Name : Rahul C Shekhar

SRN : PES1201802486

1) Create a replication set of multiple instances on a single machine.

--> Done

------------------------------------------------------------------------------------------------------------------------------------------

2) Create a collection called ‘restaurants’ having the following fields

- Address, name, id, cusine, star, locality, grade which is an embedded document having

date, grade, score.

> db.createCollection("Restaurants")

----------------------------------------------------------------------------------------------------------------------------------------

3) Add 10 records.

> db.Restaurants.insert([

{

\_id : 1,

name : "Swathi Restaurant",

cusine : "Indian",

address : "Bangalore",

star : 3,

locality : "Rajajinagar",

grade : {

date : ISODate(),

grade : "B",

score : 7

}

},

{

\_id : 2,

name : "KFC Restaurant",

cusine : "Fastfood",

address : "Bangalore",

star : 2.5,

locality : "JP Nagar",

grade : {

date : ISODate(),

grade : "C",

score : 6

}

},

{

\_id : 3,

name : "Ching's Restaurant",

cusine : "Chinese",

address : "Bangalore",

star : 3.5,

locality : "Vijaynagar",

grade : {

date : ISODate(),

grade : "B+",

score : 8

}

},

{

\_id : 4,

name : "Dosa Restaurant",

cusine : "Indian",

address : "Bangalore",

star : 3.5,

locality : "Basweshvarnagar",

grade : {

date : ISODate(),

grade : "B+",

score : 8

}

},

{

\_id : 5,

name : "Dominos Restaurant",

cusine : "Fastfood",

address : "Bangalore",

star : 3,

locality : "Vijaynagar",

grade : {

date : ISODate(),

grade : "B",

score : 6

}

},

{

\_id : 6,

name : "Food Camp Restaurant",

cusine : "Indian",

address : "Bangalore",

star : 4,

locality : "Vijaynagar",

grade : {

date : ISODate(),

grade : "B+",

score : 8

}

},

{

\_id : 7,

name : "Akshaya Restaurant",

cusine : "Indian",

address : "Bangalore",

star : 4,

locality : "Vijaynagar",

grade : {

date : ISODate(),

grade : "A",

score : 8.5

}

},

{

\_id : 8,

name : "Sahana Restaurant",

cusine : "Indian",

address : "Bangalore",

star : 3.5,

locality : "Banashankari",

grade : {

date : ISODate(),

grade : "B+",

score : 8

}

},

{

\_id : 9,

name : "Baby Restaurant",

cusine : "Italian",

address : "Bangalore",

star : 4,

locality : "Vijaynagar",

grade : {

date : ISODate(),

grade : "A",

score : 9

}

},

{

\_id : 10,

name : "PES Restaurant",

cusine : "All",

address : "Bangalore",

star : 3.5,

locality : "RR Nagar",

grade : {

date : ISODate(),

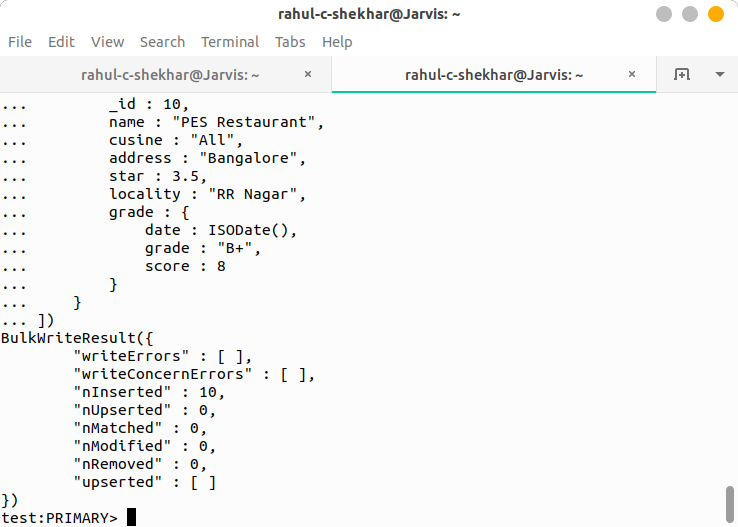
grade : "B+",

score : 8

}

}

])



----------------------------------------------------------------------------------------------------------------------------------------

11) Update 3 documents that have 3.5 star to 4 stars.

