LAB 1

MongoDB- CRUD Demonstration

SETUP:

```
C:\Users\student>

C:\Users\student>mongosh "mongodb+srv://cluster0.ddhftxd.mongodb.net/" --apiVersion 1 --username shravanics21
Enter password: *************
Current Mongosh Log ID: 660a82917c840f42b4a0552f
Connecting to:          mongodb+srv://<credentials>@cluster0.ddhftxd.mongodb.net/?appName=mongosh+2.0.0
Using MongoDB:          7.0.7 (API Version 1)
Using Mongosh:          2.0.0
mongosh 2.2.2 is available for download: https://www.mongodb.com/try/download/shell

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

1. Create a database “Student” with the following attributes Rollno, Age, ContactNo, Email-Id.

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.createCollection("Student");
{ ok: 1 }
```

2. Insert appropriate values(at least 5)

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:1,Age:21,Cont:9876,email:"antara.de9@gmail.com"});
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a82ec7c840f42b4a05530") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:2,Age:22,Cont:9976,email:"anushka.de9@gmail.com"});

{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a82ed7c840f42b4a05531") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:3,Age:21,Cont:5576,email:"anubhav.de9@gmail.com"});

{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a82ed7c840f42b4a05532") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:4,Age:20,Cont:4476,email:"pani.de9@gmail.com"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a82ed7c840f42b4a05533") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:10,Age:23,Cont:2276,email:"rekha.de9@gmail.com"});
{
  acknowledged: true,
```

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.insert({RollNo:10,Age:23,Cont:2276,email:"rekha.de9@gmail.com"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a82f47c840f42b4a05534") }
}
```

3. View the data

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.find()
[
  {
    _id: ObjectId("660a82ec7c840f42b4a05530"),
    RollNo: 1,
    Age: 21,
    Cont: 9876,
    email: 'antara.de9@gmail.com'
  },
  {
    _id: ObjectId("660a82ed7c840f42b4a05531"),
    RollNo: 2,
    Age: 22,
    Cont: 9976,
    email: 'anushka.de9@gmail.com'
  },
  {
    _id: ObjectId("660a82ed7c840f42b4a05532"),
    RollNo: 3,
    Age: 21,
    Cont: 5576,
    email: 'anubhav.de9@gmail.com'
  },
  {
    _id: ObjectId("660a82ed7c840f42b4a05533"),
    RollNo: 4,
    Age: 20,
    Cont: 4476,
    email: 'pani.de9@gmail.com'
  },
  {
    _id: ObjectId("660a82f47c840f42b4a05534"),
    RollNo: 10,
    Age: 23,
    Cont: 2276,
    email: 'rekha.de9@gmail.com'
  }
]
```

4. Write a query to update the Email-Id of a student with rollno 10.

```
  },
  {
    _id: ObjectId("660a83337c840f42b4a05535"),
    RollNo: 11,
    Age: 22,
    Name: 'ABC',
    Cont: 2276,
    email: 'rea.de9@gmail.com'
  }
]
```

5. Replace the student name from “ABC” to “FEM” of rollno 11.

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.update({RollNo:11,Name:"ABC"},{$set:{Name:"FEM"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

6. Drop the table

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Student.drop();
true
```

1. Create a collection by name Customers with the following attributes.
   Cust_id, Acc_Bal, Acc_Type

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.createCollection("Customers");
{ ok: 1 }
```

2. Insert at least 5 values into the table

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:1,Balance:200, Type:"S"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b47c840f42b4a05536") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:1,Balance:1000, Type:"Z"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b47c840f42b4a05537") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:2,Balance:100, Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b47c840f42b4a05538") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:2,Balance:1000, Type:"C"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b57c840f42b4a05539") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:2,Balance:500, Type:"C"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b57c840f42b4a0553a") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:2,Balance:50, Type:"S"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b57c840f42b4a0553b") }
}
Atlas atlas-b6pfyk-shard-0 [primary] test>

Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:3,Balance:500, Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b77c840f42b4a0553c") }
}
```

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.insert({cust_id:2,Balance:50, Type:"S"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660a83b57c840f42b4a0553b") }
}
```

3. Write a query to display those records whose total account balance is greater than 1200 of account type ‘Z’ for each customer_id.

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.aggregate (
... {$match:{Type:"Z"}},
...
... {$group : { _id : "$cust_id",
...
... TotAccBal :{$sum:"$Balance"} } },
... {$match:{TotAccBal:{$gt:1200}}});
```

4. Determine Minimum and Maximum account balance for each customer_id.

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.aggregate (
...
... {$group : { _id : "$cust_id",
...
... minAccBal :{$min:"$Balance"},
... maxAccBal :{$max:"$Balance"} }});
[
  { _id: 2, minAccBal: 50, maxAccBal: 1000 },
  { _id: 1, minAccBal: 200, maxAccBal: 1000 },
  { _id: 3, minAccBal: 500, maxAccBal: 500 }
]
```

5. Drop the table

```
Atlas atlas-b6pfyk-shard-0 [primary] test> db.Customers.drop()
true
```