

taskID=10
Date:- 14/10/2022

CRUD operations in document databases

Aim: To perform mongoose using NPM design on MongoDB designing document database and performing CRUD operations like creating, inserting, querying, finding and removing operations. like creating, inserting, querying, finding and removing operations.

steps:

Step 1: Install mongo db using following link.

<https://www.mongodb.com/try/download/community>.

Step 2: install mangosh using the below link.

<https://www.mongodb.com/doc/mongodb-shell/#download-and-install-mangosh>

Step 3: To add the mongoDB shell binary's location in your PATH environment variable:
open the control panel.

In the system and security category, click system.
click Advanced system settings. The system properties modal displays

click Environment variables

On the system variables section, select path and click edit. The edit environment variable modal displays.

click New and add the file path to your mangosh binary.

click OK to confirm your changes on each other modal.

click OK to confirm your changes.

To confirm that your prob. PATH environment variable is correctly configured to find mangosh, open a command prompt and enter the mangosh -- help command.

If your PATH is configured correctly, a list of valid commands displays.

Step-4 open mongo shell v.0 from c:\program files\mongodb\server\bin\mongod.exe.

Step-5: Type the CRUD (CREATE READ UPDATE DELETE COMMANDS GIVEN IN TEXT FILE).

CRUD OPERATIONS

```
db.createCollection("myLab")
```

{ "ok": 1 }

```
> db.myLab.insertOne({ item: "canvas", qty: 100, tags: ["cotton"], size: { h: 28, w: 35.5, uom: "cm" } })
```

{

"acknowledged": true,

"insertedId": ObjectID("627d13acc73990c074e6397c")

1 insertedId

3

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

```
{ "_id": ObjectID("627d13acc73990c074e6397c"), "item": "canvas", "qty": 100, "tags": ["cotton"], "size": { "h": 28, "w": 35.5, "uom": "cm" } }
```

```
> db.myLab.insertMany([ { item: "journal", qty: 25, tags: ["blank", "red"], size: { h: 14, w: 21, uom: "cm" } }, { item: "mat", qty: 85, tags: ["gray"], size: { h: 27.9, w: 35.5, uom: "cm" } }, { item: "mouse pad", qty: 25, tags: ["gel", "blue"], size: { h: 19, w: 22, uom: "cm" } } ])
```

{

"acknowledged": true,

"insertedIds": [

```
ObjectID("627d1598c73990c074e6397d"),
```

```
ObjectID("627d1598c73990c074e6397c"),
```

```
ObjectID("627d1598c73990c074e6397f") ]
```

```
y db.mylab.find({$y, $item: 1, qty: 1})  
> db.mylab.find({$y, $item: 1, qty: 1})  
{ "id": ObjectId("627d13acc73990c074e6397c"), "item":  
  "canvas", "qty": 100  
}  
{ "id": ObjectId("627d1598c73990c074e6397d"), "item":  
  "journal", "qty": 25  
}  
{ "id": ObjectId("627d1598c73990c074e6397e"), "item":  
  "mat", "qty": 85  
}  
{ "id": ObjectId("627d1598c73990c074e6397f"), "item":  
  "mouse pad", "qty": 25  
> db.mylab.find({$y, $item: 1, qty: 1}).pretty().  
5
```

"_id": ObjectId("627d13acc73990c074e6397c"),
"item": "canvas",
"qty": 100.

y
{ "_id": ObjectId("627d1598c73990c074e6397d"),
 "item": "journal",
 "qty": 25.

y
{ "_id": ObjectId("627d1598c73990c074e6397e"), "item": "mat",
 "qty": 85
}
{ "_id": ObjectId("627d1598c73990c074e6397f"), "item": "mouse pad",
 "qty": 25.

y
db.mylab.find({\$item: "canvas"}).pretty().sort({item: -1})

```
"_id": objectID("607d139acc73990c074e6397c"),
"item": "canvas",
"qty": 100,
"tag": [
"cotton"
],
"size": {
"height": 28,
"width": 35.5,
"unit": "cm"
}
```

y
> db.mylab.deleteOne({item: "journal"}).

= db.mylab.find({y: 1, item: 1, qty: 1}).pretty().

{}
{"_id": objectID("607d139acc73990c074e6397c"),
"item": "canvas",
"qty": 100
}
y
2
{"_id": objectID("607d1598c73990c074e6397d"),
"item": "canvas",
"qty": 100
}

y
2
{"_id": objectID("607d1598c73990c074e6397d"),
"item": "journal",
"qty": 25
}

y
2
{"_id": objectID("607d1598c73990c074e6397e"),
"item": "mat",
"qty": 85
}

y
2
{"_id": objectID("607d1598c73990c074e6397f"),
"item": "mousepad",
"qty": 85
}

thus the implementation of CRUD operations like
creating, inserting, finding and removing operations using
MongoDB is successfully executed.

14/10/25

VELTECH	EX-No.	PERFORMANCE (S)	TEST AND ANALYSIS (S)	VIVA VOCE (S)	RE-ORD (S)	FINAL (20)	WITH DATE
10	15	15	15	15	15	15	15
10	15	15	15	15	15	15	15
10	15	15	15	15	15	15	15
10	15	15	15	15	15	15	15