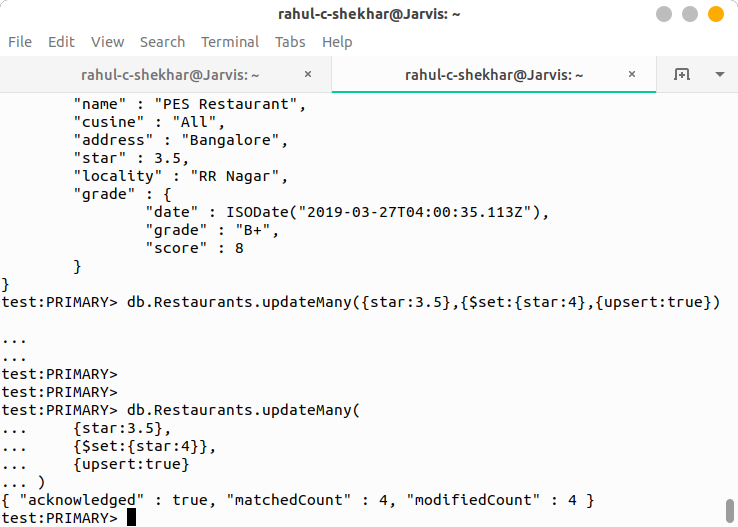
> db.Restaurants.updateMany(

{star:3.5},

{$set:{star:4}},

{upsert:true}

)



----------------------------------------------------------------------------------------------------------------------------------------

12) Update the grade of restaurants that are in Vijaynagar.

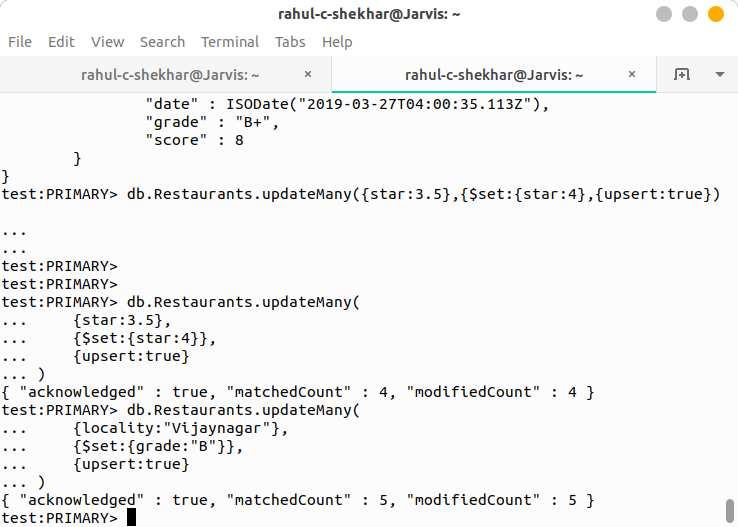
> db.Restaurants.updateMany(

{locality:"Vijaynagar"},

{$set:{grade:"B"}},

{upsert:true}

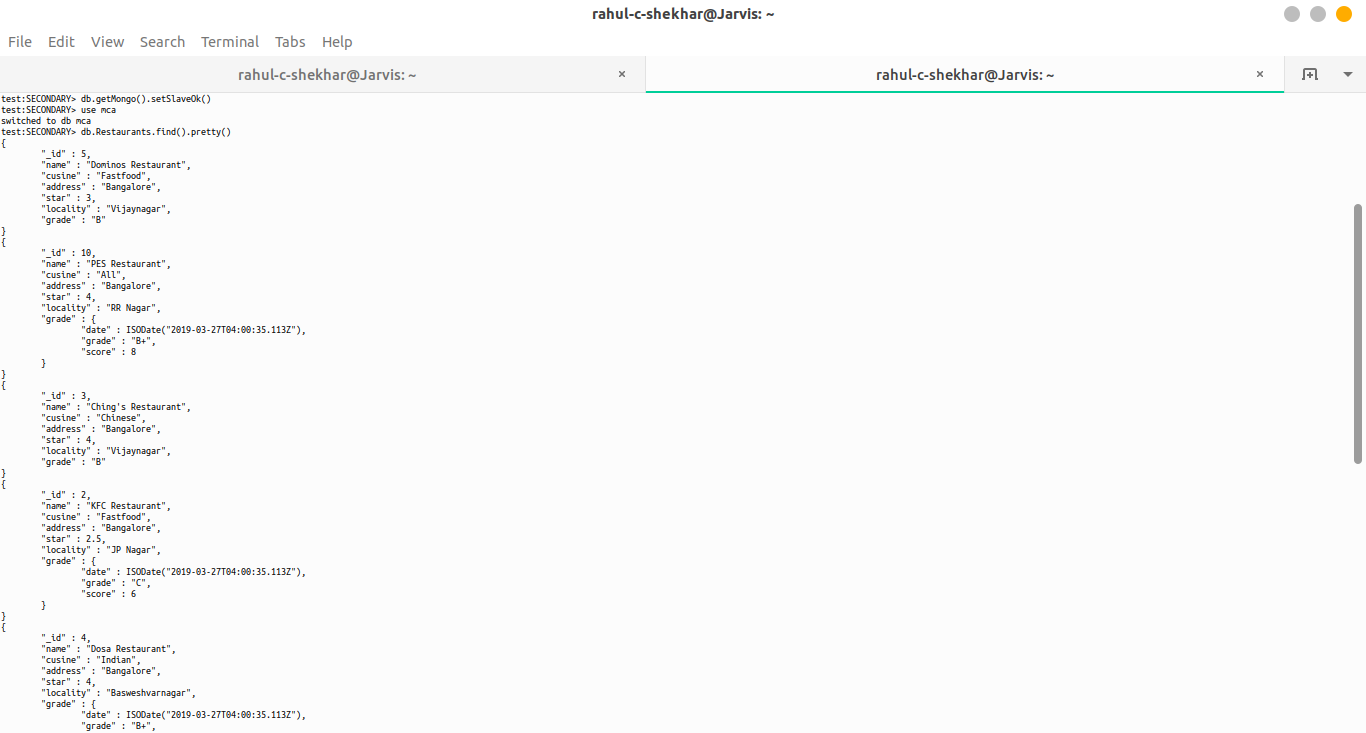
)



