**Mongo commands**

1. > show dbs

admin   0.000GB

config  0.000GB

local   0.000GB

test 0.000GB

2. > db.student.find();

{ "\_id" : ObjectId("6273990f2d39f95c8461b08f"), "Rno" : "1", "Name" : "piyush", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627399752d39f95c8461b090"), "Rno" : "2", "Name" : "abhi", "Age" : "21", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627399952d39f95c8461b091"), "Rno" : "3", "Name" : "ashley", "Age" : "20", "Class" : "TE COMP" }

{ "\_id" : ObjectId("627399c52d39f95c8461b092"), "Rno" : "4", "Name" : "hitesh", "Class" : "TE COMP" }

{ "\_id" : ObjectId("62739a122d39f95c8461b093"), "Rno" : "5", "Name" : "pratik", "Age" : "20", "Class" : "TE COMP", "Mark" : "80" }

{ "\_id" : ObjectId("62739a4c2d39f95c8461b094"), "Rno" : "6", "Name" : "piyush", "Age" : "21", "Class" : "TE COMP", "Mark" : "90" }

3.> db.student.update({'Name':'piyush'},{$set:{'Name':'maya'}});

WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

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

{

        "\_id" : ObjectId("6273990f2d39f95c8461b08f"),

        "Rno" : "1",

        "Name" : "maya",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("627399752d39f95c8461b090"),

        "Rno" : "2",

        "Name" : "abhi",

        "Age" : "21",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("627399952d39f95c8461b091"),

        "Rno" : "3",

        "Name" : "ashley",

        "Age" : "20",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("627399c52d39f95c8461b092"),

        "Rno" : "4",

        "Name" : "hitesh",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("62739a122d39f95c8461b093"),

        "Rno" : "5",

        "Name" : "pratik",

        "Age" : "20",

        "Class" : "TE COMP",

        "Mark" : "80"

}

{

        "\_id" : ObjectId("62739a4c2d39f95c8461b094"),

        "Rno" : "6",

        "Name" : "piyush",

        "Age" : "21",

        "Class" : "TE COMP",

        "Mark" : "90"

}

4.  > db.student.remove({'Name':'abhi'});

WriteResult({ "nRemoved" : 1 })

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

{

        "\_id" : ObjectId("6273990f2d39f95c8461b08f"),

        "Rno" : "1",

        "Name" : "maya",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("627399952d39f95c8461b091"),

        "Rno" : "3",

        "Name" : "ashley",

        "Age" : "20",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("627399c52d39f95c8461b092"),

        "Rno" : "4",

        "Name" : "hitesh",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("62739a122d39f95c8461b093"),

        "Rno" : "5",

        "Name" : "pratik",

        "Age" : "20",

        "Class" : "TE COMP",

        "Mark" : "80"

}

{

        "\_id" : ObjectId("62739a4c2d39f95c8461b094"),

        "Rno" : "6",

        "Name" : "piyush",

        "Age" : "21",

        "Class" : "TE COMP",

        "Mark" : "90"

}

5.>db.Student.save({\_id:ObjectId("5b8fad4ef00832a0a50b5036"),"RNO":"1","NAME":"PIYUSH","CLASS":"TE COMP","ADD":"MP"});

WriteResult({

        "nMatched" : 0,

        "nUpserted" : 1,

        "nModified" : 0,

        "\_id" : ObjectId("5b8fad4ef00832a0a50b5036")

})

>  db.Student.find().pretty();

{

        "\_id" : ObjectId("5b8fad4ef00832a0a50b5036"),

        "RNO" : "1",

        "NAME" : "PIYUSH",

        "CLASS" : "TE COMP",

        "ADD" : "MP"

}

6. > db.Stud.find( {"Rno": { $lt:"5"}}).pretty();

{

        "\_id" : ObjectId("6278cead5bca9f2c9121f313"),

        "Rno" : "1",

        "Name" : "Piyush",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cec55bca9f2c9121f314"),

        "Rno" : "2",

        "Name" : "Abhi",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cede5bca9f2c9121f315"),

        "Rno" : "3",

        "Name" : "Ashley",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cef35bca9f2c9121f316"),

        "Rno" : "4",

        "Name" : "Hitesh",

        "Class" : "TE COMP"

