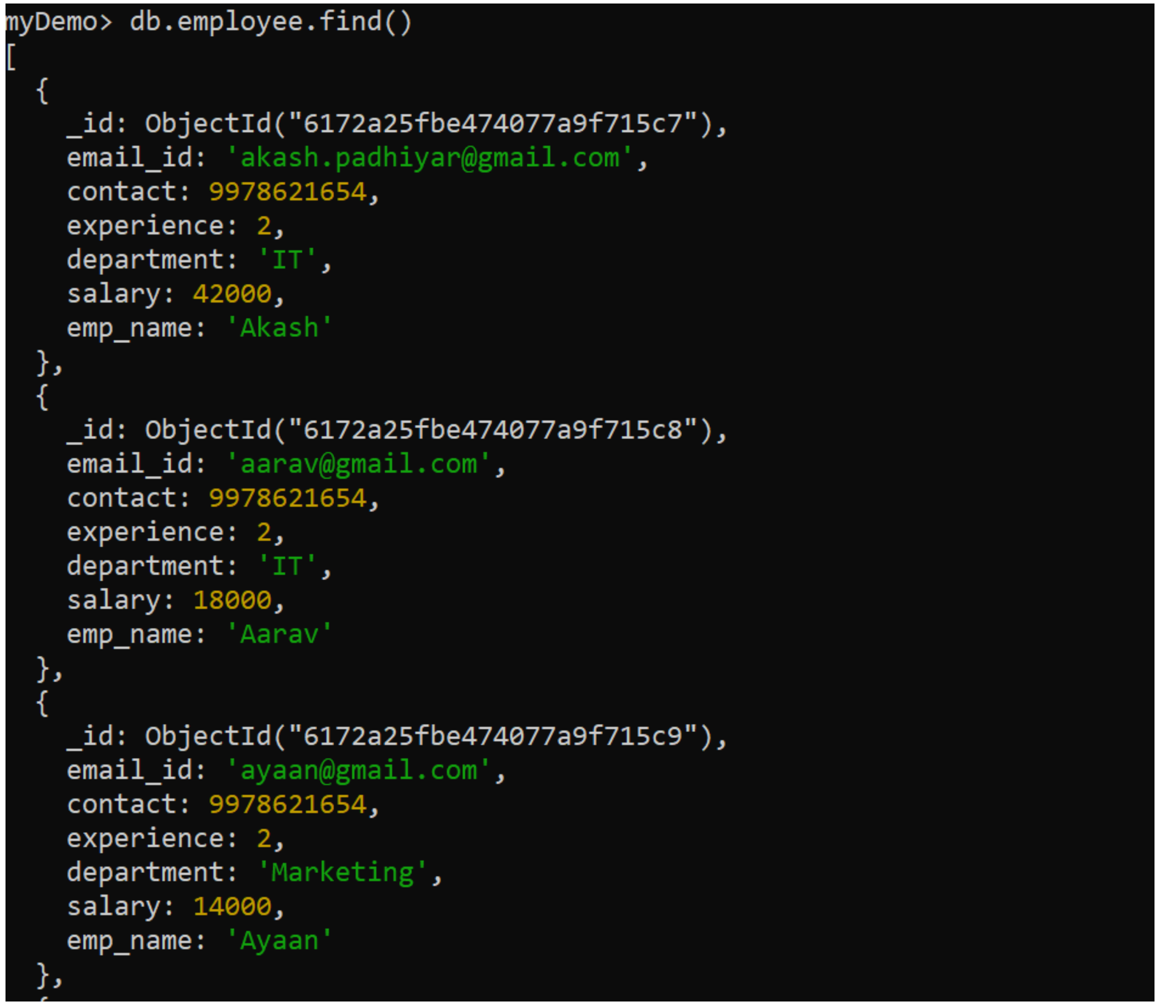
db.COLLECTION_NAME.update({},{$rename:{“oldkey”: “newkey”}}, {multi:true})

