BASIC CRUD OPERATIONS IN MONGODB IN GIT-BASH

First open git bash and type mongod and press enter then your mongo server will start now open another tab in git bash and do the code below to perform crud operations

NOTE:-> YOU CAN WRITE THE CODE IN AN EDITOR AND THEN PASTE IT IN THE BASH FOR YOU COMFORT

DELL@Stormbreaker MINGW64 ~

$ mongo

MongoDB shell version v5.0.2

connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb

Implicit session: session { "id" : UUID("b35084dd-bc89-4a98-b43b-3d09ad52c4c1") }

MongoDB server version: 5.0.2

================

Warning: the "mongo" shell has been superseded by "mongosh",

which delivers improved usability and compatibility.The "mongo" shell has been deprecated and will be removed in

an upcoming release.

We recommend you begin using "mongosh".

For installation instructions, see

https://docs.mongodb.com/mongodb-shell/install/

================

help

db.help() help on db methods

db.mycoll.help() help on collection methods

sh.help() sharding helpers

rs.help() replica set helpers

help admin administrative help

help connect connecting to a db help

help keys key shortcuts

help misc misc things to know

help mr mapreduce

show dbs show database names

show collections show collections in current database

show users show users in current database

show profile show most recent system.profile entries with time >= 1ms

show logs show the accessible logger names

show log [name] prints out the last segment of log in memory, 'global' is default

use <db\_name> set current database

db.mycoll.find() list objects in collection mycoll

db.mycoll.find( { a : 1 } ) list objects in mycoll where a == 1

it result of the last line evaluated; use to further iterate

DBQuery.shellBatchSize = x set default number of items to display on shell

exit quit the mongo shell

use showDB

switched to db showDB

db

showDB

show collections

db

showDB

db.products.insertOne({\_id: 1, name: "Pen", price: 1.20})

{ "acknowledged" : true, "insertedId" : 1 }

show collections

products

db

showDB

db.products.insertOne({\_id: 2, name: "Pencil", price: 0.80})

{ "acknowledged" : true, "insertedId" : 2 }

show collections

products

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2 }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8 }

db.products.find({name: "Pencil"})

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8 }

db.products.find({price: {$gt: 1}})

{ "\_id" : 1, "name" : "Pen", "price" : 1.2 }

DELL@Stormbreaker MINGW64 ~

$ mongo

MongoDB shell version v5.0.2

connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb

Implicit session: session { "id" : UUID("2ce20e9f-2399-49a7-b2ca-d4b7d80c5f35") }

MongoDB server version: 5.0.2

================

Warning: the "mongo" shell has been superseded by "mongosh",

which delivers improved usability and compatibility.The "mongo" shell has been deprecated and will be removed in

an upcoming release.

We recommend you begin using "mongosh".

For installation instructions, see

https://docs.mongodb.com/mongodb-shell/install/

================

db

test

use showDB

switched to db showDB

show collections

products

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2 }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8 }

db.products.finnnd({\_id: 1}, {name: 1})

uncaught exception: TypeError: db.products.finnnd is not a function :

@(shell):1:1

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2 }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8 }

db.products.find({\_id: 1}, {name: 1})

{ "\_id" : 1, "name" : "Pen" }

db.products.updateOne({\_id: 1}, {$set: {stock: 32}})

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

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2, "stock" : 32 }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8 }

db.products.updateOne({\_id: 2}, {$set: {stock: 12}})

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

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2, "stock" : 32 }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.8, "stock" : 12 }

db.products.deleteOne({\_id: 2})

{ "acknowledged" : true, "deletedCount" : 1 }

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2, "stock" : 32 }

db.products.insert({

\_id: 3,

name: "Rubber",

price: 1.30,

stock: 43,

reviews: [

{

authorName: "Saurabh",

rating: 5,

review: "Best rubber ever!!"

},

{

authorName: "Shivam",

rating: 5,

review: "Awesome rubber!!"

}

]

})

WriteResult({ "nInserted" : 1 })

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2, "stock" : 32 }

{ "\_id" : 3, "name" : "Rubber", "price" : 1.3, "stock" : 43, "reviews" : [ { "authorName" : "Saurabh", "rating" : 5, "review" : "Best rubber ever!!" }, { "authorName" : "Shivam", "rating" : 5, "review" : "Awesome rubber!!" } ] }

db.products.insert({

\_id: 2,

name: "Pencil",

price: 0.82,

stock: 12,

reviews: [

{

authorName: "Princi",

rating: 5,

review: "Cool Pencil!!"

},

{

authorName: "Sinha",

rating: 4,

review: "Nice Pencil!!"

}

]

})

WriteResult({ "nInserted" : 1 })

db.products.find()

{ "\_id" : 1, "name" : "Pen", "price" : 1.2, "stock" : 32 }

{ "\_id" : 3, "name" : "Rubber", "price" : 1.3, "stock" : 43, "reviews" : [ { "authorName" : "Saurabh", "rating" : 5, "review" : "Best rubber ever!!" }, { "authorName" : "Shivam", "rating" : 5, "review" : "Awesome rubber!!" } ] }

{ "\_id" : 2, "name" : "Pencil", "price" : 0.82, "stock" : 12, "reviews" : [ { "authorName" : "Princi", "rating" : 5, "review" : "Cool Pencil!!" }, { "authorName" : "Sinha", "rating" : 4, "review" : "Nice Pencil!!" } ] }