***Note****: Query:Green Color , Output:Red Color*

**DBMS Lab**

**Assignment No. 10: Mongo DB CRUD Operations (Student Collection)**

1. Crete a collection having name “student” with following documents:

{

“roll\_no" : 1,

"name" : "amit",

"addr" : "Loni"

}

{

"roll\_no" : 2,

"name" : "shashir",

"addr" : {

"At" : "Loni",

"Tal" : "Rahata",

"Dist" : "Ahemadnagar"

}

}

{

“roll\_no" : 11,

"name" : "sachin",

"percent\_marks" : 60.23

"addr" : "Pune"

}

{

"name" : "Rahul",

"Address”: "Kolhar"

}

Ans:

> show dbs

admin 0.000GB

assign10 0.000GB

assign9 0.000GB

config 0.000GB

local 0.000GB

test 0.000GB

> use assign10

switched to db assign10

>

> show collections

student

>

> db.student.find().pretty()

>

>

> studentDetail = [{"roll\_no":1 , "name":"amit" , "addr":"Loni"},{"roll\_no":2 , "name":"shashir" , "addr":{"At":"Loni" , "Tal":"Rahata" , "Dist":"Ahemadnagar"}},{"roll\_no":11 , "name":"sachin" , "percent\_marks":60.23 , "addr":"Pune"} , {"name":"Rahul" , "Address":"Kolhar"}]

[

{

"roll\_no" : 1,

"name" : "amit",

"addr" : "Loni"

},

{

"roll\_no" : 2,

"name" : "shashir",

"addr" : {

"At" : "Loni",

"Tal" : "Rahata",

"Dist" : "Ahemadnagar"

}

},

{

"roll\_no" : 11,

"name" : "sachin",

"percent\_marks" : 60.23,

"addr" : "Pune"

},

{

"name" : "Rahul",

"Address" : "Kolhar"

}

]

> db.student.insert(studentDetail)

BulkWriteResult({

"writeErrors" : [ ],

"writeConcernErrors" : [ ],

"nInserted" : 4,

"nUpserted" : 0,

"nMatched" : 0,

"nModified" : 0,

"nRemoved" : 0,

"upserted" : [ ]

})

