NIELIT Gorakhpur

Course name:A level   **SUBJECT:DATABASE TECHNOLOGIES**

Topic:MongoDB   **DATE:6/5/2020**

---

1. Batch insert or Inserting Multiple documents into a collection

To Insert multiple documents or Batch insert into a collection **insertMany()** method is available to do this task.

Syntax:

```
db.collection.insertMany (
   [ <document 1> , <document 2>, ... ],
   )
```

Where

- Each of these document should be separated by a **, comma**
- [<document 1> , <document 2>, ...]   is an array of the various s documents which needs to be inserted into the collection.

On Successful execution, it returns a
- A boolean acknowledged as true if the operation is successful or false if uncessful.
- An array of _id for each successfully inserted documents

Example:

```
> db.college.insertMany([{name:"star",course:"python",fees:"14000"},{name:"best",course:"c++",fees:"15000"}])
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("5eb1fcb0d424bc63930ebbff"),
                ObjectId("5eb1fcb0d424bc63930ebc00")
        ]
}
> db.college.find()
{ "_id" : ObjectId("5eb1fa4ccc9559d476ef6e2e"), "name" : "RCC", "course" : "C++" }
{ "_id" : ObjectId("5eb1fcb0d424bc63930ebbff"), "name" : "star", "course" : "python", "fees" : "14000" }
{ "_id" : ObjectId("5eb1fcb0d424bc63930ebc00"), "name" : "best", "course" : "c++", "fees" : "15000" }
```

2. remove() method to remove documents from collection

remove() method is used to remove a document from collection.

Syntax

```
db.COLLECTION_NAME.remove(DELLETION_CRITTERIA, justOne)
```

this method has two parameters One is deletion criteria and second is justOne flag.

- **DELLETION_CRITTERIA** − (Optional) deletion criteria of the documents to be removed. To delete all documents in a collection, pass an empty document ({ }).

- **justOne** – (Optional) if set to true or 1, then remove only one document i.e. the 1st document. Optional. If not specified i,e, the default value of false and it will delete all documents matching the deletion criteria.

Example1: Remove only one document

db.school.remove({"class":"12th"},1)

will remove 1st record of class 12th.

```
>
> db.school.find()
{ "_id" : ObjectId("5eb0f690ffe73fbdd0e2b1e5"), "class" : "10th", "section" : "A sction", "student count" : "39" }
{ "_id" : "01", "class" : "10th", "section" : "D Section", "student count" : "29" }
{ "_id" : "02", "class" : "10th", "section" : "C Section", "student count" : "37" }
{ "_id" : ObjectId("5eb2446ddb4e42dfda38301b"), "class" : "12th", "section" : "A section", "student count" : "33" }
{ "_id" : ObjectId("5eb24479db4e42dfda38301c"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb2448adb4e42dfda38301d"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb244a2db4e42dfda38301e"), "class" : "9th", "section" : "A section", "student count" : "40" }
>
>
> db.school.remove({"class":"12th"},1)
WriteResult({ "nRemoved" : 1 })
> db.school.find()
{ "_id" : ObjectId("5eb0f690ffe73fbdd0e2b1e5"), "class" : "10th", "section" : "A sction", "student count" : "39" }
{ "_id" : "01", "class" : "10th", "section" : "D Section", "student count" : "29" }
{ "_id" : "02", "class" : "10th", "section" : "C Section", "student count" : "37" }
{ "_id" : ObjectId("5eb24479db4e42dfda38301c"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb2448adb4e42dfda38301d"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb244a2db4e42dfda38301e"), "class" : "9th", "section" : "A section", "student count" : "40" }
>
```

Example2:Remove all records of the matching criteria

db.school.remove({"class":"12th"})

will remove all the record of class 12th.

```
> db.school.find()
{ "_id" : ObjectId("5eb0f690ffe73fbdd0e2b1e5"), "class" : "10th", "section" : "A sction", "student count" : "39" }
{ "_id" : "01", "class" : "10th", "section" : "D Section", "student count" : "29" }
{ "_id" : "02", "class" : "10th", "section" : "C Section", "student count" : "37" }
{ "_id" : ObjectId("5eb24479db4e42dfda38301c"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb2448adb4e42dfda38301d"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb244a2db4e42dfda38301e"), "class" : "9th", "section" : "A section", "student count" : "40" }
>
> db.school.remove({"class":"12th"})
WriteResult({ "nRemoved" : 2 })
> db.school.find()
{ "_id" : ObjectId("5eb0f690ffe73fbdd0e2b1e5"), "class" : "10th", "section" : "A sction", "student count" : "39" }
{ "_id" : "01", "class" : "10th", "section" : "D Section", "student count" : "29" }
{ "_id" : "02", "class" : "10th", "section" : "C Section", "student count" : "37" }
{ "_id" : ObjectId("5eb244a2db4e42dfda38301e"), "class" : "9th", "section" : "A section", "student count" : "40" }
>
```

WriteResult({ "nRemoved" : 2 })

Here it has removed all the 2 records of matching criteria.

