Name: Prerna Sunil Jadhav

Sap Id: 60004220127

Batch : C2-2

Course: Big Data Infrastructure Laboratory

Course Code: DJ19CEEL6011

## EXPERIMENT 09

AIM: Perform CRUD operations using MongoDB.

THEORY: MongoDB is an open source document oriented database. The term 'NOSQL' means Non-relational. Mongo DB isnt based on the table like relational database structure but provides an different mechanism for storage and relation of data. This format of storage is called BSON. CRUD operations include create, read, update and delete documents.

Create operation: Add new document to a collection.

```
db.user.insert one (
    {
        name: "prema",
        age: 20,
        status: "pending"
    }
)
```

Read operations: Retrieve document from a collection.

```
db. user. find (
    {
        age : { $gt : 18}} ,
    }
        nome : 1 ,
        address : 1
    }) : limit (5)
```

Update operation: Modify existing documents in a collection.

```
db. user. update many (
    {
        age: { $ 1 p } },
        { Set : { status : "reject "}}
    )
```

Delete operation: Remove documents in a collection.

```
db. users. deleteMany (
    { status : "reject" }
)
```

CONCLUSION: Thus, we are performed and studied MongoDB and its operations.

**Academic Year: 2022-2023**

| Name: | Prerna Sunil Jadhav |
|---|---|
| Sap Id: | 60004220127 |
| Class: | T. Y. B. Tech (Computer Engineering) |
| Course: | Big Data Infrastructure Laboratory |
| Course Code: | DJ19CEEL6011 |
| Experiment No.: | 09 |

**AIM:** Perform CRUD operation on MongoDB

---

script.js      Prerna Jadhav-60004220127 ✏️     NEW   MONGODB ⌄   RUN ▶ ⋮ ⌞⌝

```
1  show dbs;
```

STDIN

Input for the program ( Optional )

Output:

```
ABHI           0.000GB
Restaurants    0.000GB
Students       0.000GB
admin          0.001GB
aman           0.000GB
blog           0.000GB
case           0.000GB
config         0.000GB
cse            0.000GB
customdb       0.000GB
db             0.000GB
harryKart      0.000GB
inventory      0.000GB
local          0.000GB
products       0.000GB
```

---

script.js      Prerna Jadhav-60004220127 ✏️     NEW   MONGODB ⌄   RUN ▶ ⋮ ⌞⌝

```
1  use admin;
```

STDIN

Input for the program ( Optional )

Output:

```
switched to db admin
```

---

script.js      Prerna Jadhav-60004220127 ✏️     NEW   MONGODB ⌄   RUN ▶ ⋮ ⌞⌝

```
1  use prerna;
```

STDIN

Input for the program ( Optional )

Output:

```
switched to db prerna
```

script.js          Prerna Jadhav-60004220127 ✏          NEW   MONGODB ⌄   RUN ▶   ⋮   ⊡

```
1  db.createCollection("Employee123");
```

STDIN

Input for the program ( Optional )

Output:

{ "ok" : 1 }

---

script.js          Prerna Jadhav-60004220127 ✏          NEW   MONGODB ⌄   RUN ▶   ⋮   ⊡

```
1  db.Employee123.insert(
2    {
3      Eid:101,
4      Ename:"Akshay",
5      Eaddress:"Mumbai",
6      Emobile:1234567890
7    }
8  );
```

STDIN

Input for the program ( Optional )

Output:

WriteResult({ "nInserted" : 1 })

---

script.js          Prerna Jadhav-60004220127 ✏          NEW   MONGODB ⌄   RUN ▶   ⋮   ⊡

```
1  db.Employee123.insertOne(
2    {
3      Eid:102,
4      Ename:"Arvind",
5      Eaddress:"Mumbai",
6      Emobile:9870654321
7    }
8  );
```

STDIN

Input for the program ( Optional )

Output:

```
{
        "acknowledged" : true,
        "insertedId" : ObjectId("660d74188089574
}
```

---

script.js          Prerna Jadhav-60004220127 ✏          NEW   MONGODB ⌄   RUN ▶   ⋮   ⊡

```
1  db.Employee123.insertMany(
2    [
3      {
4        Eid: 103,
5        Ename:"Sahil",
6        Eaddress: "Pune",
7        Emobile: 8765093214
8      },
9      {
10       Eid: 104,
11       Ename: "Pranay",
12       Eaddress: "Bangalore",
13       Emobile: 74310982365
14     }
15   ]
16 );
```

STDIN

Input for the program ( Optional )

Output:

```
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("660d747739101f80b5dcd1
                ObjectId("660d747739101f80b5dcd1
        ]
}
```

```
> db.Employee123.find();
<   {
      _id: ObjectId("66038dcac78b602ec9a44675"),
      Eid: 101,
      Ename: 'Akshay',
      Eaddress: 'Mumbai',
      Emobile: 1234567890
    }
    {
      _id: ObjectId("66038dd8c78b602ec9a44676"),
      Eid: 102,
      Ename: 'Arvind',
      Eaddress: 'Mumbai',
      Emobile: 9870654321
    }
    {
      _id: ObjectId("66038e3dc78b602ec9a44677"),
      Eid: 103,
      Ename: 'Sahil',
      Eaddress: 'Pune',
      Emobile: 8765093214
    }
    {
      _id: ObjectId("66038e3dc78b602ec9a44678"),
      Eid: 104,
      Ename: 'Pranay',
      Eaddress: 'Bangalore',
      Emobile: 74310982365
    }
```

```
> db.Employee123.find({ Eaddress: "Pune" });
<   {
        _id: ObjectId("66038e3dc78b602ec9a44677"),
        Eid: 103,
        Ename: 'Sahil',
        Eaddress: 'Pune',
        Emobile: 8765093214
    }
```

```
> db.Employee123.updateMany(
      { Eaddress: "Mumbai" },
      { $set: { Esalary: 250000 } }
  );
<   {
        acknowledged: true,
        insertedId: null,
        matchedCount: 2,
        modifiedCount: 2,
        upsertedCount: 0
    }
```

```
> db.Employee123.find({ Esalary: { $gt: 150000 } });
<   {
      _id: ObjectId("66038dcac78b602ec9a44675"),
      Eid: 101,
      Ename: 'Akshay',
      Eaddress: 'Mumbai',
      Emobile: 1234567890,
      Esalary: 250000
    }
    {
      _id: ObjectId("66038dd8c78b602ec9a44676"),
      Eid: 102,
      Ename: 'Arvind',
      Eaddress: 'Mumbai',
      Emobile: 9870654321,
      Esalary: 250000
    }
```