- Example :-

db.employee.update({},{$rename: {"name":"emp_name"}}, {multi:true})



```
myDemo> db.employee.update({},{$rename:{name:"emp_name"}} ,{multi:true})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}
```

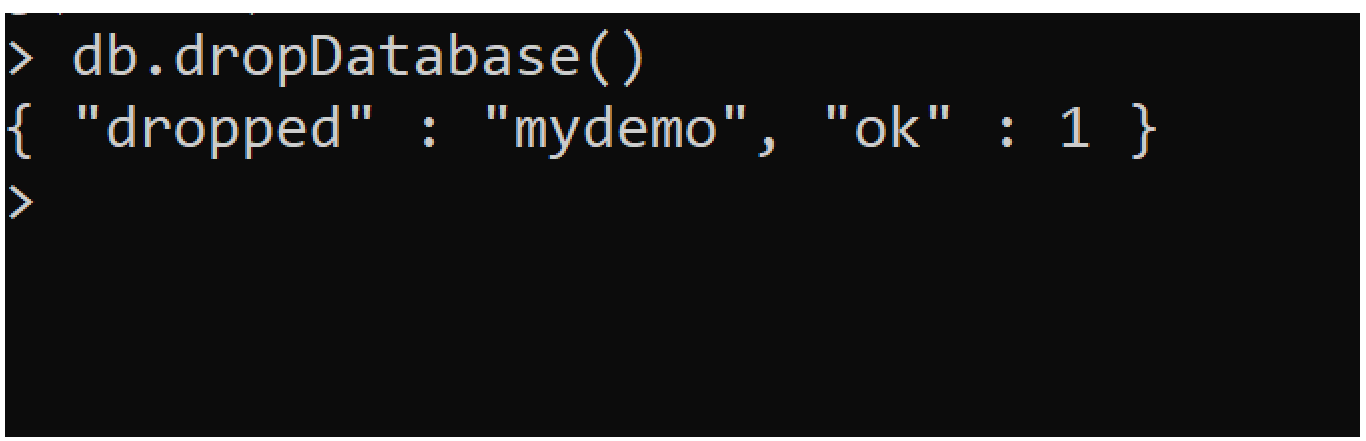


```
myDemo> db.employee.find()
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 42000,
    emp_name: 'Akash'
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c8"),
    email_id: 'aarav@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 18000,
    emp_name: 'Aarav'
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c9"),
    email_id: 'ayaan@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'Marketing',
    salary: 14000,
    emp_name: 'Ayaan'
  },
```

Drop Database

- Command :- **db.dropDatabase()**

- **This command is used to drop current database**



```
> db.dropDatabase()
{ "dropped" : "mydemo", "ok" : 1 }
>
```

