

LABORATORY PROGRAM – 1

MongoDB- CRUD Operations Demonstration (Practice and Self Study)

OBSERVATION

COMMAND WITH OUTPUT - USING ATLAS

```
Microsoft Windows [Version 10.0.22631.4890]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student>mongosh "mongodb+srv://cluster0.qh8blz4.mongodb.net/" --apiVersion 1 --username likhithcs22
Enter password: *****
Current Mongosh Log ID: 67c6c754899c67e814fa4213
Connecting to:      mongodb+srv://<credentials>@cluster0.qh8blz4.mongodb.net/?appName=mongosh+2.4.0
Using MongoDB:      8.0.5 (API Version 1)
Using Mongosh:      2.4.0

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

Atlas atlas-2vljb9-shard-0 [primary] test> show dbs
e-commerce 108.00 KiB
myDB        40.00 KiB
admin       232.00 KiB
local       15.70 GiB
Atlas atlas-2vljb9-shard-0 [primary] test> use myDB
switched to db myDB
Atlas atlas-2vljb9-shard-0 [primary] myDB> db
myDB
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.createCollection("Student");
{ ok: 1 }
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.insert({RollNo:1, Age:21, Cont:9876, email:"antara.de9@gmail.com"});
...
db.Student.insert({RollNo:2, Age:22, Cont:9976, email:"anushka.de9@gmail.com"});
...
db.Student.insert({RollNo:3, Age:21, Cont:5576, email:"anubhav.de9@gmail.com"});
...
db.Student.insert({RollNo:4, Age:20, Cont:4476, email:"pani.de9@gmail.com"});
...
db.Student.insert({RollNo:10, Age:23, Cont:2276, email:"rekha.de9@gmail.com"});
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67c6c898899c67e814fa4218') }
}
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.insert({RollNo:1, Age:21, Cont:9876, email:"antara.de9@gmail.com"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67c6c8a3899c67e814fa4219') }
}
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.insert({RollNo:2, Age:22, Cont:9976, email:"anushka.de9@gmail.com"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67c6c8f7899c67e814fa421a') }
}
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.insert({RollNo:3, Age:21, Cont:5576, email:"anubhav.de9@gmail.com"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67c6c8fb899c67e814fa421b') }
}

C:\Users\likhi>mongosh "mongodb+srv://cluster0.qh8blz4.mongodb.net/" --apiVersion 1 --username likhithcs22
Enter password: *****
Current Mongosh Log ID: 6833148466c722794490defd
Connecting to:      mongodb+srv://<credentials>@cluster0.qh8blz4.mongodb.net/?appName=mongosh+2.2.9
Using MongoDB:      8.0.9 (API Version 1)
Using Mongosh:      2.2.9
mongosh 2.5.1 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Atlas atlas-2vljb9-shard-0 [primary] test> show dbs
e-commerce 108.00 KiB
myDB        72.00 KiB
admin       312.00 KiB
local       64.34 GiB
Atlas atlas-2vljb9-shard-0 [primary] test> use myDB
switched to db myDB
Atlas atlas-2vljb9-shard-0 [primary] myDB> db
myDB
Atlas atlas-2vljb9-shard-0 [primary] myDB> show collections
Student
```

```

Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.find(
[
  {
    _id: ObjectId('67c6c898899c67e814fa4214'),
    RollNo: 1,
    Age: 21,
    Cont: 9876,
    email: 'antara.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c898899c67e814fa4215'),
    RollNo: 2,
    Age: 22,
    Cont: 9976,
    email: 'anushka.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c898899c67e814fa4216'),
    RollNo: 3,
    Age: 21,
    Cont: 5576,
    email: 'anubhav.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c898899c67e814fa4217'),
    RollNo: 4,
    Age: 20,
    Cont: 4476,
    email: 'pani.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c898899c67e814fa4218'),
    RollNo: 10,
    Age: 23,
    Cont: 2276,
    email: 'Abhinav@gmail.com'
  },
  {
    _id: ObjectId('67c6c8a3899c67e814fa4219'),
    RollNo: 1,
    Age: 21,
    Cont: 9876,
    email: 'antara.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c8f7899c67e814fa421a'),
    RollNo: 2,
    Age: 22,
    Cont: 9976,
    email: 'anushka.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c8fb899c67e814fa421b'),
    RollNo: 3,
    Age: 21,
    Cont: 5576,
    email: 'anubhav.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c8fd899c67e814fa421c'),
    RollNo: 4,
    Age: 20,
    Cont: 4476,
    email: 'pani.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6c904899c67e814fa421d'),
    RollNo: 10,
    Age: 23,
    Cont: 2276,
    email: 'rekha.de9@gmail.com'
  },
  {
    _id: ObjectId('67c6ca34899c67e814fa421e'),
    RollNo: 11,
    Age: 22,
    Name: 'FEM',
    Cont: 2276,
    email: 'rea.de9@gmail.com'
  }
]

