Lab 9 – MongoDB – CRUD

# Objective

In this lab, students learn how to update documents in a MongoDB database.

**update():** This method updates one document by default. If you want to update all documents that match the criteria using this method, you need the option {multi:true}.

update(<filter>,<update>,<option>)

The *filter* parameter specifies the criteria. For instance:

{“\_id”= 0}

{} for updating all documents

The *update* parameter specifies the changes that will be applied to a document.

**updateOne():** This method updates only the first document that matches the criteria.

updateOne(<filter>,<update>)

**updateMany():** This method updates all documents that match the criteria.

updateMany(<filter>,<update>)

# Getting Started

In this lab, you will use students.json dataset. Download students.json from Blackboard and store it in a folder named dataset.

Import the dataset into a new MongoDB database named “college” and into a Collection named “students”. You can either use the command prompt to insert the data or compass.

You should see the database *college* added to the list of your databases. To see the documents inside the database:

> use college  
> db.students.find().forEach(printjson)

or

> db.students.find().pretty()

# Submission

Your submission will be online through the provided Lab 09 Submission quiz. Please copy and paste your commands and resulting output into the associated task questions in the quiz.

## Tasks

1. Write an update statement to add new fields *program* and *term* to all documents in the *students* collection and set them to values “*CPA*” and *1*.

db.students.updateMany({},

      { $set: { "Program" : "CPA", "Term" : 1 }}

   );

{

  "acknowledged": true,

  "insertedId": null,

  "matchedCount": 27,

  "modifiedCount": 27,

  "upsertedCount": 0

}

1. Write an update statement to modify the value of the *program* field to “*BTTM”* for all documents in the *students* collection.

db.students.updateMany({},

      { $set: { "Program" : "BTTM"}}

   );

{

  "acknowledged": true,

  "insertedId": null,

  "matchedCount": 27,

  "modifiedCount": 27,

  "upsertedCount": 0

}

1. Write an update statement to modify the value of the program field to “*CPA”* for the student named *Jonie Raby*.

   db.students.updateOne({"name" : "Jonie Raby"},

      { $set: { "Program" : "CPA"}}

   );

{

  "acknowledged": true,

  "insertedId": null,

  "matchedCount": 1,

  "modifiedCount": 1,

  "upsertedCount": 0

}

1. Before executing an update statement or a delete statement, you can use the *find()* method with the update or delete criteria, to see how many documents will be affected.
   * Write the update statement.
   * How many documents are there with the value *Jonie Raby* for the *name* field? 1
   * How many documents were updated? 1

1. Write a query to show only the *program* field for the document that the value of the filed *name* is *Jonie Raby*.

db.students.find({"name" : "Jonie Raby"}, {"Program" : 1, "\_id" : 0})

[

  {

    "Program": "CPA"

  }

]

1. Write an update statement to increase the value of the *term* field by 2 for documents with *\_id* 20, 22, and 24.

db.students.bulkWrite( [

   { updateMany :

      {

         "filter" : {"\_id" : 20},

         "update" : {$inc: { "Term" : 2}}

      }

   },

   {

    updateMany :

      {

         "filter" : {"\_id" : 22},

         "update" : {$inc: { "Term" : 2}}

      }

   },

   {

    updateMany :

      {

         "filter" : {"\_id" : 24},

         "update" : {$inc: { "Term" : 2}}

      }

   }

] )

{

  "acknowledged": true,

  "insertedCount": 0,

  "insertedIds": {},

  "matchedCount": 3,

  "modifiedCount": 3,

  "deletedCount": 0,

  "upsertedCount": 0,

  "upsertedIds": {}

}

1. Write an update statement to remove the *term* field from documents that the value of the *term* filed is 3.

db.students.bulkWrite( [

   {

    updateMany :

      {

         "filter" : {"Term" : 3},

         "update" : {$unset: { "Term" : 1}}

      }

   }

] )

{

  "acknowledged": true,

  "insertedCount": 0,

  "insertedIds": {},

  "matchedCount": 3,

  "modifiedCount": 3,

  "deletedCount": 0,

  "upsertedCount": 0,

  "upsertedIds": {}

}