}

> db.Stud.find( {"Rno": { $lt:"5",$gt:"2"}}).pretty();

{

        "\_id" : ObjectId("6278cede5bca9f2c9121f315"),

        "Rno" : "3",

        "Name" : "Ashley",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cef35bca9f2c9121f316"),

        "Rno" : "4",

        "Name" : "Hitesh",

        "Class" : "TE COMP"

}

> db.Stud.find( {"Rno": { $lte:"5",$gte:"2"}}).pretty();

{

        "\_id" : ObjectId("6278cec55bca9f2c9121f314"),

        "Rno" : "2",

        "Name" : "Abhi",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cede5bca9f2c9121f315"),

        "Rno" : "3",

        "Name" : "Ashley",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cef35bca9f2c9121f316"),

        "Rno" : "4",

        "Name" : "Hitesh",

        "Class" : "TE COMP"

}

{

        "\_id" : ObjectId("6278cf0a5bca9f2c9121f317"),

        "Rno" : "5",

        "Name" : "Pratik",

        "Class" : "TE COMP"

}

Aggregate function

//USE DATABASE

> use comp;

switched to db comp

//CREATE COLLECTION WEBSITE

> db.createCollection('website');

{ "ok" : 1 }

//INSERT VALUES IN WEBSITE

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

WriteResult({ "nInserted" : 1 })

> db.website.insert({'roll':'2','name':'jitesh','amount':2000,'url':'www.yahoo.com'});

WriteResult({ "nInserted" : 1 })

> db.website.insert({'roll':'3','name':'rina','amount':3000,'url':'www.google.com'});

WriteResult({ "nInserted" : 1 })

> db.website.insert({'roll':'4','name':'ash','amount':4000,'url':'www.gmail.com'});

WriteResult({ "nInserted" : 1 })

> db.website.insert({'roll':'5','name':'ash','amount':1000,'url':'www.pvg.com'});

WriteResult({ "nInserted" : 1 })

//SUM AGGREGATE

1. > db.website.aggregate({$group:{\_id:"$name","total":{$sum:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "ash", "total" : 5000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "jitesh", "total" : 2000 }

//AVG AGGREGATE

2. > db.website.aggregate({$group:{\_id:"$name","total":{$avg:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "jitesh", "total" : 2000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "ash", "total" : 2500 }

//MIN AGGREGATION

3. > db.website.aggregate({$group:{\_id:"$name","total":{$min:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "ash", "total" : 1000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "jitesh", "total" : 2000 }

//MAX AGGREGATION

4. > db.website.aggregate({$group:{\_id:"$name","total":{$max:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "jitesh", "total" : 2000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "ash", "total" : 4000 }

//FIRST AGGREGATION

5. > db.website.aggregate({$group:{\_id:"$name","total":{$first:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "jitesh", "total" : 2000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "ash", "total" : 4000 }

//LAST AGGREGATION

6. > db.website.aggregate({$group:{\_id:"$name","total":{$last:"$amount"}}});

{ "\_id" : "harsh", "total" : 1000 }

{ "\_id" : "ash", "total" : 1000 }

{ "\_id" : "rina", "total" : 3000 }

{ "\_id" : "jitesh", "total" : 2000 }

//PUSH AGGREGATION

7. > db.website.aggregate({$group:{\_id:"$name","total":{$push:"$amount"}}});

{ "\_id" : "harsh", "total" : [ 1000 ] }

{ "\_id" : "jitesh", "total" : [ 2000 ] }

{ "\_id" : "rina", "total" : [ 3000 ] }

{ "\_id" : "ash", "total" : [ 4000, 1000 ] }

//COUNT AGGREGATION

8. > db.website.aggregate({$group:{\_id:"$name","total": {$sum:1}}});

{ "\_id" : "harsh", "total" : 1 }

{ "\_id" : "ash", "total" : 2 }

{ "\_id" : "rina", "total" : 1 }

{ "\_id" : "jitesh", "total" : 1 }

Mongodump

Command prompt

> use demo;

switched to db demo

> db.createCollection('student');

{ "ok" : 1 }

> db.student.insert({'Rno':'1','Name':'Athulya','Class':'S2mca'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'2','Name':'Arya','Class':'S2mca','Age':'21'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'2','Name':'Reshma','Class':'S2mca','Age':'25'});