```

```

Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.updateOne({"RollNo": 10}, {$set: {"email": "john.deo@gmail.com"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-2vljb9-shard-0 [primary] myDB> db.Student.find(
... {"Name": /^F/}
... )
[
  {
    _id: ObjectId('67c6ca34899c67e814fa421e'),
    RollNo: 11,
    Age: 22,
    Name: 'FEM',
    Cont: 2276,
    email: 'rea.de9@gmail.com'
  }
]
Atlas atlas-2vljb9-shard-0 [primary] myDB> |

```

MongoDB- CRUD Operations Demonstration (Practice and Self Study)

COMMAND WITH OUTPUT - USING UBUNTU TERMINAL

```

MyDataBase> use MyDataBase
already on db MyDataBase
MyDataBase> show collections
Customers
NewStudent
Student
MyDataBase> db.Student.find();
[
  {
    _id: 1,
    studName: 'Michellejacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  { _id: 3, Grade: 'VII', StudName: 'AryanDavid', Hobbies: 'Skating' },
  { _id: 2, Grade: 'VIII', StudName: 'Ram', Hobbies: 'Learning' }
]

```

```

test> use MyDataBase
switched to db MyDataBase
MyDataBase> show collections
NewStudent
NewStudent2
Student
MyDataBase> db.NewStudent2.drop();
true
MyDataBase> db.createCollection("Customers");
{ ok: 1 }
MyDataBase> db.Customers.insertMany([{cust_id:1,Balance:200, Type:"S"},]);
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d00571207666297fa3b81a') }
}
MyDataBase> db.Customers.insert({cust_id:1,Balance:1000, Type:"Z"})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d0058f207666297fa3b81b') }
}
MyDataBase> db.Customers.insert({cust_id:2,Balance:100, Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d0059c207666297fa3b81c') }
}
MyDataBase> db.Customers.insert({cust_id:2,Balance:1000, Type:"C"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d005a5207666297fa3b81d') }
}
MyDataBase> db.Customers.insert({cust_id:2,Balance:500, Type:"C"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d005ad207666297fa3b81e') }
}
MyDataBase> db.Customers.insert({cust_id:2,Balance:50, Type:"S"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d005b2207666297fa3b81f') }
}
MyDataBase> db.Customers.insert({cust_id:3,Balance:500, Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId('67d005ba207666297fa3b820') }
}

```

```

MyDataBase> db.Customers.aggregate([
...   {
...     $group: {
...       _id: "$cust_id",           // Group by cust_id
...       minAccBal: { $min: "$Balance" }, // Find the minimum Balance
...       maxAccBal: { $max: "$Balance" } // Find the maximum Balance
...     }
...   }
... ]);
[
  { _id: 3, minAccBal: 500, maxAccBal: 500 },
  { _id: 2, minAccBal: 50, maxAccBal: 1000 },
  { _id: 1, minAccBal: 200, maxAccBal: 1000 }
]

```

```

MyDataBase> db.Customers.aggregate([
...   { $match: { Type: "Z" } },
...   { $group: { _id: "$cust_id", TotAccBal: { $sum: "$Balance" } } },
...   { $match: { TotAccBal: { $gt: 1200 } } }
... ]);

```

```
MyDataBase> db.Customers.aggregate([
...   { $match: { Type: "Z" } },
...   {
...     $group: {
...       _id: "$cust_id",
...       TotAccBal: { $sum: "$Balance" }
...     }
...   },
...   {
...     $match: {
...       TotAccBal: { $gt: 200 }
...     }
...   }
... ]);
[ { _id: 3, TotAccBal: 500 }, { _id: 1, TotAccBal: 1000 } ]
```

```
bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$ mongoimport --host localhost --db MyDataBase --collection NewStudent2 --type=csv --file /home/bmscecse/Desktop/135.txt --headerline
2025-03-11T14:55:05.192+0530   connected to: mongodb://localhost/
2025-03-11T14:55:05.360+0530   3 document(s) imported successfully. 0 document(s) failed to import.
bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$ mongoexport --host localhost --db MyDataBase --collection NewStudent2 --type=json --file /home/bmscecse/Desktop/135.txt
2025-03-11T14:55:24.438+0530   error parsing command line options: unknown option "file"
2025-03-11T14:55:24.438+0530   try 'mongoexport --help' for more information
bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$ mongoexport --host localhost --db MyDataBase --collection NewStudent2 --type=json --out /home/bmscecse/Desktop/135.txt
2025-03-11T14:55:32.771+0530   connected to: mongodb://localhost/
2025-03-11T14:55:32.780+0530   exported 3 records
bmscecse@bmscecse-HP-Elite-Tower-800-G9-Desktop-PC:~$
```