Deleting many documents in collection

- **This command deletes all the documents fulfilling the criteria.**

- Syntax :-

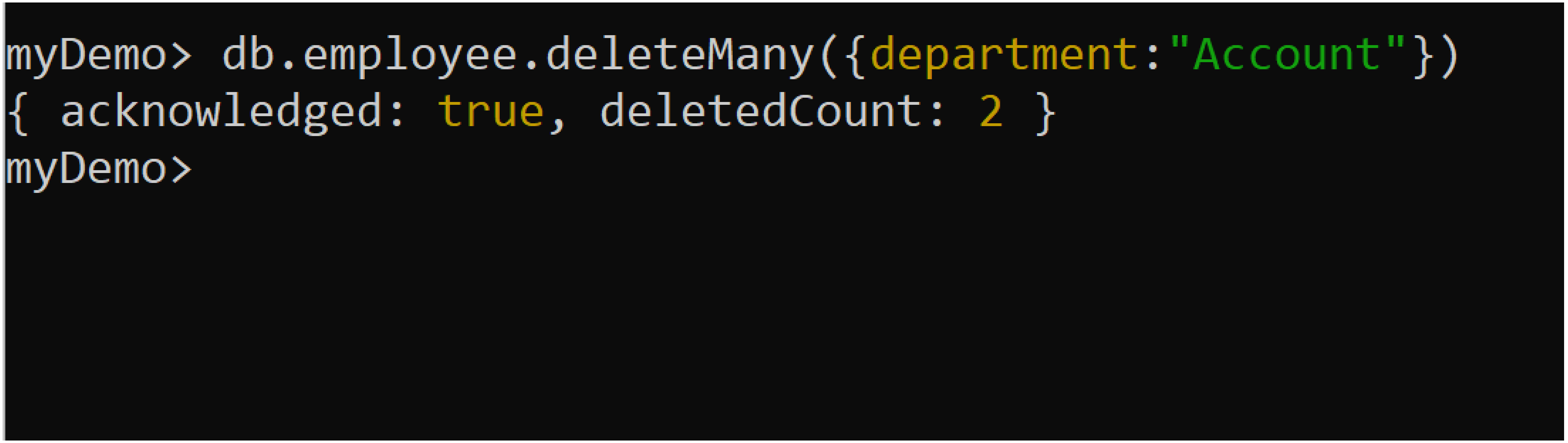
```
db.COLLECTION_NAME.deleteMany( {  } )
```

- Example :-

```
db.employee.deleteMany({“department”: “MARKETING”})
```

- Output :-

```
{ "acknowledged" : true, "deletedCount" : 2 }
```



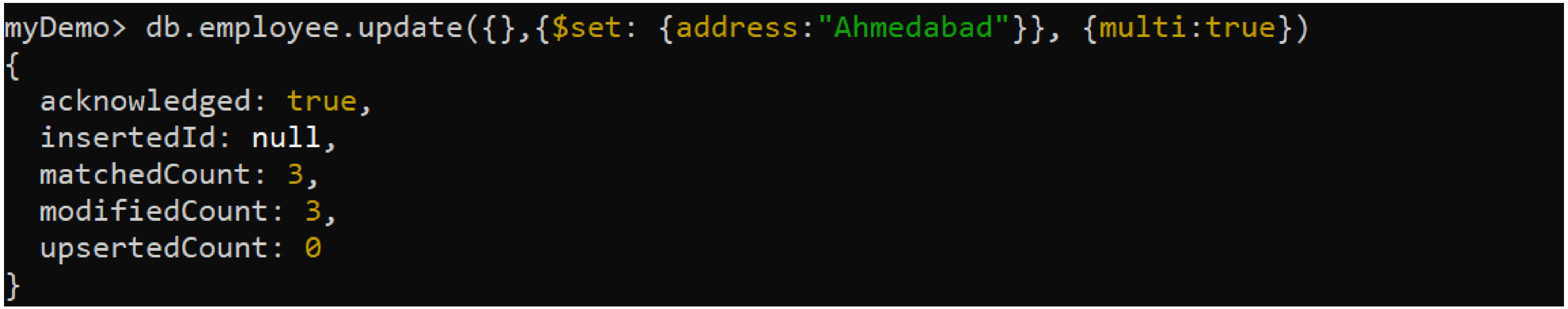
```
myDemo> db.employee.deleteMany({department:"Account"})
{ acknowledged: true, deletedCount: 2 }
myDemo>
```

