**Group B: Assignment no.1**

**MongoDB database:** create,insert delete,update operations

show dbs;

admin 40.00 KiB

config 72.00 KiB

local 64.00 KiB

use demo;

switched to db demo

db.createCollection('stud');

{ ok: 1 }

show collections;

**stud**

db.stud.insertOne({rollno:101,name:'priya',address:'abc chok'});

{

acknowledged: true,

insertedId: ObjectId(**'68d6493d0f2856402726f612'**)

}

db.stud.find();

{

\_id: ObjectId(**'68d6493d0f2856402726f612'**),

rollno: 101,

name: **'priya'**,

address: **'abc chok'**

}

db.stud.insertMany([{rollno:102,name:'riya',address:'dfg'},{rollno:103,name:'diya',address:'sdf'}]);

{

acknowledged: true,

insertedIds: {

**'0'**: ObjectId(**'68d64a640f2856402726f613'**),

**'1'**: ObjectId(**'68d64a640f2856402726f614'**)

}

}

db.stud.find();

{

\_id: ObjectId(**'68d6493d0f2856402726f612'**),

rollno: 101,

name: **'priya'**,

address: **'abc chok'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f613'**),

rollno: 102,

name: **'riya'**,

address: **'dfg'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f614'**),

rollno: 103,

name: **'diya'**,

address: **'sdf'**

}

db.stud.find().pretty();

{

\_id: ObjectId(**'68d6493d0f2856402726f612'**),

rollno: 101,

name: **'priya'**,

address: **'abc chok'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f613'**),

rollno: 102,

name: **'riya'**,

address: **'dfg'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f614'**),

rollno: 103,

name: **'diya'**,

address: **'sdf'**

}

db.stud.find({rollno:{$gt:102}});

{

\_id: ObjectId(**'68d64a640f2856402726f614'**),

rollno: 103,

name: **'diya'**,

address: **'sdf'**

}

db.stud.updateOne({rollno: 102}, {$set: {name: 'Mohini'}});

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

db.stud.find();

{

\_id: ObjectId(**'68d6493d0f2856402726f612'**),

rollno: 101,

name: **'priya'**,

address: **'abc chok'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f613'**),

rollno: 102,

name: **'Mohini'**,

address: **'dfg'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f614'**),

rollno: 103,

name: **'diya'**,

address: **'sdf'**

}

db.stud.deleteOne({rollno:101});

{

acknowledged: true,

deletedCount: 1

}

db.stud.find();

{

\_id: ObjectId(**'68d64a640f2856402726f613'**),

rollno: 102,

name: **'Mohini'**,

address: **'dfg'**

}

{

\_id: ObjectId(**'68d64a640f2856402726f614'**),

rollno: 103,

name: **'diya'**,

address: **'sdf'**

}

**Q) Problem: Create collection product having data : pid, pname, price, qty  
Change the price when price of product less than 100, update to 200. Display the record.**

show dbs;

admin 40.00 KiB

config 96.00 KiB

demo 68.00 KiB

local 72.00 KiB

db.createCollection('product');

{ ok: 1 }

db.product.insertMany([{pid:101, pname:'javabook', price:200}, {pid:102, pname:'python', price:80}, {pid:103, pname:'c++', price:160}, {pid:104, pname:'dotnet', price:500},{pid:105, pname:'mysql', price:260},{pid:106, pname:'mongodb', price:370}]);

{

acknowledged: true,

insertedIds: {

**'0'**: ObjectId(**'68d68d9b339d8df107b1238d'**),

**'1'**: ObjectId(**'68d68d9b339d8df107b1238e'**),

**'2'**: ObjectId(**'68d68d9b339d8df107b1238f'**),

**'3'**: ObjectId(**'68d68d9b339d8df107b12390'**),

**'4'**: ObjectId(**'68d68d9b339d8df107b12391'**),

**'5'**: ObjectId(**'68d68d9b339d8df107b12392'**)

}

}

db.product.find();

{

\_id: ObjectId(**'68d68d9b339d8df107b1238d'**),

pid: 101,

pname: **'javabook'**,

price: 200

}

{

\_id: ObjectId(**'68d68d9b339d8df107b1238e'**),

pid: 102,

pname: **'python'**,

price: 80

}

{

\_id: ObjectId(**'68d68d9b339d8df107b1238f'**),

pid: 103,

pname: **'c++'**,

price: 160

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12390'**),

pid: 104,

pname: **'dotnet'**,

price: 500

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12391'**),

pid: 105,

pname: **'mysql'**,

price: 260

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12392'**),

pid: 106,

pname: **'mongodb'**,

price: 370

}

db.product.updateMany({price:{lt:100}},{$set:{price:200}});

{

acknowledged: true,

insertedId: null,

matchedCount: 0,

modifiedCount: 0,

upsertedCount: 0

}

db.product.find();

{

\_id: ObjectId(**'68d68d9b339d8df107b1238d'**),

pid: 101,

pname: **'javabook'**,

price: 200

}

{

\_id: ObjectId(**'68d68d9b339d8df107b1238e'**),

pid: 102,

pname: **'python'**,

price: 80

}

{

\_id: ObjectId(**'68d68d9b339d8df107b1238f'**),

pid: 103,

pname: **'c++'**,

price: 160

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12390'**),

pid: 104,

pname: **'dotnet'**,

price: 500

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12391'**),

pid: 105,

pname: **'mysql'**,

price: 260

}

{

\_id: ObjectId(**'68d68d9b339d8df107b12392'**),

pid: 106,

pname: **'mongodb'**,

price: 370

}

test