# For Starting MongoDB

1. With basic command
   1. cd C:\Program Files\MongoDB\Server\3.4\bin
   2. enter command => mongod --dbpath D:\Lucky\MongoDBStorage\db (else it will create db in c:\\*\*)
   3. This will start server… now start mongo instance to connect as below
   4. enter command(in diff cmd) => mongo

Mongodb tutorial git =>> <https://github.com/sedouard/mongodb-mva>

1. With more properties per Traversry Media => Start as a Service
   1. Make sure to open CMD in ‘run as administrator’ mode.
   2. cd C:\Program Files\MongoDB\Server\3.4\bin
   3. enter command => mongod –directoryperdb --dbpath D:\Lucky\MongoDBStorage\db –logpath D:\Lucky\MongoDBStorage\log\mongo.log –logappend --rest –install
   4. => net start MongoDB
   5. This will start MongoDB as a service in background.
   6. To start mongo shell => mongo
   7. To stop MongoDB service => net stop MongoDB

# Mongo Shell commands

1. show dbs

This will show all DBs.

1. use <NewDbName>

Will create and switched to DB(NewDbName)

1. db – will show current DB
2. refer <https://docs.mongodb.com/manual/reference/method/db.createUser/> for more commands

db.createUser(

{

user: "lucky",

pwd: "lucky123",

roles: [ "readWrite", "dbAdmin" ]

}

)

1. db.createCollection('customers') => creates collection named ‘customers’
2. show collections
3. db.customers.insert({first\_name:”John”,last\_name:”Sena”});
4. db.customers.find(); => to list
5. Batch insert

db.customers.insert([{first\_name:"John",last\_name:"Sehgal"},{first\_name:"Rema", last\_name:"Sehgal", Gender:"Female"}]);

1. db.customers.find().pretty(); => Nice N Neat
2. To update a record =>
   1. db.customers.update({**matcher**},{**Full\_Object**});

Eg: db.customers.update({first\_name:"Rema"}, {first\_name:"Rema", last\_name:"Sehgal", Gender:"F"});

* 1. db.customers.update({matcher},{**$set**:{partial\_Object}});

eg: db.customers.update({matcher},{$set:{gender:”Female”}});

1. $inc => increase the number

Eg db.customers. update({matcher},{**$inc**:{age:5}}); => will increase the age by 5

1. $unset => to remove the field

Eg db.customers. update({matcher},{**$unset**:{age:1}}); => will remove field age

1. Upsert => if match not found , insert it else update it

db.customers.update({matcher},{**$set**:{partial\_Object}}, {**upsert:true**});

1. $rename => rename field name

Eg db.customers.update({matcher},{**$rename**:{“field\_name”:”New\_fieldName” }});

1. Remove => to delete the element from collection

Eg db.customers.remove({matcher});

To delete only first

db.customers.remove({matcher}, {**justOne:true**});

1. $or

Db.customers.find({**$or**:**[**{first\_name:”sdas”},{last\_name:”asjda”}**]**})

1. $lt => less than. also lte,gt,gte

Eg db.customers.find({age:{$lt:40}});

1. Db.customers.find().**sort({last\_name:1})**; => asc for desc -1
2. Db.customers.find().count();
3. Db.customers.find().limit(4); => will only shows first 4 entries
4. forEach

Eg: db.customers.find().forEach(function(doc){print(“Customer First Name Is:” + doc.first\_name)});