***LAB 10 :***

Perform the following queries using update, delete, rename and createcollection method :

Part – A (Use collection “Student” created in Lab-9)

use Student

db.Student.insertMany([

{ "\_id": 1, "name": "John", "age": 30, "city": "New York", "isActive": true },

{ "\_id": 2, "name": "Jane", "age": 25, "city": "Los Angeles", "isActive": false },

{ "\_id": 3, "name": "Tom", "age": 35, "city": "Chicago", "isActive": true },

{ "\_id": 4, "name": "Lucy", "age": 28, "city": "San Francisco", "isActive": true },

{ "\_id": 5, "name": "David", "age": 40, "city": "Miami", "isActive": false },

{ "\_id": 6, "name": "Eva", "age": 23, "city": "Boston", "isActive": true },

{ "\_id": 7, "name": "Nick", "age": 38, "city": "Seattle", "isActive": false },

{ "\_id": 8, "name": "Sophia", "age": 27, "city": "New York", "isActive": true },

{ "\_id": 9, "name": "Liam", "age": 32, "city": "Los Angeles", "isActive": false },

{ "\_id": 10, "name": "Olivia", "age": 29, "city": "San Diego", "isActive": true }

])

1. Update the age of John's to 31.

db.Student.updateOne({name : 'John'}, {$set: {age : 31}})

2. Update the city of all students from 'New York' to 'New Jersey'.

db.Student.updateMany({city: "New York"}, {$set: {city : 'New Jersey'}})

3. Set isActive to false for every student older than 35.

db.Student.updateMany({age: {$gt : 35}}, {$set: {isActive : 'False'}})

4. Increment the age of all students by 1 year.

db.Student.updateMany({}, {$inc: {age : 1}})

5. Set the city of 'Eva' to 'Cambridge'.

db.Student.updateMany({name: "Eva"}, {$set: {city : 'Cambridge'}})

6. Update 'Sophia's isActive status to false.

db.Student.updateMany({"name": "Sophia", }, {$set: {"isActive": 'False'}}

7. Update the city field of student aged below 30 to 'San Diego'.

db.Student.updateMany({age: {$lt : 30}}, {$set: {city : 'San Diego'}})

8. Rename the age field to years for all documents.

db.Student.updateMany({}, {$rename : {age : 'Years'}})

9. Update 'Nick' to make him active (isActive = true).

db.Student.updateMany({"name": "Sophia", }, {$set: {"isActive": 'False'}})

10. Update all documents to add a new field country with the value 'USA'.

db.Student.updateMany({}, {$set: {country : "USA"}})

11. Update 'David's city to 'Orlando'.

db.Student.updateMany({"name": "David", }, {$set: {"city": 'Orlando'}})

12. Multiply the age of all students by 2.

db.Student.updateMany({}, {$mul : {age : 1}})

13. Unset (remove) the city field for 'Tom'.

db.Student.updateMany({"name": "Tom" }, {$unset: {"city": ''}})

14. Add a new field premiumUser and to true for users older than 30.

db.Student.updateMany({age : {$gt : 30}}, {$set: {premiemUser : "True"}})

15. Set isActive to true for 'Jane'.

db.Student.updateMany({"name": "Jane", }, {$set: {"isActive": 'True'}})

16. Update isActive field of 'Lucy' to false.

db.Student.updateMany({"name": "Lucy", }, {$set: {"isActive": 'False'}})

17. Delete a document of 'Nick' from the collection.

db.Student.deleteOne({name : 'Nick'})

18. Delete all students who are inactive (isActive = false).

db.Student.deleteMany({isActive : {$eq : "False"}})

19. Delete all students who live in 'New York'.

db.Student.deleteMany({city : {$eq : 'New York'}}

20. Delete all the students aged above 35.

db.Student.deleteMany({city : {$eq : 'New York'}}

21. Delete a student named ''Olivia'' from the collection.

db.Student.deleteMany({name : {$eq :'Olivia'}})

22. Delete all the students whose age is below 25.

db.Student.deleteMany({name : {$eq :'Olivia'}})

