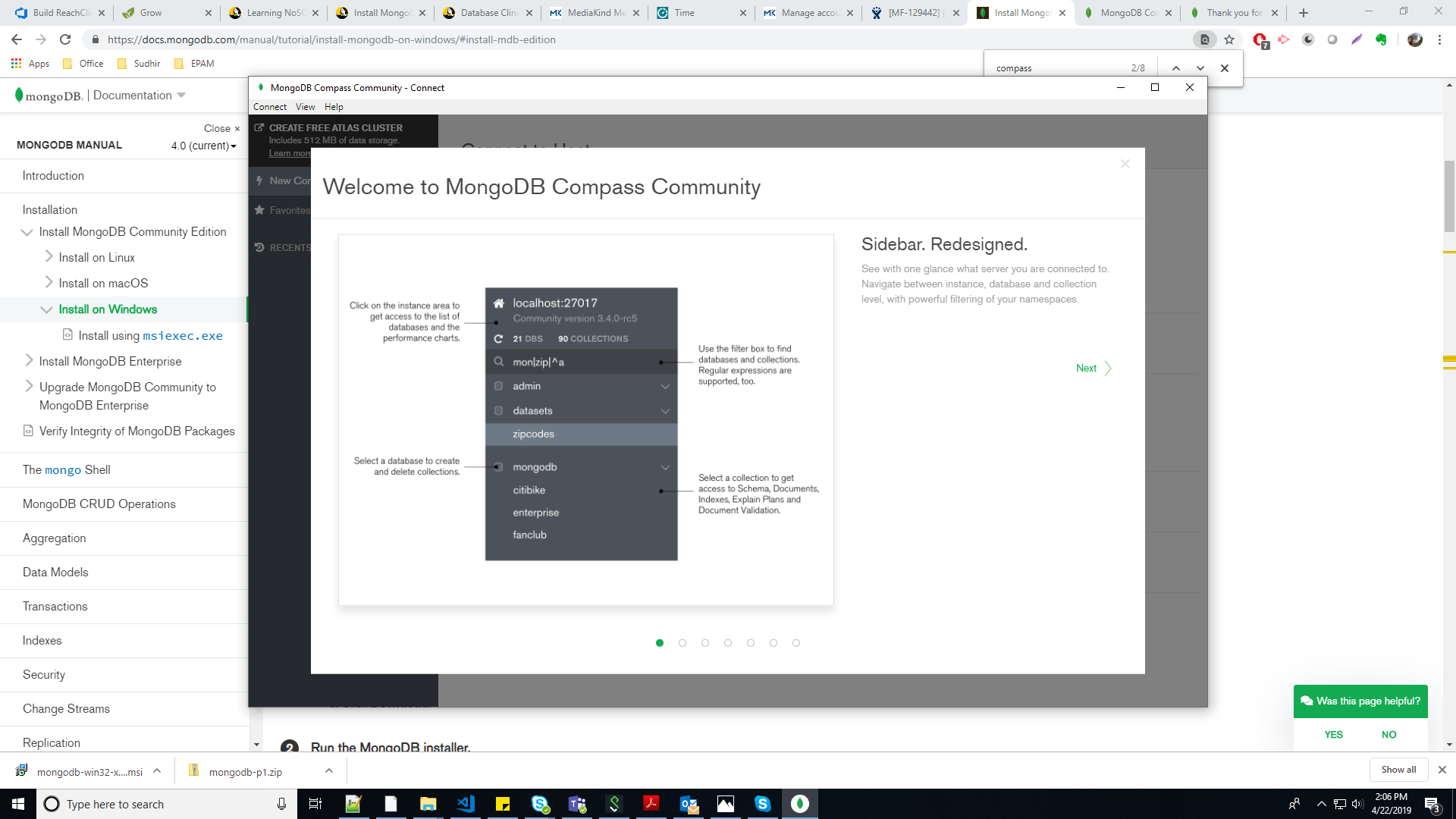
Installed from :

https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/#install-mdb-edition

Skipped /Unchecked Install MongoDb as Service..