Adding new field to all the documents in collection

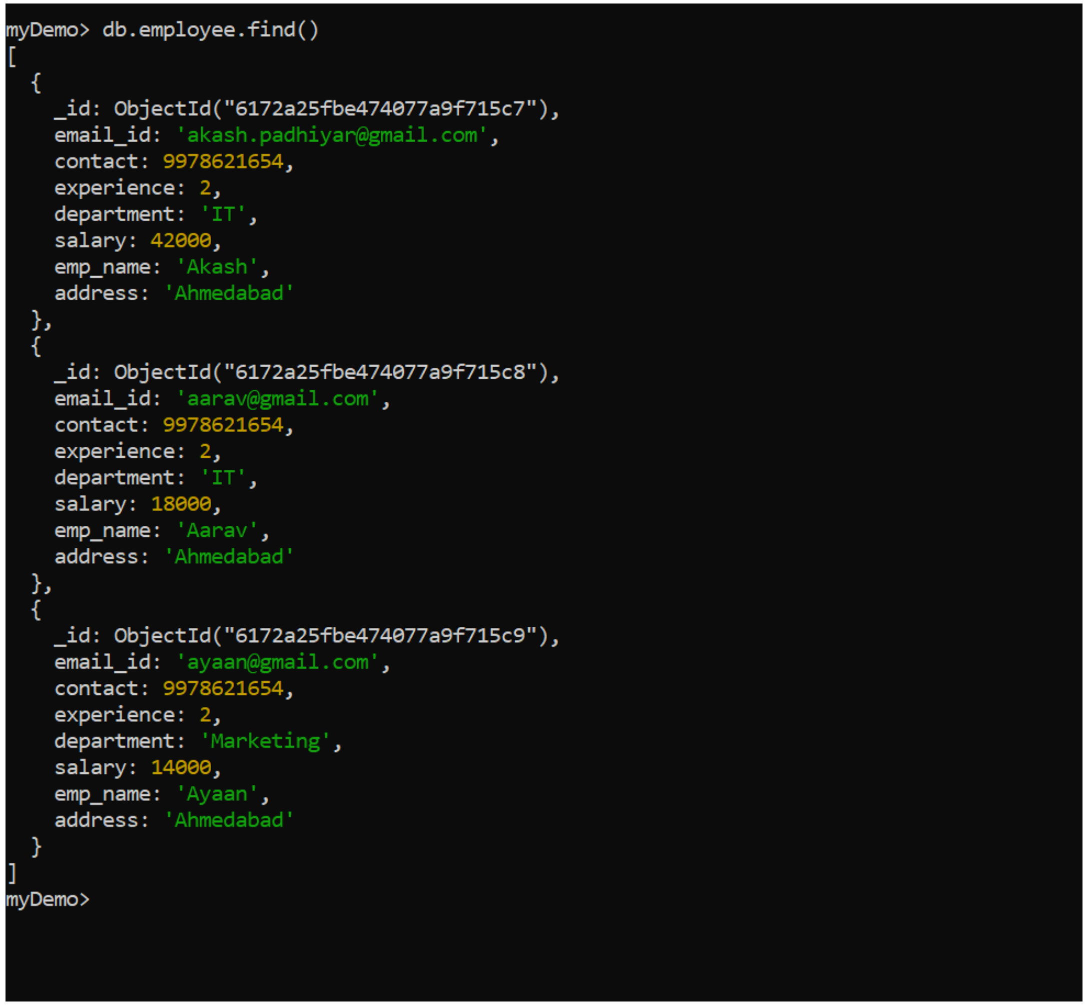
- Syntax :-

db.COLLECTION_NAME.update({},{$set:{“key” : “value”}}, {multi:true})

- By default, MongoDB will update only a single document. To update multiple documents, you need to set a parameter 'multi' to true.

