```
use dbName

show dbs

dbName.collectionname.insert({name:"Apple"})

show collections

db.collectionName.insertOne({name:"apple"})

db.collectionName.insertMany([{name:"grapes"},{name:"chocos"}])

//read data

show collections

db.getCollectionsNames()

//find method – to get all documents in a collection find is used

db.collectionName.find()

//pretty method

db.collectionName.find().pretty()

//limit the data , to get only one document

db.collectionName.findOne()

//sort the data

db.collectionName.find().sort({"name":1))  //ascending order

db.collectionName.find().sort({"name":-1})//descending order

//count

db.collectionName.find().count()

//count with condition

db.collectionName.count({name:"Apple"})

db.collectionName.countDocuments()

db.collectionName.countDocuments({name:"Apple"})

db.collectionName.estimatedDocumentCount()

//skip documents

db.collectionName.find().skip(2)

//limit

db.collectionName.find().limit(3)

db.collectionName.find().limit(3).skip(2)

db.collectionName.distinct("name") //gives an array of unique values
```

db.collectionName.find({name:"Apple"}) //find a particular document

use students

db.students.insertMany([{name:"tom",age:12,marks:12,place:"home"},{name:"Jerry",age:13,marks:19,place:"hyderabad"},{name:"Raj",age:18,marks:32,place:"kerala"},{name:"Prashant",age:27,marks:89,place:"meerat"}])

db.students.find()

db.students.find({marks:12}) //match single data

db.students.find({marks:{$in:[12,89]}}) //get multiple data using in operator

db.students.find({name:"Tom",marks:12}) //perform and operation both should match else return null

db.students.find({marks:{$lt:78}}) //get all docs with marks lessthan 78

db.students.find({marks:{$gt:78}}) //get marks greater than 78

db.students.find({$or:[{marks:{$gt:78}},{age:{$lt:26}}]}) // OR condition

update details

db.students.update({name:"Tom"},{$set:{marks:99}})//parameter1-how to find data

parameter2 – what we want to update

//multiple update

```
db.students.update({name:"navdeep"},{$set:{marks:99,age:22}})
```

db.students.updateMany({age:12},{$set:{place:"mumbai",marks:100}}) //update many

replace data , we cannot replace id

db.students.replaceOne({name:"Apple"},{name:"pomodoro",age:19,marks:65,place:"abc"})

deleting data

db.students.deleteOne({name:"pomodoro"})

//delete many

db.students.deleteMany({age:12})

db.students.deleteMany({age:{$gt:20}})

search :

db.students.find({$text:{$search:"ap"}})