CRUD in MongoDB

Prashanth B S[1]

[1]Department of Information Science & Engineering, Nitte Meenakshi Institute of Technology, Yelahanka - 560064, Bengaluru

April 17, 2022

1 CRUD in MongoDB: Update

MongoDB provides *update*() method to perform updation of a feild or a set of feilds in the document upon matching it with selection criteria. In order to update a feild $*set* flag is used. When performing an update, two possibilities occurs,

1. Either update the document with *update*() method
2. Replace a document with *save*() method

By default, the *update*() method updates a single matching document[1]. Some of the variants are,

1. *update*() : updates a single matching document with $*set* flag
2. *updateOne*() : updates a single document same as *update*()
3. *updateMany*() : updates all the matching document

The general syntax for updating the document is as shown below figure 1 The following are some

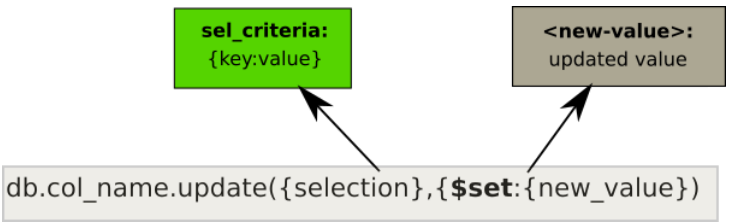

Figure 1: Syntax for Update

examples for the update, To proceed with the examples, Let us assume the dataset present in the collection "Book" under the database "BookDB" as shown below,

```
// Collection: Book, Database: BookDB,
// Dataset for the BOOKs : #3 Documents
{ "_id" : 1, "BookName" : "Lord of the Rings", "Author" : { "initials" : "J.R.R", "name"
    : "Tolkien" }, "BID" : "AE001", "tags" : [ "Fiction", "Action", "Adventure" ],
    "Price" : 100 }
{ "_id" : 2, "BookName" : "Da-Vinci Code", "Author" : { "initials" : "Dan", "name" :
    "Brown" }, "BID" : "AE002", "tags" : [ "Fiction", "Action", "Adventure", "Scifi" ],
    "Price" : 200 }
```

[1]https://www.tutorialspoint.com/mongodb/mongodb_update_document.htm

```
{ "_id" : 3, "BookName" : "The Lawyer", "Author" : { "initials" : "John", "name" :
    "Grisham" }, "BID" : "AE003", "tags" : [ "Fiction", "Action", "Adventure" ], "Price"
    : 150 }
{ "_id" : 4, "BookName" : "Angels and Demons", "Author" : { "initials" : "Dan", "name" :
    "Brown" }, "BID" : "AE004", "tags" : [ "Fiction", "Action", "Adventure", "Scifi" ],
    "Price" : 180 }
```

The following segment shows the update command variations,

```
// updating the document for the _id:4 and setting the Price = 200
> db.Book.update({_id:4},{$set:{"Price": 200}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

// updating the document based on the BookName and updating the BID
> db.Book.update({"BookName":"Angels and Demons"},{$set:{"BID": "AE005"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

// updating the initials of the Author
> db.Book.updateOne({"BookName":"Angels and Demons"},{$set:{"Author.initials": "Sir
    Dan"}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

// Adding one more duplicate record
{ "_id" : 5, "BookName" : "Angels and Demons", "Author" : { "initials" : "Dan", "name" :
    "Brown" }, "BID" : "AE004", "tags" : [ "Fiction", "Action", "Adventure", "Scifi" ],
    "Price" : 180 }

//updateMany() - set the price of the books matched with BookName to 250
> db.Book.updateMany({"BookName" : "Angels and Demons"},{$set:{"Price":250}} );
{ "acknowledged" : true, "matchedCount" : 2, "modifiedCount" : 2 }
```

Not only we can set a value, we can also unset a feild which is equivalent to removing a feild from a document. To do this use $\$unset$ flag in the place of $\$set$. The following example shows it.

```
// Lets say we want to unset few feilds matching the _id:5
> db.Book.update({"_id":5},{$unset:{"Price":"","BID":""}}) // removes Price and BID feilds
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Book.find({_id:5})
{ "_id" : 5, "BookName" : "Angels and Demons", "Author" : { "initials" : "Dan", "name" :
    "Brown" }, "tags" : [ "Fiction", "Action", "Adventure", "Scifi" ] }
```

2 CRUD in MongoDB: Delete

MongoDB's *remove*() method is used to remove a document from the collection. *remove*() method accepts two parameters. One is deletion criteria and second is justOne flag[2].

1. *deletioncriteria* : deletion criteria according to documents will be removed.
2. *justOne* : if set to true or 1, then remove only one document.

The general syntax for the *remove*() command with the above parameters is as shown below,

[2] https://www.tutorialspoint.com/mongodb/mongodb_delete_document.htm

```
> db.col_name({deletion_criteria,justOne flag= 1}
// if JustOne flag is not set, then it deletes all the matching documents
```

Some of the examples for delete are discussed below,

```
// To remove a document with _id=5
> db.Book.remove({_id:5})
WriteResult({ "nRemoved" : 1 })

// To remove all document matching the BookName = "Angels and Demons"
> db.Book.remove({"BookName": "Angels and Demons"});
WriteResult({ "nRemoved" : 2 })
```

3 CRUD-Projecting Feilds from Selection

MongoDBs *find*() method helps us to select documents based on the selection criteria and operators. Even from the selected documents, certain feilds can be projected to the output, this is optional and can be utilized for the projection. In MongoDB, when you execute *find*() method, then it displays all fields of a document. To limit this, you need to set a list of fields with value 1 or 0. 1 is used to show the field while 0 is used to hide the fields.
The general syntax is as shown ,

```
db.col_name.find({selection_criteria/empty}, {feild1:1, feild2:0});
// projects only feild1 from the selection output

// Examples : (Same Dataset)
// Listing the BookName only with BID
> db.Book.find({},{"BookName":1, "BID":1,_id:0}) // _id, and other feilds are set to 0 ,
    _id=0, explicitly specify
{ "BookName" : "Lord of the Rings", "BID" : "AE001" }
{ "BookName" : "Da-Vinci Code", "BID" : "AE002" }
{ "BookName" : "The Lawyer", "BID" : "AE003" }

// Example with selection Criteria
> db.Book.find({"Price":{$gt:100}},{"BookName":1, "BID":1,_id:0})
{ "BookName" : "Da-Vinci Code", "BID" : "AE002" }
{ "BookName" : "The Lawyer", "BID" : "AE003" }

// The projected feild can also be sorted in ascending/Descending order as follows,
// sort({key:1/-1}) 1-ascending sort. -1 -descending sort
> db.Book.find({"Price":{$gt:100}},{"BookName":1, "BID":1,_id:0}).sort({"BookName":1})
{ "BookName" : "Da-Vinci Code", "BID" : "AE002" }
{ "BookName" : "The Lawyer", "BID" : "AE003" }
```