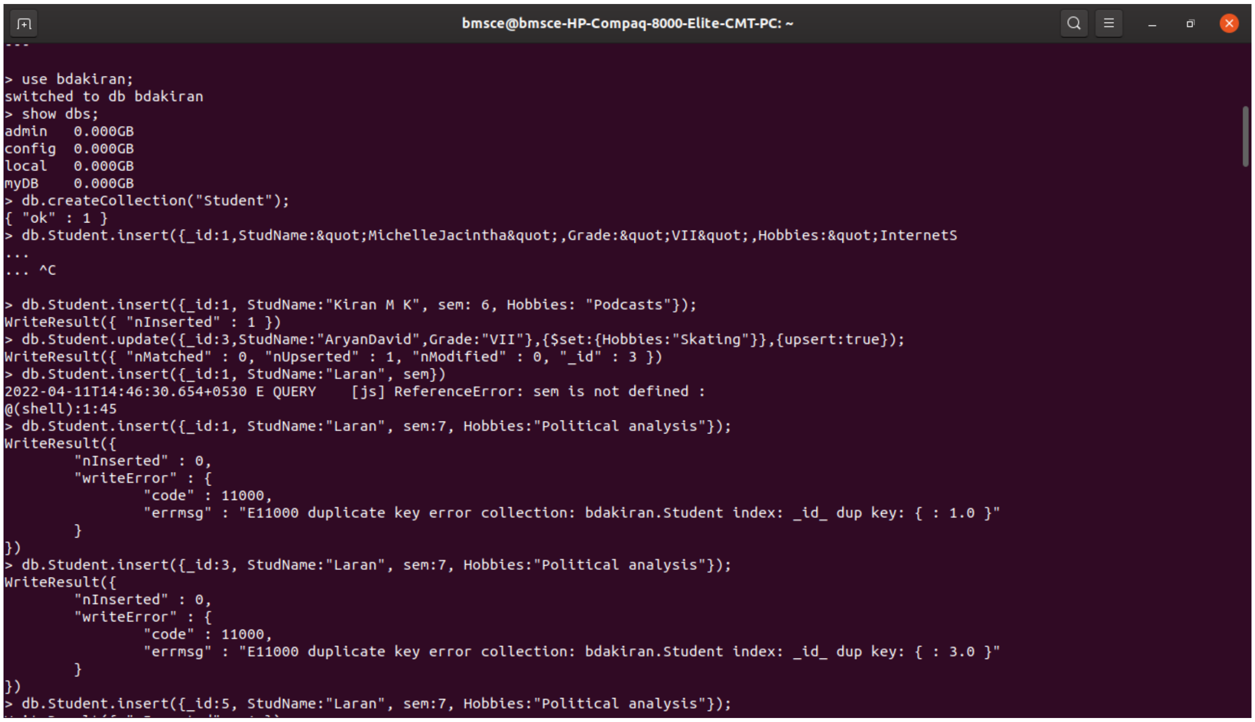
BIG DATA ANALYTICS LAB-1

11-04-2022

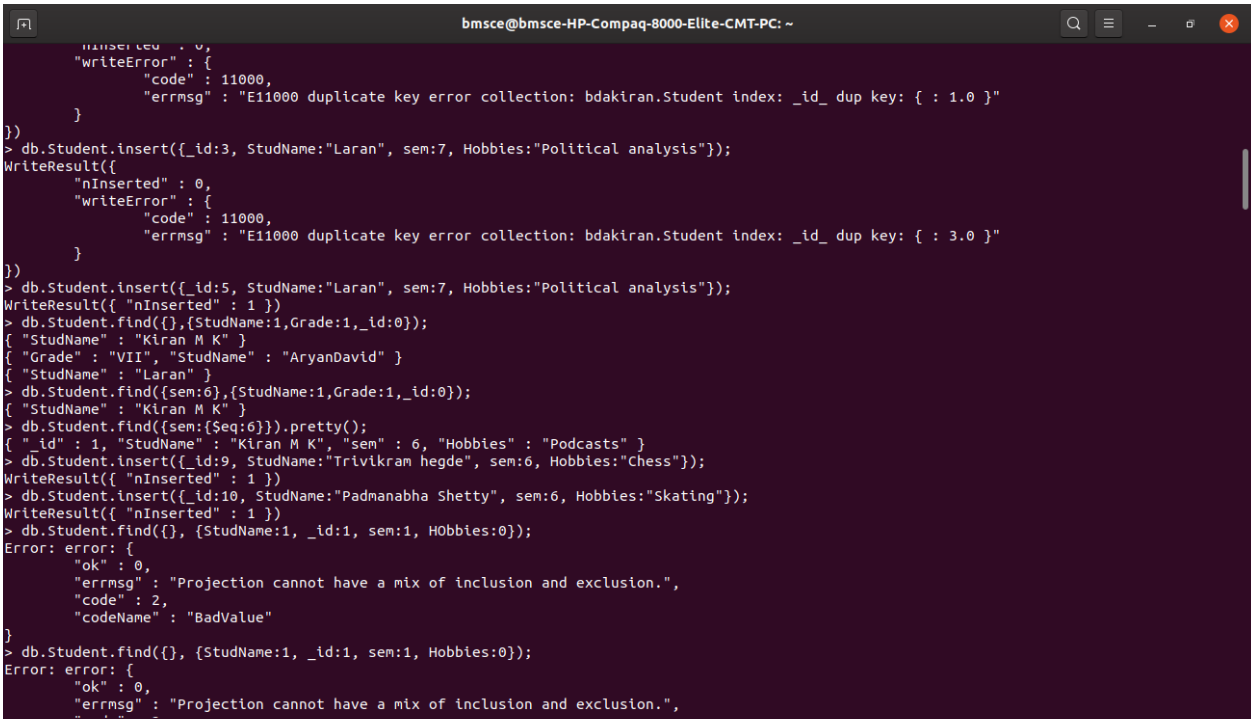
Name: Kiran M K

USN: 1BM19CS073

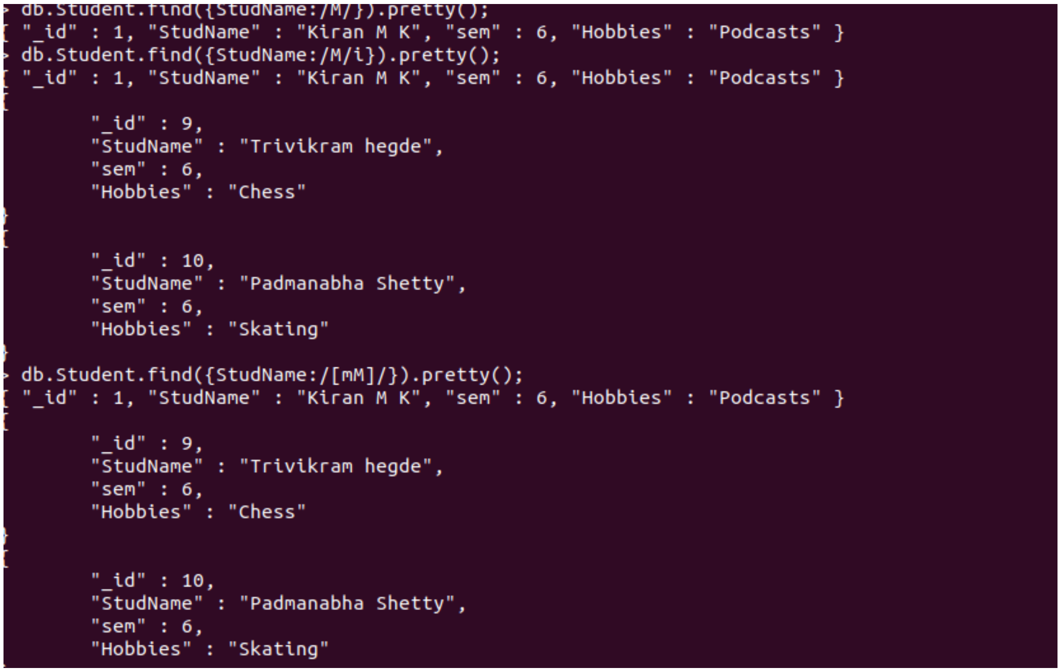
Batch: 6B2



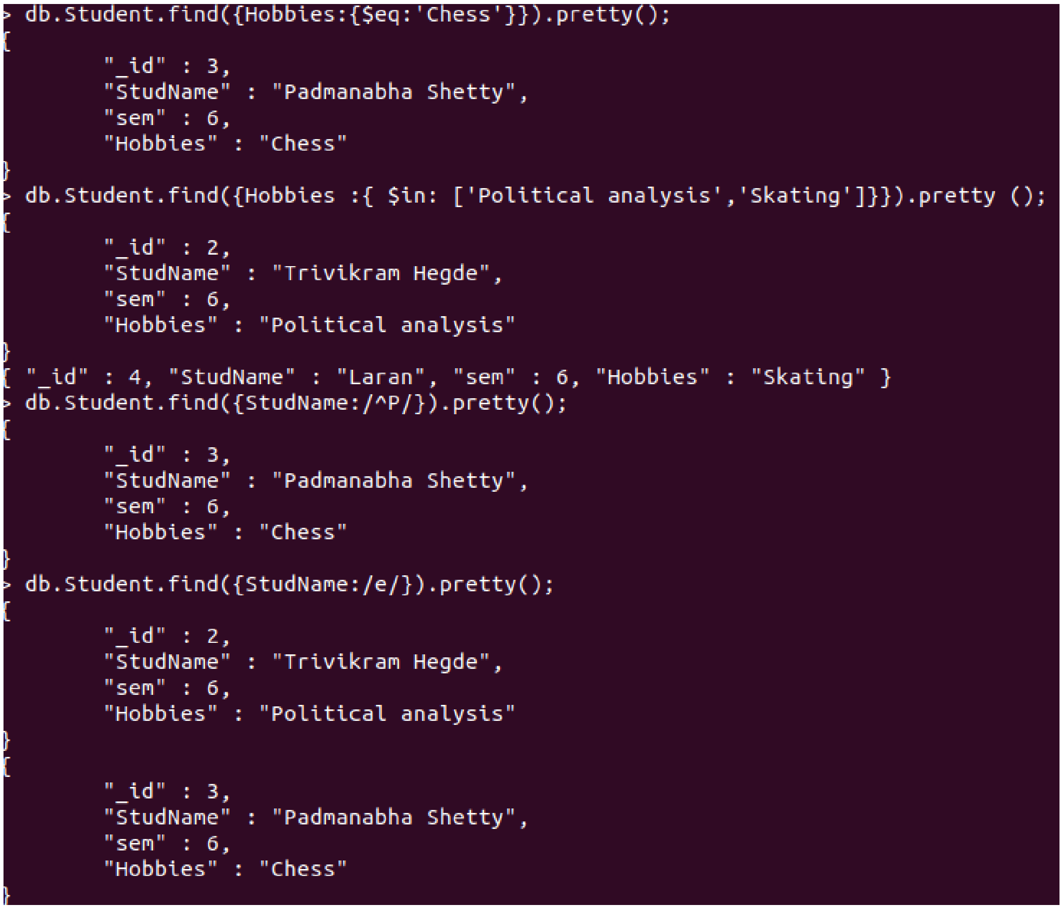
```
---

> use bdakiran;
switched to db bdakiran
> show dbs;
admin   0.000GB
config  0.000GB
local   0.000GB
myDB    0.000GB
> db.createCollection("Student");
{ "ok" : 1 }
> db.Student.insert({_id:1,StudName:&quot;MichelleJacintha&quot;,Grade:&quot;VII&quot;,Hobbies:&quot;InternetS
...
