Please enter a MongoDB connection string (Default: mongodb://localhost/): mongodb://localhost/

mongodb://localhost/

Current Mongosh Log ID: 6618a520cc627b09619f9909

Connecting to: mongodb://localhost/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.2.2

Using MongoDB: 7.0.7

Using Mongosh: 2.2.2

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------

The server generated these startup warnings when booting

2024-04-12T08:15:25.910+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

------

Warning: Found ~/.mongorc.js, but not ~/.mongoshrc.js. ~/.mongorc.js will not be loaded.

You may want to copy or rename ~/.mongorc.js to ~/.mongoshrc.js.

test> show dbs

admin 40.00 KiB

config 72.00 KiB

local 72.00 KiB

test> use employee

switched to db employee

employee> db.createCollection("employee")

{ ok: 1 }

employee> db.employee.insertOne({

... title: 101

...

...

...

employee> db.employee.insertOne({

... id: 101,

... name: "ABC",

... stree: "XYZ House",

employee>

employee>

employee> db.employee.insertOne({

... id: 101,

... name: "Joseph Thampy',

Uncaught:

SyntaxError: Unterminated string constant. (3:6)

1 | db.employee.insertOne({

2 | id: 101,

> 3 | name: "Joseph Thampy',

| ^

4 |

employee> db.employee.insertOne({

... id: 101,

... name: "Joseph Thampy",

... street: ["ABC House", "XYZ"],

... city: "Idukki",

... department: "Finance"

... })

{

acknowledged: true,

insertedId: ObjectId('6618ac75cc627b09619f990a')

}

employee> db.employee.insertMany({

... id: 102,

... name: "Kavya"

... street: ["DEF House", "WXY"],

Uncaught:

SyntaxError: Unexpected token, expected "," (4:0)

2 | id: 102,

3 | name: "Kavya"

> 4 | street: ["DEF House", "WXY"],

| ^

5 |

employee> db.employee.insertMany({

... id: 102,

... name: "kavya",

... street: ["DEF House", "WXY"],

... city: "Kottayam",

... department: "Corporate

Uncaught:

SyntaxError: Unterminated string constant. (6:12)

4 | street: ["DEF House", "WXY"],

5 | city: "Kottayam",

> 6 | department: "Corporate

| ^

7 |

employee> db.employee.insertMany({ id: 102, name: "kavya", street: ["DEF House", "WXY"], city: "Kottayam", department: "Corporate

Uncaught:

SyntaxError: Unterminated string constant. (1:109)

> 1 | db.employee.insertMany({ id: 102, name: "kavya", street: ["DEF House", "WXY"], city: "Kottayam", department: "Corporate

| ^

2 |

employee> db.employee.insertMany({

... id: 102,

... name: "Kavya"

... street: ["DEF House", "WXY"],

Uncaught:

SyntaxError: Unexpected token, expected "," (4:0)

2 | id: 102,

3 | name: "Kavya"

> 4 | street: ["DEF House", "WXY"],

| ^

5 |

employee> db.employee.insertMany({ id: 102, name: "kavya", street: ["DEF House", "WXY"], city: "Kottayam", department: "Corporate" }, { id: 103, name: "Keerthana", street: ["GHI House", "VWX"], city: "Pala", department: "Finance" }, { id: 104, name: "Lakshmi Prasannan", street: ["JKL House", "UVW"], city: "Ettumanoor", department: "Technology" })

MongoInvalidArgumentError: Argument "docs" must be an array of documents

employee> db.employee.insertMany({

... id: 102,

... name: "kavya",

... street: ["DEF House", "WXY"],

... city: "Kottayam",

... department: "Corporate"

... },

... {

... id: 103,

... name: "Keerthana",

... street: ["GHI House", "VWX"],

... city: "Pala",

... department: "Finance"

... },

... {

... id: 104,

... name: "Lakshmi Prasannan",

... street: ["JKL House", "UVW"],

... city: "Ettumanoor",

... department: "Technology"

... })

MongoInvalidArgumentError: Argument "docs" must be an array of documents

employee> db.employee.insertMany([{ id: 102, name: "kavya", street: ["DEF House", "WXY"], city: "Kottayam", department: "Corporate" }, { id: 103, name: "Keerthana", street: ["GHI House", "VWX"], city: "Pala", department: "Finance" }, { id: 104, name: "Lakshmi Prasannan", street: ["JKL House", "UVW"], city: "Ettumanoor", department: "Technology" }])