WriteResult({ "nInserted" : 1 })

> db.student.insert({'Rno':'2','Name':'Arathy','Class':'S4bca'});

WriteResult({ "nInserted" : 1 })

> db.student.find();

{ "\_id" : ObjectId("627dd9970ea1df75fa58259d"), "Rno" : "1", "Name" : "Athulya", "Class" : "S2mca" }

{ "\_id" : ObjectId("627dd9d00ea1df75fa58259e"), "Rno" : "2", "Name" : "Arya", "Class" : "S2mca", "Age" : "21" }

{ "\_id" : ObjectId("627dda240ea1df75fa58259f"), "Rno" : "2", "Name" : "Reshma", "Class" : "S2mca", "Age" : "25" }

{ "\_id" : ObjectId("627dda620ea1df75fa5825a0"), "Rno" : "2", "Name" : "Arathy", "Class" : "S4bca" }

> db.createCollection('Employee');

{ "ok" : 1 }

> db.Employee.insert({'Empno':'1','Empname':'Arathy','Dep':'Engineer'});

WriteResult({ "nInserted" : 1 })

> db.Employee.insert({'Empno':'2','Empname':'Bhavana','Dep':'Engineer','Depno':'10'});

WriteResult({ "nInserted" : 1 })

> db.Employee.insert({'Empno':'3','Empname':'Gokul','Dep':'Bussiness','Depno':'20'});

WriteResult({ "nInserted" : 1 })

> db.Employee.insert({'Empno':'4','Empname':'Amitga','Dep':'Bussiness','Depno':'20'});

WriteResult({ "nInserted" : 1 })

>

> db.Employee.find().pretty();

{

        "\_id" : ObjectId("627ddb510ea1df75fa5825a1"),

        "Empno" : "1",

        "Empname" : "Arathy",

        "Dep" : "Engineer"

}

{

        "\_id" : ObjectId("627ddb7a0ea1df75fa5825a2"),

        "Empno" : "2",

        "Empname" : "Bhavana",

        "Dep" : "Engineer",

        "Depno" : "10"

}

{

        "\_id" : ObjectId("627ddba60ea1df75fa5825a3"),

        "Empno" : "3",

        "Empname" : "Gokul",

        "Dep" : "Bussiness",

        "Depno" : "20"

}

{

        "\_id" : ObjectId("627ddbdc0ea1df75fa5825a4"),

        "Empno" : "4",

        "Empname" : "Amitga",

        "Dep" : "Bussiness",

        "Depno" : "20"

}

> use test

switched to db test

> db.createCollection('college');

{ "ok" : 1 }

> db.college.insert({'Tno':'1','Tname':'Aswathy','Class':'Te comp',});

WriteResult({ "nInserted" : 1 })

> db.college.insert({'Tno':'2','Tname':'Aswani','Class':'Te comp'});

WriteResult({ "nInserted" : 1 })

> db.college.insert({'Tno':'3','Tname':'shamili','Class':'Te comp'});

WriteResult({ "nInserted" : 1 })

> db.college.insert({'Tno':'4','Tname':'chandini','Class':'Te comp'});

WriteResult({ "nInserted" : 1 })

> db.college.find().pretty();

{

        "\_id" : ObjectId("627ddcc80ea1df75fa5825a5"),

        "Tno" : "1",

        "Tname" : "Aswathy",

        "Class" : "Te comp"

}

{

        "\_id" : ObjectId("627ddceb0ea1df75fa5825a6"),

        "Tno" : "2",

        "Tname" : "Aswani",

        "Class" : "Te comp"

}

{

        "\_id" : ObjectId("627ddd0c0ea1df75fa5825a7"),

        "Tno" : "3",

        "Tname" : "shamili",

        "Class" : "Te comp"

}

{

        "\_id" : ObjectId("627ddd270ea1df75fa5825a8"),

        "Tno" : "4",

        "Tname" : "chandini",

        "Class" : "Te comp"

}

> db.dropDatabase();

