C:\Users\Lenovo>mongosh

Current Mongosh Log ID: 64a523caba483ca969b56d24

Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.10.1

Using MongoDB: 6.0.7

Using Mongosh: 1.10.1

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

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).

You can opt-out by running the disableTelemetry() command.

------

The server generated these startup warnings when booting

2023-07-05T13:19:51.918+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

------

test> show db

MongoshInvalidInputError: [COMMON-10001] 'db' is not a valid argument for "show".

test> show dbs

admin 40.00 KiB

config 60.00 KiB

local 40.00 KiB

test> use zarana

switched to db zarana

zarana> show db

MongoshInvalidInputError: [COMMON-10001] 'db' is not a valid argument for "show".

zarana> db.table1.insertOne({"name":"zarana",age:29})

{

acknowledged: true,

insertedId: ObjectId("64a52526ba483ca969b56d25")

}

zarana> db.table1.insertOne({"name":"ankita",age:32})

{

acknowledged: true,

insertedId: ObjectId("64a52537ba483ca969b56d26")

}

zarana> show collections

table1

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

}

]

zarana> db.table1.insertMany([{"name":"ankita",age:32},{name:"KHP",age:32,contact:58589652},{fname:"ZKG",age:20}]

... )

{

acknowledged: true,

insertedIds: {

'0': ObjectId("64a525ecba483ca969b56d27"),

'1': ObjectId("64a525ecba483ca969b56d28"),

'2': ObjectId("64a525ecba483ca969b56d29")

}

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 }

]

zarana>

(To exit, press Ctrl+C again or Ctrl+D or type .exit)

------

The server generated these startup warnings when booting

2023-07-05T13:19:51.918+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

------rs\Lenovo>mongosh

Current Mongosh Log ID: 64a52699843bb0ea482e23d8

test> use zarana

switched to db zarana

zarana> db.table1.find({name:"ankita"})

[sing Mongosh: 1.10.1

{

\_id: ObjectId("64a52537ba483ca969b56d26"),/mongodb-shell/

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

}

]

zarana> db.table1.find({name:"ankita"},{name:true})

[

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'ankita' },

{ \_id: ObjectId("64a525ecba483ca969b56d27"), name: 'ankita' }

]

zarana> db.table1.find({name:"ankita"},{name:true,\_id:false})

[ { name: 'ankita' }, { name: 'ankita' } ]

zarana> db.table1.find({name:"ankita"})

[

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

}

]

zarana> db.table1.find({name:"ankita"}).limit(1)

[

{

zarana> db.table1.find()

[ name: 'ankita',

{ age: 32

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29table1.find({name:"ankita"}).limit(2)

},

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',"64a52537ba483ca969b56d26"), name: 'ankita', age: 32 }

age: 32,able1.find({name:"ankita"}).limit(1).skip(1)

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{ name: 'ankita',

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

},na> save

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> db.table1.find()

[ \_id: ObjectId("64a52537ba483ca969b56d26"),

{ name: 'ankita',

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},\_id: ObjectId("64a525ecba483ca969b56d27"),

{ name: 'ankita',

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

},na> db.table1.insertOne({"name":"ankita",age:33})

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',tId("64a52d39843bb0ea482e23d9")

age: 32

},na> db.table1.find({$or:[{name:"ankita"},{age:33}]})

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',d("64a52537ba483ca969b56d26"),

age: 32,nkita',

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{ \_id: ObjectId("64a525ecba483ca969b56d27"),

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

] age: 33

zarana> db.table1.find({name:"ZKG"},{age:20})

[

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), age: 20 }

] { \_id: ObjectId("64a52537ba483ca969b56d26"), age: 32 },

zarana> db.table1.find({name:"ZKG",age:20})

[ { \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 } ]

zarana> db.table1.find({$or:[{name:"ankita",name:"ZKG"}]})

[arana> db.table1.insertOne({"name":"ZKG",age:30})

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

] insertedId: ObjectId("64a52db2843bb0ea482e23da")

zarana> db.table1.find({$or:[{name:"ankita"},{name:"ZKG"}]})

[arana> db.table1.insertOne({"name":"ZKG",age:20})

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',tId("64a52db8843bb0ea482e23db")

age: 32

},na> db.table1.find({name:"ZKG"})

{

\_id: ObjectId("64a525ecba483ca969b56d27"), name: 'ZKG', age: 30 },

name: 'ankita',64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

age: 32

},na>

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> save 1.txt

Uncaught:

SyntaxError: Identifier directly after number. (1:7)

> 1 | save 1.txt

| ^

2 |

zarana> db.table1.find({$or:[{name:"ankita"},{name:"ZKG"}]})

[

{

\_id: ObjectId("64a52537ba483ca969b56d26"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> db.table1.find({name:"ankita"},{name:"ZKG"})

[

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'ZKG' },

{ \_id: ObjectId("64a525ecba483ca969b56d27"), name: 'ZKG' },

{ \_id: ObjectId("64a52d39843bb0ea482e23d9"), name: 'ZKG' }

]

zarana> db.table1.find({$and:[{name:"ankita"},{age:33}]})