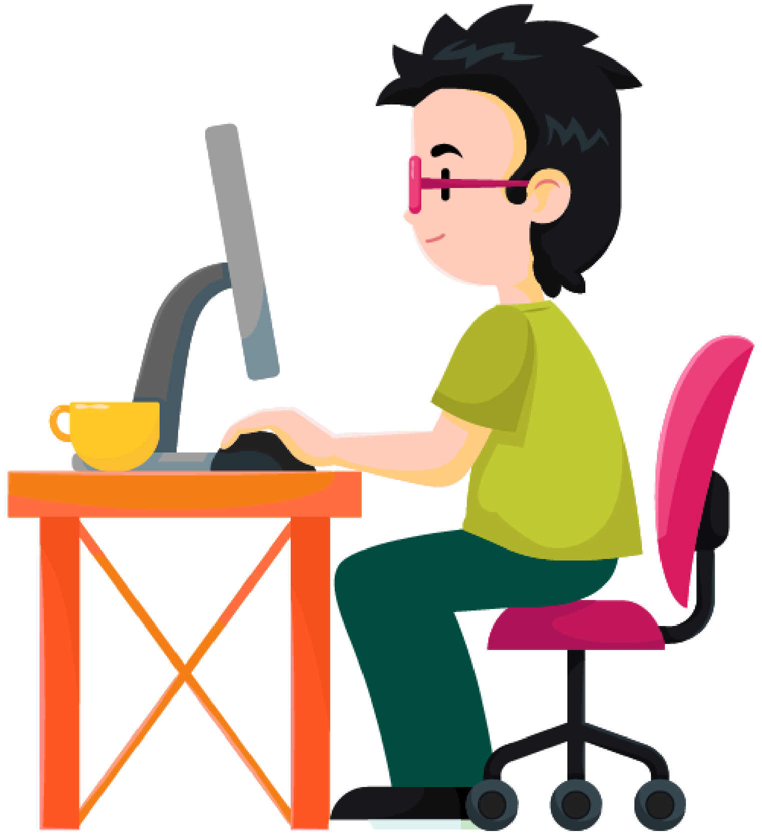


```
myDemo> db.employee.update({},{$set: {address:"Ahmedabad"}}, {multi:true})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0
}
```



```
myDemo> db.employee.find()
[
  {
    _id: ObjectId("6172a25fbe474077a9f715c7"),
    email_id: 'akash.padhiyar@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 42000,
    emp_name: 'Akash',
    address: 'Ahmedabad'
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c8"),
    email_id: 'aarav@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'IT',
    salary: 18000,
    emp_name: 'Aarav',
    address: 'Ahmedabad'
  },
  {
    _id: ObjectId("6172a25fbe474077a9f715c9"),
    email_id: 'ayaan@gmail.com',
    contact: 9978621654,
    experience: 2,
    department: 'Marketing',
    salary: 14000,
    emp_name: 'Ayaan',
    address: 'Ahmedabad'
  }
]
myDemo>
```

Get Exclusive
Video Tutorials



www.aptutorials.com

https://www.youtube.com/user/Akashtips

Get More Details

www.akashsir.com

If You Liked It !

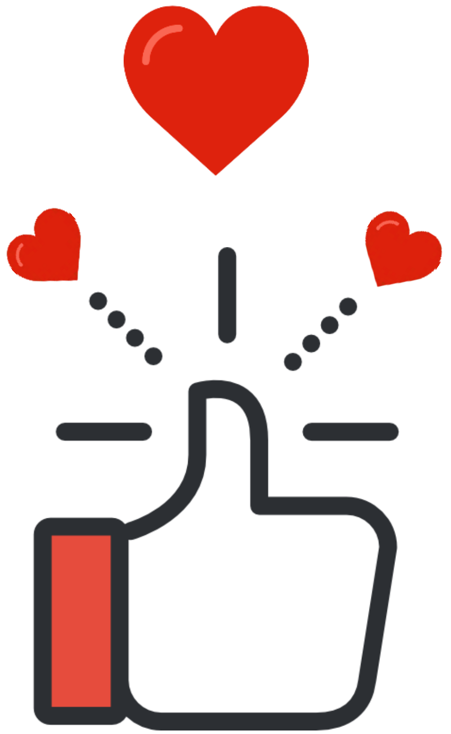
Rating Us Now

Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET

Sulekha

https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad



Connect With Me

Akash Padhiyar
#AkashSir

www.akashsir.com
www.akashtechnolabs.com
www.akashpadhiyar.com
www.aptutorials.com

# Social Info

Akash.padhiyar
Akashpadhiyar
Akash_padhiyar
+91 99786-21654

#Akashpadhiyar
#aptutorials