

Mithibai College

Department of Computer Science

M.Sc. (Data Science and AI)

Practical 3: Performing CRUD Operations in MongoDB

Date:- 10/01/2025

Submission Date:- 17/01/2025

Creating a New Database:

```
test> show dbs;
admin    40.00 KiB
config   72.00 KiB
local    80.00 KiB
test> use userdb;
switched to db userdb
userdb> |
```

Creating a New Collection:

```
userdb> db.createCollection("users");
{ ok: 1 }
userdb> |
```

Create Operation

There are two ways to create new documents to a collection in MongoDB:

1. insertOne():

```
userdb> db.users.insertOne({
... name: "Nida",
... age: 22});
{
  acknowledged: true,
  insertedId: ObjectId('6780ec6a9566f97a7376b543')
}
userdb> |
```

2. insertMany()

```

userdb> db.users.insertMany([
... {
...   name: "Aamina",
...   age: 21
... },
... {
...   name: "Firdaus",
...   age: 22
... },
... {
...   name: "Sushmita",
...   age: 21
... }
... ]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6780ed459566f97a7376b544'),
    '1': ObjectId('6780ed459566f97a7376b545'),
    '2': ObjectId('6780ed459566f97a7376b546')
  }
}
userdb> |

```

Read Operations

1. find()

```

userdb> db.users.find();
[
  { _id: ObjectId('6780ec6a9566f97a7376b543'), name: 'Nida', age: 22 },
  {
    _id: ObjectId('6780ed459566f97a7376b544'),
    name: 'Aamina',
    age: 21
  },
  {
    _id: ObjectId('6780ed459566f97a7376b545'),
    name: 'Firdaus',
    age: 22
  },
  {
    _id: ObjectId('6780ed459566f97a7376b546'),
    name: 'Sushmita',
    age: 21
  }
]
userdb> |

```

Adding filters to find()

```
userdb> db.users.find({ age: {$gt : 21}}, {name:1, age:1});
[
  { _id: ObjectId('6780ec6a9566f97a7376b543'), name: 'Nida', age: 22 },
  {
    _id: ObjectId('6780ed459566f97a7376b545'),
    name: 'Firdaus',
    age: 22
  }
]
```

2. findOne()

```
userdb> db.users.findOne({ name: 'Nida'});
{ _id: ObjectId('6780ec6a9566f97a7376b543'), name: 'Nida', age: 22 }
userdb> |
```

Update Operations

There are several ways to perform an update operation, including the following:

1. updateOne()

```
userdb> db.users.updateOne( { name: 'Nida' }, {$set: {email: 'khannida488@gmail.com'}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
userdb> |
```

2. updateMany()

```
userdb> db.users.updateMany({age: {$lt: 23}}, {$set: {status: "active"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 4,
  modifiedCount: 4,
  upsertedCount: 0
}
```

Output:

```

userdb> db.users.find({age: {$eq: 22}});
[
  {
    _id: ObjectId('6780ec6a9566f97a7376b543'),
    name: 'Nida',
    age: 22,
    email: 'khannida488@gmail.com',
    status: 'active'
  },
  {
    _id: ObjectId('6780ed459566f97a7376b545'),
    name: 'Firdaus',
    age: 22,
    status: 'active'
  }
]

```

Delete Operations

There are several ways to perform a delete operation, including the following:

1. deleteOne()

```

userdb> db.users.deleteOne({name: 'Nida'});
{ acknowledged: true, deletedCount: 1 }
userdb> db.users.find();
[
  {
    _id: ObjectId('6780ed459566f97a7376b544'),
    name: 'Aamina',
    age: 21,
    status: 'active'
  },
  {
    _id: ObjectId('6780ed459566f97a7376b545'),
    name: 'Firdaus',
    age: 22,
    status: 'active'
  },
  {
    _id: ObjectId('6780ed459566f97a7376b546'),
    name: 'Sushmita',
    age: 21,
    status: 'active'
  }
]
userdb> |

```

2. deleteMany()

```
userdb> db.users.deleteMany({age: {$lt: 22}});
{ acknowledged: true, deletedCount: 2 }
userdb> db.users.find();
[
  {
    _id: ObjectId('6780ed459566f97a7376b545'),
    name: 'Firdaus',
    age: 22,
    status: 'active'
  }
]
userdb> |
```

3. drop()

```
userdb> db.users.drop();
true
userdb> show collections;

userdb> |
```