[

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

}

]

zarana> db.table1.update

db.table1.updateMany db.table1.updateOne db.table1.updateSearchIndex

zarana> db.table1.update

db.table1.updateMany db.table1.updateOne db.table1.updateSearchIndex

zarana> db.table1.updateOne({name:"ankita"},{$set:{name:"nidhi"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'ankita',

age: 33

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> db.table1.updateOne({name:"ankita",age:33},{$set:{name:"nidhi"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{ \_id: ObjectId("64a52d39843bb0ea482e23d9"), name: 'nidhi', age: 33 },

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> db.table1.updateOne({name:"nidhi",age:33},{$set:{name:"nidhi1",age:11}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'nidhi1',

age: 11

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 20 }

]

zarana> db.table1.updateMany({name:"ZKG"},{$set:{name:"ZKG1"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 2,

modifiedCount: 2,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'nidhi1',

age: 11

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG1', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG1', age: 20 }

]

zarana> db.table1.updateMany({name:"ZKG1"},{$set:{name:"ZKG2",age:10}})

{

acknowledged: true,

insertedId: null,

matchedCount: 2,

modifiedCount: 2,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'nidhi1',

age: 11

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG2', age: 10 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG2', age: 10 }

]

zarana> db.table1.updateMany({name:"ZKG2",age:10},{$set:{name:"ZKG",age:30}})

{

acknowledged: true,

insertedId: null,

matchedCount: 2,

modifiedCount: 2,

upsertedCount: 0

}

zarana> db.table1.find()

[

{

\_id: ObjectId("64a52526ba483ca969b56d25"),

name: 'zarana',

age: 29

},

{ \_id: ObjectId("64a52537ba483ca969b56d26"), name: 'nidhi', age: 32 },

{

\_id: ObjectId("64a525ecba483ca969b56d27"),

name: 'ankita',

age: 32

},

{

\_id: ObjectId("64a525ecba483ca969b56d28"),

name: 'KHP',

age: 32,

contact: 58589652

},

{ \_id: ObjectId("64a525ecba483ca969b56d29"), fname: 'ZKG', age: 20 },

{

\_id: ObjectId("64a52d39843bb0ea482e23d9"),

name: 'nidhi1',

age: 11

},

{ \_id: ObjectId("64a52db2843bb0ea482e23da"), name: 'ZKG', age: 30 },

{ \_id: ObjectId("64a52db8843bb0ea482e23db"), name: 'ZKG', age: 30 }

]

zarana> db.table1.updateMany({name:"ZKG2",age:10},{$set:{name:"ZKG",age:30}})

{

acknowledged: true,

insertedId: null,

matchedCount: 0,

modifiedCount: 0,

upsertedCount: 0

}

zarana> db.table1.updateOne({name:"nidhi1"},{name:"nidhi2",age:11}})

Uncaught:

SyntaxError: Unexpected token, expected "," (1:58)

> 1 | db.table1.updateOne({name:"nidhi1"},{name:"nidhi2",age:11}})

| ^

2 |

zarana> db.table1.updateOne({name:"nidhi1"},{name:"nidhi2",age:11})

MongoInvalidArgumentError: Update document requires atomic operators

zarana>

**6/7/2023**

1. db.table1.updateOne({age:45},{$set:{name:"PQR"}},{upsert:true})
2. db.table1.updateMany({},{$inc:{age:10}});
3. db.table1.updateOne({},{$inc:{age:15}});
4. db.createCollection("Employee")
5. db.Employee.drop()
6. db.table1.updateOne({},{$inc:{age:-15}});
7. db.table1.updateOne({},{$mul:{age:15}});
8. db.table1.updateOne({},{$mul:{age:0.5}});
9. db.table1.updateOne({},{$mul:{age:1/8}});
10. db.table1.find({age:{$gt:25}})
11. db.table1.find({age:{$gt:25,$lt:50}})
12. db.table1.find({$and:[{age:{$gt:25}},{age:{$lt:50}}]})
13. db.table1.updateOne({},{$rename:{"age":"aged"}})
14. db.table1.find({age:{$in:[10,20,30]}})
15. db.table1.updateOne({},{$set:{join\_date:new Date()}})
16. db.table1.updateMany({},{$unset:{join\_date:""}})
17. db.table1.find({name:"ZKG"},{age:1,\_id:0})
18. db.table1.find({name:{$ne:"ZKG"}})
19. db.table1.find({name:/Z/})
20. db.table1.find({name:/^Z/})
21. db.table1.find({name:/Z[A-z]{2}/})
22. db.table1.find({name:/Z[A-z]{2}$/})
23. db.table1.find().sort({age:1})
24. db.table1.find().sort({age:-1})
25. db.table1.count({age:{$gt:30}})
26. db.table1.find().count()
27. db.table1.count()
28. db.table1.count({age:{$gt:40}})
29. db.table1.find({age:{$gt:40}}).count()

Task:

1. db.student1.find({age:{$gt:5}})
2. db.student1.updateMany({},{$inc:{standard:1}})
3. db.student1.find().sort({age:-1})
4. db.student1.find().sort({age:-1}).limit(1)
5. db.student1.deleteMany({standard:12})