ASSIGNMENT - B2

31332

TITLE:

Design and develop MongoDB queries using CRUD operations.

PROBLEM STATEMENT :-

Design and develop Mongo DB queries using CRUD operations (use CRUD operations, SAVE method, logical separators)

OBJECTIVE :-

To understand and implement CRUD operations in Mongo DB.

OUTCOME :-

1. To be able to implement command on two tier
2. To be able to implement database in Mongo DB.

S/W & H/W req.

MongoDB, 64 bit Linux/Windows OS
i5 processor

THEORY :-

Mongo DB

It is a cross platform, document oriented database that provides high performance, high availability and easy sealability. Mongo DB works on concept of collection and document. A single Mongo DB server typically has multiple databases.

Collection:

It is a group of Mongo DB documents and is equivalent to RDBMS rule.

Document

A document is a set of key value pairs. Documents have dynamic schema.

CRUD operations.

1. Create

If collection does not currently exist, insert operations wil
create the collection.

a) insert () - It can be used to insert single or array of docume

b) insert_one () - It insets a single document into a collection.

c) insert_many() - It insets multiple documents into a collection.

2. Read

The documents can be retrieved in two ways.

a) find() - It will return with all the documents in collection

b) find-one() - It returns the first document in the

3. update

The documents can be updated in four ways -

a) update() - This will update the matched docs.

b) update-one() - ——"—— first matched documen

c) update-many() - ——"—— all matched documents

d) replace-one() - This performs the same function as update-one()

4. Delete.

The documents can be deleted in two ways.

a) delete-one() - This deletes the first matched docum

b) delete-many() - This deletes all matched documents

* Logical Operators

1. $or

Joins query clauses with logical OR and returns all doc that match either clause

Syntax

{ $ or : [{ <exp1>} , { $OR}] }

2. $and

Joins query clauses with logical AND and returns al documents that match both

Syntax:

```
{ $and : [{ <exp1> } , { <exp2>}] }
```

3. $ not

Inverts the effect of a query expression and returns all documents that do not match the query expression

syntax

```
{ field: { $not: { <op-exp> } } }
```

4. $ nor

Joins query clauses with logical NOR and returns all documents that fail to match both clauses

syntax:-

```
{ $nor : [{ <exp1>} , { <exp2}] }
```

SAVE operator:-

SAVE operator is used to insert or update files If files not present it inserts or else it replaces the document. It has the same syntax as invent

CONCLUSION:-

CRUD operations were understood and implemented.
Logical operators were also implemented.