... ^C

> db.Student.insert({_id:1, StudName:"Kiran M K", sem: 6, Hobbies: "Podcasts"});
WriteResult({ "nInserted" : 1 })
> db.Student.update({_id:3,StudName:"AryanDavid",Grade:"VII"},{$set:{Hobbies:"Skating"}},{upsert:true});
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 3 })
> db.Student.insert({_id:1, StudName:"Laran", sem})
2022-04-11T14:46:30.654+0530 E QUERY    [js] ReferenceError: sem is not defined :
@(shell):1:45
> db.Student.insert({_id:1, StudName:"Laran", sem:7, Hobbies:"Political analysis"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: bdakiran.Student index: _id_ dup key: { : 1.0 }"
        }
})
> db.Student.insert({_id:3, StudName:"Laran", sem:7, Hobbies:"Political analysis"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: bdakiran.Student index: _id_ dup key: { : 3.0 }"
        }
})
> db.Student.insert({_id:5, StudName:"Laran", sem:7, Hobbies:"Political analysis"});
```



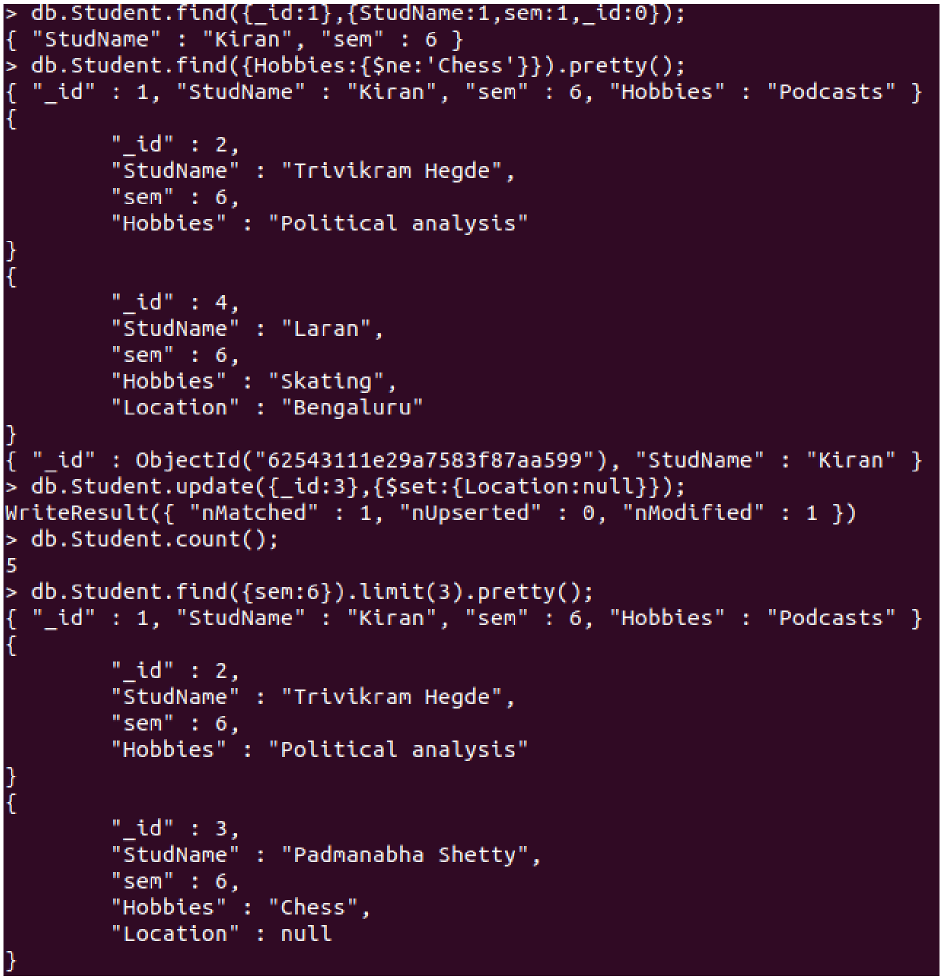
```
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: bdakiran.Student index: _id_ dup key: { : 1.0 }"
        }
})
> db.Student.insert({_id:3, StudName:"Laran", sem:7, Hobbies:"Political analysis"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: bdakiran.Student index: _id_ dup key: { : 3.0 }"
        }
})
> db.Student.insert({_id:5, StudName:"Laran", sem:7, Hobbies:"Political analysis"});
WriteResult({ "nInserted" : 1 })
> db.Student.find({},{StudName:1,Grade:1,_id:0});
{ "StudName" : "Kiran M K" }
{ "Grade" : "VII", "StudName" : "AryanDavid" }
{ "StudName" : "Laran" }
> db.Student.find({sem:6},{StudName:1,Grade:1,_id:0});
{ "StudName" : "Kiran M K" }
> db.Student.find({sem:{$eq:6}}).pretty();
{ "_id" : 1, "StudName" : "Kiran M K", "sem" : 6, "Hobbies" : "Podcasts" }
> db.Student.insert({_id:9, StudName:"Trivikram hegde", sem:6, Hobbies:"Chess"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:10, StudName:"Padmanabha Shetty", sem:6, Hobbies:"Skating"});
WriteResult({ "nInserted" : 1 })
> db.Student.find({}, {StudName:1, _id:1, sem:1, HObbies:0});
Error: error: {
        "ok" : 0,
        "errmsg" : "Projection cannot have a mix of inclusion and exclusion.",
