14/10/25 TASK10 - CRUD operations IN DOCUMENT DATABASES

Aim: To perform Mongoose using NPM design on Mongo DB designing document database and performing CRUD operations like creating, inserting, queging, finding and removing operations.

Steps:

step 1: install Mongo db using following

Step 2: Install mongosh using the below downloded.

step 3: Install mangosh ll.com ll download mangosh
b-s kel.

to add the manosh shell Binary aocation.

step 4: to open mangosh shell as su from
C:" program files.

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

Commands given in the text file.

CRUD operations:

1. Create database
db Create Collection ("products")
{"ok":1}

2. Insert one mobile phone products.
db. mobile phone products. insert one (
{ name: "Real me", price: 30000, categores}
" electronic device", stack :25, tags:
("upp");

details : {brand : "realme" : warranty : year : 3)

output :-

{ " acknowledge ": true ;

"insted id : object Id ("62a1b3acc')

}

3. find one mobile product (lead)

"db - products. find ({name : "realme 3})

Output :

{ "Id" : object ("62a1b3acc") ;

" name : "Realme",

"price " - 30,000" ;

" category " : ' Electronics")

" stock " : 25 ;

"tags : ["oppo"]

4. insert multiple mobile products.

db . mobile product . insert many ({ {

name ! " Smart phone", price : 30,000 , category = "Electronics.

stock : 50, tags : [" android", "7G"], details

[brand : "realme"]) ;

{ name : "redmi", price : 30,000, category : " Electronics"

stock = 30, tags " ("oppo") }

Output :

```
{
  "acknowledge": true
  " insert  ted ids": [
  Object id  ("62a1b5acc")
  object Id  ("b2a1b500c")
  ]
}
```

5) Display selected field

```
db. mobile product . find (), {name : 1 price :1,
                     stock :1})
```

Output :

```
{" id": object id (" 632a1b3acc")"
"name " : "realme"; "price: 25000   "stock : 25)}
{"id": object id ("62a1b5aa c"] : "name". "smart
phone", "price" : 25000. "stock : 30}
{"id": object ("62a1b500c"); "name" "Redmi"; "price"=
                25000"  , "stock : 0}}
```

6) pretty format.

```
db. mobile product. find ({}, {name =1, price =1,
 stock= 1}) . prety()
```

Output:

```
{"Id": object id ("62a1b3acc"), "name": "realme
"price": 30000,   "stock": 25}
{"id"= object id {"62a1b599 c"),"name": "smart phone";
     price : ≥ 30000,   "stock"= 30 }
```

```
{ "id": object id ( "62a1b5a9c"), " name"; "realme";
      "price"; "35000"  "stock": 35}
{ "id" = object id ( "62a1b500c"] . "name": "realmi",
   "price" 25000".  "stock": 30}
```

Delete one mobile products

```
db . mobile product . delete one ({ name:"realme"})
```

Output :

```
{"acknowledge", true" delete count:" }.
```

find display

```
db. mobile product find ({3 {name:1,
price:1, stock:1}) . mobile product
```

output :

```
{ "id": object id (" 62a1b3a cc")"
"name": "realme": " price": 30000 , "stock": 25}
{ "id": object id (" 62a1b399c");
   name": "realme", "price: "28000: "stock" = 31}.
```

VELTECH	
EX No.	10
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	4
RECORD (5)	
TOTAL (20)	14
SIGN WITH DATE	5/10

Result: thus; implementation of CRUD operations like Creating, inserting, finding, and removing operations Using mongoDB B executed successfully.