>

>

1. Insert new documents into a collection named “student”.
   1. > db.student.insert({"name":"akshay" , "roll\_no":7302 , "addr":"Mumbai"})
      1. WriteResult({ "nInserted" : 1 })
   2. >
   3. > db.student.find()
      1. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"), "roll\_no" : 1, "name" : "amit", "addr" : "Loni" }
      2. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"), "roll\_no" : 2, "name" : "shashir", "addr" : { "At" : "Loni", "Tal" : "Rahata", "Dist" : "Ahemadnagar" } }
      3. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"), "roll\_no" : 11, "name" : "sachin", "percent\_marks" : 60.23, "addr" : "Pune" }
      4. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"), "name" : "Rahul", "Address" : "Kolhar" }
      5. { "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"), "name" : "akshay", "roll\_no" : 7302, "addr" : "Mumbai" }
2. Retrieve all students’ details.
   1. > db.student.find().pretty()
      1. {
      2. "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"),
      3. "roll\_no" : 1,
      4. "name" : "amit",
      5. "addr" : "Loni"
      6. }
      7. {
      8. "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"),
      9. "roll\_no" : 2,
      10. "name" : "shashir",
      11. "addr" : {
      12. "At" : "Loni",
      13. "Tal" : "Rahata",
      14. "Dist" : "Ahemadnagar"
      15. }
      16. }
      17. {
      18. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
      19. "roll\_no" : 11,
      20. "name" : "sachin",
      21. "percent\_marks" : 60.23,
      22. "addr" : "Pune"
      23. }
      24. {
      25. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
      26. "name" : "Rahul",
      27. "Address" : "Kolhar"
      28. }
      29. {
      30. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
      31. "name" : "akshay",
      32. "roll\_no" : 7302,
      33. "addr" : "Mumbai"
      34. }
      35. >
3. Display only name field from student collection.
   1. > db.student.find({}, {"name":1}).pretty()
      1. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"), "name" : "amit" }
      2. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"), "name" : "shashir" }
      3. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"), "name" : "sachin" }
      4. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"), "name" : "Rahul" }
      5. { "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"), "name" : "akshay" }
      6. >
      7. >
4. Display first 5 student details. ( Use limit () )
   1. > db.student.find().limit(3)
      1. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"), "roll\_no" : 1, "name" : "amit", "addr" : "Loni" }
      2. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"), "roll\_no" : 2, "name" : "shashir", "addr" : { "At" : "Loni", "Tal" : "Rahata", "Dist" : "Ahemadnagar" } }
      3. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"), "roll\_no" : 11, "name" : "sachin", "percent\_marks" : 60.23, "addr" : "Pune" }
      4. >
5. Fetch the remaining documents after first 5 documents.(Use skip () )
   1. > db.student.find().skip(3)
      1. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"), "name" : "Rahul", "Address" : "Kolhar" }
      2. { "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"), "name" : "akshay", "roll\_no" : 7302, "addr" : "Mumbai" }
      3. >
   2. >
6. Find the details of student whose roll\_no is between 5 and 10.
   1. > db.student.find({"roll\_no":{$gt:1,$lt:10}}).pretty()
      1. {
      2. "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"),
      3. "roll\_no" : 2,
      4. "name" : "shashir",
      5. "addr" : {
      6. "At" : "Loni",
      7. "Tal" : "Rahata",
      8. "Dist" : "Ahemadnagar"
      9. }
      10. }
   2. >
7. Display the student details whose address is “Loni” or “Pune”.
   1. > db.student.find({$or:[{"addr":"Pune"},{"addr":"Loni"}]}).pretty()
      1. {
      2. "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"),
      3. "roll\_no" : 1,
      4. "name" : "amit",
      5. "addr" : "Loni"
      6. }
      7. {
      8. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
      9. "roll\_no" : 11,
      10. "name" : "sachin",
      11. "percent\_marks" : 60.23,
      12. "addr" : "Pune"
      13. }
8. Sort the documents of student collection in ascending order of roll\_no.
   1. > db.student.find({}).sort({roll\_no:-1})
      1. { "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"), "name" : "akshay", "roll\_no" : 7302, "addr" : "Mumbai" }
      2. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"), "roll\_no" : 11, "name" : "sachin", "percent\_marks" : 60.23, "addr" : "Pune" }
      3. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"), "roll\_no" : 2, "name" : "shashir", "addr" : { "At" : "Loni", "Tal" : "Rahata", "Dist" : "Ahemadnagar" } }
      4. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"), "roll\_no" : 1, "name" : "amit", "addr" : "Loni" }
      5. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"), "name" : "Rahul", "Address" : "Kolhar" }
      6. >
      7. >
   2. > db.student.find({}).sort({roll\_no:1})
      1. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"), "name" : "Rahul", "Address" : "Kolhar" }
      2. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"), "roll\_no" : 1, "name" : "amit", "addr" : "Loni" }
      3. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"), "roll\_no" : 2, "name" : "shashir", "addr" : { "At" : "Loni", "Tal" : "Rahata", "Dist" : "Ahemadnagar" } }
      4. { "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"), "roll\_no" : 11, "name" : "sachin", "percent\_marks" : 60.23, "addr" : "Pune" }
      5. { "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"), "name" : "akshay", "roll\_no" : 7302, "addr" : "Mumbai" }
      6. >
9. Remove all those documents having roll\_no less than 8 and greater than 3.
   1. > db.student.find().pretty()
      1. {
      2. "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"),
      3. "roll\_no" : 1,
      4. "name" : "amit",
      5. "addr" : "Loni"
      6. }
      7. {
      8. "\_id" : ObjectId("5f6ae2f648e364283d5f9c34"),
      9. "roll\_no" : 2,
      10. "name" : "shashir",
      11. "addr" : {
      12. "At" : "Loni",
      13. "Tal" : "Rahata",
      14. "Dist" : "Ahemadnagar"
      15. }
      16. }
      17. {
      18. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
      19. "roll\_no" : 11,
      20. "name" : "sachin",
      21. "percent\_marks" : 60.23,
      22. "addr" : "Pune"
      23. }
      24. {
      25. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
      26. "name" : "Rahul",
      27. "Address" : "Kolhar"
      28. }
      29. {
      30. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
      31. "name" : "akshay",
      32. "roll\_no" : 7302,
      33. "addr" : "Mumbai"
      34. }
   2. > db.student.deleteMany({"roll\_no":{$lt:3,$gt:1}})
      1. { "acknowledged" : true, "deletedCount" : 1 }
      2. > db.student.find().pretty()
      3. {
      4. "\_id" : ObjectId("5f6ae2f648e364283d5f9c33"),
      5. "roll\_no" : 1,
      6. "name" : "amit",
      7. "addr" : "Loni"
      8. }
      9. {
      10. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
      11. "roll\_no" : 11,
      12. "name" : "sachin",
      13. "percent\_marks" : 60.23,
      14. "addr" : "Pune"
      15. }
      16. {
      17. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
      18. "name" : "Rahul",
      19. "Address" : "Kolhar"
      20. }
      21. {
      22. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
      23. "name" : "akshay",
      24. "roll\_no" : 7302,
      25. "addr" : "Mumbai"
      26. }
10. Remove all those documents whose address is “Loni”.
    1. > db.student.deleteMany({"addr":"Loni"})
       1. { "acknowledged" : true, "deletedCount" : 1 }
       2. >
       3. > db.student.find().pretty()
       4. {
       5. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
       6. "roll\_no" : 11,
       7. "name" : "sachin",
       8. "percent\_marks" : 60.23,
       9. "addr" : "Pune"
       10. }
       11. {
       12. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
       13. "name" : "Rahul",
       14. "Address" : "Kolhar"
       15. }
       16. {
       17. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
       18. "name" : "akshay",
       19. "roll\_no" : 7302,
       20. "addr" : "Mumbai"
       21. }
       22. >
11. Add these details to a document where \_id=10 (Use save() )

