HPath: cd c:\Program Files\MongoDB\Server\5.0\bin

mongo(to open shell)

Create or switch to db: use demo;

Create table/collection : db.createCollection('student');

Insert value: db.student.insert({'rno':'1','name':'Ann','class':'te comp'});

db.student.insert({'rno':'6','name':'Anto','class':'te comp','age':'22'});

(title: "Tag")

Multiple values : db.student.insertMany([{'rno':'2','name':'Denny','age':'22'},{'rno':'4','name':'Smith','class':'maths'}]);

Display : db.student.find().pretty();

Update : db.student.update({'name':'Anto'},{$set:{age:21}});

Multiple : db.student.update({'name':'Ann'},{$set:{'class':'sci','name':'Helen'}});

db.RecordsDB.updateOne({name: "Marsh"}, {$set:{ownerAddress: "451 W. Coffee St. A204"}})

{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }

{ "\_id" : ObjectId("5fd993a2ce6e8850d88270b7"), "name" : "Marsh", "age" : "6 years", "species" : "Dog", "ownerAddress" : "451 W. Coffee St. A204", "chipped" : true }

db.RecordsDB.updateMany({species:"Dog"}, {$set: {age: "5"}})

{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 3 }

> db.RecordsDB.find()

{ "\_id" : ObjectId("5fd98ea9ce6e8850d88270b5"), "name" : "Kitana", "age" : "4 years", "species" : "Cat", "ownerAddress" : "521 E. Cortland", "chipped" : true }

{ "\_id" : ObjectId("5fd993a2ce6e8850d88270b7"), "name" : "Marsh", "age" : "5", "species" : "Dog", "ownerAddress" : "451 W. Coffee St. A204", "chipped" : true }

{ "\_id" : ObjectId("5fd993f3ce6e8850d88270b8"), "name" : "Loo", "age" : "5", "species" : "Dog", "ownerAddress" : "380 W. Fir Ave", "chipped" : true }

{ "\_id" : ObjectId("5fd994efce6e8850d88270ba"), "name" : "Kevin", "age" : "5", "species" : "Dog", "ownerAddress" : "900 W. Wood Way", "chipped" : true }

db.RecordsDB.replaceOne({name: "Kevin"}, {name: "Maki"})

> db.RecordsDB.find()

{ "\_id" : ObjectId("5fd98ea9ce6e8850d88270b5"), "name" : "Kitana", "age" : "4 years", "species" : "Cat", "ownerAddress" : "521 E. Cortland", "chipped" : true }

{ "\_id" : ObjectId("5fd993a2ce6e8850d88270b7"), "name" : "Marsh", "age" : "5", "species" : "Dog", "ownerAddress" : "451 W. Coffee St. A204", "chipped" : true }

{ "\_id" : ObjectId("5fd993f3ce6e8850d88270b8"), "name" : "Loo", "age" : "5", "species" : "Dog", "ownerAddress" : "380 W. Fir Ave", "chipped" : true }

{ "\_id" : ObjectId("5fd994efce6e8850d88270ba"), "name" : "Maki" }

db.RecordsDB.deleteOne({name:"Maki"})

> db.RecordsDB.find()

{ "\_id" : ObjectId("5fd98ea9ce6e8850d88270b5"), "name" : "Kitana", "age" : "4 years", "species" : "Cat", "ownerAddress" : "521 E. Cortland", "chipped" : true }

{ "\_id" : ObjectId("5fd993a2ce6e8850d88270b7"), "name" : "Marsh", "age" : "5", "species" : "Dog", "ownerAddress" : "451 W. Coffee St. A204", "chipped" : true }

{ "\_id" : ObjectId("5fd993f3ce6e8850d88270b8"), "name" : "Loo", "age" : "5", "species" : "Dog", "ownerAddress" : "380 W. Fir Ave", "chipped" : true }

db.RecordsDB.deleteMany({species:"Dog"})

> db.RecordsDB.find()

{ "\_id" : ObjectId("5fd98ea9ce6e8850d88270b5"), "name" : "Kitana", "age" : "4 years", "species" : "Cat", "ownerAddress" : "521 E. Cortland", "chipped" : true }

Display specifically :

db.student.find({$and:[{"name":"Denny"},{"age":"22"}]}).pretty();

Output :

{

"\_id" : ObjectId("6274a70e3492694406d67f2f"),

"rno" : "2",

"name" : "Denny",

"age" : "22"

}

db.student.find({$or:[{"name":"Denny"},{"name":"Anto"}]}).pretty();

output

{

"\_id" : ObjectId("62749d593492694406d67f2e"),

"rno" : "6",

"name" : "Anto",

"class" : "te comp",

"age" : 21

}

{

"\_id" : ObjectId("6274a70e3492694406d67f2f"),

"rno" : "2",

"name" : "Denny",

"age" : "22"

}

Aggregate :

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

{

"\_id" : ObjectId("62749c033492694406d67f2c"),

"rno" : "1",

"name" : "Helen",

"class" : "sci",

"fees" : 30000

}

{

"\_id" : ObjectId("62749c5a3492694406d67f2d"),

"rno" : "7",

"name" : "Liz",

"class" : "comp",

"fees" : 40000

}

{

"\_id" : ObjectId("62749d593492694406d67f2e"),

"rno" : "6",

"name" : "Anto",

"class" : "te comp",

"age" : 21,

"fees" : 20000

}

{

"\_id" : ObjectId("6274a70e3492694406d67f2f"),

"rno" : "2",

"name" : "Smith",

"age" : "23",

"fees" : 30000

}

{

"\_id" : ObjectId("6274a70e3492694406d67f30"),

"rno" : "4",

"name" : "Smith",

"class" : "maths",

"fees" : 10000

}

]