{ "ok" : 1 }

> show dbs

admin   0.000GB

config  0.000GB

demo    0.000GB

local   0.000GB

> show dbs

admin   0.000GB

config  0.000GB

demo    0.000GB

local   0.000GB

test    0.000GB

Administrator

C:\Windows\system32>cd C:\Program Files\MongoDB\Server\5.0\bin\

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

2022-05-13T09:56:18.082+0530    writing admin.system.version to dump\admin\system.version.bson

2022-05-13T09:56:18.087+0530    done dumping admin.system.version (1 document)

2022-05-13T09:56:18.089+0530    writing demo.Employee to dump\demo\Employee.bson

2022-05-13T09:56:18.092+0530    done dumping demo.Employee (4 documents)

2022-05-13T09:56:18.392+0530    writing test.college to dump\test\college.bson

2022-05-13T09:56:18.393+0530    writing demo.student to dump\demo\student.bson

2022-05-13T09:56:18.396+0530    done dumping test.college (4 documents)

2022-05-13T09:56:18.396+0530    done dumping demo.student (4 documents)

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

2022-05-13T10:05:19.043+0530    using default 'dump' directory

2022-05-13T10:05:19.044+0530    preparing collections to restore from

2022-05-13T10:05:19.059+0530    reading metadata for demo.Employee from dump\demo\Employee.metadata.json

2022-05-13T10:05:19.072+0530    reading metadata for demo.student from dump\demo\student.metadata.json

2022-05-13T10:05:19.084+0530    reading metadata for test.college from dump\test\college.metadata.json

2022-05-13T10:05:19.086+0530    restoring to existing collection demo.Employee without dropping

2022-05-13T10:05:19.087+0530    restoring to existing collection demo.student without dropping

2022-05-13T10:05:19.100+0530    restoring demo.Employee from dump\demo\Employee.bson

2022-05-13T10:05:19.100+0530    restoring demo.student from dump\demo\student.bson

2022-05-13T10:05:19.189+0530    restoring test.college from dump\test\college.bson

2022-05-13T10:05:19.210+0530    finished restoring test.college (4 documents, 0 failures)

2022-05-13T10:05:19.461+0530    continuing through error: E11000 duplicate key error collection: demo.Employee index: \_id\_ dup key: { \_id: ObjectId('627ddb510ea1df75fa5825a1') }

2022-05-13T10:05:19.461+0530    continuing through error: E11000 duplicate key error collection: demo.Employee index: \_id\_ dup key: { \_id: ObjectId('627ddb7a0ea1df75fa5825a2') }

2022-05-13T10:05:19.462+0530    continuing through error: E11000 duplicate key error collection: demo.Employee index: \_id\_ dup key: { \_id: ObjectId('627ddba60ea1df75fa5825a3') }

2022-05-13T10:05:19.464+0530    continuing through error: E11000 duplicate key error collection: demo.student index: \_id\_ dup key: { \_id: ObjectId('627dd9970ea1df75fa58259d') }

2022-05-13T10:05:19.465+0530    continuing through error: E11000 duplicate key error collection: demo.student index: \_id\_ dup key: { \_id: ObjectId('627dd9d00ea1df75fa58259e') }

2022-05-13T10:05:19.467+0530    continuing through error: E11000 duplicate key error collection: demo.Employee index: \_id\_ dup key: { \_id: ObjectId('627ddbdc0ea1df75fa5825a4') }

2022-05-13T10:05:19.468+0530    continuing through error: E11000 duplicate key error collection: demo.student index: \_id\_ dup key: { \_id: ObjectId('627dda240ea1df75fa58259f') }

2022-05-13T10:05:19.469+0530    continuing through error: E11000 duplicate key error collection: demo.student index: \_id\_ dup key: { \_id: ObjectId('627dda620ea1df75fa5825a0') }

2022-05-13T10:05:19.474+0530    finished restoring demo.Employee (0 documents, 4 failures)

2022-05-13T10:05:19.475+0530    finished restoring demo.student (0 documents, 4 failures)

2022-05-13T10:05:19.477+0530    no indexes to restore for collection test.college

2022-05-13T10:05:19.478+0530    no indexes to restore for collection demo.Employee

2022-05-13T10:05:19.480+0530    no indexes to restore for collection demo.student

2022-05-13T10:05:19.481+0530    4 document(s) restored successfully. 8 document(s) failed to restore.