23. Delete the first student whose isActive field is true.

db.Student.deleteOne({isActive : {$eq : "True"}})

24. Delete all students from 'Los Angeles'.

db.Student.deleteMany({city : {$eq : "Los Angles"}})

25. Delete all students who have city field missing.

db.Student.deleteMany({city : {$eq : ""}})

26. Rename 'city' field to 'location' for all documents.

db.Student.updateMany({},{$rename :{city : 'Location'}})

27. Rename the name field to FullName for 'John'.

db.Student.updateMany({name : 'John'},{$set :{name : 'FullName'}})

28. Rename the isActive field to status for all documents.

db.Student.updateMany({},{$rename :{isActive : 'Status'}})

29. Rename age to yearsOld for everyone from 'San Francisco' student only.

db.Student.updateMany({Location : 'San Francisco'},{$rename :{age : 'yearOld'}})

30. Create a Capped Collection named “Employee” as per follows:

a. Ecode and Ename are compulsory fields

b. Datatype of EID is int, Ename is string, Age is int and City is string Insert following documents into above “Employee” collection.

{"Ecode": 1, "Ename": "John"}

db.createCollection("Employee", {

capped : true,

size : 5120,

max : 100,

validator : {

$jsonSchema : {

bsonType : "object",

required : ["Ecode" , "Ename"],

properties : {

Ecode : {bsonType : "int"},

Ename : {bsonType : "string"},

Age : {bsonType : "int"},

City : {bsonType : "string"}

}

}

}

})

{"Ecode ": 2, "Ename": "Jane", "age": 25, "city": "Los Angeles"}

{"Ecode ": 3, "Ename": "Tom", "age": 35}

{"Ecode ": 4, "Ename": "Lucy", "age": 28, "city": "San Francisco", "isActive": true}

db.Employee.insertMany([

{"Ecode": 2, "Ename": "Jane", "age": 25, "city": "Los Angeles"},

{"Ecode": 3, "Ename": "Tom", "age": 35},

{"Ecode": 4, "Ename": "Lucy", "age": 28, "city": "San Francisco", "isActive": true},

{"Ename": "Dino"}

])

We remove dino because

{"Ename": "Dino"}

In dino validation :

It gives   
  
 [**MongoBulkWriteError:**](file:///C:\Users\student\AppData\Local\MongoDBCompass\app-1.38.2\resources\app.asar\build\index.html) Document failed validation

PART :- B

1. Display Female students and belong to Rajkot city.

2. Display students not studying in 3rd sem.

3. Display students whose city is Jamnagar or Baroda. (use: IN)

4. Display first 2 students names who lives in Baroda.

5. Display Male students who studying in 3rd sem.

6. Display sname and city and fees of those students whose roll no is less than 105.

7. Update City of all students from 'Jamnagar' City and Department as 'CE' to 'Surat'.

8. Increase Fees by 500 where the Gender is not 'Female'. (Use: Not)

9. Set the Department of all students from 'EE' and in Sem 3 to 'Electrical'.

10. Update the Fees of students in 'Rajkot' who are male.

11. Change City to 'Vadodara' for students in Sem 5 and with fees less than 10000.

12. Delete all students where the City is 'Ahmedabad' or GENDER is 'Male'.

13. Delete students whose Rollno is not in the list [101, 105, 110].

14. Delete students from the 'Civil' department who are in Sem 5 or Sem 7.

15. Delete all students who are not in the cities 'Rajkot', 'Baroda', or 'Jamnagar'.

16. Delete students whose Rollno is between 105 and 108.

17. Rename the City field to LOCATION for all students.

18. Rename the Department field to Branch where the Fees is less than 10000.

19. Rename Sname to Fullname for students with Rollno in [106, 107, 108].

20. Rename Fees to Tuition\_Fees for all students with Fees greater than 9000.

21. Rename Department to Major where the Fees is less than 15000 and Gender is 'Female'.

22. Rename City to Hometown for all students whose SEM is 3 and Department is not 'Mechanical'.