EX.NO.6 **210701510**

Import a JSON file from the command line. Apply the following actions with the data present in the JSON file where, projection, aggregation, remove, count, limit, skip and sort

AIM:

To import a JSON file from the command line and apply the following actions with the data present in the JSON file where, projection, aggregation, remove, count, limit, skip and sort using MongoDB.

PROCEDURE:

1. Open command prompt and run `mongod` to start the MongoDB server.

2. Then open another command prompt and run `mongosh` to activate MongoDB shell.

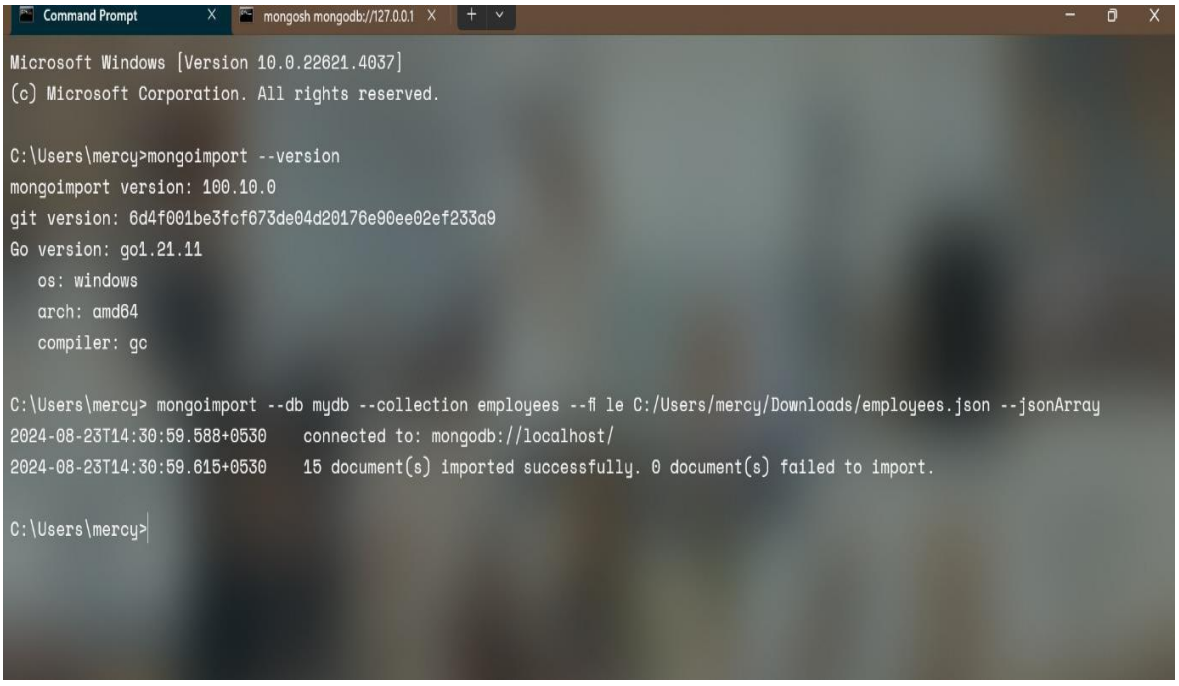
3. Create a database using use <database_name>.

4. To import the JSON file use this command:

```
mongoimport --db --mydb --collection employees --file
C:\Users\mercy\Downloads\employees.json  --jsonArray
```

5. After importing the JSON file perform specific commands for projection, aggregation, remove, count, limit and sort.

OUTPUT:



```
test> use mydb
switched to db mydb
mydb> db.employees.fi nd({}, {name: 1, salary: 1, _id: 0})
[
  { name: 'David Brown', salary: 85000 },
  { name: 'Mia White', salary: 98000 },
  { name: 'Nate Harris', salary: 75000 },
  { name: 'Olivia Clark', salary: 82000 },
  { name: 'Ivy Taylor', salary: 83000 },
  { name: 'Jack Anderson', salary: 72000 },
  { name: 'Kara Thomas', salary: 91000 },
  { name: 'Leo Jackson', salary: 94000 },
  { name: 'Frank Miller', salary: 99000 },
  { name: 'Bob Johnson', salary: 95000 },
  { name: 'Eva Davis', salary: 92000 },
  { name: 'Grace Wilson', salary: 78000 },
  { name: 'Alice Smith', salary: 90000 },
  { name: 'Henry Moore', salary: 87000 },
  { name: 'Carol Williams', salary: 105000 }
]
```

```
mydb> db.employees.aggregate([
...     { $group: { _id: "$department", totalEmployees: { $sum: 1 } } }
... ])
[
  { _id: 'Content', totalEmployees: 1 },
  { _id: 'Data', totalEmployees: 1 },
  { _id: 'IT', totalEmployees: 1 },
  { _id: 'Marketing', totalEmployees: 1 },
  { _id: 'HR', totalEmployees: 1 },
  { _id: 'Support', totalEmployees: 1 },
  { _id: 'Finance', totalEmployees: 1 },
  { _id: 'Engineering', totalEmployees: 3 },
  { _id: 'Design', totalEmployees: 2 },
  { _id: 'Business', totalEmployees: 1 },
  { _id: 'Product', totalEmployees: 1 },
  { _id: 'Sales', totalEmployees: 1 }
]
```

```
mydb> db.employees.remove({ salary: { $gt: 100000 } })
DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, fi ndOneAndDelete, or bulkWrite.
{ acknowledged: true, deletedCount: 1 }
```

```
mydb> db.employees.count({ department: "Engineering" })
DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.
3
```

```
mydb> db.employees.fi nd().limit(3)
[
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb694b'),
    employee_id: 4,
    name: 'David Brown',
    position: 'UX Designer',
    department: 'Design',
    salary: 85000
  },
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb694c'),
    employee_id: 13,
    name: 'Mia White',
    position: 'Sales Manager',
    department: 'Sales',
    salary: 98000
  },
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb694d'),
    employee_id: 14,
    name: 'Nate Harris',
    position: 'Customer Support',
    department: 'Support',
    salary: 75000
  }
]
```

```
mydb> db.employees.fi nd().skip(12)
[
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb6957'),
    employee_id: 1,
    name: 'Alice Smith',
    position: 'Software Engineer',
    department: 'Engineering',
    salary: 90000
  },
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb6958'),
    employee_id: 8,
    name: 'Henry Moore',
    position: 'Finance Analyst',
    department: 'Finance',
    salary: 87000
  }
]
```

```
mydb> db.employees.fi nd().sort({ salary: 1 })
[
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb6950'),
    employee_id: 10,
    name: 'Jack Anderson',
    position: 'Content Writer',
    department: 'Content',
    salary: 72000
  },
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb694d'),
    employee_id: 14,
    name: 'Nate Harris',
    position: 'Customer Support',
    department: 'Support',
    salary: 75000
  },
  {
    _id: ObjectId('66c84fcb1b3a03f1f1fb6956'),
    employee_id: 7,
    name: 'Grace Wilson',
    position: 'HR Specialist',
    department: 'HR',
    salary: 78000
  },
```

RESULT:

Thus to import a JSON file from the command line and apply the following actions with the data present in the JSON file where, projection, aggregation, remove, count, limit, skip and sort using MongoDB is completed successfully.