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6618b000cc627b09619f990b'),

'1': ObjectId('6618b000cc627b09619f990c'),

'2': ObjectId('6618b000cc627b09619f990d')

}

}

employee> db.employee.insertMany([

... {

... id: 105,

... name: "Merin Babu",

... street: ["TUV House", "WZQ"],

... city: "Ettumanoor",

... department: "Robotics"

... },

... {

... id: 106,

... name: "ABC",

... street: ["MNO House", "APS"],

... city: "Kottayam",

... Department: "Security"

... }

... ])

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6618b0f4cc627b09619f990e'),

'1': ObjectId('6618b0f4cc627b09619f990f')

}

}

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance'

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate'

},

{

\_id: ObjectId('6618b000cc627b09619f990c'),

id: 103,

name: 'Keerthana',

street: [ 'GHI House', 'VWX' ],

city: 'Pala',

department: 'Finance'

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology'

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics'

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security'

}

]

employee> db.employee.update({}, { $inc: {salary: 10000} })

DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance',

salary: 10000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate'

},

{

\_id: ObjectId('6618b000cc627b09619f990c'),

id: 103,

name: 'Keerthana',

street: [ 'GHI House', 'VWX' ],

city: 'Pala',

department: 'Finance'

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology'

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics'

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security'

}

]

employee> db.employee.updateMany({}, { $inc: {salary: 10000} })

{

acknowledged: true,

insertedId: null,

matchedCount: 6,

modifiedCount: 6,

upsertedCount: 0

}

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate',

salary: 10000

},

{

\_id: ObjectId('6618b000cc627b09619f990c'),

id: 103,

name: 'Keerthana',

street: [ 'GHI House', 'VWX' ],

city: 'Pala',

department: 'Finance',

salary: 10000

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology',

salary: 10000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics',

salary: 10000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security',

salary: 10000

}

]

employee> db.employee.updateMany({}, { $inc: {salary: 10000} })

{

acknowledged: true,

insertedId: null,

matchedCount: 6,

modifiedCount: 6,

upsertedCount: 0

}

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance',

salary: 30000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990c'),

id: 103,

name: 'Keerthana',

street: [ 'GHI House', 'VWX' ],

city: 'Pala',

department: 'Finance',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security',

salary: 20000

}

]

employee> db.employee.update({id: 103}, {$set: {salary: 10000}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance',

salary: 30000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990c'),

id: 103,

name: 'Keerthana',

street: [ 'GHI House', 'VWX' ],

city: 'Pala',

department: 'Finance',

salary: 10000

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security',

salary: 20000

}

]

employee> db.employee.deleteOne({salary: 10000})

{ acknowledged: true, deletedCount: 1 }

employee> db.employee.find()

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

id: 101,

name: 'Joseph Thampy',

street: [ 'ABC House', 'XYZ' ],

city: 'Idukki',

department: 'Finance',

salary: 30000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

id: 102,

name: 'kavya',

street: [ 'DEF House', 'WXY' ],

city: 'Kottayam',

department: 'Corporate',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

id: 104,

name: 'Lakshmi Prasannan',

street: [ 'JKL House', 'UVW' ],

city: 'Ettumanoor',

department: 'Technology',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

id: 105,

name: 'Merin Babu',

street: [ 'TUV House', 'WZQ' ],

city: 'Ettumanoor',

department: 'Robotics',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security',

salary: 20000

}

]

employee> db.employee.aggregate([

... {

... $sort: {"salary": -1}

... },

... {

... $project: {

... "name": 1

... }

... },

... {

... $limit: 1

... }

... ])

[

{ \_id: ObjectId('6618ac75cc627b09619f990a'), name: 'Joseph Thampy' }

]

employee> db.employee.aggregate({$count: "id"})

[ { id: 5 } ]

employee> db.employee.aggregate({$count: "total employee"})

[ { 'total employee': 5 } ]

employee> db.employee.aggregate({$project: {"name": 1, "salary": 1}})