----------------------------------------------------------------------------------------------------------------------------------------

4) Write a MongoDB query to display all the documents in the collection restaurants.

> db.Restaurants.find().pretty()



----------------------------------------------------------------------------------------------------------------------------------------

5) Write a MongoDB query to display the fields restaurant\_id, name and cuisine for all the

documents in the collection restaurant.

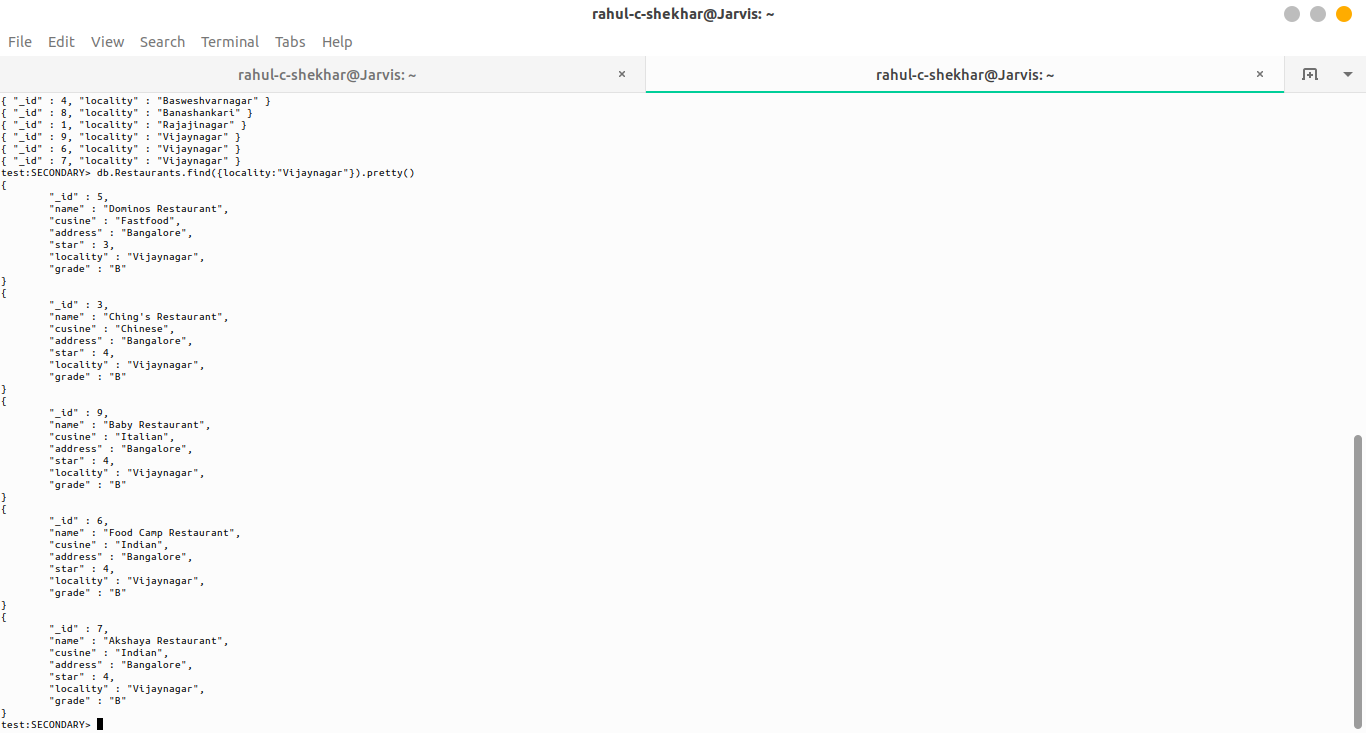
> db.Restaurants.find({},{\_id:1,name:1,cusine:1}).pretty()



----------------------------------------------------------------------------------------------------------------------------------------

6) Write a MongoDB query to display the first 5 restaurant which is in the locality “Vijaynagar”.

> db.Restaurants.find({locality:"Vijaynagar"}).pretty()



----------------------------------------------------------------------------------------------------------------------------------------