

# Task-10 :- CRUD Operations In Document Databases

10/10/25

Aim:- To perform Mongoose using MPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding and removing operations

## STEPS:-

\* Install Mongo-db using following link

<https://www.mongodb.com/tarball/community>

\* Install Mongosh using the below link

<https://www.mongodb.com/docs/mongodb-shell/#download>  
and install - mongosh.

\* To add the MongoDB Shell binary's location to your PATH environment variable :-

Open the control panel

In the System and Security category, click system click Edit, The Edit environment Variable model displays. click New and add the file path to your Mongosh binary

\* open Mongo shell h.0 from c:\Program Files\ mongo DB\server\bin\mongodb.exe

\* Type the CRUD (CREATE READ UPDATE DELETE)

(COMMANDS GIVEN IN TEXT FILE)

CRUD Operations:-

db.createCollection("mylab")

{ "ok": 1 }

- Employee (Employee\_ID, Name, Department\_ID, Job\_Title, Hire\_Date, Salary) ↗
- Department (Department\_ID, Manager\_ID) ↗
- Manager (Manager\_ID, Name)

Given below are three relations. Post a note  
whether there exists a functional dependency between job position & job

EMPLOYEE	DEPARTMENT	MANAGER
P	1	1
Q	2	2
R	3	3
S	4	4

"id": object\_id("627d1598c73940c074e6397d"),

    "item": "Canvas", "qty": 100, "tag":  
    ("cotton"), "size": { "h": 28, "w": 35.5, "cm": 11.5 }

7. db.mylab.insert many. ( { "item": "journal", "qty": 25, "tags": } )

    "blank", "red" }, size: { h: 16, w: 21, cm: "13" }

    { "item": "mat", "qty": 85, "tags": [ "grey" ], "size": { h: 27.9, w: 35.5, cm: 13 } }

    { "item": "mousepad", "qty": 25, "tags": [ "green", "blue" ], "size": { h: 19, w: 22.8, cm: 10 } }

2.

acknowledged": true

"inserted\_ids": [

    ObjectID("627d1598c73940c074e6397d")

    ObjectID("627d1598c73999c074e6397e")

    ObjectID("627d1598c73999c074e6397f")

]

}

7. db.mylab.find( { "item": "Canvas", "qty": 100 } )

{

    "item": "Canvas"

    "qty": 100

}

{

    "id": object\_id("627d1598c73940c074e6397d"),

    "item": "Journal",

    "qty": 25

]

{

";d": objectID("627d1398cf739960c074e6397f"),

```
  "item": "mousepad",
```

```
  "qty": 25
```

```
}  
db.myLab.find({item:"canvas"}).pretty().sort({item:1})
```

```
{
```

```
  "id": objectID("627d1398cf739960c074e6397f");
```

```
  "item": "canvas",
```

```
  "qty": 100
```

```
  "tags": [
```

```
    "cotton",
```

```
  ],
```

```
  "size": "
```

```
  "h": 28,
```

```
  "w": 35.5,
```

```
  " uom": "kg",
```

```
},
```

```
}
```

```
}  
db.myLab.deleteOne({item: "canvas"})
```

```
--  
--  
db.myLab.find({}).limit(1).qty: 1).pretty()
```

```
{
```

```
  "id": objectID("627d1398cf739960c074e6397f");
```

```
  "item": "canvas",
```

```
  "qty": 100
```

```
},
```

```
{
```

- $\text{Employee\_ID} \rightarrow \text{Name, Department, Job\_Title, Manager\_ID, Hire\_Date, Salary}$  (The Employee's ID determines all other attributes).
- $\text{Department} \rightarrow \text{Manager\_ID}$  (The department determines the manager's ID).
- $\text{Manager\_ID} \rightarrow \text{Name}$  (The manager's ID determines the manager's name).

"id": object id ("627d1598c73990c7ue6397d"),

"item": "Journal",

"qty": 25

"id": object id ("627d1598c73990c07ue6397f"),

"item": "mat",

"qty": 25

{ "id": object id ("627d1598c73990c07ue6397f") }

"item": "mousepad", "qty": 25}

Result:- The implementation of CRUD operations like creating, inserting, finding and removing operations using MongoDB is successfully executed

VEL TECH - CSE	
EX NO.	10
PERFORMANCE (5)	10
RESULT AND ANALYSIS (5)	10
EVIDENCE (5)	10
CRED (5)	10
DATE	10/10
	10/10