[

{

\_id: ObjectId('6618ac75cc627b09619f990a'),

name: 'Joseph Thampy',

salary: 30000

},

{

\_id: ObjectId('6618b000cc627b09619f990b'),

name: 'kavya',

salary: 20000

},

{

\_id: ObjectId('6618b000cc627b09619f990d'),

name: 'Lakshmi Prasannan',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990e'),

name: 'Merin Babu',

salary: 20000

},

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

name: 'ABC',

salary: 20000

}

]

employee> db.employee.Find({name: "ABC"})

TypeError: db.employee.Find is not a function

employee> db.employee.find({name: "ABC"})

[

{

\_id: ObjectId('6618b0f4cc627b09619f990f'),

id: 106,

name: 'ABC',

street: [ 'MNO House', 'APS' ],

city: 'Kottayam',

Department: 'Security',

salary: 20000

}

]

employee>

New Ques on 15/04/2024

test> use manu

switched to db manu

manu> db.createCollection("student")

{ ok: 1 }

manu> db.student.insertMany([

... {

... rno : 37,

... name : "Thampy",

... name : "Thampy",

manu> db.student.insertMany([

... {

... rno : 37,

... name : "Joseph Thampy",

... class : "MCA"

... },

... {

... rno : 38,

... name : "Kavya",

... class : "BCA"

... },

... {

... rno : 39,

... name : "Keerthana",

... class : "B.Tech"

... }

... rno : 40,

Uncaught:

SyntaxError: Unexpected token, expected "," (17:0)

15 | class : "B.Tech"

16 | }

> 17 | rno : 40,

| ^

18 |

manu> db.student.insertMany([ { rno : 37, name : "Joseph Thampy", class : "MCA" }, { rno : 38, name : "Kavya", class : "BCA" }, { rno : 39, name : "Keerthana", class : "B.Tech" }, {rno : 40, name : "Lakshmi Prasannan", class : "M.Tech" },{ rno : 41, name : "Manu Jose", class : "Cyber Security" }, { rno : 42, name : "Merin Babu", class : "MCA" } ] )

{

acknowledged: true,

insertedIds: {

'0': ObjectId('661cec8c80838e30949f990a'),

'1': ObjectId('661cec8c80838e30949f990b'),

'2': ObjectId('661cec8c80838e30949f990c'),

'3': ObjectId('661cec8c80838e30949f990d'),

'4': ObjectId('661cec8c80838e30949f990e'),

'5': ObjectId('661cec8c80838e30949f990f')

}

}

manu> db.student.find()

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990e'),

rno: 41,

name: 'Manu Jose',

class: 'Cyber Security'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.updateOne(

... {name : "Keerthana"},

... {$set : {name : "Keerthana KB"}}

... )

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

manu> db.student.find()

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

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

[ },

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',na KB',

class: 'BCA'ch'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',nan',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'ecurity'

},

{

\_id: ObjectId('661cec8c80838e30949f990e'),

rno: 41,

name: 'Manu Jose',,

class: 'Cyber Security'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.find()

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990e'),

rno: 41,

name: 'Manu Jose',

class: 'Cyber Security'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.deleteOne(

... {

... rno : 41

... }

... )

{ acknowledged: true, deletedCount: 1 }

manu> db.student.find()

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.updateOne(

... {

... name : "Joseph Thampy"

... },

... {

... $set : { Add : "MP

Uncaught:

SyntaxError: Unterminated string constant. (6:15)

4 | },

5 | {

> 6 | $set : { Add : "MP

| ^

7 |

manu> db.student.updateOne( { name : "Joseph Thampy" }, { $set : { Add : "MP" } } )

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

manu> db.student.find()

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA',

Add: 'MP'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.find(

... $and : [ { rno : 1 }, { name : "Kavya" } ]

Uncaught:

SyntaxError: Unexpected token, expected "," (2:5)

1 | db.student.find(

> 2 | $and : [ { rno : 1 }, { name : "Kavya" } ]

| ^

3 |

manu> db.student.find( { $and : [ { rno : 1 }, { name : "Kavya" } ] } )

manu> db.student.find( { $and : [ { rno : 2 }, { name : "Kavya" } ] } )

manu> db.student.find( { $and : [ { rno : 38 }, { name : "Kavya" } ] } )

[

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

}

]

manu> db.student.find(

... {

... $or : [ { rno : 37 }, { name : "Kavya" } ]