        "code" : 2,
        "codeName" : "BadValue"
}
> db.Student.find({}, {StudName:1, _id:1, sem:1, Hobbies:0});
Error: error: {
        "ok" : 0,
        "errmsg" : "Projection cannot have a mix of inclusion and exclusion.",
```



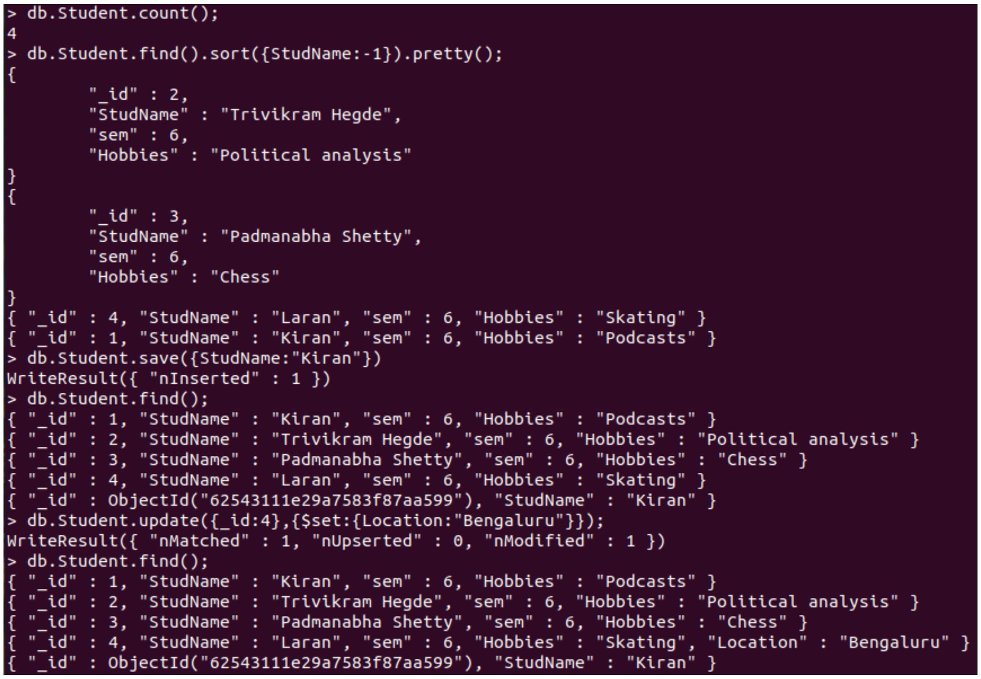
```
> db.Student.find({StudName:/M/}).pretty();
{ "_id" : 1, "StudName" : "Kiran M K", "sem" : 6, "Hobbies" : "Podcasts" }
> db.Student.find({StudName:/M/i}).pretty();
{ "_id" : 1, "StudName" : "Kiran M K", "sem" : 6, "Hobbies" : "Podcasts" }
{
        "_id" : 9,
        "StudName" : "Trivikram hegde",
        "sem" : 6,
        "Hobbies" : "Chess"
}
{
        "_id" : 10,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Skating"
}
> db.Student.find({StudName:/[mM]/}).pretty();
{ "_id" : 1, "StudName" : "Kiran M K", "sem" : 6, "Hobbies" : "Podcasts" }
{
        "_id" : 9,
        "StudName" : "Trivikram hegde",
        "sem" : 6,
        "Hobbies" : "Chess"
}
{
        "_id" : 10,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Skating"
```



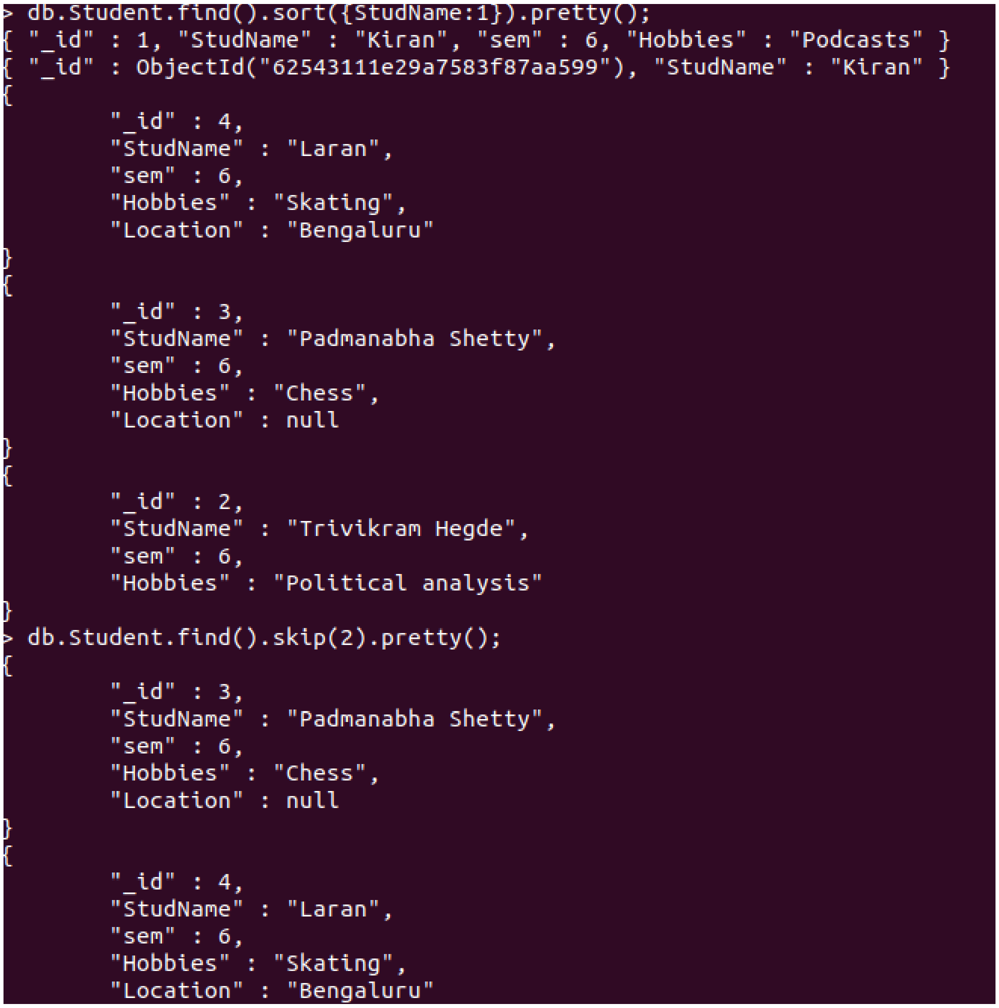
```
> db.Student.find({Hobbies:{$eq:'Chess'}}).pretty();
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess"
}
> db.Student.find({Hobbies :{ $in: ['Political analysis','Skating']}}).pretty ();
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
{ "_id" : 4, "StudName" : "Laran", "sem" : 6, "Hobbies" : "Skating" }
> db.Student.find({StudName:/^P/}).pretty();
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess"
}
> db.Student.find({StudName:/e/}).pretty();
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess"
}
```



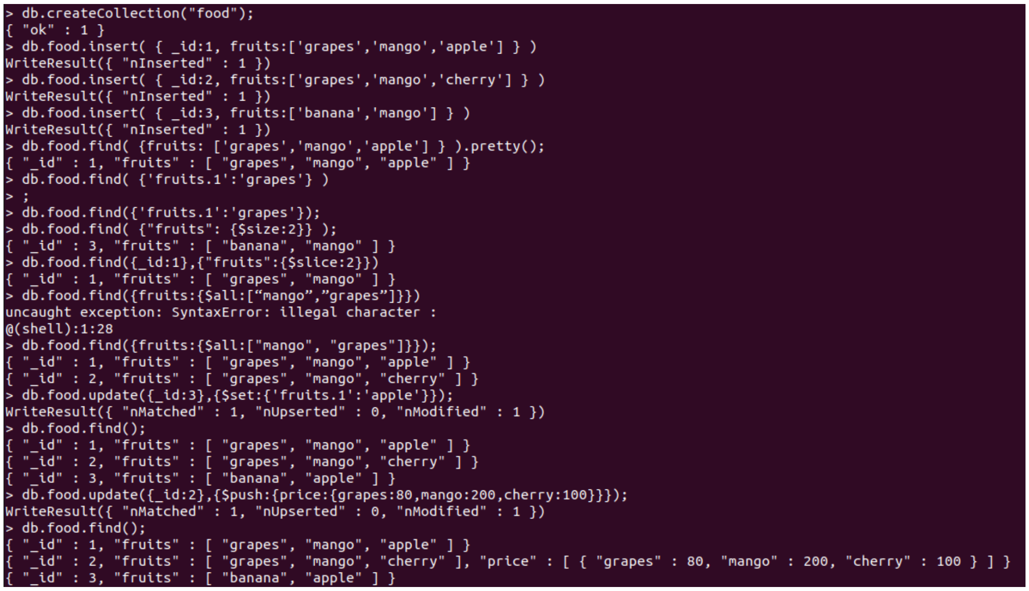
```
> db.Student.find({_id:1},{StudName:1,sem:1,_id:0});
{ "StudName" : "Kiran", "sem" : 6 }
> db.Student.find({Hobbies:{$ne:'Chess'}}).pretty();
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
{
        "_id" : 4,
        "StudName" : "Laran",
        "sem" : 6,
        "Hobbies" : "Skating",
        "Location" : "Bengaluru"
}
{ "_id" : ObjectId("62543111e29a7583f87aa599"), "StudName" : "Kiran" }
> db.Student.update({_id:3},{$set:{Location:null}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.count();