Sum :

db.student.aggregate({$group:{\_id:"$name","total":{$sum:"$fees"}}});

Op :

{ "\_id" : "Liz", "total" : 40000 }

{ "\_id" : "Helen", "total" : 30000 }

{ "\_id" : "Anto", "total" : 20000 }

{ "\_id" : "Smith", "total" : 40000 }

Average :

> db.student.aggregate({$group:{\_id:"$name","average":{$avg:"$fees"}}});

{ "\_id" : "Liz", "average" : 40000 }

{ "\_id" : "Helen", "average" : 30000 }

{ "\_id" : "Anto", "average" : 20000 }

{ "\_id" : "Smith", "average" : 20000 }

Minimum

db.student.aggregate({$group:{\_id:"$name","minimum":{$min:"$fees"}}});

{ "\_id" : "Liz", "minimum" : 40000 }

{ "\_id" : "Helen", "minimum" : 30000 }

{ "\_id" : "Smith", "minimum" : 10000 }

{ "\_id" : "Anto", "minimum" : 20000 }

Maximum

db.student.aggregate({$group:{\_id:"$name","maximum":{$max:"$fees"}}});

{ "\_id" : "Helen", "maximum" : 30000 }

{ "\_id" : "Anto", "maximum" : 20000 }

{ "\_id" : "Smith", "maximum" : 30000 }

{ "\_id" : "Liz", "maximum" : 40000 }

First

db.student.aggregate({$group:{\_id:"$name","first":{$first:"$fees"}}});

{ "\_id" : "Helen", "first" : 30000 }

{ "\_id" : "Liz", "first" : 40000 }

{ "\_id" : "Smith", "first" : 30000 }

{ "\_id" : "Anto", "first" : 20000 }

Last

db.student.aggregate({$group:{\_id:"$name","last":{$last:"$fees"}}});

{ "\_id" : "Liz", "last" : 40000 }

{ "\_id" : "Helen", "last" : 30000 }

{ "\_id" : "Smith", "last" : 10000 }

{ "\_id" : "Anto", "last" : 20000 }

Show collections list in db: show collections mycol;

db.getCollectionNames();

To drop collection: db.employee.drop();

Mongo dump

(run as administrator

path

mongodump)

[At mongo(shell)

db.getCollectionNames();

db.dropDatabase();

(Drop collection : db.employee.drop();)

exit;

Restore : mongorestore

To see weather mongo is restored:

Move to mongo

(mongo)

Show dbs]

**Create db**

use Demo

switched to db Demo

**Create collection**

> db.createCollection('student')

{ "ok" : 1 }

**Insert**

> db.student.insert({'Rno':1,'Name':'Piyush','class':'TE Comp'})

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'2','Name':'Abhi','Class':'TE COMP'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'3','Name':'Ashley','Class':'TE COMP'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'4','Name':'Hitesh','Class':'TE COMP'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'5','Name':'Pratik','Class':'TE COMP'});

WriteResult({ "nInserted" : 1 })

> db.student.find();

{ "\_id" : ObjectId("627dd954504d747f78a83a6d"), "Rno" : 1, "Name" : "Piyush", "class" : "TE Comp" }

{ "\_id" : ObjectId("627dda9c504d747f78a83a73"), "Rno" : "2", "Name" : "Abhi", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627ddb4b504d747f78a83a74"), "Rno" : "3", "Name" : "Ashley", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627ddb5d504d747f78a83a75"), "Rno" : "4", "Name" : "Hitesh", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627ddb73504d747f78a83a76"), "Rno" : "5", "Name" : "Pratik", "Class" : "TE COMP" }

**Collection-2**

> db.createCollection('employee')

{ "ok" : 1 }

> db.employee.insert({'roll':'1','name':'harsh','amount':1000,'url':'www.yahoo.com'});

WriteResult({ "nInserted" : 1 })

> db.Employee.insert({'empid':105,'Name':'Anu','city':'Kottayam'})

WriteResult({ "nInserted" : 1 })

> db.Employee.insert({'empid':105,'Name':'Anu','city':'Kottayam'})

WriteResult({ "nInserted" : 1 })

> db.employee.insert({'empid':105,'Name':'Anu','city':'Kottayam'})

WriteResult({ "nInserted" : 1 })

> db.employee.insert({'empid':107,'Name':'mary','city':'Kochi'})

WriteResult({ "nInserted" : 1 })

**Run cmd as administrator:**

C:\Program Files\MongoDB\Server\5.0\bin>mongodump

Go to C:\Program Files\MongoDB\Server\5.0\bin>dump directory

And check files

**Back to cmd shell:**

Drop database :

> db.dropDatabase()

{ "ok" : 1 }

> show dbs

admin 0.000GB

config 0.000GB

local 0.000GB

student 0.000GB

**Exit from shell:**

> exit

Bye

**Restore:**

C:\Program Files\MongoDB\Server\5.0\bin>mongorestore

security:

authorization:"enabled"

use admin

db.createUser(

{

user: "admin",

pwd: "admin",

roles: [ { "role" : "userAdminAnyDatabase","db" : "admin"}, "readWriteAnyDatabase"]

}

)

mongo -u "admin" -p "admin" --authenticationDatabase "admin”