Name: Bryce Ferreira
Roll No.: 2203129
En No: MITU20BTCS0076
Class: TY CSE CORE 3 Batch A

Exp - 5

Part A

Practical Objective:

i) To learn how to insert collections in MongoDB
ii) To learn how to read collections in MongoDB
iii) To learn how to update collections in MongoDB
iv) To learn how to delete collections in MongoDB

Prerequisite: Understanding of basic CRUD

Software: Mongodb

CO Mapping:

CO2: Query a database using SQL, PL/SQL and NoSQL commands..

Practical Outcomes: At the end of this practical student will be able to:
Perform own database and collection in NoSQL language

Theory:

1. **show dbs**
2. **use <db name>**
3. **show <collection Name>**
4. MongoDB provides the following methods to **insert** documents into a collection:

- `db.collection.insertOne({})`
- `db.collection.insertMany([{},{},{}])`
- db.collection.find().pretty()

5. MongoDB provides the following methods to **read** documents into a collection

- `db.collection.find()`
- db.collection.find().pretty()
- db.collection.find(query,projection)
- db.collection.find({firstname:"abc"},{name:1}).pretty()
- db.collection.find({firstname:"abc"},{ _id:0}).pretty()

6. MongoDB provides the following methods to **update** documents into a collection

- `db.collection.updateOne(<filter>, <update>, <options>)`
- `db.collection.updateMany(<filter>, <update>, <options>)`
- `db.collection.replaceOne(<filter>, <update>, <options>)`
- db.collection.updateOne({firstname:"abc"},{$set:{lastname:"xyz"} })

7. MongoDB provides the following methods to **update** documents into a

collection

- `db.collection.deleteMany()`
- `db.collection.deleteOne()`

db.collection.deleteOne({firstname:"abc"})

Procedure:

1. Formulate the query for given problem.
2. Write the NOSQL query with proper input.
3. Execute the query.

Practice Exercise:

1. **Create database(user) and their collections with the following data**

   Name JS
   Type Frontend
   Videos 76
   Active true

   Name MongoDB
   Type Database
   Videos 5
   Active true

   Name NodeJS
   Type backend
   Videos 52
   Active true

   Name JavaScript
   Type Frontend
   Videos 176
   Active true

2. **Find the result of given collection**
3. **Show the result in pretty format**
4. **Get the output as "mongodb"**
5. **Get the output as "mongodb"with only one name field**
6. **Get Mongodb data without ID field**
7. **Update the javascript file to FULL STACK**
8. **Update all the fields with the type value equal to "front end" and set the values of status to false**
9. **Delete the field with the type matches=Full stack**
10. **Delete all collections**

Instructions:

1. Write and execute the in MONGODB.
2. Paste the snapshot of the output in input & output section.

Part B

Code and Output:

1.

```
test> show dbs
admin    40.00 KiB
config  108.00 KiB
local    72.00 KiB
test> use user
switched to db user
user> db.collection.insertMany(
... [
...     {"Name":"JS","Type":"Frontend","Videos":76,"Active":true},
...     {"Name":"MongoDB","Type":"Database","Videos":5,"Active":true},
...     {"Name":"NodeJS","Type":"Backend","Videos":52,"Active":true},
...     {"Name":"JavaScript","Type":"Frontend","Videos":176,"Active":true}
... ]
... )
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6349053c4f5b7d3ecede067d"),
    '1': ObjectId("6349053c4f5b7d3ecede067e"),
    '2': ObjectId("6349053c4f5b7d3ecede067f"),
    '3': ObjectId("6349053c4f5b7d3ecede0680")
  }
}
```

2.

```
user> db.collection.find()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067d"),
    Name: 'JS',
    Type: 'Frontend',
    Videos: 76,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067f"),
    Name: 'NodeJS',
    Type: 'Backend',
    Videos: 52,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede0680"),
    Name: 'JavaScript',
    Type: 'Frontend',
user>
    Active: true
  }
]
```

3.

```
user> db.collection.find().pretty()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067d"),
    Name: 'JS',
    Type: 'Frontend',
    Videos: 76,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067f"),
    Name: 'NodeJS',
    Type: 'Backend',
    Videos: 52,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede0680"),
    Name: 'JavaScript',
    Type: 'Frontend',
    Videos: 176,
    Active: true
  }
]
```

4.

```
user> db.collection.find({Name:"MongoDB"}).pretty()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  }
]
```

5.

```
user> db.collection.find({Name:"MongoDB"},{Name:1}).pretty()
[ { _id: ObjectId("6349053c4f5b7d3ecede067e"), Name: 'MongoDB' } ]
```

6.

```
user> db.collection.find({Name:"MongoDB"},{_id:0}).pretty()
[ { Name: 'MongoDB', Type: 'Database', Videos: 5, Active: true } ]
```

7.

```
user> db.collection.updateOne(
... {Name:"JavaScript"},
... {
...     $set: {Name:"FULL STACK"}
... }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
user> db.collection.find().pretty()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067d"),
    Name: 'JS',
    Type: 'Frontend',
    Videos: 76,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  },
  {
user>
    Name: 'NodeJS',
    Type: 'Backend',
    Videos: 52,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede0680"),
    Name: 'FULL STACK',
    Type: 'Frontend',
    Videos: 176,
    Active: true
  }
]
```

8.

```
user> db.collection.updateMany( { Type: "Frontend" }, { $set: { Active: false } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 2,
  modifiedCount: 1,
  upsertedCount: 0
}
user> db.collection.find().pretty()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067d"),
    Name: 'JS',
    Type: 'Frontend',
    Videos: 76,
    Active: false
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067f"),
    Name: 'NodeJS',
    Type: 'Backend',
    Videos: 52,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede0680"),
    Name: 'FULL STACK',
    Type: 'Frontend',
    Videos: 176,
    Active: false
  }
]
```

9.

```
user> db.collection.deleteOne({Name:"FULL STACK"})
{ acknowledged: true, deletedCount: 1 }
user> db.collection.find().pretty()
[
  {
    _id: ObjectId("6349053c4f5b7d3ecede067d"),
    Name: 'JS',
    Type: 'Frontend',
    Videos: 76,
    Active: false
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067e"),
    Name: 'MongoDB',
    Type: 'Database',
    Videos: 5,
    Active: true
  },
  {
    _id: ObjectId("6349053c4f5b7d3ecede067f"),
    Name: 'NodeJS',
    Type: 'Backend',
    Videos: 52,
    Active: true
  }
]
```

10.

```
user> db.collection.drop()
true
user> db.collection.find().pretty()

user>
```

Observation & Learning:
Write your observation and learning after performing the task.

Conclusion:
Write statement of conclusion here.