5
> db.Student.find({sem:6}).limit(3).pretty();
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess",
        "Location" : null
}
```



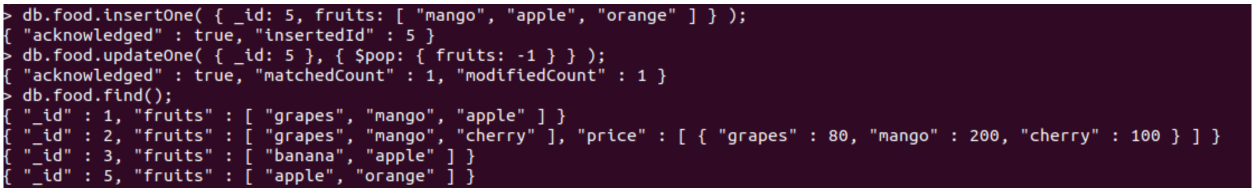
```
> db.Student.count();
4
> db.Student.find().sort({StudName:-1}).pretty();
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess"
}
{ "_id" : 4, "StudName" : "Laran", "sem" : 6, "Hobbies" : "Skating" }
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
> db.Student.save({StudName:"Kiran"})
WriteResult({ "nInserted" : 1 })
> db.Student.find();
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
{ "_id" : 2, "StudName" : "Trivikram Hegde", "sem" : 6, "Hobbies" : "Political analysis" }
{ "_id" : 3, "StudName" : "Padmanabha Shetty", "sem" : 6, "Hobbies" : "Chess" }
{ "_id" : 4, "StudName" : "Laran", "sem" : 6, "Hobbies" : "Skating" }
{ "_id" : ObjectId("62543111e29a7583f87aa599"), "StudName" : "Kiran" }
> db.Student.update({_id:4},{$set:{Location:"Bengaluru"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Student.find();
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
{ "_id" : 2, "StudName" : "Trivikram Hegde", "sem" : 6, "Hobbies" : "Political analysis" }
{ "_id" : 3, "StudName" : "Padmanabha Shetty", "sem" : 6, "Hobbies" : "Chess" }
{ "_id" : 4, "StudName" : "Laran", "sem" : 6, "Hobbies" : "Skating", "Location" : "Bengaluru" }
{ "_id" : ObjectId("62543111e29a7583f87aa599"), "StudName" : "Kiran" }
```



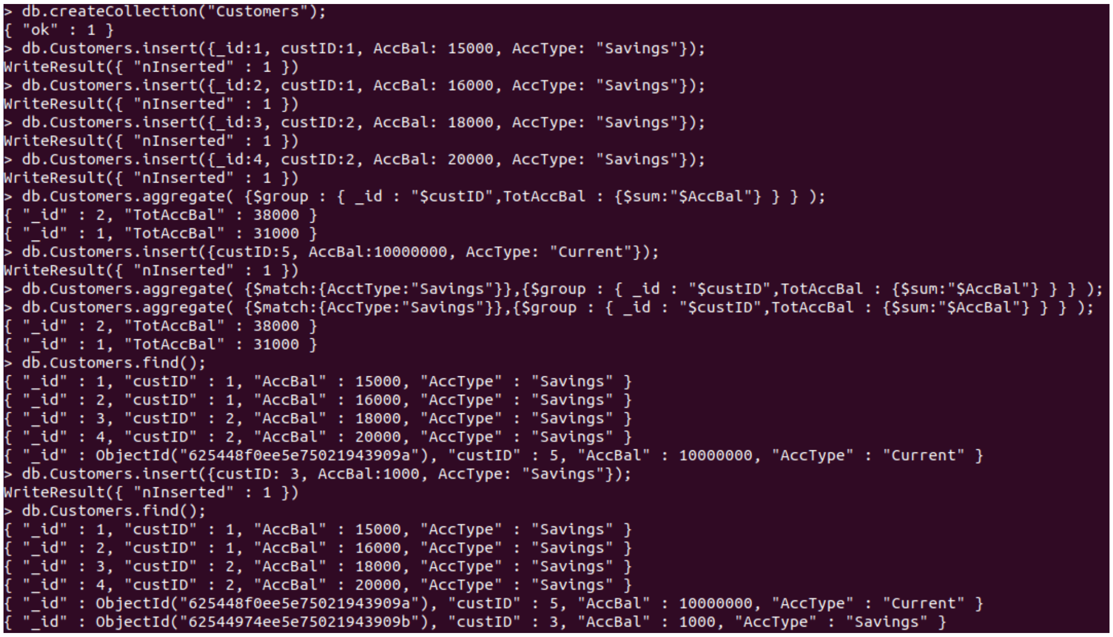
```
> db.Student.find().sort({StudName:1}).pretty();
{ "_id" : 1, "StudName" : "Kiran", "sem" : 6, "Hobbies" : "Podcasts" }
{ "_id" : ObjectId("62543111e29a7583f87aa599"), "StudName" : "Kiran" }
{
        "_id" : 4,
        "StudName" : "Laran",
        "sem" : 6,
        "Hobbies" : "Skating",
        "Location" : "Bengaluru"
}
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess",
        "Location" : null
}
{
        "_id" : 2,
        "StudName" : "Trivikram Hegde",
        "sem" : 6,
        "Hobbies" : "Political analysis"
}
> db.Student.find().skip(2).pretty();
{
        "_id" : 3,
        "StudName" : "Padmanabha Shetty",
        "sem" : 6,