... }

... )

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA',

Add: 'MP'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

}

]

manu> db.student.find( { $nor: [{ rno: 37 }, { name: "Kavya" }] } )

[

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.find(

... {

... rno : { $eq : 40 }

... }

... )

[

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

}

]

manu> db.student.find( { rno: { $ne : 40 } } )

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA',

Add: 'MP'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.find( { rno: { $lt : 40 } } )

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA',

Add: 'MP'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

}

]

manu> db.student.find( { rno: { $gt : 40 } } )

[

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

manu> db.student.find( { rno: { $lte : 40 } } )

[

{

\_id: ObjectId('661cec8c80838e30949f990a'),

rno: 37,

name: 'Joseph Thampy',

class: 'MCA',

Add: 'MP'

},

{

\_id: ObjectId('661cec8c80838e30949f990b'),

rno: 38,

name: 'Kavya',

class: 'BCA'

},

{

\_id: ObjectId('661cec8c80838e30949f990c'),

rno: 39,

name: 'Keerthana KB',

class: 'B.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

}

]

manu> db.student.find( { rno: { $gte : 40 } } )

[

{

\_id: ObjectId('661cec8c80838e30949f990d'),

rno: 40,

name: 'Lakshmi Prasannan',

class: 'M.Tech'

},

{

\_id: ObjectId('661cec8c80838e30949f990f'),

rno: 42,

name: 'Merin Babu',

class: 'MCA'

}

]

--------------------------------------22 April 2024-------------------------------------------------------

test> use comp

switched to db comp

comp> db.createCollection(website)

ReferenceError: website is not defined

comp> db.createCollection("website")

{ ok: 1 }

comp> db.insertMany(

... [

... {

... roll : 1,

... name : Amit

... amt : 1000

Uncaught:

SyntaxError: Unexpected token, expected "," (6:0)

4 | roll : 1,

5 | name : Amit

> 6 | amt : 1000

| ^

7 |

comp> db.insertMany(

... [

... {

... roll : 1,

... name : "Amit",

... amt : 1000,

... url : "www.amit.com"

... },

... {

... roll : 2,

... name : "Mukul",

... amt : 2000,

... url : "www.mukul.com"

... },

... {

... roll : 3,

... name : "Judin",

... amt : 3000,

... url : "www.judin.com"

... },

... {

... roll : 4,

... name : "Shibith",

... amt : 4000,

... url : "www.shibith.com"

... },

... {

... roll : 5,

... name : "Jisto",

... amt : 5000,

... url : "www.jisto.com"

... },

... {

... roll : 6,

... name : "Tins"

... ,

... amt : 6000,

... url : "www.tins.com"

... }

... ]

... )

TypeError: db.insertMany is not a function

comp> db.website.insertMany( [ { roll: 1, name: "Amit", amt: 1000, url: "www.amit.com" }, { roll: 2, name: "Mukul", amt: 2000, url: "www.mukul.com" }, { roll: 3, name: "Judin", amt: 3000, url: "www.judin.com" }, { roll: 4, name: "Shibith", amt: 4000, url: "www.shibith.com" }, { roll: 5, name: "Jisto", amt: 5000, url: "www.jisto.com" }, { roll: 6, name: "Tins", amt: 6000, url: "www.tins.com" }] )

{

acknowledged: true,

insertedIds: {

'0': ObjectId('66262f1d7a4bc9ebfa9f990a'),

'1': ObjectId('66262f1d7a4bc9ebfa9f990b'),

'2': ObjectId('66262f1d7a4bc9ebfa9f990c'),

'3': ObjectId('66262f1d7a4bc9ebfa9f990d'),

'4': ObjectId('66262f1d7a4bc9ebfa9f990e'),

'5': ObjectId('66262f1d7a4bc9ebfa9f990f')

}

}

comp> db.website.findAll()

TypeError: db.website.findAll is not a function

comp> db.website.find()

[

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990a'),

roll: 1,

name: 'Amit',

amt: 1000,

url: 'www.amit.com'

},

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990b'),

roll: 2,

name: 'Mukul',

amt: 2000,

url: 'www.mukul.com'

},

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990c'),

roll: 3,

name: 'Judin',

amt: 3000,

url: 'www.judin.com'

