1. **To create a database in mongoDB - use <database name>**

use training

>use training

switched to db training

**2. To check your currently selected database, use the command db**

db

>db

training

**3.0 To create the collection (table)**

>db.createCollection("employee")

**3.1 To check list of databases, use the command show dbs**

make sure you have atleast inserted one row or a document in the collection or a table

>db.employee.insert ({"name":"Krishna"})

Now run

>show dbs

local 0.000GB

training 0.000GB

In MongoDB default database is test. If you didn't create any database, then collections will be stored in test database.

**4. If you want to delete database <local>, then dropDatabase() command would be as follows −**

use local

switched to db local

>db.dropDatabase()

>{ "dropped" : "local", "ok" : 1 }

> show dbs

training 0.000GB

**5.If you want to drop a table or a collection  <employee>, then  command would be as follows −**

>use training

switched to db training

show collections

>db.employee.drop()

true

**6. To insert data into MongoDB collection, you need to use MongoDB's insert() or save() method.**

the basic command is >db.COLLECTION\_NAME.insert(document)

> db.employee.insert({

emp\_id : 'Emp100',

fname: 'Sanjay',

lname: 'Garg',

dob : new Date ("1972-06-20"),

doj : new Date ("2016-04-01"),

address : [

{

Block : '2/20',

Street : '4th Avenue',

City : 'Bangalore',

State : 'karnataka',

pin : '560037'

}

]

})

db.employee.insert({

emp\_id : 'Emp002',

fname: 'Rajiv',

lname: 'Kumar',

dob : new Date ("1972-06-20"),

doj : new Date ("2016-04-01"),

address : [

{

Block : '4/801',

Street : '3rd Road',

City : 'Mumbai',

State : 'Maharashtra',

pin : '400005'

}

]

})

db.employee.insert({

emp\_id : 'Emp030',

fname: 'Harish',

lname: 'Shah',

address : [

{

Block : '2/501',

Street : '4th Road',

City : 'Mumbai',

State : 'Maharashtra',

pin : '400025'

}

]

})

**7. To display the contents of the collection.**

>db.COLLECTION\_NAME.find() or db.COLLECTION\_NAME.find().pretty()

db.employee.find().pretty()

>db. employee.find() - display in a unstuctured way

OR

>db.employee.find().pretty() - display in a structured way

OR with where clause

>db.employee.find({'fname':'Rajiv'}).pretty()

db.employee.find({'address.City':'Mumbai'}).pretty()

8. **To update the values in the document**

>db.employee.update({'emp\_id':'Emp100'}, {$set:{'lname':'Shah'}})

>db.employee.update({'emp\_id':'Emp100'}, {$set:{'age':'46'}})

**update a nested data**

db.employee.update ({emp\_id: 'Emp100'}, { '$set': {"address.0.pin" : '560005'} });

db.employee.update ({emp\_id: 'Emp100'}, { '$set': {"address.1.pin" : '560009'} });

9. **To delete the document**

>db.employee.remove({'lname':'Shah'}, 1)

9.1. **To drop a collection**

>db.employee.drop()

**10. To truncate the table**

>db.employee.remove({})

**11. To show the limit number of records and skip the first record**

>db.employee.find().limit(1).skip(1)

>db.employee.find().count()

**12. To sort the data**

>db.employee.find({},{"fname":1}).sort({"fname":-1})