        "Hobbies" : "Chess",
        "Location" : null
}
{
        "_id" : 4,
        "StudName" : "Laran",
        "sem" : 6,
        "Hobbies" : "Skating",
        "Location" : "Bengaluru"
```



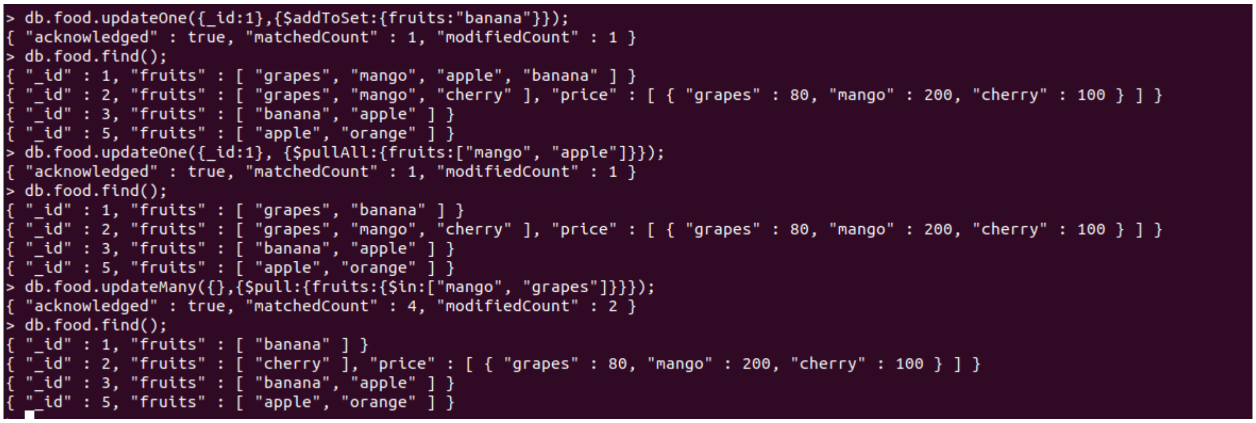
```
> db.createCollection("food");
{ "ok" : 1 }
> db.food.insert( { _id:1, fruits:['grapes','mango','apple'] } )
WriteResult({ "nInserted" : 1 })
> db.food.insert( { _id:2, fruits:['grapes','mango','cherry'] } )
WriteResult({ "nInserted" : 1 })
> db.food.insert( { _id:3, fruits:['banana','mango'] } )
WriteResult({ "nInserted" : 1 })
> db.food.find( {fruits: ['grapes','mango','apple'] } ).pretty();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
> db.food.find( {'fruits.1':'grapes'} )
> ;
> db.food.find({'fruits.1':'grapes'});
> db.food.find( {"fruits": {$size:2}} );
{ "_id" : 3, "fruits" : [ "banana", "mango" ] }
> db.food.find({_id:1},{"fruits":{$slice:2}})
{ "_id" : 1, "fruits" : [ "grapes", "mango" ] }
> db.food.find({fruits:{$all:[“mango”,”grapes”]}})
uncaught exception: SyntaxError: illegal character :
@(shell):1:28
> db.food.find({fruits:{$all:["mango", "grapes"]}});
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ] }
> db.food.update({_id:3},{$set:{'fruits.1':'apple'}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.food.find();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
> db.food.update({_id:2},{$push:{price:{grapes:80,mango:200,cherry:100}}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.food.find();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ], "price" : [ { "grapes" : 80, "mango" : 200, "cherry" : 100 } ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
```



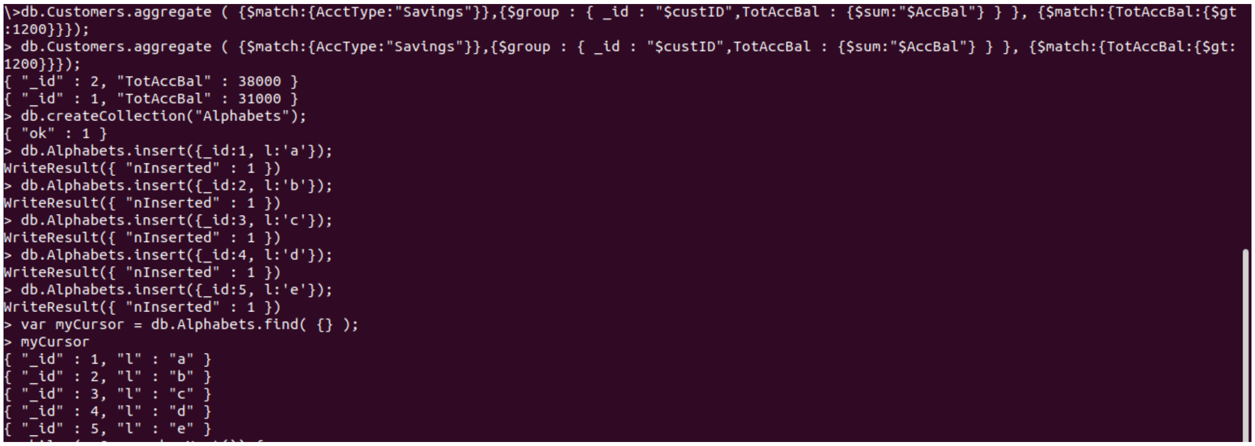
```
> db.food.insertOne( { _id: 5, fruits: [ "mango", "apple", "orange" ] } );
{ "acknowledged" : true, "insertedId" : 5 }
> db.food.updateOne( { _id: 5 }, { $pop: { fruits: -1 } } );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.food.find();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ], "price" : [ { "grapes" : 80, "mango" : 200, "cherry" : 100 } ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