},

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990d'),

roll: 4,

name: 'Shibith',

amt: 4000,

url: 'www.shibith.com'

},

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990e'),

roll: 5,

name: 'Jisto',

amt: 5000,

url: 'www.jisto.com'

},

{

\_id: ObjectId('66262f1d7a4bc9ebfa9f990f'),

roll: 6,

name: 'Tins',

amt: 6000,

url: 'www.tins.com'

}

]

comp> db.website.aggregate(

... {

... $group :

... { "Total\_Amt" : { $sum : $amt } }

... }

... )

ReferenceError: $amt is not defined

comp> db.website.aggregate( { $group: { "Total\_Amt": { $sum: "$amt" } } } )

MongoServerError[Location15955]: a group specification must include an \_id

comp> db.website.aggregate( { $group: { id : "Website", "Total\_Amt": { $sum: "$amt" } } } )

MongoServerError[Location40234]: The field 'id' must be an accumulator object

comp> db.website.aggregate( { $group: { id : "$name", "Total\_Amt": { $sum: "$amt" } } } )

MongoServerError[Location40234]: The field 'id' must be an accumulator object

comp> db.website.aggregate( { $group: { \_id : "Website", "Total\_Amt": { $sum: "$amt" } } } )

[ { \_id: 'Website', Total\_Amt: 21000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Average" : { $sum : "$amt" } } } )

[ { \_id: 'Total', Average: 21000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Minimum" : { $min : "$amt" } } } )

[ { \_id: 'Total', Minimum: 1000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Maximum" : { $max : "$amt" } } } )

[ { \_id: 'Total', Maximum: 6000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "First" : { $first : "$amt" } } } )

[ { \_id: 'Total', First: 1000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Last" : { $last : "$amt" } } } )

[ { \_id: 'Total', Last: 6000 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Push" : { $push : "$amt" } } } )

[ { \_id: 'Total', Push: [ 1000, 2000, 3000, 4000, 5000, 6000 ] } ]

comp> db.website.aggregate( { $group : { \_id : "$name", "Push" : { $push : "$amt" } } } )

[

{ \_id: 'Mukul', Push: [ 2000 ] },

{ \_id: 'Shibith', Push: [ 4000 ] },

{ \_id: 'Judin', Push: [ 3000 ] },

{ \_id: 'Tins', Push: [ 6000 ] },

{ \_id: 'Amit', Push: [ 1000 ] },

{ \_id: 'Jisto', Push: [ 5000 ] }

]

comp> db.website.aggregate( { $group : { \_id : "Total", "Count" : { $count : "$amt" } } } )

comp>

comp> db.website.aggregate( { $group : { \_id : "Total", "Count" : { $count : } } } )

Uncaught:

SyntaxError: Unexpected token (1:72)

> 1 | db.website.aggregate( { $group : { \_id : "Total", "Count" : { $count : } } } )

| ^

2 |

comp> db.website.aggregate( { $group : { \_id : "Total", "Count" : $count } } )

ReferenceError: $count is not defined

comp> db.website.aggregate( { $group : { \_id : "Total", "Count" : {$count} } } )

ReferenceError: $count is not defined

comp> db.website.aggregate( { $group : {$count: "Total Count"} } )

MongoServerError[Location40234]: The field '$count' must be an accumulator object

comp> db.website.aggregate( {$count: "Total Count"} )

[ { 'Total Count': 6 } ]

comp> db.website.aggregate( { $group : { \_id : "Total", "Add To Set" : { $addToSet : "$amt" } } } )

[

{

\_id: 'Total',

'Add To Set': [ 3000, 2000, 4000, 6000, 1000, 5000 ]

}

]

comp> db.website.aggregate( { $group : { \_id : "$name", "Add To Set" : { $addToSet : "$amt" } } } )

[

{ \_id: 'Tins', 'Add To Set': [ 6000 ] },

{ \_id: 'Judin', 'Add To Set': [ 3000 ] },

{ \_id: 'Mukul', 'Add To Set': [ 2000 ] },

{ \_id: 'Shibith', 'Add To Set': [ 4000 ] },

{ \_id: 'Jisto', 'Add To Set': [ 5000 ] },

{ \_id: 'Amit', 'Add To Set': [ 1000 ] }

]

comp>