Example 3: Remove all document if no criteria is specified

db.school.remove({})

will remove all the record from the collection.

WriteResult({ "nRemoved" : 2 })
Here it has removed all the 2 records of matching criteria

```
> db.school.find()
{ "_id" : ObjectId("5eb0f690ffe73fbdd0e2b1e5"), "class" : "10th", "section" : "A sction", "student count" : "39" }
{ "_id" : "01", "class" : "10th", "section" : "D Section", "student count" : "29" }
{ "_id" : "02", "class" : "10th", "section" : "C Section", "student count" : "37" }
{ "_id" : ObjectId("5eb244a2db4e42dfda38301e"), "class" : "9th", "section" : "A section", "student count" : "40" }
>
>
> db.school.remove({ })
WriteResult({ "nRemoved" : 4 })
> db.school.find()
>
```

WriteResult({ "nRemoved" : 4 })
> Here it has removed all the 4 records from collection and now the collection is empty.

3. update() Method : Updating the values in document

The update() method is used to update the value in the existing document of specified collection.

Syntax

> db.COLLECTION_NAME.update(SELECTION_CRITERIA, UPDATED_DATA)

where

> SELECTION_CRITERIA -- The selection criteria for the updating the document
> UPDATED_DATA -- The modifications to apply to the document. It may be simply given with
> $set: parameter.

Example 1

> **db.school.update({"class":"10th"}, {$set:{class:"11th"}})**
>
> after this command, record with class 10th will be updated to class 11th

```
>
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "10th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "9th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "9th", "section" : "B section", "student count" : "39" }
>
> db.school.update({"class":"10th"}, {$set:{class:"11th"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "11th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "9th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "9th", "section" : "B section", "student count" : "39" }
>
```

Example 2 : In case of multiple matching criteria, only 1st document will be updated

> **db.school.update({"class":"12th"}, {$set:{class:"9th"}})**
>
> after this command, 1st record with class 12th will be updated to class 9th

```
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "11th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "10th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "9th", "section" : "B section", "student count" : "39" }
>
> db.school.update({"class":"12th"}, {$set:{class:"9th"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "11th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "9th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "10th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "9th", "section" : "B section", "student count" : "39" }
```

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 }) means 1 record matched and updated one record.

4. updateMany() Method :

The updateMany() method is used to update all the matching records of collection. All other parameter and system is similar to update() method.

Syntax
db.COLLECTION_NAME.updateMany(SELECTION_CRITERIA, UPDATED_DATA)

Example:
Lets update all the records of the matching criteria,

db.school.updateMany({"class":"9th"}, {$set:{class:"12th"}})

```
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "11th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "9th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "10th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "9th", "section" : "B section", "student count" : "39" }
> db.school.updateMany({"class":"9th"}, {$set:{class:"12th"}})
{ "acknowledged" : true, "matchedCount" : 2, "modifiedCount" : 2 }
>
> db.school.find()
{ "_id" : ObjectId("5eb24d83db4e42dfda38301f"), "class" : "11th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24d89db4e42dfda383020"), "class" : "12th", "section" : "C section", "student count" : "30" }
{ "_id" : ObjectId("5eb24d8fdb4e42dfda383021"), "class" : "12th", "section" : "B section", "student count" : "35" }
{ "_id" : ObjectId("5eb24ef5db4e42dfda383022"), "class" : "10th", "section" : "A section", "student count" : "40" }
{ "_id" : ObjectId("5eb24f1ddb4e42dfda383023"), "class" : "12th", "section" : "B section", "student count" : "39" }
>
```

It results as:

{ "acknowledged" : true, "matchedCount" : 2, "modifiedCount" : 2 }

Its shows that two records are matched and 2 of them has been modified.

Assignment

1.How to batch insert documents in collection?
2.How to remove documents from collection?
3. What is the difference between update and updateMany?