Automatically MongoDb Compass will be installed



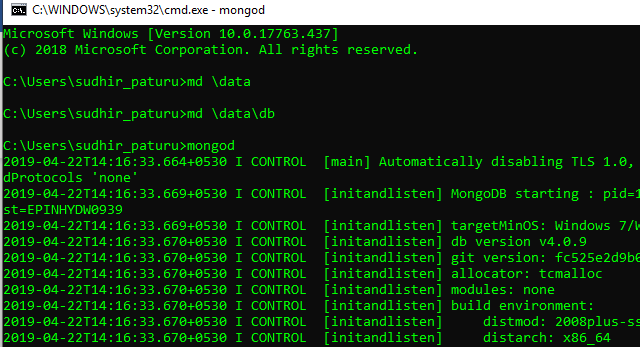
C:\Program Files\MongoDB\Server\4.0\bin

We can find mongo and mongod exe files… after installation

Add to PATH environment variables

C:\Program Files\MongoDB\Server\4.0\bin

Making data Directory for db , mongo going to look for to use store information of database.

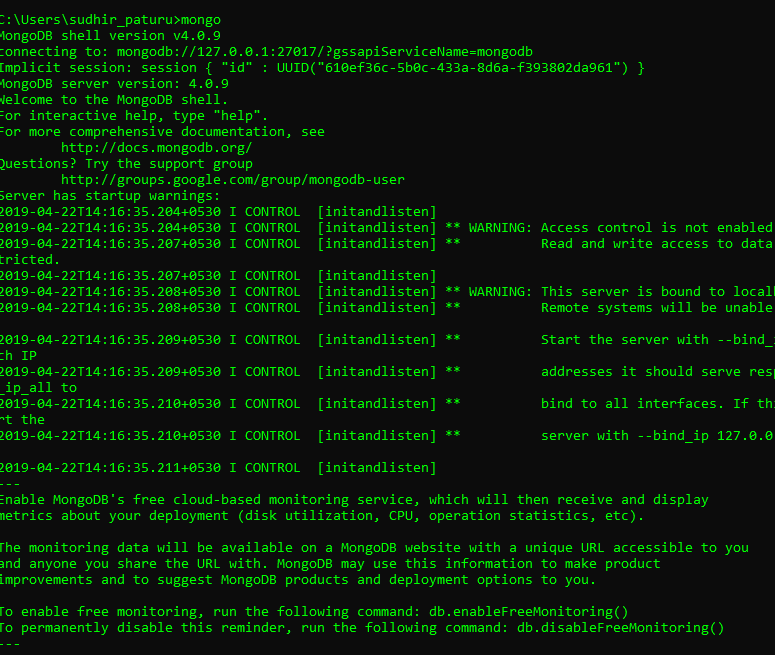


C:\Users\sudhir\_paturu>md \data

C:\Users\sudhir\_paturu>md \data\db

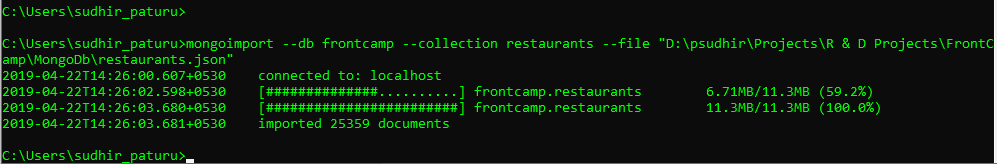
Run mongod 🡪 Starting the mongo daemon .. now we can check for mongo which is up and running

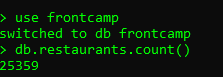
Another command prompt 🡪 “mongo”