{ "_id" : 5, "fruits" : [ "apple", "orange" ] }
```



```
> db.createCollection("Customers");
{ "ok" : 1 }
> db.Customers.insert({_id:1, custID:1, AccBal: 15000, AccType: "Savings"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({_id:2, custID:1, AccBal: 16000, AccType: "Savings"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({_id:3, custID:2, AccBal: 18000, AccType: "Savings"});
WriteResult({ "nInserted" : 1 })
> db.Customers.insert({_id:4, custID:2, AccBal: 20000, AccType: "Savings"});
WriteResult({ "nInserted" : 1 })
> db.Customers.aggregate( {$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } } );
{ "_id" : 2, "TotAccBal" : 38000 }
{ "_id" : 1, "TotAccBal" : 31000 }
> db.Customers.insert({custID:5, AccBal:10000000, AccType: "Current"});
WriteResult({ "nInserted" : 1 })
> db.Customers.aggregate( {$match:{AcctType:"Savings"}},{$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } } );
> db.Customers.aggregate( {$match:{AccType:"Savings"}},{$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } } );
{ "_id" : 2, "TotAccBal" : 38000 }
{ "_id" : 1, "TotAccBal" : 31000 }
> db.Customers.find();
{ "_id" : 1, "custID" : 1, "AccBal" : 15000, "AccType" : "Savings" }
{ "_id" : 2, "custID" : 1, "AccBal" : 16000, "AccType" : "Savings" }
{ "_id" : 3, "custID" : 2, "AccBal" : 18000, "AccType" : "Savings" }
{ "_id" : 4, "custID" : 2, "AccBal" : 20000, "AccType" : "Savings" }
{ "_id" : ObjectId("625448f0ee5e75021943909a"), "custID" : 5, "AccBal" : 10000000, "AccType" : "Current" }
> db.Customers.insert({custID: 3, AccBal:1000, AccType: "Savings"});
WriteResult({ "nInserted" : 1 })
> db.Customers.find();
{ "_id" : 1, "custID" : 1, "AccBal" : 15000, "AccType" : "Savings" }
{ "_id" : 2, "custID" : 1, "AccBal" : 16000, "AccType" : "Savings" }
{ "_id" : 3, "custID" : 2, "AccBal" : 18000, "AccType" : "Savings" }
{ "_id" : 4, "custID" : 2, "AccBal" : 20000, "AccType" : "Savings" }
{ "_id" : ObjectId("625448f0ee5e75021943909a"), "custID" : 5, "AccBal" : 10000000, "AccType" : "Current" }
{ "_id" : ObjectId("62544974ee5e75021943909b"), "custID" : 3, "AccBal" : 1000, "AccType" : "Savings" }
```



```