Name =”Ram” Roll \_no = 60

Address =”Pune”

* 1. > db.student.save({"\_id":10 ,"name":"Ram" , "roll\_no":60 , "address":"Pune"})
     1. WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "\_id" : 10 })
     2. >
     3. > db.student.find().pretty()
     4. {
     5. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
     6. "roll\_no" : 11,
     7. "name" : "sachin",
     8. "percent\_marks" : 60.23,
     9. "addr" : "Pune"
     10. }
     11. {
     12. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
     13. "name" : "Rahul",
     14. "Address" : "Kolhar"
     15. }
     16. {
     17. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
     18. "name" : "akshay",
     19. "roll\_no" : 7302,
     20. "addr" : "Mumbai"
     21. }
     22. { "\_id" : 10, "name" : "Ram", "roll\_no" : 60, "address" : "Pune" }
     23. >
     24. >

1. Update the address of roll\_no 1. Change the address as

At: “Sangamner” Tal:”Sangamner” Dist: “Ahemadnagar”

* 1. > db.student.update({"roll\_no":7302},{$set:{"address":"Gwalior"}})
     1. WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
     2. > db.student.find().pretty()
     3. {
     4. "\_id" : ObjectId("5f6ae2f648e364283d5f9c35"),
     5. "roll\_no" : 11,
     6. "name" : "sachin",
     7. "percent\_marks" : 60.23,
     8. "addr" : "Pune"
     9. }
     10. {
     11. "\_id" : ObjectId("5f6ae2f648e364283d5f9c36"),
     12. "name" : "Rahul",
     13. "Address" : "Kolhar"
     14. }
     15. {
     16. "\_id" : ObjectId("5f6ae2fe48e364283d5f9c37"),
     17. "name" : "akshay",
     18. "roll\_no" : 7302,
     19. "addr" : "Mumbai",
     20. "address" : "Gwalior"
     21. }

1. Create Index on roll\_no field
   1. > db.student.createIndex({"roll\_no":1})
      1. {
      2. "createdCollectionAutomatically" : false,
      3. "numIndexesBefore" : 1,
      4. "numIndexesAfter" : 2,
      5. "ok" : 1
      6. }
      7. >
2. Remove collection “student”
   1. > db.student.drop()
      1. true
      2. >
   2. > show collections
      1. >