-- We (Juhee Lee, Seonhye Hyeon) declare that the attached assignment is our own work in

--accordance with the Seneca Academic Policy.

--No part of this assignment has been copied manually or electronically from any other source

--(including web sites) or distributed to other students.

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Name: Juhee Lee

-- ID: 128261203

-- Name: Seonhye Hyeon

-- ID: 125635193

-- Date: December 4, 2021

-- Purpose: Lab 9 DBS311

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## 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}}) |

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

|  |
| --- |
| db.students.updateMany({}, {"$set": {"program": "BTM"}}) |

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

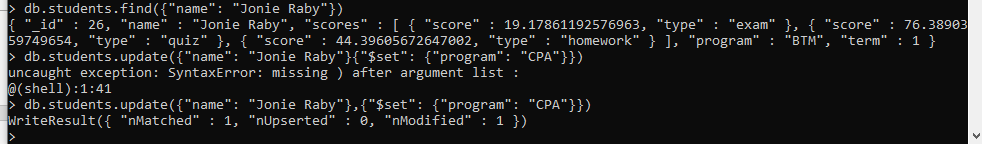
Before executing an update statement or a delete statement, you can use the *find()* method with the update or delete criteria, to see how many documents will be affected.

Write the update statement in the box below.

|  |
| --- |
| db.students.find({"name": "Jonie Raby"})  => db.students.update({"name": "Jonie Raby"},{"$set": {"program": "CPA"}}) |

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}) |

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.updateMany({"\_id": {"$in":[20,22,24]}}, {"$set": {"term":2}}) |

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

|  |
| --- |
| db.students.updateMany({"term":3}, {"$unset": {"term":1}}) |

1. Show the number of students per term Hint: Use the aggregate() function and the $group operator.

|  |
| --- |
| db.students.aggregate([{$group:{\_id:"$term", number:{$sum:1}}}]) |

1. Unwind the students collection to turn each score into its own document.

|  |
| --- |
| db.students.aggregate({"$unwind": "$scores"}).pretty() |

1. Unwind the students collection and show the average score, min score and max score per student and sort them by avgScore in descending order.

|  |
| --- |
|  |

1. Find the number of documents in the students collection.

|  |
| --- |
| db.students.count() |