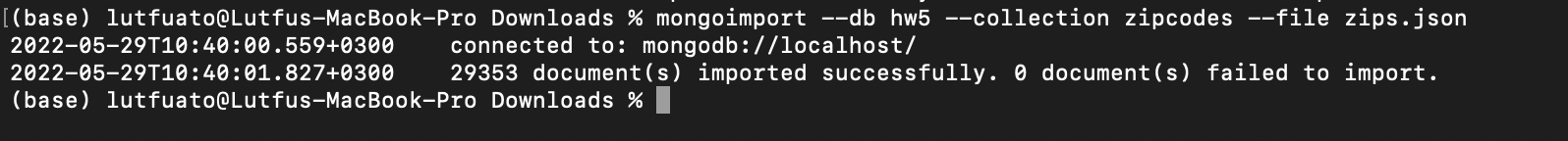
Question -1

use hw5

show collections

db.zipcodes.find().limit(15)



Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

Question – 2

db.zipcodes.find( { pop: { $gt: 30000 },state :"CA" } )

Diagram

Description automatically generated

Question – 3

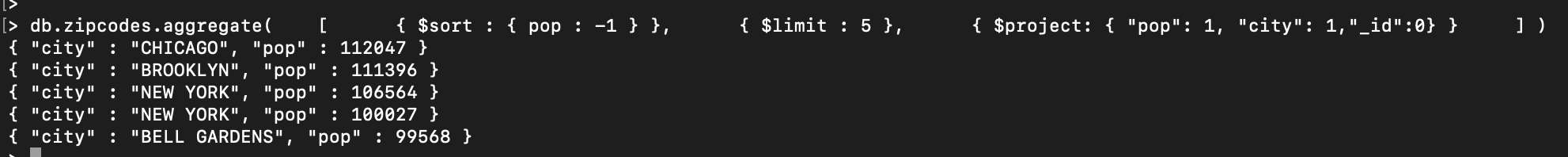
db.zipcodes.find( { $or: [ { state :{$ne : "CA" }}, {"loc.0": { $lt: -120 }, "loc.1": { $gt: 40 }} ] } )

Text

Description automatically generated

Question – 4

db.zipcodes.aggregate( [ { $sort : { pop : -1 } }, { $limit : 5 }, { $project: { "pop": 1, "city": 1,"\_id":0} } ] )



Question – 5

db.zipcodes.aggregate( [ {$group: {\_id:"$state",totalState: {$sum: 1} } } , { $match : { totalState: { $gt : 800}} },{ $sort : { totalState : -1 } } ] )

Text

Description automatically generated

Question – 6

db.createCollection("customers",{validator:{$jsonSchema:{

bsonType:"object",

required:["name","zipcode","avg\_rating"],

properties:{

name:{

bsonType:"string",

description:"mustbeastringandisrequired"

},

zipcode:{

bsonType:"string",

description:"mustbeastringandisrequired"

},

avg\_rating:{

bsonType:["double"],

minimum:0.0,

maximum:10.0,

description:"mustbeadoublein[0.0,10.0]ifthefieldexists"

},

last\_order:{

bsonType:"object",

required:["year"],

properties:{

year:{

bsonType:"int",

description:"mustbeastringifthefieldexists"

},

tags:{bsonType:["array"],minItems:0,maxItems:100,items:{bsonType:["string"],description:"tagsshouldbestringsifexists",}}

}}

}}}})

Text

Description automatically generated

Text

Description automatically generated with medium confidence

Question – 7

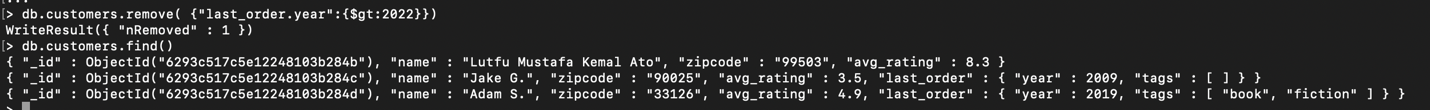
try { db.customers.insertMany( [ {name:"Lutfu Mustafa Kemal Ato", zipcode : "99503", avg\_rating:8.3}, {name:"Jake G.",zipcode:"90025",avg\_rating:3.5, last\_order : {year : NumberInt(2009) ,tags: []} }, {name:"Adam S.",zipcode:"33126",avg\_rating:4.9, last\_order : {year : NumberInt(2019) ,tags: ["book","fiction"]} }, {name:"Harold H.",zipcode:"90010",avg\_rating:1.0, last\_order : {year : NumberInt(3005) ,tags: []} } ] ); } catch (e) { print (e); }

Text

Description automatically generated

Question - 8

db.customers.remove( {"last\_order.year":{$gt:2022}})



question - 9

db.customers.updateOne({ "name" : "Lutfu Mustafa Kemal Ato"}, {$set: {"avg\_rating": 15}})

The response we recieve from the MongoDB is a WriteError response. The reason is that: We set the maximum possible value for avg\_rating to 10, but we are trying to set the value to 15. This causes an error.

Text

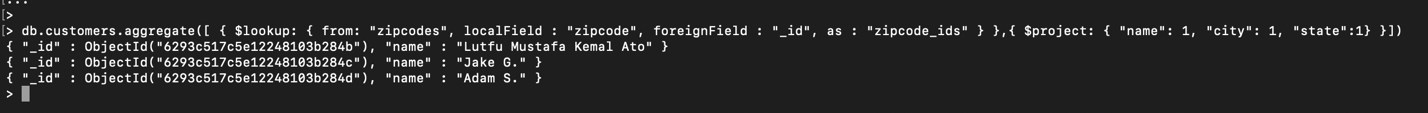
Description automatically generated

Text

Description automatically generated

question - 10

db.customers.aggregate([ { $lookup: { from: "zipcodes", localField : "zipcode", foreignField : "\_id", as : "zipcode\_ids" } },{ $project: { "name": 1, "city": 1, "state":1} }])



Question – 11

mongoexport --db hw5 --collection customers -o customers.json

