MongoDB Operations Documentation

- This document demonstrates various MongoDB operations with practical examples.
- All queries are executed using the database: trainingdb.
- Collection used: people.
- Each operation includes the command, explanation, and space for a screenshot.
- The objective is to understand CRUD operations, projections, replacements, and operators in MongoDB.

---

1. Update One Document

Query:

db.people.updateOne(

{ user_id: 3 }, { $set: { status: "failed" }}

)

Screenshot:

Explanation:

This command modifies just one document where the user_id is 3. It uses the $set operator to assign the value "failed" to the status field.

---

2. Update Multiple Document

Query:

db.people.updateMany(

{ status: "failed" }, { $inc: { age: 2 } }

)

Screenshot:

```
trainingdb> db.people.updateMany({ status: "failed" }, { $inc: { age: 2 } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
```

Explanation:

This operation increases the age by 2 for all documents that have the status field set to "failed", using the $inc operator to increment values.

---

3. Delete One Document

Query:

db.people.deleteOne(

{ age: 28 }

)

Screenshot:

```
trainingdb> db.people.deleteOne({ age: 28 })
{ acknowledged: true, deletedCount: 1 }
```

Explanation:

Removes the first document it finds in the collection where the age is 28.

---

4. Delete Multiple Documents

Query:

db.people.deleteOne(

{ age: 28 }

)

Screenshot:

```
trainingdb> db.people.deleteOne({ age: 28 })
{ acknowledged: true, deletedCount: 1 }
```

Explanation:

Removes the first document it finds in the collection where the age is 28.

---

5. Replace One Document

Query:

```
db.people.replaceOne(
  { user_id: 2 },
  { user_id: 2, age: 31, status: "passed" }
)
```

Screenshot:

Explanation:

This command replaces the entire document that matches user_id 2 with a new document containing only the provided fields (user_id, age, and status).

---

6. Projection Queries

Query:

```
db.people.find(
{ status: "passed" }, { user_id: 1, _id: 0 }
)
db.people.find(
{ status: "passed" }, { user_id: 1, status: 1, _id: 0 }
)
```

Screenshot:

Explanation:

These queries show how to use projections. The first displays only user_id (excluding _id), while the second displays both user_id and status, again omitting the _id field.

---

6. Query with $in Operator

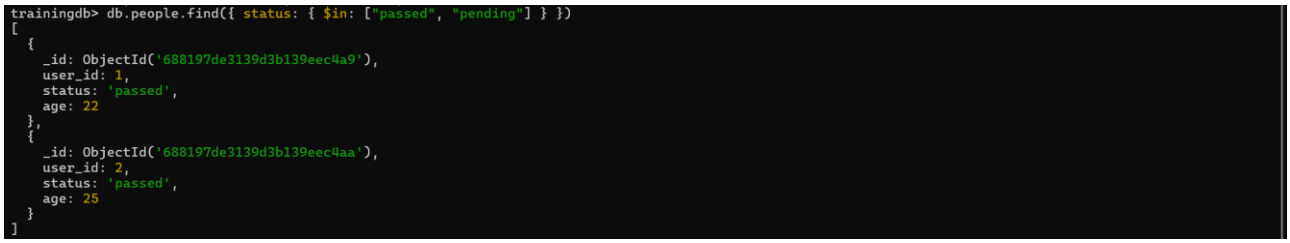
Query:

db.people.find(

{ status: { $in: ["passed", "pending"] } }

)

Screenshot:



Explanation:

This retrieves all documents where the status field is either "passed" or "pending" by using the $in operator to match multiple possible values.

Conclusion:

- This document covered basic to intermediate MongoDB operations.
- CRUD operations were performed using appropriate MongoDB commands.
- Projection, $in operator, and replacement operations were demonstrated.
- Screenshots were included to verify the output of each query.
- The practice enhances understanding of how MongoDB manages data in document-based format.