> db.food.updateOne({_id:1},{$addToSet:{fruits:"banana"}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.food.find();
{ "_id" : 1, "fruits" : [ "grapes", "mango", "apple", "banana" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ], "price" : [ { "grapes" : 80, "mango" : 200, "cherry" : 100 } ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
{ "_id" : 5, "fruits" : [ "apple", "orange" ] }
> db.food.updateOne({_id:1}, {$pullAll:{fruits:["mango", "apple"]}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.food.find();
{ "_id" : 1, "fruits" : [ "grapes", "banana" ] }
{ "_id" : 2, "fruits" : [ "grapes", "mango", "cherry" ], "price" : [ { "grapes" : 80, "mango" : 200, "cherry" : 100 } ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
{ "_id" : 5, "fruits" : [ "apple", "orange" ] }
> db.food.updateMany({},{$pull:{fruits:{$in:["mango", "grapes"]}}});
{ "acknowledged" : true, "matchedCount" : 4, "modifiedCount" : 2 }
> db.food.find();
{ "_id" : 1, "fruits" : [ "banana" ] }
{ "_id" : 2, "fruits" : [ "cherry" ], "price" : [ { "grapes" : 80, "mango" : 200, "cherry" : 100 } ] }
{ "_id" : 3, "fruits" : [ "banana", "apple" ] }
{ "_id" : 5, "fruits" : [ "apple", "orange" ] }
```



```
\>db.Customers.aggregate ( {$match:{AcctType:"Savings"}},{$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } }, {$match:{TotAccBal:{$gt
:1200}}}});
> db.Customers.aggregate ( {$match:{AccType:"Savings"}},{$group : { _id : "$custID",TotAccBal : {$sum:"$AccBal"} } }, {$match:{TotAccBal:{$gt:
1200}}}});
{ "_id" : 2, "TotAccBal" : 38000 }
{ "_id" : 1, "TotAccBal" : 31000 }
> db.createCollection("Alphabets");
{ "ok" : 1 }
> db.Alphabets.insert({_id:1, l:'a'});
WriteResult({ "nInserted" : 1 })
> db.Alphabets.insert({_id:2, l:'b'});
WriteResult({ "nInserted" : 1 })
> db.Alphabets.insert({_id:3, l:'c'});
WriteResult({ "nInserted" : 1 })
> db.Alphabets.insert({_id:4, l:'d'});
WriteResult({ "nInserted" : 1 })
> db.Alphabets.insert({_id:5, l:'e'});
WriteResult({ "nInserted" : 1 })
> var myCursor = db.Alphabets.find( {} );
> myCursor
{ "_id" : 1, "l" : "a" }
{ "_id" : 2, "l" : "b" }
{ "_id" : 3, "l" : "c" }
{ "_id" : 4, "l" : "d" }
{ "_id" : 5, "l" : "e" }
```