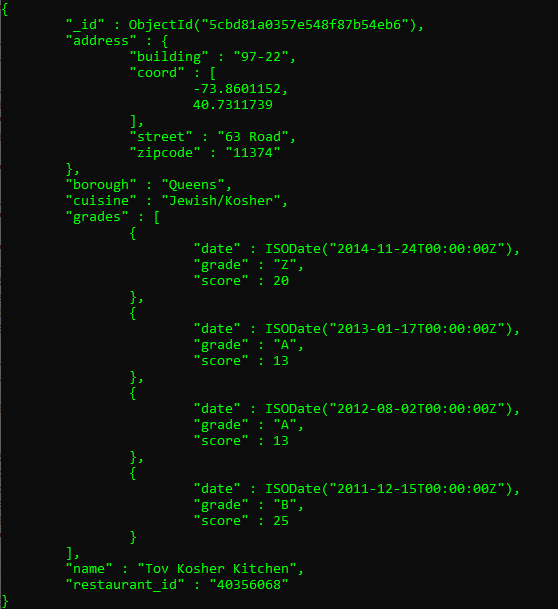


Import Restaurants Collection to Frontend DB.

mongoimport --db frontcamp --collection restaurants --file “D:\psudhir\Projects\R & D Projects\FrontCamp\MongoDb\restaurants.json”







3.1 db.restaurants.aggregate([{$match:{"borough":"Queens","cuisine":"Chinese"}},{$count:"Result"}])



728

db.restaurants.aggregate([

{$unwind:"$grades"},

{

$project:

{

\_id:”$\_id”,

maxp:{$max:"$grades.score"}

}

}])

db.restaurants.aggregate([

{$unwind:"$grades"},

{

$project:

{

\_id:"$\_id",

maxp:{$max:"$grades.score"}

}

}

])

db.restaurants.aggregate([

{$unwind:"$grades"},

{

$group:

{

\_id:null,

max: { $max: "$grades.score"}

}

},

{

$project:

{

max:1,

\_id:1

}

}

])

db.restaurants.aggregate([

{$unwind:"$grades"},

{

$sort:

{

"grades.score":-1

}

},

{

"$limit":1

},

{

$project:

{

\_id:1,

Value: "$grades.score"

}

}

],

{allowDiskUse: true}

)

3.2 db.restaurants.aggregate([ {$unwind:"$grades"}, { $sort: { "grades.score":-1 } }, { "$limit":1 }, { $project: { \_id:1, Value: "$grades.score" } } ], {allowDiskUse: true} )

db.restaurants.update(

{ "borough":"Manhattan" },

{

$push:

{

grades:{

$each:[{grade: "A",score:7,date:ISODate()}]

}

}

}

)

3.3 db.restaurants.updateMany( { "borough":"Manhattan" }, { $push: { grades:{ $each:[{grade: "A",score:7,date:ISODate()}] } } } )

db.restaurants.find({borough:"Manhattan"})

db.restaurants.aggregate(

[

{

$project:

{

index: { $indexOfArray: [ "$grades", 8 ] },

}

}

]

)

db.restaurants.find({ grades : { $slice : [8 , 1] } } )

db.restaurants.aggregate([

{

$project:

{

name: 1,

first: { $arrayElemAt: [ "$grades", 8 ] }

}

}

])

db.restaurants.aggregate([

{

$match:

{

$arrayElemAt: [ "$grades", 8 ]

}

}

])

db.restaurants.find(

{

"grades.8.score": {$lte : 7}

},

{"restaurant\_id" : 1,"name":1,"address":1,"coord":1}

);

3.4 db.restaurants.find( { "grades.8.score": {$lt : 7} }, {"restaurant\_id" : 1,"name":1,"address":1,"grades":1} );

db.restaurants.find( {

"cuisine" : "Seafood",

"grades.grade" :"B",

"date": {

$gt: ISODate("2014-02-01T00:00:00.000Z"),

$lt: ISODate("2014-03-01T00:00:00.000Z")

}

}

)

3.5db.restaurants.find( { "cuisine" : "Seafood", "grades.grade" :"B", "grades.date": { $gte: new ISODate("2014-02-01T00:00:00.000Z"), $lte: new ISODate("2014-03-01T00:00:00.000Z") } } ,{ \_id:1,borough:1} )

db.restaurants.createIndex( { name: 1 })